

Utilizando a camera digital Minolta Dimage 5 no Linux



by Guido Socher ([homepage](#))

About the author:

O Guido gosta do Linux não só pelas excelentes possibilidades deste sistema, bem como pelas pessoas envolvidas no seu desenho.



Abstract:

Recentemente, decidi comprar uma nova camera digital. Como já é habitual, procurei, em primeiro lugar, por cameras suportadas pelo Linux. Nunca compro hardware novo que não seja suportado pelo Linux. Geralmente, recomendo esta estratégia a todas as pessoas, mesmo que tenham outro sistema operativo instalado. Dá-lhe mais liberdade e flexibilidade. Assim, quando um dia se decidir só pelo Linux, não precisa de comprar hardware novo. O site www.gphoto.org deu-me a resposta: Sim, a camera Minolta Dimage 5, uma das que tinha em mente, era totalmente suportada.

Comprei a camera 5 e quis tirar as imagens da camera para o computador mas surpresa das surpresas: Todos os sites do Linux que falavam acerca da Dimage 5 diziam que a mesma trabalha com o linux, mas não consegui encontrar uma só palavra que dissesse como a pôr a trabalhar @!?!%

Foi neste momento que decidi escrever um pequeno artigo acerca do Linux e da camera Minolta Dimage.

Introdução

A camera Dimage 5 trabalha perfeitamente no Linux mas para tal precisa de configurar o sistema usb-storage. Mais especificamente, precisa de recompilar o kernel.

Concentrar-me-ei, neste pequeno artigo, somente na parte de instalação e configuração. Não explicarei como ver, editar ou arquivar as imagens digitais.

A ideia

As cameras Dimage 5 e Dimage 7 utilizam o sistema USB mass. Ou seja, no ponto de vista do Linux são um disco rígido. Pode copiar, mover, remover imagens com o seu gestor de ficheiros (e.g. konqueror) ou com os comandos da shell. Ou por outras palavras não precisa de fazer nada de especial para obter as imagens. Só precisa de "convencer" o Linux a falar para a sua máquina.

Passo a Passo

Para este artigo utilizei o Kernel 2.4.17, outros da série 2.4 poderão, também, trabalhar contudo os da série 2.2 não trabalharam visto que o sistema de armazenamento usb mass não foi portado para este série do kernel.

Edite o ficheiro unusual_devs.h que se encontra em /usr/src/linux/drivers/usb/storage/ e adicione seguintes entradas:

```
UNUSUAL_DEV( 0x0686, 0x4008, 0x0001, 0x0001,
"Minolta",
"Dimage 5",
US_SC_SCSI, US_PR_BULK, NULL, US_FL_START_STOP ),
UNUSUAL_DEV( 0x0686, 0x4006, 0x0001, 0x0001,
"Minolta",
"Dimage 7",
US_SC_SCSI, US_PR_BULK, NULL, US_FL_START_STOP ),
```

Esta é a parte mais difícil! O resto é sempre em frente. Informei o responsável do ficheiro por estas entradas. Provavelmente algumas versões futuras do kernel incluíram estas linhas.

Agora recompile o kernel e certifique-se que seleccionou as seguintes opções, para além de todas as coisas que, normalmente, precisa de seleccionar para o seu hardware.

```
# General setup
CONFIG_HOTPLUG=y

# SCSI support
CONFIG_SCSI=y
CONFIG_BLK_DEV_SD=y
CONFIG_SD_EXTRA_DEVS=40
CONFIG_SR_EXTRA_DEVS=4
CONFIG_CHR_DEV_SG=m
CONFIG_SCSI_CONSTANTS=y
CONFIG_SCSI_LOGGING=y

# File systems
CONFIG_FAT_FS=m
CONFIG_VFAT_FS=m

# USB support
CONFIG_USB=y
CONFIG_USB_DEVICEFS=y

# USB Controllers
```

```
CONFIG_USB_UHCI=m
CONFIG_USB_UHCI_ALT=m
CONFIG_USB_OHCI=m
```

```
# USB Device Class drivers
CONFIG_USB_STORAGE=m
CONFIG_USB_STORAGE_DATAFAB=y
CONFIG_USB_STORAGE_DPCM=y
CONFIG_USB_STORAGE_SDDR09=y
CONFIG_USB_STORAGE_JUMPSHOT=y
```

Pode verificar estas opções no ficheiro /usr/src/linux/.config. Penso que na realidade, não precisa das opções CONFIG_USB_STORAGE_DATAFAB, CONFIG_USB_STORAGE_DPCM, CONFIG_USB_STORAGE_SDDR09 e CONFIG_USB_STORAGE_JUMPSHOT mas estas são alguns dispositivos de leitura de memórias flash, podem-lhe fazer jeito quando adquire uma camera.

O suporte SCSI é preciso porque o usb-storage aparecerá ao utilizador como disco SCSI.

Agora compile e instale o kernel. Existe imensa documentação acerca da compilação do kernel (o kernel-HOWTO da www.linuxdoc.org ou este [artigo da Linuxfocus](#)) por conseguinte, não explicarei quaisquer detalhes. Basicamente, precisa de fazer o seguinte:

```
#crie os links em /usr/include para:
```

```
scsi -> ../src/linux/include/scsi
```

```
asm -> ../src/linux/include/asm-i386
```

```
linux -> ../src/linux/include/linux
```

```
# configure:
```

```
make xconfig
```

```
# compile:
```

```
make dep
```

```
make clean
```

```
make bzImage
```

```
make modules
```

```
make modules_install
```

```
cp /usr/src/linux/arch/i386/boot/zImage /boot/vmlinuz-2.4.17
```

```
cp /usr/src/linux/System.map /boot/System.map-2.4.17
```

```
cp /boot/System.map-2.4.17 /boot/System.map
```

```
cp /usr/src/linux/.config /boot/Config-2.4.17
```

```
#edite /etc/lilo.conf
```

```
# corra lilo (ou o grub se o utiliza)
```

```
/sbin/lilo
```

```
#faça reboot ao computador
```

Errata:

Obrigado a Seth W. Klein por apontar que já não é melhor criar ligações simbólicas a partir de /usr/include. Para uma explicação detalhada [veja este mail do Linus](#).

Ligando a camera

Ligue a camera e configure a ligação usb. Isto está descrito no manual que acompanha a camera.

Verifique com o comando `/sbin/lsmode` que tem os módulos `usb-storage` e `usb-uhci` (ou `usb-ohci`) carregados. Caso contrário, carregue-os com

```
modprobe usb-uhci
modprobe usb-storage
```

Abra o ficheiro `/proc/bus/usb/devices` e verifique que consegue encontrar a entrada para a camera:

```
T: Bus=01 Lev=01 Prnt=01 Port=00 Cnt=01 Dev#= 3 Spd=12 MxCh= 0
D: Ver= 1.00 Cls=00(>ifc ) Sub=00 Prot=00 MxPS= 8 #Cfgs= 1
P: Vendor=0686 ProdID=4008 Rev= 0.01
S: Manufacturer=MINOLTA DIMAGE CAMERA
S: Product=DIMAGE CAMERA
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr= 0mA
I: If#= 0 Alt= 0 #EPs= 2 Cls=08(stor.) Sub=06 Prot=50 Driver=usb-storage
E: Ad=03(O) Atr=02(Bulk) MxPS= 16 IvL= 0ms
```

O sistema de ficheiros `proc usb` com o ficheiro de dispositivos em cima é, normalmente, montado automaticamente mas este não é o caso, assim monte-o manualmente com o seguinte comando:

```
mount -t usbdevfs /proc/bus/usb
```

Para aqueles que estão curiosos com o significado dos números na linha do driver no ficheiro dos dispositivos aqui vai uma pequena explicação. A camera diz ao computador que tipo de protocolo fala:

```
Cls=08(stor.) -> Class usb storage
Sub=06 -> USB sub class 06= transparent SCSI =US_SC_SCSI (veja protocol.h a partir das fontes do kernel)
Prot=50 -> protocol usb bulk transfer only =US_PR_BULK (veja transport.h a partir das fontes do kernel)
```

A camera aparecerá como um disco SCSI. Se já possuir um outro qualquer disco SCSI então será `/dev/sdb1` caso contrário será `/dev/sda1`. Com o `fdisk` pode verificar onde o disco se encontra:

```
fdisk -l /dev/sda
```

```
Disk /dev/sda: 4 heads, 32 sectors, 244 cylinders
Units = cylinders of 128 * 512 bytes
```

```
Device Boot Start End Blocks Id System
/dev/sda1 * 1 245 15664 1 FAT12
```

Adicione a seguinte linha ao `/etc/fstab`:

```
/dev/sda1 /mnt/camera0 vfat rw,noauto,user 0 0
```

Crie o directório `/mnt/camera0` e permita a escrita de todos:

```
mkdir /mnt/camera0  
chmod 777 /mnt/camera0
```

```
chmod 666 /dev/sda1
```

Esta configuração permite que qualquer utilizador de montar e desmontar a camera sem ter de mudar para root. (Nota que o `chmod 777 /mnt/camera0` e `chmod 666 /dev/sda1` não são precisos de o seu comando `mount` tiver o `s-bit` para o root definido, as permissões de ficheiro `/sbin/mount` é `-rwsr-xr-x` se o `s-bit` estiver configurado sendo a opção `user` no `/etc/fstab` suficiente.)

Obtendo as imagens

Terminámos já a configuração e o setup. Agora digite:

```
mount /mnt/camera0
```

e obtenha todas as suas imagens a partir da árvore de directório abaixo de `/mnt/camera0!`

Uma palavra de aviso: Nunca desligue a camera sem a desmontar primeiro (`umount /mnt/camera0`). Pode bloquear o seu computador.

Escrevi uma pequena script para, automaticamente, retirar todas as imagens do cartão de memória Flash da camera. Esta script também monta e desmonta automaticamente a camera. Deste modo, nunca desliga a camera sem antes a desmontar. Chamei-lhe [cfimageget \(download\)](#).

Guarde este ficheiro como `cfimageget` e torne-o executável com `chmod 755 cfimageget`. Para copiar todas as imagens para o directório corrente (".") corra, simplesmente

```
cfimageget .
```

Acerca da camera

Provavelmente, tenciona comprar uma Minolta Dimage e está interessado na minha experiência com esta camera ?

No geral, posso dizer que é uma camera muito boa, em muitos aspectos excedeu as minhas expectativas. Opera perfeitamente com o Linux logo que procedeu à configuração e ao procedimento de compilação do kernel.

Pode obter especificações técnicas acerca da camera em www.dimage.minolta.com.

A camera tem, como muitas outras cameras, um grande problema. Come baterias. Fiz algumas medidas: Somente com o visor electrónico precisa de 0.6 Amperes numa voltagem de 6 Volts. Quando procede à auto-focagem e com o pequeno monitor consome em média 1 Ampere. Isto é uma média de 1 Ampere ! Com picos até aos 3 Amperes. Com 4 pequenas baterias AA NiMh (1600 mAh) a camera trabalhará cerca de 1 hora a 1,5 hora numa utilização contínua. A Minolta diz que pode tirar cerca de 200 fotografias com um par de baterias. Pessoalmente, penso que tem de tirar estas fotos bastante rápido para não ficar sem baterias. Isto é assim porque não importa se tira uma fotografia agora ou não. O que importa é o tempo em que a camera está ligada.

Referências

- As páginas usb do Linux linux-usb.org
- A página hotplug do linux linux-hotplug.sourceforge.net
O hotplug não é mais do que um conjunto, complicado, de scripts para automaticamente carregar módulos como o usb-storage. Eu prefiro uma simples entrada modprobe no /etc/rc.d/rc.local (Redhat/Mandrake) mas é consigo.
- Uma selecção de programas para ver as imagens:
gimp (www.gimp.org),
konqueror (www.kde.org),
gphoto (www.gphoto.org) Não tem necessidade da funcionalidade de descarregamento do gphoto, mas pode utilizá-lo para pesquisar qualquer directório e gerar páginas índice em html,
gqview (gqview.sourceforge.net),
xv (www.trilon.com/xv/xv.html) Este programa já existia quando muitos computadores tinham desktops à base de texto, mas ainda é bom.
- Cameras Casio e usb-storage: www.harald-schreiber.de

<p><u>Webpages maintained by the LinuxFocus Editor team</u> © Guido Socher "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org</p>	<p>Translation information: en --> -- : Guido Socher (homepage) en --> pt: Bruno Sousa <bruno/at/linuxfocus.org></p>
--	--