

Utiliser un appareil numérique Minolta Dimage 5 sous Linux



par Guido Socher ([homepage](#))



Résumé:

L'auteur:

Guido aime le système Linux non seulement pour ses larges possibilités mais aussi pour les gens qui sont impliqués dans son développement.

Traduit en Français par:
Jean Peyratout
<jean.peyratout@at.abul.org>

J'ai décidé il y a quelque temps d'acheter un appareil photo numérique. Comme d'habitude j'ai d'abord recherché les appareils reconnus sous Linux: je n'achète aucun matériel non compatible Linux et je recommande cette stratégie à tous, même si vous avez un autre système d'exploitation installé. Cela vous donne plus de liberté et de souplesse, et vous n'aurez pas à racheter de matériel le jour où vous déciderez de ne plus travailler qu'avec Linux. Le site www.gphoto.org me donna la réponse: oui, l'un des appareils que j'avais en vue, le Dimage 5 de Minolta, était pleinement reconnu.

J'ai donc acheté un Dimage 5 mais quand j'ai voulu récupérer les images dans l'ordinateur, surprise, surprise: tous les sites Linux parlant du Dimage 5 disaient bien qu'il fonctionnait avec Linux... mais impossible de trouver un seul mot expliquant comment le faire marcher @!?!%

Voilà pourquoi j'ai décidé d'écrire cet article sur le Dimage de Minolta et Linux.

Introduction

L'appareil photo numérique Dimage 5 fonctionne parfaitement sous Linux mais pour cela il faut configurer le système de stockage usb. Plus précisément il faut sans doute recompiler le noyau.

Je me focaliserai dans ce très court article uniquement sur la partie configuration et installation. Je n'expliquerai pas comment voir, éditer ou archiver les images numériques.

L'idée générale

Les appareils Dimage 5 et Dimage 7 utilisent le système de stockage USB. Cela signifie que pour Linux ils sont considérés comme des disques durs. Vous pouvez copier, déplacer, effacer des images avec n'importe quel gestionnaire de fichier (par ex. konqueror) ou avec des commandes shell. Autrement dit il ne vous faut rien de spécial pour récupérer les images. Il vous faut juste convaincre Linux de "dialoguer" avec votre appareil.

Pas à pas

J'ai utilisé le noyau 2.4.17 pour cet article. Les autres noyaux 2.4.x fonctionneront sans aucun doute aussi bien mais les noyaux 2.2.x ne fonctionneront pas car le système de stockage usb n'a pas été porté sous 2.2.x.

Éditez le fichier unusual_devs.h qui se trouve dans /usr/src/linux/drivers/usb/storage/ et ajoutez :

```
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 ),
```

Vous avez fait le plus difficile ! Le reste est plus simple. J'ai informé celui qui s'occupe de ce fichier de ces ajouts. Les versions futures du noyau inclueront ces lignes.

Maintenant recompilez le noyau en vous assurant que vous avez sélectionné les options suivantes - en plus de ce dont vous avez normalement besoin pour votre matériel.

```
# 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
```

Vous pouvez vérifier ces options dans le fichier `/usr/src/linux/.config`. Je ne crois pas que vous ayez véritablement besoin des options `CONFIG_USB_STORAGE_DATAFAB`, `CONFIG_USB_STORAGE_DPCM`, `CONFIG_USB_STORAGE_SDDR09` et `CONFIG_USB_STORAGE_JUMPSHOT` mais c'est nécessaire pour certains lecteurs de cartes flash et il est bon de pouvoir éventuellement s'en servir quand on achète un appareil photo.

Le support SCSI est nécessaire car le stockage usb apparaîtra comme si c'était un disque SCSI.

Maintenant compilez et installez le noyau. Il y a en ligne toute la documentation nécessaire sur la façon de compiler un noyau (Kernel-HOWTO de freenix ou cet article de Linuxfocus) aussi je n'entrerai pas dans les détails. En gros vous devez faire ceci :

```
#création des liens dans /usr/include pour:
```

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

```
# configuration:
```

```
make xconfig
```

```
# compilation:
```

```
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
```

```
#édition de /etc/lilo.conf
```

```
# lancer lilo (ou grub si vous l'utilisez)  
/sbin/lilo
```

```
#redémarrer l'ordinateur
```

Connecter l'appareil photo

Connectez l'appareil et configurez la connexion usb. Ceci est décrit dans le manuel livré avec l'appareil.

Vérifiez avec la commande `/sbin/lsmmod` que les modules `usb-storage` et `usb-uhci` (ou `usb-ohci`) sont chargés.

Si ce n'est pas le cas chargez-les avec la commande:

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

Ouvrez le fichier `/proc/bus/usb/devices` et vérifiez que vous voyez bien ce qui concerne l'appareil :

```
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
```

Normalement, le système de fichiers `usb` `proc` avec le fichier de périphériques ci-dessus est automatiquement monté mais si ce n'est pas le cas alors montez-le manuellement avec la commande :

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

Pour les curieux, les nombres sur la ligne du pilote dans le fichier de périphérique méritent une courte explication. L'appareil signale à l'ordinateur quel type de protocole il utilise:

```
Cls=08(stor.) -> Class usb storage  
Sub=06 -> USB sub class 06= transparent SCSI =US_SC_SCSI (cf. protocol.h dans le source du noyau)  
Prot=50 -> protocol usb bulk transfer only =US_PR_BULK (cf. transport.h dans le source du noyau)
```

L'appareil apparaîtra comme un disque SCSI. Si vous avez déjà un autre disque SCSI il sera appelé `/dev/sdb1` ou `/dev/sda1` s'il est seul. Avec `fdisk` vous pouvez vérifier où se trouve le disque :

```
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
```

Ajoutez la ligne suivante dans /etc/fstab:

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

Créez le répertoire /mnt/camera0 et donnez les droits en écriture à tous :

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

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

Ce réglage autorisera tout utilisateur à monter / démonter l'appareil sans avoir à basculer en root.

Récupérer les images

Nous voilà prêts pour ce qui est de la configuration. Maintenant tapez :

```
mount /mnt/camera0
```

et récupérez vos images depuis les répertoires sous /mnt/camera0!

Un petit mot d'avertissement : ne jamais déconnecter l'appareil sans le démonter d'abord (umount /mnt/camera0). Cela peut bloquer votre ordinateur.

J'ai écrit un petit script pour récupérer automatiquement toutes les images de la carte CompactFlash de l'appareil. Ce script monte / démonte automatiquement l'appareil. De cette façon vous ne débrancherez jamais l'appareil en oubliant de le démonter au préalable. J'ai appelé le script cfimageget (téléchargement).

Enregistrez ce fichier sous le nom cfimageget et rendez-le exécutable avec
chmod 755 cfimageget.

Pour copier toutes les images vers le répertoire courant ("."), lancez simplement

```
cfimageget .
```

À propos de l'appareil photo

Vous êtes peut-être arrivé jusqu'ici parce que vous envisagez d'acheter un Dimage Minolta et vous vous intéressez à mon expérience avec cet appareil.

En général je peux dire que c'est un très bon appareil et que sur bien des aspects il a dépassé mes attentes. Il marche à la perfection avec Linux une fois que vous avez effectué la procédure de configuration et de recompilation de noyau.

Vous pouvez obtenir les spécifications techniques de cet appareil à www.dimage.minolta.com.

L'appareil photo, comme la plupart des appareils numériques, a cependant un sérieux problème : il consomme des batteries. J'ai fait quelques mesures : juste avec le viseur électronique, il lui faut environ 0,6 Ampères en 6 Volts. Avec l'autofocus et le petit moniteur il consomme en moyenne 1 Ampère. C'est une moyenne ! ... avec des pics jusqu'à 3 Ampères.

Avec quatre petites batteries AA NiMh (1600 mAh) l'appareil pourra fonctionner durant environ une heure à une heure et demie si vous le laissez branché et que vous l'utilisez. Minolta prétend que vous pouvez prendre près de 200 photos avec un jeu de batteries. Je pense pour ma part que vous devez prendre ces 200 photos vraiment rapidement ou sinon vous serez à court de piles. Car il n'y a guère de différence que vous preniez vraiment des clichés ou non : ce qui compte est le temps durant lequel l'appareil reste sous tension.

Références

- Les pages linux usb linux-usb.org
- La page linux hotplug linux-hotplug.sourceforge.net
hotplug est un ensemble de scripts assez complexes permettant de charger automatiquement des modules comme usb-storage. Je préfère une simple entrée modprobe dans `/etc/rc.d/rc.local` (Redhat/Mandrake) mais c'est à vous de voir.
- Une sélection de programmes pour visionner des images :
gimp (www.gimp.org),
konqueror (www.kde.org),
gphoto (www.gphoto.org) Vous n'avez pas besoin de la fonction téléchargement de gphoto mais vous pouvez l'utiliser pour visiter un répertoire quelconque et générer des pages index html,
gqview (gqview.sourceforge.net),
xv (www.trilon.com/xv/xv.html) Ce programme existait déjà quand la plupart des ordinateurs n'étaient encore qu'en mode texte mais il est toujours bon.
- Appareils numériques Casio et usb-storage: www.harald-schreiber.de

Site Web maintenu par l'équipe d'édition LinuxFocus © Guido Socher "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Translation information: en --> -- : Guido Socher (homepage) en --> fr: Jean Peyratout < jean.peyratout@abul.org >
--	---