

Dynaban

Création d'un firmware open-source pour les servomoteurs de la gamme MX-64.



Les servomoteurs Dynamixel dominent le marché du servo-moteur à faible encombrement de part leur qualité et leur facilité d'utilisation. Ils sont extrêmement répandus en particulier dans le domaine de la recherche et de l'éducation supérieure. Le modèle MX-64 est particulièrement intéressant pour les raisons suivantes :

- Couple important
- Capteur de position magnétique
 - Pas d'usure mécanique du capteur (pas de contact)
 - Pas de zone morte (contrairement aux capteurs potentiométriques qui ont une zone morte de 60°)
 - Excellente précision (**erreur $\leq 0.088^\circ$**)
- Mesure du courant et donc du couple accessible à bord

Le servomoteur est commandé par un micro-processeur dont le firmware permet de réaliser des actions de base et en particulier l'asservissement en position avec un PID. Mais, ce firmware n'est pas open-source et arrive à ses limites dès que l'on désire un contrôle avancé du moteur. Vu l'intérêt porté par la communauté robotique à cette problématique et trouvant qu'il est dommage que ce soit le logiciel qui limite le potentiel du hardware, nous avons décidé de créer un **firmware alternatif open-source**.

Cette réalisation s'est faite en 2 parties :

- Le rétro-engineering matériel. Ouverture du servomoteur, identification et gestion de la communication avec les différents composants électroniques (communication série, capteur de position, capteur de courant, thermistor, driver de pont en H).
- La création logicielle. Utilisation des blocs matériels élémentaires pour créer des fonctionnalités, comme l'asservissement en position ou la gestion du protocole de communication avec l'extérieur.

Nous avons obtenu un firmware proposant les mêmes fonctionnalités que la version originale, de sorte que le code existant reste compatible avec le nouveau firmware. Ensuite, nous nous sommes penchés sur des problématiques plus avancées comme le suivi de trajectoire en utilisant une commande prédictive. Ce qui revient à résoudre le problème suivant « **connaissant la physique du moteur et la trajectoire cible, quelle est la commande idéale à envoyer au moteur à chaque instant ?** ». Nous avons embarqué un modèle du moteur qui prend en compte :

- Un modèle électrique du moteur (relations entre tension, courant, vitesse de rotation et couple)
- Un modèle des forces de frottement (frottement statique, de coulomb et visqueux)
- Un modèle inertiel

A ce stade, l'utilisation de la commande prédictive basée sur ce modèle (+contre-réaction classique) a permis :

- *L'amélioration de la précision dynamique en suivi de trajectoire dans un rapport 100 par rapport à l'approche classique de la contre-réaction seule*
- *La possibilité d'une approche « contrôle en couple » du moteur*
- *Des comportements « semi-compliant », interagissant avec l'utilisateur.*
- *La réalisation d'un bras à 1 degré de liberté anti-gravité et anti-friction, avec une masse arbitraire*

Nous avons eu d'ores et déjà plusieurs manifestations d'intérêt pour ce firmware, et les améliorations qu'il apporte.

Projet étudiant 1

Portage et continuation du firmware sur les servomoteurs de la gamme MX-28, MX-64 et MX-106

La réalisation du rétro-engineering d'un servomoteur est un projet pluridisciplinaire à fort potentiel pédagogique.

Intérêt pédagogique :

- Hardware : connaissance pratique des différents composants d'une carte électronique de contrôle.
- Développement logiciel embarqué contraint
- Mise en pratique des notions d'automatique et quantification des intérêts de l'approche
- Possibilité d'explorer de nouvelles améliorations pour le servomoteur (protocole de communication, calibration automatique des paramètres du modèle)

A la fin du projet, le servomoteur ne sera plus une boîte noire qui fait ce qu'on lui demande, l'étudiant aura une image précise du hardware, des problématiques de développement bas niveau et des algorithmes d'automatique appliqués.

Intérêt scientifique :

- Le portage du firmware à une autre gamme de servomoteurs bénéficie directement les utilisateurs et permet d'étendre l'utilisation des nouvelles techniques aux applications réelles qui utilisent rarement qu'un type de servomoteurs.
- Le firmware est très récent, des améliorations à forte valeur ajoutée n'attendent que des développeurs

Format :

Nombre d'étudiants : 2 à 6 (pas forcément de la même filière). Un binôme par type de servomoteur

Compétences nécessaires : connaissance du C

Compétences souhaitables : notions d'asservissement, connaissances en électronique

Nombre d'heures pour le portage d'un servo : 15-20 heures

niveau : Bac +4 à Bac +5 (ou profil particulier)

Environnement logiciel nécessaire :

- Aucun outil ne nécessite de licence
- Postes linux

Matériel nécessaire :

- Poste de soudage
- 1 multi-mètre et 1 oscilloscope par binôme
- Des USB2dynamixel
- 1 alimentation stabilisée par binôme (avec limitation de courant)
- Des rallonges USB

Projet étudiant 2

Contrôle d'un bras robotisé à n degrés de libertés

Le projet consiste à construire un bras avec n degrés de liberté, motorisé par des servomoteurs commandées par le firmware open-source. Une fois la phase théorique permettant d'obtenir le modèle inverse du bras, plusieurs challenges techniques sont imaginables.

- Possibilité de réaliser le bras à l'imprimante 3D avec les étudiants
- Challenge 1 : bras dessinateur (formes géométrique, écriture d'imprimerie, écriture manuscrite)
- Challenge 1+ : bras dessinateur sur surface non plane
- Challenge 2 : bras antigravité et antifriction
- Challenge 3 : bras capable de dribbler avec une balle de basket

Format :

Nombre d'étudiants : 3 à 6 (pas forcément de la même filière).

Compétences nécessaires : connaissances en C et python

Compétences souhaitables : notions d'asservissement

Nombre d'heures pour 1 challenge : ~20 heures

niveau : Bac +4 à Bac +5 (ou profil particulier)

Environnement logiciel nécessaire :

- Aucun outil ne nécessite de licence
- Postes linux

Matériel nécessaire :

- Des USB2dynamixel
- 1 alimentation stabilisée
- Des rallonges USB

Proposition de projet étudiant 3

Locomotion compliant bipède

Le projet consiste à mettre en œuvre sur un robot humanoïde, sur la base du firmware open-source, une technique de locomotion bipède utilisant un contrôle moteur compliant.

L'avantage d'une telle technique est d'absorber les chocs du à la marche, par exemple lorsque le robot pose le pied à terre, ou encore rencontre un obstacle ou une aspérité sur le sol. A l'instar du contrôle musculaire animal, les mouvements du robot peuvent s'adapter de façon directe à l'environnement.

Cependant, le contrôle robotique devient plus complexe et ne peut plus se baser sur un contrôle pur en position. Il s'agit donc d'explorer des techniques comme le « computed torque » basée sur un modèle dynamique du robot (par ailleurs en développement au sein de l'équipe Rhoban).

Dans une étape, il s'agira de maintenir le robot debout, sans se déplacer, et l'équilibrer sous l'action de perturbations aléatoires d'amplitude bornée.

Format :

Nombre d'étudiants : 3 à 6 (pas forcément de la même filière).

Compétences nécessaires : connaissances en C++

Compétences souhaitables : notions d'asservissement

Nombre d'heures : ~20 heures

niveau : Bac +4 à Bac +5 (ou profil particulier)

Environnement logiciel nécessaire :

- Aucun outil ne nécessite de licence
- Postes linux

Matériel nécessaire :

- 1 routeur wi-fi
- Des USB2dynamixel
- 1 alimentation stabilisée
- Des rallonges USB

Budget Matériel

L'équipe Rhoban peut fournir le matériel électronique et mécanique. En revanche, le matériel manquant nécessaire pour ces projets est les moteurs :

- **2 packs de 6 servomoteur de type MX-106R, disponible à l'URL suivante :**

<http://www.generationrobots.com/fr/401568-pack-de-6-servos-dynamixel-mx-106r.html>

Coût pour les 2 packs : 4375,98 euros TTC

- **2 packs de 6 servomoteurs de type MX-64R, disponible à l'URL suivante :**

<http://www.generationrobots.com/fr/401569-pack-de-6-servos-dynamixel-mx-64r.html>

Coût pour les 2 packs : 3169,98 euros TTC

Coût total : 7545,96 euros TTC

A noter que ces servos moteurs sont interchangeables pour les différents projets.