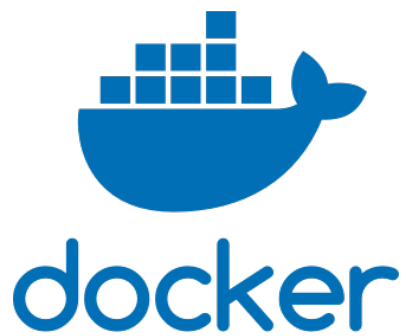




Kilodocker

Julien ANDRES
Thomas TAYLOR



Sommaire

1	Installation	2
2	Docker : Commandes de base	2
3	Kilodocker	2
3.1	Kilodocker_lite	2
3.2	Kilodocker_full	3
4	Compilation et upload	3
4.1	Compilation	3
4.2	Upload du code aux kilobots	4
5	Kilombo	4

1 Installation

Guide détaillé sur l'installation et la configuration du package docker (basé sur la doc officielle de docker¹) :

1. `sudo apt-get update`
2. `sudo apt-get install apt-transport-https ca-certificates curl software-properties-common`
3. `curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`
4. `sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"`
5. `sudo apt-get update`
6. `sudo apt-get install docker-ce`
7. `sudo groupadd docker`
8. `sudo usermod -aG docker $USER`

2 Docker : Commandes de base

- `docker image ls`
- `docker image rm`
- `docker run "votre_image"`
- `docker container ls -a`
- `docker container start -i "votre_image"`

3 Kilodocker

Les scripts et Dockerfiles sont disponible : https://github.com/JulienAndres/p_androidKilobot/tree/master/kilodocker

Les scripts prennent en argument un dossier qui sera monté dans le docker à l'adresse /root/work

3.1 Kilodocker_lite

Contient les packages essentiels pour compiler et envoyer du code aux kilobots :

- kiloGUI
- kilolib
- git

Le script pour lancer ce container est disponible sur la page dockerhub : https://hub.docker.com/r/juandres/kilodocker_lite/

Exemple : `./run_kilodocker_lite /Document/Kilobots`

¹<https://docs.docker.com/install/linux/docker-ce/ubuntu/#install-using-the-repository>

3.2 Kilodocker_full

Contient tous les packages pour le développement de Kilobots :

- kilolib
- kiloGUI
- git
- htop
- vim
- python-matplotlib
- libboost-all-dev
- subversion
- gnuplot
- kate
- leafpad
- lynx
- midori
- eog

Le script pour lancer ce container est disponible sur la page dockerhub : https://hub.docker.com/r/juandres/kilodocker_full/

Exemple : `./run_kilodocker /Document/Kilobots`

4 Compilation et upload

Toute modification faite dans un dossier autre que celui monté au lancement du container ne sera pas sauvegardé.

4.1 Compilation

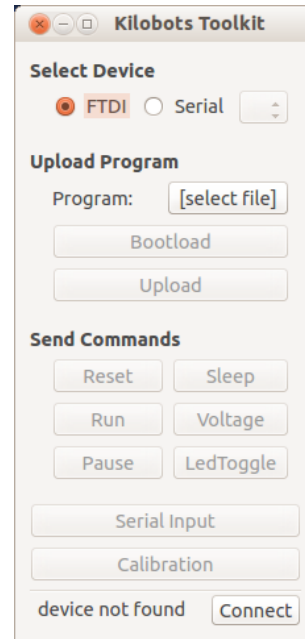
Si il faut compiler qu'un seul fichier, il suffit de copier le script `compileCode.sh` et `MakefileTemplate` (https://github.com/JulienAndres/p_androidKilobot) dans le dossier ou se trouve le fichier. Lancer maintenant le script en passant en paramètre le nom du fichier à compiler sans son extension.

Le fichier `.hex` sera alors placé dans un dossier `build`. (Adapter le Makefile généré si plusieurs fichiers)

Exemple : `./compileCode.sh test`

4.2 Upload du code aux kilobots

- Lancer kiloGUI (kilogui dans le terminal)
- Brancher le contrôleur et allumer les kilobots.
- Cliquer sur "Connect"
- Appuyer sur "Bootload" et attendre que leur LED passe bleu
- Charger le fichier .hex en appuyant sur "[Select file]"
- Envoyer le code en appuyant sur "Upload"
- Le clignotement des LEDs indique que le code est en train d'être envoyé.
- Une fois l'upload fini, appuyer sur "Run" pour démarrer les kilobots



5 Kilombo

Kilombo est un simulateur de kilobots, écrit en C. Il n'est pas inclut dans le docker. Toutefois, si vous voulez le rajouter, il suffit de décommenter les lignes correspondantes dans le Dockerfile (https://github.com/JulienAndres/p_androidKilobot/tree/master/kilodocker) et de rebuild l'image docker grâce à la commande :

- `docker build dossier_du_Dockerfile`