Chapter 13

Floppy Disk Drives and Controllers

This chapter examines, in detail, floppy disk drives and disks. It explores how floppy disk drives and disks function, how DOS uses a disk, what types of disk drives and disks are available, and how to properly install and service drives and disks. You learn about all the types of drives available for today's personal computer systems; these drives include both the 5 1/4-inch and 3 1/2-inch drives in all different versions. The chapter discusses also the newer, extra-high density (ED) 3 1/2-inch disks. Also discussed are several upgrade options including the addition of the 3 1/2-inch drives to the early PC-family systems, which enables them to be compatible with many newer systems, as well as upgrading newer systems to support the ED drives and disks.

Development of the Floppy Disk Drive

Alan Shugart is generally credited with inventing the floppy drive while working for IBM in the late 60s. In 1967, he headed the disk drive development team at IBM's San Jose lab, where and when the floppy drive was created. One of Shugart's senior engineers, David Noble, actually proposed the flexible media (then 8 inches in diameter) and the protective jacket with the fabric lining. Shugart left IBM in 1969 and took more than 100 IBM engineers with him to Memorex. He was nicknamed "The Pied Piper" because of the loyalty exhibited by the many staff members who followed him. In 1973, he left Memorex, again taking with him a number of associates, and started Shugart Associates, to develop and manufacture floppy drives. The floppy interface developed by Shugart is still the basis of all PC floppy drives. IBM used this interface in the PC, enabling them to use off-the-shelf third-party drives instead of custom building their own solutions.

Shugart actually wanted to incorporate processors and floppy drives into complete microcomputer systems at that time, but the financial backers of the new Shugart Associates wanted him to concentrate on floppy drives only. He quit (or was forced to quit) Shugart Associates in 1974, right before they introduced the Mini-floppy (5.1/4-inch) diskette drive, which of course became the standard eventually used by personal computers, rapidly replacing the 8-inch drives. Shugart Associates also introduced the Shugart Associates System Interface (SASI), which was later renamed Small Computer Systems Interface (SCSI) when it was formally approved by the ANSI committee in 1986. After being forced to leave, Shugart attempted to legally force Shugart Associates to remove his name from the company, but failed. The remnants of Shugart Associates still operates today as Shugart Corporation.

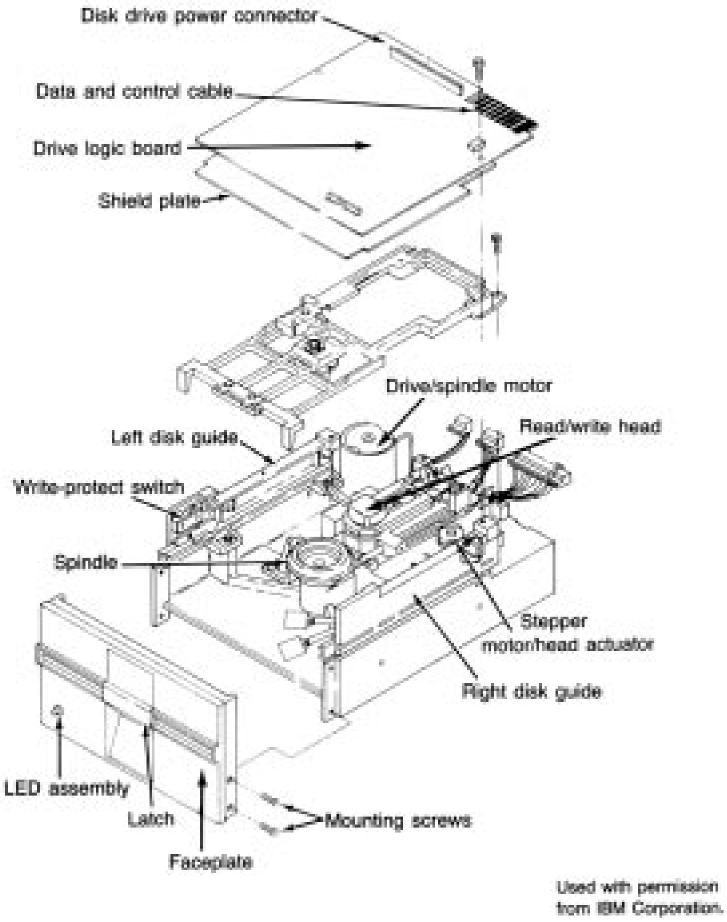
For the next few years, Shugart took time off, ran a bar, and even dabbled in commercial fishing. In 1979, Finis Conner approached Shugart to create and market 5 1/4-inch hard disk drives. Together they founded Seagate Technology and by the end of 1979 had announced the ST-506 (6M unformatted, 5M formatted capacity) drive and interface. This drive is known as the father of all PC hard disk drives. Seagate then introduced the ST-412 (12M unformatted, 10M formatted capacity) drive, which was adopted by IBM for the original XT in 1983. IBM was Seagate's largest customer for many years. Today, Seagate Technology is the largest disk drive manufacturer in the world, second only to IBM.

When you stop to think about it, Alan Shugart has had a tremendous effect on the PC industry. He (or his companies) have created the floppy, hard disk, and SCSI drive and controller interfaces still used today. All PC floppy drives are still based on (and compatible with) the original Shugart designs. The ST-506/412 interface was the de facto hard disk interface standard for many years and served as the basis for the ESDI and IDE interfaces as well. Shugart also created the SCSI interface, used in both IBM and Apple systems today.

As a side note, in the late 80s Finis Conner left Seagate and founded Conner Peripherals, originally wholly owned and funded by Compaq. Conner became Compaq's exclusive drive supplier and gradually began selling drives to other system manufacturers as well. Compaq eventually cut Conner Peripherals free, selling off most (if not all) of their ownership of the company.

Drive Components

This section describes the components that make up a typical floppy drive and examines how these components operate together to read and write data—the physical operation of the drive. All floppy drives, regardless of type, consist of several basic common components. To properly install and service a disk drive, you must be able to identify these components and understand their function (see fig. 13.1).



Used with permission from IBM Corporation.

Fig. 13.1
A typical full-height disk drive.

Read/Write Heads

A floppy disk drive normally has two read/write heads, making the modern floppy disk drive a double-sided drive. A head exists for each side of the disk, and both heads are used for reading and writing on their respective disk sides. At one time, single-sided drives were available for PC systems (the original PC had such drives), but today single-sided drives are a fading memory (see fig. 13.2).

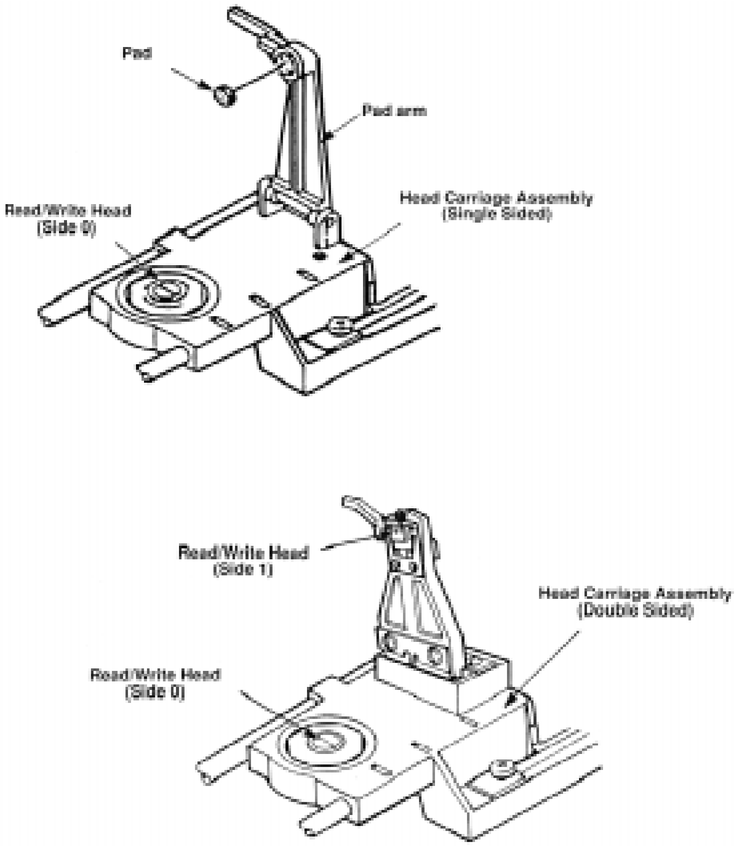


Fig. 13.2
Single- and double-sided drive head assemblies.

Note

Many people do not realize that the first head is the bottom one. Single-sided drives, in fact, use only the bottom head; the top head is replaced by a felt pressure pad (see fig 13.2). Another bit of disk trivia is that the top head (Head 1) is not directly over the bottom head—the top head is located either four or eight tracks inward from the bottom head, depending on the drive type. Therefore, what conventionally are called "cylinders" should more accurately be called "cones."

The head mechanism is moved by a motor called a *head actuator*. The heads can move in and out over the surface of the disk in a straight line to position themselves over various tracks. The heads move in and out tangentially to the tracks that they record on the disk. Because the top and bottom heads are mounted on the same rack, or mechanism, they move in unison and cannot move independently of each other. The heads are made of soft ferrous (iron) compounds with electromagnetic coils. Each head is a composite design, with a record head centered within two tunnel-erase heads in the same physical assembly (see fig. 13.3).

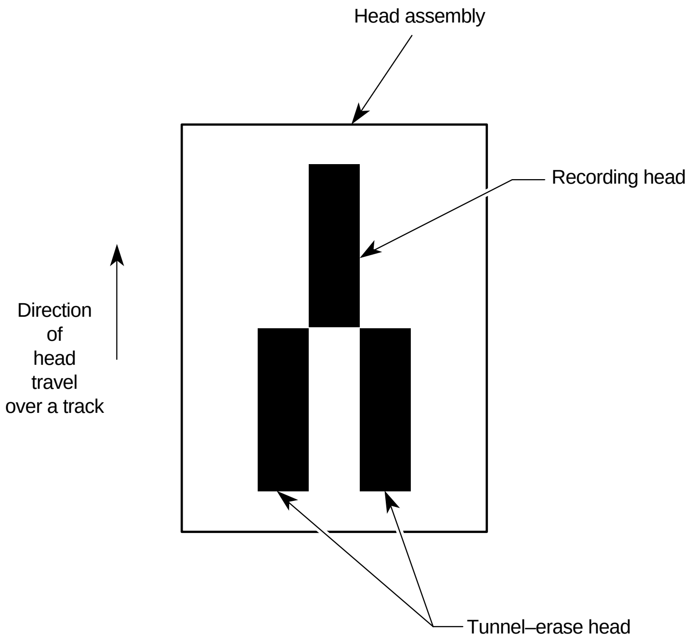


Fig. 13.3
Composite construction of a typical floppy drive head.

The recording method is called *tunnel erasure*; as the track is laid down, the trailing tunnel erase heads erase the outer bands of the track, trimming it cleanly on the disk. The heads force the data to be present only within a specified narrow "tunnel" on each track. This process prevents the signal from one track from being confused with the signals from adjacent tracks. If the signal were allowed to "taper off" to each side, problems would occur. The forcibly trimmed track prevents this problem.

Alignment is the placement of the heads with respect to the tracks they must read and write. Head alignment can be checked only against some sort of reference-standard disk recorded by a perfectly aligned machine. These types of disks are available, and you can use one to check your drive's alignment.

The two heads are spring-loaded and physically grip the disk with a small amount of pressure, which means that they are in direct contact with the disk surface while reading and writing to the disk. Because PC-compatible floppy disk drives spin at only 300 or 360 RPM, this pressure does not present an excessive friction problem. Some newer disks are specially coated with Teflon or other compounds to further reduce friction and enable the disk to slide more easily under the heads. Because of the contact between the heads and the disk, a buildup of the oxide material from the disk eventually forms on the heads. The buildup periodically can be cleaned off the heads as part of a preventive-maintenance or normal service program.

To read and write to the disk properly, the heads must be in direct contact with the media. Very small particles of loose oxide, dust, dirt, smoke, fingerprints, or hair can cause problems with reading and writing the disk. Disk and drive manufacturer's tests have found that a spacing as little as .000032 inches (32 millionths of an inch) between the heads and the media can cause read/write errors. You now can understand why it is important to handle diskettes carefully and avoid touching or contaminating the surface of the diskette media in any way. The rigid jacket and protective shutter for the head access aperture on the 3 1/2-inch disks is excellent for preventing problems with media contamination. 5 1/4-inch disks do not have the same protective elements, therefore more care must be exercised in their handling.

The Head Actuator

The *head actuator* is a mechanical motor device that causes the heads to move in and out over the surface of a disk. These mechanisms for floppy disk drives universally use a special kind of motor, a *stepper motor*, that moves in both directions an amount equal to or less than a single revolution. This type of motor does not spin around continuously; rather, the motor normally moves only a partial revolution in each direction. Stepper motors move in fixed increments, or *detents*, and must stop at a particular detent position. Stepper motors are not infinitely variable in their positioning. Each increment of motion, or a multiple thereof, defines each track on the disk. The motor can be commanded by the disk controller to position itself according to any relative increment within the range of its travel. To position the heads at track 25, for example, the motor is commanded to go to the 25th detent position.

The stepper motor usually is linked to the head rack by a coiled, split steel band. The band winds and unwinds around the spindle of the stepper motor, translating the rotary motion into linear motion. Some drives use a worm gear arrangement rather than a band. With this type, the head assembly rests on a worm gear driven directly off the stepper motor shaft. Because this arrangement is more compact, you normally find worm gear actuators on the smaller 3 1/2-inch drives.

Most stepper motors used in floppy drives can step in specific increments that relate to the track spacing on the disk. Most 48 Track Per Inch (TPI) drives have a motor that steps in increments of 3.6° (degrees). This means that each 3.6° of stepper motor rotation moves the heads from one track (or cylinder) to the next. Most 96 or 135 TPI drives have a stepper motor that moves in 1.8° increments, which is exactly half of what the 48 TPI drives use. Sometimes you see this information actually printed or stamped right on the stepper motor itself, which is useful if you are trying to figure out what type of drive you have. 5 1/4-inch 360K drives are the only 48 TPI drives available and use the 3.6°

increment stepper motor. All other drive types normally use the 1.8° stepper motor. On most drives, the stepper motor is a small cylindrical object near one corner of the drive.

A stepper motor usually has a full travel time of about 1/5 of a second—about 200 milliseconds. On average, a half-stroke is 100 milliseconds, and a one-third stroke is 66 milliseconds. The timing of a one-half or one-third stroke of the head-actuator mechanism often is used to determine the reported average-access time for a disk drive. *Average-access time* is the normal amount of time the heads spend moving at random from one track to another.

The Spindle Motor

The *spindle motor* spins the disk. The normal speed of rotation is either 300 or 360 RPM, depending on the type of drive. The 5 1/4-inch high-density (HD) drive is the only drive that spins at 360 RPM; all others, including the 5 1/4-inch double-density (DD), 3 1/2-inch DD, 3 1/2-inch HD, and 3 1/2-inch extra-high density (ED) drives, spin at 300 RPM. Most earlier drives used a mechanism on which the spindle motor physically turned the disk spindle with a belt, but all modern drives use a direct-drive system with no belts. The direct-drive systems are more reliable and less expensive to manufacture, as well as smaller in size. The earlier belt-driven systems did have more rotational torque available to turn a sticky disk because of the torque multiplication factor of the belt system. Most newer direct-drive systems use an automatic torque-compensation capability that automatically sets the disk-rotation speed to a fixed 300 or 360 RPM and compensates with additional torque for sticky disks or less torque for slippery ones. This type of drive eliminates the need to adjust the rotational speed of the drive.

Most newer direct-drive systems use this automatic-speed feature, but many earlier systems require that you periodically adjust the speed. Looking at the spindle provides you with one clue to the type of drive you have. If the spindle contains strobe marks for 50 Hz and 60 Hz strobe lights (fluorescent lights), the drive probably has an adjustment for speed somewhere on the drive. Drives without the strobe marks almost always include an automatic tachometer-control circuit that eliminates the need for adjustment. The technique for setting the speed involves operating the drive under fluorescent lighting and adjusting the rotational speed until the strobe marks appear motionless, much like the “wagon wheel effect” you see in old Western movies. The procedure is described later in this chapter, in the “Setting the Floppy Drive Speed Adjustment” section.

To locate the spindle-speed adjustment, you must consult the original equipment manufacturer’s (OEM) manual for the drive. IBM provides the information for its drives in the *Technical Reference Options and Adapters* manual as well as in the hardware-maintenance reference manuals. Even if IBM had sold the drives, they most likely are manufactured by another company, such as Control Data Corporation (CDC), Tandon, YE-Data (C. Itoh), Alps Electric, or Mitsubishi. Contact these manufacturers for the original manuals for your drives.

Circuit Boards

A disk drive always incorporates one or more *logic* boards, circuit boards that contain the circuitry used to control the head actuator, read/write heads, spindle motor, disk sensors, and any other components on the drive. The logic board represents the drive's interface to the controller board in the system unit.

The standard interface used by all PC types of floppy disk drives is the Shugart Associates SA-400 interface. The interface, invented by Shugart in the 1970s, has been the basis of most floppy disk interfacing. The selection of this industry-standard interface is the reason that you can purchase "off-the-shelf" drives (raw, or bare, drives) that can plug directly into your controller. (Thanks, IBM, for sticking with industry-standard interfacing; it has been the foundation of the entire PC upgrade and repair industry!)

Some other computer companies making non-IBM-compatible systems (especially Apple, for example) have stayed away from industry standards in this and other areas, which can make tasks such as drive repair or upgrades a nightmare—unless, of course, you buy all your parts from them. For example, in both the Apple II series as well as the Mac, Apple used nonstandard proprietary interfaces for the floppy drives.

The Mac uses an interface based on a proprietary chip called either the IWM (Integrated Woz Machine) or the SWIM (Super Woz Integrated Machine) chip, depending on which Mac you have. These interfaces are incompatible with the industry standard SA-400 interface used in IBM-compatible systems, which is based on the non-proprietary NEC PD765 chip. In fact, the Apple drives use an encoding scheme called GCR (group-coded recording), which is very different from the standard MFM (modified frequency modulation) used in most other systems. The GCR encoding scheme, in fact, cannot be performed by the NEC-type controller chips, which is why it is impossible for IBM-compatible systems to read Mac floppy disks. To Apple's credit, the Mac systems with the SWIM chip include drives that can read and write both GCR and MFM schemes, enabling these systems to read and write IBM floppy disks.

Unfortunately, because the electrical interface to the drive is proprietary, you still cannot easily (or cheaply) purchase these drives as bare units from a variety of manufacturers, as you can for IBM-compatible systems. IBM uses true industry standards in these and other areas, which is why the PC, XT, AT, and PS/2 systems, as well as most IBM-compatible vendors' systems are so open to upgrade and repair.

Logic boards for a drive can fail and usually are difficult to obtain as a spare part. One board often costs more than replacing the entire drive. I recommend keeping failed or misaligned drives that might otherwise be discarded so that they can be used for their remaining good parts—such as logic boards. The parts can be used to restore a failing drive very cost-effectively.

The Faceplate

The *faceplate*, or bezel, is the plastic piece that comprises the front of the drive. These pieces, usually removable, come in different colors and configurations.

Most drives use a bezel slightly wider than the drive. These types of drives must be installed from the front of a system because the faceplate is slightly wider than the hole in the system-unit case. Other drive faceplates are the same width as the drive's chassis; these drives can be installed from the rear—an advantage in some cases. In the later-version XT systems, for example, IBM uses this design in its drives so that two half-height drives can be bolted together as a unit and then slid in from the rear, to clear the mounting-bracket and screw hardware. On occasion, I have filed the edges of a drive faceplate to install the drive from the rear of a system—which sometimes can make installation much easier.

Connectors

Nearly all disk drives have at least two connectors—one for power to run the drive and the other to carry the control and data signals to and from the drive. These connectors are fairly standardized in the computer industry; a four-pin in-line connector (called Mate-N-Lock, by AMP), in both a large and small style is used for power (see fig. 13.4); and a 34-pin connector in both edge and pin header designs are used for the data and control signals. 5 1/4-inch drives normally use the large style power connector and the 34-pin edge type connector, whereas most 3 1/2-inch drives use the smaller version of the power connector and the 34-pin header type logic connector. The drive controller and logic connectors and pinouts are detailed later in this chapter and in Appendix A.

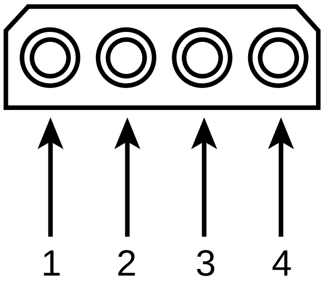


Fig. 13.4
A disk drive female power supply cable connector.

Both the large and small power connectors from the power supply are female plugs. They plug into the male portion, which is attached to the drive itself. One common problem with upgrading an older system with 3 1/2-inch drives is that your power supply only has the large style connectors, whereas the drive has the small style. An adapter cable is available from Radio Shack (Cat. No. 278-765) and other sources that converts the large style power connector to the proper small style used on most 3 1/2-inch drives.

The following chart shows the definition of the pins on the drive power-cable connectors.

Large Power Connector	Small Power Connector	Signal	Wire Color
Pin 1	Pin 4	+12 Vdc	Yellow
Pin 2	Pin 3	Ground	Black
Pin 3	Pin 2	Ground	Black
Pin 4	Pin 1	+5 Vdc	Red

Note that the pin designations are reversed between the large and small style power connectors. Also, it is important to know that not all manufacturers follow the wire color coding properly. I have seen instances in which all the wires are a single color (for example, black), or the wire colors are actually reversed from normal! For example, I once purchased the power connector adapter cables just mentioned that had all the wire colors backwards. This was not really a problem as the adapter cable was wired correctly from end to end, but it was disconcerting to see the red wire in the power supply connector attach to a yellow wire in the adapter (and vice versa)!

Not all drives use the standard separate power and signal connectors. IBM, for example, uses either a single 34-pin or single 40-pin header connector for both power and floppy controller connections in most of the PS/2 systems. In some older PS/2 systems, for example, IBM used a special version of a Mitsubishi 3 1/2-inch 1.44M drive called the MF-355W-99; which has a single 40-pin power/signal connector. Some newer PS/2 systems use a Mitsubishi 3 1/2-inch 2.88M drive called the MF356C-799MA, which uses a single 34-pin header connector for both power and signal connections.

Most standard IBM clone or compatible systems use 3 1/2-inch drives with a 34-pin signal connector and a separate small style power connector. For older PC- or AT-type systems, many drive manufacturers also sell 3 1/2-inch drives installed in a 5 1/4-inch frame assembly and have a special adapter built-in that allows the larger power connector and standard edge type signal connectors to be used. Mitsubishi, for example, sold 1.44M drives as the MF-355B-82 (black faceplate) or MF-355B-88 (beige faceplate). These drives included an adapter that enables the standard large style power connector and 34-pin edge type control and data connector to be used. Because no cable adapters are required and they install in a 5 1/4-inch half-height bay, these types of drives are ideal for upgrading earlier systems. Most 3 1/2-inch drive-upgrade kits sold today are similar and include the drive, appropriate adapters for the power and control and data cables, a 5 1/4-inch frame adapter and faceplate, and rails for AT installations. The frame adapter and faceplate enable the drive to be installed where a 5 1/4-inch half-height drive normally would go.

Drive-Configuration Devices

You must locate several items on a drive that you install in a system. These items control the configuration and operation of the drive and must be set correctly depending on which type of system the drive is installed in and exactly where in the system the drive is installed.

You must set or check the following items during installation:

- Drive select jumper
- Terminating resistor
- Diskette changeline jumper
- Media sensor jumper

You learn how to configure these items later in this chapter. In this section, you learn what function these devices perform.

Drive Select Jumper. Each drive in a controller and drive subsystem must have a unique drive number. The *drive select jumper* is set to indicate to the controller the logical number of a particular drive. The jumper indicates whether the specific drive should respond as drive 0 or 1 (A or B). Some idiosyncrasies can be found when you set this jumper in various systems because of strange cable configurations or other differences. Most drives allow four different settings, labeled DS1, DS2, DS3, and DS4. Some drives start with 0, and thus the four settings are labeled DS0, DS1, DS2, and DS3. On some drives, these jumpers are not labeled. If they are not labeled, you have several resources available for information about how to set the drive: the OEM manual, your experience with other similar drives, or simply an educated guess. First check the manual if it is available; if not, then perhaps you can rely on your past experiences and make an educated guess.

You might think that the first drive select position corresponds to A and that the second position corresponds to B, but in most cases you would be wrong. The configuration that seems correct is wrong because of some creative rewiring of the cable. IBM, for example, crosses the seven wires numbered 10 through 16 (the drive select, motor enable, and some ground lines) in the floppy interface cable between drives B: and A: to allow both drives to be jumpered the same way, as though they both were drive B:. This type of cable is shown later in this chapter.

When using a floppy cable with lines 10 through 16 twisted, you must set the DS jumper on both drives to the second (same) drive select position. This setup enabled dealers and installers to buy the drives pre-configured by IBM and install them with a minimum of hassle. Sometimes this setup confuses people who attempt to install drives properly without knowing about the twisted-cabling system. With this type of cable design, to swap drive A and B positions you can simply swap the cable connections between the drives with no jumper setting changes required. You still probably have to change the terminator resistor settings on the drives, however.

If the cable has a straight-through design, in which lines 10 through 16 are not twisted between the B and A connectors, you then need to jumper the drives as you might have thought originally—that is, drive A is set to the first drive select position, and drive B is set to the second drive select position.

If you install a drive and it either does not respond or responds in unison with the other drive in the system in calling for drive A, you probably have the drive select jumpers set incorrectly for your application. Check the DS jumpers on both drives as well as the cable for the proper setup.

Terminating Resistor. Any signal carrying electronic media or cable with multiple connections can be thought of as an electrical bus. In almost all cases, a bus must be terminated properly at each end with Terminating Resistors to allow signals to travel along the bus error free. Terminating resistors are designed to absorb any signals that reach the end of a cabling system or bus so that no reflection of the signal echoes, or bounces, back down the line in the opposite direction. Engineers sometimes call this effect *signal ringing*. Simply put, noise and distortion can disrupt the original signal and prevent proper communications between the drive and controller. Another function of proper termination is to place the proper resistive load on the output drivers in the controller and drive.

Most floppy disk cabling systems have the controller positioned at one end of the cable, and a terminating resistor network is built-in to the controller to properly terminate that end of the bus. IBM and IBM clone or compatible systems use a floppy controller design that normally allows up to two drives on a single cable. To terminate the other end of the bus properly, a terminating resistor must be set, or enabled, on the drive at the end of the cable farthest from the controller. In most systems, this drive also is the lowest-lettered drive (A) of the pair. The drive plugged into the connector in the center of the cable must have the terminating resistor (or terminator) removed or disabled for proper operation.

Because a terminating resistor is already installed in the controller to terminate the cable at that end, be concerned only with properly terminating the drive end. Sometimes a system operates even with incorrect terminator installation or configuration, but the system might experience sporadic disk errors. Additionally, with the wrong signal load on the controller and drives, you run the risk of damaging them by causing excessive power output because of an improper resistive load.

A terminating resistor usually looks like a memory chip—a 16-pin *dual in-line package* (DIP) device. The device is actually a group of eight resistors physically wired in parallel with each other to terminate separately each of the eight data lines in the interface subsystem. Normally, this “chip” is a different color from other black chips on the drive. Orange, blue, or white are common colors for a terminating resistor. Be aware that not all drives use the same type of terminating resistor, however, and it might be physically located in different places on different manufacturer's drive models. The OEM manual for the drive comes in handy in this situation because it shows the location, physical appearance, enabling and disabling instructions, and even the precise value required for the resistors. Do not lose the terminator if you remove it from a drive; you might need to reinstall it later if you relocate the drive to a different position in a system or even to a different system. Some drives use a resistor network in a *single in-line pin* (SIP) package, which looks like a slender device with eight or more pins in a line. Most terminating resistors have resistance values of 150 to 330 ohms.

Nearly all 5 1/4-inch floppy drives use a chip style terminator that must be physically installed or removed from the drive as required by the drive's position on the cable. Some 5 1/4-inch drives, especially those made by Toshiba, have a permanently installed terminating resistor enabled or disabled by a jumper labeled TR (Terminating Resistor). I prefer the jumper setup because you never have to actually remove (and possibly lose) the terminating resistor itself. Because some drives use different style and value resistors, if one is lost it can be a hassle to obtain a replacement of the correct type.

Most 3 1/2-inch drives have permanently installed, non-configurable terminating resistors. This setup is the best possible because you never have to remove or install them, and there are never any TR jumpers to set. Although some call this technique automatic termination, technically the 3 1/2-inch drives use a technique called *distributed termination*. With distributed termination, each 3 1/2-inch drive has a much higher value (1,000 to 1,500 ohm) terminating resistor permanently installed, and therefore carries a part of the termination load. These terminating resistors are fixed permanently to the drive and never have to be removed. This feature makes configuring these drives one step simpler. Because each drive is connected in parallel on the floppy cabling system, the total equivalent resistive load is calculated by the following formula:

$$1/R = 1/R1 + 1/R2 + 1/R3 + \text{etc.}$$

In a typical setup, a 330-ohm terminating resistor is permanently installed on the floppy controller, so the calculations for typical systems with one or two 3 1/2-inch drives (with built-in 1,500-ohm terminating resistors) is as follows:

$$1/R = 1/330 + 1/1{,}500 + 1/1{,}500 \text{ so } R = 229.17 \text{ ohms (two drives)}$$

$$1/R = 1/330 + 1/1{,}500 \text{ so } R = 270.5 \text{ ohms (one drive)}$$

Some confusion exists regarding the situation in which you have both 5 1/4-inch and 3 1/2-inch drives installed on a single cable. In this case, the 5 1/4-inch drive must have its terminating resistor configured as appropriate, which is to say either leave it installed or remove it depending on whether the drive is at the end of the cable. Nothing with regards to termination is done with the 3 1/2-inch drive because the terminating resistors are non-configurable. Although it sounds like mixing these might cause a problem with the termination, actually it works out quite well as the math shows. In this example, 33-ohm resistors are used both on the floppy controller and the 5 1/4-inch drive:

1/R = 1/330 + 1/1,500 so R = 270.5 ohms (3 1/2-inch drive at cable end)

1/R = 1/330 + 1/330 so R = 165 ohms (5 1/4-inch drive at cable end)

As long as a terminating resistor is at each end of the cable and the total equivalent resistance is between 100 to 300 ohms, the termination is normally considered correct. The drive in the center cable connector should not have a terminating resistor unless it employs distributed termination and has a resistor value of 1,000 ohms or higher. Note that in many cases even if the termination is improper a system seems to work fine, although the likelihood of read and write errors is greatly increased. In some cases, the drives do not work properly at all unless termination is properly configured. I have solved many intermittent floppy drive problems by correcting improper terminating resistor configurations, found even in brand new systems.

Diskette Changeline Jumper. In an XT-type system, pin 34 of the disk drive interface is not used, but in an AT system this pin is used to carry a signal called *Diskette Changeline*, or DC. Although the XT does not use pin 34, most drives that don't support the DC signal can optionally use the pin to carry a signal called *Ready* (usually labeled on the drive as RY, RDY, or SR). The Ready signal is actually never used in any IBM or IBM-compatible systems and should not be configured or in most cases the drive will not work properly.

The AT uses the Diskette Changeline signal to determine whether the disk has been changed, or more accurately, whether the same disk loaded during the previous disk access still is loaded in the floppy drive. *Disk Change* is a pulsed signal that changes a status register in the controller to let the system know that a disk has been either inserted or ejected. This register is set to indicate that a disk has been inserted or removed (changed) by default. The register is cleared when the controller sends a step pulse to the drive and the drive responds, acknowledging that the heads have moved. At this point, the system knows that a specific disk is in the drive. If the disk change signal is not received before the next access, the system can assume that the same disk is still in the drive. Any information read into memory during the previous access therefore can be reused without rereading the disk.

Because of this process, some systems can buffer or cache the contents of the file allocation table or directory structure of a disk in the system's memory. By eliminating unnecessary rereads of these areas of the disk, the apparent speed of the drive is increased. If you move the door lever or eject button on a drive that supports the disk change signal, the DC pulse is sent to the controller, thus resetting the register indicating that the disk has been changed. This procedure causes the system to purge buffered or cached data that had been read from the disk because the system then cannot be sure that the same disk is still in the drive.

AT class systems use the DC signal to increase significantly the speed of the floppy interface. Because the AT can detect whether you have changed the disk, the AT can keep a copy of the disk's directory and file allocation table information in RAM buffers. On every subsequent disk access, the operations are much faster because the information does not have to be reread from the disk in every individual access. If the DC signal has been reset (has a value of 1), the AT knows that the disk has been changed and appropriately rereads the information from the disk.

You can observe the effects of the DC signal by trying a simple experiment. Boot DOS on an AT-class system and place a formatted floppy disk with data on it in drive A. Drive A can be any type of drive except 5 1/4-inch double-density, although the disk you use can be anything the drive can read, including a double-density 360K disk, if you want. Then type the following command:

```
DIR A:
```

The disk drive lights up, and the directory is displayed. Note the amount of time spent reading the disk before the directory is displayed on-screen. Without touching the drive, enter the **DIR A:** command again, and watch the drive-access light and screen. Note again the amount of time that passes before the directory is displayed. The drive A directory should appear almost instantly the second time because virtually no time is spent actually reading the disk. The directory information was simply read back from RAM buffers or cache rather than read again from the disk. Now open and close the drive door and keep the same disk in the drive. Type the **DIR A:** command again. The disk again takes some time reading the directory before displaying anything because the AT "thinks" that you changed the disk.

The PC and XT controllers (and systems) are not affected by the status of the DC signal. These systems "don't care" about signals on pin 34. The PC and XT systems always operate under the assumption that the disk is changed before every access, and they reread the disk directory and file allocation table each time—one reason that these systems are slower in using the floppy disk drives.

A problem can occur when certain drives are installed in an AT system. As mentioned, some drives use pin 34 for a "Ready" signal. The RDY signal is sent when a disk is installed and rotating in the drive. If you install a drive that has pin 34 set to send RDY, the AT "thinks" that it is continuously receiving a disk change signal, which causes problems: usually the drive fails with a `Drive not ready` error and is inoperable.

A different but related problem occurs if the drive is not sending the DC signal on pin 34, and it should. If an AT class system is told (through CMOS setup) that the drive is any other type than a 360K (which cannot ever send the DC signal), then the system expects the drive to send DC whenever a disk has been ejected. If the drive is not configured properly to send the signal, then the system never recognizes that a disk has been changed. Therefore, even if you do change the disk, the AT still acts as though the first disk is in the drive and holds the first disk's directory and file allocation table information in RAM. This can be dangerous as the File Allocation Table (FAT) and directory information from the first disk can be partially written to any subsequent disks written to in the drive.

If you ever have seen an AT class system with a floppy drive that shows "phantom directories" of the previously installed disk, even after you have changed or removed it, you have experienced this problem firsthand. The negative side effect is that all disks after the first one you place in this system are in extreme danger. You likely will overwrite the directories and file allocation tables of many disks with information from the first disk. Data recovery from such a catastrophe can require quite a bit of work with utility programs such as the Norton Utilities. These problems with Disk Change most often are traced to an incorrectly configured drive. Another possibility is that the disk-eject sensor mechanism no longer operates correctly. A temporary solution to the problem is to press the Ctrl-Break or Ctrl-C key combination every time you change a floppy disk in the drive. These commands cause DOS to flush the RAM buffers manually and reread the directory and file allocation table during the next disk access.

All drives except 5 1/4-inch double-density (360K) drives support the Disk Change signal. Therefore, if your system thinks that one of these drives is installed, the drive is expected to provide the signal. If the system thinks that the installed drive is a 360K drive, no signal is expected on pin 34.

To summarize, PC and XT systems are not affected by pin 34, but on AT systems, non-360K drives must have pin 34 set to send Disk Change. If the drive is a 360K drive and you want to install it in an AT, pin 34 must be disabled (usually preconfigured as such, or set by removing a jumper). Never set a 360K drive (or any other drive for that matter) to send a signal called Ready (RDY) on pin 34 because IBM-compatible systems cannot use this signal. The only reason that the Ready signal exists on some drives is that it happens to be a part of the standard Shugart SA-400 disk interface that was not adopted by IBM.

Media Sensor Jumper. This configuration item exists only on the 3 1/2-inch 1.44M or 2.88M drives. The jumper selection, called the *media sensor (MS) jumper*, must be set to enable a special media sensor in the disk drive, which senses a media sensor hole found only in the 1.44M high-density and the 2.88M extra-high density floppy disks. The labeling of this jumper (or jumpers) varies greatly between different drives. In many drives, the jumpers are permanently set (enabled) and cannot be changed.

The three types of configurations with regards to media sensing are as follows:

- No media sense (sensor disabled or no sensor present)
- Passive media sense (sensor enabled)
- Active or intelligent media sense (sensor supported by Controller/BIOS)

Most systems use a *passive media sensor* arrangement. The passive media sensor setup enables the drive to determine the level of recording strength to use and is required for most installations of these drives because of a bug in the design of the Western Digital hard disk and floppy controllers used by IBM in the AT systems. This bug prevents the controller from properly instructing the drive to switch to double-density mode when you write or format double-density disks. With the media sensor enabled, the drive no longer depends on the controller for density mode switching and relies only on the drive's media sensor. Unless you are sure that your disk controller does not have this flaw, make sure that your HD drive includes a media sensor (some older or manufacturer-specific drives do not), and that it is properly enabled. The 2.88M drives universally rely on media sensors to determine the proper mode of operation. The 2.88M drives, in fact, have two separate media sensors because the ED disks include a media sensor hole in a different position than the HD disks.

With only a few exceptions, high-density 3 1/2-inch drives installed in most IBM-compatible systems do not operate properly in double-density mode unless the drive has control over the write current (recording level) via an installed and enabled media sensor. Exceptions are found primarily in systems with floppy controllers integrated on the motherboard, including most older IBM PS/2 and Compaq systems as well as most laptop or notebook systems from other manufacturers. These systems have floppy controllers without the bug referred to earlier, and can correctly switch the mode of the drive without the aid of the media sensor. In these systems, it technically does not matter whether you enable the media sensor. If the media sensor is enabled, the drive mode is controlled by the disk you insert, as is the case with most IBM-compatible systems. If the media sensor is not enabled, the drive mode is controlled by the floppy controller, which in turn is controlled by DOS.

If a disk is already formatted (correctly), DOS reads the volume boot sector to determine the current disk format, and the controller then switches the drive to the appropriate mode. If the disk has not been formatted yet, DOS has no idea what type of disk it is, and the drive remains in its native HD or ED mode.

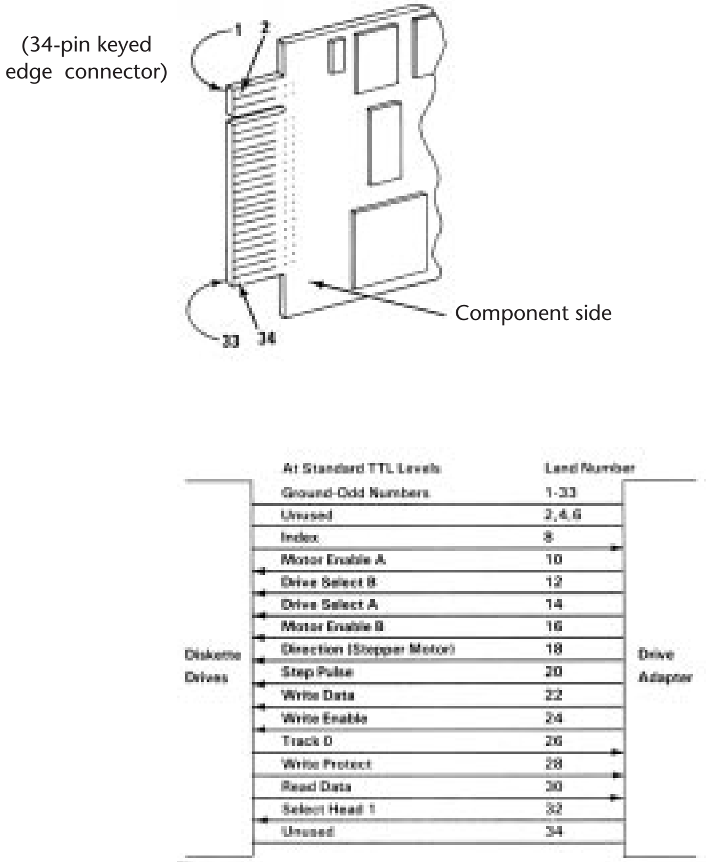
When you format a disk in systems without an enabled media sensor (such as most PS/2s), the mode of the drive depends entirely on the FORMAT command issued by the user, regardless of the type of disk inserted. For example, if you insert a DD disk into an HD drive in an IBM PS/2 Model 70 and format the disk by entering FORMAT A:, the disk is formatted as though it is an HD disk because you did not issue the correct parameters to cause the FORMAT command to specify a DD format. On a system with the media sensor enabled, this type of incorrect format fails, and you see the Invalid media or Track 0 bad error message from FORMAT. In this case, the media sensor prevents an incorrect format from occurring on the disk, a safety feature most older IBM PS/2 systems lack.

Most of the newer PS/2 systems, including all those that come standard with the 2.88M drives, have what is called an active or intelligent media sensor setup. This means that the sensor not only detects what type of disk is in the drive and changes modes appropriately, but also the drive informs the controller (and the BIOS) about what type of disk is in the drive. Systems with an intelligent media sensor do not need to use the disk type parameters in the FORMAT command. In these systems, the FORMAT command automatically “knows” what type of disk is in the drive, and formats it properly. With an intelligent media sensor, you never have to know what the correct format parameters are for a particular type of disk, the system figures it out for you automatically. Many high-end systems such as the newer PS/2 systems as well as high-end Hewlett Packard PCs have this type of intelligent media sensor arrangement.

The Floppy Disk Controller

The floppy disk controller consists of the circuitry either on a separate adapter card or integrated on the motherboard, which acts as the interface between the floppy drives and the system. Most PC- and XT-class systems used a separate controller card that occupied a slot in the system. The AT systems normally had the floppy controller and hard disk controller built into the same adapter card and also plugged into a slot. In most of the more modern systems built since then, the controller is integrated on the motherboard. In any case, the electrical interface to the drives has remained largely static, with only a few exceptions.

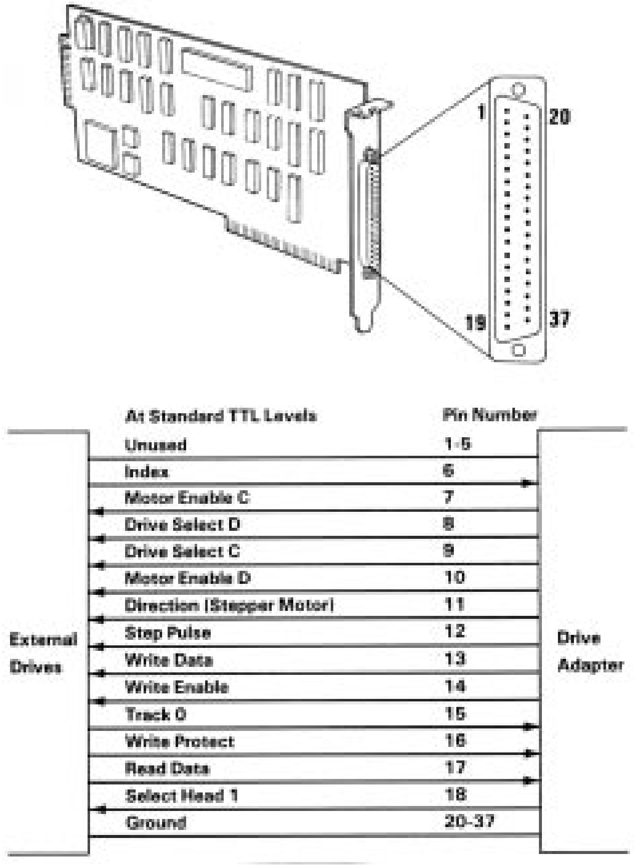
The original IBM PC and XT system floppy controller was a 3/4-length card that could drive as many as four floppy disk drives. Two drives could be connected to a cable plugged into a 34-pin edge connector on the card, and two more drives could be plugged into a cable connected to the 37-pin connector on the bracket of this card. Figures 13.5 and 13.6 show these connectors and the pinouts for the controller.



At Standard TTL Levels	Land Number
Ground-Odd Numbers	1-33
Unused	2, 4, 6
Index	8
Motor Enable A	10
Drive Select B	12
Drive Select A	14
Motor Enable B	16
Direction (Stepper Motor)	18
Step Pulse	20
Write Data	22
Write Enable	24
Track 0	26
Write Protect	28
Read Data	30
Select Head 1	32
Unused	34

Fig. 13.5

A PC and XT floppy controller internal connector.



At Standard TTL Levels	Pin Number
Unused	1-5
Index	6
Motor Enable C	7
Drive Select D	8
Drive Select C	9
Motor Enable D	10
Direction (Stepper Motor)	11
Step Pulse	12
Write Data	13
Write Enable	14
Track 0	15
Write Protect	16
Read Data	17
Select Head 1	18
Ground	20-37

Fig. 13.6
A PC and XT floppy controller external connector.

The AT used a board made by Western Digital, which included both the floppy and hard disk controllers in a single adapter. The connector location and pinout for the floppy controller portion of this card is shown in figure 13.7. IBM used two variations of this controller during the life of the AT system. The first one was a full 4.8 inches high, which used all the vertical height possible in the AT case. This board was a variation of the Western Digital WD1002-WA2 controller, sold through distributors and dealers. The second-generation card was only 4.2 inches high, which enabled it to fit into the shorter case of the XT-286 as well as the taller AT cases. This card was equivalent to the Western Digital WD1003-WA2, also sold on the open market.

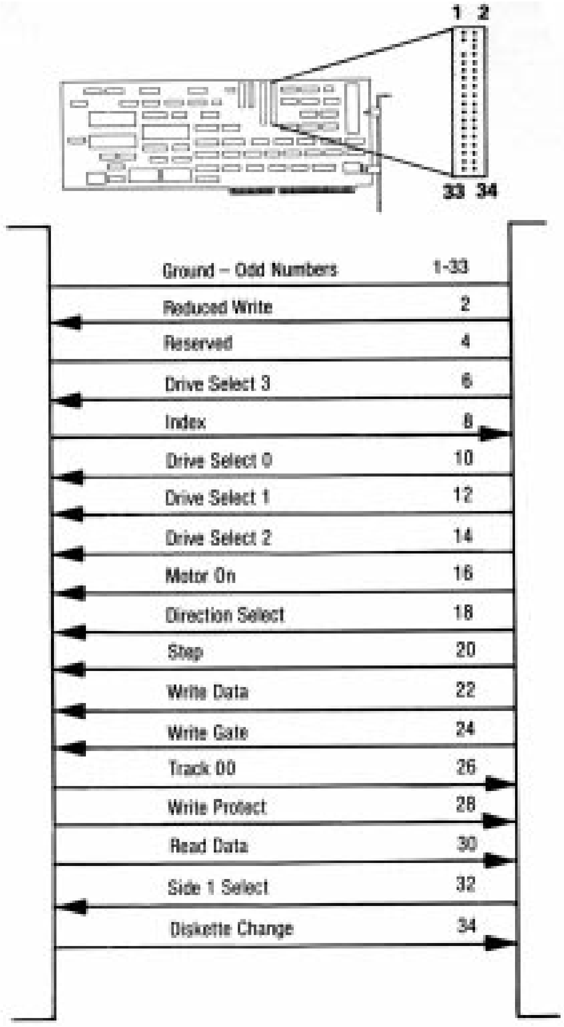


Fig. 13.7
An AT floppy controller connector.

Disk Physical Specifications and Operation

PC-compatible systems now use one of as many as five standard types of floppy drives. Also, five types of disks can be used in the drives. This section examines the physical specifications and operations of these drives and disks.

Drives and disks are divided into two classes: 5 1/4-inch and 3 1/2-inch. The physical dimensions and components of a typical 5 1/4-inch disk and a 3 1/2-inch disk are shown later in this chapter.

The physical operation of a disk drive is fairly simple to describe. The disk rotates in the drive at either 300 or 360 RPM. Most drives spin at 300 RPM, only the 5 1/4-inch 1.2M drives spin at 360 RPM (even when reading or writing 360K disks). With the disk spinning, the heads can move in and out approximately one inch and write either 40 or 80 tracks. The tracks are written on both sides of the disk and therefore sometimes are called *cylinders*. A single cylinder comprises the tracks on the top and bottom of the disk. The heads record by using a tunnel-erase procedure in which a track is written to a specified width and then the edges of the track are erased to prevent interference with any adjacent tracks.

The tracks are recorded at different widths for different drives. Table 13.1 shows the track widths in both millimeters and inches for the five types of floppy drives supported in PC systems.

Table 13.1 Floppy Drive Track-Width Specifications

Drive Type	No. of Tracks	Track Width	
5 1/4-inch 360K	40 per side	0.300 mm	0.0118 in.
5 1/4-inch 1.2M	80 per side	0.155 mm	0.0061 in.
3 1/2-inch 720K	80 per side	0.115 mm	0.0045 in.
3 1/2-inch 1.44M	80 per side	0.115 mm	0.0045 in.
3 1/2-inch 2.88M	80 per side	0.115 mm	0.0045 in.

The differences in recorded track width can result in data-exchange problems between 5 1/4-inch drives. The 5 1/4-inch drives are affected because the double-density drives record a track width nearly twice that of the high-density drives. A problem occurs, therefore, if a high-density drive is used to update a double-density disk with previously recorded data on it.

Even in 360K mode, the high-density drive cannot completely overwrite the track left by an actual 360K drive. A problem occurs when the disk is returned to the person with the 360K drive: that drive reads the new data as embedded within the remains of the previously written track. Because the drive cannot distinguish either signal, an `Abort, Retry, Ignore` error message appears on-screen. The problem does not occur if a new disk (one that never has had data recorded on it) is first formatted in a 1.2M drive with the /4 option, which formats the disk as a 360K disk.

Note

You also can format a 360K disk in a 1.2M drive with the /N:9, /T:40, or /F:360 options, depending on the DOS version. The 1.2M drive then can be used to fill the brand-new and newly formatted 360K disk to its capacity, and every file will be readable on the 40-track, 360K drive.

I use this technique all the time to exchange data disks between AT systems that have only the 1.2M drive and XT or PC systems that have only the 360K drive. The key is to start with either a new disk or one wiped clean magnetically by a bulk eraser or degaussing tool. Just reformatting the disk does not work by itself because formatting does not actually erase a disk; instead it records data across the entire disk.

In addition to a track-width specification, specifications exist for the precise placement of tracks on a disk. A 5 1/4-inch DD disk has tracks placed precisely 1/48th of an inch apart. The outermost track on side 0 (the bottom of the disk) is the starting point for measurements, and this track (cylinder 0, head 0) has a radius of exactly 2.25 inches. Because Head 1 (the top of the disk) is offset by four tracks inward from Head 0, the radius of cylinder 0, Head 1 is 2.2500 inches – (1/48 * 4) = 2.1667 inches. Therefore, to calculate the exact track radius R in inches for any specified cylinder C and head position on a 360K disk, use these formulas:

For Head 0 (bottom): R = 2.2500 inches – C/48 inches

For Head 1 (top): R = 2.1667 inches – C/48 inches

That the tracks on top of the disk (Head 1) are offset toward the center of the disk from the tracks on the bottom of the disk (Head 0) might be surprising: in effect, the cylinders are cone shaped. Figure 13.8 shows the physical relationship between the top and bottom heads on a floppy drive. In this figure, both heads are positioned at the same cylinder. You can see that the top track of the cylinder is closer to the center of the disk than is the bottom track, resulting in cylinders shaped more like cones.

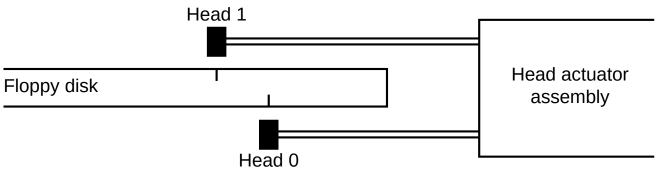


Fig. 13.8
Floppy disk drive head offset.

I first saw this track positioning in one of my data-recovery seminars. One of the experiments I perform in these courses is to stick a pin through a disk with data on it. The objective then is to recover as much data as possible from the disk. Normally, I can resolve the damage down to only a few unreadable sectors on either side of the disk and then easily recover all but these damaged sectors. As I was trying to determine the exact location of the hole by its cylinder, head, and sector coordinates, I noticed that I always had what seemed to be two groups of damaged sectors that were always located exactly four or eight (depending on the type of disk) cylinders apart. Because I had stuck the pin straight through the disk, I realized that any offset had to be in the tracks themselves. I then removed the disk from its jacket to look more closely at the holes.

When I want to “see” the tracks on a disk, I use a special type of solution called Magnetic Developer, a fine-powdered iron suspended in a trichloroethane solution. When this developer, which dries very quickly, is sprayed on the disk, the iron particles align themselves directly over magnetized areas of the disk and show very graphically the exact physical appearance and location of the tracks and sectors on the disk. You can see every individual track and sector on the disk “develop” right before your eyes! With a magnifier or low-powered microscope, I can locate the exact sectors and tracks damaged by the hole on either side of the disk. With this technique, it was easy to see that the tracks on top of the disk start and end farther toward the center of the disk. If you want to do similar experiments or simply “see” the magnetic image of your disks, you can obtain the Magnetic Developer solution from Sprague Magnetics (the address and phone number are in the “Vendor List” in Appendix B). By the way, after viewing this magnetic image, the disk "cookie" can be washed with distilled water or pure trichloroethane, placed back into a new jacket, and reused!

The high-density 5 1/4-inch disk track dimensions are similar to the double-density disk, except that the tracks are spaced precisely 1/96-inch apart, and the top head (Head 1) is offset eight tracks inward from the bottom head (Head 0). The physical head offset between Head 0 and Head 1 is the same as DD disks because twice as many tracks are in the same space as on a DD disk. The calculations for a given track radius R in inches for any cylinder C and head are as follows:

For Head 0 (bottom): R = 2.2500 inches – C/96 inches

For Head 1 (top): R = 2.1667 inches – C/96 inches

All the different 3 1/2-inch disks (DD, HD, ED) are dimensionally the same in track and cylinder spacing. The track and cylinder dimensions of these disks start with the radius of Cylinder 0, Head 0 (the bottom head outer track) defined as 39.5 millimeters. Tracks inward from this track are spaced precisely 0.1875 millimeters apart, and the top head (Head 1) is offset inward by eight tracks from Head 0. The radius, in millimeters, of the outer track on top of the disk (Cylinder 0, Head 1) therefore can be calculated as follows:

39.5 mm – (0.1875 * 8) = 38.0 mm

Now you can calculate the radius R in millimeters of any specified cylinder C and head using these formulas:

For Head 0 (bottom): R = 39.5mm – (0.1875mm * C)

For Head 1 (top): R = 38.0mm – (0.1875mm * C)

An interesting note about the dimensions of the 3 1/2-inch disks is that the dimensional standards all are based in the metric system, unlike the 5 1/4-inch disks. I could have converted these numbers to their English equivalents for comparison with the 5 1/4-inch disk figures, but rounding would have sacrificed some accuracy in the numeric conversion. For example, many texts often give the track spacing for 3 1/2-inch disks as 135 TPI (tracks per inch). This figure is an imprecise result of metric-to-English conversion and rounding. The true spacing of 0.1875 mm between tracks converts to a more precise figure of 135.4667 TPI. The figures presented here are the specifications as governed by the ANSI standards X3.125 and X3.126 for 360K and 1.2M disks, and by Sony, Toshiba, and Accurite, which all are involved in specifying 3 1/2-inch disk standards.

Disk Magnetic Properties

A subtle problem with the way a disk drive works magnetically is that the recording volume varies depending on the type of format you are trying to apply to a disk. The high-density formats use special disks that require a much higher volume level for the recording than do the double-density disks. My classes nearly always are either stumped or incorrect (unless they have read ahead in the book) when they answer this question: "Which type of disk is magnetically more sensitive: a 1.2M disk or a 360K disk?" If you answer that the 1.2M disk is more sensitive, you are wrong! The high-density disks are approximately half as sensitive magnetically as the double-density disks.

The high-density disks are called *high-coercivity disks* also because they require a magnetic field strength much higher than do the double-density disks. Magnetic field strength is measured in *oersteds*. The 360K floppy disks require only a 300-oersted field strength to record, and the high-density 1.2M disks require a 600-oersted field strength. Because the high-density disks need double the magnetic field strength for recording, you should not attempt to format a 1.2M high-density disk as though it were a 360K disk, or a 360K disk as though it were a 1.2M high-density disk.

The latter case in particular seems to appeal to people looking for an easy way to save money. They buy inexpensive 360K disks and format them in a 1.2M drive, to the full 1.2M capacity. Most of the time, this format seems to work, with perhaps a large amount of bad sectors; otherwise, most of the disk might seem usable. You should not store important data on this incorrectly formatted disk, however, because the data is recorded at twice the recommended strength and density. Eventually, the adjacent magnetic domains on the disk begin to affect each other and can cause each other to change polarity or weaken because of the proximity of these domains, and because the double-density disk is more sensitive to magnetic fields. This process is illustrated later in this chapter, in the "Media Coercivity and Thickness" section. Eventually, the disk begins to erase itself and deteriorates. The process might take weeks, months, or even longer, but the result is inevitable—a loss of the information stored on the disk.

Another problem results from this type of improper formatting: You can imprint the 360K disk magnetically with an image that is difficult to remove. The high-density format places on the disk a recording at twice the strength it should be. How do you remove this recording and correct the problem? If you attempt to reformat the disk in a 360K drive, the drive writes in a reduced write-current mode and in some cases cannot overwrite the higher-volume recorded image you mistakenly placed on the disk. If you attempt to reformat the disk in the high-density drive with the /4 (or equivalent) parameter, which indicates 360K mode, the high-density drive uses a reduced write-current setting and again cannot overwrite the recording.

You can correct the problem in several ways. You can throw away the disk and write it off as a learning experience, or you can use a bulk eraser or degaussing tool to demagnetize the disk. These devices can randomize all the magnetic domains on a disk and return it to an essentially factory-new condition. You can purchase a bulk-erasing device at electronic-supply stores for about $25.

The opposite problem with disk formatting is not as common, but some have tried it anyway: formatting a high-density disk with a double-density format. You should not (and normally cannot) format a 1.2M high-density disk to a 360K capacity. If you attempt to use one, the drive changes to reduced write-current mode and does not create a magnetic field strong enough to record on the "insensitive" 1.2M disk. The result in this case is normally an immediate error message from the FORMAT command: `Invalid media or Track 0 bad - disk unusable`. Fortunately, the system usually does not allow this particular mistake to be made.

The 3 1/2-inch drives don't have the same problems as the 5 1/4-inch drives—at least for data interchange. Because both the high-density and double-density drives write the same number of tracks and these tracks are always the same width, no problem occurs when one type of drive is used to overwrite data written by another type of drive. A system manufacturer therefore doesn't need to offer a double-density version of the 3 1/2-inch drive for systems equipped with the high-density or extra-high-density drive. The HD and ED drives can perfectly emulate the operations of the 720K DD drive, and the ED drive can perfectly emulate the 1.44M HD drive.

The HD and ED drives can be trouble, however, for inexperienced users who try to format disks to incorrect capacities. Although an ED drive can read, write, and format DD, HD, and ED disks, a disk should be formatted and written at only its specified capacity. An ED disk therefore should be formatted only to 2.88M, and never to 1.44M or 720K. *You must always use a disk at its designated format capacity*. You are asking for serious problems if you place a 720K disk in the A drive of a PS/2 Model 50, 60, 70, or 80 and enter FORMAT A:. This step causes a 1.44M format to be written on the 720K disk, which renders it unreliable at best and requires a bulk eraser to reformat it correctly. If you decide to use the resulting incorrectly formatted disk, you will eventually have massive data loss.

This particular problem could have been averted if IBM had used media sensor drives in all PS/2 systems. Drives that use the disk media-sensor hole to control the drive mode are prevented from incorrectly formatting a disk. The hardware causes the FORMAT command to fail with an appropriate error message if you attempt to format the disk to an incorrect capacity.

By knowing how a drive works physically, you can eliminate most of these user "pilot error" problems and distinguish this kind of easily solved problem from a more serious hardware problem. You will be a much better user as well as a troubleshooter of a system if you truly understand how a drive works.

Logical Operation

Each type of drive can create disks with different numbers of sectors and tracks. This section examines how DOS sees a drive. It gives definitions of the drives according to DOS and the definitions of cylinders and clusters.

How DOS Uses a Disk. A technical understanding of the way DOS maintains information on your disks is not necessary to use a PC, but you will be a more informed user if you understand the general principles.

To DOS, data on your PC disks is organized in tracks and sectors. Tracks are narrow, concentric circles on a disk. Sectors are pie-shaped slices of the disk. DOS versions 1.0 and 1.1 read and write 5 1/4-inch double-density disks with 40 tracks (numbered 0 through 39) per side and eight sectors (numbered 1 through 8) per track. DOS versions 2.0 and higher automatically increase the track density from eight to nine sectors, for greater capacity on the same disk. On an AT with a 1.2M disk drive, DOS 3.0 supports high-density 5 1/4-inch drives that format 15 sectors per track and 80 tracks per side; DOS 3.2 supports 3 1/2-inch drives that format 9 sectors per track and 80 tracks per side; DOS 3.3 supports 3 1/2-inch drives that format 18 sectors per track and 80 tracks per side. The distance between tracks and, therefore, the number of tracks on a disk is a built-in mechanical and electronic function of the drive. Tables 13.2 and 13.3 summarize the standard disk formats supported by DOS version 5.0 and higher.

Table 13.2 5 1/4-Inch Floppy Disk Drive Formats

	Double-Density 360K (DD)	**High-Density 1.2M (HD)**
Bytes per Sector	512	512
Sectors per Track	9	15
Tracks per Side	40	80
Sides	2	2
Capacity (KBytes)	360	1,200
Capacity (Megabytes)	0.352	1.172
Capacity (Million Bytes)	0.369	1.229

Table 13.3 3 1/2-Inch Floppy Disk Drive Formats

	Double-Density 720K (DD)	**High-Density 1.44M (HD)**	**Extra-High Density 2.88M (ED)**
Bytes per Sector	512	512	512
Sectors per Track	9	18	36
Tracks per Side	80	80	80
Sides	2	2	2
Capacity (KBytes)	720	1,440	2,880
Capacity (Megabytes)	0.703	1.406	2.813
Capacity (Million Bytes)	0.737	1.475	2.949

You can calculate the capacity differences between different formats by multiplying the sectors per track by the number of tracks per side together with the constants of two sides and 512 bytes per sector. Note that the disk capacity can actually be expressed in three different ways. The most common method is to refer to the capacity of a floppy by the number of KBytes (1,024 bytes equals 1K). This works fine for 360K and 720K disks, but is strange when applied to the 1.44M and 2.88M disks. As you can see, a 1.44M disk is really 1,440K, and not actually 1.44 megabytes. Because a megabyte is 1,024K, what we call a 1.44M disk is actually 1.406 megabytes in capacity. Another way of expressing disk capacity is in millions of bytes. In that case, the 1.44M disk has 1.475 million bytes of capacity. To add to the confusion over capacity expression, both megabyte and millions of bytes are abbreviated as MB or M. No universally accepted standard for the definition of M or MB exists, so throughout this book I will try to be explicit.

Like blank sheets of paper, new disks contain no information. Formatting a disk is similar to adding lines to the paper so that you can write straight across. Formatting places on the disk the information DOS needs to maintain a directory and file table of contents. Using the /S (system) option in the FORMAT command resembles making the paper a title page. FORMAT places on the disk the portions of DOS required to boot the system.

DOS reserves the track nearest to the outside edge of a disk (track 0) almost entirely for its purposes. Track 0, Sector 1 contains the DOS Boot Record (DBR) or Boot Sector, the system needs to begin operation. The next few sectors contain the File Allocation Tables (FATs), which act as the disk “room reservation clerk” that keeps records of which clusters or allocation units (rooms) on the disk have file information and which are empty. Finally, the next few sectors contain the root directory, in which DOS stores information about the names and starting locations of the files on the disk; you see most of this information when you use the DIR command.

In computer-industry jargon, this process is “transparent to the user,” which means that you don’t have to (and generally cannot) decide where information is stored on disks. That this process is “transparent,” however, doesn’t necessarily mean that you shouldn’t be aware of the decisions DOS makes for you.

When DOS writes data, it always begins by attempting to use the earliest available data sectors on the disk. Because the file might be larger than the particular block of available sectors selected, DOS then writes the remainder of the file in the next available block of free sectors. In this manner, files can become fragmented as they are written to fill a hole on the disk created by the deletion of some smaller file. The larger file completely fills the hole; then DOS continues to look for more free space across the disk, from the outermost tracks to the innermost tracks. The rest of the file is deposited in the next available free space.

This procedure continues until eventually all the files on your disk are intertwined. This situation is not really a problem for DOS because it was designed to manage files in this way. The problem is a physical one: Retrieving a fragmented file that occupies 50 or 100 separate places across the disk takes much longer than if the file were in one piece. Also, if the files were in one piece, recovering data in the case of a disaster would be much easier. Consider unfragmenting a disk periodically simply because it can make recovery from a disk disaster much easier; many people, however, unfragment disks for the performance benefit in loading and saving files that are in one piece.

How do you unfragment a disk? DOS 6.0 and higher versions include a command called DEFRAG. This utility is actually a limited version of the Norton Utilities Speedisk program. It does not have some of the options of the more powerful Norton version and is not as fast, but it does work well in most cases. Earlier versions of DOS do not provide any easy method for unfragmenting a disk, although by backing up and restoring files, you can accomplish the goal. To unfragment a floppy disk for example, you can copy all the files one by one to an empty disk, delete the original files from the first disk, and then recopy the files. With a hard disk, you can back up all the files, reformat the disk, and restore the files. This procedure is time consuming, to say the least.

Because DOS versions earlier than 6.0 did not provide a good way to unfragment a disk, many software companies have produced utility programs that can easily unfragment disks in a clean and efficient manner. These programs can restore file contiguity without reformat and restore operations. My favorite for an extremely safe, easy, and *fast* unfragmenting program is the Vopt utility, by Golden Bow. In my opinion, no other unfragmenting utility even comes close to this amazing $50 package. Golden Bow's address and phone number are in Appendix B.

Caution

These unfragmenting programs, inherently dangerous by nature, do not eliminate the need for a good backup program. Before using an unfragmenting program, make sure that you have a good backup. What shape do you think your disk would be in if the power failed during an unfragmenting session? Also, some programs have bugs or are incompatible with new releases of DOS.

Cylinders. The term *cylinder* usually is used in place of *track*. A cylinder is all the tracks under read/write heads on a drive at one time. For floppy drives, because a disk cannot have more than two sides and the drive has two heads, normally there are two tracks per cylinder. Hard disks can have many disk platters, each with two (or more) heads, for many tracks per single cylinder.

Clusters or Allocation Units. A cluster also is called an *allocation unit* in DOS version 4.0 and higher. The term is appropriate because a single cluster is the smallest unit of the disk that DOS can allocate when it writes a file. A cluster or allocation unit consists of one or more sectors—usually two or more. Having more than one sector per cluster reduces the file-allocation table size and enables DOS to run faster because it has fewer individual allocation units of the disk with which to work. The tradeoff is in some wasted disk space. Because DOS can manage space only in the cluster size unit, every file consumes space on the disk in increments of one cluster. Table 13.4 lists the default cluster sizes used by DOS for different floppy disk formats. Chapter 14, "Hard Disk Drives and Controllers," discusses hard disk cluster or allocation unit sizes.

Table 13.4 DOS Default Cluster and Allocation Unit Sizes

Floppy Disk Capacity	Cluster/Allocation	Unit Size	FAT Type
5 1/4-inch, 360K	2 sectors	1,024 bytes	12-bit
5 1/4-inch, 1.2M	1 sector	512 bytes	12-bit
3 1/2-inch, 720K	2 sectors	1,024 bytes	12-bit
3 1/2-inch, 1.44M	1 sector	512 bytes	12-bit
3 1/2-inch, 2.88M	2 sectors	1,024 bytes	12-bit

K = 1,024 bytes
M = 1,048,576 bytes

The high-density disks normally have smaller cluster sizes, which seems strange because these disks have many more individual sectors than do double-density disks. The probable reason is that because these high-density disks are faster than their double-density counterparts, IBM and Microsoft thought that the decrease in wasted disk space cluster size and speed would be welcome. You learn later that the cluster size on hard disks can vary much more between different versions of DOS and different disk sizes. Table 13.5 shows the floppy disk logical parameters.

Types of Floppy Drives

Five types of standard floppy drives are available for an IBM-compatible system. The drives can be summarized most easily by their formatting specifications (refer to tables 13.2 and 13.3).

Most drive types can format multiple types of disks. For example, the 3 1/2-inch ED drive can format and write on any 3 1/2-inch disk. The 5 1/4-inch HD drive also can format and write on any 5 1/4-inch disk (although, as mentioned, sometimes track-width problems occur). This drive can even create some older obsolete formats, including single-sided disks and disks with eight sectors per track.

As you can see from table 13.5, the different disk capacities are determined by several parameters, some of which seem to remain constant on all drives, whereas others change from drive to drive. For example, all drives use 512-byte physical sectors, which remains true for hard disks as well. Note, however, that DOS treats the sector size as though it could be a changeable parameter, although the BIOS does not.

Note also that now all standard floppy drives are double-sided. IBM has not shipped PC systems with single-sided drives since 1982; these drives are definitely considered obsolete. Also, IBM never has utilized any form of single-sided 3 1/2-inch drives, although that type of drive appeared in the first Apple Macintosh systems in 1984. IBM officially began selling and supporting 3 1/2-inch drives in 1986 and has used only double-sided versions of these drives.

Table 13.5 Floppy Disk Logical DOS-Format Parameters

	Current Formats					**Obsolete Formats**		
Disk Size (in.)	**3 1/2"**	**3 1/2"**	**3 1/2"**	**5 1/4"**	**5 1/4"**	**5 1/4"**	**5 1/4"**	**5 1/4"**
Disk Capacity (K)	**2,880**	**1,440**	**720**	**1,200**	**360**	**320**	**180**	**160**
Media Descriptor Byte	F0h	F0h	F9h	F9h	FDh	FFh	FCh	FEh
Sides (Heads)	2	2	2	2	2	2	1	1
Tracks per Side	80	80	80	80	40	40	40	40

(continues)

Table 13.5 Continued

	Current Formats					Obsolete Formats		
Disk Size (in.)	**3 1/2"**	**3 1/2"**	**3 1/2"**	**5 1/4"**	**5 1/4"**	**5 1/4"**	**5 1/4"**	**5 1/4"**
Disk Capacity (K)	**2,880**	**1,440**	**720**	**1,200**	**360**	**320**	**180**	**160**
Sectors per Track	36	18	9	15	9	8	9	8
Bytes per Sector	512	512	512	512	512	512	512	512
Sectors per Cluster	2	1	2	1	2	2	1	1
FAT Length (Sectors)	9	9	3	7	2	1	2	1
Number of FATs	2	2	2	2	2	2	2	2
Root Dir. Length (Sectors)	15	14	7	14	7	7	4	4
Maximum Root Entries	240	224	112	224	112	112	64	64
Total Sectors per Disk	5,760	2,880	1,440	2,400	720	640	360	320
Total Available Sectors	5,726	2,847	1,426	2,371	708	630	351	313
Total Available Clusters	2,863	2,847	713	2,371	354	315	351	313

The 360K 5 1/4-Inch Drive

The 5 1/4-inch low-density drive is designed to create a standard-format disk with 360K capacity. Although I persistently call these low-density drives, the industry term is “double-density.” I use “low-density” because I find the term “double-density” to be somewhat misleading, especially when I am trying to define these drives in juxtaposition to the high-density drives.

The term *double-density* arose from the use of the term *single-density* to indicate a type of drive that used frequency modulation (FM) encoding to store approximately 90 kilobytes on a disk. This type of obsolete drive never was used in any IBM-compatible systems, but was used in some older systems such as the original Osborne-1 portable computer. When drive manufacturers changed the drives to use Modified Frequency Modulation (MFM) encoding, they began using the term “double-density” to indicate it, as well as the (approximately doubled) increase in recording capacity realized from this encoding method. All modern floppy disk drives use MFM encoding, including all types listed in this section. Encoding methods such as FM, MFM, and RLL variants are discussed in Chapter 14, “Hard Disk Drives and Controllers.”

The 360K 5 1/4-inch drive normally records 40 cylinders of two tracks each, with each cylinder numbered starting with 0 closest to the outside diameter of the floppy disk. Head position (or side) 0 is recorded on the underside of the floppy disk, and Head 1 records on the top of the disk surface. This drive normally divides each track into nine sectors, but it can optionally format only eight sectors per track to create a floppy disk compatible with DOS versions 1.1 or earlier. This type of format rarely (if ever) is used today.

The 360K 5 1/4-inch drives as supplied in the first IBM systems all were full-height units, which means that they were 3.25 inches tall. Full-height drives are obsolete now and have not been manufactured since 1986. Later units used by IBM and most compatible vendors have been the half-height units, which are only 1.6 inches tall. You can install two half-height drives in place of a single full-height unit. These drives, made by different manufacturers, are similar except for some cosmetic differences.

The 360K 5 1/4-inch drives spin at 300 RPM, which equals exactly five revolutions per second, or 200 milliseconds per revolution. All standard floppy controllers support a 1:1 interleave, in which each sector on a specific track is numbered (and read) consecutively. To read and write to a disk at full speed, a controller sends data at a rate of 250,000 bits per second. Because all low-density controllers can support this data rate, virtually any controller supports this type of drive, depending on ROM BIOS code that supports these drives.

All standard IBM-compatible systems include ROM BIOS support for these drives; therefore, you usually do not need special software or driver programs to use them. This statement might exclude some aftermarket (non-IBM) 360K drives for PS/2 systems that might require some type of driver in order to work. The IBM-offered units use the built-in ROM support to enable these drives to work. The only requirement usually is to run the Setup program for the machine to enable it to properly recognize these drives.

The 1.2M 5 1/4-Inch Drive

The 1.2M high-density floppy drive first appeared in the IBM AT system introduced in August 1984. The drive required the use of a new type of disk to achieve the 1.2M format capacity, but it still could read and write (although not always reliably) the lower-density 360K disks.

The 1.2M 5 1/4-inch drive normally recorded 80 cylinders of two tracks each, starting with cylinder 0, at the outside of the disk. This situation differs from the low-density 5 1/4-inch drive in its capability to record twice as many cylinders in approximately the same space on the disk. This capability alone suggests that the recording capacity for a disk would double, but that is not all. Each track normally is recorded with 15 sectors of 512 bytes each, increasing the storage capacity even more. In fact, these drives store nearly four times the data of the 360K disks. The density increase for each track required the use of special disks with a modified media designed to handle this type of recording. Because these disks initially were expensive and difficult to obtain, many users attempted incorrectly to use the low-density disks in the 1.2M 5 1/4-inch drives and format them to the higher 1.2M-density format, which results in data loss and unnecessary data-recovery operations.

A compatibility problem with the 360K drives stems from the 1.2M drive's capability to write twice as many cylinders in the same space as the 360K drives. The 1.2M drives position their heads over the same 40 cylinder positions used by the 360K drives through *double stepping*, a procedure in which the heads are moved every two cylinders to arrive at the correct positions for reading and writing the 40 cylinders on the 360K disks. The problem is that because the 1.2M drive normally has to write 80 cylinders in the same space in which the 360K drive writes 40, the heads of the 1.2M units had to be made dimensionally smaller. These narrow heads can have problems overwriting tracks produced by a 360K drive that has a wider head because the narrower heads on the 1.2M drive cannot "cover" the entire track area written by the 360K drive. This problem and possible solutions to it are discussed later in this chapter.

The 1.2M 5 1/4-inch drives spin at 360 RPM, or six revolutions per second, or 166.67 milliseconds per revolution. The drives spin at this rate no matter what type of disk is inserted—either low- or high-density. To send or receive 15 sectors (plus required overhead) six times per second, a controller must use a data-transmission rate of 500,000 bits per second (500 kilohertz, or kHz). All standard high- and low-density controllers support this data rate and, therefore, these drives. This support of course depends also on proper ROM BIOS support of the controller in this mode of operation. When a standard 360K disk is running in a high-density drive, it also is spinning at 360 RPM; a data rate of 300,000 bits per second (300 kHz) therefore is required in order to work properly. All standard AT-style low- and high-density controllers support the 250 kHz, 300 kHz, and 500 kHz data rates. The 300 kHz rate is used only for high-density 5 1/4-inch drives reading or writing to low-density 5 1/4-inch disks.

Virtually all standard AT-style systems have a ROM BIOS that supports the controller's operation of the 1.2M drive, including the 300 kHz data rate.

The 720K 3 1/2-Inch Drive

The 720K, 3 1/2-inch, double-density drives first appeared in an IBM system with the IBM Convertible laptop system introduced in 1986. In fact, all IBM systems introduced since that time have 3 1/2-inch drives as the standard supplied drives. This type of drive also is offered by IBM as an internal or external drive for the AT or XT systems. Note that outside the IBM-compatible world, other computer-system vendors (Apple, Hewlett Packard, and so on) offered 3 1/2-inch drives for their systems well before the IBM-compatible world "caught on."

The 720K, 3 1/2-inch, double-density drive normally records 80 cylinders of two tracks each, with nine sectors per track, resulting in the formatted capacity of 720 kilobytes. It is interesting to note that many disk manufacturers label these disks as 1-megabyte disks, which is true. The difference between the actual 1 megabyte of capacity and the usable 720K after formatting is that some space on each track is occupied by the header and trailer of each sector, the inter-sector gaps, and the index gap at the start of each track before the first sector. These spaces are not usable for data storage and account for the differences between the unformatted and formatted capacities. Most manufacturers report the unformatted capacities because they do not know on which type of system you will format the disk. Apple Macintosh systems, for example, can store 800K of data on the same disk because of a different formatting technique. Note also that the 720K of usable space does not account for the disk areas DOS reserves for managing the disk (boot sectors, FATs, directories, and so on) and that because of these areas, only 713K remains for file data storage.

IBM-compatible systems have used 720K, 3 1/2-inch, double-density drives primarily in XT-class systems because the drives operate from any low-density controller. The drives spin at 300 RPM, and therefore require only a 250 kHz data rate from the controller to operate properly. This data rate is the same as for the 360K disk drives, which means that any controller that supports a 360K drive also supports the 720K drives.

The only issue to consider in installing a 720K, 3 1/2-inch drive is whether the ROM BIOS offers the necessary support. An IBM system with a ROM BIOS date of 06/10/85 or later has built-in support for 720K drives and requires no driver in order to use them. If your system has an earlier ROM BIOS date, the DRIVER.SYS program from DOS 3.2 or higher—as well as the DRIVPARM config.sys command in some OEM DOS versions—is all you need to provide the necessary software support to operate these drives. Of course, a ROM BIOS upgrade to a later version negates the need for “funny” driver software and is usually the preferred option when you add one of these drives to an older system.

The 1.44M 3 1/2-Inch Drive

The 3 1/2-inch, 1.44M, high-density drives first appeared from IBM in the PS/2 product line introduced in 1987. Although IBM has not officially offered this type of drive for any of its older systems, most compatible vendors started offering the drives as options in systems immediately after IBM introduced the PS/2 system.

The drives record 80 cylinders consisting of two tracks each with 18 sectors per track, resulting in the formatted capacity of 1.44 megabytes. Most disk manufacturers label these disks as 2-megabyte disks, and the difference between this unformatted capacity and the formatted usable result is lost during the format. Note that the 1,440K of total formatted capacity does not account for the areas DOS reserves for file management, leaving only 1423.5K of actual file-storage area.

These drives spin at 300 RPM, and in fact must spin at that speed to operate properly with your existing high- and low-density controllers. To utilize the 500 kHz data rate, the maximum from most standard high- and low-density floppy controllers, these drives could spin at only 300 RPM. If the drives spun at the faster 360 RPM rate of the 5 1/4-inch drives, they would have to reduce the total number of sectors per track to 15 or else the controller could not keep up. In short, the 1.44M 3 1/2-inch drives store 1.2 times the data of the 5 1/4-inch 1.2M drives, and the 1.2M drives spin exactly 1.2 times faster than the 1.44M drives. The data rates used by both high-density drives are identical and compatible with the same controllers. In fact, because these 3 1/2-inch high-density drives can run at the 500 kHz data rate, a controller that can support a 1.2M 5 1/4-inch drive can support the 1.44M drives also. If you are using a low-density disk in the 3 1/2-inch high-density drive, the data rate is reduced to 250 kHz, and the disk capacity is 720K.

The primary issue in a particular system utilizing a 1.44M 3 1/2-inch drive is one of ROM BIOS support. An IBM system with a ROM BIOS date of 11/15/85 or later has built-in support for these drives and no external driver support program is needed. You might need a generic AT setup program because IBM’s setup program doesn’t offer the 1.44M drive as an option. Another problem relates to the controller and the way it signals the high-density drive to write to a low-density disk. The problem is discussed in detail in the following section.

The 2.88M 3 1/2-Inch Drive

The new 2.88M drive was developed by Toshiba Corporation in the 1980s, and officially announced in 1987. Toshiba began production manufacturing of the drives and disks in 1989, and then several vendors began selling the drives as upgrades for systems. IBM officially adopted these drives in the PS/2 systems in 1991, and virtually all PS/2s sold since then have these drives as standard equipment. Because a 2.88M drive can fully read and write 1.44M and 720K disks, the change was an easy one. DOS version 5.0 or higher is required to support the 2.88M drives.

To support the 2.88M drive, modifications to the disk controller circuitry were required because these drives spin at the same 300 RPM but have an astonishing 36 sectors per track. Because all floppy disks are formatted with consecutively numbered sectors (1:1 interleave), these 36 sectors have to be read and written in the same time it takes a 1.44M drive to read and write 18 sectors. This requires that the controller support a much higher data transmission rate of 1 MHz (1 million bits per second). Most of the older floppy controllers either found on an adapter card or built into the motherboard support only the maximum of 500 kHz data rate used by the 1.44M drives. To upgrade to 2.88M drives would require that the controller be changed to one that supports the higher 1 MHz data rate.

An additional support issue is the ROM BIOS. The BIOS must have support for the controller and the capability to specify and accept the 2.88M drive as a CMOS setting. Newer motherboard BIOS sets from companies like Phoenix, AMI, and Award have support for the new extra-high density controllers.

In addition to the newer IBM PS/2 systems, most newer IBM clone and compatible systems now have built-in floppy controllers and ROM BIOS software that fully supports the 2.88M drives. Adding or upgrading to a 2.88M drive in these systems is as easy as plugging in the drive and running the CMOS Setup program. For those systems who do not have this support built-in, this type of upgrade is much more difficult. Several companies offer new controllers and BIOS upgrades as well as the 2.88M drives specifically for upgrading older systems.

Although the 2.88M drives themselves are not much more expensive than the 1.44M drives they replace, the disk media is currently still very expensive. Although you can purchase 1.44M disks for around (or under) one dollar each, the 2.88M disks can cost more than $5 per disk! As the drives become more generally available, the disk media prices should fall. The 1.44M and even 1.2M disk media also was very expensive when first introduced.

Handling Recording Problems with 1.44M 3 1/2-Inch Drives

A serious problem awaits many users who use the 1.44M 3 1/2-inch drives: If the drive is installed improperly, any write or format operations performed incorrectly on 720K disks can end up trashing data on low-density disks. The problem is caused by the controller's incapability to signal the high-density drive that a low-density recording will take place.

High-density disks require a higher write-current or signal strength when they record than do the low-density disks. A low-density drive can record at only the lower write-current, which is correct for the low-density disks; the high-density drive, however, needs to record at both high and low write-currents depending on which type of disk is inserted in the drive. If a signal is not sent to the high-density drive telling it to lower or reduce the write-current level, the drive stays in its normal high write-current default mode, even when it records on a low-density disk. The signal normally should be sent to the drive by the controller, but many controllers do not provide this signal properly for the 1.44M drives.

The Western Digital controller used by IBM enables the reduced write-current (RWC) signal only if the controller also is sending data at the 300 kHz data rate, indicating the special case of a low-density disk in a high-density drive. The RWC signal is required to tell the high-density drive to lower the head-writing signal strength to be proper for the low-density disks. If the signal is not sent, the drive defaults to the higher write-current, which should be used for only high-density disks. If the controller is transmitting the 250 kHz data rate, the controller knows that the drive must be a low-density drive and therefore no RWC signal is necessary because the low-density drives can write only with reduced current.

This situation presented a serious problem for owners of 1.44M drives using 720K disks because the drives spin the disks at 300 RPM, and in writing to a low-density disk, use the 250 kHz data rate—not the 300 kHz rate. This setup “fools” the controller into “thinking” that it is sending data to a low-density drive, which causes the controller to fail to send the required RWC signal. Without the RWC signal, the drive then records improperly on the disk, possibly trashing any data being written or any data already present. Because virtually all compatibles use controllers based on the design of the IBM AT floppy disk controller, most share the same problem as the IBM AT.

Drive and disk manufacturers devised the perfect solution for this problem, short of using a redesigned controller. They built into the drives a *media sensor*, which, when it is enabled, can override the controller’s RWC signal (or lack of it) and properly change the head-current levels within the drive. Essentially, the drive chooses the write-current level independently from the controller when the media sensor is operational.

The sensor is a small, physical or optical sensor designed to feel, or “see,” the small hole on the high-density 3 1/2-inch disks located opposite the write-enable hole. The extra hole on these high-density or extra-high density disks is the media sensor’s cue that the full write-current should be used in recording. If an ED disk is detected, the ED drive enables the vertical recording heads. Low-density disks do not have these extra holes; therefore, when the sensor cannot see a media-sensor hole, it causes the drive to record in the proper reduced write-current mode for a double-density disk.

Some people, of course, foolishly attempt to override the function of these sensors by needlessly punching an extra hole in a low-density disk, to fool the drive’s sensor into acting as though an actual high-density disk has been inserted. Several “con artist” companies have made a fast buck by selling media sensor hole-punchers to unwary or misinformed people. These “shyster” disk-punch vendors try to mislead you into believing that no difference exists between the low- and high-density disks except for the hole, and that punching the extra hole makes the low-density disk a legitimate high-density disk. This, of course, is absolutely untrue: The high-density disks are very different from low-density disks. The differences between the disks are explained in more detail later in this chapter.

Another reason that this hole-punching is needless is that if you want to record a high-density format on a low-density disk, you only have to remove the jumper from the drive that enables the media sensor. Removing the media sensor jumper still allows the drive to work properly for high-density disks writing at the full write-current level, but unfortunately also allows the higher write-current to be used on low-density disks as well because then the drive has no way of knowing the difference. If you really want to risk your data to low-density disks formatted as high-density disks, you can save yourself the cost of the $40 hole-punchers. Note that even if you attempt to format or record properly on a 720K disk, you still will be working at the higher write-current and risk trashing the disk.

Many systems, including the IBM PS/2 series, Compaq, Toshiba laptops, and many others with floppy controllers built into the motherboard, do not need 1.44M drives with media sensors. Their controllers have been fixed to allow the RWC signal to be sent to the drive even when the controller is sending the 250 kHz data rate. This setup allows for proper operation no matter what type of disk or drive is used, as long as the user formats properly. Because these systems do not have a media sensor policing users, they easily can format low-density disks as high-density disks regardless of what holes are on the disk. This has caused problems for users of the older PS/2 systems who have accidentally formatted 720K disks as 1.44M disks. When passed to a system that has an enabled media sensor, the system refuses to read the disks at all because it is not correctly formatted. If you are having disk interchange problems, make sure that you are formatting your disks correctly.

The newer PS/2 and other high-end systems from other manufacturers (Hewlett Packard, for example) use an active media sensor setup in which the user no longer has to enter the correct FORMAT command parameters to format the disk. In these systems the media sensor information is passed through the controller to the BIOS, which properly informs the FORMAT command about which disk is in the drive. With these systems, it is impossible for a user to accidentally format a disk incorrectly, and it eliminates the user from having to know anything about the different disk media types.

Handling Recording Problems with 1.2M and 360K Drives

The 5 1/4-inch drives have their own special problems. One major problem resulting in needless data destruction is that the tracks sometimes are recorded at different widths for different drives. These differences in recorded track width can result in problems with data exchange between different 5 1/4-inch drives.

As shown in table 13.1, the recorded track-width difference affects only the 5 1/4-inch drives because the 5 1/4-inch low-density drives record a track width more than twice that of the 5 1/4-inch high-density drives. This difference presents a problem if a high-density drive is used to update a low-density disk with previously recorded data on it. The high-density drive, even in 360K mode, cannot completely overwrite the track left by the 40-track drive. The problem occurs when the disk is returned to the person with the 360K drive because that drive sees the new data as "embedded" within the remains of the previously written track. The 360K drive cannot distinguish either signal, and an `Abort, Retry, Ignore` error message results.

To avoid this problem, start with a brand-new disk that has never been formatted and format it in the 1.2M drive with the /4 (or equivalent) option. This procedure causes the 1.2M drive to place the proper 360K format on the disk. The 1.2M drive then can be used to fill the disk to its 360K capacity, and every file will be readable on the 40-track 360K drive because no previous wider data tracks exist to confuse the 360K drive. I use this trick all the time to exchange data disks between AT systems that have only a 1.2M drive and XT or PC systems that have only a 360K drive. The key is to start with a brand-new disk or a disk wiped clean magnetically by a bulk eraser. Simply reformatting the disk does not work because formatting actually writes data to the disk.

Note that because all the 3 1/2-inch drives write tracks of the same width, these drives have no disk-interchange problems related to track width.

Analyzing Floppy Disk Construction

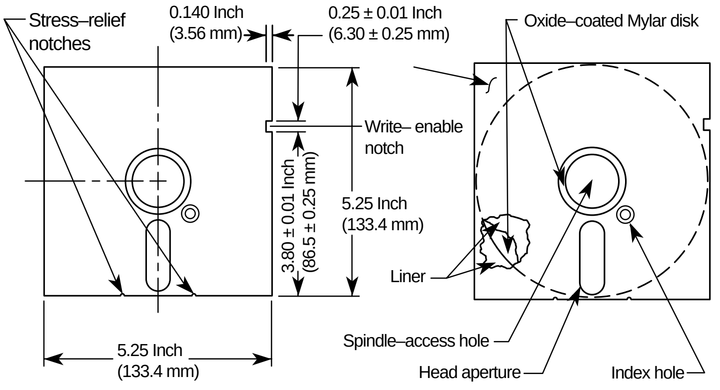
The 5 1/4-inch and 3 1/2-inch disks each have unique construction and physical properties.

The flexible (or floppy) disk is contained within a plastic jacket. The 3 1/2-inch disks are covered by a more rigid jacket than are the 5 1/4-inch disks; the disks within the jackets, however, are virtually identical except, of course, for the size.

Differences and similarities exist between these two different-size disks. This section looks at the physical properties and construction of each disk type.

When you look at a typical 5 1/4-inch floppy disk, you see several things (see fig. 13.9). Most prominent is the large, round hole in the center. When you close the disk drive's "door," a cone-shaped clamp grabs and centers the disk through the center hole. Many disks come with *hub-ring reinforcements*—thin, plastic rings like those used to reinforce three-ring notebook paper—intended to help the disk withstand the mechanical forces of the clamping mechanism. The high-density disks usually lack these reinforcements because the difficulty in accurately placing them on the disk means that they will cause alignment problems.

On the right side, just below the center of the hub hole, is a smaller, round hole called the *index hole*. If you carefully turn the disk within its protective jacket, you see a small hole in the disk. The drive uses the index hole as the starting point for all the sectors on



Courtesy of IBM Corporation

the disk—sort of the "prime meridian" for the disk sectors. A disk with a single index hole is a *soft-sectored* disk; the software (operating system) decides the actual number of sectors on the disk. Some older equipment, such as Wang word processors, use hard-sectored disks, which have an index hole to demarcate individual sectors. Do not use hard-sectored disks in a PC.

Fig. 13.9
Construction of a 5 1/4-inch floppy disk.

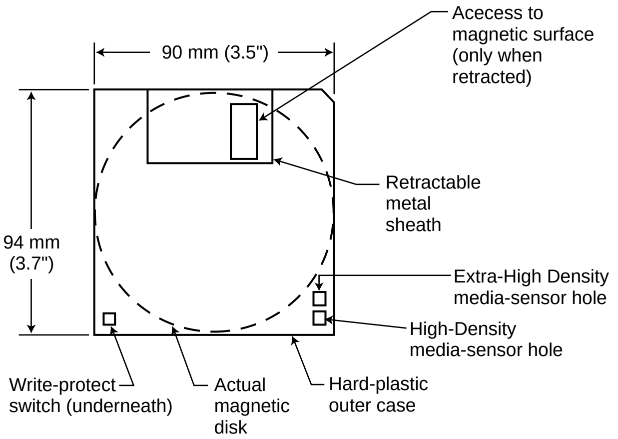
Below the hub hole is a slot shaped somewhat like a long racetrack through which you can see the disk surface. Through this *media-access hole*, the disk drive heads read and write information to the disk surface.

At the right side, about one inch from the top, is a rectangular punch from the side of the disk cover. If this *write-enable notch* is present, writing to the disk has been enabled. Disks without this notch (or with the notch taped over) are *write-protected* disks. The notch might not be on all disks, particularly those purchased with programs on them.

On the rear of the disk jacket, at the bottom, two very small, oval notches flank the head slot. The notches relieve stress on the disk and help prevent it from warping. The drive might use these notches also to assist in keeping the disk in the proper position in the drive.

Because the 3 1/2-inch disks use a much more rigid plastic case, which helps stabilize the disk, these disks can record at track and data densities greater than the 5 1/4-inch disks (see fig. 13.10). A metal shutter protects the media-access hole. The shutter is manipulated by the drive and remains closed whenever the disk is not in a drive. The media then is insulated from the environment and from your fingers. The shutter also obviates the need for a disk jacket.

Rather than an index hole in the disk, the 3 1/2-inch disks use a metal center hub with an alignment hole. The drive "grasps" the metal hub, and the hole in the hub enables the drive to position the disk properly.



On the lower-left part of the disk is a hole with a plastic slider—the write-protect/-enable hole (refer to fig. 13.9). When the slider is positioned so that the hole is visible, the disk is write-protected; the drive is prevented from recording on the disk. When the slider is positioned to cover the hole, writing is enabled, and you can record on the disk. For more permanent write-protection, some commercial software programs are supplied on disks with the slider removed so that you cannot easily enable recording on the disk.

Fig. 13.10
Construction of a 3 1/2-inch floppy disk.

On the other (right) side of the disk from the write-protect hole, there might be in the disk jacket another hole called the *media-density-selector hole*. If this hole is present, the disk is constructed of a special media and is therefore a high-density or extra-high-density disk. If the media-sensor hole is exactly opposite the write-protect hole, it indicates a 1.44M HD disk. If the media-sensor hole is located more toward the top of the disk (the metal shutter is at the top of the disk), it indicates an ED disk. No hole on the right side means that the disk is a low-density disk. Most 3 1/2-inch drives have a media sensor that controls recording capability based on the existence or absence of these holes.

Both the 3 1/2-inch and 5 1/4-inch disks are constructed of the same basic materials. They use a plastic base (usually Mylar) coated with a magnetic compound. The compound is usually a ferric- (iron-) oxide-based compound for the standard density versions; a cobalt-ferric compound usually is used in the higher coercivity (higher density) disks. A newly announced disk type, called *extended density*, uses a barium-ferric compound. The rigid jacket material on the 3 1/2-inch disks often causes people to believe incorrectly that these disks are some sort of “hard disk” and not really a floppy disk. The disk “cookie” inside the 3 1/2-inch case is just as floppy as the 5 1/4-inch variety.

Floppy Disk Types and Specifications

This section examines all the types of disks you can purchase for your system. Especially interesting are the technical specifications that can separate one type of disk from another. This section defines all the specifications used to describe a typical disk.

Single- and Double-Sided Disks. Whether a disk is single- or double-sided is really an issue only for the lower-density disks. Because no single-sided high-density drives are manufactured, no need exists for disks to match the drives. The original IBM PC had single-sided drives, but they were discontinued in 1982.

A single-sided disk is constructed of the same material as a double-sided disk. The only difference seems to be that only the single-sided disks are "certified" (whatever that means) on only one side, and the double-sided disks are certified on both sides. Because the single-sided disks are cheaper than the double-sided versions, many PC users quickly determined that they could save some money if they used the single-sided disks even in double-sided drives.

This reasoning works because it is economically impractical for disk manufacturers to make some disks with recording surfaces on one side and other disks with recording surfaces on both sides. Today's single-sided disks look, and usually behave, exactly the same as double-sided disks. The result of this—depending on the brand of disks you buy—is that you can generally format and use "single-sided" disks successfully in double-sided drives, at a savings in disk costs. Unfortunately, the savings now are so small that this practice is obsolete, if not risky.

The danger in this practice is that some manufacturers do not burnish, or polish, the unused (top) side to the same level of smoothness as the used (bottom) side. This practice can cause accelerated wear on the top head. In single-sided drives, because the top head was replaced by a soft, felt pad, the rougher top side caused no problems. For the cost, it is not worth using with the wrong disks. I recommend the conservative route: Spend the small amount of extra money for double-sided disks of the correct density, and you will rarely have to recover damaged data.

Density. *Density*, in simplest terms, is a measure of the amount of information that can be packed reliably into a specific area of a recording surface. The keyword here is *reliably*.

Disks have two types of densities: longitudinal density and linear density. *Longitudinal density* is indicated by how many tracks can be recorded on the disk, often expressed as a number of tracks per inch (TPI). *Linear density* is the capability of an individual track to store data, often indicated as a number of bits per inch (BPI). Unfortunately, both types of densities often are interchanged incorrectly in discussing different disks and drives. Table 13.6 provides a rundown of each available type of disk.

Table 13.6 Floppy Disk Media Specifications

	5 1/4-Inch			3 1/2-Inch		
Media Parameters	**Double-Density (DD)**	**Quad-Density (QD)**	**High-Density (HD)**	**Double-Density (DD)**	**High-Density (HD)**	**Extra-High Density (ED)**
Tracks Per Inch (TPI)	48	96	96	135	135	135

Bits Per Inch (BPI)	5,876	5,876	9,646	8,717	17,434	34,868
Media Formulation	Ferrite	Ferrite	Cobalt	Cobalt	Cobalt	Barium
Coercivity (Oersteds)	300	300	600	600	720	750
Thickness (Micro-In.)	100	100	50	70	40	100
Recording Polarity	Horiz.	Horiz.	Horiz.	Horiz.	Horiz.	Vert.

It is notable that IBM skipped the quad-density disk type—that is, no IBM system uses a quad-density drive or requires quad-density disks. Don't purchase a quad-density disk unless you just want a better-quality double-density disk.

Both the quad- and double-density disks store the same linear data on each track. They use the same formula for the magnetic coating on the disk, but the quad-density versions represent a more rigorously tested, higher-quality disk. The high-density disks are entirely different, however. To store the increased linear density, an entirely different magnetic coating was required. In both the 5 1/4-inch and 3 1/2-inch high-density disks, a high-coercivity coating is used to allow the tremendous bit density for each track. A high-density disk never can be substituted for a double- or quad-density disk because the write-current must be different for these very different media formulations and thicknesses.

The extra-high density 3 1/2-inch disk in the chart is newly available in some systems. This type of disk, invented by Toshiba, is available from several other vendors as well. The extra-high density disks use a barium-ferric compound to cover the disk with a thicker coating, which enables a vertical recording technique to be used. In vertical recording, the magnetic domains are recorded vertically rather than flat. The higher density results from their capability to be stacked much more closely together. These types of drives can read and write the other 3 1/2-inch disks because of their similar track dimensions on all formats.

Media Coercivity and Thickness. The *coercivity specification* of a disk refers to the magnetic-field strength required to make a proper recording on a disk. Coercivity, measured in oersteds, is a value indicating magnetic strength. A disk with a higher coercivity rating requires a stronger magnetic field to make a recording on that disk. With lower ratings, the disk can be recorded with a weaker magnetic field. In other words, *the lower the coercivity rating, the more sensitive the disk*.

Another factor is the thickness of the disk. The thinner the disk, the less influence a region of the disk has on another adjacent region. The thinner disks therefore can accept many more bits per inch without eventually degrading the recording.

When I ask someone whether the high-density disks are more sensitive or less sensitive than the double-density disks, the answer is almost always "more sensitive." But you can see that this is not true. The high-density disks are in fact as much as half as sensitive as the double-density disks. A high-density drive can record with a much higher volume level at the heads than can the standard double-density drive. For these high-density drives to record properly on a double-density disk, the drive must be capable of using a reduced write-current mode and enable it whenever the lower-density disks are installed. A big problem with users and floppy disks then can occur.

Most users do not like paying more for the high-density disks their drives can use. These users, in an attempt to save money, are tempted to use the lower-density disks as a substitute. Some users attempt to format the "regular" disk at the high-density capacity. This formatting is facilitated by DOS, which always attempts to format a disk to the maximum capacity of the drive's capabilities, unless specifically ordered otherwise through the use of proper parameters in the FORMAT command, or unless your system has an Active Media Sensor. If you use no parameters and simply enter FORMAT A:, however, the disk is formatted as though it were a high-density disk. Many users think that this procedure somehow is equivalent to using the single-sided disks in place of double-sided ones. I can assure you that this is not true—it is much worse. *Do not use double-density disks in place of high-density disks*, or you will experience severe problems and data loss from improper coercivity, media thickness, and write-current specifications.

The reasons for using the high-coercivity thin disks are simple. In designing the high-density drives, engineers found that the density of magnetic flux reversals caused adjacent flux reversals to begin to affect each other. The effect was that they started to cancel each other out, or cause shifts in the polarity of the domain. Data written at the high densities eventually began to erase itself. As an analogy, imagine a wooden track on which you place magnetic marbles, evenly spaced four inches apart in a specific pattern of magnetic polarity. At this distance, the magnetic forces from each marble are too weak to affect the adjacent marbles. Now imagine that the marbles must be placed only two inches apart. The magnetic attraction and repulsion forces now might start to work on the adjacent marbles so that they begin to rotate on their axis and thus change the direction of polarity and the data they represent.

You could eliminate the interaction of the magnetic domains by either spacing them farther apart or making the domains "weaker," therefore reducing their sphere of influence. If the marbles were made half as strong magnetically as they were before, you could get them twice as close together without any interaction between them. This is the principle behind the high-coercivity, thin media disks. Because they are weaker magnetically, they need a higher recording strength to store an image properly.

Try a simple experiment to verify this principle. Attempt to format a high-density disk in a low-density format. DOS responds with a `Track 0 bad, Disk unusable` message. The disk did not accept a recording in low-density mode because the low-density recording is also low volume. The disk cannot make a recording on the disk, and therefore the `Track 0 bad` message is displayed.

It is unfortunate for users that the opposite attempt appears to work: You can format a standard double-density disk as though it were a high-density disk, and the FORMAT command or DOS does not seem to be affected. You might notice a large number of "bad sectors," but DOS allows the format to be completed anyway.

This situation is unfortunate for two reasons. First, you are recording on the (low-density) disk with a density that requires weak magnetic domains to eliminate interaction between the adjacent domains. A low-density disk unfortunately stores magnetic domains twice as strong as they should be, and eventually they interact. You will experience mysterious data losses on this disk over the next few days, weeks, or months.

Second, you have just placed a recording on this disk at twice the signal strength it should be. This "industrial strength" recording might not be removable by a normal disk drive, and the disk might be magnetically saturated. You might not ever be able to reformat it correctly as a double-density disk because a double-density reformat uses reduced write-current. The reduced write-current might not be capable of overwriting the high write-current signal that had been recorded incorrectly. The best way to remove this "burned in" recording then is to use a bulk eraser to renew the disk by removing all magnetic information. In most cases, however, a fresh format can overcome the magnetic image of the previous one, even if the write-current had been incorrect.

Do not use the wrong media for the format you are attempting to perform. You must use the correct type of disk. Note that the 3 1/2-inch disks have a perfect mechanism—the media sensor for deterring users. If you are sure that all your high-density 3 1/2-inch drives contain this sensor, set to function (by way of a jumper or switch), you are saved from your own ignorance. A drive with a media sensor sets the write-current and operating mode by the actual disk inserted. You are prevented from taking a 720K disk and attempting to cram 1.44 megabytes on it. Remember that IBM did not make using this sensor a requirement because it fixed the controller problem that necessitated it; therefore, incorrectly formatting a disk is easy on the PS/2 system. Also, if you want to format a low-density 3 1/2-inch disk incorrectly with a high-density format using an IBM PS/2 system, you don't need one of the "disk converters" or hole punchers.

Soft and Hard Sectors. Floppy disks are either *soft sectored* or *hard sectored.* A soft-sectored disk has only one index hole on the disk surface. Once every revolution, the hole is visible through the hole in the protective jacket, and the drive, controller, and DOS use the hole to establish the location and timing of the first sector on a track. Individual sectors are defined by the controller and the software that runs the controller, hence the term soft sectored. Hard-sectored disks have an index hole as well as a separate hole for each sector on the disk marking the beginning of that sector. Hard-sectored disks are not used in IBM compatible PCs, but were used in some dedicated word processing systems and other proprietary computer systems. If you try to use a hard-sectored disk in a PC, the machine will get confused. Sometimes hard-sectored disks are not labeled specifically as hard-sectored, but rather specify "10 sectors" or "16 sectors." Because hard-sectored disks are not used in IBM-compatible systems, I have not seen these disks for sale in quite some time.

Formatting and Using High- and Low-Density Disks

This section describes how the different density capabilities of the high- and low-density drives sometimes can cause problems in formatting disks. You must always make sure that a disk initially is formatted to the density in which it was supposed to be run. In some cases, you should have a high-density drive format a low-density disk. You can perform this formatting with the correct format commands. The following section describes how to use DOS correctly so that your disks are formatted properly.

Reading and Writing 360K Disks in 1.2M Drives. Having 1.2M drives *read* or *write* to a 360K disk is a simple task. Just place a previously formatted 360K disk in the drive and use it normally. In other words, pretend that the drive is a 360K drive. Nothing special must be done. You can either read or write on the disk with absolutely no problems…yet.

You will have a problem if you decide to return the disk to a 360K drive and attempt to read it. Remember that the recorded track width of the 1.2M drive is half the track width of the 360K drive; therefore, if any tracks have been *previously recorded by an actual 360K drive*, the tracks are twice as wide as the tracks recorded by the 1.2M drive. If you write to the disk with the 1.2M drive, you cannot overwrite the entire track width—only the center portion of it. When you return this disk to a 360K drive, the wider head system in the 360K drive then sees two signals on any overwritten tracks, and the new data is nestled within the image of the old data that could not be completely covered by the 1.2M drive. An immediate `Abort, Retry, Ignore` error from DOS usually is displayed for any updated portions of the disk.

To solve this problem easily, if you want to record data in an AT 1.2M drive and later read it properly in a 360K drive, make sure that you use *brand-new* disks for recording in the 1.2M drive. Because a new disk has no magnetic information on it, the smaller recorded track width can be written on the 1.2M drive and read properly in the 360K drive: The more narrow track is written in “clean space.” The 360K drive, therefore, no longer is confused by any “ghost images” of previously recorded wider tracks. Other than starting with a brand-new disk, your only other option is to use a disk erased by a bulk eraser. You cannot erase a disk by reformatting it if it has been in use. Formatting records actual data on the disk, and causes the track-width problem. The new or bulk-erased disk in fact must be formatted by the 1.2M drive for this procedure to work again. Remember the simple rule: Any track recorded by a 360K drive *cannot be overwritten* by a 1.2M drive, even in the 360K format.

How do you format a 360K disk in a 1.2M drive? If you just execute the FORMAT command without parameters, DOS attempts to format the disk to its maximum capacity. Because the 1.2M drive has no media-sensing capability, DOS assumes that the disk capability is equal to the maximum capability of the drive and attempts to create a 1.2M format on the disk. The write-current is increased also during a recording in this format, which is incompatible with the 360K media. To format the 360K disk correctly, therefore, look at the alternative command examples in table 13.7.

Table 13.7 Proper Formatting of 5 1/4-Inch 360K Disks in a 1.2M Drive

	DOS Version						
Command	**6.x**	**5.x**	**4.x**	**3.3**	**3.2**	**3.1**	**3.0**
FORMAT d: /4	Yes	Yes	Yes	Yes	Yes	Yes	Yes

FORMAT d: /N:9 /T:40	Yes	Yes	Yes	Yes	Yes	No	No
FORMAT d: /F:360	Yes	Yes	Yes	No	No	No	No

d: = The drive to format
N = Number of sectors per track
T = Tracks per side
F = Format capacity

Note that DOS versions prior to 3.0 do not support 1.2M drives. Each example command accomplishes the same function, which is to place on a 360K disk a 40-track, 9-sector format using reduced write-current mode.

Reading and Writing 720K Disks in 1.44M and 2.88M Drives. The 3 1/2-inch drives do not have the same problems as the 5 1/4-inch disks—that is, at least not with data interchange. Because both the high- and low-density drives write the same number of tracks and are the same width, one type of drive can be used to overwrite data written by another type of drive. Because of this capability, IBM does not need to offer a low-density version of the 3 1/2-inch drives for the PS/2 systems. These systems (except Models 25 and 30) include only the HD or ED drives, which are capable of imitating perfectly the 720K drives in the Model 25 and Model 30. The high-density drives can be trouble, however, in the hands of an inexperienced (or cheapskate) user. You *must* make sure that you use only the 1.44M high-density disks in the 1.44M format, and only the 720K disks in the 720K format. You will encounter serious problems if you stick a 720K disk in a drive in a PS/2 without a media sensor drive and enter the command FORMAT A:. If you decide to use the formatted disk anyway, massive data loss eventually will occur.

Problems with incorrect formatting could have been averted if IBM had universally used disk drives that included the media sensor. This special switch senses the unique hole found only on the right side of high-density disks. Drives that use this hole to control the status of reduced write-current never can format a disk incorrectly. The hardware saves you by causing the FORMAT command to end in failure with an appropriate error message if you attempt to format the disk incorrectly. All the newer PS/2 systems from IBM include drives with an active media sensor that allows disks to be formatted correctly with no parameters on the FORMAT command. The information FORMAT needs about what type of disk is in the drive is supplied by the BIOS directly on these systems; no user supplied parameters are necessary.

The 1.44M drives and 720K drives do not have all the same problems as the 5 1/4-inch drives, primarily because all the 3 1/2-inch drives have the same recorded track width. The 1.44M or 2.88M drive has no problem recording 720K disks. For these reasons (and more), I applaud the industry move to the 3 1/2-inch drives. The sooner we stop using 5 1/4-inch disk drives, the better.

The only other problem with formatting, other than incorrectly selecting a drive without a media sensor or failing to enable it during the drive-installation procedure, is naive users attempting to format disks at a capacity for which they were not designed. *You must have a 720K (double-density) disk to write a 720K format; a 1.44M (high-density) disk to* *write a 1.44M format; and a 2.88M (extra-high density) disk to write an extra-high density format.* No ifs, ands, or buts. This chapter has explained already that this requirement stems from differences in the coercivity of the media and the levels of recording current used in writing the disks.

When you enter a standard FORMAT command with no parameters, DOS normally attempts to format the disk to the drive's maximum capacity as indicated by the BIOS. If you insert a 720K disk in a 1.44M drive, therefore, and enter the FORMAT command with no parameters, DOS attempts to create a 1.44M format on the disk. If the drive has a passive media sensor, the FORMAT command aborts with an error message. The media sensor does not communicate to DOS the correct information to format the disk—it just *prevents* incorrect formatting. You still must know the correct commands. If the system supports active media sensing, then no FORMAT command parameters are necessary as the BIOS will supply the correct parameters based on the type of disk in the drive. Table 13.8 shows the correct FORMAT command and parameters to use in formatting a 720K disk in a 1.44M drive.

Table 13.8 Proper Formatting of 3 1/2-Inch 720K Disks in a 1.44M or 2.88M Drive

	DOS Version			
Command	**6.x**	**5.x**	**4.x**	**3.3**
FORMAT d: /N:9 /T:80	Yes	Yes	Yes	Yes
FORMAT d: /F:720	Yes	Yes	Yes	No

d: = The drive to format
N = Number of sectors per track
T = Tracks per side
F = Format capacity

Note that DOS versions prior to 3.3 do not support 1.44M drives, and versions prior to 5.0 do not support 2.88M drives.

Reading and Writing 1.44M Disks in 2.88M Drives. The 2.88M extra-high density (ED) drive used in some newer systems, such as virtually the entire PS/2 line, is a welcome addition to any system. This drive offers a capacity twice as great as the standard 1.44M HD drive, and also offers full backward compatibility with the 1.44M HD drive and the 720K DD drive.

The 2.88M ED drive uses a technique called *vertical recording* to achieve its great linear density of 36 sectors per track. This technique increases density by magnetizing the domains perpendicular to the recording surface. By essentially placing the magnetic domains on their ends and stacking them side by side, density increases enormously.

The technology for producing heads that can perform a vertical or perpendicular recording has been around awhile. It is not the heads or even the drives that represent the major breakthrough in technology; rather, it is the media that is special. Standard disks have magnetic particles shaped like tiny needles that lie on the surface of the disk. Orienting these acicular particles in a perpendicular manner to enable vertical recording is very difficult. The particles on a barium-ferrite floppy disk are shaped like tiny, flat, hexagonal platelets that easily can be arranged to have their axis of magnetization perpendicular to the plane of recording. Although barium ferrite has been used as a material in the construction of permanent magnets, no one has been able to reduce the grain size of the platelets enough for high-density recordings.

Toshiba has perfected a glass-crystallization process for manufacturing the ultra fine platelets used in coating the barium-ferrite disks. This technology, patented by Toshiba, is being licensed to a number of disk manufacturers, all of whom are producing barium-ferrite disks using Toshiba's process. Toshiba also made certain modifications to the design of standard disk drive heads to enable them to read and write the new barium-ferrite disks as well as standard cobalt or ferrite disks. This technology is being used not only in floppy drives but also is appearing in a variety of tape drive formats.

The disks are called 4M disks in reference to their unformatted capacity. Actual formatted capacity is 2,880K, or 2.88M. Because of space lost in the formatting process, as well as space occupied by the volume boot sector, file-allocation tables, and root directory, the total usable storage space is 2,863K.

A number of manufacturers are making these drives, including Toshiba, Mitsubishi, Sony, and Panasonic. During the next few years, they should become more popular in higher-end systems. Table 13.9 shows the correct FORMAT command and parameters to use in formatting a 1.44M disk in a 2.88M drive, especially if your system does not support the active media sensor.

Table 13.9 Proper Formatting of 3 1/2-Inch 1.44M Disks in a 2.88M Drive

	DOS Version	
Command	**6.x**	**5.x**
FORMAT d: /N:18 /T:80	Yes	Yes
FORMAT d: /F:1.44	Yes	Yes

d: = The drive to format
N = Number of sectors per track
T = Tracks per side
F = Format capacity

Note that DOS versions prior to 5.0 do not support the 2.88M drive. Also, most 2.88M installations have an active media sensor that automatically formats the disk correctly as determined by the media sensor. With active media sensing, the parameters indicating disk capacity are not necessary.

FORMAT Command Summary. This section is a short guide to the DOS FORMAT command. With newer DOS versions supporting more and different types of disk hardware, the once-simple FORMAT command has become more complex. Especially with the advent of DOS V5.0, the number of parameters and options available for the FORMAT command have increased dramatically. This section discusses the FORMAT command and these optional parameters. You see a simple guide to proper formatting of disks, as well as a thorough description of the FORMAT command parameters and options.

This chapter emphasized that a specific disk must always be formatted to its designated capacity. Formatting a disk to a capacity different from what it was designed for results only in an eventual loss of data from the disk. Because all the higher-density drives can format all the lower-density disks of the same form factor, knowing when a particular command option is required can be pretty complicated.

The basic rule is that a drive always formats in its native mode unless specifically instructed otherwise through the FORMAT command parameters. Therefore, if you insert a 1.44M HD disk in a 1.44M HD A drive, you then can format that disk by simply entering FORMAT A—no optional parameters are necessary in that case. If you insert any other type of disk (DD, for example), you absolutely *must* enter the appropriate parameters in the FORMAT command to change the format mode from the default 1.44M mode to the mode appropriate for the inserted disk. Even though the drive might have a media sensor that can detect which type of disk is inserted in the drive, in most cases the sensor does not communicate to the controller or DOS, which does not know which disk it is. An exception to this is the 2.88M drive installations that support active media sense. Most 2.88M drive installations support this advanced feature, which means that the media sensor will communicate the type of the inserted diskette to the controller and DOS. In this case no parameters are ever needed when formatting diskettes no matter what type is inserted. The FORMAT command will automatically default to the proper type as indicated by the active sensors on the 2.88M drive. I have even seen 1.44M drive installations with active media sensing (certain Hewlett Packard systems, for example), but this is rare.

In most cases of 1.44M drive installations, the media sensor in the drive is passive, and in effect all the sensor does is force the FORMAT command to fail if you do not enter the correct parameters for the inserted disk type.

Table 13.10 shows the proper format command for all possible variations in drive and disk types. It shows also which DOS versions support the various combinations of drives, disks, and FORMAT parameters.

To use this table, just look up the drive type and disk type you have. You then can see the proper FORMAT command parameters to use as well as the DOS versions that support the combination you want.

Table 13.10 Proper Disk Formatting

Drive Type	Disk Type	DOS Version	Proper FORMAT Command
5 1/4-inch 360K	DD 360K	DOS 2.0+	FORMAT d:

5 1/4-inch 1.2M	HD 1.2M	DOS 3.0+	FORMAT d:
5 1/4-inch 1.2M	DD 360K	DOS 3.0+	FORMAT d: /4
5 1/4-inch 1.2M	DD 360K	DOS 3.2+	FORMAT d: /N:9 /T:40
5 1/4-inch 1.2M	DD 360K	DOS 4.0+	FORMAT d: /F:360
3 1/2-inch 720K	DD 720K	DOS 3.2+	FORMAT d:
3 1/2-inch 1.44M	HD 1.44M	DOS 3.3+	FORMAT d:

Drive Type	**Disk Type**	**DOS Version**	**Proper FORMAT Command**
3 1/2-inch 1.44M	DD 720K	DOS 3.3+	FORMAT d: /N:9 /T:80
3 1/2-inch 1.44M	DD 720K	DOS 4.0+	FORMAT d: /F:720
3 1/2-inch 2.88M	ED 2.88M	DOS 5.0+	FORMAT d:
3 1/2-inch 2.88M	HD 1.44M	DOS 5.0+	FORMAT d: /F:1.44
3 1/2-inch 2.88M	DD 720K	DOS 5.0+	FORMAT d: /F:720

+ = Includes all higher versions
d: = Specifies drive to format
DD = double-density
HD = high-density
ED = extra-high density

Note

If the drive and installation you are using supports active (intelligent) media sense, no diskette type parameters are required. The drive will automatically communicate the type of the installed diskette to the FORMAT program. This is normal for most 2.88M drive installations.

With the advent of DOS 5.0, the FORMAT command has received a number of new functions and capabilities, all expressed through two new parameters: /Q (Quickformat) and /U (Unconditional). Precisely describing the effect of these parameters on the FORMAT command is difficult, especially considering that they have different effects on hard disks and floppy disks. Table 13.11 summarizes the functions of these new parameters and relates the new functions to the older versions of DOS.

Table 13.11 DOS FORMAT Command Internal Operations

Hard Disk Format Operations:	**DOS 2–4**	**DOS 5 and Higher**				
FORMAT Command Parameters	**Any**	**None**	**None**	**/Q**	**/U**	**/Q/U**
Disk Previously Formatted?	**–**	**Yes**	**No**	**Yes**	**Yes**	**Yes**
Check Volume Boot Sector	No	Yes	Yes	Yes	No	Yes
Save UNFORMAT Information	No	Yes	No	Yes	No	No

(continues)

Read Verify (Scan) Disk	Yes	Yes	Yes	No	Yes	No
Overwrite VBS, FATs & Root Dir.	Yes	Yes	Yes	Yes	Yes	Yes
Overwrite Data Area	No	No	No	No	No	No

Table 13.11 Continued

Floppy Disk Format Operations:		**DOS 2–4**	**DOS 5 and Higher**				
FORMAT Command Parameters		**Any**	**None**	**None**	**/Q**	**/U**	**/Q/U**
Disk Previously Formatted?		**–**	**Yes**	**No**	**Yes**	**Yes**	**Yes**
Check Volume Boot Sector	No	Yes	Yes	Yes	No	Yes	
Save UNFORMAT Information	No	Yes	No	Yes	No	No	
Read Verify (Scan) Disk	Yes	Yes	Yes	No	Yes	No	
Overwrite VBS, FATs & Root Dir.	Yes	Yes	Yes	Yes	Yes	Yes	
Overwrite Data Area	Yes	No	Yes	No	Yes	No	

/Q = Quick Format
/U = Unconditional Format
"–" = Does Not Matter
VBS = DOS Volume Boot Sector
FAT = File Allocation Table

From this table you should be able to discern the function of a specific FORMAT command relative to the use of the /Q and /U parameters. For example, suppose that you are using DOS V5.0 and you insert a brand-new 1.44M disk in a 1.44M drive A: on your system. If you enter the command FORMAT A: with no other parameters, what will happen? By looking at table 13.11, you can see that the default operation of the FORMAT command in this case would be as follows:

1. Check the DOS volume boot sector.
2. Perform a read verify (or scan) of the entire disk.
3. Overwrite the DVB, FATs, and the root directory.
4. Overwrite the entire data area of the disk.

These functions do not necessarily happen in this order; in fact, the last three items listed occur simultaneously as the format progresses. Now suppose that you write some files on this disk and reenter the same FORMAT A: command. As you can see from table 13.11, the functions of the FORMAT command are very different this time. The steps occur something like this:

1. Check the DOS volume boot sector.
2. Save UNFORMAT information.
3. Perform a read verify (or scan) of the entire disk.

4. Overwrite the DVB, FATs, and the root directory.

The default operation of FORMAT on a disk that is already formatted has changed dramatically with DOS V5.0. The biggest difference between this and older versions of DOS is that DOS V5.0 and higher versions (by default) save a backup copy of the disk's DOS volume boot sector, file-allocation tables, and root directory. This information, which is placed in a special format in sectors near the end of the disk, is designed to be utilized by the UNFORMAT command to restore these areas of the disk and therefore undo the work of the FORMAT command. In addition to saving this critical UNFORMAT information, the FORMAT command also defaults to *not* overwriting the data area of the disk; therefore, the UNFORMAT command can "restore" the disk data. The UNFORMAT does not actually restore the data—only the saved UNFORMAT information. The disk data is never lost. Older DOS versions do not check the disk to see whether it is formatted and always overwrite the entire floppy disk.

The /Q parameter stands for Quickformat. The basic function of /Q, to eliminate the (sometimes lengthy) read verify scan for disk defects that otherwise would occur, can be performed only on a disk that already is formatted. Any existing defect marks on the disk are preserved by using /Q. The net effect of /Q is to greatly speed up the formatting procedure for disks already formatted. It's a quick way to delete all the files from a disk quickly and efficiently.

The /U parameter stands for Unconditional. This parameter has two distinctly different effects depending on whether you are formatting a floppy disk or a hard disk. On a floppy disk, the /U parameter instructs the FORMAT command to overwrite the entire disk and skip saving UNFORMAT information because it would be useless anyway if the data were overwritten. On a hard disk, the purpose of /U is only to suppress saving UNFORMAT information. FORMAT on a hard disk *never* overwrites the data area of the disk, even with the /U parameter! If you have experience with the FORMAT command, you know that FORMAT never has overwritten data on a hard disk, no matter what version of IBM or MS-DOS you are using. (On some older OEM versions, such as Compaq and AT&T, DOS did overwrite the entire hard disk.)

When you combine the /Q and /U parameters, you get the fastest reformat possible. /Q prevents the scan for defects, which is the longest operation during formatting, and /U eliminates saving UNFORMAT information. The FORMAT command is restricted to simply erasing the DOS volume boot sector, FATs, and root directory, which it can do very quickly. In fact, a format using the /Q and /U parameters takes only a few seconds to complete no matter how large the disk.

For more information on the DOS FORMAT command, a master FORMAT command reference chart is in Appendix A. This explicit chart explains what all the FORMAT parameters do, and even describes some useful undocumented parameters I discovered.

Caring for and Handling Floppy Disks and Drives

Most computer users know the basics of disk care. Disks can be damaged or destroyed easily by the following:

- Touching the recording surface with your fingers or anything else

- Writing on a disk label with a ballpoint pen or pencil
- Bending the disk
- Spilling coffee or other substances on the disk
- Overheating a disk (leaving it in the hot sun or near a radiator, for example)
- Exposing a disk to stray magnetic fields

Despite all these cautions, disks are rather hardy storage devices; I can't say that I have ever destroyed one by just writing on it with a pen, because I do so all the time. I am careful, however, not to press too hard, which can put a crease in the disk. Also, simply touching a disk does not necessarily ruin it but rather gets the disk and your drive head dirty with oil and dust. The danger to your disks comes from magnetic fields that, because they are unseen, can sometimes be found in places you never dreamed of.

For example, all color monitors (and color TV sets) have, around the face of the tube, a degaussing coil used to demagnetize the shadow mask inside when the monitor is turned on. The coil is connected to the AC line and controlled by a thermistor that passes a gigantic surge of power to the coil when the tube is powered on, which then tapers off as the tube warms up. The degaussing coil is designed to remove any stray magnetism from the shadow mask at the front area of the tube. Residual magnetism in this mask can bend the electron beams so that the picture appears to have strange colors or be out of focus.

If you keep your disks anywhere near (within one foot) of the front of the color monitor, you expose them to a strong magnetic field every time you turn on the monitor. Keeping disks in this area is not a good idea because the field is designed to demagnetize objects, and indeed works well for demagnetizing disks. The effect is cumulative and irreversible.

Another major disk destructor is the telephone. The mechanical ringer in a typical telephone uses a powerful electromagnet to move the striker into the bell. The ringer circuit uses some 90 volts, and the electromagnetic fields have sufficient power to degauss a disk lying on the desk next to or partially underneath the phone. *Keep disks away from the telephone*. A telephone with an electronic ringer might not cause this type of damage to a disk, but be careful anyway.

Another source of powerful magnetic fields is an electric motor, found in vacuum cleaners, heaters or air conditioners, fans, electric pencil sharpeners, and so on. Do not place these devices near areas where you store disks.

Airport X-Ray Machines and Metal Detectors. People associate myths with things they cannot see, and we certainly cannot see data as it is stored on a disk, nor the magnetic fields that can alter the data.

One of my favorite myths to dispel is that the airport X-ray machine somehow damages disks. I have a great deal of experience in this area from having traveled around the country for the past ten years or so with disks and portable computers in hand. I fly about 150,000 miles per year, and my portable computer equipment and disks have been through X-ray machines more than 100 times each year.

Most people commit a fatal mistake when they approach the airport X-ray machines with disks or computers: they don't pass the stuff through! Seriously, X-rays are in essence just a form of light, and disks and computers are just not affected by X-rays at anywhere near the levels found in these machines. What can damage your magnetic media is the *metal detector*. Time and time again, someone with magnetic media or a portable computer approaches the security check. They freeze and say, "Oh no, I have disks and a computer—they have to be hand inspected." The person then refuses to place the disk and computer on the X-ray belt, and either walks through the metal detector with disks and computer in hand or passes the items over to the security guard, in very close proximity to the metal detector. Metal detectors work by monitoring disruptions in a weak magnetic field. A metal object inserted in the field area causes the field's shape to change, which the detector observes. This principle, which is the reason that the detectors are sensitive to metal objects, can be dangerous to your disks; the X-ray machine, however, is the safest area through which to pass either your disk or computer.

The X-ray machine is not dangerous to magnetic media because it merely exposes the media to electromagnetic radiation at a particular (very high) frequency. Blue light is an example of electromagnetic radiation of a different frequency. The only difference between X-rays and blue light is in the frequency, or wavelength, of the emission.

Electromagnetic radiation is technically a form of wave energy characterized by oscillating electric and magnetic fields perpendicular to one another. An electromagnetic wave is produced by an oscillating electric charge. This wave is not the same thing as a magnetic field. When matter intercepts electromagnetic energy, the energy is converted to thermal, electrical, mechanical, or chemical energy, but not to a magnetic field. Simply put, an electromagnetic wave generates either heat or an electrical alternating current in an object through which the wave passes.

I have been electrically shocked, for example, by touching metal objects in the vicinity of a high-powered amateur-radio transmitter. Your microwave oven induces thermal (kinetic) or even electrical energy in objects because of the same principle. Although a microwave oven is designed to induce kinetic energy in an irradiated substance's molecules, most of you know that when you place conductive (metal) objects in the microwave, an alternating electrical current also is generated, and you might even see sparks. This activity is a generation of electrical or mechanical energy, not of a magnetic field. Because a disk is not a good conductor, the only noticeable effect a high-powered electromagnetic field has on a floppy disk is the generation of kinetic (or thermal) energy. In other words, the only way that X-rays, visible light, or other radiation in these areas of the electromagnetic spectrum can damage a disk is by heating it.

Consider also that if electromagnetic radiation could truly magnetize a disk as a magnetic field can, all magnetic media (disks and tapes) in the world would be in danger. Much electromagnetic radiation is passing through you at this moment, and through all your disks and tapes as well. There is no danger of magnetic damage because the radiation's effect on an object is to impart electrical, thermal, mechanical, or chemical energy—*not to magnetize the object*. I am *not* saying that you cannot harm a disk with electromagnetic radiation, because you certainly can; the damage, however, is from the heating effects of the radiation.

You probably know what the sun's extremely powerful electromagnetic radiation can do to a disk. Just leave a disk lying in direct sunlight a while, and you can see the thermal effects of this radiation. A microwave oven would have basically the same cooking effect on a disk, only more intense! Seriously, at the levels of electromagnetic radiation to which we normally are exposed, or which are present in an airport X-ray machine, there is certainly no danger to your disks. The field strength is far too low to raise the temperature of the disk in any perceptible manner, and this radiation has no magnetic effect on a disk.

Some people worry about the effect of X-ray radiation on their system's EPROM (erasable programmable read-only memory) chips. This concern might actually be more valid than worrying about disk damage because EPROMs are erased by certain forms of electromagnetic radiation. In reality, however, you do not need to worry about this effect either. EPROMs are erased by direct exposure to very intense ultraviolet light. Specifically, to be erased, an EPROM must be exposed to a 12,000 uw/cm2 UV light source with a wavelength of 2,537 angstroms for 15 to 20 minutes, and at a distance of one inch. Increasing the power of the light source or decreasing the distance from the source can shorten the erasure time to a few minutes. The airport X-ray machine is different by a factor of 10,000 in wavelength, and the field strength, duration, and distance from the emitter source are nowhere near what is necessary for EPROM erasure. Be aware that many circuit-board manufacturers use X-ray inspection on circuit boards (with components including EPROMs installed) to test and check quality control during manufacture.

I have conducted my own tests: I passed one disk through different airport X-ray machines for two years, averaging two or three passes a week. The same disk still remains intact with all the original files and data, and never has been reformatted. I also have several portable computers with hard disks installed; one of them has been through the X-ray machines safely every week for more than four years. I prefer to pass computers and disks through the X-ray machine because it offers the best shielding from the magnetic fields produced by the metal detector standing next to it. Doing so also significantly lowers the "hassle factor" with the security guards because if I have it X-rayed, they usually do not require that I plug it in and turn it on.

Drive-Installation Procedures

The procedure for installing floppy drives is simple. You install the drive in two phases. The first phase is to configure the drive for the installation, and the second is to perform the physical installation. Of these two steps, the first one usually is the most difficult to perform, depending on your knowledge of disk interfacing and whether you have access to the correct OEM drive manuals.

Drive Configuration

Configuring a floppy drive consists of setting the jumpers and switches mounted on the drive to match the system in which the drive will be installed, as well as tailoring the function of the drive to the installer's requirements. Every drive has a stable of jumpers and switches, and many drives are different from each other. You will find no standards for what these jumpers and switches are called, where they should be located, or how they should be implemented. There are some general guidelines to follow, but to set up a specific drive correctly and know all the options available, you must have information from the drive's manufacturer, normally found in the original equipment manufacturer's (OEM) manual. The manual is a "must have" item when you purchase a disk drive.

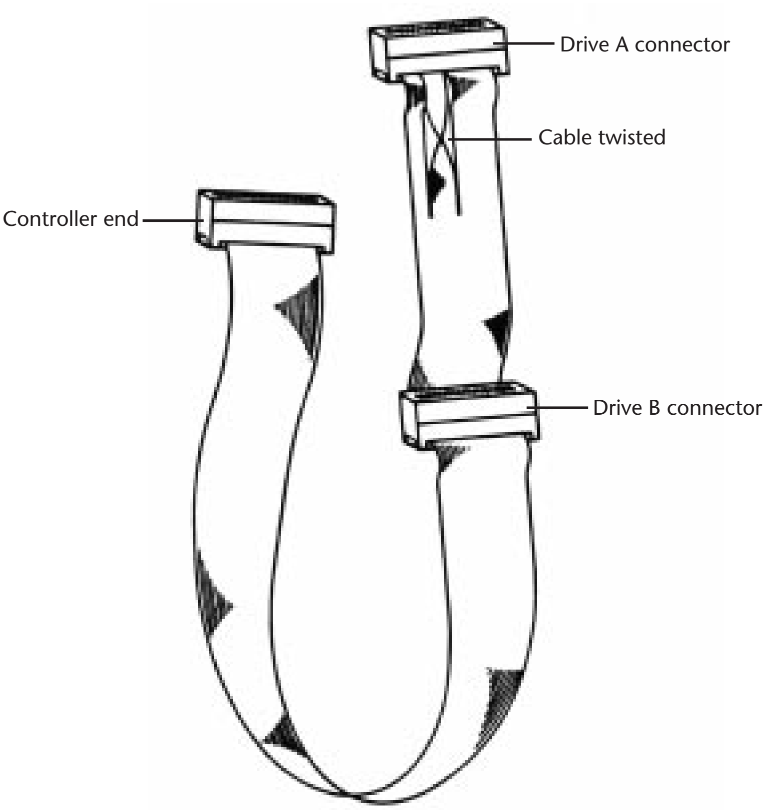
Although additional options might be available, most drives have several configuration features that must be set properly for an installation. These standard options typically need attention during an installation procedure:

- Drive select jumper
- Terminating resistor
- Diskette changeline or ready jumper
- Media sensor jumper

Each configuration item was discussed in more detail earlier in this chapter. The following section describes how these items are to be set for various installations.

Floppy drives are connected by a cabling arrangement called a *daisy chain*. The name is descriptive because the cable is strung from controller to drive to drive in a single chain. All drives have a drive select (sometimes called DS) jumper that must be set to indicate a certain drive's physical drive number. The point at which the drive is connected on the cable does not matter; the DS jumper indicates how the drive should respond. Most drives allow four settings, but the controllers used in all PC systems support only two on a single daisy-chain cable. The PC and XT floppy controllers, for example, support four drives but only on two separate cables—each one a daisy chain with a maximum of two drives.

Every drive on a particular cable must be set to have unique drive select settings. In a normal configuration, the drive you want to respond as the first drive (A) is set to the first drive select position, and the drive you want to respond as the second drive (B) is set to the second drive-select position. On some drives, the DS jumper positions are labeled 0, 1, 2, and 3; other drives use the numbers 1, 2, 3, and 4 to indicate the same positions.



For some drives then, a setting of DS0 is drive A. For others, however, DS1 indicates drive A. Likewise, some drives use a setting of DS1 for drive B, and others use a DS2 setting to indicate drive B. On some drives, the jumpers on the drive circuit board are unlabeled! In this case, consult the drive's manual to find out the descriptions of each jumper setting on the drive. A typical daisy-chain drive cable with this included "twist" is connected as shown in figure 13.11.

Fig. 13.11

A floppy controller cable showing the location of "the twist."

Make sure that the DS settings for every drive on a single daisy-chain cable are different, or both drives respond to the same signals. If you have incorrect DS settings, both drives respond simultaneously or neither drive responds at all.

The type of cable you use can confuse the drive select configuration. IBM puts in its cables a special twist that electrically changes the DS configuration of the drive plugged in after the twist. This twist causes a drive physically set to the first DS position (A) to appear to the controller to be set to the second DS position (B). If the first drive on the cable was before the twist in the cable and was set to the second DS position (B), the controller would see a conflict. To the controller, both drives would appear to be set to the second DS position (B), although physically they looked as though they were set differently. In essence, the system would think that two B drives were installed. The adjustment for this problem is simple: When this type of cable is used, both drives should be set to the second DS position. The drive plugged in to the connector farthest from the controller, which is after the twist in the cable, then would have the physical second-DS-position setting appear to be changed to a first-DS-position setting. Then the system would see this drive as A, and the drive plugged into the middle cable connector still would appear as B.

An IBM-style floppy cable is a 34-pin cable with lines 10 through 16 sliced out and cross-wired (twisted) between the drive connectors (refer to fig. 13.11). This twisting "cross-wires" the first and second drive-select and motor-enable signals, and therefore inverts the DS setting of the drive following the twist. All the drives in a system using this type of cable, therefore—whether you want them to be A or B, are physically jumpered the same way; installation and configuration are simplified because both floppies can be preset to the second DS position. Some drives used by IBM, in fact, have had the DS "jumper" setting permanently soldered into the drive logic board.

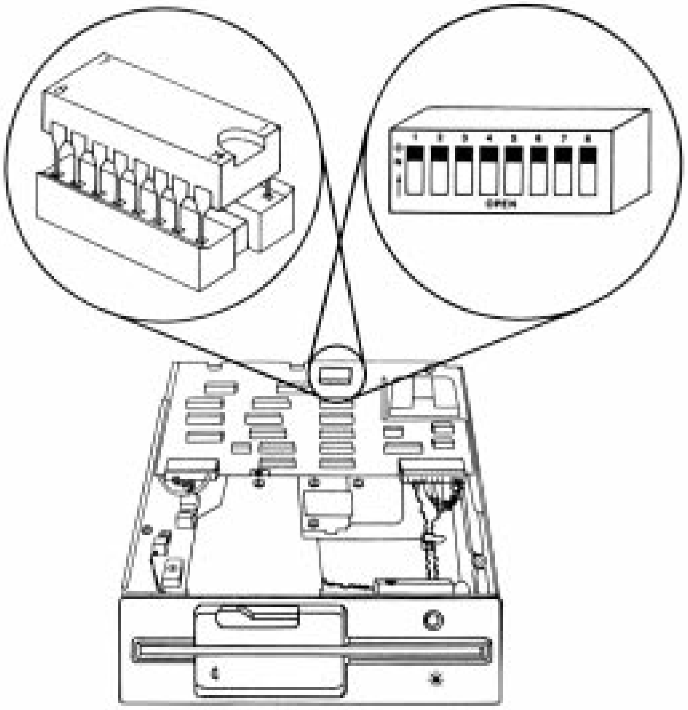
Most bare drives you purchase have the DS jumper already set to the second position, which is correct for the majority of systems that use a cable with the twisted lines. Although this setting is correct for the majority of systems, if you are using a cable with no twist, you will have to alter this setting on at least one of the two drives. Some systems come with only a single floppy drive and no provisions for adding a second one. These types of systems often use a floppy cable with only one drive connector attached. This type of cable does not have any twisted lines, so how do you set up a drive plugged into this cable? Because there is no twist, the DS setting you make on the drive is exactly what the controller sees. You can attach only one drive, and it should appear to the system as A—therefore, set the drive to the first DS position.

Most IBM-compatibles use a floppy cable with the twisted lines between the drive connectors. Drives plugged into this type of cable have their DS jumpers set to the second position. Drives on a single floppy cable or a cable with no twisted lines are set to the first DS position.

A terminating resistor should be placed (or enabled) in any drive plugged into the physical end of a cable. The function of this resistor is to prevent reflections or echoes of signals from reaching the end of the cable. All new drives have this resistor installed by default. The terminating resistor should be removed or disabled for drives that are not the farthest away from the controller. Most 3 1/2-inch floppy drives use the distributed-termination technique, in which the installed terminating resistors are permanently installed, and are non-removable and cannot be disabled. The resistor value in these drives is adjusted appropriately so that, in effect, the termination is distributed among both drives. When you mix 5 1/4-inch and 3 1/2-inch drives, you should enable or disable the terminators on the 5 1/4-inch drives appropriately, according to their position on the cable and ignore the non-changeable settings on the 3 1/2-inch drives.

In a typical cabling arrangement for two 5 1/4-inch floppies, for example, the terminating resistor is installed in drive A (at the end of the cable), and this resistor is removed from the other floppy drive on the same cable (B). The letter to which the drive responds is not important in relation to terminator settings; the important issue is that the drive at the end of the cable has the resistor installed and functioning and that other drives on the same cable have the resistor disabled or removed.

The terminating resistor usually looks like a memory chip; it might be white, blue, black, gray, or some other color, and memory chips usually are just black. IBM always labels the resistor with a T-RES sticker for easy identification. On some systems, the resistor is a built-in device enabled or disabled by a jumper or series of switches. If you have the removable type, make sure to store the resistor in a safe place because you might need it later. Figure 13.12 shows the location and appearance of the terminating resistor or switches on a typical floppy drive. Because most 3 1/2-inch drives have a form of automatic termination, there is no termination to configure; also, some 5 1/4-inch drives, such as Toshiba drives, have a permanently installed terminating resistor enabled or dis-



abled by a jumper labeled TM.

Fig. 13.12
A typical floppy drive terminating resistor, or termination switches.

Table 13.12 explains how a drive should be configured relative to the drive-select jumper and terminating resistor. You can use the table as a universal drive-select and terminating resistor configuration chart that applies to all types of drives, including floppy disk drives and hard disks.

Table 13.12 Configuring Drive-Select Jumpers and Terminating Resistors

Drive	Twisted Cable	Straight Cable
A: drive (end connector)	DS = second TR installed	DS = first TR installed
B: drive (center connector)	DS = second TR removed	DS = second TR removed

DS = Drive select position
TR = Terminating resistor

The assumption in table 13.12 is that you always plug drive B into the center connector on the cable and drive A into the end connector. This arrangement might seem strange at first, but it is virtually required if you ever assemble a single-drive system. The logical first (A) drive should be the end, or last, drive on the cable, and should be terminated. The twist in the cable is almost always between the two drive connectors on a cable and not between the controller and a drive.

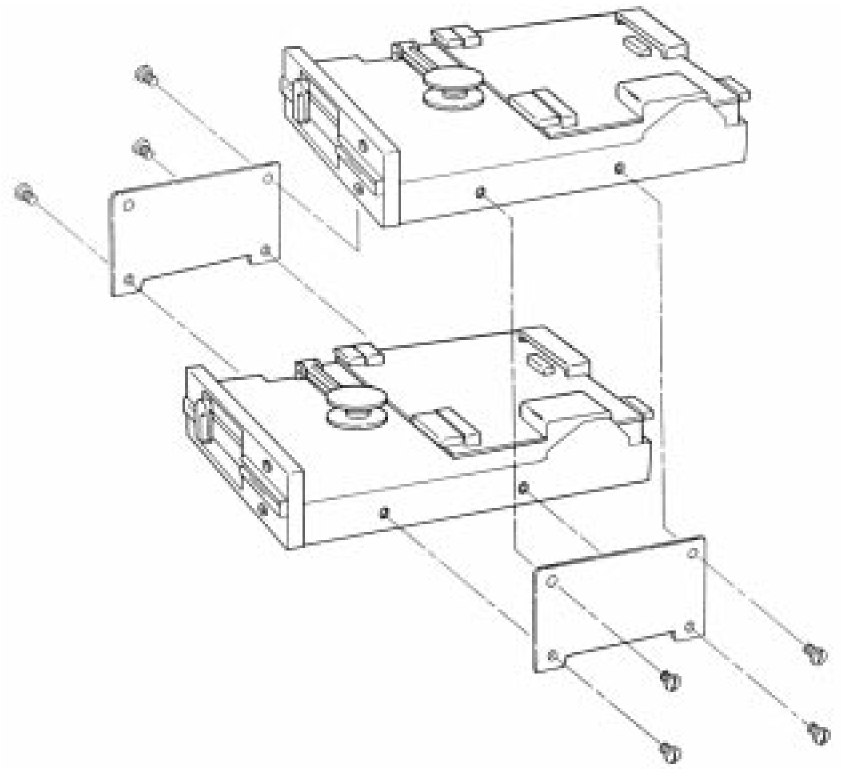
Two other options might be available for you to set: the status of pin 34 on the drive's connector, and the function of a media-sensor feature. The guidelines for setting these options follow.

If the drive is a 5 1/4-inch 360K drive, set the status of pin 34 to Open (disconnected) regardless of the type of system in which you are installing the drive. The only other option normally found for pin 34 on 360K drives is Ready (RDY), which is incorrect. If you are using only a low-density controller, as in a PC or XT, pin 34 is ignored no matter what is sent on it. If the drive you are installing is a 5 1/4-inch 1.2M or 3 1/2-inch 720K, 1.44M, or 2.88M drive, be sure to set pin 34 to send the Disk Change (DC) signal. The basic rule is simple:

For 360K drives only, pin 34 = Open (disconnected)

For any other drive, pin 34 = Disk Change

The media-sensor setting is the easiest to describe. Only 1.44M and 2.88M drives have a media sensor. The best rule to follow is to set these drives so that the sensor is enabled; this step enables the sensor to control the drive's recording mode and, therefore, the drive's write-current level.



Physical Installation

When you physically install a drive, you plug in the drive. Here, your concerns are using the correct brackets and screws for the system and the correct drive you are installing.

A special bracket usually is required whenever you install a half-height drive in place of an earlier full-height unit (see fig. 13.13). The brackets enable you to connect the two half-height drives together as a single full-height unit for installation. Remember also that nearly all floppy drives now use metric hardware; only the early, American-manufactured drives use the standard English threads.

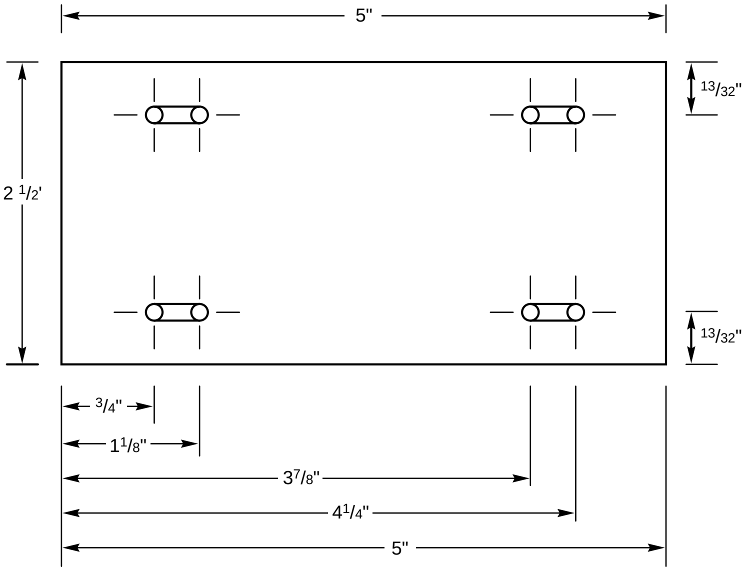
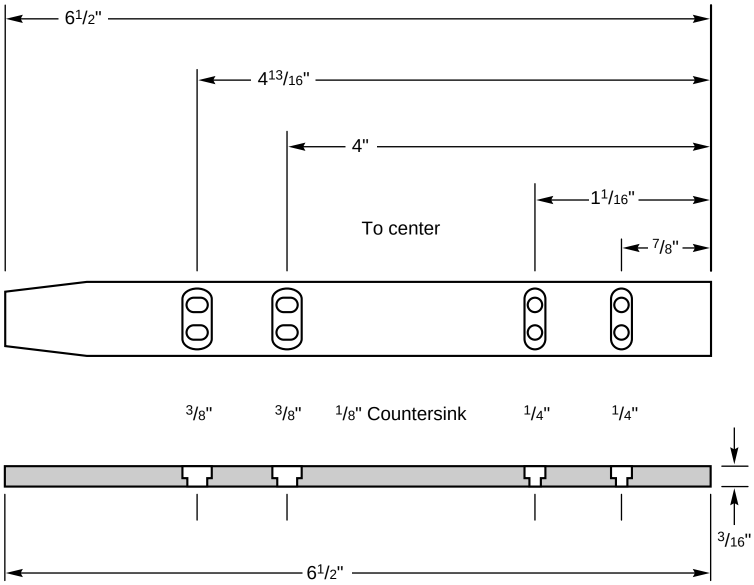


Fig. 13.13
Installing half-height drives in a full-height bay with adapter plates.

You can get these adapter plates from most vendors who sell drives, but sometimes they charge as much as $10 for basically a piece of sheet metal with four holes drilled in it! Several companies listed in Appendix B specialize in cables, brackets, screw hardware, and other items useful in assembling systems or installing drives. The template shown in figure 13.14 also will guide you if you want to make your own. I usually use a piece of galvanized sheet metal like that used in ventilation ductwork for the stock, which can be easily obtained at most hardware stores.

Fig. 13.14
The dimensions of a typical drive adapter plate.

Another piece of drive-installation paraphernalia you need is the rails used in installing disk drives in AT systems. Most IBM-compatible systems follow the IBM standard for rail design. Again, you can purchase these from some of the vendors listed in Appendix B. If you want to construct your own, figure 13.15 shows the construction of a typical IBM-style drive rail. These rails can be made from metal, but usually are made from plastic. They probably can even be made from wood. Drives installed in an AT are grounded to the system chassis through a separate ground wire and tab, which is why the rails do not need to be made from a conductive material. I find it more cost effective to purchase the rails rather than make them.



As you might expect, Compaq uses a slightly different rail construction. The vendors mentioned in Appendix B that sell cables, brackets, and other installation accessories also carry the Compaq-style rails.

When you connect a drive, make sure that the power cable is installed properly. The cable normally is keyed so that it cannot be plugged in backward. Also, install the data and control cable. If no key is in this cable, which allows only a correct orientation, use the colored wire in the cable as a guide to the position of pin 1. This cable is oriented correctly when you plug it in so that the colored wire is plugged into the disk drive connector toward the cut-out notch in the drive edge connector.

Fig. 13.15

A typical AT-drive mounting rail.

Floppy Drive Installation Summary

To install and set up a floppy drive properly, you must understand and set up primarily four different configuration items on the drive as follows:

- Drive select (DC) jumper setting
- Terminating resistor (TR) enabled or disabled
- Send disk change (DC) or no signal on pin 34
- Enable media sensor

This section explained the proper settings for these items in virtually any installation situation you might encounter.

For more information about configuring and installing a specific drive, you can use several resources. Obviously, this book contains much information about configuring and installing floppy disk drives—make sure that you have read it all! The best source of information about certain drives or controllers is the original equipment manufacturer's (OEM) documentation. These manuals tell you where all configuration items are located on the drive; what they look like; and how to set them. Unfortunately, most of the time you do not receive this detailed documentation when you purchase a drive or controller; instead, you must contact the OEM to obtain it.

Troubleshooting and Correcting Problems

The majority of floppy drive problems are caused primarily by improper drive configuration, installation, or operation. Unfortunately, floppy drive configuration and installation is much more complicated than the average technician seems to realize. Even if you had your drive "professionally" installed, it still might have been done incorrectly.

This section describes some of the most common problems that stem from improperly installing or configuring a drive. Also discussed are several problems that can occur from improperly using drives and disks. Solutions to these problems are presented also.

Handling the "Phantom Directory" (Disk Change)

One of the most common mistakes people make when installing a disk drive is incorrectly setting the signals sent by the drive on pin 34 of the cable to the controller. All drives *except* the 360K drive must be configured so that a Disk Change (DC) signal is sent along pin 34 to the controller.

If you do not enable the DC signal when the system expects you to, you might end up with trashed disks as a result. For example, a PC user with disk in hand might say to you, "Moments ago, this disk contained my document files, and now it seems as though my entire word processing program disk has mysteriously transferred to it. When I attempt to run the programs that now seem to be on this disk, they crash or lock up my system." Of course, in this case the disk has been damaged, and you will have to perform some data-recovery magic to recover the information for the user. My book, *Que's Guide to Data Recovery,* contains more information about data-recovery techniques. A good thing about this particular kind of problem is that recovering most—if not all—the information on the disk is entirely possible.

You also can observe this installation defect manifested in the "phantom directory" problem. For example, you place a disk with files on it in the A drive of your AT-compatible system and enter the **DIR A:** command. The drive starts spinning, the access light on the drive comes on, and after a few seconds of activity, the disk directory scrolls up the screen. Everything seems to be running well. Then you remove the disk and insert in drive A a different disk with different files on it and repeat the **DIR A:** command. This time, however, the drive barely (if at all) spins before the disk directory scrolls up the screen. When you look at the directory listing that has appeared, you discover in amazement that it is the same listing as on the first disk you removed from the drive.

Understand that the disk you have inserted in the drive is in danger. If you write on this disk in any way, you will cause the file-allocation tables and root-directory sectors from the first disk (which are stored in your system's memory) to be copied over to the second disk, thereby "blowing away" the information on the second disk. Most AT-compatible systems with high- or low-density controllers utilize a floppy disk caching system that buffers the FATs and directories from the floppy disk that was last read in system RAM. Because this data is kept in memory, these areas of the disk do not have to be reread as frequently. This system greatly speeds access to the disk.

Opening the door lever or pressing the eject button on a drive normally sends the Disk Change signal to the controller, which in turn causes DOS to flush out the floppy cache. This action causes the next read of the disk drive to reread the FAT and directory areas. If this signal is not sent, the cache is not flushed when you change a disk, and the system acts as though the first disk still is present in the drive. Writing to this newly inserted disk writes not only the new data but also either a full or partial copy of the first disk's FAT and directory areas. Also, the data is written to what was considered free space on the first disk, which might not be free on the subsequent disk and results in damaged files and data.

This problem has several simple solutions. One is temporary; the other is permanent. For a quick, temporary solution, press Ctrl-Break or Ctrl-C immediately after changing any disk, to force DOS to manually flush the floppy I/O buffers. This method is exactly how the old CP/M operating system used to work. After pressing Ctrl-Break or Ctrl-C, the next disk access rereads the FAT and directory areas of the disk and places fresh copies in memory. In other words, you must be sure that every time you change a disk, the buffer gets flushed. Because these commands work only from the DOS prompt, you must not change a disk while working in an application.

A more permanent and correct solution to the problem is simple—just correct the drive installation. In my experience, incorrect installation is the root cause of this problem nine out of ten times. Remember this simple rule: *If a jumper block is on the disk drive labeled DC, you should install a jumper there*. If you are absolutely certain that the installation was correct—for example, the drive has worked perfectly for some time, but then suddenly develops this problem—check the following list of items, all of which can prevent the Disk Change signal from being sent:

- Drive configuration/Setup. Make sure that the DC jumper is enabled; check CMOS Setup.
- Bad cable. Check for continuity on pin 34.
- Bad Disk Change sensor. Clean sensor or replace drive and retest.
- Bad drive logic board. Replace drive and retest.

- Bad controller. Replace controller and retest.
- Wrong DOS OEM version.

The last of these checklist items can stump you because the hardware seems to be functioning correctly. As a rule, you should use only the DOS supplied by the same OEM as the computer system on the system. For example, use IBM DOS on IBM systems, Compaq DOS on Compaq systems, Zenith DOS on Zenith systems, Toshiba DOS on Toshiba systems, Tandy DOS on Tandy systems, and so on. This problem is most noticeable with some laptop systems that apparently have a modified floppy controller design, such as some Toshiba laptops. On many of these systems, you *must* use the correct (Toshiba, for example) OEM version of DOS.

Handling Incorrect Media-Sensor Operation

Incorrect media-sensor operation occurs on only 1.44M or 2.88M, 3 1/2-inch, high-density drives—the only drives that have a media sensor. Again, this is largely a drive-configuration problem because the installer did not enable the sensor when it should have been enabled. You would think that the sensor would be set correctly when you purchase a drive, but that is not always the case. Never assume that a drive is preconfigured properly for your system. Remember that drive manufacturers sell drives for systems other than IBM-compatibles. Sometimes it is hard to remember that many other types of computers exist other than just IBMs or IBM clones.

If the media sensor is not operational, the controller likely will leave the drive in a state in which high write-current always is applied to the heads during write operations. This state is OK for high-density disks, but when low-density disks are used, random and sporadic read and write failures occur, usually ending with the DOS message `Abort, Retry, Ignore, Fail?`.

Another symptom of incorrect media-sensor operation is generating double-density disks that seem eventually to lose data—perhaps over a few weeks or months. This loss often can be traced back to an improperly configured media sensor on the drive. In some systems, the problem might be more obvious, such as not being capable of formatting or writing successfully on 720K disks. If your system can format a 720K disk to 1.44M without punching any extra holes in the disk, it is an immediate alert that the media sensor is not enabled.

Handling Problems Caused by Using Double-Density Disks at High Density

If you attempt to format a 5 1/4-inch, double-density disk at high-density format, you usually will hear several retries from the drive as DOS finds a large amount of bad sectors on the disk. When the format is completed, hundreds of kilobytes in bad sectors usually are reported. Most people would never use this disk. Because the 5 1/4-inch disks are so radically different from one another in terms of magnetic coercivity and media formulation, the double-density disks do not work well carrying a high-density format.

Problems are seen more often with the 3 1/2-inch disks because the double-density disks are not nearly as different from the high-density disks compared to the 5 1/4-inch versions, although they indeed are different. Because the 3 1/2-inch DD and HD disks differ less, however, a double-density, 3 1/2-inch disk usually accepts a high-density format with no bad sectors reported. This acceptance is unfortunate because it causes most users to feel that they are safe in using the disk for data storage.

A 3 1/2-inch, double-density disk with a 1.44M, high-density format initially seems to work with no problem. If you fill this type of double-density disk with 1.44M of data and store it on a shelf, you will notice that eventually the recording degrades, and the data becomes unreadable. Several months might pass before you can detect the degradation, but then it is too late. From talking to hundreds of my clients, I have found that the average "half life" of such a recording is approximately six months from the time the data is written to the time that one or more files suddenly have unreadable sectors. In six months to a year, much of the rest of the disk rapidly degrades until all the data and files have extensive damage. The recording simply destroys itself during this time. I have substantiated this situation with my own testing and my clients' experiences. If the data is reread and rewritten periodically before any degradation is noticeable, then the recording can be "maintained" for longer periods of time before data is lost.

The technical reasons for this degradation were explained earlier in this chapter. In a sense, the disk eventually performs a self-erasure operation. Again, the time frame for damage seems to be approximately six months to a year from the time the data is written. I certainly expect my disks to hold data for more than six months; in fact, data written properly to your disks should be readable many, many years from now.

If you have been using double-density, 3 1/2-inch disks with high-density formats, you are asking for problems. Using these types of disks for backup, for instance, is highly inappropriate! Many people use double-density disks as high-density disks to save money. You should realize that high-density disks are not very expensive anymore; *data-recovery services, however, are very expensive*.

If you have a disk that has been formatted improperly and is developing read problems, the first thing to do is DISKCOPY the disk immediately to another proper-density disk. Then you can survey the damage and make repairs to the new copy. For a more detailed investigation of the subject of data recovery, see *Que's Guide to Data Recovery*.

Handling Track-Width Problems from Writing on 360K Disks with a 1.2M Drive

As discussed earlier in this chapter, the 5 1/4-inch, high-density drives usually write a narrower track than the 5 1/4-inch, double-density drives. Therefore, when you use a high-density drive to update a double-density disk originally formatted or written in a double-density drive, the wider tracks written by the double-density drive are not completely overwritten by the high-density drive. Of course, if the double-density disk is newly formatted and subsequently written in only a high-density drive (although at the proper 360K format), there is no problem with overwrites—that is, until you update the disk with a double-density (wide-track) drive and then update it again with a high-density (narrow-track) drive. In that case, you again have a wider track with a narrow-track update embedded within—but not completely covering it.

You must remember *never* to use a high-density drive to write on a double-density disk previously written by a double-density drive. This procedure makes the disk unreadable by the double-density drive, but usually still readable by the high-density drive. In fact, the best way to recover information from a disk that has been incorrectly overwritten in this manner is to use a high-density disk drive to perform a DISKCOPY operation of the disk to a new, blank, never previously formatted, low-density disk.

Handling Off-Center Disk Clamping

Clamping the disk off-center in the drive has to be absolutely the most frequently encountered cause of problems with floppy drives. In my worldwide troubleshooting seminars, we run the PC systems with the lid off for most of the course. Whenever someone has a problem reading or booting from a floppy disk, I look down at the top of the exposed disk drive on the system while it is spinning to see whether the disk has been clamped by the drive hub in an off-center position. *More often than not, that is the problem.* I know that the disk is clamped off-center because the disk wobbles while it rotates. Ejecting and reinserting the disk so that it is clamped properly usually makes the disk reading or booting problem disappear immediately. This step might solve the problem in most cases, but it is not much help if you have formatted or written a disk in an off-center position. In that case, all you can do is try to DISKCOPY the improperly written disk to another disk and attempt various data-recovery operations on both disks.

I use a technique for inserting floppy disks that has eliminated this problem for me. After inserting a disk into a drive, I always take an extra half-second to wiggle the drive lever or door, first down, and then up, and then down again to clamp the disk rather than simply push the door or lever down once to clamp it. The reason is that the first partial closing of the lever serves to center the disk in its jacket so that the second motion allows the drive hub to clamp the disk properly in a centered position. If I were in charge of training for a large organization, I would make sure that all the basic PC starter classes taught proper disk handling, including insertion and on-center clamping in the drive.

Note that the 3 1/2-inch drives are virtually immune to this type of problem because of the different type of clamping and centering mechanisms they use. Some 5 1/4-inch drives have adopted a more reliable clamping mechanism similar to the 3 1/2-inch drives. Canon makes some of these new 5 1/4-inch drives, used by IBM and Compaq. The newest version used by IBM in some of its PS/2 systems is totally motorized. You merely slide the disk into the drive slot, and the drive grabs the disk and electrically pulls it in and centers it. These drives also include a motorized eject button.

Realigning Misaligned Drives

If your disk drives are misaligned, you will notice that other drives cannot read disks created in your drive, and you might not be able to read disks created in other drives. This situation can be dangerous if you allow it to progress unchecked. If the alignment is bad enough, you probably will notice it first in the incapability to read original application program disks, while still being able to read your own created disks. The Drive Probe program from Accurite for checking the alignment and operation of floppy drives is discussed later in this chapter.

To solve this problem, you can have the drive realigned. I don't always recommend realigning drives because of the low cost of simply replacing the drive compared to aligning one. Also, an unforeseen circumstance catches many people off guard: You might find that your newly aligned drive might not be able to read all your backup or data disks created while the drive was out of alignment. If you replace the misaligned drive with a new one and keep the misaligned drive, you can use it for DISKCOPY purposes to transfer the data to newly formatted disks in the new drive.

Repairing Floppy Drives

Attitudes about repairing floppy drives have changed over the years primarily because of the decreasing cost of drives. When drives were more expensive, people often considered repairing the drive rather than replacing it. With the cost of drives decreasing every year, however, certain labor or parts-intensive repair procedures have become almost as expensive as replacing the drive with a new one.

Because of cost considerations, repairing floppy drives usually is limited to cleaning the drive and heads and lubricating the mechanical mechanisms. On drives that have a speed adjustment, adjusting the speed to within the proper operating range also is common. Note that most newer half-height drives and virtually all 3 1/2-inch drives do not have an adjustment for speed. These drives use a circuit that automatically sets the speed at the required level and compensates for variations with a feedback loop. If such an auto-taching drive is off in speed, the reason usually is that the circuit failed. Replacement of the drive usually is necessary.

Cleaning Floppy Drives

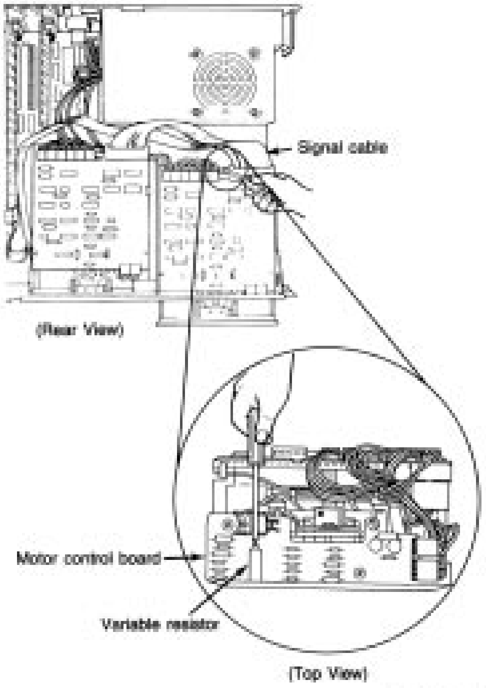
Sometimes read and write problems are caused by dirty drive heads. Cleaning a drive is easy; you can proceed in two ways. In one method, you use one of the simple head-cleaning kits available from computer- or office-supply stores. These devices are easy to operate and don’t require the system unit to be open for access to the drive. The other method is the manual method: You use a cleaning swab with a liquid such as pure alcohol, Freon, or trichloroethane. With this method, you must open the system unit to expose the drive and, in many cases (especially in earlier full-height drives), also remove and partially disassemble the drive. The manual method can result in a better overall job, but usually the work required is not worth the difference.

The cleaning kits come in two styles: The wet type uses a liquid squirted on a cleaning disk to wash off the heads; the dry kit relies on abrasive material on the cleaning disk to remove head deposits. I recommend that you never use the dry drive-cleaning kits. Always use a wet system in which a liquid solution is applied to the cleaning disk. The dry disks can prematurely wear the heads if used improperly or too often; wet systems are very safe to use.

The manual drive-cleaning method requires that you have physical access to the heads, in order to swab them manually with a lint-free foam swab soaked in a cleaning solution. This method requires some level of expertise: Simply jabbing at the heads incorrectly with a cleaning swab might knock the drive heads out of alignment. You must use a careful in-and-out motion, and lightly swab the heads. No side-to-side motion (relative to the way the heads travel) should be used; this motion can snag a head and knock it out of alignment. Because of the difficulty and danger of this manual cleaning, for most applications I recommend a simple wet-disk cleaning kit because it is the easiest and safest method.

One question that comes up repeatedly in my seminars is “How often should you clean a disk drive?” Only you can answer that question. What type of environment is the system in? Do you smoke cigarettes near the system? If so, cleaning would be required more often. Usually, a safe rule of thumb is to clean drives about once a year if the system is in a clean office environment in which no smoke or other particulate matter is in the air. In a heavy-smoking environment, you might have to clean every six months or perhaps even more often. In dirty industrial environments, you might have to clean every month or so. Your own experience is your guide in this matter. If DOS reports drive errors in the system by displaying the familiar DOS `Abort, Retry, Ignore` prompt, you should clean your drive to try to solve the problem. If cleaning does solve the problem, you probably should step up the interval between preventive-maintenance cleanings.

In some cases, you might want to place a (very small) amount of lubricant on the door mechanism or other mechanical contact points inside the drive. *Do not use oil;* use a pure silicone lubricant. Oil collects dust rapidly after you apply it and usually causes the oiled mechanism to gum up later. Silicone does not attract dust in the same manner and can be used safely. Use very small amounts of silicone; do not drip or spray silicone inside the drive. You must make sure that the lubricant is applied only to the part that needs it. If the lubricant gets all over the inside of the drive, it may cause unnecessary problems.



Used with permission from IBM Corporation.

Setting the Floppy Drive Speed Adjustment

Most older 5 1/4-inch floppy disk drives, especially full-height drives, have a small variable resistor used to adjust the drive's rotational speed. In particular, the Tandon and CDC full-height drives used by IBM in the PC and XT systems have this adjustment. The location of this variable resistor is described in the hardware-maintenance reference manuals IBM sells for these systems.

If you have a Tandon drive, you make the adjustment through a small, brass screw on a variable resistor mounted on the motor control board, attached to the rear of the drive (see fig. 13.16). The resistor is usually blue, and the screw is brass. To gauge the speed, you can use a program such as Drive Probe, by Accurite; IBM's Advanced Diagnostics, supplied with the hardware-maintenance and service manual; or even a purely mechanical method that relies on a fluorescent light to act as a strobe.

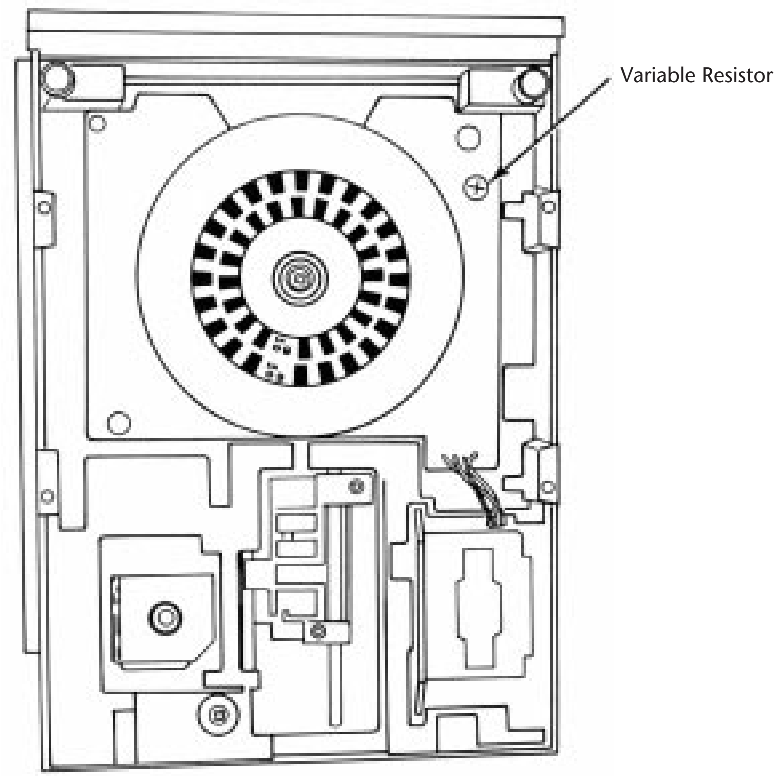


Fig. 13.16

The drive-speed adjustment for the Tandom TM-100 series drive.

The software methods use a disk to evaluate the running speed of the drive. Usually, you turn the screw until the speed reads correctly (300 RPM) according to the program you use. The mechanical method requires you to remove the drive from the system and place it upside down on a bench. Sometimes the drive is set sideways on the power supply so that the drive's case is grounded. Then the underside of the drive is illuminated by a standard fluorescent light. The light acts as a strobe that flashes 60 times per second because of the cycling speed of the AC line current. On the bottom of the drive spindle are strobe marks for 50 Hz and 60 Hz (see fig. 13.17). Because 60 Hz power is used in the United States, you should use the 60 Hz marks. The 50 Hz marks are used for (50 cycle) European power. While the drive is running, turn the small screw until the strobe marks appear to be stationary, much like the "wagon wheel effect" you see in old western movies. When the marks are completely stationary as viewed under the light, the drive's rotation speed is correct.

Fig. 13.17
Strobe marks and speed adjustment on a typical half-height drive.

With CDC drives, the adjustment resistor is mounted on the logic board, which is on top of the drive. The small, brass screw to the left of the board is the one you want. Other drives also might have an adjustment. The best way to tell whether a drive has a speed adjustment is to look for the telltale strobe marks on the spindle of the drive. If the marks are there, the drive probably has an adjustment; if the marks are not there, the drive probably has an automatic speed circuit and requires no adjustment. The OEM manual for the drive has information about all these adjustments (if any) and where you make them.

Aligning Floppy Disk Drives

Aligning disk drives is usually no longer done because of the high relative cost. To align a drive properly requires access to an oscilloscope (for about $500), a special analog-alignment disk ($75), and the OEM service manual for the drive; also, you must spend half an hour to an hour aligning the drive.

A new program, Drive Probe, by Accurite, uses special test disks called High-Resolution Diagnostic (HRD) disks. These disks are as accurate as the analog alignment disks (AAD) and eliminate the need for an oscilloscope to align a drive. You cannot use any program that relies on the older Digital Diagnostic Disk (DDD) or Spiral format test disks because they are not accurate enough to use to align a drive. The Drive Probe and HRD system can make an alignment more cost-effective than before, but it is still a labor-intensive operation. Figure 13.18 shows the main menu of the Accurite Drive Probe program, which offers various functions, including fully automatic or manual drive testing.

Fig. 13.18
The main menu screen from Accurite Drive Probe.

With the price of most types of floppy drives hovering at or below the $75 mark, aligning drives usually is not a cost-justified alternative to replacement. One exception exists. In a high-volume situation, drive alignment might pay off. Another alternative is to investigate local organizations that perform drive alignments, usually for $25 to $50. Weigh this cost against the replacement cost and age of the drive. I have purchased new 360K floppy drives for *as low as $30*. At these prices, alignment is no longer a viable option.

Summary

This chapter examined floppy drives and floppy media (disks) in great detail. One of the most important things to do when installing a drive in a system is to make sure that the drive is configured correctly. This chapter discussed drive configuration also. With this information, installing drives correctly should be an easy task.

This chapter also discussed many problems that confound floppy drive users, such as reading and writing double-density disks with high-density drives. It discussed thoroughly the differences between high- and double-density drives and disks, and mentioned the consequences of using the wrong type of disk in the wrong drive. Simple drive servicing, such as cleaning and speed adjustment, were explained so that these operations can be performed in-house. After reading this chapter, you should know much about floppy drives. Chapter 14 discusses hard disk drives and controllers.