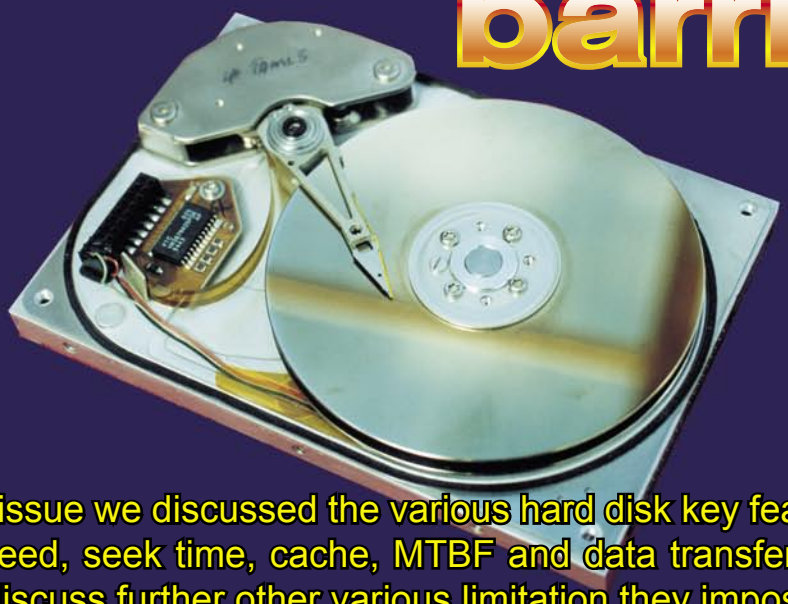
# Hard disk barriers



**In the last issue we discussed the various hard disk key features, such as spindle speed, seek time, cache, MTBF and data transfer rates. In this issue we discuss further other various limitation they impose on a system (or is being imposed on them), be that by the BIOS or the operating sys-**

prepared by Vlado Damjanovski

## Introduction

When the personal computer (PC) was first developed, designers had to decide how many bytes would be designated for addressing particular memory locations within the system, including hard drive memory. It was also necessary to specify how the address bytes would be structured to access that memory. Originally, these bytes were divided into cylinder, head, and sector (CHS) address locations that related directly to the physical layout of the hard drives. Unfortunately, the designers of the system BIOS (Basic Input/Output System) and the ATA (Advanced Technology Attachment) interface did not set up the total bytes used for addressing in the same manner, nor did they define the same number of bytes for the cylinder, head, and sector addressing. The differences in the CHS configurations required that there be a translation of the address when data was sent from the system (using the system BIOS) and the ATA interface. This is what caused the problems that were found at 528 MB and 4.2 GB, which limited systems to smaller capacity drives.

The new limitation involves the total addressing space that was defined for the system BIOS. A few years ago, most PC systems were limited to accessing drives with a capacity of 8.4 GB or less (7.9 GB on some systems.) The BIOS's of these systems did not have adequate address locations to access more than 8.4 GB.

A solution to this problem required updating the system BIOS to one that supports the interrupt 13 extensions or patching the BIOS with the extension support. Today, most PC system BIOS's support the interrupt 13 extensions, allowing systems to address drives of extraordinarily large capacities.

## The problem

The ATA interface uses 28 bit addressing which supports drives that are $2^{28}$X 512 bytes (2 to the power of 28, times 512 bytes) or 137 GB. Unfortunately, most system BIOS's use 24 bit addressing which only allows access to $2^{24}$ X 512 bytes or 8.4 GB. (This number really multiplies out to be 8.6GB but because of the way the BIOS uses the bit, only 8.4 GB can be accessed.) When the system wants to read or write data to the disk, the BIOS has to use a **software interrupt**. The main interrupt that is used to access the disk drive is interrupt 13h. This interrupt was assigned 24 bits of addressing, which only allows the system to access 8.4GB on a disk drive. System designers were aware of this limitation and defined extensions for interrupt 13. This allows for a quad-word or 64 bits of addressing, which is equal to $2^{64}$ X 512 bytes or 9.4 x $10^{21}$ bytes. That is 9.4 Tera Giga bytes or over a trillion times as large as an 8.4 GB drive.

## Systems affected

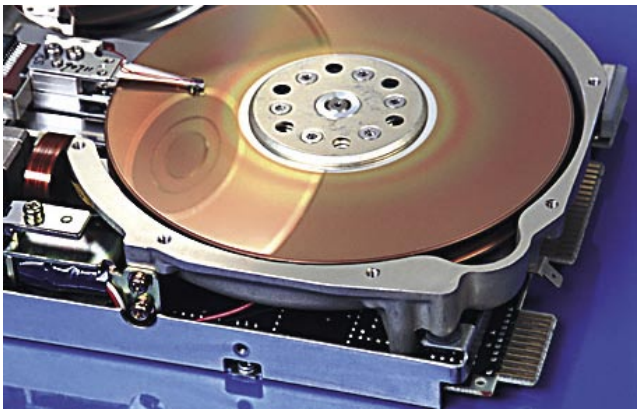
Very few systems built in 1997 (or before) properly support the BIOS interrupt 13 extensions. However, all major BIOS manufacturers today have BIOS's that support the extensions. By mid 1998, most new systems had this support. Systems without the support can be modified to use drives greater than 8.4 GB.

## The solution

There are a number of things that can be done to update a system in order to allow the use of large drives.

The options available are:

1. Obtain a new version of BIOS that supports the interrupt 13 extensions from the system or BIOS manufacturer.
2. Load software on the system that links into the BIOS to add the interrupt 13 support. There are several vendors that offer software for this purpose. Ontrack's Disk Manager for IBM (version 9) can be downloaded at no charge from the following web site: http://www.storage.ibm.com/hdd/support/download.htm
3. Use an intelligent host adapter whose BIOS supports interrupt 13 extensions. This adds the cost of additional hardware, but may be the best solution for certain systems.



## What's the next limitation?

The next limitation with the ATA interface should occur at 137 GB. (Some systems and operating systems may encounter other unforeseen limitations before this.) At 137 GB, the 28 bits of addressing on the ATA bus run out. Some possible solutions for this problem follow:

1. The ATA's Feature Register could be used to add an additional 8 bits giving 28+8=36 or 35.2 TB of addressable space.
2. The size of each sector could be increased. For example, a sector size of 4096 bytes would increase the maximum size of the drives to 2.2 TB.
3. The industry could switch to a completely different interface. The IEEE 1394 interface is the most likely candidate and may gain popularity before one of the other options needs to be implemented.

## Operating system limitation

With the increase in disk drive size, there is another limitation that affects the user that cannot be corrected by updating the BIOS. This is due to an inherent limitation within the operating systems. The most widely used are Windows 95, 98 and now 2000. The earlier versions of Windows operating systems (3.1 and 95 A) only support a maxi-

mum partition size of 2.1GB. This means that drives over 2.1GB will have to be partitioned into several logical drives, C:, D:, and so on. 8.4GB drives will require at least 4 logical drives.

Here are the limitations of the maximum addressable hard disk space with the various operating systems:

**DOS 6.22 or LESS**: DOS 6.22 or less does not support drives greater than 8.4GB.

**Windows 95**: Windows 95 version A (standard version) does support extended interrupt 13 so it can support drives with capacities greater than 8.4 GB. Because of the limitation of the FAT16 file system, a minimum of five partitions will need to be created on the hard drive. This is caused by the 2.048 GB partition limitation of a FAT 16 Operating System. Microsoft has identified that Windows 95 does NOT support drive capacities greater than 32 GB. They recommend that users either upgrade to Windows 98 or Windows NT 4.0, Service Pack 4 (or newer). Reference Microsoft Article Q246818 for more information. NOTE: Windows 95 Upgrade is NOT considered a standard version. It requires that MS-DOS or Windows 3.1x be installed. This means that a Windows Upgrade will NOT support drive capacities greater than 8.4 GB.

**Windows 95B / OSR2, Windows 95C and Windows 98**: Windows 95B (OSR2), Windows 95C and Windows 98 does support extended interrupt 13 which allows the operating system to support drives larger than 8.4 GB. These operating systems also support FAT 32. This file system lets the user create partitions larger than 2.048 GB. FAT 32 can only be used on hard drives whose capacity exceeds 512 megabytes. Microsoft has identified that Windows 95 does NOT support drive capacities greater than 32 GB. They recommend that users either upgrade to Windows 98 or Windows NT 4.0, Service Pack 4 (or newer) or Windows 2000. Reference Microsoft Article Q246818 for more information. Microsoft has also identified occasions when the operating system may misreport cluster sizes (and/or disk capacities) on drives whose capacities exceed 32 GB. Reference Microsoft Article Q243450 for the fix to this issue.

**Windows NT 4.0**: Windows NT 4.0 WILL support drive capacities greater than 8.4 GB provided.

**Windows 2000** Supports FAT 16, FAT 32 and NTFS file systems. As mentioned above, FAT 16 file system supports partition sizes up to 2.048 GB. FAT 32 file system supports up to 2TB (Terabyte=1,000GB) and, according to Microsoft, two Terabytes should be considered the partition size limit for NTFS file systems. For further information on partition and volume size limitations for Windows 2000 please see Microsofts Windows 2000 Article on Maximum Volume Sizes

In **Linux**, the maximum file sizes depends on a lot of parameters (e.g. as block size for the various file systems ext2/ext3), and is likely to evolve depending on the kernel version and architecture. Nonetheless, the minimum available, according to the file system limits, is currently generally near or superior to 2TB (1Tb=1024 GB) and can go up to 4PB (1PB=1024 TB) for JFS filing system variation. Unfortunately, these values are also limited to maximum block device size, which in the Linux kernel before the 2.4 is limited (for X86 architecture only) to 2TB.

The latest Linux kernel 2.4 (which runs on Red Hat 7, Mandrake 8 and SuSe 7) can support not only larger file sizes but each IDE controller actually supports two separate disks (hard drives, cdrom drives, etc.) which appear under Linux as separate block devices. Linux 2.4 has improved on Linux 2.2's support of IDE by more than doubling the number of IDE controllers allowed in a system to 10. (Previously, 4 was the maximum allowed.) This boosts Linux to a theoretical limit of 20 IDE devices.

*Sources: www.ibm.com; www.linux.org; www.microsoft.com*