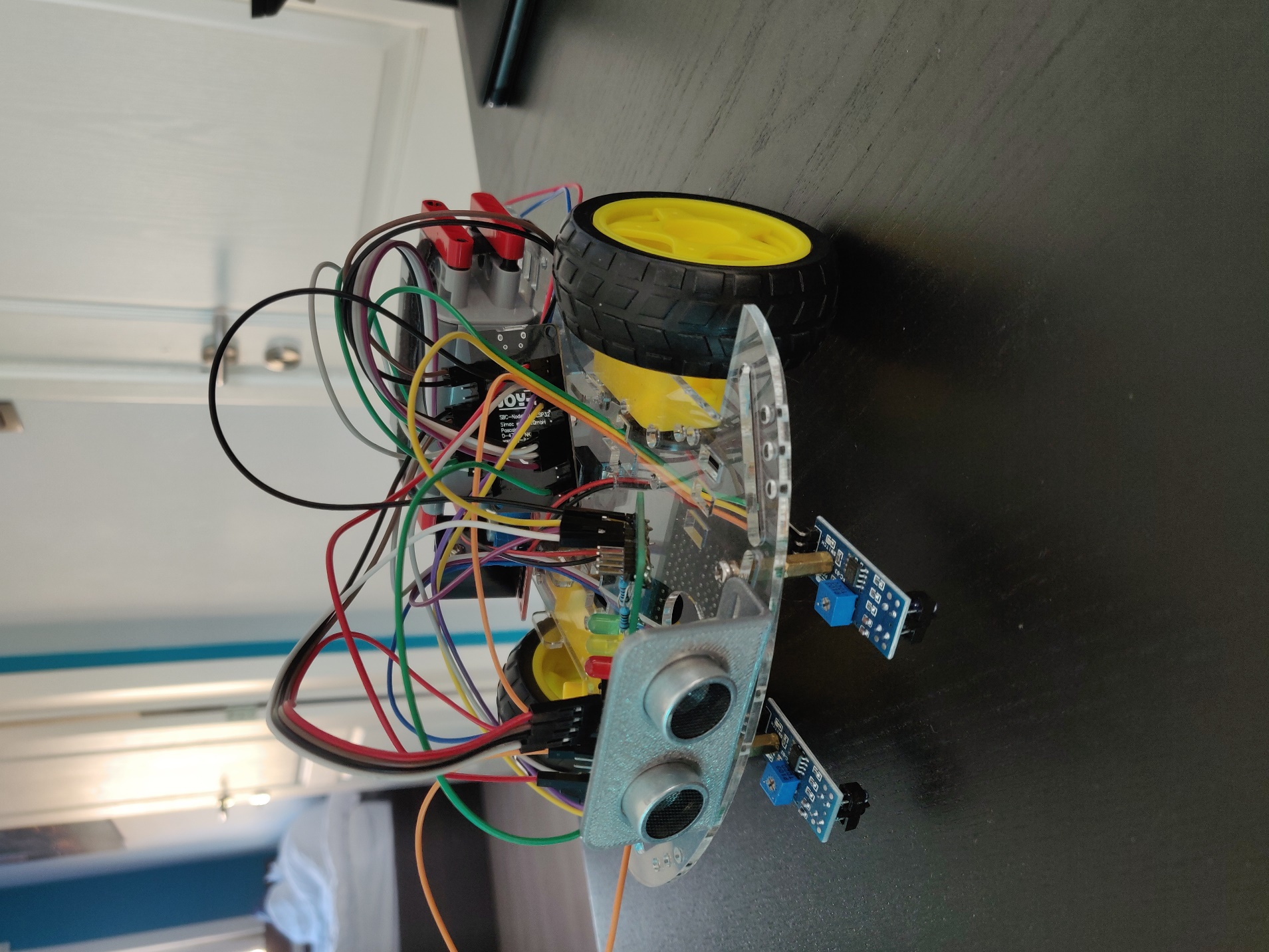
Iot Project : Line following car



Inhoud

[Electronica component 3](#_Toc73187030)

[Benodigdheden: 3](#_Toc73187031)

[ESP-32 3](#_Toc73187032)

[Smart Arduino car voor het frame 3](#_Toc73187033)

[Motor sturing 4](#_Toc73187034)

[IR sensoren 4](#_Toc73187035)

[Ultrasone sensor 5](#_Toc73187036)

[Jumper wires 5](#_Toc73187037)

[Batterij 5](#_Toc73187038)

[Wat zal het kosten? 6](#_Toc73187039)

[Beschrijving aansluitingen 7](#_Toc73187040)

[Code 10](#_Toc73187041)

[Bibliotheken 10](#_Toc73187042)

[WIFI 10](#_Toc73187043)

[Declaratie pinnen 11](#_Toc73187044)

[Ingangen en uitgangen 12](#_Toc73187045)

[Begin void loop 13](#_Toc73187046)

[Optioneel 13](#_Toc73187047)

[Vooruit 14](#_Toc73187048)

[Stop 15](#_Toc73187049)

[Timer 20 seconden 15](#_Toc73187050)

[Draaien 16](#_Toc73187051)

[Blynk 17](#_Toc73187052)

[schema 18](#_Toc73187053)

# Electronica component

## Benodigdheden:

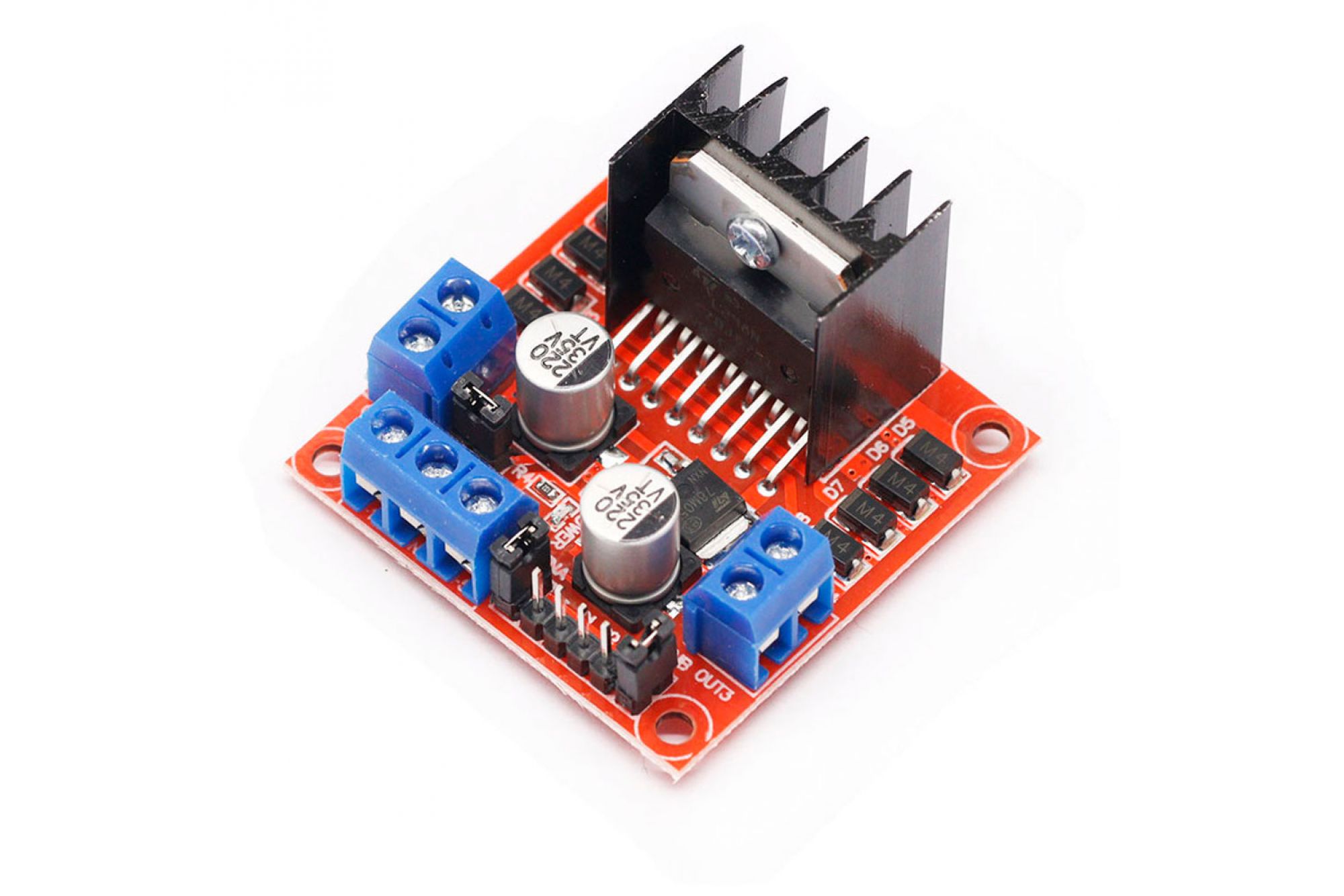
### ESP-32



### Smart Arduino car voor het frame

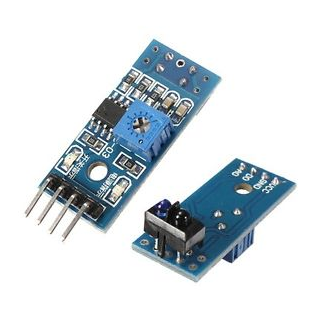


### Motor sturing



Voor de motor sturing heb ik een L298N gebruikt deze kan 2 motoren aansturen. En kan heeft een voedingspanning die 12v kan zijn en kan ook onze esp32 voeden door gebruik te maken van de 5V pin op het bord.

### IR sensoren



Ik heb in mijn auto 2 sensoren gebruikt.

### Ultrasone sensor



### Jumper wires



### Batterij

Een batterij houder bevind zich al reeds in de auto kit maar ik gebruik voor mijn project een iets grotere voedingsbron.

# Wat zal het kosten?

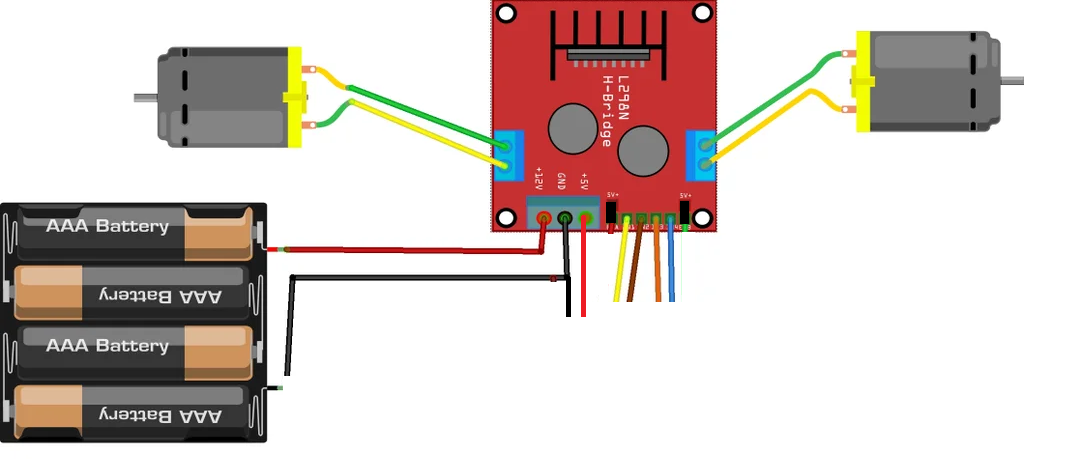
|  |  |  |  |
| --- | --- | --- | --- |
| Componenten | Prijs per stuk | Aantal nodig | Totaal prijs |
| ESP-32 | € 12,89 | 1 | € 12,89 |
| Body kit | € 8,99 | 1 | € 8,99 |
| L298N | € 2,59 | 1 | € 2,598 |
| IR sensor | € 1,19 | 2 | € 1,195 |
| Ultrasone sensor | € 2,45 | 1 | € 2,45 |
| Totaal |  |  | € 28,21 |

# Beschrijving aansluitingen

Nu dat we alles hebben gaan we is kijken naar het aansluitschema:

Eerst even de pin layout van de ESP-32

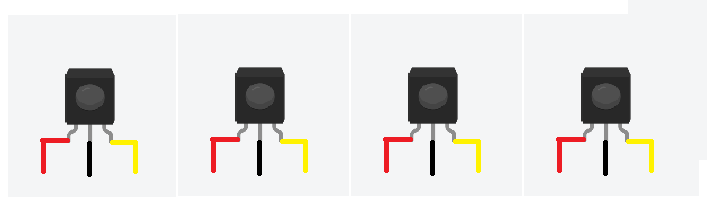
De motor sturing aansluiting:



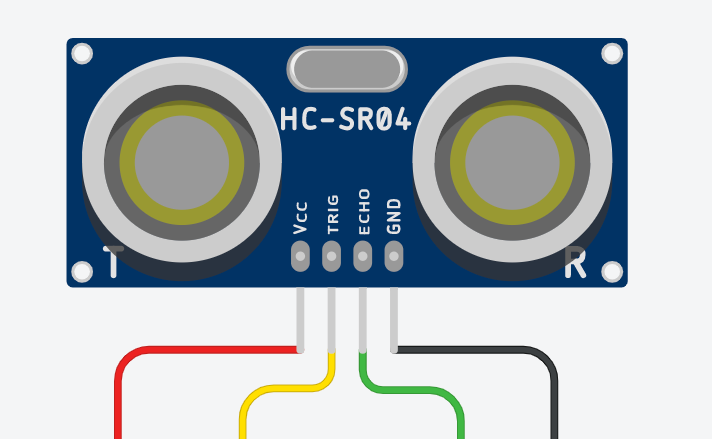
gnd Vin 33 21 19 18 5 32

De IR Sensoren

We gebruiken van de Ir sensoren de digitale uitgang.

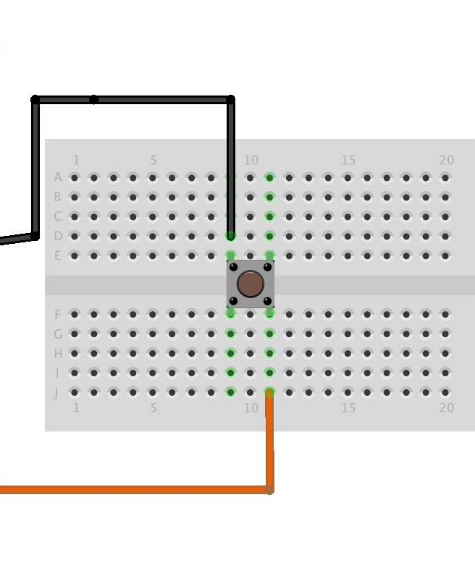


3,3v gnd 25 3,3v gnd 26

 De ultrasone sensor

3,3v 27 14 gnd

En tot slot de bypass drukknop



3,3V

22

Met al deze informatie en aansluitingen zijn we klaar met de hardware en

kunnen we overgaan naar het volgende stuk de arduino code.

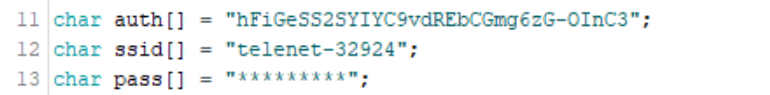
# Code

## Bibliotheken



Hier zetten we alle bibliotheken klaar en definiëren we de pinnen voor de ultrasone sensor.

## WIFI

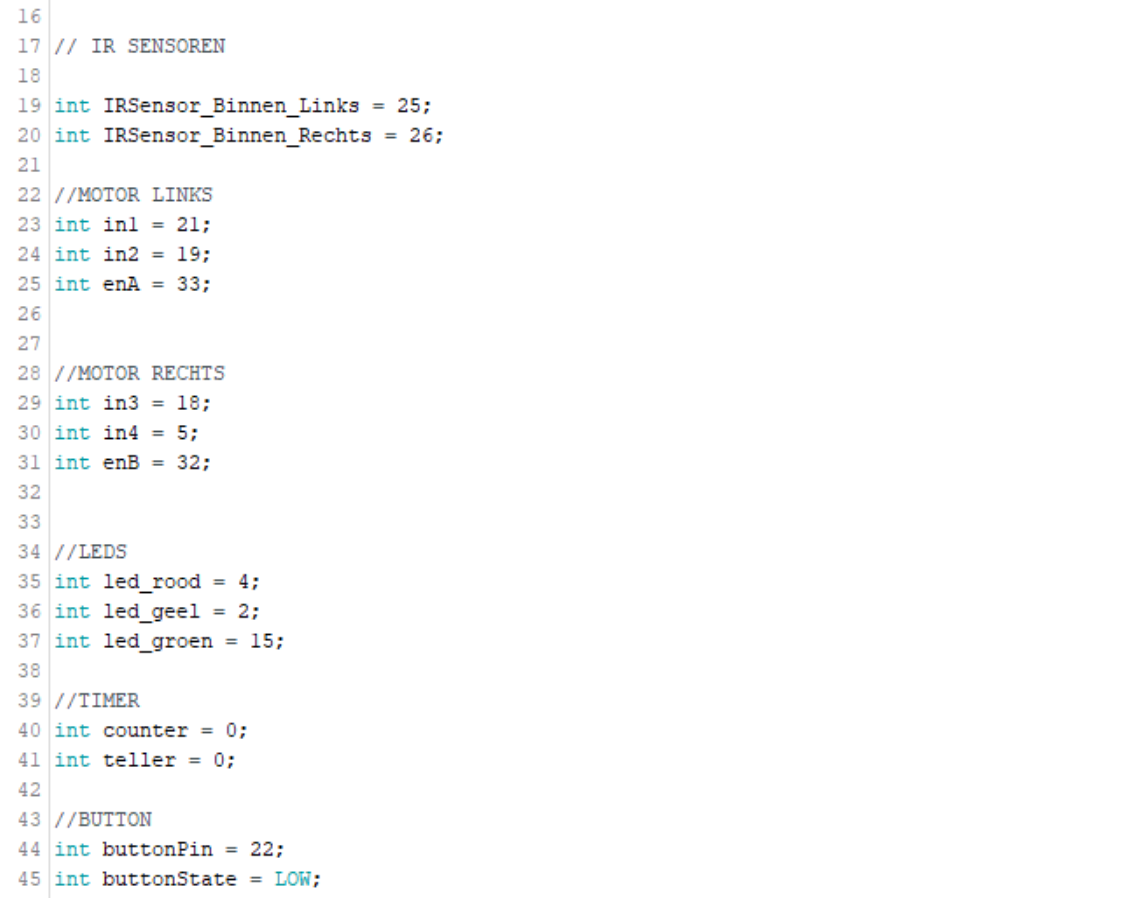


Hier vul je je wifi gegevens in en ok de autorisatie code van blynk in.



Deze float stellen we in voor de tijd en afstand van de gemeten waarde van de ultrasone sensor.

## Declaratie pinnen



Nu declareren we wat er op welke pin aangesloten is en stellen ook een timer en een teller in.

## Ingangen en uitgangen

In de void setup maken we eerst verbinding met de blynk server en stellen dan alle inputs en outputs in.

## Begin void loop

In het eerste deel van de voil loop laten we blynk runnen en stelle we de ultrasone sensor in en meten we de afstand die ons een waarde geeft.



## Optioneel

Dit gedeelte is optioneel maar is wel handig om de waardes van de sensoren te kunnen lezen in de seriële monitor.

## Vooruit

De auto vooruit laten rijden doen we als volgt door de in1 tot en met in4 hoog of laag in te stellen kunnen we kiezen hoe de motoren draaien in dit geval staan ze vooruit. Via de enable kanalen A en B kunne we de snelheid instellen gaande van 0 tot 255. Ook worden hier de juiste leds aangezet en word er een counter gestart, als de auto te lang geen lijn zou zien dan stopt deze.

## Stop

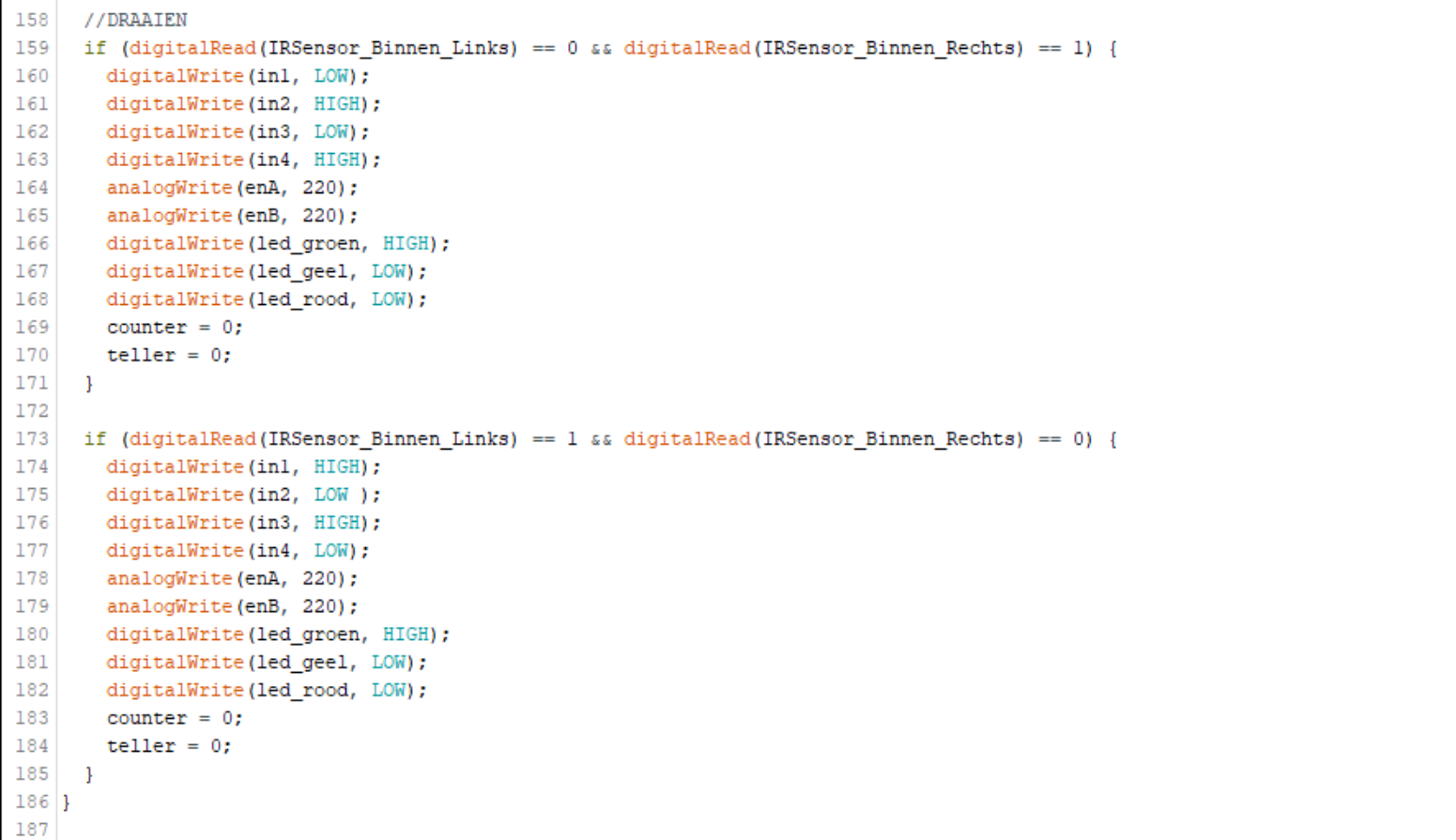
Hier laten we de wagen stoppen op een dwarslijn en laten we ook een teller starten als de teller zijn waarden bereikt heeft dat overeenkomt met 20 seconden zal de auto terug starten. Hoe we dat doen zie je in de volgende afbeelding.

## Timer 20 seconden



Hier laten we de auto terug starten na de gewenste waarde. Of er moet op de bypass knop op de auto worden geduwd.

## Draaien



Met deze code gaan we zorgen dat de wagen mooi op de lijn blijft rijden door simpel weg te gaan corrigeren dit doen we door 1 van de motoren de andere kant op te laten draaien waardoor de wagen niet zal afwijken. Dit is ook het einde van de void loop.

## Blynk



Met deze code gaan we op blynk een drukknop instellen die wanneer je deze indrukt zijn waarde naar de ESP-32 stuurt als deze waarde 1 is zal de auto doorrijden als deze aan een dwarslijn staat .

# Schema