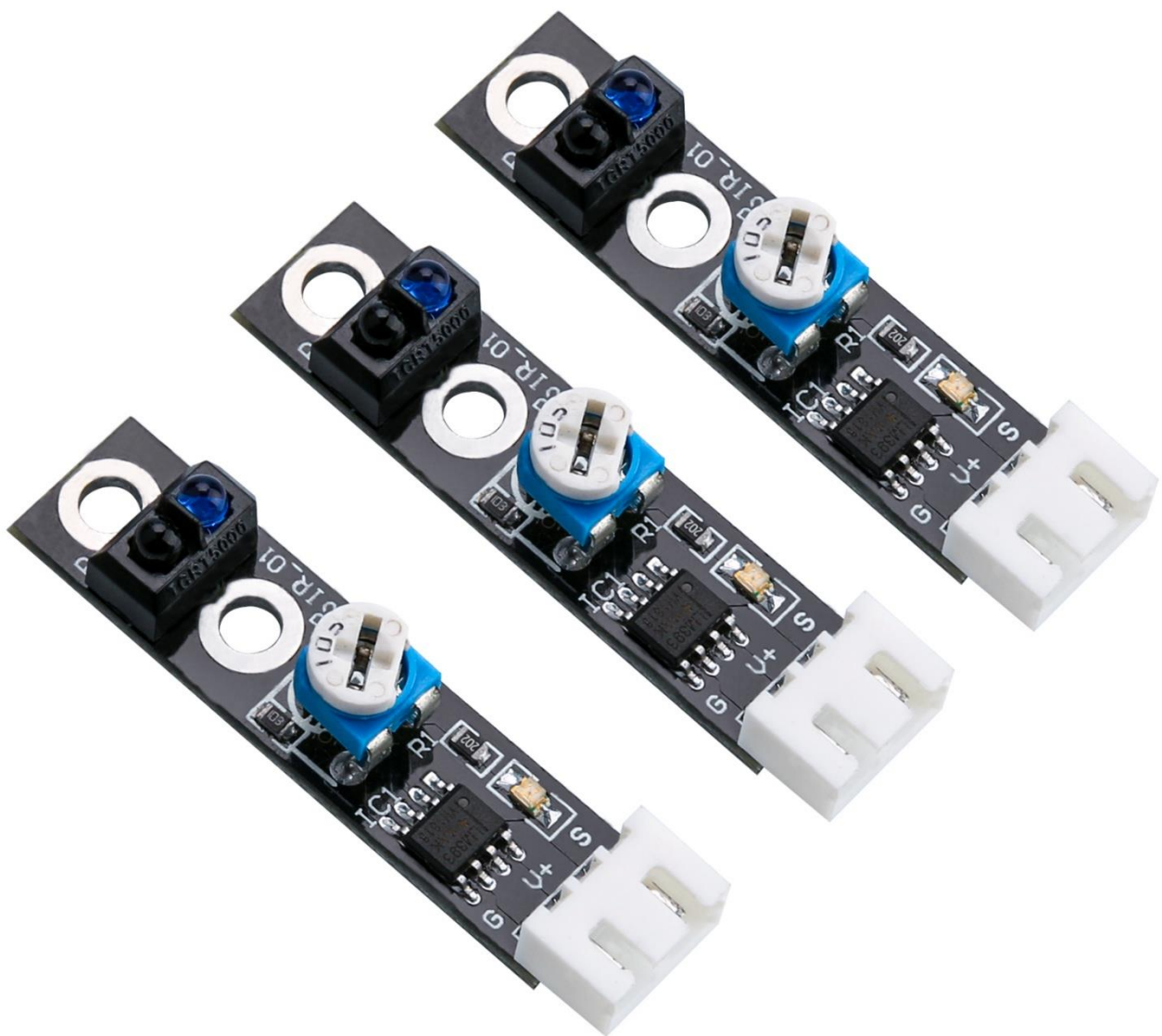


# Leçon n°5 – Suivi de ligne (Line tracking car)



---

## Points clés de la leçon

Dans cette leçon, nous allons apprendre comment contrôler le véhicule afin que celui-ci suive un parcours tracé au sol.

Sommaire:

- Utiliser un capteur "line tracking"

- Comprendre les principes de la détection de ligne

- Implémenter le code

Préparatifs:

- Le Robot équipé de trois capteurs "line tracking"

- Un câble USB (fourni)

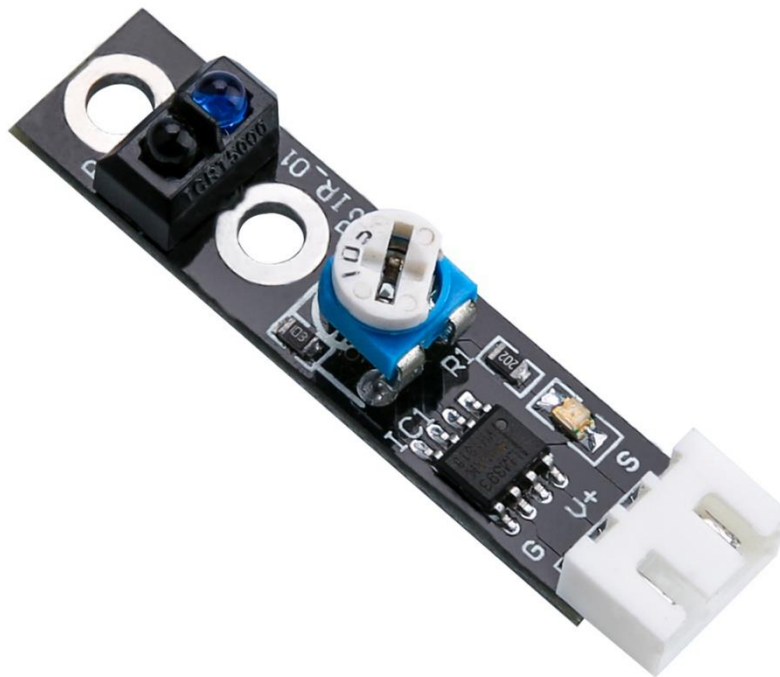
- Un rouleau d'adhésif de couleur noir (non fourni)

## I . Créer un parcours

Dans un premier temps, il est nécessaire de créer le parcours. Il est possible par exemple d'en créer un en collant de l'adhésif noir au sol (type ruban adhésif pour électricien). Faites bien attention de créer des courbures douces, sinon le véhicule ne sera pas en mesure de suivre correctement le changement de direction. Nous vous conseillons une dimension minimum de 40\*60cm.



## II. Connecter les modules et déboguer



Vue générale du capteur. Au centre de celui-ci est implantée une résistance variable réglable (via le tournevis fourni) qui permet d'ajuster la distance de détection et la sensibilité du capteur.

### III. Déverser le code

Après vous être assuré du bon montage du robot et avoir créé le parcours au sol, il ne vous reste plus qu'à déverser le code sur la carte Elegoo UNO.

Le code est le suivant:

```
int in1=6;
int in2=7;
int in3=8;
int in4=9;
int ENA=5;
int ENB=11;
int ABS=115;
void _mForward()
{
  analogWrite(ENA,ABS);
  analogWrite(ENB,ABS);
  digitalWrite(in1,HIGH);//digital output
  digitalWrite(in2,LOW);
  digitalWrite(in3,LOW);
  digitalWrite(in4,HIGH);
  Serial.println("go forward!");
}

void _mBack()
{
  analogWrite(ENA,ABS);
  analogWrite(ENB,ABS);
  digitalWrite(in1,LOW);
  digitalWrite(in2,HIGH);
  digitalWrite(in3,HIGH);
  digitalWrite(in4,LOW);
  Serial.println("go back!");
}

void _mleft()
```

```
{  
  analogWrite(ENA,ABS);  
  analogWrite(ENB,ABS);  
  digitalWrite(in1,HIGH);  
  digitalWrite(in2,LOW);  
  digitalWrite(in3,HIGH);  
  digitalWrite(in4,LOW);  
  Serial.println("go left!");  
}
```

```
void _mright()  
{  
  analogWrite(ENA,ABS);  
  analogWrite(ENB,ABS);  
  digitalWrite(in1,LOW);  
  digitalWrite(in2,HIGH);  
  digitalWrite(in3,LOW);  
  digitalWrite(in4,HIGH);  
  Serial.println("go right!");  
}
```

```
void _mStop()  
{  
  digitalWrite(ENA,LOW);  
  digitalWrite(ENB,LOW);  
  Serial.println("Stop!");  
}
```

```
void setup()  
{  
  Serial.begin(9600);  
}
```

```
void loop() {  
  int num1,num2,num3;  
  num1=digitalRead(10);  
  num2=digitalRead(4);
```

```

num3=digitalRead(2);
if((num1==0)&&num2&&num3)
{
    _mleft();          //The sensor detected that right car turn left immediately when
it meets black line
    delay(2);
    while(1){
        num2=digitalRead(2);          //Cycle to judge degree of intermediate sensor,
        if(num2==1)
        { _mleft();          //If num2==1 does not go to the middle position,
continue to turn left.
        delay(2);}
        else
        break;          //Detection of num2==0 instructions turned over,
out of the loop, detection of three sensors' statusand then make appropriate action
    }          //The following and so on
}

```

```

else if(num2&&num1&&(num3==0))
{
    _mright();
    delay(2);
    while(1)
    {
        num2=digitalRead(2);
        if(num2==1){
            _mright();
            delay(2);}
        else
        break;
    }
}
else
{
    _mForward();
    delay(2);
}

```

```
}
```

```
}
```

Ouvrez le fichier "Line\_Tracking\_Car /Line\_Tracking\_Car.ino" et déversez le code sur la carte Elegoo UNO. Veillez préalablement à avoir correctement branché la carte Elegoo UNO sur l'ordinateur via le câble USB fourni et avoir retiré le module BLUETOOTH. Débranchez le câble USB, mettez le Robot en marche et placez-le sur le parcours: le robot se déplace en suivant la ligne.

Note : si le robot ne suit pas correctement le parcours, vous pouvez essayer de régler les potentiomètres des capteurs pour augmenter la sensibilité de détection.

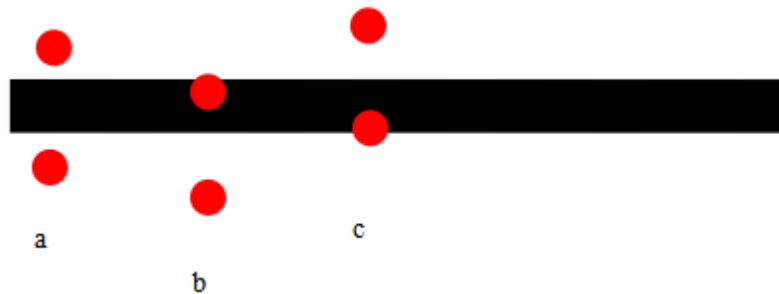


## IV. Principe de fonctionnement

### Les capteurs "line tracking"

Les capteurs "line tracking" sont placés sous le robot, à l'avant. Ils sont chacun équipés de deux leds infrarouges, l'une émettrice, l'autre réceptrice. Une surface de couleur noire absorbe la majeure partie des ondes infrarouges, créant une différence de réflexion avec une surface de couleur vive. Le capteur détecte donc s'il se situe au-dessus d'une surface noire ou pas.

**En associant plusieurs détecteurs, il est donc possible de déterminer la position relative du robot par rapport à une ligne noire donc d'agir sur sa trajectoire afin de se recentrer sur cette ligne.**



- a. Aucune détection n'est faite par les détecteurs. Cela signifie que la ligne est au centre. Le robot a donc ordre de continuer d'avancer sans opérer de correction de trajectoire.
- b. Le capteur de droite détecte la ligne. Le robot a donc ordre d'opérer une correction de trajectoire et de tourner vers la droite.
- c. Le capteur de gauche détecte la ligne. Le robot a donc ordre d'opérer une correction de trajectoire et de tourner vers la gauche

Nous venons de voir le principe de base de la détection de ligne. C'est un algorithme simple qui est utilisé dans notre exemple. Des versions plus élaborées existent qui permettent un contrôle plus fin de la trajectoire.

---

**Astuces:**

- (1) Les tournants doivent être doux. Le robot ne peut opérer de virage brutal.
- (2) Il est vivement conseillé d'utiliser de l'adhésif noir. Cependant, vous pouvez essayer d'autres couleurs afin de déterminer celles qui fonctionnent le mieux ou le moins bien.
- (3) Après avoir expérimenté et compris le "Line Tracking", vous pouvez imaginer par vous-même d'autres applications qui permettent de confiner les déplacements du robot à une zone précise.
- (4) robot à une région déterminée.