MaO

Table des matières

1 Cartes sons

```
// plusieurs cartes (pour éventuellement les réordonner)
# cat /proc/asound/cards

// quels drivers sont chargés (pour éventuellement les black-lister)
# lsmod | grep snd

// pulse-audio
-> paneau de configuation >> son >> test des hauts parleurs

// vlc
-> sortie audio (pulseaudio)
```

2 Carte son TerraTec DMX6Fire

Dans `Patchbay/Router`, `Digital Mix` permet de merger la stéréo du micro et du pc sur la sortie front de la carte.

- Monitor Inputs : Mic L, Mic R.
  Il s'agit des entrées physiques sur la carte son (le micro correspond aux entrées 5 et 6...). Ici ça ne sert à rien de les activer si l'on ne veut pas faire de "*[Digital Mix]*".
  - Enceinte gauche : Left *[Mic L =>max]*
  - Enceinte droite : Right *[Mic R =>max]*

- Monitor PCMs : PCM Out1, PCM Out2.
  Il s'agit des entrées logiques (Audio/System/Playback via Jack). De base les logiciels se branchent sur PCM1 et PCM2. Ici ça ne sert à rien de les activer si l'on ne veut pas faire de "*[Digital Mix]*".
  - Enceinte gauche : Left *[PCM Out 1 =>max]*
  - Enceinte droite : Right *[PCM Out 2 =>max]*
- Patchbay/Routeur : PCM, Mic, Mix (Logiciels, le micro ou les 2).
  Il s'agit des sorties analogiques qui sortent sur les enceintes. Je ne me suis jamais servi que de *[HW Out1]* et 2. Le plus simple c'est de les cabler sur *[PCM Out1]* et 2 : on utilise directement les sorties des logiciels. *[Phono Mic]* permet de tester le micro (seul). *[Digital Mix]* permet de récupérer la jauge de gauche.
  - Enceinte gauche : HW/Out1 (L) *[Digital Mix L]*
  - Enceinte droite : HW/Out2 (R) *[Digital Mix R]*
- Hardware setting : (Pour allumer le micro).
  Là je laisse presque tout par défaut, ou sinon je vois pas vraiment de différence.
  - Line In Selector : *[Front Input]*
  - Switch Phono/Mic : *[Mic]*
- Analog Volume : (Le plus important : volumes d'E/S sur la carte).
  - Enceinte gauche : DAC 0 Front L *[100]*
  - Enceinte droite : DAC 1 Front R : *[100]*
  - Micro gauche : ADC 4 Phono (L) : *[130]*
  - Micro droit ADC 5 Phono (R) : *[130]*

3 Jackd

```
# apt-get install qjackctrl vlc-plugin-jack
```

Fichier */etc/security/limits.conf* :

```
@audio -memlock   500000
@audio -rtprio 100
@audio -nice -10
```

Pour que ça prenne effet :

```
$ groups
... audio ...

# /etc/init.d/gdm restart
$ ulimit -l -r
max locked memory       (kbytes, -l) 500000
real-time priority              (-r) 100
```

`qjackctrl` permet de connecter les softs entre eux et avec la carte son. Il y a 3 onglets présentant les sorties à gauches et les entrées à droites :

- Audio : les entrées/sorties audio (dont le micro et les enseintes via la carte son).
- Midi : nouveau standard MIDI (pas encore super utilisé)
- Alsa : ancien standard MIDI (les ports ont une adresse X :Y pratique en ligne de commande)

Nous en reparlerons plus loin avec le MIDI.

3.1 vu-mètres

```
# apt-get install meterbridge
$ meterbridge -t vu X    // ('X' pour un port bidon sinon ne se lance pas)
```

3.2 Vlc

sortie audio (jackd)

```
# apt-get install vlc-plugin-jack
```

3.3 Ardour2

Rien à dire, fait le job pour la prise de son audio.

3.4 SooperLooper

On peut récupérer le paquet sur les mirroirs Debian, (bien qu'apt n'y arrive pas...).
Dans audio, il faut brancher :

- System/capture_5 (micro) -> SooperLooper/common_in_1,
- SooperLooper/common_out_1 -> System/playback_1 (PCM Out 1)

On peut aussi brancher le clavier MIDI, puis redéfinir les touches, principalement :

- "record" : commencer la boucle (efface tout), puis la faire tourner.
- "overdub" : enregister par dessus.
- "pause" : pas moyen d'arrêter la boucle sinon.

La durée maximale de la boucle est affichée ("mem"). Elle peut être changée en configuration mais alors il faut quitter puis redémarer le soft.

```
$ slgui
```

4 MIDI

Dans `qjackctl! il y a 3 onglets présentant les sorties à gauches et les entrées à droites`

- Audio : les entrées/sorties audio (dont le micro et les ensintes via la carte son).
- Midi : nouveau standard MIDI (pas encore super utilisé)
- Alsa : ancien standard MIDI (les ports ont une adresse X :Y pratique en ligne de commande)

Il faut utiliser `a2jmidid` pour faire un pont de Alsa-Midi vers Jack-Midi. Sinon, pas moyen de communiquer entre eux.

```
$ a2j -e &
```

Des ports "Midi Trought" entrées et sorties simulent (driver linux ?) un bus MIDI : plutôt que de cabler A à B on peut intercaler le bus au millieu (pour concentrer depuis ou diffuser vers plusieurs équipements).

4.1 Clavier virtuel

```
# apt-get install jack-keyboard
$ jack-keyboard
$ jack-keyboard
```

Le clavier de jack nécessite un pont vers Alsa-Midi (peut d'applications sont compatibles Jack-Midi) mais il est pratique car il possède une sortie et une entrée. De ce fait on peut s'en servir de vu-mètre MIDI : par exemple cabler 2 claviers et constater que l'on pilote le second avec le premier.

4.2 Clavier réel

A priori il n'y a besoin d'aucun réglage de la carte graphique pour rendre le port MIDI accessible. Si ça ne marche pas ça peut être dû aux les piles du clavier (ça marche avec un transfo de 600 mA). La sortie MIDI de la carte son apparaît dans le panneau "ALSA" de jackctrl.

4.3 Lecteur

Dans `qjackctl` on retrouve le numéro du port (Alsa-Midi) que l'on souhaite connecter à la sortie du lecteur. Ou via `pmidi -l`. Cela peut être le clavier virtuel jack-keyboard, un synthétiseur, les deux, le bus "Midi Trought"...

```
# apt-get install pmidi
$ wget http://www.chaumetsoftware.com/canta/midi/...\
BrassensGeorges/Brassens_-_Supplique_Pour_Etre_Enterre_Sur_La_Plage_De_Sete.mid

// 14:0 : chez moi c'est Midi Through Port 0
$ pmidi -p 14:0 fichier.mid
```

4.4 Synthétiseurs

Sans eux on n'a pas de son. Ils se branchent sur les sorties `Audio:system` via qjackctl.

4.4.1 Orgue

Orgue de la fin des années 1960 qui marche sorti de la boîte.

```
# apt-get install foo-yc20
$ foo-yc20
```

4.4.2 Yoshimin

Ihm avec une table de mixage qui permet d'affecter les pistes MIDI à différents instruments.

```
# apt-get install yoshimi
$ yoshimi
```

La banque d'instruments (et les "preset") est dispo dans */usr/share/yoshimi*, il faut l'ajouter en configuration.

`yoshimi` a une option `-a` pour utilier les port Alsa-Midi.
Il possède son propre clavier virtuel.

4.4.3 fluidsynth

Synthétiseurs "standard" avec une large banque d'instruments.

```
# apt-get install fluidsynth fluid-soundfont-gm
$ fluidsynth -j /usr/share/sounds/sf2/FluidR3_GM.sf2
```

Il peut prendre en entrée un fichier MIDI :

```
$ fluidsynth -j /usr/share/sounds/sf2/FluidR3_GM.sf2 fichier.mid
```

Il peut être utilisé via l'IHM `qsynth` :

```
# apt-get install qsynth fluid-soundfont-gm
```

Il faut ajouter la banque d'instruments en configuration : */usr/share/sounds/sf2*

4.4.4 timidity

Synthétiseurs "standard" avec une large banque d'instruments et également la possibilité d'utiliser celle de fluidsynth.

```
# apt-get install timidity freepats
$ timidity -iA -Oj -B2,8 // entré midi alsa, sortie jack, moins de latence
TiMidity starting in ALSA server mode
Opening sequencer port: 128:0 128:1 128:2 128:3

// rq: il énumère ses ports Alsa-Midi d'entrées, pour pmidi par exemple :
$ pmidi -p 128:0 fichier.mid
```

On peut changer de banque à la fin du fichier */etc/timidity/timidity.cfg*.

Il peut prendre en entrée un fichier MIDI, et il permet de transposer.

```
$ timidity -Oj camarde.mid       // en Ré M
$ timidity -Oj camarde.mid -K -2 // en Do M
$ timidity -Oj camarde.mid -K -4 -T 75 // en Sib M (tuba) et 1/2 fois plus lent (75%)
```

Il peut être utilisé via une IHM minimaliste, qui permet aussi de transposer :

```
# apt-get install timidity-interfaces-extr
$ timidity -ig -Oj // ihm gtk
```

Il est fournit également en mode serveur, mais de base il semble configuré pour alsa.

```
# apt-get install timidity-daemon
```

4.5 Editeur de partition

Permet de charger un fichier midi, de transposer les pistes une par une, puis de sauver au format rosegarden.

```
# apt-get install rosegarder
$ timidity -iA -Oj
$ rosegarden
```

Il faut brancher la sortie (Alsa-Midi) "out 1 General Midi" de rosegarden sur TiMidity, et les sortie Audio de TiMidity sur le système.