Syntax

Capturing audio with `ffmpeg` and ALSA is pretty much straightforward:

```
ffmpeg -f alsa <input_options> -i <input_device> ... output.wav
```

See the FFmpeg ALSA input device documentation for more info.

Selecting the input card

`input_device` tells `ffmpeg` which audio capturing card or device you would like to use. To get the list of all installed cards on your machine, you can type `arecord -l` or `arecord -L` (longer output).

To list recording cards or devices:

```
$ arecord -l

**** List of CAPTURE Hardware Devices ****
card 0: ICH5 [Intel ICH5], device 0: Intel ICH [Intel ICH5]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: ICH5 [Intel ICH5], device 1: Intel ICH - MIC ADC [Intel ICH5 - MIC ADC]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: ICH5 [Intel ICH5], device 2: Intel ICH - MIC2 ADC [Intel ICH5 - MIC2 ADC]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: ICH5 [Intel ICH5], device 3: Intel ICH - ADC2 [Intel ICH5 - ADC2]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: U0x46d0x809 [USB Device 0x46d:0x809], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

We can see there are 2 audio cards installed that provide capturing capabilities, namely "card 0" (Intel ICH5) and "card 1" (Microphone on the USB web cam). The easiest thing to do is to reference each of them directly using `-f alsa -i hw:0` or `-f alsa -i hw:1`:

```
ffmpeg -f alsa -i hw:1 -t 30 out.wav
```

That will give us a 30 seconds WAV audio output, recorded from our USB camera's **default recording device** (microphone). The default recording device can be selected using the `alsamixer` tool (see below) or specifying the device using an additional parameter Y in `hw:<X>,<Y>`, where <X>=card, <Y>=device. For example, to select "MIC2 ADC" from Intel card (look above at the list), we would use:

```
ffmpeg -f alsa -i hw:0,2 -t 30 out.wav
```

The best way is to select your card and default recording device with the `alsamixer` tool, because some audio cards have a complicated way of selecting the default input through the `ffmpeg` command line.

Surviving the reboot

If you reboot your machine, you will notice sometimes your cards get reordered, so "card 0" is listed as USB Audio and "card 1" is listed as Intel audio card. You might want to play with udev rules, but there is an easier solution for this. Typing `arecord -L` will give us a little bit more detailed listing of recording devices:

```
# arecord -L
null
    Discard all samples (playback) or generate zero samples (capture)
default:CARD=ICH5
    Intel ICH5, Intel ICH5
    Default Audio Device
sysdefault:CARD=ICH5
    Intel ICH5, Intel ICH5
    Default Audio Device
front:CARD=ICH5,DEV=0
    Intel ICH5, Intel ICH5
    Front speakers
surround40:CARD=ICH5,DEV=0
    Intel ICH5, Intel ICH5
    4.0 Surround output to Front and Rear speakers
surround41:CARD=ICH5,DEV=0
    Intel ICH5, Intel ICH5
    4.1 Surround output to Front, Rear and Subwoofer speakers
surround50:CARD=ICH5,DEV=0
    Intel ICH5, Intel ICH5
    5.0 Surround output to Front, Center and Rear speakers
surround51:CARD=ICH5,DEV=0
    Intel ICH5, Intel ICH5
    5.1 Surround output to Front, Center, Rear and Subwoofer speakers
default:CARD=U0x46d0x809
    USB Device 0x46d:0x809, USB Audio
    Default Audio Device
sysdefault:CARD=U0x46d0x809
    USB Device 0x46d:0x809, USB Audio
    Default Audio Device
front:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    Front speakers
surround40:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    4.0 Surround output to Front and Rear speakers
surround41:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    4.1 Surround output to Front, Rear and Subwoofer speakers
surround50:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    5.0 Surround output to Front, Center and Rear speakers
surround51:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    5.1 Surround output to Front, Center, Rear and Subwoofer speakers
surround71:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    7.1 Surround output to Front, Center, Side, Rear and Woofer speakers
iec958:CARD=U0x46d0x809,DEV=0
    USB Device 0x46d:0x809, USB Audio
    IEC958 (S/PDIF) Digital Audio Output
```

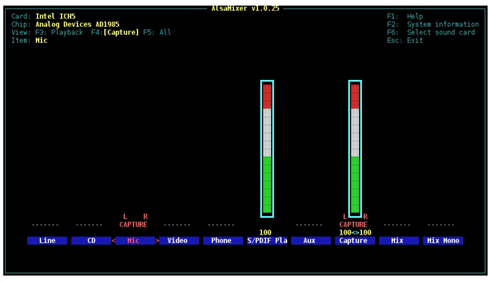
We can tell `ffmpeg` exactly what card we want to use, specifying the exact card's name, no matter which ordering it is, like this:

```
ffmpeg -f alsa -i default:CARD=U0x46d0x809 -t 30 out.wav
```

This way, you're always asking for the input from that certain device (the default recording device from the USB Audio device) and will never mix things up.

ALSA mixer tool

You might find useful a tool named `alsamixer`.



It will let you visually select, for each specified card (Intel or USB), which recording device do you want to use (if the specified card has got multiple inputs, like Line-In, CD-In, Mic, etc), so you can just run `alsamixer`, press F6 to choose the card, and then use TAB key to switch to recording devices (pressing it multiple times just switches between playback/recording/all devices), after that just use arrow keys to highlight desired device and just hit the SPACEBAR key to select it (and up/down, page up/dn keys to change the input volume).

Input options

The only useful audio input options for ALSA input are `-ar` (audio sample rate) and `-ac` (audio channels).

Specifying audio sampling rate/frequency will force the audio card to record the audio at that specified rate. Usually the default value is "44100" (Hz). Specifying audio channels will force the audio card to record the audio as mono, stereo or even 2.1/5.1 (if supported by your audio card). Usually the default value is "1" (mono) for Mic input and "2" (stereo) for Line-In input.

Examples

Record audio from your microphone

When doing screencast recordings, you usually want to record your voice too:

```
ffmpeg -f alsa -ac 1 -ar 44100 -i hw:0 -t 30 out.wav
```

Looking at our example device listing, this would be the same as this:

```
ffmpeg -f alsa -ac 1 -ar 44100 -i default:CARD=ICH5 -t 30 out.wav
```

Record audio from an application

Load the `snd_aloop` module:

```
modprobe snd-aloop pcm_substreams=1
```

Set the default ALSA audio output to one substream of the Loopback device in your `.asoundrc` (or `/etc/asound.conf`)

```
# .asoundrc
pcm.!default { type plug slave.pcm "hw:Loopback,0,0" }
```

You can now record audio from a running application using:

```
ffmpeg -f alsa -ac 2 -ar 44100 -i hw:Loopback,1,0 out.wav
```

Record audio from an application while also routing the audio to an output device

Load the `snd_aloop` module:

```
modprobe snd-aloop pcm_substreams=1
```

Set up your `.asoundrc` (or `/etc/asound.conf`) like so:

```
# .asoundrc
pcm.multi {
    type route;
    slave.pcm {
        type multi;
        slaves.a.pcm "output";
        slaves.b.pcm "loopin";
        slaves.a.channels 2;
        slaves.b.channels 2;
        bindings.0.slave a;
        bindings.0.channel 0;
        bindings.1.slave a;
        bindings.1.channel 1;
        bindings.2.slave b;
        bindings.2.channel 0;
        bindings.3.slave b;
        bindings.3.channel 1;
    }

    ttable.0.0 1;
    ttable.1.1 1;
    ttable.0.2 1;
    ttable.1.3 1;
}

pcm.!default {
        type plug
        slave.pcm "multi"
}

pcm.output {
        type hw
        card <Your Output Device Name>
}

pcm.loopin {
        type plug
        slave.pcm "hw:Loopback,0,0"
}

pcm.loopout {
        type plug
        slave.pcm "hw:Loopback,1,0"
}
```

where the output `pcm` points to the output device you want the audio to go to.

You can now record audio from a running application using:

```
ffmpeg -f alsa -ac 2 -ar 44100 -i loopout out.wav
```


Last modified on Feb 25, 2016, 9:40:44 PM

t a g s

alsa audio capture