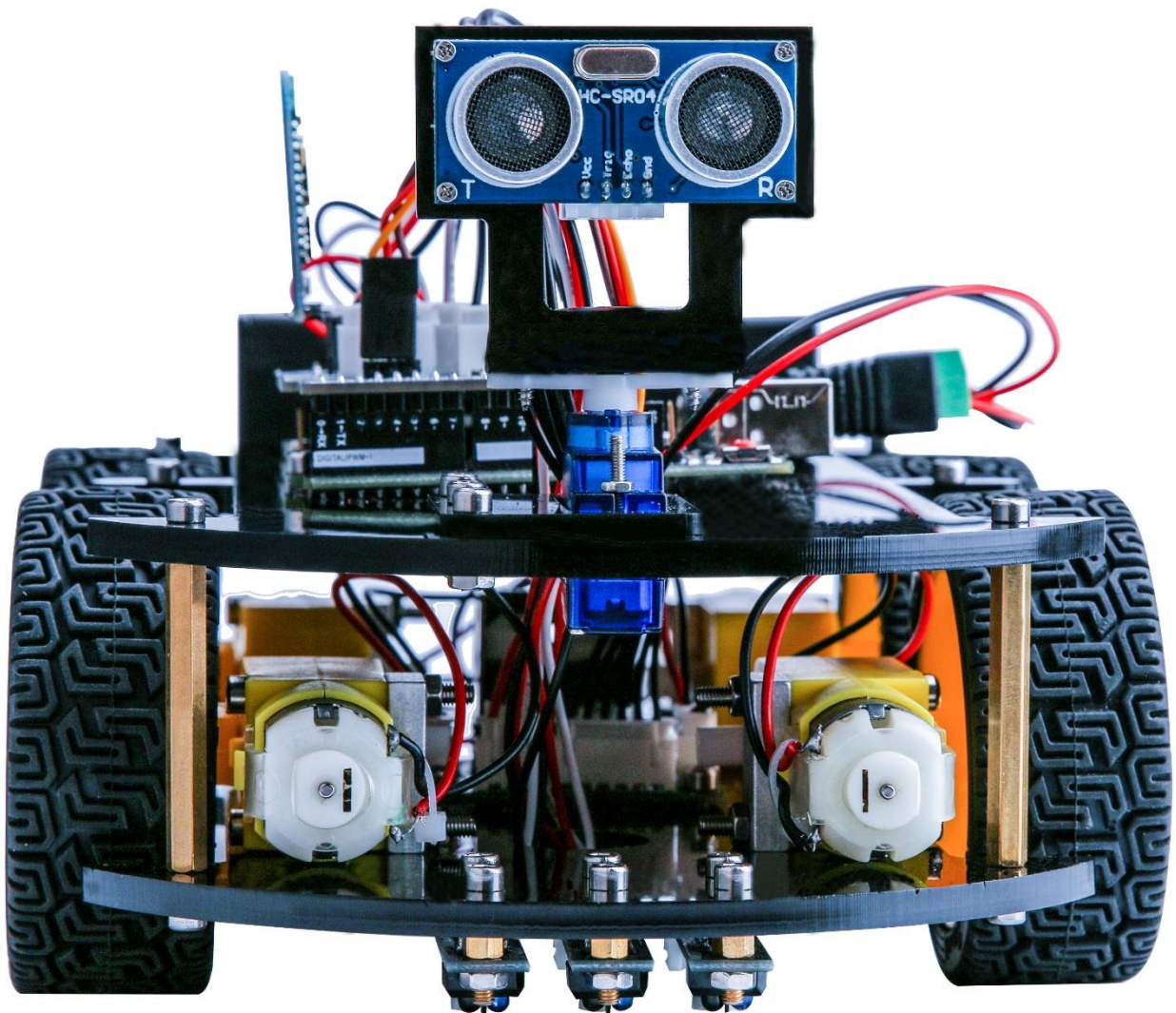
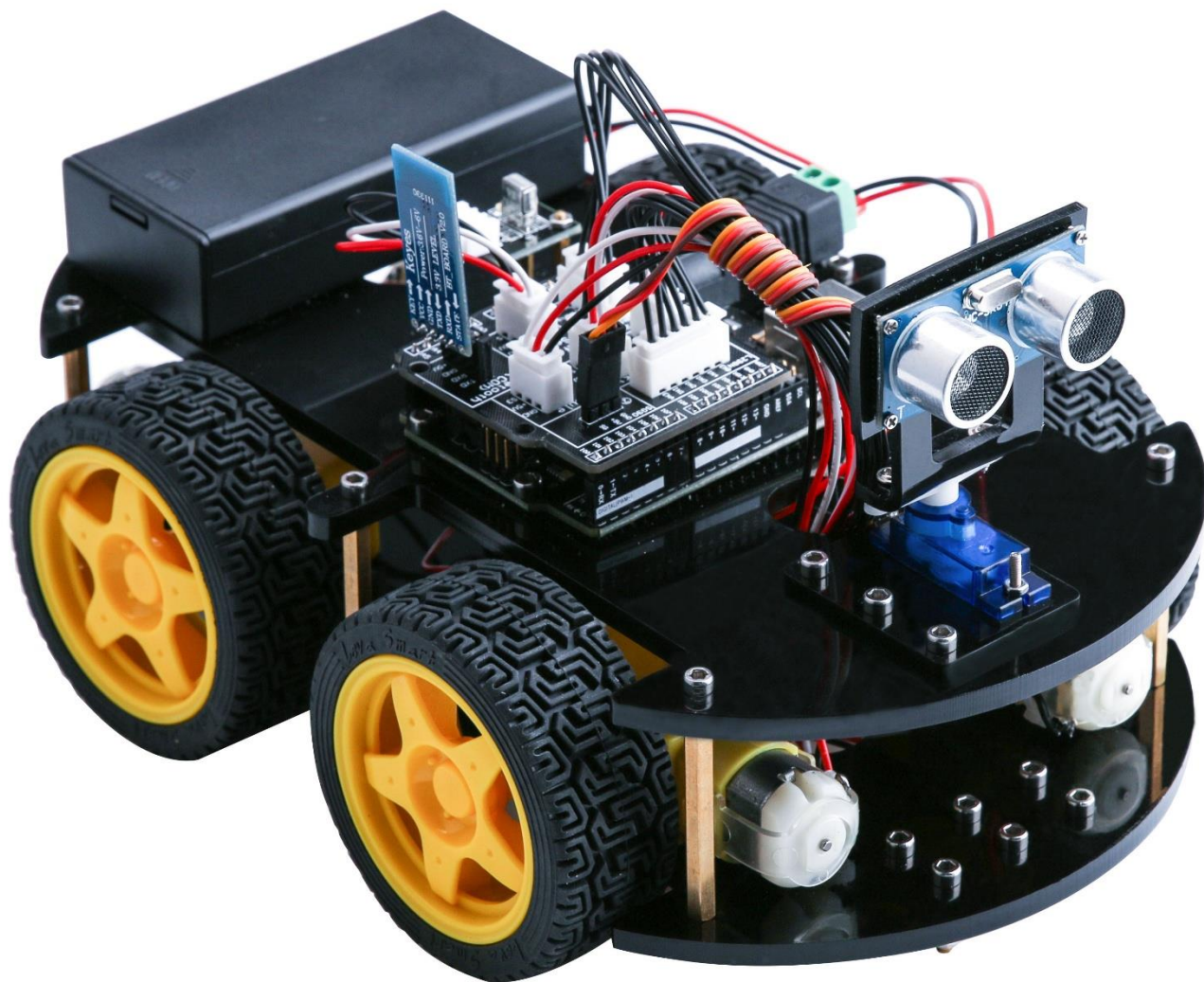


Leçon 1 – Mettre le Robot en mouvement

(Make the Car Move)





Points clés de la leçon

Pour cette première leçon, nous allons commencer par une petite présentation du robot en se focalisant sur les pièces qui le composent. Puis nous attaquerons par la présentation de la manière la plus simple de permettre au robot de se déplacer.

Sommaire:

- Survol des différentes parties du robot
- Utilisation des programmes de base

Préparatifs:

- Le robot
- Un câble USB (fourni)

I . Introduction

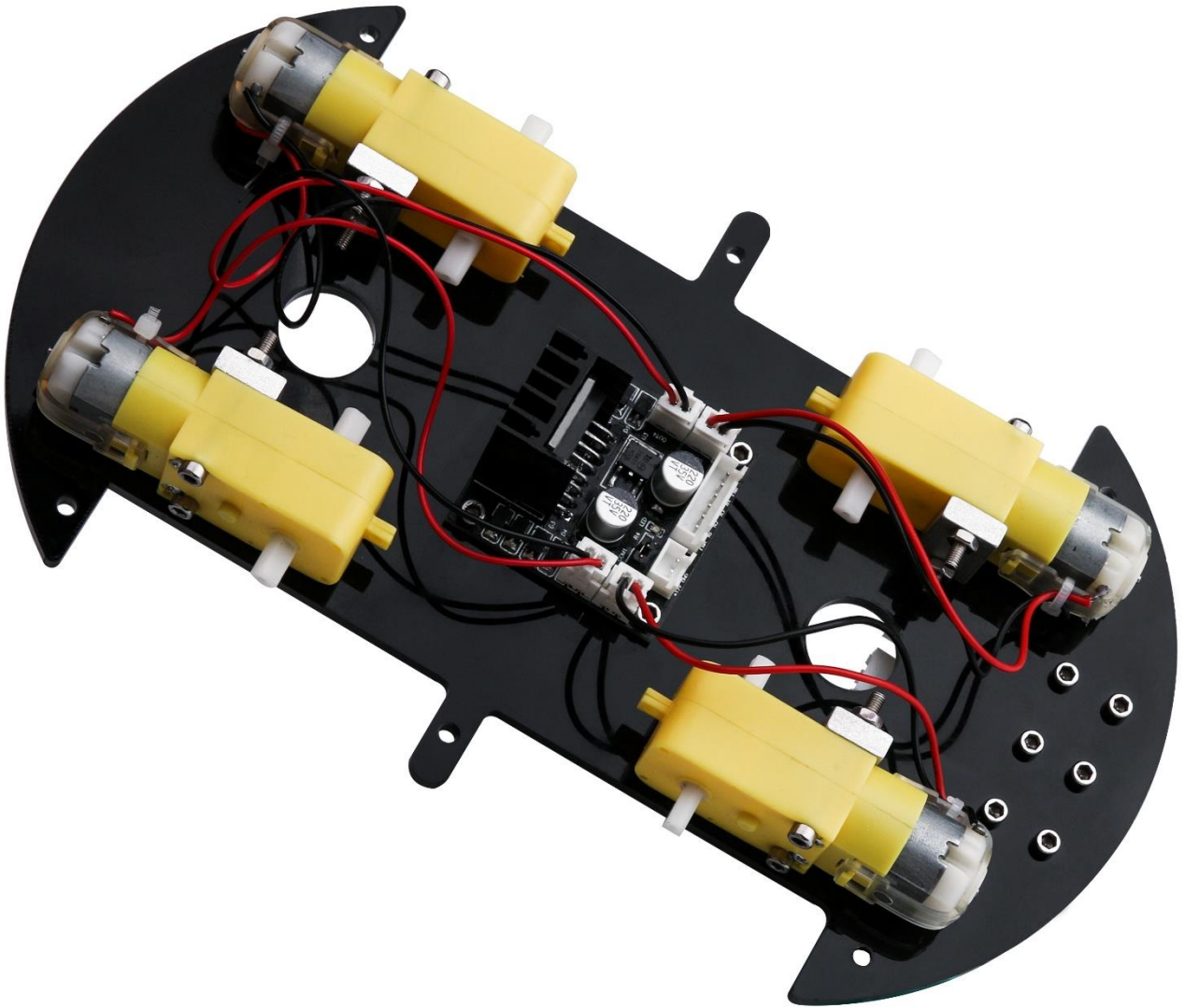
Ce kit permet d'assembler un véhicule extrêmement flexible, conçu pour les domaines de l'éducation, du divertissement, de la compétition.

Vous allez pouvoir expérimenter des phénomènes physiques et technologiques comme l'infrarouge, les ultrasons, le Bluetooth dans le but de faire de la commande à distance ou du déplacement automatique en évitement d'obstacle ou suivi de parcours.

Il vous sera aussi possible d'ajouter d'autres fonctions en utilisant des capteurs disponibles dans nos "kits de démarrage" : buzzers, leds, sounds sensors etc... et de doter votre robot de fonctionnalités inédites.

Commençons par une présentation des différentes parties du robot, qui va nous en sommes sûrs, vous tenir compagnie un long moment.

Les moteurs / la carte de contrôle moteurs:

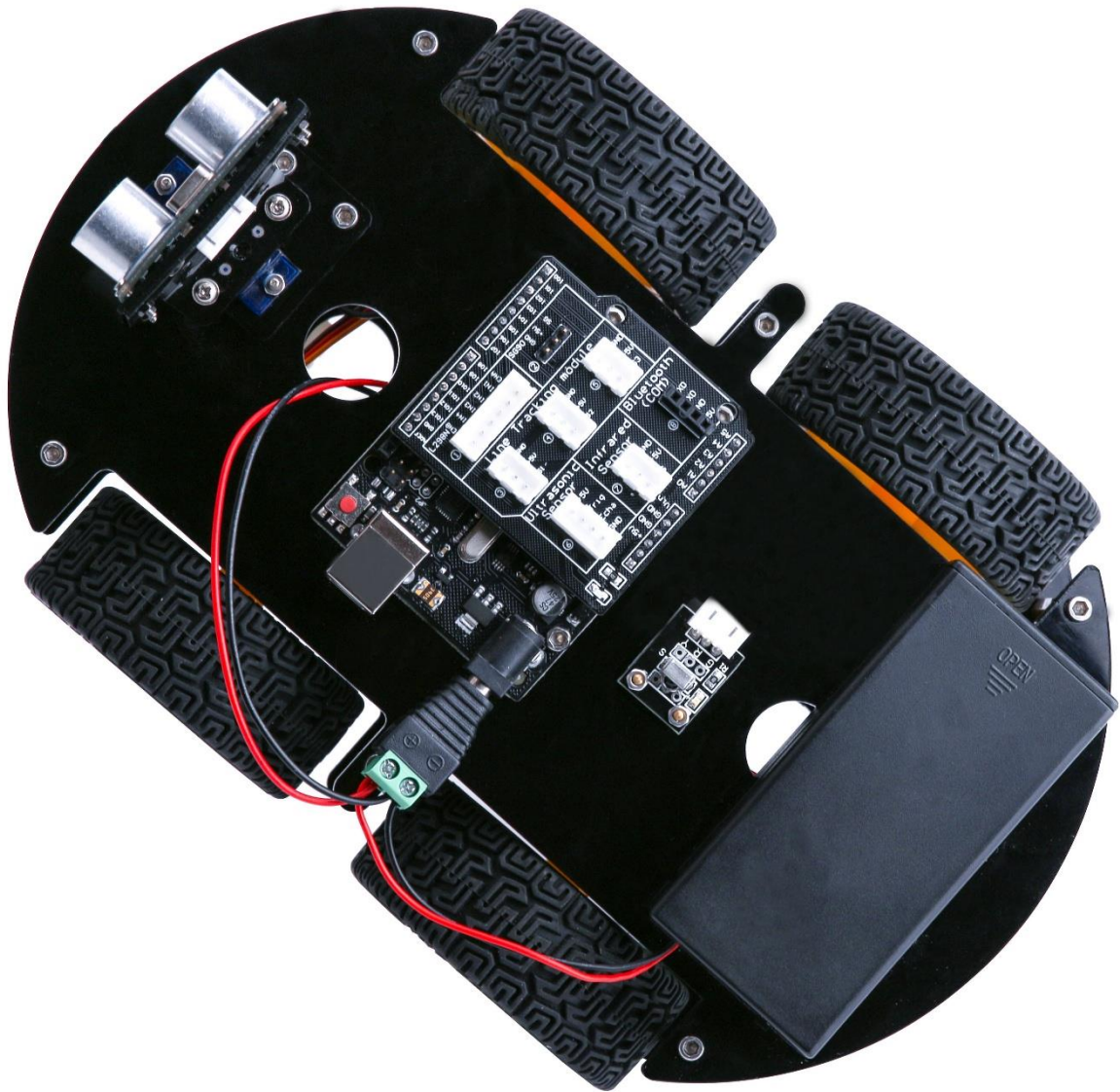


Les capteurs « Line Tracking »



Les capteurs infrarouges pour le "Line Tracking"

Le Module Ultrasons monté sur un servomoteur / La carte Elegoo Uno et la carte Extension V5 / Le récepteur IR / Le bloc Batteries



Le module ultrasons est monté sur un servomoteur / La carte Elegoo Uno et la carte Extension V5 / Le récepteur IR / Le bloc Batteries

En détails:

1. Bloc batterie avec l'interrupteur Marche/Arrêt du robot
2. Les moteurs et roues
3. Les platines acryliques (x2)
4. L298N carte de contrôle pour le pilotage des moteurs
5. Carte de contrôle Elegoo UNO : le cerveau
6. Carte d'extension V5: permet de connecter aisément tous les capteurs
7. Servomoteur SG90 : permet d'orienter avec précision le module Ultrasons
8. Module Ultrasons : permet de faire des mesures de distances
9. Capteurs "Line tracking module": utilisés pour le suivi de lignes noires
10. Récepteur infrarouge et sa télécommande pour un contrôle à distance
11. Module Bluetooth: permet un contrôle du robot avec les appareils mobiles de type smartphones, tablettes.


II. Déverser le code

Dans chaque leçon, vous allez devoir déverser des codes exemples permettant de vous familiariser avec les fonctionnalités du robot.

Pour ce faire, branchez le robot avec le câble USB (fourni) à un ordinateur dans lequel vous avez inséré le CD (fourni) contenant les codes exemples.



Ouvrez le fichier "AUTO_GO_\AUTO_GO_.ino" et déversez le sur la carte Elegoo UNO de votre robot

| J深度U盘 (G:) > 工作项目 > 4WD car tutorial and code > code > AUTO_GO_ | | |
|--|----------------|----------------|
| 名称 | 修改日期 | 类型 |
|  AUTO_GO_.ino | 2016/1/26 1:57 | Arduino Source |

Débranchez le câble USB, posez le robot au sol et mettez-le en marche. Vous observez le robot se mettre en mouvement.

III. Principes

Comment fonctionne la carte L298N

La carte L298N est branchée de la manière suivante

L298N GND est reliée à la masse de la batterie du robot;

L298N VCC est reliée au + de la batterie du robot;

La carte Elegoo UNO est aussi connectée à la batterie.

ENA et ENB contrôlent la vitesse des moteurs gauches et droites de manière séparée par une sortie PWM.

IN1, IN2, IN3, IN4: IN1 et IN2 sont utilisées pour le contrôle des moteurs 1, IN3 et IN4 sont utilisées pour le contrôle des moteurs 2.

| ENA | IN1 | IN2 | ETAT DES MOTEURS |
|-----|-----|-----|------------------|
| 0 | X | X | STOP |
| 1 | 0 | 0 | FREIN |
| 1 | 0 | 1 | EN AVANT |
| 1 | 1 | 0 | EN ARRIERE |
| 1 | 1 | 1 | FREIN |

IV. Mettre le robot en mouvement

Première étape: piloter le moteur

Tout d'abord, nous allons mettre le moteur en mouvement sans nous soucier de la vitesse de rotation.

La carte L298N est reliée à la carte Elegoo UNO comme suit:

| | |
|-------|----------------------|
| L298N | Carte d'extension V5 |
| ENA | 5 |
| IN1 | 6 |
| IN2 | 7 |
| IN3 | 8 |
| IN4 | 9 |
| ENB | 11 |

Nous allons réaliser un programme simple, permettant de mettre les roues de droite en mouvement 0.5s dans un sens, puis 0.5s dans le sens inverse après une pause de 0.5s.

Ouvrez le fichier "right_wheel_rotation\right_wheel_rotation.ino"



Le code est le suivant:

```
/*In1 connected to the 9 pin,  
In2 connected to the 8 pin, ENA pin 10,*/  
int ENA=5;  
int IN1=6;  
int IN2=7;  
void setup()  
{  
  pinMode(IN1,OUTPUT);  
  pinMode(IN2,OUTPUT);  
  pinMode(ENA,OUTPUT);  
  digitalWrite(ENA,HIGH);  
}  
void loop()  
{
```

```
digitalWrite(IN1,HIGH);  
digitalWrite(IN2,LOW);    //Right wheel forward  
delay(500);  
digitalWrite(IN1,LOW);  
digitalWrite(IN2,LOW);    //Right wheel stop  
delay(500);  
digitalWrite(IN1,LOW);  
digitalWrite(IN2,HIGH);    //Right wheel back  
delay(500);  
digitalWrite(IN1,LOW);  
digitalWrite(IN2,LOW);    //Right wheel stop  
delay(500);  
}
```

Déversez le code sur la carte Elegoo UNO, débranchez le câble USB, posez le robot au sol et mettez-le en marche. Vous pourrez constater que le robot se comporte comme exposé.

De la même manière, vous pouvez mettre en mouvement les roues de gauche :

"Left_wheel_rotation\ Left_wheel_rotation.ino"



Le code est le suivant:

```
/*In3 connected to the 7 pin,  
  In4 connected to the 6 pin, ENB pin 5,*/  
int ENB=11;  
int IN3=8;  
int IN4=9;  
void setup()  
{  
  pinMode(IN3,OUTPUT);  
  pinMode(IN4,OUTPUT);  
  pinMode(ENB,OUTPUT);  
  digitalWrite(ENB,HIGH);  
}  
void loop()  
{  
  digitalWrite(IN3,LOW);  
  digitalWrite(IN4,HIGH);      //Left wheel forward  
  delay(500);  
  digitalWrite(IN3,LOW);  
  digitalWrite(IN4,LOW);      //Left wheel stop  
  delay(500);  
  digitalWrite(IN3,HIGH);  
  digitalWrite(IN4,LOW);      //Left wheel back  
  delay(500);  
  digitalWrite(IN3,LOW);  
  digitalWrite(IN4,LOW);      //Left wheel stop  
  delay(500);  
}
```

Deuxième étape : bouger en avant et en arrière

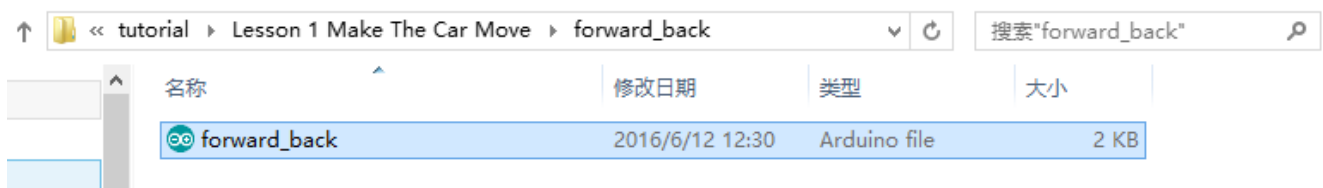
Vous pouvez maintenant écrire des programmes pour mettre le robot en mouvement

| | | | |
|------------------------|----------|------------|-------|
| ROBOT | En avant | En arrière | Arrêt |
| Roues de gauche | En avant | En arrière | Arrêt |
| Roue de droite | En avant | En arrière | Arrêt |

| | | | |
|------------------------|------------|------------|-------|
| ROBOT | A gauche | A droite | Arrêt |
| Roues de gauche | En arrière | En avant | Arrêt |
| Roue de droite | En avant | En arrière | Arrêt |

Nous allons écrire un programme simple permettant de mettre le robot en mouvement
En avant 0.5s, Arrêt 0.5s, En arrière 0.5s, Arrêt 0.5s.

Ouvez le fichier "forward_back\ forward_back.ino"



Le code est le suivant:

```
int ENA=5;  
int IN1=6;  
int IN2=7;  
int ENB=11;  
  
int IN3=8;  
int IN4=9;  
  
void setup()  
{  
    pinMode(IN1,OUTPUT);  
    pinMode(IN2,OUTPUT);  
    pinMode(IN3,OUTPUT);  
    pinMode(IN4,OUTPUT);  
    pinMode(ENA,OUTPUT);  
    pinMode(ENB,OUTPUT);  
}
```

```

digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);

}
void loop()
{
digitalWrite(IN1,HIGH);
digitalWrite(IN2,LOW);          // left wheel goes forward
digitalWrite(IN3,LOW);
digitalWrite(IN4,HIGH);        // right wheel goes forward
delay(500);
digitalWrite(IN1,LOW);
digitalWrite(IN2,LOW);          //left wheel holds still
digitalWrite(IN3,LOW);
digitalWrite(IN4,LOW);          // right wheel holds still
delay(500);
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);          //left wheel is back up
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);          // right wheel is back up
delay(500);
digitalWrite(IN1,LOW);
digitalWrite(IN2,LOW);          // left wheel holds still
digitalWrite(IN3,LOW);
digitalWrite(IN4,LOW);          // right wheel holds still
delay(500);
}

```

Déversez le programme sur la carte Elegoo UNO du robot après avoir branché celui-ci à l'ordinateur via le câble USB.

Débranchez le câble, posez le robot au sol, mettez-le en marche et observez son comportement.

Troisième étape : écrire un code

Il peut sembler difficile de créer un code complet permettant de mettre le robot en mouvement, surtout lorsque l'on veut en enchaîner une dizaine voire plus.

Nous vous conseillons donc de bien séparer les mouvements dans différentes fonctions : Aller en avant, Aller en arrière Le corps du programme (Loop()) appellera tour à tour les fonctions souhaitées et permettra une lecture plus aisée.

Le code est le suivant:

```
/******
```

```
Forward sub function
```

```
functions: Move forward
```

```
*****/
```

```
void forward( )
```

```
{
```

```
digitalWrite(IN1,HIGH);
```

```
digitalWrite(IN2,LOW); //Left wheel forward
```

```
digitalWrite(IN3,LOW);
```

```
}
```

```
/******
```

```
Forward sub function
```

```
functions: Move backward
```

```
*****/
```

```
void back( )
```

```
{
```

```
digitalWrite(IN1,LOW);
```

```
digitalWrite(IN2,HIGH); //Left wheel back
```

```
digitalWrite(IN3,HIGH);
```

```
digitalWrite(IN4,LOW); //Right wheel back
```

```

}
/*****
turnLeftsub function
functions: Turn left
*****/
void turnLeft( )
{
    digitalWrite(IN1,HIGH);
    digitalWrite(IN2,LOW);    //Left wheel back
    digitalWrite(IN3,HIGH);

}
/*****
turn Right sub function
functions: Turn right
*****/
void turnRight( )
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);    //Left wheel forward
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);    //Right wheel back
}
/*****
stop sub function
functions: Stop
*****/
void _stop()
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,LOW);    //Left wheel stop
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,LOW);    //Right wheel stop
}

```


Quatrième étape : mouvements automatiques

Pour finir, nous commençons à mettre le robot en mouvement automatiquement : en avant 0.4s, en arrière 0.4s, à gauche 0.4s, à droite 0.4s

Vous pouvez accéder à ce code via le fichier "AUTO_GO_\ AUTO_GO_.ino"



Le code est le suivant:

```
/*define logic control output pin*/
int in1=6;
int in2=7;
int in3=8;
int in4=9;
/*define channel enable output pins*/
int ENA=5;
int ENB=11;
/*define forward function*/
void _mForward()
{
    digitalWrite(ENA,HIGH);
    digitalWrite(ENB,HIGH);
    digitalWrite(in1,HIGH);//digital output
    digitalWrite(in2,LOW);
    digitalWrite(in3,LOW);
    digitalWrite(in4,HIGH);
    Serial.println("Forward");
}
/*define back function*/
void _mBack()
{
    digitalWrite(ENA,HIGH);
```

```
digitalWrite(ENB,HIGH);
digitalWrite(in1,LOW);
digitalWrite(in2,HIGH);
digitalWrite(in3,HIGH);
digitalWrite(in4,LOW);
Serial.println("Back");
}
/*define left function*/
void _mleft()
{
digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);
digitalWrite(in1,HIGH);
digitalWrite(in2,LOW);
digitalWrite(in3,HIGH);
digitalWrite(in4,LOW);
Serial.println("Left");
}
/*define right function*/
void _mright()
{
digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);
digitalWrite(in1,LOW);
digitalWrite(in2,HIGH);
digitalWrite(in3,LOW);
digitalWrite(in4,HIGH);
Serial.println("Right");
}
/*put your setup code here, to run once*/
void setup() {
Serial.begin(9600); //Open the serial port and set the baud rate to 9600
/*Set the defined pins to the output*/
pinMode(in1,OUTPUT);
pinMode(in2,OUTPUT);
pinMode(in3,OUTPUT);
```

```
pinMode(in4,OUTPUT);
pinMode(ENA,OUTPUT);
pinMode(ENB,OUTPUT);
}
/*put your main code here, to run repeatedly*/
void loop() {
  _mForward();
  delay(1000);
  _mBack();
  delay(1000);
  _mleft();
  delay(1000);
  _mright();
  delay(1000);
}
```

Déversez le programme sur la carte Elegoo UNO du robot après avoir branché celle-ci à l'ordinateur via le câble USB.

Débranchez le câble, posez le robot au sol, mettez-le en marche et observez-le. mportementcomportement.