# The Linux Sound HOWTO

Jeff Tranter, Jeff_Tranter@Mitel.COM v1.6, 6 June 1994

This document describes sound support for Linux. It lists the supported sound hardware, describes how to configure the kernel drivers, and answers frequently asked questions. The intent is to bring new users up to speed more quickly and reduce the amount of traffic in the usenet news groups.

## Contents

**1 Introduction** **1**

1.1 Acknowledgments . . . 1

1.2 Revision History . . . 1

1.3 New versions of this document . . . 2

1.4 Feedback . . . 2

1.5 Other Sources of Information . . . 2

1.6 Version Information . . . 3

**2 Supported Sound Hardware** **3**

2.1 Sound Cards . . . 3

2.2 PC Speaker . . . 4

2.3 Parallel Port . . . 4

**3 Configuring Linux for Sound Support** **4**

3.1 Installing the Sound Card . . . 4

3.2 Configuring the Kernel . . . 5

3.3 Creating the Device Files . . . 5

3.4 Testing the Installation . . . 5

**4 Applications Supporting Sound** **7**

**5 Answers To Frequently Asked Questions** **7**

5.1 What are the various sound device files? . . . 7

5.2 How can I play a sound sample? . . . 8

5.3 How can I record a sample? . . . 8

5.4 Can I have more than one sound card? . . . 8
5.5 Error: `No such file or directory` for sound devices . . . 9
5.6 Error: `No such device` for sound devices . . . 9
5.7 Error: `No space left on device` for sound devices . . . 9
5.8 Error: `device busy` for sound devices . . . 9
5.9 I still get `device busy` errors! . . . 9
5.10 Partial playback of digitized sound file . . . 10
5.11 There are pauses when playing MOD files . . . 10
5.12 Compile errors when compiling sound applications . . . 10
5.13 SEGV when running sound binaries that worked previously . . . 10
5.14 What known bugs or limitations are there in the sound driver? . . . 10
5.15 What version of the sound driver I should use? . . . 11
5.16 What do all the sound driver configuration options mean? . . . 11
5.17 What future enhancements are planned for the sound driver? . . . 12
5.18 Where are the sound driver ioctls() etc. documented? . . . 12
5.19 What CPU resources are needed to play or record without pauses? . . . 12
5.20 Problems with a PAS16 and an Adaptec 1542 SCSI host adaptor . . . 13
5.21 Problems with the FM synthesizer on a SoundBlaster Pro 1 . . . 14
5.22 Is the GUS-MAX supported? . . . 14
5.23 What if my sound card is not supported? . . . 14
5.24 Is it possible to read and write samples simultaneously? . . . 14

# 1 Introduction

This is the Linux Sound HOWTO document. It is intended as a quick reference covering everything you need to know to install and configure sound support under Linux. Frequently asked questions about sound under Linux are answered, and references are given to some other sources of information on a variety of topics related to computer generated sound and music.

The scope is limited to the aspects of sound cards pertaining to Linux. See the other documents listed in the *Other Sources of Information* section for more general information on sound cards and computer sound and music generation.

## 1.1 Acknowledgments

Much of this information came from the `Readme` files provided with the sound driver source code, by Hannu Savolainen (`hannu@voxware.pp.fi`). Thanks go to Hannu and the many other people who developed the Linux kernel sound drivers and utilities.

Thanks to Matt Welsh's Linuxdoc-SGML package, this HOWTO is now available in several formats, all generated from a common source file.

## 1.2 Revision History

**Version 1.1**

first version; posted to SOUND channel of Linux activists mailing list only

**Version 1.2**

minor updates; first version available on archive sites

**Version 1.3**

converted to SGML; now available in several formats using Matt Welsh's Linuxdoc-SGML tools; appearance changed due to new format, only minor changes to content

**Version 1.4**

minor tweaking of SGML; added answer on PAS16 and Adaptec1542A SCSI adaptor incompatibilities

**Version 1.5**

2.5a sound driver is now in 1.1 kernel distribution; note on GUS-MAX support; other minor updates

**Version 1.6**

added info on "no space on device" error; added note that Hacker's Guide is in a "hidden" directory; added question on bidirectional mode; info on "device busy" errors; other minor changes

## 1.3 New versions of this document

New versions of this document will be periodically posted to `comp.os.linux.announce`. They will also be uploaded to various anonymous ftp sites that archive such information including `sunsite.unc.edu:/pub/Linux/docs/HOWTO`.

## 1.4 Feedback

If you have any suggestions, corrections, or comments on the HOWTO, please send them to the author and I will try to incorporate them in the next release.

## 1.5 Other Sources of Information

*The Linux Sound User's Guide* covers all of the user visible aspects of using sound under Linux in much more detail (approximately 40 pages). If you are interested in sound under Linux you should definitely get this document. The current version is ALPHA 0.1, and is available on `tsx-11.mit.edu` in the directory `/pub/linux/ALPHA/LDP`. I will continue to maintain the Sound-HOWTO as a concise guide for users who want to get sound up and running, or just find out what is required, without having to read the full user's guide.

Hannu Savolainen has written a draft version of the *Hacker's Guide to VoxWare*. The latest version is draft 2, and can be found on `nic.funet.fi` in `/pub/OS/linux/ALPHA/sound`.

The following FAQs are regularly posted to the usenet newsgroup `news.announce` as well as being archived at the site `rtfm.mit.edu` in the directory `/pub/usenet/news.answers`:

```
PCsoundcards/generic-faq (Generic PC Soundcard FAQ)
PCsoundcards/soundcard-faq (comp.sys.ibm.pc.soundcard FAQ)
PCsoundcards/gravis-ultrasound/faq (Gravis UltraSound FAQ)
audio-fmts/part1 (Audio file format descriptions)
audio-fmts/part2 (Audio file format descriptions)
```

The FAQs also list several product specific mailing lists and archive sites. The following Usenet news groups discuss sound and/or music related issues:

```
alt.binaries.sounds.* (various groups for posting sound files)
alt.binaries.multimedia (discussions on Multimedia)
alt.sb.programmer (Soundblaster programming topics)
comp.multimedia (Multimedia topics)
comp.music (Computer music theory and research)
comp.sys.ibm.pc.soundcard (IBM PC soundcard topics)
```

The Linux Activists mailing list has a SOUND channel. To find out how to join the mailing list, send mail to `linux-activists-request@joker.cs.hut.fi`.

The files `Readme`, `Readme.linux`, and `CHANGELOG` included with the kernel sound driver source code contain useful information about the sound card drivers. These can typically be found in the directory `/usr/src/linux/drivers/sound`.

The *Linux Software Map* (LSM) is an invaluable reference for locating Linux software. Searching the LSM for keywords such as *sound* is a good way to identify applications related to sound hardware. The LSM can be found on various anonymous FTP sites, including `sunsite.unc.edu:/pub/Linux/docs/LSM.gz`.

## 1.6 Version Information

At time of writing the latest Linux sound driver was version 2.5a, and was included in the Linux kernel version 1.1.10 and later. You can also obtain the 2.5 sound driver separately, and apply it as a patch to earlier kernels (such as 1.0).

# 2 Supported Sound Hardware

## 2.1 Sound Cards

The following sound cards are supported by the Linux kernel:

- Roland MPU-401 MIDI interface
- AdLib
- SoundBlaster (version 1 and 2) and compatibles, including ThunderBoard and ATI Stereo F/X
- SoundBlaster Pro (version 1 and 2)
- SoundBlaster 16
- ProAudioSpectrum 16 (and the compatible Logitech SoundMan 16)
- Advanced Gravis UltraSound (GUS)

Other sound cards that are claimed to be compatible with one of the supported sound cards *may* work if they are hardware (i.e. register level) compatible.

The Sound Galaxy NX Pro is supported as a SoundBlaster compatible, provided that you add `#define __SGNXPRO__` to the sound driver `local.h` file.

The Linux kernel supports the SCSI port provided on some sound cards (e.g. ProAudioSpectrum 16). There is also support for CD-ROM drives attached to the Soundblaster Pro and SoundBlaster 16 CD-ROM port (see the file `/usr/src/linux/drivers/block/README.sbpcd`).

A kernel patch to support joystick ports, including those provided on some sound cards, is also available.

## 2.2 PC Speaker

An alternate sound driver is available that requires no additional sound hardware; it uses the internal PC speaker. It is mostly software compatible with the sound card driver, but, as might be expected, provides much lower quality output and has much more CPU overhead. The results seem to vary, being dependent on the characteristics of the individual loudspeaker. For more information, see the documentation provided with the release.

The current version is 0.6, and can be found at site `sunsite.unc.edu` in the file `pub/Linux/kernel/misc-patches/pcsndrv-0.6.tar.z`.

## 2.3 Parallel Port

Another option is to build a digital to analog converter using a parallel printer port. This provides better sound quality but still has a lot of CPU overhead. The PC sound driver package mentioned above supports this, and includes instructions for building the necessary hardware.

# 3 Configuring Linux for Sound Support

Configuring Linux to support sound involves the following steps:

1. Installing the sound card.
2. Configuring and building the kernel for sound support.
3. Creating the device files.
4. Testing the installation.

## 3.1 Installing the Sound Card

To install the card, follow the instructions provided by the manufacturer. Be sure to note down the jumper settings for IRQ, DMA channel, etc. If you are unsure, use the factory defaults. Try to avoid conflicts with other devices (e.g. ethernet cards, SCSI host adaptors) if possible.

## 3.2 Configuring the Kernel

If you are using a recent kernel (0.99pl14 or later), the sound drivers are included with the kernel release. Follow the usual procedure for building the kernel. When you run `make config`, a configuration program will ask you what sound card options you want. Carefully read the information displayed by this program.

If you are upgrading from an older sound driver, make sure that the files `/usr/include/sys/soundcard.h` and `/usr/include/sys/ultrasound.h` are symbolic links to the corresponding files in `/usr/include/linux`.

It's good idea to read the `Readme` files in the kernel `drivers/sound` directory since there could be some last minute information. The file `CHANGELOG` contains a list of enhancements and new features since the previous version.

## 3.3 Creating the Device Files

The first time the kernel sound driver is configured, you need to create the sound device files. The easiest way to do this is to cut the short shell script from the end of the file `Readme.linux` (or possibly `Readme`) in the directory `/usr/src/linux/drivers/sound`, and run it as root.

If your device entries already exist, you might want to ensure they are correct, e.g. `/dev/audio` should have major and minor device numbers 14 and 4. If they are not, or if you are in doubt, run the above script and it will replace any existing entries with correct ones.

Some older Linux distributions provided install scripts which created incorrect sound device files. You may also have a `/dev/MAKEDEV` script for creating device files. Using the script included with the kernel sound driver is preferred since it should always be up to date with the latest supported sound devices.

## 3.4 Testing the Installation

You can now follow these steps to verify the sound hardware and software:

1. Reboot with the new kernel.

Follow your usual procedure for installing and rebooting the new kernel (keep the old kernel around in case of problems, of course).

2. Verify that the sound card is recognized during kernel initialization.

Check for a message such as the following on powerup (if they scroll by too quickly to read, you may be able to retrieve them with the "dmesg" command):

```
snd2 <SoundBlaster Pro 3.2> at 0x220 irq 5 drq 1
snd1 <Yamaha OPL-3 FM> at 0x388 irq 0 drq 0
```

This should match your sound card type and jumper settings.

The driver may also display some error messages and warnings during boot. Watch for these when booting the first time after configuring the sound driver.

If no sound card is detected when booting, there are two possible reasons:

- The configuration of the driver is incorrect and the driver was not able to detect your card in the given I/O address, or
- The sound driver was configured to be inactive or you booted with an old kernel (a common error).

3. Check device file `/dev/sndstat`.

Reading the sound driver status device file should provide additional information on whether the sound card driver initialized properly. Sample output should look something like this:

```
% cat /dev/sndstat

Sound Driver:2.5 (Wed Apr 20 19:57:25 EDT 1994 root@fizzbin)
Config options: 1aa2

HW config:
Type 2: SoundBlaster at 0x220 irq 5 drq 1
Type 1: AdLib at 0x388 irq 0 drq 0

PCM devices:
0: SoundBlaster Pro 3.2

Synth devices:
0: Yamaha OPL-3

Midi devices:
0: SoundBlaster

Mixer(s) installed
```

If the cat command displays "No such device", the sound driver is not active in the kernel. Make sure that you booted with the newly compiled kernel.

If the printout contains no devices (PCM, Synth or MIDI), your soundcard was not detected. Verify that the "HW config" section contains correct information.

4. Play a simple sound file.

Get hold of a sample sound file, and send it to the sound device as a basic check of sound output, e.g.

```
% cat endoftheworld >/dev/dsp
% cat crash.au >/dev/audio
```

Some sample sound files can be obtained from the file `snd-data-0.1.tar.Z`.

5. Verify sound recording.

If you have sound input capability, you can do a quick test of this using commands such as the following:

```
# record 4 seconds of audio from microphone
% dd bs=8k count=4 </dev/audio >sample.au
# play back sound
% cat sample.au >/dev/audio
```

If these tests pass, you can be reasonably confident that the sound hardware and software are working. If you experience problems, read the FAQ section of this document.

# 4 Applications Supporting Sound

Because *The Linux Sound User's Guide* describes the available Linux applications in detail, I will only give here a sample of the types of applications that you likely want if you have a sound card under Linux.

As a minimum, you will likely want to obtain the following sound applications:

- audio file format conversion utility (e.g. `Sox`)
- mixer utility (e.g. `aumix` or `xmix`)
- digitized file player/recorder (e.g. `play`)
- MOD file player (e.g. `tracker`)
- MIDI file player (e.g. `mp`)

There are text-based as well as GUI-based versions of most of these tools. There are also some more esoteric applications (e.g. speech synthesis) that you may wish to try.

# 5 Answers To Frequently Asked Questions

This section answers some of the questions that have been commonly asked on the Usenet news groups and mailing lists.

## 5.1 What are the various sound device files?

**/dev/audio**

Sun workstation compatible audio device (only a partial implementation, does not support Sun ioctl interface, just u-law encoding)

**/dev/dsp**

digital sampling device

**/dev/mixer**

sound mixer

**/dev/sequencer**

MIDI, FM, and GUS access

**/dev/midi**

MIDI device (not yet implemented in current sound driver)

**/dev/sndstat**

displays sound driver status when read

**/dev/audio1**

for second sound card

**/dev/dsp1**

for second sound card

The PC speaker driver provides the following devices:

**/dev/pcaudio**

equivalent to /dev/audio

**/dev/pcsp**

equivalent to /dev/dsp

**/dev/pcmixer**

equivalent to /dev/mixer

## 5.2 How can I play a sound sample?

Sun workstation (.au) sound files can be played by sending them to the `/dev/audio` device. Raw samples can be sent to `/dev/dsp`. Using a program such as `play` is preferable, as it will recognize most file types and set the sound card to the correct sampling rate, etc.

## 5.3 How can I record a sample?

Reading `/dev/audio` or `/dev/dsp` will return sampled data that can be redirected to a file. A program such as `vrec` makes it easier to control the sampling rate, duration, etc. You may also need a mixer program to select the appropriate input device.

## 5.4 Can I have more than one sound card?

Up to two sound cards is supported. It's possible to install a Gravis UltraSound or MPU-401 with a SoundBlaster, SoundBlaster Pro, SoundBlaster16 or ProAudioSpectrum16. It's *not* possible to have a ProAudioSpectrum16 and SoundBlaster at the same time (the PAS16 has an SB emulator in it). It's also not possible to have more than one card of the same type at the same time – for example, a GUS + GUS combination is not possible.

You can change the sound card configuration parameters at boot time using command line options from a boot loader such as LILO. See the kernel sound driver file `Readme.linux` for details.

## 5.5 Error: `No such file or directory` for sound devices

You need to create the sound driver device files. See the section on creating device files. If you do have the device files, ensure that they have the correct major and minor device numbers (some older CD-ROM distributions of Linux may not create the correct device files during installation).

## 5.6 Error: `No such device` for sound devices

You have not booted with a kernel containing the sound driver or the I/O address configuration doesn't match your hardware. Check that you are running the newly compiled kernel and verify that the settings entered when configuring the sound driver match your hardware setup.

## 5.7 Error: `No space left on device` for sound devices

This can happen if you tried to record data to /dev/audio or /dev/dsp without creating the necessary device file. The sound device is now a regular file, and has filled up your disk partition. You need to run the script described in the *Creating the Device Files* section of this document.

## 5.8 Error: `device busy` for sound devices

Only one process can open a given sound device at one time. Most likely some other process is using the device in question. One way to determine this is to use the fuser command:

```
% fuser -v /dev/dsp
/dev/dsp:             USER       PID ACCESS COMMAND
                      tranter    265 f....  tracker
```

In the above example, the fuser command showed that process 265 had the device open. Waiting for the process to complete or killing it will allow the sound device to be accessed once again.

## 5.9 I still get `device busy` errors!

(the following explanation was supplied by Harald Albrecht `albrecht@igpm.rwth-aachen.de`)

Some soundcards support using DMA channel 0. The sound driver configuration program allows this, and the kernel compiles properly, but accessing the sound device results in a "device busy" error message.

The reason is that the Linux kernel reserves DMA channel 0 for DRAM refresh. This is no longer true for modern 386/486 boards which use their own refresh logic. You can correct it by changing this line in the file /usr/src/linux/kernel/dma.c:

```
static volatile unsigned int dma_chan_busy[MAX_DMA_CHANNELS] = {
                1, 0, 0, 0, 1, 0, 0, 0
};
```

Replace the first 1 with a 0; this enables DMA channel 0. Don't do the same with DMA channel 4 as this is cascade and won't work! The code should now look like this:

```
static volatile unsigned int dma_chan_busy[MAX_DMA_CHANNELS] = {
                0, 0, 0, 0, 1, 0, 0, 0
};
```

Recompile and reboot with the new kernel.

## 5.10 Partial playback of digitized sound file

The symptom is usually that a sound sample plays for about a second and then stops completely or reports an error message about "missing IRQ" or "DMA timeout". Most likely you have incorrect IRQ or DMA channel settings. Verify that the kernel configuration matches the sound card jumper settings and that they do not conflict with some other card.

## 5.11 There are pauses when playing MOD files

Playing MOD files requires considerable CPU power. You may have too many processes running or your computer may be too slow to play in real time. Your options are to:

- try playing with a lower sampling rate or in mono mode
- eliminate other processes
- buy a faster computer
- buy a more powerful sound card (e.g. Gravis UltraSound)

## 5.12 Compile errors when compiling sound applications

The version 1.0c and earlier sound driver used a different and incompatible `ioctl()` scheme. Obtain newer source code or make the necessary changes to adapt it to the new sound driver. See the sound driver `Readme` file for details.

Also ensure that you have used the latest version of `soundcard.h` and `ultrasound.h` when compiling the application. See the installation instructions at beginning of this text.

## 5.13 SEGV when running sound binaries that worked previously

This is probably the same problem described in the previous question.

## 5.14 What known bugs or limitations are there in the sound driver?

See the `Readme` and `CHANGELOG` files included with the sound driver kernel source.

## 5.15 What version of the sound driver I should use?

If you are using version 1.0c or earlier, you definitely need to upgrade. Version 1.0c is not compatible with the applications written for version 2.0 or later.

There have been no significant changes after version 2.0, so if you don't have problems and that particular version fulfills your requirements, there are no compelling reasons to move to a more recent version (this should be true at least until September 1994).

The latest official version is in the latest Linux kernel distribution. There may also be some test and prototype versions lying around. If the version number is smaller than 2.9, the version should be quite safe. Any driver release having a version number of the form 2.99.XX is an incompletely implemented and experimental test release.

If you are interested in development of the sound driver, join the linux activists SOUND channel.

## 5.16 What do all the sound driver configuration options mean?

During configuration of the sound driver, a `configure` program is compiled and executed. This program asks you some questions and then generates the header file `local.h` that defines the sound card configuration.

The configuration file defines (or undefines) the following symbols:

```
Symbol                  Meaning
======                  =======
KERNEL_SOUNDCARD        enable/disable sound driver
CONFIGURE_SOUNDCARD     enable/disable sound driver
EXCLUDE_PAS             ProAudioSpectrum support
EXCLUDE_SB              SoundBlaster support
```

```
EXCLUDE_ADLIB           AdLib support
EXCLUDE_GUS             Gravis UltraSound support
EXCLUDE_MPU401          MPU-401 MIDI interface support
EXCLUDE_SBPRO           SoundBlaster Pro support
EXCLUDE_SB16            SoundBlaster 16 support
EXCLUDE_AUDIO           Digitized voice support
EXCLUDE_MIDI            MIDI interface support
EXCLUDE_YM3812          FM synthesizer (YM3812/OPL-3) support
EXCLUDE_SEQUENCER       MIDI sequencer support
EXCLUDE_PRO_MIDI        SoundBlaster Pro MIDI support
EXCLUDE_CHIP_MIDI       MIDI on CHIP support
SBC_IRQ                 SoundBlaster IRQ number
SBC_DMA                 SoundBlaster DMA channel
SB16_DMA                SoundBlaster 16 DMA channel
SB16_MIDI_BASE          base address of SoundBlaster 16 MIDI port
PAS_IRQ                 ProAudioSpectrum IRQ number
PAS_DMA                 ProAudioSpectrum DMA channel
GUS_IRQ                 Gravis UltraSound IRQ number
GUS_DMA                 Gravis UltraSound DMA channel
GUS_BASE                base address of Gravis UltraSound
MPU_IRQ                 MPU-401 IRQ number
MPU_BASE                base address of MPU-401 port
DSP_BUFFSIZE            DMA buffer size
```

Several other defines are also created, setting such things as the sound driver revision level and the time and date when configure was run.

There are other parameters that are not set by the configure program (e.g. SoundBlaster base address). If you need to change these, edit the file `sound_config.h`.

To disable the sound driver, run `make config` and answer "no" to the "Sound card support?" question.

## 5.17 What future enhancements are planned for the sound driver?

The sound driver is not just for Linux, it also supports several other Intel-based Unix operating systems. The package is now called "VoxWare". Some of the enhancements being considered are:

- implementing full MIDI support
- patch manager support
- document sound card driver (Hacker's Guide)
- support for new sound cards
- miscellaneous bug fixes

## 5.18 Where are the sound driver ioctls() etc. documented?

These are documented in the *Hacker's Guide to VoxWare*, currently available in draft form. The latest version is draft 2, and can be found on `nic.funet.fi` in `/pub/OS/linux/ALPHA/sound`. Note that this directory is "hidden" and will not appear in directory listings. If you "cd" to the directory and use the FTP "dir" command, the files *are* there.

## 5.19 What CPU resources are needed to play or record without pauses?

There is no easy answer to this question, as it depends on:

- whether using PCM sampling or FM synthesis
- sampling rate and sample size
- which application is used to play or record
- Sound Card hardware
- disk I/O rate, CPU clock speed, cache size, etc.

In general, any 386 machine should be able to play samples or FM synthesized music on an 8 bit soundcard with ease.

Playing MOD files, however, requires considerable CPU resources. Some experimental measurements have shown that playing at 44kHz requires more than 40% of the speed of a 486/50 and a 386/25 can hardly play faster than 22 kHz (these are with an 8 bit card sound such as a SoundBlaster). A card such as the Gravis UltraSound card performs more functions in hardware, and will require less CPU resources.

These statements assume the computer is not performing any other CPU intensive tasks.

Converting sound files or adding effects using a utility such as Sox is also much faster if you have a math coprocessor. The kernel driver itself does not do any floating point calculations, though.

## 5.20 Problems with a PAS16 and an Adaptec 1542 SCSI host adaptor

(the following explanation was supplied by `seeker@indirect.com`)

Linux only recognizes the 1542 at address 330 (default) or 333, and the PAS only allows the MPU-401 emulation at 330. Even when you disable the MPU-401 under software, something still wants to conflict with the 1542 if it's at its preferred default address. Moving the 1542 to 333 makes everyone happy.

Additionally, both the 1542 and the PAS-16 do 16-bit DMA, so if you sample at 16-bit 44KHz stereo and save the file to a SCSI drive hung on the 1542, you're about to have trouble. The DMAs overlap and there isn't enough time for RAM refresh, so you get the dread "PARITY ERROR - SYSTEM HALTED" message, with no clue to what caused it. It's made worse because a few second-party vendors with QIC-117 tape drives recommend setting the bus on/off times such that the 1542 is on even longer than normal. Get the SCSISEL.EXE program from Adaptec's BBS or several places on the internet, and reduce the BUS ON

time or increase the BUS OFF time until the problem goes away, then move it one notch or more further. SCSISEL changes the EEPROM settings, so it's more permanent than a patch to the DOS driver line in CONFIG.SYS, and will work if you boot right into Linux (unlike the DOS patch). Next problem solved.

Last problem - the older Symphony chipsets drastically reduced the timing of the I/O cycles to speed up bus accesses. None of various boards I've played with had *any* problem with the reduced timing except for the PAS-16. Media Vision's BBS has SYMPFIX.EXE that's supposed to cure the problem by twiddling a diagnostic bit in Symphony's bus controller, but it's not a hard guarantee. You may need to:

- get the motherboard distributor to replace the older version bus chip,
- replace the motherboard, or
- buy a different brand of sound card.

Young Microsystems will upgrade the boards they import for around $30 (US); other vendors may be similar if you can figure out who made or imported the motherboard (good luck). The problem is in ProAudio's bus interface chip as far as I'm concerned; *nobody* buys a $120 sound card and sticks it in a 6MHz AT. Most of them wind up in 25-40MHz 386/486 boxes, and should be able to handle *at least* 12MHz bus rates if the chips are designed right. Exit soapbox (stage left).

The first problem depends on the chipset used on your motherboard, what bus speed and other BIOS settings, and the phase of the moon. The second problem depends on your refresh option setting (hidden or synchronous), the 1542 DMA rate and (possibly) the bus I/O rate. The third can be determined by calling Media Vision and asking which flavor of Symphony chip is incompatible with their slow design. Be warned, though - 3 of 4 techs I talked to were brain damaged. I would be very leery of trusting *anything* they said about someone else's hardware, since they didn't even know their own very well.

## 5.21 Problems with the FM synthesizer on a SoundBlaster Pro 1

The newer SB Pro has an OPL-3 FM chip, but the older version 1 used the OPL-2. The sound driver assumed the presence of an OPL-3. Version 2.5 of the sound driver corrects this problem.

## 5.22 Is the GUS-MAX supported?

The GUS-MAX is not explicitly supported yet, but it will work partially with the current sound driver. The driver does not know about the additions such as the mixer or 16 bit sampling. Booting your system and initializing the card under MS-DOS and then booting Linux (using ctrl-alt-del) should allow it to work.

Full GUS-MAX (and GUS + 16 bit daughtercard) support should be provided in the kernel driver in the near future.

## 5.23 What if my sound card is not supported?

First, make sure you really have an unsupported sound card. A few cards are compatible with supported cards (e.g. Logitech SoundMan 16 is compatible with ProAudioSpectrum 16). Post your question to the net

or the Linux activists SOUND channel.

If your card truly is not supported, here are some options:

- replace it with a supported sound card
- write the driver yourself
- ask Hannu Savolainen to add support to the sound driver

The *Hacker's Guide to Voxware* has some comments on which sound cards may be supported in future.

## 5.24 Is it possible to read and write samples simultaneously?

Due to hardware limitations, this is not possible with most sound cards. The only supported card that can do this is the ProAudioSpectrum16. See the section on "bidirectional mode" in the *Hacker's Guide to Voxware* for more information.