Via 686 Audio Driver for Linux

Jeff Garzik

Via 686 Audio Driver for Linux

by Jeff Garzik

Table of Contents

Chapter 1. Introduction

The Via VT82C686A "super southbridge" chips contain AC97-compatible audio logic which features dual 16-bit stereo PCM sound channels (full duplex), plus a third PCM channel intended for use in hardware-assisted FM synthesis.

The current Linux kernel audio driver for this family of chips supports audio playback and recording, but hardware-assisted FM features, and hardware buffer direct-access (mmap) support are not yet available.

This driver supports any Linux kernel version after 2.4.10.

Please send bug reports to the mailing list <`linux-via@gtf.org`>. To subscribe, e-mail <`majordomo@gtf.org`> with

```
subscribe linux-via
```

in the body of the message.

Chapter 2. Driver Installation

To use this audio driver, select the CONFIG_SOUND_VIA82CXXX option in the section Sound during kernel configuration. Follow the usual kernel procedures for rebuilding the kernel, or building and installing driver modules.

To make this driver the default audio driver, you can add the following to your /etc/conf.modules file:

```
alias sound via82cxxx_audio
```

Note that soundcore and ac97_codec support modules are also required for working audio, in addition to the via82cxxx_audio module itself.

Chapter 3. Submitting a bug report

3.1. Description of problem

Describe the application you were using to play/record sound, and how to reproduce the problem.

3.2. Diagnostic output

Obtain the via-audio-diag diagnostics program from http://sf.net/projects/gkernel/ and provide a dump of the audio chip's registers while the problem is occurring. Sample command line:

```
./via-audio-diag -aps > diag-output.txt
```

3.3. Driver debug output

Define `VIA_DEBUG` at the beginning of the driver, then capture and email the kernel log output. This can be viewed in the system kernel log (if enabled), or via the dmesg program. Sample command line:

```
dmesg > /tmp/dmesg-output.txt
```

3.4. Bigger kernel message buffer

If you wish to increase the size of the buffer displayed by dmesg, then change the `LOG_BUF_LEN` macro at the top of linux/kernel/printk.c, recompile your kernel, and pass the `LOG_BUF_LEN` value to dmesg. Sample command line with `LOG_BUF_LEN` == 32768:

```
dmesg -s 32768 > /tmp/dmesg-output.txt
```

Chapter 4. Known Bugs And Assumptions

Low volume

Volume too low on many systems. Workaround: use mixer program such as xmixer to increase volume.

Chapter 5. Thanks

Via for providing e-mail support, specs, and NDA’d source code.

MandrakeSoft for providing hacking time.

AC97 mixer interface fixes and debugging by Ron Cemer <roncemer@gte.net>.

Rui Sousa <rui.sousa@conexant.com>, for bugfixing MMAP support, and several other notable fixes that resulted from his hard work and testing.

Adrian Cox <adrian@humboldt.co.uk>, for bugfixing MMAP support, and several other notable fixes that resulted from his hard work and testing.

Thomas Sailer for further bugfixes.

Chapter 6. Random Notes

Two /proc pseudo-files provide diagnostic information. This is generally not useful to most users. Power users can disable CONFIG_SOUND_VIA82CXXX_PROCFS, and remove the /proc support code. Once version 2.0.0 is released, the /proc support code will be disabled by default. Available /proc pseudo-files:

```
/proc/driver/via/0/info
/proc/driver/via/0/ac97
```

This driver by default supports all PCI audio devices which report a vendor id of 0x1106, and a device id of 0x3058. Subsystem vendor and device ids are not examined.

GNU indent formatting options:

```
-kr -i8 -ts8 -br -ce -bap -sob -l80 -pcs -cs -ss -bs -di1 -nbc -lp -psl
```

Via has graciously donated e-mail support and source code to help further the development of this driver. Their assistance has been invaluable in the design and coding of the next major version of this driver.

The Via audio chip apparently provides a second PCM scatter-gather DMA channel just for FM data, but does not have a full hardware MIDI processor. I haven’t put much thought towards a solution here, but it might involve using SoftOSS midi wave table, or simply disabling MIDI support altogether and using the FM PCM channel as a second (input? output?)

Chapter 7. Driver ChangeLog

7.1. Version 1.9.1

- DSP read/write bugfixes from Thomas Sailer.
- Add new PCI id for single-channel use of Via 8233.
- Other bug fixes, tweaks, new ioctls.

7.2. Version 1.1.15

- Support for variable fragment size and variable fragment number (Rui Sousa)
- Fixes for the SPEED, STEREO, CHANNELS, FMT ioctls when in read & write mode (Rui Sousa)
- Mmaped sound is now fully functional. (Rui Sousa)
-

Make sure to enable PCI device before reading any of its PCI config information. (fixes potential hotplug problems)

- Clean up code a bit and add more internal function documentation.
- AC97 codec access fixes (Adrian Cox)
- Big endian fixes (Adrian Cox)
- MIDI support (Adrian Cox)
- Detect and report locked-rate AC97 codecs. If your hardware only supports 48Khz (locked rate), then your recording/playback software must upsample or downsample accordingly. The hardware cannot do it.
- Use new pci_request_regions and pci_disable_device functions in kernel 2.4.6.

7.3. Version 1.1.14

- Use VM_RESERVE when available, to eliminate unnecessary page faults.

7.4. Version 1.1.12

- mmap bug fixes from Linus.

7.5. Version 1.1.11

- Many more bug fixes. mmap enabled by default, but may still be buggy.
- Uses new and spiffy method of mmap'ing the DMA buffer, based on a suggestion from Linus.

7.6. Version 1.1.10

- Many bug fixes. mmap enabled by default, but may still be buggy.

7.7. Version 1.1.9

- Redesign and rewrite audio playback implementation. (faster and smaller, hopefully)

- Implement recording and full duplex (DSP_CAP_DUPLEX) support.
- Make procfs support optional.
- Quick interrupt status check, to lessen overhead in interrupt sharing situations.
- Add mmap(2) support. Disabled for now, it is still buggy and experimental.
- Surround all syscalls with a semaphore for cheap and easy SMP protection.
- Fix bug in channel shutdown (hardware channel reset) code.
- Remove unnecessary spinlocks (better performance).
- Eliminate "unknown AFMT" message by using a different method of selecting the best AFMT_xxx sound sample format for use.
- Support for realtime hardware pointer position reporting (DSP_CAP_REALTIME, SNDCTL_DSP_GETxPTR ioctls)

- Support for capture/playback triggering (DSP_CAP_TRIGGER, SNDCTL_DSP_SETTRIGGER ioctls)

- SNDCTL_DSP_SETDUPLEX and SNDCTL_DSP_POST ioctls now handled.

- Rewrite open(2) and close(2) logic to allow only one user at a time. All other open(2) attempts will sleep until they succeed. FIXME: open(O_RDONLY) and open(O_WRONLY) should be allowed to succeed.

- Reviewed code to ensure that SMP and multiple audio devices are fully supported.

7.8. Version 1.1.8

- Clean up interrupt handler output. Fixes the following kernel error message:

  ```
  unhandled interrupt ...
  ```

- Convert documentation to DocBook, so that PDF, HTML and PostScript (.ps) output is readily available.

7.9. Version 1.1.7

- Fix module unload bug where mixer device left registered after driver exit

7.10. Version 1.1.6

- Rewrite via_set_rate to mimic ALSA basic AC97 rate setting
- Remove much dead code
- Complete spin_lock_irqsave -> spin_lock_irq conversion in via_dsp_ioctl
- Fix build problem in via_dsp_ioctl
-