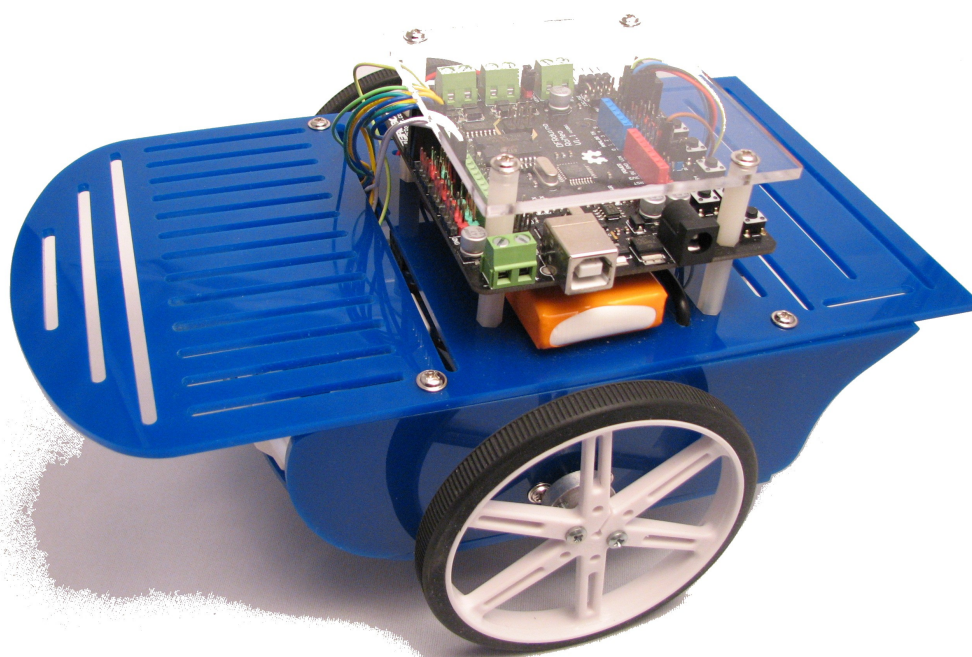




GEEROS "ERECTUS"

DOCUMENTATION COMPLÈTE



Date de dernière mise à jour : 17/02/2014

Table des matières

1 - Introduction.....	<u>4</u>
2 - Matériel inclus.....	<u>4</u>
2.1 - Eléments constituant Geeros « Erectus ».....	<u>5</u>
2.2 - Diagramme des échanges de signaux.....	<u>6</u>
3 - Conformité.....	<u>6</u>
4 - Installation de l'IDE Arduino.....	<u>7</u>
4.1 - Installation principale.....	<u>7</u>
4.2 - Installation de la bibliothèque complémentaire FlexiTimer2.....	<u>7</u>
4.3 - Installation de la bibliothèque complémentaire digitalWriteFast.....	<u>7</u>
4.4 - Installation de la bibliothèque complémentaire I2Cdev.....	<u>8</u>
4.5 - Installation de la bibliothèque complémentaire MPU6050.....	<u>8</u>
4.6 - Téléchargement d'un programme sur la carte Arduino Romeo.....	<u>9</u>
5 - Précautions d'emploi.....	<u>10</u>
5.1 - Connexions d'alimentation sur la carte Romeo.....	<u>10</u>
5.2 - Recharge de la batterie.....	<u>10</u>
5.3 - Précautions d'utilisation.....	<u>11</u>
6 - Utilisation.....	<u>12</u>
6.1 - Mise en route standard.....	<u>12</u>
6.2 - Mise en route spectaculaire.....	<u>12</u>
6.3 - Suivi de ligne.....	<u>13</u>
6.3.1 - Préparation.....	<u>13</u>
6.3.2 - Fonctionnement.....	<u>14</u>
6.3.3 - Asservissement.....	<u>14</u>
6.4 - Comment arrêter Geeros ?.....	<u>15</u>
7 - Fonctionnement d'un gyropode.....	<u>16</u>
7.1 - Maintien en équilibre.....	<u>16</u>
7.2 - Mouvement.....	<u>17</u>
7.3 - Démarrage.....	<u>17</u>
7.4 - Arrêt.....	<u>18</u>
8 - Expériences additionnelles.....	<u>19</u>
8.1 - Commande des moteurs en tension.....	<u>19</u>
8.2 - Asservissement des moteurs en vitesse.....	<u>22</u>

9 - Autre documentation disponible.....	<u>26</u>
10 - Important.....	<u>27</u>
11 - ANNEXE 1 – Circuit de suivi de ligne.....	<u>28</u>

1 - Introduction

Geeros « Erectus » est un robot gyropode ouvert et open-source, un concentré de technologie vous permettant de faire de nombreuses expériences. Il est basé sur une carte Romeo (compatible Arduino), différents capteurs et deux moteurs électriques permettant de le mettre en mouvement tout en assurant son maintien en équilibre.

Une documentation détaillée de la carte Arduino Romeo ainsi que les programmes associés au fonctionnement de Geeros, intégrant tous les commentaires nécessaires à leur compréhension, sont téléchargeables sur notre site Web à l'adresse suivante:

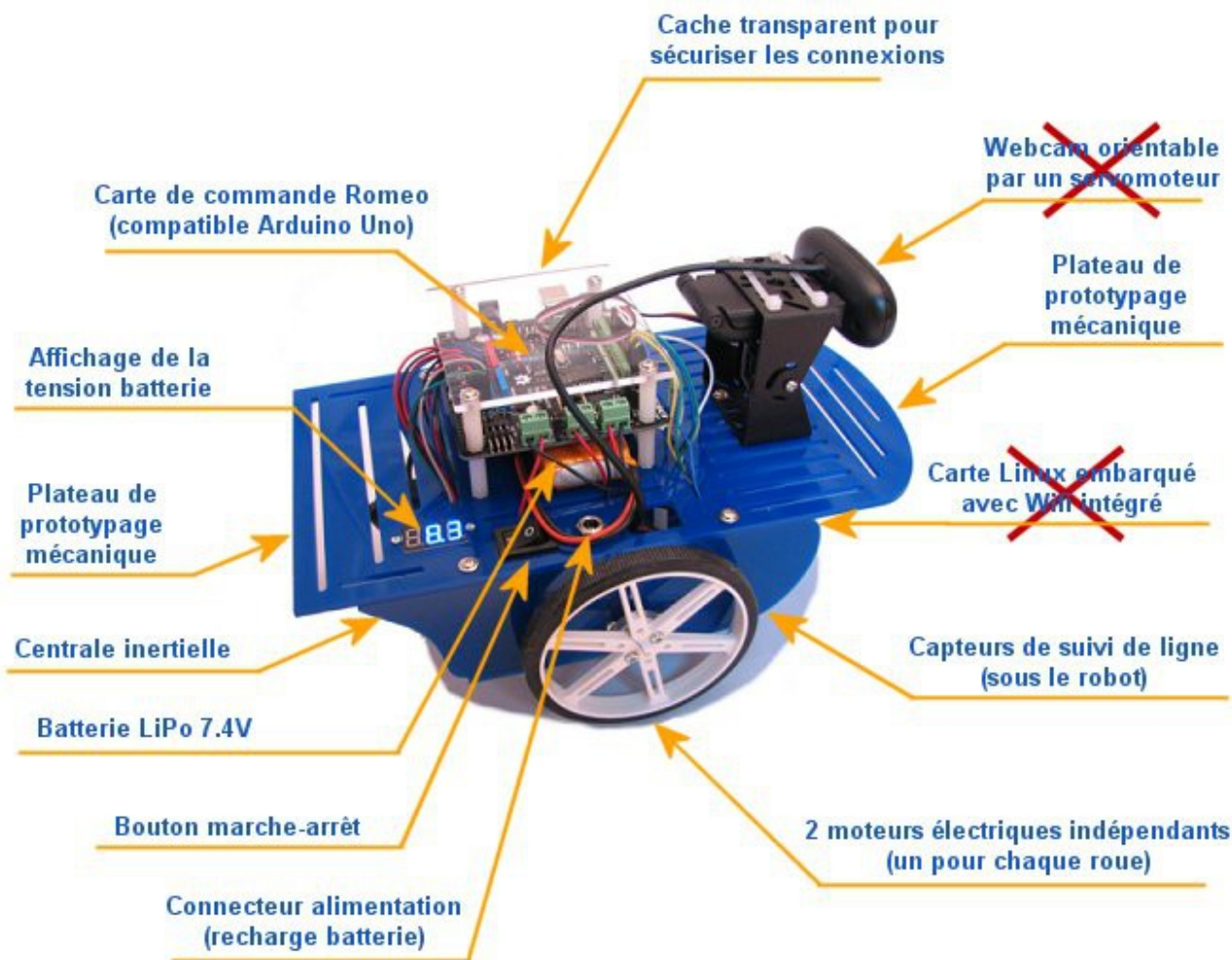
<http://www.3sigma.fr/telechargements>

2 - Matériel inclus

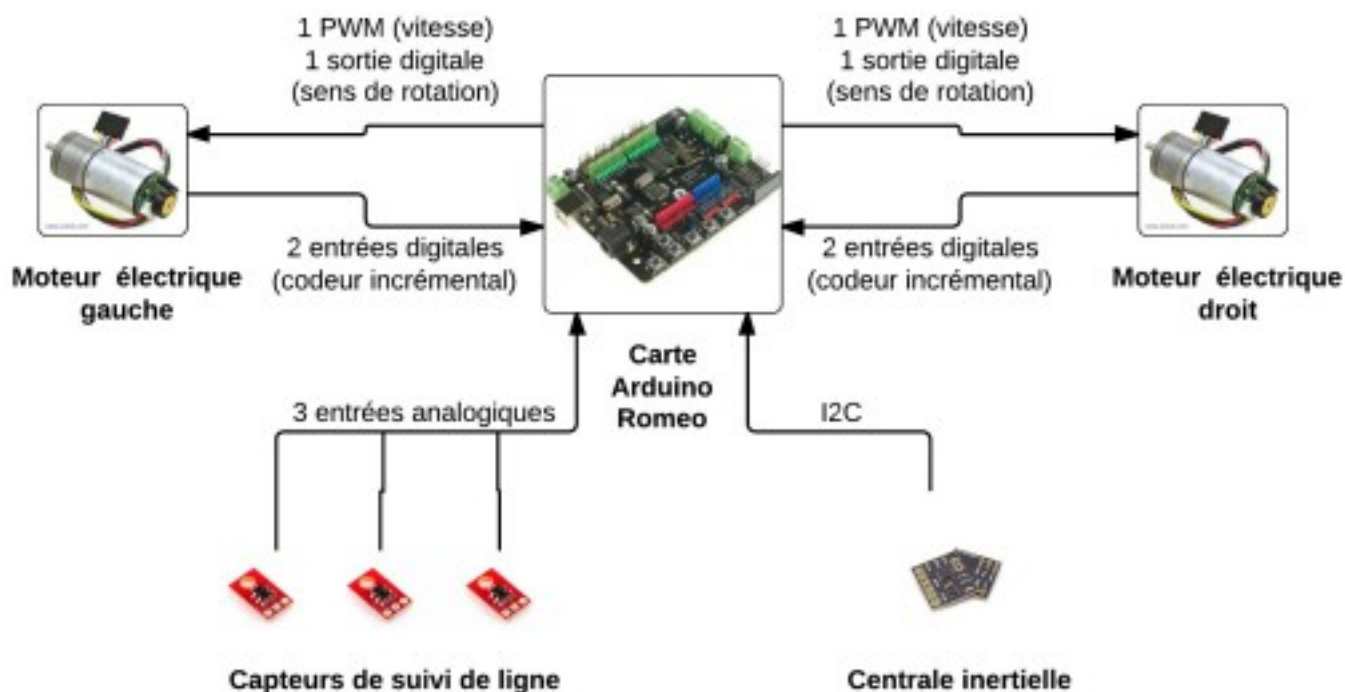
Geeros est livré monté et **fonctionnel (programmé et testé par nos soins avant la livraison)**. Il est composé des éléments suivants:

- Le robot gyropode lui-même
- 1 câble USB A-B pour la programmation de la carte Romeo
- 1 alimentation 12V, 1A
- 1 clé allen pour visser ou dévisser les roues
- 1 platine de prototypage électronique vierge
- 2 entretoises 30mm et 2 entretoises 50mm en métal, positionnées sous l'avant du robot pour lester ce dernier. Elles sont amovibles et peuvent être utilisées pour ajouter des éléments supplémentaires au robot.

2.1 - Éléments constituant Geeros « Erectus »



2.2 - Diagramme des échanges de signaux



3 - Conformité

Le robot Geeros, **dans sa configuration livrée aux clients**, est conforme à la directive 1999/EC.



4 - Installation de l'IDE Arduino

Geeros intègre une carte Romeo, compatible Arduino Uno. **Le robot est déjà programmé à la livraison.** Cependant, si vous souhaitez télécharger de nouveaux programmes, vous devez au préalable installer l'IDE Arduino et quelques bibliothèques additionnelles.

4.1 - Installation principale

Télécharger et installer l'IDE Arduino (<http://arduino.cc/en/Main/Software>). Il suffit de décompresser l'archive téléchargée dans le répertoire de votre choix.

ATTENTION !

Avant de pouvoir compiler les programmes fournis avec Geeros, vous devez suivre les instructions suivantes pour installer les bibliothèques complémentaires **FlexiTimer2**, **digitalWriteFast**, **I2Cdev**, **MPU6050** et **SoftwareServo**.

4.2 - Installation de la bibliothèque complémentaire FlexiTimer2

Cette bibliothèque permet d'exécuter à cadence fixe une partie du programme Arduino.

Vous pouvez la télécharger à l'adresse suivante: <http://www.3sigma.fr/telechargements/FlexiTimer2.zip>.

Une fois téléchargée, décompressez-la dans le répertoire des librairies de votre installation Arduino (typiquement, Documents\Arduino\libraries).

4.3 - Installation de la bibliothèque complémentaire digitalWriteFast

Cette bibliothèque permet de lire et d'écrire plus rapidement sur les entrées-sorties digitales de l'Arduino.

Vous pouvez la télécharger à l'adresse suivante:

<http://www.3sigma.fr/telechargements/digitalWriteFast.zip>.

Une fois téléchargée, décompressez-la dans le répertoire des librairies de votre installation Arduino (typiquement, Documents\Arduino\libraries).

4.4 - Installation de la bibliothèque complémentaire I2Cdev

Cette bibliothèque permet de communiquer avec des composants utilisant le protocole I2C, comme le capteur accélérométrique et gyroscopique 3 axes intégré à Geeros.

Vous pouvez la télécharger à l'adresse suivante:

<http://www.3sigma.fr/telechargements/I2Cdev.zip>.

Une fois téléchargée, décompressez-la dans le répertoire des bibliothèques de votre installation Arduino (typiquement, Documents\Arduino\libraries).

4.5 - Installation de la bibliothèque complémentaire MPU6050

Cette bibliothèque intègre de nombreuses fonctions permettant d'accéder aux informations mesurées par le MPU6050, qui est le capteur accélérométrique et gyroscopique 3 axes intégré à Geeros.

Vous pouvez la télécharger à l'adresse suivante:

<http://www.3sigma.fr/telechargements/MPU6050.zip>.

Une fois téléchargée, décompressez-la dans le répertoire des bibliothèques de votre installation Arduino (typiquement, Documents\Arduino\libraries).

ATTENTION !

L'environnement Arduino doit être redémarré après l'installation des bibliothèques complémentaires.

4.6 - Téléchargement d'un programme sur la carte Arduino Romeo

ATTENTION !

Pour télécharger un programme sur la carte Romeo, il est impératif de basculer au préalable le bouton marche-arrêt du robot sur la position « I ». Geeros doit par ailleurs être posé sur l'arrière.

La procédure à suivre pour la programmation est la suivante:

- Poser Geeros sur l'arrière pour qu'il ne se mette pas en route à la fin du téléchargement (en effet, le programme téléchargé démarre automatiquement à la fin de son téléchargement). Si vous modifiez les programmes livrés, vous devrez vous assurer de conserver un mode de fonctionnement dans lequel le contrôle de verticalité est désactivé pour ne pas voir le robot s'enfuir sitôt terminé le téléchargement de votre nouveau programme
- Connecter le câble USB reliant l'ordinateur et la carte Romeo
- Mettre le robot sous tension en appuyant sur le bouton marche-arrêt
- Lancer le téléchargement immédiatement

5 - Précautions d'emploi

Nous insistons sur le fait que Geeros et un robot d'étude et de développement qui nécessite un certain nombre de précautions d'emploi.

5.1 - Connexions d'alimentation sur la carte Romeo

Il est impératif de faire très attention aux connexions de l'alimentation de la carte Romeo (sous le cache en plexiglas transparent) car celle-ci n'est pas protégée contre les inversions de polarité. Une erreur de connexion sur les bornes d'alimentation risque d'entraîner la destruction du sous-ensemble de gestion d'alimentation de la carte et de rendre celle-ci inutilisable. Le robot étant livré connecté et fonctionnel, il est préférable de ne pas modifier les branchements sur les connecteurs d'alimentation.

5.2 - Recharge de la batterie

L'alimentation 12V, 1A fournie est destinée à recharger la batterie LiPo du robot en la connectant sur le connecteur jack situé à côté du bouton de marche-arrêt. Bien que cette batterie possède une protection intégrée contre les surcharges et les sous-charges, il n'est pas conseillé de la recharger avec une alimentation pouvant fournir un courant supérieur à 1A.

Si une autre alimentation devait être utilisée pour la recharger, il est important de vérifier que sa polarité est « positif au centre du connecteur ».

Il faut compter environ 2 heures pour une demi-recharge de la batterie et 8 heures pour une recharge complète.

Les robots Geeros 2 (livrés après le 1^{er} juillet 2013) possèdent par ailleurs un afficheur LCD permettant de visualiser la tension en sortie de la batterie. Lorsqu'une batterie est totalement chargée avec l'alimentation 12V, sa tension de sortie est d'environ 8.4 V. Lorsque la tension descend au dessous de 7V, il est bon d'envisager une recharge de la batterie.

5.3 - Précautions d'utilisation

Geeros doit être utilisé dans les conditions suivantes:

- Au sol. Ne jamais l'utiliser en hauteur (sur une table, par exemple) à cause des risques de chute
- A l'intérieur. Les sols extérieurs présentent souvent des aspérités et des obstacles surdimensionnés par rapport à la taille du robot. Par ailleurs, Geeros craint l'eau et l'humidité

Geeros peut être reprogrammé à votre guise. Vous pouvez par exemple faire des programmes permettant d'étudier la commande des moteurs électriques. Mais attention: il est **fortement déconseillé** de faire des expériences de fonctionnement « rotor bloqué » avec une tension d'alimentation du moteur trop élevée. Ce type d'expérience peut générer des courants trop forts qui réduisent la durée des vie des éléments.

6 - Utilisation

6.1 - Mise en route standard

Le programme pré-chargé dans le robot à la livraison lui permet de tenir en équilibre sur ses deux roues. La procédure à suivre est simple:

- Poser le robot au sol sans l'allumer, penché vers l'arrière
- Le mettre en route en appuyant sur le bouton marche-arrêt
- Le redresser. Lorsqu'il atteint la verticale, l'asservissement d'équilibre démarre automatiquement

Remarque :

Si le robot est parfaitement équilibré quand il est éteint (ce qui est normalement le cas lorsque vous le recevez), l'activation de l'asservissement de verticalité s'accompagne d'un mouvement de translation quasi-nul. En revanche, si le robot est naturellement déséquilibré (ce qui peut arriver si vous rajoutez de nouveaux capteurs, par exemple), l'activation de l'asservissement de verticalité s'accompagnera d'un mouvement de translation, le temps que le robot détermine automatiquement l'angle d'inclinaison lui permettant de se maintenir en équilibre.

6.2 - Mise en route spectaculaire

Il est possible de démarrer le robot lorsqu'il est posé sur l'avant à l'arrêt. Dans ce cas, le basculement de l'interrupteur sur la position « I » démarre immédiatement le robot, qui retrouve tout seul son équilibre.

Ce mode de démarrage est très spectaculaire mais attention ! Il faut prévoir quelques dizaines de centimètres sans obstacles pour qu'il ait la place d'avancer pour se redresser.

6.3 - Suivi de ligne

6.3.1 - Préparation

Une étape préalable est la réalisation d'un circuit. Vous pouvez pour cela imprimer les morceaux de lignes fournis en annexe (chapitre 11) pour construire un circuit personnalisé.

Les autres conditions préalables au bon fonctionnement sont les suivantes:

- Les capteurs de suivi de lignes doivent être présents (ils sont facilement démontables...)
- Vous devez avoir téléchargé sur la carte Arduino Romeo l'application GeerosErectusSuiviLigne.ino, téléchargeable à l'adresse <http://www.3sigma.fr/telechargements>
- Vous devez éventuellement avoir réglé le seuil de détection dans ce programme. Celui-ci est réglé pour fonctionner dans la majorité des cas mais si vous constatez que le robot a du mal à détecter la ligne, vous pouvez diminuer la valeur de la variable globale « SEUIL », en procédant par pas de 50. A l'inverse, si vous constatez que le robot fait des fausses détections, vous devrez augmenter ce seuil

Concernant ce dernier point, il est important de conserver en mémoire les éléments susceptibles de jouer sur la détection de la ligne:

- Le contraste entre la ligne et le fond. Les meilleurs résultats sont obtenus en réalisant un circuit à partir de morceaux de lignes noires imprimées sur du papier blanc (voir chapitre 11). Si vous avez un sol très clair ou très foncé, vous pouvez également réaliser un circuit directement parterre en utilisant du ruban adhésif pour tracer les lignes. Cependant, vous aurez dans ce cas du mal à tracer des courbes. Or, un angle sur une ligne correspond localement à un rayon de courbure nul, que le robot ne pourra pas suivre instantanément (d'autant plus que Geeros ne peut pas s'arrêter avant de tourner, l'arrêt immédiat d'un gyropode n'étant pas possible, voir 7.4).
- La distance entre les capteurs et le sol: il faut trouver un bon compromis car plus la distance est faible, plus la ligne sera détectée correctement, mais plus les capteurs risqueront de rencontrer des obstacles. Par exemple, si une feuille du circuit n'est pas très bien plaquée au sol, les capteurs du robot risquent de s'y accrocher

6.3.2 - Fonctionnement

La procédure à suivre pour démarrer le suivi de ligne sur Geeros est la suivante (rappelons que le programme Arduino adéquat doit avoir été chargé au préalable sur le robot, voir ci-dessus):

- Positionner le robot à droite de la ligne à suivre (quand on regarde le robot et la trajectoire par l'arrière), en direction de celle-ci, avec un angle d'environ 30° en direction de celle-ci
- Relever le robot pour activer l'asservissement de verticalité
- Après 1 s, Geeros démarre. Lorsqu'il croise la ligne, il doit normalement la détecter. Dans ce cas, la led verte de la carte Romeo s'allume. Si la ligne n'est pas détectée, cela signifie que le contraste et/ou l'éclairage de la pièce n'est pas suffisant
- Une fois la ligne détectée, Geeros tente de la suivre

6.3.3 - Asservissement

L'asservissement de suivi de ligne est du type proportionnel-dérivée, qui est un régulateur assez peu courant. La variable asservie est la distance entre le robot et la ligne ; la commande du système est une tension différentielle entre les deux moteurs, ce qui correspond directement à une commande de vitesse de rotation. C'est ce dernier point qui conduit à l'utilisation d'un régulateur PD.

L'action proportionnelle joue en fait le rôle de l'action intégrale dans un asservissement plus classique: si le robot roule parallèlement mais non centré sur la ligne, cette action va le ramener au centre.

L'action dérivée joue quant à elle le rôle de l'action proportionnelle d'un asservissement classique: lorsque le robot s'éloigne de la ligne, cette action le ramène sur cette dernière.

Il est important de conserver à l'esprit que la mesure de la distance entre le robot et la ligne est réalisée grâce à trois capteurs optiques analogiques mais qui fournissent dans les faits une information de type tout ou rien car en général les bords des lignes sont nets: le capteur est au-dessus de la ligne ou à côté, il n'y a pas vraiment d'intermédiaire. L'écart entre le robot et la ligne est donc une quantité discontinue qui, passée par l'action dérivée (même filtrée), conduit à des à-coups de rotation du robot assez brutaux.

Remarque :

Avec les réglages par défaut, Geeros peut quitter la ligne (dans ce cas, la LED verte de la carte Romeo s'éteint par plus aucun capteur n'est au dessus de la ligne). Ce n'est pas un problème, l'action proportionnelle est là pour ramener le robot dans le droit chemin.

6.4 - Comment arrêter Geeros ?

Deux options sont possibles:

- Vous souhaitez l'arrêter totalement: il suffit de basculer le bouton de marche-arrêt sur « 0 »
- Vous souhaitez désactiver l'asservissement de verticalité mais pas arrêter totalement le robot : il suffit de le basculer vers l'arrière jusqu'à l'arrêt de la rotation des roues. Vous pouvez ensuite le poser, toujours sur l'arrière, il ne démarrera pas tant que vous ne le redresserez pas à la verticale

7 - Fonctionnement d'un gyropode

Ce chapitre décrit le principe de fonctionnement d'un gyropode, qu'il est utile de connaître pour comprendre son comportement.

7.1 - Maintien en équilibre

Deux capteurs sont essentiels pour le maintien en équilibre: un accéléromètre et un gyroscope. Geeros est équipé d'un capteur qui communique avec la carte Arduino Romeo via une interface I2C. Ce capteur combine un accéléromètre 3 axes et un gyroscope 3 axes. Seul un axe accélérométrique et un axe gyroscopique sont utilisés dans l'asservissement de verticalité.

Un accéléromètre 3 axes permet de mesurer l'accélération du gyropode **plus** l'accélération de la pesanteur dans 3 axes orthogonaux liés au robot. L'algorithme de maintien en équilibre utilise la mesure de l'accélération longitudinale: en effet, quand le gyropode est incliné sans mouvement, l'accéléromètre mesure la pesanteur multipliée par l'angle d'inclinaison:

$$\text{acc} = g \cdot \sin(\theta)$$

La valeur de la pesanteur (g) étant connue (9.81 m/s^2) et l'angle d'inclinaison étant en général faible ($\sin(\theta) \sim \theta$), cela nous permet d'obtenir:

$$\theta = \text{acc}/g$$

Mais la connaissance de l'angle d'inclinaison n'est pas suffisante. En effet, deux cas de figure peuvent se présenter:

- le gyropode est incliné et est en train de tomber (l'inclinaison augmente)
- le gyropode est incliné mais est en train de se redresser (l'inclinaison diminue)

Dans le premier cas, il faudra agir plus fortement que dans le second cas. La distinction entre les deux cas se fait grâce à la mesure de la vitesse de chute (ou de redressement) fournie par un des axes (celui qui est parallèle à l'axe des roues) de mesure du gyroscope.

Avec ces deux mesures (1 axe accélérométrique et 1 axe gyroscopique), nous possédons en théorie assez d'informations pour maintenir correctement le gyropode à la verticale. Cependant, nous avons fait précédemment l'hypothèse d'un gyropode incliné sans mouvement pour la mesure de l'accélération. Or, dans le cas général le gyropode est en mouvement et peut avoir une accélération longitudinale qui va se superposer à l'accélération de la pesanteur:

$$\text{acc_mesurée} = g \cdot \sin(\theta) + \text{acc_longitudinale}$$

Cette accélération longitudinale étant inconnue, on ne peut pas en théorie remonter à l'angle d'inclinaison. Pour cette raison, l'asservissement de verticalité de Geeros intègre un filtre permettant d'estimer l'angle d'inclinaison à partir des deux mesures de l'accéléromètre et du gyroscope.

Remarque :

Geeros est capable de se maintenir en équilibre même s'il n'est pas parfaitement équilibré en « statique » (lorsqu'il est éteint). Dans ce cas, il penchera vers l'avant (même à l'arrêt) s'il y a trop de poids sur l'arrière et vice-versa.

7.2 - Mouvement

Avant tout, il est important de bien comprendre comment fonctionne un gyropode: lorsqu'il est en mouvement, il est penché en avant dans le sens de la marche pour maintenir son équilibre. En effet, la force de motorisation ne s'applique pas sur le centre de gravité du robot mais au point de contact avec le sol. La force de motorisation engendre donc un couple de rotation du gyropode autour de son centre d'inertie et a tendance à le faire tomber. Pour contrer ce couple de chute, le gyropode asservi se penche en avant pour utiliser la pesanteur.

Le gyropode peut donc rester à l'équilibre (mais pas à la verticale) en mouvement lorsque le couple de rotation généré par les moteurs est compensé par le couple de rotation généré par la pesanteur. Il se déplace alors penché dans le sens de la marche.

7.3 - Démarrage

Comme indiqué précédemment, Geeros est penché vers l'avant lorsqu'il se déplace. Or, il est vertical à l'arrêt. La phase de démarrage doit permettre de passer d'un état à l'autre (tout se fait automatiquement). Cette phase permettant de passer de l'état « vertical à l'arrêt » à l'état « penché vers l'avant en mouvement » est la suivante:

- Les moteurs vont tout d'abord tourner dans le sens inverse de la marche
- Cela a pour effet de faire pencher le gyropode vers l'avant
- Les moteurs tournent ensuite dans le sens de la marche, leur vitesse de rotation et l'angle d'inclinaison se compensant mutuellement pour éviter la chute

Nous nous trouvons donc en présence du paradoxe suivant: pour avancer, un gyropode doit d'abord (légèrement) reculer.

IMPORTANT !

Si vous démarrez Geeros collé à un mur à l'arrière, il ne pourra par conséquent pas démarrer puisqu'il ne pourra pas initialement reculer !

7.4 - Arrêt

Nous avons vu précédemment que lorsque Geeros est en mouvement, il est penché en avant dans le sens de la marche pour maintenir son équilibre. Quand il doit s'arrêter et retrouver la verticale, il doit transitoirement se pencher vers l'arrière, dans le sens contraire au mouvement, pour freiner (sinon, il chute vers l'avant). La seule façon de réaliser ça est de faire en sorte que la base du gyropode aille transitoirement plus vite que son centre de gravité. Par conséquent, pour s'arrêter, le gyropode doit tout d'abord accélérer !

S'il roule déjà à la vitesse maximale permise par les moteurs, il ne pourra plus accélérer et sera donc incapable de s'arrêter. La vitesse du robot est limitée à 50 cm/s dans les différentes applications de pilotage, ce qui permet de conserver aux moteurs une réserve d'accélération suffisante pour que ceux-ci puisse réaliser un arrêt du robot.

Une dernière conséquence est que l'arrêt immédiat d'un robot gyropode n'est pas possible. Si on coupe brutalement l'alimentation des moteurs, le robot va chuter, entraîné par sa vitesse. Ce point est important pour les applications de suivi de ligne car il empêche le robot de suivre des virages trop serrés.

8 - Expériences additionnelles

Geeros est un système pédagogique multi-expériences qui permet non seulement d'étudier de près le fonctionnement d'un gyropode, mais également d'autres aspects plus spécifiques, comme la commande de moteurs électriques.

ATTENTION !

Veillez à sécuriser le robot lorsque vous réalisez les expériences décrites par la suite, pour qu'il ne risque pas de tomber d'une table, par exemple.

8.1 - Commande des moteurs en tension

Cette expérience permet de changer la vitesse de rotation des moteurs en appliquant une tension variable par l'intermédiaire d'une application qui s'exécute sur votre ordinateur.

Le programme Arduino peut être téléchargé à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosArduinoCommandeMoteursEnTension_x.y.zip (x.y correspond au numéro de version du programme).

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à envoyer des signaux PWM sur le pont en H intégré à la carte Romeo afin de faire varier la tension d'alimentation des moteurs. Cette tension peut être modifiée interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosInterfaceCommandeMoteursEnTension_x.y.zip (x.y correspond au numéro de version du programme).

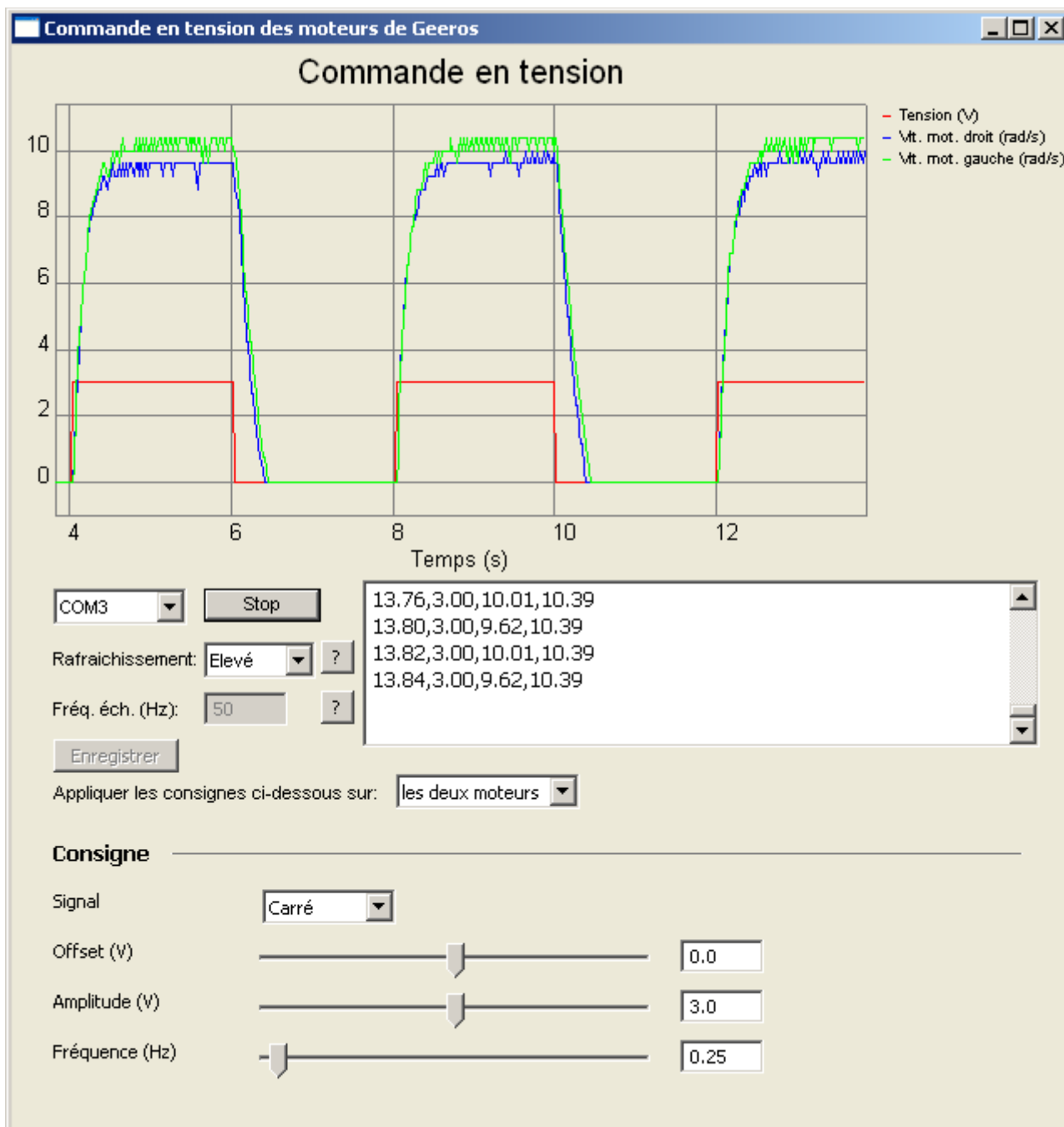
Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur GeerosInterfaceCommandeMoteursEnTension_x.y.exe.

Pour piloter la tension d'alimentation (et donc la vitesse) des moteurs depuis votre ordinateur, il vous suffit de suivre les étapes suivantes :

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- connecter la carte Romeo à l'ordinateur avec le câble USB fourni

- télécharger le programme Arduino « GeerosArduinoCommandeMoteursEnTension » sur la carte Romeo
- lancer l'application « GeerosInterfaceCommandeMoteursEnTension »

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
 - la tension de commande (V, en rouge)
 - les vitesses mesurées (rad/s, en bleu et vert)
- zone de sélection du port série: choisir le port série sur lequel est connectée votre carte Romeo et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme Arduino. Attention: ce paramètre doit être spécifié en Hz, alors que les valeurs correspondantes dans le programme Arduino (CADENCE_MS et TSDATA) sont spécifiées en ms. La relation entre les deux est la suivante:
$$\text{freq (Hz)} = 1000 / \text{cadence (ms)}$$
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.
Attention : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- Zone « Consigne »: elle permet de modifier la consigne de tension avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel

IMPORTANT !

Si l'affichage ne suit pas les consignes même avec une fréquence de rafraîchissement très lente, veuillez diminuer la cadence d'échantillonnage dans le programme Arduino (augmenter les valeurs TSDATA et CADENCE_MS)

IMPORTANT !

Si vous comparez la vitesse de rotation des deux moteurs, vous obtiendrez probablement une valeur différente, bien que la tension de commande soit la même : c'est normal, ceci est dû à la disparité des caractéristiques des moteurs à courant continu.

D'un point de vue pédagogique, ce point permet de souligner la nécessité de réaliser un asservissement de vitesse : si l'on souhaite une vitesse de rotation précise, on ne peut pas se contenter d'une commande en boucle ouverte.

8.2 - Asservissement des moteurs en vitesse

Cette expérience permet d'asservir la vitesse de rotation des moteurs en appliquant une consigne de vitesse par l'intermédiaire d'une application qui s'exécute sur votre ordinateur.

Le programme Arduino peut être téléchargé à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosArduinoAsservissementMoteursEnVitesse_x.y.zip (x.y correspond au numéro de version du programme).

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à calculer, grâce à un régulateur de type PID, la tension de commande à appliquer aux moteurs (via commande PWM du pont en H intégré sur la carte Romeo) pour qu'il suive la consigne de vitesse spécifiée.

Notez que dans ce programme, la vitesse mesurée est en fait la moyenne glissante des 10 derniers échantillons de mesure de vitesse instantanée, ce qui permet d'avoir une mesure plus lisse. En effet, la résolution de la mesure instantanée est la suivante:

$$2*\pi/(Ts*CPR*ratio)$$

avec :

- Ts : cadence d'échantillonnage (0.01 s)
- CPR : nombre d'impulsions par tour du codeur (48)
- ratio : rapport de réduction du moteur (34)

Ceci donne une résolution de $2*\pi/(0.01*1632) = 0.4$ rad/s. Le moyennage permet d'améliorer cette valeur.

La consigne de vitesse peut se fixer interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

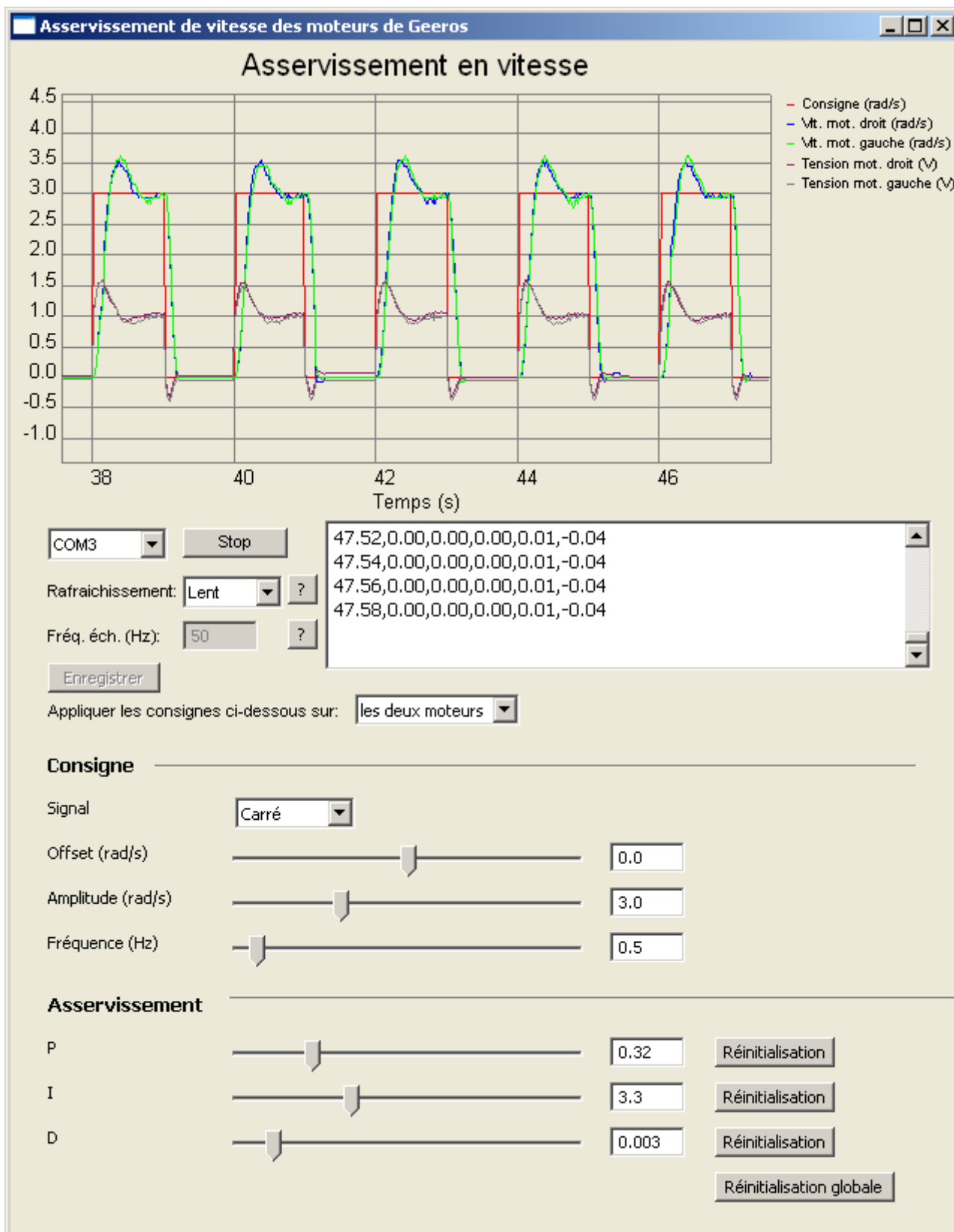
Son nom est de la forme GeerosInterfaceAsservissementMoteursEnVitesse_x.y.zip (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur GeerosInterfaceAsservissementMoteursEnVitesse_x.y.exe.

Pour changer la consigne de vitesse du moteur depuis votre ordinateur, il vous suffit de suivre les étapes suivantes (remarque: n'exécutez pas de nouveau celles que vous avez déjà effectuées):

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- connecter la carte Romeo à l'ordinateur avec le câble USB fourni
- télécharger le programme Arduino « GeerosArduinoAsservissementMoteursEnVitesse » sur la carte Romeo
- lancer l'application « GeerosInterfaceAsservissementMoteursEnVitesse »

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
 - la consigne de vitesse (rad/s, en rouge)
 - les vitesses mesurées (rad/s, en bleu et en vert)
 - les tensions de commande (V, en marron et en gris)
- zone de sélection du port série: choisir le port série sur lequel est connectée votre carte Romeo et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme Arduino. Attention: ce paramètre doit être spécifié en Hz, alors que la valeur correspondante dans le programme Arduino (TSDATA) est spécifiée en ms. La relation entre les deux est la suivante:
$$\text{freq (Hz)} = 1000 / \text{cadence (ms)}$$
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.
Attention : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- Zone « Consigne »: elle permet de modifier la consigne de vitesse avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel
- Zone « Asservissement »: elle permet de modifier les gains du régulateur PID pendant le fonctionnement du système. Cela permet de voir l'influence de ces gains sur les performances de l'asservissement et de régler précisément ce dernier.

IMPORTANT !

Si l'affichage ne suit pas les consignes même avec une fréquence de rafraîchissement très lente, veuillez diminuer la cadence d'échantillonnage dans le programme Arduino (augmenter la valeur TSDATA). Ne pas modifier la valeur CADENCE_MS, sous peine de déstabiliser l'asservissement.

9 - Autre documentation disponible

Cette documentation, bien que baptisée « complète », ne détaille pas certains points qui sont présents dans une autre documentation disponible sur simple demande à l'adresse contact@geeros.com:

- Documentation pédagogique: elle contient tous les éléments relatifs à la mise en équations du robot ainsi que les documents Maple et modèles MapleSim associés. Cette documentation, utile dans le cadre de l'enseignement, n'est pas en accès libre car elle contient des informations qui peuvent être les solutions d'exercices proposés par les professeurs. Ceux-ci sont donc les seuls habilités à y avoir accès.

10 - Important

Geeros est un produit « vivant » en constant développement pour 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 (contact@geeros.com).

Ne restez jamais bloqué sans nous contacter !

Pour tout problème ou toute requête, contactez-nous à l'adresse contact@geeros.com.

11 - ANNEXE 1 – Circuit de suivi de ligne

Les deux pages suivantes peuvent être imprimées pour créer un circuit de suivi de ligne pour Geeros. Vous pouvez créer d'autres branches de circuit avec n'importe quel logiciel de dessin, à condition de respecter les mêmes caractéristiques:

- l'épaisseur de la ligne est de 200 pixels, ce qui permet de couvrir la largeur des 3 capteurs de suivi de ligne du robot
- le rayon de courbure ne doit pas être trop petit. Le virage proposé ci-après peut être considéré comme la limite réalisable par le robot

L'impression doit être réalisée sur des feuilles de papier blanc, en échelle de gris (ce qui permet d'obtenir un « vrai » noir), avec une qualité suffisamment bonne. Plus le contraste sera faible, plus le seuil de détection de ligne (voir la variable globale « SEUIL » dans le programme Arduino de suivi de ligne) devra être faible.

Lors de la réalisation d'un circuit, il est important de bien joindre les feuilles (voir de les faire chevaucher) pour qu'il n'y ait pas d'espace entre elles, même sur les bords éloignés de la ligne. En effet, si le robot quitte la ligne et qu'il rencontre un espace entre deux feuilles, il risque de croire qu'il a retrouvé la ligne et partir définitivement dans la mauvaise direction.

Enfin, nous vous conseillons de relire le paragraphe 6.3, relatif à la mise en œuvre de Geeros pour le suivi de ligne.

