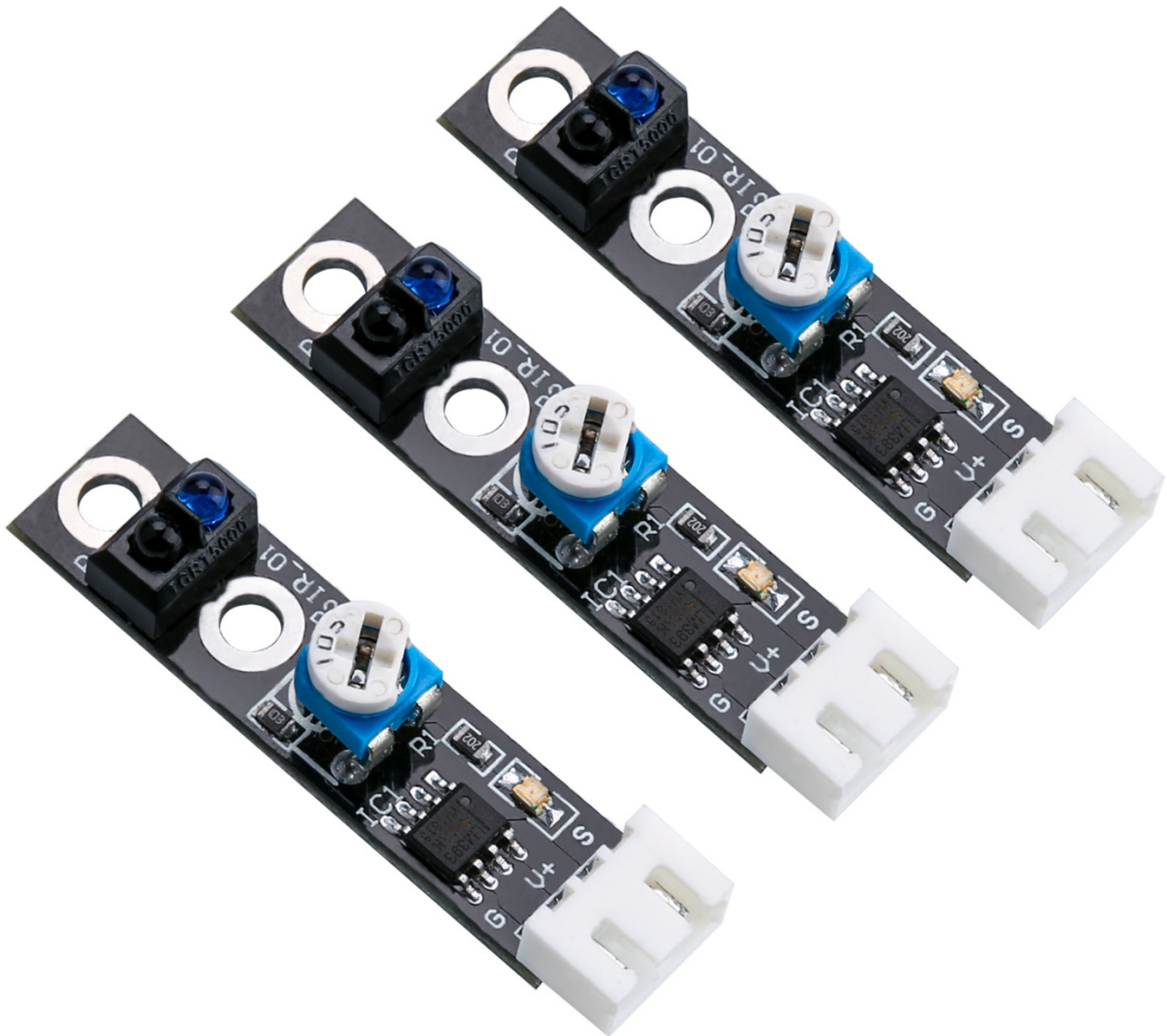


Lektion 5 Linien- Verfolgung



Punkte des Abschnitts

In dieser Lektion werden wir lernen, wie man das Auto kontrolliert, um eine Landebahn zu bewegen.

Lernteile:

- ◆ Erfahren Sie, wie Sie die Linien- Erkennungs -Module verwenden
- ◆ Erlernen Sie das Prinzip der Linienverfolgung
- ◆ Erfahren Sie, wie Sie das Tracking über die Programmierung implementieren können

Vorbereitungen:

- ◆ Ein Auto (mit Batterie ausgestattet)
- ◆ Ein USB cable
- ◆ Drei Linien- Erkennungsmodule
- ◆ Eine Rolle schwarzes Band

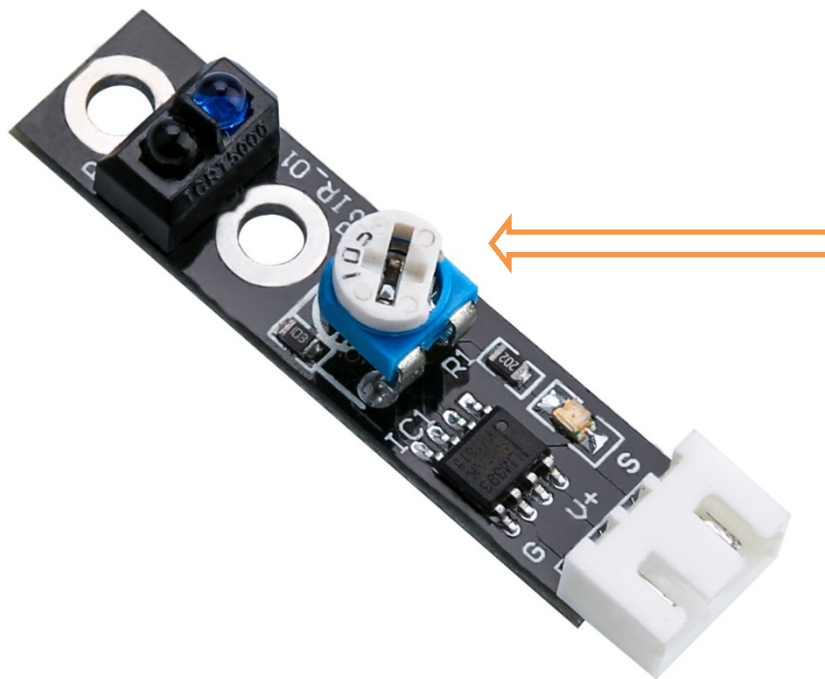
I . Landebahn aufbauen

Materialien: elektrisches Klebeband (schwarzes Band)

Zuerst müssen wir eine eigene Fahrbahn bauen. Wir können einen Kreislauf durch aufkleben von schwarzem Band auf ein geeignetes Papier oder dem Boden machen. Vor dem aufkleben können Sie eine Piste mit dem Stift ziehen und dann mit elektrischem Klebeband bekleben. Achten Sie darauf, die Ecke so glatt wie möglich zu machen. Weil das Auto aus der Linie herauskommt, wenn der Winkel zu klein ist. Aber wenn Sie es noch schwieriger machen wollen, können sie natürlich auch einen kleineren Winkel ausprobieren. Die Größe der Fahrbahn ist in der Regel nicht kleiner als 40 * 60 cm.



II. Verbinden der Module und debuggen



Das Bauteil, auf das der nette kleine Pfeil zeigt, ist ein Potentiometer. Es kann die Empfindlichkeit des Linienverfolgungsmoduls durch Änderung seines Widerstandswertes einstellen.

III. Programm hochladen

Nachdem Sie die Fahrbahn, auf der das Auto fahren soll, aufgebaut und die Module verbunden haben, müssen Sie das Programm nur noch auf die UNO- Controller- Platine hochladen.

Der Code ist wie folgt:

```
#define lineTracking1 10
#define lineTracking2 4
#define lineTracking3 2

int in1=6;
int in2=7;
int in3=8;
int in4=9;
int ENA=5;
int ENB=11;
int ABS=115;

bool num1, num2, num3;
bool flag;

void _mForward(){
  analogWrite(ENA,ABS);
  analogWrite(ENB,ABS);
  digitalWrite(in1,HIGH);
  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() {
  num1 = digitalRead(lineTracking1);
  num2 = digitalRead(lineTracking2);
  num3 = digitalRead(lineTracking3);
```

```
  if(num2){
    _mForward();
  }
  else if(num1) {
    while(1) {
      num1 = digitalRead(lineTracking1);
      if(num1) {
        _mleft();
      }
      else{
        break;
      }
    }
  }
  else if(num3) {
    while(1) {
      num3 = digitalRead(lineTracking3);
      if(num3) {
        _mright();
      }
      else{
        break;
      }
    }
  }
}
```

```

    }
}
else{
    num1 = digitalRead(lineTracking1);
    num3 = digitalRead(lineTracking3);
    if(num1 && !num3){
        _mleft();
        while(1){
            num3 = digitalRead(lineTracking1);
            if(num3) {
                break;
            }
        }
    }
    else if(num3 && !num1){
        _mright();
        while(1){
            num1 = digitalRead(lineTracking1);
            if(num1) {
                break;
            }
        }
    }
}
}
}

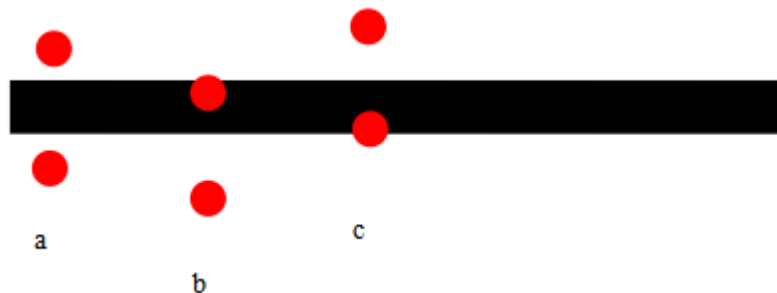
```

Öffnen Sie die Datei Line_Tracking_Car /Line_Tracking_Car.ino und laden Sie das Programm auf die UNO-Controller-Karte. Nach dem Trennen des Autos auf den Computer können Sie den Netzschalter einschalten und das Auto auf die Fahrbahn stellen. Dann wird das Auto den Linien folgen. Wenn Sie feststellen, dass es sich nicht wie erwartet bewegt, stellen Sie bitte das Potentiometer auf dem Linienverfolgungsmodul entsprechend ein.

IV. Einführung in das Grundprinzip

Linienverfolgungsmodul

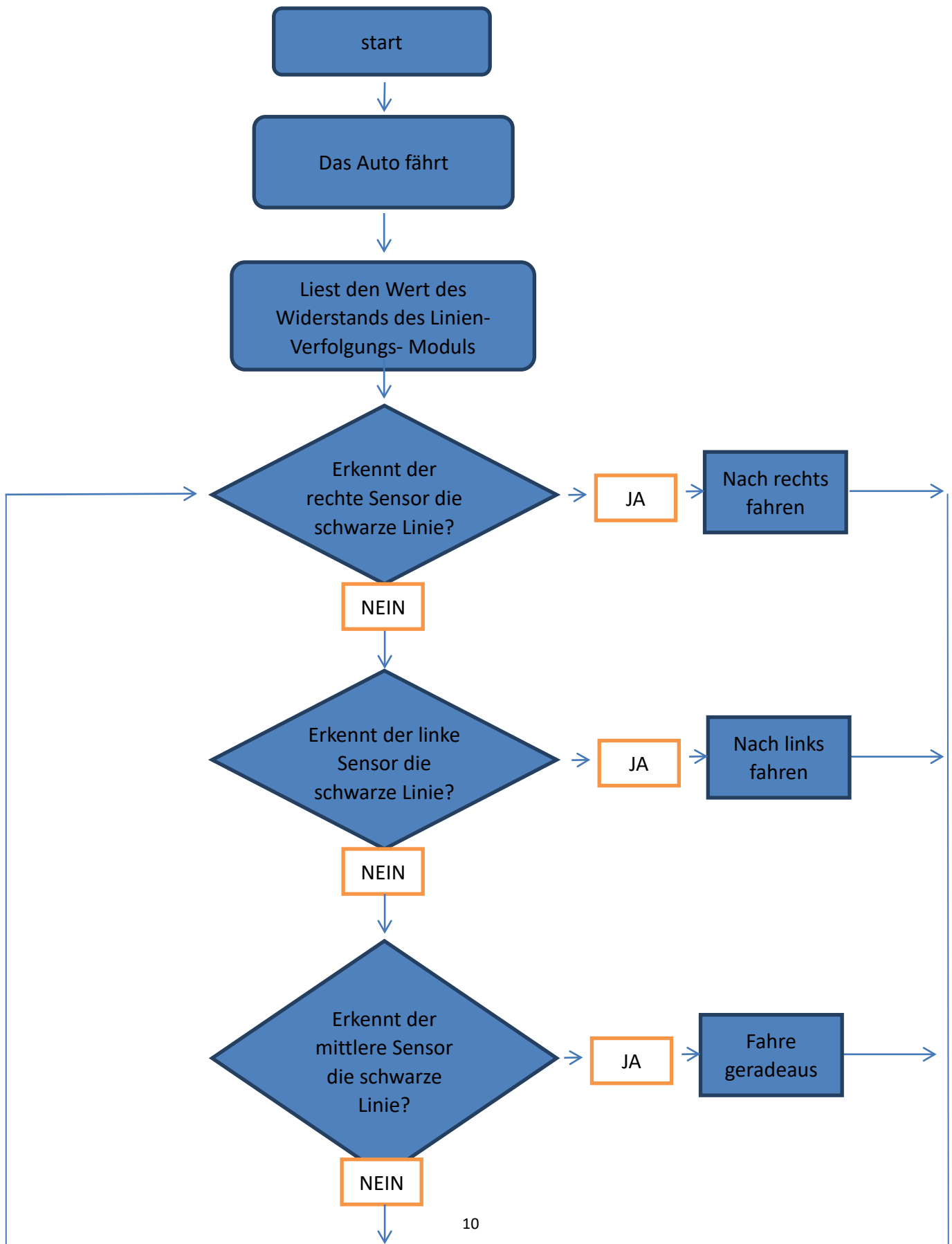
Die Linienverfolgungssensoren sind die beiden Komponenten, die sich vor dem Auto befinden. Der Linienverfolgungssensor besteht aus einem Infrarot-Senderrohr und einem Infrarot-Empfängerrohr. Erstere ist eine LED, die den Infrarotstrahl übertragen kann, während letzterer ein Photowiderstand ist, der nur für das Infrarotlicht verantwortlich ist. Die Lichtreflexionen von der schwarzen Oberfläche unterscheidet sich von der von einer weißen Oberfläche. Daher unterscheidet sich die Intensität des reflektierten Infrarotlichts, das das Auto an der schwarzen Straße empfängt, von der Intensität durch die Reflexionen der weißen Straße. Die Widerstandsgröße ändert sich ebenfalls. Entsprechend dem Prinzip des Spannungsteilers unter dem Serienwiderstand kann der Bewegungsweg bestimmt werden, indem die Farbe der Straße unterhalb des Fahrzeugs von der Spannung des Sensors abgeleitet wird.



A → Das Auto bewegt sich entlang der schwarzen Linie. Eines der Linienverfolgungsmodule befindet sich auf der linken Seite der Linie und das andere ist auf der rechten Seite. Sie können die schwarze Linie nicht erkennen.

B → Das Auto wird beginnen sich nach rechts zu bewegen. Das Modul auf der linken Seite kann die schwarze Linie erkennen, dann sendet es Signal an die Anschaltbaugruppe und das Auto wird links abbiegen.

C → Das Auto wird beginnen sich nach links zu bewegen. Das Modul auf der rechten Seite kann die schwarze Linie erkennen, dann wird es Signal an die Controller-Platine senden und das Auto wird rechts abbiegen.



Von oben können wir das Prinzip des Linien- verfolgungs- Autos sehen. Nach dem Start des Autos muss das Line-Tracking-Modul nur die schwarze Linie auf der Fahrbahnoberfläche erfassen, entsprechend dem Programm entsprechende Aktionen vornehmen.

Dies ist ein einfaches Struktogramm des Auto-Line-Tracking-Programms. Es gibt viele komplexere Algorithmen wie z.B. PID. Also, nachdem die Funktion der Linienverfolgung erfolgreich geklappt hat, können Sie auf eigene Faust versuchen, mehr Algorithmen zur Kontrolle Autos zu entwickeln.

Kleine Tipps

- (1) Die Kurventeile der Strecke sollten so glatt wie möglich sein. Wenn der Kurvenradius zu klein ist, wird das Auto sehr wahrscheinlich über die Fahrbahn hinaus weiter fahren.
- (2) Die Fahrstrecke kann mit dem Schwarzweißband oder einem Papier beliebiger Farbe gemacht werden, Hauptsache sie unterscheidet sich von der Farbe des Weges.
- (3) Neben der Linienverfolgung können wir die Phantasie ausdehnen, um andere Programme zu entwickeln, die auf den Linienverfolgungsprinzipien basieren, wie diejenigen, die das Auto innerhalb einer Region unabhängig von ihrer Bewegung beschränken.