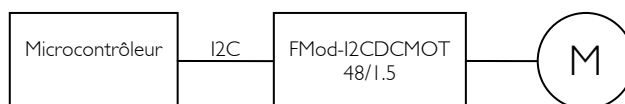


La carte FMod-I2CDCMOT 48/1.5 peut-être contrôlée directement à partir d'un microcontrôleur présentant une interface série de type I2C. FiveCo recommande l'utilisation d'une carte FMod-TCP DB ou FMod-TCP BOX pour la configuration et la familiarisation avec la carte FMod-I2CDCMOT 48/1.5, mais il est toutefois possible de sauter cette étape et de configurer la carte directement à partir du microcontrôleur. Ce document décrit la marche à suivre.



Exemple de moteur

Prenons comme exemple un moteur (24V), avec un courant maximum continu de 1A, avec un codeur en quadrature de 500 CPR.

500 CPR (cycle per revolution) permet de connaître 2000 (500x4) états différents en faisant un tour moteur = 2000 PPR (pulses per revolution).

Quand ce moteur tourne à 3000 tr/min = 50 tr/sec, notre module compte 50 tr/sec * 2000 (pulses/tr) = 10000 (pulses/sec).

Etape 1 – Régler les paramètres PID et ceux du mouvement

1. **Déconnecter le moteur de sa mécanique.**
2. Ecrire le registre CURRENTMAX (0x2A) avec le courant nécessaire pour l'application, avec notre exemple à 1A, la valeur à écrire est 0x00010000.
3. Ecrire 0x01 dans le registre AUTOTUNING (0x39) pour démarrer la recherche automatique des paramètres.
 - o Le moteur se met à tourner pendant environ 2 secondes.
 - o Lire ce registre plusieurs fois et attendre qu'il revienne à 0 (OK) ou à 4 (erreur).
 On peut aussi attendre 4 secondes pour être sûr que l'autotuning est terminé.
4. Sauvegardez tous les paramètres qui viennent d'être modifiés avec la fonction SAVEUSERPARAMETERS (0x03)

Etape 2 – Tester le moteur en position

Uniquement après avoir configuré les paramètres PID et ceux du mouvement.

1. (option) écrire 0 dans le registre POSITION (0x26); ainsi, la position actuelle (à l'arrêt) du moteur est à 0 (pulses).
2. Ecrire 2000 (0x000007D0) dans le registre INPUT (0x21), ceci sera la nouvelle position (goal) à atteindre (en pulses).
3. Ecrire 0x05 dans le registre REGULATIONMODE (0x20), pour configurer le module en mode position avec suivi de trajectoire.

Le moteur se met à tourner et s'arrête sur la position 2000, avec notre exemple 2000 représente 1 tour du moteur.
4. (option) pour un nouveau but à atteindre (pulses), écrire la nouvelle valeur dans le registre INPUT (0x21),

Il n'est pas nécessaire de configurer REGULATIONMODE qui est déjà dans le mode position avec suivi de trajectoire.

Etape 3 – Tester le moteur en vitesse

Uniquement après avoir configuré les paramètres PID et ceux du mouvement.

1. Ecrire 10'000 (0x00002710) dans le registre INPUT (0x21), ceci sera la nouvelle vitesse à atteindre (pulses/sec).
2. Ecrire 0x04 dans le registre REGULATIONMODE (0x20), pour configurer le module en mode vitesse.

Le moteur se met à tourner à 10'000 pulses/sec (= 3'000 tr/min pour notre exemple).
3. (option) pour une nouvelle vitesse à atteindre, écrire la nouvelle valeur dans le registre INPUT (0x21),

Il n'est pas nécessaire de configurer REGULATIONMODE qui est déjà dans le mode vitesse.

Etape 4 – Configuration de la position de référence (Homing)

Il existe plusieurs manières pour définir la position de référence. Nous allons utiliser la plus répandue.

1. A une des extrémités du mouvement mécanique, nous fixons un capteur de fin de course (compatible 5 volts): un switch, une barrière optique ou un capteur inductif.
2. Si le capteur doit être alimenté, il faut le relier électriquement à la carte FMod-I2CDCMOT (DB) 48/1.5, 0v et 5v doivent être connecté aux bornes du module. Si c'est un capteur non alimenté (switch), connecter une borne au 5v de la carte FMod-I2CDCMOT (DB) 48/1.5
3. Relier la ligne de retour du capteur sur la borne Limit 1 (L1-Home) de la carte FMod-I2CDCMOT (DB) 48/1.5

4. Regarder la documentation du capteur, et observez de quel type est le signal en sortie. Très souvent il n'y a qu'un seul transistor (bipolaire ou MOS).
5. Si le transistor de sortie est un bipolaire-PNP ou MOS-P, il est nécessaire d'avoir une résistance qui définit l'état 0 = résistance pull-down. Si le transistor de sortie est un bipolaire-NPN ou MOS-N, il est nécessaire d'avoir une résistance qui définit l'état 5v = résistance pull-up.
6. Les pull-up ou pull-down sont déjà existantes sur la carte FMod-I2CDCMOT (DB) 48/1.5. Il faut par contre définir celle qu'il faut activer. Lisez le registre OPTIONS (0x2C) et modifiez le bit 5 à 1 pour activer les pull-up, à 0 pour activer les pull-down.
7. Maintenant il s'agit de savoir quand votre capteur est actif: lorsque sa sortie vaut 0v (0) ou 5v (1)?

Si vous n'êtes pas sûr en lisant la documentation de votre capteur, utilisez un voltmètre et mesurez la tension lorsque votre mécanique est sur le capteur. Si vous observez 5v, votre capteur est actif à 1, si vous observez 0v, votre capteur est actif à 0.

Pour vérifier que votre système est correctement câblé, déplacez votre mécanique loin du capteur, le signal DOIT s'inverser!
8. Ecrire le registre LIMIT1SETUP (0x50) avec le bit 0 à la valeur 1 et le bit 1 à la valeur définie au point précédent.
9. Quand votre mécanique est loin du capteur L1, lisez le registre POSITION. Quand vous déplacez la mécanique dans la direction du capteur, relisez le registre POSITION: est-il plus grand (+) ou plus petit (-).

Ceci définit le mode de Homing avec la limit 1: L1+ ou L1- .
10. Ecrire le registre HOMINGOPTIONS (0x48) avec la valeur 0x00 BB 4B 0y, avec y=8 pour L1-, ou y=9 pour L1+
11. Décidez quelle sera la valeur attribuée à la position quand L1 sera activé lors du homing et écrivez cette valeur dans le registre HOMINGPOSITION (0x4B).
12. Dès que le système sera calibré, où doit-il aller se positionner ?

Décidez de la position que vous souhaitez et écrivez-la dans HOMINGINPUT; si vous n'avez pas d'idée, écrivez la même valeur que celle du point précédent.
13. Sauvegardez tous les paramètres qui viennent d'être modifiés avec la fonction SAVEUSERPARAMETERS (0x03) et attendez ~0.5seconde pour que les paramètres soient enregistrés en EEPROM.
14. Lancez le Homing avec la fonction HOMING (0x49).

Le moteur tourne jusqu'à ce que la limit L1-Home soit activé, puis la référence est prise, et finalement, le système va à sa position de repos.

Auteur : Xavier Greppin 11.2006

Révision 1.0