

# Projet Arduino

## Voiture robotisée & contrôlée par Bluetooth

Bernard Anwar (en collaboration avec Chillat Quentin)

### Rapport Semaine 5 (le 25/01/2024)

#### Matériels utilisés

- Pour le gant
  - Un connecteur (adaptateur)
  - Pile 6LR61 Pour le gant
- Pour la voiture
  - Un capteur à distance ultrasons HC-SR04
  - Jeux de cordons mâles-femelles
  - Support pour fixer le module HC-SR04

#### Schéma du câblage et Assemblage

Le connecteur du gant est branché au VIN (fil rouge sur la borne positive) et au GND (fil noir sur la borne négative)

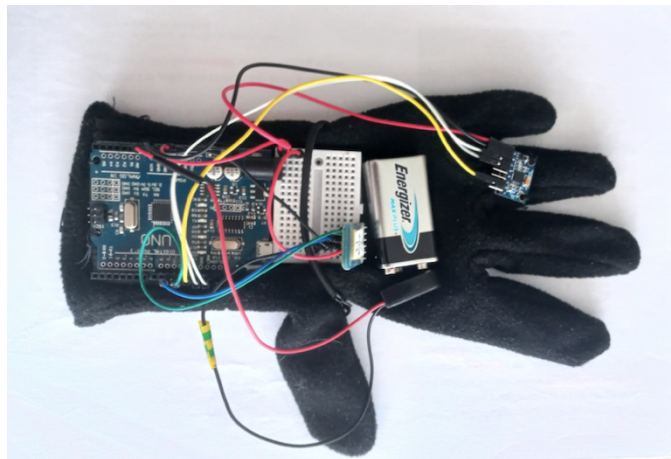
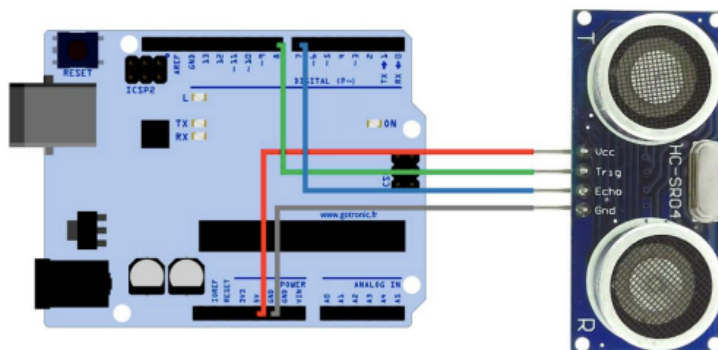
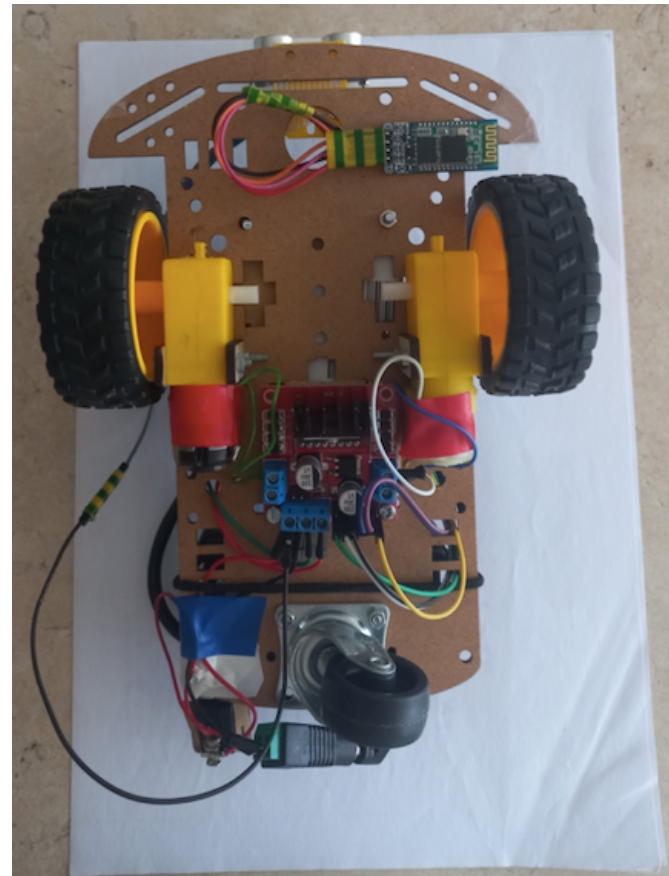
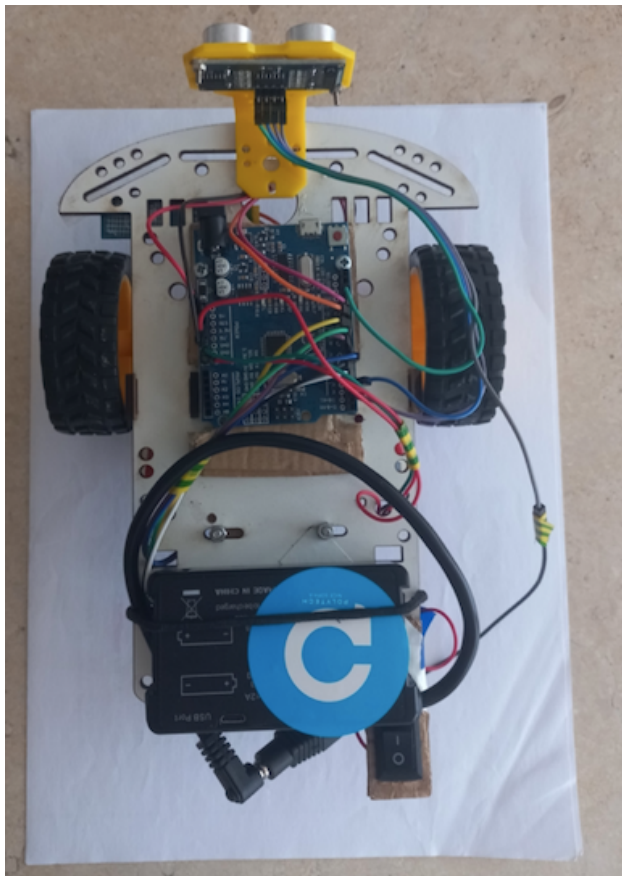


Tableau de correspondance carte Arduino et module HC-SR04:

Carte Arduino	HC-SR04
5V	Vcc
GND	Gnd
3	Echo
2	Trig





## Programme de communication récepteur (HC-06) pour prendre en compte le module HC-SR04

### Codage côté récepteur (HC-06)

```
#include <SoftwareSerial.h>
SoftwareSerial bluetoothSerial(13, 12); // RX, TX (Connectez RX du HC-06 à 10 et TX à 11)

int16_t accelerometer_data[2] = {0};
String receivedString = ""; // Chaîne pour stocker les données reçues

const int trig=2;
const int echo=3;

float distance;
float lecture_echo;

int ENA=9; // Moteur DROIT
int IN1=4;
int IN2=5;

int ENB=10; //Moteur GAUCHE
int IN3=6;
int IN4=7;

void setup() {
  Serial.begin(38400);
  bluetoothSerial.begin(38400); // Initialisez le port série pour la communication Bluetooth

  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);

  analogWrite(ENA, 255);
  analogWrite(ENB, 255);
}

void loop() {

  pinMode(trig, OUTPUT); //trig sur sortie 10
  pinMode(echo, INPUT); //echo sur entrée 10

  digitalWrite(trig, HIGH); //on allume le signal
  delayMicroseconds(10);
  digitalWrite(trig, LOW); //on eteint le signal
  lecture_echo=pulseIn(echo, HIGH); //on enregistre le temps d'allumage du signal
  distance=lecture_echo*0.017; //on converti le temps en distance
  // distance=(temps*2)/0.034=temps*0.017
```

```

// Vérifier s'il y a des données disponibles sur Bluetooth
while (bluetoothSerial.available()) {
    char receivedChar = bluetoothSerial.read();

    // Si une virgule ou un saut de ligne est reçu, cela signifie la fin d'une valeur
    if (receivedChar == ',' || receivedChar == '\n') {
        // Convertir la chaîne reçue en entier
        int receivedValue = receivedString.toInt();

        // Mettre à jour les valeurs du tableau d'accéléromètre à partir des données reçues
        for (int i = 0; i < 2; i++) {
            if (accelerometer_data[i] == 0) {
                accelerometer_data[i] = receivedValue;
                break;
            }
        }

        // Réinitialisez la chaîne pour la prochaine valeur
        receivedString = "";

        // Si le tableau est complètement construit, afficher chaque valeur
        if (accelerometer_data[0] != 0 && accelerometer_data[1] != 0 ) {
            Serial.print("Tableau d'accéléromètre complet : ");
            Serial.print(accelerometer_data[0]);
            Serial.print(", ");
            Serial.println(accelerometer_data[1]);
            if(distance>5){
                if(accelerometer_data[1]<-10000){ //avancer
                    digitalWrite(IN1,HIGH);
                    digitalWrite(IN2,LOW);
                    digitalWrite(IN3,LOW);
                    digitalWrite(IN4,HIGH);
                }
                else if(accelerometer_data[1]>10000){//reculer
                    digitalWrite(IN1,LOW);
                    digitalWrite(IN2,HIGH);
                    digitalWrite(IN3,HIGH);
                    digitalWrite(IN4,LOW);
                }

                else if(accelerometer_data[0]<-10000){//tourner a gauche
                    digitalWrite(IN1,HIGH);
                    digitalWrite(IN2,LOW);
                    digitalWrite(IN3,HIGH);
                    digitalWrite(IN4,LOW);
                }

                else if(accelerometer_data[0]>10000){//tourner a droite
                    digitalWrite(IN1,LOW);
                    digitalWrite(IN2,HIGH);
                    digitalWrite(IN3,LOW);
                    digitalWrite(IN4,HIGH);
                }
                else{ //static
                    digitalWrite(IN1,LOW);
                    digitalWrite(IN2,LOW);
                    digitalWrite(IN3,LOW);
                    digitalWrite(IN4,LOW);
                }

                // Réinitialisez le tableau pour la prochaine série de valeurs
                memset(accelerometer_data, 0, sizeof(accelerometer_data));
            }
        }
        else {
            if(accelerometer_data[1]>10000){//reculer
                digitalWrite(IN1,LOW);
                digitalWrite(IN2,HIGH);
                digitalWrite(IN3,HIGH);
                digitalWrite(IN4,LOW);
            }
            else{ //static
                digitalWrite(IN1,LOW);
                digitalWrite(IN2,LOW);
                digitalWrite(IN3,LOW);
                digitalWrite(IN4,LOW);
            }

            // Réinitialisez le tableau pour la prochaine série de valeurs
            memset(accelerometer_data, 0, sizeof(accelerometer_data));
        }
    }
    // Ajoutez le caractère à la chaîne en cours de réception
    receivedString += receivedChar;
}
}

```

## **Tâches effectuées :**

Nous avons perfectionné l'exécution des instructions par la voiture.

Nous avons ajouté une pile au gant, via un connecteur, pour avoir plus de liberté de mouvement.

Nous avons ajouté un module HC-SR04 pour détecter les obstacles et les éviter.

En cas d'obstacle, la voiture s'arrête et ne peut que reculer.

Nous avons modifié le programme de communication Bluetooth HC-06 (de la voiture) pour prendre en considération le module HC-SR04. Nous avons modifié le programme pour que la trajectoire de la voiture soit la plus rectiligne possible.

## **Tests effectués :**

Nous avons testé l'autonomie du gant (sans le branchement à l'ordinateur).

Nous avons testé l'orientation des roues selon l'inclinaison du poignet muni du gant.

Nous avons testé les trajectoires de la voiture.

Lorsque le poignet est baissé, la voiture avance.

Lorsque le poignet est levé, la voiture recule.

Lorsque le poignet est incliné à droite, la voiture va à droite.

Lorsque le poignet est incliné à gauche, la voiture va à gauche.

La voiture s'arrête lorsqu'elle rencontre un obstacle (dans ce cas elle ne peut que reculer).

## **Réalisation à prévoir pour la semaine 6.**

Attendre la réception de la commande (une carte PCB double face, fil pour PCB, headers, entretoises, un interrupteur pour PCB, une coque) pour professionnaliser notre projet et mettre une coque sur la voiture.