

Das Schaltplan-Navigations-System für Hobbyprojekte

Dieses Dokument beschreibt ein Simulationsprogramm, das uns die erfolgreiche Umsetzung von Hobbyprojekten oder Maker-Projekten viel leichter macht. Oder zumindest von allen Projekten mit eigener Elektrik.

Dabei nimmt es den Bau eines einfachen Roboters als Beispielprojekt für den Einstieg in die Hobbyelektrik.

Inhalt

1: „Wozu Simulieren?“ am Beispiel Putzroboter.....	2
1.1 Mechanik.....	2
1.2 Informatik.....	3
1.3 Elektrik.....	4
2. Das Schaltplan-Navigations-System.....	5
2.1 ein einfaches Beispiel: eine Leuchtdiode.....	5
2.2 ein fortgeschrittenes Beispiel: ein Schaltregler.....	7
3. Schlussfolgerung.....	8

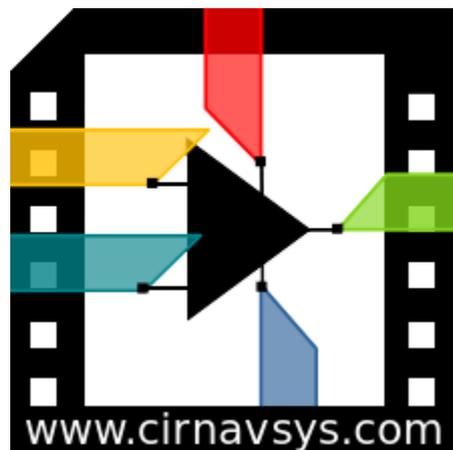


Abbildung 1: Das Logo des Schaltplan-Navigations-Systems

1: „Wozu Simulieren?“ am Beispiel Putzroboter

Zunächst sollten wir uns fragen: Brauchen wir für ein Maker-Projekt überhaupt eine Schaltungssimulation? Man will ja oft einfach loslegen, real etwas bauen und nicht so sehr sich mit Computersimulationen beschäftigen. Machen wir dazu ein Beispiel, ein typisches Hobbyprojekt für den Einstieg in die Robotik ist der Putzroboter.



Abbildung 2: selbstgebauter Putzroboter in Aktion.

1.1 Mechanik

Die Mechanik dieses Putzroboters besteht aus einem Brett, an das man links und rechts jeweils ein Rad mit Elektromotor dran geschraubt hat, noch ein drittes Rad zur Stabilität und vorne zwei Staubwedel, die den Dreck zusammen kehren. Die Mechanik ist hier schnell gebaut:

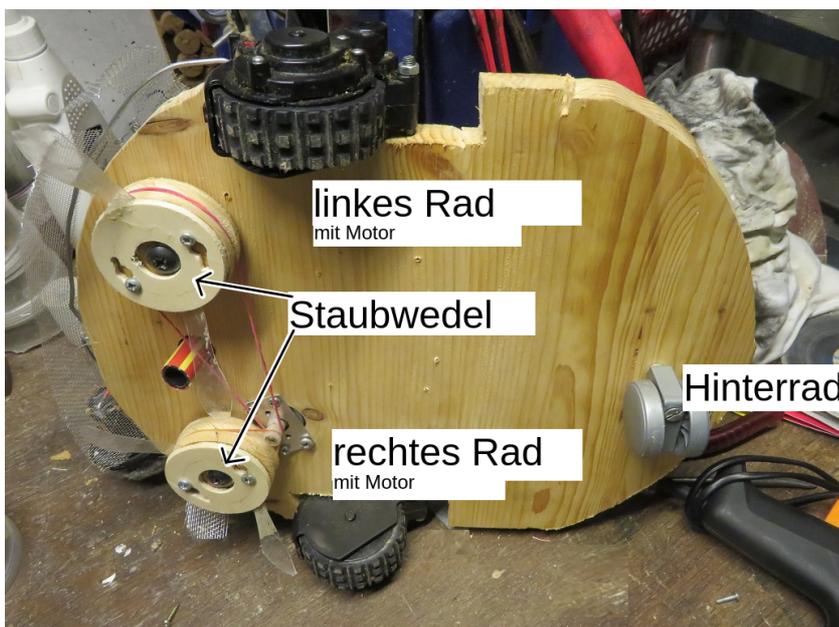
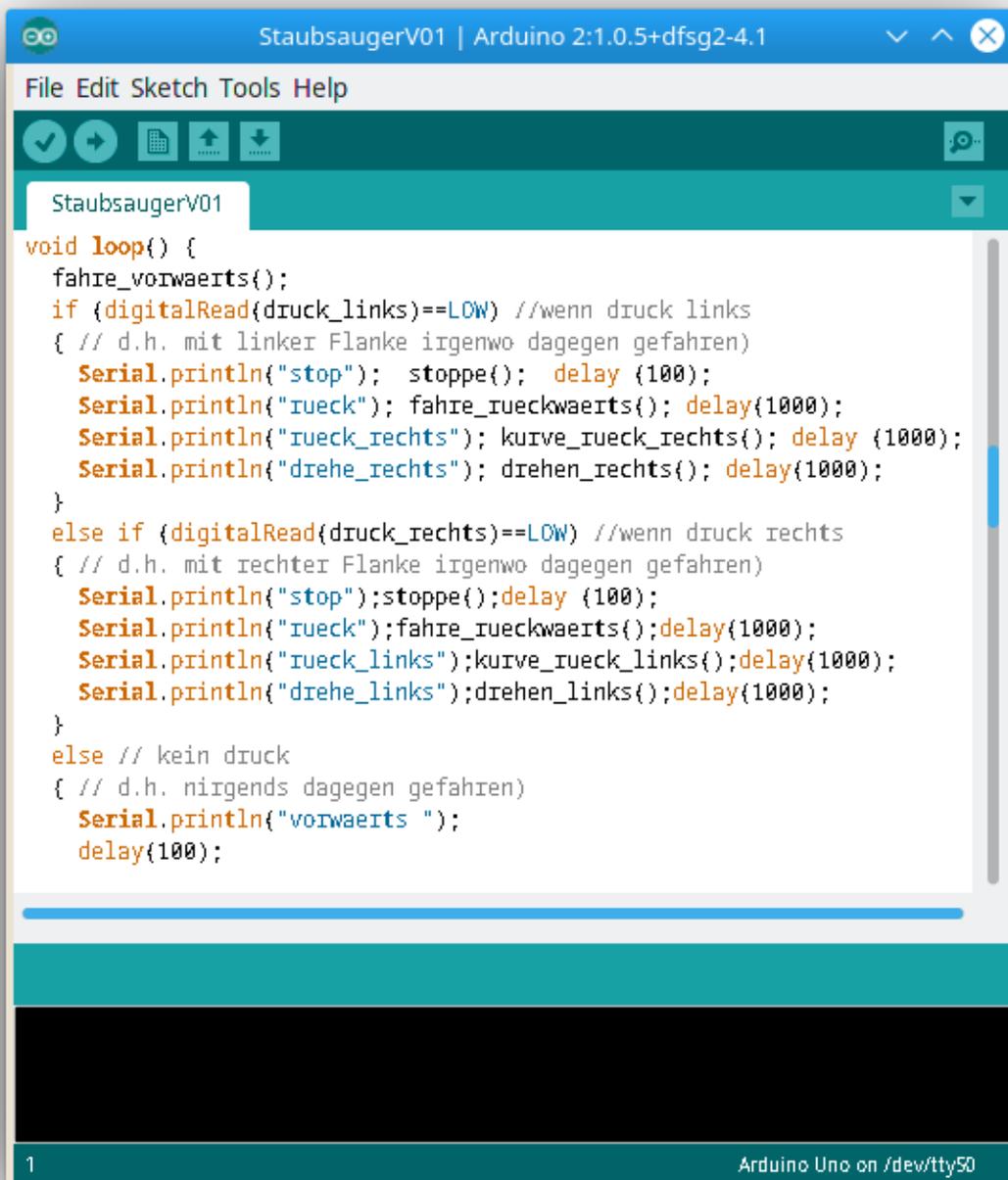


Abbildung 3: Mechanik des Putzroboters

1.2 Informatik

Der zweite Teil ist eine Steuerung für den Roboter. Das Programm eines einfachen Putzroboters ist nicht so kompliziert. Der fährt einfach geradeaus, bis er irgendwo dagegen geknallt ist. Dann dreht er um, und fährt in irgend eine andere Richtung. Und das macht er so lange, bis er überall war und an allen Stellen sauber gemacht hat. Und deswegen ist so ein Putzroboter ein gutes Projekt für den Einstieg in die Robotik: Man kann mit wenig Aufwand ein einfaches Programm schreiben, und das funktioniert einigermaßen. Oder man kann - wenn man Lust dazu hat - das Programm verbessern, sodass der Roboter in möglichst kurzer Zeit den Raum an allen Stellen geputzt hat. Da können wir einfach ausprobieren welche Programme funktionieren, und nebenbei das Programmieren lernen.



```
void loop() {
  fahre_vorwaerts();
  if (digitalRead(druck_links)==LOW) //wenn druck links
  { // d.h. mit linker Flanke irgendwo dagegen gefahren)
    Serial.println("stop"); stoppe(); delay (100);
    Serial.println("rueck"); fahre_rueckwaerts(); delay(1000);
    Serial.println("rueck_rechts"); kurve_rueck_rechts(); delay (1000);
    Serial.println("drehe_rechts"); drehen_rechts(); delay(1000);
  }
  else if (digitalRead(druck_rechts)==LOW) //wenn druck rechts
  { // d.h. mit rechter Flanke irgendwo dagegen gefahren)
    Serial.println("stop");stoppe();delay (100);
    Serial.println("rueck");fahre_rueckwaerts();delay(1000);
    Serial.println("rueck_links");kurve_rueck_links();delay(1000);
    Serial.println("drehe_links");drehen_links();delay(1000);
  }
  else // kein druck
  { // d.h. nirgends dagegen gefahren)
    Serial.println("vorwaerts ");
    delay(100);
  }
}
```

Abbildung 4: Die Programmschleife zum Steuern des Putzroboters

1.3 Elektrik

Aber damit der Roboter überhaupt fährt, muss er auch verdrahtet werden.

Aus den Steuersignalen des Mikrocomputers müssen stärkere Ströme werden, um die Motoren anzutreiben. Dafür brauchen wir Treiberschaltungen. Und eine Zustandsanzeige mit LEDs, ob der Staubbeutel voll ist, wäre auch schön. Aber sowohl für die Treiberschaltungen, als auch für die Zustandsanzeige müssen wir Bauteile auswählen, und sie richtig mit einander verdrahten.

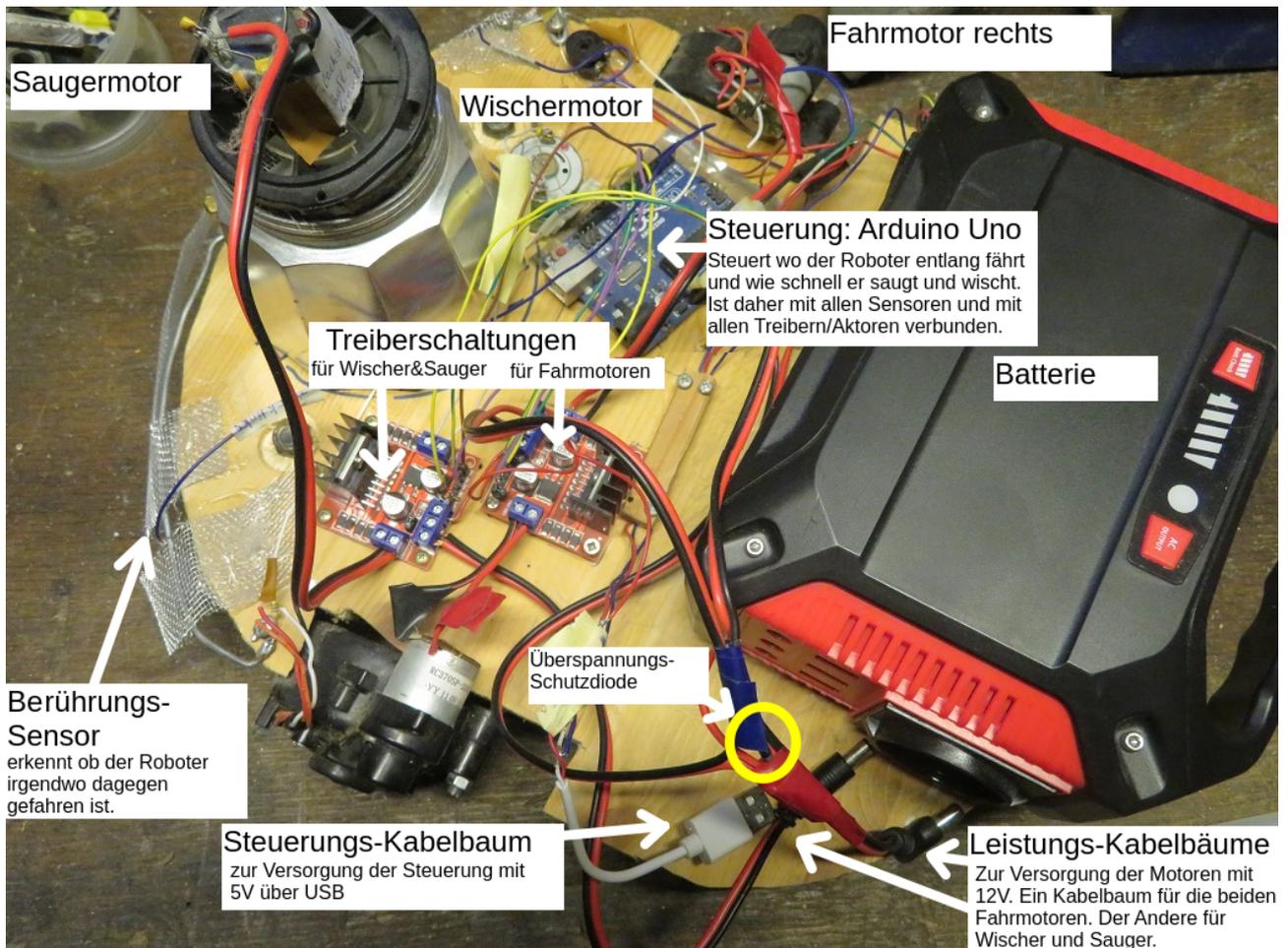


Abbildung 5: Elektrischer Aufbau des Putzroboters. Mit Beschreibung der meisten Bauteile.

An diesem Bild erkennen wir, dass die Elektrik richtig aufzubauen in diesem Projekt etwas schwieriger ist, als die anderen beiden Abschnitte zuvor.

Hier müssen alle diese Bauteile mit einander verdrahtet werden. Und zwar richtig verdrahtet werden. Denn ein Fehler in der Elektrik führt oft dazu, dass irgend ein Bauteil überlastet ist und durchbrennt, und das Projekt dann nicht funktioniert, sondern nur Frust bereitet. Wir können also nicht einfach ausprobieren was funktioniert, denn das geht meistens schief. Ausser wir haben sehr viel Glück. Wenn wir hier unnötige Frustration vermeiden wollen, dann sollten wir vorher simulieren:

- Welchen Vorwiderstand brauche ich vor einer LED, dass sie zwar hell leuchtet, aber nicht „zu hell“ so dass sie durchbrennt?
- Funktioniert die Treiberschaltung mit diesem Transistor, oder sollte man einen anderen nehmen?

2. Das Schaltplan-Navigations-System

Solche Fragen wie oben entscheiden darüber ob ein Projekt funktioniert.

Dafür sollten wir in die Schaltung hinein sehen können. Sehen können, was an jedem Bauteil passiert.

Das Schaltplan-Navigations-System zeigt uns genau das: Es zeigt uns als Animationsvideo, wieviel Strom durch jedes Bauteil fließt und welche Spannung an jedem Bauteil anliegt.

2.1 ein einfaches Beispiel: Leuchtdiode

Fangen wir zunächst mit einer Zustandsanzeige mit einer LED an. Damit so eine LED leuchtet, muss ein Strom durch die Diode fließen. Der Strom darf nicht zu klein sein, sonst leuchtet sie nicht oder nur so schwach, dass man es gar nicht sieht. Er darf aber auch nicht zu gross sein, sonst brennt die LED durch. Die allermeisten LEDs haben ein maximal-Limit von ca 20mA.¹

In der folgenden Abbildung sehen wir aber, dass bei 9V und einem Vorwiderstand von 100Ω hier 82mA durch die LED fließen würden. Die Diode würde also beim Einschalten durchbrennen:

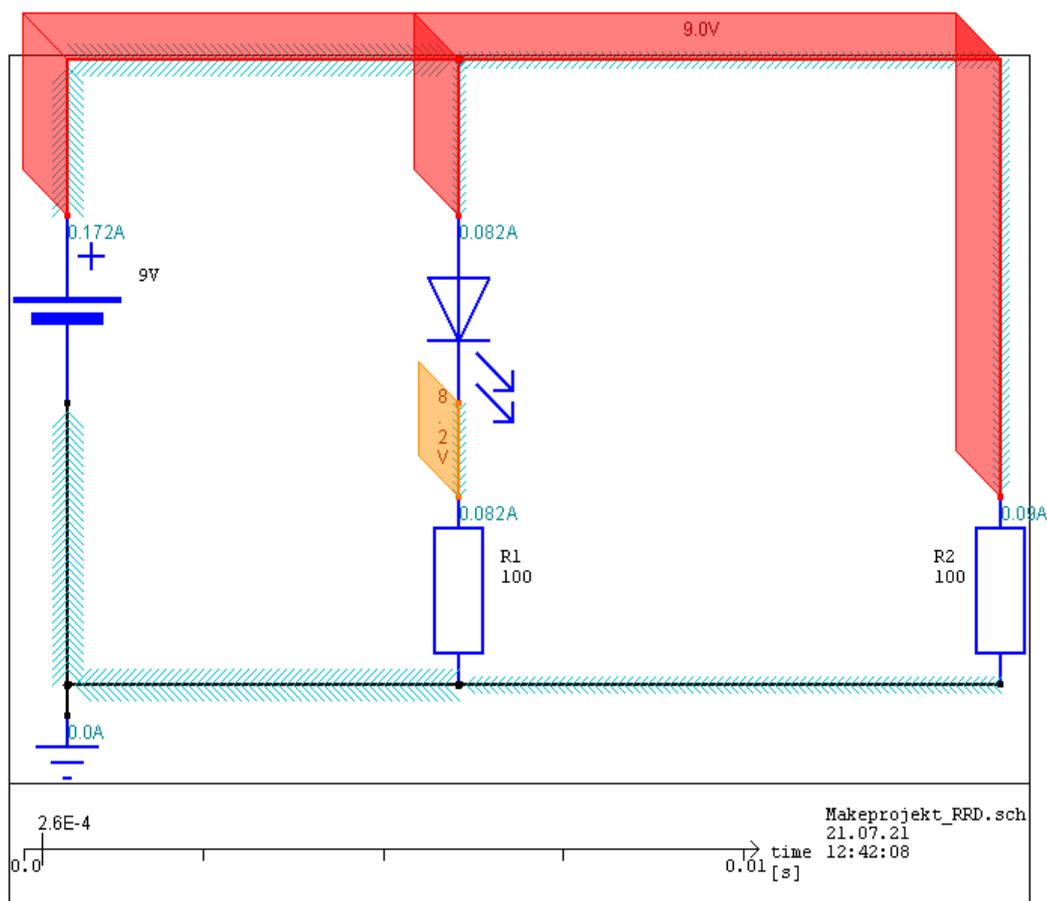


Abbildung 6: Eine einfache Schaltung mit einer Batterie, zwei Widerständen und einer LED als Ausgabe des Schaltplan-Navigations-Systems

Deswegen passen wir im Schaltplan-Editor (hier in KiCad EESchema) den Vorwiderstand R1 an, und ändern ihn auf 500Ω:

¹ Beispielsweise bei den LEDs von einem Online-Versandhändler namens „Conrad“ sehen wir, dass bei den meisten LEDs „20mA“ dran steht, siehe <https://www.conrad.de/de/o/leds-bedrahtet-0212029.html>

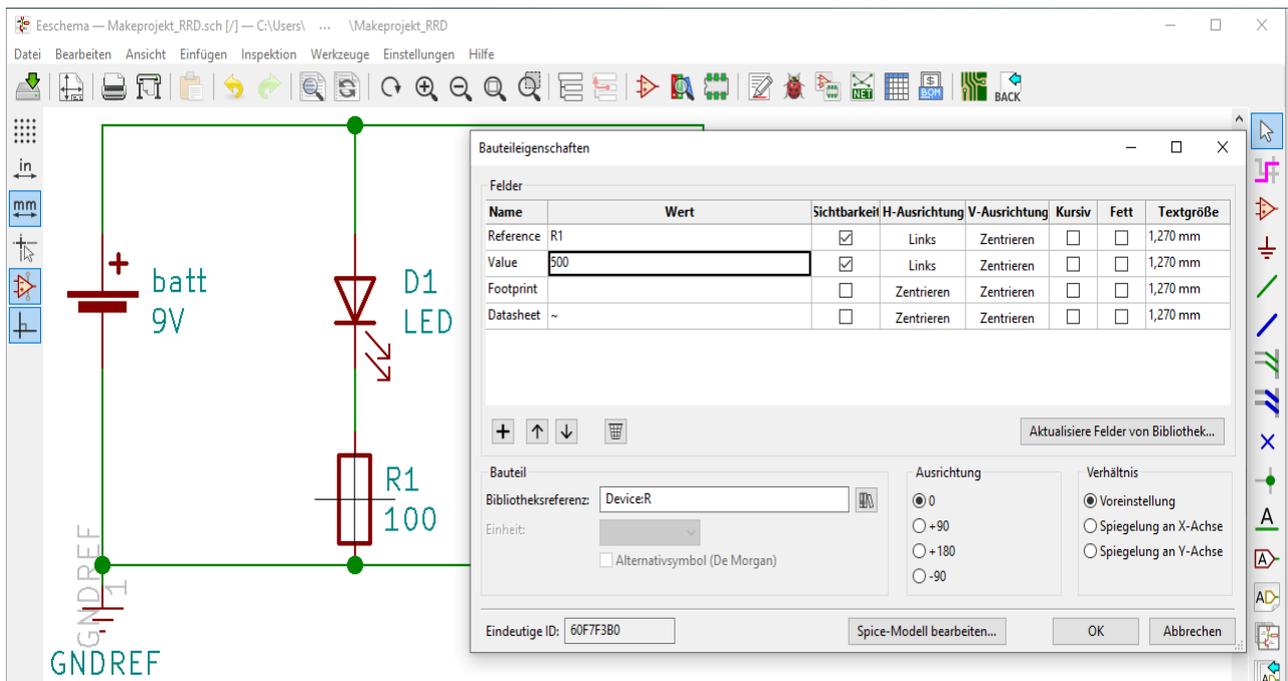


Abbildung 7: Ändern des Wertes von R1 von 100 Ohm zu 500 Ohm

Wir simulieren die Schaltung erneut, und sehen, dass der Strom durch die LED nur noch 16,5mA beträgt. Also etwas unter dem Maximalwert von 20mA bleibt. So wird die LED gut sichtbar leuchten, aber nicht durchbrennen:

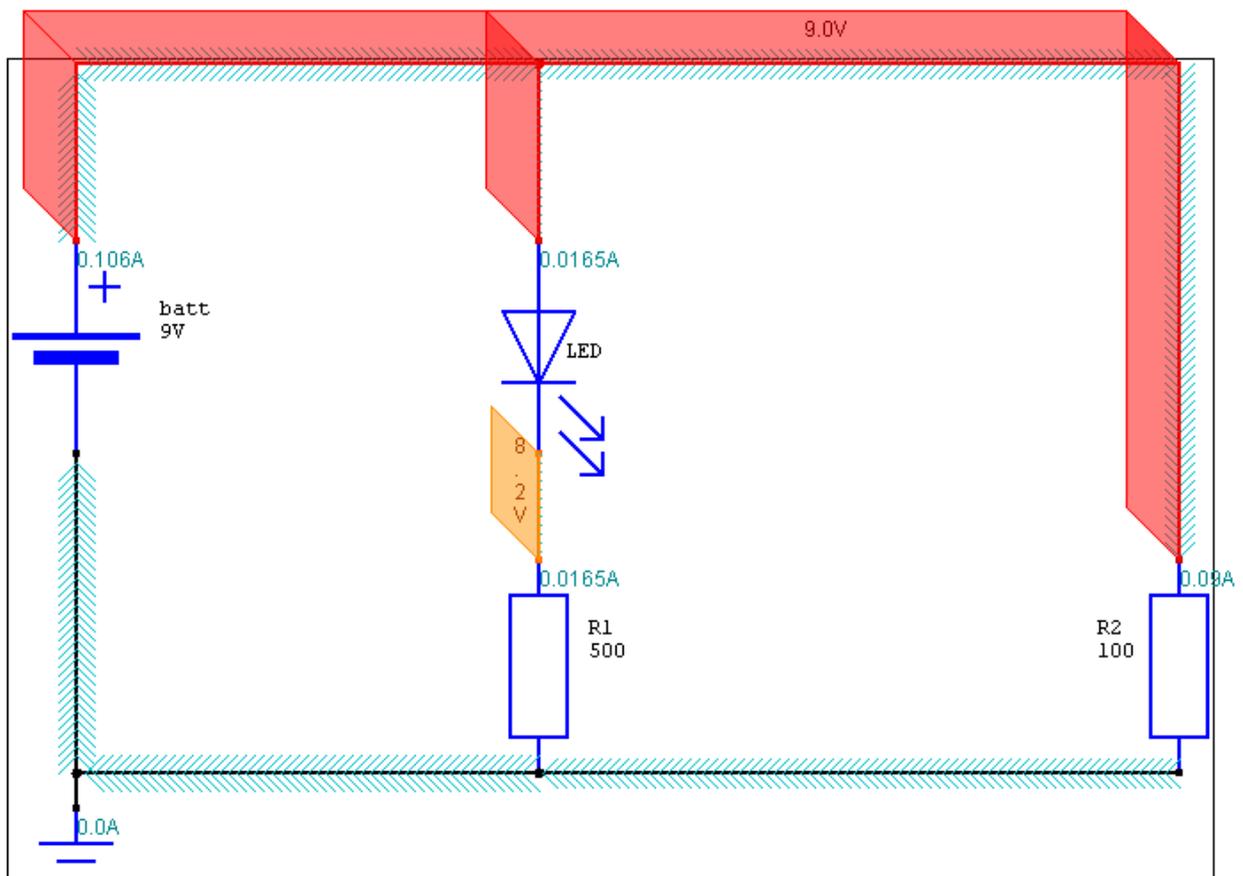


Abbildung 8: Erneute Simulation nach der Anpassung von R1 auf 500 Ohm

Probieren wir es danach real aus:

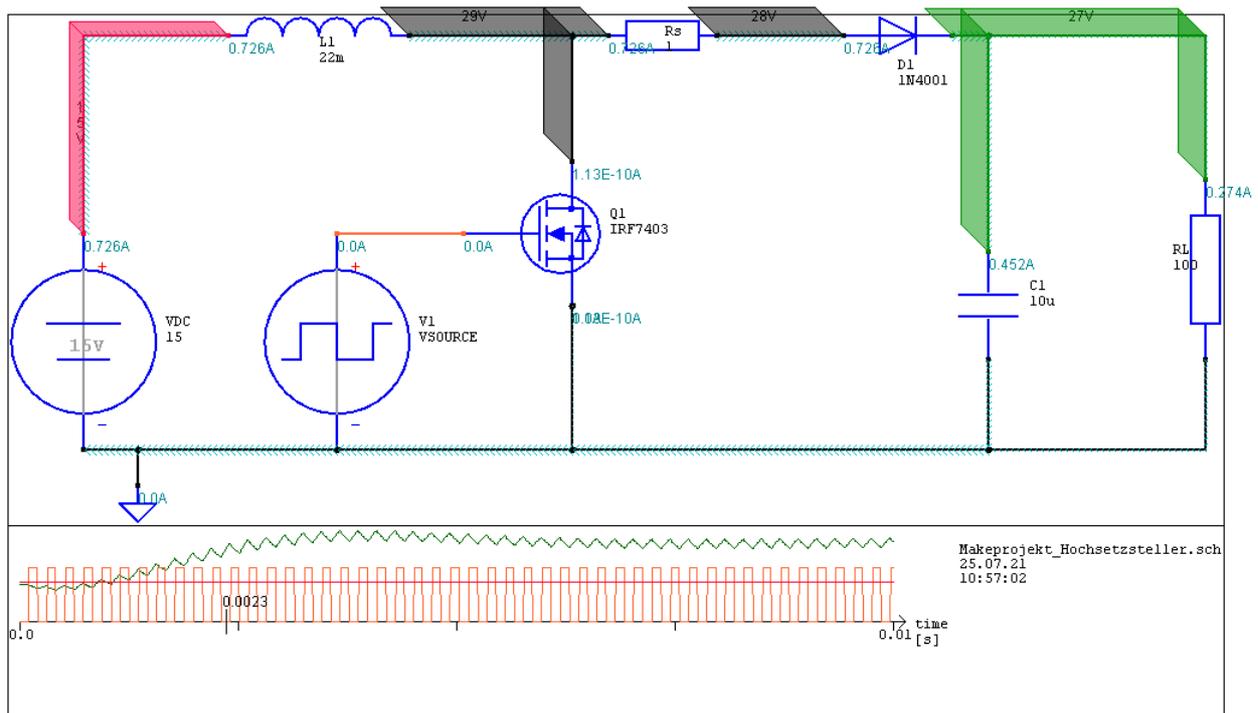


Abbildung 11: Hochsetzsteller mit sperrendem Transistor. Der Strom wird an den Ausgang geleitet. Sobald der Transistor ausgeschaltet ist und sperrt, muss der Strom von der Spule sich einen anderen Weg suchen. Es gibt aber nur einen Weg: den zum Ausgang (grün). Egal wie gross die Spannung am Ausgang ist, der Strom fliesst trotzdem dort hin. Er fliesst in den Kondensator C1 und erhöht dort die Spannung. Daher steigt die Spannung an C1 immer weiter an, bis der Stromzufluss aus der Spule gleich gross ist, wie der Stromabfluss über die 100 Ohm von RL. In diesem Beispiel ist dieses Gleichgewicht „Stromzufluss gleich Stromabfluss“ bei etwa 30V Ausgangsspannung erreicht. Es dauert in diesem Beispiel etwa 3ms, bis dieses Gleichgewicht erreicht ist.

Damit so eine Schaltung funktioniert, müssen die Frequenz aus der PWM von V1 zu dem Spulenwert von L1 und C1 passen. Dafür gibt es entweder komplizierte Berechnungen aus Elektrotechnik-Handbüchern oder man führt eine Simulation durch. Und wenn die Werte nicht passen, dann versucht man es eben mit anderen Werten und einer erneuten Simulation. Bis man eine Schaltung hat, die so funktioniert, wie man sie haben will.

3. Schlussfolgerung

In allen Projekten, welche die irgendwelche Elektrik enthalten, ist es sehr sinnvoll eine genaue Übersicht zu haben, was wie verdrahtet werden muss, und welche Bauteile an welcher Stelle geeignet sind, und wo nicht. Denn sonst braucht man sehr viel Glück, damit das Projekt funktioniert.

Die schnellste Möglichkeit, eine solche Übersicht zu erhalten, ist eine Schaltplan-Simulation. Eine andere gute Möglichkeit ist ein Elektrotechnik-Studium, das dauert aber länger.

Zwar ist auch der Einstieg in die Simulation nie ganz einfach, denn man muss sich mit einem neuen Programm vertraut machen und auf die manchmal komischen Meldungen vom Simulationsprogramm reagieren. Doch im Vergleich zu dem Ärger, den man bei der Fehlersuche in einer falsch aufgebauten oder sogar durchgebrannten Schaltung hat, ist die Simulation ein Kinderspiel.