

## **Compte-Rendu n°17 Projet**

### **I-Débugage du moteur avant gauche qui ne fonctionnait pas**

Suite à de nombreux tests réalisés (vérification de code/ changement de pin en cas d'un dysfonctionnement de l'arduino mega....) suite à une panne du moteur de la chenille avant gauche du robot, il s'avérait que le problème venait de fils de connexions défectueux pour le pin 1 et 2 du moteur avant gauche. Ce problème a été résolu et le moteur tourne

### **II-Fin du travail sur le Bluetooth pour le contrôle de la motorisation.**

A partir de l'application Arduino Bluetooth Control, j'ai réussi à pouvoir contrôler mon robot à partir de mon téléphone. Un changement d'application a résolu le problème de non détection du robot. En effet, l'application plus connue "Bluetooth electronics" n'était pas compatible avec mon téléphone. Avec la nouvelle application Arduino Bluetooth Control, le robot au nom de "militech" apparaissait alors dans la liste des appareils bluetooth.

#### **Écriture du code Arduino:**

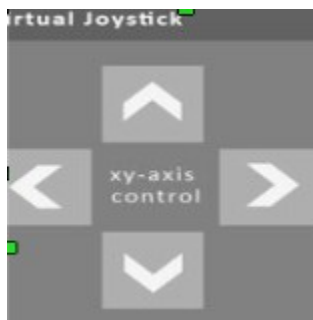
Les instructions que l'on peut actuellement donner au robot à partir du téléphone sont:

- avancer
- reculer
- tourner a gauche
- tourner à droite

**Explication du code** :Le téléphone est connecté au composant Bluetooth du robot et envoie des informations, notamment des lettres.

R pour Droite, L pour gauche , F pour avancer et B pour reculer. Il suffit alors d'assigner ces lettres à des touches virtuelles paramétrées sur l'application du téléphone : c'est un joystick virtuel. En appuyant sur ces touches, les lettres assignées à ces mêmes touches sont envoyées par l'application et réceptionnées par le module Bluetooth puis par l'arduino méga et ensuite traduites en instructions (avancer/reculer...).

Par exemple, en appuyant sur la flèche du haut : l'application envoie un F à la carte Arduino, qui est traduit en l'avancement du robot vers l'avant.



*Joystick Virtuel*

Voici ci- joint une capture d'écran du code permettant la communication Bluetooth entre le Robot et le téléphone.

```
39 void loop() {
40   while (BlueT.available()) {
41     Serial.print(char(BlueT.read())); } // permet la réception de lettres
42   while (Serial.available()) {
43     BlueT.write(char(Serial.read())); }
44
45   if (BlueT.available()){
46     Data=char(BlueT.read());
47     if (Data=='F') {
48       digitalWrite(pin1AvantDroit,HIGH);
49       digitalWrite(pin2AvantDroit,LOW);
50       digitalWrite(pin1ArriereDroit,HIGH);
51       digitalWrite(pin2ArriereDroit,LOW);
52       digitalWrite(pin1AvantGauche,HIGH); //le même sens et le robot avance tout droit
53       digitalWrite(pin2AvantGauche,LOW);
54       digitalWrite(pin1ArriereGauche,HIGH);
55       digitalWrite(pin2ArriereGauche,LOW);
56
57       digitalWrite(moteurArriereDroit,HIGH);
58       digitalWrite(moteurArriereGauche,HIGH); //activation des quatres moteurs
59       digitalWrite(moteurAvantDroit,HIGH);
60       digitalWrite(moteurAvantGauche,HIGH);
61       delay(500);
62     }
63     else if (Data=='B') {
64       digitalWrite(pin2AvantDroit,HIGH);
65       digitalWrite(pin1AvantDroit,LOW);
66       digitalWrite(pin2ArriereDroit,HIGH);
67       digitalWrite(pin1ArriereDroit,LOW);
68       digitalWrite(pin2AvantGauche,HIGH); //le robot avance tout droit
69       digitalWrite(pin1AvantGauche,LOW);
70       digitalWrite(pin2ArriereGauche,HIGH);
71       digitalWrite(pin1ArriereGauche,LOW);
72
73       digitalWrite(moteurArriereDroit,HIGH);
74       digitalWrite(moteurArriereGauche,HIGH); //activation des quatres moteurs
75       digitalWrite(moteurAvantDroit,HIGH);
76       digitalWrite(moteurAvantGauche,HIGH);
77       delay(500);
78     }
79     else if (Data=='R') {
80       digitalWrite(pin1AvantDroit,LOW);
81       digitalWrite(pin2AvantDroit,HIGH);
82       digitalWrite(pin1ArriereDroit,LOW);
83       digitalWrite(pin2ArriereDroit,HIGH); //tourne a droite
84       digitalWrite(pin1AvantGauche,HIGH);
85       digitalWrite(pin2AvantGauche,LOW);
86       digitalWrite(pin1ArriereGauche,HIGH);
87       digitalWrite(pin2ArriereGauche,LOW);
88
89       digitalWrite(moteurArriereDroit,HIGH);
90       digitalWrite(moteurArriereGauche,HIGH); //activation des quatres moteurs
91       digitalWrite(moteurAvantDroit,HIGH);
92       digitalWrite(moteurAvantGauche,HIGH);
93     }
94   }
```

```

96 | | else if (Data=='L') {
97 | |   digitalWrite(pin1AvantDroit,HIGH);
98 | |   digitalWrite(pin2AvantDroit,LOW);
99 | |   digitalWrite(pin1ArriereDroit,HIGH);           //tourne a gauche
100 | |   digitalWrite(pin2ArriereDroit,LOW);
101 | |   digitalWrite(pin1AvantGauche,LOW);
102 | |   digitalWrite(pin2AvantGauche,HIGH);
103 | |   digitalWrite(pin1ArriereGauche,LOW);
104 | |   digitalWrite(pin2ArriereGauche,HIGH);
105 | |
106 | |   digitalWrite(moteurArriereDroit,HIGH);
107 | |   digitalWrite(moteurArriereGauche,HIGH);
108 | |   digitalWrite(moteurAvantDroit,HIGH);           //activation des quatres moteurs
109 | |   digitalWrite(moteurAvantGauche,HIGH);
110 | |   delay(500);
111 | | }
112 | |   Serial.println(Data);
113 | | }
114 | |
115 | }

```

## II- complétion du code de détection d'obstacle

Certaines distances ont été encore une fois modifiées dans le code de détection d'obstacle :  
 Le robot se heurtait encore parfois à des murs : il ne tournait pas assez tôt. Suite à plusieurs tests, j'ai alors pris la décision d'augmenter la distance à partir de laquelle notre robot allait tourner :

distance de détection d'obstacle des capteurs avant : passée de 40 à 60 cm  
 distance de détection d'obstacle des capteurs latéraux et diagonaux passée de 30 à 45 cm.

Le robot fonctionne alors parfaitement lorsqu'il rencontre des murs ou des objets larges. Toutefois, il peut lui arriver de ne pas détecter certains obstacles fins, comme des pieds de chaises.

Solution envisagée dans le futur: mettre un LIDAR pour faire une synergie de capteurs et améliorer la détection d'obstacle.