

Linux系统中用ALSA驱动声卡流程详解 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/245/2021_2022_Linux_E7_B3_BB_E7_BB_c103_245862.htm

一、什么是ALSA Advanced Linux Sound Architecture 的简称为 ALSA ，译成中文的意思是 Linux 高级声音体系（这是我直译的，可能译的不对）。一谈到体系就有点范围就太大了，所以ALSA不仅仅是包括对声卡的支持和驱动。ALSA具有如下特征：小虫网络技术 1、对所有音频接口的高效支持，从普通用户的声卡到专业级别多路音频设备。 2、声卡驱动完全模块化设计。 3、SMP and thread-safe design。 4、开发库（alsa-lib）为程序设计提供了简单、方便，并且拥有有高级的效果和功能。 5、支持旧版本的OSS API 接口，能为大多数的OSS应用程序提供兼容。OSS是一个商业性的驱动，OSS有一个简装版的代码已经移入内核和ALSA，其中alsa-oss就是。OSS公司据说目前已经并不存在了。我们没有必要用OSS公司提供的商业版本。用ALSA和OSS简装版足够。二、关于硬件驱动驱动的必备基础 1、如何查看硬件芯片在Linux操作系统中，所有的硬件都是以芯片组来区分的，品牌并不是最重要的。硬件最重要的标识是芯片组。所以您在讨论区求助的时候，只说硬件品牌，而不提供芯片组，大家是帮助不了您的，切记。我们查看硬件的芯片组是的命令是 `lspci -v` 或者是 `dmesg`，由于 `dmesg` 输出的信息不太多，不够直观。所以经常用的还是 `lspci -v`。也可以用 `lshw` 获取。最方便的还是 `lspci -v`。初学者还是用 `lspci -v` 更好一点。

[root@localhost beinan]# `lspci -v`我们运行 `lspci -v` 后，如果查看声卡芯片组，发会现有类似下面的一段： [root@localhost

```
beinan]# lspci -v Multimedia audio controller: Intel Corporation
82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M)AC97 Audio
Controller (rev 03) Subsystem: Hewlett-Packard Company:
Unknown device 3080 Flags: bus master, medium devsel, latency 0,
IRQ 5 I/O ports at 1c00 size=256 I/O ports at 18c0 size=64 Memory
at e0100c00 (32-bit, non-prefetchable) size=512 Memory at
e0100800 (32-bit, non-prefetchable) size=256 Capabilities: [50]
Power Management version 22、系统内核版本 [root@localhost
beinan]# uname -r -m -p -i 2.6.11-1.1369_FC4 i686 i686 i386
```

上面的表示的是系统的内核版本，处理器架构等。提示：如果您自己编译内核，还要安装kernel-devel（或kernel-source），这个是在系统光盘或者映像文件中有带。如果您想自己用kernel.org下载内核，自己编译，然后再来驱动声卡，也可以到kernel.org上下载最新的内核源码。一般情况下，如果系统默认的内核不能驱动声卡，大多得更换内核，我们建议您先从发行版所提供的升级包来升级内核及源码。比如Fedora的扩展项目软件包更新速度极快，我们能通过apt和yum进行kernel的升级。在启用新内核时，我们不建议您删除老内核。因为现在Fedora/Redhat引入apt和yum更新工具，系统的完整性极为重要。如果系统被破坏，用更新工具变得极为困难。比如Fedora的apt和yum升级源升级的内核也解决不了您的问题，我们可能需要自己通过kernel.org下载内核来编译。但也不能删除以前的内核包及源码包。一是为了安全，毕竟我们自己编译内核不是百分之百的成功。另外系统自带的内核及源码包被系统很多软件依赖。apt和yum每次更新软件包的时候都要检查系统的完整性。对于有些发行版，可以直接

从从 kernel.org 下载最新的内核。如果从系统提供的内核源码（kernel-devel或kernel-source）编译，可能解决不了您的问题。

3、modprobe.conf的内容。在Fedora Core 4.0中，内核模块的添加或定义别名是在/etc/modprobe.conf文件。在其它版本可能是modules.conf。如果您的系统中存在modprobe.conf，我们就以这个文件为准。不同发行版有不同的定义文件，比如slackware是定义在/etc/modules.conf中，但也要在/etc/rc.d/rc.modules打开相关驱动模块。modprobe.conf或者module.conf就是对系统已经加载的模块进行相应的配置，比如设置别名等。这些一般都是自通过工具自动生成的，我们也可以通过查看硬件的文档和站点，也可以自己添加。为什么这样添加，而不是那样的呢？我也不懂了，因为人家文档那样说的，我不会开发，如果懂的开发，可能会明白。请理解理解我。比如我用的是855的主板，系统驱动用的是snd-intel8x0，在/etc/modprobe.conf的配置是如下的内容。下面这段内容是通过alsaconf配置工具自动生成的。

```
alias
snd-card-0 snd-intel8x0options snd-card-0 index=0options
snd-intel8x0 index=0remove snd-intel8x0 { /usr/sbin/alsactl store 0
>/dev/null 2>& .1 || : . }. /sbin/modprobe -r --ignore-remove
snd-intel8x04、lsmod 查看已经加载的内核模块。如果一个设备在内核的编译中是以模块方式加载的，主要是通过lsmod来查看。
```

[root@localhost beinan]# lsmod可能初学Linux的弟兄会问，内核的module存放在哪里，我们能不能自己来手工加载模块。内核的模块放在/lib/modules/内核的版本/kernel/，比如在Fedora 4.0中，我们可以看到如下的：

小虫网络技术

```
[root@localhost beinan]# ls
```

/lib/modules/2.6.11-1.1369_FC4/kernel/arch/ crypto/ drivers/ fs/
lib/ net/ sound/其中，sound 就是声卡模块存放目录，大多数的
硬件驱动放在 drivers目录，fs 是文件系统模块的目录。net
是与网络有关的存放目录，比如一些网络协议支持的模块、
防火墙支持的模块等。arch 是cpu方面 ... 如果我们想自己加
载模块，就到这些目录中查看相应模块的信息，然后用
modprobe 来加载。 [root@localhost beinan]# modinfo
snd-intel8x0 查看一个模块的信息，我们用 modinfo 来查看，
所要查看的模块不要带 .ko 或者.o 之类的。比如查看到类似下
面的信息。 description: Intel
82801AA,82901AB,i810,i820,i830,i840,i845,MX440. SiS 7012. Ali
5455通过这个我们是可以知道这是Intel 集成声卡，通过lspci -v
得到的声卡信息，感觉他们很相近。所以就能尝试用这个模
块来驱动。 [root@localhost beinan]# modprobe snd-intel8x0用
modprobe 加载了模块，然后我们通过 lsmod 就能看到了。对
于声卡模块是这么加载的，其它设备的驱动模块也是如此。
100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com