
brickhid - Connexion directe entre un joystick Bluetooth et les LEGO Mindstorms EV3

Copyright © 2014 pabr@pabr.org
Tous droits réservés. (All rights reserved.)

Comment contrôler les LEGO Mindstorms EV3 avec des périphériques Bluetooth tels que les manettes SIXAXIS et DualShock 3 de la PS3.

READ THE HYPERTEXT VERSION HERE:
<http://www.pabr.org/bricks/brickhid/brickhid.fr.html>

Historique des versions		
1.1	2015-10-21	Ajouté une recommandation sur les systèmes d'exploitation alternatifs.
1.0	2014-10-24	Première publication.

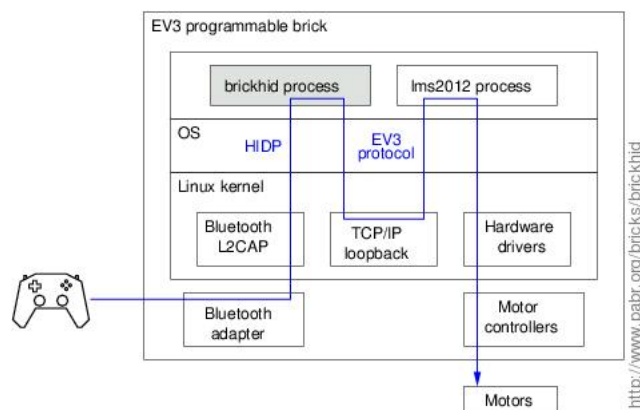
Table des matières

1. Introduction	3
2. Prérequis	3
2.1. Pièces LEGO	3
2.2. Joystick Bluetooth	4
2.3. WiFi	5
2.4. Ordinateur et logiciels tiers	5
3. Logiciel : brickhid	5
3.1. Environnement	5
3.2. Interface Bluetooth	5
3.3. Interface EV3	6
3.4. Fichier de configuration	6
4. Mise en oeuvre	6
4.1. Configuration du WiFi	6
4.2. Installation et démarrage de brickhid	7
4.3. Appariement du joystick Bluetooth avec la brique EV3	8
4.4. Utilisation	8
5. Problèmes identifiés	8
6. Perspectives	8
Bibliographie	9

1. Introduction

La plate-forme LEGO Mindstorms EV3 améliore considérablement la puissance de calcul, la connectivité et la programmabilité par rapport à la génération précédente, Mindstorms NXT. Il y a tellement de nouvelles fonctionnalités que certaines ne sont pas pleinement exploitées. En particulier, la brique EV3 est équipée d'une pile de protocoles Bluetooth générique, mais ne l'utilise que pour communiquer avec des logiciels Mindstorms au moyen d'un protocole spécifique.

Dans ce projet nous montrons comment un petit programme en C tournant sur la brique EV3, **brickhid**, peut gérer la communication en Bluetooth avec un joystick de PlayStation 3 et traduire les mouvements des leviers et les pressions sur les boutons en commandes pour les moteurs.



D'autres avant nous ont déjà montré qu'il est possible d'utiliser des joysticks Bluetooth ou USB avec les Mindstorms EV3 et NXT, mais généralement avec un programme tournant sur un troisième appareil (PC, portable, tablette ou ordinateur embarqué). Ici la connexion Bluetooth s'établit directement entre le joystick et la brique EV3.

D'une façon plus générale, ce projet montre que la brique EV3 peut interagir avec de nombreux périphériques Bluetooth, même sans remplacer le *firmware* d'origine :

- autres modèles de joysticks, claviers, souris, enregistreurs d'activité sportive, capteurs médicaux, montres intelligentes, équipements domotiques, récepteurs GPS ;
- fonctions Bluetooth standard des téléphones mobiles : transfert de fichiers, partage de contacts et d'agendas, gestion des appels téléphoniques, accès aux réseaux mobiles ;
- peut-être également : oreillettes sans fil et autres périphériques audio/vidéo.

Note . Pour exploiter toutes les fonctionnalités de la plate-forme EV3, il est préférable d'installer un système d'exploitation alternatif tel que EV3DEV. Voir par exemple [BRICKFPV].

2. Prérequis

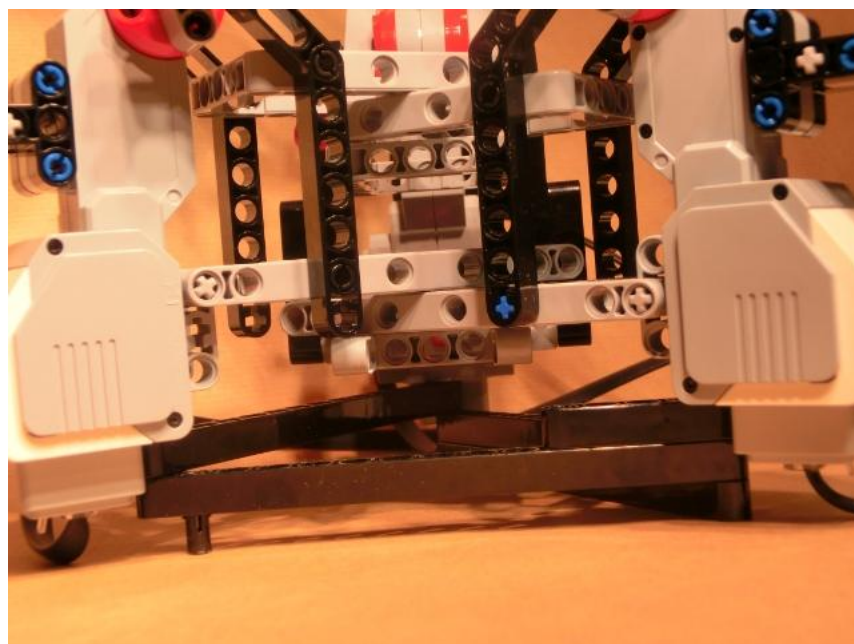
2.1. Pièces LEGO

Le véhicule à chenilles de la vidéo est le modèle TRACK3R standard de la boîte Mindstorms EV3 de base (31313), sauf que la brique programmable a été déplacée pour laisser apparaître le port USB Hôte.

Notre robot delta utilise des pièces de la boîte EV3 Éducation Expansion (45560), ainsi qu'un grand moteur EV3 supplémentaire (45502).



Le positionnement n'est pas très précis (du moins par rapport à l'état de l'art des robots delta), malgré les efforts réalisés pour rigidifier le châssis avec trois couches de poutres et de cadres rectangulaires imbriqués à 120°. Malgré cela, ce modèle peut servir pour l'initiation aux robots à cinématiques parallèles. Pour améliorer la précision, il faudrait envisager d'ajouter une réduction à engrenages et un système de pré-tensionnement.



2.2. Joystick Bluetooth

Ce projet a été testé avec un SIXAXIS de 2007. Le DualShock 3 devrait convenir également.

On pourra facilement obtenir ces périphériques d'occasion à faible coût au fur et à mesure que les nouvelles générations de consoles envahissent le marché.

2.3. WiFi

Nous utilisons WiFi pour télécharger des fichiers vers la brique EV3 et accéder à sa ligne de commande. C'est la façon la plus simple de tester **brickhid**. Attention, le *firmware* EV3 d'origine n'est compatible qu'avec un seul modèle d'adaptateur WiFi (WNA1100 0846:9030).

Il est probablement possible d'exécuter des programmes à partir d'une carte microSD contenant un *firmware* modifié, mais nous n'avons pas exploré cette approche.

2.4. Ordinateur et logiciels tiers

Un PC Linux générique avec un compilateur croisé pour ARM est recommandé pour compiler **brickhid**, obtenir un shell sur la brique EV3, transférer des fichiers, apparier le joystick Bluetooth et démarrer le programme.

Les utilisateurs ne disposant pas d'un compilateur croisé peuvent soit utiliser notre binaire **brickhid** précompilé, soit compiler et exécuter **brickhid** sur le PC plutôt que sur la brique EV3. La communication avec le *runtime* Mindstorms s'établira alors par WiFi plutôt que sur l'interface loopback.

Ce projet n'utilise pas l'environnement graphique de programmation Mindstorms pour Windows/OSX.

3. Logiciel : brickhid

Télécharger le code source : [brickhid-1.0.tar.gz](http://www.pabr.org/bricks/brickhid/brickhid-1.0.tar.gz)

Voir le code source : [brickhid-1.0/](http://www.pabr.org/bricks/brickhid/brickhid-1.0/)

Compiler :

```
user@PC $ wget http://www.pabr.org/bricks/brickhid/brickhid-1.0.tar.gz
user@PC $ tar xzf brickhid-1.0.tar.gz
user@PC $ cd brickhid-1.0
user@PC $ make
```

3.1. Environnement

La brique EV3 est basée sur un Linux embarqué tournant sur un processeur ARM.

La plate-forme ne dispose que de 64 Mo de RAM dont une bonne partie est déjà utilisé par le système de fichiers (cramfs), le système d'exploitation, le *runtime* Mindstorms, et les programmes EV3 de l'utilisateur (chargés en ramdisk). Le processus *runtime* Mindstorms consomme environ 50 % du CPU au repos. **brickhid** ne peut donc pas se permettre de consommer trop de ressources. D'où la décision de programmer en C.

3.2. Interface Bluetooth

La brique EV3 contient un adaptateur Bluetooth générique et le *firmware* inclut BlueZ, l'implémentation Bluetooth standard de Linux.

La norme Bluetooth définit des "profils", qui sont essentiellement des protocoles dédiés pour toutes sortes d'appareils. Pour les claviers, souris et joystick, le profil qui nous intéresse s'appelle HIDP (*Human Interface Device Profile*). HIDP est supporté dans la plupart des distributions Linux, et les claviers et souris Bluetooth fonctionnent généralement sans difficulté. Le noyau Linux fournit aussi une interface

appelée HIDRAW qui permet aux applications d'utiliser des périphériques HID plus exotiques, tels que certains joysticks.

Malheureusement le noyau Linux du *firmware* EV3 d'origine a été compilé sans HIDP ni HIDRAW, probablement pour économiser de la mémoire. On peut le vérifier comme suit pour voir si cette situation évolue dans les versions futures :

```
root@EV3:~# zcat /proc/config.gz | grep 'HIDP\|HIDRAW'
# CONFIG_BT_HIDP is not set
# CONFIG_HIDRAW is not set
```

Par conséquent, **brickhid** doit s'interfacer avec la pile Bluetooth à un niveau plus bas : L2CAP. **brickhid** ouvre une socket Bluetooth L2CAP et accepte des connexions sur les PSMs 17 and 19 qui sont alloués à HIDP.

brickhid n'implémente pas l'intégralité de la norme HID. La version actuelle ne sait décoder que les messages du joystick de la PS3. Mais il est facile d'ajouter des décodeurs pour d'autres périphériques HID.

3.3. Interface EV3

brickhid ne contrôle pas les moteurs directement. Il envoie des *bytecodes* EV3 (dont des commandes pour les moteurs) au *runtime* Mindstorms, **lms2012**. La communication se fait par une connexion TCP loopback, avec le protocole EV3 qui est utilisé sur WiFi.

Le protocole EV3 prévoit une commande pour asservir la position angulaire d'un moteur [https://github.com/mindboards/ev3sources/blob/master/lms2012/c_output/source/c_output.c#L38], mais elle n'est apparemment pas implémentée [https://github.com/mindboards/ev3sources/blob/master/lms2012/d_pwm/Linuxmod_AM1808/d_pwm.c#L2492]. **brickhid** implémente une boucle d'asservissement simple en utilisant les commandes **OUTPUT_GET_COUNT** [https://github.com/mindboards/ev3sources/blob/master/lms2012/c_output/source/c_output.c#L47] et **OUTPUT_SPEED** [https://github.com/mindboards/ev3sources/blob/master/lms2012/c_output/source/c_output.c#L32]. Ceci induit une latence non négligeable dans la démonstration avec le robot delta.

3.4. Fichier de configuration

Au démarrage **brickhid** lit un fichier de configuration qui spécifie comment les pressions sur les boutons et les mouvements des leviers doivent être converties en commandes pour les moteurs.

brickhid-1.0/ps3_tank.config

brickhid-1.0/ps3_delta.config

4. Mise en oeuvre

4.1. Configuration du WiFi

- Brancher l'adaptateur WiFi dans le port USB Host de la brique EV3.
- Allumer la brique EV3.
- Dans les menus EV3 :
 - dérouler le menu "outils" ;
 - sélectionner "WiFi" ;

- cocher la case "WiFi" ;
- sélectionner "Connections" ;
- sélectionner le point d'accès WiFi dans la liste, puis attendre ;
- saisir la clé WEP/WPA si nécessaire ;
- sélectionner "*Connect*" ;
- Dans les menus EV3 :
 - dérouler le menu "outils" ;
 - sélectionner "Brick info" ;
 - conserver l'adresse IP de la brique EV3 (par exemple 192.168.1.2) pour un usage ultérieur.

4.2. Installation et démarrage de brickhid

- Dans un shell sur un PC connecté au même réseau WiFi :

```
user@PC $ telnet 192.168.1.2
login: root
root@EV3:~#
```

(Remplacer 192.168.1.2 par l'adresse WiFi de la brique EV3 obtenue précédemment.)

```
root@EV3:~# dropbear
root@EV3:~# hcitool dev
Devices:
    hci0    11:22:33:44:55:66
```

Conserver l'adresse Bluetooth de la brique EV3 (11:22:33:44:55:66 dans cet exemple) pour un usage ultérieur.

- Dans un autre shell, si vous n'avez pas compilé les sources, il faut télécharger l'exécutable ARM et un exemple de fichier de configuration :

```
user@PC $ wget http://www.pabr.org/bricks/brickhid/brickhid-1.0/brickhid.arm
user@PC $ wget http://www.pabr.org/bricks/brickhid/brickhid-1.0/ps3_tank.config
```

Puis les installer sur la brique EV3 :

```
user@PC $ scp brickhid.arm ps3_tank.config root@192.168.1.2:/var/volatile/
password: (press return)
```

(Remplacer 192.168.1.2 par l'adresse WiFi de la brique EV3 obtenue précédemment.)

- Dans le premier shell :

```
root@EV3:~# killall -STOP udevd
root@EV3:~# killall bluetoothd
root@EV3:~# hciconfig hci0 up pscan
root@EV3:~# cd /var/volatile
```

```
root@EV3:~# chmod a+x brickhid.arm
root@EV3:~# ./brickhid.arm ps3_tank.config
```

brickhid s'interface alors avec le *runtime* Mindstorms et émet une tonalité grave. Il attend ensuite une connexion Bluetooth.

4.3. Appariement du joystick Bluetooth avec la brique EV3

Pour plus d'informations au sujet de cette procédure, voir [SIXLINUX].

- Brancher le joystick PS3 à un PC Linux avec un cordon USB.
- Dans un shell sur le PC :

```
user@PC $ wget http://www.pabr.org/sixlinux/sixpair.c
user@PC $ gcc -o sixpair sixpair.c -lusb
user@PC $ sudo ./sixpair 11:22:33:44:55:66
Current Bluetooth master: xx:xx:xx:xx:xx:xx
Setting master bd_addr to 11:22:33:44:55:66
```

(Remplacer 11:22:33:44:55:66 par l'adresse Bluetooth de la brique EV3 obtenue précédemment.)

- Débrancher le joystick PS3 du PC.

4.4. Utilisation

- Presser le bouton PS du joystick PS3. Lorsque le joystick PS3 réussit à se connecter à la brique EV3, **brickhid** émet une tonalité aiguë.
- À partir de ce moment **brickhid** convertit les mouvements des leviers et les pressions sur les boutons en commandes pour les moteurs comme indiqué dans son fichier de configuration.
- Vous pouvez arrêter **brickhid** en appuyant sur le bouton PS. Alternativement, vous pouvez aussi envoyer Control+C sur la ligne de commande ou presser le bouton au dos du joystick PS3 avec un trombone.

5. Problèmes identifiés

- Pour utiliser des applications Mindstorms communiquant par Bluetooth en même temps que le joystick, il faut probablement éditer `/etc/bluetooth/main.conf`, décommenter la ligne **DisablePlugins=network,input**, et redémarrer **bluetoothd** au lieu de le désactiver complètement.
- L'interface TCP du *runtime* Mindstorms n'est apparemment disponible que lorsque le WiFi est activé. **brickhid** s'arrête lorsqu'on désactive le WiFi manuellement et lorsqu'on retire l'adaptateur du port USB.

6. Perspectives

- supporter davantage de périphériques HID ;
- faire en sorte que le *runtime* Mindstorms voie les périphériques HID comme des capteurs EV3, afin que des programmes EV3 puissent les utiliser ;
- supporter les fonctions de retour HID telles que les vibrations du DualShock 3 et les LEDs du PlayStation Move.

Bibliographie

[SIXLINUX] *Utilisation du joystick de la PlayStation 3 en mode Bluetooth avec Linux* . <http://www.pabr.org/sixlinux/sixlinux.fr.html> .

[BRICKFPV] *Pilotage en immersion avec les LEGO Mindstorms EV3* . <http://www.pabr.org/bricks/brickfpv/brickfpv.fr.html> .