

Installation sur l'Udoo

Version

1.0

Introduction

Ce tutoriel a été écrit pour pouvoir reproduire la partie logiciel du prototype du dispositif d'interaction sonore. Tout le code du projet ainsi que les fichiers sont disponible sur [ce dépôt github](#). Si vous constatez des erreurs dans ce tutorial, merci de contacter [la personne actuellement en charge du projet](#).

Création de la carte et installation d'Udoobuntu

Cette partie est à réaliser sur votre ordinateur. Ouvrez un terminal.

```
$ df -h                                # trouvez le nom de la carte
$ sudo umount /dev/[nom device]         # ex: "/dev/sdb1"
$ sudo dd bs=1M if[img_file_path] of=/dev/[sd_name] # ex: "/dev/sdb", le nom moins le chiffre
```

Une image de l'OS (Udoobuntu) est joint au dossier. Il est également possible de le récupérer sur le [site d'Udoo](#).

Pour être sûr que les données se sont bien écrites sur la carte.

```
$ sudo sync
```

Maintenant la carte micro SD est prête à être utilisée. Vous pouvez la retirer de votre ordinateur et l'insérer dans la carte *Udoo*. Après avoir branché l'*Udoo* à un écran, et avoir connecté un clavier, une souris, et un moyen d'avoir internet (câble Ethernet ou Wifi).

Installation Pure Data

Avant de pouvoir installer *Pure Data* et *Pure Data Extended*, il faut supprimer la version incomplète pré-installée dans *Udoobuntu*.

```
sudo apt-get purge puredata-core      #remove la version pré-installer
sudo apt-get install openssh-server   #pour communiquer en ssh
```

De plus il faut changer la configuration du clavier car de base elle utilise le clavier anglais. À taper dans un terminal, ou à copier dans le fichier *"etc/rc.local"* (pour le charger automatiquement à chaque fois).

```
setxkbmap fr
```

Installation des dépendances

```
sudo apt-get install openssh-server automake libtool libasound2-dev
```

À partir de ce moment il est possible de continuer l'installation à distance en ssh, utilisez la commande *ifconfig* pour récupérer l'adresse IP de la carte *Udoo*.

Installation Pure Data

Récupérer la version de *Pure Data* sur [le site de Miller Puckett](#).

```
tar -xvf pd-[version].udoo.tar.gz      # décompressez l'archive
cd pd[version]
./autogen.sh                           # suivre les instructions du INSTALL.txt
./configure --enable-jack
make
sudo make install
```

Pure Data est maintenant installé, pour le lancer il suffit de taper la commande *sudo pd*. Le *sudo* est optionnel, cela permet juste à pure data d'être plus réactif pour garantir du pseudo temps réel.

Installation Pure Data Extended

Récupération de bibliothèque nécessaire pour utiliser *pd-extended* et ses différentes externes. Certains sont peut-être inutiles pour notre application mais il est préférable d'installer la totalité pour ne pas être embêté par la suite.

```
sudo apt-get install fakeroot ttf-dejavu libfftw3-dev libspeex-dev libvorbis-dev libflac-dev libsndfile1-dev libsamplerate0-dev libgav
telecharger du site de pd-extended les codes sources
tar -xvf Pd-extended_[version]-source.tar.bz2
cd pd-extended
cd packages/linux_make
make install          # ATTENTION, la compilation est très longue + de 2 heures
make package
sudo dpkg -i Pd-[version]-extended.deb
```

Pure Data extended est fonctionnelle.

Lien utilise:

- <http://www.udoo.org/getting-started/creating-a-bootable-micro-sd-card-from-image/>
- <http://msp.ucsd.edu/software.html>

- <http://www.udoo.org/forum/viewtopic.php?f=27&t=478>

Installation Flex

Flex est une bibliothèque utilisée pour compiler des *Externals* pour Pd-extended. On peut télécharger les sources, depuis [le site de Flex](#). Il faut décompresser l'archive. Pour l'installation, on peut retrouver les instructions d'installation dans le fichier *build.txt*

```
bash build.sh pd gcc
```

Il faut peut être lancer plusieurs fois la commande en éditant les fichiers nécessaires, si besoin.

```
bash build.sh pd gcc install
```

compilation d'un external

Pour installer les externals, il faut se déplacer dans les dossiers où sont les fichiers sources des externals. Pour le projet les externals à compiler sont : *arduinoPack*, *arduinoUnPack*, *struct_rythme*, qui sont dans le dossier PSAR/Src/external. À faire dans chaque dossier des externals. Exemple le dossier "PSAR/Src/external/arduinoPack".

```
bash [path flex]/flex/build.sh pd gcc          # le chemin vers le dossier flex
bash {path flex}/flex/build.sh pd gcc install
```

Pour ajouter un external à Pd-extended, une fois qu'il a été compilé il faut l'ajouter le chemin vers le fichier compilé dans Pd-extended : Édition >> préférence >> New.

Annexe

Recommandation pour l'utilisation de Pd-extended sur Udo

Au démarrage de Pd-extended, le logiciel affiche de nombreux **Warning** de *ALSA* mais cela n'a pas d'incidence sur le logiciel. De plus il y a un léger problème d'affichage, la barre de navigation est au dessus de l'écran, pour la voir il faut mettre la fenêtre en plein écran (clic droite >> maximise sur la barre de tâche), puis la redimensionner.

Sauvegarde et restauration de la carte Udo

À partir de l'image de la carte:

```
gunzip ./udoo-20150422-sdb.img.gz | sudo dd of=/dev/[name_device]    #ex: "of=/dev/sdb"
sudo dd if=/dev/[name_device] | gzip -9 > ./udoo-20150222-sdb.img.gz
```

Sauvegarde de la carte:

```
sudo dd bs=1M if=/dev/sdb | gzip > ./savecard.img.gz
```

tutoriel

Le tutoriel a été écrit en **Markdown**, et mis en *pdf* avec le site [Dillinger](#).

Licence

-A renseigner-