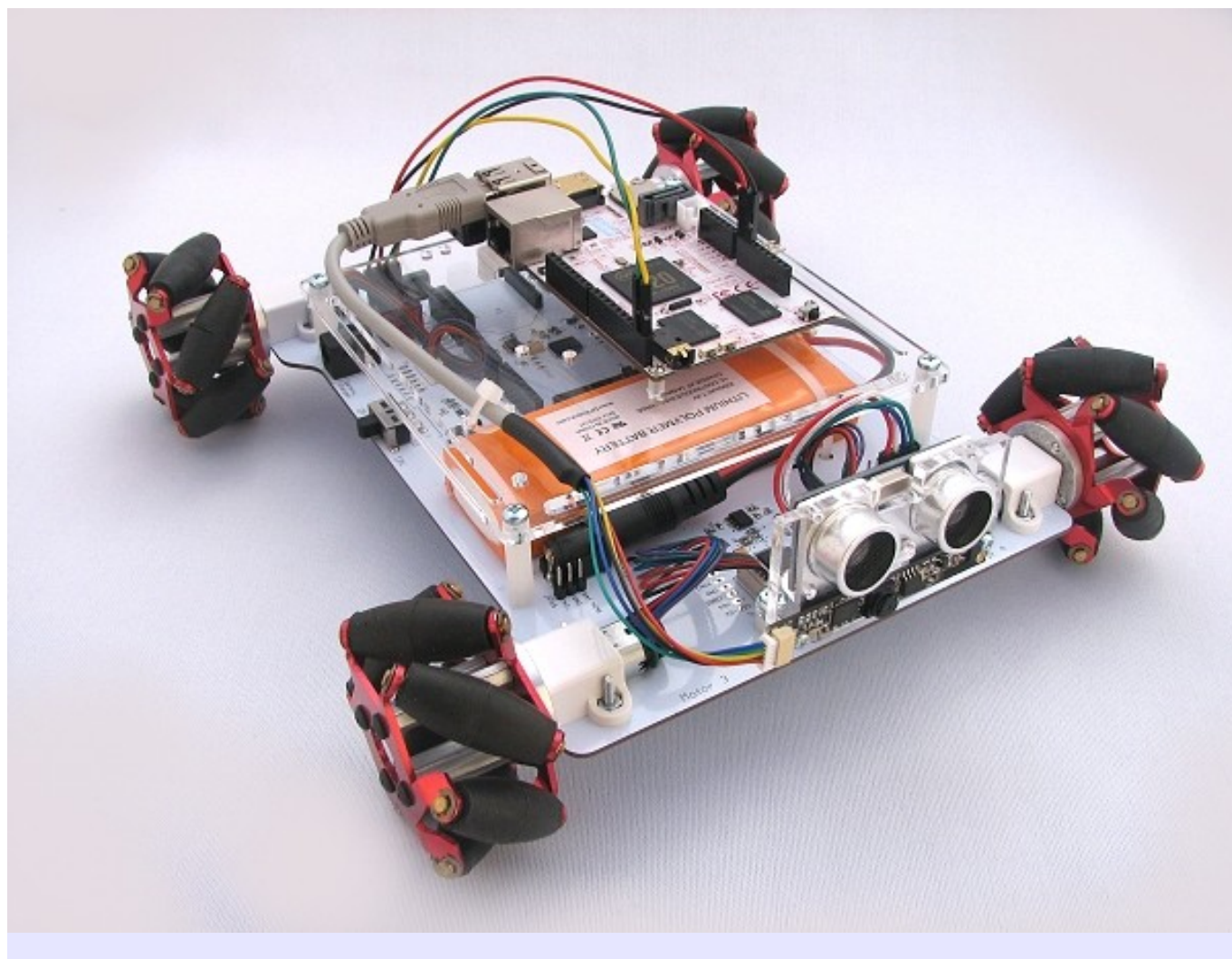




T-QUAD

API PYTHON



Date de dernière mise à jour : 08/06/2017

Table des matières

1 - Introduction.....	<u>3</u>
2 - Principe de fonctionnement.....	<u>4</u>
3 - Fonctions de l'API.....	<u>5</u>
4 - Utilisation.....	<u>14</u>
4.1 - Exécution du serveur de pilotage.....	<u>14</u>
4.1.1 - Lancement du serveur de pilotage en ligne de commande.....	<u>14</u>
4.1.2 - Lancement du serveur de pilotage via MyViz.....	<u>17</u>
4.2 - Exemples d'utilisation de l'API.....	<u>19</u>
4.2.1 - Suivi de cercle en « translation ».....	<u>19</u>
4.2.2 - Suivi de cercle en « rotation ».....	<u>20</u>
4.2.3 - Mouvements successifs.....	<u>20</u>
5 - Important.....	<u>21</u>

1 - Introduction

Cette documentation présente l'API de pilotage en Python, c'est-à-dire un ensemble de fonctions de haut niveau utilisables dans un script ou dans l'interpréteur Python et permettant d'interagir avec le robot.

Les programmes et applications associés à ces activités sont téléchargeables ici :

<https://github.com/3sigma/T-Quad-API-Python>

2 - Principe de fonctionnement

L'utilisation de l'API Python met en œuvre deux composantes :

- un serveur de pilotage du robot, qui doit au préalable être lancé (voir plus loin) :
 - soit en ligne de commande
 - soit via une application MyViz
- des commandes exécutées sur le robot :
 - soit dans un script Python
 - soit interactivement dans l'interpréteur Python

Plus précisément :

- les commandes de l'API permettant de modifier l'état du robot (par exemple pour lui donner une consigne de vitesse longitudinale) envoient des données au serveur de pilotage, qui actionne les moteurs en conséquence
- les commandes de l'API permettant de lire les valeurs de variables du robot réceptionnent ces valeurs lorsqu'elles sont envoyées par le serveur de pilotage

3 - Fonctions de l'API

L'API de pilotage de T-Quad en Python (« TQuad_API ») se trouve dans le fichier API.py, dans le répertoire « programmes_python » de l'archive.

Elle contient les fonctions suivantes :

TQuad_API()

- Description : fonction d'initialisation à exécuter impérativement au début de chaque programme

TensionMoteurArriereDroit(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de tension d'alimentation au moteur arrière droit pendant une certaine durée.

TensionMoteurArriereGauche(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de tension d'alimentation au moteur arrière gauche pendant une certaine durée.

TensionMoteurAvantDroit(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de tension d'alimentation au moteur avant droit pendant une certaine durée.

TensionMoteurAvantGauche(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de tension d'alimentation au moteur avant gauche pendant une certaine durée.

TensionMoteurs(tensionArriereDroit, tensionArriereGauche, tensionAvantDroit, tensionAvantGauche, duree=-1)

- Paramètres :
 - tensionArriereDroit : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V) du moteur arrière droit, saturée en interne entre -6 et 6V
 - tensionArriereGauche : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V) du moteur arrière gauche, saturée en interne entre -6 et 6V
 - tensionAvantDroit : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V) du moteur avant droit, saturée en interne entre -6 et 6V
 - tensionAvantGauche : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Tension de commande (V) du moteur avant gauche, saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de tension d'alimentation aux quatre moteurs pendant une certaine durée.

TensionMouvementLongitudinal(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de tension d'alimentation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement longitudinal

TensionMouvementLateral(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de tension d'alimentation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement latéral

TensionMouvementPivot(tension, duree=-1)

- Paramètres :
 - tension : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V), saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de tension d'alimentation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement de rotation autour de la verticale

TensionMouvement(tensionMouvementLongitudinal, tensionMouvementLateral, tensionMouvementPivot, duree=-1)

- Paramètres :
 - tensionMouvementLongitudinal : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V) des quatre moteurs, correspondant à un mouvement longitudinal. Cette tension est saturée en interne entre -6 et 6V
 - tensionMouvementLateral : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V) des quatre moteurs, correspondant à un mouvement latéral. Cette tension est saturée en interne entre -6 et 6V
 - tensionMouvementPivot : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Tension de commande (V) des quatre moteurs, correspondant à un mouvement de rotation autour de la verticale. Cette tension est saturée en interne entre -6 et 6V
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de tension d'alimentation aux quatre moteurs pendant une certaine durée, afin de réaliser une combinaison de mouvements

VitesseMoteurArriereDroit(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse de rotation au moteur arrière droit pendant une certaine durée.

VitesseMoteurArriereGauche(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse de rotation au moteur arrière gauche pendant une certaine durée.

VitesseMoteurAvantDroit(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse de rotation au moteur avant droit pendant une certaine durée.

VitesseMoteurAvantGauche(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse de rotation au moteur avant gauche pendant une certaine durée.

VitesseMoteurs(vitesseArriereDroit, vitesseArriereGauche, vitesseAvantDroit, vitesseAvantGauche, duree=-1)

- Paramètres :
 - vitesseArriereDroit : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s) du moteur arrière droit, saturée en interne entre -20 et 20 rad/s
 - vitesseArriereGauche : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s) du moteur arrière gauche, saturée en interne entre -20 et 20 rad/s
 - vitesseAvantDroit : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s) du moteur avant droit, saturée en interne entre -20 et 20 rad/s
 - vitesseAvantGauche : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s) du moteur avant gauche, saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de vitesse de rotation aux quatre moteurs pendant une certaine durée.

VitesseMoteursMouvementLongitudinal(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de vitesse de rotation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement longitudinal

VitesseMoteursMouvementLateral(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de vitesse de rotation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement latéral

VitesseMoteursMouvementPivot(vitesse, duree=-1)

- Paramètres :
 - vitesse : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (rad/s), saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de vitesse de rotation aux quatre moteurs pendant une certaine durée, pour que le robot ait un mouvement de rotation autour de la verticale

VitesseMoteursMouvement(vitesseMoteurMouvementLongitudinal, vitesseMoteurMouvementLateral, vitesseMoteurMouvementPivot, duree=-1)

- Paramètres :
 - vitesseMoteurMouvementLongitudinal : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse (rad/s) des quatre moteurs, correspondant à un mouvement longitudinal. Cette consigne est saturée en interne entre -20 et 20 rad/s
 - vitesseMoteurMouvementLateral : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse (rad/s) des quatre moteurs, correspondant à un mouvement latéral. Cette consigne est saturée en interne entre -20 et 20 rad/s
 - vitesseMoteurMouvementPivot : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse (rad/s) des quatre moteurs, correspondant à un mouvement de rotation autour de la verticale. Cette consigne est saturée en interne entre -20 et 20 rad/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne des consignes de vitesse aux quatre moteurs pendant une certaine durée, afin de réaliser une combinaison de mouvements

AvancerLongitudinal(vitesseLongitudinale, duree=-1)

- Paramètres :
 - vitesseLongitudinale : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse longitudinale (m/s), saturée en interne entre -0.5 et 0.5 m/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse longitudinale au robot pendant une certaine durée

AvancerLateral(vitesseLaterale, duree=-1)

- Paramètres :
 - vitesseLaterale : réel ou chaîne de caractères (expression Python valide du type ' $3 * \text{math.sin}(t)$ ', t étant reconnu comme le temps courant). Consigne de vitesse latérale (m/s), saturée en interne entre -0.5 et 0.5 m/s
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une consigne de vitesse latérale au robot pendant une certaine durée

Pivoter(vitesseRotation, source_ximes, duree=-1)

- Paramètres :
 - vitesseRotation : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (deg/s), saturée en interne entre -360 et 360 deg/s
 - source_ximes : entier. Définition de la source de mesure de rotation :
 - 0 : odométrie (calcul à partir de la vitesse de rotation des roues). Peu précis
 - 1 : mesure par l'axe z du gyroscope. Plus précis mais plus bruité
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne au robot une consigne de vitesse de rotation autour de la verticale pendant une certaine durée

Mouvement(vitesseLongitudinale, vitesseLaterale, vitesseRotation, source_ximes, duree=-1)

- Paramètres :
 - vitesseLongitudinale : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Consigne de vitesse longitudinale (m/s), saturée en interne entre -0.5 et 0.5 m/s
 - vitesseLaterale : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Consigne de vitesse latérale (m/s), saturée en interne entre -0.5 et 0.5 m/s
 - vitesseRotation : réel ou chaîne de caractères (expression Python valide du type '3 * math.sin(t)', t étant reconnu comme le temps courant). Consigne de vitesse de rotation (deg/s), saturée en interne entre -360 et 360 deg/s
 - source_ximes : entier. Définition de la source de mesure de rotation :
 - 0 : odométrie (calcul à partir de la vitesse de rotation des roues). Peu précis
 - 1 : mesure par l'axe z du gyroscope. Plus précis mais plus bruité
 - duree : réel (optionnel). Durée de la manœuvre en secondes. Valeur par défaut : -1 (la valeur de la tension reste constante jusqu'à ce qu'elle soit changée par une nouvelle commande)
- Description : cette fonction donne une combinaison de consignes de mouvement au robot pendant une certaine durée

InitialiserPsi(valeur)

- Paramètres :
 - valeur : réel. Valeur d'initialisation (rad)
- Description : cette fonction réinitialise l'intégrateur permettant de calculer l'angle de rotation du robot à partir de l'intégration de la mesure du gyroscope autour de l'axe vertical

LireVariable(variable)

- Paramètres :
 - variable : chaîne de caractères. Variable à lire, parmi la liste suivante :
 - Temps : temps courant (s)
 - omegaArriereDroit : vitesse de rotation du moteur arrière droit (rad/s)
 - omegaArriereGauche : vitesse de rotation du moteur arrière gauche (rad/s)
 - omegaAvantDroit : vitesse de rotation du moteur avant droit (rad/s)
 - omegaAvantGauche : vitesse de rotation du moteur avant gauche (rad/s)
 - vxmes : composante longitudinale de vitesse du robot (m/s)
 - vymes : composante latérale de vitesse du robot (m/s)
 - ximes : composante de rotation autour de la verticale de la vitesse du robot (rad/s)
 - distance : distance brute mesurée par le capteur de distance (cm)
 - distanceFiltre : distance filtrée (cm)
 - gz : vitesse de rotation autour de la verticale mesurée par le gyroscope (rad/s)
 - psi : angle de rotation du robot autour de la verticale, calculée à partir de l'intégration de gz (rad)
 - L1 : tension mesurée par le capteur de suivi de ligne n°1 (entre 0 et 5V)
 - L2 : tension mesurée par le capteur de suivi de ligne n°2 (entre 0 et 5V)
 - L3 : tension mesurée par le capteur de suivi de ligne n°3 (entre 0 et 5V)
- Description : cette fonction permet de lire la valeur d'une variable du robot

4 - Utilisation

4.1 - Exécution du serveur de pilotage

Cette étape est indispensable car l'API interagit avec un serveur de pilotage, sans lequel rien ne peut fonctionner (voir chapitre 2).

Ce serveur peut être exécuté en ligne de commande ou via une application MyViz.

Tout ce qui suit suppose que le robot est allumé et que l'ordinateur y est connecté, soit en Wifi, soit en Ethernet (voir la documentation générale du robot : https://github.com/3sigma/T-Quad/raw/master/T-Quad_General.pdf).

4.1.1 - Lancement du serveur de pilotage en ligne de commande

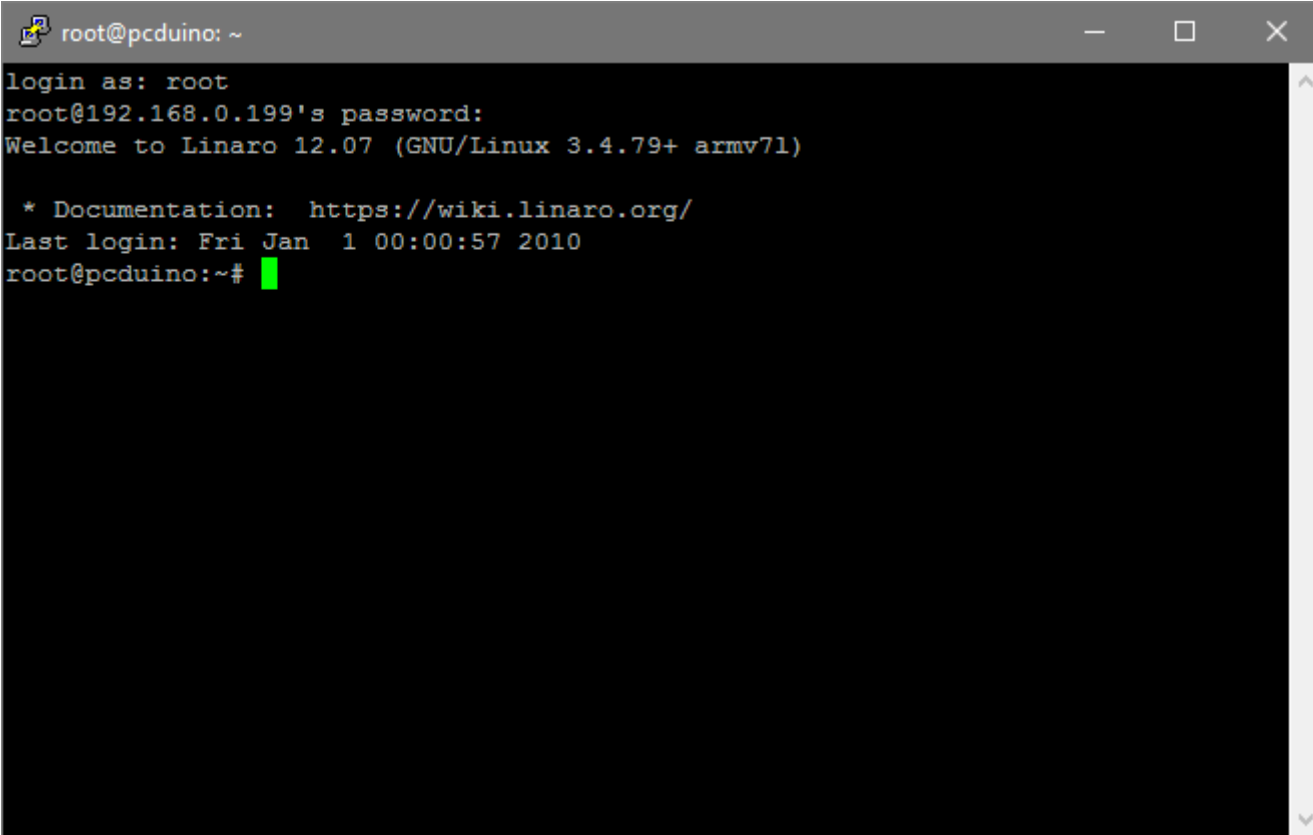
Le lancement du serveur de pilotage en ligne de commande nécessite tout d'abord de se connecter au robot en SSH. Ceci se fait de la façon suivante, en fonction de la plate-forme utilisée :

- Windows : utiliser PuTTY (<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>).
Après avoir lancé l'exécutable (putty.exe) ouvrir une connexion SSH avec les paramètres suivant :
 - IP : 192.168.0.199 (adresse par défaut du robot en Wifi ; si vous êtes connecté en Ethernet filaire, l'adresse IP du robot est affichée sur son écran OLED)
 - port : 22
 - login : root
 - mot de passe : tqad
- Linux ou macOS :
 - ouvrir un terminal et exécuter la commande suivante :

```
ssh root@192.168.0.199
```


(192.168.0.199 est l'adresse par défaut du robot en Wifi ; si vous êtes connecté en Ethernet filaire, l'adresse IP du robot est affichée sur son écran OLED)
 - entrer le mot de passe : tqad

Sur Windows, la console PuTTY est similaire à ceci (exemple avec la version pcDuino) :

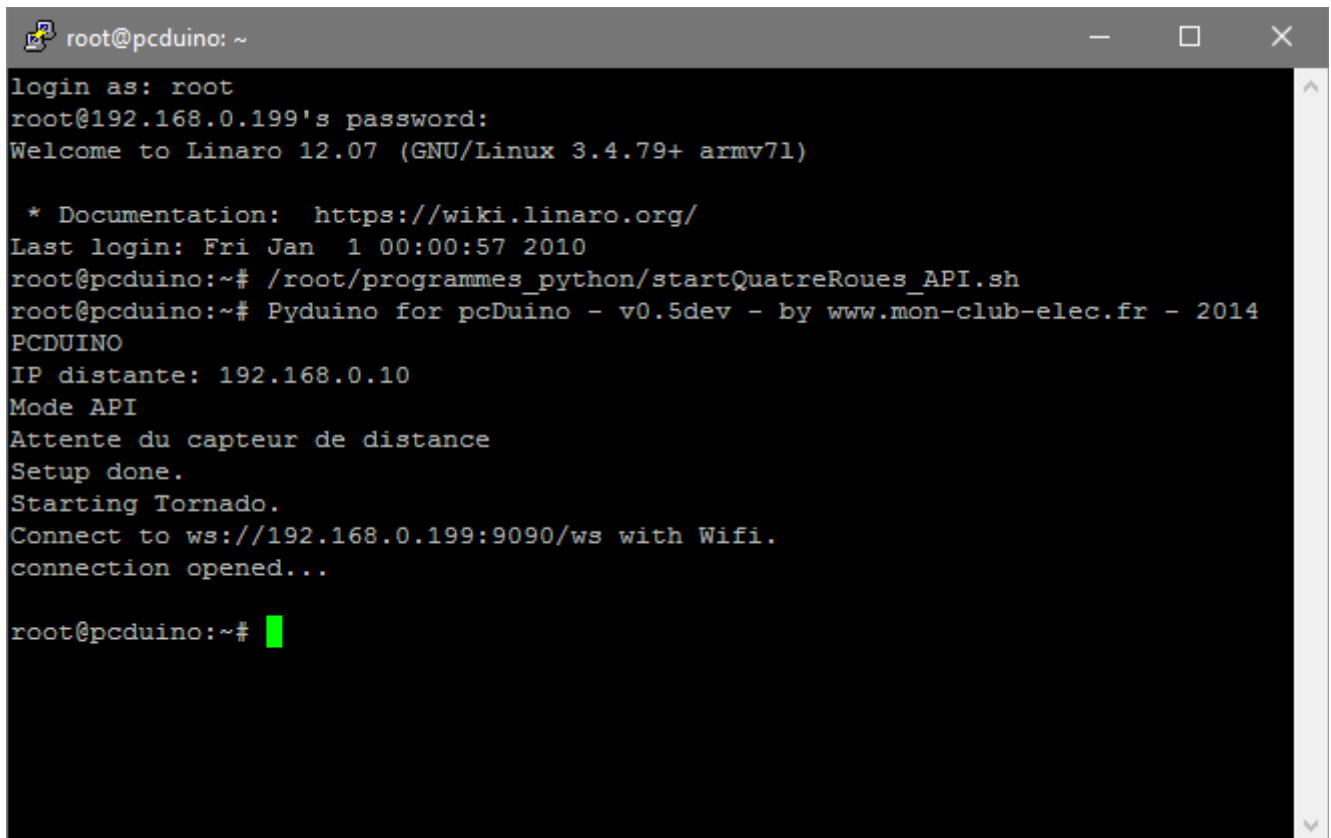


```
root@pcduino: ~  
login as: root  
root@192.168.0.199's password:  
Welcome to Linaro 12.07 (GNU/Linux 3.4.79+ armv7l)  
  
* Documentation: https://wiki.linaro.org/  
Last login: Fri Jan  1 00:00:57 2010  
root@pcduino:~#
```

Le serveur de pilotage se lance alors avec la commande suivante :

```
/root/programmes_python/startQuatreRoues_API.sh
```

Le résultat dans la fenêtre de commande est le suivant :



```
root@pcduino: ~
login as: root
root@192.168.0.199's password:
Welcome to Linaro 12.07 (GNU/Linux 3.4.79+ armv7l)

* Documentation:  https://wiki.linaro.org/
Last login: Fri Jan  1 00:00:57 2010
root@pcduino:~# /root/programmes_python/startQuatreRoues_API.sh
root@pcduino:~# Pyduino for pcDuino - v0.5dev - by www.mon-club-elec.fr - 2014
PCDUINO
IP distante: 192.168.0.10
Mode API
Attente du capteur de distance
Setup done.
Starting Tornado.
Connect to ws://192.168.0.199:9090/ws with Wifi.
connection opened...

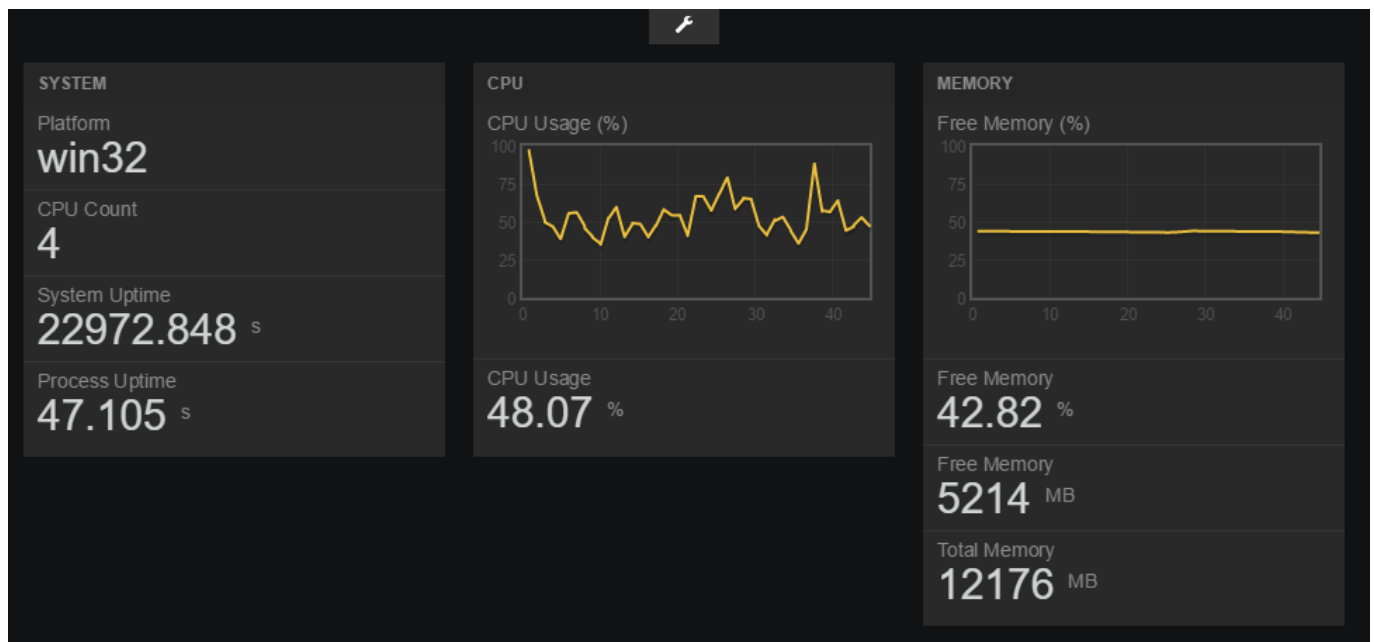
root@pcduino:~#
```

Le serveur de pilotage tourne alors en tâche de fond. Il est maintenant possible d'exécuter des commandes de l'API (voir plus loin).

4.1.2 - Lancement du serveur de pilotage via MyViz

L'utilisation de MyViz pour lancer le serveur de pilotage est intéressante car cela permet en parallèle de visualiser en temps-réel les valeurs de certaines variables du robot.

Après avoir téléchargé et installé le logiciel (<http://www.3sigma.fr/Telechargements-MyViz.html>), lancez son exécution. Le tableau de bord initialement affiché sera similaire à la capture d'écran ci-dessous :



Ce tableau de bord n'est qu'un exemple de ce qui peut être réalisé avec MyViz.

Charger ensuite le tableau de bord relatif à l'activité « API » dans MyViz. Pour cela, il faut tout d'abord récupérer ce dernier sur votre ordinateur, à partir du lien suivant :

https://raw.githubusercontent.com/3sigma/T-Quad-API-Python/master/MyViz/T-Quad_API_Reseau.json

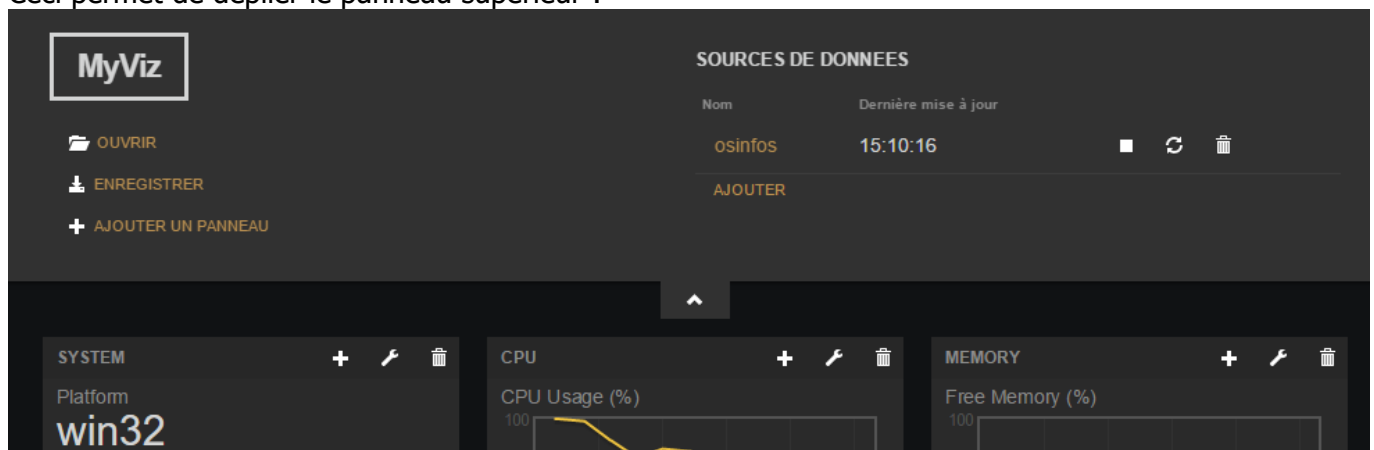
Il se trouve également dans l'archive suivante :

<https://github.com/3sigma/T-Quad-API-Python/archive/master.zip>

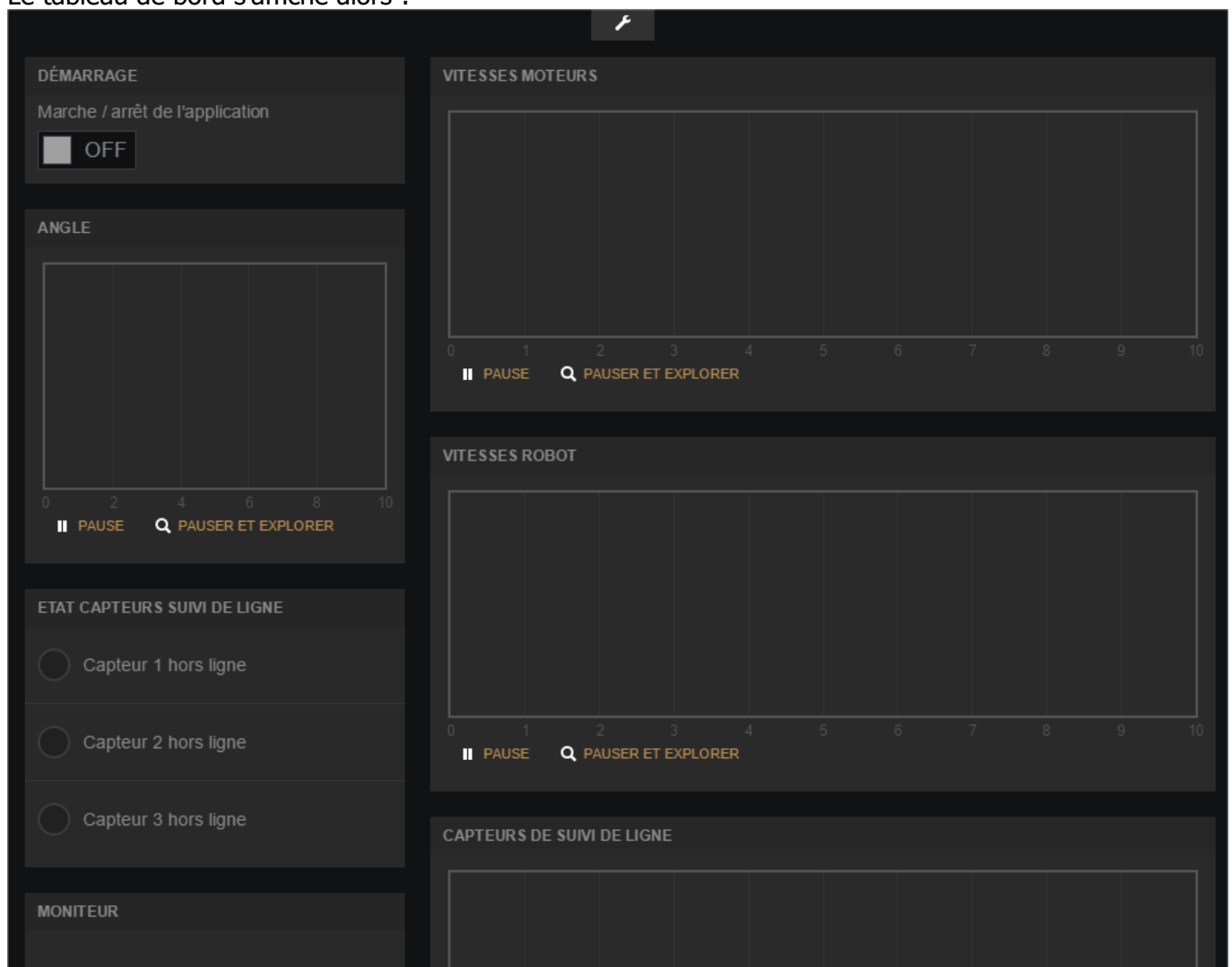
Pour l'ouvrir dans MyViz, il suffit ensuite de cliquer sur la clé en haut de la fenêtre :



Ceci permet de déplier le panneau supérieur :



Cliquez sur « Ouvrir » et sélectionnez le fichier T-Quad_API_Reseau.json que vous venez de télécharger. Le tableau de bord s'affiche alors :



Le démarrage du serveur de pilotage se fait via la bouton marche-arrêt en haut à gauche.

Attention : il faut attendre quelques secondes avant de voir les courbes s'afficher.

4.2 - Exemples d'utilisation de l'API

Les exemples présentés ci-dessous doivent être exécutés sur le robot :

- soit dans l'interpréteur Python, lancé depuis le répertoire /root/programmes_python
- soit sous la forme d'un script préalablement envoyé dans le répertoire /root/programmes_python

Pour échanger des fichiers entre l'ordinateur hôte et le robot, nous recommandons l'utilisation des logiciels suivants

- FileZilla (<https://filezilla-project.org/>): multi-plateforme
- WinSCP (<https://winscp.net/eng/docs/lang:fr>) sur Windows

4.2.1 - Suivi de cercle en « translation »

Dans cet exemple, le robot suit un cercle sans changer de cap. Sa trajectoire est la combinaison de 3 mouvements :

- une translation longitudinale sinusoïdale
- une translation latérale sinusoïdale déphasée de 90° par rapport à la translation
- une rotation nulle autour de la verticale

Nous souhaitons par ailleurs exécuter cette manœuvre pendant 10 s. Le code est le suivant :

```
from API import TQuad_API
```

```
TQuad = TQuad_API()
```

```
TQuad.Mouvement('0.2*math.cos(t)', '0.2*math.sin(t)', 0, 0, 10)
```

4.2.2 - Suivi de cercle en « rotation »

Dans cet exemple, le robot suit un cercle en pointant l'avant vers ce dernier. Sa trajectoire est la combinaison de 3 mouvements :

- une translation longitudinale nulle
- une translation latérale constante (la translation latérale est définie dans le repère lié au robot)
- une rotation constante autour de la verticale

Nous souhaitons par ailleurs exécuter cette manœuvre pendant 13 s, la vitesse de rotation étant mesurée sur l'axe verticale du gyroscope. Le code est le suivant :

```
from API import TQuad_API
```

```
TQuad = TQuad_API()
```

```
TQuad.Mouvement(0, -0.2, 30, 1, 13)
```

4.2.3 - Mouvements successifs

Dans cet exemple, nous demandons au robot de se déplacer successivement en longitudinal, en latéral et en rotation. Le code est le suivant :

```
from API import TQuad_API
```

```
TQuad = TQuad_API()
```

```
TQuad.AvancerLongitudinal(0.2, 2)
```

```
TQuad.AvancerLateral(0.2, 2)
```

```
TQuad.Pivoter(180, 0, 2)
```

5 - Important

T-Quad est un produit « vivant » en constant développement pour l'améliorer ou lui ajouter de nouvelles fonctionnalités. Si vous avez des idées ou des besoins pour des développements spécifiques, n'hésitez pas à nous contacter (support@3sigma.fr).

Ne restez jamais bloqué sans nous contacter !

Pour tout problème ou toute requête, contactez-nous à l'adresse support@3sigma.fr