



Conception et Réalisation d'un Système de Contrôle Optimal d'Orientation d'un Hydrophone Ultrasonore

Y. Ouahab et H. Khelladi

U S T H B, BP 32 El Allia, Bab-Ezzouar, Alger, Algérie, 16000 Alger, Algérie
ouahab.yahia@hotmail.com

L'objectif de notre travail est de réaliser une commande PID pour un moteur à courant continu, dans le but de l'asservir en orientation. Le fonctionnement consistera à donner une consigne d'orientation en entrée, puis la comparer avec une valeur de retour (feedback) et de ce fait apporter une régulation du système pour qu'il agisse avec précision et fidélité par rapport à la consigne de départ. La régulation se fera évidemment en boucle fermée pour avoir un retour de consigne. Notre dispositif sera réalisé à base du microcontrôleur ARM LPC Expresso qui permettra un contrôle assez rigoureux des différents paramètres PID moyennant une conversion analogique/numérique pour les afficher, par la suite, sur un écran LCD. L'implémentation de ces paramètres est basée sur la méthode de Ziegler-Nichols (ZN). De plus, nous aurons besoin d'un circuit pont en H pour gérer le sens de rotation du moteur et fournir un courant suffisant afin de faire tourner le moteur. Enfin, un compas se place sur l'hydrophone pour le retour d'information (Feedback).

1 Introduction

Actuellement les moteurs à courant continu sont utilisés dans diverses applications robotiques, allant des jouets aux engins spatiaux [1,2]. Dans ces applications la vitesse et la position sont généralement contrôlées par la proportion d'intégration (PI), la proportion-dérivation (PD), et la proportion-intégration dérivation (PID). En raison de la haute performance de contrôle de la vitesse et de la position, le régulateur PID a été inclus dans notre réalisation [3,4]. Le travail développé dans cette étude est consacré à la conception et réalisation d'un système automatique de balayage pour l'analyse du profil vibratoire d'un transducteur via un hydrophone ultrasonore.

2 Partie hardware

2.1 Descriptif général

Le dispositif réalisé comprend essentiellement trois parties, introduction des consignes, algorithme de commande, et le retour d'information (Fig. 1). La consigne sur l'orientation ainsi que les différents paramètres PID sont introduits moyennant une conversion analogique/numérique du microcontrôleur ARM LPC Expresso sur lequel est basé notre travail, puis visualisés sur un afficheur LCD. Le signal PWM généré par le microcontrôleur est insuffisant pour le moteur, c'est la raison pour laquelle nous avons jugé utile d'insérer un circuit pont en H pour garantir non seulement la fourniture du courant nécessaire pour faire tourner le moteur mais aussi pour gérer le sens de rotation de ce dernier. Le retour d'information (Feedback) est assuré par un compas via le module capture qui mesure la fréquence du signal. Enfin, un algorithme de commande PID est implémenté sur le microcontrôleur et cela pour optimiser au mieux le contrôle d'orientation.

2.2 Calculateur

Dans le cadre de cette étude, nous nous sommes intéressés à une nouvelle famille de microcontrôleurs qui est ARM-Cortex M3-LPC XPRESSO -17XX de chez NXP Semiconductor (Fig. 1). Ce sont des microcontrôleurs 32-bits à base d'architecture ARM-CORTEX M3 qui incorpore un pipeline à trois étages ainsi qu'une architecture HARVARD dont le CPU opère à une fréquence de 100 MHz et peut aller jusqu'à 120 MHz. Dans le cadre de cette réalisation, nous avons choisi le LPC XPRESSO- 1769. Ce dernier étant le microcontrôleur le plus performant de la gamme 17XX, ce qui explique notre choix.

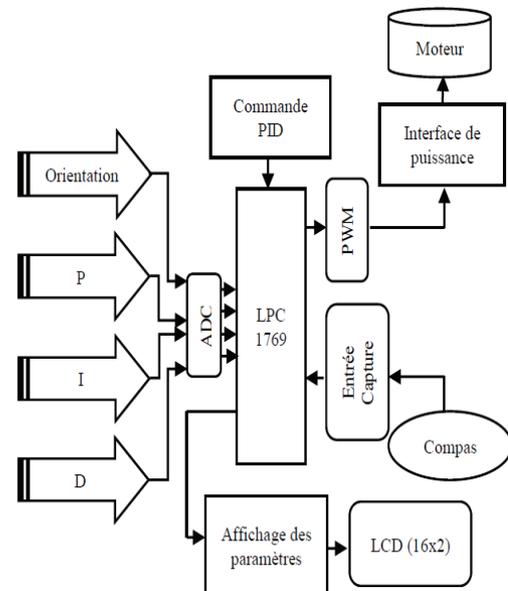


Figure 1: Schéma synoptique.

2.3 Moteur à courant continu

Nous avons utilisé un moteur à courant continu avec une vitesse nominale de 130 (tr/min) et un couple nominal de 58.8mN.m. Le poids de ce moteur est de 60g. La tension d'entrée nominale est de 12V. Notons que la modulation de la largeur d'impulsion de commutation (PWM) est la technique de contrôle de la vitesse la plus largement utilisée pour les moteurs à courant continu, c'est la raison pour laquelle nous avons jugé utile de l'utiliser dans le cadre de cette réalisation [5]. De plus pour commander le moteur, L293D représentant un double pont en H est utilisé (Fig. 2). Ce circuit permet d'inverser la tension aux bornes du moteur pour inverser son sens de rotation, mais aussi il sert à fournir un courant suffisamment important pour faire tourner le moteur qui ne marcherait pas si on le branchait directement à la sortie d'un microcontrôleur.

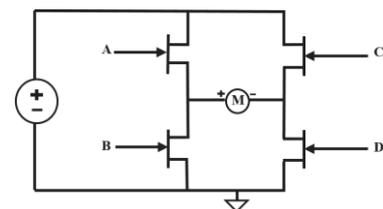


Figure 2: Schéma synoptique du L293D.

2.4 Compas

Le module CMPS03 dont nous disposons incorpore un capteur de champ magnétique, le KMZ51 de PHILIPS assez sensible pour détecter le champ magnétique terrestre. Ainsi, il sera possible de repérer notre hydrophone par rapport au nord puisque le module indique toujours l'angle qu'il fait par rapport au nord. Le CMPS03 dispose d'une sortie PWM (pin 4) qui délivre une impulsion dont la largeur est proportionnelle à l'angle. Pour ce faire, le timer 16 bits du microcontrôleur 18Fxx intégré au module est utilisé pour faire varier la largeur d'impulsion de 1ms correspondant à 0° (nord) jusqu'à 36,99ms correspondant à 359,9°, autrement dit, une résolution de 100µs/° avec un offset de 1ms. On peut aussi exploiter les informations délivrées par ce module au travers d'une interface I2C entre lui et un microcontrôleur et ce en lisant directement dans ses registres.

En résumé de son fonctionnement, il faut d'abord envoyer une « start- condition » ou « start-bit », ensuite l'adresse du module (0x0C), puis vient le numéro du registre qu'on veut lire (de 0 à 15) et enfin le « stop-bit ». Dans le cas d'une deuxième lecture, il suffit d'envoyer un « repeated start-bit » suivi de l'adresse du module (0X0C) et du numéro du deuxième registre qu'on veut lire. L'information sur l'angle est obtenue par une lecture du registre 1 avec des valeurs allant de 0 à 255 correspondants à un tour complet (360°). Pour une meilleure résolution, la lecture des registres 2 et 3 s'impose puisque l'information est donnée sur 16 bits, la valeur 3599 représente alors l'angle 360°. Pour ce qui est des autres registres, il s'agit de registres pour le test et la calibration du module. Notons qu'un dispositif d'affichage de type 2x16 est inclus dans cette réalisation et cela pour afficher en premier temps la consigne demandée (orientation) ainsi que les différents paramètres PID, par la suite il servira à suivre en temps réel la rotation de l'hydrophone.

3 Partie Software

3.1 Calcul de l'angle de rotation

Nous avons utilisé le module compas CMPS03 présenté précédemment. Aussi, nous avons choisi d'exploiter l'information sur l'angle de lacet non pas par communication I2C, mais plutôt à travers sa sortie PWM car plus facile à élaborer et moins sensible aux bruits (Fig. 3). Rappelons, qu'en mode PWM, le compas délivre une impulsion dont la largeur est proportionnelle à l'angle qu'il fait par rapport au nord terrestre.

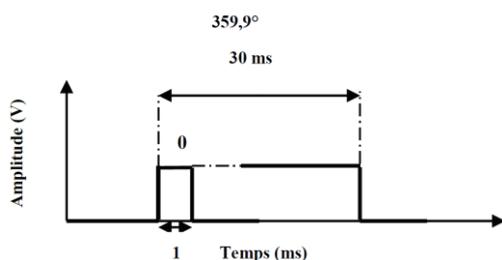


Figure 3: Schéma explicatif de la relation angle/largeur d'impulsion.

3.1.1 Calcul de la largeur d'impulsion

Le microcontrôleur ARM-LPC1769 que nous avons utilisé dispose de deux entrées capture par timer. Il faut savoir qu'une entrée capture permet de prendre des clichés du contenu du timer associé, et ce lors de la détection d'un événement sur cette dernière. Les événements possibles sont la détection d'un front montant, un front descendant, ou bien les deux à la fois sans oublier qu'il est aussi possible de générer une interruption lors de la détection d'événements.

Le principe de calcul de la largeur d'impulsion élaboré est le suivant : tout d'abord, nous avons configuré les entrées captures du timer 2 sur un front montant et activé l'interruption associée. Ensuite, dès que l'impulsion arrive sur l'entrée capture, la valeur du timer 2 à cet instant précis est affectée au registre CR0 déclenchant ainsi une interruption. Ce qui nous permet de sauvegarder cette valeur dans une variable X. Evidemment le timer 2 ne s'arrête pas de compter, il suffit alors d'imposer une condition sur le retour de l'entrée de capture au repos (état logique bas) puis de lire la valeur du timer 2 à cet instant, et la stocker par la suite dans une autre variable Y. Ainsi en soustrayant la variable X à la variable Y, nous obtenons une valeur numérique correspondant à l'intervalle de temps dans lequel l'entrée capture est à l'état logique haut, autrement dit la largeur d'impulsion. Enfin, il ne reste plus qu'à convertir la valeur numérique (Y-X) en temps (ms). Comme la fréquence de comptage de timer est de 25 MHz, la conversion de la valeur numérique (Y-X) en temps (ms) se fait comme suit :

$$\text{Largeur d'impulsion (ms)} = \frac{\text{valeur numérique (Y-X)}}{25 \cdot 10^6} \quad (1)$$

$$\text{valeur numérique (Y-X)} / 25 \cdot 10^6$$

En se reportant au datasheet du CMPS03, la sensibilité est de 100 µs/°, donc on peut déduire la valeur de l'angle suivant la relation ci-dessous :

$$\text{Angle (°)} = \frac{\text{Largeur d'impulsion (ms)}}{100(\mu\text{s})} \quad (2)$$

Ci-dessous, est illustré l'organigramme de calcul de la largeur d'impulsion :

3.2 Régulateur PID

Les contrôleurs (PID) sont largement utilisés dans les systèmes de contrôle industriels en raison du nombre réduit de paramètres à l'écoute [1,2]. Ils fournissent des signaux de commande qui sont proportionnels à l'erreur entre le signal de référence et la sortie réelle (action proportionnelle), à l'intégrale de l'erreur (action intégrale), et à la dérivée de l'erreur (action dérivée) :

$$u(t) = K_p [e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{d}{dt} e(t)] \quad (3)$$

Avec K_p est le gain proportionnel, T_d représente la constante de temps de dérivation, et T_i est la constante de temps d'intégration.

La transformée de Laplace de la fonction de transfert correspondante $K(s)$ est donnée par :

$$K(s) = K_p \left[1 + \frac{1}{T_i(s)} + T_d(s) \right] \quad (4)$$

Ces fonctions sont suffisantes pour la plupart des processus de contrôle. Le régulateur PID est principalement utilisé pour ajuster un gain approprié proportionnel (KP), gain intégrale (KI) et gain différentiel (KD) pour atteindre les performances de la commande optimale [6].

3.3 Méthode de Ziegler-Nichols en boucle fermée

Le principe de réglage des paramètres PID par la méthode Ziegler-Nichols est illustré à la figure 5.

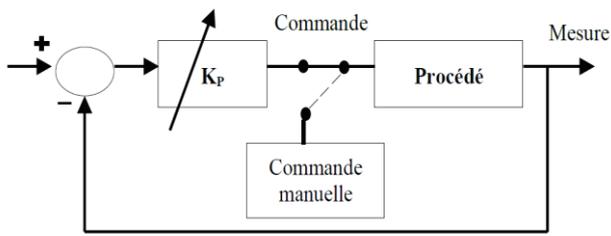


Figure 5: Schéma de principe du régulateur PID Ziegler-Nichols

Le régulateur est en mode automatique avec une faible valeur de K_p . Les actions I et D sont inhibées en mettant $T_i = T_{i\max}$ et $T_D = 0$. Nous augmentons progressivement le gain K_p du correcteur proportionnel agissant seul jusqu'à l'obtention de la bonne oscillation de la boucle (pompage).

3.3.1 Exploitation du résultat de pompage de la boucle

On relève le gain limite (K_{pLim}) conduisant au pompage de la boucle et la période des oscillations (T_{Lim}) correspondant à ce fonctionnement à partir de n'importe quel point d'observation (sortie du régulateur, sortie du procédé) (Fig. 6).

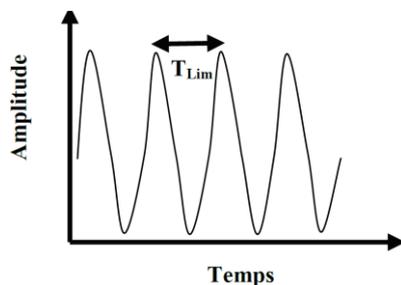


Figure 6: Courbe de la période de l'oscillation T_{Lim}

3.3.2 Réglage des paramètres du régulateur PID

Ziegler & Nichols proposent de calculer les paramètres du régulateur choisi à l'aide des recommandations suivantes [7,8] :

Tableau 1: Réglage ZIEGLER-Nichols.

Réglage des paramètres			
Régulateur	K_p	T_i	T_d
PID	$0.58 K_{pLim}$	$0.5 T_{Lim}$	$0.125 T_{Lim}$

3.4 Discrétisation de la loi de commande

En ce qui concerne la régulation, elle se fera par programmation à l'aide de l'algorithme correspondant à la fonction de transfert du régulateur PID [9,10]. Ce dernier est obtenu après discrétisation de la loi de commande continue du régulateur PID. Il se résume aux équations ci-dessous [5] :

Erreur = valeur de consigne – valeur de retour
 Intégrale = Intégrale + Erreur
 Dérivée = Erreur – Erreur précédente
 Sortie = (P x Erreur) + (I x Intégrale) + (D x Dérivée).

3.5 Programme principal

Le programme se fera en boucle infinie relevant continuellement les valeurs des paramètres PID et celle de la consigne (angle) puis les afficher sur le LCD, ainsi à la moindre variation d'un quelconque potentiomètre on obtiendra sa valeur correspondante sur l'afficheur. En appuyant sur le bouton poussoir, nous initialisons le timer 1 à 0,5s ainsi que l'interruption relative à son débordement.

Par la suite lors du débordement, le programme basculera à la routine d'interruption qui comporte notamment l'algorithme du régulateur PID. Le moteur tournera alors avec l'angle voulu. L'organigramme suivant explique le déroulement du programme (Fig. 7).

3.6 Résultats et discussion

Lors des tests effectués, nous avons constaté que la réaction du moteur dépendait des paramètres PID et que le moteur se mettait en marche mais ne s'arrêtait pas à la position désirée et cela parce que la valeur du PID n'était pas adaptée à cet angle. Alors nous avons choisi d'autres valeurs du PID commençant par des petites valeurs afin d'arriver à l'angle désiré avec un bon compromis entre précision et stabilité en se basant sur la technique de Ziegler-Nichols. Nous avons remarqué qu'il fallait choisir une faible valeur du P (action proportionnelle) pour une consigne très petite, car l'action proportionnelle fait augmenter l'erreur pour diminuer le temps de montée, et donc automatiquement le moteur va dépasser l'angle désirée et ne s'arrêtera jamais.

4 Conclusion

Dans cette étude, une approche pour le réglage des paramètres du régulateur PID a été présentée et appliquée à la commande en orientation d'un moteur à courant continu. Le régulateur PID numérique est conçu de telle sorte qu'il minimise la somme des temps, temps de montée, dépassement maximal et l'erreur. Les résultats obtenus grâce à la manipulation de moteur à courant continu montrent que régulateur PID conçu avec l'approche proposée peut améliorer la performance dynamique du système. Le régulateur PID conçu avec la méthode de Ziegler-Nichols présente des performances satisfaisantes se traduisant par un bon comportement du moteur avec un dépassement qui diminue, un temps de stabilisation et de montée relativement faibles.

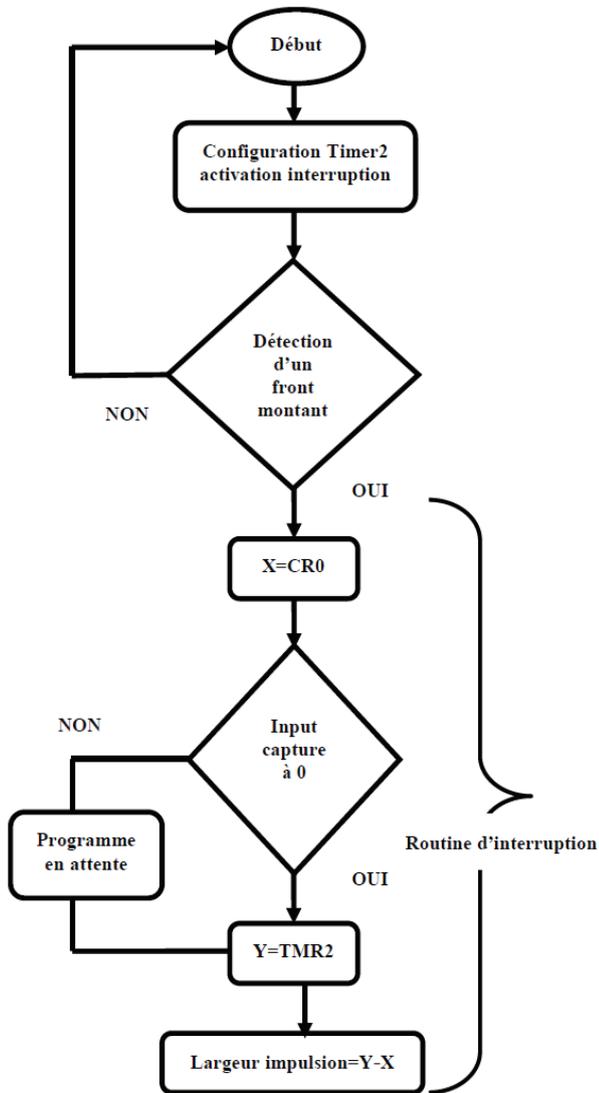


Figure 4: Organigramme de calcul de la largeur d'impulsion.

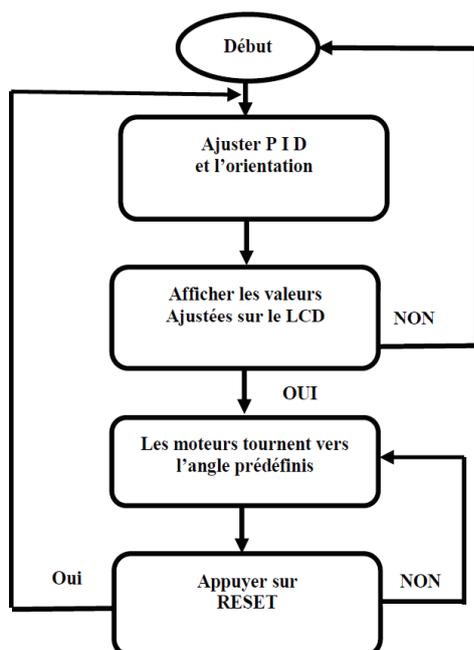


Figure 7: Organigramme du programme principal

Références

- [1] B. Allaoua, B. Gasbaoui and B. Mebarki, Setting Up PID DC Motor Speed Control Alteration Parameters Using Particle Swarm Optimization Strategy, *Leonardo Electronic Journal of Practices and Technologies*, ISSN 1583-107, 19-32 (2009).
- [2] W. P. Aung, Analysis on Modeling and Simulink of DC Motor and its Driving System Used for Wheeled Mobile Robot, World Academy of Science, *Engineering and Technology* **32**, 299-306 (2007).
- [3] S. Duman, D. Maden and U. Guvenc, Determination of the PID Controller Parameters for Speed and Position Control of DC Motor using Gravitational Search Algorithm, *7th International Conference on Electrical and Electronics Engineering (ELECO)*, 1225-1229 (2011).
- [4] N. Tutkun, D. Maden, Determination of the PID controller parameters for a DC shunt motor using the genetic algorithm method, *National Conference on Electrical, Electronics and Computer Engineering (ELECO)*, 330-333 (2010).
- [5] A. R. Paul, M. George, Brushless DC motor control using digital PWM techniques, *International Conference on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN)*, 733-738 (2011).
- [6] R. G. Kanojiya, P. M. Meshram, Optimal tuning of PI controller for speed control of DC motor drive using particle swarm optimization, *International Conference on Advances in Power Conversion and Energy Technologies (APCET)*, 1-6 (2012).
- [7] S. Umamaheswari, V. Palanisamy and M. Chidambaram, A simple method of tuning PID controllers for interval plant of cold rolling mill, *International Conference on Control, Automation, Communication and Energy Conservation (INCACEC)*, 1-6 (2009).
- [8] J. G. Zielgler, N. B. Nichols, Optimum settings for automatic controllers, *ASME Transactions*, **64**, 759 (1942).
- [9] A. Meghebbar, *Introduction aux systèmes linéaires numériques*, Université Aboubekr Belkaid Tlemcen Faculté de Technologie Département de Génie Electrique et Electronique /GEE.
- [10] Y. Granjon, *AUTOMATIQUE Systèmes linéaires, non linéaires, à temps continu, à temps discret, représentation d'état Cours et exercices corrigés*, Paris, 2001, 2010 ISBN 978-2-10-055087-6.