

Fast, Enhanced: The New ATA and ATAPI Standards

Written by **Rakesh Dubey**

A huge number of inexpensive, capable PC computers are built around IDE-style hard disks. Until recently, the improved speed and flexibility of SCSI systems proved a compelling argument to stay away from IDE-based computers. But manufacturers are cooperating more closely to improve the specifications for IDE-style equipment. The latest round of IDE products are faster than ever, and recent specifications promise to connect more kinds of equipment than hard disks. A welter of names has sprung up, however, confusing buyers and vendors alike. This article sorts out the terms to see just what features we can depend on and what's still vapor.

(Note that sidebars and marginal notes in the printed journal are denoted here by smaller type with bars above and below the item.)

Although most NEXTSTEP systems use SCSI hard disk drives, recent improvements in IDE features—especially increased transfer speeds, support for more and larger hard disks, and support for other kinds of storage devices—have made IDE more attractive for computer users looking for a lower-cost desktop solution.

But the winds of change are scattering misleading acronyms around like leaves: IDE, EIDE, Fast

ATA, ATA-1, ATA-2, ATA-3, and ATAPI.

Current compatibility problems stem from competing advances, but as the industry moves forward, players converge. The basic improvement areas are the physical attachments including bus adaptor circuitry and device controller circuitry, a specification for a command set that device drivers use to query and control the device, and relations to the rest of the system, most notably system BIOS.

A LOOK BACK

Before the advent of a standard PC design, there were several hardware ^{layers} for managing hard disks. A computer system might have an adaptor card that would be connected to a controller card in the disk assembly. The controller card in turn would interface with the electronics on the disk drive itself.

The adaptor provided the hardware interface between the host bus and drive controller. It took signals from the host bus and converted them into a form that the drive controller could understand. The controller interpreted the signals received from the adaptor into signals that controlled the drive. The drive handled the medium, keeping the disks spinning at a constant rate, positioning the heads, and so on.

The ST-506 interface was originated by Shugart (now Seagate) for communicating between the disk and the PC host system. The ST-506 (also ST-506/412) could support RLL, MFM, or similar schemes for data encoding.

The ST-506/412 drives used a very low-level hardware interface: the operation of the drives depended on adaptors. Adaptors were difficult for users to configure, and the design of adaptors, controllers, and drives was very much interdependent, presenting several problems:

- It wasn't possible to increase data transfer speeds.
- The host adaptor had to parse raw data from the device.
- Users had to buy an adaptor that would work with a drive.

- The physical cabling was complex, with one cable for data and another for control signals.

In the ST-506 interface, data was delivered as a serial bit-stream to the host from the disk controller. The entire data (including formatting information) was transferred to disk, and the rate of data transfer was tied to the controller. Unfortunately, such a scheme limited the speed of data transfer from the disk to the device as well as the kind of devices that could be connected to a controller.

The ST-506/412 (MFM/RLL) disk drives with their cumbersome adaptors were difficult to configure. The fact that they had both a data cable and a control cable plagued them with reliability problems.

The IBM PC XT improved on this concept by integrating the adaptor and controller. The resulting hardware (called the *controller*) could directly interface with the disk.

The final step was moving the controller from the host system and putting it on the disk itself.

A company called Xebec was the first to begin to integrate these separate circuits. Western Digital had an early association with the technology because they had designed controllers for IBM's PC AT (WD 1002). The later ESDI disk drives (pioneered by Maxtor) could be much faster than ST-506s, but they had the same fundamental limitation. The drive-controller interface was specified at a low level, specific to each device, leading to the expected compatibility problems between the disk and the controller.

ATA = IDE

More than 90% of hard disks sold today for PCs are sold under the rubric of IDE. Until recently, these conformed to the ATA-1 specification.

The first ATA (AT Attachment) disk drives were made for Compaq in 1985 because its portable (actually *luggable*) computer didn't have an extra slot for a hard disk controller. Compaq, Western Digital, and Control Data Corporation worked together to integrate the WD 1003 controller into a Conner disk. The result was significantly faster than the prevalent MFM and RLL type drives,

cost less, and was easy to configure.

Compaq popularized the term IDE to refer to ATA disks. The name IDE is shorthand for Integrated Device Electronics and comes from the fact that these drives integrated the controllers in the disk drives themselves. IDE is a generic term and can be used to refer to any integrated disk, including SCSI. However, it is popularly used only to refer to ATA disks and does not include the mutually incompatible IDE disks and controllers for XTs, ATs, and MCA (IBM's Micro Channel Architecture).

Compared to the ST-506, configuring IDE drives was easy. They were universally supported by all PC BIOS, required no device drivers, and had no need for termination. Only one jumper was needed for configuring a driver as a slave. Last but not the least, the same signal lines were connected to both master and slave disks, so there was no need to distinguish between them physically.

By integrating the controller on the disk itself and by providing a parallel (8- or 16-bit) interface to the host, it was possible to move data faster even with a lower clock speed. This reduction in clock speed gave disk manufacturers a lot of freedom in designing more reliable disks. Because adaptor hardware was minimal, requirements for device drivers were simpler than for RLL/MFM, ESDI, or SCSI drives.

EIDE-2.eps – Integrating adaptor, controller, and driver circuitry compresses the path data takes between the storage drive and memory

IDE disk drive capacities have been limited to 504 megabytes because older BIOS and ATA-1 specifications made different assumptions about disk drive geometries. The effective capacity was dictated by the lower limits of the two, as shown in the following table:

	BIOS	IDE	Limitation
Maximum cylinders	1024	65536	1024
Maximum heads	255	16	16
Maximum sector/track	63	255	63
Maximum capacity	7.84 GB	127.5 GB	504 MB

The need for larger hard disk capacities forced a workaround: more recent BIOS “fake” the statistics. For example, if a disk is actually 2048 cylinders, 16 heads, and 63 sectors per track, the BIOS may translate this to 1024 cylinders, 32 heads, and 63 sectors per track.

Unfortunately, NEXTSTEP 3.2 IDE device drivers got statistics from the BIOS and then used those numbers to query the disk drive directly, with the result that the “upper” 1024 cylinders were out of view, which generated system panics. The fix has been for the user to reconfigure the disk drive to conform to the BIOS and ATA-1 lower limits or, if the disk drive hardware permits, to use jumpers to “split” the hardware into two logically independent units. NEXTSTEP 3.3 IDE drivers query the hard disk circuitry directly, bypassing the BIOS and avoiding the bug.

ATA-1 FEATURES

Disk manufacturers quickly recognized the benefits of ATA disks, but they went on to add features of their own that led to many incompatibilities. A standard (now referred to as ATA-1) was put together by industry in 1989 to formalize the AT Attachment specification to standardize the existing command set and interface timings. Most of the disks that ship today conform to the ATA-1 specification. ST-506 and ESDI disk drives are obsolete.

The CAM (Common Access Method) committee that formalized the ATA specification chose to use the term ATA instead of IDE, because there was no agreement on what IDE means. ATA is a PC AT-only specification (as the name says).

SFF COMMITTEES

A lot of recent innovation in the PC hardware has originated in SFF (Small Forms Factor) committees. Each committee is created ad hoc, composed of engineers from major manufacturers, to address a particular problem associated with an ID number. Because these SFF committees have very low overhead and can keep pace with changing technology, their standards can truly precede hardware development. Some notable ones are connector standards for 2.5, 1.8, and 1.3-inch disk drives (SFF-8004, 8005, 8006), local bus timings for ATA drives (SFF-8011), and a power management interface for ATA disks (SFF-8018). Most of the changes that occurred in transition from ATA-1 to ATA-2 were approved by SFF first. SFF committees are not standards bodies. They are a cooperative, consensus-building process between manufacturers (whose recommendations eventually are accepted by standards bodies).

The ATA interface specification deals with electrical signals and connectors between a hard disk and its IDE controller as well as a required set of registers (a register file), a command set for communication between the host and the disk, and some timing modes. The host doesn't know about low-level details (like formatting, data encoding, and so on) of the device. Rather, the host knows how to transfer data to the device through the device driver, but not how the actual data is represented and stored on the device. This more abstract interface has more room for flexibility, functionality, and growth (of course, it's harder for manufacturers to design as well). All modern disks (ATA, SCSI) are now designed this way.

ATA-1 AT A GLANCE

ATA-1 provides a physical and register-level interface for connecting hard disks directly to the host computer. It permits low-cost, easy-to-configure hardware, relying on the BIOS built into PCs.

When introduced: 1990.

Major sponsors: Standardized by CAM committee.

Timing modes: PIO modes 0, 1, and 2 and multiword DMA mode 0.

Interface: Introduced the register file and the ATA command set: One 16-bit register for data, seven additional control registers.

Signals, cables, connections: Specified a 40-pin connector (16 bits are for data transfers and the rest are for control) and a maximum 18-inch cable.

ATA-2 FEATURES

In their rush to compete, manufacturers added many things to the ATA-1 standard after it was ratified. The ATA-2 specification standardized these features while ensuring backward-compatibility with the old ATA-1 specification.

Faster timing modes (modes 3 and 4) were added to support the emerging local bus controllers. New commands allowed queries to find out the fastest common timing mode among connected

devices and then set all devices to that mode.

Several vendors created their own commands to implement power management. Essentially, there are two states: normal mode and sleep mode. When the device receives a "sleep" command from the host (the device driver), it will spin-down the disk and power-off some hardware. This substantially reduces the energy consumption. When the host needs some data, the device driver then sends a "wake-up" command to the disk and waits for the disk to be ready and available for data transfers. Basically it's a trade-off: the more circuitry is disabled, the less power it takes to operate the device, but the longer it takes to reactivate the circuits after each power-down event to access the disk.

ATA-2 AT A GLANCE

Changes from ATA-1 include:

- Better specified power management
 - Better specified reset mechanism
 - Specification of PIO modes 3 and 4, multiword DMA 1 and 2
 - Better signal quality on the ATA bus
 - Use of IORDY signal for mode 3 and 4 devices
 - Better support of disks bigger than 504 megabytes
 - Improved description of command set and task register file
-

ATAPI FEATURES

ATAPI is a big improvement designed to allow low-cost, nondisk peripherals to be connected to a personal computer in a standard fashion. It addresses the cost-conscious consumer market.

ATAPI standardizes the way to connect devices other than hard disk drives to a PC bus. Currently there are many nonstandard, vendor-specific ways to connect devices to a PC (like a backup tape to floppy controller or parallel printer port). This is not desirable. Not only is it difficult for

consumers to configure these devices, but also hardware manufacturers have to provide software drivers that are plagued with compatibility problems.

ATAPI (AT Attachment Packet Interface) is a protocol that uses the ATA physical interface and register file for interfacing an ATAPI device (like CD-ROM) to the ATA controller. Any ATAPI-compliant device—not just hard disk drives—can be connected to an ATA connector without the need of an extra adaptor card.

An ATAPI device is transparent to the system BIOS and ATA traffic. It ignores the bus until it receives special commands. These attention commands are new ATA commands not recognized by ATA devices, alerting the ATAPI device to forthcoming activity.

ATAPI does not use the same command set as ATA. Instead it uses a slightly simplified SCSI command set with some improvements for communicating between the host and the ATAPI device. It has better support for audio commands and CD-ROM support. But the changes are minor so that a device driver can ignore them and treat SCSI devices and ATAPI devices in the same way. The ATAPI commands are transmitted to the ATAPI device using the ATA physical cable and register file. When the ATAPI command packet stream is done, the device receives a "go-to-sleep" command, freeing the ATA bus for standard ATA traffic.

This eliminates the need for new connectors, cables, and a new command set. The result allows simple and low-cost standard peripherals that are easier for users to configure, because they don't use any new resources (like I/O ports or IRQ).

ATAPI protocol is designed in such a way that the software drivers are simpler to write. For example, NeXT uses the same SCSI disk classes for ATAPI. Only the hardware interface layer of the device driver differs.

Western Digital drafted the ATAPI specification, which is still being discussed in various SFF committees. The ATAPI standard for CD-ROMs is close to being approved.

Compared to SCSI, ATAPI enables a very inexpensive class of devices. Other than hard disk drives, the only ATAPI-compliant devices available in the market today are CD-ROMs, but ATAPI tapes and other peripherals are expected to follow. An ATAPI standard for tapes is

being discussed.

In the future it will be possible to connect the same ATAPI disks to any register ®le-oriented hardware interface, such as the Enhanced Parallel Port (EPP).

ATAPI AT A GLANCE

ATAPI de®nes a transport layer protocol for connecting devices to an ATA interface to provide a more extensible and general-purpose interface to the ATA register ®le.

When introduced: February 1994 (®nal draft).

Major sponsors: Western Digital and the SFF 8020 committee.

Commands: Few additions to the ATA command set, the most important of which is the ATAPI packet command, which signals a forthcoming SCSI command delivered to the device.

Timing modes: Same as ATA-1.

Power management: Same as ATA-1 but mandatory.

Interface: Same register layout as ATA-1 but with new de®nitions.

Signals, cables, connections: Same as ATA-1.

EIDE AND FAST ATA

EIDE (Enhanced IDE) is a marketing movement spearheaded by Western Digital that combines elements from ATA-2 and ATAPI speci®cations but requires certain changes in BIOS and hardware.

Western Digital's *Enhanced IDE Implementation Guide* seems to cover all of the ATAPI standard, incorporates many features of ATA-2 (notably missing some timing modes and omitting power management), permits use of large hard disks, speci®es new BIOS enhancements, and recommends two ATA controllers. Unfortunately, EIDE is not speci®c in some areas, which leaves room for other vendors' products to deviate, with possible incompatibilities. Furthermore, the user has to ensure

that the operating system, system BIOS, and hardware are all compatible. Despite creating confusion in the marketplace, this program has been successful.

Not to be outdone, Seagate and Quantum countered Western Digital's Enhanced IDE by promoting Fast ATA (for ATA-1) and Fast ATA-2 (for ATA-2) programs. Addressing only disk drive details (mainly timing), these are more modest in their objective than EIDE, which is more of a PC implementation guide. Because they're simple, they're easier to comply with than EIDE.

EIDE AT A GLANCE

EIDE was designed to remove the limitations and disadvantages that have relegated IDE to a PC-only interface and thereby allow it to be a successful choice in the workstation and commercial computer markets.

Major sponsors: Western Digital and later a few other disk vendors.

Features: (a) Support of high-capacity disk drives, (b) support of high-speed host transfers, (c) support of multiple IDE peripherals, (d) support of nondisk IDE peripherals, (e) provision of cost-effective high-performance solutions, (f) maintenance of industry compatibility, and (g) maintenance of AT-IDE's "ease of use" tradition, according to the Western Digital *Enhanced IDE Implementation Guide*, Revision 5.0, November 10, 1993.

FAST ATA AT A GLANCE

Fast ATA is a simple scheme for labeling fast disks. It is a true subset of an existing industry standard. Since timing modes 2, 3, and 4 are optional for ATA drivers, Fast ATA conformance means that such drives support faster timings. Like many other good things in the computer industry, this hasn't caught on in the marketplace.

When introduced: 1994.

Major sponsors: Seagate/Quantum.

Features: Fast ATA drives must support PIO mode 3 and DMA multiword mode 1. They must also support Logical Block Addressing (LBA) and Read/Write Multiple commands. Fast ATA-2 drives will also support PIO mode 4 and DMA multiword mode 2.

BIOS

ATA support in a system involves the disk drive itself, adaptor and/or controller hardware, the system BIOS, and possibly a controller BIOS.

You need an enhanced BIOS to support enhanced hard disks. There's no BIOS standard for getting information about new disks—there are many competing standards and confusion in the marketplace. Many ATA incompatibilities and restrictions result from BIOS and PC hardware nonstandardization.

The Phoenix Enhanced Disk Drive Specification (version 1.0, January 25, 1994) seems to be taking hold as a standard.

RECENT INDUSTRY DEVELOPMENTS

The industry aim is support for the ATA-2-compliant disks and ATAPI devices without the need for any new device drivers. The hardware and BIOS vendors (Phoenix, Award, AMI, and others) would like BIOSs to be updated so that they transparently support the new devices without the need for any extra drivers or configurations.

Microsoft will be supplying device drivers for ATA and ATAPI devices with their multitasking operating systems. IBM will be doing the same with their OS/2 offering. It appears that PC vendors that are not Intel-based (like PowerPC) will be supporting ATA/ATAPI in their systems. Western Digital recently has been trying to market IDE to the workstation and traditionally non-PC computer markets.

Intel has put forward a specification for a PCI bus-mastering ATA specification (SFF 8038). This will allow compliant implementations to transfer data with little CPU intervention. Most ATA disks available today support DMA transfers.

A new proposed ATA-3 specification includes an optional security mode, some power mode enhancements, DMA-only drives, dynamic power selection for memory cards, and several hardware improvements. NeXT plans to support ATA-3 drivers when they become available (probably late 1996).

It's impossible to make the best use of a fast IDE disk unless it is put on a fast local bus. Unfortunately, there's no hardware standard for local bus IDE support (VL-bus and PCI). The IDE interface won't reach its full potential unless a PCI bus-mastering DMA becomes the norm instead of programmed I/O.

Currently all the host data transfers from ATA and ATAPI devices to the system use programmed I/O (this is not a specification limitation, but an implementation issue), which is slow. Moreover, neither the ATA nor the ATAPI standard allows SCSI-2 style connect-disconnect. This means that for good performance ATA disks and ATAPI devices shouldn't share the same cable. (SCSI connect-disconnect provides a feature by which a shared bus can be used more efficiently by connected devices. If a request is made to the device for data and the device knows that this request will take a while, it can relinquish use of the bus and regain it when the data is available. Meanwhile other devices can use the bus.)

PROGRAMMED I/O AND BUS-MASTERING DMA

The CPU, memory, and I/O devices are connected by the same bus. Normally, the CPU is the master of the bus—that is, it initiates data transfers between memory and itself.

Programmed I/O refers to interrupt-driven data transfer under CPU control. The CPU asks the device to get some data. When the data is available, the CPU is interrupted by the device and data is moved (in chunks of bytes) from the I/O device to the CPU and then from the CPU to memory. This is slower but has the advantage of simplicity (hardware cost).

Bus mastering is a well-defined protocol by which an I/O device can request the CPU to relinquish control of the bus and then transfer data between memory and itself (that is, the I/O device). Because there is no intermediary, data transfers can be very fast.

WHAT NEXT IS DOING ABOUT IT

NEXTSTEP 3.3 has improved support for large-capacity disks (disks greater than 528 megabytes). In addition to the standard IDE driver, NEXTSTEP 3.3 includes a revised IDE driver that resolves the geometry problem. This updated driver asks the disk for its geometry instead of obtaining it from the BIOS. If you are currently using the NEXTSTEP 3.2 IDE driver and will be upgrading to

NEXTSTEP 3.3 (or later), you may use the revised IDE driver, but you will have to both reset the BIOS settings and reinitialize the drive (reinstall all operating systems). However, you may continue to use the standard IDE driver with no difference in operation.

The 3.31 version of EIDE/ATAPI drivers supports both ATA-1 and ATA-2 disks as well as ATAPI CD-ROMs. The Enhanced IDE driver is designed to support both enhanced and plain old IDE disks. For compatibility reasons, NeXT still supplies a driver for nonenhanced disks.

NeXT is not likely to support VL-bus IDE controllersÐthere are too many different implementations for NeXT's driver to cover. However, NeXT hopes to support several PCI IDE implementations in the future.

NeXT has no test suite for qualifying a CD-ROM as ATAPI-compliant. Until CD-ROM manufacturers become compatible, users will have the usual Ðrst-generation hardware problems.

NeXT plans to include EIDE/ATAPI compatibility tests in its certiÐcation suite. Most compatibility problems seem to occur in ATAPI CD-ROM drive support. NeXT will keep a list of supported CD-ROM drives.

Phoenix Technologies' Enhanced Disk Drive SpeciÐcation should solve all BIOS-related problems with large ATA disks.

IMPORTANT NEXTANSWERS

- Refer to Nextanswer 1650 (large IDE drive problems with NEXTSTEP 3.2).
 - Refer to Nextanswer 1933 for details on our EIDE/ATAPI device driver.
 - See Nextanswer 1921 (copy images overview) if you want to install NEXTSTEP 3.3 using an ATAPI CD-ROM.
 - Archive sites: <ftp://ftp.symbios.com/pub/standards/io>
-

*Rakesh Dubey is a software engineer at NeXT. You can reach him by e-mail at **Rakesh_Dubey@next.com**.*

Table of contents

<http://www.next.com/HotNews/Journal/OSJ/SummerContents95.html>