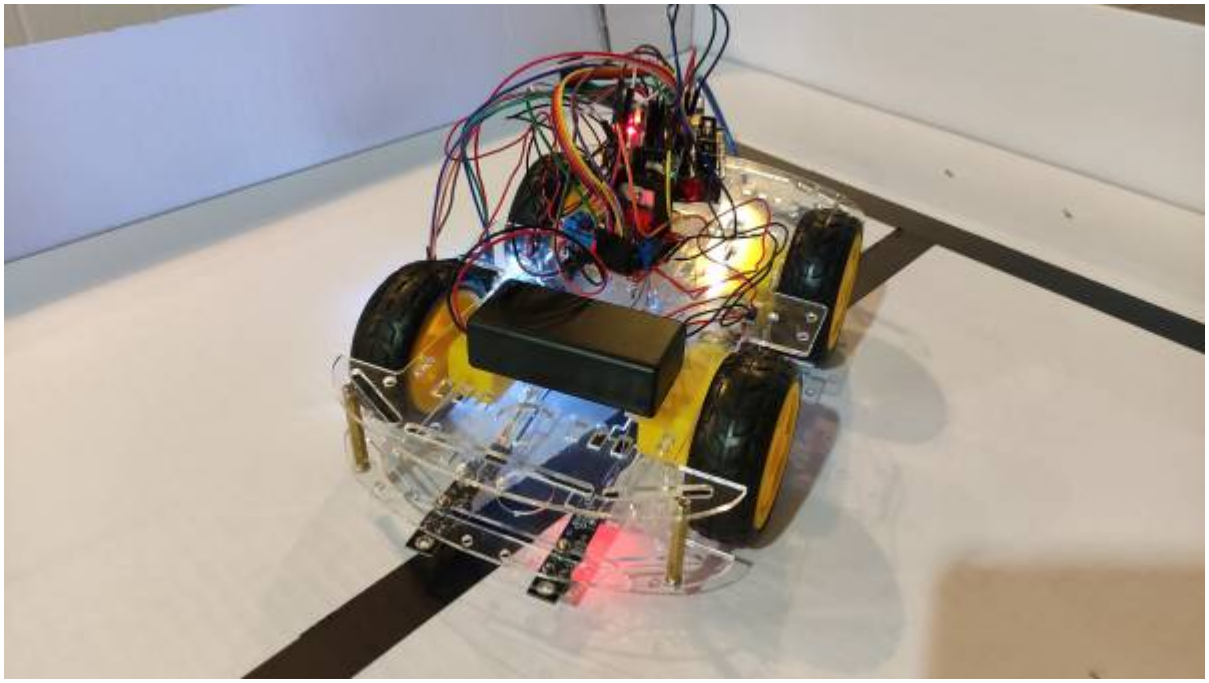
# Color Car Projekt



Video

**Durchgeführt von**

- Laura Panke
- Yago Agustin Illing Arizti (397505)
- Konrad Joachim Kloska (400913)
- Maximilian Thomas Frank Bürger
- Yunli Chen
- Max Dallinger

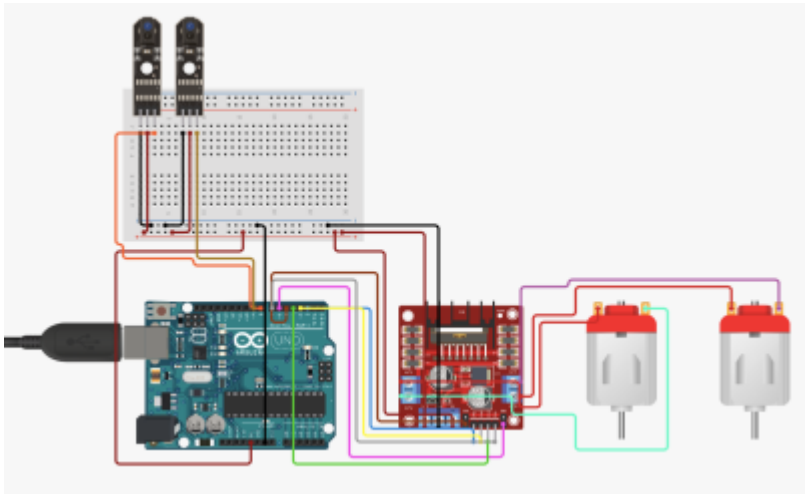
# Aufbau

Wir haben für dieses Projekt folgende Bauteile verwendet:

| **Komponente** | **Modell** | Stückzahl |
|---|---|---|
| Arduino Uno | Uno Atmega 238 | 1 |
| Motortreiber | L298N | 1 |
| IR Entfernungssensor | TCRT5000 | 2 |
| DC Motor | | 4 |
| Farbsensor | TCS3200 | 1 |
| Infrarot IR Diode | TSOP382 | 1 |

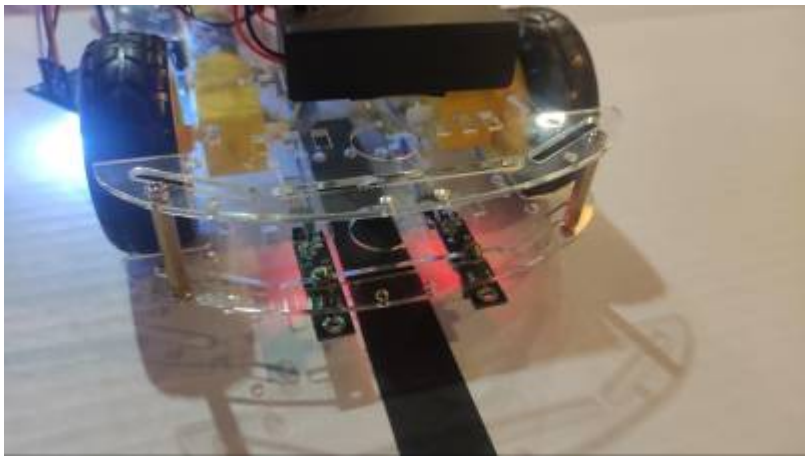
Das Fahrzeuggestell und die Räder haben wir als set bestellen lassen.

Die Motoren und die Sensoren haben wir, wie in der Abbildung zu sehen, angeschlossen. Auf die einzelnen Sensoren und den Motor Driver gehen wir später genauer ein.



# Sensoren

## Infrarotsensoren



Die Infrarotsensoren überprüfen, ob die Fläche auf die Sie zeigen Licht reflektiert. Absorbiert eine Fläche das Licht (in unserem Fall bei einem schwarzen Klebeband) sendet der Sensor über die digitalen Pins des Arduinos den Wert 1 an den Arduino. Andernfalls wird bei einer Fläche, welche das Licht reflektiert, der Wert 0 an den Arduino gesendet.

Wir platzieren das Fahrzeug so, dass der Klebestreifen zwischen den Sensoren liegt. Werden für beide Sensoren eine 0 ausgelesen, dann fährt das Fahrzeug geradeaus.

Liest der rechte Sensor eine 1 und der linke eine 0, dann fährt das Fahrzeug nach links. Analog fährt das Fahrzeug nach links bei einem rechten Sensor Input von 1 und einem linken Sensor Input von 0.

## Farbsensensor (TCS3200/230)

Der Farbsensor besitzt drei Fotodioden (Rot, Grün und Blau) Mit dem Arduino werden die Werte für die drei Fotodioden ausgelesen.

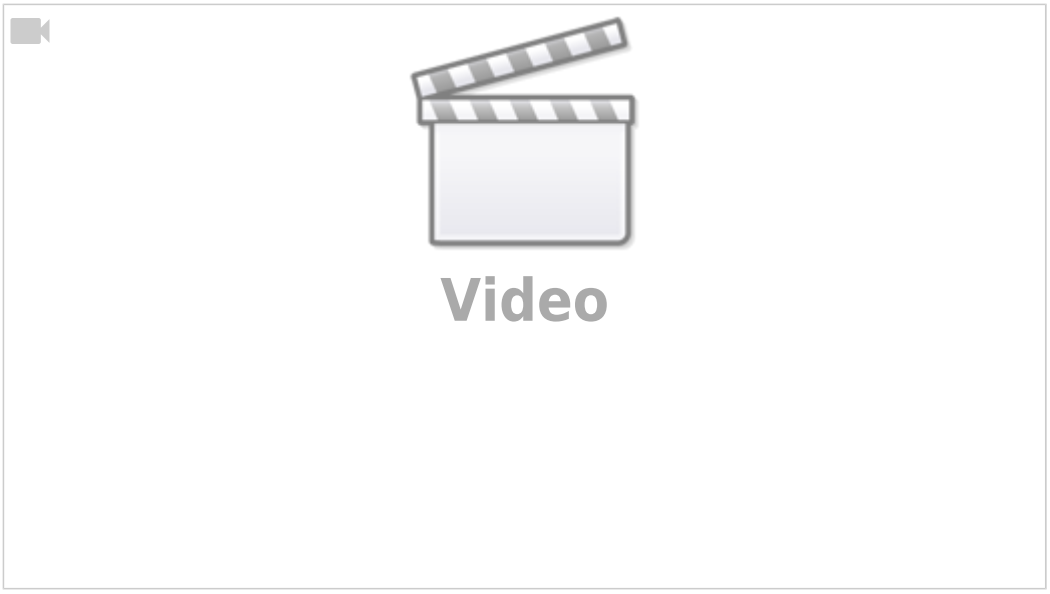
```
75    digitalWrite(s2,LOW);
76    digitalWrite(s3,LOW);
77    int red = GetData();
```

Die S2/S3-Pegel legen fest, welcher Satz von Fotodioden verwendet wird LOW/LOW steht für ROT, LOW/HIGH steht für Blau und HIGH/HIGH steht für Grün.

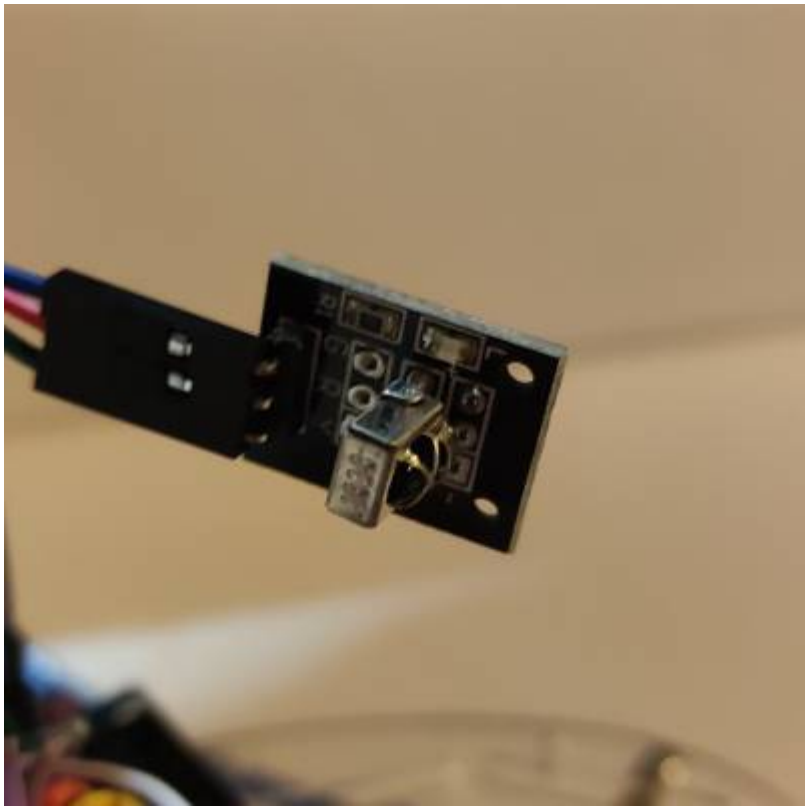
```
291  int GetData(){
292      data=pulseIn(out,LOW);
293      //Serial.print(data);
294      //Serial.print("\t");
295      delay(20);
296      return data;
297  }
298
```

Hier wird gewartet, bis „out“ den Wert LOW annimmt, dann wird die Dauer gemessen und gestoppt, wenn „out“ wieder auf HIGH ist. Es wird eine Zeitdauer gemessen, die mit der Frequenz zusammenhängt, da der Sensor eine von der Farbe abhängigen Frequenz angibt. Je höher die Frequenz, desto kürzer die Dauer.

Wir haben den Farbsensor auf ein Rotes Klebeband gehalten die Werte für Rot, Grün und Blau notiert. Werden vom Sensor wieder ähnliche Werte registriert, dann hält das Fahrzeug.



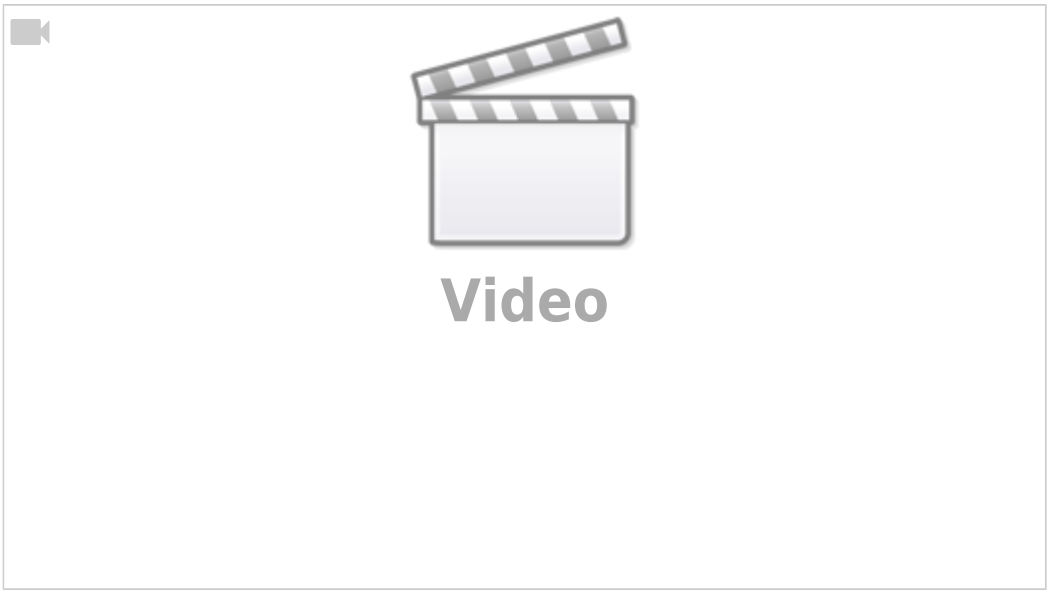

## IR - Receiver und Fernbedienung



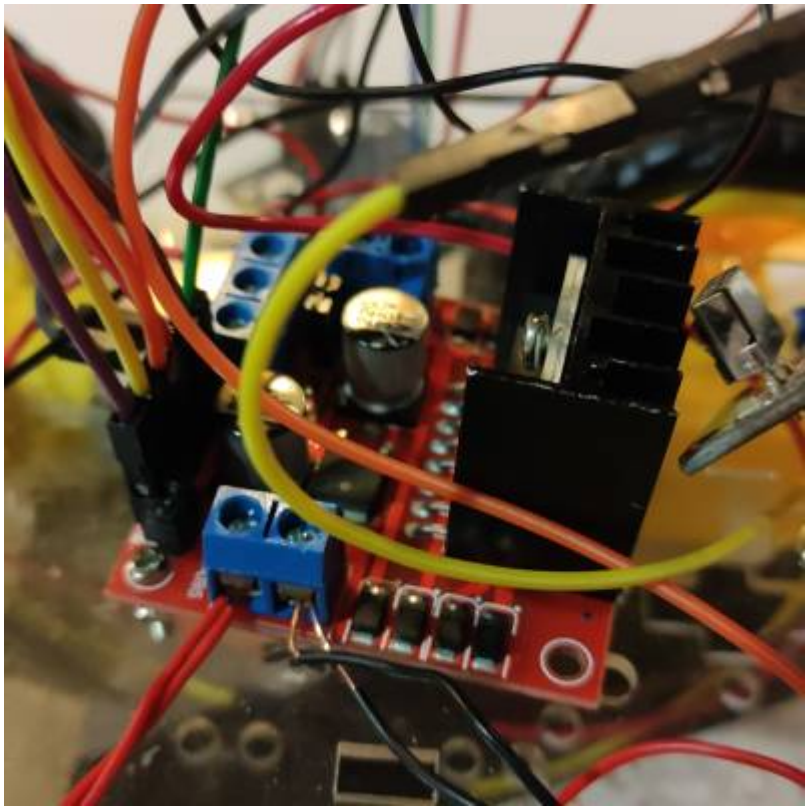
Mit dem IR - Receiver und der Library IRremote lesen wir Anfragen, der Fernbedienung aus. Bei dem Erhalten vom Signal des Power-Buttons wird das Fahrzeug an und aus geschaltet.

Wird vom Fahrzeug eine Kreuzung erkannt, fährt das Fahrzeug erst weiter, sobald ein korrektes Signal von der Fernbedienung erkannt wird.

- ListenpunktRückspulen (Linkskurve)
- Pause/Play (Vorwärtsfahren)
- Vorspulen (Rechtskurve)

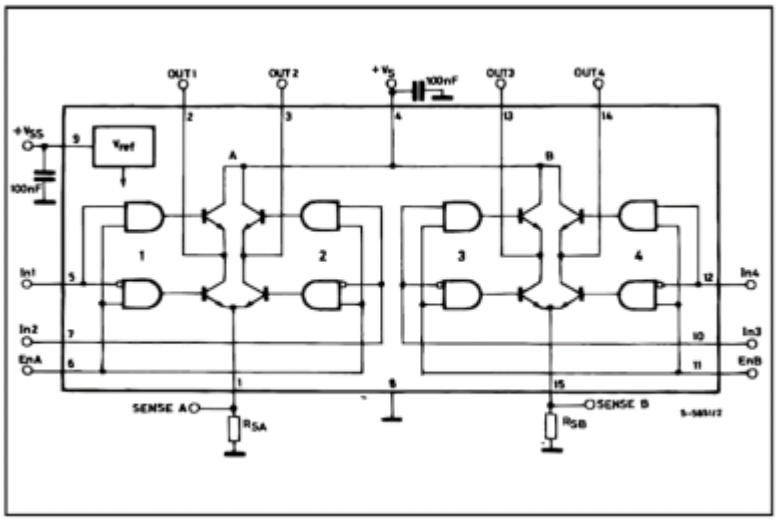



## Motor driver (L298N)



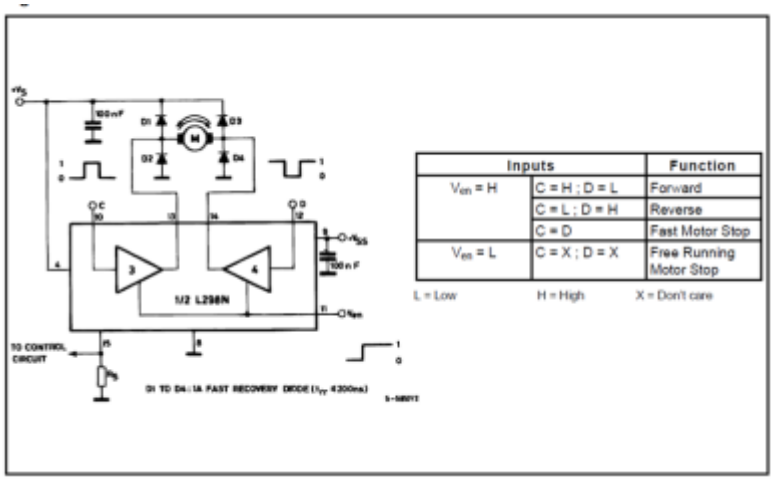
Um das Motor mit dem Mikrocontroller zu steuern, wird ein Motortreiber zwischen Motor und Mikrocontroller verwendet, weil die GPIO von Arduino nur maximal 40 mA liefern können. Hier wird ein integrierter Motortreiber L298N ausgewählt. Der Motortreiber besitzt 2 H-Brücken und eine Treiberschaltung, die gleichzeitig 2 bidirektionalen Kanäle oder 4 unidirektionalen Kanäle bietet.

Der Motortreiber verfügt über 4 Ausgänge 4 Ausgänge und 4 Eingänge das Anschließen der Motoren, 2 Enable Eingänge für jeweiligen Kanal, einen Eingang für 12V Spannungsversorgung und einen Ausgang für 5V Spannungsversorgung. Die folgende Abbildung zeigt den Aufbau des Treibers.




Die Motoren sollen sich in beide Richtungen drehen können, dafür habe wir eine bidirektionale Konfiguration ausgewählt. Jeder Kanal hat jeweils zwei Ausgänge die durch zwei Eingänge gesteuert werden. Ein Enable Pin sorgt für das Ein- und Ausschalten eines Kanals. Die zwei Ausgänge eines Kanals werden an den DC Motor angeschlossen. Um den Motor drehen zu lassen, muss einer der Eingänge auf HIGH gesetzt werden, und der andere Eingang auf LOW. Um die Geschwindigkeit zu steuern, wird einer der Enable Eingänge durch PWM moduliert. Die folgende Abbildung zeigt ein

Beispiel für das Steuern des Motors.




| Inputs | | Function |
|---|---|---|
| $V_{en}$ = H | C = H ; D = L | Forward |
| | C = L ; D = H | Reverse |
| | C = D | Fast Motor Stop |
| $V_{en}$ = L | C = X ; D = X | Free Running Motor Stop |

L = Low H = High X = Don't care

From:
http://www.labprepare.tu-berlin.de/wiki/ - **Project Sci.Com Wiki**

Permanent link:
**http://www.labprepare.tu-berlin.de/wiki/doku.php?id=color_car&rev=1648539372**

Last update: **2022/03/29 09:36**