Robot

1 -Présentation du module

Le robot est une partie du projet qui met en relation trois domaines : la mécanique, l’électronique, et l’informatique. Sa principale utilité est de pouvoir exécuter une action de déplacement de manière précise. Cette action est récupérée depuis un port série sur une Arduino. Les déplacements que peut faire le robot sont :

* Avancer/reculer en ligne droite d’une distance donné
* Tourner dans le sens horaire/anti-horaire d’un angle donné

Ces fonctions élémentaires permettent de pouvoir faire se déplacer le robot en n’importe quel point de l’espace, de manière précise.

-Matériel

Arduino :

Le robot embarque une carte Arduino Méga permettant de recevoir les ordres et de commander les moteurs. L’avantage de ce modèle d’Arduino est de pouvoir connecter plusieurs ports séries. Ceux-ci nous seront utiles pour pouvoir commander le lidar et transmettre les informations.

Moteurs :

Deux moteurs à roues codeuses permettent de réaliser les déplacements. Ces moteurs sont composés de lasers optiques qui envoient des impulsions lorsque le moteur réalise un certain nombre de tour. L’avantage de ces moteurs et de pouvoir récupérer des informations sur la rotation des moteurs par l’Arduino.

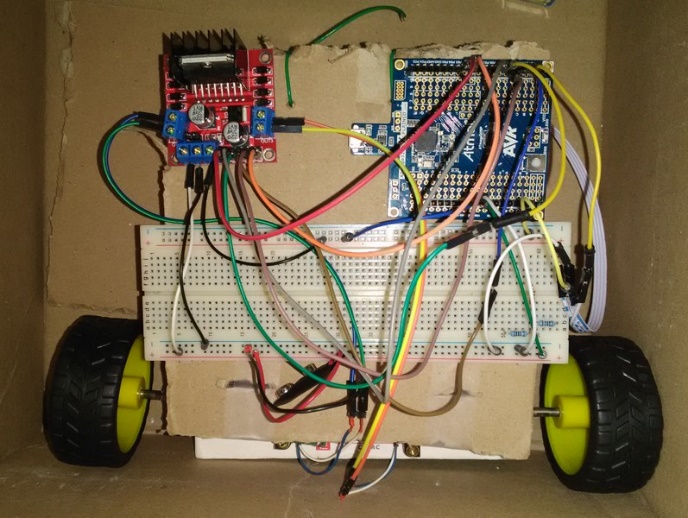
Pont en H L298N :

Ce circuit permet de faire le lien entre l’Arduino, les moteurs et l’alimentation des moteurs. Le L298N permet ainsi de récupérer grâce à 3 fils les ordres de l’Arduino. Les deux premiers servent à savoir dans quel sens faire tourner le moteur, en y appliquant des états haut ou bas (0 ou 5V), et le dernier permet de donner la vitesse demandé grâce à un signal PWM. Ce circuit est alimenté avec du 12V pour pouvoir ensuite alimenter les moteurs. Il renvoi finalement sur les deux fils du moteur le bon signal en 12V pour pouvoir faire tourner dans le sens horaire/anti-horaire le moteur, à la vitesse demandée.

Roues :

Le robot ne possède que trois roues. Les deux premières sont positionnées sur les moteurs, et la dernière est une roue folle qui est à l’avant du robot, et qui permet de faire rouler le robot dans toutes les directions. La rotation du robot s’effectue juste avec les deux moteurs : On actionne le premier moteur dans un sens, et le second dans le sens inverse.

2-Travail effectué

Premiers pas

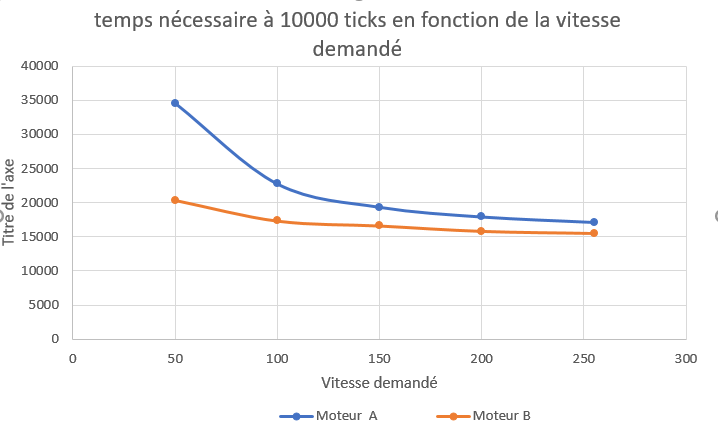
Notre premier but était de réaliser un robot sur un bout de carton pour bien comprendre comment actionner les moteurs avec l’Arduino.

Nous avons tout d’abord mis du temps à bien câbler et comprendre le fonctionnement du pont en H, car plusieurs câbles défectueux sont venus nous rendre la tâche plus difficile. Une fois les câbles changés, nous avons pu faire les premiers tests du robot.

Premier support du robot en carton

Nous avons ensuite changé de support en récupérant un châssis en plexiglass. Nous avons dû pour cela réaliser de pièces à la découpeuse laser pour adapter la fixation des moteurs. En réalisant les premiers tests sur le nouveau support nous nous sommes vite rendu compte que les moteurs ne tournent pas à la même vitesse, bien que l’on leurs envoi le même ordre. Cela a pour conséquence que le robot ne se déplace pas en ligne droite.

Coefficient de frottement

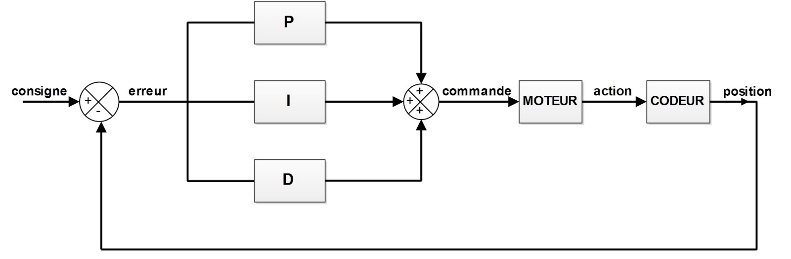
Nous nous sommes donc demandé si l’on pouvait trouver une relation linéaire entre la vitesse demandée, et la vitesse mesuré sur chaque moteur. Cela nous aurait permis d’appliquer un facteur sur la vitesse demandé sur un des moteurs, et ainsi avoir la même vitesse finale sur les deux moteurs. Nous avons donc réalisé un code permettant de calculer le temps nécessaire pour réaliser un certain nombre de tour. En exécutant ce code avec différents vitesse demandés sur les deux moteurs, nous avons eu comme résultat ces deux courbes :

Les courbes nous ont montrés que le frottement n’était pas du tout linéaire. Une solution simple et efficace n’était donc pas envisageable en appliquant un coefficient de proportionnalité.

Asservissement

Un asservissement en vitesse s’est donc imposé comme la solution à notre problème ?

« Mais qu’est-ce qu’un asservissement en vitesse ??? »

L’asservissement d’un moteur est une méthode permettant de réguler précisément la vitesse de rotation du moteur en fonctionnant en boucle. Cette boucle commence par envoyer un ordre au moteur, puis calcul la vitesse de rotation du moteur grâce à la roue codeuse, et réajuste l’ordre donné pour que la vitesse du moteur soit au plus proche de la valeur attendue. Cet algorithme est exécuté indéfiniment pour réajuster en temps réel la valeur de la vitesse.

Source : naytech.blogspot.com

La réalisation de cet asservissement qui s’exécute toutes les 10 milliseconde permet ainsi que les deux moteurs tournent à la même vitesse.

Notre robot peut grâce à cela se déplacer en ligne droite, et tourner sur lui-même.

Calcul de distance

Une variable de comptage est incrémentée à chaque impulsion lu sur les roues codeuses. Celle-ci nous permet de réaliser l’asservissement en vitesse, mis nous permet aussi de connaitre la distance effectuée par le robot. Le rapport de réduction étant de 1:80 et sachant que l’on récupère deux impulsion par tour, on sait que lorsque l’on reçoit une impulsion, la roue à parcourut 1/160ième de son périmètre. On peut donc en connaissant le périmètre de nos roues, aisément déduire le nombre d’impulsion nécessaire pour que le robot parcourt une distance donnée. Comme les moteurs ne peuvent pas être arrêter, et comportent une roue libre, nous avons fait attention à faire ralentir le robot lorsqu’il commence à se rapprocher de son point d’arrivé. Pour ce qui est de la rotation, le calcul est réalisé en connaissant l’entraxe de nos roues. Les roues possédant un léger décalage d’alignement, nous avons été contraints d’appliquer une correction sur la valeur mesurée sur l’entraxe des roues. Celle-ci ne s’appliquant que pour la rotation dans le sens horaire, le robot ne peut finalement exécuter que des tours dans ce sens, pour avoir une mesure précise de la rotation effectuée.

Problèmes rencontrés

Nous avons grillé nos premières roues codeuses en appliquant une tension trop forte dessus car la data sheet des roues codeuse était dans un premier temp introuvable, et nous avons donc branchés la roue codeuses selon la data sheet d’une roue codeuse semblable. Nous avons était obligé d’en récupérer de nouvelles.

Bilan

Finalement le robot se déplace bien tout droit d’une distance donnée et réalise bien l’angle demandé en tournant sur lui-même. Cette rotation ne se fait que dans le sens horaire mais cela n’est pas très pénalisant. Le robot est autonome grâce à la batterie lipo 12V qu’il embarque qui alimente les moteurs, l’Arduino et le pont en H.