

# Motortreiber (H-Brücke)

## Motor ist nicht gleich Motor

Es gibt eine Vielzahl an **unterschiedlichen Motor-Typen**, die alle unterschiedlich aufgebaut sind und unterschiedlich betrieben werden müssen.

Wir sprechen hier von ganz gewöhnlichen **DC-Motoren (Gleichstrom-Motoren)**, die genau zwei Anschlüsse besitzen und die man prinzipiell auch einfach an eine Batterie anschließen könnte. Würde man die Batterie anders herum anschließen, würde sich der Motor anders herum drehen.

In den folgenden Bildern siehst du einige unterschiedliche DC-Motoren, die sich rund um das Arduino bewährt oder auch nicht bewährt haben:



**Ein 5V-Ventilator** (eigentlich unbrauchbar)

*Leider enthält er eine interne Elektronik, so dass er nur in eine Richtung betrieben werden kann.*

*Aufgrund der internen Elektronik kann er auch nicht in seiner Geschwindigkeit geregelt werden. Es geht nur „an“ und „aus“ ...*



**Ein billiger 5V DC-Motor** (sehr begrenzt brauchbar)

*Er lässt sich in beide Richtungen betreiben und auch in engen Grenzen in der Geschwindigkeit regeln.*

*Er dreht viel zu schnell, hat kaum Kraft und zieht zu viel Strom, so dass ohne externe Stromversorgung das Arduino oft abstürzt ...*



**Ein Plastik-Getriebe-Motor** (brauchbar bis gut)

*Es gibt ihn mit und ohne zusätzliches Rad (Reifen).*

*Er lässt sich in beide Richtungen betreiben und einigermaßen akzeptabel in der Geschwindigkeit regeln, verliert bei geringer Drehzahl aber schnell an Kraft. Eigentlich bräuchte er 6V ...*



**Ein Metall-Getriebe-Motor** (sehr überzeugend)

*Er lässt sich in beide Richtungen betreiben, hat erheblich mehr Kraft als man dem kleinen Ding zutraut und verliert auch nicht so schnell an Kraft, wenn man ihn herunter regelt.*

**Vorsicht:** *Es gibt ihn in unzähligen Abwandlungen. Achte auf Bezeichnungen wie „3-6V“ und „40 rpm (rounds per minute) bei 6V“ !!!*

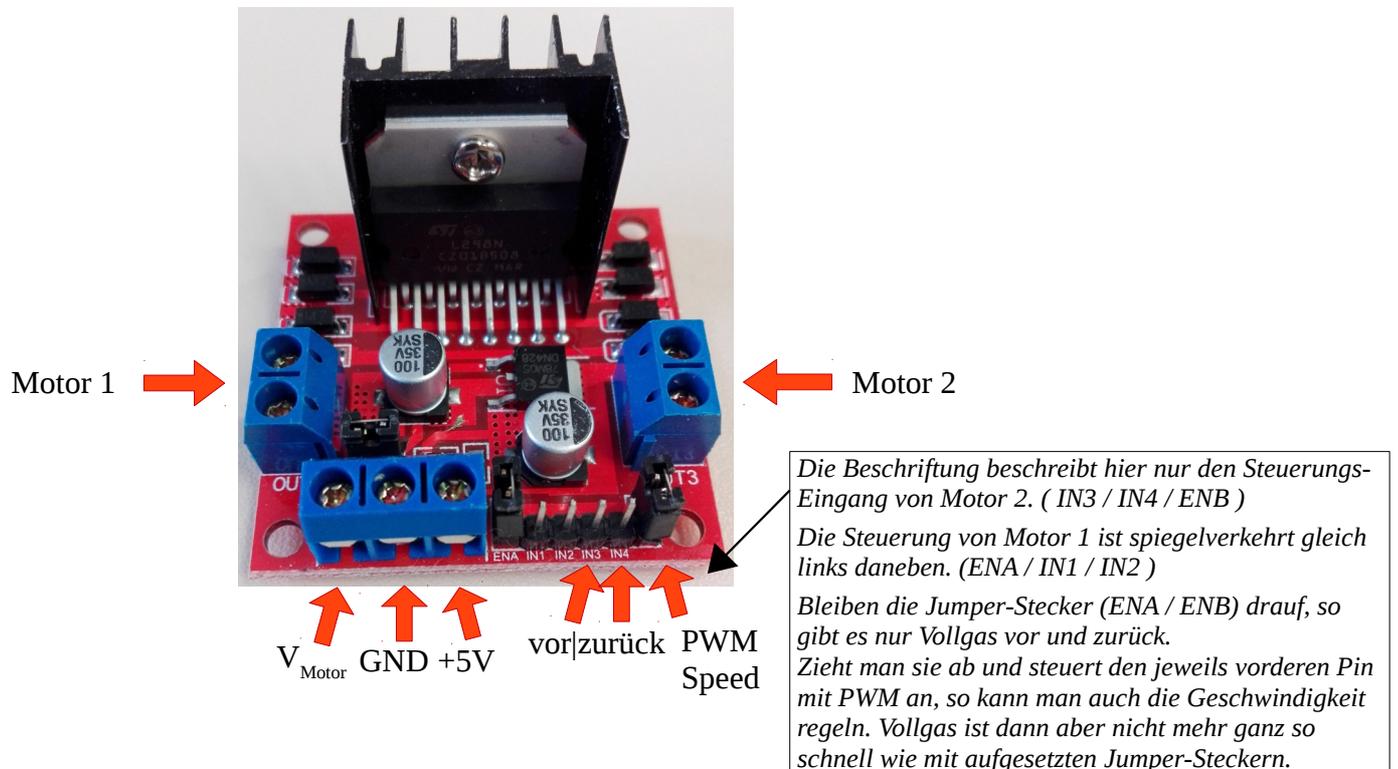
**Wieso** brauchen wir dann eigentlich noch **zusätzliche Hardware** wie die im folgenden beschriebenen H-Brücke?

**Ein DC-Motor stellt eine induktive Last dar** und „feuert“ eine selbst generierte Spannung an das Arduino zurück, die der Steuer-Spannung entgegen wirkt und höher als diese sein kann. **Das Arduino würde abstürzen oder zerstört werden ...**

Deshalb schaltet man zwischen Arduino und Motor eine zusätzliche Hardware, den Motor-Treiber oder auch H-Brücke genannt. Damit kann man den Motor dann einfach und gefahrlos vorwärts und rückwärts laufen lassen und ihn in der Geschwindigkeit regeln ... und genau das brauchen wir ...

## Eine günstige H-Brücke

Das Bild zeigt eine günstig erhältliche Doppel-H-Brücke, mit der man 2 Motoren ansteuern kann, deren Spannung sogar bis zu 12 V betragen kann ...



**Links und rechts** werden die Motoren an die beiden **blauen 2er-Terminals** angeschlossen.  
(Motor 1 und Motor 2)

Das **vordere blaue 3er-Terminal** dient der Spannungsversorgung:

**rechts** 5V (vom Arduino) für die Elektronik der H-Brücke,  
**mittig** GND (vom Arduino),  
**links** 5-12V für die Motoren.

Der linke Anschluss wird in der Regel mit dem Plus-Pol eines zusätzlichen, externen Netzteils verbunden, dessen Minus-Pol dann als zweites Kabel in die Mitte kommt.

**Vorne rechts** findet man die **Steuer-Pins**:

**IN1 , IN2** für die Richtung des linken Motors,  
**IN3 , IN4** für die Richtung des rechten Motors.

Hier bekommt jeder Motor 2 Anschlüsse zum Arduino.

Einer wird auf HIGH, der andere auf LOW gesetzt. Nun dreht sich der Motor in eine Richtung. Vertauscht man HIGH und LOW, so dreht sich der Motor anders herum.

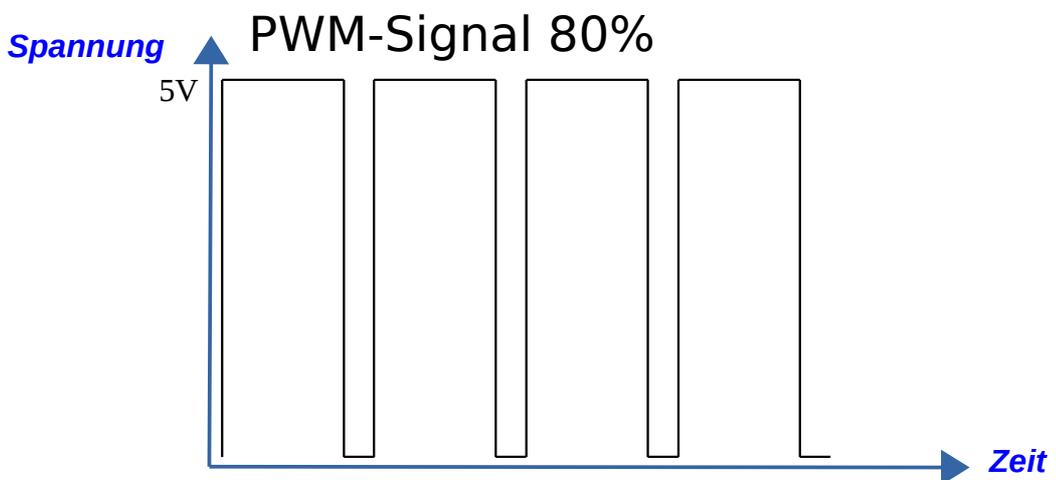
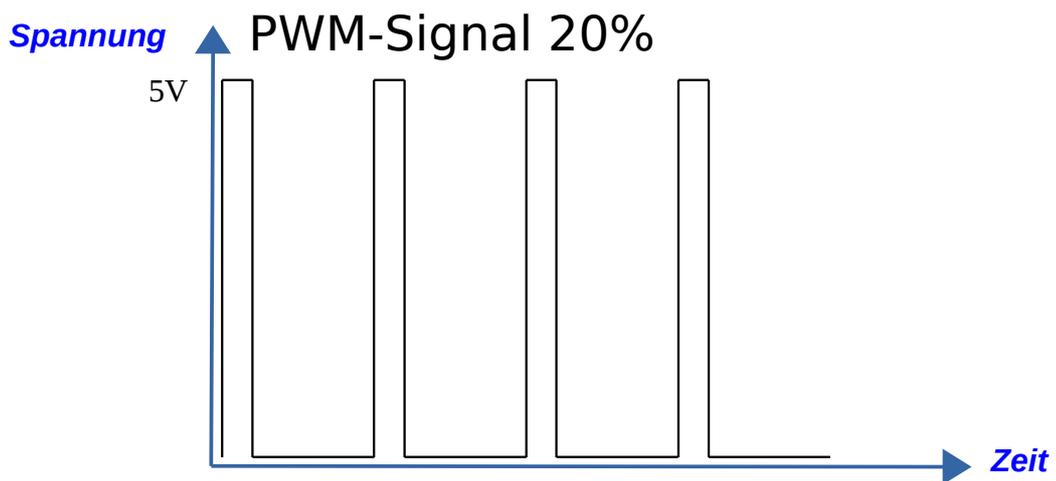
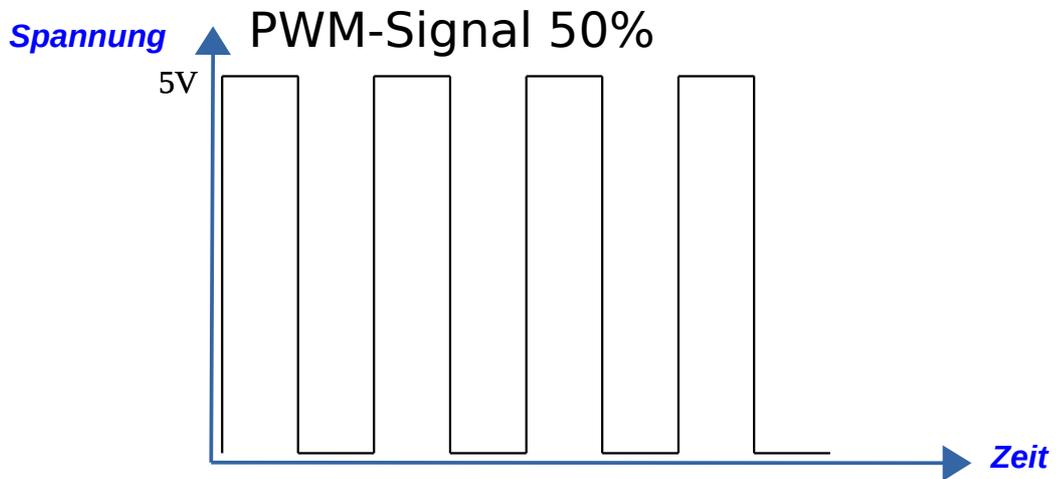
**ENA , ENB** Links und rechts neben den IN-Pins mit einem Jumper-Stecker darauf.

Sind diese Jumper-Stecker aufgesteckt, so kann man nur vorwärts und rückwärts drehen, aber nicht die Geschwindigkeit regeln.

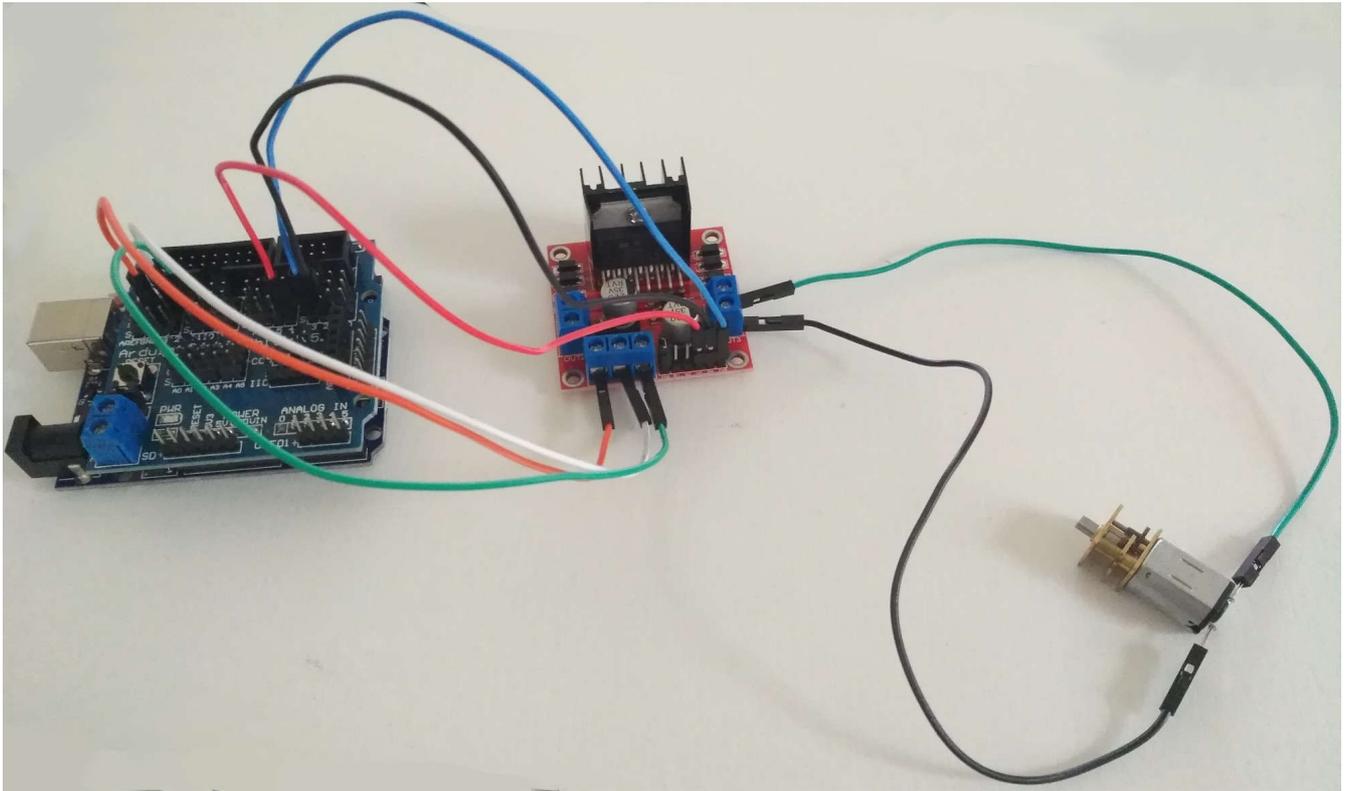
Zieht man die Jumper-Stecker ab, so geht erst mal gar nichts mehr. Verbindet man nun die vorderen ENA- und ENB-Pins mit je einem PWM-Pin des Arduino, so kann man zusätzlich die Geschwindigkeit regeln. Ein PWM-Wert (Duty Cycle) von 0 bedeutet Stillstand, 100 entspricht „Vollgas“. Werte dazwischen sorgen für unterschiedliche Geschwindigkeit. PWM wird auf der nächsten Seite erklärt.

# Pulsweitenmodulation (PWM)

Ein **PWM (Pulsweitenmodulation) Signal** ist eine **Rechteck-Spannung**, deren Anteil HIGH einer Periode man einstellen kann. **Nur 6 Pins (3, 5, 6, 9, 10, 11) des Arduino Uno unterstützen PWM.**



## Fertiger Aufbau



Spannungs- und Strom-Versorgung der H-Brücke von beliebigen V- bzw. G-Pins des Sensor-Shield. Dies funktioniert aber nur, solange der Motor nicht zu viel Strom zieht ohne Weiteres. Wird mehr Strom benötigt, so muss man eine externe 5V-Spannungs-Quelle an die blauen Pins des Sensor-Shield anschließen. Das kann z.B. ein Handy-Lade-Gerät oder eine Power-Bank sein, solange man mit 5V auskommt.

**Schließe NIE eine andere Spannung als 5V an die blauen Pins des Sensor-Shield an !!!**

Richtungs-Steuerung hier über Pins 2 und 4, PWM über Pin 3.