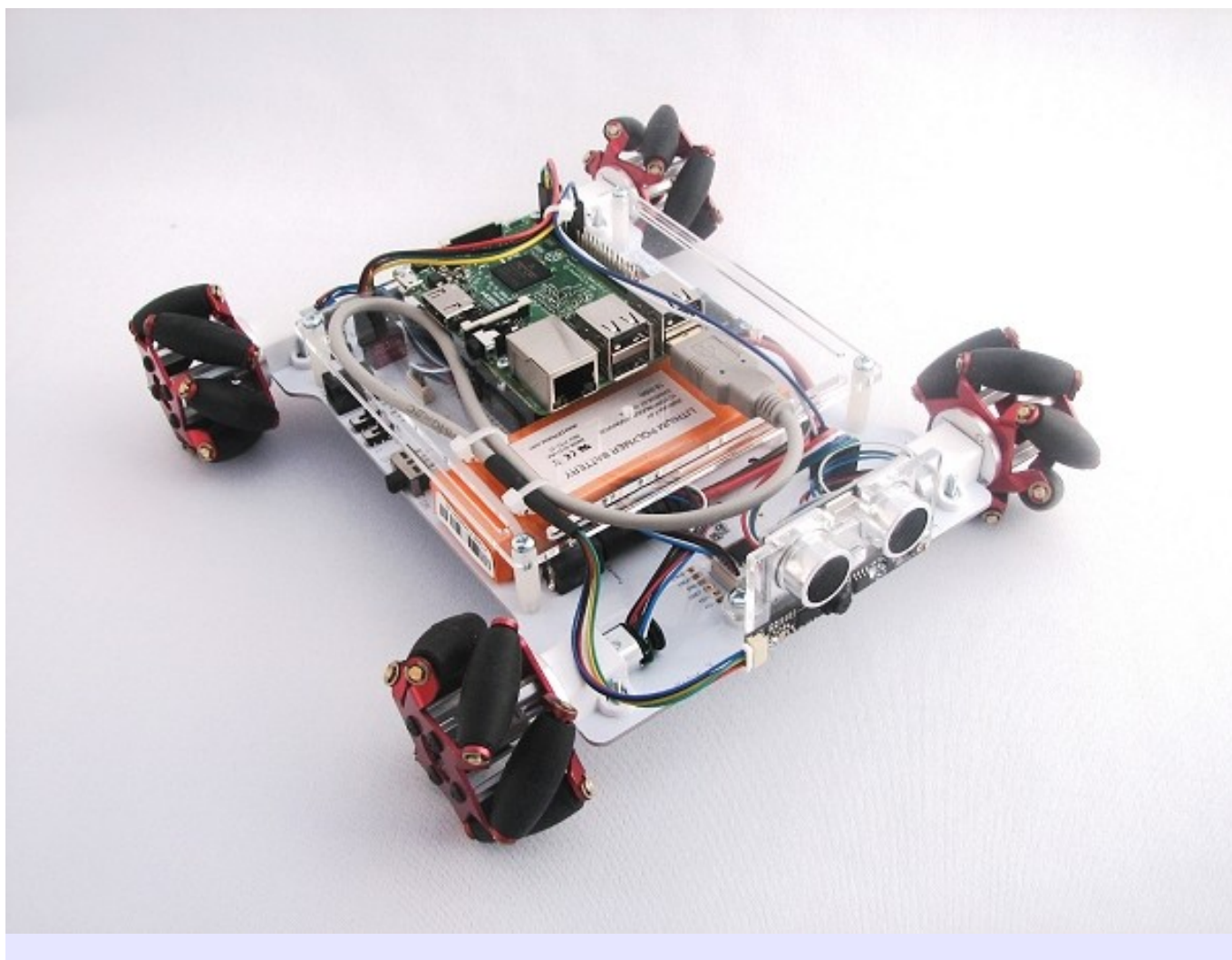




T-QUAD

CONTRÔLE AVEC ROUES HOLONOMES



Date de dernière mise à jour : 07/11/2017

Table des matières

1 - Introduction.....	<u>3</u>
2 - Principe de fonctionnement.....	<u>4</u>
3 - Programme Arduino.....	<u>7</u>
4 - Activités.....	<u>8</u>
4.1 - Contrôle général des mouvements du robot.....	<u>8</u>
4.1.1 - Programme Python.....	<u>8</u>
4.1.2 - Application MyViz.....	<u>9</u>
4.2 - Asservissement d'angle.....	<u>14</u>
4.2.1 - Programme Python.....	<u>14</u>
4.2.2 - Application MyViz.....	<u>15</u>
4.3 - Asservissement de distance avec mouvement latéral.....	<u>19</u>
4.3.1 - Programme Python.....	<u>19</u>
4.3.2 - Application MyViz.....	<u>20</u>
5 - Important.....	<u>24</u>

1 - Introduction

Cette documentation présente différentes activités mettant en œuvre le robot T-Quad avec ses 4 roues holonomes (roues Mecanum).

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

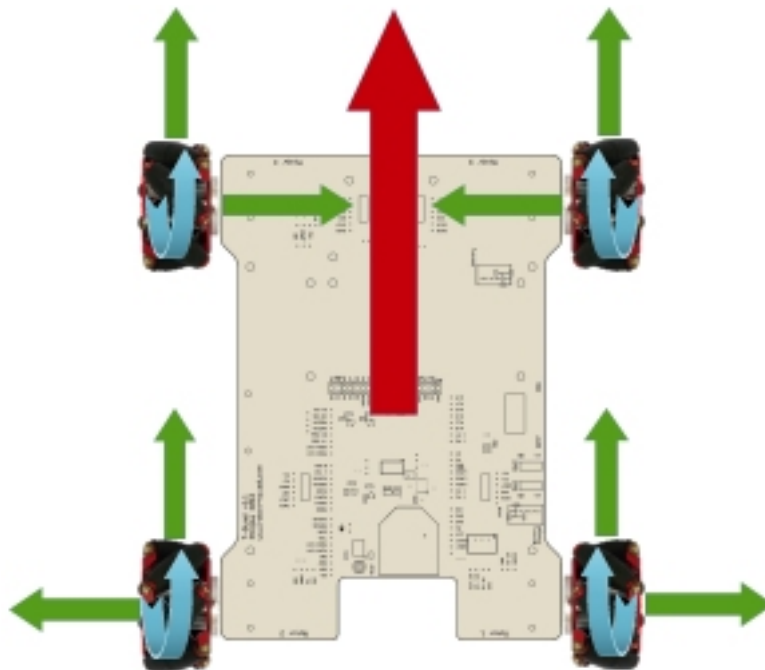
<https://github.com/3sigma/T-Quad-Quatre-Roues>

2 - Principe de fonctionnement

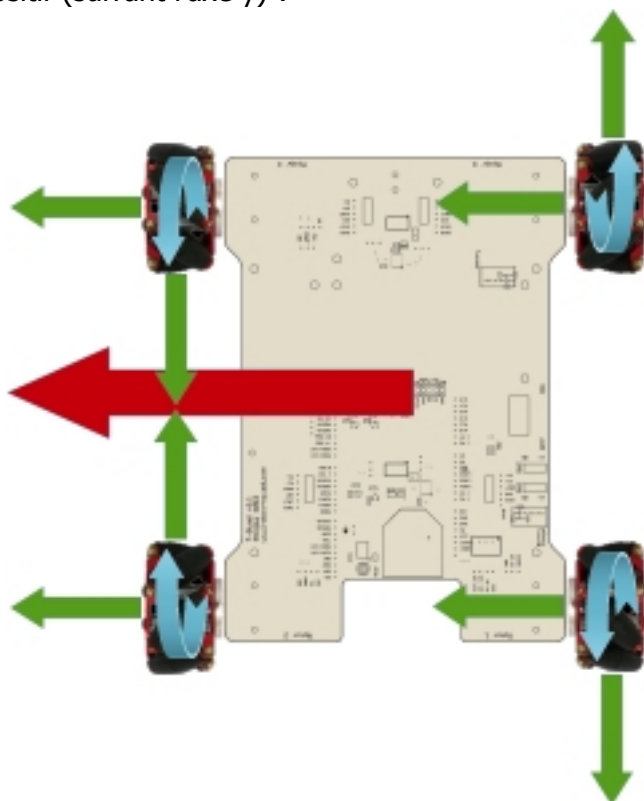
Le contrôle des mouvements du robot se font en fonction de la combinaison des rotations des différentes roues. En effet, chaque roue est constituée d'un assemblage de rouleaux qui, lorsqu'ils sont en contact avec le sol et que la roue tourne, créent deux forces perpendiculaires, l'une parallèle au plan de la roue et l'autre parallèle à son axe de rotation.

Il est ainsi possible de créer 3 types de mouvements en fonction du sens de rotation de chaque roue, en supposant que toutes les vitesses de rotation sont identiques :

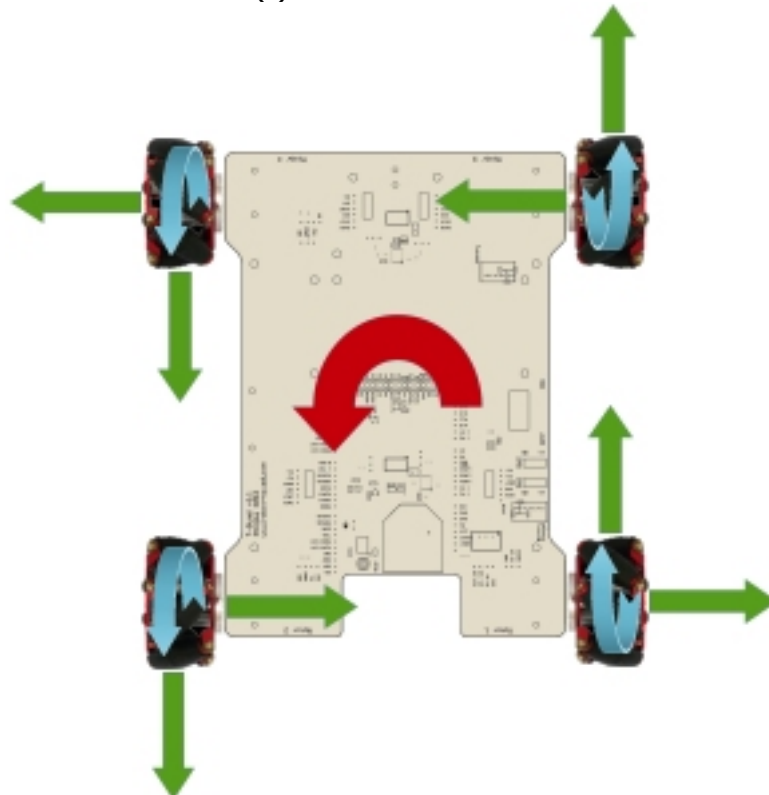
Mouvement longitudinal positif (suivant l'axe x) :



Mouvement latéral positif (suivant l'axe y) :



Rotation positive autour de l'axe vertical (z) :



Bien sûr, ces 3 mouvements peuvent combinés pour faire déplacer le robot dans la direction souhaitée avec l'orientation voulue.

Il est important de noter que les 4 roues ne sont pas identiques : elle sont appairées.

Par ailleurs, le montage doit se faire de telle sorte que pour chaque roue, les rouleaux soient orientés vers le centre du robot.

3 - Programme Arduino

Dans le cas où ces activités se font avec un programme Python exécuté sur un éventuel mini-ordinateur ajouté au robot (pcDuino, Raspberry Pi ou autre), il faut charger sur la carte Arduino Mega le firmware Arduino, composé des fichiers téléchargeables à cette adresse :

https://github.com/3sigma/T-Quad/tree/master/Arduino/Firmware_i2c

Ils sont également contenus dans l'archive suivante :

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

ATTENTION !

Ce firmware est déjà chargé à la livraison du robot : si vous n'avez encore jamais reprogrammé l'Arduino Mega et s'il n'existe pas de nouvelle version de ce firmware, ce n'est pas nécessaire de le charger à nouveau.

Dans le cas contraire, la procédure à suivre pour la programmation est la suivante:

- Si ce n'est pas déjà fait, installez l'IDE Arduino et les bibliothèques additionnelles nécessaires (voir la documentation générale : https://github.com/3sigma/T-Quad/raw/master/T-Quad_General.pdf)
- Ouvrez le programme Firmware_i2c.ino téléchargé précédemment dans l'IDE Arduino
- Mettre le robot sous tension en appuyant sur le bouton marche-arrêt
- Connecter le câble USB reliant l'ordinateur et la carte Arduino Mega
- Lancer le téléchargement

Noter que le comptage des interruptions permettant de mesurer la vitesse de rotation des moteurs est réalisé par ce programme, même lorsque le robot intègre un mini-ordinateur (pcDuino, Raspberry Pi ou autre). En effet, le système d'exploitation standard de ce type de mini-ordinateur ne permet pas de faire ces manipulations avec suffisamment de performances. Cette tâche est donc sous-traitée à la carte Arduino Mega.

4 - Activités

4.1 - Contrôle général des mouvements du robot

L'objectif est simplement de piloter les mouvements du robot en translation longitudinale et latérale et en rotation, en utilisant une application de contrôle à distance.

4.1.1 - Programme Python

Dans le cas où cette activité se fait avec un programme Python exécuté sur un éventuel mini-ordinateur ajouté au robot (pcDuino, Raspberry Pi ou autre), ce programme est déjà présent sur la carte. Cependant, il peut être nécessaire de mettre ce programme à jour avec la dernière version, disponible à l'adresse suivante :

https://raw.githubusercontent.com/3sigma/T-Quad-Quatre-Roues/master/programmes_python/QuatreRoues.py

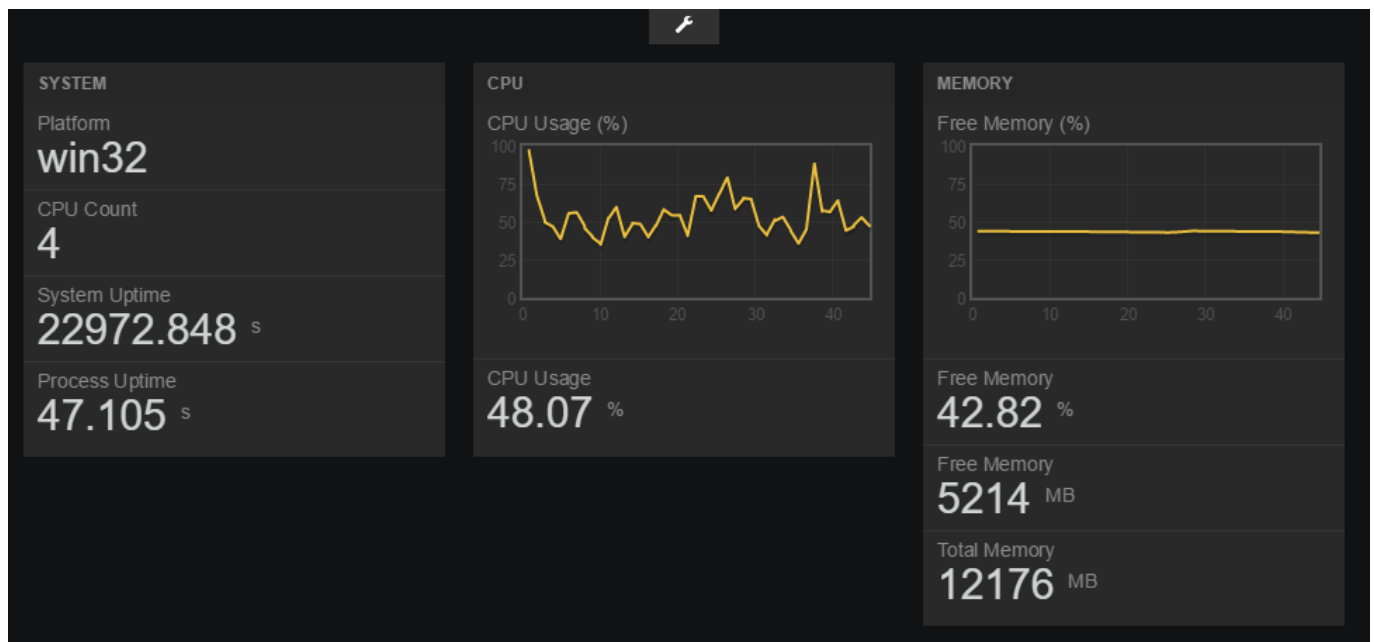
Il est également contenu dans l'archive suivante :

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

4.1.2 - Application MyViz

Les activités réalisées sur le robot T-Quad utilisent en général le logiciel MyViz, très souple pour créer des tableaux de bord de pilotage et de visualisation de données.

Après l'avoir téléchargé (<http://www.3sigma.fr/Telechargements-MyViz.html>) et installé, 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. Nous verrons plus loin comment charger celui correspondant à l'expérience que nous souhaitons réaliser dans ce chapitre.

Pour réaliser cette activité, les conditions suivantes doivent être remplies :

- le robot doit être allumé
- l'ordinateur doit être connecté en Wifi au robot
- le firmware Arduino doit être chargé (voir chapitre 3)

Charger ensuite le tableau de bord de pilotage 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-Quatre-Roues/master/MyViz/T-Quad_QuatreRoues_Reseau.json

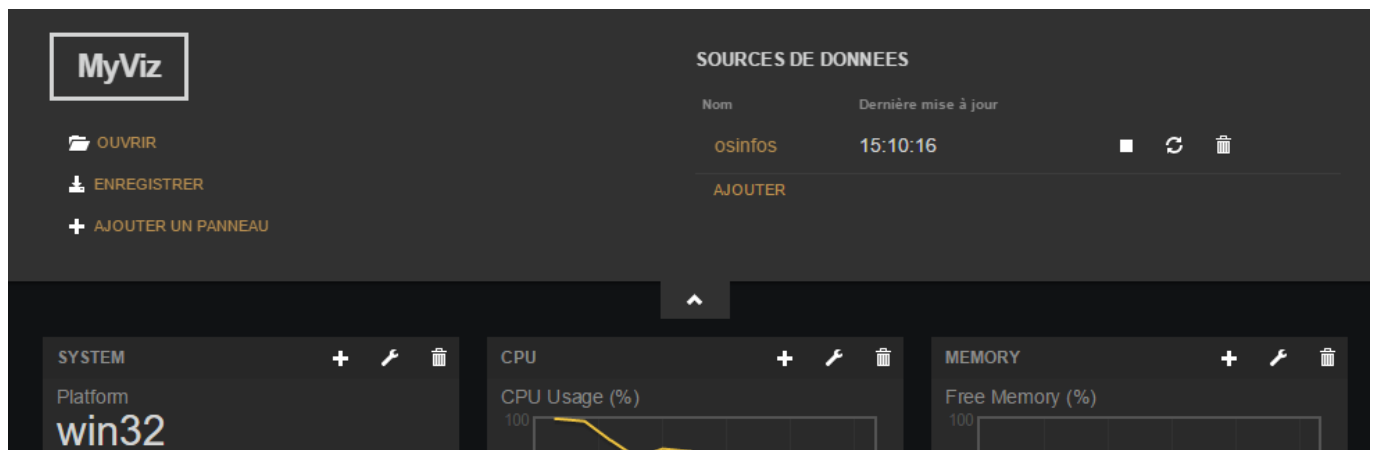
Il se trouve également dans l'archive suivante :

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

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

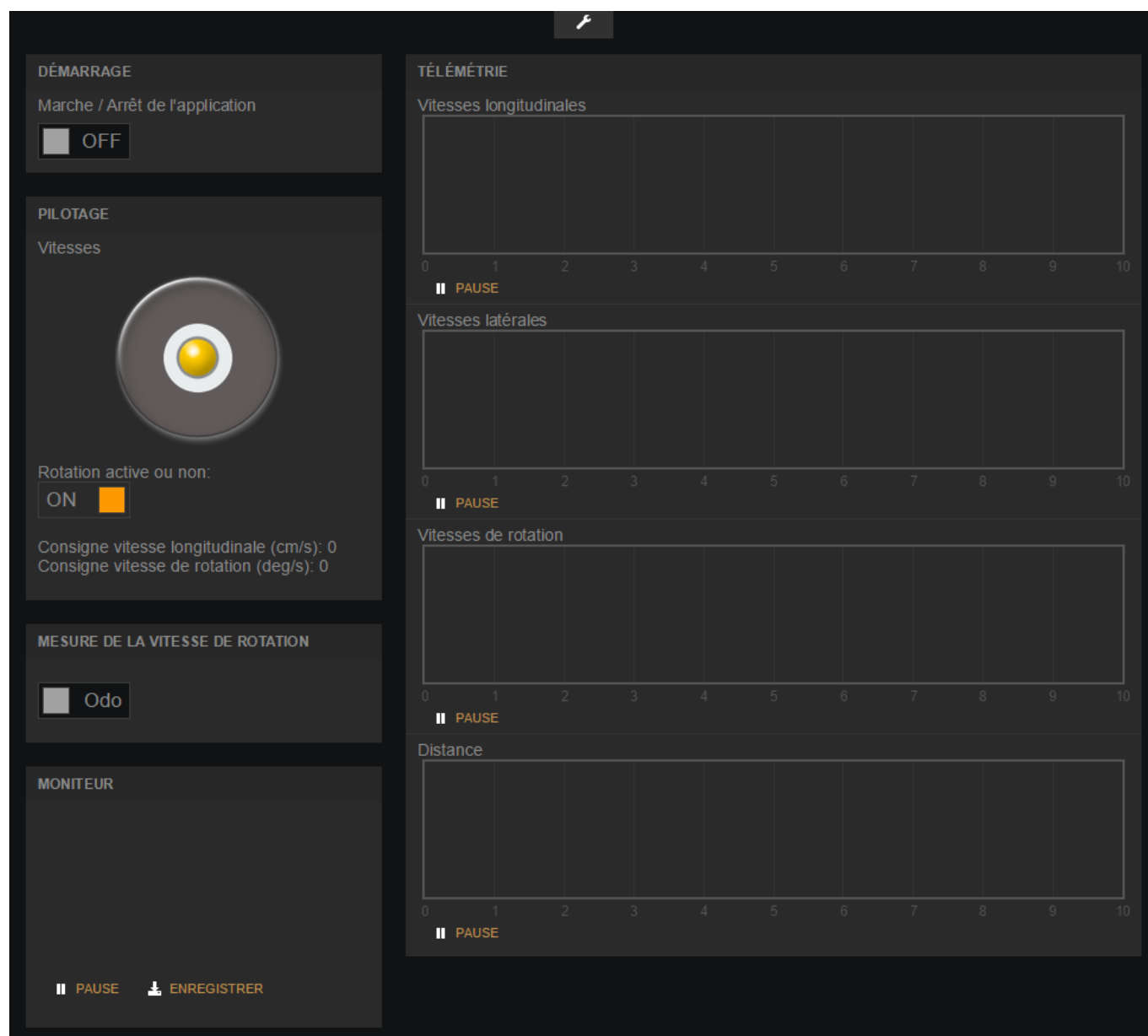


Ceci permet de déplier le panneau supérieur :



Cliquez sur « Ouvrir » et sélectionnez le fichier T-Quad_QuatreRoues_Reseau.json que vous venez de télécharger.

Le tableau de bord s'affiche alors :



Les différents éléments de l'interface sont les suivants (en commençant par la colonne de gauche et en finissant par celle de droite) :

- « Démarrage » : bouton ON/OFF de l'application de pilotage
Attention : lors du passage sur « ON », il faut attendre quelques secondes avant de voir apparaître les courbes de télémétrie et de pouvoir piloter le robot
- « Pilotage » : ce bloc est composé de deux éléments :
 - un « Joypad » pour contrôler deux degrés de liberté
 - un interrupteur d'activation ou non de la rotation du robot
 - lorsque la rotation est active, la translation latérale permise par les roues holonomes est inactive. Dans ce cas :
 - lorsque le centre de la boule jaune du joypad se trouve à l'intérieur du disque blanc, seule la rotation du robot autour de son axe vertical est possible. La vitesse de rotation dépend de la position de la boule
 - lorsque le centre de la boule jaune du joypad se trouve à l'intérieur de l'anneau gris, la consigne de mouvement donnée au robot est la combinaison d'une vitesse longitudinale et d'une vitesse de rotation, en fonction de la position de la boule
 - lorsque la rotation est inactive, la translation latérale permise par les roues holonomes est active. Dans ce cas :
 - lorsque le centre de la boule jaune du joypad se trouve à l'intérieur du disque blanc, aucun mouvement n'est possible
 - lorsque le centre de la boule jaune du joypad se trouve à l'intérieur de l'anneau gris, la consigne de mouvement donnée au robot est la combinaison d'une vitesse longitudinale et d'une vitesse latérale, en fonction de la position de la boule
- « Mesure de la vitesse de rotation » :
 - « Odo » : par défaut, la vitesse de rotation est calculée par odométrie, à partir de la vitesse de rotation des roues. Cependant, sur un robot à roues holonomes, une certaine quantité d'énergie est perdue dans les roues et cette vitesse de rotation calculée est toujours supérieure à la vitesse de rotation réelle
 - « Gyro » : quand cette option est activée, la vitesse de rotation du robot est mesurée grâce à la mesure gyroscopique autour de l'axe vertical. Cette mesure donne une meilleure image de la vitesse de rotation réelle du robot, mais à cause des bruits et de l'offset du capteur, la rotation mesurée n'est pas nulle même si le robot est immobile. Cela peut donc conduire à de légers mouvements de ce dernier même si la consigne de pilotage est nulle. Par ailleurs, si le robot est posé « sur cales » (et non pas sur ses roues) pour faire des tests, activer l'option « Gyro » n'a pas de sens puisque le robot ne peut pas tourner. Il faut alors obligatoirement choisir l'option « Odo » pour éviter une divergence des asservissements
- « Moniteur » : cette zone permet de visualiser sous forme textuelle la totalité des mesures renvoyées par le robot

- « Télémétrie » : la colonne de droite du tableau de bord est consacrée au tracé de courbes à partir de mesures effectuées sur le robot :
 - « Vitesses longitudinales » : consigne de vitesse longitudinale et mesure effectuée à partir de la rotation des roues
 - « Vitesses latérales » : consigne de vitesse latérale et mesure effectuée à partir de la rotation des roues
 - « Vitesses de rotation » : consigne de vitesse de rotation et mesure effectuée à partir de la rotation des roues et par le gyroscope
 - « Distance » : distance brute mesurée par le capteur ultrason et distance filtrée

En fonctionnement, ce tableau de bord peut avoir l'allure suivante :



4.2 - Asservissement d'angle

L'objectif est d'asservir l'angle de rotation du robot autour de son axe vertical, en fonction d'une consigne transmise à distance. Cette application permet de choisir entre une mesure de rotation réalisée par odométrie (à partir de la mesure de la vitesse de rotation des roues) ou à partir des données du gyroscope.

4.2.1 - Programme Python

Dans le cas où cette activité se fait avec un programme Python exécuté sur un éventuel mini-ordinateur ajouté au robot (pcDuino, Raspberry Pi ou autre), ce programme est déjà présent sur la carte. Cependant, il peut être nécessaire de mettre ce programme à jour avec la dernière version, disponible à l'adresse suivante :

https://raw.githubusercontent.com/3sigma/T-Quad-Quatre-Roues/master/programmes_python/QuatreRoues_AsservissementAngle.py

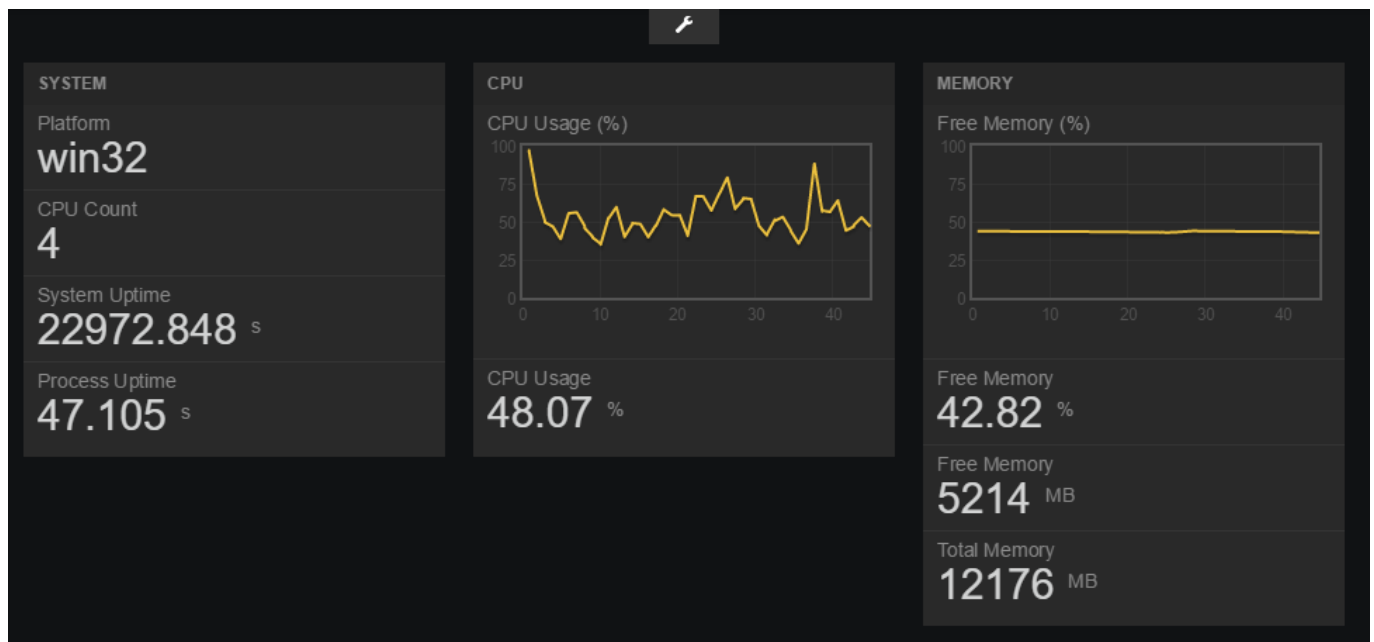
Il est également contenu dans l'archive suivante :

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

4.2.2 - Application MyViz

Les activités réalisées sur le robot T-Quad utilisent en général le logiciel MyViz, très souple pour créer des tableaux de bord de pilotage et de visualisation de données.

Après l'avoir téléchargé (<http://www.3sigma.fr/Telechargements-MyViz.html>) et installé, 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. Nous verrons plus loin comment charger celui correspondant à l'expérience que nous souhaitons réaliser dans ce chapitre.

Pour réaliser cette activité, les conditions suivantes doivent être remplies :

- le robot doit être allumé
- l'ordinateur doit être connecté en Wifi au robot
- le firmware Arduino doit être chargé (voir chapitre 3)

Charger ensuite le tableau de bord d'asservissement en angle 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-Quatre-Roues/master/MyViz/T-Quad_QuatreRoues_AsservissementAngle_Reseau.json

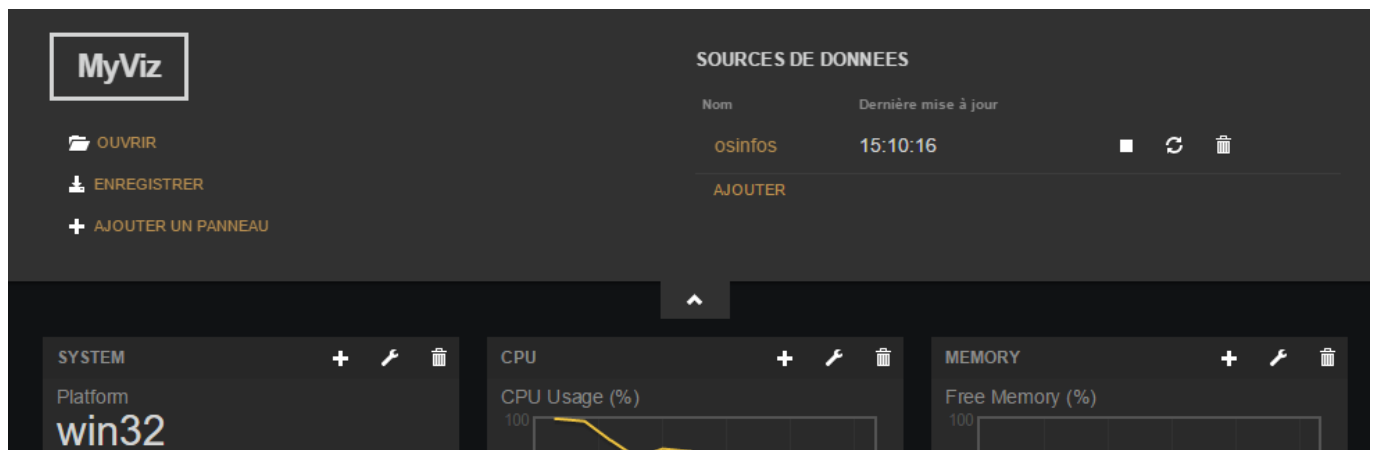
Il se trouve également dans l'archive suivante :

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

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

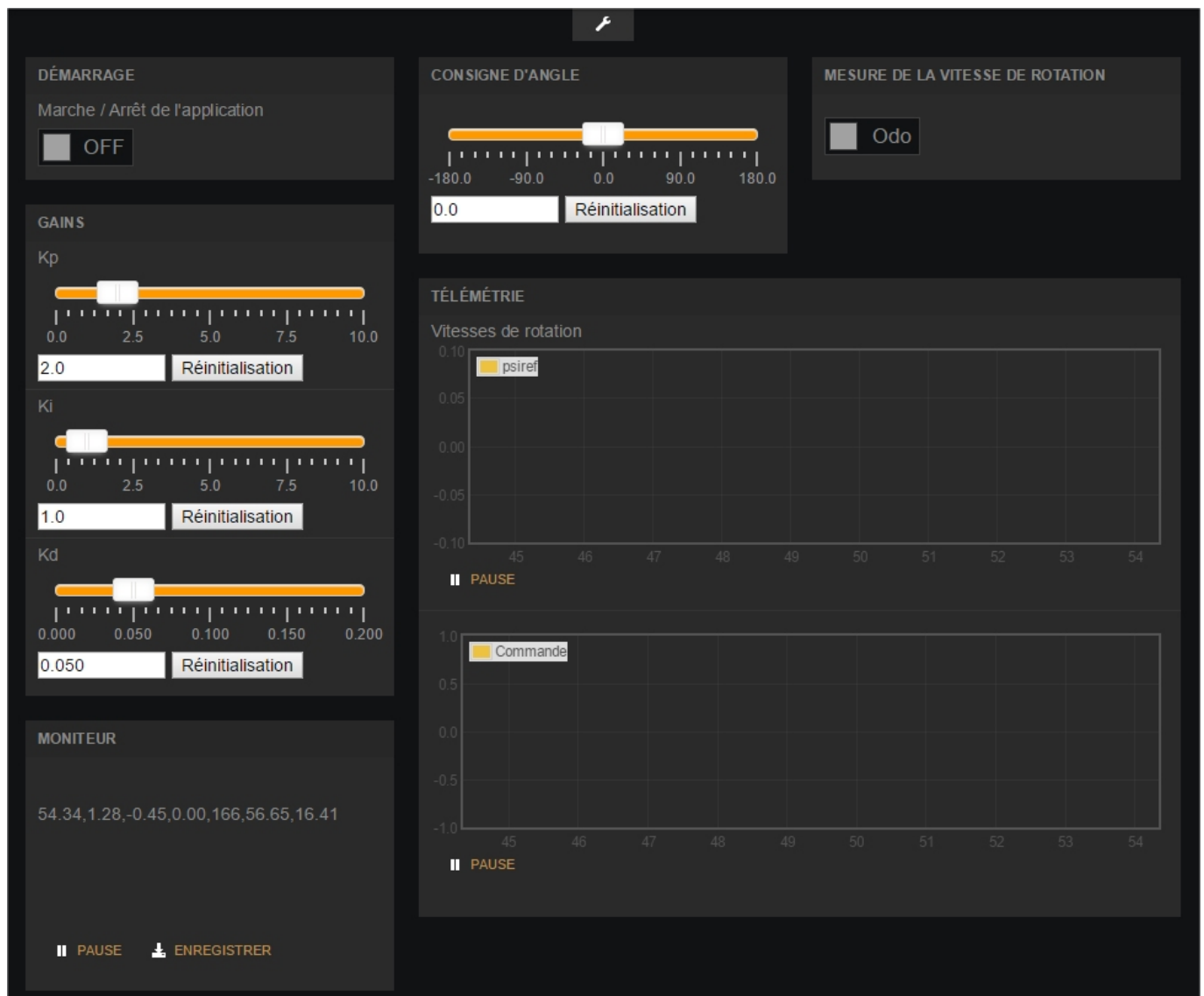


Ceci permet de déplier le panneau supérieur :



Cliquez sur « Ouvrir » et sélectionnez le fichier T-Quad_QuatreRoues_AsservissementAngle_Reseau.json que vous venez de télécharger.

Le tableau de bord s'affiche alors :

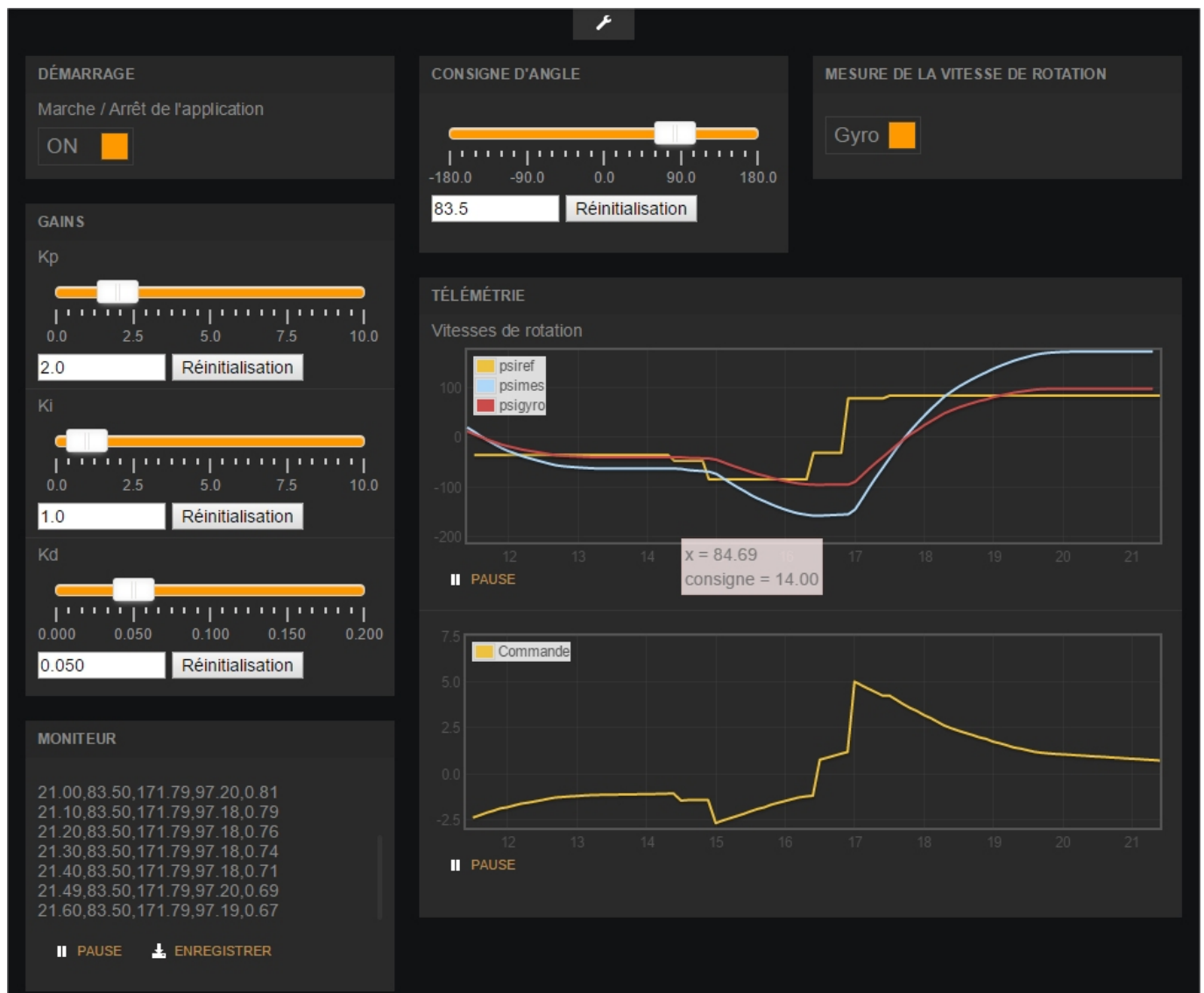


Son utilisation est a priori intuitive. Il faut cependant noter les points suivants :

- le démarrage de l'application se fait via la bouton marche-arrêt en haut à gauche
Attention : il faut attendre quelques secondes avant de voir apparaître les courbes de télémétrie et de pouvoir piloter le robot

- dans le bloc « Mesure de la vitesse de rotation », l'interrupteur a deux positions
 - « Odo » : par défaut, la vitesse de rotation est calculée par odométrie, à partir de la vitesse de rotation des roues
 - « Gyro » : quand cette option est activée, la vitesse de rotation du robot est mesurée grâce à la mesure gyroscopique autour de l'axe vertical.
- Attention** : à cause des bruits et de l'offset du capteur, la rotation mesurée n'est pas nulle même si le robot est immobile. Cela peut donc conduire à de légers mouvements de ce dernier même si la consigne de pilotage est nulle. Par ailleurs, si le robot est posé « sur cales » (et non pas sur ses roues) pour faire des tests, activer l'option « Gyro » n'a pas de sens puisque le robot ne peut pas tourner. Il faut alors obligatoirement choisir l'option « Odo » pour éviter une divergence des asservissements

En fonctionnement, ce tableau de bord peut avoir l'allure suivante :



4.3 - Asservissement de distance avec mouvement latéral

L'objectif est de faire un « suivi de mur » : le robot se déplace latéralement, tout en asservissant sa distance à un mur, mesurée par le capteur ultrason.

4.3.1 - Programme Python

Dans le cas où cette activité se fait avec un programme Python exécuté sur un éventuel mini-ordinateur ajouté au robot (pcDuino, Raspberry Pi ou autre), ce programme est déjà présent sur la carte. Cependant, il peut être nécessaire de mettre ce programme à jour avec la dernière version, disponible à l'adresse suivante :

https://raw.githubusercontent.com/3sigma/T-Quad-Quatre-Roues/master/programmes_python/QuatreRoues_LateralDistance.py

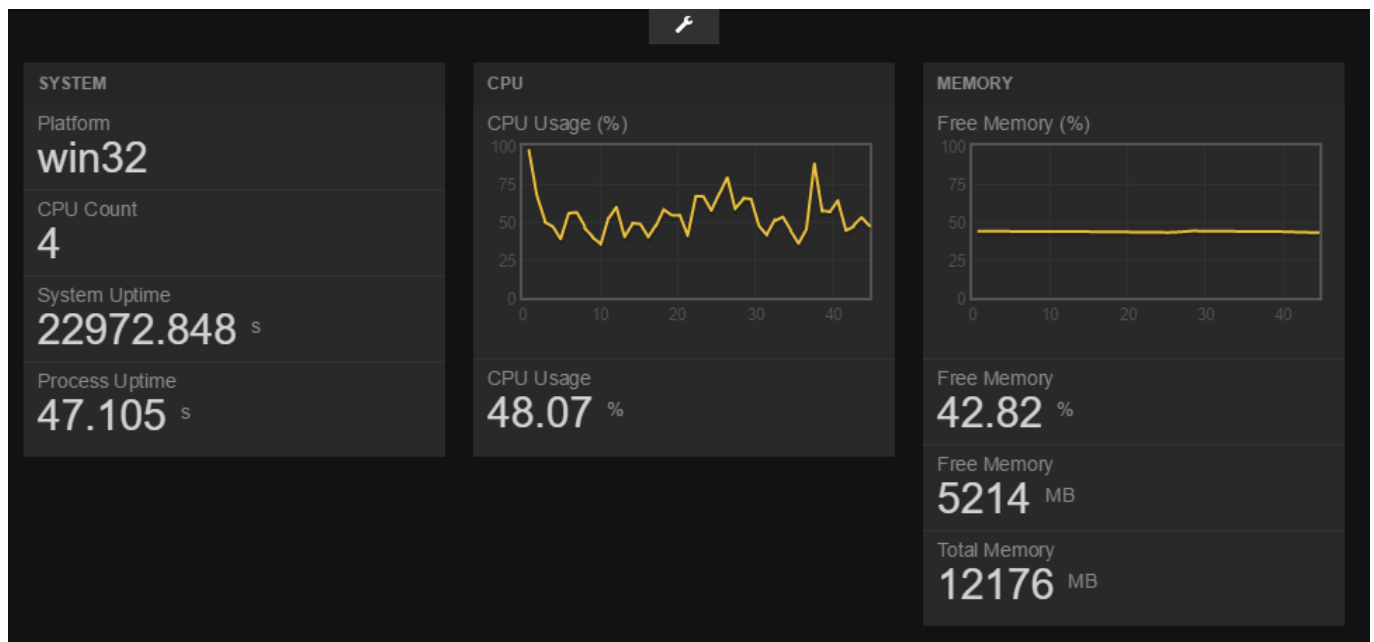
Il est également contenu dans l'archive suivante :

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

4.3.2 - Application MyViz

Les activités réalisées sur le robot T-Quad utilisent en général le logiciel MyViz, très souple pour créer des tableaux de bord de pilotage et de visualisation de données.

Après l'avoir téléchargé (<http://www.3sigma.fr/Telechargements-MyViz.html>) et installé, 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. Nous verrons plus loin comment charger celui correspondant à l'expérience que nous souhaitons réaliser dans ce chapitre.

Pour réaliser cette activité, les conditions suivantes doivent être remplies :

- le robot doit être allumé
- l'ordinateur doit être connecté en Wifi au robot
- le firmware Arduino doit être chargé (voir chapitre 3)

Charger ensuite le tableau de bord de suivi de mur 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-Quatre-Roues/master/MyViz/T-Quad_QuatreRoues_LateralDistance_Reseau.json

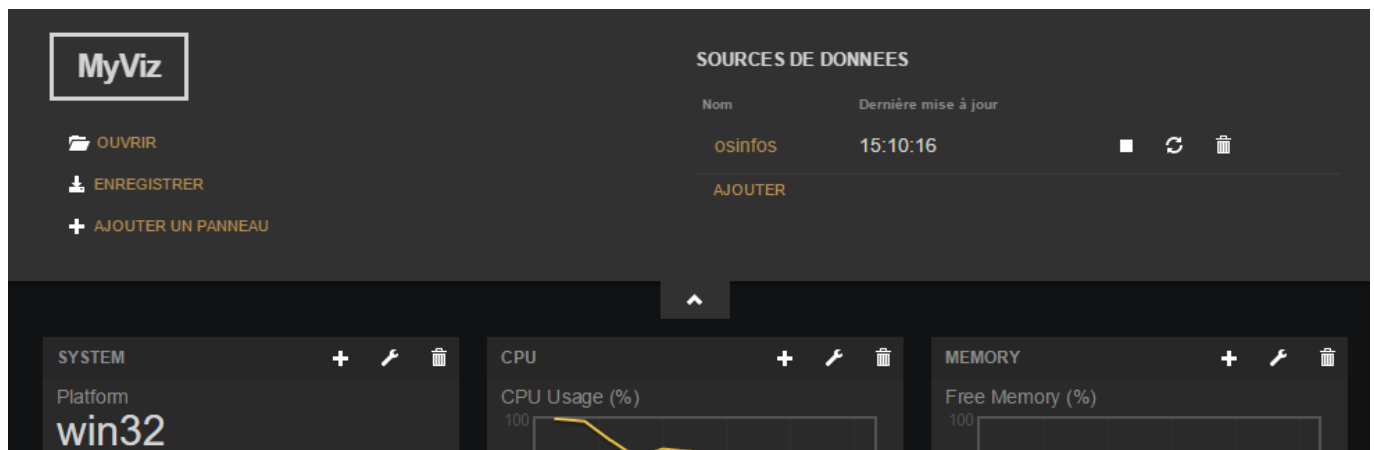
Il se trouve également dans l'archive suivante :

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

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

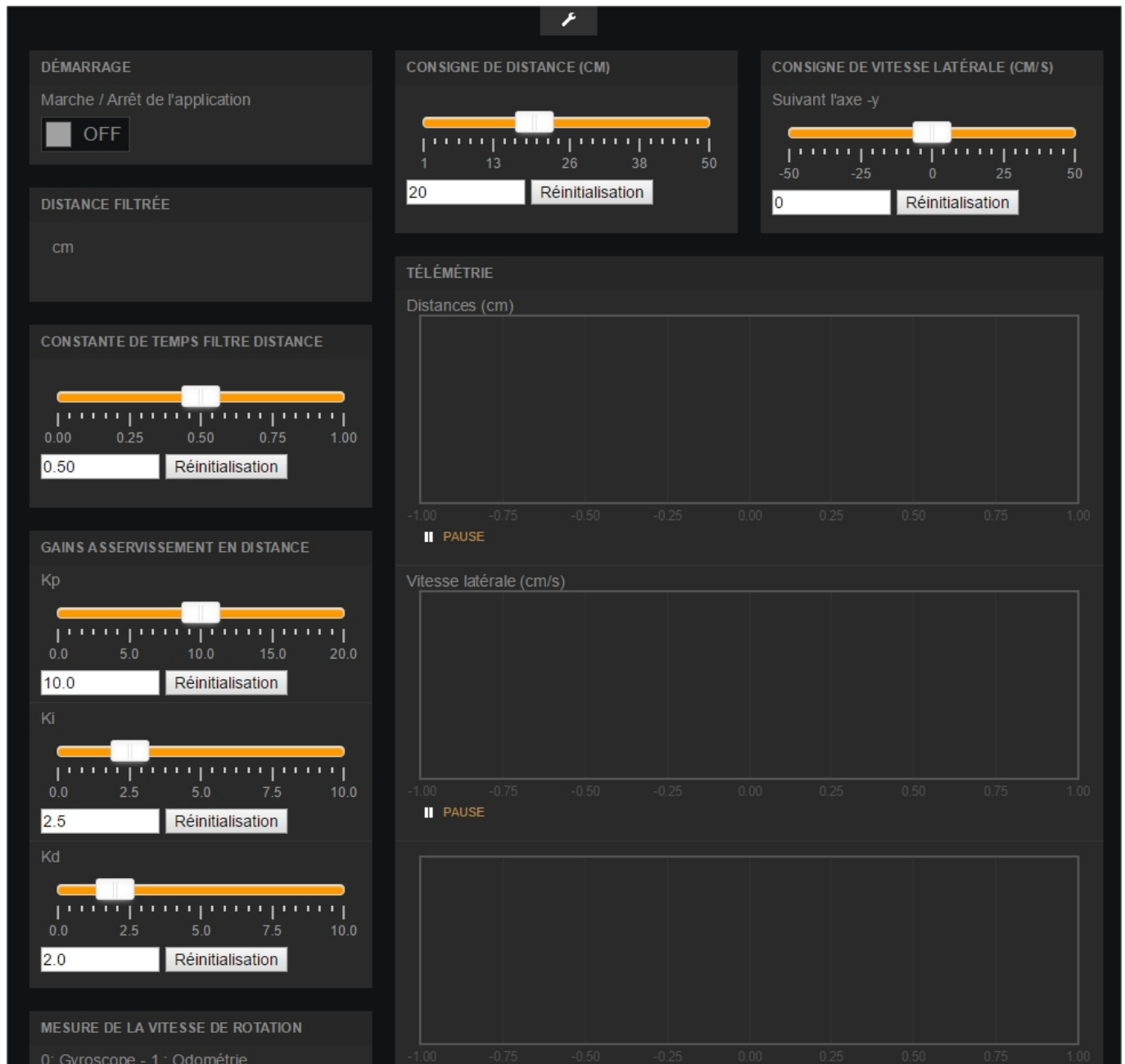


Ceci permet de déplier le panneau supérieur :



Cliquez sur « Ouvrir » et sélectionnez le fichier T-Quad_QuatreRoues_LateralDistance_Reseau.json que vous venez de télécharger.

Le tableau de bord s'affiche alors :



Son utilisation est a priori intuitive. Il faut cependant noter les points suivants :

- le démarrage de l'application se fait via la bouton marche-arrêt en haut à gauche
Attention : il faut attendre quelques secondes avant de voir apparaître les courbes de télémétrie et de pouvoir piloter le robot
- la mesure de la vitesse de rotation du robot peut se faire suivant une combinaison de la mesure faite par le gyroscope et de celle faite à partir de la vitesse de rotation des roues
- la consigne de vitesse latérale est donnée suivant l'axe -y, de sorte que lorsque le curseur se trouve sur la gauche (négatif, donc), le robot se déplace sur la gauche

En fonctionnement, ce tableau de bord peut avoir l'allure suivante :



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