

Lektion 2 Bluetooth Auto

Punkte in diesem Abschnitt

In den letzten Lektionen haben wir gelernt, wie die Komponenten von Auto alleine arbeiten können. Gut, aber nicht gut genug. In der Lektion werden wir etwas cooler machen, also wir werden Ihnen beibringen, alle Funktionen in einer einzigen App zusammenzufassen und DIY Projekte zu ermöglichen.

Lernteile:

- ◆ **Erfahren Sie, wie Sie das Bluetooth Modul und die Bluetooth APP**

verwenden

- ◆ **Lernen Sie wie man ein Auto per Bluetooth steuert**
- ◆ **Schreiben Sie Programme, um diese Funktion zu implementieren**

implementieren

Vorbereitung:

- ◆ **Ein Auto (mit Batterie)**
- ◆ **Ein USB Kabel**
- ◆ **Ein Bluetooth modul**
- ◆ **Smartphone (iOS oder Android)**

I . Bluetooth Modul

Bluetooth ist ein Industriestandard für die Datenübertragung zwischen Geräten über kurze Distanz per Funktechnik. Geräte nach den Standards der Bluetooth SIG senden als Short Range Devices in einem lizenzfreien ISM-Band zwischen 2,402 GHz und 2,480 GHz, und Aufbau persönlicher Bereichsnetzwerke (PANs). Es gibt auch HF-Protokolle wie ZigBee und Wi-Fi.



Dieses Smart Car Kit enthält das Bluetooth Modul "HC-08", das serielle Daten per Bluetooth an andere Geräte senden kann.

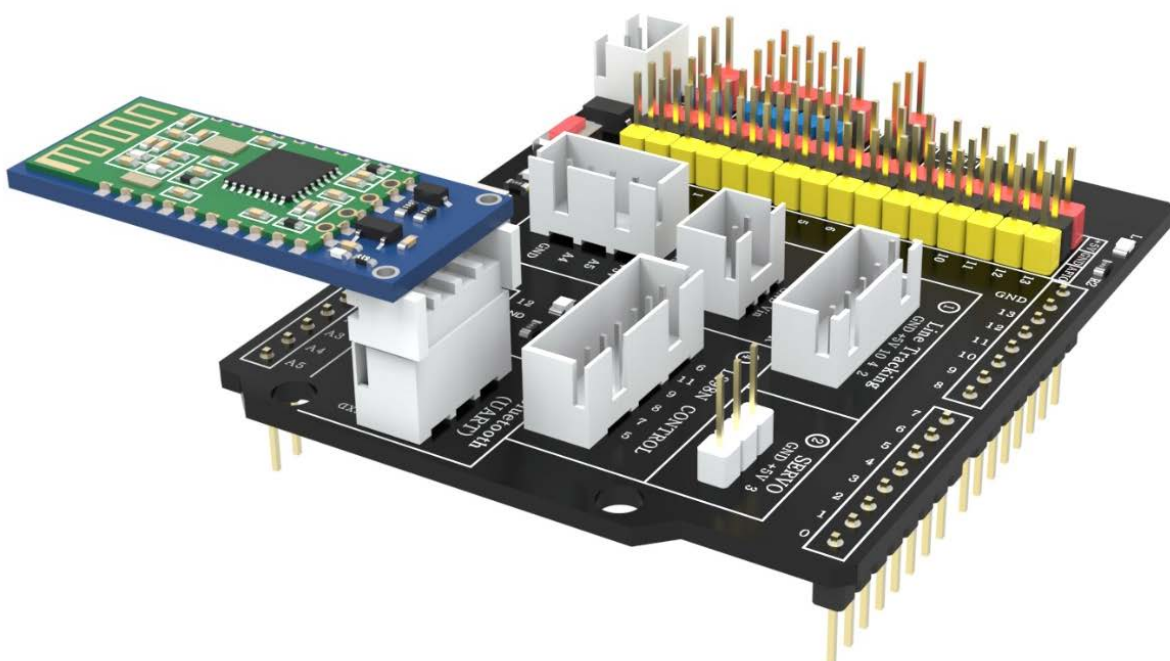


HC-08 kommuniziert mit UNO über das Pin **RX/TX** des hields.

***WARNING:**

The RX/TX pin will be occupied when the sketch is uploading.

Please unplug the Bluetooth module until done uploading.

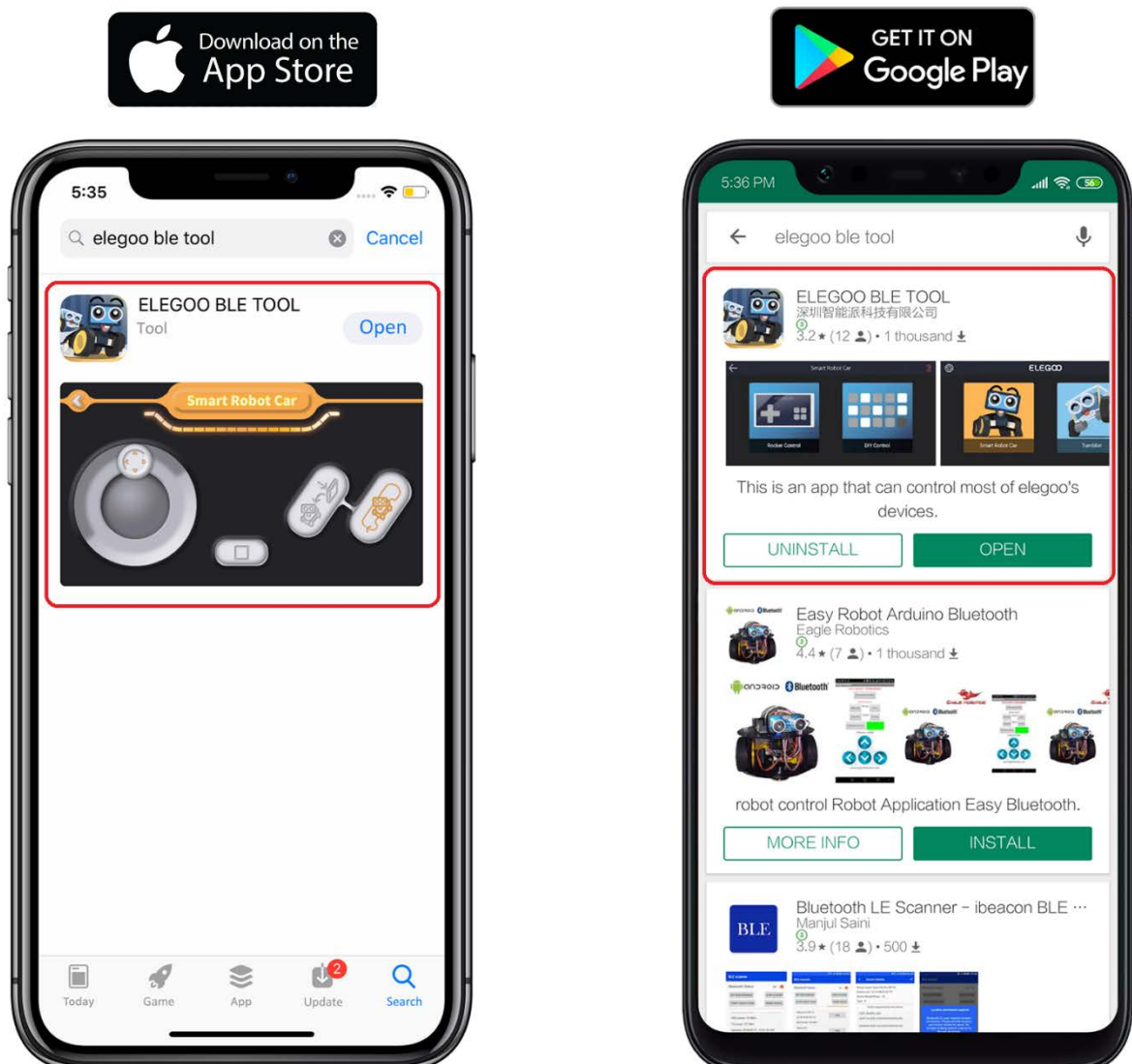


II. Anfang mit Bluetooth APP

Bevor wir starten, verbinde das HC-08 Bluetooth Modul mit dem Erweiterungsboard und schalte den Strom ein.

SCHRITT 1: Installiere die APP

Suche die neueste "ELEGOO BLE TOOL" im [App Store](#) and [Google Play](#) und installiere sie.



SCHRITT 2: App Einstellungen

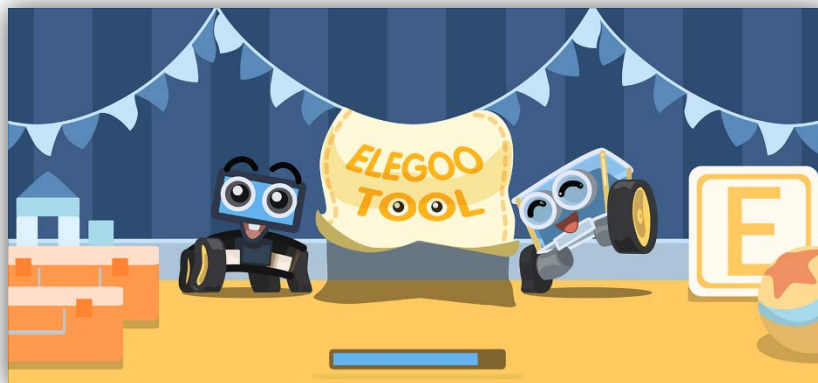
Schalten Sie zuerst die Bluetooth-Funktion Ihres Mobiltelefons ein.



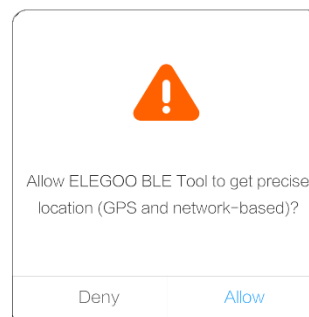
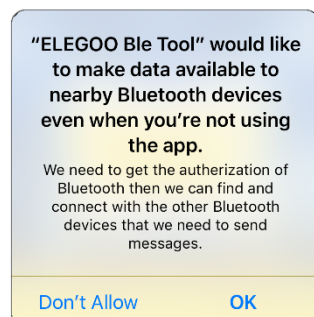
Please note:

The Smart Cars CANNOT be paired in settings page, it can only be connected with "Elegoo BLE Tool" App.

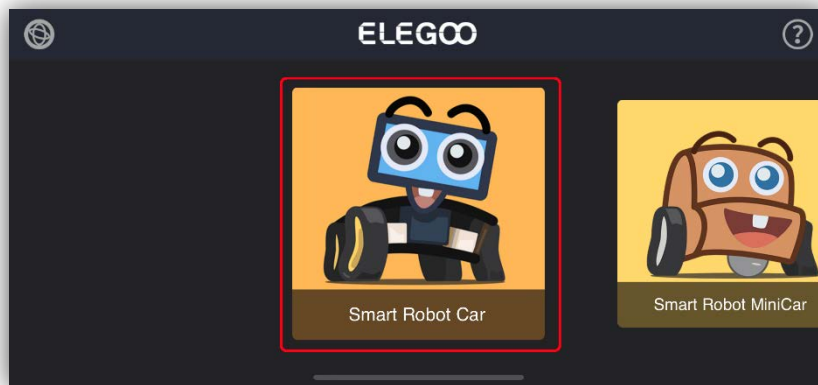
❖ Öffne die App "Elegoo BLE Tool"



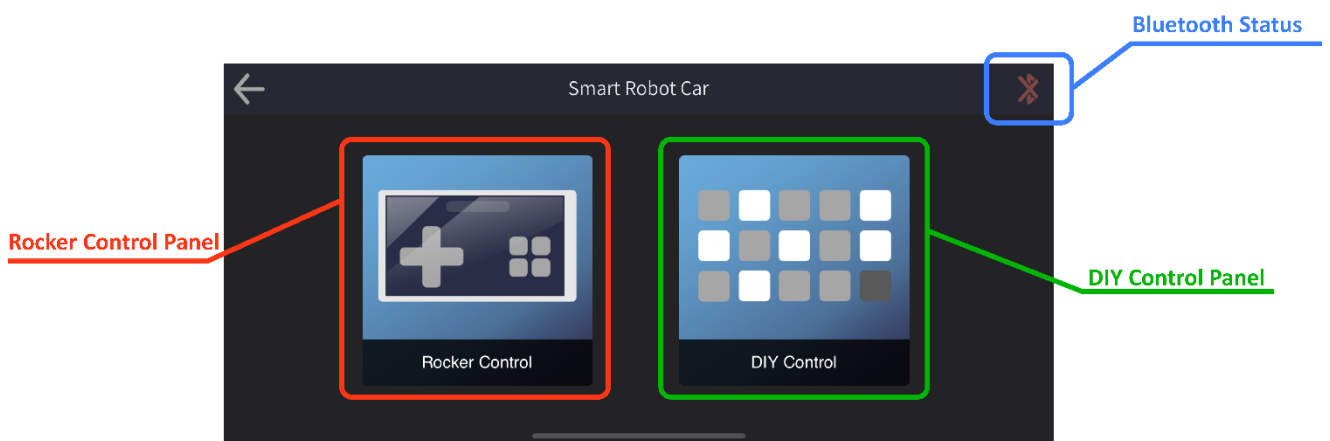
The first time you open the app, the system will pop up a permission request. Please select the "Allow" option to ensure the app works.



- ❖ Wähle "Smart Robot Car".



- ❖ Die Funktion des Icons auf der Smart Robot Car Homepage:



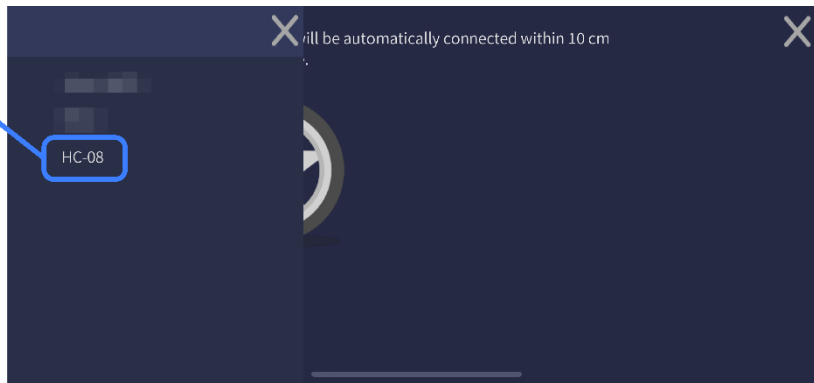
Klick auf “”, du gelangst in Bluetooth-Suche Interface.

Lege das Handy in die Nähe des Fahrzeugs (innerhalb von 10 cm). Die App erkennt das Smart Auto und stellt automatisch eine Verbindung her.

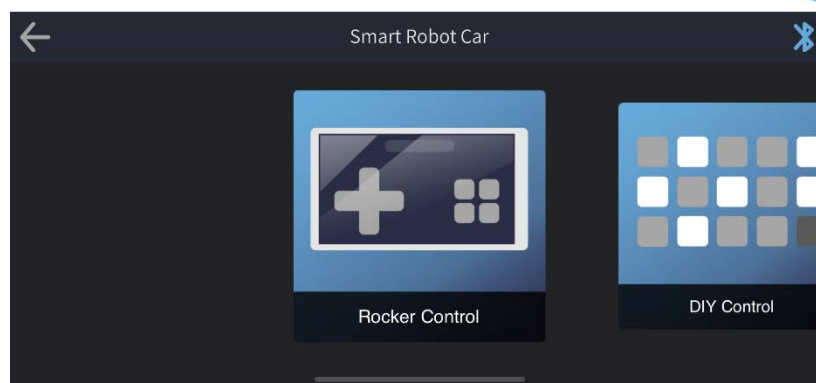


Oder öffne Bluetooth Geräte durch Tippen des Icon “” oben links und wähle “HC-08” um eine Bluetooth-Verbindung manuell herzustellen.

Select "HC-08" to connect Smart Car manually

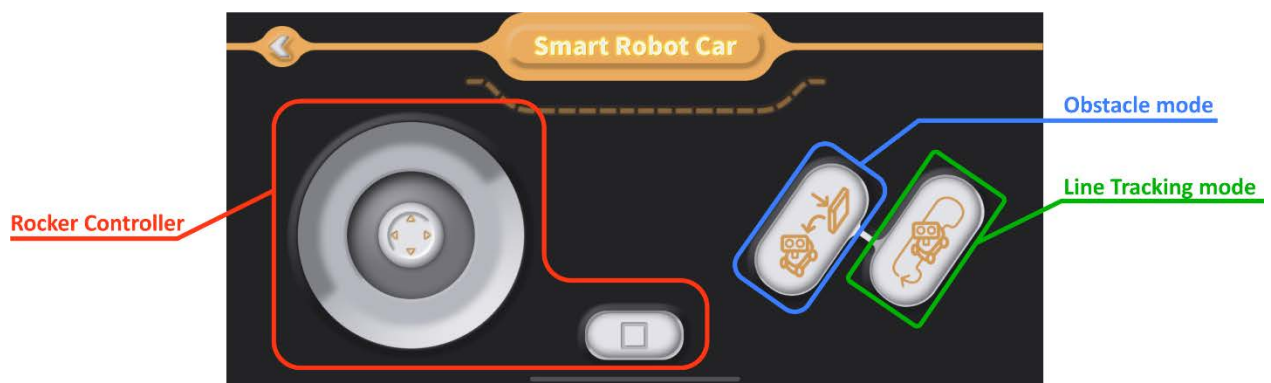


❖ Wenn die Verbindung erfolgreich ist, wird das Bluetooth-Status-Icon blau!



Toll, jetzt ist das Smart Car bereits mit der App verbunden, wir können es auf zwei Arten steuern.

Das Rocker Control Bedienfeld in "Elegoo BLE Tool" App.



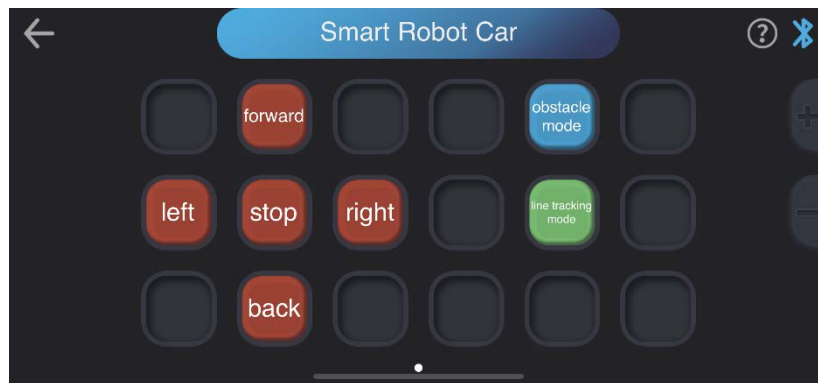
Die Funktionen des Rocker Control Bedienfelds lassen sich in drei Teile unterteilen:

Rocker Controller: Die Bewegung des Autos lässt sich frei und flüssig steuern und beim Drücken der quadratischen Taste stoppen.

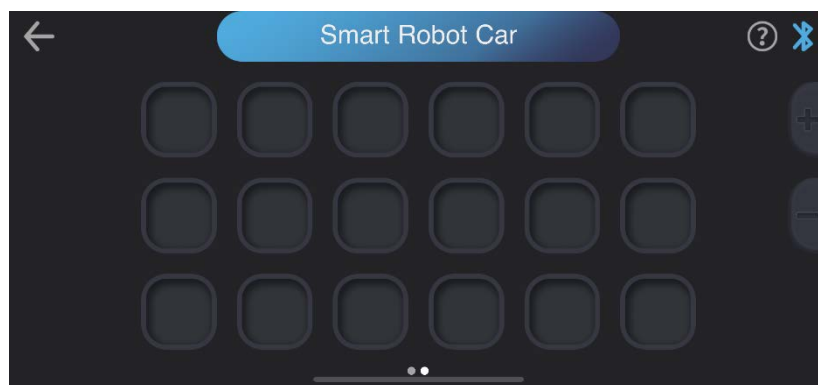
Hindernis Modus: Es wird ins Hindernisvermeidung-Modus umgewandelt (siehe Lektion 2).

Linienverfolgung Modus: Es wird ins Linienverfolgung-Modus umgewandelt (siehe Lektion 3).

❖ Das DIY-Bedienfeld in “Elegoo BLE Tool” App.

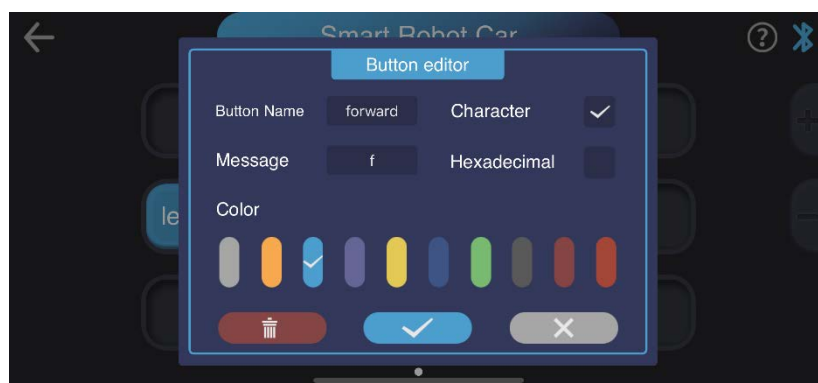


In der Standardeinstellung sind nur einige Rest-Tasten vorhanden, du sollst sie definieren.



Drücke die zu definierende Taste lange, wird es ein Optionsfeld “Taste Editor” erscheinen (siehe unten). Du sollst die Anweisung für “Button Name”, “Message” eingeben und die Farbe wählen.

(Alle voreingestellten Messages sind vom Zeichentyp. Sie müssen also nur die Option "Character" aktivieren.)



❖ Die Vergleichstabelle der Standard Messages und ihre Funktionen befinden sich unten.

Funktion	VORWÄRTS	RÜCKWÄRTS	LINKS	RECHTS	STOP	Linienverfolgungsmodus	Hindernismodus
Message	f	b	l	r	s	1	2

*Groß- und Kleinschreibung

III. Smart Car Core Code

In dieser Lektion erfahren wir alle Funktionen des Smart Autos, daher müssen wir die Codes aus allen vorherigen Lektionen in einen kompletten Sketch zusammenfassen.



Öffne diese Datei durch:

"\Elegoo Smart Robot Car Kit V2.0\Lesson 2 Bluetooth Cat\SmartCar_Core"

Ein großer Teil des Codes wurde in den vorherigen Lektionen erläutert. Hier wird der Bluetooth-betreffende Teil aus dem kompletten Sketch erläutert.

Bluetooth Daten Code Ausschnitt:

```
void getBTData() {  
  if(Serial.available()) {  
    switch(Serial.read()) {  
      case 'f': func_mode = Bluetooth; mov_mode = FORWARD; break;  
      case 'b': func_mode = Bluetooth; mov_mode = BACK; break;  
      case 'l': func_mode = Bluetooth; mov_mode = LEFT; break;  
      case 'r': func_mode = Bluetooth; mov_mode = RIGHT; break;  
      case 's': func_mode = Bluetooth; mov_mode = STOP; break;  
      case '1': func_mode = LineTeacking; break;  
      case '2': func_mode = ObstaclesAvoidance; break;  
      default: break;  
    }  
  }  
}
```

Dieser Codeblock definiert die Funktion "getBTData()". Wenn serielle Port Daten erfasst werden, wird Funktion erfolgreich ausgeführt. Die erkennt den Inhalt dieser Daten und konvertiert sie in den Befehl der Funktion der nächsten Ebene.

Zum Beispiel: Wenn du die Wippe im Rocker-Bedienfeld nach vorne schiebst, sendet dein Smartphone über Bluetooth einen Buchstaben 'f' an das Auto. Nach Erhalt dieses Buchstaben aktiviert der UNO Mikrocontroller die Funktion "getBTData()" und übergibt diese Funktion. Der Buchstabe 'f' ist in den Bewegungsmodus " FORWARD " umgewandelt.

Bluetooth Befehl Code:

```
void bluetooth_mode() {
  if(func_mode == Bluetooth){
    switch(mov_mode){
      case FORWARD: forward(); break;
      case BACK:     back();    break;
      case LEFT:     left();    break;
      case RIGHT:    right();   break;
      case STOP:     stop();    break;
      default: break;
    }
  }
}
```

Nach der letzten Funktion ist das Bluetooth-Signal bereits in einen direkten Befehl umgewandelt, der von "bluetooth_mode ()" empfangen wird und bestimmt, welche Funktionen ausgeführt werden.

Zum Beispiel: In der letzten Funktion wurde der Buchstabe 'f' in " FORWARD " umgewandelt, "bluetooth_mode()" wird diesen Befehl in die Funktion "forward()" umwandeln.

Bewegung Code:

```
void forward(bool debug = false){
  analogWrite(ENA, carSpeed);
  analogWrite(ENB, carSpeed);
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);
  if(debug) Serial.println("Go forward!");
}

//ect...

void stop(bool debug = false){
  digitalWrite(ENA, LOW);
  digitalWrite(ENB, LOW);
  if(debug) Serial.println("Stop!");
}
```

Am Ende befinden sich die Funktionen zur Steuerung des Fahrzeugs bei Bewegung. Sie steuern den Output des UNO Boards, wodurch die Motoren betrieben werden.