Bras Robot Ksr10 USB

Configuration du raspberryPi3 pour les commandes USB

Script Python pour contrôler le bras robotique USB Maplin sur votre RaspberryPI.

Prerequis :

```
  Raspian
  Python 2.7
  pip
  pyusb Library
```

Installation :

Créez un nouveau fichier de règles udev dans /etc/udev/rules.d/85-robotarm.rules avec le contenu

```
  SUBSYSTEM=="usb", ATTRS{idVendor}=="1267", ATTRS{idProduct}=="0000",
ACTION=="add", GROUP="plugdev", MODE="0666"
```

Ajoutez votre utilisateur au groupe plugdev en utilisant la commande:

```
  sudo usermod -aG plugdev yourusername
```

Redémarrez le Pi avec la commande:

```
  sudo shutdown -r now
```

Rendre le script exécutable avec la commande:

```
  chmod 755 maplinrobot.py
```

Installez pip avec la commande:

```
  sudo apt-get install python-pip -y
```

Istallez la bibliothèque pyusb via pip avec la commande:

```
  sudo pip install pyusb
```

Ouvrez le script et modifiez-le en fonction de vos besoins (voir la section Utilisation d'exemples pour plus d'informations).

Exécuter ./maplinrobot.py . Si vous rencontrez des problèmes lors de l'exécution en tant qu'utilisateur normal, essayez d'exécuter le script en tant que root.

Déplacement du bras

Les commandes sont stockées dans un dictionnaire. Les commandes valides à envoyer au bras sont:

Regarder le Bras, l' USB étant à gauche du Bras.

```
  'base-anti-clockwise'M5 - Tourne la base dans le sens inverse des
aiguilles d'une montre
  'base-clockwise' M5 - Tourne la base dans le sens  des aiguilles d'une
montre
  'shoulder-up' M4- soulève l'épaule   D à G
  'Shoulder-down' M4- Diminue l'épaule G à D
  'elbow-up' M3- soulève le coude
  'elbow-down' M3- Abaisse le coude
  'wrist-up' M2- soulève le poignet  Bas vers Haut
  'wrist-down' M2- Diminue le poignet Haut vers Bas
  'Grip-open' M1- Ouvre la poignée
  'Grip-close' M1- Ferme la poignée
  'Light-on' - Allume la LED dans la poignée
  'Light-off' - Éteint la LED de la poignée
  'Stop' - Arrête tout mouvement du bras
```

Exemple d'utilisation :

maplinrobot.py

Au bas du script, décommettez les lignes ci-dessous:

s = MaplinRobot() s.MoveArm(t=1.0, cmd='base-clockwise')

Cela fera tourner la base du bras dans le sens des aiguilles d'une montre pendant 1 seconde. La durée de chaque commande est définie en passant une valeur flottante au paramètre t .

L'argument passé au paramètre cmd peut être n'importe quelle commande dans la section

Déplacement du bras ci-dessus.

Mouvement maxi des axes

M5 (la base) = 12 s de gauche à droite MAXI (6s en partant du milieu)

M4 = 8 s de gauche à droite MAXI (4s en partant de la position verticale)

M3 = 8 s de bas en haut MAXI (7s pour baisser le bras, 1s seconde en moins du fait du poids du bras qui descend plus vite...)

M2 = 7 s de bas en haut MAXI (la pince étant baissée au maximum)

M1(la pince) = 1 s pour ouvrir lorsque la pince est fermée

Code Python utiliser pour la pince

```
def M1Ferme(deplam1f):
   s.MoveArm(deplam1f, cmd='grip-close')
   s.MoveArm(1, cmd='stop')
```

```
def M1Ouvert(deplam1o):
   s.MoveArm(deplam1o, cmd='grip-open')
   s.MoveArm(1, cmd='stop')
```

```
 M1Ferme(1.25)
 M1Ouvert(0)
```

Drivers KSR10 pour Win10

- Programme et drivers pour robot KSR10 win10 pilotage USB

From:
https://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
https://chanterie37.fr/fablab37110/doku.php?id=start:rasberry:ksr10

Last update: **2023/01/27 16:08**