

Le RaspberryPi : Utiliser le son, la synthèse vocale et la reconnaissance vocale avec le Raspberry-Pi

Par X. HINAULT – Octobre 2012 – www.mon-club-elec.fr

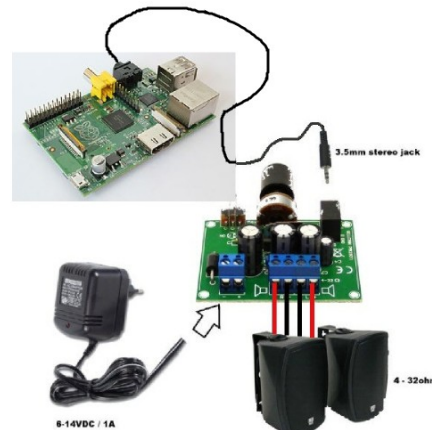


Matériel nécessaire pour les fonctions utilisant la sortie audio

- Le Raspberry dispose d'une sortie audio jack 3,5mm. Pour pouvoir utiliser les sons avec le RaspberryPi, la solution la plus simple consiste à utiliser une [paire d'enceintes de PC](#). Choisir un modèle avec ampli intégré et alimentation dédiée et si possible miniature. On pourrait aussi utiliser un simple [casque audio](#).



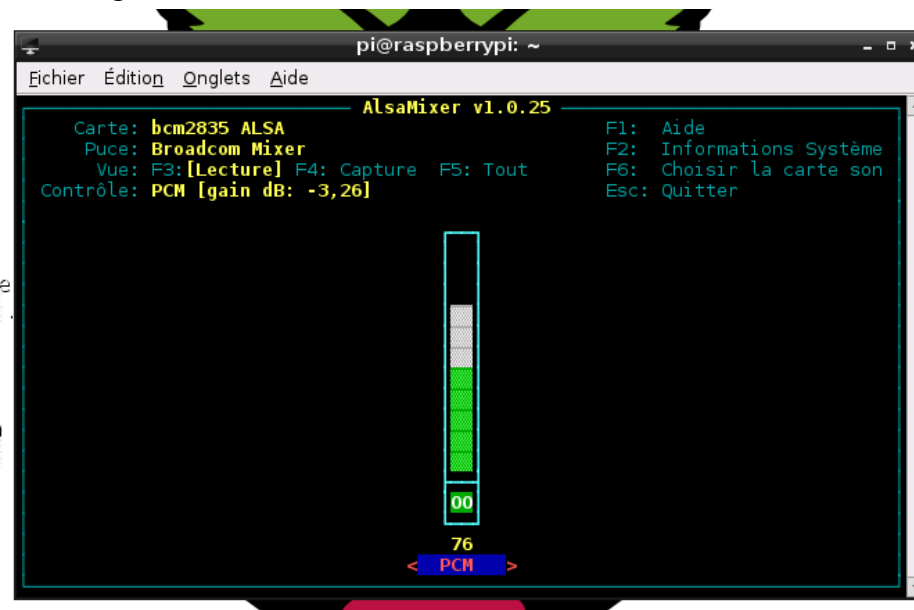
- Noter qu'en situation « embarquée », sur un robot ou autre, il pourra être intéressant d'utiliser une [petite carte ampli avec un simple haut-parleur](#).



Exemple avec kit Velleman ampli 2x5W MK190 (dispo chez Gotronic – réf 01359)

Technique : Pour info : Paramétrage de la carte son

- Pour accéder à la configuration du son, le plus simple est d'utiliser **alsamixer**, utilitaire qui s'appelle en ligne de commande
- `$ alsamixer`
- On obtient une interface sommaire qui permet de gérer les niveaux, entrées, sorties à l'aide des touches de fonctions :



- Note : j'ai tenté d'installer PulseAudio qui est une bonne interface pour gérer les entrées sorties/audio mais cela a posé des problèmes (plus de son...)

D'une manière générale, sur le RaspberryPi, j'ai l'impression qu'il ne faut pas trop modifier les paramètres audio (alsa) par défaut, qui fonctionnent et toute modification a l'air d'entraîner des problèmes.

- La liste des périphériques est accessible avec la commande

```
cat /proc/asound/cards
```

- Il peut être également nécessaire selon les problèmes rencontrés, ou en cas d'utilisation d'une carte son USB, de créer un fichier `.asoundrc` dans le `home/user` avec le contenu suivant :

```
pcm.!default {
type hw
card 1
}
ctl.!default {
type hw
card 1
}
```

}

- Plus d'infos ici : <http://alsa.opensrc.org/.asoundrc>

Lecture de fichiers sons : lecteur polyvalent mplayer

- Pour lire des fichiers sons, notamment des bruitages de toutes sortes, il est nécessaire d'utiliser un lecteur de fichier son. Un lecteur polyvalent est le lecteur mplayer, aussi bien pour la vidéo que les fichiers audio.
- Installation

```
$ sudo apt-get install mplayer
```

```
$ sudo apt-get install mencoder
```

- Utilisation : en ligne de commande, mplayer suivi du nom du fichier à lire :

```
$ mplayer mes_sons/r2d2.mp3
```

- Vous trouverez dans l'archive suivante plusieurs sons utiles à tester : http://www.mon-club-elec.fr/mes_downloads/mes_sons.tar.gz

```
$ cd
```

```
$ mkdir Telechargements
```

```
$ cd Telechargements/
```

```
$ wget http://www.mon-club-elec.fr/mes_downloads/mes_sons.tar.gz
```

- puis ouvrir le gestionnaire de fichier et extraire l'archive. Amusez-vous bien !

Remarque

A ce stade, vous disposez d'une possibilité très intéressante : il vous suffit de télécharger des fichiers de sons de bruitages (r2d2, alarmes, etc...) et, sans besoin de les convertir, vous pouvez les lire directement sur le raspberryPi.

Vous pouvez ainsi ajouter toutes sortes d'effets sonores à vos applications en quelques minutes !

Vous pouvez également lire des fichiers audio de votre choix.

- Pour lire fichier Flash : Flash n'existe pas pour ARM. Installer le paquet gnash à la place et tester si ça marche... pas pour moi.

Synthèse vocale

Introduction

- Avec la synthèse vocale, on rentre dans un domaine qui fait toujours un peu rêver : un robot, une application qui parle, ça lui donne un petit côté humain... Il est possible de le faire avec des cartes d'extension dédiées, notamment avec Arduino, mais les coûts sont tout de même assez élevés, et en tout cas beaucoup plus qu'un simple RaspberryPi...
- **La bonne nouvelle, c'est que le système Raspbian que nous avons installé sur RaspberryPi est capable de réaliser la synthèse vocale à partir de chaînes de caractères : nous allons donc ajouter de ce pas cette possibilité à notre système embarqué.**

Une fois que la synthèse vocale sera installée sur le raspberryPi, il sera ensuite possible de déclencher la synthèse vocale à partir d'Arduino comme nous le verrons ultérieurement. Vous pourrez ainsi annoncer la distance d'un obstacle, dire quelque chose à l'appui sur un bouton poussoir, etc...

Non, non, vous ne rêvez pas !

Solution 1 : synthétiseur vocal espeak et son interface graphique gespeaker

- Ce synthétiseur de voix permet la synthèse de voix à partir de chaîne de texte. Il offre plusieurs voix masculines et féminines. Son son est un peu métallique cependant. Il existe également une interface graphique appelée gespeaker.

- Installation :

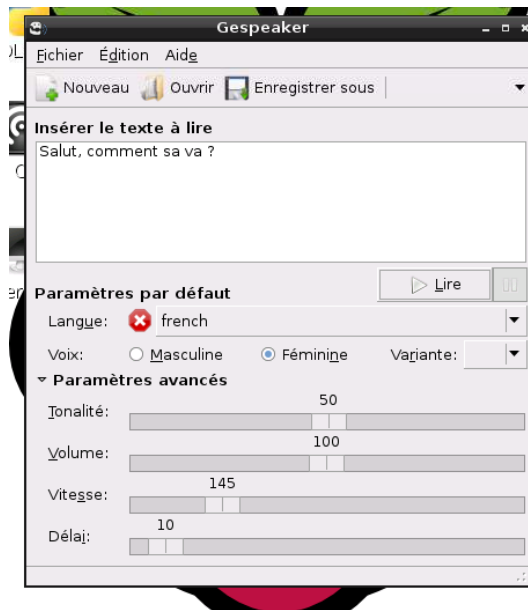
```
$ sudo apt-get install libportaudio0
```

```
$ sudo apt-get install espeak
```

```
$ sudo apt-get install gespeaker
```

- Utilisation :

- soit lancement en interface graphique depuis le menu Application > Son et Vidéo > Gespeaker. On obtient une interface permettant de générer de la synthèse vocale.



- soit lancement en ligne de commande :

```
$ espeak -v fr -s 140 'lecture de texte avec ispik'
```

- Note : lors de l'installation initiale de espeak sur le raspberryPi, il semblait que les voix n'étaient pas installées au bon endroit.
 - Pour les trouver : `find / -name 'espeak-data' -type d`
 - Ensuite se placer dans le répertoire où elles se trouvent : `cd /usr/lib/arm-linux-gnueabi/espeak-data`
 - puis créer répertoire de destination attendu : `sudo mkdir /usr/share/espeak-data`
 - puis copier le contenu : `sudo cp * /usr/share/espeak-data/ -r`

Solution 2 : synthétiseur vocal picotts

- picotts est un synthétiseur vocal qui donne une bonne qualité de voix féminine. C'est la voix d'Android. A priori pas dispo avec raspbian... Ceci étant, il devrait être possible de compiler les sources... à suivre..
- Une documentation utile est disponible ici : <http://forums.debian.net/viewtopic.php?f=16&t=38976>
- Tout d'abord ajouter le dépôt des sources :

```
$ sudo geany /etc/apt/sources.list
```

Coller les 2 lignes :

```
deb http://archive.raspbian.org/raspbian wheezy main contrib non-free rpi
deb-src http://archive.raspbian.org/raspbian wheezy main contrib non-free rpi
```

Ensuite, mettre à jour avec :

```
$ sudo apt-get update
```

- Ensuite :

```
$ apt-get update
$ apt-get install fakeroot
$ apt-get build-dep libttspico-utils
$ cd
$ mkdir my_build/
$ cd /my_build/
$ apt-get source libttspico-utils
$ ls
$ cd svox-1.0-<version>/
$ dpkg-buildpackage -rfakeroot -us -uc
```

Ensuite c'est parti pour un moment... compter un bon 1/4h d'heure... On est quand même en train de faire une compil' sur le raspberryPi lui-même... excusez du peu...

Puis ensuite, on peut faire l'install

```
$ cd
$ cd my_build
$ ls
```

On obtient plusieurs fichiers *.deb :

libttspico0..armhf.deb

libttspico-data..all.deb

libttspico-dev...armhf.deb

libttspico-utils...armhf.deb

Il faut installer dans l'ordre pico0, pico-data et pico-utils (le pico-dev n'est pas indispensable) avec :

```
sudo dpkg -i <packagename>_<version>_<architecture>.deb
```

ce qui donne :

```
$ sudo dpkg -i libttspico0..armhf.deb  
$ sudo dpkg -i libttspico-data..all.deb  
$ sudo dpkg -i libttspico-utils...armhf.deb
```

Ensuite, on peut tester la synthèse vocale Pico sur notre système RaspberryPi :

```
$ pico2wave -l fr-FR -w test.wav « Salut les amis. Je mappelle pico »  
$ aplay test.wav
```

Et bingo, ça marche !! Pas belle la vie ? **Donc la synthèse vocale pico est opérationnelle sur le RaspberryPi...** beaucoup mieux que la voix métallique de Espeak, même si c'est déjà sympa quand même.

Enregistrement sons

- Plusieurs solutions possibles : rec, arecord, ... à utiliser selon ses besoins. Le plus simple est d'utiliser arecord déjà installé.
- Installation : installer le paquet alsa-utils qui contient arecord. Déjà installé par défaut.

```
$ sudo apt-get install alsa-utils
```

- Utilisation : `$ arecord <params>`

```
$ arecord
```

- Information : `$ man arecord`
 - -l donne la liste des périphériques
 - -t type de fichier parmi voc, wav, raw, au
 - -r débit en Hz
 - -d durée de l'enregistrement en secondes
 - -V activation vumètre = ne marche pas..
 - -c nombre de canaux 1 par défaut
 - -f cd (16 bit little endian, 44100, stereo)
- Exemples :

```
$ arecord -d 5 test.wav
```

```
$ arecord -d 5 -r 16000 -f cd test.wav # enregistre 5 secondes en qualité CD à 16000hz dans fichier test.wav
```

PS : utilisation de -f cd limite les « craquements » de début et fin d'enregistrement...

Note technique : par défaut, mon micro de webcam n'était détecté. J'ai créé le fichier .asoundrc avec la commande

```
$ geany .asoundrc
```

et copier coller le code suivant qui indique d'utiliser le micro de la webcam hw:1,0 pour la capture par défaut :

```
pcm.!default {
    type asym
    playback.pcm {
        type plug
        slave.pcm "hw:0,0"
    }
    capture.pcm {
        type plug
        slave.pcm "hw:1,0"
    }
}
```

Plus de détails ici : http://wiki.audacityteam.org/wiki/USB_mic_on_Linux

Conversion de fichier son

- Un utilitaire polyvalent : sox
- Installation : le paquet sox

```
$ sudo apt-get install sox
```

- Utilisation :

```
$ sox test.wav fichier.flac rate 16k # convertit le fichier *.wav en fichier *.flac avec échantillonnage 16000hz
```

- Information : \$ man sox

Reconnaissance vocale

- En combinant l'enregistrement de son et la conversion, il est possible sur le RaspberryPi d'utiliser la reconnaissance vocale de Google sous réserve d'être connecté à internet (via le routeur) !
- Il suffit d'enregistrer un fichier son puis de l'envoyer vers le serveur Google pour obtenir la chaîne texte correspondante. A suivre...

```
$ arecord -d 5 -r 16000 -f cd test.wav
```

```
$ sox test.wav fichier.flac rate 16k
```

```
$ wget -q -U "Mozilla/5.0" --post-file fichier.flac --header="Content-Type: audio/x-flac; rate=16000" -O -  
"http://www.google.com/speech-api/v1/recognize?lang=fr&client=chromium" | sed 's/^\.*\[{.*:"\(.*\)", ".*$/\1/'
```

- Exécuter les commandes suivantes en parlant après la première : la 3ème commande doit renvoyer le texte correspondant à ce que vous avez dit !