

Building a modular sound driver

The following information is current as of linux-2.1.85. Check the other readme files, especially README.OSS, for information not specific to making sound modular.

First, configure your kernel. This is an idea of what you should be setting in the sound section:

<M> Sound card support

<M> 100% Sound Blaster compatibles (SB16/32/64, ESS, Jazz16) support

I have SoundBlaster. Select your card from the list.

<M> Generic OPL2/OPL3 FM synthesizer support

<M> FM synthesizer (YM3812/OPL-3) support

If you don't set these, you will probably find you can play .wav files but not .midi. As the help for them says, set them unless you know your card does not use one of these chips for FM support.

Once you are configured, make zliilo, modules, modules_install; reboot. Note that it is no longer necessary or possible to configure sound in the drivers/sound dir. Now one simply configures and makes one's kernel and modules in the usual way.

Then, add to your /etc/modprobe.conf something like:

```
alias char-major-14-* sb
install sb /sbin/modprobe -i sb && /sbin/modprobe adlib_card
options sb io=0x220 irq=7 dma=1 dma16=5 mpu_io=0x330
options adlib_card io=0x388      # FM synthesizer
```

Alternatively, if you have compiled in kernel level ISAPnP support:

```
alias char-major-14 sb
post-install sb /sbin/modprobe "-k" "adlib_card"
options adlib_card io=0x388
```

The effect of this is that the sound driver and all necessary bits and pieces autoloading on demand, assuming you use kernel d (a sound choice) and autoclean when not in use. Also, options for the device drivers are set. They will not work without them. Change as appropriate for your card. If you are not yet using the very cool kernel d, you will have to "modprobe -k sb" yourself to get things going. Eventually things may be fixed so that this kludgery is not necessary; for the time being, it seems to work well.

Replace 'sb' with the driver for your card, and give it the right options. To find the filename of the driver, look in /lib/modules/<kernel-version>/misc. Mine looks like:

```
adlib_card.o # This is the generic OPLx driver
opl3.o # The OPL3 driver
```

README.modules.txt

sb.o # <<The SoundBlaster driver. Yours may differ.>>
sound.o # The sound driver
uart401.o # Used by sb, maybe other cards

Whichever card you have, try feeding it the options that would be the default if you were making the driver wired, not as modules. You can look at function referred to by `module_init()` for the card to see what args are expected.

Note that at present there is no way to configure the io, irq and other parameters for the modular drivers as one does for the wired drivers.. One needs to pass the modules the necessary parameters as arguments, either with `/etc/modprobe.conf` or with command-line args to `modprobe`, e.g.

```
modprobe sb io=0x220 irq=7 dma=1 dma16=5 mpu_io=0x330
modprobe adlib_card io=0x388
```

recommend using `/etc/modprobe.conf`.

Persistent DMA Buffers:

The sound modules normally allocate DMA buffers during `open()` and deallocate them during `close()`. Linux can often have problems allocating DMA buffers for ISA cards on machines with more than 16MB RAM. This is because ISA DMA buffers must exist below the 16MB boundary and it is quite possible that we can't find a large enough free block in this region after the machine has been running for any amount of time. The way to avoid this problem is to allocate the DMA buffers during module load and deallocate them when the module is unloaded. For this to be effective we need to load the sound modules right after the kernel boots, either manually or by an init script, and keep them around until we shut down. This is a little wasteful of RAM, but it guarantees that sound always works.

To make the sound driver use persistent DMA buffers we need to pass the `sound.o` module a `"dmabuf=1"` command-line argument. This is normally done in `/etc/modprobe.conf` like so:

```
options sound          dmabuf=1
```

If you have 16MB or less RAM or a PCI sound card, this is wasteful and unnecessary. It is possible that machine with 16MB or less RAM will find this option useful, but if your machine is so memory-starved that it cannot find a 64K block free, you will be wasting even more RAM by keeping the sound modules loaded and the DMA buffers allocated when they are not needed. The proper solution is to upgrade your RAM. But you do also have this improper solution as well. Use it wisely.

I'm afraid I know nothing about anything but my setup, being more of a text-mode guy anyway. If you have options for other cards or other helpful hints, send them to me, Jim Bray, jb@as220.org, <http://as220.org/jb>.