

Information regarding the Enhanced IDE drive in Linux 2.6

The hdparm utility can be used to control various IDE features on a running system. It is packaged separately. Please Look for it on popular linux FTP sites.

```
*** IMPORTANT NOTICES:  BUGGY IDE CHIPSETS CAN CORRUPT DATA!!
*** =====
*** PCI versions of the CMD640 and RZ1000 interfaces are now detected
*** automatically at startup when PCI BIOS support is configured.
***
*** Linux disables the "prefetch" ("readahead") mode of the RZ1000
*** to prevent data corruption possible due to hardware design flaws.
***
*** For the CMD640, linux disables "IRQ unmasking" (hdparm -u1) on any
*** drive for which the "prefetch" mode of the CMD640 is turned on.
*** If "prefetch" is disabled (hdparm -p8), then "IRQ unmasking" can be
*** used again.
***
*** For the CMD640, linux disables "32bit I/O" (hdparm -c1) on any drive
*** for which the "prefetch" mode of the CMD640 is turned off.
*** If "prefetch" is enabled (hdparm -p9), then "32bit I/O" can be
*** used again.
***
*** The CMD640 is also used on some Vesa Local Bus (VLB) cards, and is *NOT*
*** automatically detected by Linux.  For safe, reliable operation with such
*** interfaces, one *MUST* use the "cmd640.probe_vlb" kernel option.
***
*** Use of the "serialize" option is no longer necessary.
```

Common pitfalls:

- 40-conductor IDE cables are capable of transferring data in DMA modes up to udma2, but no faster.
 - If possible devices should be attached to separate channels if they are available. Typically the disk on the first and CD-ROM on the second.
 - If you mix devices on the same cable, please consider using similar devices in respect of the data transfer mode they support.
 - Even better try to stick to the same vendor and device type on the same cable.
-

This is the multiple IDE interface driver, as evolved from hd.c.

It supports up to 9 IDE interfaces per default, on one or more IRQs (usually

ide.txt

14 & 15). There can be up to two drives per interface, as per the ATA-6 spec.

```
Primary:   ide0, port 0x1f0; major=3; hda is minor=0; hdb is minor=64
Secondary: ide1, port 0x170; major=22; hdc is minor=0; hdd is minor=64
Tertiary:  ide2, port 0x1e8; major=33; hde is minor=0; hdf is minor=64
Quaternary: ide3, port 0x168; major=34; hdg is minor=0; hdh is minor=64
fifth..    ide4, usually PCI, probed
sixth..    ide5, usually PCI, probed
```

To access devices on interfaces > ide0, device entries please make sure that device files for them are present in /dev. If not, please create such entries, by using /dev/MAKEDEV.

This driver automatically probes for most IDE interfaces (including all PCI ones), for the drives/geometries attached to those interfaces, and for the IRQ lines being used by the interfaces (normally 14, 15 for ide0/ide1).

Any number of interfaces may share a single IRQ if necessary, at a slight performance penalty, whether on separate cards or a single VLB card. The IDE driver automatically detects and handles this. However, this may or may not be harmful to your hardware.. two or more cards driving the same IRQ can potentially burn each other's bus driver, though in practice this seldom occurs. Be careful, and if in doubt, don't do it!

Drives are normally found by auto-probing and/or examining the CMOS/BIOS data. For really weird situations, the apparent (fdisk) geometry can also be specified on the kernel "command line" using LILO. The format of such lines is:

```
ide_core.chs=[interface_number.device_number]:cyls,heads,sects
or ide_core.cdrom=[interface_number.device_number]
```

For example:

```
ide_core.chs=1.0:1050,32,64 ide_core.cdrom=1.1
```

The results of successful auto-probing may override the physical geometry/irq specified, though the "original" geometry may be retained as the "logical" geometry for partitioning purposes (fdisk).

If the auto-probing during boot time confuses a drive (ie. the drive works with hd.c but not with ide.c), then an command line option may be specified for each drive for which you'd like the drive to skip the hardware probe/identification sequence. For example:

```
ide_core.noprobe=0.1
or
ide_core.chs=1.0:768,16,32
ide_core.noprobe=1.0
```

Note that when only one IDE device is attached to an interface, it should be jumpered as "single" or "master", *not* "slave". Many folks have had "trouble" with cdroms because of this requirement, so the driver now probes for both units, though success is more likely when the drive is jumpered correctly.

Courtesy of Scott Snyder and others, the driver supports ATAPI cdrom drives

ide.txt

such as the NEC-260 and the new MITSUMI triple/quad speed drives. Such drives will be identified at boot time, just like a hard disk.

If for some reason your cdrom drive is **not** found at boot time, you can force the probe to look harder by supplying a kernel command line parameter via LILO, such as:

```
ide_core.cdrom=1.0      /* "master" on second interface (hdc) */
or
ide_core.cdrom=1.1      /* "slave" on second interface (hdd) */
```

For example, a GW2000 system might have a hard drive on the primary interface (/dev/hda) and an IDE cdrom drive on the secondary interface (/dev/hdc). To mount a CD in the cdrom drive, one would use something like:

```
ln -sf /dev/hdc /dev/cdrom
mkdir /mnt/cdrom
mount /dev/cdrom /mnt/cdrom -t iso9660 -o ro
```

If, after doing all of the above, mount doesn't work and you see errors from the driver (with dmesg) complaining about `status=0xff', this means that the hardware is not responding to the driver's attempts to read it. One of the following is probably the problem:

- Your hardware is broken.
- You are using the wrong address for the device, or you have the drive jumpered wrong. Review the configuration instructions above.
- Your IDE controller requires some nonstandard initialization sequence before it will work properly. If this is the case, there will often be a separate MS-DOS driver just for the controller. IDE interfaces on sound cards usually fall into this category. Such configurations can often be made to work by first booting MS-DOS, loading the appropriate drivers, and then warm-booting linux (without powering off). This can be automated using loadlin in the MS-DOS autoexec.

If you always get timeout errors, interrupts from the drive are probably not making it to the host. Check how you have the hardware jumpered and make sure it matches what the driver expects (see the configuration instructions above). If you have a PCI system, also check the BIOS setup; I've had one report of a system which was shipped with IRQ 15 disabled by the BIOS.

The kernel is able to execute binaries directly off of the cdrom, provided it is mounted with the default block size of 1024 (as above).

Please pass on any feedback on any of this stuff to the maintainer, whose address can be found in linux/MAINTAINERS.

The IDE driver is modularized. The high level disk/CD-ROM/tape/floppy drivers can always be compiled as loadable modules, the chipset drivers can only be compiled into the kernel, and the core code (ide.c) can be compiled as a loadable module provided no chipset support is needed.

When using ide.c as a module in combination with kmod, add:

ide.txt

alias block-major-3 ide-probe

to /etc/modprobe.conf.

When ide.c is used as a module, you can pass command line parameters to the driver using the "options=" keyword to insmod, while replacing any ',' with ';'.

Summary of ide driver parameters for kernel command line

For legacy IDE VLB host drivers (ali14xx/dtc2278/ht6560b/qd65xx/umc8672) you need to explicitly enable probing by using "probe" kernel parameter, i.e. to enable probing for ALI M14xx chipsets (ali14xx host driver) use:

- * "ali14xx.probe" boot option when ali14xx driver is built-in the kernel
- * "probe" module parameter when ali14xx driver is compiled as module ("modprobe ali14xx probe")

Also for legacy CMD640 host driver (cmd640) you need to use "probe_vlb" kernel parameter to enable probing for VLB version of the chipset (PCI ones are detected automatically).

You also need to use "probe" kernel parameter for ide-4drives driver (support for IDE generic chipset with four drives on one port).

To enable support for IDE doublers on Amiga use "doubler" kernel parameter for gayle host driver (i.e. "gayle.doubler" if the driver is built-in).

To force ignoring cable detection (this should be needed only if you're using short 40-wires cable which cannot be automatically detected - if this is not a case please report it as a bug instead) use "ignore_cable" kernel parameter:

- * "ide_core.ignore_cable=[interface_number]" boot option if IDE is built-in (i.e. "ide_core.ignore_cable=1" to force ignoring cable for "ide1")
- * "ignore_cable=[interface_number]" module parameter (for ide_core module) if IDE is compiled as module

Other kernel parameters for ide_core are:

- * "nodma=[interface_number.device_number]" to disallow DMA for a device
- * "noflush=[interface_number.device_number]" to disable flush requests
- * "nohpa=[interface_number.device_number]" to disable Host Protected Area
- * "noprobe=[interface_number.device_number]" to skip probing
- * "nowerr=[interface_number.device_number]" to ignore the WRERR_STAT bit

ide.txt

* "cdrom=[interface_number.device_number]" to force device as a CD-ROM

* "chs=[interface_number.device_number]" to force device as a disk (using CHS)

Some Terminology

IDE = Integrated Drive Electronics, meaning that each drive has a built-in controller, which is why an "IDE interface card" is not a "controller card".

ATA = AT (the old IBM 286 computer) Attachment Interface, a draft American National Standard for connecting hard drives to PCs. This is the official name for "IDE".

The latest standards define some enhancements, known as the ATA-6 spec, which grew out of vendor-specific "Enhanced IDE" (EIDE) implementations.

ATAPI = ATA Packet Interface, a new protocol for controlling the drives, similar to SCSI protocols, created at the same time as the ATA2 standard. ATAPI is currently used for controlling CDROM, TAPE and FLOPPY (ZIP or LS120/240) devices, removable R/W cartridges, and for high capacity hard disk drives.

mlord@pobox.com

Wed Apr 17 22:52:44 CEST 2002 edited by Marcin Dalecki, the current maintainer.

Wed Aug 20 22:31:29 CEST 2003 updated ide boot options to current ide.c comments at 2.6.0-test4 time. Maciej Soltysiak <solt@dns.toxicfilms.tv>