IDE e SoundBlaster 32 creative - HOWTO

Antonio Madesani <madesani@versilia.toscana.it> 0.2 del 18/8/1998

Informazioni sull'utilizzo ed il settaggio dell'interfaccia IDE presente su una scheda sonora SoundBlaster 32 Creative.

1 Differenze delle varie versioni di questo HOWTO:

v 0.1

le note considerano il settaggio della macchina dell'autore e questo crea qualche difficoltà ad utilizzarle in un caso generale

v 0.2

corretti gli inevitabili refusi, ho cercato di rendere più generale il discorso

v 0.3

corretti alcuni refusi e inserita una appendice per un caso particolare che si può presentare nel settaggio (a me è capitato)

2 Introduzione

Scopo di questo mini HOWTO è quello di fornire alcune informazioni che possano essere utili per poter utilizzare la porta IDE collegata ad una scheda sonora.

È necessario, prima di cominciare, fare una serie di precisazioni in modo da definire i limiti di queste note. L'autore è un semplice utente Linux e quindi non uno "specialist" e questo significa che le informazioni saranno limitate ad un ambito abbastanza ristretto.

Le informazioni si riferiscono ad una distribuzione RedHat 5.1 con kernel 2.0.35 non Red Hat. Questo discorso del kernel non RH nasce dal fatto che in un paio di occasioni mi sono trovato ad avere difficoltà in fase di compilazione con i kernel forniti insieme alla distribuzione ed allora ....

La scheda sonora che ho installato è una pnp Sound Blaster 32 della Creative Labs; mi pare che venga indicata anche come AWE 32. Personalmente non uso la scheda sonora se non per ascoltare, ogni tanto, un poco di musica mentre lavoro e quindi in queste note non sarà possibile trovare informazioni dettagliate sulla messa a punto della parte sonora. Ho avuto qualche difficoltà ad applicare le patch per la AWE 32 ed alla fine ho deciso di non applicarle in quanto non erano indispensabili. Il mio scopo era unicamente l'utilizzo della porta IDE sulla scheda.

Le versioni del software utilizzato sono:

- Distribuzione Red Hat 5.1 aggiornata 15/7/98
- kernel-2.0.35
- isapnptools-1.11-4
- Il lettore di CdRom è un IDE-ATAPI 24x.

La mia configurazione prevede tre HD, un masterizzatore Mitsumi 2801TE ed il lettore 24x.

3 Brevi note sulle IDE in Linux

Le schede madri, per lo meno le recenti, hanno due interfacce IDE che vengono gestite senza bisogno di starci molto a pensare: basta attaccarci qualche cosa opportunamente settato e tutto funziona. La cose possono complicarsi un pochino quanto si sfruttano le altre due collocate su schede esterne.

Le Ide sono:

```
Prima:    ide0, porta 0x1f0; major=3;  hda is minor=0; hdb is minor=64
Seconda:  ide1, porta 0x170; major=22; hdc is minor=0; hdd is minor=64
Terza:    ide2, porta 0x1e8; major=33; hde is minor=0; hdf is minor=64
Quarta:   ide3, porta 0x168; major=34; hdg is minor=0; hdh is minor=64
```

`/usr/src/linux/drivers/block/ide.c` controlla automaticamente la presenza di periferiche a queste interfacce ed assegna loro i relativi indirizzi e gli IRQ. Questo significa che, se non diversamente settato, ad una certa interfaccia corrisponde un certo indirizzo ed un certo irq.

Le porte standard ed i relativi irq sono:

```
ide0=0x1f0,0x3f6,14
ide1=0x170,0x376,15
ide2=0x1e8,0x3ee,11
ide3=0x168,0x36e,10
```

Notare che il primo parametro riserva 8 indirizzi contigui, mentre il secondo ne riserva uno solo. Per un approfondimento provare a dare una occhiata a `/proc/ioports`.

A questo punto possiamo cominciare a settare il nostro sistema. Considero ovviamente che la scheda sia stata correttamente installata e che, nel dubbio, sia possibile consultare i parametri che il costruttore consiglia per un suo funzionamento ottimale.

Il kernel, per lo meno quelli fino ai 2.0.XX, non hanno la possibilità di riconoscere automaticamente le schede pnp. Per riuscire a farglielo fare vi sono due strade:

1. patchare il kernel (strategia che non ho mai provato)
2. utilizzare isapnptools, un programma che consente di passare le corrette informazioni sulla scheda al sistema nella fase di boot.

Io ho utilizzato questa seconda strada ed è quella di cui si parlerà qui. Linux, per poter utilizzare una pnp, oltre ai servigi di isapnptools, vuole che la scheda sia caricata come modulo. Questo significa compilare opportunamente il kernel.

A questo scopo è necessario rispondere Y alle tre domande che si riferiscono alla parte che riguarda la gestione dei moduli.

```
*
* Loadable module support
*
Enable loadable module support (CONFIG_MODULES) [Y/n/?]
Set version information on all symbols for modules (CONFIG_MODVERSIONS) [Y/n/?]
Kernel daemon support (e.g. autoload of modules) (CONFIG_KERNELD) [Y/n/?]
*
```

Quando si arriva al suono:

```
*
* Sound
*
Sound card support (CONFIG_SOUND) [M/n/y/?]
```

si risponde con M e quindi si chiede di configurarlo come modulo.

A questo punto, la macchina lavoricchia un pochino e, se abbiamo precedentemente compilato il kernel con la richiesta della componente sonora, viene chiesto se vogliamo utilizzare la vecchia configurazione:

```
Old configuration exists in '/etc/soundconf'. Use it [Y/n/?]
```

Io, la prima volta ho preferito rispondere n in modo da essere sicuro di inserire i valori corretti.

Le domande a cui è necessario rispondere Y (in alcune distribuzioni vi può essere l'opzione M invece che Y) sono:

```
Sound Blaster (SB, SBPro, SB16, clones) support (CONFIG_SB) [Y/n/?]
Generic OPL2/OPL3 FM synthetizer support (CONFIG_ADLIB) [Y/n/?]
/dev/dsp and /dev/audio support (CONFIG_AUDIO)  [Y/n/?]
MIDI interface support (CONFIG_MIDI) [Y/n/?]
FM synthetizer (YM3812/OPL-3) support (CONFIG_YM3812) [Y/n/?]
lowlevel sound driver support [Y/n/?]
```

I valori predefiniti proposti per l'impostazione della scheda li ho accettati così come mi sono stati proposti. Nell'eventualità che possano sorgere conflitti, modificare quanto necessario. Orientativamente penso che l'unico valore che potrebbe creare problemi è la scelta dell'irq. La SB ama alla follia utilizzare 5 o 7.

Si compila il kernel, io uso:

```
# make zImage
```

poi si compilano e si installano i moduli:

```
# make modules
# make modules_install
```

Come probabilmente tutti sanno, se sono presenti i moduli della vecchia compilazione o del vecchio kernel, make install fa del casino e quindi è opportuno intervenire a mano, ovvero: prima di dare `make modules` e `make module_install`, è opportuno mettere da parte quello che può essere presente, a tale scopo:

```
# mv /lib/modules/2.0.XX /lib/modules/2.0.XX.old
```

XX indica il numero della versione del kernel (nel caso descritto potrebbe essere 35). Se questa directory non esiste, tanto meglio, significa che non avete precedentemente compilato il kernel con la richiesta di gestione moduli. Trasferito `zImage` in `/boot/` o dove siete abituati a metterlo, cambiato il nome secondo la solita metodica, lanciato lilo per avvertirlo che abbiamo un nuovo kernel, incrociamo le dita e facciamo:

```
# shutdown -r 0
```

Rilanciato, dovremmo avere la conferma che i moduli vengono visti e riconosciuti. Se così non fosse bisogna verificare il lavoro per individuare il punto dove qualche cosa è andato storto. Possono ovviamente aiutare i messaggi di errore che escono nella fase di boot.

Se poi il sistema non parte significa che il casino fatto supera il livello tolleranza ed allora si deve ripartire con il dischetto di emergenza, reinstallare il vecchio kernel che vi siete ben guardati dal buttare via e fare una verifica un poco più approfondita. Io suggerisco di tenere a portata di mano un blocco per appunti, quando si introducono dei cambiamenti in modo da ricostruire "esattamente" quello che si è fatto. Un modo per non arrivare al punto di non ritorno è mantenere le vecchie impostazioni rinominando i vecchi file per poterli rimettere al loro posto in caso problemi. Facciamo finta che tutto sia andato per il meglio, si tratta di vedere se la nostra scheda sonora viene regolarmente vista. A tale scopo:

```
# cat /dev/sndstat
```

dovrebbe venir fuori qualche cosa tipo:

```
# cat /dev/sndstat
Sound Driver:3.5.4-960630 (Sat Aug 1 22:35:15 CEST 1998 root,
Linux raptor.optitaly.it 2.0.35 #2 Sat Aug 1 22:11:56 CEST 1998 i586 unknown)
Kernel: Linux raptor.optitaly.it 2.0.35 #1 Sun Aug 2 12:25:33 CEST 1998 i586
Config options: a80002

Installed drivers:
Type 1: OPL-2/OPL-3 FM
Type 2: Sound Blaster
Type 7: SB MPU-401

Card config:
(Sound Blaster at 0x220 irq 5 drq 1,5)
(OPL-2/OPL-3 FM at 0x388 drq 0)

Audio devices:

Synth devices:

Midi devices:

Timers:
0: System clock

Mixers:
```

Se il sistema alla richiesta di "leggere" sndstat vi insulta significa, ovviamente che qualche cosa non è andato per il verso giusto.

Ricominciare il processo di verifica e leggere un poco di informazioni ("HOWTO come vi amo!") relative all'uso dei moduli.

Tanto che ci sono, consiglio di leggere quell'opera d'arte di: Appunti Linux del benemerito Daniele Giacomini al quale vanno tutti i nostri ringraziamenti e di consultare gli archivi del gruppo Pluto.

Se invece tutto procede con il vento in poppa si tratta di fare intervenire isapnptools. Se non è installato lo installiamo:

```
rpm -Uvh isapnptools-1.11-4
```

La versione potrebbe indubbiamente essere diversa; cercate di utilizzare quella più recente consigliata dalla vostra distribuzione. Se volete un consiglio, niente esperimenti con versioni sperimentali o di dubbia origine fino a quando tutto non funziona alla perfezione.

A questo punto lanciate:

```
# pnpdump > /etc/isapnp.conf
```

questo consente di avere il file completo di configuarazione con tutti i parametri predisposti ad essere settati sulla vostra macchina. I parametri sono tutti commentati con con il solito cancelletto e quindi ci basta "scommentare" quelli che vogliamo scegliere togliendo il cancelletto.

La parte che ci interessa è:

```
(CONFIGURE CTL0048/266633 (LD 1
#     Compatible device id PNP0600
#     ANSI string -->IDE<--

# Multiple choice time, choose one only !

#     Start dependent functions: priority preferred
#       IRQ 10.
#             High true, edge sensitive interrupt (by default)
# (INT 0 (IRQ 10 (MODE +E)))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0168
#             Maximum IO base address 0x0168
#             IO base alignment 1 bytes
#             Number of IO addresses required: 8
# (IO 0 (BASE 0x0168))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x036e
#             Maximum IO base address 0x036e
#             IO base alignment 1 bytes
#             Number of IO addresses required: 2
# (IO 1 (BASE 0x036e))

#       Start dependent functions: priority acceptable
#       IRQ 11.
#             High true, edge sensitive interrupt (by default)
 (INT 0 (IRQ 11 (MODE +E))) # da scommentare
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x01e8
#             Maximum IO base address 0x01e8
#             IO base alignment 1 bytes
#             Number of IO addresses required: 8
 (IO 0 (BASE 0x01e8)) # da scommentare
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x03ee
#             Maximum IO base address 0x03ee
#             IO base alignment 1 bytes
#             Number of IO addresses required: 2
 (IO 1 (BASE 0x03ee)) # da scommentare

#       Start dependent functions: priority acceptable
#       IRQ 10, 11, 12 or 15.
#             High true, edge sensitive interrupt (by default)
# (INT 0 (IRQ 10 (MODE +E)))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0100
#             Maximum IO base address 0x01f8
#             IO base alignment 8 bytes
```

```
#             Number of IO addresses required: 8
# (IO 0 (BASE 0x0100))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0300
#             Maximum IO base address 0x03fe
#             IO base alignment 2 bytes
#             Number of IO addresses required: 2
# (IO 1 (BASE 0x0300))

#       Start dependent functions: priority functional
#       IRQ 15.
#             High true, edge sensitive interrupt (by default)
# (INT 0 (IRQ 15 (MODE +E)))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0170
#             Maximum IO base address 0x0170
#             IO base alignment 1 bytes
#             Number of IO addresses required: 8
# (IO 0 (BASE 0x0170))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0376
#             Maximum IO base address 0x0376
#             IO base alignment 1 bytes
#             Number of IO addresses required: 1
# (IO 1 (BASE 0x0376))

#     End dependent functions
 (ACT Y)   # da scommentare
))
```

Le righe dove ho tolto il # che indica il commento le ho indicate. Mi raccomando di ricordare di togliere il commento alla riga `(ACT Y)`.

Un'altra cosa, rammentata nell'utile Sound Blaster AWE 32/64 HOWTO di Marcus Brinkmann (sarebbe utile leggerlo), è quella di verificare che in fondo alla sbrodolata di `isapnp.conf` sia presente la linea:

```
(WAITFORKEY)
```

che, a detta di Marcus, qualche volta è assente nelle vecchie versioni di `pnpdump`.

A questo punto un piccolo inciso. Sulla mia macchina, per qualche motivo che non conosco, la ide2 risulta impegnata e quindi non è disponibile per essere usata per attaccarci qualche cosa. Ho risolto il problema settando in `isapnp.conf` i valori corrispondenti alla ide3. Non dovrebbero esserci difficoltà a fare un cambiamento di questo tipo. L'unica cosa è ricordare che il cambiamento implica tenerne conto negli altri settaggi, compreso quello che vedremo dopo per individuare il nome della periferica che corrisponde al nostro cdrom. Fine dell'inciso.

A questo punto si tratta di sincronizzare gli interventi al momento del boot iniziale. Mi spiego.

Al momento del boot, prima è necessario che intervenga isapnptools con i suoi settaggi e poi può arrivare la gestione dei moduli.

Nel mio sistema questo è ottenuto tramite `/etc/rc.d/rc.sysinit` dove ci sono le due chiamate, nella sequenza corretta:

- gestione pnp:

```
# set up pnp
if [ -x /sbin/isapnp -a -f /etc/isapnp.conf ]; then
    if [ -n "$PNP" ]; then
        echo "Setting up ISA PNP devices"
        /sbin/isapnp /etc/isapnp.conf
    else
        echo "Skipping ISA PNP configuration at users request"
    fi
fi
```

- e successivamente quello per la gestione di moduli:

```
# Enter root and /proc into mtab.
mount -f /
mount -f /proc

if [ -f /proc/ksyms ]; then
    USEMODULES=y
else
    USEMODULES=
fi

# Get the modules ready to go -- we use awk here as cut is in /usr/bin
rm -f /lib/modules/preferred
if [ -n $USEMODULES ]; then
    set `cat /proc/cmdline`
    while [ $# -gt 0 ]; do
        if echo $1 | grep '^BOOT_IMAGE=' > /dev/null ; then
            image=`echo $1 | awk -F= '{ print $2 }'`
            kernelfile=`/sbin/lilo -I $image`
            if [ -n "$kernelfile" ]; then
                kernelname=`echo $kernelfile | awk -F- '{ print $1 }'`
                versioninfo=`echo $kernelfile | sed "s|${kernelname}-||"`
                if [ "$kernelname" = "/boot/vmlinuz" -a \
                     -d /lib/modules/$versioninfo -a \
                     $versioninfo != `uname -r` ]; then
                     ln -sf $versioninfo /lib/modules/preferred
                fi
            fi
        fi
        shift
    done
fi
```

Vi sono varie maniere per fare questo, e alcune distribuzioni, inseriscono automaticmente la parte che gestisce le schede pnp al momento del lancio del tool. Non buttatevi a ricopiare tutto, nell'eventualità che quello che ho riportato non sia uguale a ciò che avete nel vostro sistema, cercate di individuare le varie parti per capire se realmente è necessario modificare qualche cosa. Ancora una volta consiglio di leggere l'HOWTO di Marcus, citato prima, sulla SoundBlaster AWE.

Vediamo ora come vengono indicate le varie periferiche collegate.

```
ide0    master  hda     slave   hdb
ide1    master  hdc     slave   hdd
ide2    master  hde     slave   hdf
ide3    master  hdg     slave   hdf
```

A seconda che la periferica da collegare sia settata su master o slave, diventerà o hde oppure hdf. Una volta deciso, si crea un link in /dev

```
# ln -s /dev/hde cdrom (se è settata come master)
```

Può risultare utile, ma probabilmente non indispensabile, inserire un append in lilo:

```
append ="hde=cdrom"
```

con il quale gli dico che cdrom coincide con la periferica hde. A questo punto faccio partire lilo e poi rilancio:

```
# shutdown -r 0
```

Se tutto è andato bene dovrei in `/var/log/dmsg` trovare, all'inizio le informazioni che ho passato con l'append di lilo:

```
ide_setup: hde=cdrom
```

poi poco più sotto, il blocco delle informazioni delle periferiche ide. Nel mio caso:

```
hda: IBM-DJAA-31270, 1220MB w/96kB Cache, CHS=620/64/63
hdb: FUJITSU MPB3043ATU E, 4125MB w/0kB Cache, CHS=525/255/63
hdc: QUANTUM SIROCCO1700A, 1628MB w/75kB Cache, CHS=3309/16/63
hdd: CR-2801TE, ATAPI CDROM drive
hde: BCD-24X 1997-06-27, ATAPI CDROM drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
ide1 at 0x170-0x177,0x376 on irq 15
ide2 at 0x1e8-0x1ef,0x3ee on irq 11
```

se le cose stanno così non dovrebbero esserci problemi a fargli vedere il cdrom collegato alla ide della scheda sonora.

Se qualche cosa non dovesse andare per il verso giusto, provare a rilanciare e fate attenzione ai messaggi di boot per verificare che il settaggio della schede pnp avvenga prima del caricamento e riconoscimento delle dipendenze dei moduli.

Altro non so cosa dire ad eccezione che con una configurazione in cui il cdrom è collegato ad una scheda pnp succede che se fate un upgrade da dischetto, il lettore di cdrom non viene visto in quanto prevede tutta la trafila di cui abbiamo parlato. Per risolvere questo problema io utilizzo il masterizzatore collegato a ide1 (hdd) come semplice lettore.

La procedura fino a questo punto illustrata è quella che logicamente dovrebbe portare al risultato che vogliamo ottenere. Il problema che io ho avuto, e che mi ha costretto a rivedere alcune cose, riguarda ide2 che, sulla mia macchina, risulta già occupata da qualche cosa (mi piacerebbe tanto sapere che cosa) e quindi non consente di utilizzarla per attaccarci il nostro benedetto cdrom. In questo caso vi sono due cose da modificare:

- la prima riguarda il settaggio di isapnp.config;
- la seconda il fatto che la scheda che io uso (le informazioni sono sulle caratteristiche fornite dal produttore) pretende di avere come irq11.

Per quel che riguarda modificare isapnp.conf è estremamente intuitivo: si tolgono i commenti alla prima parte, quella che comincia con:

```
#     Start dependent functions: priority preferred
#       IRQ 10.
#             High true, edge sensitive interrupt (by default)
# (INT 0 (IRQ 10 (MODE +E)))
#       Logical device decodes 16 bit IO address lines
#             Minimum IO base address 0x0168
```

utilizzare quanto detto precedentemente per vedere le righe su cui intervenire. Si tratta però di introdurre anche la modifica dell'irq che deve essere cambiato da 10 a 11. Ovvero:

```
(INT 0 (IRQ 11 (MODE +E)))
```

se si decide di lasciare inalterata la seconda parte, quella relativa alla ide2, è necessario, in quella sostituire l'11 con il 10. In parole povere i due irq vanno invertiti sulle due porte. Io ho preferito rimettere in questa seconda parte i segni di commento (#) alle righe che a conti fatti non mi interessano. quelle relative ad ide2.

In caso di cambiamenti, ricordare di lasciare scommentato:

```
(ACT Y)
```

Cambia anche l'appende in lilo che diventa:

```
append ="ide2=noprobe ide3=0x168,0x36e,11 hdg=cdrom"
```

in /dev cambiare il link a cdrom che deve puntare ad hdg invece che hde. Io ho notato che ide2=noprobe non pare strettamente indispensabile ma evita una segnalazione nella fase di boot del sistema. Con questo settaggio interrogando dmesg ottengo:

all'inizio

```
ide_setup: ide2=noprobe
ide_setup: ide3=0x168,0x36e,11
ide_setup: hdg=cdrom
```

che sono le informazioni che ho passato tramite lilo. Per quanto riguarda le informazioni sulle periferiche ho:

```
hda: IBM-DJAA-31270, 1220MB w/96kB Cache, CHS=620/64/63
hdb: FUJITSU MPB3043ATU E, 4125MB w/0kB Cache, CHS=525/255/63
hdc: QUANTUM SIROCCO1700A, 1628MB w/75kB Cache, CHS=3309/16/63
hdd: CR-2801TE, ATAPI CDROM drive
hdg: BCD-24X 1997-06-27, ATAPI CDROM drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
ide1 at 0x170-0x177,0x376 on irq 15
ide3 at 0x168-0x16f,0x36e on irq 11
```

A questo punto penso proprio di aver fornito tutte le informazioni in mio possesso; attendo solo eventuali feedback.

4 Disclaimer

Questa è una versione che ha inevitabilmente bisogno di una messa a punto per togliere grossolani errori, inesattezze e quant'altro di inesatto o ambiguo.

Avrebbe bisogno anche di aggiunte da parte di chi conosce il comportamento di altre distribuzioni, altre configurazioni o altre situazioni che si possono presentare.

In ogni caso, usate queste informazioni con cautela, tenendo conto che vi sono ampie possibilità che siano presenti errori tali da compromettere la stabilità del sistema.

Attendo quindi consigli, correzioni, insulti ma non attribuitemi responsabilità di disastrose cadute dei vostri sistemi.

Antonio Madesani madesani@versilia.toscana.it