

Lektion 1 – Das Auto bewegen

Punkte dieses Abschnitts

Lernziel:

- ◆ Erfahren Sie, wie Sie Arduino IDE verwenden können
- ◆ Machen Sie das Auto fahren, indem Sie das Programm hochladen

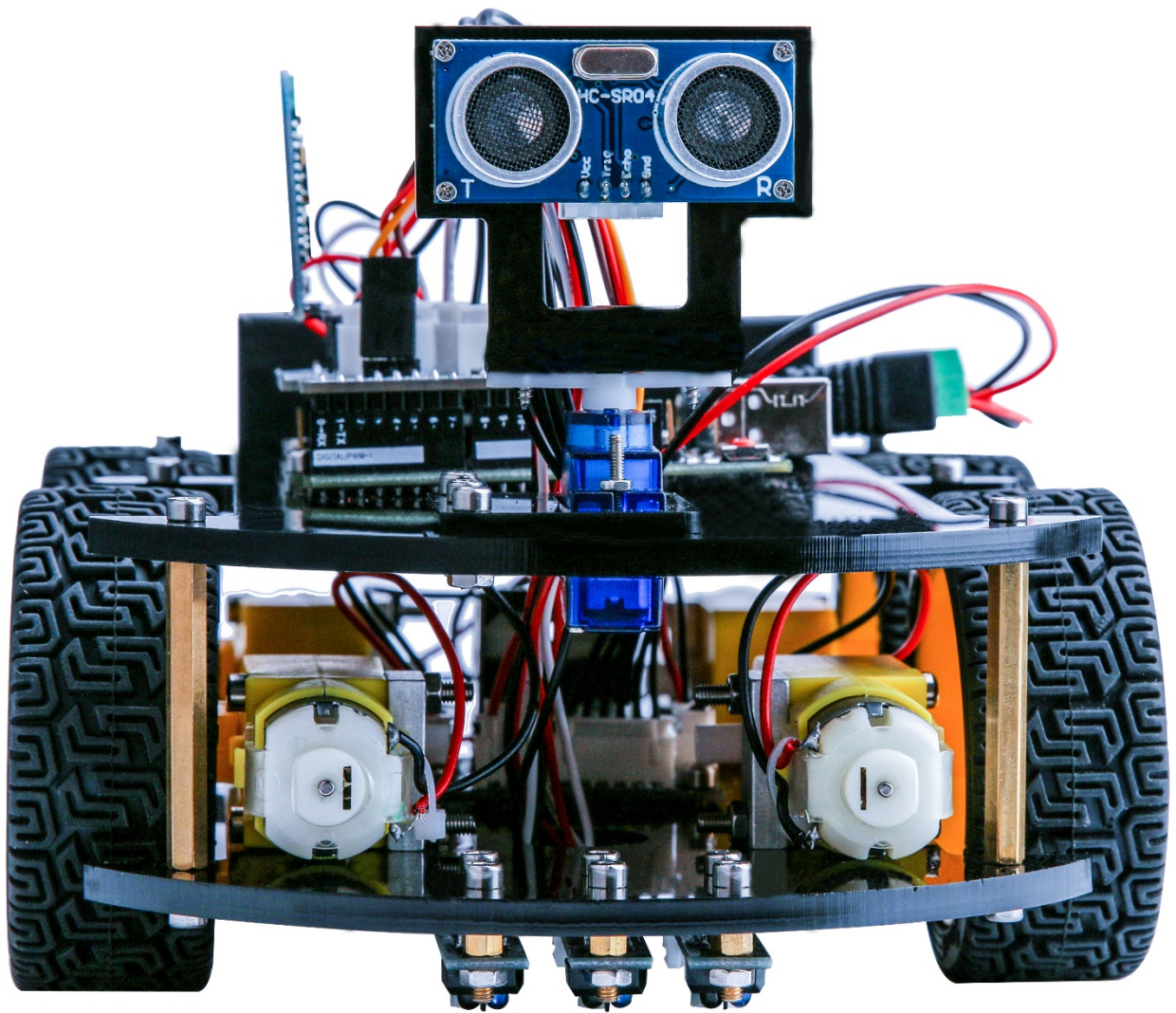
Vorbereitungen:

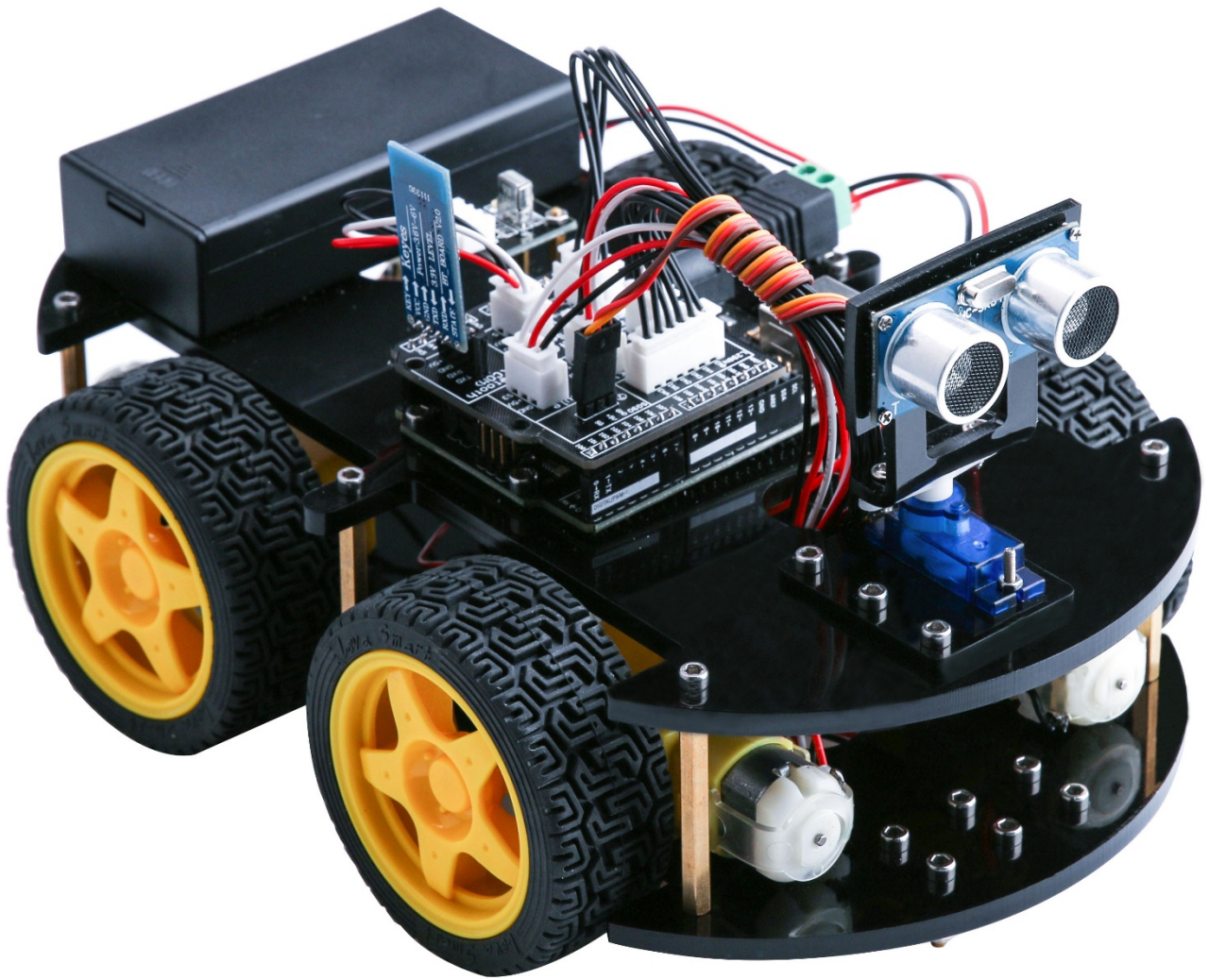
- ◆ Ein Auto (mit einer Batterie)
- ◆ Ein USB-Kabel

I . Einführung des Autos

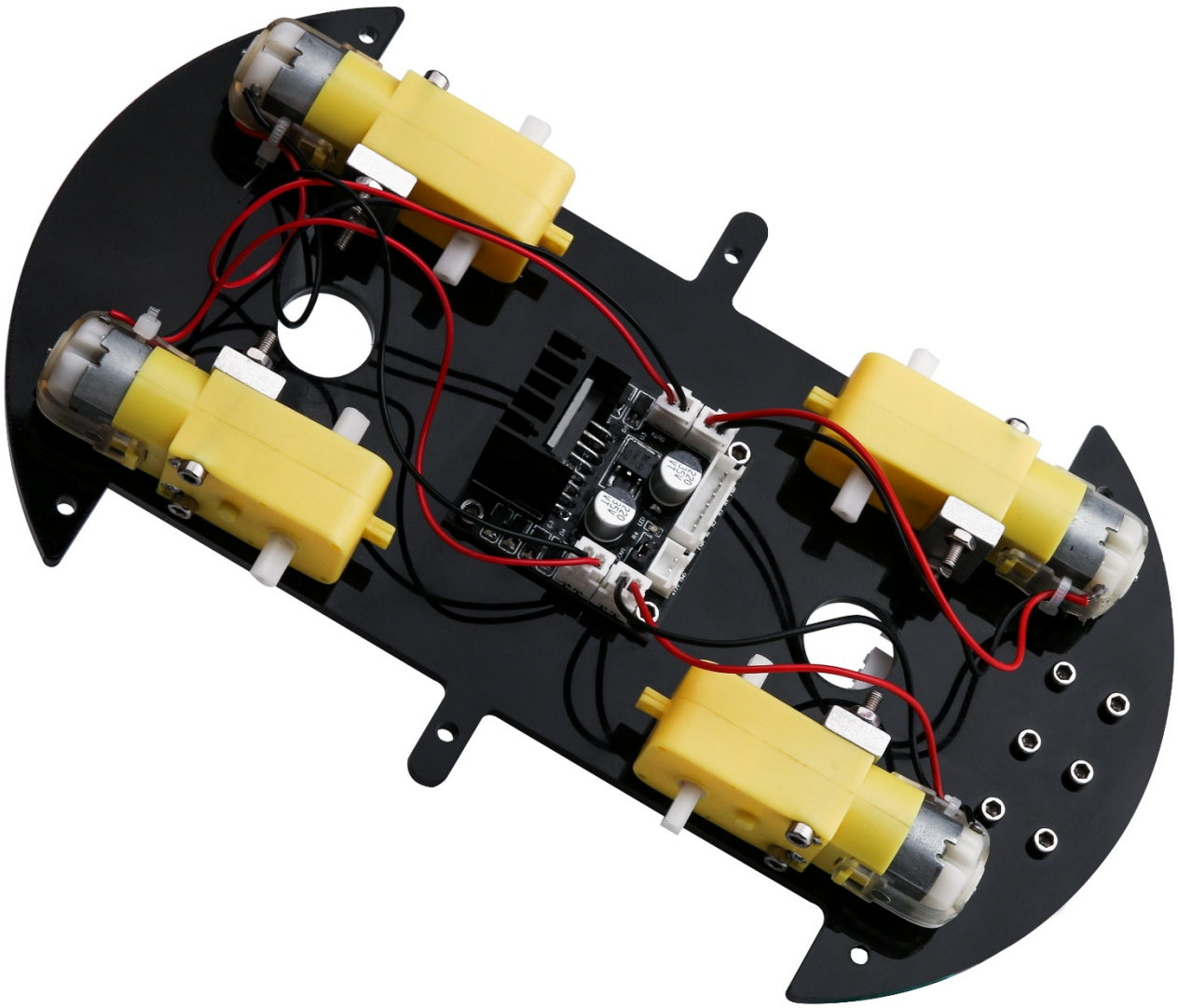
Dieser Bausatz ist ein äußerst flexibler Fahrzeugbausatz, der speziell für Bildungs-, Wettkampf- und Unterhaltungszwecke entwickelt wurde. Die obere Platte des Bausatzes ist direkt mit 9-Gramm-Lenkmotor kompatibel. Es trägt auch Überschall-Sensor, Batterie und andere feste vorgegebene Löcher, um die Installation von verschiedenen Sensoren zu erleichtern. Dies ist ein sehr lustiger und vielseitiger Roboter, der Lern- und Produktionszwecke erfüllt. Mit ihm können Sie diverse interessante Ideen wie Bluetooth und Infrarot-Fernbedienung, automatische Vermeidung von Hindernissen und Linieninspektion umsetzen.

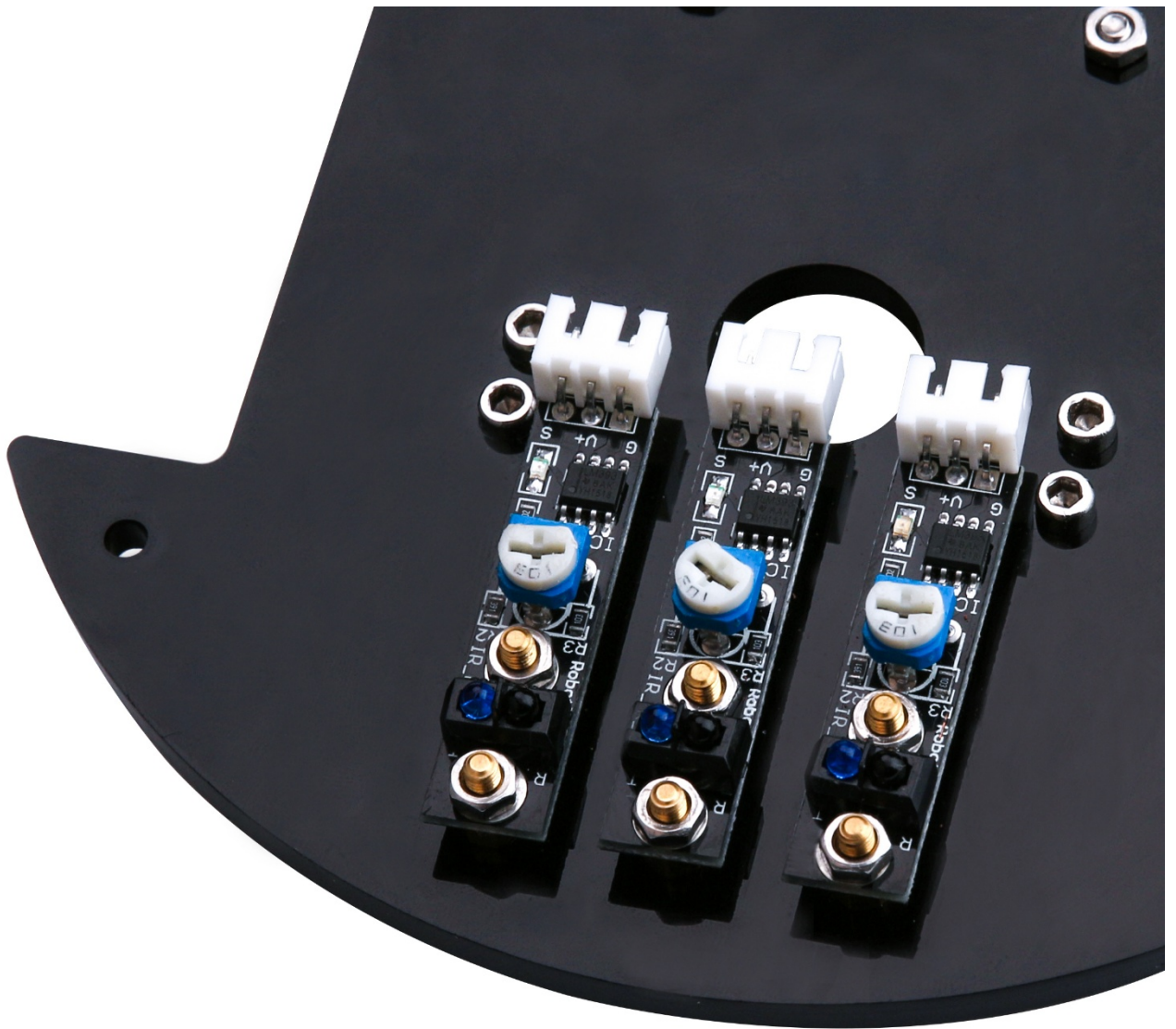
Lassen Sie uns das kleine Fahrzeug beschreiben, das uns in künftiger Zeit begleiten wird.

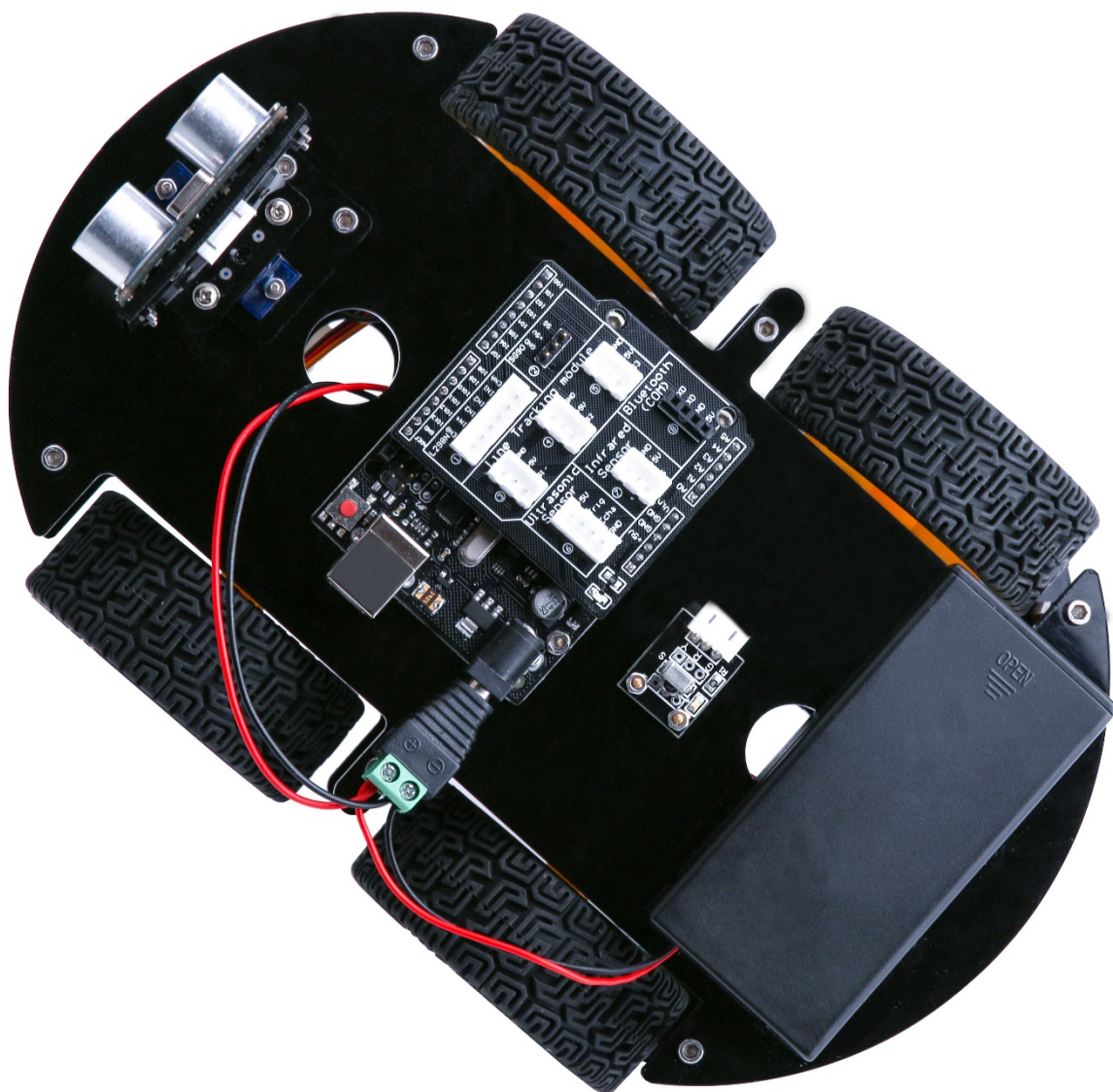




Das Auto ist wie unten aufgebaut:







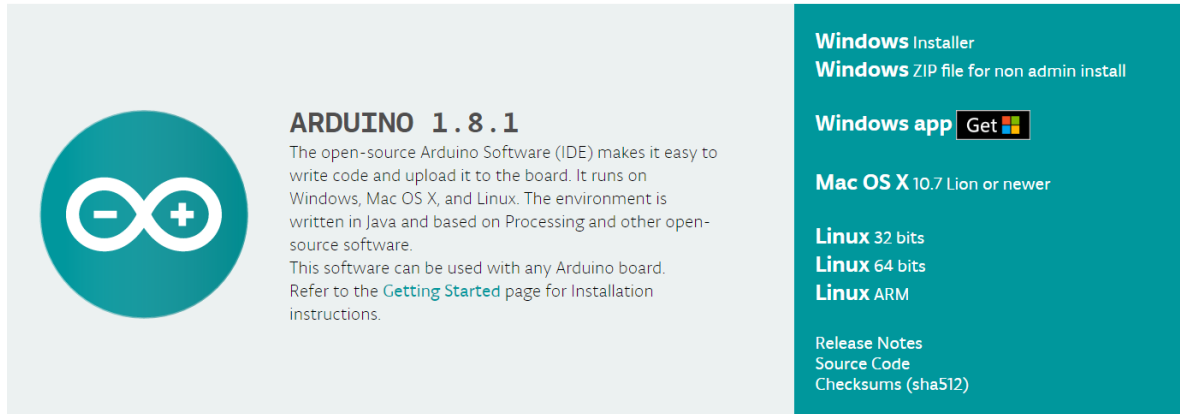
Funktion jedes Teils:

1. Batteriehalter mit Schalter: dient als Netzteil für das Fahrzeug
2. Elektromotor + Rad: lässt das Fahrzeug fahren 😊
3. Acrylplatte: der Rahmen des Autos
4. L298N Motor Fahrkarte: Lässt die Motoren drehen
5. UNO-Controller-Board: das Gehirn des Autos, steuert alle Teile
6. V5 Sensor Erweiterungskarte: kombiniert mit dem UNO, machen die Verbindung einfacher
7. Servo und Cloud-Plattform: Den GP2Y0A21 Distanzsensor um 180 Grad drehen lassen
8. Ultraschallsensormodul: Distanzmessung und Hindernisvermeidung
9. Line-Tracking-Modul: Schwarz-Weiß-Sensor zur Erkennung der weißen und schwarzen Bahnen
10. Infrarot-Empfänger und Fernbedienung: Bereitstellung der Infrarot-Fernbedienung Funktion
11. Bluetooth-Modul: Stellen Sie die Bluetooth-Steuerfunktion bereit

II. Programm hochladen

Jede Bewegung des Fahrzeugs wird durch das Programm gesteuert, so dass es notwendig ist, das Programm installiert und korrekt eingerichtet zu haben.

STEP 1: Gehen Sie zu <https://www.arduino.cc/de/Main/Software> und finden Sie die Downloadseite.



Die auf dieser Website verfügbare Version ist in der Regel die neueste Version, und die aktuelle Version kann neuer sein als die Version im Bild.

STEP2: Laden Sie die Entwicklungssoftware herunter, die für das Betriebssystem Ihres Computers geeignet ist. Nimm Windows als Beispiel hier.

Windows Installer

Sie können es mit dem EXE-Installationspaket oder dem grünen Paket installieren.

Support the Arduino Software

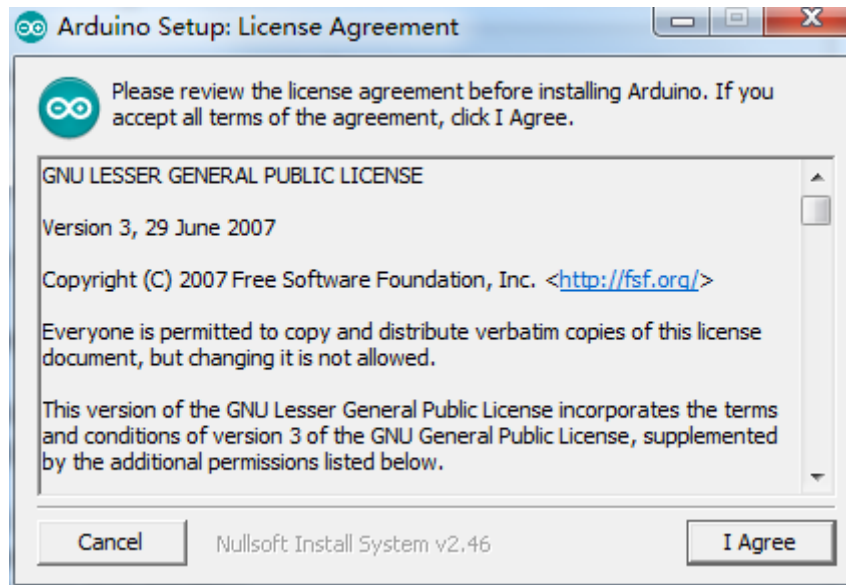
Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



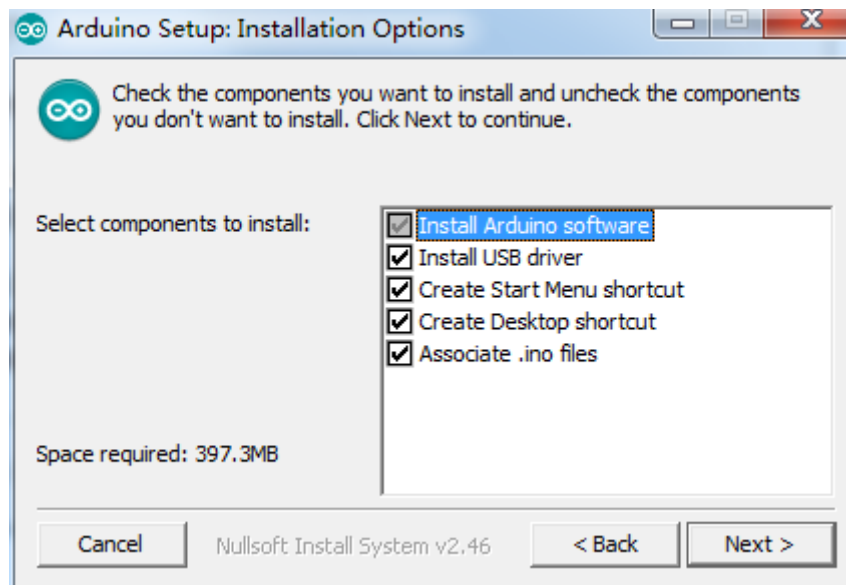
JUST DOWNLOAD

Drücken Sie die Taste "JUST DOWNLOAD", um die Software herunterzuladen.

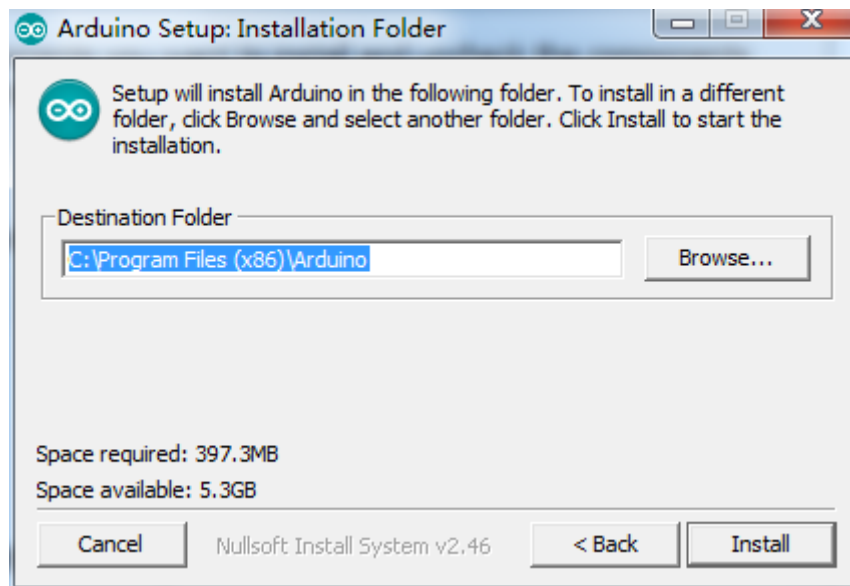
Diese sind in den von uns zur Verfügung gestellten Materialien enthalten, und die Versionen unserer Materialien waren die neuesten Versionen, als dieser Kurs gemacht wurde.



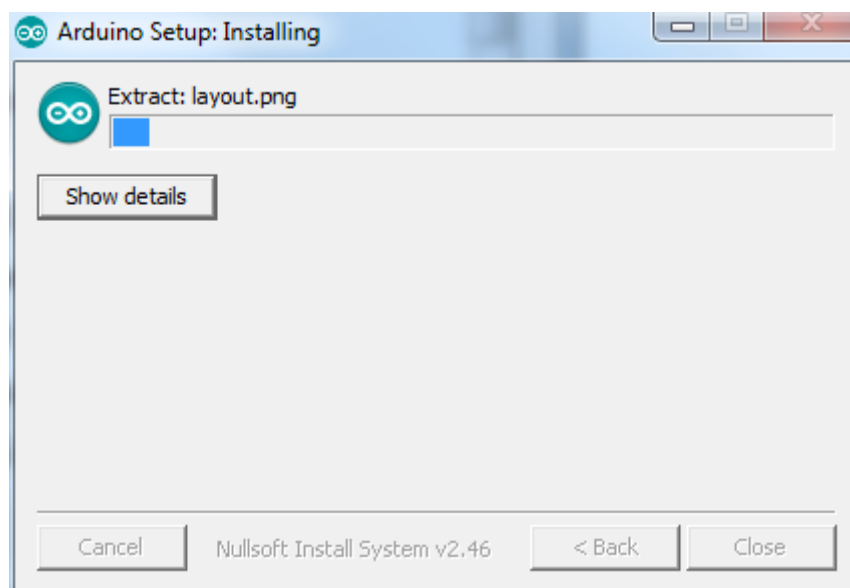
Wählen Sie Ich stimme zu, um die folgende Schnittstelle zu sehen



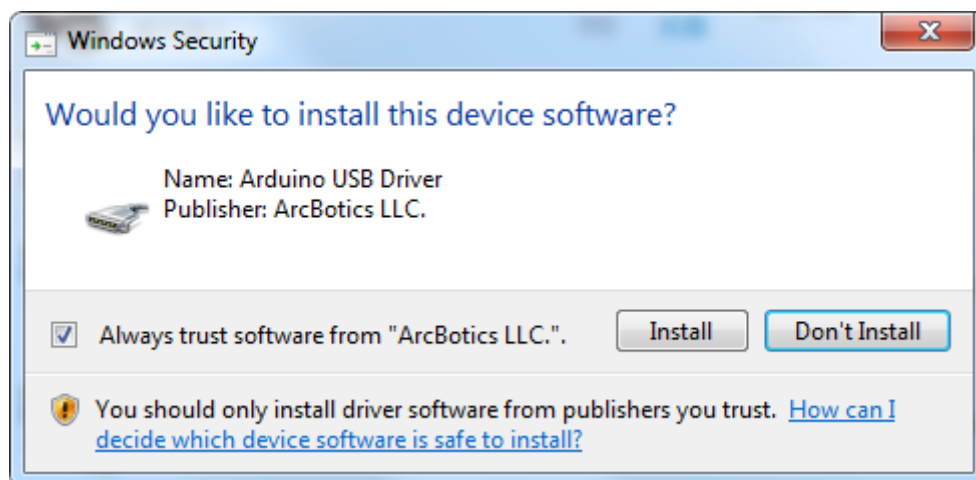
Wählen Sie Weiter



Drücken Sie Install, um die Installation zu starten

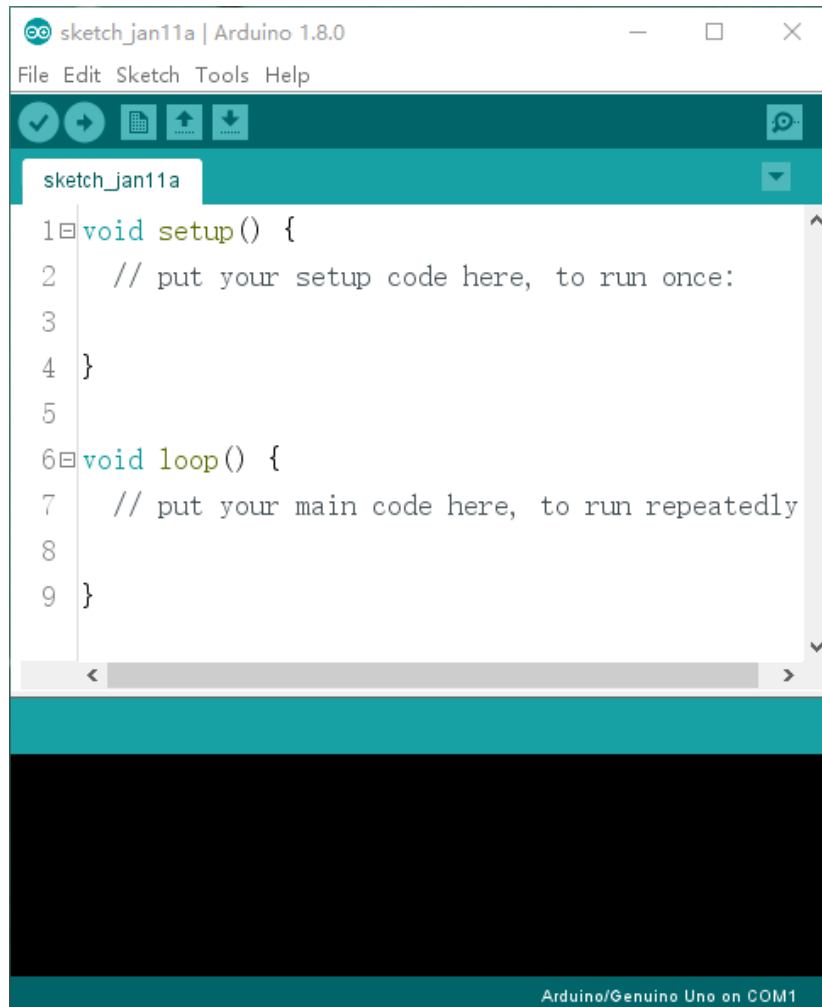


Schließlich erscheint die folgende Schnittstelle, Sie sollten Installieren wählen, um die Richtigkeit der Entwicklung zu gewährleisten

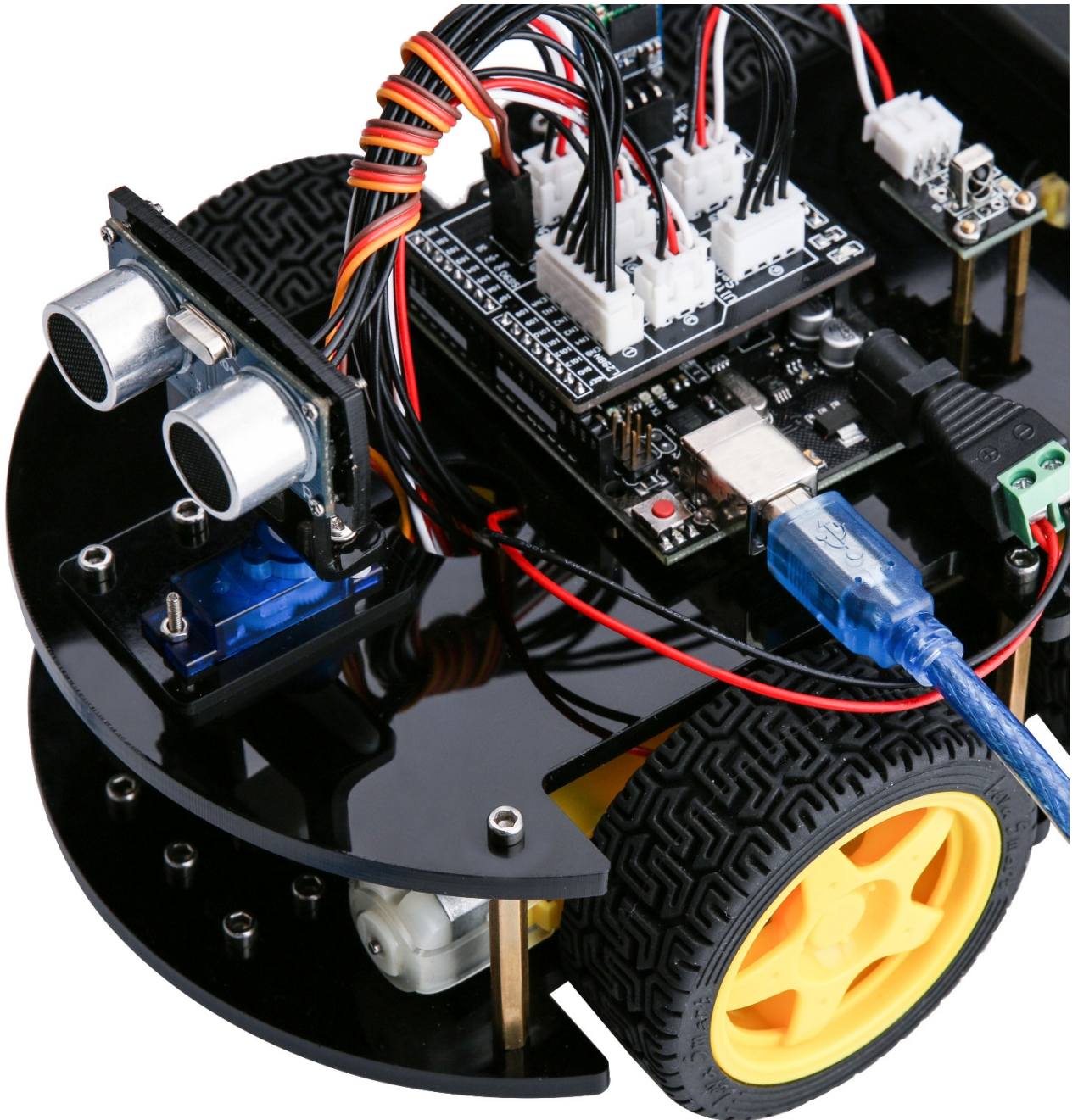


Als nächstes erscheint das folgende Symbol auf dem Desktop

Doppelklicken Sie auf die gewünschte Entwicklungsumgebung



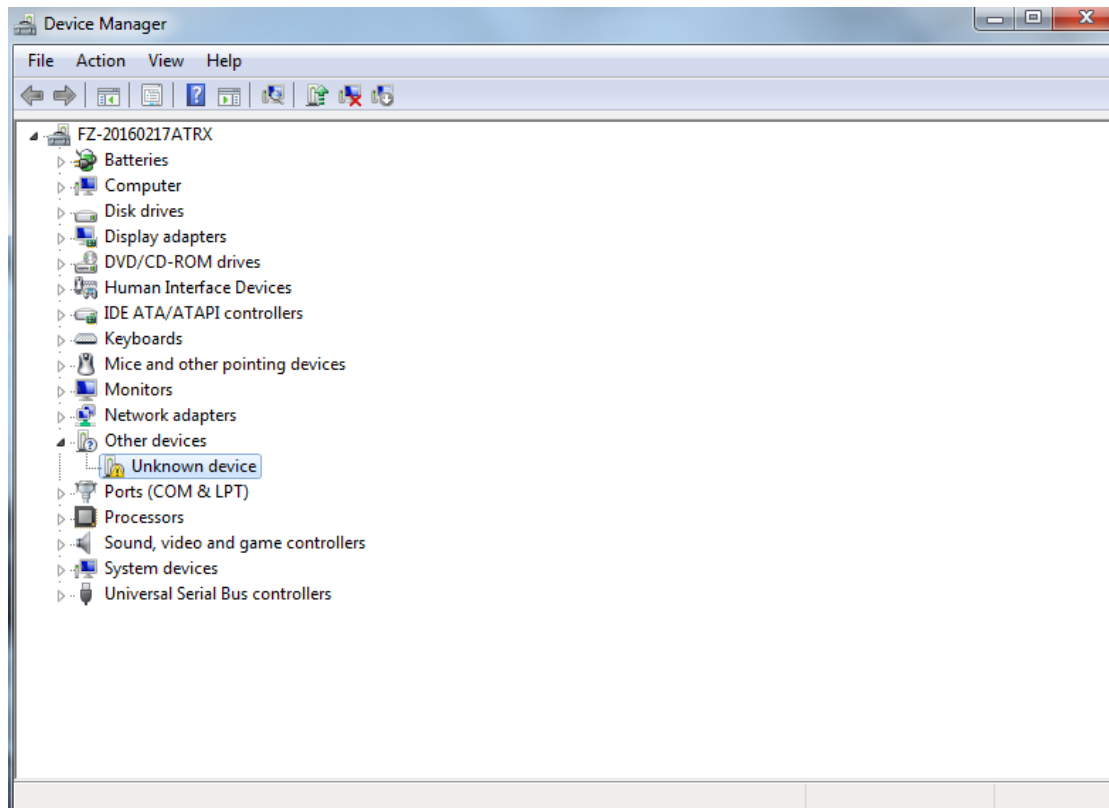
STEP3: Verbinden Sie das Auto mit dem Computer.



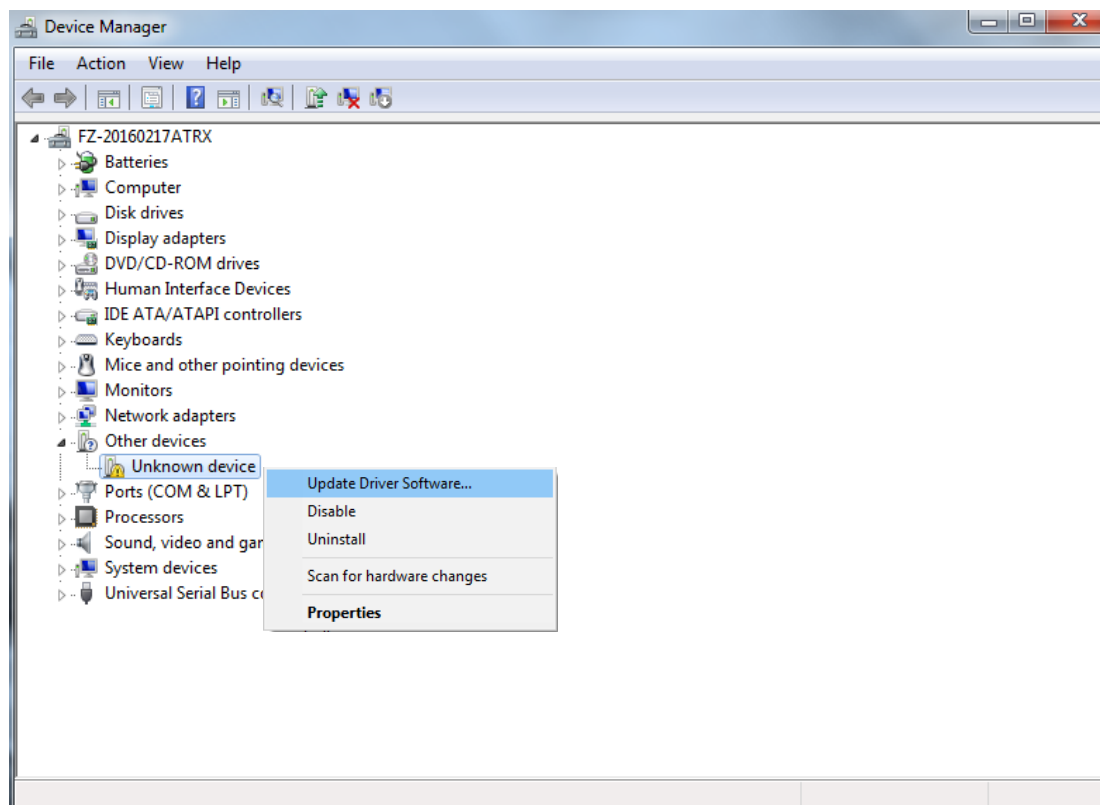
STEP 4: Öffnen Sie IDE - Tool - Port. Wenn Sie den richtigen Port sehen, bedeutet dies, dass das Fahrzeug korrekt mit dem Computer verbunden ist. In diesem Fall kannst du direkt zu STEP 5 springen. Andernfalls müssen Sie den Treiber auf folgende Weise installieren.

Öffnen Sie den Geräte-Manager, indem Sie mit der rechten Maustaste auf Arbeitsplatz - Management -

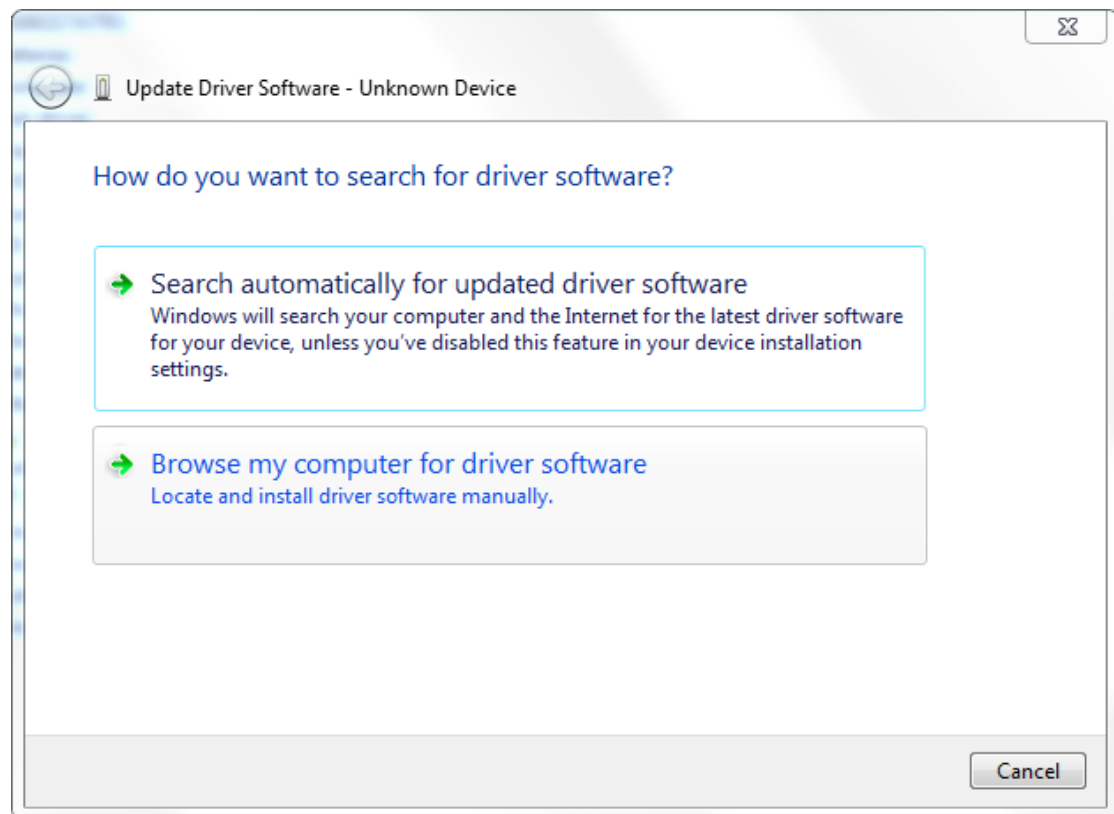
Geräte-Manager klicken



Klicken Sie mit der rechten Maustaste auf unbekanntes Gerät ----- Update Gerät Software



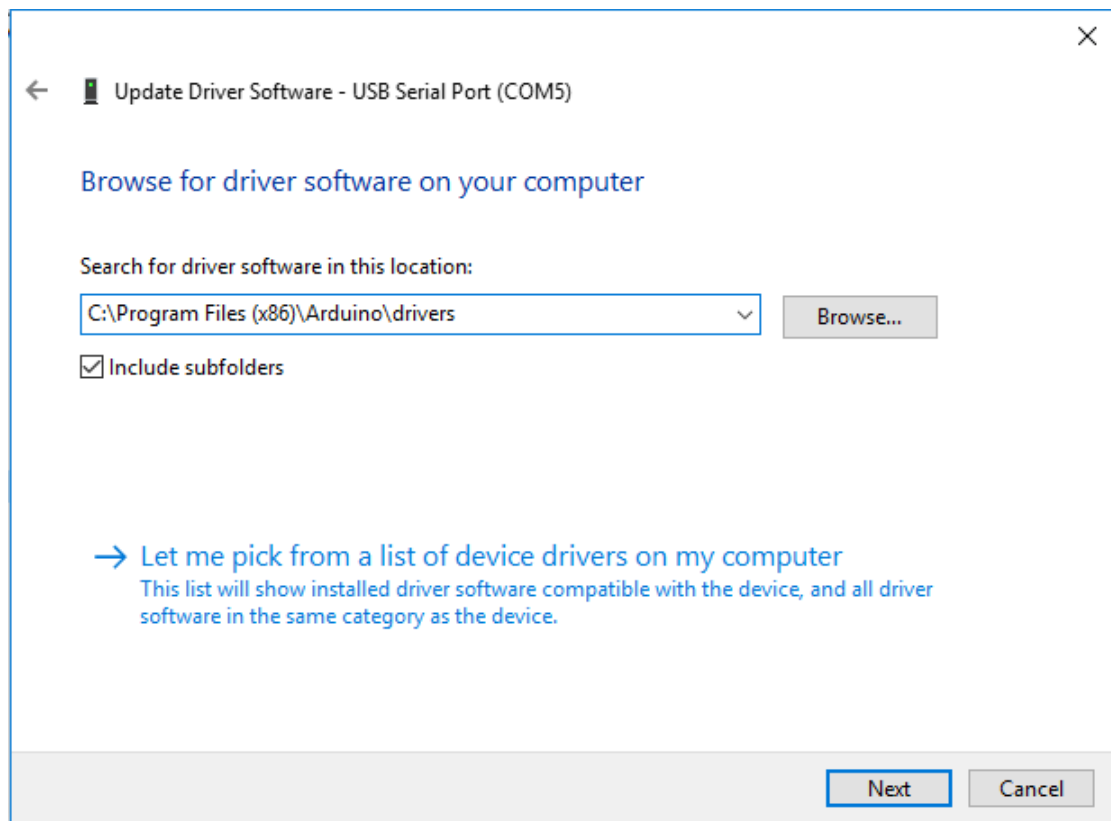
Es zeigt an, dass der Treiber nicht installiert wurde, und Sie müssen auf meinen Computer auf Treiber-Software klicken, um die Treiber zu finden. Die Laufwerke befinden sich im Arduino-Ordner. Normalerweise installieren Sie den Ordner in C: \ Program Files (x86) \ Arduino.



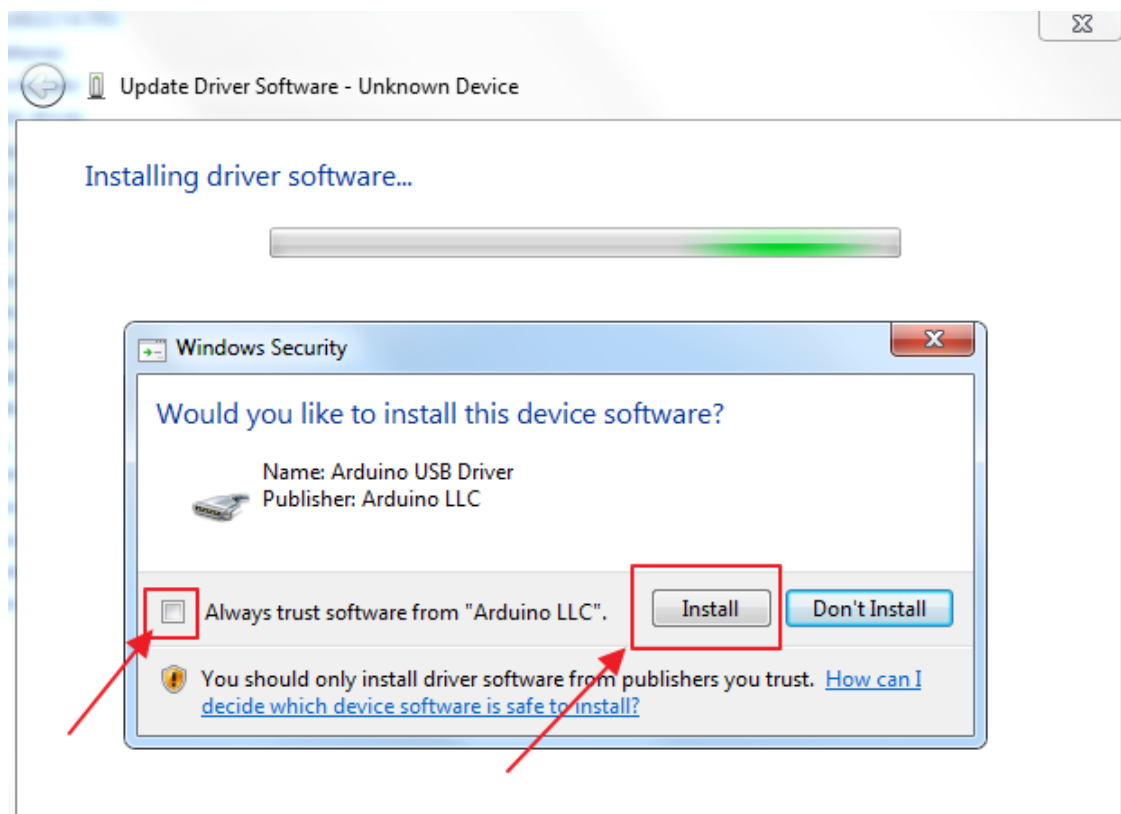
Arduino Installationsordner

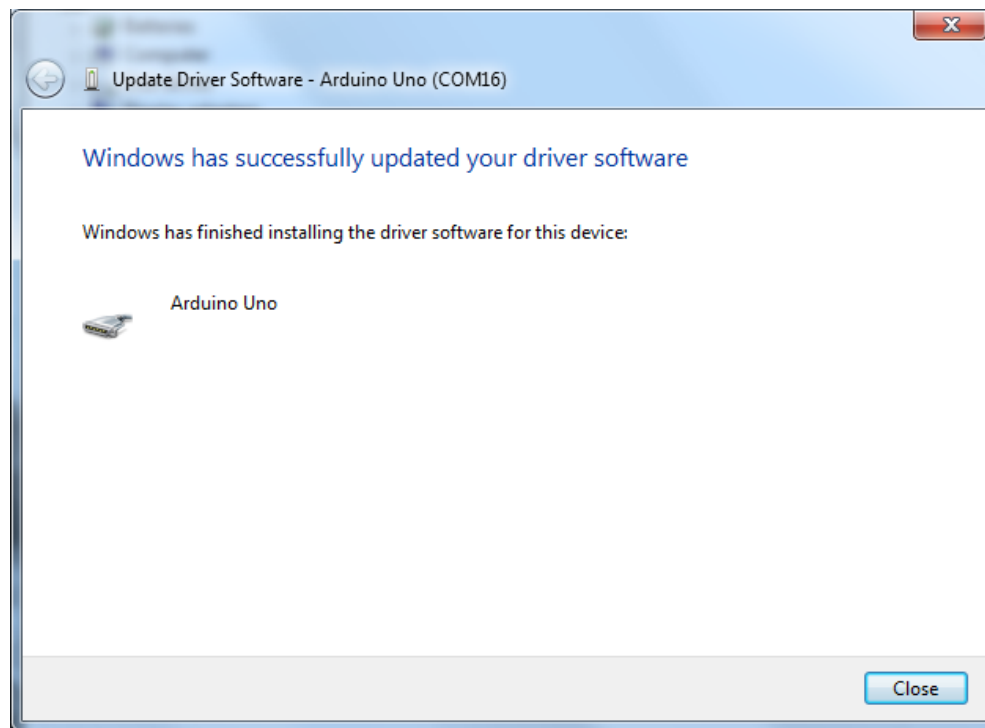
This PC > Win10 (C:) > Program Files (x86) > Arduino >				
Name	Date modified	Type	Size	
drivers	1/16/2017 9:18 AM	File folder		
examples	1/16/2017 9:18 AM	File folder		
hardware	1/16/2017 9:18 AM	File folder		
java	1/16/2017 9:18 AM	File folder		
lib	1/16/2017 9:18 AM	File folder		
libraries	1/16/2017 9:18 AM	File folder		
reference	1/16/2017 9:18 AM	File folder		
tools	1/16/2017 9:18 AM	File folder		
tools-builder	1/16/2017 9:18 AM	File folder		
arduino.exe	1/9/2017 7:35 PM	Application	395 KB	
arduino.l4j.ini	1/9/2017 7:35 PM	Configuration sett...	1 KB	
arduino_debug.exe	1/9/2017 7:35 PM	Application	392 KB	
arduino_debug.l4j.ini	1/9/2017 7:35 PM	Configuration sett...	1 KB	
arduino-builder.exe	1/9/2017 7:32 PM	Application	3,192 KB	
libusb0.dll	1/9/2017 7:32 PM	Application extens...	43 KB	
msvcpr100.dll	1/9/2017 7:32 PM	Application extens...	412 KB	
msvcpr100.dll	1/9/2017 7:32 PM	Application extens...	753 KB	
revisions.txt	1/9/2017 7:32 PM	Text Document	81 KB	
uninstall.exe	1/16/2017 9:18 AM	Application	404 KB	
wrapper-manifest.xml	1/9/2017 7:35 PM	XML Document	1 KB	

Wählen Sie den Arduino-Treiberordner aus

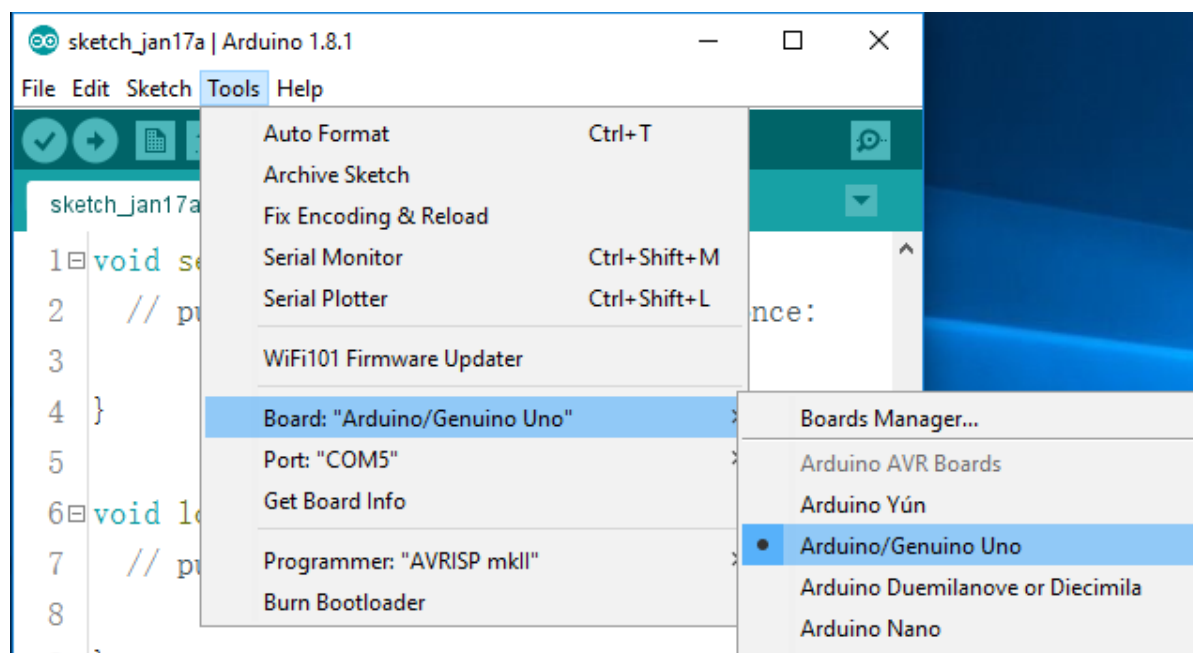


Installiere das Arduino USB Gerät

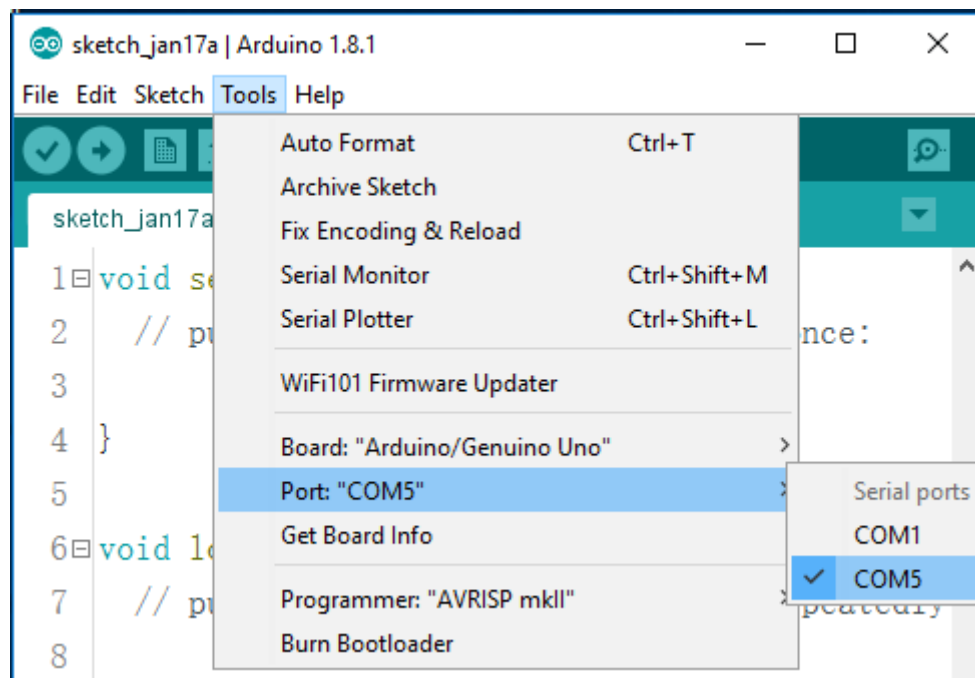




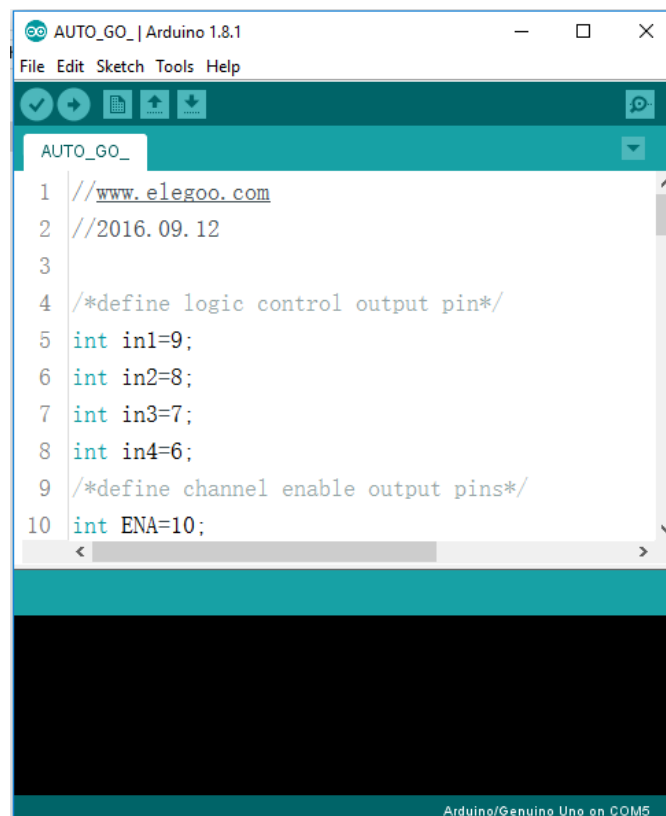
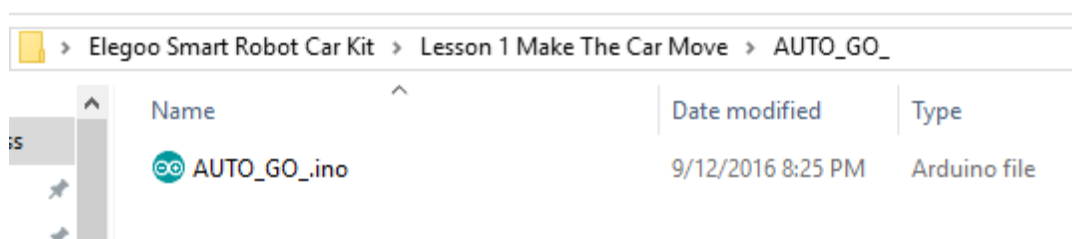
**STEP5: Nach dem Installieren des Treibers öffnen Sie bitte die IDE und klicken Sie dann auf Tools
--- Board --- Arduino / Genuino Uno**



STEP6: Klicken Sie auf Werkzeuge --- Port --- COMx (Arduino / Genuino Uno)



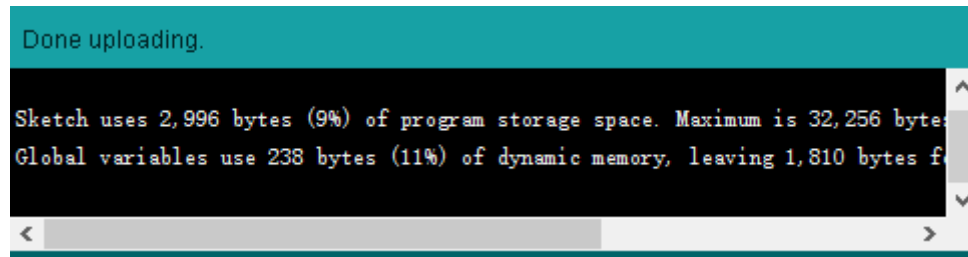
STEP7: Öffnen Sie die Datei AUTO_GO_ \ AUTO_GO_.ino und laden Sie die UNO-Controller-Karte hoch



(TIPPS: Das Bluetooth-Modul sollte herausgezogen werden, wenn Sie das Programm jedes Mal hochladen, oder kann sonst passieren, dass das Programm Probleme hat beim hochladen.)

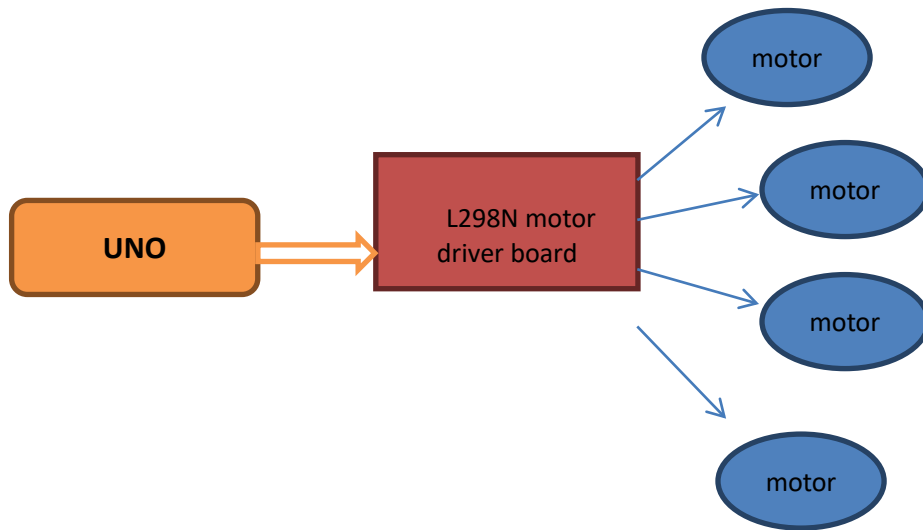


Das Bild oben zeigt, dass es erfolgreich hochgeladen wird.



STEP8: Schauen wir uns die Ergebnisse an. Laden Sie das Programm auf die UNO-Controller-Karte. Nach dem Trennen des Autos auf den Computer können Sie den Netzschalter einschalten und das Fahrzeug auf den Boden stellen. Dann werden Sie sehen, wie sich das Auto bewegt.

III. Beschreibung des Grundprinzips



So verwenden Sie L298N Motor Treiber Board

Die Definition der Anschlüsse auf der L298N-Karte wurde oben markiert. Die Motoren sollten mit der L298N-Platine verbunden sein, wie das Bild oben, und wenn Sie die Drehrichtung eines der Motoren entgegengesetzt vorfinden, ändern Sie bitte die Verbindungsposition der schwarzen und roten Drähte.

L298N GND ist mit Batteriekasten GND verbunden;

L298N VCC ist mit Batteriekasten VCC verbunden;

UNO-Board ist auch mit dem Batteriefach verbunden.

L298N 5V kann hier nicht an UNO 5V angeschlossen werden;

ENA und ENB steuern die Geschwindigkeit des rechten Motors und die Geschwindigkeit des linken Motors separat durch PWM.

IN1, IN2, IN3, IN4: IN1 und IN2 dienen zur Steuerung des rechten Motors, IN3 und IN4 dienen zur Steuerung des linken Motors. Über das Prinzip, schau bitte auf das Blatt unten: (Wir nehmen rechten Motor zum Beispiel)

ENA	IN1	IN2	DC MOTOR STATUS
0	X	X	STOP
1	0	0	BREMSSEN
1	0	1	VORWÄRTS
1	1	0	RÜCKWÄRTS
1	1	1	BARKING

IV. Das Auto fahren lassen

Der erste Schritt: Den Motor drehen lassen

Wir werden versuchen, den Motor ohne Drehzahlregelung zu bewegen. Weil es einfacher ist, ein Programm ohne Geschwindigkeitsregelung zu schreiben.

Zuerst einmal sehen wir die Verbindung des Motors der L298N Bord, wir verwenden die Arduino Stifte 5, 6, 7, 8, 9, 10 um das Auto zu steuern. Stift 6 und 7 steuern das rechte Rad. Stift 8 und 9 steuern das linke Rad. Stift 5 und 10 steuern ENA und ENB.

So ist die Verbindung wie unten...

L298N	V5 expansion board
ENA	5
IN1	6
IN2	7
IN3	8
IN4	9
ENB	11

Basierend auf dem oben angegebenen Blatt entwerfen wir zunächst ein einfaches Programm, um das rechte Rad um 0,5s in positiver Richtung zu drehen, 0,5s zu stoppen, 0,5s in negativer Richtung zu drehen und 0,5s zu stoppen. Und das Rad wird das ganze dann wiederholen.

Verbinde die UNO-Controller-Karte mit dem Computer, öffne die Datei right_wheel_rotation \ right_wheel_rotation.ino



Der Code sieht wie folgt aus:

```
/*In1 verbunden mit dem Pin 9,  
In2 verbunden mit dem Pin 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);    //Rechtes Rad vorwärts
  delay(500);
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,LOW);    // Rechtes Rad stoppt
  delay(500);
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,HIGH);   // Rechtes Rad rückwärts
  delay(500);
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,LOW);    // Rechtes Rad stoppt
  delay(500);
}

```

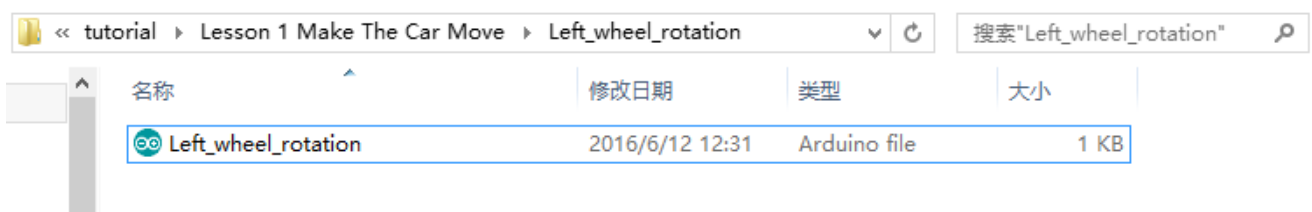
Laden Sie das Programm auf die UNO-Karte, trennen Sie es vom Computer und schalten Sie dann die Stromversorgung des Fahrzeugs ein. Sie werden sehen, dass das rechte Rad sich bewegt, wie Sie es erwartet haben.

Wenn sich das Auto nicht bewegt, drücken Sie die Reset-Taste auf der UNO-Karte.

Wenn sich die Bewegungsrichtung des Motors von der eingestellten Richtung unterscheidet, können Sie den Anschluss mit den schwarzen und roten Kabeln vom Motor auf die L298N-Platine ändern.

Dann lassen wir das linke Rad in gleicher Weise drehen.

Verbinde die UNO-Controller-Karte mit dem Computer, öffne die Datei Left_wheel_rotation \ Left_wheel_rotation.ino



Der Code sieht wie folgt aus:

```

/*In3 verbunden mit dem Pin 7 ,
  In4 verbunden mit dem Pin 6 , 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);          //Linkes Rad vorwärts
  delay(500);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,LOW);          // Linkes Rad stoppt
  delay(500);
  digitalWrite(IN3,HIGH);
  digitalWrite(IN4,LOW);          // Linkes Rad Rückwärts
  delay(500);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,LOW);          // Linkes Rad stoppt
  delay(500);
}

```

Laden Sie das Programm auf die UNO-Karte, trennen Sie es vom Computer und schalten Sie dann die Stromversorgung des Fahrzeugs ein. Sie werden sehen, dass das rechte Rad sich bewegt, wie Sie es erwartet haben.

Der zweite Schritt: Vorwärts und rückwärts fahren

Nach dem Debuggen des Autos können Sie Programme schreiben, um das Auto zu bewegen.

Unten ist die Art und Weise aufgezeigt, wie sich das Auto bewegt:

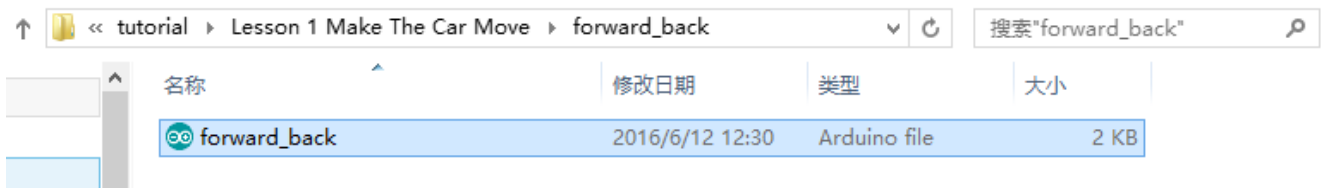
CAR	vorwärts	rückwärts	stop
Left wheel	vorwärts	rückwärts	stop
Right wheel	vorwärts	rückwärts	stop

CAR	Nach Links drehen	Nach Rechts drehen	stop
Left wheel	rückwärts	vorwärts	Stop
Right wheel	vorwärts	rückwärts	stop

Als nächstes werden wir ein einfaches Programm schreiben, um das Auto vorwärts fahren zu lassen für

0,5s, dann stoppen für 0.5s, dann rückwärts für 0.5s und dann stoppen für 0.5s.

Verbinde die UNO-Controller-Karte mit dem Computer, öffne die Datei forward_back \ forward_back.ino



Der Code sieht folgendermaßen aus:

```
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);          // Linkes rad dreht vorwärts
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);        // Rechtes Rad dreht vorwärts
  delay(500);
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,LOW);          //Linkes Rad bleibt stehen
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,LOW);          // Rechtes Rad bleibt stehen
```



```

delay(500);
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);          //Linkes Rad dreht Rückwärts
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);          // Rechtes Rad dreht Rückwärts
delay(500);
digitalWrite(IN1,LOW);
digitalWrite(IN2,LOW);          // Linkes Rad bleibt stehen
digitalWrite(IN3,LOW);
digitalWrite(IN4,LOW);          // Rechtes Rad bleibt stehen
delay(500);
}

```

Laden Sie das Programm auf die UNO-Karte, trennen Sie es vom Computer und schalten Sie dann die Stromversorgung des Fahrzeugs ein. Sie werden sehen, dass das rechte Rad sich bewegt, wie Sie es erwartet haben.

Der dritte Schritt: Schreiben Sie das Programm

Es kann ein schwieriges für Sie sein, das ganze Programm zu schreiben, um das Auto automatisch fahren zu lassen. Also trennen wir die Bewegungen in eine andere Funktion, zum Beispiel nach vorne und nach links. Und wenn wir das Programm im letzten Schritt schreiben, können wir die Funktion aufrufen. Als nächstes fangen wir an Programme für jede Bewegung zu schreiben:

Der Code sieht folgendermaßen aus:

```

/*****

```

```

****  Unterfunktion weiterleiten

```

```

Funktion: Vorwärts fahren

```

```

*****

```

```

***/ void forward( )

```

```

{

```

```

    digitalWrite(IN1,HIGH);

```

```

    digitalWrite(IN2,LOW);          //Linkes Rad vorwärts

```

```

    digitalWrite(IN3,LOW);

```

```

    digitalWrite(IN4,HIGH);          //Rechtes Rad vorwärts

```

```

}

```

```

/*****
****  Unterfunktion weiterleiten
Funktion:  Rückwärts fahren
*****/
void back( )
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);    //Linkes Rad rückwärts
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);    //Rechtes Rad rückwärts

```

```
}  
/*****  
**** turnLeft sub function  
Funktion: nach Links drehen  
*****/  
***/ void turnLeft( )  
{  
  digitalWrite(IN1,HIGH);  
  digitalWrite(IN2,LOW);      //Linkes Rad rückwärts  
  digitalWrite(IN3,HIGH);  
  digitalWrite(IN4,LOW);      //Rechtes Rad vorwärts  
}  
/*****  
**** turn Right sub function  
Funktion: nach rechts drehen:  
*****/  
***/ void turnRight( )  
{  
  digitalWrite(IN1,LOW);  
  digitalWrite(IN2,HIGH);      //Linkes Rad vorwärts  
  digitalWrite(IN3,LOW);  
  digitalWrite(IN4,HIGH);      //Rechtes Rad rückwärts  
}  
/*****  
**** stop sub function  
Funktion: Stop  
*****/  
***/ void _stop()  
{  
  digitalWrite(IN1,LOW);  
  digitalWrite(IN2,LOW);      //Linkes Rad stopp  
  digitalWrite(IN3,LOW);  
  digitalWrite(IN4,LOW);      //Rechtes Rad stop  
}
```

Der vierte Schritt: Automatisch fahren

Schließlich fangen wir an, das Programm zu schreiben, um das Auto automatisch fahren zu lassen: vorwärts fahren für 0.4s – rückwärts für 0.4s - links drehen für 0.4s - rechts drehen für 0.4s.

Verbinden Sie die UNO-Controller-Karte mit dem Computer, Öffnen Sie die Datei AUTO_GO_ \ AUTO_GO_.ino



Der Code dafür sieht folgendermaßen aus:

```
/*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;

/*deklariere die vorwärts funktion*/
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");
}

/* deklariere die rückwärts funktion*/
void _mBack()
{
```

```

digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);
digitalWrite(in1,LOW);
digitalWrite(in2,HIGH);
digitalWrite(in3,HIGH);
digitalWrite(in4,LOW);
Serial.println("Back");
}
/* deklariere die links drehen funktion*/
void _mleft()
{
digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);
digitalWrite(in1,HIGH);
digitalWrite(in2,LOW);
digitalWrite(in3,HIGH);
digitalWrite(in4,LOW);
Serial.println("Left");
}
/* deklariere die rechts drehen funktion*/
void _mright()
{
digitalWrite(ENA,HIGH);
digitalWrite(ENB,HIGH);
digitalWrite(in1,LOW);
digitalWrite(in2,HIGH);
digitalWrite(in3,LOW);
digitalWrite(in4,HIGH);
Serial.println("Right");
}
/* Setzen Sie Ihren Setup-Code hier, um ihn einmal durchlaufen zu lassen */
void setup() {
Serial.begin(9600); // Öffnen Sie den seriellen Port und stellen Sie die Baudrate auf 9600 ein

```



```

/* Setzen Sie die definierten Pins auf den Ausgang */
pinMode(in1,OUTPUT);
pinMode(in2,OUTPUT);
pinMode(in3,OUTPUT);
pinMode(in4,OUTPUT);
pinMode(ENA,OUTPUT);
pinMode(ENB,OUTPUT);
}

/* Setzen Sie Ihren Hauptcode hier, um ihn wiederholt laufen zu lassen */
void loop() {
  _mForward();
  delay(1000);
  _mBack();
  delay(1000);
  _mleft();
  delay(1000);
  _mright();
  delay(1000);
}

```

Laden Sie das Programm auf die UNO-Karte, trennen Sie es vom Computer und schalten Sie dann die Stromversorgung des Fahrzeugs ein. Sie werden sehen, dass das rechte Rad sich bewegt, wie Sie es erwartet haben.