

# Moteurs – Pont H – PWM

---

## Résumé du démon

### Moteur

Rappel : le pack de piles fournit une **tension**, qui s'exprime en **Volts**. La tension à disposition sur le robot PRisme est d'environ 4 à 5V selon l'état de charge des piles.

Quand on connecte un moteur à une source de tension (comme une pile), il se met à tourner à une vitesse déterminée.

Si on inverse la tension (en connectant la pile à l'envers par exemple), le moteur tourne à la même vitesse, mais en sens inverse.

Nous avons vu lors de la dernière séance comment alimenter des LED à l'aide du microcontrôleur du PRisme. Malheureusement, celui-ci n'est pas capable de fournir suffisamment de puissance pour alimenter un moteur. Comment peut-on alors allumer, éteindre ou inverser le sens de rotation des moteurs du robot ? C'est le rôle du pont en H

### Pont H

Un **transistor** est un composant électronique qui fonctionne comme un **interrupteur**, que l'on peut commander à l'aide d'une tension électrique. Le microcontrôleur n'est pas capable de fournir la puissance nécessaire à faire tourner un moteur, mais il peut contrôler un transistor.

Le pont H est un montage de quatre transistors, qui permettent de connecter le moteur à l'alimentation de diverses façons.

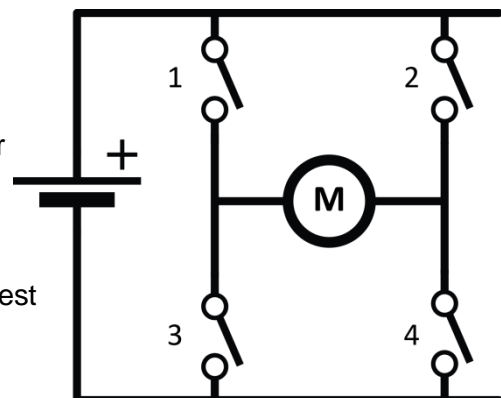
Voici deux exemples :

#### Marche avant

En fermant les interrupteurs 1 et 4, le circuit passe par le moteur dans un sens.

#### Marche arrière

En fermant les interrupteurs 2 et 3, le circuit est maintenant connecté dans l'autre sens.



### PWM

On désire maintenant, en plus d'aller en avant et en arrière, faire avancer le robot plus lentement. Il faut savoir tout d'abord que la vitesse de rotation d'un moteur est directement liée à sa tension d'alimentation. La figure ci-contre illustre ce fait par une droite, qui indique simplement que **plus la tension est élevée, plus le moteur tourne vite**.

Etant donné que la tension d'alimentation est fixée par le pack de piles, il faudra trouver une autre astuce pour faire tourner le moteur moins vite. On peut par exemple débrancher et rebrancher alternativement le moteur. S'il n'est alimenté que la moitié du temps, il s'ensuit

que le robot devrait avancer deux fois moins vite !

C'est le principe de base du PWM, qui est un module hardware disponible sur le PRisme. La librairie Robopoly comporte une fonction pour faire avancer les moteurs à une vitesse déterminée en utilisant ce module.

## Programmation

Le pont H se branche comme indiqué dans le document « *Connectique du PRisme* » à télécharger sur <http://robopoly.epfl.ch>

Pour commander les moteurs, on utilise la fonction

```
SetupMotorPWM(Gauche, Droite);
```

Où *Gauche* et *Droite* sont respectivement la vitesse du moteur gauche et du moteur droit, exprimés comme un nombre entre -100 et 100. À -100, le moteur est à pleine vitesse en marche arrière, et à 100, il est à pleine vitesse en marche avant. Avec 0, le moteur est à l'arrêt.

**A noter :** la fonction `SetupMotorPWM` initialise le module PWM à chaque appel. Après cela, la consigne est transmise aux moteurs en permanence. Il suffit donc de l'appeler à chaque fois que l'on veut changer la vitesse des moteurs, et pas plus!

Exemple : le code suivant **ne fonctionne pas** !

```
int main void()
{
    while(1)
    {
        SetupMotorPWM(50,50);
    }
}
```

Dans cet exemple, la boucle `while` contient l'initialisation du module PWM, puis le réglage de la consigne des moteurs. Ceci a pour effet de désactiver et réinitialiser le module PWM en permanence, et par conséquent, il ne se produira (presque) rien sur les moteurs.

L'exemple suivant fonctionnera bien mieux :

```
int main void()
{
    SetupMotorPWM(50, 50);
    while(1)
    {
        waitms(100) ;
    }
}
```

Ici, on initialise une seule fois le PWM, puis on attend sans rien faire pendant que le module s'occupe de faire avancer les moteurs.

Pour plus d'infos, consultez le document « *Guide de la librairie Robopoly* » que vous pouvez télécharger sur le site web du club.