Baccalauréat technologique
série STI2D

Session 2014

Projet n°2 : Robot Ramoneur

Épreuve de projet en enseignement spécifique à la spécialité

Contrat individuel n°5: LEDUC Quentin

Attestation de la prestation de nettoyage

Lycée de la Plaine de l'ain
Amberieu en Bugey
01500

Professeur : J.F.JOYON

Sommaire :

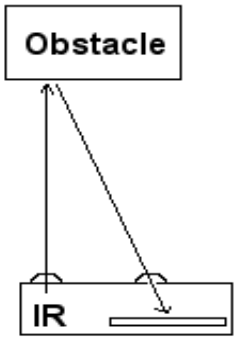
I- Définition du projet :
(présentation de groupe)

II-Solution permettant de mesurer des distances et de calculer la distance parcourue par le robot pendant le cycle de ramonage.

II-1- Solution permettant de mesurer des distances avec un dispositif de comptage et une résolution.

Dans le cahier des charges, ma fonction principale est de créer des données : de distances et de temps. Pour créer des donnés comme celles-ci, il faut que nous disposions d'un système capable de convertir des informations analogiques en informations numériques.

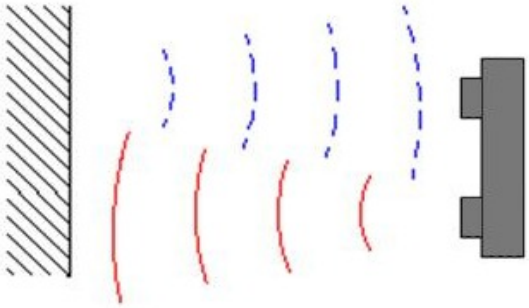
Pour alors mesurer des distances notre premier choix c'est tourné vers les capteurs de distances , qui sont utilisés dans de nombreux systèmes grâce à leurs efficacités. Pour les mesures de distances, deux capteurs différents peuvent être utilisés, tel que les capteurs infrarouges, qui sont constitués d'un récepteur qui détecte l'intensité lumineuse dans la gamme des lumières infrarouges et d'un émetteur de lumière infrarouge. Le capteur infrarouge est utilisé en capteur de distance en mesurant l'angle avec lequel le rayon réfléchi arrive sur le récepteur. En fonction de la distance entre l'émetteur et le récepteur, on peut en déduire la distance de l'obstacle...



... Ou bien les capteurs à ultrasons. Les capteurs ultrasons fonctionnent en mesurant le temps de retour d'une onde sonore inaudible par l'homme émise par le capteur. La vitesse du son étant à peu près stable, on en déduit la distance à l'obstacle.

Les capteurs ultrasons fournis ont souvent la forme d'une paire d'yeux car il y a deux parties essentielles : **l'émetteur** et **le récepteur**.

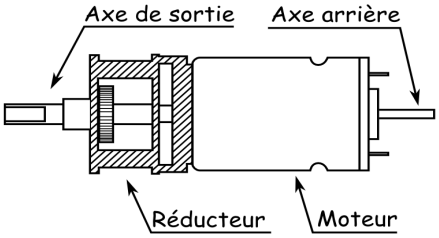
L'émetteur émet un son à une fréquence définie (généralement autour de 40 kHz) et le récepteur collecte le son répercuté par les obstacles. La distance aux objets est calculée par le temps mis par le son pour revenir au récepteur.



Mais ces capteurs ne sont pas appropriés pour l'endroit de l'utilisation du robot, la cheminée. La cheminée qui à l'intérieur est couverte de suie va, en effet, annuler les capacités du capteur infrarouge, car la suie va absorber les rayons émis par l'émetteur et donc ne transmettre aucune information. Pour le capteur à ultrason c'est l'espace qui est trop restreint car les ondes d'émission et de transmission vont rebondir contres les parois et donc brouiller le signal qui sera donc faussé.

Après en avoir déduit que les capteurs ne donneraient aucune solution concrète, nous nous sommes tournés vers un autre système qui réunirait deux solutions, une pour la mesure de distance et une autre pour le calcul de distance pendant un cycle de ramonage, ce système est celui de l'encodeur.

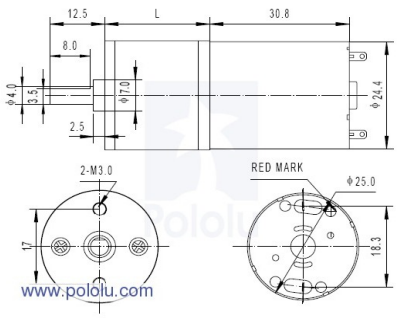
Un encodeur est un système qui permet d'incrémenter des états hauts à la suite. Nous allons installer ce système directement sur une roue du robot ramoneur, ce qui va nous permettre, avec un calcul de périmètre de la roue, de pouvoir définir de combien avance le robot sur un tour de roue et donc de mesurer et calculer la distance parcourue par le robot pendant le cycle de ramonage. Généralement les encodeurs rotatifs se fixent sur des moteurs avec un réducteur qui ce trouve entre l'axe de sortie et le moteur.



II-2- Test et validation.

Pour notre projet nous nous sommes procurés un encodeur avec moteur intégré CC Métal Pololu 25Dx54L avec réducteur 75:1 .

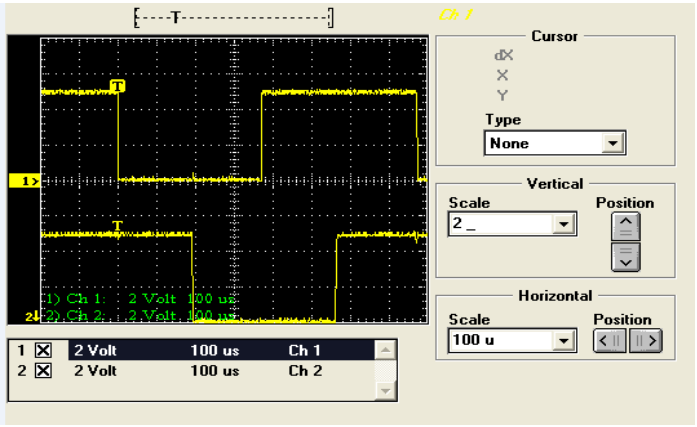
Couleur	Fonction
Rouge	Alimentation moteur (3-9 V)
Noir	Masse moteur
Vert	Masse encodeur
Bleu	Alimentation encodeur (3.5-20 V)
Jaune	Sortie encodeur A
Blanc	Sortie encodeur B



L'encodeur du Moteur CC Pololu mesure la rotation d'un disque magnétique situé à l'arrière de l'axe moteur. Cet encodeur à effet Hall offre une résolution de 48 pas par tour de l'axe moteur, ce qui après réduction correspond à une résolution de 3592 pas par tour de l'axe de sortie.

Mais qu'est-ce que l'effet Hall ? L'effet Hall a été découvert en 1879 par Edwin Herbert Hall et son principe est que lorsque un courant électrique traverse un matériau baignant dans un champ magnétique, il engendre une tension perpendiculaire à celui-ci, qui est appelé, tension Hall.

L'encodeur requiert une tension d'entrée comprise entre 3.5-20 V et consomme au maximum 10 mA. Les deux sorties A et B renvoient un signal carré entre 0 et la tension d'entrée avec un décalage de phase d'environ 90° . Chaque front montant et descendant des deux encodeurs indique un pas de révolution.
La fréquence des fronts indique donc la vitesse de rotation qui est de 3592 pas par tour de l'axe de sortie, et l'ordre des transitions permet de connaître la direction.



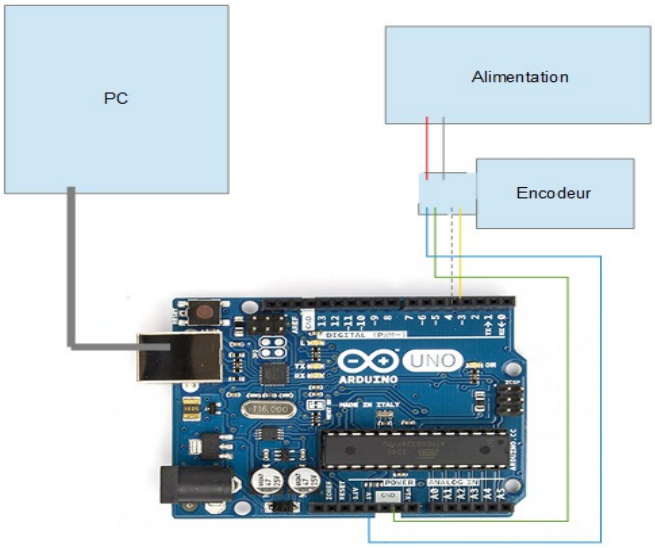
Pour la résolution de la mesure nous avons préféré une résolution au centimètre près pour avoir une marge précise.

II-3- Solution permettant de calculer la distance parcourue pendant le cycle de ramonage.

Pour ensuite faire le calcul de la distance parcourue pendant le cycle, nous allons utiliser un logiciel de programmation. En effet le logiciel Arduino va nous permettre, grâce à une carte Arduino UNO de créer un programme qui nous donnera les informations de distance parcourue.

Nous pouvons commencer à utiliser le logiciel Arduino avec la carte Arduino UNO, l'encodeur et une alimentation pour l'encodeur sur les bornes appropriées.

(Schéma) :



Création du programme via Arduino :

Tout d'abord nous devons faire apparaître les variables des impulsions des sorties A et B de l'encodeur :

```
// --- Affectation des broches A/B ---
int encodeur_pulseA = 3; // affectation de la sortie sur la broche 3
int encodeur_pulseB = 4; // affectation de la sortit sur la broche 4
```

Puis créer la variable de nombre d'impulsion enregistrée grâce aux variables précédentes :

```
int N_puls = 0;          // Variable représentant le nombre d'impulstion
```

Grâce à ces variable nous pouvons déjà lire le nombre d'impulsions de l'encodeur en mettant le nombre de bauds au maximum et sans aucun délai, autrement le programme prendrait en compte juste les informations entre chaque délai et les données seraient faussées.

```
Serial.begin(115200);
```

Grâce aux dernières informations nous pouvons faire apparaître la variable de distance : On sait aussi que une pulsation fait avancer le robot de 0,42cm.

```
N_distance = N_puls*0.42;
```

On connaît alors les informations de distance qui sont affichées dans le moniteur série...

```
Serial.print("Nombre d'impulstion = " );
Serial.println(N_puls);
Serial.print("Distance parcourue =");
Serial.println(N_distance);
```

...et que un tour de roue (11,9cm) est atteint en 28 pulsations et que 1 mètre est parcouru en 238 pulsations.

III- Solution permettant de mesurer un temps et de calculer la durée de cycle de ramonage.

III-1- Système permettant de mesurer un temps.

Mesurer un temps est l'une de mes fonctions principale dans mon cahier des charges. Pour mesurer le temps nous utiliserons un module RTC (real time clock), ce module se branche directement sur la carte Arduino via une carte Tinkerkit. Ce module est une horloge à temps réel (Real Time Clock), qui nous permet de connecter simplement la puce DS1307 à la carte arduino UNO, en utilisant le protocole I2C.

Le module RTC fonctionne avec l'interface I2C. Le protocole I2C est un bus de données qui a émergé de la «guerre des standards» lancée par les acteurs du monde électronique. Conçu par Phillips pour les applications de domotique et d'électronique domestique, il permet de relier facilement un microprocesseur et différents circuits, notamment ceux d'une télévision moderne: récepteur de la télécommande, réglages des amplificateurs basses fréquences, tuner, horloge, gestion de la prise péritel, etc.

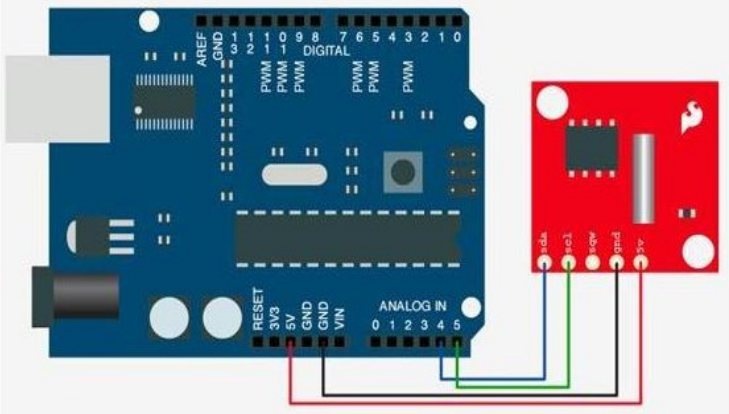
Il existe d'innombrables périphériques exploitant ce bus, il est même implémenté par logiciel dans n'importe quel microcontrôleur. Le poids de l'industrie de l'électronique grand public a permit des prix très bas grâce à ces nombreux composants.

Ce bus porte parfois le nom de TWI (Two Wire Interface) chez certains constructeurs.

III-2-Test et validation

Pour faire le calcul de la durée du cycle de ramonage, nous utiliserons un module RTC avec la carte de programmation Arduino UNO. Avec le logiciel Arduino, nous utiliserons la librairie DS1307, et la librairie Wire, qui permet de communiquer avec les composants qui utilisent le protocole I2C, cela va permettre le bon fonctionnement programme.

```
#include <Wire.h>
#include "DS1307.h"
```



Pour les données nous avons choisi une résolution pour une seconde.

Pour faire le calcul du temps du cycle de ramonage nous allons faire une soustraction dans le programme : temps actuel-temps initial. Ce qui nous donnera :

```
void calcul_SEC()
    {
   buttonState = digitalRead ( buttonPin ) ;
    if (buttonState == HIGH)
    {
      SEC= ((clock.hour)*36E2)+((clock.minute)*60)+ (clock.second);
      Serial.println(SEC);
      comptage++;
      if (comptage==1)
      {
        result1= SEC;
      }
      if (comptage==2)
      {
       result2 = SEC;
       resultfinal=(result2-result1);
       comptage=0;
```

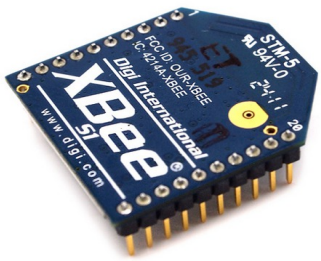
IV- Solution permettant d'envoyer et de mémoriser les informations de distances et de temps de ramonage.

IV-1- Système permettant d'envoyer des informations.

Pour permettre d'envoyer les informations de distance et de temps du cycle de ramonage, nous utiliserons le module Xbee. Le module Xbee est un module de la gamme Arduino, il permet de d'envoyer des informations à distance, sans fil.

Le module Arduino Xbee Shield permet à une carte Arduino Uno de communiquer sans fil en utilisant le protocole Zigbee. Il dispose d'une portée de 30 m en intérieur. La carte se connecte facilement sur les connecteurs des Uno prévus à cet effet.

ZigBee est un protocole de haut niveau permettant la communication de petites radios, à consommation réduite, basée sur la norme IEEE 802.15.4.



L'IEEE joue un rôle très important dans l'établissement de normes. Celle-ci est faite par la *IEEE Standards Association*. Elle assure la publication de ses propres normes et des autres textes rédigés par des membres de son organisation.

Le 802.15.4 est un protocole de communication défini par l'IEEE. Il est destiné aux réseaux sans fil de la famille des LR WPAN (Low Rate Wireless Personal Area Network) du fait de leur faible consommation, de leur faible portée et du faible débit des dispositifs utilisant ce protocole.

Pour un affichage des donnés reçues au pupitre, nous avions pensé à installer un afficheur écran LCD pour que les informations s'affichent au fur et à mesure, grâce à un autre programme sur Arduino.

IV-2-Système permettant de sauvegarder des informations.

Pour sauvegarder les informations qui ont été crées par les programmes de mesure de temps et de distance, nous devons d'abord stocker ces données dans la mémoire du microprocesseur Arduino UNO : l'EEPROM grâce à un programme auquel nous insérerons la librairie EEPROM :

```
#include <EEPROM.h>
```

Ensuite nous enverrons ces informations via la puce Xbee au pupitre.

V-Conclusion.

A ce stade du projet, les solutions de mesure de distance, de calcul de distance parcourue sont opérationnelles, il ne reste plus qu'à les insérer dans le programme principal du robot. Nous devons encore terminer de programmer les solutions d'envois, et de stockage des données. Nous ne disposerons pas d'afficheur écran LCD pour lire les données donc nous devrons lire les données sur un autre système.

VI- Annexe.

RTC (real time clock) : document technique

Fonctionnalités :

- Horloge temps réel : Année, mois, jour, heure, minute, seconde. Compensation des années bissextiles.
- Stockage NVRAM 56 byte
- Interface Série (I2C)
- Alimentation DC 5V
- Sauvegarde automatique (batterie Lithium 3V)
- Consommation très faible (500nA en mode battery backup)

Encodeur : document technique

- Diamètre: 25 mm
- Longeur: 66 mm
- Poids: 100 g
- Diamètre de l'axe: 4 mm

- Ratio de réduction: 74.83:1
- Vitesse à vide (pour 6V): 130 t/min
- Courant à vide (pour 6V): 450 mA
- Courant de blocage(pour 6V): 6000 mA
- Couple de blocage(pour 6V): 9.36 kg/m
- Longueur du câble: 20.3 cm
- Connectique: connecteur femelle 1x6 2.54 mm
- Tension nominale: 6 V (3-9 V)
- Encodeur:
    - Type: Encodeurs à effet Hall 2 canaux
    - Résolution: 48 pas/t interne, 3592 pas/t sortie
    - Tension nominale: 3.5-20 V
    - Courant maximum: 10 mA

Module Xbee : Document technique

- RF 2,4 GHz : portée allant jusqu'à 120 m (sans obstruction)
- Jusqu'à 250 kbps
- Interface série (UART CMOS 3,3V)
- Utilisé pour l'émission et la réception
- Petite taille, bas coût et faible consommation d'énergie (1 mW)

Arduino UNO: Document technique

alimentation: via port USB ou 7 à 12 V sur connecteur alim.

- microprocesseur: ATMega328
- mémoire flash: 32 kB
- mémoire SRAM: 2 kB
- mémoire EEPROM: 1 kB
- 14 broches d'E/S dont 6 PWM
- 6 entrées analogiques 10 bits
- intensité par E/S: 40 mA
- cadencement: 16 MHz
- bus série, I2C et SPI
- gestion des interruptions
- fiche USB B
- dimensions: 74 x 53 x 15 mm