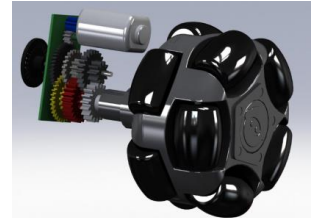


Mise en situation

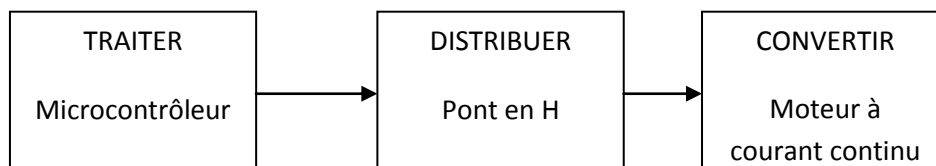
Comment piloter les roues du robot Rovio ?



Cahier des charges

Le cahier des charges est le suivant :

- Le traitement des informations doit se faire par un microcontrôleur (TRAITER),
- Les roues devront pouvoir être pilotées dans les deux sens et à vitesse variable (DISTRIBUER),
- L'actionneur est un moteur à courant continu (CONVERTIR).



Commande d'un moteur à courant continu

Voir cours

Choix des composants

Choix du microprocesseur :

Tout microprocesseur peut-être utilisé dans notre application. Pour des raisons de programmation rapide et graphique, notre choix se portera sur un microcontrôleur de chez microchip. Nous utiliserons un 16F88. Il possède 5 ports 2 ports programmables en sortie, soit 16 sorties.



Ce microprocesseur est programmable via le logiciel flowcode.

Choix du distributeur

Le pilotage d'un moteur à courant continu à partir d'un microcontrôleur se fait par l'intermédiaire d'une interface de puissance, il doit pouvoir inverser le sens de rotation des roues.

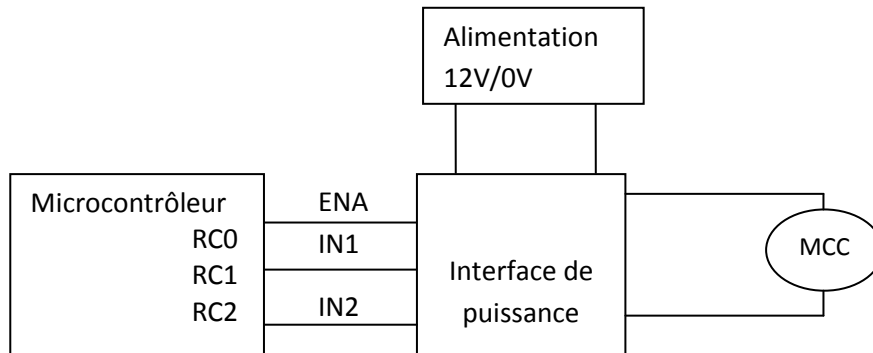
L'interface de puissance le plus adapté est donc un pont en H piloté par trois entrées dont voici la table de vérité :

ENable	IN1	IN2	Etat moteur
0	X	X	Arrêt
1	1	0	Sens 1
1	0	1	Sens 2

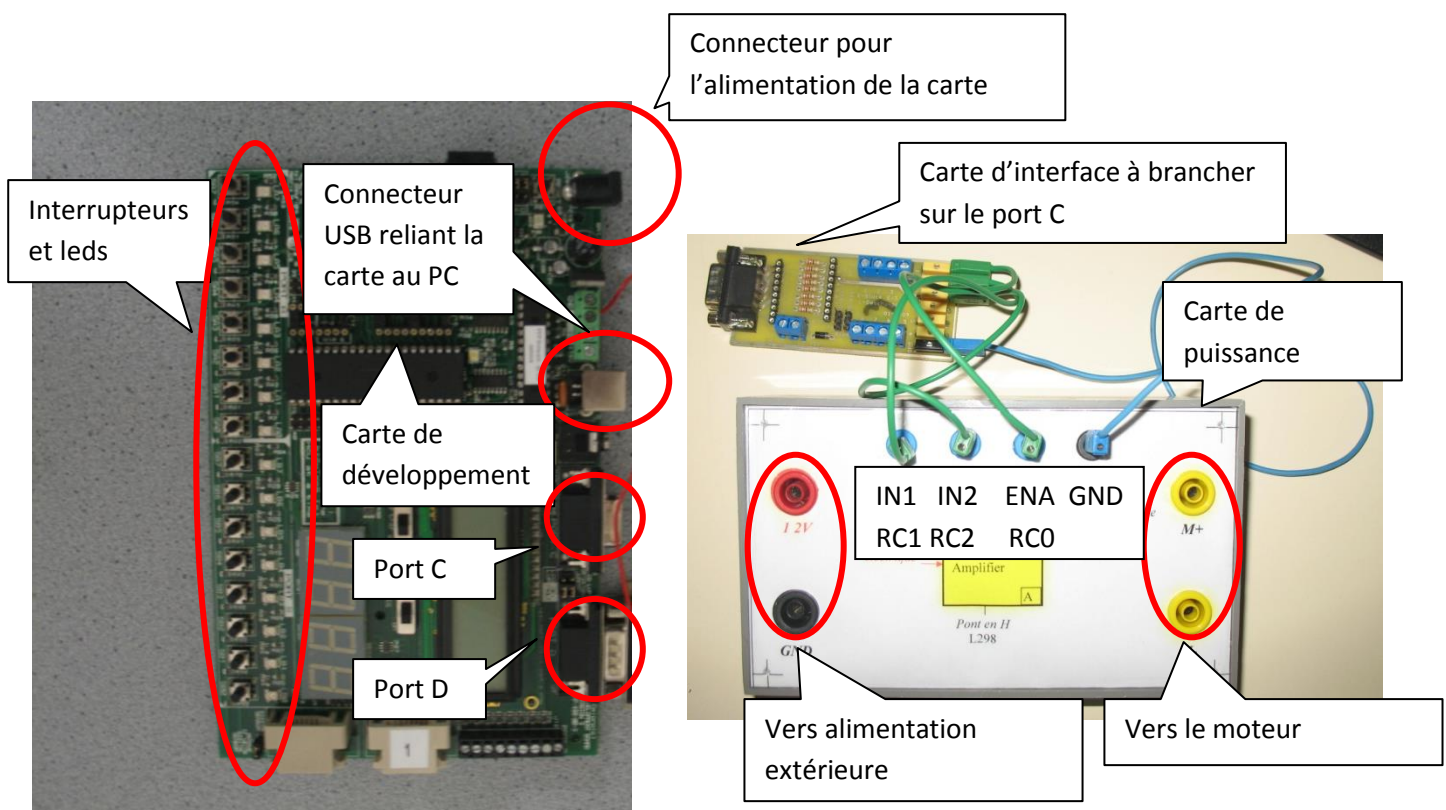
Assemblage de l'ensemble

Il faut alors relier 3 sorties du microcontrôleur aux 3 entrées de commande de ce pont en H (EN, IN1 et IN2)

- La sortie RC0 est reliée de manière arbitraire à EN(able)
- Les sorties RC1 et RC2, sont deux sortie PWM (Pulse Wild Modulation), elles seront donc reliées à IN1 et IN2, afin de permettre la commande dans les deux sens à vitesse variable.
- Une alimentation extérieure est nécessaire car le moteur à courant continu nécessite beaucoup de puissance.



Câblage de l'ensemble



Les sorties à large d'impulsion, sont les sorties RC1 et RC2 du microcontrôleur :


- Relier la sortie RC0 à l'entrée Enable de l'interface de puissance
- Relier la sortie RC1 (Sortie MLI 1) à l'entrée IN1 de l'interface de puissance
- Relier la sortie RC2 (Sortie MLI 2) à l'entrée IN2 de l'interface de puissance.


Brancher la carte d'interface sur le port C,

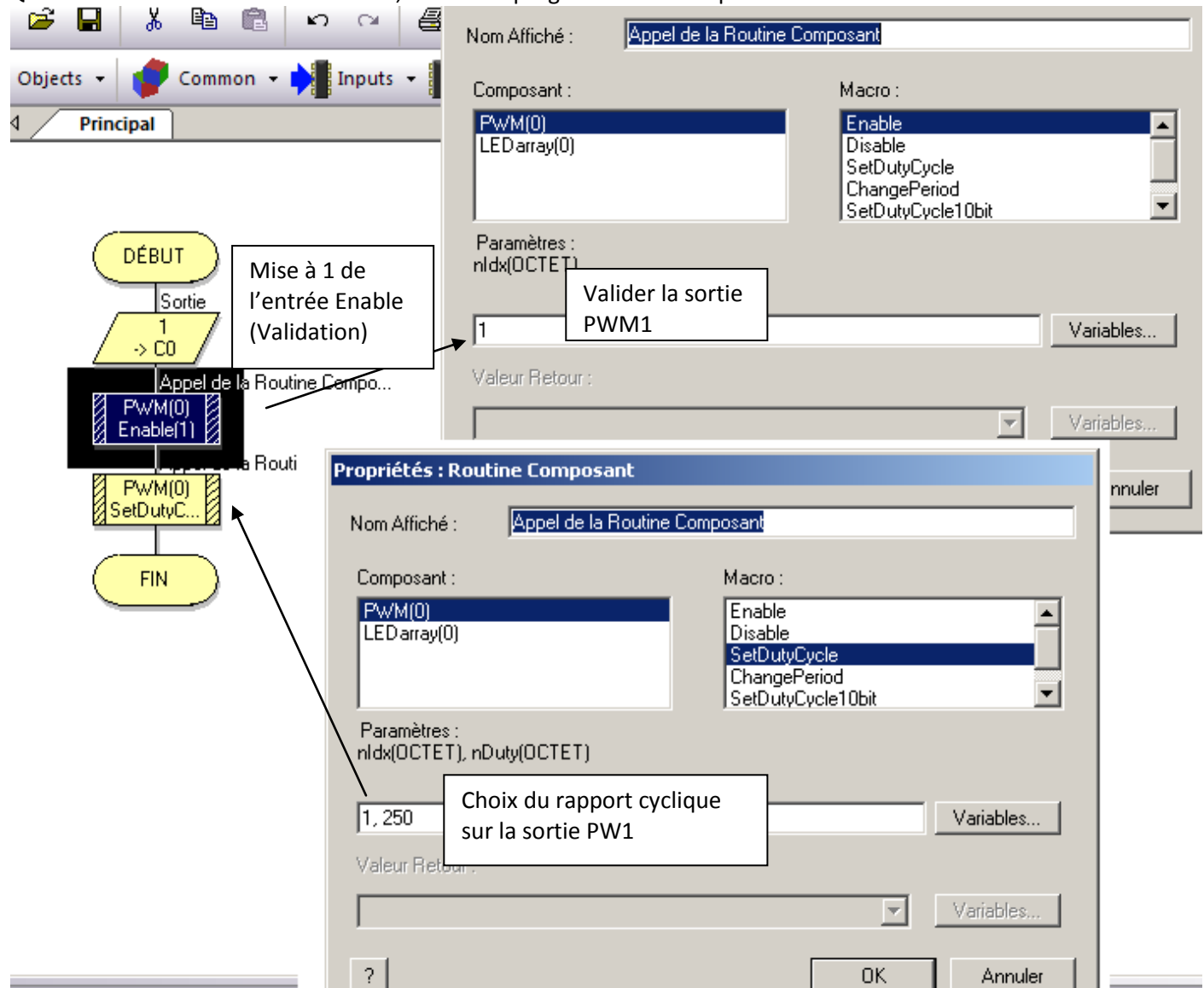
L'alimentation de la carte d'interface, doit être reliée à l'alimentation extérieure (12V/0V) car le moteur à courant continu nécessite beaucoup de courant.

Brancher le moteur à courant continu à la carte de puissance.

Pilotage d'un moteur à courant continu

Pour piloter un moteur à courant continu sous Flowcode, on utilise l'outil PWM (Pulse Width Modulation) . Il se trouve dans l'onglet « Mechatronics ». Il permet de générer les signaux pour piloter l'interface de puissance. c'est cet interface de puissance qui fournit l'énergie électrique au moteur à courant continu.

Q1. A l'aide de la notice de Flowcode, réaliser le programme suivant puis lancer la simulation  :




The image shows a Flowchart and two PWM configuration windows. The Flowchart starts with a 'DÉBUT' (Start) block, followed by a 'Sortie 1 -> C0' (Output 1 to C0) block. This is followed by a 'PwM(0) Enable(1)' block, which is annotated with 'Mise à 1 de l'entrée Enable (Validation)' (Set Enable input to 1 (Validation)). Below this is a 'PwM(0) SetDutyC...' block, annotated with 'Choix du rapport cyclique sur la sortie PW1' (Choice of duty cycle on output PW1). The Flowchart ends with a 'FIN' (End) block.


The 'Appel de la Routine Composant' (Component Routine Call) window shows the following configuration:


- Nom Affiché : Appel de la Routine Composant
- Composant : PwM(0), LEDarray(0)
- Macro : Enable, Disable, SetDutyCycle, ChangePeriod, SetDutyCycle10bit
- Paramètres : nIdx(OCTET)
- Paramètre 1 : 1 (Annotated with 'Valider la sortie PWM1' (Validate output PWM1))
- Valeur Retour : (empty)

The 'Propriétés : Routine Composant' (Component Routine Properties) window shows the following configuration:

- Nom Affiché : Appel de la Routine Composant
- Composant : PwM(0), LEDarray(0)
- Macro : Enable, Disable, SetDutyCycle, ChangePeriod, SetDutyCycle10bit
- Paramètres : nIdx(OCTET), nDuty(OCTET)
- Paramètre 1 : 1, 250 (Annotated with 'Choix du rapport cyclique sur la sortie PW1' (Choice of duty cycle on output PW1))
- Valeur Retour : (empty)

Q2. Télécharger  le programme dans le pic et le tester. Commenter les résultats de la simulation.

Q3. Réaliser un programme qui permet de faire tourner le moteur à 75% de sa vitesse dans le sens 1 si on n'appuie pas sur l'interrupteur A0 et à 50% de sa vitesse dans le sens 2 si on appuie sur A0. Vous utiliserez l'outil de décision .

Penser à insérer votre programme dans une boucle infinie afin de répéter le programme .