Zen Cape Audio Guide

by Brian Fraser Last update: Nov 1, 2022

Guide has been tested onBeagleBone (Target):PC OS (host):Debian 11.5

Works on at least Linux kernels 4.9+

This document guides the user through

- 1. Getting the audio playback working on a USB Audio Adapter
- 2. Playing PCM (wave) files via a C program through asound.

Table of Contents

4
4
4
4
5

Formatting

- 1. Commands for the host Linux's console are show as:
 (host)\$ echo "Hello PC world!"
- 2. Commands for the target (BeagleBone) Linux's console are shown as: (bbg) \$ echo "Hello embedded world!"
- 3. Almost all commands are case sensitive.

Revision History

• Nov 1, 2022: Updated to Debian Bullseye and use of USB Audio Adapter

1. Configure and Play Audio

- On the target, sound tools and libraries (aplay, alsamixer, lsusb) (bbg)\$ sudo apt-get install usbutils alsa-utils i2c-tools
- 2. Plug in your USB Audio Adapter to your target. Ensure it is detected: (bbg)\$ lsusb Possible output: Bus 001 Device 003: ID 1b3f:2008 Generalplus Technology Inc. USB Audio Device Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
 - If needed, run dmesg to find out information about how the Linux kernel tried to load drivers, etc.
 - This guide tested with: "Generalplus Technology Inc. USB Audio Device" from Envel "C-Media Electronics, Inc. Audio Adapter (Unitek Y-247A)" from Cable Creation
- 3. Set the default sound card for ALSA on the target:

```
    View sound cards available:
(bbg)$ cat /proc/asound/cards
    Output:
1 [Device ]: USB-Audio - USB Audio Device
GeneralPlus USB Audio Device at usb-musb-hdrc.1-1, full speed
```

• Set the default, changing the "1" below to the desired device number listed above: (bbg)\$ sudo nano /etc/asound.conf

```
Set contents to:
defaults.pcm.card 1
defaults.ctl.card 1
```

- 4. Plug in speakers or headphones into the audio output on the USB Audio Adapter.
 - WARNING: When testing your audio, do not put headphones in your ear. A very loud sound is possible if there are problems, and this could cause an injury to your ear.

Just drape the head-phones beside your ears so you can hear it, but not be injured if it goes wrong. Once you know the audio levels are fine, then using head-phones normally is fine.

- 5. Save a WAVE file to your NFS folder on your host (say, sample.wav).
- 6. On the target, play the file with: (bbg) \$ aplay sample.wav

```
7. Change the volume: (bbg) $ alsomixer
```

- Use the left/right arrows to select different channels to adjust.
- Use the up/down arrows to change the volume.
- Press 'M' to mute or unmute channels.
- Press ESC to exit (and save changes).
- Change the volume of wave data playback by changing the Speaker channel.
- 8. Troubleshooting:
 - You can list available playback devices to ensure the configuration succeeded:

```
(bbg)$ aplay -1
**** List of PLAYBACK Hardware Devices ****
card 1: Device [USB Audio Device], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

- You can also list PCM playback devices: (bbg) \$ aplay -L
- You can view same info: (bbg) \$ cat /proc/asound/cards
- Occasionally playing a file via NFS can fail. If it fails, try copying the wave file to your home folder on the target first.
- You can playback a sound on a specific card using a command such as: (bbg)\$ aplay -D default:CARD=Device ./my-file.wav
 Where the CARD parameter can be found by running `aplay -L`, look for "CARD="
- You can play back white noise to test the hardware: (bbg) \$ speaker-test
- If the output from the software looks like it should be playing audio but no sound is generated, ensure you have speakers or headphones plugged into the correct audio-out jack, and that you have fully pushed in the connector (may hear a small click as it goes in).

```
    If you get the following error with aplay, it likely means you have not yet set your default audio device, or your USB Audio Adapter is not connected:
        ALSA lib confmisc.c:767:(parse_card) cannot find card '0'
            ALSA lib conf.c:4745:(_snd_config_evaluate) function snd_func_card_driver
            returned error: No such file or directory
            ALSA lib conf.c:4745:(_snd_config_evaluate) function snd_func_concat
            returned error: No such file or directory
            ALSA lib conf.c:4745:(_snd_config_evaluate) function snd_func_concat
            returned error: No such file or directory
            ALSA lib confmisc.c:1246:(snd_func_refer) error evaluating name
            ALSA lib conf.c:4745:(_snd_config_evaluate) function snd_func_refer
            returned error: No such file or directory
            ALSA lib conf.c:5233:(snd_config_expand) Evaluate error: No such file or
            directory
            ALSA lib conf.c:2660:(snd_pcm_open_noupdate) Unknown PCM default
            aplay: main:830: audio open error: No such file or directory
```

2. (Optional) Other Tools Command-line Tools

2.1 Recording (Untested)

- **1.** Record with:
 - (bbg) \$ arecord -r 44100 -f S16_LE -c 2 testRecording.wav
- 2. Things to do to prove out the recording capabilities more:
 - Types of microphones, and mic settings need to be investigated.
 - Volume controls for recording need to be investigated.
 - Tested using an audio cable from Zen headphone jack back into Zen mic port. Able to record poor quality quiet audio using the above command. Recommend using an audio file processing tool (such as GoldWave for Windows) to view recordings while debugging.

2.2 MP3 Player

- 1. Install the mpg123 package: (bbg)\$ sudo apt-get update (bbg)\$ sudo apt-get install mpg123
 - For this to work, you must have internet access. Test by pinging Google.
 - Note that installing this pulls in a number of other packages at the same time.
- 2. Copy an MP3 file to your NFS share.
- 3. Play the MP3: (bbg) \$ mpg123 sample.mp3
 - You can view the amount of CPU consumed during playback by loading a new terminal to the target:
 (hk a) & target
 - (bbg)\$ **top**

This will show you the CPU usage of mplayer (~5% on my test).

• You can also use other command-line MP3 players such as mplayer (apt-get install it). Takes up ~250 MB to install its libraries.

2.3 Text to Speech¹

- 1. Install the text-to-speech engine: (bbg) \$ sudo apt-get update (bbg) \$ sudo apt-get install libttspico-utils
 - This will take about 6MB.
- 2. Generate a wave file: (bbg)\$ pico2wave -w testAudio.wav 'All your bits are belong to us.'
- 3. Playback the audio: (bbg) \$ aplay testAudio.wav

3. C Program to Play PCM Audio

- 1. On the host, install the asound development library (for the header files) (host) \$ sudo apt-get update (host) \$ sudo apt-get install libasound2-dev
- 2. On the BeagleBone, check if asound is already installed:
 (bbg)\$ cd /usr/lib/arm-linux-gnueabihf/
 (bbg)\$ ls libasound*
 libasound.so.2 libasound.so.2.0.0
 - If you don't see a libasound.so.2, then install the asound library: (bbg)\$ sudo apt-get update (bbg)\$ sudo apt-get install libasound2
- 3. Your host will need a copy of asound's .so file from the BeagleBone in order to build an application which uses asound to run on the target. So copy the .so file to the NFS mounted shared folder:
 - On the host, create a folder on the shared NFS space (host)\$ mkdir ~/cmpt433/public/asound_lib_BBB (host)\$ chmod a+rwx ~/cmpt433/public/asound_lib_BBB
 - On the BeagleBone, copy the file to the mount and name it libasound.so: (bbg)\$ cd /usr/lib/arm-linux-gnueabihf/ (bbg)\$ cp libasound.so.2.0.0 /mnt/remote/asound_lib_BBB/libasound.so
- 4. Download the wave_player.c example from the course website.
 - In a directory on your host, such as in ~/cmpt433/work/pcmExample/, copy the wave player.c and Makefile.
 - Create a wave-files/ sub-directory of this folder, and copy a wave file into it. For example, copy in the provided drum sounds wave files.
 - Edit wave_player.c and change the SOURCE_FILE constant to the filename of your choice (assumed to be in the wave-files/ folder)
 - Note that the program assumes the files are 16-bit, signed little endian, 44.1kHz, mono files (which is true of the drum sounds). If your sounds are different, you'll need to change the settings in wave player.c
- **5.** Cross-compile the example code by running make:
 - (host) \$ cd ~/cmpt433/work/pcmExample/

(host)\$ make

 Makefile will build the wave_player.c code into wave_player and place it in ~/cmpt433/public/myApps/. To do this, it needs the asound library you copied in previous steps. GCC is told to look for any necessary libraries in the following command already in the Makefile:

LFLAGS = -L\$ (HOME) / cmpt433/public/asound_lib_BBB

- The wav target in the Makefile copies the wave-files/ folder into the myApps/ folder.
- See comments in wave_player.c for details on how application works.

6. Run the application:

- (bbg)\$ cd /mnt/remote/myApps/ (bbg)\$ wave_player
- You should hear the drum sound selected in the .c file. The drum sounds are are quite short.
- For reference, the part of the application which actually sends data to be played is the call to snd_pcm_writei() in the Audio_playFile() function.

This call is blocking: it waits until the data has been transmitted to the ALSA sub-system for playback. However, there is some hardware buffering, so the sound may not have actually stopped when snd_pcm_writei() returns.

You can use this delay to send more data, hopefully fast enough so that the sound has no jitter. Or, if you want to exit, you may want to call <code>snd_pcm_drain()</code> first so that all buffers play out without clipping the end of your wave file.

- 7. Troubleshooting:
 - When trying to compile, if you get the following error: fatal error: alsa/asoundlib.h: No such file or directory

Ensure you have libasound2-dev installed on your host PC.

• When running wave_player, if you see: error while loading shared libraries: libasound.so.2: cannot open shared object file: No such file or directory

Then you likely need to install libasound2 on the target.

- If you don't hear any sound when running wave_player, use aplay to ensure your hardware is configured correctly and your mixer level is set correctly.
- If you see an error:

ALSA lib confmisc.c:768:(parse_card) cannot find card '0'

it likely means that you don't have the BB-BONE-AUDI-02 cape loaded; see previous section for loading the audio cape.