Zip Drive Mini-HOWTO

Kyle Dansie, `dansie@ibm.net` v2.4, 10 gennaio 1999

Questo documento fornisce una veloce guida di riferimento alla configurazione e all'utilizzo di un lettore ZIP sotto Linux. Traduzione di Giovanni Bortolozzo (*borto at pluto.linux.it*).

Indice

1 Introduzione

Il lettore Iomega ZIP è un dispositivo a dischi rimovibili piuttosto popolare. È disponibile in diverse versioni:

- Parallel Port
- SCSI
- IDE - ATAPI
- Plus

La più popolare sembra essere la versione per porta parallela. È in giro da diverso tempo e costa abbastanza poco. Anche la versione SCSI è disponibile da un po', ma poiché necessita di una scheda SCSI che molti non hanno a disposizione, non è così popolare come la precedente. In questo momento, la versione ATAPI sembra essere quella più facilmente reperibile. La Plus è la versione più recente del lettore ZIP: ha la capacità di rilevare automaticamente a quale porta (parallela o scsi che sia) è connesso. Questo documento è incentrato sulla versione per porta parallela, ma offre anche alcune dritte per le altre versioni.

Questo documento incorpora informazioni collezionate e pubblicate da altri, in particolare:

- Grant Guenther, grant@torque.net
- Scot Wilcoxon, sewilco@fielday.mn.org
- Joe Mack, mack@ncifcrf.gov
- Byron Jeff, byron@cc.gatech.edu

Li ringrazio per i loro importanti contributi, ed accetto la responsabilità per qualsiasi errore io abbia introdotto.

Un ringraziamento particolare a Grant Guenther che scrisse la versione 0.26 del programma ppa oltre alla versione originale di questo HOWTO. Se si trova un errore qualsiasi o inesattezze in questo documento, mi si informi. *dansie@ibm.net*

1.1 Dove reperire questo documento

La versione più aggiornata di questo documento risiede a

`http://www.njtcom.com/dansie/zip-drive.html`

Per la traduzione italiana si faccia riferimento alla home page dell' *Italian Linux Documentation Project* `http://wwww.pluto.linux.it/ildp/` .

2 Partenza a Razzo

OK, si è comprato un nuovo lettore ZIP ed ora si vuole usarlo. Alcune distribuzioni dispongono del modulo già compilato. Si può provare a dare `insmod ppa` per cominciare (ci si assicuri di averlo collegato e di aver inserito un disco). Se non c'è il programma ppa, allora bisogna configurare il kernel. La mia macchina ha due porte parallele, una per lo ZIP e l'altra per la stampante. Ciò rende le cose semplici, senza bisogno di scambiare i cavi.

Configurazione del kernel:

- cd /usr/src/linux
- make xconfig
- supporto per scsi = Y
- supporto per scsi disk = Y
- supporto per Iomega zip come modulo
- supporto per la stampante anche questo come modulo
- salvare ed uscire
- make dep
- make clean
- make zImage oppure zlilo o zdisk
- make modules
- make modules_install

Ora per usare il lettore:

- caricare il modulo: `insmod ppa`
- creare un mount point: `mkdir /zip`
- inserire un dischetto preformattato per win$ozz nel lettore.
- montare il disco: `mount -t vfat /dev/sda4 /zip`
- usare qualsiasi comando per i file :`l /zip, ls /zip, df, cp,`
- quando si ha finito: `umount /zip`

Se si hanno problemi con uno qualsiasi dei passi precedenti, si vedano le sezioni che seguono per istruzioni più dettagliate.

3 Configurare il kernel per il lettore ZIP

Per usare il lettore ZIP con Linux, si deve possedere un kernel configurato con il supporto per il sistema SCSI, il supporto per i dischi SCSI e il supporto per l'adattatore host utilizzato. Se non si ha familiarità con la compilazione del kernel, per maggiori informazioni si dovrebbe leggere il

Linux Kernel HOWTO `http://metalab.unc.edu/mdw/HOWTO/Kernel-HOWTO.html` .

Si deve iniziare il processo di costruzione di un kernel con il passo di configurazione. Qui, si identificano i componenti specifici del kernel di cui si ha bisogno. Il primo passo è `cd /usr/src/linux`. Esistono diversi modi per operare la reale configurazione. Sotto X io uso **make xconfig**. Ci sono pure `make menuconfig` o `make config` per il prompt a linea di comando. Il metodo più facile è con xconfig.

Nella sezione **SCSI Support** (Supporto SCSI) si imposti **SCSI support** = **Y**. Si imposti anche **SCSI disk support** = **Y** (Supporto per dischi SCSI).

Nella sezione **SCSI low-level drivers** (Driver SCSI a basso livello) si imposti **IOMEGA Parallel Port ZIP drive SCSI support** = **M** (Supporto SCSI per il lettore IOMEGA ZIP su porta parallela). La M sta per moduli.

Nella sezione **Character Devices** (Device a caratteri) si trovi ed imposti **Parallell Printer support** = **M** (Supporto per la stampante su porta parallela).

Se non si è molto sicuri di queste cose, si usi zdisk per il passo di make. Questo compilerà e installerà il kernel nel dischetto. Se si fa qualche casino, si avrà ancora un buon sistema avviabile sul disco fisso.

Ora di compili il kernel con questi passi.

- make dep
- make clean
- make zImage o zlilo o zdisk
- make modules
- make modules_install

Suggerimento se si vuole creare un registro dell'output del passo make zImage si può usare

```
make zImage 2>&1 | tee zImage.out
```

Questa è veramente un buona cosa da fare, se si hanno degli errori di compilazione o altri conflitti nella configurazione, ora si ha un file con i messaggi d'errore o gli errori di compilazione. Questo rende le richieste d'aiuto molto più facili.

Da qualche parte è necessario anche caricare i moduli. Sul mio sistema ho aggiunto un paio di righe al file boot.local.

- insmod ppa
- insmod lp

Il mio sistema è una Suse 5.3 e questo file è in /etc/rc.d. La posizione di questo file varierà da distribuzione a distribuzione, ma da qualche parte ci sarà un file per le modifiche locali da fare all'avvio. Si dovrebbe usare il file adeguato per la propria distribuzione.

Ora si faccia `shutdown` e il `reboot`.

Si può anche compilare tutto o parte del sistema SCSI come modulo. Se lo si fa, ci si assicuri di caricare scsi.o, poi sd.o e alla fine il driver per il proprio adattatore host, prima di provare ad accedere al lettore ZIP.

3.1 Versione SCSI

Si già si ha un disco SCSI nel proprio sistema e si sta connettendo il lettore ZIP allo stesso controller, non è necessaria alcuna configurazione aggiuntiva del kernel. Altrimenti, molto probabilmente si dovrà compilare un nuovo kernel.

Se si sta compilando un kernel per supportare la versione SCSI del lettore ZIP, si dovrebbe selezionare il supporto SCSI e il supporto per i dischi SCSI. Si deve anche selezionare un driver per la scheda di interfaccia che si userà. Se si ha uno ZIP Zoom, si selezioni il driver aha152x.

Ci si assicuri di leggere la documentazione per il proprio adattatore nello SCSI HOWTO e qualsiasi file README nella sottodirectory driver/scsi dell'albero dei sorgenti di Linux. Si faccia attenzione ai parametri in linea di comando che si dovranno usare per aiutare il kernel ad inizializzare l'adattatore.

Per esempio, se si sta usando una scheda ZIP Zoom, si dovrà aggiungere qualcosa di simile a

```
aha152x=0x340,11,7,1
```

al comando di avvio (o includerlo nel proprio file /etc/lilo.conf in un comando append). Questa dice al driver l'indirizzo della porta e l'IRQ della propria scheda ZIP Zoom - ci si assicuri che questi numeri corrispondano al modo in cui sono impostati i ponticelli sulla scheda.

Si dovrebbe leggere anche il *BOOTPROMPT HOWTO* `http://metalab.unc.edu/mdw/HOWTO/BootPrompt-HOWTO.html` di Paul Gortmaker per informazioni sulla configurazione del proprio kernel con LILO e LOADLIN.

4 Il lettore ZIP

Ci sono almeno cinque o sei versioni del lettore Iomega ZIP 100. La pagina della Iomega per questi lettori è a `http://www.iomega.com/product/zip/` . Tutte accettano speciali cartucce simili ai dischetti da 3.5 che possono contenere 100 megabyte di dati. Il realtà il disco contiene 96 cilindri di 2048 settori ognuno da 512 byte di capacità, ovvero un totale di 96 Megabyte.

Il lettore esterno è disponibile in tre varianti:

- Parallel port
- SCSI
- Plus

mentre il lettore interno è disponile nelle seguenti varianti:

- SCSI
- IDE
- ATAPI

Le versioni interne possono essere installate in uno slot per i lettori da 3.5 o da 5.25. Le versioni esterne sono racchiuse in un piccolo case di plastica blue, ed alimentate con un alimentatore esterno.

Tutti i lettori hanno un grosso bottone sul pannello anteriore, che serve per espellere il disco. Linux blocca la porta mentre sta usando il drive, ma se viene premuto il bottone mentre questa è bloccata, il lettore ZIP se ne ricorderà ed espellerà il disco non appena il software la sblocca.

4.1 La versione per porta parallela Zip100

Questo lettore è descritto a `http://www.iomega.com/product/zip/zippp.html` .

Il driver per Linux è disponibile in un paio di versioni diverse. La versione 0.26 del programma pppa.c è distribuita assieme ai kernel 2.0.x. Se il proprio kernel è più vecchio del 2.0.35 allora si ha il programma **VECCHIO**. Si veda la *homepage di David Campbell* `http://www.torque.net/~campbell/` per maggiori informazioni su questo programma. Durante la stesura di questo documento, l'ultima la versioni disponibile è la 1.42.

Le caratteristiche di auto configurazione del programma 1.4x sono molto carine. NON si deve dire al programma qual'è l'indirizzo base della porta parallela. Il programma cercherà la porta parallela negli indirizzi standard e connette lì il lettore quando lo trova.

Il problema più comune è il caricare il modulo per la stampante lp prima del programma ppa. Il programma ppa deve essere caricare prima del modulo lp. Il modulo lp si riserverà la porta di cui ha bisogno il lettore ZIP ed è per questo che deve essere caricato prima il programma ppa.

Ci si assicuri inoltre che i cavi siano attaccati bene.

Se si è fatto tutto bene :) si vedrà un messaggio come questo all'avvio.

```
VFS: Mounted root (ext2 filesystem) readonly.
Adding Swap: 102528k swap-space (priority -1)
ppa: Version 1.42
ppa: Probing port 03bc
ppa: Probing port 0378
ppa:     SPP port present
ppa:     EPP 1.9 with hardware direction protocol
ppa: Found device at ID 6, Attempting to use EPP 32 bit
ppa: Communication established with ID 6 using EPP 32 bit
ppa: Probing port 0278
ppa:     SPP port present
scsi0 : Iomega parport ZIP drive
scsi : 1 host.
  Vendor: IOMEGA    Model: ZIP 100          Rev: D.09
  Type:   Direct-Access                     ANSI SCSI revision: 02
Detected scsi removable disk sda at scsi0, channel 0, id 6, lun 0
SCSI device sda: hdwr sector= 512 bytes. Sectors= 196608 [96 MB] [0.1 GB]
sda: Write Protect is off
  sda: sda4
lp2 at 0x0278, (polling)
```

Si possono vedere questi messaggi con dmesg | more. Si possono salvare i messaggi in un file con dmesg > dmesg.txt. Questa cosa è spesso utile se si sta provando a fare il debug di un'installazione.

Si veda inoltre la sezione 8 (Ottenere maggioni informazioni).

4.2 La versione SCSI esterna

Questo lettore è descritta a `http://www.iomega.com/product/zip/zipscsi.html` .

La versione SCSI esterna del lettore ZIP ha due connettori DB25F, e due switch di configurazione. Uno switch seleziona l'indirizzo di destinazione del lettore. La scelta è limitata agli indirizzi 5 o 6. L'altro abilita un terminatore interno, nel caso il lettore sia l'ultimo di una catena. I connettori SCSI a 25 pin usano l'ormai familiare collegamento in stile Macintosh. Il lettore è venduto con un cavo di tipo Macintosh, ma cavi standard e convertitori sono facilmente reperibili nel caso si usi un controller SCSI con connettore Centronics o ad alta densità.

Non ho visto un lettore SCSI interno, ma mi aspetto abbia un connettore SCSI di tipo DIP a 50 pin con gli stessi due switch.

Ci si assicuri che l'indirizzo di destinazione scelto non sia in conflitto con altri dispositivi SCSI presenti nello stesso bus. Ci si assicuri inoltre che l'ultimo drive fisico della catena abbia la terminazione attivata o di installare un terminatore esterno.

Se si ha un disco SCSI interno o un CD-Rom, e si connette il proprio ZIP ad un controller già a disposizione, si dovrebbe controllare che non ci siano altri terminatori nella scheda che debbano essere rimossi. Solo le due terminazioni estreme del bus SCSI devono essere terminate. Se il proprio bus è parzialmente interno e parzialmente esterno, ci dovrebbe essere essere un terminatore nell'ultimo dispositivo esterno ed uno nell'ultimo dispositivo interno, ma non ci dovrebbero essere terminatori nella scheda stessa.

Assicurarsi che tutti i cavi siano ben fissati.

La configurazione del kernel è praticamente la stessa di quella per la versione interna. Si veda la prossima sezione sulla versione SCSI interna per un esempio di configurazione.

4.2.1 Il controller ZIP ZOOM

Iomega produce un controller SCSI chiamtato ZIP Zoom. Questo si basa in pratica sulla famiglia di adattatori **Adaptec AHA1520**. Ha un connettore esterno DB25F di tipo Macintosh, compatibile con il cavo venduto assieme al lettore ZIP.

Linux supporta questo adattatore con il driver **aha152x**.

4.3 La versione SCSI interna

Si installi l'hardware come descritto nella Installation and Reference Guide, annotando quale SCSI ID, IRQ e indirizzo di porta I/O sono stati usati (queste informazioni saranno usate più tardi). Le cose andranno subito lisce se il lettore ed il controller usano diversi ID SCSI.

Attenzione La scheda zip zoom è impostata in hardware per l'ID SCSI numero 7. L'ID SCSI nella riga di comando qui sotto è per la scheda non per il lettore. La scheda ed il lettore devono usare numeri di ID SCSI diversi. Per esempio se la scheda usa il 7 allora si usi 5 o qualche altro numero per il lettore.

Si ricompili il kernel dopo aver configurato ed incluso il supporto per 'SCSI', 'SCSI disk' e 'AHA152X/2825'. INSTALLARE IL NUOVO KERNEL :-(

Si dermini qual è la riga di comando da passare al kernel:

`aha152x=[I/O Port][,IRQ][,SCSI ID][,RECONNECT][,PARITY]`

Per esempio:

`aha152x=0x140,10,7,1`

Se si usa LILO si aggiunga la riga di comando per il kernel al proprio file lilo.conf, usando il comando append.

(ie. append = aha152x=0x140,10,7,1)

Se si usa LOADLIN si aggiunga la riga di comando per il kernel al comando usato per inizializzare loadlin.

```
(ie. loadlin c:\vmlinuz root=/dev/hda3 aha152x=0x140,10,5,1 ro)
```

4.4 Lo ZIP Plus - il driver IMM

Questo lettore è descritto a `http://www.iomega.com/product/zip/zipplus.html` .

Questa è una **nuovissima** versione del lettore ZIP esterno. Lo **Zip Plus** non usa il driver ppa: usa invece il driver **imm**. *Philippe Andersson* ha inviato questa eccellente sezione sul driver imm per lo Zip Plus. Grazie Philippe.

Per prima cosa si ricompili il proprio kernel per includere le seguenti cose:

- supporto per i moduli
- supporto SCSI
- supporto per dischi SCSI
- lp come modulo (se serve)

Si noterà che non si è selezionato alcun driver SCSI a basso livello. Ok, va tutto bene...lo si compilerà separatamente più tardi.

Adesso, bisogna procurarsi il codice sorgente del driver dalla home page di Dave Campbell (*http://www.torque.net/˜campbell/imm.tar.gz*). La versione 0.18 è quella corrente mentre sto scrivendo questo documento (agosto 98). Si faccia l'unzip e l'untar di questo da qualche parte (ad esempio sotto `/usr/src`). Poi semplicemente, si lanci `make`. Si otterrà il modulo di cui si ha bisogno (`imm.o`). Lo si copi in

```
/lib/modules/$(uname
-r)/scsi
```

.

Se è caricato il modulo lp (si controlli con **lsmod**(1)), lo si scarichi (`rmmod lp`), e poi si carichi `imm.o` (`insmod imm`), e tutto dovrebbe essere a posto.

Se il lettore ZIP non è connesso e alimentato quando si fa questa operazione, si riceverà il messaggio - init_module: Device or resource busy, e il modulo non sarà caricato (si potrebbe pensare che a questo punto sia necessario spegnere il PC e connettore lo ZIP, in realtà basta assicurarsi che il modulo non sia caricato e il lettore sia spento, poi basta collegarlo, accenderlo e caricare il modulo).

Se lo ZIP è connesso ed acceso, ma non c'è un disco al suo interno, il modulo si caricherà e si otterrà un messaggio che dice che non è in grado di leggere la tavola delle partizioni del disco. Nessun problema, in quanto la tavola delle partizioni sarà letta automaticamente quando si inserisce uno ZIP.

Se c'era uno ZIP, verrà mostrata una serie completa di informazioni, compreso un elenco delle partitizioni definite nel disco ed il loro stato di protezione dalla scrittura. Ecco qui un esempio del messaggio che si ottiene quando viene caricato il modulo:

```
vger:~# insmod imm
imm: Version 0.18
imm: Probing port 03bc
imm: Probing port 0378
imm:     SPP port present
imm:     ECP with a 16 byte FIFO present
imm:     PS/2 bidirectional port present
imm:     Passed Intel bug check.
imm: Probing port 0278
```

```
scsi0 : Iomega ZIP Plus drive
scsi : 1 host.
  Vendor: IOMEGA    Model: ZIP 100 PLUS      Rev: J.66
  Type:   Direct-Access                      ANSI SCSI revision: 02
Detected scsi removable disk sda at scsi0, channel 0, id 6, lun 0
SCSI device sda: hdwr sector= 512 bytes. Sectors= 196608 [96 MB] [0.1 GB]
sda: Write Protect is off
 sda: sda1
vger:~#
```

Un ultimo punto da sottolineare è che questa versione del driver blocca anche la porta del lettore quando è montato il disco. Se viene premuto il tasto di eject in questa situazione non succede niente, ma il lettore se ne ricorderà ed espellerà il disco non appena viene smontato.

4.5 La versione ATAPI

Questo lettore è descritto a `http://www.iomega.com/product/zip/zipatapi.html` .

Per un certo periodo è stata in produzione una versione IDE del lettore. Penso che adesso sia stata rimpiazzata dalla versione ATAPI.

Donald Stidwell mi ha spedito questi commenti sulla versione ATAPI. Grazie Don.

Uso il lettore Zip ATAPI e funziona sia con il kernel 2.0.32 che con il 2.0.33. L'ho usato sia sotto RH 5.0 che sotto OpenLinux 1.2 (la distribuzione che uso attualmente). Per farlo funzionare sotto OpenLinux, ho semplicemente abilitato **ATAPI floppy support** (Supporto per floppy ATAPI) nel kernel. OpenLinux non ha questo supporto compilato di default.

Non è necessario nessun altro driver. Sarà montato come una partizione estesa sulla partizione 4, i.e. il mio si monta su hdb4. Lo monto sotto /mnt/zip con l'opzione noauto, sebbene sono sicuro che non ci siano problemi reali con l'automount. Una piccola nota sull'espulsione dei dischi: io smonto sempre il disco prima di espellerlo.

Ci sono istruzioni più dettagliate per l'installazione della versione ATAPI nel numero di maggio 1998 della Linux Gazette. Si veda la sezione 2 cent tip.

```
<url url="http://www.linuxgazette.com/issue28/lg_tips28.html#atapi">
```

4.6 La versione IDE

Non ho usato la versione IDE. Eric Backus mi ha spedito questi commenti. Grazie Eric.

Io ne ho uno. Mi è stato dato assieme al mio computer Getaway 2000 circa un anno fa. Penso che la maggior parte di questi lettori siano stati venduti da grosse compagnie OEM come questa, prima che fosse disponibile la versione ATAPI del lettore ZIP.

Le buone nuove su questo lettore: per supportarlo non è necessario alcun modulo o modifica del kernel. Appare al kernel come un normale disco fisso IDE. Io l'ho usato senza problemi con i kernel 2.0.31 e 2.0.32.

Le cattiva notizie su questo lettore: poiché non usa ATAPI, non può usare la SCSI-to-ATAPI translation, il che implica che non è possibile usare gli mtools per proteggere in scrittura i dischi (o per espellerli, per quanto possa importare).

4.7 La versione USB

Questo lettore è descritto a `http://www.iomega.com/product/zip/imac.html` .

Questi lettori sembrano molto «cool» e anche l'interfaccia USB sembra buona. Il solo problema è che in questo momento l'interfaccia USB per Linux è ancora in sviluppo. Quindi per ora questo lettore non è utilizzabile sotto Linux. Si veda a questo indirizzo per le novità:

`http://peloncho.fis.ucm.es/~inaky/USB/news.html`

4.8 La versione ZIP250

Questo lettore è descritto a `http://www.iomega.com/product/zip/zip250.html` .

A gennaio 1999 questo lettore è una novità. Ne ho appena coprato uno e funziona molto bene sotto Linux. Si usi semplicemente il driver IMM scaricabile da `http://www.torque.net/~campbell/imm.tar.gz` . Si imposti il driver in maniera analoga che con lo zip plus. In pratica, semplicemente si scarichi il driver, lo si scompatti e si esegua make. Poi si copi l'eseguibile imm.o nella directory dei moduli. Per esempio /lib/modules/2.0.35/scsi (se 2.0.35 è la versione del proprio kernel). Se non si è sicuri della versione si usi uname -r.

5 Usare il lettore ZIP

Se si sono creati tutti i componenti richiesti, il kernel dovrebbe riconoscere il proprio adattatore ed il lettore al boot. Se si sta usando un modulo caricabile per il proprio lettore, la discussione che segue è applicabile una volta che sia stato caricato il modulo.

In linea di massima la versione SCSI e parallela del lettore si comportano nello stesso modo, tranne per il fatto che la versione parallela è in qualche modo più lenta.

5.1 Fdisk, mke2fs, mount, ecc.

Una volta noto il nome del device per il proprio lettore ZIP, si è a posto. Si può manipolare il disco con i normali comandi di gestione dei dischi di Linux. `fdisk` (o anche `cfdisk`) è usato per manipolare le tavole delle partizioni sul disco. `mke2fs` può essere usato per formattare una partizione con il filesystem ext2, il più comunemente usato in Linux. `mount` è usato per connettere una partizione formattata nella propria gerarchia di directory.

Se non si ha familiarità con questi strumenti, si dovrebbero studiare le relative pagine di manuale. Si faccia attenzione poiché adesso come adesso esistono diverse versione del programma `fdisk`.

Descriverò due scenari comuni.

5.2 Un disco DOS preformattato

Se si ha un disco ZIP con una struttura DOS dei file originariamente creata dagli strumenti della Iomega, l'analisi delle partizioni dovrebbe dire che il disco ha una sola partizione, `/dev/sda4`.

Si deve creare un posto dove montare il disco, diciamo `/zip`, e poi montarlo come un filesystem MS-DOS:

```
mkdir /zip
mount -t vfat /dev/sda4 /zip
```

Si potrebbe usare msdos invece di vfat. vfat supporta i nomi lunghi per i file mentre msdos no. Ora, i file nel disco dovrebbero apparire in /zip. Mentre il disco è montato, non si sarà in grado di rimuoverlo. Quando si ha finito con il disco si può smontarlo per rilasciarlo e staccarlo dalla propria gerarchia di directory.

```
umount /zip
```

Una volta creato il mount point /zip, non è necessario ricrearlo ancora, e quindi si può riutilizzare più tardi per montarci qualcos'altro.

5.3 Riformattazione come un disco Linux nativo

Se si vuole cancellare un disco ZIP e creare un file system nativo di Linux, di deve usare fdisk sull'intero disco:

```
fdisk /dev/sda
```

e cancellare qualsiasi partizione esistente (con il comando `d`). Poi si crei una nuova partizione con il comando `n`. La si crei primaria e con numero di partizione 1, poi si usi `w` per scrivere la tavola delle partizioni sul disco e si esca con `q`.

Si formatti la partizione con

```
mke2fs /dev/sda1
```

(L'1 è il numero che si è dato a questa partizione con `fdisk`). Ora si può montare il disco:

```
mount -t ext2 /dev/sda1 /zip
```

(riusando il mount point creato in precedenza).

5.4 Le voci in fstab

Il file /etc/fstab permette di configurare il comando mount. Mi piace poter montare e scrivere nel disco zip da qualsiasi account nel sistema. Ho aggiunto queste due righe alla fine del file fstab:

```
/dev/sda1    /zip          ext2          noauto,rw,user,nosuid,sync
/dev/sda4    /zipdos       vfat          noauto,rw,user,nosuid,sync,mode=0777
```

Queste voci assumono che ci siano le directory /zip e /zipdod. Se si vuole che tutti gli utenti possono leggere e scrivere, ci si assicuri di impostare i permessi. Per esempio **chmod 666 /zip**. Si deve eseguire chmod come root. Le voci in fstab assumono inoltre che i propri dischi dos siano partizionato come 4 (sda4 nel mio caso) e che i propri dischi ext2 siano partizionati come 1 (sda1). Si può sapere di più sulle opzioni di fstab con **man mount**.

Con le suddette voci di fstab si può montare un disco dos con **mount /zipdos**. Se si sta usando un disco ext2 si usi invece **mount /zip**.

5.5 Il disco ZIP Tools

C'è un altro po' di lavoro da fare se si vuole usare il disco che viene venduto con il lettore ZIP. Prima della distribuzione viene abilitata la protezione software dalla scrittura. Molti hanno per prima cosa sbloccato il disco sotto DOS prima di provare ad usarlo con Linux. Linux non può accedere a un disco bloccato, e questo dev'essere sbloccato con gli strumenti della Iomega.

Un programma nativo per Linux per gestire la caratteristica di protezione dalla scrittura ed altre cose, si pensa sarà presto disponibile.

6 Prestazioni

Con la versione standard dei kernel 2.0.x si riceve la versione 0.26 del programma ppa.c. Questo è un programma semplice ma molto affidabile. È stato sottoposto ad un uso intenso ed ora funziona piuttosto bene. La sola cosa brutta di questo programma è che non è molto veloce. Sulla mia macchina ho la versione 2.0.30 del kernel e due porte parallele. È pure una macchina da pochi soldi con un pentium 133, 16 mega di memoria e un disco eide della maxtor da 5 giga. La macchina mi è costata 500 dollari americani mentre il disco ne costa 250. Sono tirchio ma non troppo.

Ho messo il lettore ZIP nella seconda porta e la stampante nella prima. La seconda era una vecchia scheda senza la modalità epp. Ciò fa sì che il lettore funzioni in modalità a 4 bit.

Per verificare le prestazioni ho copiato il file tar del netscape dal disco fisso allo ZIP. Questo file è circa 10.6 mega. Per terminare l'operazione ci sono voluti 7 minuti e 15 secondi. Per trasferire dallo ZIP al disco fisso ci sono voluti 7 minuti. Non proprio esaltaltante, vero?

Allora ho fatto 2 modifiche. Per prima cosa ho passato il lettore ZIP nella prima porta e sono andato nel BIOS ed ho impostato la modalità EPP per la prima porta. Allo stesso tempo ho collegato la stampante nella seconda porta.

La seconda modifica che ho fatto è stato di scaricarmi la nuova versione del programma dalla *pagina di David Campbells* `http://www.torque.net/~campbell/` . Poi l'ho installato. Si dia un'occhiata al readme incluso.

Poi ho eseguito ancora lo stesso test copiando il file tar di Netscape nel e dal lettore ZIP. La copia nel disco fisso ha durato 1:55 minuti. Dal disco fisso allo ZIP invece c'ha messo 1 minuto giusto. Si confrontino questi valori con i 7:15 minuti e 7:00 minuti del vecchio programma. Si noti il notevole incremento di prestazioni rispetto alla versione 0.26 del programma. Tutto ciò inoltre è stato fatto usando il disco ZIP con un file system vfat.

Poi ho formattato un disco ZIP con il formato ext2. Ho eseguito ancora lo stesso test. La copia verso il disco fisso ha durato solo 50 secondi, mentre la copia dal disco fisso al disco ZIP solo 40.

Sommario delle Prestazioni

Per questo test ho spostato un file da 10.6 mega. La modalità standard a 4 bit della porta e la versione 0.26 del programma è risultata la combinazione più lenta con tempi di 7:00 e 7:15 minuti.

Il nuovo programma (1.41) e la modalità EPP con un filesystem ext2 è la combinazione più veloce con tempi di 50 e 40 secondi.

7 Filza di Assilli Quoditiani (aka Frequently Asked Questions)

Queste sono alcune delle domande che sono state fatte un sacco di volte. Invito a leggere le risposte prima di rifarle un'altra volta!

7.1 Posso collegare una stampante nel lettore ZIP per porta parallela?

Sì, ma: attualmente non è possibile avere contemporaneamente i driver lp e ppa attivi sulla stessa porta parallela. Durante lo sviluppo iniziale del gestore è stato progettato un protocollo di condivisione delle risorse, ma non sarà dispobile prima del rilascio di Linux 2.2.

Se si ama vivere sul filo del rasoio, si prenda il kernel 2.1.x e si può così avere la condivisione della porta. Si veda `http://www.cyberelk.demon.co.uk/parport.html` .

Il modo più semplice per fare questa cosa con un kernel 2.0.x è di avere due porte parallele. Ho comprato la mia seconda per circa 15 dollari e funziona alla grande. Si può usarne una per stampare e l'altra per lo ZIP. Semplicemente si compili un kernel con sia il driver ppa che il driver lp come moduli. Uso la versione più nuova (1.41) di ppa, quindi semplicemente carico ppa per primo (insmod ppa). Questo trova il lettore zip nella prima porta. Poi carico lp (insmod lp) e questo assegna la stampante alla seconda porta.

Se si usa la versione 0.26 del programma allora si aggiorni alla versione 1.41 o successiva. La sola risposta che si otterrà per domande su questo driver è di aggiornare il programma.

7.2 C'è l'intenzione di supportare le porte EPP/ECP in PPA?

La versione 0.26 del programma non supporta EPP. La versione 1.41 **supporta** EPP. Questo programma è molto più veloce. Si veda la sezione sulle prestazioni per maggiori dettagli.

7.3 Possono eseguire Linux da un lettore ZIP?

C'è gente che l'ha fatto. Questo sembra una gran cosa per fare un ottimo disco di ripristino o per testare un'installazione.

`http://metalab.unc.edu/LDP/HOWTO/mini/ZIP-Install.html`

7.4 Posso fare il boot da un lettore ZIP?

Dipende da cosa si usa come controller. Se il proprio controller ha una ROM BIOS con il codice per fare il boot dai target 5 o 6, allora sì, si può fare il boot da uno ZIP SCSI. Non è possibile fare il boot da uno ZIP su porta parallela.

Se si ha DOS nel proprio disco principale e si vuole avviare Linux da un disco ZIP, lo si può fare usando LOADLIN. L'immagine del proprio kernel deve essere da qualche parte nel disco DOS, ma il resto del proprio sistema Linux può essere nello ZIP. Ci si assicuri che il kernel che si usa abbia i giusti driver compilati al suo interno.

Per quel che ne so, non c'è nessun floppy di avvio dell'installazione per una qualsiasi delle distribuzioni più popolari che contenga il driver per lo ZIP su parallela.

7.5 Perché Iomega usa 4 come numero della partizione?

Questa è una delle domande più popolari, ma non penso che nessuno abbia la risposta definitiva. Probabilmente non c'è una vera ragione.

7.6 Posso fare in modo che il disco sia montato all'avvio?

Tutto quello che c'è da fare è aggiungere una riga al proprio file `/etc/fstab`. Per esempio, se si ha sempre un disco DOS nel lettore quando si fa il boot, si può mettere

```
/dev/sda4   /zip  vfat  defaults  0 0
```

in `fstab`. A seconda di come è fatta la propria distribuzione, gli script di inizializzazione potrebbero provare ad eseguire fsck sulle partizioni elencate nel proprio fstab. Si rammenti che ciò potrebbe creare problemi se si dimentica di mettere il disco nel lettore quando si avvia, oppure se si mette il disco sbagliato.

7.7 Cosa succende se non c'è nessun disco quando faccio il boot?

Il kernel proverà a leggere la tavola delle partizioni, ma l'operazione andrà in time out. Non è niente di cui preoccuparsi. Una volta inserito un disco e provato a farci qualcosa il kernel si accorgerà che adesso c'è e rileggerà la tavola delle partizioni.

SUGGERIMENTO: quando si cambiano i dischi, è sempre una buona idea usare fdisk per verificare la struttura delle partizioni nel nuovo disco.

Il BIOS in alcuni controller SCSI proverà a leggere la tavola delle partizioni del proprio disco durante il boot del sistema. Se non si può disabilitare questo controllo, si sarà costretti ad effettuare sempre il boot con un disco nel lettore.

7.8 Posso usare il lettore su porta parallela con un effettivo disco SCSI?

L'adattatore da parallela a SCSI PPP-3 è implementato come un unico chip ASIC che Iomega chiama VPIO. È inserito nella scheda di controllo del lettore ZIP. È effettivamente presente un bus SCSI, ma non in una forma utilizzabile.

Sebbene non abbia provato a confrontare le due schede, sembra logico che il VPIO rimpiazzi il circuito convenzionale di buffering elettrico che sarebbe richiesto se il bus SCSI fosse esteso al di fuori dell'involucro.

7.9 Posso usare PPA con i lettori di nastri paralleli della Iomega?

No. Questi sono lettori floppy-tape. Non c'entra lo SCSI.

Diversi hanno manifestato interesse nell'applicare il miei metodi per provare a determinare il protocollo e sviluppare il driver per questi tape. Il primo passo è riuscire ad avere il driver per DOS che funzioni sotto DOSemu. Non ho ancora notizia di qualcuno che abbia avuto successo nel farlo.

7.10 Il PPA funzionerà con il SyQuest EZ135 su porta parallela?

No. Il EZ135 è un disco IDE con un convertitore da parallela ad IDE ShuttlePort al suo interno.

Comunque è stato fatto un po' di lavoro su questo driver. Si veda

`http://www.torque.net/linux-pp.html`

8 Dove trovare ulteriori informazioni

Le versioni più recenti del programma di gestione per lo zip su porta parallela (ppa.c) e le informazioni tecniche correnti su tale driver possono essere trovare a

`http://www.torque.net/~campbell/`

Le informazioni su molti driver per dispositivi esterni su porta parallela possono essere trovate a

`http://www.torque.net/linux-pp.html`

Informazioni sulla condivisione della porta parallela e sulla porta parallela passante si trovano a

`http://www.cyberelk.demon.co.uk/parport.html`

Installazione di Linux da un lettore zip (per macchine senza un CD)

`http://metalab.unc.edu/mdw/HOWTO/mini/Install-From-ZIP.html`

Un'interfaccia X per il programma ziptool può essere trovata a

`http://www.scripps.edu/~jsmith/jazip/`

Informazioni sulla configurazione di Lilo le si può trovare a

`http://metalab.unc.edu/mdw/HOWTO/mini/LILO.html`

Informazioni sull'uso dei bootprompt le si può trovare a

`http://metalab.unc.edu/mdw/HOWTO/BootPrompt-HOWTO.html`

Se si vuole installare Linux sullo Zip ed eseguire il sistema dallo Zip

`http://metalab.unc.edu/LDP/HOWTO/mini/ZIP-Install.html`

Le pagine web della Iomega sono a

`http://www.iomega.com/` .

Se si cercano informazioni generiche sulla programmazione della porta parallela del proprio PC, non è male visitare

`http://www.lvr.com/parport.htm`