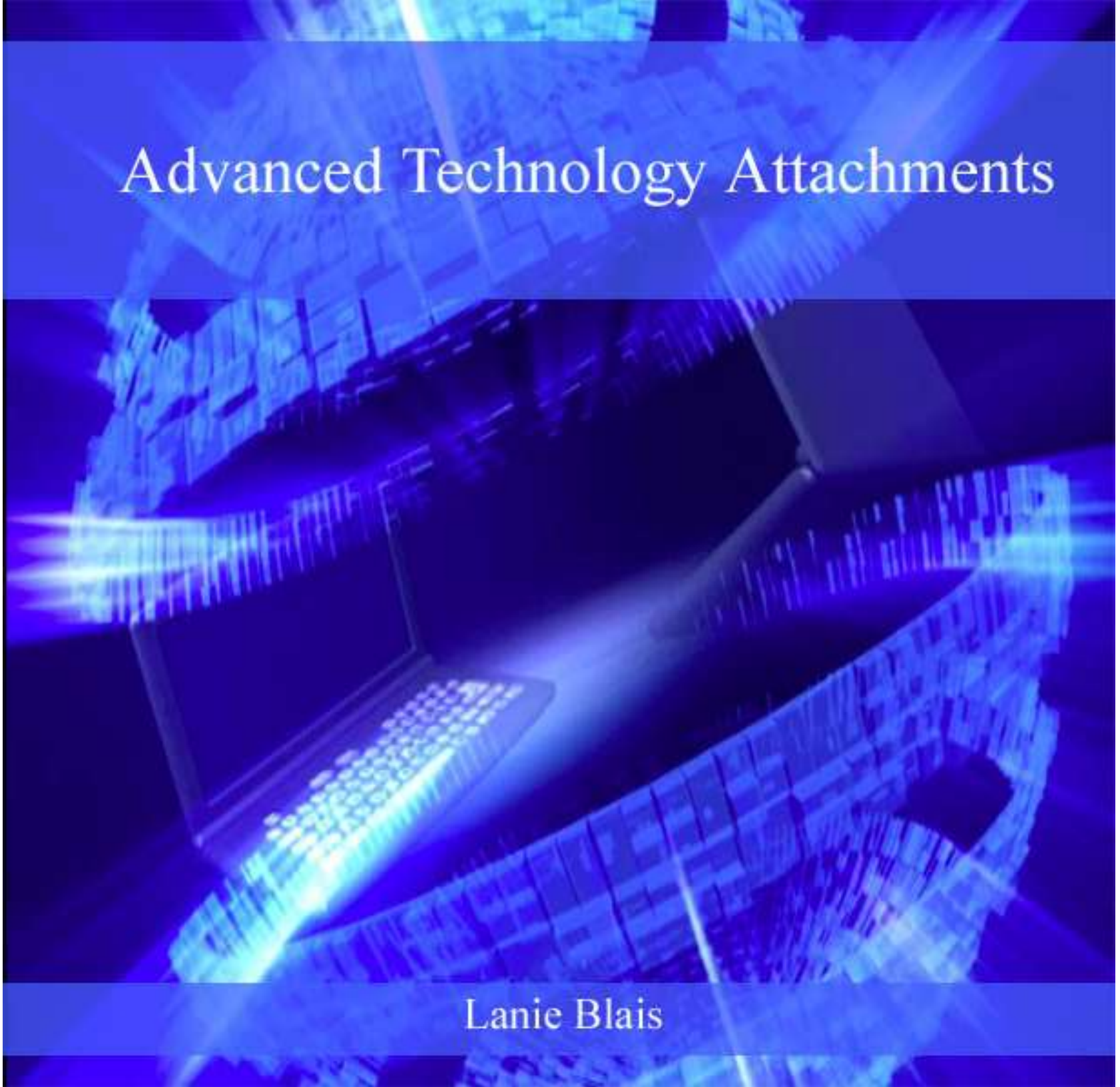
Advanced Technology Attachments



Lanie Blais

First Edition, 2012

ISBN 978-81-323-3022-6

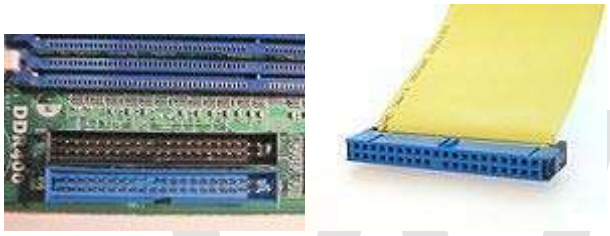
Published by:
Research World
4735/22 Prakashdeep Bldg,
Ansari Road, Darya Ganj,
Delhi - 110002
Email: info@wtbooks.com

Table of Contents

Chapter 1

Parallel ATA

Parallel ATA



ATA connector on the right, with two motherboard ATA sockets on the left.

Type	Internal storage device connector
Production history	
Designer	Western Digital, subsequently amended by many others
Designed	1986
Superseded by	Serial ATA (2003)
General specifications	
Hot pluggable	No
External	No
Cable	40 or 80 wires ribbon cable
Pins	40
Data	
Width	16 bits
Bandwidth	16 MB/s originally

	later 33, 66, 100 and 133 MB/s
Max. devices	2 (master/slave)
Protocol	Parallel

Pin out

Pin 1	Reset
Pin 2	Ground
Pin 3	Data 7
Pin 4	Data 8
Pin 5	Data 6
Pin 6	Data 9
Pin 7	Data 5
Pin 8	Data 10
Pin 9	Data 4
Pin 10	Data 11
Pin 11	Data 3
Pin 12	Data 12
Pin 13	Data 2
Pin 14	Data 13
Pin 15	Data 1
Pin 16	Data 14
Pin 17	Data 0
Pin 18	Data 15
Pin 19	Ground
Pin 20	Key or VCC_in
Pin 21	DDRQ
Pin 22	Ground

Pin 23	I/O write
Pin 24	Ground
Pin 25	I/O read
Pin 26	Ground
Pin 27	IOCHRDY
Pin 28	Cable select
Pin 29	DDACK
Pin 30	Ground
Pin 31	IRQ
Pin 32	No connect
Pin 33	Addr 1
Pin 34	GPIO_DMA66_Detect
Pin 35	Addr 0
Pin 36	Addr 2
Pin 37	Chip select 1P
Pin 38	Chip select 3P
Pin 39	Activity
Pin 40	Ground

Parallel ATA (**PATA**), originally ATA, is an interface standard for the connection of storage devices such as hard disks, solid-state drives, floppy drives, and optical disc drives in computers. The standard is maintained by X3/INCITS committee. It uses the underlying **AT Attachment** (**ATA**) and **AT Attachment Packet Interface** (**ATAPI**) standards.

The Parallel ATA standard is the result of a long history of incremental technical development, which began with the original AT Attachment interface, developed for use in early PC AT equipment. The ATA interface itself evolved in several stages from Western Digital's original **Integrated Drive Electronics** (**IDE**) interface. As a result, many near-synonyms for ATA/ATAPI and its previous incarnations are still in common informal use. After the introduction of Serial ATA in 2003, the original ATA was retroactively renamed *Parallel ATA*.

Parallel ATA cables have a maximum allowable length of only 18 in (457 mm). Because of this length limit the technology normally appears as an internal computer storage interface. For many years ATA provided the most common and the least expensive interface for this application. It has largely been replaced by Serial ATA (SATA) in newer systems.

History and terminology

The standard was originally conceived as "PC/AT Attachment" as its primary feature was a direct connection to the 16-bit ISA bus introduced with the IBM PC/AT. The name was shortened to "AT Attachment" to avoid possible trademark issues. It is not spelled out as "Advanced Technology" anywhere in current or recent versions of the specification; it is simply "AT Attachment".

IDE and ATA-1

The first version of what is now called the ATA/ATAPI interface was developed by Western Digital under the name *Integrated Drive Electronics* (IDE). Together with Control Data Corporation (who manufactured the hard drive part) and Compaq Computer (into whose systems these drives would initially go), they developed the connector, the signaling protocols, and so on with the goal of remaining software compatible with the existing ST-506 hard drive interface. The first such drives appeared in Compaq PCs in 1986.

The term *Integrated Drive Electronics* refers not just to the connector and interface definition, but also to the fact that the drive controller is integrated into the drive, as opposed to a separate controller on or connected to the motherboard. The interface cards used to connect a parallel ATA drive to, for example, a PCI slot are not drive controllers, they are merely bridges between the host bus and the ATA interface. Since the original ATA interface is essentially just a 16-bit ISA slot in disguise, the bridge was especially simple in case of an ATA connector being located on an ISA interface card. The integrated controller presented the drive to the host computer as an array of 512-byte blocks with a relatively simple command interface. This relieved the mainboard and interface cards in the host computer of the chores of stepping the disk head arm, moving the head arm in and out, and so on, as had to be done with earlier ST-506 and ESDI hard drives. All of these low-level details of the mechanical operation of the drive were now handled by the controller on the drive itself. This also eliminated the need to design a single controller that could handle many different types of drives, since the controller could be unique for the drive. The host need only ask for a particular sector, or block, to be read or written, and either accept the data from the drive or send the data to it.

The interface used by these drives was standardized in 1994 as ANSI standard X3.221-1994, *AT Attachment Interface for Disk Drives*. After later versions of the standard were developed, this became known as "ATA-1".

A short-lived, seldom-used implementation of ATA was created for the IBM XT and similar machines that used the 8-bit version of the ISA bus. It has been referred to as "XTA" or "XT Attachment."

Second ATA interface

When PC motherboard makers started to include onboard ATA interfaces in place of the earlier ISA plug-in cards, there was usually only one ATA connector on the board, which could support up to two hard drives. At the time in combination with the floppy drive, this was sufficient for most people, and eventually it became common to have two hard drives installed. When the CD-ROM was developed, many computers would have been unable to accept these drives if they had been ATA devices, due to already having two hard drives installed. Adding the CD-ROM drive would have required removal of one of the drives.

SCSI was available as a CD-ROM expansion option at the time, but devices with SCSI were more expensive than ATA devices due to the need for a smart interface that is capable of bus arbitration. SCSI typically added US$ 100-300 to the cost of a storage device, in addition to the cost of a SCSI host adapter.

The less-expensive solution was the addition of a dedicated CD-ROM interface, typically included as an expansion option on a sound card. It was included on the sound card because early business PCs did not include support for more than simple beeps from the internal speaker, and tuneful sound playback was considered unnecessary for early business software. When the CD-ROM was introduced, it was logical to also add digital audio to the computer at the same time (for the same reason, sound cards tended to include a gameport interface for joysticks). An older business PC could be upgraded in this manner to meet the Multimedia PC standard for early software packages that used sound (which required the sound card) and colorful video animation (which required the CD-ROM as floppy disks simply did not have the necessary data capacity).

The second drive interface initially was not well-defined. It was first introduced with interfaces specific to certain CD-ROM drives such as Mitsumi, Sony or Panasonic drives, and it was common to find early sound cards with two or three separate connectors each designed to match a certain brand of CD-ROM drive. This evolved into the standard ATA interface for ease of cross-compatibility, though the sound card ATA interface still usually supported only a single CD-ROM and not hard drives.

This second ATA interface on the sound card eventually evolved into the second motherboard ATA interface which was long included as a standard component in all PCs. Called the "primary" and "secondary" ATA interfaces, they were assigned to base addresses 0x1F0 and 0x170 on ISA bus systems.

EIDE and ATA-2

In 1994, about the same time that the ATA-1 standard was adopted, Western Digital introduced drives under a newer name, **Enhanced IDE** (EIDE). These included most of the features of the forthcoming ATA-2 specification and several additional enhancements. Other manufacturers introduced their own variations of ATA-1 such as "Fast ATA" and "Fast ATA-2".

The new version of the ANSI standard, *AT Attachment Interface with Extensions ATA-2* (X3.279-1996), was approved in 1996. It included most of the features of the manufacturer-specific variants.

ATA-2 also was the first to note that devices other than hard drives could be attached to the interface:

3.1.7 Device: Device is a storage peripheral. Traditionally, a device on the ATA interface has been a hard disk drive, but any form of storage device may be placed on the ATA interface provided it adheres to this standard.

ATAPI

As mentioned in the previous sections, ATA was originally designed for, and worked only with hard disks and devices that could emulate them. The introduction of ATAPI (ATA Packet Interface) by a group called the Small Form Factor committee allowed ATA to be used for a variety of other devices that require functions beyond those necessary for hard disks. For example, any removable media device needs a "media eject" command, and a way for the host to determine whether the media is present, and these were not provided in the ATA protocol.

The Small Form Factor committee approached this problem by defining ATAPI, the "ATA Packet Interface". ATAPI is actually a protocol allowing the ATA interface to carry SCSI commands and responses; therefore all ATAPI devices are actually "speaking SCSI" other than at the electrical interface. In fact, some early ATAPI devices were simply SCSI devices with an ATA/ATAPI to SCSI protocol converter added on. The SCSI commands and responses are embedded in "packets" (hence "ATA Packet Interface") for transmission on the ATA cable. This allows any device class for which a SCSI command set has been defined to be interfaced via ATA/ATAPI.

ATAPI devices are also "speaking ATA", as the ATA physical interface and protocol are still being used to send the packets. On the other hand, ATA hard drives and solid state drives do not use ATAPI.

ATAPI devices include CD-ROM and DVD-ROM drives, tape drives, and large-capacity floppy drives such as the Zip drive and SuperDisk drive.

The SCSI commands and responses used by each class of ATAPI device (CD-ROM, tape, etc.) are described in other documents or specifications specific to those device classes and are not within ATA/ATAPI or the T13 committee's purview.

ATAPI was adopted as part of ATA in INCITS 317-1998, *AT Attachment with Packet Interface Extension (ATA/ATAPI-4)*.

UDMA and ATA-4

The ATA/ATAPI-4 also introduced several "Ultra DMA" transfer modes. These initially supported speeds from 16 MByte/s to 33 MByte/second. In later versions faster Ultra DMA modes were added, requiring a new 80-wire cable to reduce crosstalk. The latest versions of Parallel ATA support up to 133 MByte/s.

Current terminology

The terms "integrated drive electronics" (IDE), "enhanced IDE" and "EIDE" have come to be used interchangeably with ATA (now Parallel ATA, or PATA).

In addition there have been several generations of "EIDE" drives marketed, compliant with various versions of the ATA specification. An early "EIDE" drive might be compatible with ATA-2, while a later one with ATA-6.

Nevertheless a request for an "IDE" or "EIDE" drive from a computer parts vendor will almost always yield a drive that will work with most Parallel ATA interfaces.

Another common usage is to refer to the specification version by the fastest mode supported. For example, ATA-4 supported Ultra DMA modes 0 through 2, the latter providing a maximum transfer rate of 33 megabytes per second. ATA-4 drives are thus sometimes called "UDMA-33" drives, and sometimes "ATA-33" drives. Similarly, ATA-6 introduced a maximum transfer speed of 100 megabytes per second, and some drives complying to this version of the standard are marketed as "PATA/100" drives.

Drive size limitations

The original ATA specification used a 28-bit addressing mode, allowing for the addressing of 2^{28} (268,435,456) sectors (blocks) of 512 bytes each, resulting in a maximum capacity of 128 GiB (137 GB). The BIOS in early PCs imposed smaller limits such as 8.46 GB, with a maximum of 1024 cylinders, 256 heads and 63 sectors, but this was not a limit imposed by the ATA interface.

ATA-6 introduced 48-bit addressing, increasing the limit to 128 PiB (144 PB). As a consequence, any ATA drive of capacity larger than about 137 gigabytes must be an ATA-6 or later drive. Connecting such a drive to a host with an ATA-5 or earlier interface will limit the usable capacity to the maximum of the interface.

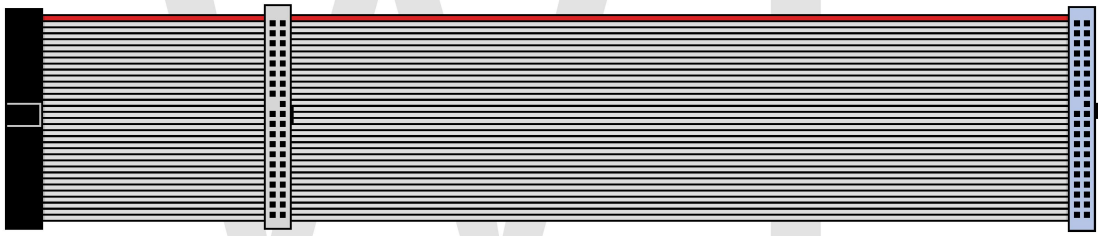
Some operating systems, including Windows XP pre-SP 1, and Windows 2000, disable 48-bit LBA by default, requiring the user to take extra steps to use the entire capacity of an ATA drive larger than about 137 gigabytes. Older operating systems, such as Windows 98, do not support 48-bit LBA at all.

Obsolescence

For a long period of time, ATA ruled as the primary storage device interface and in some systems a third and fourth motherboard interface was provided (for example, Promise Ultra-100), for up to eight ATA devices attached to the motherboard.

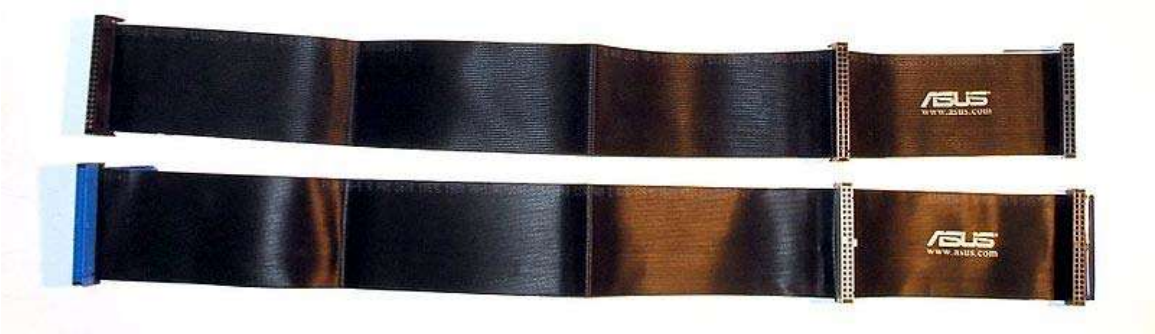
After the introduction of SATA (Serial ATA), use of Parallel ATA declined, and new motherboards had only a single PATA connector, for up to two PATA optical drives, along with (typically) six or more SATA connectors for hard drives and other devices. In new computers, the parallel ATA interface is rarely used, and several PC chipsets have removed support for PATA, and motherboard vendors still wishing to offer ATA with those chipsets must include an additional interface chip.

Parallel ATA interface



Parallel ATA cables transfer data 16 bits at a time. The traditional cable uses 40-pin connectors attached to a ribbon cable. Each cable has two or three connectors, one of which plugs into an adapter interfacing with the rest of the computer system. The remaining connector(s) plug into drives.

ATA's cables have had 40 wires for most of its history (44 conductors for the smaller form-factor version used for 2.5" drives — the extra four for power), but an 80-wire version appeared with the introduction of the *Ultra DMA/33* (*UDMA*) mode. All of the additional wires in the new cable are ground wires, interleaved with the previously defined wires to reduce the effects of capacitive coupling between neighboring signal wires, reducing crosstalk. Capacitive coupling is more of a problem at higher transfer rates, and this change was necessary to enable the 66 megabytes per second (MB/s) transfer rate of *UDMA4* to work reliably. The faster *UDMA5* and *UDMA6* modes also require 80-conductor cables.



ATA cables:
40 wire ribbon cable (top)
80 wire ribbon cable (bottom)

Though the number of wires doubled, the number of connector pins and the pinout remain the same as 40-conductor cables, and the external appearance of the connectors is identical. Internally the connectors are different; the connectors for the 80-wire cable connect a larger number of ground wires to a smaller number of ground pins, while the connectors for the 40-wire cable connect ground wires to ground pins one-for-one. 80-wire cables usually come with three differently colored connectors (blue, black, and gray for controller, master drive, and slave drive respectively) as opposed to uniformly colored 40-wire cable's connectors (commonly all gray). The gray connector on 80-conductor cables has pin 28 CSEL not connected, making it the slave position for drives configured cable select.

Round parallel ATA cables (as opposed to ribbon cables) were eventually made available as they were believed to have less effect on computer cooling and were easier to handle; however, only ribbon cables are supported by the ATA specifications.

Pin 20

In the ATA standard pin 20 is defined as (mechanical) key and is not used. This socket on the female connector is often obstructed, requiring pin 20 to be omitted from the male cable or drive connector, making it impossible to plug it in the wrong way round; a male connector with pin 20 present cannot be used. However, some flash memory drives can use pin 20 as VCC_in to power the drive without requiring a special power cable; this feature can only be used if the equipment supports this use of pin 20.

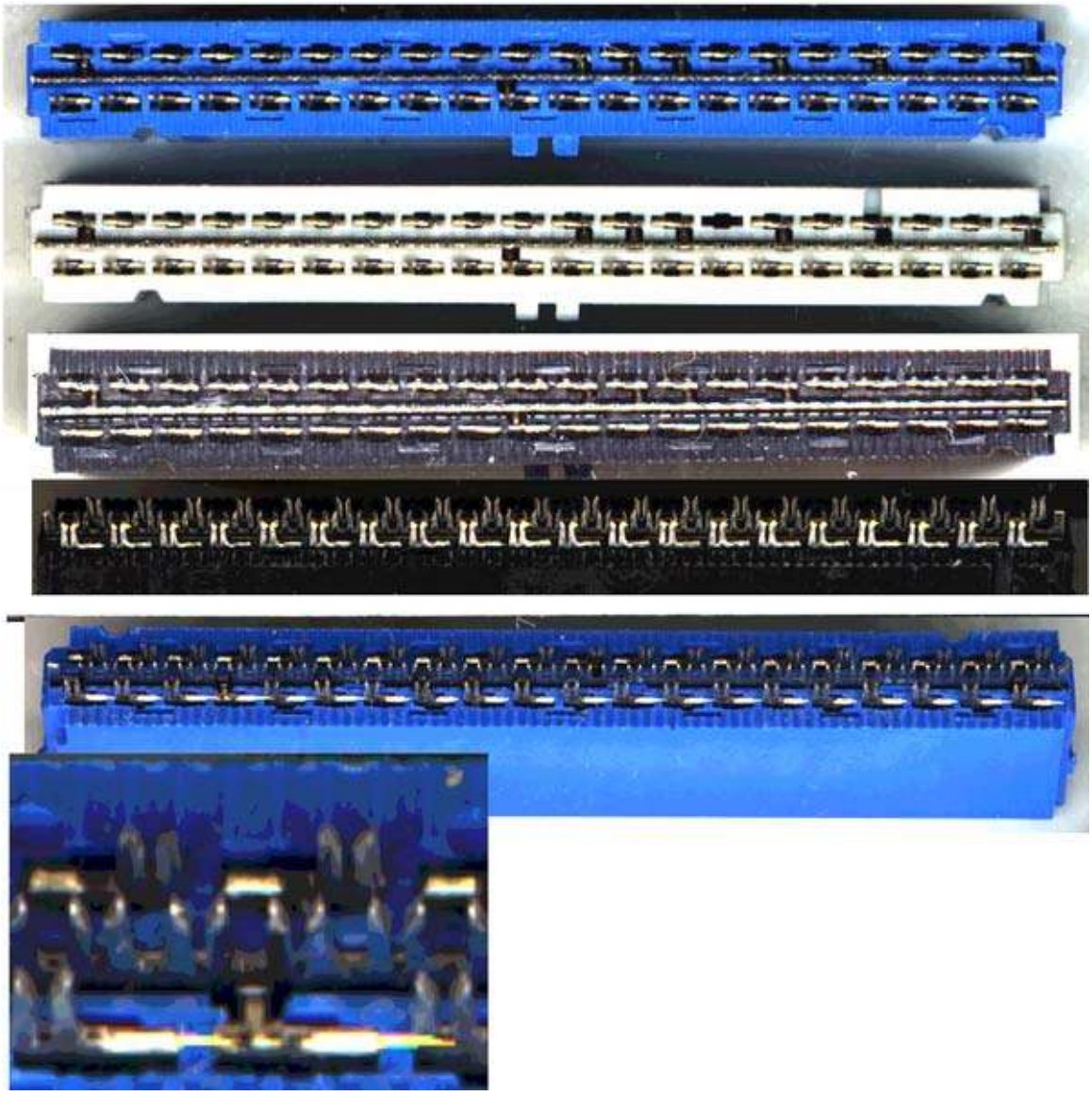
Pin 28

Pin 28 of the gray (slave/middle) connector of an 80 conductor cable is not attached to any conductor of the cable. It is attached normally on the black (master drive end) and blue (motherboard end) connectors.

Pin 34

Pin 34 is connected to ground inside the blue connector of an 80 conductor cable but not attached to any conductor of the cable. It is attached normally on the gray and black connectors.

Differences between connectors on 80-conductor cables



The image shows PATA connectors after removal of strain relief, cover, and cable. Pin one is at bottom left of the connectors, pin 2 is top left, etc., except that the lower image of the blue connector shows the view from the opposite side, and pin one is at top right.

Each contact comprises a pair of points which together pierce the insulation of the ribbon cable with such precision that they make a connection to the desired conductor without harming the insulation on the neighboring wires. The center row of contacts are all

connected to the common ground bus and attached to the odd numbered conductors of the cable. The top row of contacts are the even-numbered sockets of the connector (mating with the even-numbered pins of the receptacle) and attach to every other even-numbered conductor of the cable. The bottom row of contacts are the odd-numbered sockets of the connector (mating with the odd-numbered pins of the receptacle) and attach to the remaining even-numbered conductors of the cable.

Note the connections to the common ground bus from sockets 2 (top left), 19 (center bottom row), 22, 24, 26, 30, and 40 on all connectors. Also note (enlarged detail, bottom, looking from the opposite side of the connector) that socket 34 of the blue connector does not contact any conductor but unlike socket 34 of the other two connectors, it does connect to the common ground bus. On the gray connector, note that socket 28 is completely missing, so that pin 28 of the drive attached to the gray connector will be open. On the black connector, sockets 28 and 34 are completely normal, so that pins 28 and 34 of the drive attached to the black connector will be connected to the cable. Pin 28 of the black drive reaches pin 28 of the host receptacle but not pin 28 of the gray drive, while pin 34 of the black drive reaches pin 34 of the gray drive but not pin 34 of the host. Instead, pin 34 of the host is grounded.

The standard dictates color-coded connectors for easy identification by both installer and cable maker. All three connectors are different from one another. The blue (host) connector has the socket for pin 34 connected to ground inside the connector but not attached to any conductor of the cable. Since the old 40 conductor cables do not ground pin 34, the presence of a ground connection indicates that an 80 conductor cable is installed. The wire for pin 34 is attached normally on the other types and is not grounded. Installing the cable backwards (with the black connector on the system board, the blue connector on the remote device and the gray connector on the center device) will ground pin 34 of the remote device and connect host pin 34 through to pin 34 of the center device. The gray center connector omits the connection to pin 28 but connects pin 34 normally, while the black end connector connects both pins 28 and 34 normally.

Multiple devices on a cable

If two devices attach to a single cable, one must be designated as *device 0* (commonly referred to as *master*) and the other as *device 1* (*slave*). This distinction is necessary to allow both drives to share the cable without conflict. The *master* drive is the drive that usually appears "first" to the computer's BIOS and/or operating system. On old BIOSes (Intel 486 era and older), the drives are often referred to by the BIOS as "C" for the master and "D" for the slave following the way DOS would refer to the active primary partitions on each.

The mode that a drive must use is often set by a jumper setting on the drive itself, which must be manually set to *master* or *slave*. If there is a single device on a cable, it should be configured as *master*. However, some hard drives have a special setting called *single* for this configuration (Western Digital, in particular). Also, depending on the hardware and software available, a single drive on a cable can work reliably even though configured as

the *slave* drive (this configuration is most often seen when a CD ROM has a channel to itself).

Cable select

A drive mode called *cable select* was described as optional in ATA-1 and has come into fairly widespread use with ATA-5 and later. A drive set to "cable select" automatically configures itself as master or slave, according to its position on the cable. Cable select is controlled by pin 28. The host adapter grounds this pin; if a device sees that the pin is grounded, it becomes the master device; if it sees that pin 28 is open, the device becomes the slave device.

This setting is usually chosen by a jumper setting on the drive called "cable select", usually marked *CS*, which is separate from the "master" or "slave" setting.

Note that if two drives are configured as *master* and *slave* manually, this configuration does not need to correspond to their position on the cable. Pin 28 is only used to let the drives know their position on the cable; it is not used by the host when communicating with the drives.

With the 40-wire cable it was very common to implement cable select by simply cutting the pin 28 wire between the two device connectors; putting the slave device at the end of the cable, and the master on the middle connector. This arrangement eventually was standardized in later versions. If there is just one device on the cable, this results in an unused stub of cable, which is undesirable for physical convenience and electrical reasons. The stub causes signal reflections, particularly at higher transfer rates.

Starting with the 80-wire cable defined for use in ATAPI5/UDMA4, the master device goes at the end of the 18-inch (460 mm) cable—the black connector—and the slave device goes on the middle connector—the gray one—and the blue connector goes onto the motherboard. So, if there is only one (master) device on the cable, there is no cable stub to cause reflections. Also, cable select is now implemented in the slave device connector, usually simply by omitting the contact from the connector body.

Master and slave clarification

Although they are in extremely common use, the terms "master" and "slave" do not actually appear in current versions of the ATA specifications. The two devices are simply referred to as "device 0" and "device 1", respectively, in ATA-2 and later.

It is a common myth that the controller on the master drive assumes control over the slave drive, or that the master drive may claim priority of communication over the other device on the channel. In fact, the drivers in the host operating system perform the necessary arbitration and serialization, and each drive's onboard controller operates independently of the other.

The terms "master" and "slave" have not been without controversy. In 2003, the County of Los Angeles, California, US requested that, when possible, suppliers stop using the terms because the county found them unacceptable in light of its "cultural diversity and sensitivity".

Serialized, overlapped, and queued operations

The parallel ATA protocols up through ATA-3 require that once a command has been given on an ATA interface, it must complete before any subsequent command may be given. Operations on the devices must be serialized—with only one operation in progress at a time—with respect to the ATA host interface. A useful mental model is that the host ATA interface is busy with the first request for its entire duration, and therefore can not be told about another request until the first one is complete. The function of serializing requests to the interface is usually performed by a device driver in the host operating system.

The ATA-4 and subsequent versions of the specification have included an "overlapped feature set" and a "queued feature set" as optional features, both being given the name "Tagged Command Queuing", a reference to a set of features from SCSI which the ATA version attempts to emulate. However, support for these is extremely rare in actual parallel ATA products and device drivers because these feature sets were implemented in such a way as to maintain software compatibility with its heritage as originally an extension of the ISA bus. This implementation resulted in excessive CPU utilization which largely negated the advantages of command queuing. By contrast, overlapped and queued operations have been common in other storage buses, in particular, SCSI's version of tagged command queuing had no need to be software compatible with ISA's APIs, allowing it to attain high performance with low overhead on buses which supported first party DMA like PCI. This has long been seen as a major advantage of SCSI.

The Serial ATA standard has supported native command queueing since its first release, but it is an optional feature for both host-adapters and target-devices. Many less expensive PC motherboards do not support NCQ. Many SATA/II hard drives sold today support NCQ, while no removable (CD/DVD) drives do because the ATAPI command set used to control them prohibits queued operations.

Two devices on one cable — speed impact

There are many debates about how much a slow device can impact the performance of a faster device on the same cable. There is an effect, but the debate is confused by the blurring of two quite different causes, called here "Lowest speed" and "One operation at a time".

"Lowest speed"

It is a common misconception that, if two devices of different speed capabilities are on the same cable, both devices' data transfers will be constrained to the speed of the slower device.

For all modern ATA host adapters this is not true, as modern ATA host adapters support *independent device timing*. This allows each device on the cable to transfer data at its own best speed. Even with older adapters without independent timing, this effect only applies to the data transfer phase of a read or write operation. This is usually the shortest part of a complete read or write operation.

"One operation at a time"

This is caused by the omission of both overlapped and queued feature sets from most parallel ATA products. Only one device on a cable can perform a read or write operation at one time, therefore a fast device on the same cable as a slow device **under heavy use** will find it has to wait for the slow device to complete its task first.

However, most modern devices will report write operations as complete once the data is stored in its onboard cache memory, before the data is written to the (slow) magnetic storage. This allows commands to be sent to the other device on the cable, reducing the impact of the "one operation at a time" limit.

The impact of this on a system's performance depends on the application. For example, when copying data from an optical drive to a hard drive (such as during software installation), this effect probably doesn't matter: Such jobs are necessarily limited by the speed of the optical drive no matter where it is. But if the hard drive in question is also expected to provide good throughput for other tasks at the same time, it probably should not be on the same cable as the optical drive.

HDD passwords and security

The disk lock is a built-in security feature in the disk. It is part of the ATA specification, and thus not specific to any brand or device. The disk lock can be enabled and disabled by sending special ATA commands to the drive. If a disk is locked, it will refuse all access until it is unlocked.

A disk always has two passwords: A User password and a Master password. Most disks support a Master Password Revision Code. Reportedly some disks can report if the Master password has been changed, or if it still the factory default. The revision code is word 92 in the IDENTIFY response. Reportedly on some disks a value of 0xFFFE means the Master password is unchanged. The standard does not distinguish this value.

A disk can be locked in two modes: High security mode or Maximum security mode. Bit 8 in word 128 of the IDENTIFY response shows which mode the disk is in: 0 = High, 1 = Maximum.

In High security mode, the disk can be unlocked with either the User or Master password, using the "SECURITY UNLOCK DEVICE" ATA command. There is an attempt limit, normally set to 5, after which the disk must be power cycled or hard-reset before unlocking can be attempted again. Also in High security mode the SECURITY ERASE UNIT command can be used with either the User or Master password.

In Maximum security mode, the disk cannot be unlocked without the User password — the only way to get the disk back to a usable state is to issue the SECURITY ERASE PREPARE command, immediately followed by SECURITY ERASE UNIT. In Maximum security mode the SECURITY ERASE UNIT command requires the User password and will completely erase all data on the disk. The operation is slow, it may take longer than half an hour or more, depending on the size of the disk. (Word 89 in the IDENTIFY response indicates how long the operation will take.)

While the ATA disk lock is intended to be impossible to defeat without a valid password, there are workarounds to unlock a drive. Many data recovery companies offer unlocking services, so while the disk lock will deter a casual attacker, it is not secure against a qualified adversary.

External parallel ATA devices

It is extremely uncommon to find external PATA devices that directly use the interface for connection to a computer. PATA is primarily restricted to devices installed internally, due to the short data cable specification. A device connected externally needs additional cable length to form a U-shaped bend so that the external device may be placed alongside, or on top of the computer case, and the standard cable length is too short to permit this.

For ease of reach from motherboard to device, the connectors tend to be positioned towards the front edge of motherboards, for connection to devices protruding from the front of the computer case. This front-edge position makes extension out the back to an external device even more difficult. Ribbon cables are poorly shielded, and the standard relies upon the cabling to be installed inside a shielded computer case to meet RF emissions limits.

All external PATA devices, such as external hard drives, use some other interface technology to bridge the distance between the external device and the computer. USB is the most common external interface, followed by Firewire. A bridge chip inside the external devices converts from the USB interface to PATA, and typically only supports a single external device without cable select or master/slave.

ATA standards versions, transfer rates, and features

The following table shows the names of the versions of the ATA standards and the transfer modes and rates supported by each. Note that the transfer rate for each mode (for example, 66.7 MB/s for UDMA4, commonly called "Ultra-DMA 66", defined by ATA-5) gives its maximum theoretical transfer rate on the cable. This is simply two bytes multiplied by the effective clock rate, and presumes that every clock cycle is used to transfer end-user data. In practice, of course, protocol overhead reduces this value.

Congestion on the host bus to which the ATA adapter is attached may also limit the maximum burst transfer rate. For example, the maximum data transfer rate for conventional PCI bus is 133 MB/s, and this is shared among all active devices on the bus.

In addition, no ATA hard drives existed in 2005 that were capable of measured sustained transfer rates of above 80 MB/s. Furthermore, sustained transfer rate tests do not give realistic throughput expectations for most workloads: They use I/O loads specifically designed to encounter almost no delays from seek time or rotational latency. Hard drive performance under most workloads is limited first and second by those two factors; the transfer rate on the bus is a distant third in importance. Therefore, transfer speed limits above 66 MB/s really affect performance only when the hard drive can satisfy all I/O requests by reading from its internal cache — a very unusual situation, especially considering that such data are usually already buffered by the operating system.

As of April 2010 mechanical hard disk drives can transfer data at up to 157 MB/s, which is beyond the capabilities of the PATA/133 specification. High-performance flash drives can transfer data at up to 308 MB/s.

Only the Ultra DMA modes use CRC to detect errors in data transfer between the controller and drive. This is a 16 bit CRC, and it is used for data blocks only. Transmission of command and status blocks do not use the fast signaling methods that would necessitate CRC. For comparison, in Serial ATA, 32 bit CRC is used for both commands and data.

Features introduced with each ATA revision

Standard	Other Names	New Transfer Modes	Maximum disk size (512 byte sector)	Other New Features	ANSI Reference
IDE (pre-ATA)	IDE	PIO 0	2 GiB (2.1 GB)	22-bit logical block addressing (LBA)	-
ATA-1	ATA, IDE	PIO 0, 1, 2 Single-word DMA 0, 1, 2 Multi-word DMA 0	128 GiB (137 GB)	28-bit logical block addressing (LBA)	X3.221-1994 (obsolete since 1999)

ATA-2	EIDE, Fast ATA, Fast IDE, Ultra ATA	PIO 3, 4 Multi-word DMA 1, 2		PCMCIA connector	X3.279-1996 (obsolete since 2001)
ATA-3	EIDE	Single-word DMA modes dropped		S.M.A.R.T., Security, 44 pin connector for 2.5" drives	X3.298-1997 (obsolete since 2002)
ATA/ATAPI-4	ATA-4, Ultra ATA/33	Ultra DMA 0, 1, 2 aka UDMA/33		AT Attachment Packet Interface (ATAPI) (support for CD-ROM, tape drives etc.), Optional overlapped and queued command set features, Host Protected Area (HPA), CompactFlash Association (CFA) feature set for solid state drives	NCITS 317-1998
ATA/ATAPI-5	ATA-5, Ultra ATA/66	Ultra DMA 3, 4 aka UDMA/66		80-wire cables; CompactFlash connector	NCITS 340-2000
ATA/ATAPI-6	ATA-6, Ultra ATA/100	UDMA 5 aka UDMA/100	128 PiB (144 PB)	48-bit LBA, Device Configuration Overlay (DCO), Automatic Acoustic Management (AAM)	NCITS 361-2002
ATA/ATAPI-7	ATA-7, Ultra ATA/133	UDMA 6 aka UDMA/133 SATA/150		SATA 1.0, Streaming feature set, long logical/physical sector feature set for non-packet devices	NCITS 397-2005 (vol 1) NCITS 397-2005 (vol 2) NCITS 397-2005 (vol 3)
ATA/ATAPI-8	ATA-8	—		Hybrid drive featuring non-volatile cache to speed up critical OS files	In progress

Speed of defined transfer modes

	Transfer Modes	
Mode	**#**	**Maximum transfer rate**

		(MB/s)
PIO	0	3.3
	1	5.2
	2	8.3
	3	11.1
	4	16.7
Single-word DMA	0	2.1
	1	4.2
	2	8.3
Multi-word DMA	0	4.2
	1	13.3
	2	16.7
	3	20
	4	25
Ultra DMA	0	16.7
	1	25.0
	2 (Ultra ATA/33)	33.3
	3	44.4
	4 (Ultra ATA/66)	66.7
	5 (Ultra ATA/100)	100
	6 (Ultra ATA/133)	133

Related standards, features, and proposals

ATAPI Removable Media Device (ARMD)

ATAPI devices with removable media, other than CD and DVD drives, are classified as ARMD (ATAPI Removable Media Device) and can appear as either a super-floppy (non-partitioned media) or a hard drive (partitioned media) to the operating system. These can be supported as bootable devices by a BIOS complying with the **ATAPI Removable Media Device BIOS Specification**, originally developed by Compaq Computer Corporation and Phoenix Technologies. It specifies provisions in the BIOS of a personal computer to allow the computer to be bootstrapped from devices such as Zip drives, Jaz drives, SuperDisk (LS-120) drives, and similar devices.

These devices have removable media like floppy disk drives, but capacities more commensurate with hard drives, and programming requirements unlike either. Due to limitations in the floppy controller interface most of these devices were ATAPI devices, connected to one of the host computer's ATA interfaces, similarly to a hard drive or CD-ROM device. However, existing BIOS standards did not support these devices. An ARMD-compliant BIOS allows these devices to booted from and used under the operating system without requiring device-specific code in the OS.

A BIOS implementing ARMD allows the user to include ARMD devices in the boot search order. Usually an ARMD device is configured earlier in the boot order than the hard drive. Similarly to a floppy drive, if bootable media is present in the ARMD drive, the BIOS will boot from it; if not, the BIOS will continue in the search order, usually with the hard drive last.

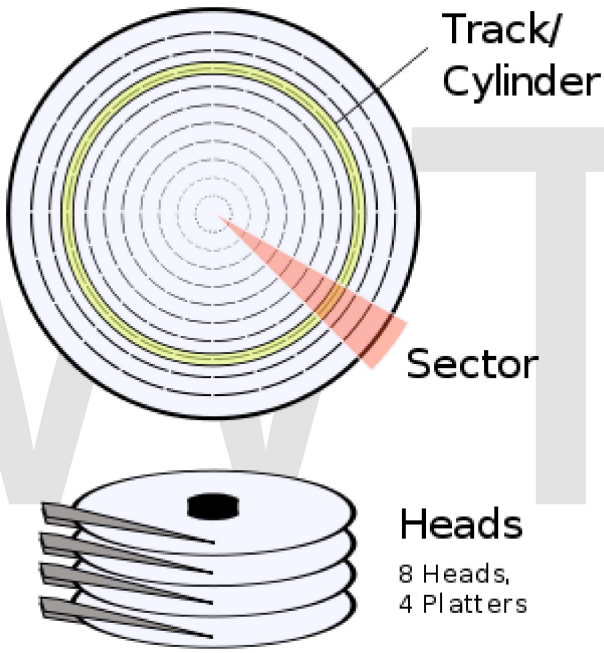
There are two variants of ARMD, ARMD-FDD and ARMD-HDD. Originally ARMD caused the devices to appear as a sort of very large floppy drive, either the primary floppy drive device 00h or the secondary device 01h. Some operating systems required code changes to support floppy disks with capacities far larger than any standard floppy disk drive. Also, standard-floppy disk drive emulation proved to be unsuitable for certain high-capacity floppy disk drives such as Iomega Zip drives. Later the ARMD-HDD, ARMD-"Hard disk device", variant was developed to address these issues. Under ARMD-HDD, an ARMD device appears to the BIOS and the operating system as a hard drive.

ATA over Ethernet

In August 2004, Sam Hopkins and Brantley Coile of Coraid specified a lightweight ATA over Ethernet protocol to carry ATA commands over Ethernet instead of directly connecting them to a PATA host adapter. This permitted the established block protocol to be reused in storage area network (SAN) applications.

Chapter 2

Cylinder-Head-Sector



Cylinder-head-sector, also known as **CHS**, was an early method for giving addresses to each physical block of data on a hard disk drive. In the case of *floppy* drives, for which the same exact *diskette* medium can be truly *low-level formatted* to different capacities, this is still true.

Though CHS values no longer have a direct physical relationship to the data stored on disks, *pseudo* CHS values (which can be *translated* by disk electronics or software) are still being used by many utility programs.

Definitions

Heads

Data is written to and read from the surface of a platter by a device called a head. Naturally, a platter has 2 sides and thus 2 surfaces on which data could be manipulated; usually there are 2 heads per platter—one on each side, but not always. (Sometimes the term *side* is substituted for *head,* since platters might be separated from their head assemblies; as is definitely the case with the removable media of a *floppy* drive.)

Tracks

The tracks are the thin concentric circular strips on a floppy disk or platter surface which comprise the magnetic medium to which data is written by the drive heads. These magnetic strips form a circle and are (therefore) two-dimensional. At least one head is required to read a single track. All information stored on the hard disk is recorded in tracks.

Cylinders

A cylinder comprises the same track number on each platter, spanning all such tracks across each platter surface that is able to store data (without regard to whether or not the track is "bad"). Thus, it is a three-dimensional structure. Any track comprising part of a specific cylinder can be written to and read from while the actuator assembly remains stationary, and one way in which hard drive manufacturers have increased drive access speed has been by increasing the number of platters which can be read at the same time.

As larger hard disks have come into use, a cylinder has become also a logical, rather than a physical, disk structure: standardised at 16,065 sectors (i.e. 255 tracks multiplied by 63 sectors per track).

Sectors

Tracks are subdivided. Each subdivision is called a sector, which is the smallest storage unit on a hard drive. Typically, a sector will hold 512 bytes of information. Some vendors of hard drives, and some software developers, are attempting to create a new standard for the future by revising the amount of data stored in a sector to 4,096 bytes.

Blocks and Clusters

In MSDOS and Windows communities the phrases *allocation unit* or *cluster* are used interchangeably to represent the group of sectors. Cluster is the smallest unit for the file system.

The Unix communities employ the term *block* to refer to a sector or group of sectors. For example, the Linux fdisk utility normally displays partition table information using 1024-

byte *blocks*, but also uses the word *sector* to help describe a disk's size in the phrase, *63 sectors per track*.

Note: The terms *cluster* and *block* are also often used separately from the context of physical disks. It is still, by convention, a power of 2 multiple of 512 bytes.

CHS Addressing

Hence, each Sector of data can be addressed by specifying a cylinder, head, and sector. The following formulas detail the CHS geometry.

The number of sectors on one side of a platter is:

```
Sectors Per Side = Tracks Per Side * Sectors Per track
```

Each side of a drive "disk" or "platter" will have one head. So the calculation for the Total Number of Sectors is:

```
Total Number of Sectors = Sectors Per Side * Number of Heads
```

Knowing that the standard size of a sector is 512 bytes, the calculation for the total size of the drive is:

```
Total storage capacity of a hard drive = Total Number of Sectors *
512 bytes per sector
```

Logical blocks in modern computer systems are typically 512 bytes each, though ISO 9660 CDs use 2048-byte blocks.

CHS to LBA mapping

CHS-tuples can be mapped onto LBA (Logical Block Addressing) addresses using the following formula:

$$A = (c \cdot N_{\text{heads}} + h) \cdot N_{\text{sectors}} + s - 1$$

Where A is the LBA address, N_{heads} is the number of heads on the disk, N_{sectors} is the number of sectors per track, and (c,h,s) is the CHS address.

Examples

A "1.44 MB" floppy disk has 80 tracks (numbered 0 to 79), 2 heads (numbered 0 to 1) and 18 sectors per track (numbered 1 to 18). Therefore, its capacity in sectors is computed as follows:

```
Total Number of Sectors = (80 * 18) * 2 = 2880
```

```
                                                    2880*512 bytes/sector =
1474560 bytes (1.44 MB)
```

The 16-byte entries within an MBR or EBR Partition Table have CHS-*tuples* which are limited to only (1023,254,63) for a total of 1024 cylinders, 255 heads and 63 sectors (values for cylinders and heads start at zero (0 ~ 1023 cylinders, 0 ~ 254 heads), and sector values start at one). For computers whose BIOS code was also limited to using only these CHS values, what was the largest size hard disk on which every *sector* could be accessed? Starting with the formula above, but also including the term, `512 bytes/sector`, the hard disk could be no larger than:

```
((1024 * 63) * 255) * 512 = 8,455,716,864 bytes (about 7.8 GiB)
```

History

Earlier hard drives used in the PC, such as MFM and RLL drives, divided each cylinder into an equal number of sectors, so the CHS values matched the physical properties of the drive. A drive with a CHS *tuple* of (500, 4, 32) would have 500 tracks per side on each platter, two platters (4 heads), and 32 sectors per track, with a total of 32,768,000 bytes (about 32.8 MB, or 31.25 MiB).

ATA/IDE drives were much more efficient at storing data and have replaced the now *archaic* MFM and RLL drives. They use zone bit recording (ZBR), where the number of sectors dividing each track varies with the location of groups of tracks on the surface of the platter. Tracks nearer to the edge of the platter contain more blocks of data than tracks close to the spindle, because there is more physical space within a given track near the edge of the platter. Thus, the CHS addressing scheme cannot correspond directly with the physical geometry of such drives, due to the varying number of sectors per track for different regions on a platter. Because of this, many drives still have a surplus of sectors (less than 1 cylinder in size) at the end of the drive, since the total number of sectors rarely, if ever, ends on a cylinder boundary.

An ATA/IDE drive can be set in the system BIOS with any configuration of cylinders, heads and sectors that do not exceed the capacity of the drive (or the BIOS), since the drive will convert any given CHS value into an actual address for its specific hardware configuration. This however can cause compatibility problems

For operating systems such as Microsoft DOS or older version of Windows, each partition must start and end at a cylinder boundary. Only some of the most modern operating systems (Windows XP included) may disregard this rule, but doing so can still cause some compatibility issues, especially if the user wants to perform dual booting on the same drive. Microsoft does not follow this rule with internal disk partition tools since Windows Vista.

Chapter 3

ATA over Ethernet

ATA over Ethernet (**AoE**) is a network protocol developed by the Brantley Coile Company, designed for simple, high-performance access of SATA storage devices over Ethernet networks. It is used to build storage area networks (SANs) with low-cost, standard technologies.

Operating system support

The following operating systems provide ATA over Ethernet (AoE) support:

OS	Support	Third-party drivers
Linux	Native (2.6.11+)	Coraid
Windows	Third-party	StarWind Software AoE Initiator , WinAoE , and WinVBlock
Mac OS X 10.4 and up	Third-party	2DegreesFrost
Mac OS X 10.5 and 10.6	Third-party	Small Tree Communications
Solaris	Third-party	Coraid
FreeBSD	Third-party	Coraid (outdated)
OpenBSD	Native (4.5-current)	
VMware	Third-party	Coraid
Plan 9 from Bell Labs	Native	

Linux target support

Linux can function as an AoE target using one of these independently-developed implementations:

- **vblade**, a userspace daemon that is part of the *aoetools* package.
- **kvblade**, a Linux kernel module.
- **ggaoed**, a userspace daemon that takes advantage of Linux-specific performance features.

Hardware support

The Coraid company offers an array of AoE SAN appliances under the EtherDrive brand, along with diskless gateways that add network-attached storage functionality, using the NFS or SMB protocols, to one or more AoE appliances.

In 2007 LayerWalker announced the world's first single-chip AoE hardware solution called miniSAN running at both Fast and Gigabit Ethernet grades. The miniSAN product family offers standard AoE server functions plus other management features that targets PC, consumer and SMB markets.

Protocol description

AoE runs on layer 2 Ethernet. AoE does not use internet protocol (IP), it cannot be accessed over the Internet or other IP networks. In this regard it is more comparable to Fibre Channel over Ethernet than iSCSI.

This approach makes AoE more lightweight (with less load on the host), makes it easier to implement, provides a layer of inherent security, and offers higher performance. The AoE specification is 12 pages compared with iSCSI's 257 pages.

ATA encapsulation

SATA (and older PATA) hard drives use the Advanced Technology Attachment (ATA) protocol to issue commands, such as read, write, and status. AoE encapsulates those commands inside Ethernet frames and lets them travel over an Ethernet network instead of a SATA or 40-pin ribbon cable. By using an AoE driver, the host operating system is able to access a remote disk as if it were directly attached.

The encapsulation of ATA provided by AoE is simple and low-level, allowing the translation to happen either at high performance or inside a small, embedded device, or both.

Routability

AoE runs directly on top of Ethernet, instead of an intermediate protocol such as TCP/IP. This reduces the significant CPU overhead of TCP/IP. However, this means that routers cannot be used to route a packet across disparate networks (such as the Internet). Instead, AoE packets can travel within a single local Ethernet storage area network (eg, a set of computers connected to the same switch or in the same VLAN).

Security

The non-routability of AoE is a source of inherent security (ie, an intruder can't connect through a router—they must physically plug into the local Ethernet switch where Ethernet frame tunneling over routed networks is not in use). However, there are no AoE-specific mechanisms for password verification or encryption. Additional security may be implemented at the file-system level. Certain AoE targets such as Coraid Storage appliances, vblade and GGAOED, support access lists ("masks") allowing connections only from specific MAC addresses (which can be spoofed).

Config string

The AoE protocol provides a mechanism for host-based cooperative locking. When more than one AoE initiator is using an AoE target, they must communicate. The hosts need a way to avoid interfering with one another as they use and modify the data on the shared AoE device.

One option provided by AoE is to use the storage device itself as the mechanism for determining the access of particular hosts. The AoE protocol includes a "config string" feature. The config string can record who is using the device. (It can also record any other information.) If more than one host tries to set the config string simultaneously, only one succeeds. The other host is informed of the conflict.

Related concepts

Although AoE is a simple network protocol, it opens up a complex realm of storage possibilities. To understand and evaluate these storage scenarios, it helps to be familiar with a few concepts.

Storage area networks

A SAN allows the physical hard drive to be removed from the server that uses it, and placed on the network. A SAN interface is similar in principle to non-networked interfaces such as SATA or SCSI. Most users will not use a SAN interface directly. Instead, they will connect to a server that uses a SAN disk instead of a local disk. Direct connection, however, can also be used.

When using a SAN network to access storage, there are several potential advantages over a local disk:

- It is easier to add storage capacity and the amount of storage is practically unlimited.
- It is easier to reallocate storage capacity.
- Data may be shared.
- Additionally, compared to other forms of networked storage, SANs are low-level and high performance

Utilizing storage area networks

To utilize a SAN disk, the host must format it with a filesystem. However, unlike a SATA or SCSI disk a SAN hard drive may be accessed by multiple machines. This is a source of both danger and opportunity.

Traditional filesystems (such as FAT or ext3) are designed to be accessed by a single host, and will cause unpredictable behavior if accessed by multiple machines. Such filesystems may be used, and AoE provides mechanisms whereby an AoE target can be guarded against simultaneous access (see: Config String).

Shared disk file systems allow multiple machines to use a single hard disk safely by coordinating simultaneous access to individual files. These filesystems can be used to allow multiple machines access to the same AoE target without an intermediate server or filesystem (and at higher performance).

Chapter 4

Device Configuration Overlay & Automatic Acoustic Management

Device Configuration Overlay

Device configuration overlay (**DCO**) is a hidden area on many of today's hard disk drives (HDDs). Usually when information is stored in either the DCO or host protected area (HPA), it is not accessible by the BIOS, OS, or the user. However, certain tools can be used to modify the HPA or DCO.

Uses

The Device Configuration Overlay (DCO), which was first introduced in the ATA-6 standard, "allows system vendors to purchase HDDs from different manufacturers with potentially different sizes, and then configure all HDDs to have the same number of sectors. An example of this would be using DCO to make an 80-gigabyte HDD appear as a 60-gigabyte HDD to both the (OS) and the BIOS.... Given the potential to place data in these hidden areas, this is an area of concern for computer forensics investigators. An additional issue for forensic investigators is imaging the HDD that has the HPA and or DCO on it. While certain vendors claim that their tools are able to both properly detect and image the HPA, they are either silent on the handling of the DCO or indicate that this is beyond the capabilities of their tool."

Guidance Software EnCase

One tool that handles the DCO is FastBloc SE. Guidance Software, creator of the popular computer forensics tool EnCase, acknowledges about their software write-blocking utility: "Fastbloc SE supports HPA and DCO as well as a combination of the two. Of note, the HPA is removed temporarily so the disk is not modified at the end, but DCO and the combination of HPA and DCO permanently alters the disk." EnCase claims to handle the HPA and DCO effectively using its DOS imaging utility or LinEn, its Linux imaging utility. This doesn't seem to be the case, however. LinEn 6.01 was validated by

the National Institute of Justice in October 2008, and they found that "The tool does not remove either Host Protected Areas (HPAs) or DCOs. However, the Linux test environment automatically removed the HPA on the test drive, allowing the tool to image sectors hidden by an HPA. The tool did not acquire sectors hidden by a DCO."

AccessData Forensic Toolkit (FTK)

AccessData, arguably Guidance Software's biggest competitor, offers its imaging utility free of charge. FTK Imager 2.5.3.14 was validated by the National Institute of Justice in June 2008. Their findings indicated that "If a physical acquisition is made of a drive with hidden sectors in either a Host Protected Area or a Device Configuration Overlay, the tool does not remove either an HPA or a DCO. The tool did not acquire sectors hidden by an HPA."

Automatic Acoustic Management

Automatic acoustic management (**AAM**) is a method for reducing acoustic emanations in AT Attachment (ATA) mass storage devices, such as ATA hard disk drives and ATAPI optical disc drives. AAM is an optional feature set for ATA/ATAPI devices; when a device supports AAM, the acoustic management parameters are adjustable through a software or firmware user interface.

The ATA/ATAPI sub-command for setting the level of AAM operation is an 8-bit value from 0 to 255. Most modern drives ship with the vendor-defined value of 0x00 in the acoustic management setting. This often translates to the max-performance value of 254 stated in the standard. Values between 128 and 254 (0x80 - 0xFE) enable the feature and select most-quiet to most-performance settings along that range. Though hard drive manufacturers may support the whole range of values, the settings are allowed to be banded, so many values could provide the same acoustic performance.

Though there is no definition of the function implemented to provide acoustic management in the ATA standard, most drives use power control of the head-positioning servo to reduce vibration induced by the head positioning mechanism. Western Digital calls this InteliSeek(tm) which uses only enough head acceleration to position the head at the target track and sector "just in time" to access data. Previous seek mechanisms used maximum power and acceleration to position the head. This operation induced the familiar clicking vibration emanating from a seeking hard drive. Western Digital provides a demonstration flash movie illustrating just-in-time head positioning on their web site.

In order to provide best acoustic performance, some drive manufacturers may limit the maximum seek velocity of the heads for AAM operation. This degrades performance by increasing the average seek time: some head movements are forced to wait an additional disk rotation before accessing data because the head was unable to move to the target position during the first rotation due to velocity limits. For example, benchmark tests with SiSoftware Sandra Lite on a Samsung HD154UI (1.5TB, SATA300, 3.5", 5400rpm, 32MB Cache) hard drive showed no measurable performance impact for an AAM setting

of 190, but the drive did become noticeably more quiet than the disabled setting (0). Selecting the most-quiet setting (128) caused average random access time to increase about 10% while quieting improved noticeably over the middle setting. On this drive, some quieting is available without performance impact, and even more quieting is available if some performance degradation is acceptable.

By contrast, a Western Digital WD1001FALS-00J7B0 (1TB, SATA300, 3.5", 7200rpm, 32MB Cache) disk drive shows a decrease from 18ms to 12.5ms by changing the value from 128 to 254, with little to no increase in noise. This drive did appear to be slightly less quiet than the Samsung in tests. Users should read manuals carefully for available settings of individual drives in each application.

AAM operates independently of advanced power management settings. However, selecting lower head acceleration (quieter operation) uses less power, so energy-conscious users might prefer the most-quiet setting (128) for power management purposes.

INCITS (formerly NCITS) first standardized AAM in the ATA/ATAPI-6 specification.

Chapter 5

Tagged Command Queuing & Disk Array Controller

Tagged Command Queuing

Tagged Command Queuing (**TCQ**) is a technology built into certain ATA and SCSI hard drives. It allows the operating system to send multiple read and write requests to a hard drive. ATA TCQ is not identical in function to the more efficient native command queuing (NCQ) used by SATA drives. SCSI TCQ does not suffer from the same limitations as ATA TCQ.

Before TCQ, an operating system was only able to send one request at a time. In order to boost performance, it had to decide the order of the requests based on its own, possibly incorrect, idea of what the hard drive was doing. With TCQ, the drive can make its own decisions about how to order the requests (and in turn relieve the operating system from having to do so). The result is that TCQ can improve the overall performance of a hard drive if it is implemented correctly.

Overview

For efficiency the sectors should be serviced in order of proximity to the current head position, rather than in the order received. The queue is constantly receiving new requests and fulfilling and removing existing requests, and re-ordering the queue according to the current pending read/write requests and the changing position of the head. The exact reordering algorithm may depend upon the controller and the drive itself, but the host computer simply makes requests as needed, leaving the controller to handle the details.

This queuing mechanism is sometimes referred to as "elevator seeking", as the image of a modern elevator in a building servicing multiple calls and processing them to minimise travel illustrates the idea well.

If the buttons for floors 5, 2, and 4 are pressed in that order with the elevator starting on floor 1, an old elevator would go to the floors in the order requested. A modern elevator

processes the requests to stop at floors in the logical order 2, 4, and 5, without unnecessary travel. Non-queueing disk drives service the requests in the order received, like an old elevator; queueing drives service requests in the most efficient order. This may improve performance slightly in a system used by a single user, but may dramatically increase performance in a system with many users making widely varied requests on the disk surface.

Comparison of SCSI TCQ, ATA TCQ, and SATA NCQ

SCSI TCQ

SCSI TCQ is the first popular version of TCQ and is still popular today. It allows tasks to be entered into a queue using one of three different modes:

- head of queue
- ordered
- simple

In *head of queue mode*, unique to SCSI TCQ, a task is pushed into the front of a queue, ahead of all other tasks including other pending head of queue tasks.. This mode is not used much because it can cause resource starvation when abused.

In *ordered mode*, a task must execute after all older tasks have completed and before all newer tasks begin to execute excluding newer head of queue tasks.

Simple mode allows tasks to execute in any order that does not violate the constraints on the tasks in the other two modes. After a command in a task is completed, a notification is sent by the device that completed the command to the host bus adapter. Whether or not SCSI TCQ causes massive interrupt overhead depends on the bus being used to connect the SCSI host bus adapter. On Conventional PCI, PCI-X, PCI Express, and other buses that permit it, first party DMA allows for low interrupt overhead. The older ISA bus required a SCSI host adapter to generate a interrupt to cause the CPU to program the third-party DMA engine to perform a transfer, and then required another interrupt to notify the CPU that a task in the queue was finished, causing high CPU overhead.

SCSI TCQ Tag Length

The SCSI-3 protocol permits 64 bits to be used in the tag field, allowing up to 2^{64} tasks in one task set to be issued before requiring that some of them complete before any more commands be issued. However, different protocols that implement the SCSI protocol might not permit the use of all 64 bits. For example, older parallel SCSI permits 8 bits of tag bits, iSCSI permits up to 32 tag bits, and Fibre Channel permits up to 16 bits of tag with tag `0xFFFF` reserved. This flexibility allows the designer of a protocol to trade off queuing ability against cost. Networks that can be large, such as iSCSI networks, benefit from more tag bits to deal with the larger number of disks in the network and the larger latencies such large networks generate, while smaller-scale networks, such as parallel

SCSI chains, do not have enough disks or latency to need many tag bits and can save money by using a system supporting fewer bits.

ATA TCQ

ATA TCQ was developed in attempt to bring the same benefits as SCSI to ATA drives. It is available in both Parallel and Serial ATA.

This effort was not very successful because the ATA bus started out as a reduced-pin-count ISA bus. The requirement for software compatibility made ATA host bus adapters act like ISA bus devices without first party DMA. When a drive was ready for a transfer, it had to interrupt the CPU, wait for the CPU to ask the disk what command was ready to execute, respond with the command that it was ready to execute, wait for the CPU to program the host bus adapter's third party DMA engine based on the result of that command, wait for the third party DMA engine to execute the command, and then had to interrupt the CPU again to notify it when the DMA engine finished the task so that the CPU could notify the thread that requested the task that the requested task was finished. Since responding to interrupts uses CPU time, CPU utilization rose quickly when ATA TCQ was enabled . Also, since interrupt service time can be unpredictable, there are times when the disk is ready to transfer data but is unable to do so because it must wait for a CPU to respond to the interrupt so that the CPU knows that it needs to program the third party DMA engine.

Therefore, this standard was rarely implemented because it caused high CPU utilization without improving performance enough to make this worthwhile. This standard allows up to 32 outstanding commands per device.

SATA NCQ

SATA NCQ is a modern standard which drastically reduces the number of required CPU interrupts compared to ATA TCQ. Like ATA TCQ, it allows up to 32 outstanding commands per device, but was designed to take advantage of the ability of SATA host bus adapters that are not emulating parallel ATA behavior to support first party DMA. Instead of interrupting the CPU before the task to force it to program the host bus adapter's DMA engine, the hard drive tells the host bus adapter which command it wants to execute, causing the host bus adapter to program its integrated first-party DMA engine with the parameters that were included in the command that was selected by the hard drive when it was first issued, and then the DMA engine moves the data needed to execute the command. To further reduce the interrupt overhead, the drive can withhold the interrupt with the task completed messages until it gathers many of them to send at once, allowing the operating system to notify many threads simultaneously that their tasks have been completed. If another task completes after such an interrupt is sent, the host bus adapter can concatenate the completion messages together if the first set of completion messages has not been sent to the CPU. This allows the hard disk firmware design to trade off disk performance against CPU utilization by determining when to withhold and when to send completion messages.

Disk Array Controller

A **disk array controller** is a device which manages the physical disk drives and presents them to the computer as logical units. It almost always implements hardware RAID, thus it is sometimes referred to as **RAID controller**. It also often provides additional disk cache.

A *disk array controller* name is often improperly shortened to a *disk controller*. The two should not be confused as they provide very different functionality.

Front-end and back-end side

Disk array controller provides front-end interfaces and back-end interfaces.

- Back-end interface communicates with controlled disks. Hence protocol is usually ATA (a.k.a. PATA; incorrectly called IDE), SATA, SCSI, FC or SAS.
- Front-end interface communicates with a computer's host adapter (HBA, Host Bus Adapter) and uses:
    - one of ATA, SATA, SCSI, FC; these are popular protocols used by disks, so by using one of them a controller may transparently emulate a disk for a computer
    - somewhat less popular protocol dedicated for a specific solution: FICON/ESCON, iSCSI, HyperSCSI, ATA over Ethernet or InfiniBand

A single controller *may* use different protocols for back-end and for front-end communication. Many enterprise controllers use FC on front-end and SATA on back-end.

Enterprise controllers

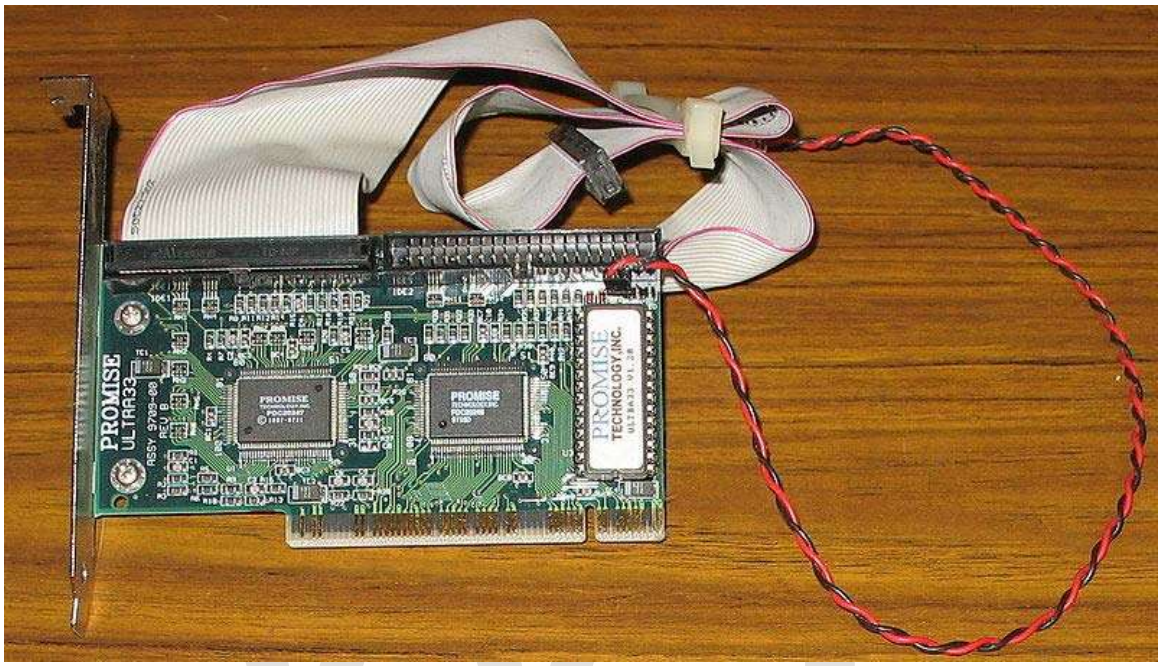
In a modern enterprise architecture disk array controllers are parts of physically independent enclosures, such as disk arrays placed in a storage area network (SAN) or network-attached storage (NAS) servers.

Those external disk arrays are usually purchased as an integrated subsystem of RAID controllers, disk drives, power supplies, and management software. It is up to controllers to provide advanced functionality (various vendors name these differently):

- automatic failover to another controller (transparent to computers transmitting data)
- long-running operations performed without downtime
    - forming a new RAID set
    - reconstructing *degraded* RAID set (after a disk failure)
    - adding a disk to online RAID set
    - removing a disk from a RAID set (rare functionality)

    - partitioning a RAID set to separate volumes/LUNs
- snapshots
- Business Continuance Volumes (BCV)
- replication with a remote controller....

Simple controllers



Promise Technology ATA RAID controller

A simple disk array controller may be fit inside a computer, either as a PCI expansion card or just built into the motherboard. Such controller usually provides host bus adapter (HBA) functionality itself to save physical space. Hence it is sometimes called a **RAID adapter**.

More recently (February 2007) Intel has started integrating their own Matrix RAID controller in their more upmarket motherboards giving control over 4 devices and an additional 2 SATA connectors, totalling to 6 SATA connections (3Gbit/s each). For backward compatibility one IDE connector enabling to connect 2 ATA devices (100 Mbit/s) is also present.

History

While hardware RAID controllers were available for a long time, they always required expensive SCSI hard drives and aimed at the server and high-end computing market. SCSI technology advantages include allowing up to 15 devices on one bus, independent data transfers, hot-swapping, much higher MTBF.

Around 1997, with the introduction of ATAPI-4 (and thus the Ultra-DMA-Mode 0, which enabled fast data transfers with less CPU utilization) the first ATA RAID controllers were introduced as PCI expansion cards. Those RAID systems made their way to the consumer market, where the users wanted the fault-tolerance of RAID without investing in expensive SCSI drives.

ATA drives make it possible to build RAID systems at lower cost than with SCSI, but most ATA RAID controllers lack a dedicated buffer or high-performance XOR hardware for parity calculation. As a result, ATA RAID performs relatively poorly compared to most SCSI RAID controllers. Additionally, data safety suffers if there is no battery backup to finish writes interrupted by a power outage.

Chapter 6

Host Protected Area & Etherdrive

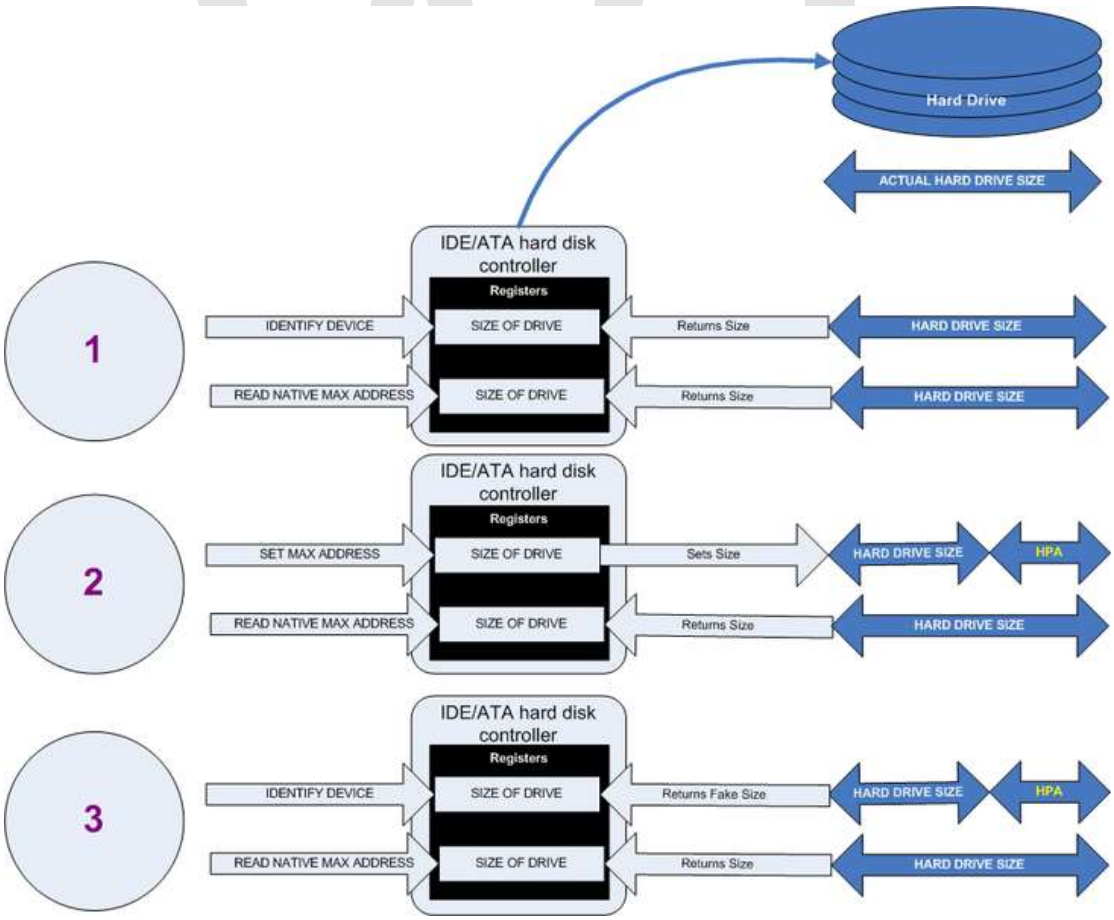
Host Protected Area

The **host protected area**, sometimes referred to as **hidden protected area**, is an area of a hard drive that is not normally visible to an operating system (OS).

History

HPA was first introduced in the ATA-4 standard cxv (T13, 2001).

How it works



Creation of an HPA. The diagram shows how a host protected area (HPA) is created. # IDENTIFY DEVICE returns the true size of the hard drive. READ NATIVE MAX

ADDRESS returns the true size of the hard drive. # SET MAX ADDRESS reduces the reported size of the hard drive. READ NATIVE MAX ADDRESS returns the true size of the hard drive. An HPA has been created. # IDENTIFY DEVICE returns the now fake size of the hard drive. READ NATIVE MAX ADDRESS returns the true size of the hard drive, the HPA is in existence.

The IDE controller has registers that contain data that can be queried using ATA commands. The data returned gives information about the drive attached to the controller. There are three ATA commands involved in creating and utilizing a hidden protected area. The commands are:

- IDENTIFY DEVICE
- SET MAX ADDRESS
- READ NATIVE MAX ADDRESS

Operating systems use the IDENTIFY DEVICE command to find out the addressable space of a hard drive. The IDENTIFY DEVICE command queries a particular register on the IDE controller to establish the size of a drive.

This register however can be changed using the SET MAX ADDRESS ATA command. If the value in the register is set to less than the actual hard drive size then effectively a host protected area is created. It is protected because the OS will work with only the value in the register that is returned by the IDENTIFY DEVICE command and thus will never be able to address the parts of the drive that lie within the HPA.

The HPA is useful only if other software or firmware (e.g. BIOS) is able to utilize it. Software and firmware that are able to utilize the HPA are referred to as 'HPA aware'. The ATA command that these entities use is called READ NATIVE MAX ADDRESS. This command accesses a register that contains the true size of the hard drive. To use the area, the controlling HPA-aware program changes the value of the register read by IDENTIFY DEVICE to that found in the register read by READ NATIVE MAX ADDRESS. When its operations are complete, the register read by IDENTIFY DEVICE is returned to its original fake value.

Use

- HPA can be used by various booting and diagnostic utilities, normally in conjunction with the BIOS. An example of this implementation is the Phoenix FirstBIOS, which utilizes BEER (boot engineering extension record) and PARTIES (protected area run-time interface extension services).
- Computer manufacturers may use the area to contain a preloaded OS for install and recovery purposes (instead of providing DVD or CD media).
- Dell notebooks hide Dell MediaDirect utility in HPA. IBM and LG notebooks hide system restore software in HPA.
- HPA is also used by various theft recovery and monitoring service vendors. For example the laptop security firm Computrace use the HPA to load software that

reports to their servers whenever the machine is booted on a network. HPA is useful to them because even when a stolen laptop has its hard drive formatted the HPA remains untouched.

- HPA can also be used to store data that is deemed illegal and is thus of interest to government and police computer forensics teams.
- Some vendor-specific external drive enclosures (Maxtor) are known to use HPA to limit the capacity of unknown replacement hard drives installed into the enclosure. When this occurs, the drive may appear to be limited in size (e.g. 128 GB), which can look like a BIOS or dynamic drive overlay (DDO) problem. In this case, one must use software utilities that use READ NATIVE MAX ADDRESS and SET MAX ADDRESS to change the drive's reported size back to its native size, and avoid using the external enclosure again with the affected drive.
- Some rootkits hide in the HPA to avoid being detected by anti-rootkit and antivirus software.

Identification and manipulation

Identification of HPA on a hard drive can be achieved by a number of tools and methods.

Identification tools

- The Sleuth Kit (free, open software) by Brian Carrier. (HPA identification is currently Linux-only.)
- The ATA Forensics Tool (TAFT) by Arne Vidstrom.
- EnCase by Guidance Software
- Access Data's Forensic Toolkit

Identification methods

Using Linux, there are a couple of ways to detect the existence of an HPA. The latest Linux versions will print a message when the system is booting. For example:

```
dmesg | less
[...]
hdb: Host Protected Area detected.
    current capacity is 12000 sectors (6 MB)
    native  capacity is 120103200 sectors (61492 MB)
```

With program hdparm, version >= 8.0, where X is your drive letter:

```
 hdparm -N /dev/sdX
```

For versions of hdparm < 8, one can compare the number of sectors output from 'hdparm -I' with the number of sectors reported for the hard drive model's published statistics.

Manipulation tools

Creating and manipulating HPA on a hard drive can be achieved by a number of tools.

- HDAT2 by Lubomir Cabla.
- setmax by Andries E. Brouwer
- Feature Tool by Hitachi Global Storage Technologies.
- MHDD (created by Dmitry Postrigan) is a freeware tool for hard drives that among other low-level functionalities provides information about the HPA state of a disk and can manipulate it.
- hdparm is a Linux program for reading and writing ATA and SATA hard drive parameters.
- FreeBSD has the hw.ata.setmax sysctl which can be set to 1.

Manipulation methods

Using the Linux program hdparm with version >=8.0 you can modify the HPA directly. Where ABC is the number of visible sectors and X is the drive letter:

```
hdparm -NpABC /dev/sdX
```

Etherdrive

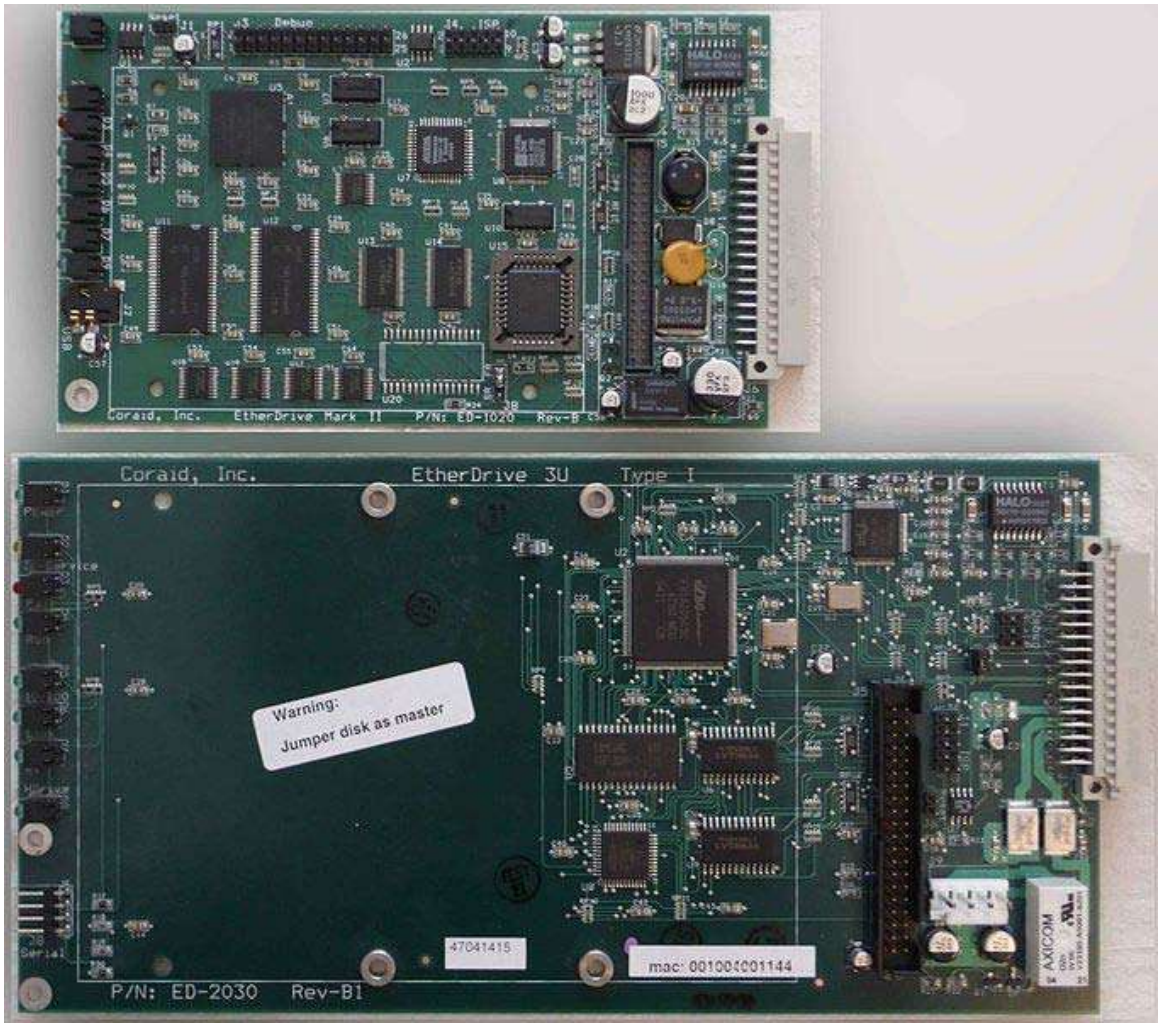


Figure 1. Early EtherDrives:
(Top) 2.5 PATA Blade
(Bottom) 3.5 PATA Blade

Figure 2. SR EtherDrive Cluster
These 60 Units can hold over 780 TeraBytes at RAID5 with 60 hot spares.

EtherDrive is a brand name for a variety of storage area network devices based upon the ATA over Ethernet (AoE) protocol. EtherDrive is a registered trademark of Coraid, Inc.. EtherDrive is an invented word, owned and used as a trademark by Coraid since 2002 and registered with the United States Patent and Trademark Office in 2004. The word was invented by Brantley W. Coile as a portmanteau of the words Ethernet and disk drive.

History

The first commercial transaction involving interstate commerce of an EtherDrive branded product was to Geoff Collyer. Some of the first EtherDrive products were embedded Z80 based boards that acted as AoE converters (see Figure 1). These boards were attached like a daughterboard to PATA disk drives mounted and connected to a backplane that provided only power and an RJ-45 jack. These original PATA blades were aligned in a single row per shelf and addressed by over Ethernet with a shelf:slot address. The slot component eventually became referred to as a LUN.

Today, EtherDrive devices provide RAID 0, 1, 5, 10, and JBOD. These use AoE to talk to the initiator but provide a hardware RAID close to disk. The device also provides a block scrubbing mechanism called RAIDShield that systematically checks every block on every disk for potential failures. When a bad block is encountered, the block is reconstructed from parity and written to another part of the disk.

EtherDrives have been used as storage for high altitude atmospheric research and aeronautical applications. Combined with SSD disks the technology is an easy solution to

data acquisition in the embedded space. Since it uses AoE the device is presented to the host OS as block storage, and thus the EtherDrive requires minimum overhead from the host system.

Chapter 7

Logical Block Addressing & Intel Rapid Storage Technology

Logical Block Addressing

Logical block addressing (LBA) is a common scheme used for specifying the location of blocks of data stored on computer storage devices, generally secondary storage systems such as hard disks.

LBA is a particularly simple linear addressing scheme; blocks are located by an integer index, with the first block being LBA 0, the second LBA 1, and so on.

IDE standard included 22-bit LBA as an option, which was further extended to 28-bit with the release of ATA-1 (1994) and to 48-bit with the release of ATA-6 (2003). Most hard drives released after 1996 implement Logical block addressing.

Overview

In logical block addressing, only one number is used to address data, and each linear base address describes a single block.

The LBA scheme replaces earlier schemes which exposed the physical details of the storage device to the software of the operating system. Chief among these was the cylinder-head-sector (CHS) scheme, where blocks were addressed by means of a tuple which defined the cylinder, head, and sector at which they appeared on the hard disk. CHS didn't map well to devices other than hard disks (such as tapes and networked storage), and was generally not used for them. CHS was used in early MFM and RLL drives, and both it and its successor Extended Cylinder-Head-Sector (ECHS) were used in the first ATA drives. However, current disk drives use zone bit recording, where the number of sectors per track depends on the track number. Even though the disk drive will report some CHS values as sectors per track (SPT) and heads per cylinder (HPC), they have little to do with the disk drive's true geometry.

LBA was first introduced in SCSI as an abstraction. While the drive controller still addresses data blocks by their CHS address, this information is generally not used by the SCSI device driver, the OS, filesystem code, or any applications (such as databases) that access the "raw" disk. System calls requiring block-level I/O pass LBA definitions to the storage device driver; for simple cases (where one volume maps to one physical drive), this LBA is then passed directly to the drive controller.

In RAID devices and SANs and where logical drives (LUNs) are composed via LUN virtualization and aggregation), LBA addressing of individual disk should be translated by a software layer to provide uniform LBA addressing for the entire storage device.

Enhanced BIOS

The earlier IDE standard from Western Digital introduced 22 bit LBA addressing; in 1994, the ATA-1 standard allowed for 28 bit addresses in both LBA and CHS modes. The CHS scheme used 16 bits for cylinder, 4 bits for head and 8 bits for sector, counting sectors from 1 to 255. This means the reported number of heads never exceeds 16 (0-15), the number of sectors can be 255 (1-255; though 63 is often the largest used) and the number of cylinders can be as large as 65,536 (0-65535), limiting disk size to 128 GiB (≈137.4 GB), assuming 512 byte sectors. These values can be accessed by issuing the ATA command "Identify Device" (ECh) to the drive.

However IBM BIOS implementation defined in the INT 13H disk access routines used quite a different 24-bit scheme for CHS addressing, with 10 bits for cylinder, 8 bits for head, and 6 bits for sector, or 1024 cylinders, 256 heads, and 63 sectors. This INT 13H implementation had pre-dated the ATA standard, as it was introduced when the IBM PC had only floppy disk storage, and when hard disk drives were introduced on the IBM PC/XT, INT 13H interface could not be practically redesigned due to backward compatibility issues. Overlapping ATA CHS mapping with BIOS CHS mapping produced the lowest common denominator of 10:4:6 bits, or 1024 cylinders, 16 heads, and 63 sectors, which gave the practical limit of 1024×16×63 sectors and 528 Mbytes (504 MiB), assuming 512 byte sectors.

In order for BIOS to overcome this limit and successfully work with large hard drives, a CHS translation scheme had to be implemented in BIOS disk I/O routines which would convert between 24-bit CHS used by INT 13H and 28-bit CHS numbering used by ATA. The translation scheme was called **Large** or **Bit Shift Translation**. This method would remap 16:4:8 bit ATA cylinders and heads to 10:8:6 bit scheme used by INT 13H, generating much more "virtual" drive heads than the physical disk reported. This increased the practical limit to 1024×256×63 sectors, or 8.4 Gbytes (7.8 GiB).

To further overcome this limit, INT 13H Extensions were introduced with **BIOS Enhanced Disk Drive Services** specification, which removed practical limits on disk size for operating systems which are aware of this new interface, such as *DOS 7.0* component in Windows 95. This *Enhanced BIOS* subsystem supports LBA addressing

with **LBA** or **LBA-Assist** method, which uses native 28-bit LBA for addressing ATA disks and performs CHS conversion as needed.

The **Normal** or **None** method reverts to the earlier 10:4:6 bit CHS mode which does not support addressing more than 528 Mbytes.

Until the release of ATA-2 standard in 1996, there were a handful of large hard drives which did not support LBA addressing, so only Large or Normal methods could be used. However using the Large method also introduced portability problems, as different BIOSes often used different and incompatible translation methods, and hard drives partitioned on a computer with BIOS from a particular vendor often could not be read on a computer with a different make of BIOS. The solution was to use conversion software such as OnTrack Disk Manager, EZ-Drive, etc., which installed to the disk's OS loader and replaced INT 13H routines at boot time with custom code. This software could also enable LBA and INT 13H Extensions support for older computers with non LBA-compliant BIOSes.

The current 48-bit LBA scheme, introduced in 2003 with ATA-6 standard, allows addressing up to 128 PiB. Current PC-Compatible computers support INT 13H Extensions, which use 64-bit structures for LBA addressing and should encompass any future extension of LBA addressing, though modern operating systems implement direct disk access and do not use the BIOS subsystems, except at boot load time. However, the common DOS style Master boot record partition table only supports disk partitions up to 2 TiB in size. For large partitions this needs to be replaced by another scheme for instance the GUID Partition Table which has the same 64-bit limit as the current INT 13H Extensions. Support for this is poor as of 2010 due to Windows requiring Extensible Firmware Interface additions to the BIOS to boot using GPT.

CHS conversion

LBA and CHS equivalence with 16 heads per cylinder

LBA Value	CHS *Tuple*
0	0, 0, 1
1	0, 0, 2
2	0, 0, 3
62	0, 0, 63
945	0, 15, 1
1007	0, 15, 63
1008	1, 0, 1
1070	1, 0, 63
1071	1, 1, 1
1133	1, 1, 63
1134	1, 2, 1

2015	1, 15, 63
2016	2, 0, 1
16,127	15, 15, 63
16,128	16, 0, 1
32,255	31, 15, 63
32,256	32, 0, 1
16,450,559	16319, 15, 63
16,514,063	16382, 15, 63

CHS (cylinder/head/sector) tuples can be mapped to LBA address with the following formula:

$$LBA = ((C \times HPC) + H) \times SPT + S - 1$$

where,

- C, H and S are the cylinder number, the head number, and the sector number
- LBA is the logical block address
- HPC is the maximum number of heads per cylinder (reported by disk drive, typically 16 for 28-bit LBA)
- SPT is the maximum number of sectors per track (reported by disk drive, typically 63 for 28-bit LBA)

LBA addresses can be mapped to CHS tuples with the following formula:

$$C = LBA \div (SPT \times HPC)$$
$$H = (LBA \div SPT) \bmod HPC$$
$$S = (LBA \bmod SPT) + 1$$

where

- mod is the modulo operation, i.e. the remainder, and
- $\div$ is integer division, i.e. the quotient of the division.

According to the ATA specifications, "If the content of words (61:60) is greater than or equal to 16,514,064 then the content of word 1 [the number of logical cylinders] shall be equal to 16,383." Therefore for LBA 16450559, an ATA drive may actually respond with the CHS *tuple* (16319, 15, 63), and the number of cylinders in this scheme must be much larger than 1024 allowed by INT 13H.

OS dependencies

Operating systems that are sensitive to BIOS-reported drive geometry include Solaris, DOS and Windows NT family (NT, 2000, XP, Server 2003, Vista, and 7) where NTLDR uses Master boot record which addresses the disk using CHS; x86-64 and Itanium versions of Windows can partition the drive with GUID Partition Table which uses LBA addressing.

Some operating systems do not require any translation because they do not use geometry reported by BIOS in their boot loaders. Among these operating systems are BSD, Linux, Mac OS X, OS/2 and ReactOS.

Intel Rapid Storage Technology

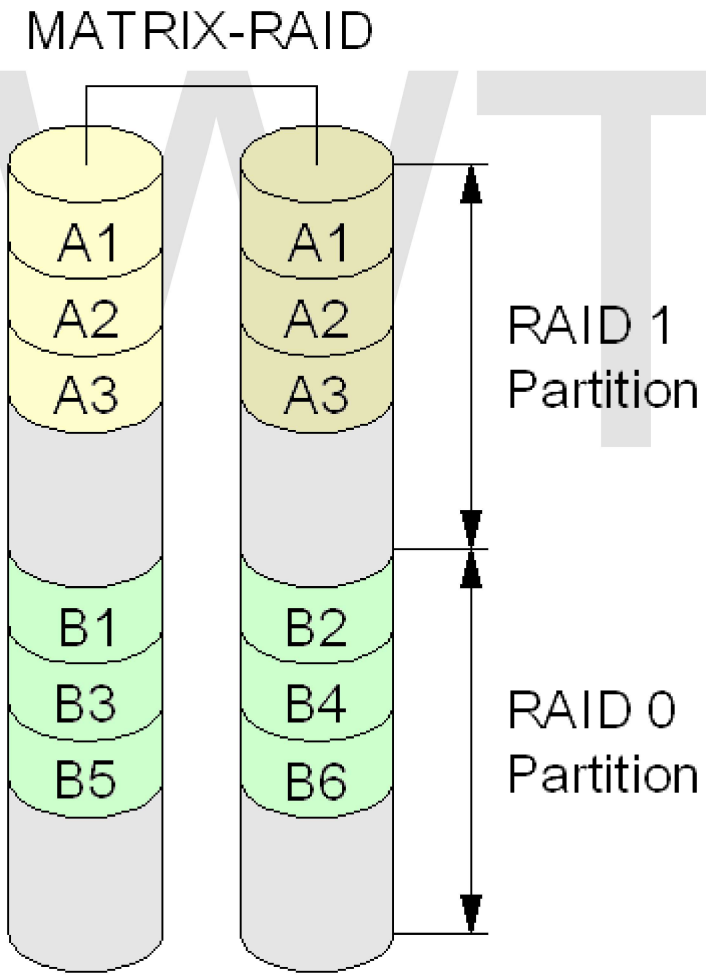


Diagram of a Matrix RAID setup.

Intel Rapid Storage Technology (formerly Intel Matrix RAID) is a firmware RAID system, rather than hardware RAID or software RAID. It first appeared in the ICH6R "southbridge" chip. Intel has continued to use an 'R' at the end of the southbridge's name (e.g. ICH9**R** instead of ICH9) to indicate when a southbridge contains their Matrix RAID technology and no other upgrades. Complicating the matter is that instead of "R," a "DO," "DH," etc has indicated a southbridge that combined RAID with non-RAID-related upgrades to the southbridge. Like all RAID, Intel Matrix RAID employs two or more physical hard disks which the operating system will treat as a single disk, in order to increase redundancy which avoids data loss (as all RAID levels except RAID 0 do), and/or to increase the speed at which data is written to and/or read from a disk.

Intel Matrix RAID is *not* a new RAID level. One of the features that Intel Matrix RAID has, which many other RAID implementations lack, is that different areas (e.g. partitions or logical volumes) on the same disk can be assigned to different RAID devices. Currently—in the ICH10R -- RAID 0, RAID 1, RAID 10, and RAID 5 are supported.

Intel's recommended setup is to put any critical applications and data on a RAID 1, 5, or 10 volume. The thinking being that protection from losing the user's personal data and the OS and program configuration settings is more important than having the pure performance (speed) increase of RAID 0. On the other hand, the RAID 0 volume in Matrix RAID is recommended mostly for working with large files, such as videos during editing, and for non-critical files where fast storage will increase performance (swap files, for example, or read-only files that are backed-up on a separate PC).

In 2010, Intel renamed the Intel Matrix RAID to Intel Rapid Storage Technology and replaced the graphical user interface with a simpler and less advanced version.

Operating system support

Linux supports Matrix RAID through **DM-RAID** and MD-RAID. DM-RAID does not provide a graphical utility to configure the arrays or notify the user of disk errors/failures, and will not activate the Intel Matrix RAID on many motherboards (due to incompatibilities). All the functionality of the Windows driver is also not available; such as creation of RAID volumes (which must be performed in the ROM, or using Windows).

FreeBSD and MidnightBSD support Intel Matrix RAID using ataraid, managed through atacontrol. However, there are critical reliability issues which include array device renaming when a disk in an array is replaced, an array being considered healthy if the machine reboot/crashes during an array rebuild, and kernel panics when a disk is lost or is removed from the bus. Some of these problems, when experienced in combination, could result in the loss of an entire array (even in the case of RAID 1).

Windows has full support for Intel Matrix RAID, including creation of RAID volumes.

VMware ESXi 4 does not support any RAID function nor Intel Matrix RAID based on Intel ICHxR controllers.VMware Community Thread.

PGPDisk does not support Intel Matrix RAID based on Intel ICHxR, and does not support standalone drives if the "RAID" mode is enabled on the motherboard.

Chapter 8

Hard Disk Drive

Hard disk drive

Interior of a hard disk drive

Date invented December 24, 1954

Invented by An IBM team led by Rey Johnson

A **hard disk drive** (HDD) is a non-volatile, random access device for digital data. It features rotating rigid platters on a motor-driven spindle within a protective enclosure. Data is magnetically read from and written to the platter by read/write heads that float on a film of air above the platters.

Introduced by IBM in 1956, hard disk drives have fallen in cost and physical size over the years while dramatically increasing in capacity. Hard disk drives have been the dominant device for secondary storage of data in general purpose computers since the early 1960s. They have maintained this position because advances in their areal recording density have kept pace with the requirements for secondary storage. Today's HDDs operate on high-speed serial interfaces; i.e., serial ATA (SATA) or serial attached SCSI (SAS).

History

Hard disk drives were introduced in 1956 as data storage for an IBM accounting computer and were developed for use with general purpose mainframe and mini computers.

Driven by areal density doubling every two to four years since their invention, HDDs have changed in many ways, a few highlights include:

- Capacity per HDD increasing from 3.75 megabytes to greater than 1 terabyte, a greater than 270 thousand to 1 improvement.
- Size of HDD decreasing from 87.9 cubic feet (a double wide refrigerator) to 0.002 cubic feet (2½-inch form factor, a pack of cards), a greater than 44 thousand to 1 improvement.
- Price decreasing from about $15,000 per megabyte to less than $0.0001 per megabyte ($100/1 terabyte), a greater than 150 million to 1 improvement.
- Average access time decreasing from greater than 0.1 second to a few thousandths of a second, a greater than 40 to 1 improvement.
- Market application expanding from general purpose computers to most computing applications including consumer applications.

Technology

Magnetic recording

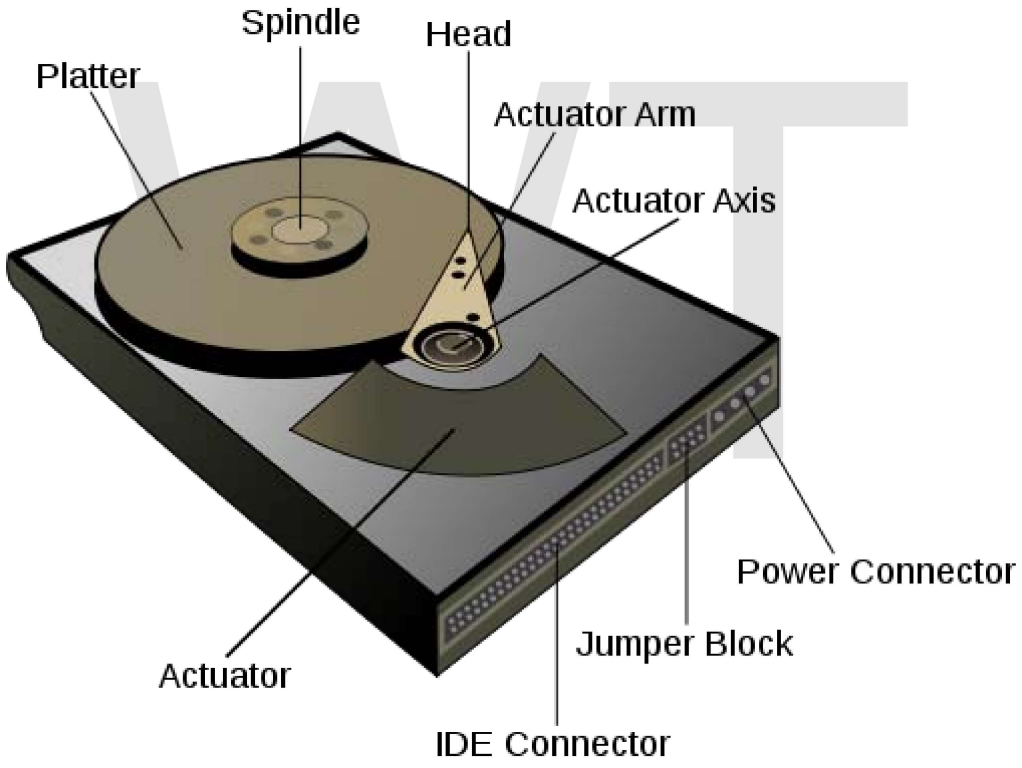
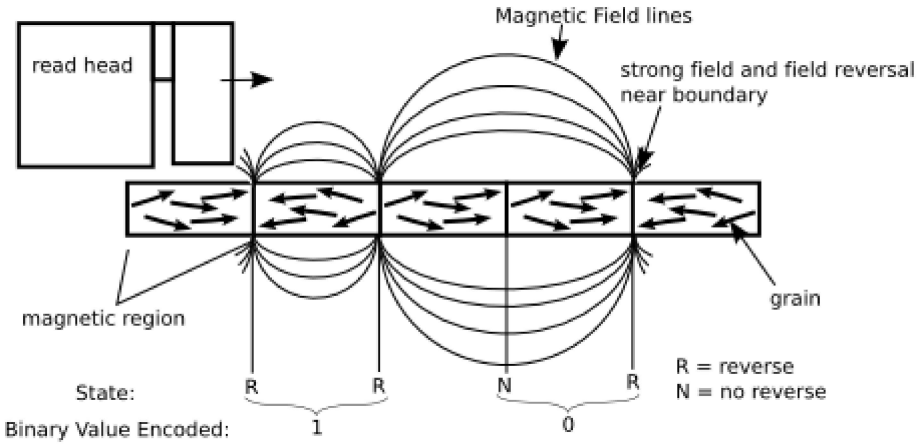


Diagram of a computer hard disk drive

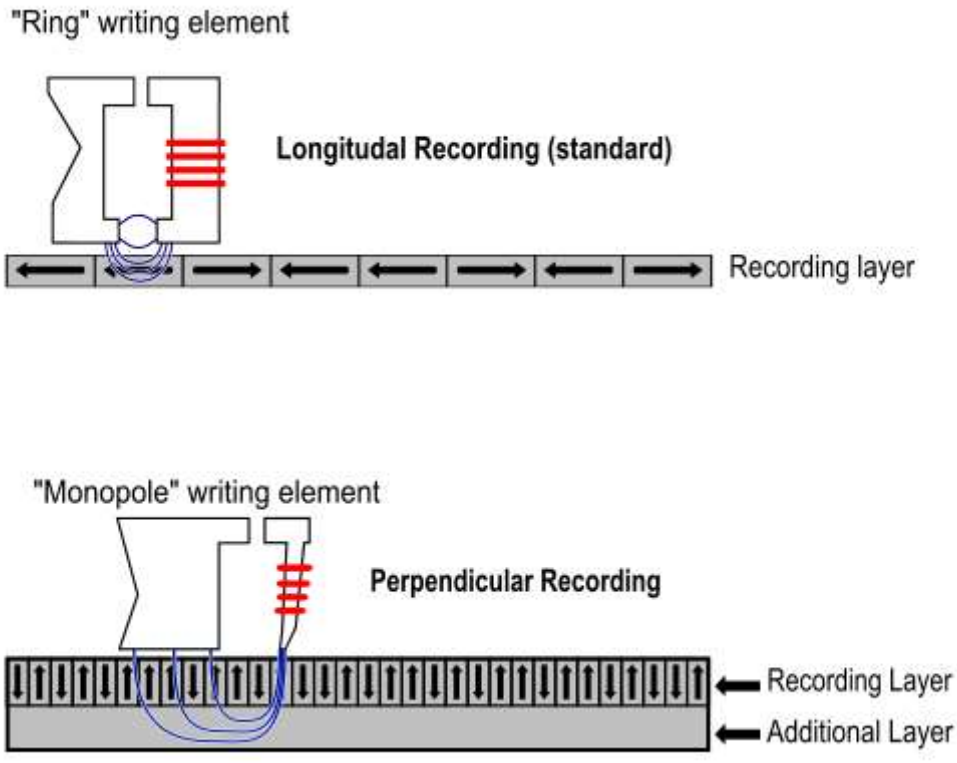
HDDs record data by magnetizing ferromagnetic material directionally. Sequential changes in the direction of magnetization represent patterns of binary data bits. The data are read from the disk by detecting the transitions in magnetization and decoding the originally written data. Different encoding schemes, such as Modified Frequency Modulation, group code recording, run-length limited encoding, and others are used.

A typical HDD design consists of a spindle that holds flat circular disks called platters, onto which the data are recorded. The platters are made from a non-magnetic material,

usually aluminum alloy or glass, and are coated with a shallow layer of magnetic material typically 10–20 nm in depth, with an outer layer of carbon for protection. For reference, standard copy paper is 0.07–0.18 millimetre (70,000–180,000 nm).



A cross section of the magnetic surface in action. In this case the binary data are encoded using frequency modulation.



Perpendicular recording

The platters are spun at speeds varying from 3,000 RPM in energy-efficient portable devices, to 15,000 RPM for high performance servers. Information is written to, and read

from a platter as it rotates past devices called read-and-write heads that operate very close (tens of nanometers in new drives) over the magnetic surface. The read-and-write head is used to detect and modify the magnetization of the material immediately under it. In modern drives there is one head for each magnetic platter surface on the spindle, mounted on a common arm. An actuator arm (or access arm) moves the heads on an arc (roughly radially) across the platters as they spin, allowing each head to access almost the entire surface of the platter as it spins. The arm is moved using a voice coil actuator or in some older designs a stepper motor.

The magnetic surface of each platter is conceptually divided into many small sub-micrometer-sized magnetic regions referred to as magnetic domains. In older disk designs the regions were oriented horizontally and parallel to the disk surface, but beginning about 2005, the orientation was changed to perpendicular to allow for closer magnetic domain spacing. Due to the polycrystalline nature of the magnetic material each of these magnetic regions is composed of a few hundred magnetic grains. Magnetic grains are typically 10 nm in size and each form a single magnetic domain. Each magnetic region in total forms a magnetic dipole which generates a magnetic field.

For reliable storage of data, the recording material needs to resist self-demagnetization, which occurs when the magnetic domains repel each other. Magnetic domains written too densely together to a weakly magnetizable material will degrade over time due to physical rotation of one or more domains to cancel out these forces. The domains rotate sideways to a halfway position that weakens the readability of the domain and relieves the magnetic stresses. Older hard disks used iron(III) oxide as the magnetic material, but current disks use a cobalt-based alloy.

A write head magnetizes a region by generating a strong local magnetic field. Early HDDs used an electromagnet both to magnetize the region and to then read its magnetic field by using electromagnetic induction. Later versions of inductive heads included metal in Gap (MIG) heads and thin film heads. As data density increased, read heads using magnetoresistance (MR) came into use; the electrical resistance of the head changed according to the strength of the magnetism from the platter. Later development made use of spintronics; in these heads, the magnetoresistive effect was much greater than in earlier types, and was dubbed "giant" magnetoresistance (GMR). In today's heads, the read and write elements are separate, but in close proximity, on the head portion of an actuator arm. The read element is typically magneto-resistive while the write element is typically thin-film inductive.

The heads are kept from contacting the platter surface by the air that is extremely close to the platter; that air moves at or near the platter speed. The record and playback head are mounted on a block called a slider, and the surface next to the platter is shaped to keep it just barely out of contact. This forms a type of air bearing.

In modern drives, the small size of the magnetic regions creates the danger that their magnetic state might be lost because of thermal effects. To counter this, the platters are coated with two parallel magnetic layers, separated by a 3-atom layer of the non-

magnetic element ruthenium, and the two layers are magnetized in opposite orientation, thus reinforcing each other. Another technology used to overcome thermal effects to allow greater recording densities is perpendicular recording, first shipped in 2005, and as of 2007 the technology was used in many HDDs.

Components

A hard disk drive with the disks and motor hub removed showing the copper colored stator coils surrounding a bearing at the center of the spindle motor. The orange stripe along the side of the arm is a thin printed-circuit cable. The spindle bearing is in the center. The actuator is in the upper left.

A typical hard disk drive has two electric motors; a disk motor to spin the disks and an actuator (motor) to position the read/write head assembly across the spinning disks.

The disk motor has an external rotor attached to the disks; the stator windings are fixed in place.

Opposite the actuator at the end of the head support arm is the read-write head (near center in photo); thin printed-circuit cables connect the read-write heads to amplifier electronics mounted at the pivot of the actuator. A flexible, somewhat U-shaped, ribbon cable, seen edge-on below and to the left of the actuator arm continues the connection to the controller board on the opposite side.

The head support arm is very light, but also stiff; in modern drives, acceleration at the head reaches 550 *G*s.

The silver-colored structure at the upper left of the first image is the top plate of the actuator, a permanent-magnet and moving coil motor that swings the heads to the desired position (it is shown removed in the second image). The plate supports a squat neodymium-iron-boron (NIB) high-flux magnet. Beneath this plate is the moving coil, often referred to as the *voice coil* by analogy to the coil in loudspeakers, which is attached to the actuator hub, and beneath that is a second NIB magnet, mounted on the bottom plate of the motor (some drives only have one magnet).

The voice coil itself is shaped rather like an arrowhead, and made of doubly coated copper magnet wire. The inner layer is insulation, and the outer is thermoplastic, which bonds the coil together after it is wound on a form, making it self-supporting. The portions of the coil along the two sides of the arrowhead (which point to the actuator bearing center) interact with the magnetic field, developing a tangential force that rotates the actuator. Current flowing radially outward along one side of the arrowhead and radially inward on the other produces the tangential force. If the magnetic field were uniform, each side would generate opposing forces that would cancel each other out. Therefore the surface of the magnet is half N pole, half S pole, with the radial dividing line in the middle, causing the two sides of the coil to see opposite magnetic fields and produce forces that add instead of canceling. Currents along the top and bottom of the coil produce radial forces that do not rotate the head.

Error handling

Modern drives also make extensive use of Error Correcting Codes (ECCs), particularly Reed–Solomon error correction. These techniques store extra bits for each block of data that are determined by mathematical formulas. The extra bits allow many errors to be fixed. While these extra bits take up space on the hard drive, they allow higher recording densities to be employed, resulting in much larger storage capacity for user data. In 2009, in the newest drives, low-density parity-check codes (LDPC) are supplanting Reed-Solomon. LDPC codes enable performance close to the Shannon Limit and thus allow for the highest storage density available.

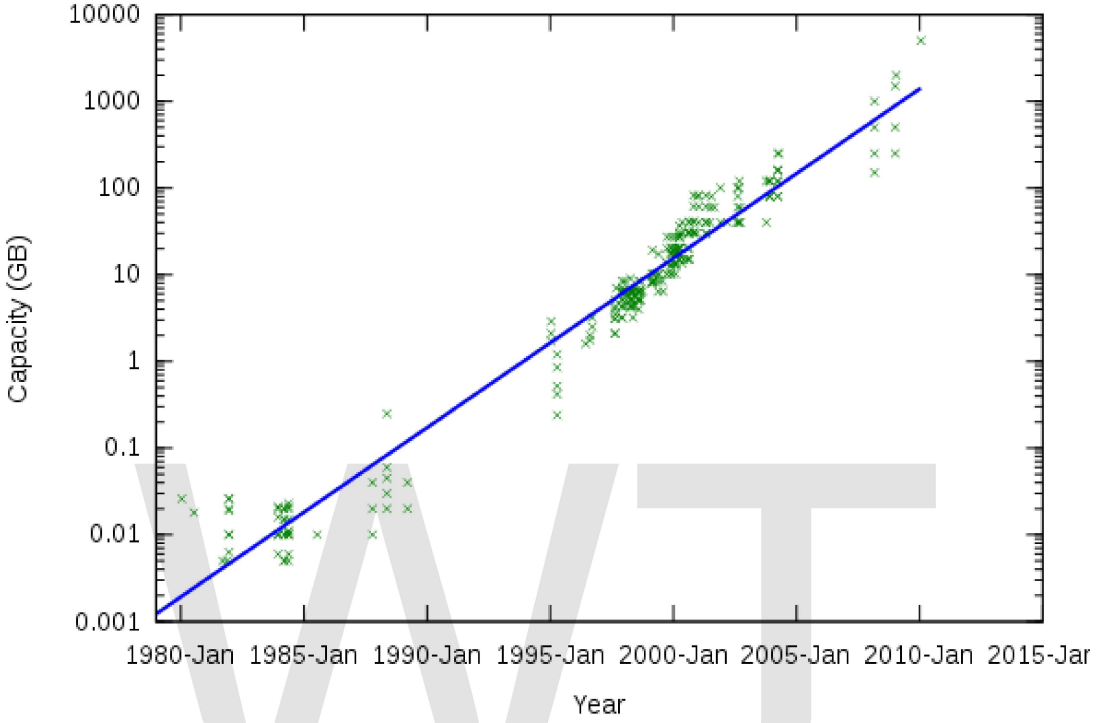
Typical hard drives attempt to "remap" the data in a physical sector that is going bad to a spare physical sector—hopefully while the errors in that bad sector are still few enough that the ECC can recover the data without loss. The S.M.A.R.T. system counts the total number of errors in the entire hard drive fixed by ECC, and the total number of remappings, in an attempt to predict hard drive failure.

Future development

Because of bit-flipping errors and other issues, perpendicular recording densities may be supplanted by other magnetic recording technologies. Toshiba is promoting bit-patterned

recording (BPR), whiles Xyratex are developing heat-assisted magnetic recording (HAMR).

Capacity



PC hard disk drive capacity (in GB) over time. The vertical axis is logarithmic, so the fit line corresponds to exponential growth.

Capacity measurements

A disassembled and labeled 1997 hard drive. All major components were placed on a mirror, which created the symmetrical reflections.

Hard disk manufacturers quote disk capacity in multiples of SI-standard powers of 1000, where a *terabyte* is 1000 gigabytes and a *gigabyte* is 1000 megabytes. With file systems that report capacity in powers of 1024, available space appears somewhat less than advertised capacity. The discrepancy between the two methods of reporting sizes had serious financial consequences for at least one hard drive manufacturer when a class action suit argued the different methods effectively misled consumers.

Semiconductor memory chips are organized so that memory sizes are expressed in multiples of powers of two. Hard disks by contrast have no inherent binary size. Capacity is the product of the number of heads, number of tracks, number of sectors per track, and the size of each sector. Sector sizes are standardized for convenience at 256 or 512 and more recently 4096 bytes, which are powers of two. This can cause some confusion because operating systems may report the formatted capacity of a hard drive using binary prefix units which increment by powers of 1024. For example, Microsoft Windows reports disk capacity both in a decimal integer to 12 or more digits and in binary prefix units to three significant digits.

A one terabyte (1 TB) disk drive would be expected to hold around 1 trillion bytes (1,000,000,000,000) or 1000 GB; and indeed most 1 TB hard drives will contain slightly more than this number. However some operating system utilities would report this as around 931 GB or 953,674 MB. (The actual number for a formatted capacity will be

somewhat smaller still, depending on the file system.) Following are the several ways of reporting one Terabyte.

SI prefixes (hard drive)	equivalent	Binary prefixes (OS)	equivalent
1 TB (Terabyte)	$1 * 1000^4$ B	0.9095 TiB (Tebibyte)	$0.9095 * 1024^4$ B
1000 GB (Gigabyte)	$1000 * 1000^3$ B	931.3 GiB (Gibibyte)	$931.3 * 1024^3$ B
1,000,000 MB (Megabyte)	$1{,}000{,}000 * 1000^2$ B	953,674.3 MiB (Mebibyte)	$953{,}674.3 * 1024^2$ B
1,000,000,000 kB (Kilobyte)	1,000,000,000 * 1000 B	976,562,500 KiB (Kibibyte)	976,562,500 * 1024 B
1,000,000,000,000 B (byte)	-	1,000,000,000,000 B (byte)	-

Addressing data on large drives

The capacity of an HDD can be calculated by multiplying the number of cylinders by the number of heads by the number of sectors by the number of bytes/sector (most commonly 512). Drives with the ATA interface and a capacity of eight gigabytes or more behave as if they were structured into 16383 cylinders, 16 heads, and 63 sectors, for compatibility with older operating systems. Unlike in the 1980s, the cylinder, head, sector (C/H/S) counts reported to the CPU by a modern ATA drive are no longer actual physical parameters since the reported numbers are constrained by historic operating-system interfaces and with zone bit recording the actual number of sectors varies by zone. Disks with SCSI interface address each sector with a unique integer number; the operating system remains ignorant of their head or cylinder count.

The old C/H/S scheme has been replaced by logical block addressing. In some cases, to try to "force-fit" the C/H/S scheme to large-capacity drives, the number of heads was given as 64, although no modern drive has anywhere near 32 platters.

Not all the space on a hard drive is available for user files. The operating system file system uses some of the disk space to organize files on the disk, recording their file names and the sequence of disk areas that represent the file. Examples of data structures stored on disk to retrieve files include the MS DOS file allocation table (FAT), and UNIX inodes, as well as other operating system data structures. This file system overhead is usually less than 1% on drives larger than 100 MB.

For RAID drives, data integrity and fault-tolerance requirements also reduce the realized capacity. For example, a RAID1 drive will be about half the total capacity as a result of data mirroring. For RAID5 drives with x drives you would lose 1/x of your space to parity. RAID drives are multiple drives that appear to be one drive to the user, but provides some fault-tolerance.

A general rule of thumb to quickly convert the manufacturer's hard disk capacity to the standard Microsoft Windows formatted capacity is 0.93*capacity of HDD from manufacturer for HDDs less than a terabyte and 0.91*capacity of HDD from manufacturer for HDDs equal to or greater than 1 terabyte.

HDD Formatting

The presentation of an HDD to its host is determined by its controller. This may differ substantially from the drive's native interface particularly in mainframes or servers.

Modern HDDs, such as SAS and SATA drives, appear at their interfaces as a contiguous set of logical blocks; typically 512 bytes long but the industry is in the process of changing to 4,096 byte logical blocks.

The process of initializing these logical blocks on the physical disk platters is called *low level formatting* which is usually performed at the factory and is not normally changed in the field. *High level formatting* then writes the file system structures into selected logical blocks to make the remaining logical blocks available to the host OS and its applications.

Form factors

5¼″ full height 110 MB HDD,
2½″ (8.5 mm) 6495 MB HDD,
US/UK pennies for comparison.

Six hard drives with 8″, 5.25″, 3.5″, 2.5″, 1.8″, and 1″ disks, partially disassembled to show platters and read-write heads, with a ruler showing inches.

Mainframe and minicomputer hard disks were of widely varying dimensions, typically in free standing cabinets the size of washing machines (e.g. HP 7935 and DEC RP06 Disk Drives) or designed so that dimensions enabled placement in a 19" rack (e.g. Diablo Model 31). In 1962, IBM introduced its model 1311 disk, which used 14 inch (nominal size) platters. This became a standard size for mainframe and minicomputer drives for many years, but such large platters were never used with microprocessor-based systems.

With increasing sales of microcomputers having built in floppy-disk drives (FDDs), HDDs that would fit to the FDD mountings became desirable, and this led to the evolution of the market towards drives with certain **Form factors**, initially derived from the sizes of 8-inch, 5.25-inch, and 3.5-inch floppy disk drives. Smaller sizes than 3.5 inches have emerged as popular in the marketplace and/or been decided by various industry groups.

- **8 inch:** 9.5 in × 4.624 in × 14.25 in (241.3 mm × 117.5 mm × 362 mm)
  In 1979, Shugart Associates' SA1000 was the first form factor compatible HDD, having the same dimensions and a compatible interface to the 8″ FDD.

- **5.25 inch:** 5.75 in × 3.25 in × 8 in (146.1 mm × 82.55 mm × 203 mm)
  This smaller form factor, first used in an HDD by Seagate in 1980, was the same size as full-height 5¼-inch-diameter (130 mm) FDD, 3.25-inches high. This is twice as high as "half height"; i.e., 1.63 in (41.4 mm). Most desktop models of drives for optical 120 mm disks (DVD, CD) use the half height 5¼″ dimension, but it fell out of fashion for HDDs. The Quantum Bigfoot HDD was the last to use it in the late 1990s, with "low-profile" (≈25 mm) and "ultra-low-profile" (≈20 mm) high versions.
- **3.5 inch:** 4 in × 1 in × 5.75 in (101.6 mm × 25.4 mm × 146 mm) = 376.77344 cm³
  This smaller form factor, first used in an HDD by Rodime in 1983, was the same size as the "half height" 3½″ FDD, i.e., 1.63 inches high. Today it has been largely superseded by 1-inch high "slimline" or "low-profile" versions of this form factor which is used by most desktop HDDs.
- **2.5 inch:** 2.75 in × 0.275–0.59 in × 3.945 in (69.85 mm × 7–15 mm × 100 mm) = 48.895–104.775 cm^3
  This smaller form factor was introduced by PrairieTek in 1988; there is no corresponding FDD. It is widely used today for hard-disk drives in mobile devices (laptops, music players, etc.) and as of 2008 replacing 3.5 inch enterprise-class drives. It is also used in the Playstation 3 and Xbox 360 video game consoles. Today, the dominant height of this form factor is 9.5 mm for laptop drives (usually having two platters inside), but higher capacity drives have a height of 12.5 mm (usually having three platters). Enterprise-class drives can have a height up to 15 mm. Seagate has released a wafer-thin 7mm drive aimed at entry level laptops and high end netbooks in December 2009.
- **1.8 inch:** 54 mm × 8 mm × 71 mm = 30.672 cm³
  This form factor, originally introduced by Integral Peripherals in 1993, has evolved into the ATA-7 LIF with dimensions as stated. It was increasingly used in digital audio players and subnotebooks, but is rarely used today. An original variant exists for 2–5GB sized HDDs that fit directly into a PC card expansion slot. These became popular for their use in iPods and other HDD based MP3 players.
- **1 inch:** 42.8 mm × 5 mm × 36.4 mm
  This form factor was introduced in 1999 as IBM's Microdrive to fit inside a CF Type II slot. Samsung calls the same form factor **"1.3 inch" drive** in its product literature.
- **0.85 inch:** 24 mm × 5 mm × 32 mm
  Toshiba announced this form factor in January 2004 for use in mobile phones and similar applications, including SD/MMC slot compatible HDDs optimized for video storage on 4G handsets. Toshiba currently sells a 4 GB (MK4001MTD) and 8 GB (MK8003MTD) version and holds the Guinness World Record for the smallest hard disk drive.

3.5-inch and 2.5-inch hard disks currently dominate the market.

By 2009 all manufacturers had discontinued the development of new products for the 1.3-inch, 1-inch and 0.85-inch form factors due to falling prices of flash memory, which is slightly more stable and resistant to damage from impact and/or dropping.

The inch-based nickname of all these form factors usually do not indicate any actual product dimension (which are specified in millimeters for more recent form factors), but just roughly indicate a size relative to disk diameters, in the interest of historic continuity.

Current hard disk form factors

Form factor	Width	Height	Largest capacity	Platters (Max)
3.5″	102 mm	25.4 mm	3 TB (2010)	5
2.5″	69.9 mm	7–15 mm	1.5 TB (2010)	4
1.8″	54 mm	8 mm	320 GB (2009)	3

Obsolete hard disk form factors

Form factor	Width	Largest capacity	Platters (Max)
5.25″ FH	146 mm	47 GB (1998)	14
5.25″ HH	146 mm	19.3 GB (1998)	4
1.3″	43 mm	40 GB (2007)	1
1″ (CFII/ZIF/IDE-Flex)	42 mm	20 GB (2006)	1
0.85″	24 mm	8 GB (2004)	1

Performance characteristics

Access Time

The factors that limit the time to access the data on a hard disk drive (Access time) are mostly related to the mechanical nature of the rotating disks and moving heads. Seek time is a measure of how long it takes the head assembly to travel to the track of the disk that contains data. Latency is rotational delay incurred because the desired disk sector may not be directly under the head when data transfer is requested. These two delays are on the order of milliseconds each. The bit rate or data transfer rate once the head is in the right position creates delay which is a function of the number of blocks transferred; typically relatively small, but can be quite long with the transfer of large contiguous files.

An HDD's **Average Access Time** is its average Seek time which technically is the time to do all possible seeks divided by the number of all possible seeks, but in practice is determined by statistical methods or simply approximated as the time of a seek over one-third of the number of tracks

Defragmentation is a procedure used to minimize delay in retrieving data by moving related items to physically proximate areas on the disk. Some computer operating systems perform defragmentation automatically. Although automatic defragmentation is intended to reduce access delays, the procedure can slow response when performed while the computer is in use.

Access time can be improved by increasing rotational speed, thus reducing latency and/or by decreasing seek time. Increasing areal density increases throughput by increasing data rate and by increasing the amount of data under a set of heads, thereby potentially reducing seek activity for a given amount of data. Based on historic trends, analysts predict a future growth in HDD areal density (and therefore capacity) of about 40% per year. Access times have not kept up with throughput increases, which themselves have not kept up with growth in storage capacity.

Seek time

Average Seek time ranges from 3 ms for high-end server drives, to 15 ms for mobile drives, with the most common mobile drives at about 12 ms and the most common desktop type typically being around 9 ms. The first HDD had an average seek time of about 600 ms and by the middle 1970s HDDs were available with seek times of about 25 ms. Some early PC drives used a stepper motor to move the heads, and as a result had seek times as slow as 80–120 ms, but this was quickly improved by voice coil type actuation in the 1980s, reducing seek times to around 20 ms. Seek time has continued to improve slowly over time.

Latency

Latency is the delay for the rotation of the disk to bring the required disk sector under the read-write mechanism. It depends on rotational speed of a disk, measured in revolutions per minute (RPM). Average rotational delay is shown in the table below, based on the empirical relation that the average latency in milliseconds for such a drive is one-half the rotational period:

Spindle [rpm]	Average latency [ms]
4200	7.14
5400	5.56
7200	4.17
10000	3
15000	2

Data transfer rate

As of 2010, a typical 7200 rpm desktop hard drive has a sustained "disk-to-buffer" data transfer rate up to 1030 Mbits/sec. This rate depends on the track location, so it will be higher for data on the outer tracks (where there are more data sectors) and lower toward

the inner tracks (where there are fewer data sectors); and is generally somewhat higher for 10,000 rpm drives. A current widely used standard for the "buffer-to-computer" interface is 3.0 Gbit/s SATA, which can send about 300 megabyte/s from the buffer to the computer, and thus is still comfortably ahead of today's disk-to-buffer transfer rates. Data transfer rate (read/write) can be measured by writing a large file to disk using special file generator tools, then reading back the file. Transfer rate can be influenced by file system fragmentation and the layout of the files.

HDD data transfer rate depends upon the rotational speed of the platters and the data recording density. Because heat and vibration limit rotational speed, advancing density becomes the main method to improve sequential transfer rates. Areal density advances by increasing both the number of tracks across the disk and the number of sectors per track, the later will increase the data transfer rate (for a given RPM). Since data transfer rate performance only tracks one of the two components of areal density, its performance improves at lower rate,

Power consumption

Power consumption has become increasingly important, not only in mobile devices such as laptops but also in server and desktop markets. Increasing data center machine density has led to problems delivering sufficient power to devices (especially for spin up), and getting rid of the waste heat subsequently produced, as well as environmental and electrical cost concerns. Heat dissipation directly tied to power consumption, and as drive age, disk failure rates increase at higher drive temperatures. Similar issues exist for large companies with thousands of desktop PCs. Smaller form factor drives often use less power than larger drives. One interesting development in this area is actively controlling the seek speed so that the head arrives at its destination only just in time to read the sector, rather than arriving as quickly as possible and then having to wait for the sector to come around (i.e. the rotational latency). Many of the hard drive companies are now producing Green Drives that require much less power and cooling. Many of these Green Drives spin slower (<5,400 rpm compared to 7,200, 10,000 or 15,000 rpm) and also generate less waste heat. Power consumption can also be reduced by parking the drive heads when the disk is not in use reducing friction, adjusting spin speeds according to transfer rates, and disabling internal components when not in use.

Also in systems where there might be multiple hard disk drives, there are various ways of controlling when the hard drives spin up since the highest current is drawn at that time.

- On SCSI hard disk drives, the SCSI controller can directly control spin up and spin down of the drives.

- On Parallel ATA (aka PATA) and Serial ATA (SATA) hard disk drives, some support power-up in standby or PUIS. The hard disk drive will not spin up until the controller or system BIOS issues a specific command to do so. This limits the power draw or consumption upon power on.

- Some SATA II hard disk drives support staggered spin-up, allowing the computer to spin up the drives in sequence to reduce load on the power supply when booting.

Power management

Most hard disk drives today support some form of power management which uses a number of specific power modes that save energy by reducing performance. When implemented an HDD will change between a full power mode to one or more power saving modes as a function of drive usage. Recovery from the deepest mode, typically called Sleep, may take as long as several seconds.

Audible noise

Measured in dBA, audible noise is significant for certain applications, such as DVRs, digital audio recording and quiet computers. Low noise disks typically use fluid bearings, slower rotational speeds (usually 5,400 rpm) and reduce the seek speed under load (AAM) to reduce audible clicks and crunching sounds. Drives in smaller form factors (e.g. 2.5 inch) are often quieter than larger drives.

Shock resistance

Shock resistance is especially important for mobile devices. Some laptops now include active hard drive protection that parks the disk heads if the machine is dropped, hopefully before impact, to offer the greatest possible chance of survival in such an event. Maximum shock tolerance to date is 350 g for operating and 1000 g for non-operating.

Access and interfaces

Hard disk drives are accessed over one of a number of bus types, including parallel ATA (P-ATA, also called IDE or EIDE), Serial ATA (SATA), SCSI, Serial Attached SCSI (SAS), and Fibre Channel. Bridge circuitry is sometimes used to connect hard disk drives to buses that they cannot communicate with natively, such as IEEE 1394, USB and SCSI.

For the ST-506 interface, the data encoding scheme as written to the disk surface was also important. The first ST-506 disks used Modified Frequency Modulation (MFM) encoding, and transferred data at a rate of 5 megabits per second. Later controllers using 2,7 RLL (or just "RLL") encoding caused 50% more data to appear under the heads compared to one rotation of an MFM drive, increasing data storage and data transfer rate by 50%, to 7.5 megabits per second.

Many ST-506 interface disk drives were only specified by the manufacturer to run at the 1/3 lower MFM data transfer rate compared to RLL, while other drive models (usually more expensive versions of the same drive) were specified to run at the higher RLL data transfer rate. In some cases, a drive had sufficient margin to allow the MFM specified model to run at the denser/faster RLL data transfer rate (not recommended nor

guaranteed by manufacturers). Also, any RLL-certified drive could run on any MFM controller, but with 1/3 less data capacity and as much as 1/3 less data transfer rate compared to its RLL specifications.

Enhanced Small Disk Interface (ESDI) also supported multiple data rates (ESDI disks always used 2,7 RLL, but at 10, 15 or 20 megabits per second), but this was usually negotiated automatically by the disk drive and controller; most of the time, however, 15 or 20 megabit ESDI disk drives were not downward compatible (i.e. a 15 or 20 megabit disk drive would not run on a 10 megabit controller). ESDI disk drives typically also had jumpers to set the number of sectors per track and (in some cases) sector size.

Modern hard drives present a consistent interface to the rest of the computer, no matter what data encoding scheme is used internally. Typically a DSP in the electronics inside the hard drive takes the raw analog voltages from the read head and uses PRML and Reed–Solomon error correction to decode the sector boundaries and sector data, then sends that data out the standard interface. That DSP also watches the error rate detected by error detection and correction, and performs bad sector remapping, data collection for Self-Monitoring, Analysis, and Reporting Technology, and other internal tasks.

SCSI originally had just one signaling frequency of 5 MHz for a maximum data rate of 5 megabytes/second over 8 parallel conductors, but later this was increased dramatically. The SCSI bus speed had no bearing on the disk's internal speed because of buffering between the SCSI bus and the disk drive's internal data bus; however, many early disk drives had very small buffers, and thus had to be reformatted to a different interleave (just like ST-506 disks) when used on slow computers, such as early Commodore Amiga, IBM PC compatibles and Apple Macintoshes.

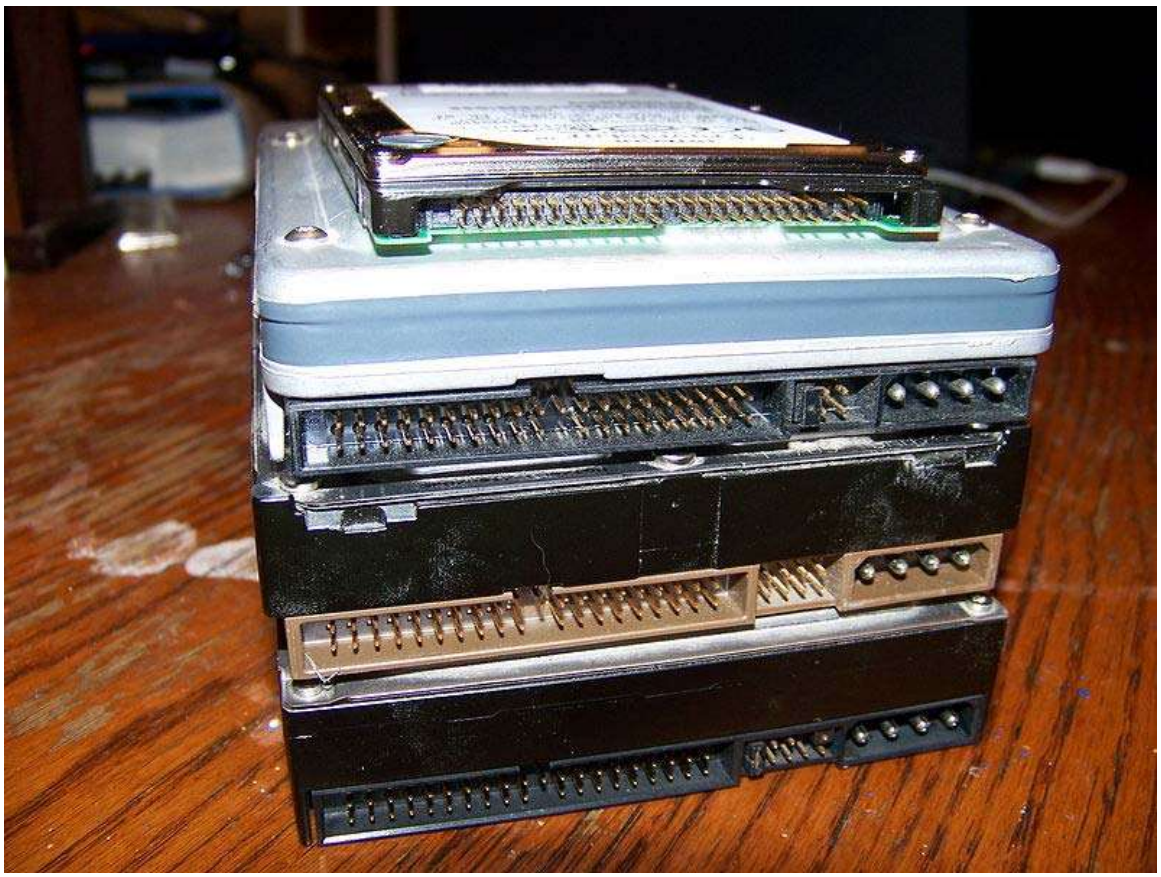
ATA disks have typically had no problems with interleave or data rate, due to their controller design, but many early models were incompatible with each other and could not run with two devices on the same physical cable in a master/slave setup. This was mostly remedied by the mid-1990s, when ATA's specification was standardized and the details began to be cleaned up, but still causes problems occasionally (especially with CD-ROM and DVD-ROM disks, and when mixing Ultra DMA and non-UDMA devices).

Serial ATA does away with master/slave setups entirely, placing each disk on its own channel (with its own set of I/O ports) instead.

FireWire/IEEE 1394 and USB(1.0/2.0) HDDs are external units containing generally ATA or SCSI disks with ports on the back allowing very simple and effective expansion and mobility. Most FireWire/IEEE 1394 models are able to daisy-chain in order to continue adding peripherals without requiring additional ports on the computer itself. USB however, is a point to point network and does not allow for daisy-chaining. USB hubs are used to increase the number of available ports and are used for devices that do not require charging since the current supplied by hubs is typically lower than what's available from the built-in USB ports.

Disk interface families used in personal computers

Notable families of disk interfaces include:



Several Parallel ATA hard disk drives

- Historical **bit serial interfaces** connect a hard disk drive (HDD) to a hard disk controller (HDC) with two cables, one for control and one for data. (Each drive also has an additional cable for power, usually connecting it directly to the power supply unit). The HDC provided significant functions such as serial/parallel conversion, data separation, and track formatting, and required matching to the drive (after formatting) in order to assure reliability. Each control cable could serve two or more drives, while a dedicated (and smaller) data cable served each drive.
    - ST506 used MFM (Modified Frequency Modulation) for the data encoding method.
    - ST412 was available in either MFM or RLL (Run Length Limited) encoding variants.
    - Enhanced Small Disk Interface (ESDI) was an industry standard interface similar to ST412 supporting higher data rates between the processor and the disk drive.

- Modern **bit serial interfaces** connect a hard disk drive to a host bus interface adapter (today typically integrated into the "south bridge") with one data/control cable. (As for historical *bit serial interfaces* above, each drive also has an additional power cable, usually direct to the power supply unit.)
    - Fibre Channel (FC), is a successor to parallel SCSI interface on enterprise market. It is a serial protocol. In disk drives usually the Fibre Channel Arbitrated Loop (FC-AL) connection topology is used. FC has much broader usage than mere disk interfaces, and it is the cornerstone of storage area networks (SANs). Recently other protocols for this field, like iSCSI and ATA over Ethernet have been developed as well. Confusingly, drives usually use *copper* twisted-pair cables for Fibre Channel, not fibre optics. The latter are traditionally reserved for larger devices, such as servers or disk array controllers.
    - Serial ATA (SATA). The SATA data cable has one data pair for differential transmission of data to the device, and one pair for differential receiving from the device, just like EIA-422. That requires that data be transmitted serially. Similar differential signaling system is used in RS485, LocalTalk, USB, Firewire, and differential SCSI.
    - Serial Attached SCSI (SAS). The SAS is a new generation serial communication protocol for devices designed to allow for much higher speed data transfers and is compatible with SATA. SAS uses a mechanically identical data and power connector to standard 3.5-inch SATA1/SATA2 HDDs, and many server-oriented SAS RAID controllers are also capable of addressing SATA hard drives. SAS uses serial communication instead of the parallel method found in traditional SCSI devices but still uses SCSI commands.

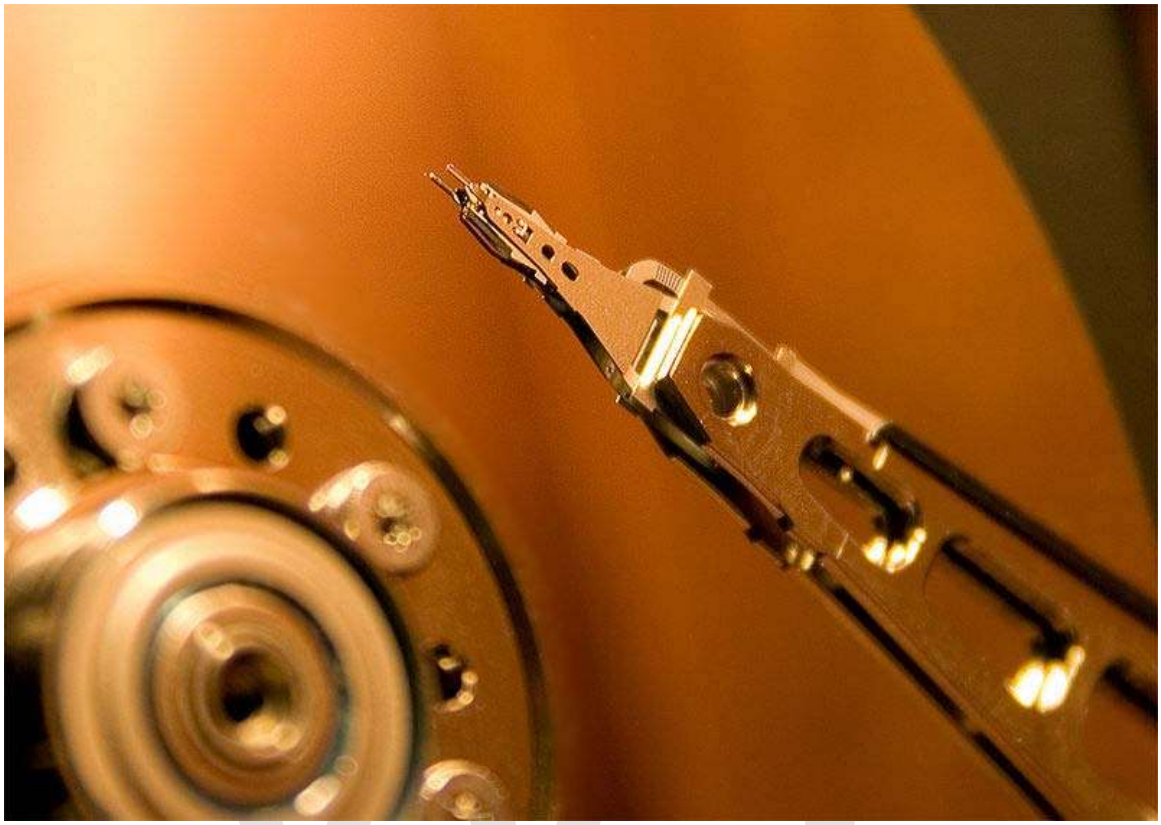
Inner view of a 1998 Seagate hard disk drive which used Parallel ATA interface

- **Word serial interfaces** connect a hard disk drive to a host bus adapter (today typically integrated into the "south bridge") with one cable for combined data/control. (As for all *bit serial interfaces* above, each drive also has an additional power cable, usually direct to the power supply unit.) The earliest versions of these interfaces typically had a 8 bit parallel data transfer to/from the drive, but 16-bit versions became much more common, and there are 32 bit versions. Modern variants have serial data transfer. The word nature of data transfer makes the design of a host bus adapter significantly simpler than that of the precursor HDD controller.
    - Integrated Drive Electronics (IDE), later renamed to ATA, with the alias P-ATA ("parallel ATA") retroactively added upon introduction of the new variant Serial ATA. The original name reflected the innovative integration of HDD controller with HDD itself, which was not found in earlier disks. Moving the HDD controller from the interface card to the disk drive helped to standardize interfaces, and to reduce the cost and complexity. The 40-pin IDE/ATA connection transfers 16 bits of data at a time on the data cable. The data cable was originally 40-conductor, but later higher speed requirements for data transfer to and from the hard drive led to an "ultra DMA" mode, known as UDMA. Progressively swifter versions of this standard ultimately added the requirement for a 80-conductor variant of the same cable, where half of the conductors provides grounding necessary for enhanced high-speed signal quality by reducing cross talk. The interface for 80-conductor only has 39 pins, the missing pin acting as a key to prevent incorrect insertion of the connector to an incompatible socket, a common cause of disk and controller damage.
    - EIDE was an unofficial update (by Western Digital) to the original IDE standard, with the key improvement being the use of direct memory access (DMA) to transfer data between the disk and the computer without the involvement of the CPU, an improvement later adopted by the official ATA standards. By directly transferring data between memory and disk, DMA eliminates the need for the CPU to copy byte per byte, therefore allowing it to process other tasks while the data transfer occurs.
    - Small Computer System Interface (SCSI), originally named SASI for Shugart Associates System Interface, was an early competitor of ESDI. SCSI disks were standard on servers, workstations, Commodore Amiga, and Apple Macintosh computers through the mid-1990s, by which time most models had been transitioned to IDE (and later, SATA) family disks. Only in 2005 did the capacity of SCSI disks fall behind IDE disk technology, though the highest-performance disks are still available in SCSI and Fibre Channel only. The range limitations of the data cable allows for external SCSI devices. Originally SCSI data cables used single ended (common mode) data transmission, but server class SCSI could use differential transmission, either low voltage differential (LVD) or high voltage differential (HVD). ("Low" and "High" voltages for differential SCSI are relative to SCSI standards and do not meet the meaning of low voltage and high voltage as used in general electrical engineering contexts,

as apply e.g. to statutory electrical codes; both LVD and HVD use low voltage signals (3.3 V and 5 V respectively) in general terminology.)

Acronym or abbreviation	Meaning	Description
SASI	Shugart Associates System Interface	Historical predecessor to SCSI.
SCSI	Small Computer System Interface	Bus oriented that handles concurrent operations.
SAS	Serial Attached SCSI	Improvement of SCSI, uses serial communication instead of parallel.
ST-506	Seagate Technology	Historical Seagate interface.
ST-412	Seagate Technology	Historical Seagate interface (minor improvement over ST-506).
ESDI	Enhanced Small Disk Interface	Historical; backwards compatible with ST-412/506, but faster and more integrated.
ATA (PATA)	Advanced Technology Attachment	Successor to ST-412/506/ESDI by integrating the disk controller completely onto the device. Incapable of concurrent operations.
SATA	Serial ATA	Modification of ATA, uses serial communication instead of parallel.

Integrity



An IBM HDD head resting on a disk platter. Since the drive is not in operation, the head is simply pressed against the disk by the suspension.

Close-up of a hard disk head resting on a disk platter. A reflection of the head and its suspension is visible on the mirror-like disk.

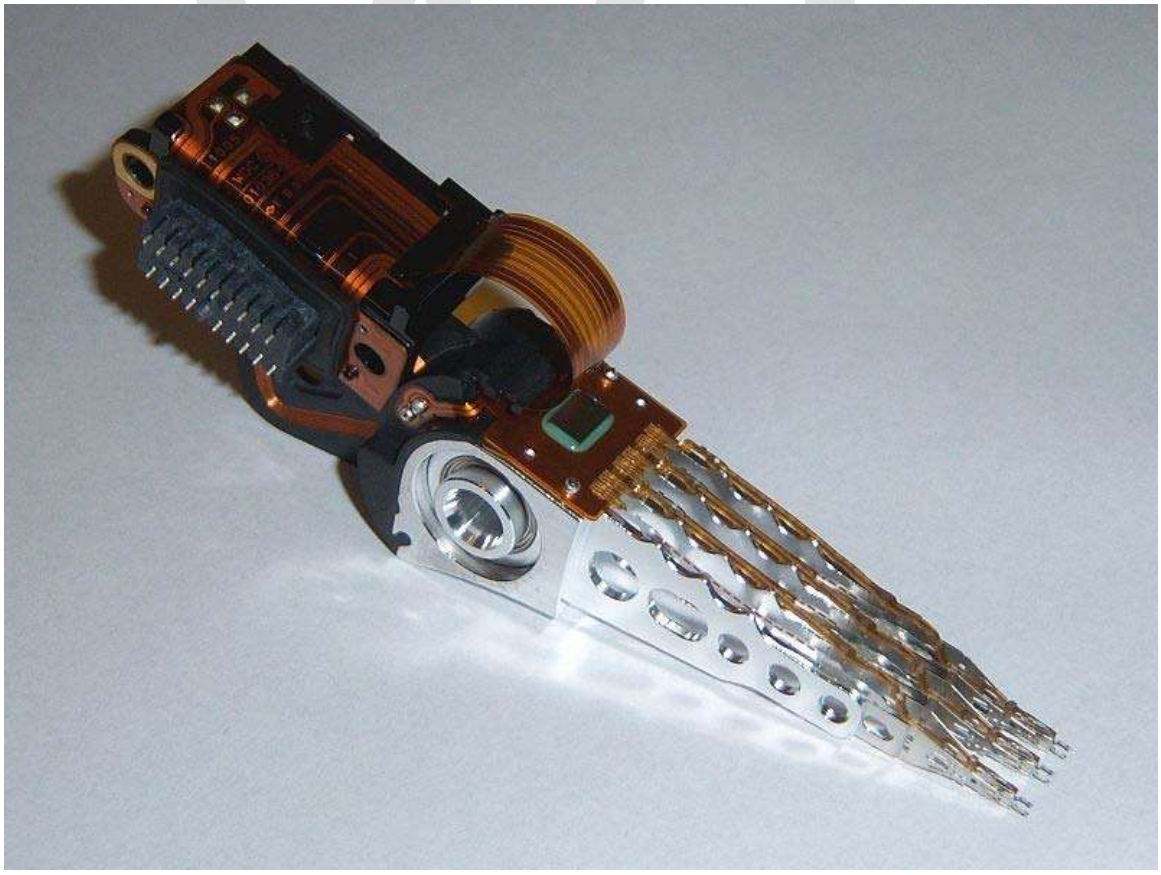
Due to the extremely close spacing between the heads and the disk surface, hard disk drives are vulnerable to being damaged by a head crash—a failure of the disk in which the head scrapes across the platter surface, often grinding away the thin magnetic film and causing data loss. Head crashes can be caused by electronic failure, a sudden power failure, physical shock, contamination of the drive's internal enclosure, wear and tear, corrosion, or poorly manufactured platters and heads.

The HDD's spindle system relies on air pressure inside the disk enclosure to support the heads at their proper *flying height* while the disk rotates. Hard disk drives require a certain range of air pressures in order to operate properly. The connection to the external environment and pressure occurs through a small hole in the enclosure (about 0.5 mm in breadth), usually with a filter on the inside (the *breather filter*). If the air pressure is too low, then there is not enough lift for the flying head, so the head gets too close to the disk, and there is a risk of head crashes and data loss. Specially manufactured sealed and

pressurized disks are needed for reliable high-altitude operation, above about 3,000 m (10,000 feet). Modern disks include temperature sensors and adjust their operation to the operating environment. Breather holes can be seen on all disk drives—they usually have a sticker next to them, warning the user not to cover the holes. The air inside the operating drive is constantly moving too, being swept in motion by friction with the spinning platters. This air passes through an internal recirculation (or "recirc") filter to remove any leftover contaminants from manufacture, any particles or chemicals that may have somehow entered the enclosure, and any particles or outgassing generated internally in normal operation. Very high humidity for extended periods can corrode the heads and platters.

For giant magnetoresistive (GMR) heads in particular, a minor head crash from contamination (that does not remove the magnetic surface of the disk) still results in the head temporarily overheating, due to friction with the disk surface, and can render the data unreadable for a short period until the head temperature stabilizes (so called "thermal asperity", a problem which can partially be dealt with by proper electronic filtering of the read signal).

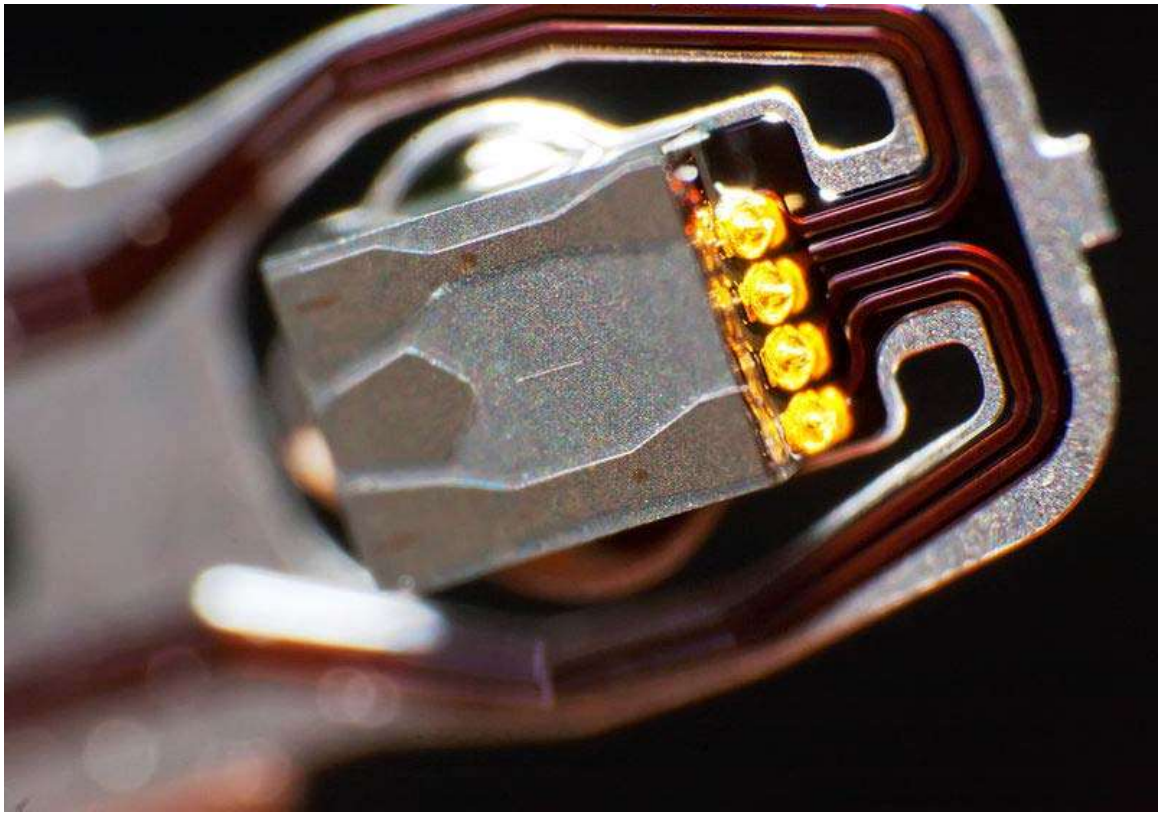
Actuation of moving arm



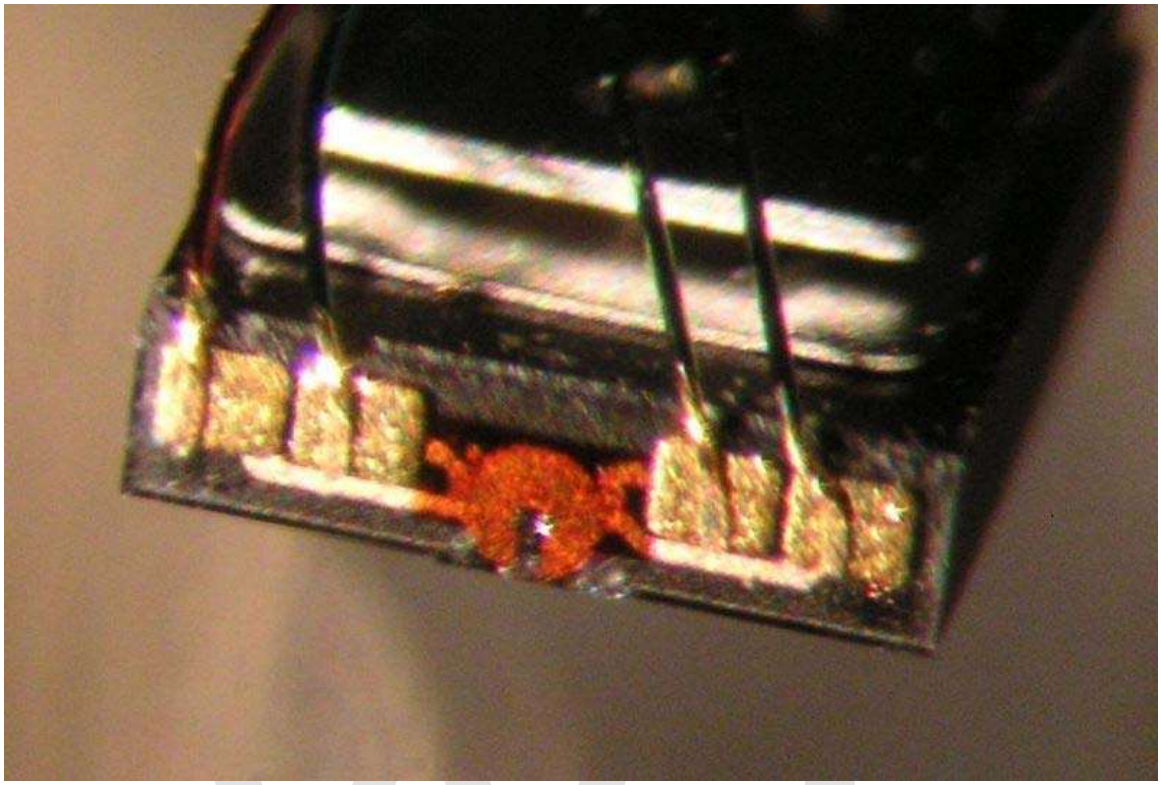
Head stack with actuator coil on the left side (partly hidden by the controller interface) and read/write heads on the right side

The hard drive's electronics control the movement of the actuator and the rotation of the disk, and perform reads and writes on demand from the disk controller. Feedback of the drive electronics is accomplished by means of special segments of the disk dedicated to servo feedback. These are either complete concentric circles (in the case of dedicated servo technology), or segments interspersed with real data (in the case of embedded servo technology). The servo feedback optimizes the signal to noise ratio of the GMR sensors by adjusting the voice-coil of the actuated arm. The spinning of the disk also uses a servo motor. Modern disk firmware is capable of scheduling reads and writes efficiently on the platter surfaces and remapping sectors of the media which have failed.

Landing zones and load/unload technology



A read/write head from a circa-1998 Fujitsu 3.5-inch hard disk. The area pictured is approximately 2.0 mm x 3.0mm.



Microphotograph of an older generation hard disk head and slider (1990s). The size of the front face (which is the "trailing face" of the slider) is about 0.3 mm × 1.0 mm. It is the location of the actual head (magnetic sensors). The non-visible bottom face of the slider is about 1.0 mm × 1.25 mm (so-called "nano" size) and faces the platter. It contains the lithographically micro-machined air bearing surface (ABS) that allows the slider to fly in a highly controlled fashion. One functional part of the head is the round, orange structure visible in the middle—the lithographically defined copper coil of the write transducer. Also note the electric connections by wires bonded to gold-plated pads.

Modern HDDs prevent power interruptions or other malfunctions from landing its heads in the data zone by **parking** the heads either in a **landing zone** or by unloading (i.e., **load/unload**) the heads. Some early PC HDDs did not park the heads automatically and they would land on data. In some other early units the user manually parked the heads by running a program to park the HDD's heads.

A **landing zone** is an area of the platter usually near its inner diameter (ID), where no data are stored. This area is called the Contact Start/Stop (CSS) zone. Disks are designed such that either a spring or, more recently, rotational inertia in the platters is used to park the heads in the case of unexpected power loss. In this case, the spindle motor temporarily acts as a generator, providing power to the actuator.

Spring tension from the head mounting constantly pushes the heads towards the platter. While the disk is spinning, the heads are supported by an air bearing and experience no physical contact or wear. In CSS drives the sliders carrying the head sensors (often also

just called *heads*) are designed to survive a number of landings and takeoffs from the media surface, though wear and tear on these microscopic components eventually takes its toll. Most manufacturers design the sliders to survive 50,000 contact cycles before the chance of damage on startup rises above 50%. However, the decay rate is not linear: when a disk is younger and has had fewer start-stop cycles, it has a better chance of surviving the next startup than an older, higher-mileage disk (as the head literally drags along the disk's surface until the air bearing is established). For example, the Seagate Barracuda 7200.10 series of desktop hard disks are rated to 50,000 start-stop cycles, in other words no failures attributed to the head-platter interface were seen before at least 50,000 start-stop cycles during testing.

Around 1995 IBM pioneered a technology where a landing zone on the disk is made by a precision laser process (*Laser Zone Texture* = LZT) producing an array of smooth nanometer-scale "bumps" in a landing zone, thus vastly improving stiction and wear performance. This technology is still largely in use today (2008), predominantly in desktop and enterprise (3.5 inch) drives. In general, CSS technology can be prone to increased stiction (the tendency for the heads to stick to the platter surface), e.g. as a consequence of increased humidity. Excessive stiction can cause physical damage to the platter and slider or spindle motor.

Load/Unload technology relies on the heads being lifted off the platters into a safe location, thus eliminating the risks of wear and stiction altogether. The first HDD RAMAC and most early disk drives used complex mechanisms to load and unload the heads. Modern HDDs use ramp loading, first introduced by Memorex in 1967, to load/unload onto plastic "ramps" near the outer disk edge.

All HDDs today still use one of these two technologies listed above. Each has a list of advantages and drawbacks in terms of loss of storage area on the disk, relative difficulty of mechanical tolerance control, non-operating shock robustness, cost of implementation, etc.

Addressing shock robustness, IBM also created a technology for their ThinkPad line of laptop computers called the Active Protection System. When a sudden, sharp movement is detected by the built-in accelerometer in the Thinkpad, internal hard disk heads automatically unload themselves to reduce the risk of any potential data loss or scratch defects. Apple later also utilized this technology in their PowerBook, iBook, MacBook Pro, and MacBook line, known as the Sudden Motion Sensor. Sony, HP with their HP 3D DriveGuard and Toshiba have released similar technology in their notebook computers.

This accelerometer-based shock sensor has also been used for building cheap earthquake sensor networks.

Disk failures and their metrics

Most major hard disk and motherboard vendors now support S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology), which measures drive characteristics

such as operating temperature, spin-up time, data error rates, etc. Certain trends and sudden changes in these parameters are thought to be associated with increased likelihood of drive failure and data loss.

However, not all failures are predictable. Normal use eventually can lead to a breakdown in the inherently fragile device, which makes it essential for the user to periodically back up the data onto a separate storage device. Failure to do so can lead to the loss of data. While it may sometimes be possible to recover lost information, it is normally an extremely costly procedure, and it is not possible to guarantee success. A 2007 study published by Google suggested very little correlation between failure rates and either high temperature or activity level; however, the correlation between manufacturer/model and failure rate was relatively strong. Statistics in this matter is kept highly secret by most entities. Google did not publish the manufacturer's names along with their respective failure rates, though they have since revealed that they use Hitachi Deskstar drives in some of their servers. While several S.M.A.R.T. parameters have an impact on failure probability, a large fraction of failed drives do not produce predictive S.M.A.R.T. parameters. S.M.A.R.T. parameters alone may not be useful for predicting individual drive failures.

A common misconception is that a colder hard drive will last longer than a hotter hard drive. The Google study seems to imply the reverse—"lower temperatures are associated with higher failure rates". Hard drives with S.M.A.R.T.-reported average temperatures below 27 °C (80.6 °F) had higher failure rates than hard drives with the highest reported average temperature of 50 °C (122 °F), failure rates at least twice as high as the optimum S.M.A.R.T.-reported temperature range of 36 °C (96.8 °F) to 47 °C (116.6 °F).

SCSI, SAS, and FC drives are typically more expensive and are traditionally used in servers and disk arrays, whereas inexpensive ATA and SATA drives evolved in the home computer market and were perceived to be less reliable. This distinction is now becoming blurred.

The mean time between failures (MTBF) of SATA drives is usually about 600,000 hours (some drives such as Western Digital Raptor have rated 1.4 million hours MTBF), while SCSI drives are rated for upwards of 1.5 million hours. However, independent research indicates that MTBF is not a reliable estimate of a drive's longevity. MTBF is conducted in laboratory environments in test chambers and is an important metric to determine the quality of a disk drive before it enters high volume production. Once the drive product is in production, the more valid metric is annualized failure rate (AFR). AFR is the percentage of real-world drive failures after shipping.

SAS drives are comparable to SCSI drives, with high MTBF and high reliability.

Enterprise S-ATA drives designed and produced for enterprise markets, unlike standard S-ATA drives, have reliability comparable to other enterprise class drives.

Typically enterprise drives (all enterprise drives, including SCSI, SAS, enterprise SATA, and FC) experience between 0.70%–0.78% annual failure rates from the total installed drives.

Eventually all mechanical hard disk drives fail, so to mitigate loss of data, some form of redundancy is needed, such as RAID or a regular backup system.

External removable drives

External removable hard disk drives connect to the computer using a USB cable or other means. External drives are used for:

- Backup of files and information
- Data recovery
- Disk cloning
- Running virtual machines
- Scratch disk for video editing applications and video recording.

A 6 GB Seagate Pocket hard drive with USB cable extended next to a 2 GB CompactFlash card.

Larger models often include full-sized 3.5" PATA or SATA desktop hard drives. Features such as biometric security or multiple interfaces generally increase cost.

Market segments

- As of July 2010, the highest capacity consumer HDDs are 3 TB.
- **"Desktop HDDs"** typically store between 120 GB and 2 TB and rotate at 5,400 to 10,000 rpm, and have a media transfer rate of 0.5 Gbit/s or higher. (1 GB = 10^9 bytes; 1 Gbit/s = 10^9 bit/s)
- **Enterprise HDDs** are typically used with multiple-user computers running enterprise software. Examples are
    - transaction processing databases;
    - internet infrastructure (email, webserver, e-commerce);
    - scientific computing software;
    - nearline storage management software.

    The fastest enterprise HDDs spin at 10,000 or 15,000 rpm, and can achieve sequential media transfer speeds above 1.6 Gbit/s. and a sustained transfer rate up to 1 Gbit/s. Drives running at 10,000 or 15,000 rpm use smaller platters to mitigate increased power requirements (as they have less air drag) and therefore generally have lower capacity than the highest capacity desktop drives. Enterprise drives commonly operate continuously ("24/7") in demanding environments while delivering the highest possible performance without sacrificing reliability. Maximum capacity is not the primary goal, and as a result the drives are often offered in capacities that are relatively low in relation to their cost.

- **Mobile HDDs** or laptop HDDs, smaller than their desktop and enterprise counterparts, tend to be slower and have lower capacity. A typical mobile HDD spins at either 4200 rpm, 5200 rpm, 5400 rpm, or 7200 rpm, with 5400 rpm being the most prominent. 7200 rpm drives tend to be more expensive and have smaller capacities, while 4200 rpm models usually have very high storage capacities. Because of smaller platter(s), mobile HDDs generally have lower capacity than their greater desktop counterparts.

The exponential increases in disk space and data access speeds of HDDs have enabled the commercial viability of consumer products that require large storage capacities, such as digital video recorders and digital audio players. In addition, the availability of vast amounts of cheap storage has made viable a variety of web-based services with extraordinary capacity requirements, such as free-of-charge web search, web archiving, and video sharing (Google, Internet Archive, YouTube, etc.).

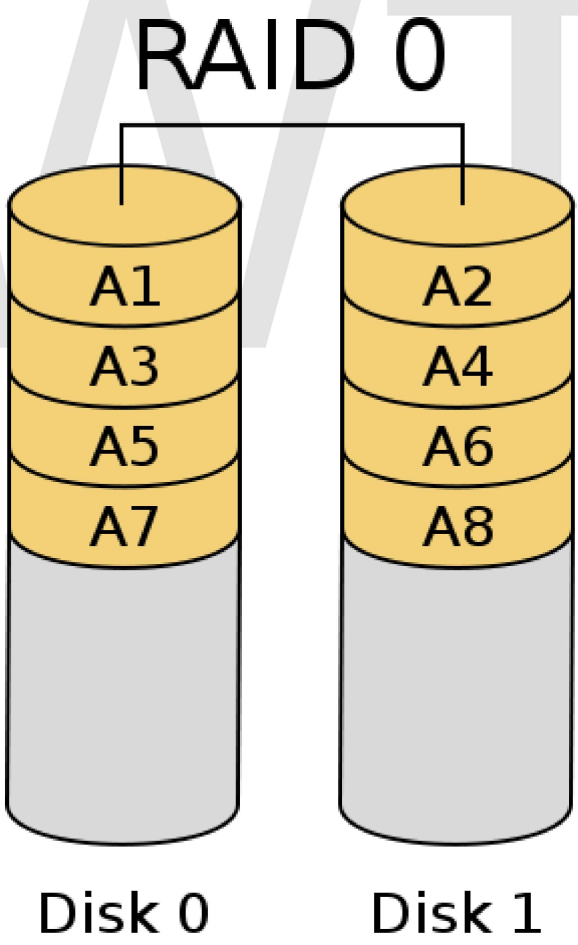
Sales

Worldwide revenue from shipments of HDDs is expected to reach $27.7 billion in 2010, up 18.4% from $23.4 billion in 2009 corresponding to a 2010 unit shipment forecast of 674.6 million compared to 549.5 million units in 2009.

Icons

Hard drives are traditionally symbolized as either a stylized stack of platters (in orthographic projection) or, more abstractly, as a cylinder. This is particularly found in schematic diagrams or on indicator lights, as on laptops, to indicate hard drive access. In most modern operating systems, hard drives are instead represented by an illustration or photograph of a hard drive enclosure. These are illustrated below.

Today, hard drives are symbolized by a picture of the enclosure.



Schematically, hard drives may be represented by cylinders or stacks of platters, as in this RAID diagram.

The cylinder schematic derives from hard drives internally being a stack of platters, as in these 1970s vintage disk pack (cover removed).

Manufacturers

More than 200 companies have manufactured hard disk drives. Today most drives are made by Seagate, Western Digital, Hitachi, Samsung, and Toshiba (though Toshiba does not manufacture 3.5 inch drives).

Chapter 9

Solid-State Drive

An SSD in standard 2.5-inch (64 mm) form-factor

DDR SDRAM based SSD

PCI attached IO Accelerator SSD

PCI-E, DRAM, and NAND based SSD

A **solid-state drive** (SSD) is a data storage device that uses solid-state memory to store persistent data with the intention of providing access in the same manner of a traditional block i/o hard disk drive. SSDs are distinguished from traditional hard disk drives (HDDs), which are electromechanical devices containing spinning disks and movable read/write heads. SSDs, in contrast, use microchips which retain data in non-volatile memory chips and contain no moving parts. Compared to electromechanical HDDs, SSDs are typically less susceptible to physical shock, quieter, and have lower access time

and latency. SSDs use the same interface as hard disk drives, thus easily replacing them in most applications.

As of 2010, most SSDs use NAND-based flash memory, which retains memory even without power. SSDs using volatile random-access memory (RAM) also exist for situations which require even faster access, but do not necessarily need data persistence after power loss, or use external power or batteries to maintain the data after power is removed.

A hybrid drive combines the features of an HDD and an SSD in one unit, containing a large HDD, with a smaller SSD cache to improve performance of frequently accessed files. These can offer near-SSD performance in most applications (such as system startup and loading applications) at a lower price than an SSD. These are not suitable for data-intensive work, nor do they offer the other advantages of SSDs.

Development and history

Early SSDs using RAM and similar technology

The origins of SSDs came from the 1950s using two similar technologies, magnetic core memory and card capacitor read-only store (CCROS). These auxiliary memory units, as they were called at the time, emerged during the era of vacuum tube computers. But with the introduction of cheaper drum storage units, their use was discontinued. Later, in the 1970s and 1980s, SSDs were implemented in semiconductor memory for early supercomputers of IBM, Amdahl and Cray; however, the prohibitively high price of the built-to-order SSDs made them quite seldom used.

In 1978, Texas Memory Systems introduced a 16 kilobyte (KB) RAM solid-state drive to be used by oil companies for seismic data acquisition. The following year, StorageTek developed the first modern type of solid-state drive. The Sharp PC-5000, introduced in 1983, used 128 kilobyte solid-state storage cartridges, containing bubble memory. In 1984 Tall grass Company had a tape back up unit of 40 MB with a solid state 20 MB unit built in. The 20 MB unit could be used instead of a hard drive. In September 1986, Santa Clara Systems introduced BatRam, 4 megabyte (MB) mass storage system expandable to 20 MB using 4 MB memory modules. The package included a rechargeable battery to preserve the memory chip contents when the array was not powered. 1987 saw the entry of EMC Corporation into the SSD market, with drives introduced for the mini-computer market. However, EMC exited the business soon after.

Flash-based SSDs

In 1994, STEC,_Inc. bought Cirrus Logic's flash controller operation, allowing the company to enter the flash memory business for consumer electronic devices.

In 1995, M-Systems introduced flash-based solid-state drives. They had the advantage of not requiring batteries to maintain the data in the memory (required by the prior volatile

memory systems), but were not as fast as the DRAM-based solutions. Since then, SSDs have been used successfully as HDD replacements by the military and aerospace industries, as well as for other mission-critical applications. These applications require the exceptional mean time between failures (MTBF) rates that solid-state drives achieve, by virtue of their ability to withstand extreme shock, vibration and temperature ranges.

BiTMICRO made a number of introductions and announcements in 1999 around flash-based SSDs including an 18 gigabyte 3.5 in SSD. Fusion-io announced a PCIe-based SSD with 100,000 input/output operations per second (IOPS) of performance in a single card with capacities up to 320 gigabytes in 2007. At Cebit 2009, OCZ demonstrated a 1 terabyte (TB) flash SSD using a PCI Express ×8 interface. It achieves a maximum write speed of 654 megabytes per second (MB/s) and maximum read speed of 712 MB/s. In December 2009, Micron Technology announced the world's first SSD using a 6 gigabits per second (Gbit/s) or 768 (MB/s) SATA interface.

Enterprise flash drives

Enterprise flash drives (EFDs) are designed for applications requiring high I/O performance (IOPS), reliability, and energy efficiency. In most cases an EFD is an SSD with a higher set of specifications compared to SSDs which would typically be used in notebook computers. The term was first used by EMC in January 2008, to help them identify SSD manufacturers who would provide products meeting these higher standards. There are no standards bodies who control the definition of EFDs, so any SSD manufacturer may claim to produce EFDs when they may not actually meet the requirements. Likewise there may be other SSD manufacturers that meet the EFD requirements without being called EFDs.

Secure digital card drives

Secure digital cards are particularly simple and inexpensive SSDs though an equally inexpensive USB flash memory adapter may be needed to attach it if the computer does not have a built-in SD flash memory card reader(/writer). These cards can boot live SD operating systems.

Architecture and function

The key components of an SSD are the controller and the memory to store the data. The primary memory component in an SSD had been DRAM volatile memory since they were first developed, but since 2009 it is more commonly NAND flash non-volatile memory. Other components play a less significant role in the operation of the SSD and vary between manufacturers.

Controller

Every SSD includes a controller that incorporates the electronics that bridge the NAND memory components to the host computer. The controller is an embedded processor that

executes firmware-level code and is one of the most important factors of SSD performance. Some of the functions performed by the controller include:

- Error correction (ECC)
- Wear leveling
- Bad block mapping
- Read scrubbing and read disturb management
- Read and write caching
- Garbage collection
- Encryption

The performance of the SSD can scale with the number of parallel NAND flash chips used in the device. A single NAND chip is relatively slow, due to narrow (8/16 bit) asynchronous IO interface, and additional high latency of basic IO operations (typical for SLC NAND, ~25 µs to fetch a 4K page from the array to the IO buffer on a read, ~250 µs to commit a 4K page from the IO buffer to the array on a write, ~2 ms to erase a 256 kiB block). When multiple NAND devices operate in parallel inside an SSD, the bandwidth scales, and the high latencies can be hidden, as long as enough outstanding operations are pending and the load is evenly distributed between devices. Micron and Intel initially made faster SSDs by implementing data striping (similar to RAID 0) and interleaving in their architecture. This enabled the creation of ultra-fast SSDs with 250 MB/s effective read/write speeds.

Memory

Flash memory-based

Most SSD manufacturers use non-volatile NAND flash memory in the construction of their SSDs due to the lower cost compared to DRAM and the ability to retain the data without a constant power supply, ensuring data persistence through sudden power outages. Flash memory SSDs are slower than DRAM solutions, and some early designs were even slower than HDDs after continued use. This problem was resolved by controllers that came out in 2009 and later.

Flash memory-based solutions are typically packaged in standard disk drive form factors (1.8-, 2.5-, and 3.5-inch), or smaller unique and compact layouts due to the compact memory.

Single-level cell (SLC) vs multi-level cell (MLC)

Lower priced drives usually use multi-level cell (MLC) flash memory, which is slower and less reliable than single-level cell (SLC) flash memory. This can be mitigated or even reversed by the internal design structure of the SSD, such as interleaving, changes to writing algorithms, and higher over-provisioning (more excess capacity) with which the wear-leveling algorithms can work.

DRAM-based

SSDs based on volatile memory such as DRAM are characterized by ultrafast data access, generally less than 10 microseconds, and are used primarily to accelerate applications that would otherwise be held back by the latency of flash SSDs or traditional HDDs. DRAM-based SSDs usually incorporate either an internal battery or an external AC/DC adapter and backup storage systems to ensure data persistence while no power is being supplied to the drive from external sources. If power is lost, the battery provides power while all information is copied from random access memory (RAM) to back-up storage. When the power is restored, the information is copied back to the RAM from the back-up storage, and the SSD resumes normal operation (similar to the hibernate function used in modern operating systems).

SSDs of this type are usually fitted with DRAM modules of the same type used in regular PCs and servers, which can be swapped out and replaced by larger modules.

A *remote, indirect memory-access disk (RIndMA Disk)* uses a secondary computer with a fast network or (direct) Infiniband connection to act like a RAM-based SSD, but the new faster flash memory based SSDs already available in 2009 are making this option not as cost effective.

Cache or buffer

A flash-based SSD typically uses a small amount of DRAM as a cache, similar to the cache in Hard disk drives. A directory of block placement and wear leveling data is also kept in the cache while the drive is operating. Data are not permanently stored in the cache. One SSD controller manufacturer, SandForce, does not use an external DRAM cache on their designs, but still achieve very high performance. Eliminating the external DRAM enables a smaller footprint for the other flash memory components in order to build even smaller SSDs.

Battery or super capacitor

Another component in higher performing SSDs is a capacitor or some form of battery. These are necessary to maintain data integrity such that the data in the cache can be flushed to the drive when power is dropped; some may even hold power long enough to maintain data in the cache until power is resumed. In the case of MLC flash memory, a problem called *lower page corruption* can occur when MLC flash memory loses power while programming an upper page. The result is data written previously and presumed safe can be corrupted if the memory is not supported by a super capacitor in the event of a sudden power loss. This problem does not exist with SLC flash memory.

Host interface

The host interface is not specifically a component of the SSD, but it is a key part of the drive. The interface is usually incorporated into the controller discussed above. The interface is generally one of the interfaces found in HDDs. They include:

- Serial ATA
- Serial attached SCSI (generally found on servers)
- PCI Express
- Fibre Channel (almost exclusively found on servers)
- USB
- Parallel ATA (IDE) interface (mostly replaced by SATA)
- (Parallel) SCSI (generally found on servers; mostly replaced by SAS)

Form factor

The size and shape of any device is largely driven by the size and shape of the components used to make that device. Traditional HDDs and optical drives are designed around the rotating platter or optical disc along with the spindle motor inside. If an SSD is made up of various interconnected integrated circuits (ICs) and an interface connector, then its shape could be virtually anything imaginable because it is no longer limited to the shape of rotating media drives. Some solid state storage solutions come in a larger chassis that may even be a rack-mount form factor with numerous SSDs inside. They would all connect to a common bus inside the chassis and connect outside the box with a single connector.

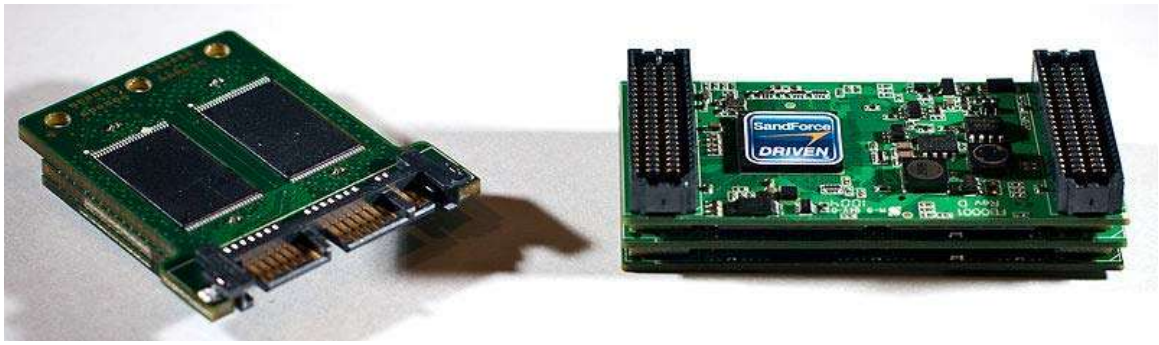
Standard HDD form factors

The benefit of using a current HDD form factor would be to take advantage of the extensive infrastructure already in place to mount and connect the drives to the host system. These traditional form factors are known by the size of the rotating media, e.g., 5.25", 3.5", 2.5", 1.8", not by the dimensions of the drive casing.

Box form factors

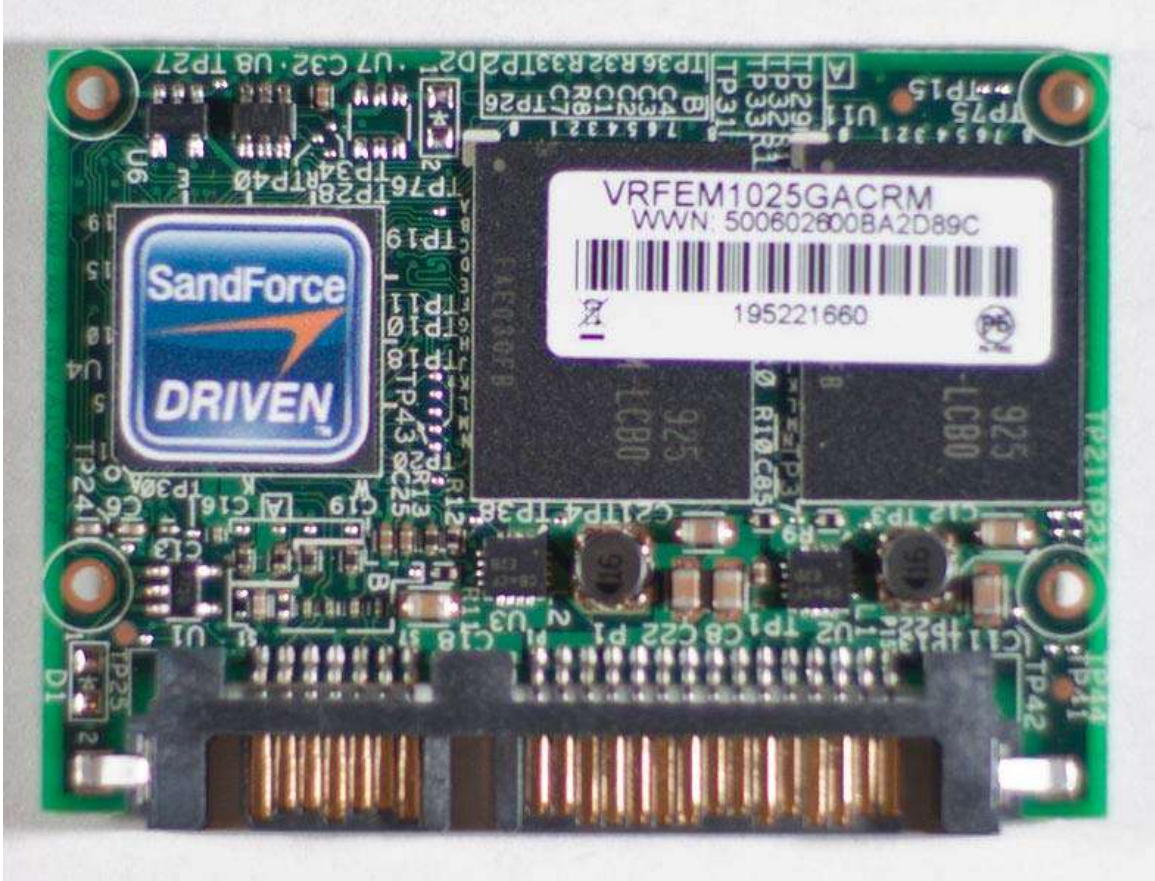
Many of the DRAM-based solutions use a box that is often designed to fit in a rack-mount system. The number of DRAM components required to get sufficient capacity to store the data along with the backup power supplies requires a larger space than traditional HDD form factors.

Bare-board form factors

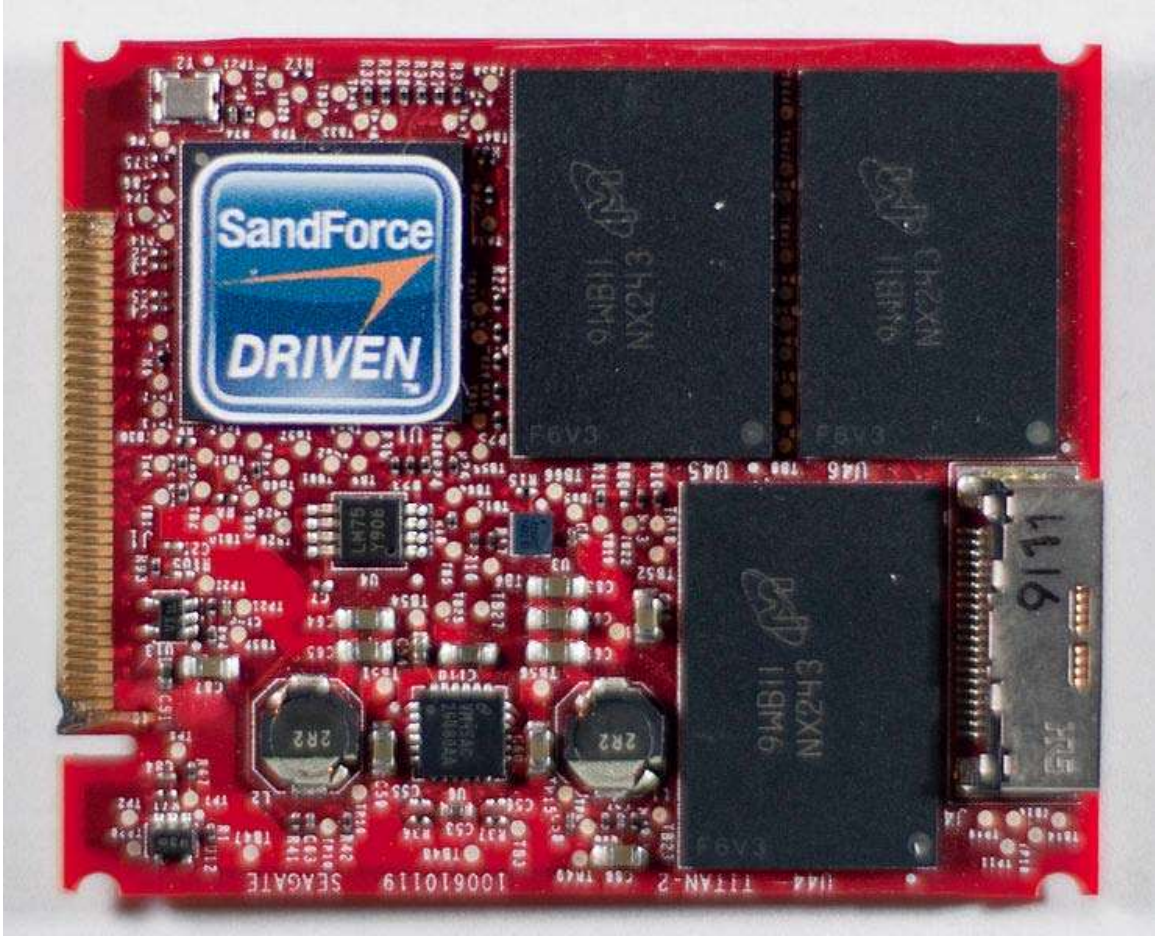


Viking Modular SATA Cube and AMP SATA Bridge multi-layer SSDs

Viking Modular SATADIMM based SSD



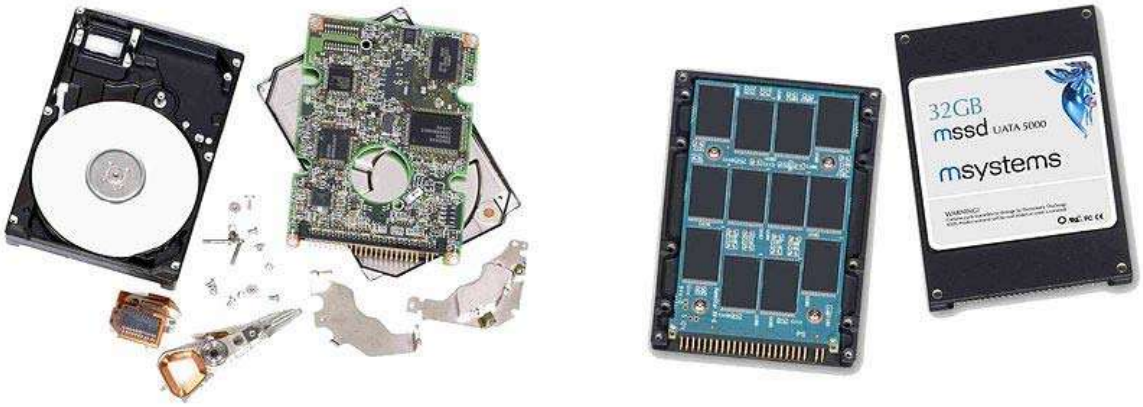
MO-297 SSD form factor



Custom connector SATA SSD

Form factors which were more common to memory modules are now being used by SSDs to take advantage of their flexibility in laying out the components. Some of these include PCIe, mini PCIe, mini-DIMM, MO-297, and many more. At least one manufacturer, InnoDisk, is producing a drive that sits directly on the SATA connector on the motherboard without any other support or mechanical mount. The SATADIMM from Viking Modular uses an empty DDR3 DIMM slot on the motherboard to provide power to the drive with a separate SATA connector to provide the data connection back to the computer. The result is an easy to install SSD with a capacity equal to drives that typically take a full 2.5 in expansion slot. Some SSDs are based on the PCIe form factor and connect both the data interface and power through the PCIe connector to the host. These drives can use either direct PCIe flash controllers or a PCIe-to-SATA bridge device which then connects to SATA flash controller(s).

Comparison of SSD with hard disk drives



The disassembled components of a hard disk drive (left) and of the PCB and components of a solid-state drive (right)

Making a comparison between SSDs and ordinary (spinning) HDDs is difficult. Traditional HDD benchmarks are focused on finding the performance aspects where they are weak, such as rotational latency time and seek time. As SSDs do not spin, or seek, they may show huge superiority in such tests. However, SSDs have challenges with mixed reads and writes, and their performance may degrade over time. SSD testing must start from the (in use) full disk, as the new and empty (fresh out of the box) disk may have much better write performance than it would show after only weeks of use.

Comparisons reflect typical characteristics, and may not hold for a specific device.

Attribute or characteristic	Solid-state drive	Hard disk drive
Spin-up time	Instantaneous.	May take several seconds. With a large number of drives, spin-up may need to be staggered to limit total power drawn.
Random access time	About 0.1 ms - many times faster than HDDs because data is accessed directly from the flash memory	Ranges from 5–10 ms due to the need to move the heads and wait for the data to rotate under the read/write head
Read latency time	Generally low because the data can be read directly from any location; In applications where hard disk seeks are the limiting factor, this results in faster boot and application launch times.	Generally high since the mechanical components require additional time to get aligned
Consistent read performance	Read performance does not change based on where data is stored on an SSD	If data is written in a fragmented way, reading

		back the data will have varying response times
Defragmentation	SSDs do not benefit from defragmentation because there is little benefit to reading data sequentially and any defragmentation process adds additional writes on the NAND flash that already have a limited cycle life.	HDDs may require defragmentation after continued operations or erasing and writing data, especially involving large files or where the disk space becomes low.
Acoustic levels	SSDs have no moving parts and make no sound	HDDs have moving parts (heads, spindle motor) and have varying levels of sound depending upon model
Mechanical reliability	A lack of moving parts virtually eliminates mechanical breakdowns	HDDs have many moving parts that are all subject to failure over time
Susceptibility to environmental factors	No flying heads or rotating platters to fail as a result of shock, altitude, or vibration	The flying heads and rotating platters are generally susceptible to shock, altitude, and vibration
Magnetic susceptibility	No impact on flash memory	Magnets or magnetic surges can alter data on the media
Weight and size	The weight of flash memory and the circuit board material are very light compared to HDDs	Higher performing HDDs require heavier components than laptop HDDs (which are light, but not as light as SSDs)
Parallel operation	Some flash controllers can have multiple flash chips reading and writing different data simultaneously	HDDs have multiple heads (one per platter) but they are connected, and share one positioning motor.
Write longevity	Solid state drives that use flash memory have a limited number of writes over the life of the drive. SSDs based on DRAM do not have a limited number of writes.	Magnetic media do not have a limited number of writes.
Software encryption limitations	NAND flash memory cannot be overwritten, but has to be rewritten to previously erased blocks. If a software encryption program encrypts data already on the SSD, the overwritten data is still unsecured, unencrypted, and accessible (drive-based hardware encryption does	HDDs can overwrite data directly on the drive in any particular sector.

	not have this problem). Also data cannot be securely erased by overwriting the original file without special "Secure Erase" procedures built into the drive.	
Cost	As of February 2011, NAND flash SSDs cost about (US)$1.20–2.00 per GB	As of February 2011, HDDs cost about (US)$0.05/GB for 3.5 in and $0.10/GB for 2.5 in drives
Storage capacity	As of October 2010, SSDs come in different sizes up to 2TB but are typically 512GB or less	As of October 2010, HDDs are typically 2-3TB or less
Read/write performance symmetry	Less expensive SSDs typically have write speeds significantly lower than their read speeds. Higher performing SSDs have a balanced read and write speed.	HDDs generally have slightly lower write speeds than their read speeds.
Free block availability and TRIM	SSD write performance is significantly impacted by the availability of free, programmable blocks. Previously written data blocks that are no longer in use can be reclaimed by TRIM; however, even with TRIM, fewer free, programmable blocks translates into reduced performance.	HDDs are not affected by free blocks or the operation (or lack) of the TRIM command
Power consumption	High performance flash-based SSDs generally require 1/2 to 1/3 the power of HDDs; High performance DRAM SSDs generally require as much power as HDDs and consume power when the rest of the system is shut down.	High performance HDDs generally require between 12-18 watts; drives designed for notebook computers are typically 2 watts.

Commercialization

Cost and capacity

The technological trend is a 2 year 50% decline in costs, while capacities continue to double at the same rate. As a result, flash-based solid-state drives are becoming increasingly popular in markets such as notebook PCs and sub-notebooks for enterprises, Ultra-Mobile PCs (UMPC), and Tablet PCs for the healthcare and consumer electronics sectors.

Availability

CompactFlash card used as SSD

Solid-state drive (SSD) technology has been marketed to the military and niche industrial markets since the mid-1990s..

Along with the emerging enterprise market, SSDs have been appearing in ultra-mobile PCs and a few lightweight laptop systems, adding significantly to the price of the laptop, depending on the capacity, form factor and transfer speeds. As of 2008, some manufacturers have begun shipping affordable, fast, energy-efficient drives priced at $350 to computer manufacturers. For low-end applications, a USB flash drive may be obtainable for anywhere from $10 to $100 or so, depending on capacity, or a CompactFlash card may be paired with a CF-to-IDE or CF-to-SATA converter at a similar cost. Either of these requires that write-cycle endurance issues be managed, either by not storing frequently written files on the drive, or by using a flash file system. Standard CompactFlash cards usually have write speeds of 7 to 15 MB/s while the more expensive upmarket cards claim speeds of up to 40 MB/s.

One of the first mainstream releases of SSD was the XO Laptop, built as part of the One Laptop Per Child project. Mass production of these computers, built for children in developing countries, began in December 2007. These machines use 1,024 MiB SLC NAND flash as primary storage which is considered more suitable for the harsher than normal conditions in which they are expected to be used. Dell began shipping ultra-portable laptops with SanDisk SSDs on April 26, 2007. Asus released the Eee PC subnotebook on October 16, 2007, and after a successful commercial start in 2007, it was expected to ship several million PCs in 2008, with 2, 4 or 8 gigabytes of flash memory. On January 31, 2008, Apple Inc. released the MacBook Air, a thin laptop with optional 64 GB SSD. The Apple store cost was $999 more for this option, as compared to that of

an 80 GB 4200 rpm Hard Disk Drive. Another option, the Lenovo ThinkPad X300 with a 64 gigabyte SSD, was announced by Lenovo in February 2008, and is, as of 2008, available to consumers in some countries. On August 26, 2008, Lenovo released ThinkPad X301 with 128GB SSD option which adds approximately $200 US.

The Mtron SSD

In 2008, low end netbooks appeared with SSDs. In 2009, SSDs began to appear in laptops.

On January 14, 2008, EMC became the first enterprise storage vendor to ship flash-based SSDs into its product portfolio.

In late 2008, Sun released the *Sun Storage 7000 Unified Storage Systems* (codenamed Amber Road), which use both solid state drives and conventional hard drives to take advantage of the speed offered by SSDs and the economy and capacity offered by conventional hard disks.

Dell began to offer optional 256 GB solid state drives on select notebook models in January 2009.

In May 2009, Toshiba launched a laptop with a 512 GB SSD.

As of October 2010, Apple's MacBook Air line carry solid state drives, standard.

In December 2010, OCZ RevoDrive X2 PCIe SSD available in 100GB to 960GB capacities delivering speeds over 740MB/s sequential speeds and random small file writes up to 120,000 IOPS.

Quality and performance

SSD is a rapidly developing technology. In November 2010, Tom's Hardware made recommendations based on user needs. "You usually won't regret buying an SSD with a SandForce SF-1200 controller. The balance between throughput, I/O performance, and application performance is still impressive. Corsair, G.Skill, Patriot, OCZ, Runcore, and others have suitable offerings. Look for best prices, warranty, and maybe an installation kit for your desktop PC if needed. Mobile users can only get rock bottom power consumption if they stay with Toshiba hardware, which isn't the fastest. Intel SSDs offer a good compromise between power and performance, but you have to be aware of their limited write performance. Samsung's 470-series only shows marginal weaknesses across our benchmark suite. If Samsung were to price the drive more aggressively, it would be a more compelling option."

Performance of flash SSDs is difficult to benchmark. In a test done by Xssist, using IOmeter, 4 KB RANDOM 70/30 RW, queue depth 4, the IOPS delivered by the Intel X25-E 64 GB G1 started around 10,000 IOPs, and dropped sharply after 8 minutes to 4,000 IOPS, and continued to decrease gradually for the next 42 minutes. IOPS vary between 3,000 to 4,000 from around the 50th minutes onwards for the rest of the 8+ hours test run.

This only affected write performance with consumer grade drives. Enterprise grade drives avoid this problem by overprovisioning, and by employing wear-leveling algorithms that only move data around when the drives are not being heavily utilized.

Applications

Until 2009, SSDs were mainly used in those aspects of mission critical applications where the speed of the storage system needed to be as fast as possible. Since flash memory has become a common component of SSDs, the falling prices and increased densities have made it more financially attractive for many other applications. Organizations that can benefit from faster access of system data include equity trading companies, telecommunication corporations, and video streaming and editing firms. The list of applications which could benefit from faster storage is vast. Any company can assess the ROI from adding SSDs to their own applications to best understand if that will be cost effective for them.

Flash-based Solid-state drives can be used to create network appliances from general-purpose PC hardware. A write protected flash drive containing the operating system and application software can substitute for larger, less reliable disk drives or CD-ROMs. Appliances built this way can provide an inexpensive alternative to expensive router and firewall hardware.

SSDs based on an SD card with a live SD operating system are easily write-locked. Combined with a cloud computing environment or other writable medium, to maintain persistence, an OS booted from a write-locked SD card is robust, rugged, reliable, and impervious to permanent corruption. If the running OS degrades, simply turning the machine off and then on returns it back to its initial virgin uncorrupted state and thus is particularly solid. The SD card installed OS does not require removal of corrupted components since it was write-locked though any written media may need to be restored.

SSD-optimized file systems

There are a number of file systems which are optimized for solid-state drives. Some of the more popular or notable are listed below.

Microsoft Windows

Versions of Windows prior to Windows 7 are optimized for hard disk drives rather than SSDs. Windows Vista includes ReadyBoost to exploit characteristics of USB-connected flash devices, but for SSDs it only improves the partition alignment to prevent read-modify-write operations because the SSD is typically aligned on 4 KB sectors and the OS is based on 512 byte sectors and they are not aligned. The proper alignment really does not help the SSD's endurance over the life of the drive. Some Vista operations, if not disabled, can shorten the life of the SSD. Disk defragmentation should be disabled because the location of the file components on an SSD doesn't significantly impact its performance, but moving the files to make them contiguous using the Windows Defrag routine will cause write wear on the limited number of P/E cycles on the SSD. The Superfetch feature will not materially change the performance of the system and causes additional overhead on the system and SSD, although it does not cause wear.

Windows 7 is optimized for SSDs as well as for hard disks. The OS looks for the presence of an SSD and operates differently with that drive. If an SSD is present, Windows 7 will disable disk defragmentation, Superfetch, ReadyBoost, and other boot-time and application prefetching operations. It also includes support for the TRIM command to reduce garbage collection of data which the OS has already determined is no longer valid (without TRIM the SSD would be unaware of this data being invalid).

ZFS

Solaris can use SSDs as a performance booster for ZFS. An SSD can be used for the ZFS Intent Log (ZIL), where it is named the SLOG. This is used every time a synchronous write to the disk occurs. An SSD may also be used for the level 2 Adaptive Replacement Cache (L2ARC), which is used to cache data for reading. When used either alone or in combination, large increases in performance are generally seen.

Linux systems

The vital TRIM function is supported by the Linux OS starting with 2.6.33 kernel (available early 2010). The ext4 filesystem is supported when mounted using option "discard". The most recent disk utilities (and therefore installation software that make use of them) do also apply proper partition alignment.

Standardization organizations

The following are noted standardization organizations and bodies that work to create standards for solid-state drives (and other computer storage devices). It also includes organizations who promote the use of solid-state drives. This is not necessarily an exhaustive list.

Organization or Committee	Subcommittee of:	Purpose
INCITS	N/A	Coordinates technical standards activity between ANSI in the USA and joint ISO/IEC committees worldwide
T10	INCITS	SCSI
T11	INCITS	FC
T13	INCITS	ATA
JEDEC	N/A	Develops open standards and publications for the microelectronics industry
JC-64.8	JEDEC	Focuses on solid-state drive standards and publications
NVMHCI	N/A	Provides standard software and hardware programming interfaces for nonvolatile memory subsystems
SATA-IO	N/A	Provides the industry with guidance and support for implementing the SATA specification
SFF Committee	N/A	Works on storage industry standards needing prompt attention when not addressed by other standards committees
SNIA	N/A	Develops and promotes standards, technologies, and educational services in the management of information
SSSI	SNIA	Fosters the growth and success of solid state storage