

COMPTE RENDU

SEANCE 5

Date : 31/01/2022
Heure : 15h à 18h
BOURRY Amir Groupe 1

Objectifs

Créer la communication Bluetooth entre un téléphone portable et les moteurs.

Préparer le capteur de distance communiquant avec les moteurs.

Assembler le cylindre à LED.

Barre LED

Le cylindre supportant la bande LED a été réalisé avec une imprimante 3D, celui-ci tournera sur lui-même lors de l'actionnement du Rover.



Communication Bluetooth

Nous utilisons une puce communicante HC-06 pour le Bluetooth. Elle fonctionne avec la bibliothèque SoftwareSerial. Nous avons configuré la puce en lui donnant un identifiant et un mot de passe pour assurer la connexion avec le téléphone.

```
#include<SoftwareSerial.h>
#define RX 11
#define TX 3
SoftwareSerial Blue (RX, TX);
char Data;
int PWM;
```

On dispose de deux sorties, TX qui transmet les informations et RX qui les reçoit.

```
if (Blue.available()) {  
  Data = Blue.read();  
  Serial.println("premier IF");  
  if (Data=='A'){  
    PWM = Blue.parseInt();  
    Serial.println(PWM);  
    analogWrite(ENA, PWM);  
    analogWrite(ENB, PWM);  
  }  
}
```

On reçoit les messages Bluetooth. On vérifie qu'ils portent le message «'A » qui est l'identifiant du message bluetooth qu'envoie le téléphone, si c'est le cas on note le premier entier qui suit et on allume les moteurs avec comme valeur d'entrée la valeur de la barre de contrôle sur le téléphone pour moduler la puissance des moteurs via le Bluetooth.

Le téléphone contrôle la puce HC-06 grâce à l'application « Bluetooth Electronics ».

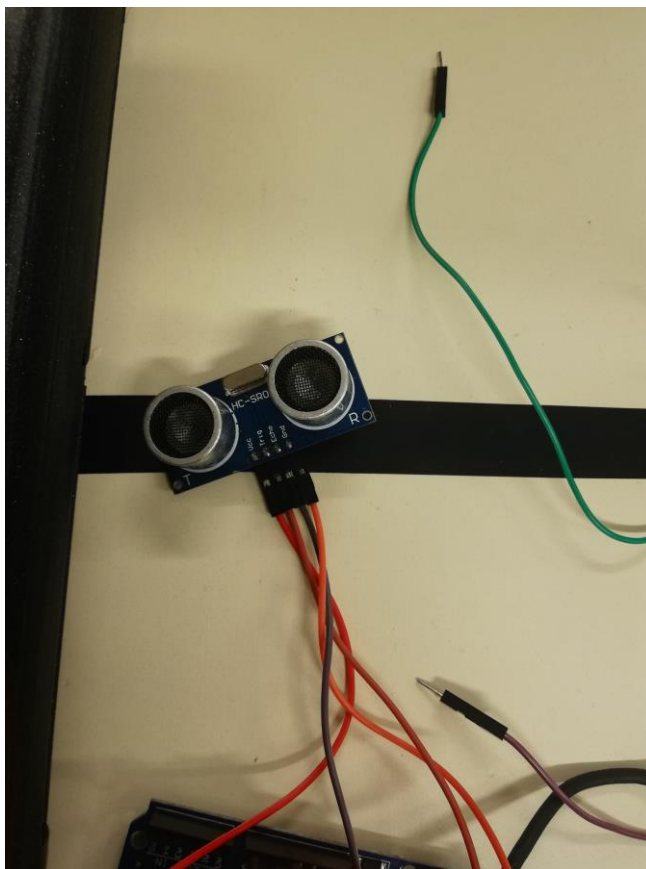
Mesure de distance et contrôle des moteurs

Afin de configurer le mode automatique de déplacement du rover, nous utilisons une puce de mesure de distance. Elle envoie une onde électromagnétique et mesure le temps que celle-ci met à revenir. Elle porte donc un émetteur et un récepteur. Si la distance entre le rover et l'objet qui lui fait face est inférieure à 10 centimètres, le rover s'arrête, sinon, il continue en pleine puissance.

```
digitalWrite(trig, HIGH);  
delayMicroseconds(10);  
digitalWrite(trig, LOW);  
Serial.print("la distance est : ");  
Serial.println(sonar.ping_cm());  
if (sonar.ping_cm()<10){  
  analogWrite(ENA, 0);  
  analogWrite(ENB, 0);  
}  
if (sonar.ping_cm()>10){  
  analogWrite(ENA, 255);  
  analogWrite(ENB, 255);  
}
```

On envoie une onde pendant 10 millisecondes, on s'arrête, puis on mesure la distance avec la bibliothèque NewPing qui contient la fonction ping_cm(). On allume le moteur si la distance est supérieure à 10 centimètres, on l'éteint sinon.

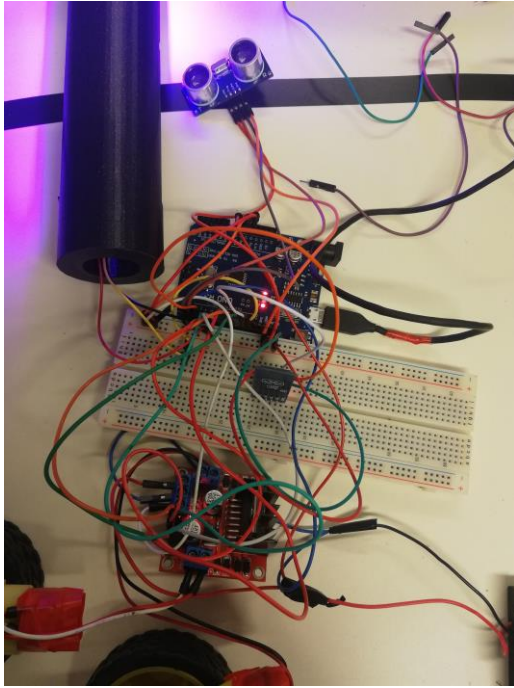
Il reste donc à permettre au Rover de se repérer selon les deux axes X et Y de la réalité afin de lui permettre de se déplacer dans un « monde » en 2D.



Problèmes rencontrés

- Problème de connexion au Bluetooth qui a été réglé en inversant TX et RX (car il ne s'agissait pas de la carte mais de la puce HC-06)

Montage actuel



Il faudra améliorer le câble management en rassemblant les câbles allant vers les mêmes endroits.

Conclusion

Cette séance a été celle qui a vu le plus d'avancements sur notre projet, il faut maintenant finir l'assemblage du rover en y incrustant les circuits et plaque d'essai. Il faut aussi paramétrer le contrôle bluetooth de nos moteurs ainsi que le repérage dans l'espace du Rover en mode automatique et rajouter un écran pour le choix du mode (automatique ou manuel). Nous sommes plus enthousiastes sur notre projet maintenant que celui-ci prend de plus en plus forme.

A Valbonne, le 01/02/2022.

BOURRY Amir PeiP G1.