Table of Contents

Quick Reference

Common *DrivePro* Features

DrivePro has many features. This section describes how to perform some commonly used functions. It is recommended that you read the rest of the manual as soon as possible to better understand what is actually being done with each function.

Run **DOS6INST**.**EXE** found on your *DrivePro* diskette.

- Preferably make the *DrivePro* diskette bootable, and boot from it.
- Insert the *DrivePro* disk, go to that drive, and type: **DRIVEPRO**

You must be at the prompt for that drive and enter *DrivePro* - you cannot be at another drive and specify the *DrivePro* drive. If loaded, some software will interfere with the initial *DrivePro* registration.

- Physically install the hard drive(s) (see pg), ensuring that the jumper settings are correct.
- Run: **DRIVEPRO /X /IDE** and answer any questions.
- Your IDE drive will be installed and bootable in about a minute.

If you are experiencing difficulties getting the IDE Quick Installation to work, try running the following sequence of parameters. Note that this combination will write over any data on drive **0**. **DRIVEPRO /X /IDE /OVER0 /CUSTOM /PSK /B**

All switches are described on page and by running **DRIVEPRO /?** You may need to use switches, so be sure to read about all of them at least once.

The Guided Mode will lead you through the steps necessary to install a new hard drive. You may use this procedure on IDE drives if you would like greater control over the installation (such as partition sizes), but the previous procedure is quicker and easier for IDE drives.

- Install the hard drive(s) (see pg) ensuring that the jumper settings are correct.
- Run: **DRIVEPRO /X /G**
- You will be guided through the installation and prompted for input along the way.

This procedure is for use on hard drives that were previously set-up and working but have lost their configuration by, for example, moving the drive to a different system, or the CMOS battery going dead.

- Run **DRIVEPRO**
- Choose **GET LOST PARAMETERS** from the functions menu.
- The heads, cylinders, and sectors for the drive will be displayed in the results window. These are the parameters with which the drive was last partitioned. Note that merely retrieving this information does not change the system. You must also either use **BIOS Drive Table** to set to the correct CMOS Drive Type, or if no correct BIOS type exists or is definable (which would only happen if you moved the drive to a different machine), you must create a *DrivePro* Custom Drive Type. The easiest way to create a Custom Drive Type for this situation is to run **DRIVEPRO /MBR**.

For DOS versions 3.31 through 5, run **DRIVEPRO /MBR** and select the 'Other DOS Regular MBR'. You must have a copy of FDISK on a diskette. With MS-DOS 6.0 or above, you must run **FDISK /MBR**. With 5.0, either way may be used. **DRIVEPRO /MBR** will not work with the DOS6.0 FDISK unless you first unPKLITE the **FDISK.EXE** These procedures overwrite the Master Boot Code without affecting the partition table or format. You must reboot for the changes to take effect. Note that the **DRIVEPRO /MBR** feature can also be used to install a *DrivePro* IDE Antiviral MBR, or a Custom Drive Type for Plug-And-Play capabilities without affecting any partitioning or data on the drive.

Common Error Messages

Following is a description and fix suggestions for several error messages experienced when using *DrivePro*.

OS ERROR - To fix this error, run MHDRIVE ensuring that the correct parameters are used to initialize the drive. After DOS is reloaded, SYS the hard drive.

LOAD ERR - The Master Boot Code attempts to read the DOS Boot Record in the partition through standard INT13 calls. The BIOS is unable to perform this read. This may be caused by sector error, drive electronics failure, or any other problem that can cause that BIOS error. Often this is the result of complicated partitioning and can be resolved by repartitioning in a more simple fashion and reformatting.

NO ACTIVE - There is no active primary partition on the primary drive. To remedy this, you either need to create one, or more likely, make the existing one active. If there is already a primary partition on the primary drive that merely is not bootable, use FDISK or *DrivePro* to make it bootable with DRIVEPRO /PARTACTIVE

1782, 1790, FIXED DISK ERROR, HD CONTROLLER ERROR - These are errors reported by the BIOS at boot time. They indicate that either the machine was unable to successfully communicate with the drive, or that the drive failed its self diagnostics. You must resolve the hardware error before attempting to use DrivePro or any other software on the drive. These errors do not occur if the drive is not set in CMOS since the machine only tests a drive it is set to a type in CMOS.

HARDWARE ERROR - This message is reported by *DrivePro* for a drive in the BIOS Data window. Usually it is reported if there was a hardware error at boot-up. Since the drive did not pass its diagnostics, niether *DrivePro* nor any other software will be able to successfully work with the drive until this hardware error is resolved. Check jumpers, power, correct cabling, and the interface card. Try individually swapping the drive, the cable, the interface card, and the machine on which you are installing the drive. If you swap some hardware and are then able to access the drive, you know that whatever piece you just swapped was the problem.

NO ROM BASIC - There is no active primary partition and no boot floppy so the machine attempts to load BASIC which originally was stored on ROM in the machine. Many newer machines do not have this ROM and so the machine will report this error. This error can be avoided by booting from a floppy or making a partition on the drive active.

Introduction

DrivePro contains all the utilities needed to set up, diagnose, and recover hard drives and hard drive controllers on 80X86 type personal computers.

Many of the tools in *DrivePro* directly access and manipulate the hard drive and controller. Because the possibility for loss of data does exist, we recommend that all data be backed-up from *any* drive in the machine before using the program. However, so that you are not afraid of using the software, we recommend you become familiar with *DrivePro* by using to setup and work on a test hard-drive that does not have critical data on it. For any operation that may result in loss of data, *DrivePro* posts a warning and will require confirmation.

DrivePro is continually being added to and improved. It is likely that there have been additions or changes since the printing of this manual. It is therefore very strongly recommended that you first read the README.TXT file included on the diskette which will contain the most current information.

System Requirements

DrivePro can only be run and installed on a system that adheres to the following :

- The CPU must be an Intel 80X86 compatible. PC/XT and MCA-interface computers are not supported. Toshiba laptops are designed to limit the drives supported to a select few, and thus do not benefit from the *DrivePro* Custom Drive Type.
- The only known software conflicts are those that require their own MBRs, currently this only means Windows NT. *DrivePro* MBRs are supported by Novell 3.11 and greater, and by OS/2.

The following requirements apply only while actually running the software. Once you are through setting-up the system, it may be returned to its optimal setup and run as usual.

- No other software should be resident in memory during registration or use of *DrivePro.* Specifically, it is known that SHARE will interfere with the registration of *DrivePro.* To avoid conflicts and ensure that there is sufficient memory, you should make bootable (pg.) and boot from the *DrivePro* diskette.
- *DrivePro* has been tested on and is compatible with the following operating systems: MS DOS 3.3 and above, IBM PC DOS 3.3 and above, Compaq DOS 3.3 and above, DR DOS 6, Novell DOS 7, and Tandy DOS 3.3. Other operating systems may not be supported.
- We recommend against running under Microsoft *Windows*, Quarterdeck's *Deskview*, or any other multitasking software, since *DrivePro* must regularly access the hardware directly.
- You must disable any disk-caching hardware or software including SMARTDRV.SYS and caching controllers. Disk caching interferes with partitioning, low-level and high-level formatting by caching data that should be directly coming from or going to the drive.

About This Manual

This manual contains not only the information necessary to successfully use the software, but also some in-depth information concerning the workings of hard drives and controllers. You will find this additional information helpful in diagnosing hard drive related problems.

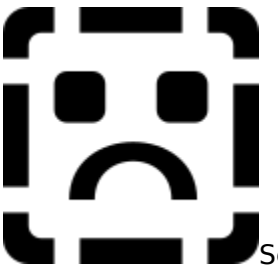
Two symbols are used to quickly identify important sections:

 Identifies an important point or warning.

 Identifies technical information. Although to run the software it is not necessary to read these sections**,** we recommend that you read them in order to better understand hard drives and to aid in troubleshooting.

Additionally, the ↵ symbol instructs you to press the **Enter** or **Return** key.

This manual is arranged into several sections. The first chapter explains in detail *DrivePro's* main menu items (the "Functions" window). If you are unfamiliar with hard drive installation procedures, you should first read **Physical Drive Installation** on page

 See the **README.TXT** file on the diskette for updated information not contained in this manual.

Using the Software

First Time Use

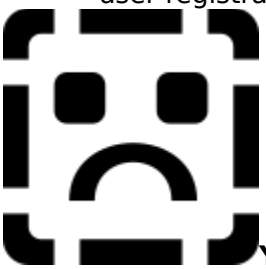
North-American Versions

After agreeing to the licensing agreement, first write-protect your original *DrivePro* diskette, then perform a DISKCOPY of it and place the original in safe storage, using it only to make a replacement working copy. This way, you will always have the intact original of *DrivePro* from which to make a working copy. If your **original** diskette does become corrupted or infected, there *will* be a replacement charge.

It is best if the *DrivePro* floppy is bootable, since diskette swaps are required otherwise. For this reason, the first thing you should do before running *DrivePro* is to make the diskette bootable. Do this with either the SYS program included with DOS or with MHSYS from Micro House included on the *DrivePro* diskette. Example: type "SYS A: B:" or "MHSYS A: B:" to copy the system files from a bootable diskette in drive A: to the *DrivePro* floppy in drive B: (Adapt the parameters to your system's particular configuration.) It is also a good idea, if possible, to place a copy of DOS's FDISK on the *DrivePro* diskette for situations in which you may want to install a DOS Master Boot Record. If you will be installing different DOS versions, you can make a bootable *DrivePro* diskette for each version.

DrivePro must be accessible on a floppy drive to work. You may run a copy from a hard drive but the floppy must still be accessible for verification purposes. Also, special 'save' files will be written to the diskette, and the disk must be written to in order to be registered, so in order to be able to use all features the diskette cannot be write-protected. (Some people write-protect their diskette to prevent virus infection, but the best defense against infection is write-protected, clean originals.)

When running the software for the first time you will be prompted for your personal name, company name and serial number. Your serial number is found on the *DrivePro* diskette (the 5¼ if you have both). If you encounter an error while attempting to perform this step on your diskette, you should **reboot from the *DrivePro* diskette** (see pg. 9) and attempt the procedure again. Booting from the *DrivePro* floppy prevents the loading of any software that may conflict with *DrivePro*'s user registration procedure.



You will only be prompted for this user information once, so be sure to enter it correctly!

Once this is done and the Main Menu appears, you may then use or exit the program.

Non North-American Versions

Non North-American versions of *DrivePro* end in 'C' (such as *DrivePro* 1.16C). These versions cannot be used except from the original disk. They are already serialized so do not require that step. You should still make the diskettes bootable, however, by using DOS's SYS or Micro House's MHSYS (see pg. 9). If the original diskette becomes corrupted or infected you need to contact your distributor concerning a replacement diskette.

Before using *DrivePro*, the drive(s) need to be physically installed into the system. For assistance with drive installation, refer to page , **Physical Drive Installation**.

To run *DrivePro* switch to the A: or B: floppy drive containing it and type:

DRIVEPRO <Return or Enter>

Standard Switches

You may use "switches" that alter the manner in which *DrivePro* runs. Each switch must have its own "/" which must be preceded by a space. An example is "DRIVEPRO /S /IDE". A thorough understanding of all the parameters is not necessary to use the program, but some may be required or useful under certain conditions.

/? Display all command line options including any new ones added since this printing.

/X Speed up exploding windows.

/IDE Execute IDE Quick Installation.

This is the 1 minute IDE installation feature. Just physically install the IDE drive, run *DrivePro* using this switch, and the IDE drive will be ready for use in about 1 minute (*DrivePro* will find and set the correct CMOS drive type, partition, DOS format, transfer the operating system, and reboot). The following switches are used ***only*** with the /IDE switch:

/CUSTOM Use a Custom Drive Type.

This creates a software custom drive type, allowing use of the drive's complete capacity, as well as "plug-and-play" capability.

This feature is typically used if: 1) your BIOS does not support your drive and does not have a 'user-definable' option or 2) plug-and-play capabilities are desired. Please see page 54 for complete details on this feature.

/OVER0 Reinstall drive, overwriting any existing information on hard drive 0 (the "master" or "C" drive). If the IDE hard drive has information on it and this switch is not used then the drive will not be set up by *DrivePro*. Remember to backup any information that might be needed later!

/OVER1 Reinstall physical drive 1, (the "slave" or "D" drive), overwrite any existing information. See /OVER0 above.

/IDE (cont.)

/OVERBOTH Reinstall (overwriting any existing information) both physical drive 0 ("master" or "C") and physical drive 1 ("slave" or "D"). See /OVER0 above.

/NOCMOS Do not set the CMOS drive type.

Use this switch if *DrivePro* cannot find the system BIOS or if you have already set or will be setting the CMOS manually. Not often used.

/VERIFY Perform surface scan during high-level format to help ensure data integrity.

This feature is rarely needed for IDE drives since most IDE drives have internal electronics that automatically manage defects, and even those that do not are low-level formatted at the factory and should arrive to you with no defects. Using this feature will greatly increase the time taken by *DrivePro* to setup your hard drive. It is only needed when setting up a used or older IDE drive, or one that is in questionable condition.

/PSK Enable safe POST SeeK.

Use this switch when using the /CUSTOM switch on Compaq and other "finicky"

systems. Such systems do a seek to the last cylinder of the drive during the POST (Power-On Self Test) which fails if the CMOS is set to a type larger than the drive resulting in a C: DRIVE FAILURE / F1 TO CONTINUE (Though in most cases pressing F1 causes the drive to come up fine). *DrivePro* will normally choose the CMOS drive type closest to the capacity of the drive, even if this is larger than the drive's capacity. On most systems this works fine, because when using the /CUSTOM switch, the *DrivePro* Custom Drive Type overrides the CMOS drive type setting. /PSK instructs *DrivePro* to only set the CMOS type to one which is less than the capacity of the drive, which is needed for Compaq and other finicky systems.

/G Enter Guided Installation mode.

Guides through the steps necessary for a new hard drive installation, asking for responses whenever a choice is encountered. You should use this option if you are installing a non-IDE drive. IDE drives may be installed with this option, but using /IDE is quicker (see pg 47).

/B Do not search for the BIOS drive table.

This switch is necessary for systems which do not have a standard AT BIOS drive table. It is important to note that using this switch disables *DrivePro's* BIOS drive table features. If your system has a BIOS drive table but *DrivePro* cannot find it, please call Technical Support.

/S Do not prompt for saving the results log on exit. The results log will not be saved unless the user specifically chooses the menu item for saving it.

/DIAG Display the diagnostics screen.Use of this switch will display the diagnostics screen also available through Enhanced Menu / Diagnostics / Show Report. The diagnostics screen displays the BIOS Data, Drive Inquiry, DBR, and MBR/Partition Table all on one screen! This is useful in troubleshooting a system that will not boot.

/M Use monochrome mode. For monochrome monitors or LCD.

/C Use color mode.

This is rarely necessary since *DrivePro* will automatically detect most mono or color video adapters.

/?A Display Advanced command line switch options.

Advanced Switches

These switches are for the advanced *DrivePro* user.

/MBR Install or replace an MBR without affecting the partition table.

DrivePro asks what type of Master Boot Record to install and installs the MBR without affecting the partition table or DOS system areas and so does not affect data on the drive.

This would be used to: 1) install an Antiviral MBR onto an existing drive, 2) to replace a *DrivePro* MBR with a Regular DOS MBR (same as using FDISK /MBR), 3) to install a *DrivePro* Custom Drive Type for taking advantage of Plug-And-Play, or 4) to restore a *DrivePro* Custom Drive Type that was accidentally erased. Note that the Custom Drive Type will be based on the drive's partition table, not the current CMOS settings, so you don't need to worry about having the system configured correctly before using.

/A<:filename> Enter Automatic mode.

If a file name has been specified, it is executed. If it does not, it is begun to be recorded. Also see page 47.

/F Fast load.

Does not load the drive parameter and other databases. This allows *DrivePro* to run in less memory and load slightly faster.

/U Get the drive parameters by BIOS interrupt call.

By default, *DrivePro* retrieves the current drive parameters from the hard drive parameter table, unless the parameters are found to be invalid, in which case, *DrivePro* will normally try an INT13 call. If the drive parameters shown in the "BIOS DATA"
window are not correct or there is a "Parameter Error" then you may try this command line switch. MFM/RLL (ST506/412) and IDE drives normally have their parameters stored in the hard drive parameter tables. SCSI drives normally require an INT13 call, and ESDI drives may function either way, depending on the controller card.

/K Do not check for TSR/Device Driver checking at start-up.

Not recommended, this prevents the warning messages.

/W Do not check for Microsoft Windows™ and Desqview™.

Not recommended, this switch allows *DrivePro* to run under Windows or DesqView.

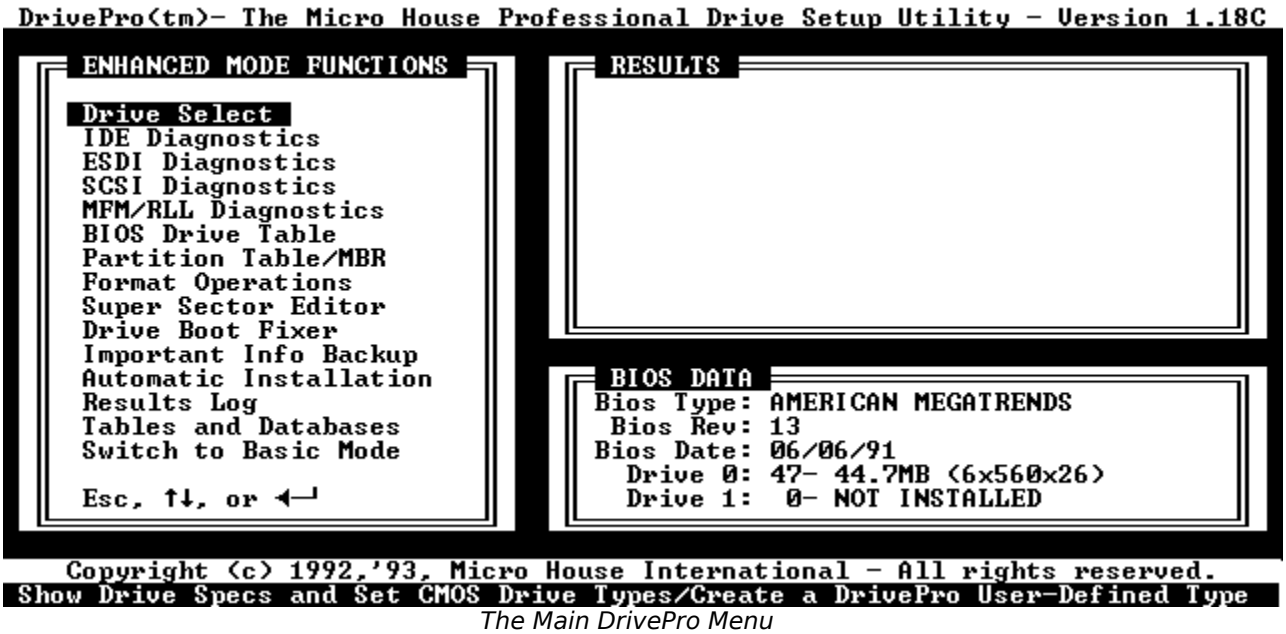
/E(segment:offset, limit) Specify BIOS drive table address and size limit. Rarely used.

/Y(segment:offset) Specify Reinitialize Drive/DOS Drive Parameter Table Address. Used in cases of memory conflict due to non-standard memory addressing of an expansion card or of the main board. Rarely used.

/Z(chksum,type) Specify DOS Boot Record checksum and type.

Generally used only when a Micro House technician is assisting you with an unsupported old or OEM DOS version.

The Main Selection Menu and Status Windows



The Main DrivePro Menu

There are three windows on this screen. The **FUNCTIONS** window contains the menu items, use the and arrows to move to an item, or simply enter the first letter of the item to go

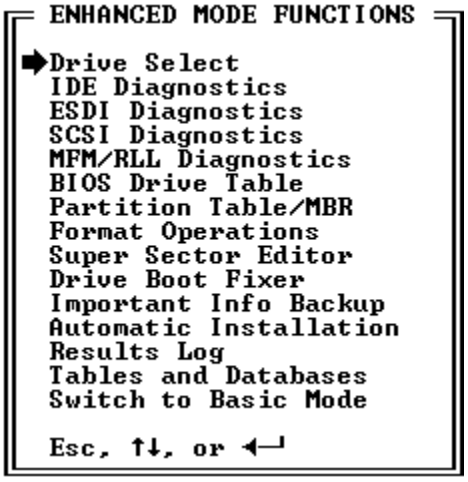
directly to it. (In the case of more than one item with the same first letter, *DrivePro* will toggle between the items.) Pressing selects the requested item. If you do not specify otherwise by switch, *DrivePro* will default to the Basic Mode. This mode has most of the necessary features available to install a drive. The Enhanced Mode allows more features, but is usually not necessary and not recommended until you are familiar with the features available from the Basic Mode.

The **RESULTS** window is where the results of an operation are placed. Press the **TAB** key to move to the results window and scroll up and down with the and arrow keys. This is helpful if the results from previous operations have scrolled off the screen. See page 50 for more information on this feature.

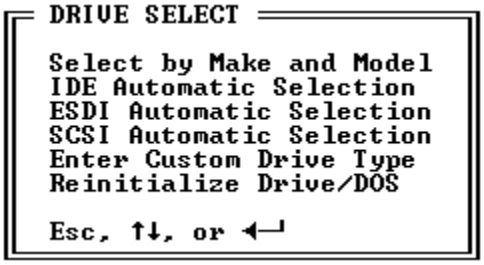
The **BIOS DATA** window contains information on the system BIOS and how the system is currently recognizing any drives. Note that this window will reflect a parameter override if a *DrivePro* Custom Drive Type is installed.

Menu Items

The Functions Menu
DRIVE SELECT



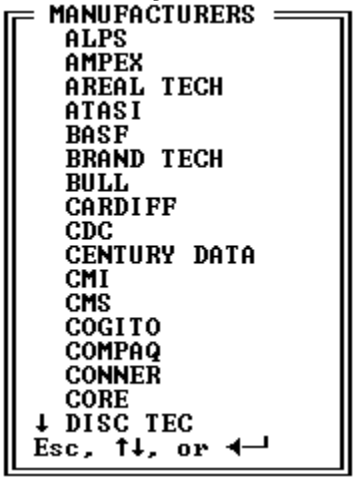
Drive Select



The Drive Select Menu

This menu selection leads to more specific options used to determine the appropriate drive parameters, and to set the system's CMOS correctly (drive type and parameters).

Select by Make and Model

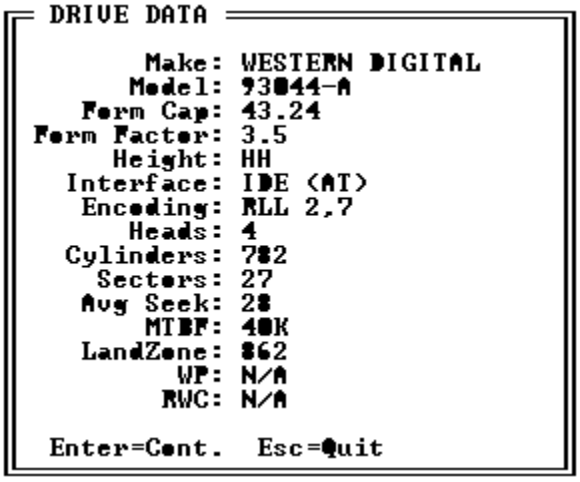


The Manufacturer List

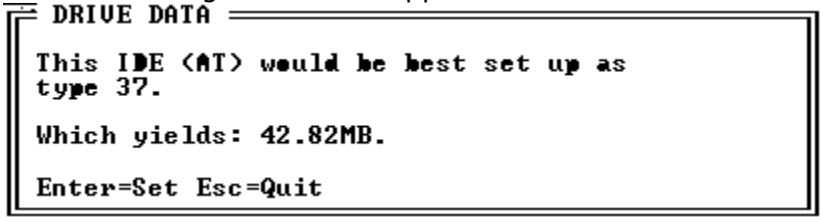
Choose this menu item to view the hard drive specifications and to setup the CMOS for the correct drive type and parameters. Specifications are included for nearly 2,000 hard drives. See the Glossary on page **Error! Bookmark not defined.** for a complete explanation of each of these parameters.

The list of drives, as well as most of the databases in *DrivePro,* have a quick search feature. Typing in the partial or full text of the item that you are locating will quickly jump to that section of the database. For example, typing "WES" will quickly take you to Western Digital.

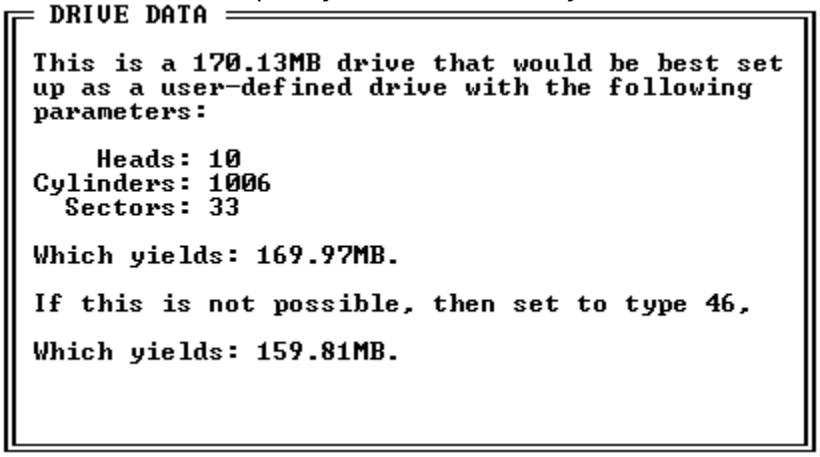
After selecting the correct hard drive you will be presented with the specifications for that drive:



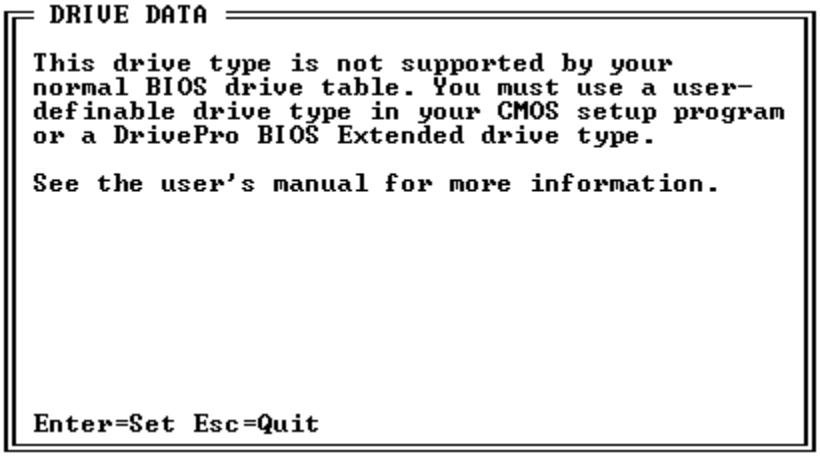
Press when done viewing the specifications to set the CMOS settings. If there is a type in your BIOS drive table that correctly matches the drive being installed then the following window will appear:



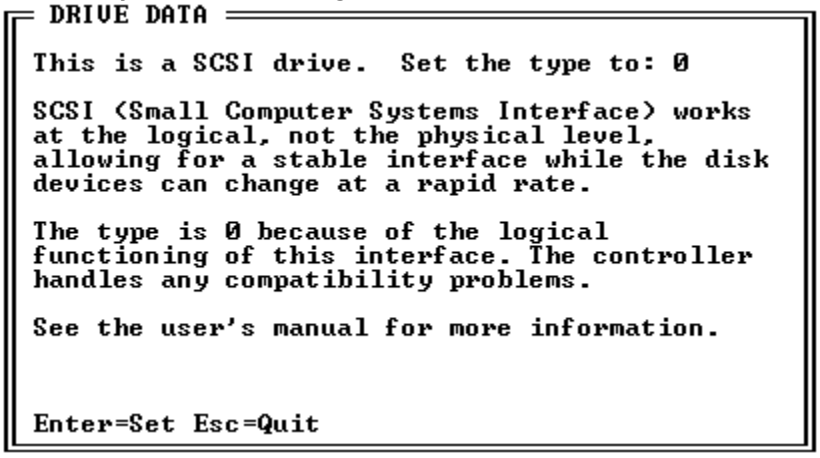
Your BIOS drive table may not have an entry that matches your drive and would lead to loss of capacity. In such a case you will see a screen that indicates this condition:



The type recommended depends on the drive being installed and the current system BIOS. A recommended type will not be shown if there is no type in the BIOS that even roughly matches the drive. The following window will appear:

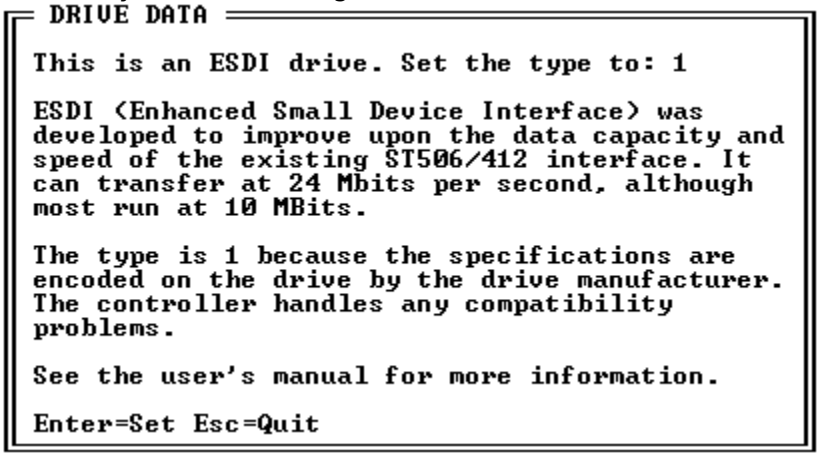


If you are installing a SCSI drive then this window will appear:



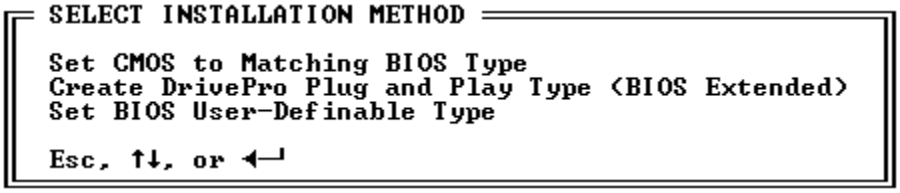
SCSI drives are not identified by your system as hard drives. They are controlled as separate devices by the SCSI controller. One of the advantages to this is that you can have two ESDI, IDE, or MFM/RLL hard drives in addition to one or many SCSI drives. Since the SCSI hard drives are not considered by your system to be hard drives, you do not indicate in the CMOS that they exist. The SCSI controller is responsible for dealing with the drive and interfacing with DOS.

If you are installing an ESDI drive then this window will appear:

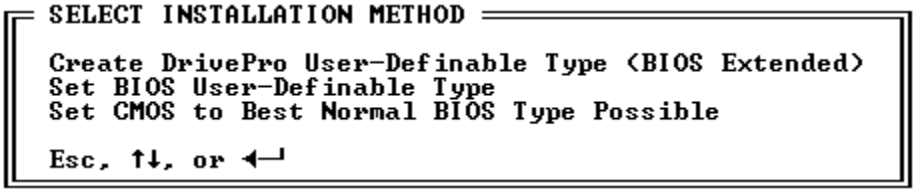


To use an ESDI drive in your system, set the CMOS to drive type 1. The ESDI controller card will replace the system's own hard drive parameters with its own. This enables the ESDI card to "override" the system's type 1 with its own parameters.

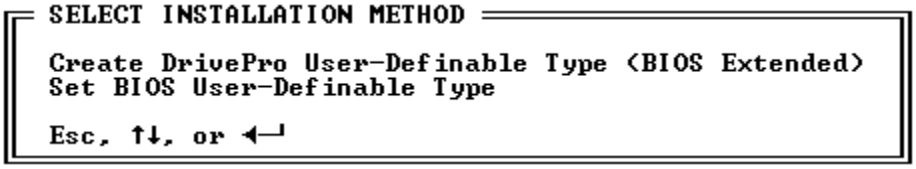
After viewing one of the above windows press to continue on to the next menu. One of three menus will then appear depending on your hard drive and BIOS (in the case of SCSI or ESDI no additional menus will be displayed, but instead you will be prompted to set the CMOS to type 0 or 1 respectively):



Menu 1: If the hard drive has an exact match in the BIOS table.



Menu 2: If the hard drive does not have an exact match in the BIOS table.



If your BIOS does not have a user-definable type in CMOS then this option will not appear on the above menus. This option will also not appear if *DrivePro* cannot find the user-definable type in your BIOS, even if it exists. If *DrivePro* does not support your user-definable CMOS type, you should contact Micro House Technical Support. An explanation of each of the options follow.

Set CMOS to Matching BIOS Type

This option will set the CMOS to the exact matching drive type in the BIOS drive table. If you have been given this option, then an exact match in the BIOS has been found. All of the drive's capacity will be available for use. You may still wish to install a Plug-and-Play drive type. See page 54 for details on Plug-and-Play capabilities.

Set CMOS to Best Normal BIOS Type Possible

This option will set the CMOS to the *closest* matching drive type in the BIOS drive table. If you are given this option then there is not a type that matches perfectly, and you will lose some disk capacity by using this option. The capacity of the drive that will be available when using the Best Normal BIOS Type was displayed in the previous window.

Create DrivePro User-Definable Type or

Create DrivePro Plug and Play Drive Type

These options are identical. A Custom Drive Type is called a Plug-and-Play type if it is being installed even though an exact match exists in the BIOS drive table, or the CMOS has a user-definable type.

After selecting the drive to configure (0 or 1), you will be asked to confirm the overwriting of the Master Boot Record (MBR). This must be done when creating a Custom Drive Type. The partition table will be preserved, but the Master Boot Program will be overwritten.

There are advantages and disadvantages to creating a *DrivePro* Custom Drive Type. We recommended that you have a complete understanding of the Custom Drive Type before installing one. See page 54 for details.

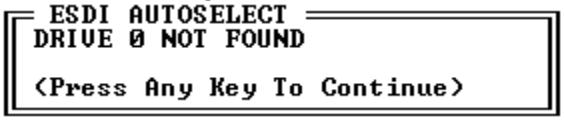
BIOS User-Definable Type

If your BIOS supports a user-definable drive type in its CMOS setup, and *DrivePro* can locate it, then you probably will want to set it with the proper parameters for the drive being installed. Remember to record physically on the drive's case the parameters that you use, to aid in future drive recovery. For example, if the system ever loses its setup, then you will need to enter in the *exact* original parameters to get the drive to function correctly.

You may additionally **Create DrivePro and User-Definable Drive Type** feature. The advantage is that the *DrivePro* Custom Drive Type is independent of the BIOS, so the drive will still operate correctly if the system loses its CMOS settings or if you move the drive to another system. You merely have to set the CMOS to any type other than 0 or "Not Installed." A further discussion of the advantages and disadvantages of the Custom Drive Type is found on page 54.

IDE, ESDI, and SCSI Automatic Selection

This feature will interrogate the drive for its parameters. If *DrivePro* cannot recognize your IDE or ESDI hard drive, then you will receive a message similar to the following:

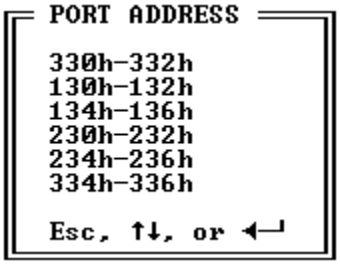


Try setting the drive type to 1, rebooting, and then running this feature again.

For the SCSI Automatic Selection to work *DrivePro* must support your SCSI controller. SCSI Automatic Selection requires you to select from a list of supported controllers.

As of this writing only three SCSI controllers are supported, but many others fall under these two categories. If your controller is not specifically listed, try selecting the ones that are.

You will then be asked for the controller's port address. The port address is usually set by jumpers on the controller.



The above addresses will vary for each different SCSI controller. *DrivePro* will only be able to access your controller if the correct port address is selected. If *DrivePro* fails to find the controller, then check your controller manual and jumper settings for the current port address settings.

Enter Custom Drive Type

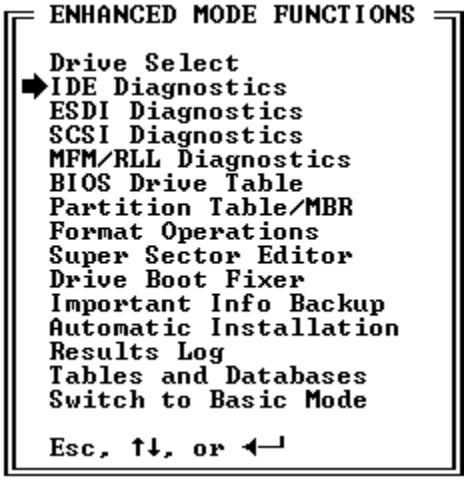
This feature is used to enter the parameters and install a *DrivePro* Custom Drive Type. This feature is used when you know the parameters you want to use and are having trouble installing a Custom Drive Type. Note that as a safety feature, the partition is marked inactive. Therefore, if you are using this to define a Custom MBR and want the drive to boot, you will need to next go to the Partition Table Editor and change the partition to BOOTABLE = 'YES'. If you *are* replacing an overwritten or corrupted MBR, an easier way to accomplish this is to run *DrivePro* with the /MBR switch (see pg. 10).

Reinitialize Drive/DOS

This option will inform DOS of the drive's parameters and reload DOS. An example of this is if you just installed a *DrivePro C*ustom Drive Type (also called "BIOS Extended") and do not wish to completely reboot. Using this option will leave you at the DOS prompt since it executes DOS initialization code. You can then run DrivePro or any other program and the drive parameters will be set correctly. This menu option is not commonly used, though the stand-alone program version of this function, MHDRIVE.EXE, is. See page 64 for info on MHDRIVE.

Drive Diagnostics

IDE, ESDI, SCSI AND
MFM/RLL DIAGNOSTICS

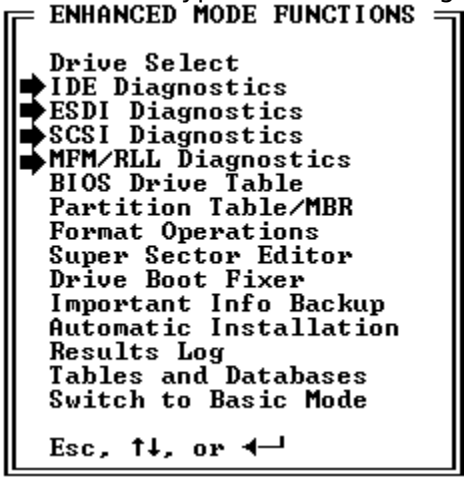


IDE, ESDI, SCSI and MFM/RLL Diagnostics

These features are only available through the Enhanced Mode Menu.

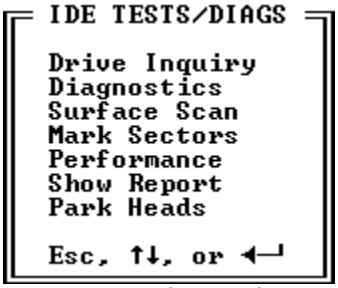
The manner in which your drive will be diagnosed depends on its interface. The interface is the means by which the hard drive, controller, and computer communicate.

There are currently four popular interfaces for the PC; ESDI, IDE, SCSI and MFM/RLL (actually ST506/412 but referred to as MFM/RLL by us so as to not cause confusion). Each type has its own diagnostic menu items.



Each interface type menu option is explained in full detail below. We highly recommend that you read the technical sections that follow pertaining to the interface type of the drive you will be working on. Having a full understanding of the technologies used will greatly increase your ability to troubleshoot hard drive and controller problems.

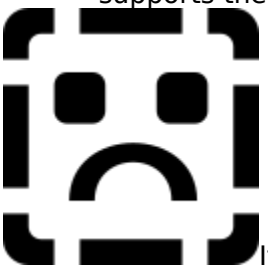
Diagnostics



IDE Diagnostics Menu (other diagnostic menus have similar selections)

Drive Inquiry

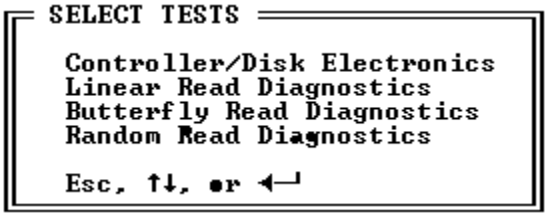
Selecting this option will interrogate the drive(s) and return its specifications. Pressing after the drive specifications are displayed will advance you through menu options used to set the CMOS and install the desired type of MBR. These options are explained fully in the **Drive Select** section which begins on page 14. MFM/RLL drives do not support **Drive Inquiry** and do not contain this menu item. SCSI **Drive Inquiry** will also allow you to retrieve information on other devices, such as tape backups, CD ROM drives, etc. connected to the SCSI controller if they support the SCSI **Device Inquiry** command, and will supply capacity data if the device being interrogated supports the SCSI **Read Capacity** command.



If *DrivePro* encounters problems identifying a SCSI controller or SCSI device then check the SCSI controller to ensure that the firmware (BIOS) and interrupts are **enabled**. Not all SCSI devices will support **Device Inquiry** and not all SCSI controllers are currently supported by *DrivePro*.

Diagnostics

This menu option leads to several diagnostic test selections. An option is toggled for selection/deselection by pressing while on an item. The tests toggled for execution will have a small filled-in box to the left of them. Selecting **ESC** exits the window and executes the tests selected.



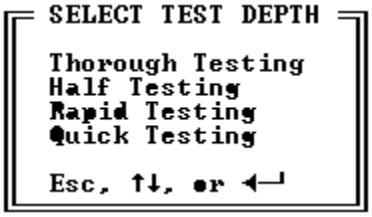
Diagnostics Menu

Controller/Disk Electronics

This option executes the controller and disk internal diagnostics. Since these tests are dependent on the controller and drive, some will not support all commands sent to them. If this is the case with your controller and drive then messages such as "Can't test controller RAM" (meaning that the controller does not support the controller RAM test) will appear in the **RESULTS** window. This is perfectly normal and does not mean that there is a problem with your controller or drive.

Linear, Butterfly and Random Read Tests

These test the drive's ability to read data from the drive. These are all **non-destructive** tests.



After selecting the test depth, the test will be run and the results placed into the **RESULTS** window.

Surface Scan

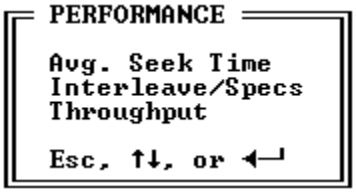
This feature gives you four levels of testing to perform on your drive media. Typically the second level will find any bad sectors, but you can run the longer and more thorough tests to be absolutely sure. Currently *DrivePro* will automatically mark bad any sectors that fail the tests, but at the lowest level. This means they must further be marked bad in the FAT through a utility program especially designed for this purpose, or a **Regular** (not Quick) high-level format. Be sure to check your README file as this feature is bound to change!

Mark Sectors

This feature is used to mark and unmark specific sectors as defective at the lowest level. See the above notes concerning Surface Scan.

Performance

This menu option has several performance test options.



Seek Time

Tests the drive for average, track-to-track and full-stroke seek times. The seek times are obtained as follows:

Average Seek: Seeks are performed within the highest and lowest cylinders for all possible seek lengths. The number of cylinders minus 1 seeks are performed. This is a more exact measure of average seek time than doing a 1/3-stroke seek as some other programs do.

Track-To-Track Seek: The time it takes the drive to seek between adjacent tracks. 512 seeks are performed.

Full-Stroke Seek: The time it takes the drive to seek from the lowest cylinder to the highest cylinder. A total of 256 seeks are performed.

These tests are usually accurate but certain variables such as system timing, BIOS type, and transfer rate might cause this figure to deviate from true slightly. *DrivePro*'s figures will usually not exactly match the manufacturer's due to the fact

that their tests usually do not take into account system and controller overhead, and are measured in an idealized way.

Average seek times are most accurate on non-translated drives. Average seek time for SCSI drives may not be a valid test because SCSI drives always perform translation. Average seek times on translated ESDI and IDE drives may deviate from true because the seek time between logical tracks may differ from physical tracks, but still can be useful in comparisons.

See also the section below concerning Throughput.

Interleave/Specs

Checks the drive for its current interleave and determines the optimum interleave. The optimum interleave given is only valid for the system that the drive is currently installed in. Interleave testing is usually only needed for MFM/RLL (ST506/412) drives.

This test involves the reformatting of a track on the drive (track 128). Due to this fact it will not work with most IDE drives, and many ESDI and SCSI drives, and would not be necessary anyway.

Most newer drives only use an interleave of 1:1 as they have a full track buffer. Interleave optimization is not required on these drives.

If you want to permanently adjust the interleave on a drive, redo the low-level format (this can be done non-destructively). See the section entitled "**Format Operations"** for further details.

Throughput

The most widely-published performance rating is **seek** time. Most software utilities will report the **seek** time of the drive. While this can be useful and is, in fact, one of the most critical aspects of a drive's performance, a more practical measure of drive performance is throughput. **Seek** time is a rating of the time involved when moving the drive heads across the platters. In addition to seek time, however, there also are additional factors such as **Latency**, which is the time involved in spinning the platter around so that the desired data is below the heads, **Head Switch Time**, which is the time involved in switching to a different head of the head assembly, and Controller, BIOS, and System Bus-to-Memory overhead. These factors all affect the speed at which desired information can be retrieved and made available to applications. By performing data transfers from the drive, all of these factors can be measured together as overall system throughput. Dramatic performance (throughput) increases can be realized by such innovations as track buffers, caching, and local bus controllers. Please note: Manufacturers of controller cards and drives often publish throughput ratings for their products, but these figures are based on ideal conditions, and for that device only, rather than the whole system, and so may differ from the actual overall system throughput achieved, and measured by *DrivePro*. Note also that performance can be affected by the type and manner of hard drive access of a particular application. An application which does more random access of information will perform more poorly than one that confines its accesses mainly to a particular area of the drive.

Show Report

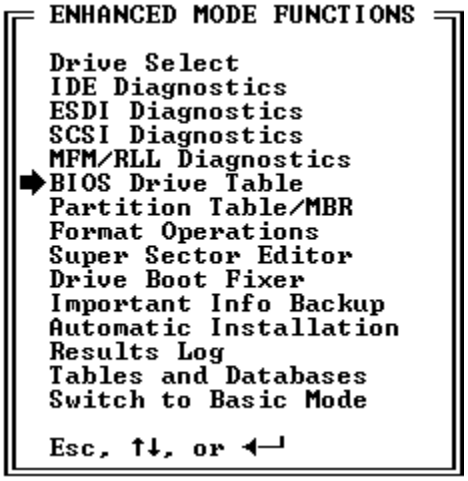
This feature displays the IDE/ESDI Drive Inquiry, BIOS Data, DOS Boot Record, and Partition Table/MBR windows all on one page. This is a great help in troubleshooting drive or controller problems.

Park Heads

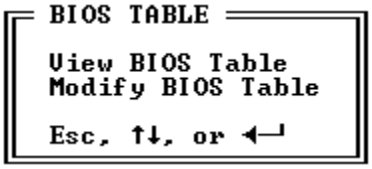
This feature has been moved here from the Enhanced Main Menu. It is used to park the hard drive heads at a safe position for transport or when the computer is idle or turned off. Note that most newer hard drives now park their heads automatically when turned off. Also, laptop computers often "spin down" their internal IDE drives when idle for a few minutes, which parks the heads. It is important to ensure the hard drive heads are parked when transporting a drive, and as a general safety measure when the drive is idle, to help prevent data loss. See also the section later in this manual on **Park Heads**.

Bios Drive Table

BIOS DRIVE TABLE



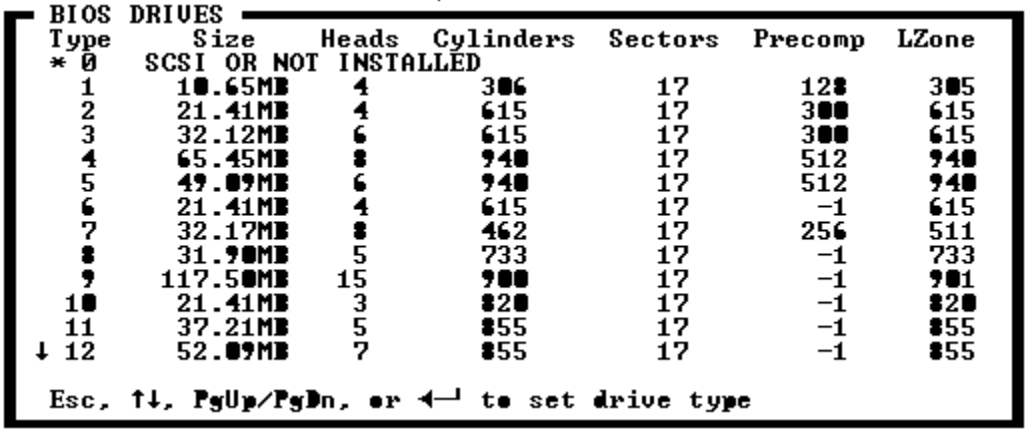
When in Basic Mode, you will directly be placed in the View BIOS Table window. When in Enhanced Mode, you will presented with the following options:



The BIOS Drive Table Menu

View BIOS Table

Choose this to view the systems BIOS table and optionally set the CMOS for a table entry. Use the and arrows to move to a drive type and to set the CMOS to that type.



The View BIOS Table Screen (your display may vary slightly)

The current setting(s) for drive 0 and 1 will have an asterisk (*) next to them. As portrayed in the diagram, this system has one or both of its drive types set to 0: SCSI or not installed.

If your BIOS supports a CMOS user-definable drive type you may set it by scrolling down to the user-definable type (usually type 46 or 47) and press . You will then be

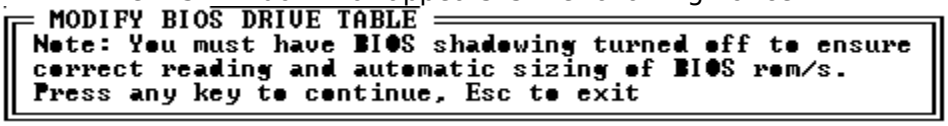
able to enter in the parameters manually, and set the CMOS to use that drive type. In most cases you should enter 0 for Precomp and LZone as they are not used by most newer (non MFM/RLL or ST506/412) hard drives. MFM/RLL (ST506/412) hard drives usually require Precomp and LZone, which can be obtained by selecting the drive's make and model from the **Drive Select** menus, as described in the corresponding section of this manual.

Modify BIOS Table

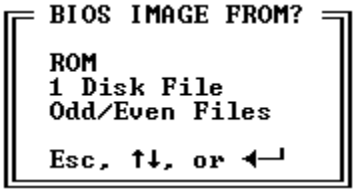
With this option you may create a CMOS entry that supports your particular drive, though to actually store this table into your BIOS table you will need a piece of hardware- an EPROM programmer (burner). All types other than 1, 2, 15 (reserved and not shown) and the last entry may be edited. Types 1 and 2 are reserved to maintain compatibility. Certain programs, including *DrivePro*, assume these two types to be set to the manufacturer's settings. Type 15 cannot be used because it is a chain to higher drive types (above 15) in the CMOS. The last entry may not be edited since it has no entry following it for checksum correction (see below).

Please note: **This procedure will work with most but not all BIOS types**. Most BIOS's do what is called a checksum. The checksum does a calculation on all of the bytes in the BIOS. *DrivePro* tries to compensate for the bytes that change by adding or subtracting from the bytes in the next entry in the drive table. This is not always effective. If the BIOS fails the checksum when booting then the system will halt and not boot-up. It can then be assumed that this feature will not work with your particular BIOS.

The first window that appears is the following notice:



Be sure to heed this notice and continue only if shadowing is disabled. Shadowing must be disabled through your system's setup program. Older systems do not have shadowing and so this is not a problem. You will then be presented with the source options:



ROM	Loads the table in from the system's BIOS.
1 Disk File	Loads the table in from a single disk file that was previously saved.
Odd/Even files	Loads the table in from two disk files (odd and even) that were previously saved.

After selecting the source you may then edit the table. Use the and arrows to move to a drive type and to edit that type. **TAB** and **SHIFT+TAB** move between the fields. Once the type or types are edited choose **F2** to save.

MODIFY BIOS DRIVE TABLE

BIOS segment is: F000
BIOS ROM detected is 64K.
Note: checksum is fixed by changing the next entry.

BIOS DRIVE TABLE EDIT

Type	Size	Cyls	Heads	Sectors	Precomp	ECC	Ctrl	LZone
1	10.65MB	306	4	17	128	0	0	305
2	21.41MB	615	4	17	300	0	0	615
3	32.12MB	615	6	17	300	0	0	615
4	65.45MB	940	8	17	512	0	0	940
5	49.09MB	940	6	17	512	0	0	940
6	21.41MB	615	4	17	-1	0	0	615
7	32.17MB	462	8	17	256	0	0	511
8	31.90MB	733	5	17	-1	0	0	733
9	117.50MB	900	15	17	-1	0	8	901
10	21.41MB	820	3	17	-1	0	0	820
11	37.21MB	855	5	17	-1	0	0	855

Esc, Tab/Shift-Tab, ↑↓, PgUp/PgDn, ←┘, or F2 to save

The BIOS Drive Table Edit screen (your display may vary slightly)

Note that when one entry is changed the entry below it will also change to compensate. If compensation is not possible because of the byte values, you will get a message from *DrivePro* and you should try modifying another entry for that drive type. When done, press F2 to save your changes. When saving you have two options: **1 Disk File** or **Odd/Even Files**. Choose **1 Disk File** if your BIOS is contained on one chip. Choose **Odd/Even Files** if your BIOS is contained on two chips.

The final option is the type of file to output. Choosing Binary Format is adequate for most EPROM programmers. A select few require Hexadecimal input. Refer to your EPROM programmer manual if you are unsure as to the type required.

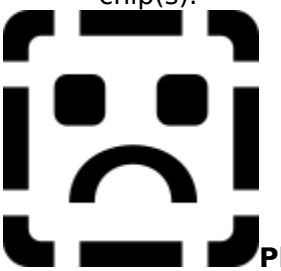
BIOS IMAGE FILE TYPE?

Binary Format
Hex Format

Esc, ↑↓, or ←┘

The BIOS Image File Type menu

You will then be prompted for the file name(s). After saving you will need to exit the program and run your EPROM programmer software to program (burn) the new chip(s).



Please note! Neither Micro House International nor any of its distributors can be held liable for the consequences of copying copyrighted BIOS's. Some BIOS manufacturers protect their software with a license agreement that prohibits the modification and duplication of their BIOS. We are not promoting the illegal modification and distribution of these BIOS's. We advise you to contact the manufacturer of your particular BIOS to receive, in writing, permission to modify the software.

BIOS Technical Information

The rest of this chapter explains the BIOS in detail. Reading this section is not necessary for the use of the software but is recommended.

Most 286 and 386 computers use two BIOS chips. Each chip only has a data path 8-bits wide but the CPU can handle a 16-bit data path. If the designers were to put the whole BIOS on one chip, access to it would be through a slow 8-bit path. By placing it on two 8-bit chips data can be transferred from each chip at the same time thereby creating a 16-bit BIOS data path. The access is altered for each byte of data. One chip contains all the odd numbered bytes and the other contains all the even numbered bytes. If you were to look at the BIOS area with debug (use the command DF000:0) you would probably see a copyright message such as: CCOOPPYYRRIIGGHHTT. This is because each of the two chips has a copyright message in it and when they are "meshed" together they form this strange duplicate message. More current system designs use only one BIOS chip. They get by the 8-bit data problem by utilizing "BIOS shadow RAM." This places a copy of the BIOS into fast RAM memory which is accessed through 16, 32, or even 64-bit data paths. In these types of systems ensure that the shadow feature is turned on, except when running *DrivePro*, or access to the BIOS will be slower, thus programs will run noticeably slower.

Most current systems that use two chips use 27256 type chips. The "256" is the number of kilobits that the chip can hold. If you divide this number by 8 you can get the number of bytes that the chip holds which is 32KB. So two of them together equal 64KB, the normal size for current BIOS's. Some older systems use two 27128 chips which is a total of 32KB (16KB x 2) and a few newer systems use two 27512 chips which is a total of 128KB (64KB x 2). The systems that use a single BIOS chip usually have one 27512 (64KB).

Partition Table/MBR

```
ENHANCED MODE FUNCTIONS

Drive Select
IDE Diagnostics
ESDI Diagnostics
SCSI Diagnostics
MFM/RLL Diagnostics
BIOS Drive Table
➡Partition Table/MBR
Format Operations
Super Sector Editor
Drive Boot Fixer
Important Info Backup
Automatic Installation
Results Log
Tables and Databases
Switch to Basic Mode

Esc, ↑↓, or ←┘
```

Partition Table/MBR

After the low-level format (if it was required) you will need to partition the drive. Partitioning sets up the basic drive configuration and other information that DOS and other operating systems will need to access the drive. This section of the disk is also referred to as the Master Boot Record (MBR).

```
SET UP PARTITIONS

Automatic Partitioning
Create Master Boot Rec.
Create A DOS Partition
Modify Logical Drives
Delete Last Partition
Edit Partition Table
Save Partition Table
Undo All Changes

Esc, ↑↓, or ←┘
```

Partition table/MBR Menu

The partition table editor status box will appear below this menu:

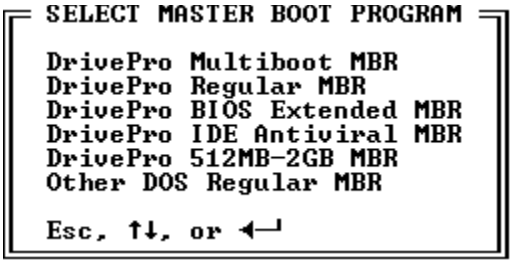
```
PARTITION TABLE EDITOR
BOOT  SYSTEM          FROM:              TO:          PRIOR    TOTAL     TOTAL
ABLE  TYPE         HD  CYL SECT   HD  CYL SECT  SECTORS  SECTORS    SPACE
NO    DOS HUGE      1    0   1     7  774  33        33   204567  104.7MB
NO    UNUSED        0    0   0     0    0   0         0        0    0.0MB
NO    UNUSED        0    0   0     0    0   0         0        0    0.0MB
NO    UNUSED        0    0   0     0    0   0         0        0    0.0MB
DRIVEPRO NEW BIOS EXTENDED MASTER BOOT RECORD    TAB/SHIFT-TAB, F1=HELP
```

Partition Table Editor status box

The available partition setup menu options are explained below.

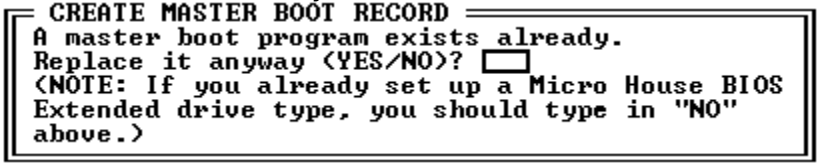
Automatic Partitioning

This is a shortcut that optionally creates a new MBR and partitions all of the drive into the largest partitions possible (1 DOS Huge partition in DOS 5.0) with the first (or only) partition marked bootable.



Selecting the *DrivePro* Multiboot MBR is usually the best choice (MBR types are explained on page 26).

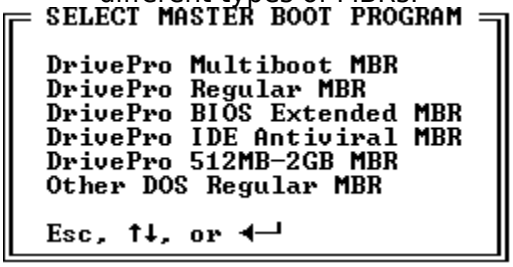
If there is already a MBR on the drive then the this window will appear:



If you have previously installed a *DrivePro* Custom Drive Type (through **Drive Select** or **Drive Inquiry**, for example) then enter *NO*. If you choose *YES* then your custom type will be partially overwritten! A typical installation may involve installing a Custom Drive Type, rebooting to enable the custom Drive Type, then selecting **Automatic Partitioning, N-O** as a shortcut for partitioning. After being created, the partition must be high-level formatted.

Create Master Boot Record

This feature will create the desired MBR and optionally wipe clear the partition table. The master boot record (MBR) is located on the first sector of the hard disk. It is loaded and executed after the BIOS has completed its other power-on or reset procedures. Every disk must have an MBR to be accessible. *DrivePro* offers several different types of MBRs:



DrivePro Multiboot MBR

If your hard disk has several active partitions marked bootable then this MBR will prompt you with "Boot (1x3x)?" (for a system with partitions 1 and 3 marked bootable) every time you start the system. The user responds with a keystroke selecting the number of the partition from which to boot (in this case '1' or '3'). If no selection is received within 15-30 seconds, the Multiboot Program will automatically boot the partition marked as the DEFAULT partition. This feature is useful if you have two or more different versions of DOS or other operating systems on the drive.

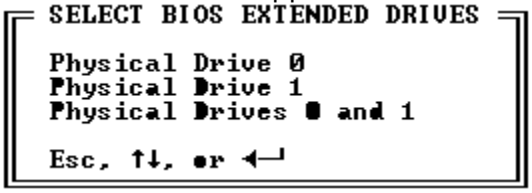
If you choose this type of MBR and you only have one partition or only one partition marked active ('bootable') then it will operate exactly like a *Micro House* Regular MBR (see below). The Multiboot partition can be used for most installations.

DrivePro Regular MBR

This type of MBR functions almost exactly like the one that the DOS FDISK program installs. The difference is that FDISK is not required (from which the Other DOS Regular MBR is extracted) for the installation, and so this function is faster.

DrivePro BIOS Extended MBR

This MBR allows you to have custom drive parameters independent of the table in the BIOS. See page 54 for more info on the *DrivePro* Custom Drive Type. This entry window will appear after selecting this type of MBR:



Choose the drive(s) for which you would like to define a Custom Drive Type. Press to continue or **ESC** to return to the previous menu. The final entry window defines the first (or only) drive parameters:

The ADDRESS field may be left at its default. This is the location in memory to place the drive parameters at boot. 0000:0300 (and 0000:0310 for the second hard drive if needed) is the address where most of the BIOSes that support their own user-definable drive types place their parameters, so it is reasonable to assume that this is a good place for the information.

The CYL, HDS, and SECTORS fields will contain the current system parameters. If these values are not appropriate, then you should enter the correct ones. The appropriate values can be found by using **Diagnostics, Drive Inquiry**.

The final four parameters: PRECOMP, ECC, CRTL, and LZONE are not used with IDE drives and may be left blank (zero).You must reboot the computer at this point so that the MBR can be loaded and executed for the new custom parameters to take effect. After rebooting, the drive must be partitioned and high-level formatted.

DrivePro IDE Antiviral MBR

This MBR is currently only supported on IDE drives. It is a strong defense against Boot Sector Viruses, that is, viruses that install themselves in the boot sector of a drive. If the *DrivePro* IDE Antiviral MBR is ever attacked by such a virus, at the next boot of the system, it will automatically post a message on screen, destroy the virus, and reboot the machine to completely rid the system of the virus. No user intervention is necessary. Boot sector viruses, such as the Stone virus and the famous Michaelangelo virus account for a large percentage of viral infections. Please note that for full protection, you should also backup regularly, use only disks from reliable sources, and obtain virus detection software to augment the *DrivePro* IDE Antiviral MBR.

DrivePro 512MB-2GB (>1024 cylinder) MBR

This feature is only available if the drive parameters for the drive (found in the BIOS data window at the Main Menu) indicate that the drive has >1024 cylinders, and if the system BIOS supports >1024 cylinders which *DrivePro* determines by doing an

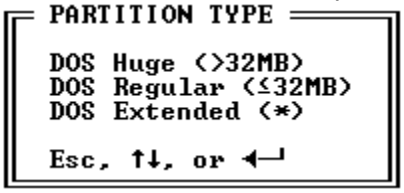
invisible test. The feature is used for drives that have more than 1024 cylinders and cannot be translated below that number of cylinders, namely, large ST506/412 drives and large ESDI drives with controller cards that do not have a BIOS. It is also used for any drive (besides SCSI) that is larger than 512MB. Typically, large drives that need this support are IDE since ESDI and MFM/RLL drives are rarely that large, and SCSI controller cards normally handle the large drives automatically.

Other DOS Regular MBR

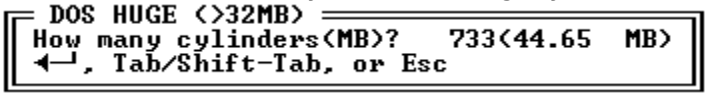
This type of MBR is extracted from DOS's FDISK. You must insert a disk containing FDISK or copy FDISK to your *DrivePro* disk to use this feature.

Create a DOS Partition

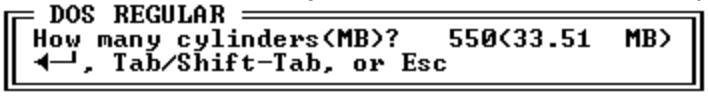
This option creates a standard DOS partition. A partition may only be created if there is enough room left on the drive for it, or if a partition can be erased to create room for the new partition.



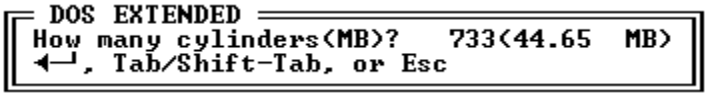
Select the type of partition to create. **DOS Huge** is only supported by MS/PC DOS versions 4.00 and later (Compaq DOS 3.31 and DR DOS 6.x will also support DOS Huge) and allows partitions of up to 8GB in size (although 4GB is the largest partition size supported by the *DrivePro* High-Level (DOS) Formatter). We recommend this type of partition if you have a DOS version that supports it. You will be prompted for the number of cylinders or megabytes to use for the partition. The default value is the number of cylinders or megabytes remaining or available on that particular drive:



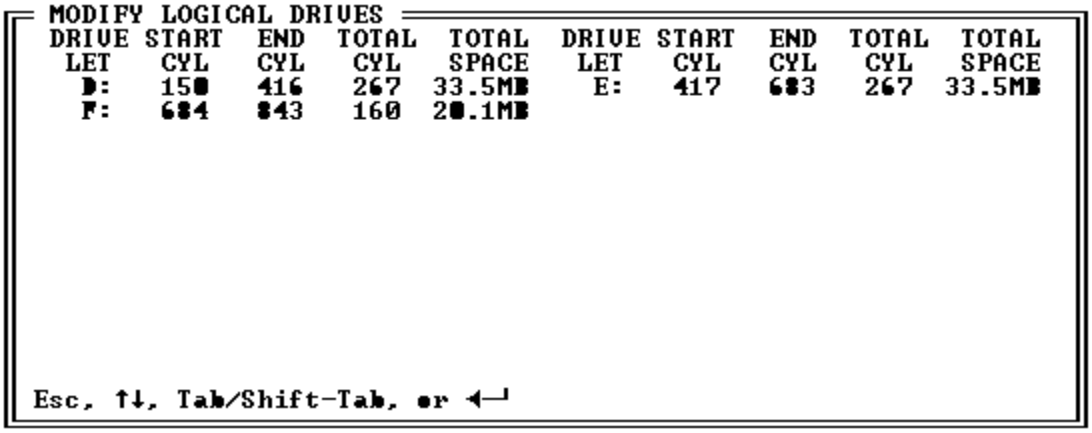
DOS Regular creates the first partition in sizes of 33.5MB or less. You will be prompted for the number of cylinders to use for the partition. The default value is the number of cylinders for a 33.5MB or less partition:



DOS Extended is used to create the partitions for the remaining disk space after a **DOS Regular** partition has been created. You will be prompted for the number of cylinders to use for the partition. The default is the number of cylinders remaining after the DOS regular partition (33.5MB or less) has been created:



This is the only partition type that has an additional option window.



You will be prompted for each DOS extended logical drive (size in cylinders). In the example above there are three logical drives defined.

Modify Logical Drives

Choose this option to modify logical drive sizes contained in a DOS extended partition (see the above information on creating a DOS extended partition). You must already have a DOS extended partition set up before selecting this option.

Delete Last Partition

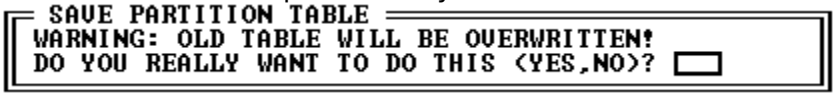
Deletes the last partition used in the table. Note that all data in a deleted partition may be lost!

Edit Partition Table

Allows the editing of the partition table. While in the editor you may use the following keys: **TAB** and **SHIFT+TAB** move between fields, the **SPACE BAR** changes selections in the **Bootable** and **System Type** fields and **ESC** returns to the **Set Up Partitions** menu and keeps changes made. Nothing is written to disk, however, unless **Save Partition Table** is selected from the menu.

Save Partition Table

Saves the partition table to disk. If there was already a partition table on the disk then it will be permanently overwritten!



You must type the complete word *Y-E-S* for the changes to be saved.

Undo All Changes

Choosing this menu item will set the table back to its original state before you changed any part of the table. This will not be possible if **Save Partition Table** has already been selected.

When you are through editing the table be sure to save it and press **ESC** to exit the **Partition Table/MBR** menu. (Pressing **ESC** without saving first will prompt you to save changes.)

Partition Table and Drive Parameter Limitations

There are many DOS versions. Each version has its own limitations. We recommend that you use the latest version of either MS-DOS (Microsoft) or DR DOS (Digital Research).

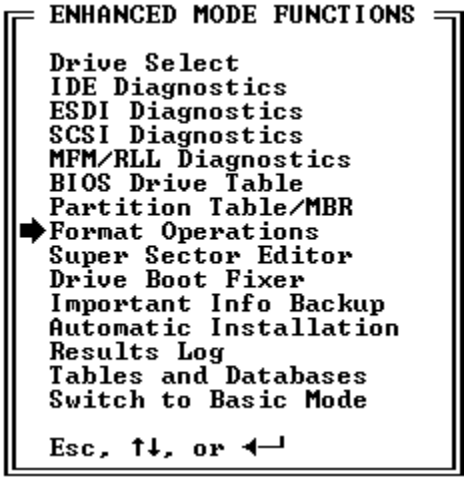
CLUSTER SIZE				PARTITION SIZE
MS/PC-DOS 2.x	MS/PC-DOS 3.3	Compaq DOS 3.31	MS/PC-DOS 4.x & 5.0 DR DOS 6.0 Novell DOS 7.0	
4KB	4KB	4KB	4KB	0 to 15.9MB
8KB	2KB	2KB	2KB	16 to 32MB
N/A	N/A	2KB	2KB	32.1 to 63.9MB
N/A	N/A	4KB	2KB	64 to 127.9MB
N/A	N/A	8KB	4KB	128 to 255.9MB
N/A	N/A	8KB	8KB	256 to 511.9MB
N/A	N/A	16KB	16KB	512MB+

All current versions of DOS have a limitation of a maximum of 1,024 cylinders. If the drive cannot be translated to less than 1,024 cylinders, you must 1) use the >1,024 Cylinder Support from *DrivePro*, 2) use a replacement block device driver in your CONFIG.SYS, or 3) truncate the drive at 1024 cylinders (resulting in the loss of drive capacity). Older BIOSes (regardless of the OS) are limited to 1,024 cylinders, but some newer BIOSes can use up to 4,096 cylinders, useful only with OSes other than DOS. Standard AT BIOSes are then also limited to 63 sectors per track and 16 heads. If you are using a SCSI or ESDI controller with BIOS, the AT BIOS is not used for drive access and so the number of heads can go up to 256. Without the *DrivePro* > 1,024 Cylinder Support or a replacement DOS block device driver in your CONFIG.SYS, MFM and RLL drives that exceed these limits must be truncated to meet the restrictions, losing some disk space. IDE drives can be instructed to use parameter translations that utilize the entire disk with no loss of disk space, up to 528 MB. SCSI and ESDI drives automatically use translations and thus lose no disk space (ESDIs up to 512MB, SCSIs up to 2 GB).

WARNING: If an MFM, RLL, or IDE is told to use parameters that exceed the DOS restrictions (number of cylinders greater in number than 1024), DOS will accept and *use these parameters* with no error message. Some BIOS tables, in fact, even have entries that are (currently) quite invalid! For example, some BIOS tables have 15 x 1224 x 17 as parameters for type 46. These parameters may even be the physical parameters for some drives, but if you select these parameters for a drive and the drive is told to write to a logical sector beyond the 1024 cylinder limit, DOS will write to the drive's cylinder 0 in a 'wrap-around' effect **without giving any error to the user**. This is due to little known anomaly in DOS. Writing to this location can be disastrous, since it will write over, in approximate order, the Master Boot Code and Partition Table, the DOS Boot Record, the FATs, the Root Directory, and then user data! Note that the drive will perform correctly until the disk is

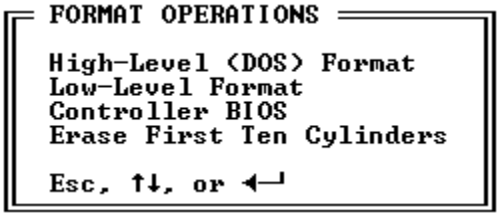
filled to the point that it begins to 'wrap-around.' So, even though the drive boots and can be written to and read from, the DOS restrictions still do apply! The results could become disastrous in time. Note also that this is a DOS software limitation and does not necessarily apply to other operating systems, and hopefully not to future DOS versions. Also note that interfaces other than SCSI are usually limited to 528 MB, unless you use a *DrivePro* 512MB-2GB MBR, or replacement DOS block device driver.

Format Operations



Format Operations

After selecting **Format Operations,** you will be presented with the following four options:



The formatting of a drive consists of :

Low-level formatting (most newer drives do not require low-level formatting)

Partitioning

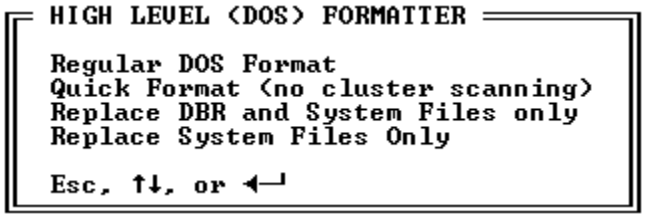
High-level formatting

The *DrivePro* menus do not follow this order, and are ordered by frequency of use. The **Automatic Editor** and **Guided Installation** do follow the normal flow of functions for installation.

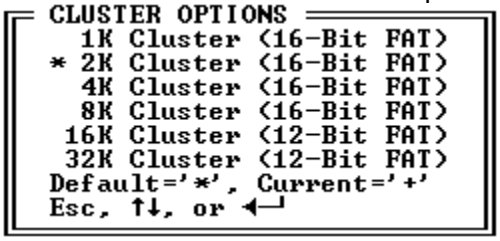
High-Level Format

High-Level (DOS) Format

A hard drive must be high-level (DOS) formatted after it has been partitioned. Upon selecting **High-Level (DOS) Format** in Enhanced Mode from the **Format Operations** menu you will be presented with the following choices (in Basic Mode, only the first two options will be presented):



Choose **Regular DOS Format** to have the entire hard drive scanned to record any bad DOS clusters (allocation units). **Quick Format** enables *DrivePro* to format any size drive in under 30 seconds because it does not scan the disk for bad clusters, and can be used when the drive is new or known to be defect-free. **Quick Format** usually is used on IDE, SCSI and ESDI hard drives since many drives automatically manage and "hide" defects and new drives are shipped with no defects visible to the operating system. The next two options should only be used by experienced users or when the drive has been backed-up completely!



Cluster Options - Choose Cluster Size

If you are in the Enhanced Mode and select the **High-Level Format** feature (either **Regular** or **Quick**), you will be prompted for the cluster size for the partition. The default cluster size (the cluster size that would have been selected by DOS's format) is indicated by an asterisk immediately to the left of the setting and will be the default highlit setting. There will be more than one cluster size that you can choose. The actual choices will depend on the size of the partition. Note that if you do not select the standard cluster size, diagnostic utilties such as *DrivePro* and Norton Utilities may report the DOS Boot Record / FATs as invalid. The FAT *is* valid, it just doesn't have the *expected* values. Do ***NOT***! let such a utility 'fix' your FAT as the result of doing so is that you are likely to lose all the data in that partition.

xxClusters are used in order to improve disk read/write performance, but using clusters wastes some disk space. Performance is improved by using clusters because there is overhead for each cluster read or write. Therefore, if the same amount of diskspace uses fewer clusters because they are larger, there are less units of cluster read/write overhead. Writing to and reading from each allocation unit costs certain memory and time overhead because DOS must keep an allocation table entry for each allocation unit. Accessing the FATs from RAM is much faster than accessing the FATs on the hard drive so DOS keeps a certain amount of the FATs in RAM. It will only load a certain amount in RAM, however, or else large FATs would occupy an unacceptable amount of system memory. If the FAT is small, it will be loaded entirely into RAM. If the FAT is large, only a portion of it will be loaded, and DOS will slow down as it loads other parts of the FAT as they are needed (overwriting parts that are already in memory). The designers of DOS decided to keep disk-access performance high, the usage of memory low, and the resources required to load a file low by keeping the number of allocation units low. The number of allocation units is kept low by using multiple sectors per allocation unit.

While using large amounts of disk space per allocation unit yields desirable effects, it does have the drawback of wasting increasingly larger amounts of disk space. Each allocation unit is the minimum amount of space that part of a file can be written to (this is what is meant by allocation unit). So if a file takes up any amount at all of an

allocation unit, it will effectively take up *all* of that allocation unit. The disk space (as sectors) that gets allocated to an allocation unit but does not actually contain any data is referred to as "slack space". As the cluster size increases, the same files that would take up many smaller allocation units take up few large clusters. In real situations, files rarely take up exact numbers of allocation units. So there is almost always some disk space being 'wasted'. The only way to avoid this would be to have the allocation units consist of one sector each, but this would cause slower read/write, and slower access time through the FAT. It is important to note that the larger the allocation units, the more disk space will be wasted on that partition. And large allocation units are used for larger partitions, so in general, the larger the partition, the greater percentage of the disk will be wasted.

If cluster sizes are supposed to optimize between performance and wasted disk space, why would you interfere with the defaults of DOS? Simply because the designers of DOS encoded their estimations of what was best for every user in every situation in the world, but did not provide a way for users to configure the cluster sizes to their own configurations. *DrivePro* finally allows users to determine the cluster size for partitions. In what ways can you benefit from determining the cluster size yourself?

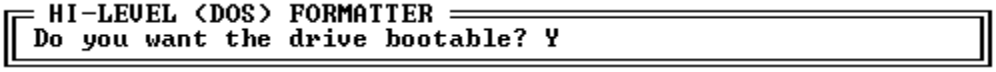
As mentioned previously, large partitions typically waste a very large amount of space. Usually this is not a problem since there is so much space. However, for users who do not wish to waste as much space, the allocation unit size can be selected at a lower size so that less space gets wasted for each allocation unit that is only partially used. Adjusting cluster sizes may also be used strategically across multiple partitions. In a configuration involving multiple partitions, at least one would have a large cluster size for the OS and programs, and at least one would have small clusters for data files. This configuration assumes that programs are large files, few in number, and that data files are small and numerous. Of course some programs may have many overlay files, database files, or other types of many, small files. For such programs you may reconsider the cluster size for that program, perfaps putting it on a partition with smaller cluster sizes. The larger cluster size tends to waste more disk space, but this tendency is compensated for by the fact that there aren't a large number of files and the files tend to occupy a large number of allocation units. The advantage is that the programs stored on large clusters will load noticebly faster. For the partition(s) that will contain data files, a smaller cluster size is likely in order. The smaller cluster size will waste less disk space since there will be less slack space. The files may load more slowly, but not substantially since they only occupy a few allocation units anyway. Of course two partitions are the minimum needed to benefit from using non-standard cluster sizes. You may feel that more than two partitions would benefit your system.

Replace DBR and System Files Only is similar to the DOS "SYS" command and will install a fresh DOS Boot Record and System Files on the drive, without affecting the FATs, or other data. Replacement of the DOS Boot Record is not recommended in a drive recovery situation, unless you are certain the partition table is correct, and have saved the corrupted DOS Boot Record. **Replace System Files Only** replaces only the DOS system files without affecting the DOS Boot Record and is the recommended step when attempting to recover a drive, provided the partition table and the DOS Boot Record are correct or have been manually corrected using the **Drive Boot Fixer**, and all vital data has been backed-up off of the hard drive.

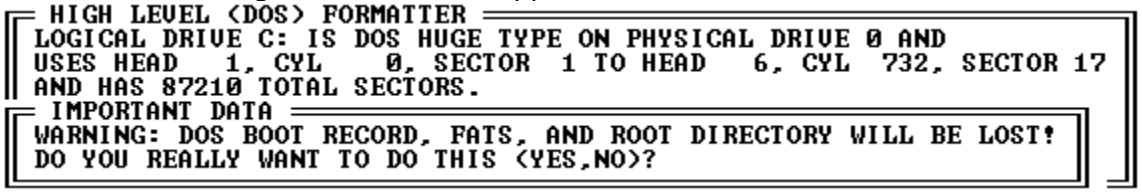
Please note that the recommended **first** step in any drive or data recovery situation is to get the data backed-up off the drive. Making the drive bootable should only be attempted **after** any vital data has been backed-up off of the hard drive! Consult a good book on hard drives and hard drive recovery for further information.

If you have more than one drive installed you will be asked for the drive letter to

format (usually drive C: or D:). You may enter AUTO to have all partitions formatted, with the first one made bootable. After making your selection the following window will appear:



Answer *Y* if you would like the DOS system files to be placed on the hard drive (if the *DrivePro* diskette does not have the system files on it, you will be asked during the high-level format to place a disk with the DOS system files into the floppy drive). The following screen will then appear:



This is your last chance to back out of the format! All data may be lost on the drive if you answer *YES*. Note that you must type the full word to continue, as the letter '*Y*' alone is not sufficient. If you chose to make the drive bootable, *DrivePro* will read the system files off of the disk and place them onto the hard drive during the format process.

The high-level format is the last step necessary to prepare the drive for use. Once the high-level format is done you may boot the system from the hard drive (providing you marked the partition as bootable previously and told the high-level formatter to make it bootable as well).

If the hard drive will not boot on its own then run *DrivePro* and select **Partition Table/MBR**. Look at the partition table windows at the bottom of the screen to ensure that the first partition is marked BOOTABLE-'YES'. If it is not, then choose **Edit Partition Table** and change the first partition to BOOTABLE-'YES'. Save your changes and reboot with the floppy pulled out. If the hard drive will still not boot, then redo the high-level format, being sure to answer '*Y*' to the question "Do you want the drive bootable?".

Low-Level Format

The low-level format sets up the sector headers and boundaries necessary for all disk read/write operations.

When to low-level format a hard drive

Most ST506/412 and ESDI drives need to be low-level formatted when initially installed. Most SCSI, and all IDE drives do not.

You may also want to do a low-level to "clean-up" a used drive. Redefining the sector boundaries and identifying any new bad sectors is a good idea when a drive starts to act poorly. One drive company informed us that ST506/412 drives should be low-leveled at least once a year to ensure that the sector boundaries are in order, even if the drive appears to be working fine. A **complete backup** should be done on such a drive, and then you would use either the **Controller BIOS** option, the *DrivePro* **BIOS INT13 Low-Level** option using the Non-Destructive mode, or the software included with or optional for the controller you are using (the choice is explained below). **To "clean-up" IDE drives we recommend using Erase First Ten Cylinders instead of low-leveling the drive, since many IDE drives can only be low-leveled at the factory and do not support user low-level formatting!**

Most ESDI and SCSI drives cannot be low-leveled through third-party software (such as *DrivePro*), and thus **must** be low-leveled via the controller's firmware or software

included with the controller. In those cases choose **Controller BIOS**, or use the appropriate supplied software for that controller.

MFM/RLL (ST506/412) drives and some ESDI drives (where the controller uses a BIOS INT13 low-level format) are low-leveled using the *DrivePro* **BIOS INT13 Low-Level** option.

IDE Low-Level Information

We recommend that you do *not* low-level IDE hard drives. The manufacturers do not recommend it and neither do we. In some cases, the bad track table is in jeopardy of being erased. The table is never placed in the same location for every drive and so it is impossible to know where to find it. To add to the hazards of an IDE low-level, the manufacturers of IDE drives never include a printed copy of the table with the drive. **Instead of a low-level on an IDE drive we suggest that you first try the Erase First Ten Cylinders feature**.

As far as low-leveling IDE drives, four possibilities exist:

The drive will ignore the low-level completely.

The drive will accept the low-level but only erase the areas containing data. This is actually only wiping clean the drive since the sector boundaries are not redefined.

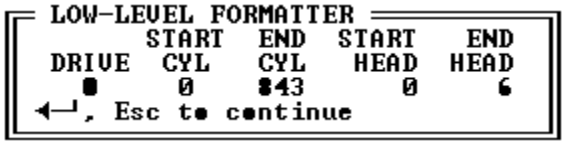
The drive will accept the low-level completely. Sector boundaries will be defined and the interleave may in rare cases be adjusted (note that almost all IDE drives will not allow the interleave to be adjusted since they have full track buffers and will ignore any interleave adjustment, and leave the interleave set to 1:1). A small fraction of IDE drives fit into this category.

The drive will accept the low-level and erase vital information thereby rendering the drive unusable.

We recommend that you contact the manufacturer of your IDE hard drive for more information and advice should you need or wish to low-level an IDE hard drive.

Using the Low-Level Formatter

Select **Low-Level Format** and the following entry window will appear:



Drive

If you have more than one drive installed then enter in the drive number (0 or 1). The default is 0. If the program can only detect one drive, then this field cannot be changed from the default of 0.

Start Cyl/End Cyl

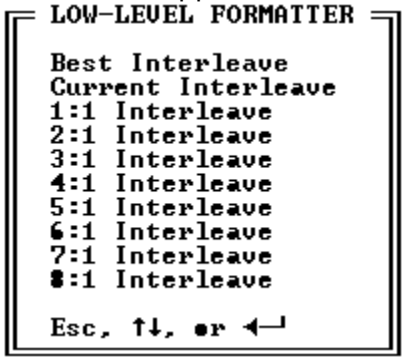
Enter in the cylinders at which to start and end the format. The defaults are the drive's first and last cylinders. If you wish to format only one cylinder you would set the start and end cylinder to the same value.

Start Head/End Head

Enter in the heads at which to start and end the format. The defaults are the drive's first and last heads. If you wish to format only one head you can set the start and end head to the same value.

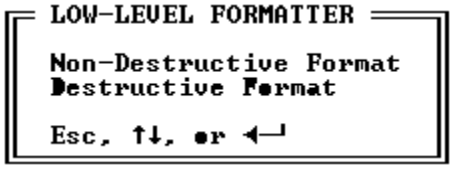
Press **ESC** when done modifying the parameters. The Interleave option window will

then appear:



Select the interleave to use (see the glossary for an explanation of interleave). **Best Interleave** will attempt to determine the optimum interleave on the current system, which is usually the best choice, unless you know what interleave you wish to use. Note that most IDE & ESDI drives do not always allow a true low-level (see above). In these cases the interleave will *not* be adjusted and will remain at 1:1 (the optimum setting for these drives) even if you make a selection other than 1:1. These drives contain a track buffer which renders interleave setting meaningless.

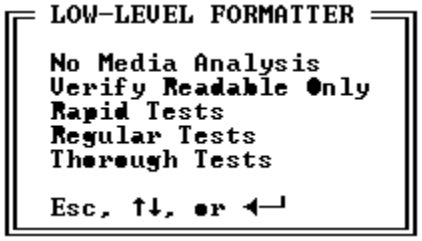
The next window is the non-destructive/destructive options box:



Choose **Destructive Format** if it is okay for the data to be destroyed, which saves the time involved in saving and restoring the data on each track as it is being formatted, speeding up the low-level format process. Choose **Non-Destructive Format** to low-level the drive but preserve the data. This is recommended once a year to help prolong the life of the drive and safety of data or adjust the interleave (if it can be adjusted), to improve performance, on an up-and-running hard drive, all without having to necessarily **restore** all the data afterwards.

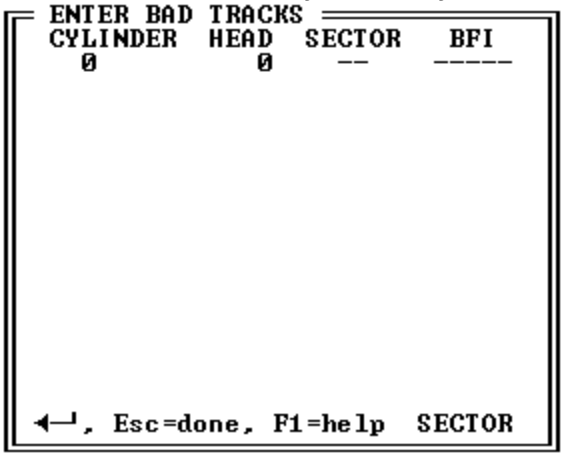
Even if you will be doing a non-destructive format, ***BACK UP YOUR DATA!*** The "non-destructive" format is only non-destructive if no power failures or other errors are encountered. This procedure reads the data off the disk into memory, formats the sector, verifies the sector, and then places the data back onto the sector. If power or the drive/controller electronics fail during this procedure YOU ARE LIKELY TO LOSE SOME DATA! Take the time to backup, you may be glad you did!

The next options window allows you to select the degree of media analysis:



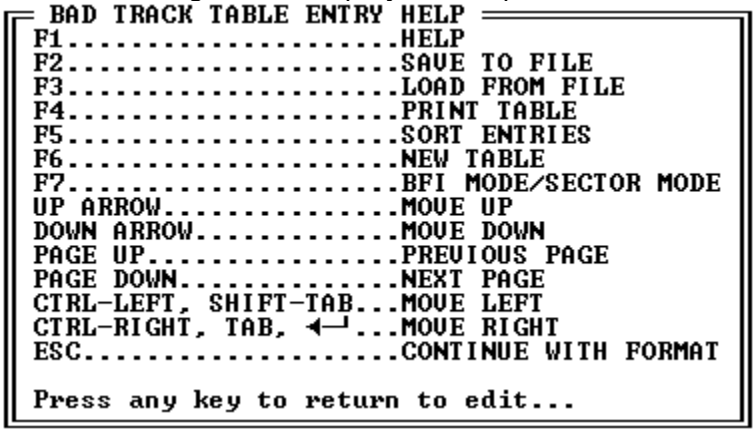
The degree of thoroughness increases from the top down. **No Media Analysis** is very quick but does no testing, all the way down to **Thorough Tests** which do very thorough testing but take a long time to complete.

In the final entry window you should enter in the drive's known defective areas:



For most drives, there is a list affixed to the drive case listing these defects. You should enter these into this window, even if you choose to do thorough testing. The drive manufacturers have sensitive equipment that can detect problem areas and *future* possible problem areas. These areas are the ones that they supply you with on the drive case. Not all drives include this table, especially IDE drives. Some labels specify sector and some BFI. Pressing F7 will toggle between the two modes. The word 'SECTOR' or 'BFI' will appear in the lower right corner of the entry window.

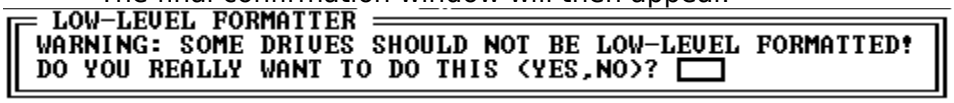
Pressing F1 will display the help window for bad track entry.



DrivePro will allow you to read a previously saved Bad Track Table, save a Bad Track Table to disk, print the Bad Track Table, wipe clean the Bad Track Table, etc.. To see the options, press F1 for help.

If you have no defects to enter, or you are done entering them, then press the **ESC** key to continue with the low-level format setup.

The final confirmation window will then appear.



This is your last chance to stop the format. Remember, all data on the drive will be lost if you selected a destructive format! The full word *Y-E-S* must be entered in (the letter *Y* alone is not sufficient).

Low-Level Technical Information

The rest of this chapter explains low-level format information for each of the interface types in detail. Reading this section is not necessary for use of the software.

ESDI Drives

Most of these drives ***must*** be low-leveled via the controllers on-board firmware. Most have variable sectors per track. They also use a technique called sector skew. This moves the sectors slightly from track to track to account for track-to-track seek time and head switch time. This figure can be calculated by dividing the track-to-track seek time and/or head switch time of the drive by 16.6ms and then multiplying this number by the number of sectors per track. Round the final figure up and you will have the optimum sector skew value. Currently, *DrivePro* does not support this feature. If you would like to implement it on your drive then you will need to run the low-level code on the controller's BIOS or software included with the card (if available).

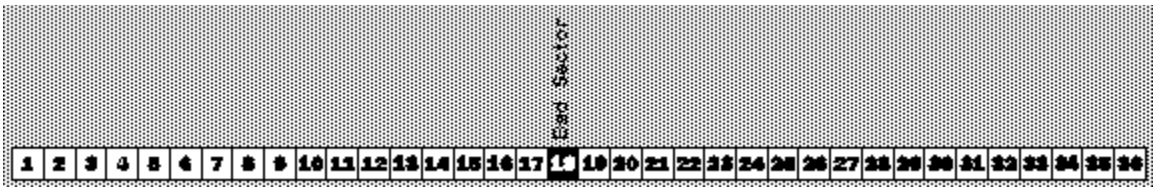
There is also another factor that comes into play: sector sparing. This reduces the number of sectors on each track by one and places defect information on the drive. The application will see less defects since only the drive is aware of the spare sectors. This reduces the total capacity of your drive but is useful if the drive has a large amount of defects and your application requires a defect-free drive.

SECTOR SPARING FOR A 36 SECTOR DRIVE

OPERATING SYSTEM SEES 35 GOOD SECTORS AND NO BAD SECTORS

SAME DRIVE WITHOUT SECTOR SPARING

OPERATING SYSTEM SEES 35 GOOD SECTORS AND 1 BAD SECTOR



IDE Drives

See the above information on the low-leveling of these drives (page 34). Most IDE drives have variable sectors per track and use sector sparing. They also usually utilize wedge servos. These servos mark the track and sector boundaries. Wedge servos are written to the platters by a special machine. The drive itself is unable to recreate these. It is impossible to low-level this type of IDE drive because there is no way to rewrite these servos unless the special machine is used. The drive electronics are intelligent enough to refuse a low-level command. On these types of drives the program simply "scrubs" clean the data areas.

SCSI Drives

Most SCSI drives need to be low-leveled through the firmware on the controller or through the software that comes with the controller. Some SCSI drives ignore a low-level command. In this case the only thing that can be done is to "scrub" clean the data areas. *DrivePro* will properly low-level format only if the SCSI controller supports the BIOS INT13 format command properly.

ST506/412 Drive

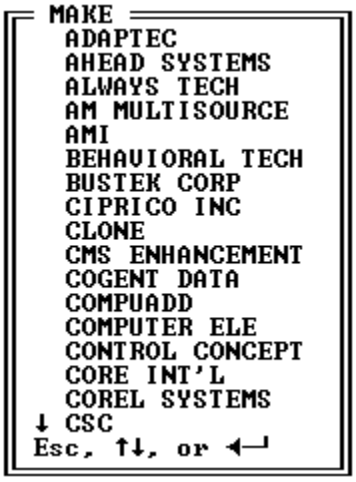
These are the simplest of all the interface types. There are two types of encoding schemes used. They are MFM and RLL. MFM always uses 17 sectors per track and RLL usually uses 26. Be sure to choose a controller that matches the type of the drive you are installing.

A warning about RLL: Do not use MFM drives on an RLL controller! It can be done (and often is) but the life of the hard drive will be greatly reduced and may result in the loss of data. This is because RLL uses higher bit densities requiring better media.

Controller BIOS

This menu option will locate the controller and execute any diagnostics or setup code that may be contained in its firmware. This is synonymous to executing DOS's DEBUG command and entering in the controller's firmware address, such as "G=C800:5".

This feature is especially useful with ESDI controllers that require the drive be setup through the controller. The Western Digital WD1007-xx is one such series of controller.



First, select the make and model of your controller. The list of controllers, as well as most of the databases in *DrivePro* have a quick search feature. Typing in the partial or full text of the item that you are locating will quickly jump to that section of the database. For example, typing "WES" will quickly take you to Western Digital.

After selecting the make and model, *DrivePro* will do its best to search the computer's memory for the controller. It will then give you one or more options depending on the controller you've selected. If you select an address and the system 'locks-up' then reboot and check the documentation and jumper settings on the card for the correct BIOS address. *DrivePro* has default addresses for many controllers and will indicate the default address of available by an asterisk ('*') to the left of the address.

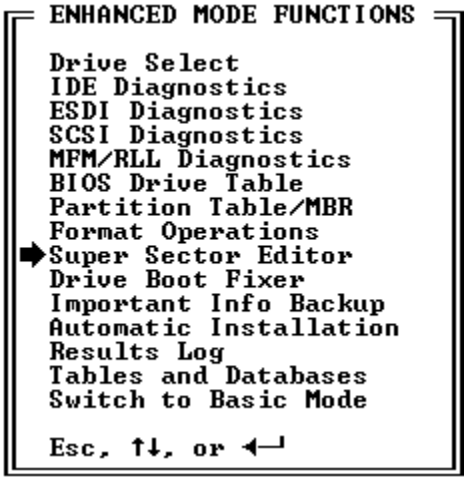
Complete functional specifications for controller cards can be found in **Tables & Databases, Controller Cards**.

Erase First Ten Cylinders

This feature erases the first ten cylinders on the hard drive. **If you feel that an IDE drive needs to be low-level formatted, use this option instead!** This will erase only the first ten cylinders of the disk that are designated for information storage (MBR, partition table, DBR, FAT, etc.) and so makes the boot area of the drive as clean as when new. This is also a quick way of removing virus fragments, a corrupted MBR or foreign partition table (such as Novell), and you will then be working with a 'like-new' drive. This may be used on all interface types (ESDI, IDE, SCSI, MFM/RLL). This feature has been known to 'fix' drives that hung when booting due to garbage being left in supposedly unused areas of the drive.

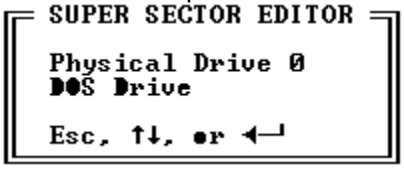
Of course, all data on the hard drive will be lost!

Super Sector Editor



Super Sector Editor

The Super Sector Editor will allow you to view and alter disk sectors.



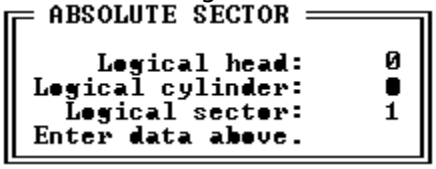
A **Physical Drive 1** option will appear if more than one hard drive is installed.

Physical Drive 0

This option can be selected for a drive that has not been DOS formatted yet or can not be accessed by DOS, or when accessing sectors by their physical address. It has a submenu with only two options:

Selecting the **Master Boot Record/Partition Table** option will take you to the MBR/partition table section of the hard drive.

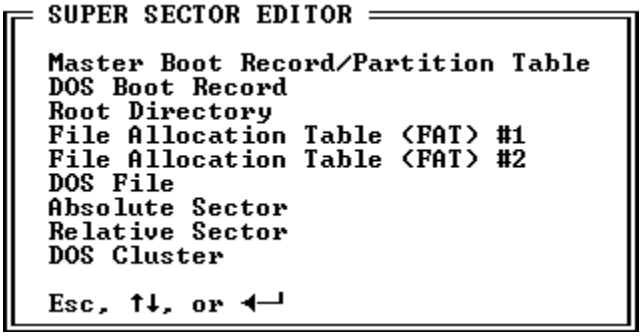
Selecting **Absolute Sector** will present you with the following data entry window:



See below for available options while in the Super Sector Editor.

DOS Drive

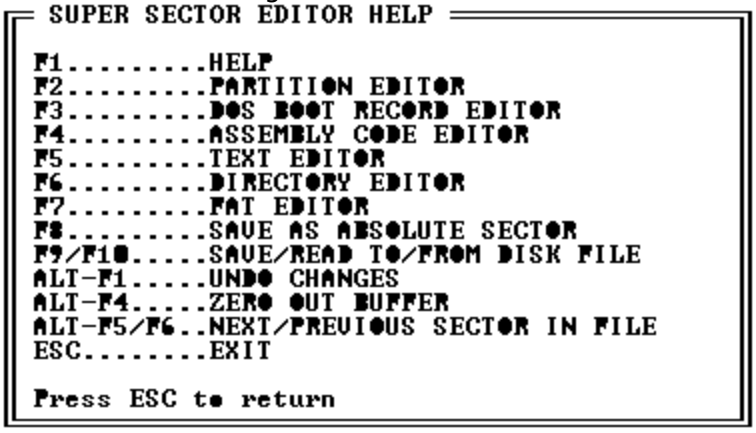
This option contains a submenu with several features:



Select the location of the disk that you want to inspect. For most of the options the sector is initially shown in its special format. This is selected automatically.

When viewing a sector in the default view mode, the offset is shown on the left, followed by a colon (':') and 16 hex bytes, and finally a representation of those bytes as printable characters. A byte value of 0, 7 or 8 is represented as an inverse-video '0', '7', or '8'. A carriage return character is represented as an inverse-video 'D' and a line feed as an inverse-video 'A'. A byte value of 255 (FF in hex) is represented as an inverse-video 'F'. You can toggle between editing the hex bytes or the characters by pressing the **TAB** key. On a color monitor any bytes that are changed from their original value will be shown in yellow, and those that are unchanged will be shown in white. If the changes are written to disk, all changes in yellow will become white to reflect this.

While viewing a sector in the HEX format the available options are:

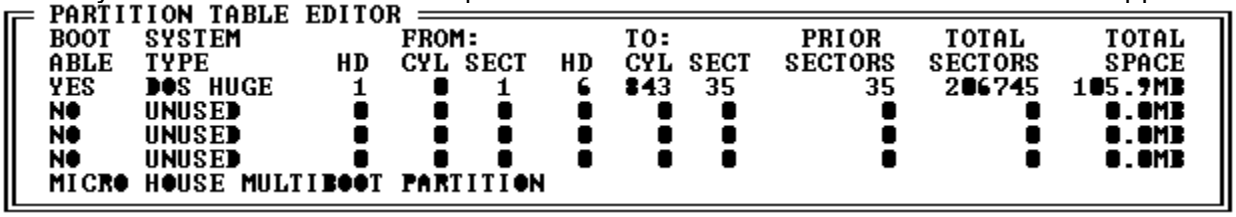


F1 - Help

Displays all available commands while in the **Super Sector Editor**. Also check here for any new features that may have been added since this writing.

F2 - Partition Editor

This will force the display to a partition table. Note that the sector you are editing may not be a partition table but you may still view and edit it as one. If the sector you are on contains a valid partition table then a window similar to this will appear:

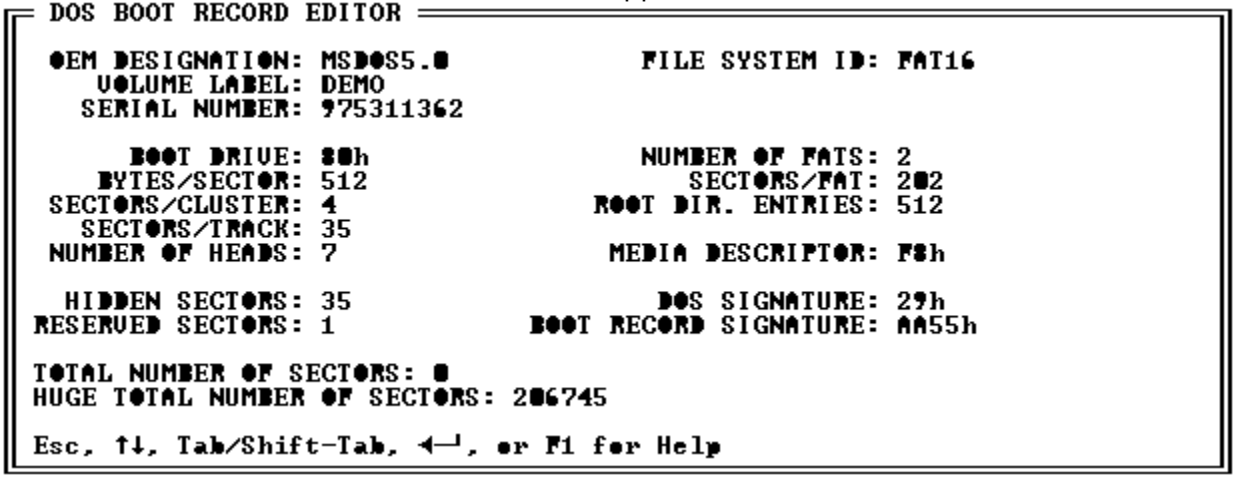


While in the editor you may use the following keys: **TAB** and **SHIFT+TAB** move

between fields, the **SPACE BAR** selects values in the **Bootable** and **System Type** fields and **ESC** exits back to the hex editor and keeps any changes made. By pressing F1 you can view the options available while in the Partition Table Editor, including the ability to rapidly advance to Extended Partition Tables for all Logical Drives on the physical hard drive (providing you are editing the Master Boot Record sector set up properly and containing a valid Extended Partition and Logical Drives).

F3 - DOS Boot Record (DBR) Editor

This will force the display to a DBR. Note that the sector you are editing may not be a DBR but you may still view and edit it as one. If the sector you are on is a valid DBR then a window similar to this will appear:



The DBR for the first (or only) partition can usually be found on head 1 ,cylinder 0, sector 1.

Invalid parameters in the DBR will result in a message on-screen indicating that the DBR is invalid. Fields that are clearly invalid will be marked in red (or another shade on monochrome displays) to aid in restoring a DBR. If the sector you are editing is not a DBR sector this will occur, and you should not make any changes unless it is in fact a DBR sector.

You can use the **Repair DBRs** feature described in the **Drive Boot Fixer** chapter on page **Error! Bookmark not defined.** to recreate a true DBR sector, provided the partition table is accurate.

F4 - Assembly Code Editor

This view is very helpful for patching program code without having to load it into DEBUG or having to edit the source code of a program and then recompiling. An understanding of assembly language is required.

F5 - Text Editor

The entire edit area will be in text. This is helpful for editing sections of text files. Note that the reverse video characters mean the same in both this editor and in the HEX display (see above for explanations of each one).

These values can be typed-in (except for the 0, which must be entered in HEX mode) by holding down the ALT-key, typing the byte value on the numeric keypad, and then releasing the ALT-key.

F6 - Directory Editor

This will force the display to a DOS directory. Note that the sector you are editing

may not be a directory but you may still view and edit it as one.

F7 - File Allocation Table (FAT) Editor

This will force the display to a DOS FAT. Note that the sector you are editing may not be a FAT but you may still view and edit it as one. *DrivePro* will attempt to automatically determine whether the FAT that you are editing is 12-bit or 16-bit, but in data recovery you should ensure that it is correct before making any changes.

F8 - Save As Absolute Sector.

You have the option to save the current sector to any sector on the hard drive. Be careful, since you can overwrite vital data on the hard drive using this powerful feature!

F9 - Save/Read To/From Disk File

This will save the current sector to a disk file or read from a file into the current sector. You will be prompted for the file name. You may optionally precede the file name with a drive and directory designation.

Alt-F1 - Undo All Changes

Puts the sector back to its original state before any changes were made, causing any changes to be discarded.

Alt-F4 - Zero-out Buffer

Clears the entire sector to zeroes.

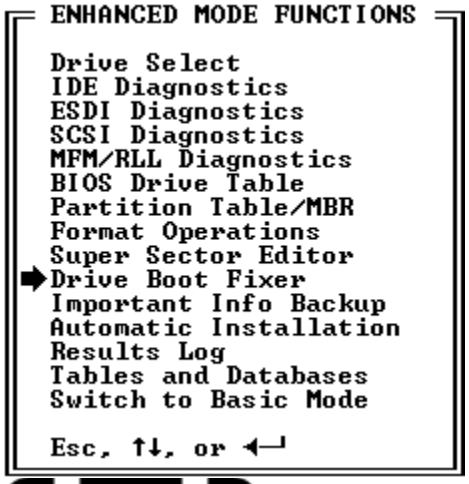
Alt-F5/F6 - Next/Previous Sector In File

Moves to the next or previous sector if currently editing a file. *DrivePro* will emit a beep if there is no previous or next sector found or if the sector is not a valid file sector.

ESC - Exit

Will prompt for save if changes were made, and return to the Main Menu.

Drive Boot Fixer

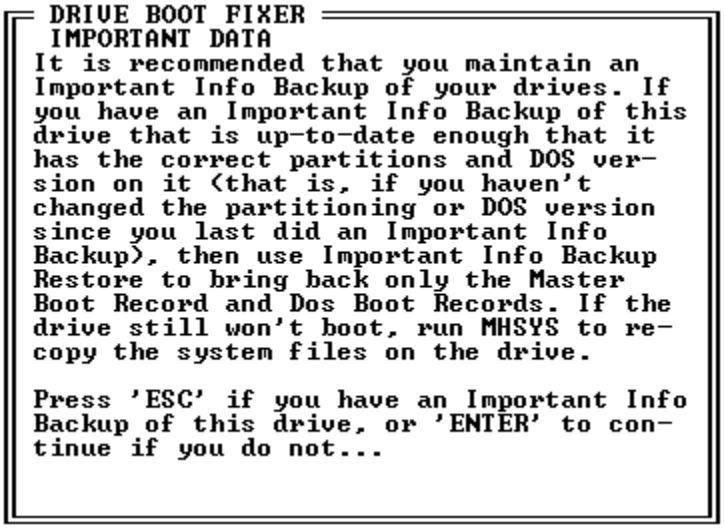


Important Note: Your first priority in any drive or data recovery situation is to back-up any important data off of the drive, by booting from a floppy and switching to the hard drive prompt(s), or by repairing the vital areas one at a time until you can get to the hard drive, and then backing-up data to floppies or other media. **DO NOT WRITE ANY FILES OR DATA TO THE HARD DRIVE!** Making the drive bootable is the second priority, once you know your important data is safe! Consult a good hard drive and data recovery book for more information.

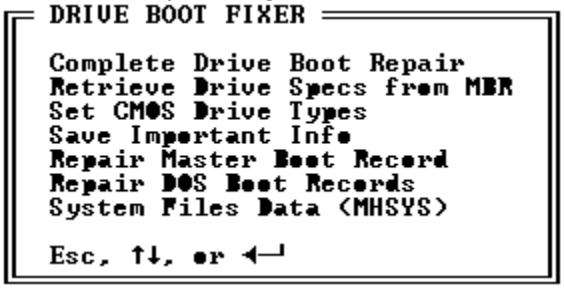
You may encounter drives that simply will not boot. Sometimes the only way you have to correct this situation is to reformat the drive which will destroy all data.

Drive Boot Fixer has been designed to help alleviate this problem. It will attempt to reconstruct the areas of the disk that are required for booting. It can reset the CMOS drive type, backup vital system areas in preparation for editing to recover files, and repair or create a new master boot record (MBR) and DOS boot record (DBR), which may be all that is required to recover the drive. You can also optionally place a new copy of DOS on the drive, after ensuring the MBR and DBR are correct, by using the **High-Level Formatter** in **Format Operations**, or the included utility, MHSYS. You should first boot from a floppy and use MHDRIVE if it has a *DrivePro* BIOS Extended MBR, and then copy all your important files off the hard disk. **Remember** that your data is first priority, and getting the drive to boot is secondary.

The first window that appears is an instructional window. Please take the time to read this if using **Drive Boot Fixer** for the first time.



After pressing the next window to appear is the options menu:

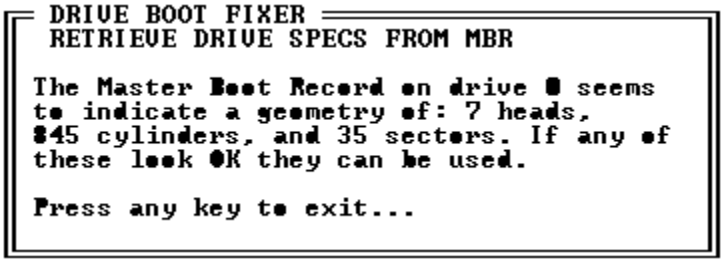


Complete Drive Boot Repair

This selection will perform all of the below steps in the order that they appear. You may press **ESC** at any time during this procedure to abort.

Retrieve Drive Specs From MBR

This option is ideal for systems that have lost their CMOS setup information. *DrivePro* will look at the drive and return what the CMOS settings were when it was initially setup. This is particularly useful for IDE drives because they can have a multitude of different translation specifications. **To access these drives properly, the exact parameters used to configure the drive must be reentered**. You will receive an information window similar to the following:



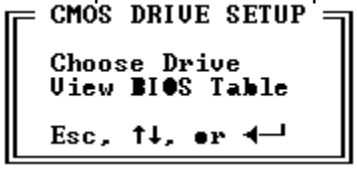
Of course the actual values are dependent on the drive. Upon exiting you must choose the next menu option **Set CMOS Drive Types** to set the CMOS to the specifications given in this window. If you cannot get this feature to recognize the hard drive, it is likely that the drive is not attached correctly or has gone bad.

One final note on this feature: If the partition table is damaged or changed

from its original setup, then this feature will return the new invalid incorrect parameters!

Set CMOS Drive Types

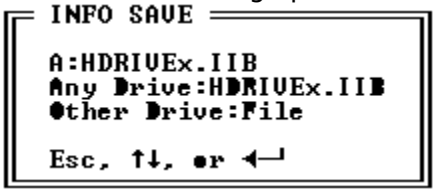
After an initial information window, this selection gives you two options by which to setup the CMOS drive type (if the CMOS drive type is already set correctly, you can press **ESC** to skip this step):



Choose Drive allows you to select the drive by make and model. **View BIOS Table** allows you to set the CMOS by selecting an entry in the BIOS drive table. You could also set the drive type by selecting **Drive Select** or **BIOS Drive Table** from the main *DrivePro* menu.

Save Important Info

This option allows you to save important sections of the drive data to a file before you attempt repairs, so that you can always restore the drive to the last time this option was performed. This option is identical to the main *DrivePro* menu option **Important Info Backup**. After an initial information window, this selection gives you the following options:

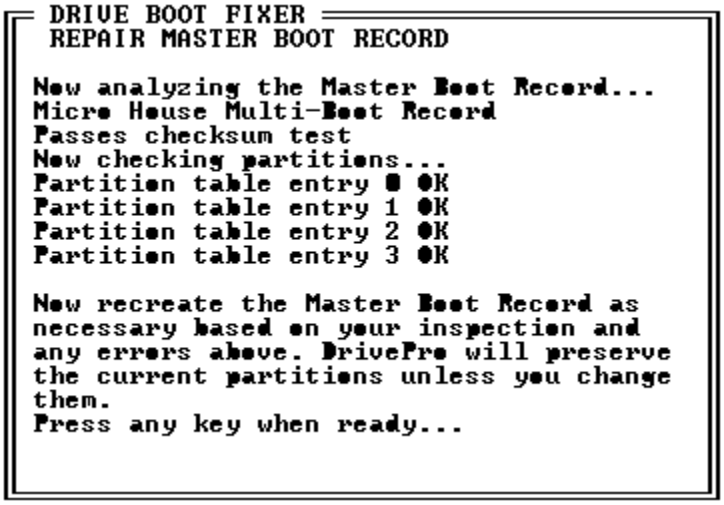


WARNING: You should not save to the hard drive! Normally the save is done to a floppy drive.

MHSAVE.EXE is a stand-alone utility that performs the same function of saving the important system info. Either MHSAVE or *DrivePro's* **Important Info Backup** should be performed regularly, preferably every time the computer is booted (you would put MHSAVE in your AUTOEXEC.BAT file), to drastically simplify the repair of future drive problems, and help prevent data loss. Information on MHSAVE can be found on page 65.

Repair Master Boot Record

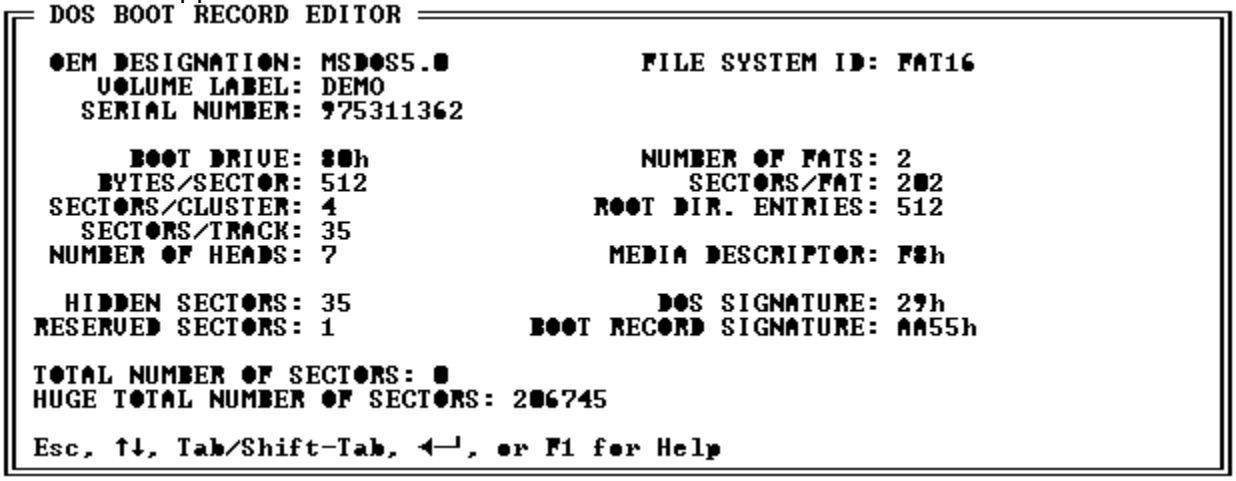
This selection analyzes the Master Boot Record (MBR) for errors. An informational window will appear after the analysis is complete:



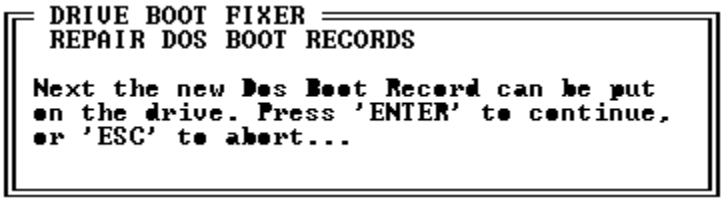
If no errors were found, then you may press the **ESC** key to exit. If errors were found, then press any other key to go to the Partition table/MBR editor. This editor is exactly the same as the **Partition Table/MBR** option found on the main *DrivePro* menu *except that using the option to do a* **New Master Boot Record** *while in* **Repair Master Boot Record** *does not erase the partition table*. This is especially useful in situations in which the Master Boot Program needs to be overwritten, either because it is corrupted by virus or soft errors, or because a *DrivePro* Custom Drive Type was accidentally created and needs to be removed without disturbing the partition table. Information on the **Partition Table/MBR** can be found on page 26.

Repair DOS Boot Records

This feature analyzes the DOS Boot Records (DBR) for errors. You will have the option to repair any discrepencies. An informational window will appear after the analysis is complete. If all information looks correct, nothing appears in red, and *DrivePro* reports the DBR checksum as "OK" then you may press ESC to exit the **Drive Boot Fixer**. Press any other key to enter the DOS Boot Record Editor to make any further changes and check over all changes made. The DOS Boot Record editor will then appear:



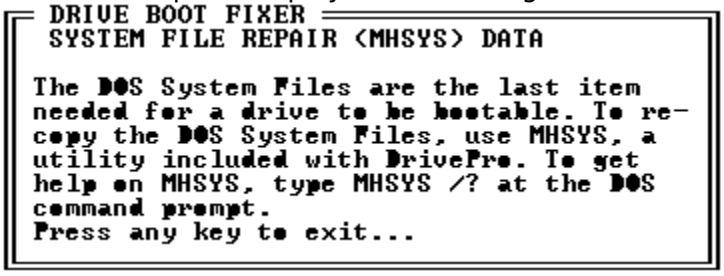
After making your changes press the **ESC** key and the following information window will appear:



If you do not want the changes that you've made to be written to the drive then press the **ESC** key. Otherwise, press **ENTER** and your changes will be written to disk.

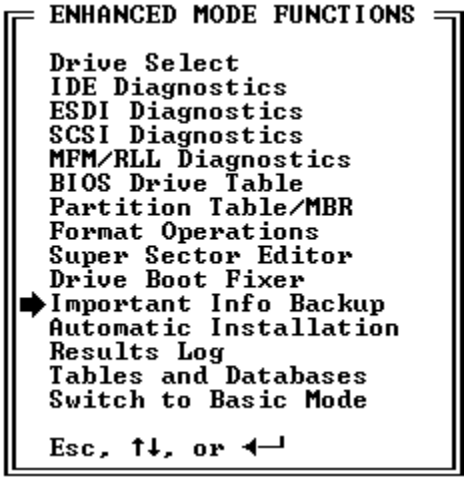
System Files Data (MHSYS)

This option displays the following information window:



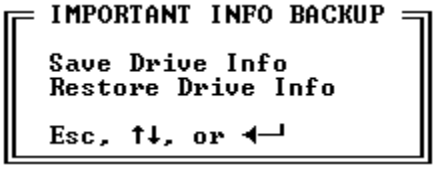
You should not use the SYS command included with DOS when trying to recover a drive, as it may overwrite the DBR and leave the drive in a worse state than before! Instead use MHSYS, or the option **Replace System Files Only** under **High-Level (DOS) Formatting** in the **Format Operations** menu which merely copies the system files onto the drive. You should ideally do the previous steps of Drive Boot Fixer, and check the root directory and FATs after repairing the DBR, to avoid the unlikely possibility of overwriting about 100K of user data (for system files), and to gather data to diagnose and fully repair the drive. It is recommended that you consult a good book on hard drives and data recovery for more information. Additional information on the MHSYS utility can be found on page 65.

Improtant Info Backup

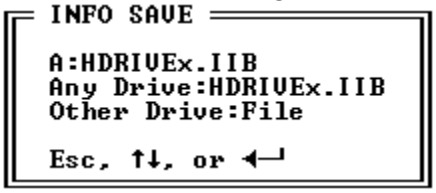


Important Info Backup

Losing the Master Boot Record (MBR), the partition table, DOS Boot Record (DBR), the File Allocation Table (FAT) or the CMOS drive settings will cause the system to be unable to access the drive correctly. With **Important Info Backup** you will be able to save the vital disk information to a file for later retrieval in the case of disaster. If you are a technician that installs many systems, we suggest that you use this option on every system you set up, that is, save the important system info for every machine to a diskette that will remain with the machine. If any of the mentioned information is lost from the system, as long as the drive can still be written to, running **Restore System Info** will restore the state of the system at the time of the last system save. This can save a great deal of time with the newer systems that have many settings and with IDE drives that must use a "user-definable" drive type. Specifically, with the IDE translated parameters, if you don't use the original specifications (heads, cylinders and sectors) when resetting this, the IDE drive will not work properly. Note that if determining the drive's original parameters is your primary task, this is best accomplished through **Get Lost Parameters** in Basic Mode or **Drive Boot Fixer, Retrieve Drive Specs From MBR** in Enhanced Mode, and entering the parameters into CMOS.



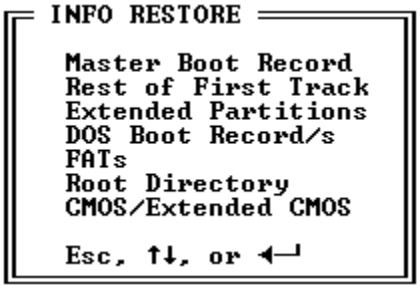
After selecting **Save Drive Info** you have several "save to" options:



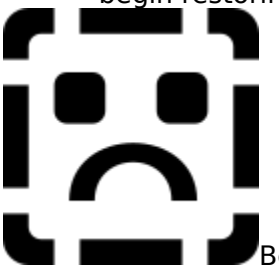
You have the option to save to a hard drive, floppy, or other media. You should save to a floppy drive or other drive (such as a network drive or other backup type device). If you save to the hard drive and it fails, there may be no way to retrieve the data.

When restoring the **Important Info** from the disk file you have the option to choose

which items to restore:



Toggle an item for execution by pressing the key and **ESC** when through selecting to begin restoring the data.

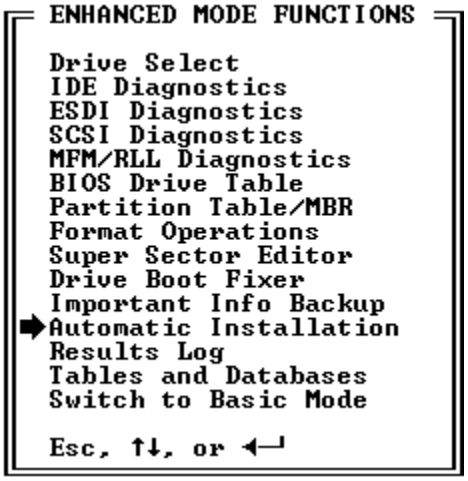


Be sure that you only restore the information you need. **Be careful not to restore drive info that is obsolete. Restoring obsolete data will permanently destroy any changes made to the disk since the last system info backup and can result in data loss.** Typically, the only data that will be obsolete is the FAT since it is altered each time a DOS directory is altered (a file created, deleted or changed). The Root Directory could also be obsolete if it was changed since the last backup.

The MHSAVE Program

You will find a stand-alone program on the distribution disk called **MHSAVE.EXE**. This program can be executed from the AUTOEXEC.BAT file. It serves the same function as **Important Info Backup,** but can be executed at every boot, thus making the system info file more closely up-to-date (though the system info will not be perfectly up-to-date if changes are made after the save). It is *very* important to have current data in the backup file and this utility is very useful in accomplishing this. See page 65 for more information on MHSAVE.

Automatic Installation



Automatic Installation

This menu selection has three areas:

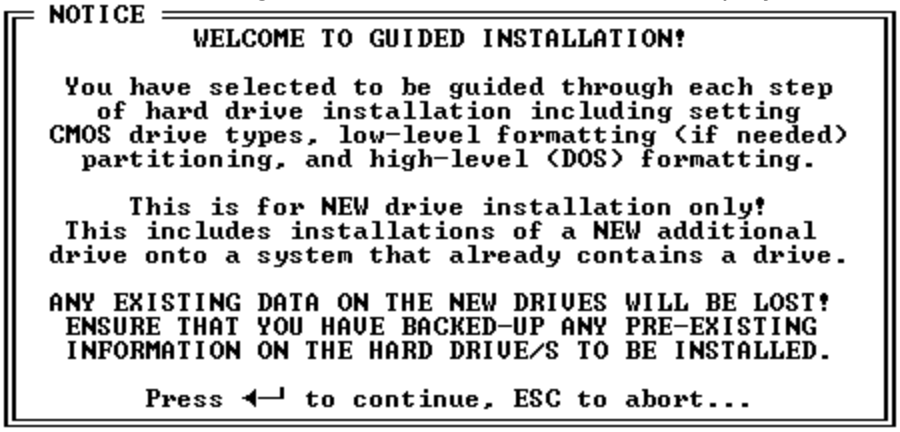
The Automatic Installation Menu

Guided Installation

This menu item will guide you though all of the steps required to install a new hard drive or drives. This should be selected for new drive installations due to its ease of use.

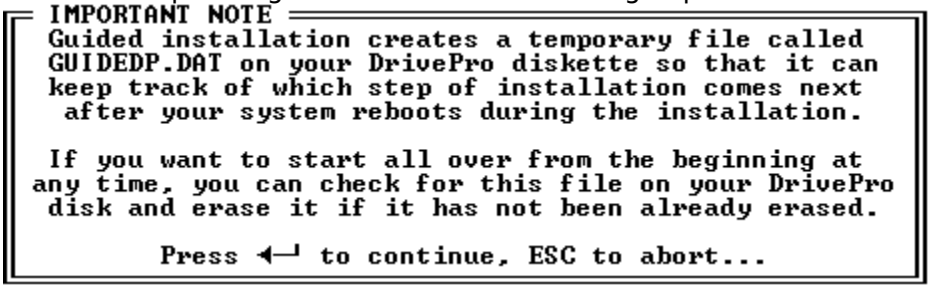
If you are installing a new IDE drive, IDE Quick Setup (see below) should be chosen instead of Guided Install. This is much faster and requires no rebooting until installation is complete.

The following informational windows will be displayed after choosing **Guided Install**:



Note: If ESC is pressed during Guided Install then *DrivePro* will exit you to the DOS prompt. You will not be returned to the Main Menu.

After pressing to continue the following important note will be displayed:



Guided Installation requires the system to be rebooted after setting the CMOS drive type. You will be walked through each step necessary for installation of the hard drive(s). These steps are not listed here as they are self-explanatory and vary for each different drive type and system.

IDE Quick Setup

This option is only valid for IDE type hard drives. It will perform the following steps all automatically without rebooting until the process is complete:

Determine parameters for the drive from the drive automatically

Set the systems CMOS drive type or install a Custom Drive Type

Partition

DOS format

Transfer DOS to the drive so that it can boot

This selection is the quickest and easiest way to install new IDE drives. Note: A standalone version of the IDE Quick Setup only, called EZ-Drive, is available as a separate product from your distributor.

Start Macro Recorder

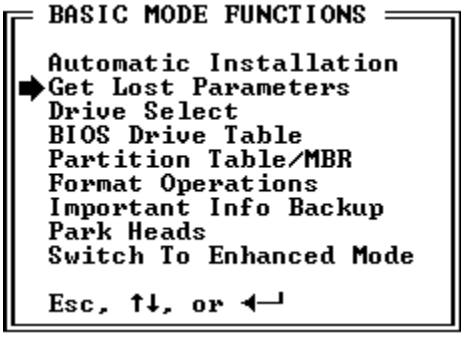
This and the following feature are only available when in the Enhanced Menu.

Use this feature to record the keystrokes of a macro. Typically macros are used when many similar drives are to be installed. Merely install one drive, recording all the keystrokes, and the keystrokes can be played back to duplicate the procedures on successive drives. While recording, reboots are simluated, so go ahead and answer "YES" whenever a reboot is required. When you are through recording, select Automatic Installation again and select **Stop Macro Recording**. You will be prompted for the filename to which to save the macro.

Start Macro Playback

To load and execute a macro from file on diskette, either use this menu command or run *DrivePro* with the /Afilename switch.

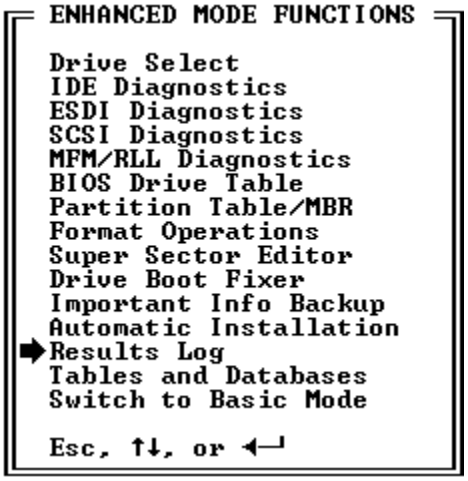
Get Lost Parameters



Get Lost Parameters

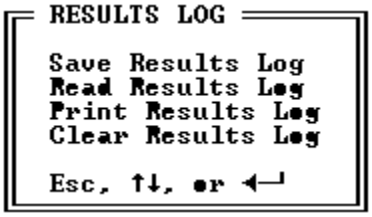
Selecting this feature will return the parameters with which the drive(s) were originally partitioned. This is useful in situations in which the CMOS settings have been lost, and you are trying to set the drive correctly in CMOS. Note that this actually obtains the parameters from the partition table, and does not merely return the manufacturers recommended parameters. This means that DrivePro will return the correct (appropriate) parameters regardless of what parameters were originally used. This feature is the same as the feature **Drive Boot Fixer**, **Retrieve Drive Specs From MBR**, in Enhanced Mode.

Results Log



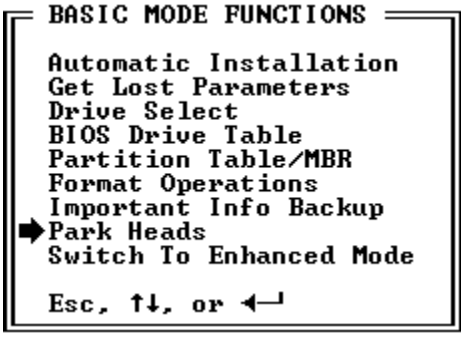
Results Log

The **RESULTS LOG** window logs the selection and results of operations. Selecting **Results Log** allows the manipulation of this log. Note that the log may only be read from within *DrivePro* as it is not merely text but also contains formatting information.



From the MAIN MENU you can select and deselect the Results Log Window by pressing the **TAB** key and then scroll up and down with the and arrow keys. This is useful when the results from previous operations have scrolled off the screen. When exiting *DrivePro,* you will be prompted to save the log to a file. Using the /S command-line switch at start-up causes *DrivePro* to omit prompting to save the log, and of course the results log will not be saved unless you specifically go into the menu above and save it.

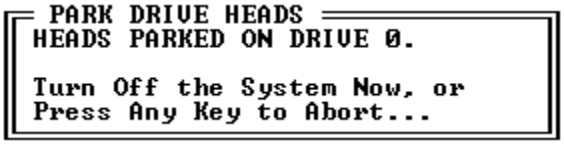
Park Heads



Park Heads

This feature is only present on the Main Menu in Basic Mode. When in Enhanced Mode, it is available from the applicable Diagnostics submenus.

This feature moves the drive's heads to a safe zone away from data. It is recommended that you park the drive heads before shipping or any other movement of the computer. The heads should also be parked when the drive is idle. If the heads are 'parked' and the drive sustains a jolt during moving, unused cylinders will be damaged instead of data-laden ones. Newer hard drives automatically park themselves when power is removed so this feature will usually not be needed. **Park Heads** may be used in any case without damage to the hard drive so it is good practice to assume it is needed if you are not absolutely sure. The following message window will appear after the heads are parked:



The heads may become unparked if you press any key before turning off the system.

You may optionally use the standalone park program on the distribution disk called MHPARK.EXE. This program serves the same function as this menu item.

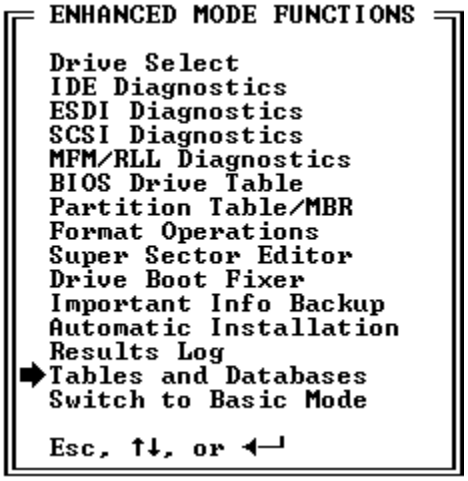
Why are heads "parked" and why do the newer drives not need it? Hard drives move the heads across the platters much like a record player moves its needle across the record. If you turn the power off when the needle is in the middle of a record, the needle will be sitting on "data" or a song. If the record player gets severely bumped the needle will jump up and come crashing back down on that "data." The song is likely to be permanently damaged.
This is exactly what happens to a hard drive. When the drive is turned off its heads (like the record players needle) will be sitting on data, the exact data that was being read before it was turned off. This data could be lost if the drive is jolted. Parking the heads on a hard drive moves them to a "safe" zone away from data- like the record player's needle arm being

returned to its holder and locked down.

The type of drive we have been talking about so far uses a type of motor to move the heads called a stepper motor. This motor moves the heads one step at a time and it takes power to move them back and forth.

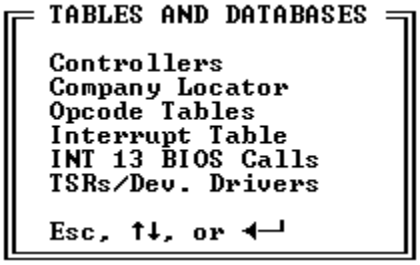
The newer drives use a type of motor called voice coil. They are called voice coil because the motor works much like a speaker. It takes power to move them one way but to move them back it takes no power at all. This is due to a spring. The electricity, with greater force than the spring, pushes the heads out. When power is removed the spring no longer has resistance against it and the heads move back in to a "safe" zone. Usually, a small magnet or pin will lock it in place, with varying degrees of strength. This is why these drives do not need to be parked. A few of the newer stepper motor drives also auto-park using the last bit of power after the computer has been turned off with its power switch. Some also store energy in a capacitor which dumps its energy on power-down, providing enough energy to park the heads.

Tables & Databases



Tables & Databases

This menu leads to reference databases commonly used when working with hard drives and controllers.



Controllers

This section contains a database of all common hard drive controller card specifications.

Company Locator

Locate company phone and address information quickly by using this database.

Opcode Tables

Assembly language opcodes with definitions can be found here. This is useful when using the Super Sector Editor's assembly code editor. These are also available when in the Assembly Editor by pressing CTRL-F1, when on the line of assembly code.

Interrupt Table

The systems interrupt table is contained in this section with the current vector address settings.

INT 13 BIOS Calls

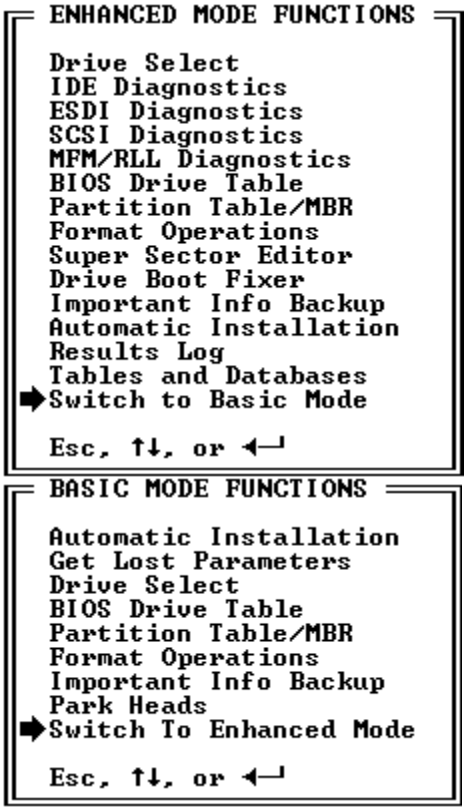
The complete list of INT 13 BIOS calls are listed in this section. These are the

routines that are contained in the BIOS for drive access and manipulation.

TSRs/Dev. Drivers

This option will display all installed TSR (Terminate and Stay Resident) programs and device drivers. It will also display all programs that are currently installed in the system's memory. This is for informational purposes only and can be used when *DrivePro* reports on start-up that TSRs and/or Device Drivers have been detected in memory that could conflict with *DrivePro* to determine what is in memory.

Switch To Enhanced/Basic Mode



Switch to Enhanced Mode
Switch to Basic Mode

Selecting this menu item will switch you to the other Main Menu mode.

Basic Mode is designed to have only the essential hard drive setup functions on it, to help a new *DrivePro* user or provide a more condensed menu for the experienced user. Enhanced Mode has all of *DrivePro's* powerful features available for the experienced user. They can be easily toggled back and forth depending on your needs and preferences.

Appendix A

DRIVE TYPE TABLES EXPLAINED
&
THE DRIVEPRO
CUSTOM DRIVE TYPE

Drive Type Tables Explained

When the original IBM AT system BIOS was designed it contained only 14 drive types in its table (type 15 was reserved). At that time there were only 14 drive geometries so the 14 table entries were sufficient. As drive technology advanced, more and more geometries required support and IBM increased its BIOS drive table entries. Eventually there were hundreds of drive geometries on the market. IBM increased their table 7 times with subsequent releases. The total number of drives supported reached 33, which did not adequately approach the hundreds on the market. Both IBM and the clone BIOS manufacturers came up with their own solutions to this problem. Most of the clone BIOS manufacturers created a BIOS that has a user-definable drive type. With this type you can enter in your own geometries. We believe that this is what type 15 in the IBM BIOS was originally intended for but was never implemented. This would have saved a lot of headaches over the years! IBM came up with their own solution, but only for some of the PS/2 line. These systems come standard with ROM on disk. When setting up the system, a 3MB partition is created on the hard drive and all boot and setup information is loaded onto the hard drive in this partition from a 2.88MB floppy diskette. These ROM files are described as IML (Initial Microcode Load) files. With this method BIOS upgrades are as simple as obtaining the revised ROM diskette from your IBM dealer.

The DrivePro Custom / BIOS Extended / Plug & Play Drive Type - How it Works

This feature allows you to have a software version of the user-definable drive CMOS type in any system. What it does is redirect the pointer that looks for the drive parameters from the BIOS drive tables to a RAM memory location where custom parameters have been defined. No user-accessible system RAM is used, only a few bytes that would have been used by the CMOS anyway.

The DrivePro Custom / BIOS Extended / Plug & Play Drive Type - What it is Used For

Basically, the Custom Drive Type overrides the systems CMOS. This is beneficial in situations in which the CMOS is either inadequate or where the drive will be moved frequently from system to system, or where the CMOS drive type may be lost. However, the Custom Drive Type requires precautions when booting from a floppy.

The most typical use for the Custom Drive Type is to compensate for a BIOS that does not support a particular drive type. Often, especially for IDE drives which do not *need* to use their physical parameters, a CMOS Drive Type can be found that sacrifices only a small percentage of the disk capacity. If no type is a satisfactory match, you can configure and use your BIOS User-Definable Drive Type if present. However, many CMOSes do not have a user-definable type. In such a situation, you must: 1) upgrade your BIOS to one that does have a user-definable type, 2) use an EPROM programmer (burner) to write a new drive table to the CMOS, or, most

inexpensively and most easily, 3) install a *DrivePro* Custom Drive Type.

The second most typical use for the Custom Drive Type besides compensating for limited BIOSes is to use it for Plug-and-Play capabilities. This term refers to the fact that the drive type is independent of the CMOS. The advantages of the drive type being independent of the CMOS are: 1) the drive can be moved to any system regardless of the BIOS and can be booted from easily, and 2) the parameters are stored on the drive so are not in danger of being lost due to CMOS battery failure. In either of these cases, the CMOS needs only to be set to anything but 0 or not installed (to get the system to look for the drive) and the Custom Drive Type will supply the system with the correct drive parameters.

MS-DOS Installation and *DrivePro*

The only drawbacks of using a Custom Drive Type are 1) the danger of being overwritten by other software and 2) the fact that the drive must be booted from in order for the code to be executed. Some actions that may overwrite the Custom Drive Type include MS-DOS 5.0's **upgrade** installation program and image-restoring from tape backups. Since image-restoring restores the MBR that was previously on the drive, you must either do a file restore, or reinstall the Custom Drive Type after the restore. To safely install MS-DOS 5.0's upgrade (upgrade version only, not the normal complete installation): 1) copy the files over from a machine where it is already installed, or 2) upgrade to diskettes and then copy the DOS files from the floppies to the drive. To install MS-DOS 6.0/6.2 's upgrade or MS-DOS 5.0 or 6.0/6.2 's complete versions, **DO NOT FOLLOW MICROSOFT'S INSTRUCTIONS. THEIR DOCUMENTATION IS INCORRECT**. Instead, use the DOS6INST program provided on your *DrivePro* diskette. Or, run MHDRIVE, press **ENTER** as many times as needed until prompted to insert a DOS system diskette, then insert the DOS SETUP Disk and hit **ENTER**. If installing MS-DOS 6.0/6.2, press return, then select the second option "Continue Setup and replace your current version of DOS". Note that for the installation of **any** software, *the only totally safe way to boot from a diskette is to run MHDRIVE and insert the desired boot diskette when asked to by MHDRIVE.*

It is safe to access floppies after booting from the hard drive since the parameters have been loaded. However, if you ever need to boot from floppy (for diagnostics, custom OS's, whatever) **YOU MUST RUN MHDRIVE** AND INSERT THE FLOPPY WHEN INSTRUCTED TO DO SO BY MHDRIVE. (You may alternatively run MHDRIVE /B from the AUTOEXEC.BAT on a boot floppy [see pg. 64])

Appendix B

A DRIVE/DOS INSTALLATION PRIMER

A Drive Installation Primer

Occassionally *DrivePro* will be unable to complete an installation of a hard drive due to an unsupported operating system, an unusual system configuration, or other circumstances. In that case, it is important to be able to continue the installation where *DrivePro* left off. Therefore, following is the sequence of events involved in a typical installation.

À Physical drive installation. Your drive must be installed and have the jumpers configured correctly. A good test for a hardware failure is to set the CMOS drive type to 1 so that the BIOS tests for the drive. If you get a "HDD FAILURE" or "HDD CONTROLLER FAILURE" or similar message, OR if *DrivePro* displays "HARDWARE ERROR" for drive 0 in the BIOS DATA window, you must stop and resolve the hardware error before the machine and *DrivePro* can correctly communicate with the drive. Check cabling, jumpers, and try swapping hardware until you determine what component is failing. If the data for drive 0 reads "PARAMETER ERROR" then either a BIOS user-definable or *DrivePro* Custom Drive Type is setup with invalid parameters, or *DrivePro* does not recognize your BIOS.

Á Selecting the drive type. A very useful feature of *DrivePro* is its ability to recommend the best manner in which to install a drive. However, if *DrivePro* does not recognize your BIOS, it will not be able to recommend a drive type. If your BIOS is not recognized, use the setup utility (or diskette) included with your system to pick the best drive type yourself, or set the CMOS drive type to 1 and install a *DrivePro* Custom Drive Type. Then reboot. Remember: Whenever you have a Custom Drive Type installed, you must either boot from the hard drive or run MHDRIVE so that the correct parameters are loaded from the Custom Drive Type. If you boot from the hard drive and do not see the DRIVEPRO CUSTOM DRIVE TYPE INSTALLED message, either there is not one installed, or the Custom Drive Type has been partially or fully overwritten, or the drive is failing before it loads the Custom Drive Type.

Â Low-level format. Only for MFM/RLL (ST506/412), ESDI and SCSI drives. If the controller has low-leveling capabilities, you should use it and no other software. This applies to some MFM/RLL (ST506/412) controllers, most ESDI controllers, and all SCSI controller s (some SCSI controllers have an included or optional format utility on a floppy diskette that is used). IDE drives are always shipped already low-level formatted from the factory, and rarely require low-levelling. **Low-level IDE drives at your own risk!**

Ã Partitioning. After rebooting so that the correct parameters show up for drive 0 in the BIOS DATA window (this is a must!) partition your drive to your specifications with *DrivePro*, making sure each drive has one Primary Partition and any remaining space in Logical Drives in an Extended Partition, with the Primary Partition on the first drive marked bootable. Note that at this point you can actually partition with DOS FDISK, but *DrivePro* is by far the more powerful and easier

method.

Ä High-level Formatting. After partitioning you must high level format each partition. DrivePro has two high-level formats; the "quick" when you know the drive has no bad clusters and the "regular" when you would like to invest the time required to check for bad clusters.

Appendix C

SOLUTIONS TO
COMMON PROBLEMS

No matching type in BIOS for IDE drive.

If the BIOS has a user-definable drive type and the total cylinders of the drive are under 1024 (and sectors 63 or less and heads 16 or less), use the physical parameters. It is preferable to use the physical parameters because they minutely enhance drive access speed since a translation is not required, and are probably the first parameters that someone trying to recover the drive will guess to be parameters used to install the drive.

If the BIOS does not have a user-definable drive type or the cylinders of the drive are over 1024 (or other DOS limitations are exceeded), you may select the closest fitting table entry which does not use the entire disk, or create a *DrivePro* Custom Drive Type. Note that you must not set the number of cylinders greater than 1024 with any version of DOS, unless you use the *DrivePro* 512MB-2GB Drive Type or a replacement block device driver for DOS that supports over 1024 cylinders. To exceed these limitations without one of these installed may result in loss of all data on the drive! In addition, you cannot use more than 63 sectors or 16 heads, unless your controller has an on-board BIOS that allows for a greater number than that, because of AT-standard BIOS limitations. You must use a translated set of parameters that equal the same capacity without exceeding these limitations. To use the best matching BIOS entry, choose the entry that is equal to or less than, but never more than, the drive's capacity in megabytes.

Other options are hardware ones, including upgrading your system BIOS, a controller with the appropriate BIOS, and purchasing an EPROM programmer (burner) to physically alter the BIOS.

IDE drive was moved to another system or computer lost its CMOS setup. Now the IDE drive won't work properly. (Need to determine drive's original parameters.)

The parameters for the IDE drive must be set identically as when the computer was initially set up. Note that different parameters can yield the same capacity, so using just any parameters that yield the correct capacity does not necessarily mean you are using the correct parameters. To determine from the drive's partition table the original parameters used to setup the drive, use *DrivePro*'s **Get Lost Parameters** feature, or the stand-alone MHDRIVE utility. Upon determining the correct parameters, either 1) match them to a drive type in the CMOS, or configure the CMOS user-definable. Alternatively, you may run *DrivePro* /MBR to set up a *DrivePro* Custom Drive Type with the correct parameters without erasing the partition table. The drive may then be used in any machine as long as the CMOS is set to a value besides 0 or not installed.

CMOS shows drives capacity as less than it should be.

CMOS and FDISK calculate drive capacity by the formula HEADS x CYLINDERS x SECTORS / 2048. DIR, CHKDSK, and *DrivePro* use the formula HEADS x CYLINDERS x SECTORS x 512. CxHxSx512 yields decimal number of bytes, which when divided by 1,000,000 gives you decimal megabytes. CxHxS/2048 yields binary megabytes.

There are 1,000,000 bytes in each decimal megabyte. There are 1,048,576 bytes in a binary megabyte. Therefore, if the CMOS's value for the capacity of the drive is less by ~5% than some other indicator, the difference is due to different calculation methods.

How do I remove the *DrivePro* Custom Drive Type and just get back to a standard set-up?

Type '**DRIVEPRO /MBR**' and select 'Other DOS Regular MBR'. You must have a copy of FDISK on your *DrivePro* diskette or a disk containing FDISK handy to insert when prompted for it. Or, with MS DOS 5.0 or above, type '**FDISK /MBR**' Both of these operations will overwrite the MBR with a standard DOS MBR without destroying or otherwise altering the partition table.

"NO ACTIVE REBOOT" message when booting from hard drive

This message appears when there is no partition marked active (or BOOTABLE- 'YES') and there is a *DrivePro* MBR or Custom Drive Type installed (If it is not a *DrivePro* installation and it is not an older true IBM system, you will get the message "NO ROM BASIC" instead and the computer will lock up. On a older true IBM system, you will actually get a BASIC prompt and be in ROM BASIC). Select **Partition Table/MBR**, **Partition Table Editor** and press spacebar to toggle the BOOTABLE field from NO to YES. Save and reboot for the changes to take effect.

"C: Drive Failure. F1 to continue"

If after pressing F1 your drive goes on to boot, the parameters set in CMOS are not supported by the drive (for some Conner, Maxtor, and a few other older drive models not supporting "Universal Translation") or are too large for the drive (on a Compaq system). If a *DrivePro* Custom Drive Type is installed, merely set the CMOS for type 1. If not, check the jumper settings and manufacturer's recommended parameters for the hard drive (using the **IDE AutoSelect** feature in *DrivePro).* Correct the CMOS setup and try again. If no matching CMOS drive type can be found and the BIOS does not have a user-definable drive type, use '**DRIVEPRO /MBR**' and select the '*DrivePro* BIOS Extended MBR' to install a *DrivePro* Custom Drive Type on the drive.

"Hard Drive Controller Failure" message

Most likely your controller is not functioning. Try switching controllers, cables, and drives, and make sure that the cables are attached correctly, that the jumpers are set correctly, and try different bus slot locations. Also check the section above on "C: Drive Failure, Press F1 to Continue".

"Error reading Fixed Disk" or similar message

If this message appears when booting from the hard drive, but you can boot from a floppy and then access the hard drive, it is most likely that the parameters used to configure the drive slightly exceeded the drive's capacity. Use slightly lower parameters, then re-partition and re-format the drive. Reboot for the changes to take effect.

My BIOS has no provisions for displaying or editing the drive type table during boot-up.

Use **BIOS Drive Table, View BIOS Table** to view, configure user-definable drive types, and set drive types.

What drive type is used for SCSI drives?

In most installations set the type to "0" or "none," and the SCSI controller and drive will take it from there. Also, its a good idea not to low-level the drive unless the controller came with its own firmware or software to do so. Use **Controller BIOS** to access the firmware on the card if it exists.

What drive type is used for ESDI drives?

In most installations set the type to "1" and the ESDI controller and drive will take it from there. Most of these drives need to be setup using **Controller BIOS**.

The error message "1790" appears at boot-up.

A 1790 error is normal for an unformatted drive. Simply do the low-level if this is a new installation. If this is an existing installation then the drive has probably "lost" its format and cannot be salvaged by normal means. If the data has been backed-up, you can try redoing the low-level format, followed by the normal partitioning and high-level (DOS) format, finally restoring your backup to the drive. If this works, the drive may still be close to total failure, so it is wise to not place any important information on it.

Drive won't boot after FDISK and DOS format, but Norton Disk Doctor corrects this. Why?

This is a problem that usually arises with some IDE drives. DOS appears to have some problems marking the first partition of a drive "active." The first partition must be marked "active" for it to boot. FDISK is supposed to do this for you on the first partition but it sometimes fails to do so. The most problematic DOS version is 3.3. Norton Disk Doctor fixes this because this simply marks the first partition as "active". You can also fix this by running FDISK and setting the first partition to "active", or by using *DrivePro*'s Partition Table/MBR editor and marking the first partition BOOTABLE -"YES."

ESDI drive shows less capacity than it actually has when in FDISK.

If using a Western Digital ESDI controller or compatible: when in debug (G=xxxx:5) you must select menu item 7, **Change drive type and exit**. Then use the +/- keys to select a set of parameters that is equal to your drives' capacity. These parameters need not match the actual physical parameters of the drive. These are the parameter's logical translation that DOS will refer to. If you are not using a Western Digital controller then refer to your controller manual for information on the above. Also, see note earlier in this manual concerning different calculations of disk capacity.

Unable to delete a Novell or other non-DOS partition or unable to DOS format such a partition.

This is particularly troublesome with IDE and SCSI drives due to the fact that they usually cannot be low-level formatted. Select **Format Operations** from the main menu, then select **Erase First Ten Cylinders**. *DrivePro*'s **Partition/MBR Editor** will also

allow you to edit or delete a non-DOS partition table entry, so you can also use that option or the **Super Sector Editor** to remove the Novell or non-DOS partition.

"Hard drive controller failure" after installing a second IDE drive.

You should carefully check and reconfirm the jumper settings and check or switch out the cabling, and try changing such jumpers as I/O Channel Ready and the Drive Slave Present jumpers (consult the drive manufacturer's Technical Support for additional help with this), and can also try switching the two hard drives around, setting the jumpers on the Master drive to be the Slave, and on the Slave drive to be the Master. Also, due to the proprietary beginnings of the IDE interface, IDE drives are very frequently incompatible with IDE drives from other manufacturers. This is especially true of some older Conner and Seagate drives. The solution to this problem when adding a second IDE drive is to use drives of the same make, or alternatively, using the included utility ADDADRV to allow the second drive to be used on a second controller card.

Ran FDISK or Partition Table/MBR on an IDE drive several times, but it won't save the partition tables.

Translation values into the CMOS setup that are greater than the capacity of the IDE drive or parameters that the IDE drive does not support can cause this problem. Try another set of heads, cylinders and sectors. Be sure that you don't go over the drive's total capacity in megabytes. You should also recheck jumper settings and check or switch out the cabling.

Two IDE drives will not work together even though I have the Master/Slave settings set correctly.

Check or switch out the cabling. Recheck the I/O Channel Ready and Drive Slave Present jumpers (contact the drive or controller manufacturer's Technical Support for more information). In some cases both IDE drives must be of the same make in order to work together. If the two IDE drives work independently of each other, try switching the drives in terms of Master and Slave. If the two drives work independently but not together then the only solutions include to use two drives of the same make, or to use the included utility ADDADRV and a second controller.

Installed an IDE hard drive; now the floppy drives don't work.

If you have installed an IDE drive and now the floppy drives don't work or even light up during boot, then you have hooked the IDE cables up backwards.

Most IDE drives do not show the location of pin-1. If you can't locate pin-1, then a good rule-of-thumb is to assume pin-1 is the closest pin to the DC power connector.

Drive doesn't spin.

1) Check that drive has power supplied to it.

2) Some of the older hard drives have a problem with the media becoming sticky after years of use. The heads get stuck to the platter and prevent it from spinning. To get them spinning again, make sure you touch a good ground (like bare metal on the case of your computer if it is plugged into a good 3-prong outlet), remove the drive from the case, turn the drive over and slightly turn the spindle (be careful!) to unstick it. It should then work long enough to make a backup. Get your data off

quickly and replace the drive!

Need to use only the floppy section of an AT hard/floppy controller card but no jumper is provided to disable. How do I disable the hard drive from the controller?

In most cases you can disable the hard drive section of the controller by setting the hard drive port address to the secondary setting (170h-177h) . You must also move the controller to an 8-bit slot.

Long boot time after installing a drive into an AT.

Some AT's may continue to retry booting from the newly installed disk several times even though it has not even been formatted. You will have to wait until the computer is done with the retry process before the system will boot to the floppy drive. Then you can proceed to do the hard drive setup.

No response from MFM/RLL (ST506/412) hard drive after installing.

Change the cables. Note: Do not use a floppy cable on a hard drive. Check to make sure that the drive has power and is spinning. Recheck the jumper settings also.

"Drive active" LED always stays lit.

XT or PC installations: Check cable connections. Try to reverse or replace the cables. If this doesn't correct the problem, then the drive is probably bad. AT installation: This is not a problem. On AT systems the drive is always selected and therefore the LED is always lit. Also, you can try turning off all power to the system, waiting 10-15 seconds, and then turning the system back on.

"Nothing done exit" appears when doing a low-level format with a Western Digital Controller.

You didn't press the <Y> key when prompted. Restart the low-level format.

"Hard disk drive not ready" or "01" error code when booting.

Possibly one of two problems: 1) System BIOS drive table doesn't support the controller and drive or 2) the power supply is overloaded- replace it with one of higher wattage or remove or replace some of the cards with lower-power ones to save power.

Error code "80" while doing a low-level format.

Drive select jumper is incorrectly set on the hard drive or the cables are on backwards. Possibly could be bad cables, no power to drive, or simply a bad drive.

Error codes "20" or "40" while doing a low-level format.

Check cable connections. Try to reverse or replace the cables. Could also be a bad drive.

After DOS format, message "Insert disk and press ENTER" appears.

Motherboard switches are set for the incorrect number of floppy drives (PC or XT). Check to see if RAM disk drivers are present, if so then set the motherboard switches to include the RAM drives.

"Bad track 0" using DOS 3.1

Make sure the line, "BUFFERS=99" is in the CONFIG.SYS file.

"Bad track 0" using DOS 2.1

DOS 2.1 cannot support a drive that has bad tracks above 16.7MB. Upgrade the system's DOS to 3.1 or better.

Getting intermittent operation errors or read/write errors at random.

Ensure that the termination resistors on the hard drive(s) are properly placed. Also ensure that the power supply can support the added hard drive and voltages from the power supply are within range.

Getting a "recal error" or a "no drive attached" error.

Check the drive select jumpers for proper installation. On MFM/RLL (ST506/412) or ESDI drives ensure that the data cables are attached to the proper drives. Also ensure that pin-1 on the controller is attached to pin-1 on the drive.

Appendix D

SUPPLEMENTAL UTILITIES

Supplemental Utilities -

A Quick Reference

The following has been provided as a quick reference to each of the utility programs included on the *DrivePro* diskette.

MHDRIVE [/B /?]

This utility **must** be used if you have placed a *DrivePro* Custom Drive Type on the hard drive, and want to boot from a floppy disk, and still have access to the hard drive. If you do not use it, you risk data loss on your hard drive due to the 1024 cylinder limitation of DOS.

Optional command line parameters are contained in the []. They are:

/B Instructs the program to run in batch file mode. In this mode, MHDRIVE will only reset the drive parameters if it is required. After resetting the parameters, MHDRIVE will reload DOS, all without user-intervention. You can put MHDRIVE on a bootable floppy disk and as the first line in the AUTOEXEC.BAT file put "**MHDRIVE /B**". A diskette that runs MHDRIVE is the only type of floppy you should boot from when using a system with a hard-drive that has a *DrivePro* Custom (BIOS Extended) Drive Type. Note that MHDRIVE does not run if it is not needed, so **you can safely boot from this floppy on any machine, even one that does not use a *DrivePro* Custom Drive Type.**

/? Displays all command line options.

The proper use of this program is:

MHSAVE [/Dx /N /Bfilename /S /0 /1 /C /F /U /?]

Optional command line parameters are contained in the []. They are:

/Dx Backup important info to drive x (A:,B:,C:,etc.)

The default is A:. You should always save to a floppy or other drive, not the hard drive. You will need a formatted disk in the A: or B: drive after the system boots. MHSAVE will prompt you when it is time to insert the disk (unless the **/N** option is used).

/N Disable prompt for floppy disk insertion

MHSAVE will not prompt you for the save diskette. This is useful for systems that have a CMOS setting to not boot from the A: drive. With these systems you may just leave a formatted disk in the drive and every time you boot the file will be saved on the diskette.

/Bfilename Specify filename for backup file

/S Only show space required for important info backup(s)

/0 Only save physical drive 0 important info

/1 Only save physical drive 1 important info

/C Only save CMOS information

/F DOS format the floppy disk first

/U Don't use disk parameter tables, only disk BIOS

/? Displays all available parameters

MHPARK

Parks the drive heads for protection of data during shipping.

MHSYS [SourceDrive:] DestinationDrive: [/?]

Purpose: MHSYS is used to copy the system files from one drive to another. The *DrivePro* installation normally performs this task, so this utility is normally only used to make the *DrivePro* diskette bootable upon arrival, assuming that SYS.COM is not available. MHSYS will transfer from one floppy to another in one pass instead of the multiple passes used by SYS, and can transfer to the same floppy drive. It also does not affect the DOS Boot Record like the DOS SYS utility and so is also useful in drive recovery. However, MHSYS does not reposition existing files so it can only be used on blank floppies or floppies which have been formatted with the /B option (leave space for system files). MHSYS will only make your *DrivePro* diskette bootable if you use it before writing any additional files to it.

DestinationDrive: The drive to which to copy the DOS system. The colon is necessary and must be used. This parameter must be supplied.

Optional command line parameters are contained in the []. They are:

SourceDrive: The drive from which to load the DOS system. The colon is necessary and must be used. If no drive is specified then the system is loaded from the current drive. Typical uses are MHSYS A: A: or MHSYS C: A:

/? Displays all command line options.

Purpose: Add-A-Drive is a CONFIG.SYS block device driver that allows use of the Secondary Port Address while in the DOS operating system. This allows two hard-drive controllers to be used in the same system. This means that you can mix interface types, add a third drive, use drives that conflict with each other, or any combination of these!

A common use of Add-A-Drive is to install the new IDE drive, and keep the MFM or RLL drive by putting it on the second card. You can eventually insert a second IDE and have 3 drives running in the system!

Other common uses include using 2 cards for 2 IDE drives from different manufacturers which are incompatible, or mixing two ESDI cards that are using different sectoring.

Use: Add the line: **Device = ADDADRV.HDD** to your CONFIG.SYS file.

Requirements:

1. System must have a 286 CPU or higher.
2. DOS revision must be V3.0 or higher.
3. A second Controller or Host Adapter which must have the option to set a Secondary Port Address to 170h, to disable IRQ14, and to completely disable or set to a secondary address the floppy drive. (Seagate ST-07A and ST-08A AT/IDE Host Adapters can do this, as example.)

Almost all problems experienced with AddADrive are due to misconfigured hardware. If you are experiencing problems, you should consult the **ADDADRV.DOC** file on the *DrivePro* diskette. Only if you are unable to resolve your problems by meticulously following the instructions in the **DOC** file on the diskette should you consult technical support..

Appendix E

BIOS POST CODES

The following table lists the codes returned by the POST (Power On Self Test) when an IBM system detects an error. Only the codes pertaining to hard and floppy disks are contained here.

POST and Diagnostics Error Codes:

01x Undetermined problem errors
6xx Floppy drive/adapter errors
601 Floppy drive/adapter POST failure
602 Drive test failure;
disk boot record is not valid
606 Disk change line function failure;
drive error
607 Disk is write protected; drive error
608 Bad command; drive error
610 Disk initialization failure; track 0 bad
611 Time-out; drive error
612 Bad Controller chip
613 Bad Direct Memory Access; drive error
614 Bad Direct Memory Access;
boundary overrun
615 Bad index timing; drive error
616 Drive speed error
621 Bad seek; drive error
622 Bad Cyclic Redundancy Check;
drive error
623 Record not found; drive error
624 Bad address mark; drive error
625 Bad Controller chip; seek error
626 Disk data compare error
17xx Fixed disk errors
1701 Fixed disk POST error
1702 Fixed disk adapter error
1703 Fixed disk drive error
1704 Fixed disk adapter or drive error
1780 Fixed disk 0 failure

1781 Fixed disk 1 failure
1782 Fixed disk controller failure
1790 Fixed disk 0 error
1791 Fixed disk 1 error
7306 Disk change line function failure; drive error
7307 Disk is write protected; drive error
7308 Bad command; drive error
7310 Disk initialization failure; track 0 bad
7311 Time-out; drive error
7312 Bad Controller chip
7313 Bad Direct Memory Access;
drive error
7314 Bad Direct Memory Access; boundary overrun

7315	Bad index timing; drive error	
7316	Drive speed error	
7321	Bad seek; drive error	
7322	Bad Cyclic Redundancy Check; drive error	
7323	Record not found; drive error	
7324	Bad address mark; drive error	
7325	Bad Controller chip; seek error	
104xx	PS/2 ESDI Fixed disk errors	
10480	PS/2 ESDI Fixed disk 0 failure	
10481	PS/2 ESSI Fixed disk 1 failure	
10482	PS/2 ESDI Fixed disk controller	failure
10483	PS/2 ESDI Fixed disk controller	failure
10490	PS/2 ESDI Fixed disk 0 error	
10491	PS/2 ESDI Fixed disk 1 error.	

Appendix F

INTERFACE TYPES

Technical Information On The Interface Types

This appendix contains background and technical information on each of the interface types. Reading this section is not necessary for the use of the software, but it is recommended.

Enhanced Small Device Interface (ESDI) Overview

The hard drive manufacturer Maxtor was instrumental in bringing this interface to market in 1983. It is similar to the ST506/412 interface in that it uses the same cables and a lot of the same pin signals, but this is where the similarity ends. ESDI can transfer data at up to 24Mb per second, although most transfer at 10Mbs. In the technical explanations that follow, all comparisons are done to ST506/412 as this was ESDI's predecessor. Please see the Glossary for more background information on this interface.

ESDI Technical Features

Most ESDI drives have drive select jumpers that will allow drive unit numbers between 0 and 6, although only types 0 and 1 are valid with most controllers (you will notice that this is similar to SCSI drives). There are usually only three jumpers that define the 7 drive select options. These three jumpers are a binary representation of the seven possible selections.

The circuitry that separates the data being read from the drives read/write heads, the clock-data separator, is contained on the hard drive itself and not on the controller card. Previously, with ST506/412 drives this circuitry was incorporated into the controller card and not on the drive. With ESDI, drive manufacturers are able to design a clock-data separator that is perfectly matched to their hard drive. Throughput is a lot quicker since data is recovered from the heads before it is sent down the cable to the interface and thus noise margins are better.

Data on these drives is stored in much higher densities than on ST506/412 drives. The sectors per track (SPT) are usually 33, 34, 35, 48 or higher and may even have variable sectors. That is, the outer tracks may have more SPT than the inner tracks.

As with IDE and SCSI drives, the ESDI drive is somewhat intelligent. It will accept commands (or 'opcodes') to perform various functions on its own and will report back when done.

When the computer accesses the hard drive it does so through an interrupt. INT13 is the hard drive interrupt. The programs necessary to access the hard drive are located in the computers BIOS and are accessed through this interrupt (a list of these INT routines are contained in **Tables & Databases**). ESDI controllers fall into two basic categories regarding this interrupt:

Autotyping ESDI controllers

These controllers actually replace the computers own BIOS INT13 programs. INT13 is redirected to the controller's own set of programs contained in its firmware. This firmware knows exactly how to handle an ESDI drive. You must set the computers drive type (in CMOS) to 1 so that this firmware can locate it. Type 1 is normally a 10MB drive, but not in this case. Most controllers issue an ESDI General Configuration command to the drive to get its actual parameters. *DrivePro* issues a Drive Inquiry command to the controller to get the controller data and drive's specifications.

WD1005 and Compatible Controllers:

These controllers are accessed in the same manner as ST506/412 controllers. There is no INT13 replacement and the drive type (in CMOS) must be set to the drives actual heads, cylinders and sectors. This type of controller is out dated and very rare due to the fact that it may only be used with drives that fall within the limits of the computers BIOS and DOS limitations (no more than 1,024 cylinders, 16 heads, and 63 sectors, giving 512MB).

DrivePro is compatible with both types of controllers.

ESDI Quirks, Hazards And Pitfalls

Many ESDI controllers are in use that do not a BIOS. These controllers neither translate nor lowlevel format drives. For installation purposes, these drives should be treated as ST506/412 drives, in that they need to be lowlevelled with software (not firmware) and that they are limited to 1,024 cylinders unless you some sort of support besides DOS, such as *DrivePro*.

Some ESDI controllers have firmware placed on them that is only compatible with a specific ESDI hard drive. Also, some controllers are rated at a certain MHz speed (10MHz, 15MHz, or 24Mhz). In all cases, always ensure that the controller is compatible with the drive being installed.

Integrated Drive Electronics (IDE) Overview

This interface originated in 1988 when a number of peripheral suppliers got together to design an interface that could be built right into the new low-cost personal computers. See the Glossary for complete background information on this interface.

IDE Technical Features

These drives have the controller built right onto the drive itself. Only a small paddle card or connection built onto the motherboard is needed to interface computer to drive. And since the drive and controller are designed, manufactured, and tested by the same company there are *no* compatibility problems between the drive, controller and computer.

IDE drives usually use translation schemes to get around DOS limitations so that they may be used in a wide range of systems that may not support their actual physical parameters. The physical parameters are the actual heads, cylinders, and sectors per track of the drive. The logical or translation specifications are specifications that fall within the rules of DOS and the systems BIOS. When DOS issues a seek command to the IDE drive it does so with the logical parameters. The drive then does a mathematical translation and goes to the actual location. This makes low-level formatting these drives very tricky. ***Therefore, DrivePro will not format every IDE drive successfully!***

The throughput of these drives is not as outstanding as ESDI or SCSI. Placing these disks in a high-load environment (such as a Novell network) is not recommended. Instead, use ESDI or SCSI drives. But performance always improves as technology advances.

IDE drives use a drive-select scheme called 'master' and 'slave'. Ensure that your drive is configured properly. Only two IDE drives may be attached to one cable (one drive in XT installations).

Data on these drives is stored in much higher densities than on ST506/412 drives. The sectors per track (SPT) are usually 33, 34, 35, 48 or higher and may even have variable sectors. That is, outer tracks may have more SPT than the inner tracks.

As with SCSI and ESDI, the IDE drive is somewhat intelligent. It will accept

commands (or 'opcodes') to perform various functions on its own and will report back when done.

When the computer accesses the IDE hard drive it does so through an interrupt. INT13 is the hard drive interrupt. The programs necessary to access the hard drive are located in the computers BIOS and are accessed through this interrupt (a list of these INT routines are contained in **Tables & Databases**).

IDE Quirks, Hazards And Pitfalls

It has been rumored that IDE drives can be 'destroyed' by low-leveling them. The only drives that we've had reports of being permanently ruined in this manner are older Microscience drives (but even this has not been confirmed). If an IDE drive *can* be successfully lowlevelled with software, *DrivePro* can probably do it. Assume that all data will be lost, even when in the non-destructive mode. You may also lose the bad track table. This table is placed on the drive by the manufacturer and is never in the same place. We suggest you perform a thorough media analysis before using the drive.

IDE drives seem to have a small problem with some older computer BIOS revisions (pre-1991). The IDE drive will pause for a few seconds to a few minutes when data is being accessed. Simply wait for the drive to finish its read/write cycle. This can be permanently remedied with a BIOS upgrade.

BIOSes older than 1988 may not recognize the IDE interface. In this case, the machines will boot and then error with a message instructing the user to press F1 to continue. When the user presses F1, the machine will proceed to boot from the IDE drive with any other problems until the next reboot. Having to press F1 cannot be done away with without a BIOS upgrade.

Not all IDE drives are compatible. We suggest that you use the same make, and, if possible, the same model for both drives on a two drive chain. Some models may used together with either drive as master. Some models can be mixed together but one may need to be the master. The Seagate ST 157A, for example, will only work with another ST 157A (on rare occasions you can set this drive up with another manufacturer's IDE drive, but the ST 157A usually must be the master drive).

Small Computer System Interface (SCSI) Overview

This interface originated in the late 70s' by Shugart Associates. It evolved from SASI (Shugart Associates System Interface). See the Glossary for the complete background on this interface.

SCSI Technical Features

The SCSI interface is much more than just for controlling hard drives. Up to 8 devices (including hard drives) can be chained to a SCSI interface. The interface itself is considered one of these devices, which leaves room for 7 other devices, each of which could have 8 subdevices (providing your SCSI devices and software support this configuration- 56 devices!). The drive is an intelligent SCSI device. Commands are sent down the SCSI bus to the drive and it performs the task.

The system does not access the SCSI hard drive and interface in the normal manner. The drive type (in CMOS) is set to 0 or 'No drives'. The firmware or installed device driver handle the communication tasks.

SCSI drives use a drive select scheme called Target and Logical Unit Number (or LUN). Each target, from 0 to 7, has up to 8 LUNs. Most SCSI controllers require the first drive to be set to LUN 0 and the second to be LUN 1 on Target 0, or the first drive to be target 0, LUN 0, and the second to be target 1, LUN 0. The controller must have a target number too as it is a SCSI device. Most controllers use Target 7, which has the

highest priority. Ensure that your drive is configured properly.

Data on these drives is stored in much higher densities than on ST506/412 drives. The sectors per track (SPT) are usually 33, 34, 35, 48 or higher and may even have variable sectors. That is, outer tracks may have more SPT than the inner tracks.

As with IDE and ESDI, the SCSI drive is intelligent. It will accept commands (or 'opcodes') to perform various functions on its own and will report back when done.

When the computer accesses the hard drive it does so through an interrupt. INT13 is the hard drive interrupt. The programs necessary to access the hard drive are located in the computers BIOS and are usually accessed through this interrupt. Most times, the firmware on the SCSI controller or the loaded SCSI device driver redirect INT13 to themselves.

Adaptec ASPI

ASPI (Advanced Storage Peripheral Interface) was developed by Adaptec to help standardize the software interface to SCSI controllers. If you are using an Adaptec SCSI controller you may need to load the ASPI drivers which are available from Adaptec.

SCSI Quirks, Hazards And Pitfalls

Some SCSI controllers require you to use a device driver in the CONFIG.SYS file. If you are experiencing difficulties check with the controller manufacturer to see if a driver is required.

The SCSI controller is usually set to device 7. Typically, set the first SCSI drive to device number 0. Most SCSI controllers look for a device 0 to install as drive C:.

ST506/412 Interface Overview

This interface was developed by Seagate Technologies (originally Shugart) in 1980 solely for use with their ST-506 hard drive (a 5 MB drive). It was latter revised in 1981 with a feature called "buffered seek" for their ST-412 drive (a 10MB drive). "Buffered seek" allows the controller to save seek steps and then do the full seek in the optimum fashion, decreasing seek time. This interface transfers data at around 5Mb per second which is slow to today's standards.

ST506/412 Technical Features

Most ST506/412 drives have drive select jumpers that will allow drive unit numbers between 1 and 4, although only types 1 and 2 are valid with most controllers.

Data on these drives is stored by using one of two encoding schemes, MFM and RLL. Unlike the other interface types, the controller must be purchased with the correct encoding schemes to match the drive.

These drives are not very intelligent. There are no opcodes or special features as with the other interface types.

When the computer accesses the hard drive it does so through an interrupt. INT13 is the hard drive interrupt. The programs necessary to access the hard drive are located in the computer's BIOS and are accessed through this interrupt (a list of these INT routines are contained in **Tables & Databases**). ST506/412 drives are accessed through the standard BIOS.

ST506/412 Quirks, Hazards And Pitfalls

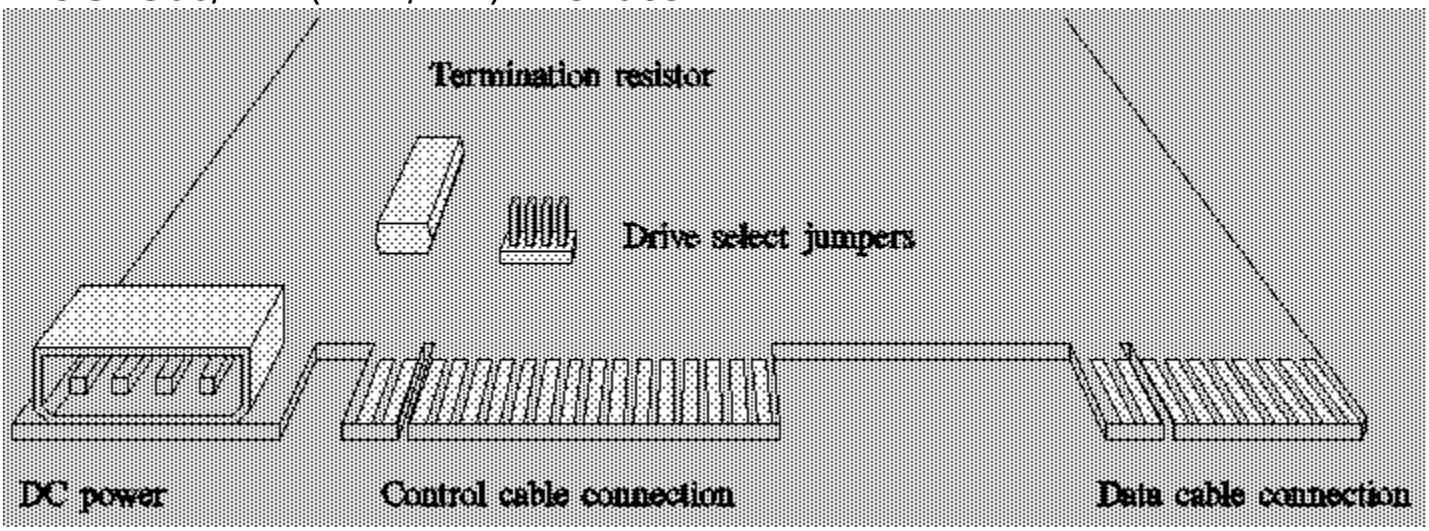
Being that it is one of the first types of interfaces ever designed for PC to hard drive installations it is not too intelligent or efficient. This is the only type of interface that does not support the Drive Inquiry feature which allows *DrivePro* to automatically identify the drives parameters.

The ST506/412 drives have no sector-sparing (relocation of defects on the disk surface) nor automatic bad-sector mapping. Due to the lack of these features it is very important that you enter the defects into the bad-track table before the drive is low-level formatted.

Appendix G

Physical Drive Installation

The ST-506/412 (MFM/RLL) Interface



This layout will vary from drive to drive, but the basic connections will always be present.

Terminating Resistor Pack

This is a socketed resistor pack that is usually yellow, but can sometimes be black or blue. It can be found on the underside of the drive or near the drive control and data cables. Remove these only according to the instructions for the type of control cable used (twisted or non-twisted).

Drive Select Jumpers

There will be 2-4 drive select jumpers depending on the make and model of the hard drive. Next to these will be several other jumpers. The position to the far left or far right will be drive select 0 (sometimes the numbering starts at 1). Usually only four of these jumpers are actually used for the drive select function. The others are reserved for other drive parameters.

Power Cable

Each drive requires its own power cable. The cable is keyed so that it can be inserted in only one orientation, the correct one.

Finding Pin 1 On All Cables

Pin 1 on the control cable can be located by one of three possible markings, a triangle stamped on the connector close to pin 1, a colored stripe on wire #1 (usually red or blue) or a keying tab inside of the connector that will not allow the cable to be inserted incorrectly.

Drive Data Cable

This is the small 20-pin ribbon cable. One data cable is required for each hard drive.

Drive Control Cables

This is the large 34-pin ribbon cable. One control cable is required for every two hard drives. There are several types of drive control cables. The type of cable used determines where the drive select jumpers and termination resistors are to be placed on the hard drives. These cables and the location of jumpers and termination resistors are described below.

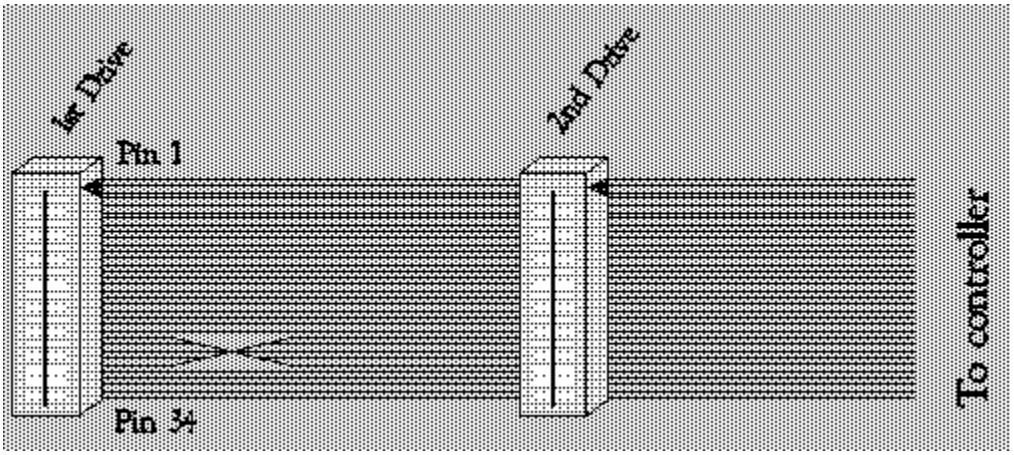
Single Drive Control Cable

General Use: This cable is used in systems with only one hard drive.

Hard Drive Jumper Settings: Place the drive select jumper in the first position.

Termination Resistor: Leave the termination resistor on the drive.

Dual Drive Control Cable With Twist



General Use: This cable is used in systems with one or two hard drives.

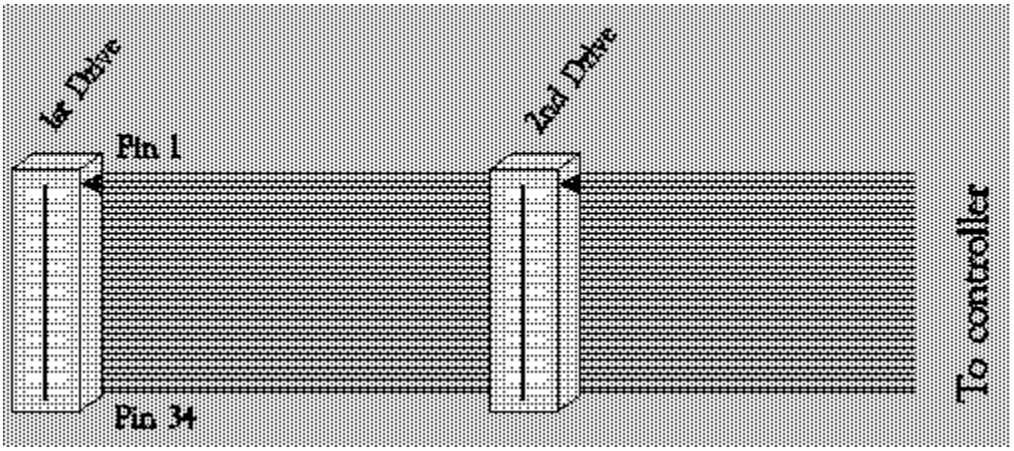
Hard Drive Jumper Settings And Termination Resistor: For the first hard drive (usually drive C:) or only hard drive, set the drive select jumper in the second position. Leave termination resistor in place. For the second hard disk (usually drive D:) set the drive select jumper in the second position and remove the termination resistor.

In this case, the jumpers are set to the second position on both of the drives because the cables twist reverses the drive select pins.

Hard Drive Connection: The middle connector is for the second hard drive (usually drive D:). The last connector is for the first hard drive (usually drive C:) or the only

hard drive, if there is only one drive connected to the controller card.

Dual Drive Control Cable With No Twist

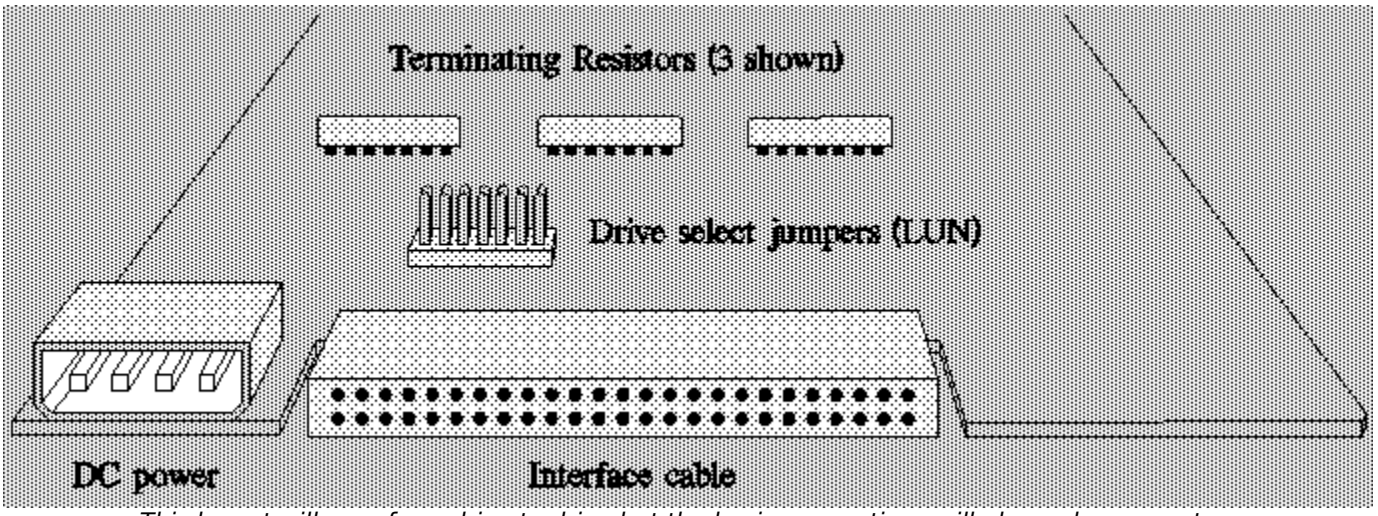


General Use: This cable is used in systems with one or two hard drives.

Hard Drive Jumper Settings And Termination Resistor: For the first hard drive (usually drive C:) or the only hard drive, set the drive select jumper in the first position and leave the termination resistor in place. For the second hard disk (usually drive D:) set the drive select jumper in the second position and remove the termination resistor.

Hard Drive Connection: The middle connector is for the second hard drive (usually drive D:). The last connector is for the first hard drive (usually drive C:) or the only hard drive, if there is only one drive connected to the controller card.

The SCSI Interface



This layout will vary from drive to drive but the basic connections will always be present.

Terminating Resistor Packs

These are socketed resistor packs that are usually yellow and sometimes black or blue. On SCSI drives there are several of these on each hard drive. They can be found on the underside of the drive or near the drive header cable. Remove all resistors for drives (or devices) connected in the middle of the daisy chain. Leave

terminators in place for the first and last devices in the physical chain. SCSI controllers usually have terminators on them to terminate the start of the chain. Therefore, in a typical installation (external drives only), the drive placed at the physical end of the cable will have installation, and the drive physically between the previously mentioned drive and the controller will not have termination. SCSI controller cards can usually control up to seven SCSI hard drives or other SCSI devices.

Drive Select Jumpers (LUN)

Drive selection on SCSI devices are sometimes called Logical Unit Numbers (LUN). There will be 3-12 jumpers. The position to the far left or right will be drive select 0 (sometimes the numbering starts at 1). Some drives use three jumpers that determine the binary coded number of the drive (jumper 1= 2_0, jumper 2=2_1, jumper 3=2_2). On the opposite side will usually be the parity jumper which may or may not be needed depending on the controller that the drive is attached to.

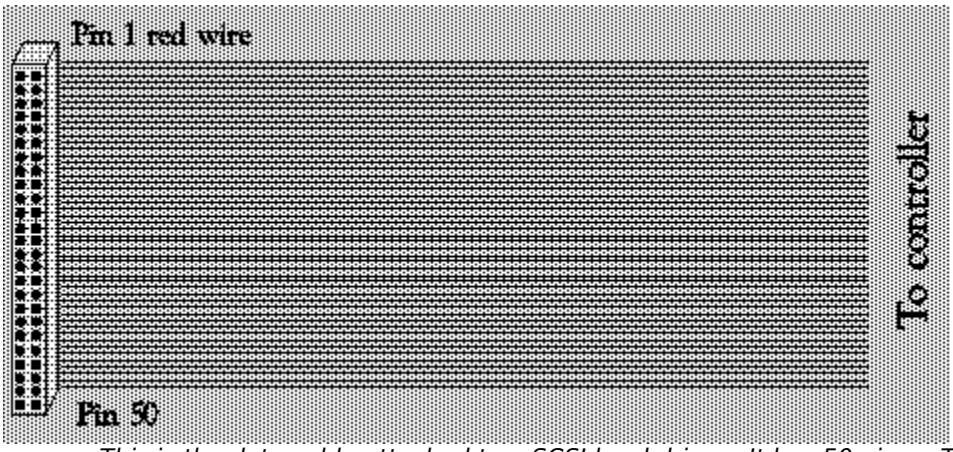
Power Cable

Each drive requires its own power cable. The cable is keyed so that it can only be inserted one way.

Finding Pin 1 On The Cable

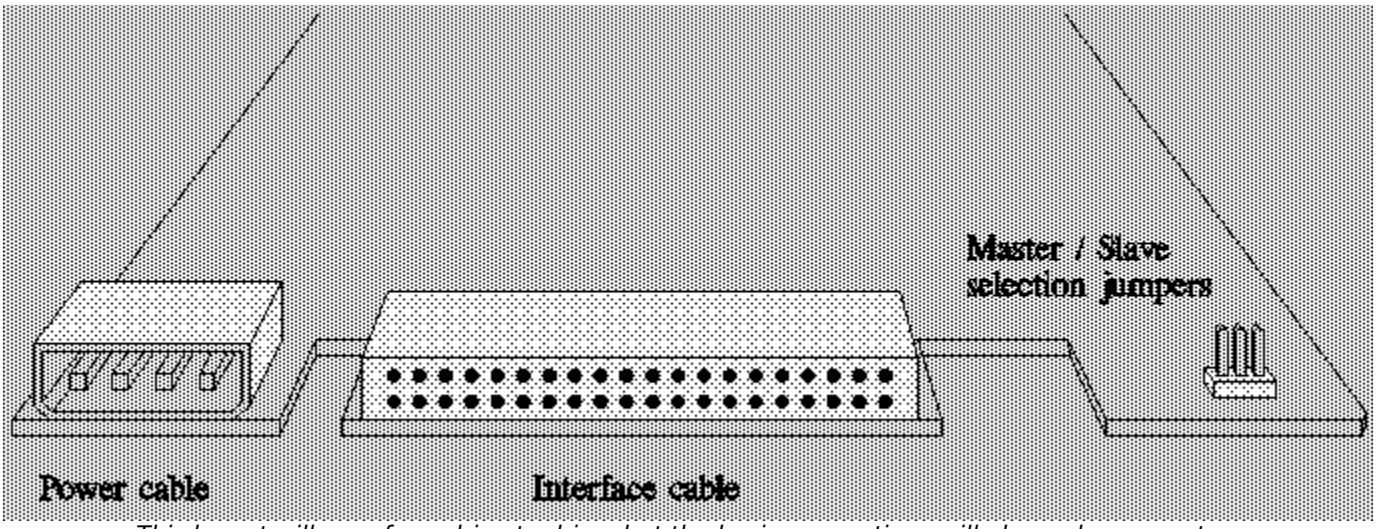
Pin 1 on the control cable can be located by one of three possible markings, a triangle stamped on the connector close to pin 1, a colored stripe on wire #1 (usually red or blue) or a keying tab inside of the connector that will not allow the cable to be inserted incorrectly.

SCSI Drive Header Cable



This is the data cable attached to a SCSI hard drive. It has 50 pins. There is no separate data cable as on ST506/412 hard drives. The cables do not have a twist.

The IDE Interface



This layout will vary from drive to drive, but the basic connections will always be present.

Terminating Resistor Packs

IDE hard drives do not normally have terminating resistor packs.

Drive Select Jumpers (Master/Slave)

The drives connected to the IDE interface are called MASTER and SLAVE. The first (or only) hard drive is the master drive and the second (if attached) is the slave drive. Note that this only applies to AT IDE hard drives. XT IDE drives cannot be daisy-chained together. If two IDE drives are to be used in an XT system, then the controller must have two separate connections.

The jumpers for master and slave selection are normally located on the underside of the drive, close to the data cable.

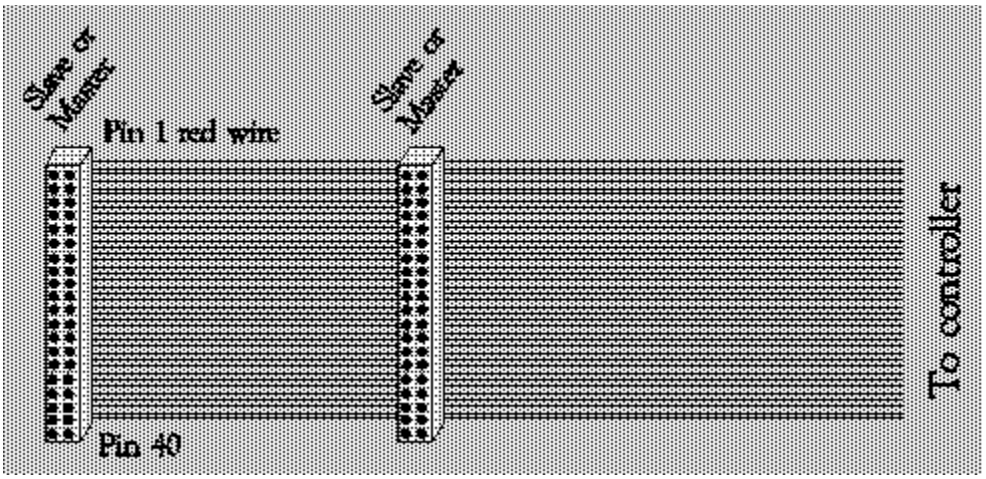
Power Cable

Each drive requires its own power cable. The cable is keyed so that it can only be inserted one way.

Finding Pin 1 On The Cable

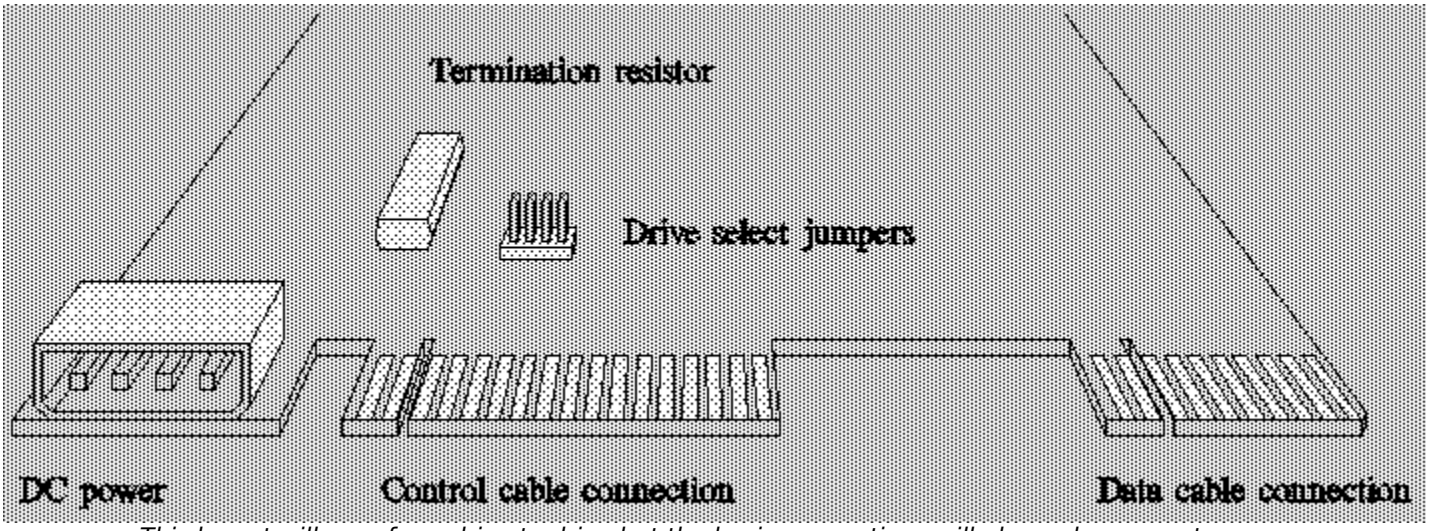
Pin 1 on the control cable can be located by one of three possible markings: a triangle stamped on the connector close to pin 1, a colored stripe on wire #1 (usually red or blue), or a keying tab inside of the connector that will not allow the cable to be inserted incorrectly.

IDE Drive Header Cable



This is the data cable attached to an IDE hard drive. It has 40 pins (IBM uses a 44- or 72-pin connector) and should not exceed 24" in length. There is no separate data cable as on ST506/412 hard drives. The cables do not have a twist. Either master or slave may be attached in either of the connection locations.

The ESDI Interface



This layout will vary from drive to drive but the basic connections will always be present.

Terminating Resistor Pack

This is a socketed resistor pack that is usually yellow and sometimes black or blue. It can be found on the underside of the drive or near the drive control and data cables. Remove these only according to the instructions for the type of control cable used (twisted or non-twisted).

Drive Select Jumpers

There will be 3-12 drive select jumpers depending on the make and model of the hard drive. The position to the far left or far right will be drive select 0 (sometimes the numbering starts at 1). Usually only four of these jumpers are actually used for the drive select function. The others are reserved for other drive parameters.

Power Cable

Each drive requires its own power cable. The cable is keyed so that it can only be inserted one way.

Finding Pin 1 On All Cables

Pin 1 on the control cable can be located by one of three possible markings: a triangle stamped on the connector close to pin 1, a colored stripe on wire #1 (usually red or blue), or a keying tab inside of the connector that will not allow the cable to be inserted incorrectly.

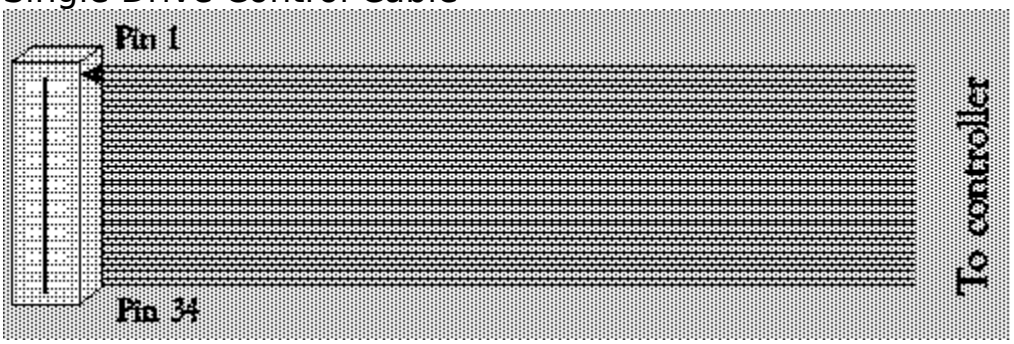
Drive Data Cable

This is the smaller 20-pin ribbon cable. One data cable is required for each hard drive.

Drive Control Cables

This is the large 34-pin ribbon cable. One control cable is required for every two hard drives. There are several types of drive control cables. The type of cable used determines where the drive select jumpers and termination resistors are to be placed on the hard drives. These cables and the location of jumpers and termination resistors are described below.

Single Drive Control Cable



General Use

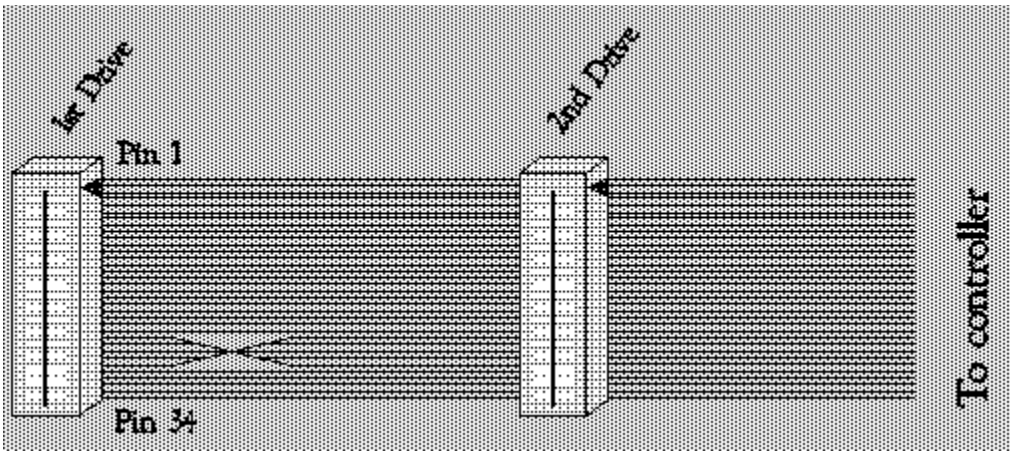
This cable is used in systems with only one hard drive.

Jumper Settings And Termination Resistor

Place the drive select jumper in the first position.

Leave the termination resistor on the drive.

Dual Drive Control Cable With Twist



General Use

This cable is used in systems with one or two hard drives.

Jumper Settings And Termination Resistor

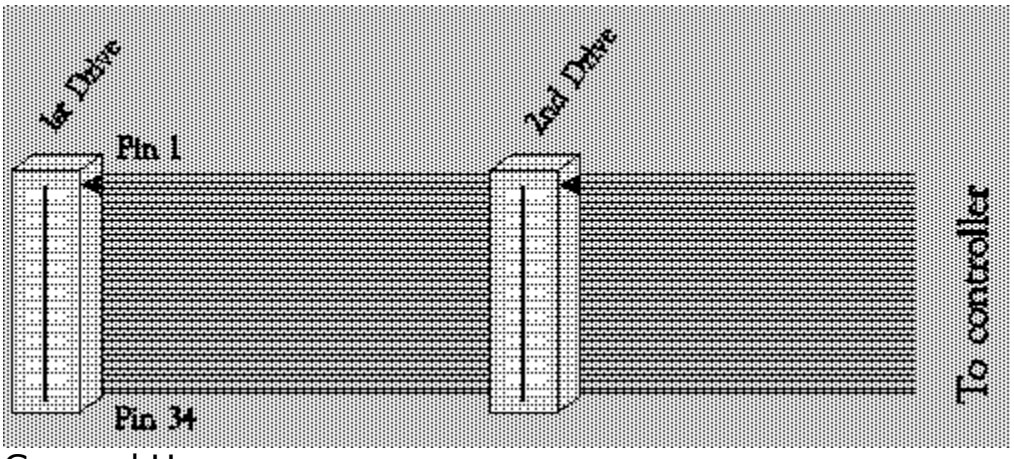
For the first hard drive (usually drive C:) or only hard drive, set the drive select jumper in the second position. Leave termination resistor in place. For the second hard disk (usually drive D:) set the drive select jumper in the second position and remove the termination resistor.

The jumpers are set to the second position on both the drives in this case because the cables twist reverses the drive select pins.

Hard Drive Connection

The middle connector is for the second hard drive (usually drive D:). The last connector is for the first hard drive (usually drive C:) or only hard drive, if there is only one drive connected to the controller card.

Dual Drive Control Cable With No Twist



General Use

This cable is used in systems with one or two hard drives.

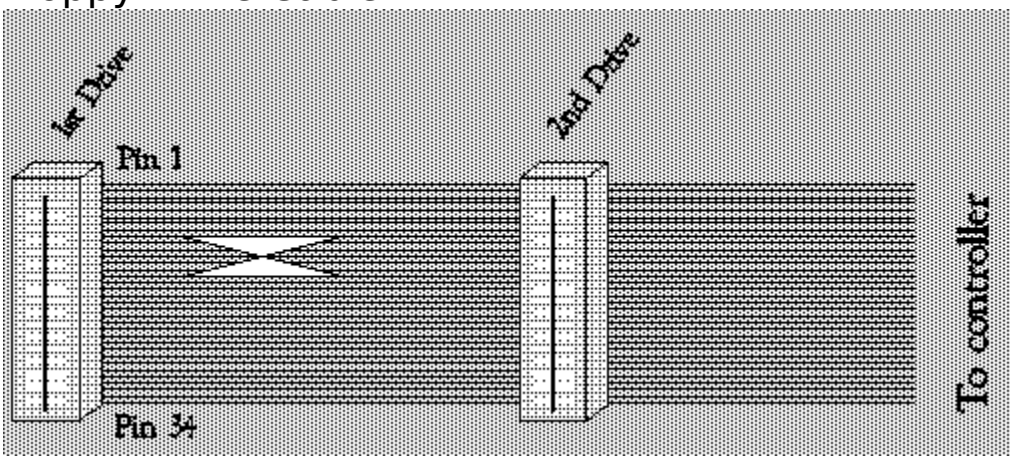
Jumper Settings And Termination Resistor

For the first hard drive (usually drive C:) or only hard drive, set the drive select jumper in the first position. Leave termination resistor in place. For the second hard disk (usually drive D:) set the drive select jumper in the second position and remove the termination resistor.

Hard Drive Connection

The middle connector is for the second hard drive (usually drive D:). The last connector is for the first hard drive (usually drive C:) or only hard drive, if there is only one drive connected to the controller card.

Floppy Drive Cable



Do not substitute a floppy drive control cable for a ST-506/412 or ESDI hard drive cable.

Note that the twist is larger and closer to pin 1.

Glossary

A

ACTUATOR

The device that moves the read/write heads across the platter surfaces. There are two kinds of actuators that are commonly used, stepper motor actuators and voice-coil actuators. The voice-coil actuator is the faster and sturdier of the two, but is more expensive and usually used only in larger more expensive drives. To give you an idea of the speed difference, a stepper motor drive takes from 65 to 100 milliseconds (65 to 100 thousandths of a second) on average to move from one track to another as compared to the voice-coil actuator that usually only requires 30 to 40 milliseconds; twice the speed of the stepper motor. Voice-coil actuated hard drives are usually auto-parking and the stepper motor types usually are not.

ADDRESS MARK

Two byte address at the beginning of both the ID field and the data field of the track format.

ALTERNATE SECTOR

See SECTOR SPARING.

ALLOCATION UNIT

See CLUSTER.

AREAL DENSITY

Bit density (bits per inch, or BPI) multiplied by track density (tracks per inch, or TPI), or bits per square inch of the disk surface. Bit density is measured around a track (circumferential on the disk), and track density is radially measured.

ARLL

Same as RLL 3,9. See RLL.

ASYNCHRONOUS DATA

Data sent in serial or parallel mode without a clock pulse. Time intervals between transmitted bits or bytes may be of unequal lengths.

AT INTERFACE

See IDE & ATA.

ATA

AT Attachment interface

ATA defines a universally agreed upon register set and a 40-pin connector and its associated signals. This is the AT bus or IDE interface. The specification document is available through the X3T9 committee as X3T9.2/90-143. Also see IDE.

AVERAGE LATENCY

A measurement of how long a drive must wait before a specified bit of data rotates under the heads. An average figure is one half a platter rotation. Most drives turn at 60 revolutions per second (or 3,600 per minute) making the value the same for most drives: 8.4 milliseconds. The average latency can be slowed if data is being read or written sporadically so that the average latency period has to pass again and again. The average latency of a drive can be considerably affected by inefficient software, slowing the performance of even a "fast" drive.

AZIMUTH

The angular distance in the horizontal plane, usually measured as an angle from true track location.

B

BCAI

Byte Count After Index

Used in defect mapping to indicate the position of defects with relation to index.

BFI

Bytes From Index

Used in defect mapping to indicate the position of defects with relation to index.

BIT DENSITY

or Bits Per Inch

Expressed as "BPI", bit density defines how many bits can be written onto one inch of a track on a disk surface. Bit density is measured around a track (circumferential on the disk), and it is usually specified for "worst case," which is the inner track. Data is the densest in the inner tracks where track circumferences are the smallest. See Track Density, Areal Density.

BLOCK

A group of bytes handled, stored, and accessed as a logical data unit, such as an individual file record. Typically, one block of data is stored as one physical sector of data on a disk drive.

BPI

Bits Per Inch

See Bit Density

BUFFERED SEEK

A feature of the ST412 interface. In buffered mode, head motion is postponed until a string of step pulses can be sent to the drive. These pulses represent the number of tracks that the head is to be stepped over and are sent much faster than the heads can move. The pulses are saved or buffered, then the optimum head movement to the correct track is performed.

C

CACHE MEMORY

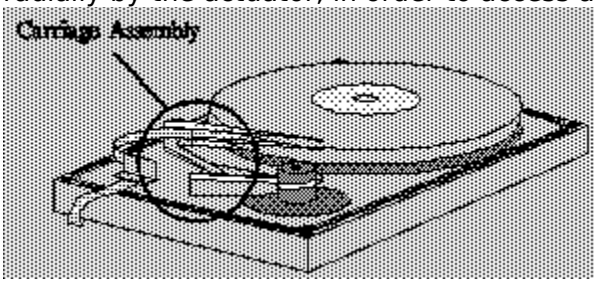
Cache Memory allows the system to load data from the hard disk to memory. The system may then refer to memory for information instead of going back to the hard disk, thereby increasing the processing speed. Today's top of the line ESDI hard disk controllers are equipped with their own cache memory. Some, such as the Everex EV-353, have up to 16 megabytes of controller cache memory.

CAPACITY

This is the total space in megabytes available on the hard disk. Confusing unformatted capacity and formatted capacity is a common mistake. Be sure that you are purchasing a drive with the correct formatted capacity. You could lose over 50 megabytes on a large drive after formatting due to the space taken up by defining the sector boundaries. For example the Maxtor XT-4380E has 384 megabytes of unformatted space but only 319 megabytes of formatted space available to the user. This is a 65 megabyte difference!

CARRIAGE ASSEMBLY

Assembly which holds read/write heads and roller bearings. It is used to position the heads radially by the actuator, in order to access a track of data.



CLUSTER

Also called Allocation Unit. An operating system term describing the number of sectors that the operating system allocates each time disk space is needed. For example, if the cluster size is 16K (Thirty-two, 512-byte sectors per cluster) then every file will use 16K even though the actual file size may be less, and most files on average that are larger than 16K will have 8K of wasted space in their last cluster.

CONTROLLER

A controller is the printed circuit board required to interpret data access commands from host computer (via the bus), and send track seeking, read/write, and other control signals to and from a hard drive.

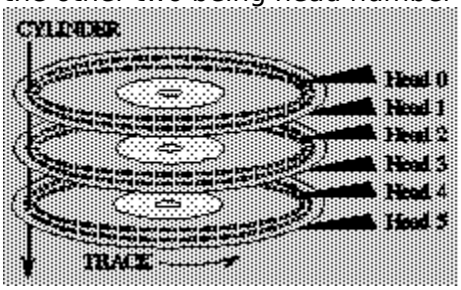
CRC

Cyclic-Redundancy-Check

Used to verify data block integrity. In a typical scheme, 4 CRC bytes are added to each user data block. The 4 bytes are computed from the user data by digital logic chips. The mathematical model is polynomials with binary coefficients. When reading back data, the CRC bytes are read and compared to new CRC bytes computed from the read back block to detect a read error. The read back error check process is mathematically equivalent to dividing the read block, including its CRC, by a binomial polynomial. If the division remainder is zero, the data is error free.

CYLINDER

The cylindrical surface formed by identical track numbers on vertically stacked disks. At any location of the head positioning arm, all tracks under all heads are the cylinder. Cylinder number is one of the three address components required to find a specific address, the other two being head number and sector number.



CYLINDER SKEW

See Skewing.

D

DAISY CHAIN

A way of connecting multiple drives to one controller. The controller drive select signal is routed serially through the drives, and is intercepted by the drive whose number matches. The disk drives have switches or jumpers on them which allow the user to select the drive number desired. No two devices can have the same number.

DTR

Data Transfer Rate

Speed at which bits are sent: In a disk storage system, the communication is between CPU and controller, plus controller and the disk drive. Typical units are bits per second (BPS) or bytes per second, e.g. ST506/412 interface allows 5 Mbits/sec., ESDI allows up to 24 Mbits/sec. transfer rates.

DEDICATED SERVO SYSTEM

A complete disk surface and head are dedicated for servo data. Hard drives with an odd (not even) head count will have a dedicated servo. Also see Servo Track, Embedded Servo System.

DISK PLATTER

For hard (rigid) disks, a flat, circular disk substrate, coated on both sides with a magnetic substance (iron oxide or thin film metal media) for non-volatile data storage. The substrate may consist of metal, plastic, or even glass. Surfaces of disks are usually lubricated to minimize wear during drive start-up or power down.

DMA

Direct Memory Access

A means of data transfer between peripheral and host memory without processor intervention. This is a means of avoiding the "bottleneck" if the data has to go through the processor to the memory and so is a very fast way to load data on or off of a peripheral into

or out of memory.

DRIVE-SELECT JUMPERS

These set the control channel for the hard drive so that the controller knows which drive it's controlling. See the section on hard drive installation and your hard drive manual for the proper setting of these jumpers.

DRIVE TYPE

A number representing a standard configuration of physical parameters: cylinders, heads, and sectors per track of a particular type of disk drive. Each AT system BIOS contains a list of drive types that the system considers "standard types." These types are not necessarily the same from one BIOS to the next. See the chapter on BIOS tables for a listing of these.

DROP-IN/DROP-OUT

Types of disk media defects usually caused by a pin size hole in the disk coating. If the coating is interrupted, the magnetic flux between medium and head is zero. A large interruption will induce two extraneous pulses, one at the beginning and one at the end of the pin-hole (2 DROP-INs). A small coating interruption will result in no playback from a recorded bit (a DROP-OUT).

E

ECC

Error Correction Code

The ECC hardware in the controller used to interface the drive to the system can typically correct a single burst error of 11 bits or less. This maximum error burst correction length is a function of the controller. With some controllers the user is allowed to the select this length. The most common selection is 11.

EMBEDDED SERVO SYSTEM

Servo data is embedded or superimposed along with data on every cylinder as opposed to having a dedicated servo track. See Dedicated Servo System.

ESDI

Enhanced Small Device Interface

An ad-hoc group of controller and device manufacturers (led by Maxtor Corporation) met to develop a standard that would increase the data capacity and speed of the existing ST506/412 interface.

The first standards document was released in 1983. This initial release defined the Enhanced Small Disk Interface but after the Enhanced Small Tape Interface was defined, it was decided in October, 1983 to merge the two standards into one. This new interface became the Enhanced Small Device Interface (ESDI). In 1985 a version suitable for optical disks was released. In 1987 it was agreed that the ESDI definition for tape did not have enough acceptance and would not be incorporated in the standard.

The ESDI standard has been approved by the ISO (International Organization for Standardization) as ISO 10222:198x, and X3T9 standard X3.170. The complete ESDI standards document can be obtained from the X3T9 committee or Global Engineering Documents at 800-854-7179.

ESDI can transfer data up to 24 megabits per second, although most transfer at 10

megabits. It is an "intelligent" controller that can potentially, not only handle hard drives, but also floppy drives, tape backup units, CD and handle direct file transfer between these devices. This interface also performs a lot better error checking than the ST506/412 standard. Used on many of the large capacity hard drives.

Usually the drive type is set to 1 (one) and the on-board BIOS on the ESDI interface handles any compatibility problems. When implemented in a DOS system, access to the host system is done through interrupt 13 (INT13).

F

FCI

Flux Changes Per Inch

Synonymous with FRPI (flux reversals per inch). In MFM recording 1 FCI equals 1 BPI (bit per inch). In RLL encoding schemes, 1 FCI generally equals 1.5 BPI.

FAT

File Allocation Table

What the operating system uses to keep track of which clusters (allocation units) are allocated to which files and which are available for use. There are typically two copies of the FAT in case one gets damaged.

FLUX CHANGE

Location on the data track, where the direction of magnetization reverses in order to define a 1 or 0 bit. See FCI and FRPI for more information.

FM ENCODING

Frequency Modulation Encoding

This is an outdated encoding scheme that is no longer in use. It used up to half of the disk space with timing signals for the encoding process. The technology was refined and replaced with a new standard called MFM encoding. See MFM, RLL Encoding.

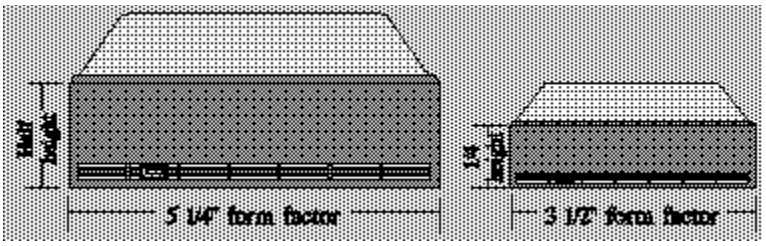
FORMAT

The purpose of a low-level format is to record the "header" data that organize the tracks into sequential sectors on the disk surfaces. This information is never altered during normal read/write operations. Header information identifies the sector number and also contains the head and cylinder address.

The purpose of a high-level format is to scan and verify the disk surface if needed, create the system areas and place the operating system software on the hard drive, so that the drive is bootable and mapped out for file storage.

FORM FACTOR

This is the hard drives physical external size and the mounting space it will take up. Usually 3.5 inch half height or 5.25 inch half height. Smaller hard drives are now available, such as 2.5 inch, and 1.8 inch, which are useful in laptop applications.



G

"G"

A G is a unit of force applied to a mass at rest equal to the force exerted on it by gravity. Hard disk drive shock specifications are usually called out in Gs. A shock specification of 40 Gs non-operating means that a drive will not suffer any permanent damage if subjected to a 40 G shock when not turned on. This is roughly equivalent to a drop of the drive to a hard surface from a distance of 1 inch!

GIGABYTE

A unit of storage equal to exactly 1,073,741,824 bytes (1,000 megabytes). Some of the newer hard drives and optical drives are in the gigabyte range.

H

HEAD

The head is the small magnetic device that reads and writes data off of the magnetic media. Electrical pulses are sent to the head to create magnetic areas on the media and these magnetic areas create electrical pulses in the head when reading the data off the media. There are usually several read/write heads in a hard drive. On larger drives a single head and platter surface is reserved for the servo. This keeps track of where the rest of the heads are positioned (See Voice-Coil Actuators). So the term "heads" usually only pertains to actual data heads, those that read/write. Such is the case with typical drive listings. There may be eight platter surfaces but only seven data heads. Only one of the several read/write heads is in use at any one time. The controller can only handle data from one head at a time, which does not result in any loss of speed as the computer can only take in so much data at one time. There are three types of heads commonly used, composition, monolithic, and thin film.

Head Capacities:

Head Type	BPI	TPI	AREAL DENSITY (BPI x TPI)
Composition	12,000	1,000	$12x10_6$ (or 12,000,000)
Monolithic	8,000	450	$3.6x10_6$ (or 3,600,000)
Thin Film	25,000	1,500	$37.5x10_6$ (or 37,500,000)

HEAD CRASH

As a normal operation, a head landing occurs when the disk drive is turned on or off. When the heads land, a thin film of lubricant on the disk surface protects the disk from damage. A head crash occurs when the head and disk damage each other during landing because of rough handling or because a small particle gets between them. Head crash is a catastrophic failure condition and causes permanent damage and loss of data.

HEAD LANDING AND TAKEOFF

In many Winchester drives, the head is in contact with the platter when the drive is not powered. During the power up cycle, the disk begins rotation and an "air cushion" is established as the disk spins up to full RPM (rotations per minute, usually 3,600). This air bearing prevents any mechanical contact between head and disk. This is analogous to a plane taking off and landing.

HEAD PARKING

Parking the heads places them in a safe zone away from data on the platter, usually the highest cylinder on the drive. When a stepper motor drive is turned off its heads land on the platter wherever it was last doing a read/write. If the drive is jolted, data under the heads can be lost. The stepper motor drive must be parked through software which places the head at a safe zone. Voice-coil drives automatically park themselves due to the design of the solenoid-spring mechanism (See Voice-Coil Actuators). Head parking software is not needed on these drives and should not be used.

HEAD SKEW

See Skewing.

HEIGHT

The drive's height. See Form Factor.

I

IDE

Imbedded Drive Electronics

This interface originated in 1988 when a number of peripheral suppliers formed the Common Access Method Committee to push an industry-wide effort of adopting a standard software interface for SCSI peripherals. Part of their goal was to specify what is now known as the ATA (AT Attachment interface) which would allow an interface to be designed into the new low cost AT-compatible motherboards. A standard was established and approved by the X3T9 committee and sent to ANSI for approval. The complete ATA standards document can be obtained from the X3T9 committee and Global Engineering Documents as document X3T9.2/90-143 at 800-854-7179.

The ATA interface is usually not mentioned, it is encompassed in the term IDE. ATA refers to the interface itself and IDE to the hard drive.

The hard drives used with this type of interface are "intelligent" devices that have most of the controller functions built into the drive circuitry. These drives link up to the computer via a 40-pin connector (IBM uses a 44-pin or 72-pin connector) and a small paddle card, or directly to the motherboard on some newer computers. Most use RLL encoding and 36 or more sectors per track which is unacceptable for most BIOS drive type tables. Most IDE drives do a translation so that they may be used in systems that do not support the drive type needed. The physical information is the actual heads, cylinders, and sectors per track

of the hard drive. The logical information is the heads, cylinders, and sectors per track, that the drive and BIOS table can agree on.

If you don't have the logical conversion information when installing an IDE drive into a system, the drive type should be set to the type that most closely matches, but is not more than the size in megabytes of the drive being installed. If your BIOS allows manual parameters to be entered, then enter the physical heads, cylinders, and sectors per track for the drive. For example, if the IDE hard drive to be installed is 21 megabytes in size, type 2 would be selected in the 286 BIOS setup for this drive, as it is the closest match in megabytes to the IDE drive.

All IDE hard drives are factory low-level formatted. Low-level formatting the IDE hard drive by the user is not recommended. The factory defect list and cylinder/head skewing may be erased and the drive destroyed by the low level-format.

ID FIELD

See sector header

INDEX PULSE

The Index Pulse is the starting point for each disk track. The index pulse provides initial synchronization for sector addressing on each individual track.

INDEX TIME

The time interval between similar edges of the index pulse, which measures the time for the disk to take one revolution. This information is used by a disk drive to verify correct rotational speed of the media.

INTERLEAVE

Most hard drives can transfer data faster then some CPUs or computers can accept it. Interleaving alleviates this problem by forcing the drive to read the data a little slower. This is accomplished by renumbering the sectors. Instead of going from 1 to 17 sequentially, they are staggered.

If the interleave is 3 then the sectors are renumbered: **1**-7-13-**2**-8-14-**3**-9-15-**4**-10-16-**5**-11-17-**6**-12. This allows the CPU to store the data for sector #1 while #7 and #13 are under the head, and then continue with sector #2 when it's ready. In one revolution, approximately six sectors of 512 bytes will have been read and stored by the CPU. In three revolutions, all of the sectors will have been read and stored. Most PC/XTs use an interleave of three or four. Most PC/ATs use an interleave of 1 or 2. It really depends on the speed of the CPU and the controller electronics. An interleave of three should be fine for most XT installations.

DrivePro will allow you to adjust the interleave after the low level is done. It will even find an optimum interleave for the system and drive you are using or tell you the current interleave being used.

IPI

Intelligent Peripheral Interface

A hard drive interface used with the larger 8" and 14" mainframe and minicomputers.

L

LANDING ZONE

This is the section of the disk that is designed as the safe zone for head parking (See Head Parking).

LATENCY (ROTATIONAL)

See Average Latency

LOW-LEVEL FORMAT

The first step in preparing a drive to store information after physical installation is complete. The process sets up the "handshake" between the drive and the controller. In an XT system, the low-level format is usually done using DOS's debug utility. In an AT system, *DrivePro* can be used to do low-level formatting, either by accessing the controller's built-in firmware, or by using the INT13 interface. Some SCSI and ESDI controllers may include or have optional special software for that particular controller to do the low-level formatting.

Do not low-level IDE drives unless specifically told to do so by the manufacturer! They have been factory low-leveled and damage may occur if the low-level is redone.

LUN

Logical Unit Number

The unit's logical device number. Each device affiliated with a certain SCSI target must have its own unique LUN. Also see Daisy Chain, SCSI.

M

MEGABYTE

A unit of storage equal to exactly 1,048,576 bytes (roughly 1 million bytes). Most hard drives are in the megabyte range. Also see Gigabyte.

MFM ENCODING

Modified Frequency Modulation Encoding

A type of encoding scheme that converts the digital bits from the computer into a pattern of magnetic changes or "flux reversals" that are then stored on the hard drive. MFM uses a fixed length encoding scheme. Flux reversals on the disk always will be evenly spaced in time so that the beginning of one bit can be separated from another. This type of scheme allows even single bit errors to be detected easily and corrected by the controller electronics. The encoding/decoding process is done on the controller card not at the hard drive. MFM is a modified version of FM which does away with the need for timing signals

allowing it to pack twice as much data on the same drive. It has completely replaced the FM encoding technology and is very widely used at this time. Also see FM Encoding, RLL Encoding.

MCA

Micro Channel Architecture

The type of bus used in the IBM PS/2 line of computers. Each MCA hard disk controller board can support two hard disks. If two hard disk boards are installed in a PS/2 system only one drive may be attached to each controller (you can never have more then a maximum of two hard drives in a PS/2 system, no matter how many controller cards you have). If you have one ESDI and one ST506/412 drive, the MCA architecture always selects the ESDI drive as drive C:.

MTBF

Mean Time Between Failures

This figure is supposed to indicate how long a drive is expected to last between needing repairs. The figure is normally rounded to the nearest thousand or sometimes even five thousand. The manufacturers really have no solid way of arriving at these figures so they rely on a battery of stress tests and mathematics based on testing of individual component lifetimes that are analyzed and worked into real time working condition hours. MTBF values are measured in power-on hours (conditions with the drive on).

MTTR

Mean Time To Repair

The average time to repair a given unit.

MEDIA

The magnetic layers of a disk or tape. Also, loosely, the disk or tape itself.

MEDIA DEFECT

A media defect can cause a considerable reduction of the read signal (missing pulse or Drop-Out) or create an extra pair of pulses (Drop-In). The factories that manufacture the hard drives have equipment that can detect media defects which the standard controller might not. These areas will potentially give the user problems. Even if your own extensive diagnostics finds less defects than the list supplied with the drive, enter in all defects when low level formatting to ensure error free operation. Also see Drop-In/Drop-Out.

N

NRZ

Non-Return To Zero

A method of magnetic recording of digital data in which a flux reversal denotes a one bit and no flux reversal, a zero bit. NRZ recording requires an accompanying synchronization clock to define each cell time (unlike MFM or RLL recording). ESDI drives usually use this type of encoding.

P

PARITY

A computer data-checking method using an extra bit in which the total number of binary 1's (or 0's) in a byte is always odd or always even; thus, in a parity scheme, every byte has seven or eight bits of data and one parity bit. If using odd parity and the number of 1 bits comprising the byte of data is not odd, the 9th or parity bit is set to 1 to create the odd parity. In this way, a byte of data can be checked for accurate transmission by simply counting the bits for an odd parity indication. If the count is ever even, an error is indicated.

PARTITION

A section of a hard drive designated for a specific operating system and space. A hard drive can have up to four partitions under DOS. There are different partitioning limitations for each DOS version.

Version 2.0 may not have a hard drive over 16MB.

Versions from 2.1 but before 3.30 may not have a hard drive over 32MB. You may use a replacement DOS block device driver to overcome this limitation.

Versions from 3.30 but before 4.00 may have multiple partitions no greater than 32MB. Therefore a 100-megabyte drive will have to be partitioned into four parts: 32MB, 32MB, 32MB and 4 MB. Compaq DOS 3.31 allows partitions larger than 32MB, however.

Versions from 4.00 but before 5.00 may have partitions up to 512MB.

Version 5.00 allows for partitions up to 2 gigabytes.

These are only MS and PC-DOS limitations and not limitations of the hard drives themselves. On ST506/412 drives a virtual split may be done to overcome these limitations (See virtual split in the glossary).

PIO

Programmed Input/Output

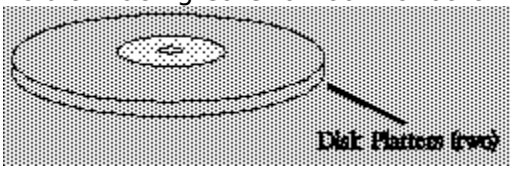
A means of data transfer that requires the use of the host processor. See DMA for more information.

PLATED THIN FILM DISKS

Magnetic disk memory media having its surface plated with a thin coating of a metallic alloy instead of being coated with oxide. Plated disks hold far more data than coated media. Also see Platter and Coated media.

PLATTER

The circular aluminum or other surface where the actual data is stored. They are usually either coated with an oxide substance (like a floppy disk) or plated. Coating will hold far less magnetic particles than plating. Coated media can hold up to 20,000 magnetic domains in an inch of track while plated media can hold up to 50,000. Plated media are also extremely hard and much less susceptible to head crashes. Because of the precision and cost needed in making plated media, it is much more expensive and is generally used only in higher-capacity drives. Newer drives use glass platters which are smoother and can hold a much greater amount of data in less space.



Q

QIC-36 Interface

A 50-pin tape drive interface that has become an industry standard.

QIC-02 Interface

Tape drive software command set that has become an industry standard.

R

RADIAL

A way of connecting multiple drives to one controller. In radial operation, all output signals are active even if the drive is not selected. Also see Daisy Chain.

RECALIBRATE

Return to Track Zero. A common disk drive function in which the heads are returned to track 0 (outermost track) to ensure good positioning by having a stable starting point.

RWC

Reduced Write Current

A signal input (to some older drives) which decreases the amplitude of the write current at the actual drive head. Normally this signal is specified to be used during inner track write operations to lessen the effect of adjacent bit "crowding." When installing a drive in a system, the number requested is the first track number to begin the area of reduced write current. That track and all subsequent tracks will be written with reduced write current.

RLL

Run Length Limited

A type of encoding scheme that requires less flux reversals (changes in the media) for a given amount of data (see MFM). The logic used is to simply replace each sequence of 1's and 0's with an encoded sequence that allows greater packing density, since it ensures that flux reversals stay farther apart than with MFM. This allows a lot more data to be placed on the disk than MFM. In RLL 2,7 the "run length limit"(limit of 0 bits next to each other) is 7 (this is limited so that the clock can be kept synchronized with data given that there are very slight speed variations in the rotation of the platter). The codes are chosen so that sequences of zeros in the codes always range from 2 to 7. This allows for a fifty percent increase in disk space over MFM encoding. RLL 3,9 (commonly called ARLL) is also used which further increases disk space (up to 100% from MFM) but also increases the potential for bit errors stemming from rotational speed variations. RLL 2,7 is more commonly used. With the exception of IDE and SCSI, the encoding/decoding process is done on the controller card not at the hard drive.

A warning about RLL: Do not use MFM drives on an RLL controller! It can be done (and often is) but the life of the hard drive will be greatly reduced and may result in the loss of data.

ROTATIONAL SPEED

The speed at which the media spins. On a 5-1/4 or 3-1/2" Winchester drive it is usually 3,600 RPM. The rotational speed affects latency, so a higher rotational speed allows faster access to data. Some drives go up a few hundred more RPM due to this reason. See also Latency.

S

SA-400 Interface

Shugart Associates designed the SA-400 floppy disk drive in 1978, it was the first floppy drive to gain wide acceptance. This drive utilized a 34-pin cable that is still used in floppy drives today. The interface was later modified for hard drives and this modified version eventually became the ST-506 interface. The SA-400 pinouts have been slightly modified over the years, but the industry standard floppy interface is still referred to as the SA-400.

SCSI

Small Computer Systems Interface

This interface originated as the SASI (Shugart Associates System Interface) in 1979. It was one of several disk interfaces that worked at a logical level instead of the widely accepted device level. Working at a logical level allowed for a stable interface while the disk devices could change rapidly.

In 1980, Shugart Associates attempted to replace IPI (Intelligent Peripheral Interface) through the X3T9 standards committee but were not able to, due to limited industry acceptance. NCR added features to Shugart's original interface and in 1982 the X3T9 decided to start a project for SCSI which was to be based on SASI. During the project, optical WORM commands were added, no longer limiting SCSI to disks.

In 1984 NCR released the NCR 5380. This chip includes on-chip single-ended drivers/receivers, which allowed the industry to produce inexpensive SCSI interfaces. This is the chip that Apple Computer originally implemented in their Macintosh computer and which gained SCSI widespread acceptance.

In 1986 ANSI approved SCSI as ANSI X3.131-1986. Also in 1986 X3T9 began the SCSI-2 project to incorporate the CCS (Common Command Set) into SCSI plus numerous other improvements and additions. In 1990 SCSI-2 was submitted to ANSI but as of this writing, it has not yet been approved. The complete SCSI standards document can be obtained from ANSI.

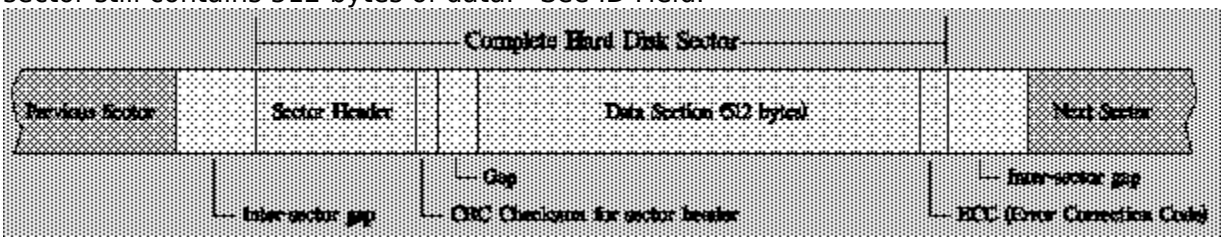
As you can gather from the above history, this interface is not only designed for hard drives. It is a device interface that leaves most of the interfacing to the devices attached to it. Up to eight targets, each having up to eight logical units or devices, such as hard drives, tape

drive units, floppy drives, and even printers and computers, can be chained together. Each device must have its own unique Target number and Logical Unit Number (LUN) so that it can be identified. These are set by jumpers on the devices.

Usually the drive type is set to 0 (zero) and the on-board BIOS on the SCSI host adapter (which is a SCSI device on the daisy chain) handles any compatibility problems. When implemented in a DOS system, access to the host system is done through interrupt 13 (INT13).

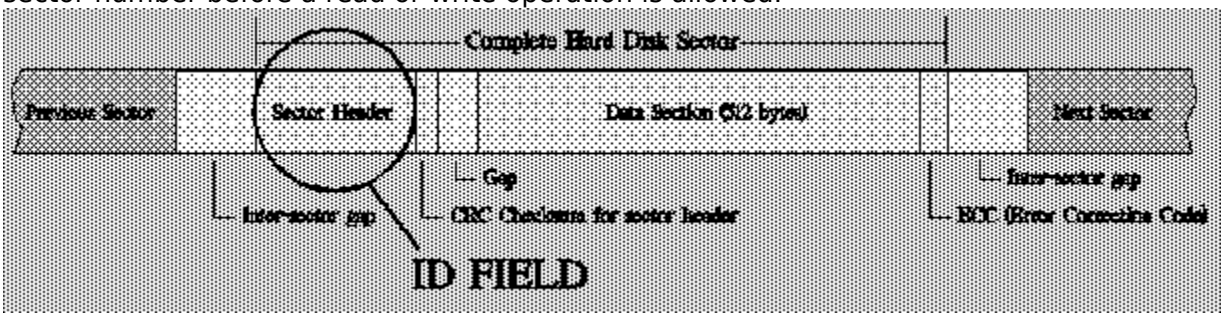
SECTOR

A section of one track is called a sector. Each sector is defined with magnetic markings and an identification number from 0 to 65,535 (this identification number is contained in the sector header). All current hard drives use 512 bytes of data per sector (in the data section). All tracks have the same amount of sectors even though the tracks are much larger near the outside of the platter than the inside. This is only done by DOS to avoid extra complications and as a result gives up much valuable disk space. More advanced recording methods have been introduced such as ZBR (Zone Bit Recording) in which tracks on the outside cylinders have more sectors per track than the inside cylinders, but each sector still contains 512 bytes of data. See ID Field.



SECTOR HEADER

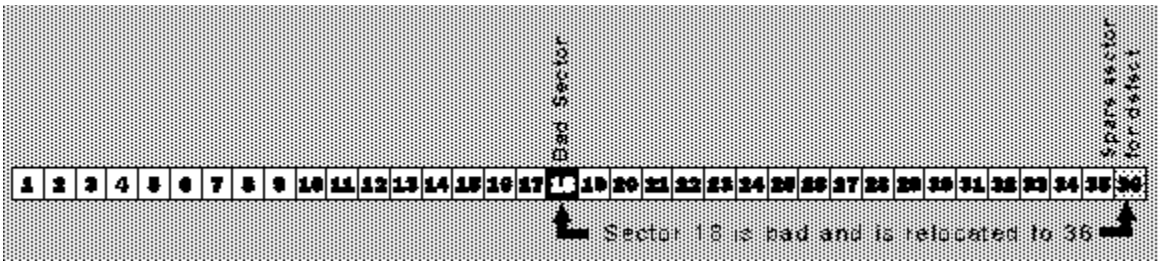
The address portion of a sector. The sector header (ID field) is written during the format operation. It includes the cylinder, head, and sector number of the current sector. This address information is compared by the disk controller with the desired head, cylinder, and sector number before a read or write operation is allowed.



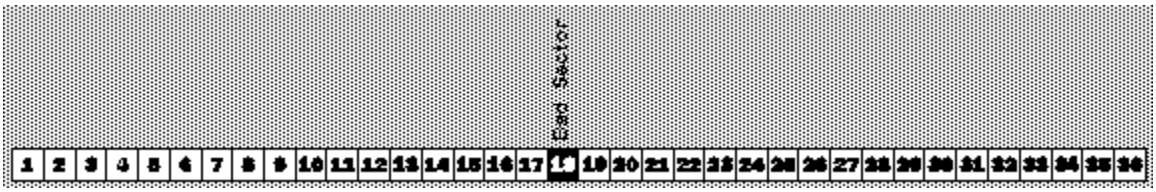
SECTOR SPARING

This reduces the number of sectors on each track by one and places defect information on it. The application will see less defects as only the drive is aware of the spare sectors. This reduces the total capacity of your drive but is useful if the drive has a large amount of defects and your application requires a defect free drive.

SECTOR SPARING FOR A 36 SECTOR DRIVE
OPERATING SYSTEM SEES 35 GOOD SECTORS AND NO BAD SECTORS



SAME DRIVE WITHOUT SECTOR SPARING
OPERATING SYSTEM SEES 35 GOOD SECTORS AND 1 BAD SECTOR



SEEK

The radial movement of the heads to a specified track.

SEEK TIME

Seek time usually refers to the average time it takes the heads to move between one track to another, on average. This is the seek time referred to in the hard drive listings in this book. There is also the track-to-track seek time which reflects how long it takes the heads to move between adjacent tracks, and the full stroke seek time, which calculates how long it takes for the heads to move from track 0 all the way to the last track.

SERVO TRACK

A prerecorded reference track on the dedicated servo surface of certain disk drives. The position of the carriage assembly is adjusted based on the strength of the received servo signal from the servo head so that the data heads are precisely positioned over the data. See Embedded Servo System for a description of the other type of servo encoding.

SKEWING

Skewing is the repositioning of the first sector on each track to synchronize with the time required to switch heads or cylinders (tracks). A skew factor is added when formatting a disk to improve performance by reducing wasted disk revolutions.

Head skew reflects the time it takes for a drive to switch between head groups. It will be set to 0 if the drive does not support head skewing and most don't.

SMD

Storage Module Device

A hard drive interface used with the larger 8" and 14" mainframe and minicomputers hard drives. This interface is being phased out and replaced with SCSI.

SPINDLE

The rotating hub structure to which the disks are attached.

SPINDLE MOTOR

This is the motor that spins the platters. These motors are direct connect, they never use belts or gears. Most spindle motors spin at 3,600 RPM and are controlled precisely at this

speed. They are usually mounted outside of the enclosed platter area. Maxtor uses a unique design in some of their drives placing it in between the actual platters thus allowing for more platters in a 5 1/4" casing.

SPINDLE MOTOR GROUND STRAP

Most hard drives have this strap attached to the circuit board pressing against the spindle or spindle motor. This dissipates the static generated by operation and grounds the spindle motor casing. When you hear the drive making high-pitched or scraping noises it is usually this strap. Placing a drop of oil between it and the spindle or adjusting it will usually stop the noise.

SPT

Sectors Per Track.

ST-506/412 INTERFACE

This interface was developed by Seagate Technologies (originally Shugart) in 1980 solely for use with their ST-506 hard drive (a 5 megabyte formatted drive). It was later revised in 1981 with a feature called "buffered seek" for their ST-412 (a 10 megabyte formatted drive). Today it is still a very common hard drive interface among personal computers. Most of the smaller capacity drives use this type of interface. You will not see this interface on drives larger than 140 megabytes due to its design limitations. These controllers transfer data at around 5 megabits per second which is relatively slow according to today's standards.

When implemented in PC & XT type systems, these interfaces utilize an on-board BIOS that contains a specific table of hard drive parameters. The hard disk to be installed must match one of the drive parameters in the table (some controllers allow user-definable parameters). When implemented in AT (80286) systems, the BIOS in the host computer has a table of hard drive parameters. The hard disk to be installed must match one of the drive parameters in the table (some AT BIOS manufacturers allow for a user-definable drive type).

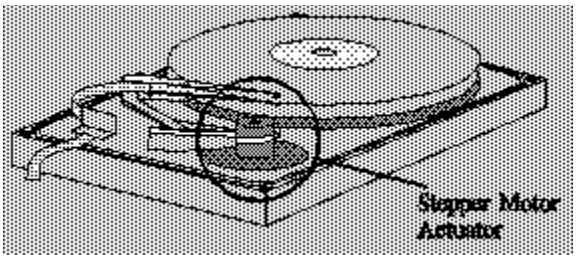
Due to BIOS and DOS limitations, the maximum heads are 16 and cylinders 1,024 (unless a replacement block device driver is used for DOS). There are no such limitations on the other types of PC interfaces due to their conversion techniques and direct manipulation of the hard drives.

STEP

An increment or decrement of the head positioning arm to move the heads in or out respectively, one track from their current position. In buffered mode, the head motion is postponed until the last of a string of step pulses has been received. See Buffered Seek.

STEPPER MOTOR ACTUATOR

The head actuator is responsible for moving the heads back and forth over the platters. A stepper motor actuator uses a motor that moves the heads by rotating the motor a "step" at a time. By rotating the motor a precise number of steps and then converting these steps into linear motion, the heads are moved. Alignment is assured by a metal band that can cause the drive to be put out of alignment if overheated or worn. Most hard drives, especially inexpensive ones, use this kind of actuator. Floppy drives also use this kind of actuator. Also see Actuator, Voice-coil Actuator.



STEP PULSE

The pulse sent from the controller to the stepper motor on the step interface signal line to initiate a step operation.

SYNCHRONOUS DATA

Data sent, usually in serial mode, with a clock pulse defining the intervals between bits.

T

THIN-FILM HEADS

A read/write head whose read/write element is deposited using integrated circuit techniques rather than being manually fabricated by grinding ferrite and hand-winding coils. Also see Head.

TERMINATING RESISTORS

Hard drives are always shipped with the terminating resistor in place. This is a small resistor pack usually located near the bottom or near the drive select jumpers. They are usually yellow, and sometimes black or blue. They provide electrical signal termination so that the control signals do not echo along the drive cables. They also provide the proper electrical load for the controller and not using them properly could damage the controller. See the section on hard drive installation for the proper use of these resistors.

TPI

Tracks Per Inch

A measurement of track density. Hard drives usually have a TPI of over 1,000. In contrast, 360 kilobyte floppy drives have a TPI of 48.

TRACK

The concentric circles that hold data on a disk platter. A track is composed of a circle of magnetic flux changes. Each track is divided into sectors, which are normally 512 bytes in length.

TRACK DENSITY

or TPI

Track density (or Tracks Per Inch) defines how many tracks can be written onto a disk surface. Track density is radially measured.

V

VIRTUAL SPLIT

A virtual split is a logical (not actual) split of the disk drive. Some controller cards support this option when low level formatting to make one physical drive appear as two drives to the operating system. This overcomes the partitioning limitations of some earlier versions of MS/PC-DOS.

VOICE-COIL ACTUATOR

The head actuator is responsible for moving the heads back and forth over the platters. Voice-coil actuators use a solenoid (a magnet that pulls on a metal rod) to pull the heads toward the center of the platter. The heads are placed on a hinge mechanism with a spring that pulls in the other direction. When the solenoid is let go (magnetism released) the heads get pulled back to the outer edge of the platters and are held there by a small magnet. The terms "voice-coil" is used due to the likeness of this system to speakers, since speakers pull and push a cone. Voice-coil actuators are a lot faster than stepper actuators because they don't have to "step" a little at a time, they just fly to their destination. The voice-coil knows what track it's on by reading information, called the "servo" data, that has been permanently placed between the tracks. Some of the larger capacity drives use one whole platter surface to store this information. That is why you will see a drive that has only seven data heads but four platters (there are usually two heads for every platter). The eighth is reserved for the servo platter. See Actuator, Stepper Motor Actuator.

W

WEDGE SERVO SYSTEM

See Embedded Servo System

WINCHESTER DRIVE

A term that originated from the old IBM drives in the 1960's that had 30 megabytes of removable media and 30 megabytes of fixed media. Thus the name 30-30, which is the caliber of the rifles made by the Winchester gun factory. The name now refers to all PC hard drives.

WRITE CURRENT

See Reduced Write Current.

WORM DRIVE

Write Once Read Mostly
Write Once Read Many

WORM drives contain removable cartridges that can be written to once and read from indefinitely, similar to a phonograph record.

WP

Write Precompensation

The varying of the timing of the head current from outer tracks to the inner tracks of a disk to compensate for the bit shifting that occurs on the inner cylinders which pack more data

into a smaller area (See Sectors). Sometimes this number must be entered at time of low-level formatting, and is only required on some oxide-media drives.

X

XSMD
Extended Storage Module Device Interface

See SMD.

XT INTERFACE

See IDE, ATA.

Z

ZBR
Zone Bit Recording

Trademark of Seagate Technology. A media optimization technique where the number of sectors per track is dependent upon the cylinder circumference; e.g., tracks on the outside cylinders have more sectors per track than the inside cylinders. The ZBR format is only done at the factory.

 *These drives have been factory low-level formatted and should not be low-level formatted by the end-user!*