

## Midi with alsa



by Guido Socher ([homepage](#))

### *About the author:*

Guido loves Linux not only because it is fun to discover the great possibilities of this systems but also because of the people involved in its design.



### *Abstract:*

Midi is a file format which does not contain sampled digital sound but contains a description on how to make the sound. This is similar to sheet with notes where each note is a symbolic representation of a tone. Midi files are therefore very small in size compared to e.g wav files.

To convert midi to actual sound you need a midi synthesizer. This is a piece of hardware or software which uses a number of digital sound samples from an instrument (e.g piano) to generate the tones. These sound samples are known as "soundfont".

This article will explain how to use Alsa (advanced linux sound architecture) as a midi synthesizer interface.

---

## Introduction

Just a few years ago you could find famous sound cards like the "Gravis UltraSound" or "SoundBlaster Gold" in any computer shop. Both cards had internal hardware midi synthesizer. Unfortunately you can't buy these cards anymore. Even if you could they would not fit into a modern computer because they were ISA bus cards. Today's computer use the PCI bus and have often already a sound chip in the chipset on the motherboard. Hardly any of these chipsets supports midi synthesis. If you look at the list of alsa supported soundcards you will probably wonder which of these cards should I now buy. The decision is unfortunately easier than it looks. There is today only one card which supports midi synthesis that you can actually buy in the normal computer shops: Soundblaster live.

If you do not own a Soundblaster live card then you can emulate midi synthesis in software and I will explain how to do that.

## What you need

Alsa is currently in a transition phase. The stable release 0.5 is aging and 0.9 is still under development. If you actually want to use midi applications with alsa then you need 0.5 because there are hardly any applications for 0.9 and the interfaces between 0.5 and 0.9 are incompatible (note: PCM sampled sound, wav files, and applications basing on sampled sound are not a problem with 0.9. I am talking about midi.)

The packages for 0.5 are:

- alsa-driver-0.5.12a.tar.bz2
- alsa-lib-0.5.10b.tar.bz2
- alsa-utils-0.5.10.tar.bz2

Information on how to compile alsa can be found in the alsa howto (see references). Basically you need to run

```
tar jxvf alsa-driver-0.5.12a.tar.bz2
cd alsa-driver-0.5.12a
./configure --with-sequencer=yes --with-oss=yes --with-isapnp=no
make
make install
./snddevices
```

for the driver and "./configure;make;make install" for the rest.

After installation you can load the modules into your kernel. For soundblaster live (=emu10k1 chip) this is:

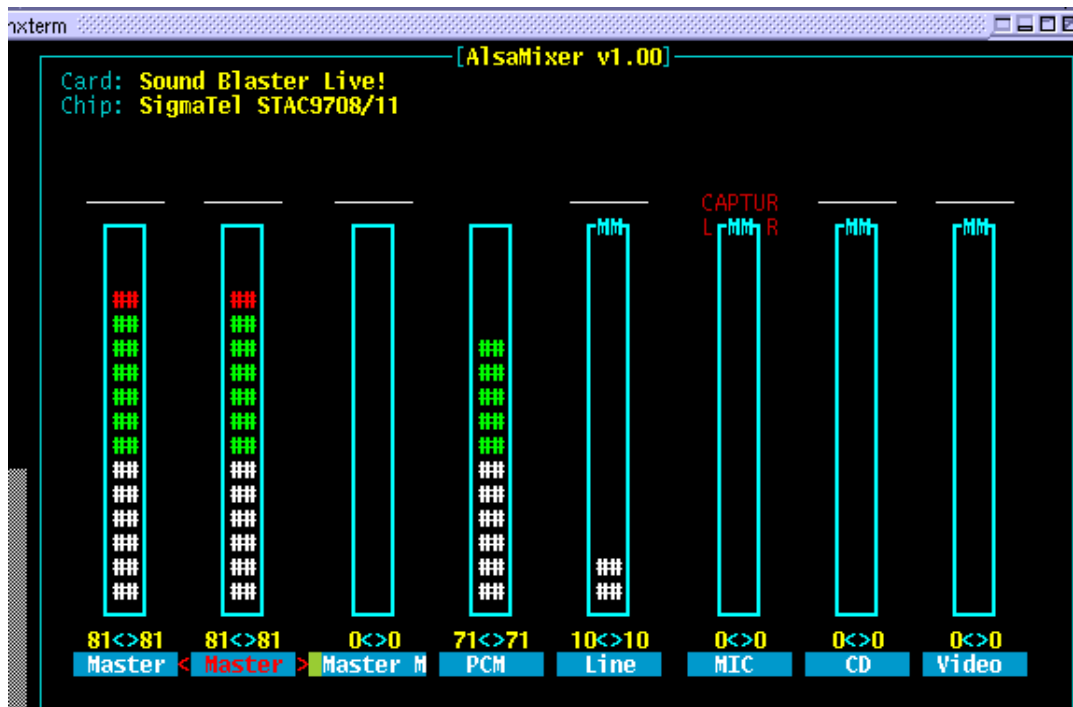
```
modprobe snd-card-emu10k1
modprobe snd-synth-emu10k1
modprobe snd-seq-midi
modprobe snd-pcm-oss
modprobe snd-mixer-oss
modprobe snd-seq-oss
```

For other cards just replace the first two lines with the module for your card (e.g snd-card-via686a for an internal soundcard bases on the via 686 chip). Mandrake and Suse support alsa and there you can also use the hardware detection and configuration program from the linux distribution (harddrake and yast2). If you have absolutly no idea what kind of chip you have then run "lspci -v" (the lspci command belongs to a package called pciutils).

Now it is time to test that sound works. Run the program

```
alsamixer
```

and type "m" to un-mute the master and pcm volume and then use the arrow keys to change the volume. Hit Esc to exit alsamixer when done.



To store the settings in `/etc/asound.conf` run the command

```
/usr/sbin/alsactl store
```

Now run

```
play flute.wav
```

and you should hear some sound. If not go to `/proc/asound` and check `/proc/asound/devices` and `/proc/asound/oss-devices`. There should be "mixer" and "digital audio" (my files).

This was the test that the pcm oss emulation and sound in general works. Further down I will explain what to add in `/etc/modules.conf` to do everything automatically but the next step is to get midi to work.

## midi

If you have a sound card with internal midi synthesizer (sound blaster live) then you need to load the "soundfont" (the sound samples) into the chip of your soundcard. If you do not have such a card then you can install timidity (see references for url to download) and use it as a alsa midi synthesizer. Programs running on top of alsa will not notice the difference.

### Loading the soundfont

Check that you have the program `sfxload` (`/bin/sfxload`) installed. If not install it. It is part of a package called `awesfx` (check the CDs of your linux distribution or <http://mitglied.lycos.de/iwai/awedrv.html>).

Next copy the file 8MBGMSFX.SF2 from your Soundblaster Live CD (/mnt/cdrom/AUDIO/Common/SFBANK/8MBGMSFX.SF2) to /etc/midi/8MBGMSFX.SF2. To load the soundfont run the command:

```
/bin/sfxload /etc/midi/8MBGMSFX.SF2
```

A good test program is pmidi (see references). Run  
pmidi -l  
This should produce the printout:

```
Port Client name Port name
64:0 External MIDI 0 MIDI 0-0
65:0 Emu10k1 WaveTable Emu10k1 Port 0
65:1 Emu10k1 WaveTable Emu10k1 Port 1
65:2 Emu10k1 WaveTable Emu10k1 Port 2
65:3 Emu10k1 WaveTable Emu10k1 Port 3
```

Now run

```
pmidi -p 65:0 test.mid
```

and you should hear midi sound. Cool!

### Using TiMidity as a software synthesizer

Download the TiMidity++-2.11.3.tar.gz package (see references at the end) unpack it (tar zxvf TiMidity++-2.11.3.tar.gz) and edit the file common.makefile.in. Here you uncomment the CFLAGS line for pentium gcc:

```
CFLAGS = -O3 -mpentium -march=pentium -fomit-frame-pointer \
-funroll-all-loops -malign-double -ffast-math
```

Now you can configure timidity with many nice graphical interfaces but we are primarily interested in the "--enable-alsaseq" option. Still it does not harm to enable some GUI stuff:

```
./configure --enable-ncurses --enable-xaw --enable-spectrogram --enable-xaw=dynamic
--enable-audio=oss,alsa --enable-alsaseq --prefix=/usr/local/timidity-2.11.3
make
make install
```

This installs timidity into /usr/local/timidity-2.11.3/bin to not overwrite an already existing timidity installation from your linux distribution. We install timidity because I have not seen a single linux distribution where the alsaseq interface was enabled.

For timidity you need also soundfonts. They are called this time instrument files. A good and complete set of instrument files is quite big (about 10Mb). The fastest way to get them is therefore to install the timidity++ from your distribution (CDrom) and copy the files from there (e.g timidity++-2.11.3-1.i386.rpm download for redhat 7.3). To copy the files from /usr/share/timidity/instruments to /usr/local/timidity-2.11.3/share/timidity/instruments run:

```
cd /usr
find share/timidity -print | cpio -dump /usr/local/timidity-2.11.3
```

Now we are ready to test our installation and start timidity as root:

```
/usr/local/timidity-2.11.3/bin/timidity -iA -B2,8 -Os -EFreverb=0
TiMidity starting in ALSA server mode
set SCHED_FIFO
Opening sequencer port: 128:0 128:1
```

then pmidi -l:

```
Port Client name Port name
128:0 Client-128 TiMidity port 0
128:1 Client-128 TiMidity port 1
```

Voila, we have 2 ports with the TiMidity synthesizer.  
Now run

```
pmidi -p 128:0 test.mid
```

and you should hear midi sound.

## **/etc/modules.conf**

For automatic configuration and module loading add this to your /etc/modules.conf if you have a soundblaster live card:

```
alias char-major-116 snd
alias char-major-14 soundcore
alias snd-card-0 snd-card-emu10k1
alias sound-slot-0 snd-card-0
alias sound-service-0-0 snd-mixer-oss
alias sound-service-0-1 snd-seq-oss
alias sound-service-0-3 snd-pcm-oss
alias sound-service-0-8 snd-seq-oss
alias sound-service-0-12 snd-pcm-oss
alias midi snd-synth-emu10k1
below snd-seq-oss snd-synth-emu10k1
post-install snd-synth-emu10k1 /bin/sfxload /etc/midi/8MBGMSFX.SF2 ; alsactl restore
# uncomment to save volume settings at shutdown:
#pre-remove snd-synth-emu10k1 alsactl store
```

For a different card without midi synthesizer, e.g internal via686:

```
alias char-major-116 snd
alias char-major-14 soundcore
```

```
alias snd-card-0 snd-card-via686a
alias sound-slot-0 snd-card-0
alias sound-service-0-0 snd-mixer-oss
alias sound-service-0-3 snd-pcm-oss
alias sound-service-0-12 snd-pcm-oss
# restore original mixer:
post-install snd-card-via686a alsactl restore
# uncomment to save volume settings at shutdown:
#pre-remove snd-synth-emu10k1 alsactl store
```

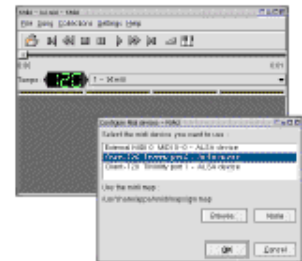
To start timidity automatically at startup I suggest to add the following line to /etc/init.d/alsasound (this script gets installed with the driver but is not active. To activate use the command chkconfig)

```
echo "starting timidity"
timidity=/usr/local/timidity-2.11.3/bin/timidity # do not forget the "&" in the next line:
$timidity -iA -B2,8 -Os -EFreverb=0 > /dev/null &
```

## Using alsa and midi

You have used the pmidi program above to play midi files. KDE includes a very nice player called kmid (not to be confused with kmidi). Kmid can be compiled with or without alsa support. Redhat has it compiled for OSS. Mandrake and Suse use Alsa. It is possible to use the Mandrake binaries with Redhat.

This article is also meant to explain the alsa installation which is needed for other programs that will be explained in future articles. We will e.g have an article on Jazz. Jazz is a mide sequencer and and midi file editor. At the end of the article you find also references to other applications.



There are also programs like timidity (timidity -ig will start the gtk GUI) or kmidi (not kmid) which already include the midi synthesizer in software. In these cases you do not need kernel internal midi capabilities whether supported by hardware or emulated in timidity software. In general it is however better to have a common midi api than to re-implement a midi synthesizer in every software.

Alsa is currently undergoing big changes. As said earlier version 0.9 does not work with most applications however the version 0.5 as discussed in this article has also its problems. The OSS sequencer emulation (/dev/sequencer) does not work (it works only with old soundblaster awe cards which you can't buy anymore). That means only alsa aware midi programs will work. This will hopefully change with version 0.9. This article is however also useful in the future when version 0.9 will have more support since only "small" things such as the names of modules have changed between 0.5 and 0.9. The basic principles will still be the same.

## References

- The alsa howto: [www.amelek.gda.pl/avr/](http://www.amelek.gda.pl/avr/)

- The Alsa project: <http://www.alsa-project.org/>
- Download TiMidity++-2.11.3.tar.gz from this server: [TiMidity++-2.11.3.tar.gz](#)
- Timidity software midi synthesizer: <http://www.goice.co.jp/member/mo/timidity/dist/>
- Midi-howto: <http://www.midi-howto.com/>
- pmidi midi player: [pmidi-1.4.2.tar.gz](#)  
(from <http://download.sourceforge.net/pmidi/> or  
<http://www.parabola.demon.co.uk/alsa/pmidi.html> )
- A short and a long midi file for testing: test.mid bigstar.mid
- download page for this article
- Midi applications for Linux <http://www.linuxsound.at/midi.html>
- Sound & MIDI Software For Linux <http://linux-sound.org/one-page.html> (or  
<http://www.linuxsound.at>)

---

<p>Webpages maintained by the LinuxFocus Editor team © Guido Socher "some rights reserved" see <a href="http://linuxfocus.org/license/">linuxfocus.org/license/</a> <a href="http://www.LinuxFocus.org">http://www.LinuxFocus.org</a></p>	<p>Translation information: en --&gt; -- : Guido Socher (homepage)</p>
---	--