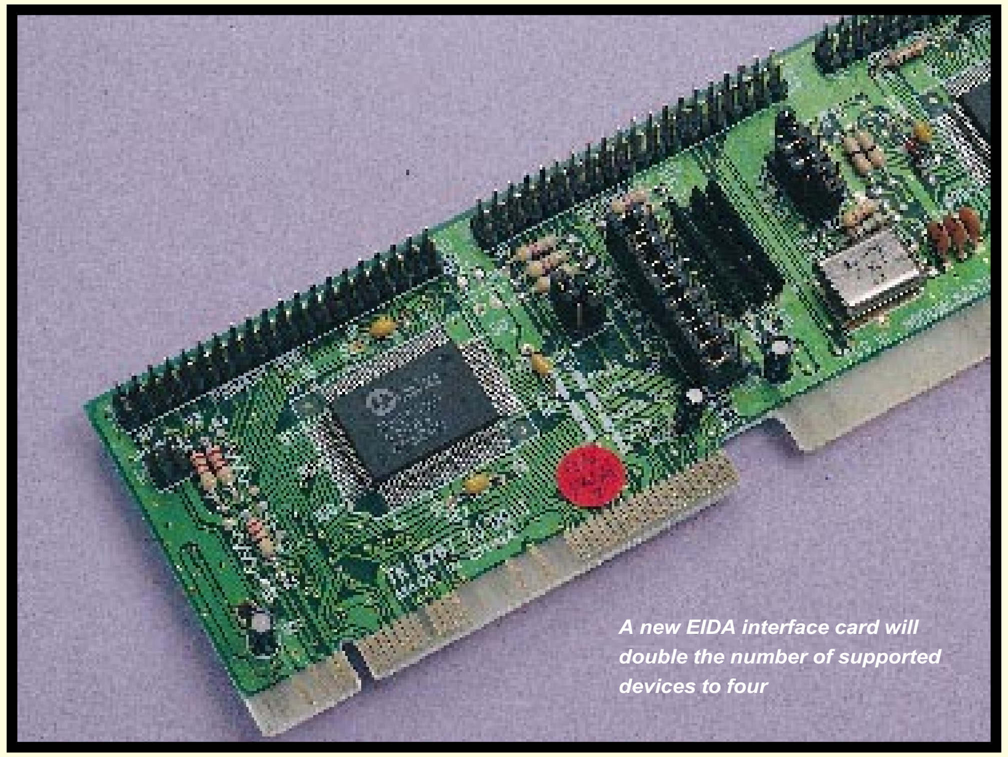
Magic **card** trick

A bigger hard disk is increasingly affordable, but it could cause compatibility problems. Roger Gann shows how you can use EIDE to overcome them.

Given the bloated nature of modern 32-bit operating systems and applications, these days it isn't so much a question of "if" but of "when shall I fit a bigger hard disk?" Luckily, big new hard disks have never been more affordable. Almost uniquely among computer components, they have simultaneously grown in size and speed while their prices have plummeted. Today, you can pick up a good-quality 1Gb drive for £160 or less – just two years ago I paid £700 (sob!).

The drop in price is great, but the overall increase in size can be a problem, at least from the compatibility point of view. It's now becoming difficult to buy disks smaller than 540Mb and older machines can't directly support these larger disks without resorting to software trickery. This is no big deal for DOS+Windows 3.1x, but for Windows 95, which uses 32-bit protected-mode drivers, the presence of real-mode drivers can result in reduced performance.

A much more elegant solution is a low-cost hardware upgrade to give support for larger drives and let Windows 95 handle the drives directly. In this month's column I'll take you through the process of upgrading to Enhanced IDE (EIDE) and I'll complete the piece next month by showing you how to install a new large hard disk.



A new EIDA interface card will double the number of supported devices to four

Size solutions

There are several solutions to the IDE size problem (see page 295). Most large EIDE hard disks come complete with special device driver software that permits the PC to recognise the entire capacity of the drive. They work by placing a BIOS overlay in the boot sector, which is the same for all IDE drives, regardless of size. The overlay is loaded as soon as the drive boots, and it provides the support for large partitions that is missing from an older BIOS.

The down side of this solution is that it won't give you a second IDE channel, which is useful for connecting up slow IDE devices such as CD-ROMs and tape streamers. Also, it's likely that Windows 95 will run its file system in the slower MSDOS compatibility mode. Even under Windows 3.1x, a special driver is still required, to restore both 32-bit Disk Access and 32-bit File Access. Another more worrying problem is that if for any reason you lose the special driver, DOS won't be able to "see" the big hard disk.

A better option is to buy an inexpensive EIDE interface card. These cost about £35 and are a direct replacement for the £10 multi I/O cards fitted in most popular PCs. Examples include the VL-Bus Promise EIDE2300-Plus and the ISA Bus Future Domain IDE-16042. Both include two IDE channels and a floppy-drive interface. The Promise also features enhanced serial and parallel ports, which is a useful upgrade if you have a fast modem or a printer that needs a "bi-di" printer port. And, of course, they both include an on-board BIOS that supports enhanced IDE: this supplements your existing BIOS.

The only drawbacks I can think of is that you'll possibly lose an expansion slot, and the additional BIOS steals a little (16Kb) Upper Memory.

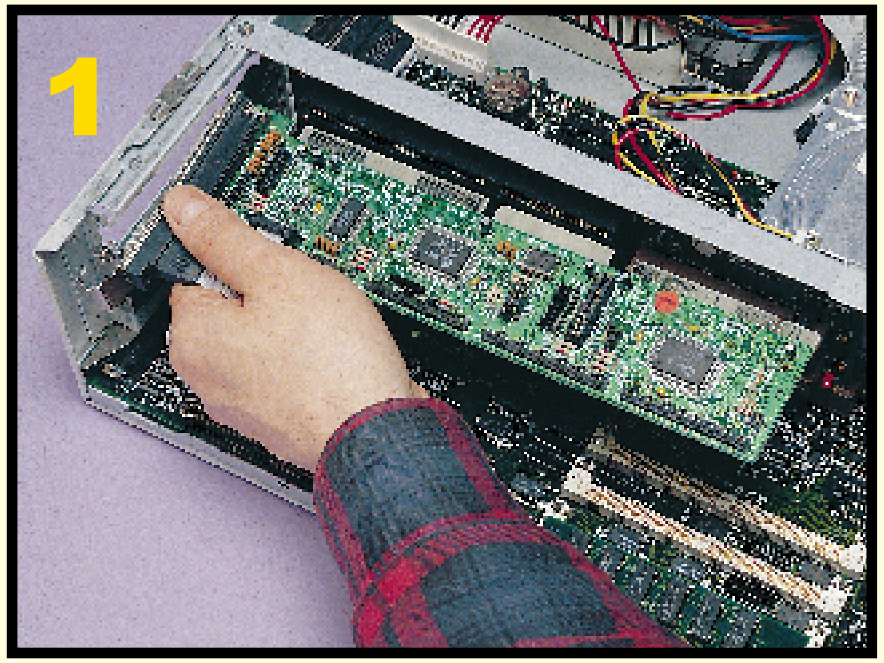
Another good reason for going down the EIDE card route is that you gain a second IDE channel. This doubles the number of IDE devices supported to four: you could, in theory, hang four IDE hard disks off such a card, with a master and slave drive attached to each channel. However, a more likely scenario would be a pair of Enhanced IDE hard disks on the primary or EIDE channel and slower IDE devices, designed for the ATA Packet Interface (ATAPI) standard such as CD-ROM drives or tape streamers attached to the secondary or IDE channel.

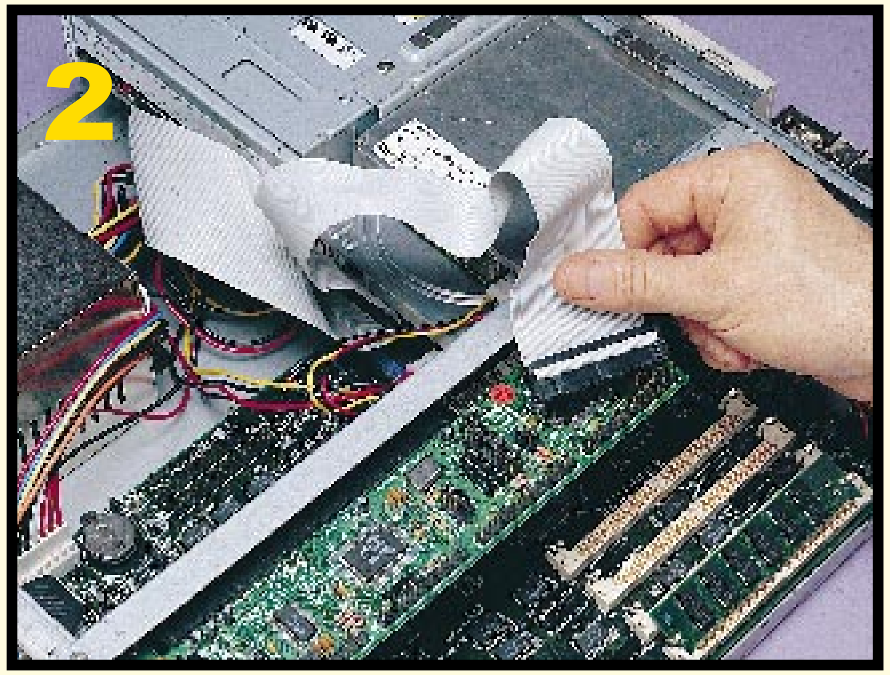
The received wisdom is that it's a good idea to give IDE devices with wildly differing data transfer rates separate IDE channels. At its peak, an EIDE drive using Mode 4 can transfer data at rates as fast as 16Mb/sec; a typical quad-speed CD-ROM drive knocks it out at 600Kb/sec. The lowest common denominator rule applies again here and there will be occasions when the data transfer rate of the hard disk will be pulled down to that of the CD-ROM, especially where both are in use simultaneously, for example when you're installing software from CD-ROM. With 8X CD-ROM data transfer rates now approaching 1.2Mb/sec and beyond, this speed differential is set to become less of a problem, but it's still something most of us have to watch out for.

Yet another benefit of EIDE is speed. A new EIDE interface card will support the lightning fast data throughput of the new drives . There are four types, or modes, of Polled I/O or PIO transfers and the fastest drives now support Mode 4 which, as we've seen, can offer data transfer rates as high as 16.6Mb/sec. The most your existing Mode 0 IDE can manage is 3.3Mb/sec.

A word of caution here: these are maximum burst rate figures, and the actual data throughput for Mode 3 and Mode 4 drives will be significantly lower than those numbers indicate. Note too that to upgrade to EIDE you'll need a motherboard with a local bus. These speeds demand a 32-bit VL-Bus or PCI local bus; the 16-bit 8MHz ISA bus can't support these levels of throughput.

So, as a prelude to upgrading your hard disk, we'll replace your existing IDE interface card with a new EIDE interface card. As we'll be leaving your existing hard disk *in situ*, there's no special need to back it up; and in any case, you probably already have recent backups of your hard disk, don't you?





(1) *Here, a new EIDE disk controller is fitted into place. This one is designed for a VL-bus*

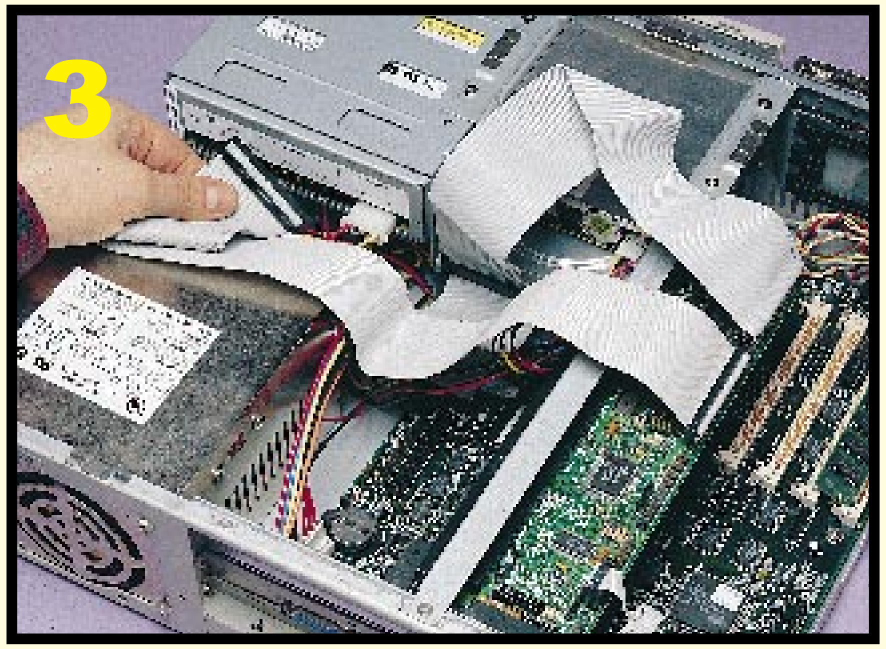
(2) *It is possible to daisy-chain the hard disk and CD-ROM drive together on the same channel, but this is not recommended due to their different data transfer rates*

Step-by-Step

INSTALLING ENHANCED IDE

Step 1

● Make the usual preparations: clear a work space and get out your Phillips screwdriver. Power down and unplug the PC from the mains and disconnect all other leads, such as printers and mice. Before handling any electrical components, discharge any static electricity you might be carrying by earthing yourself: touch a metal pipe, for instance, or the



(3) *The CD-ROM drive is generally fitted into the secondary IDE channel on the EIDE card*

PC's chassis. Undo the four or five screws and take the lid off the PC.

Step 2

● Locate the existing IDE interface card and disconnect the ribbon cables from it. You may also have to unplug any external serial ports. Undo the screw holding the card in and extract the card from its slot. Put it somewhere safe. If your PC features I/O integrated in the motherboard, there's nothing to remove, but you'll have to disable at least the on-board IDE interface, and more if the new card has a floppy interface and serial and parallel ports.

Step 3

● Configure the new EIDE card before you fit it. These early cards are definitely not plug-and-play; they are often festooned with jumpers and may require adjustment to avoid clashes. For example, the EIDE2300Plus defaults to using DMA Channel 1, which is the same as the default setting of a SoundBlaster card. If you have one of these you should use the alternative, DMA 3, and move the appropriate jumpers. Also, don't forget that the second IDE channel steals another IRQ, typically 15, so make sure this IRQ is free. When you're done, find a spare slot for the new EIDE interface card and insert it. Tighten up the fixing screw.

If you're adding a PCI EIDE controller, note that most PCI systems can't map IRQs 14 and 15 to a PCI slot. Since those interrupts are typically needed by a disk controller, you should have a workaround, which comes in the form of a ribbon cable and paddle card. Luckily, this can plug in to the otherwise unusable ISA "shared" slot, so you don't lose a slot. This card allows the controller to tap interrupts 14 and 15 through the ISA connection. Some newer PCI motherboards offer a separate connector, called a legacy connector, near one of the PCI slots to provide the needed IRQ. For these systems, you simply connect the ribbon cable from the EIDE card to the legacy connector rather than to a second ISA card.

Step 4

● Now refit the hard disk drive's 40-way ribbon cable. Modern ribbon cables are polarised with a notch to prevent you inserting them the wrong way round, but some aren't, so look for a red or coloured stripe on the ribbon cable. This indicates Pin 1. Look at the socket on the new interface card for the Pin 1 label to correctly orient the plug. Be sure to plug it in to the E-IDE/primary (and not the IDE) port on the EIDE interface card, making sure to align the coloured edge with Pin 1.

Because of the very high data transfer rates possible thanks to EIDE, you have to be careful with the ribbon cable and you mustn't use one longer than 18 inches. Don't forget to plug in the 34-way floppy ribbon cable on to the floppy interface on the EIDE card, making sure to get it the right way round. Reconnect any serial or game port cables, too.

Step 5

● If you have an IDE CD-ROM, now's the time to give it its very own IDE channel. Use a separate 40-way ribbon cable to connect it to the secondary IDE channel on the EIDE card. Don't forget to set it to be a "master" IDE device if it has been set to a "slave" previously (though in my experience, this doesn't seem to matter too much). Do this by moving a jumper at the back of the drive.

Step 6

● Reassemble the PC, put the lid back on, do up the screws, plug all the cables back in and power up the PC. Check the CMOS setup and get the BIOS to auto-identify the hard disk just to make sure that all the cabling has been installed correctly. Some EIDE cards may let their own BIOS handle the hard disk totally and insist therefore that you set the Drive Type on the PC's BIOS to "Not Installed". Your PC should reboot at this point. If it doesn't, re-check the cabling, especially the Pin 1 process.

Step 7

● Now's the time to install the card's Windows 3.1x and Windows 95 drivers. For Windows 95, fire up Control Panel, click on the "Add New Hardware" icon and opt for the manual install. Select "Hard Disk Controllers" and click on the "Have Disk" button. Place the driver disk in the floppy and click on "OK". When that's done, restart Windows 95. ▣

Explanation of acronyms and terms used

● Hard drive and peripheral interface standards

BIOS	Basic input/output system.
EIDE	Enhanced integrated drive electronics.
IDE	Integrated drive electronics.

● Other terms

ATAPI	AT Attachment Packet Interface.
CMOS	Complementary metal-oxide semiconductor.
DMA	Direct Memory Access.
IRQ	Interrupt Request Line
PIO	Polled (or programmed) input/output.

The IDE ceiling

While a PC's standard BIOS is theoretically capable of supporting hard disks as large as 8Gb and IDE drives can reach 130Gb in size, the real size limitation is just 504Mb. This low figure is the result of a clash between the drive geometries (the number of cylinders, heads and sectors per track) supported by each hardware standard. Unfortunately, these differ substantially and DOS defaults to the lowest common denominator in each case, hence the 504Mb limit. The table below highlights the problem:

	BIOS	IDE	Lowest value
Max. sectors per track	63	255	63
Max. heads	255	16	16
Max. cylinders	1,024	65,536	1,024

The maximum size of a hard disk drive is derived from multiplying these values, (sectors x heads x cylinders) by 512 (the number of bytes in a sector), so you'll end up with the following capacities, in bytes:

```
8,422,686,720
136,902,082,560
528,482,304
```

Or, to express these values in megabytes, we divide them by 10,242 or 1,048,576:

```
8,032.5Mb
130,560Mb
504Mb
```

And that's how we wind up with the half gigabyte IDE limitation. Two years ago it wasn't a problem as IDE drives were no bigger than 420Mb and only SCSI drives were larger. As they used their own BIOS, the limit didn't apply to them. However, with today's large IDE drives, the 504Mb limit is a very real problem for older PCs.

PCW Contacts

Roger Gann can be contacted either by post c/o *PCW* or via email at **rgann@mcgilivray.win-uk,net**