



[RN-Wissen Home](#) - [Community Home](#) - [Alle Artikel](#) - [Mitglieder](#) - [Moderatoren](#) - [Bilderliste](#) - [Letzte Änderungen](#)
[Kategorien](#) - [Beliebte Seiten](#) - [Sackgassenartikel](#) - [Artikel ohne Kategorie](#) - [Neue Artikel](#) - [Anmelden](#)

Getriebemotoren Ansteuerung

aus RN-Wissen, der freien Wissensdatenbank

Ist der Entschluss gefasst, für seinen Roboter oder andere Konstruktion einen Getriebemotor zu verwenden, so müssen sich Gedanken gemacht werden, wie er angesteuert werden kann. Gewöhnlich sollen die Motoren ja sowohl in Bezug auf Drehrichtung und Geschwindigkeit über eine Programmiersprache angesteuert werden. Daher wird in der Regel erst mal ein Controllerboard benötigt. Leider kann man an die wenigsten Controllerboards einen Getriebemotor direkt anschließen, eine der wenigen Ausnahmen ist das RN-Control-Board und das RNBFR-Board. Bei beiden Boards ist es einfach, dort müssen die Getriebemotoren einfach nur an die Schraubklemmen angeschlossen werden, danach kann man sofort mit der Programmierung loslegen.

Ich möchte hier aber aufzeigen, wie man Motoren auch bei anderen Boards anschließen kann. Die gleiche Technik läßt sich natürlich auch nutzen, um z.B. mehr als zwei Motoren an RN-Control anzuschließen.

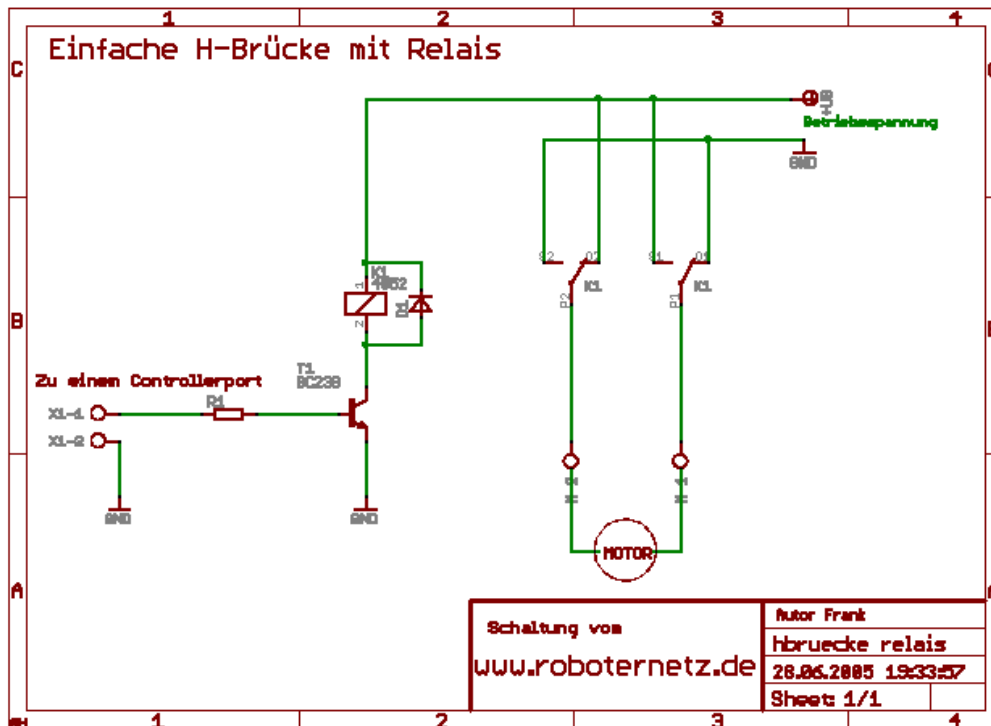
Inhaltsverzeichnis

- 1 Ansteuerung mit Relais
- 2 Ansteuerung mit Relais und MOS-FET
- 3 Ansteuerung mit MOS-FET
 - 3.1 Neufassung
 - 3.2 Grundprinzip
 - 3.3 Elektrische Ansteuerung der Gates
 - 3.3.1 Schaltung direkt am Port
 - 3.3.2 Schaltung mit einem Transistor
 - 3.3.3 Schaltung mit zwei Transistoren
 - 3.3.4 Integrierte Gate-Treiber
 - 3.4 Logische Ansteuerung der Gates
 - 3.4.1 softwaremäßige Ansteuerung
 - 3.4.2 Einfache, gemeinsame Ansteuerung
 - 3.4.3 Ansteuerung mit logischer Verknüpfung
 - 3.4.3.1 Heidingscher Encoder (Beispiel für Logic)
 - 3.4.3.2 PR Encoder
 - 3.4.3.3 PRK Encoder
 - 3.4.4 Support
 - 3.5 Alte Schaltungsbeschreibung
 - 3.5.1 Kritik an der Schaltung
- 4 Ansteuerung mit einem LeistungsOPV
 - 4.1 Heidingscher Motortreiber
 - 4.2 Gleichstromansteuerung
- 5 Ansteuerung mit Treiber ICs
 - 5.1 Ansteuerung mit Treiber IC L293 D
 - 5.2 Ansteuerung mit dem Schaltkreis L298
 - 5.3 Ansteuerung mit L620x
- 6 L6202 für 1 Motor
- 7 L6205 für 2 Motoren
 - 7.1 weitere Treiber-ICs
- 8 Noch mehr Power gewünscht?
- 9 Endstufen mit I2C-Bus / RS232
- 10 Und nie vergessen Motoren zu entstören
- 11 Siehe auch
- 12 Weblinks

Ansteuerung mit Relais

Die einfachste Methode, um Motoren per Controller anzusteuern, erreicht man durch die Verwendung von Relais. Mit einem Relais, das

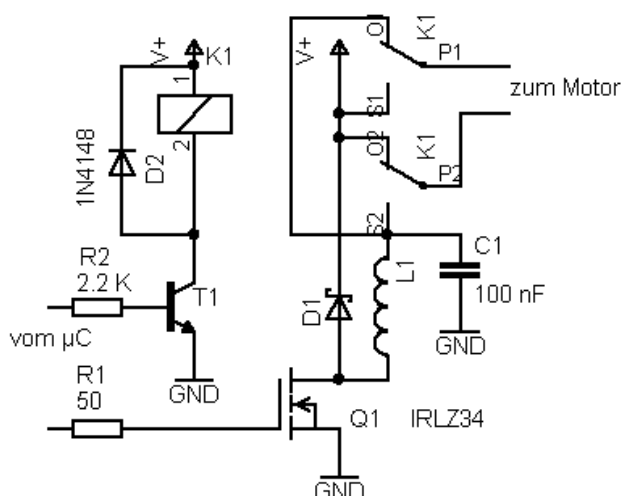
zwei Umschaltkontakte besitzt, lässt sich über einen einzelnen Controllerport bequem die Drehrichtung wechseln.



Da auch Relais wegen des Strombedarfes nicht direkt vom Controller geschaltet werden können, wurde in dem oberen Beispiel ein Transistor vorangestellt. Die Diode dient dazu, die Spannungen abzuleiten, die beim Ausschalten des Relais induziert werden. Ohne Diode könnte der Transistor oder sogar der Controller durch die induzierte Spannung beschädigt werden. Die Schaltung ist also in der Lage, die Drehrichtung umzuschalten, jedoch nicht in der Lage, den Motor zu stoppen. Um den Motor ganz auszuschalten, könnte man man 2 getrennte Relais für die beiden Umschaltkontakte nutzen. Ein Schaltbild können wir uns ersparen, da das Ganze recht ähnlich aussieht. Der große Nachteil von Relaischaltungen ist, dass die Geschwindigkeit des Motors nicht geregelt werden kann, zudem haben Relais bei vielen Schaltvorgängen einen gewissen Verschleiß. Das Ausschalten einer induktiven Last ist dabei für die Relaiskontakte besonders schädlich. Vorteil der Schaltung ist jedoch, das man auch große Lasten/Motoren schalten kann.

Ansteuerung mit Relais und MOS-FET

Die Schaltung ist sehr ähnlich wie die vorherige Relaischaltung. Es wird aber in die Masseleitung hinter dem Relais ein MOSFET und eine Freilaufdiode eingesetzt. Dadurch werden die Nachteile der einfachen Relaischaltung behoben. Durch den MOSFET läßt sich der Motor in der Geschwindigkeit regeln und ggf. abschalten.



Es sind prinzipiell alle N-Kanal MOS-FETs geeignet. Für die hier gezeigte direkte Steuerung vom μC sollte es ein Logik-Level-Typen sein. Der FET IRLZ34 sollte ohne Kühlung bis etwa 5 A genügen. Die Freilaufdiode D1 muß für den Motorstrom ausgelegt sein, und bei PWM Steuerung genügend schnell sein (z.B. Shottky SB540). Die Induktivität L1 (ca. 10...100 μH) und der Kondensator C1 dienen zur

Funkentstörung und können bei kurzem Kabel zum Motor auch weggelassen werden. Um die Umschaltverluste und Funkstörungen gering zu halten, sollte die PWM-Frequenz nicht unnötig hoch liegen. Da das Relais nur Strom verbraucht, wenn es angezogen wird, sollte man den Motor so anschließen, dass in der bevorzugten Drehrichtung (beim Roboter Vorwärtsfahren) das Relais nicht angezogen ist. Die Schaltung ist sicher gegen Querströme und es können durch Fehler im Programm des μ C kaum Schäden entstehen. Das Relais sollte erst umgeschaltet werden, wenn der Motor steht, dann gibt es keine Funken an den Relaiskontakten.

Der Verdrahtungsaufwand insgesamt ist minimal und sogar kleiner als bei Motortreiber-ICs, wie dem L298, da keine zusätzlichen Bauteile benötigt werden. Daher ist die Schaltung insbesondere für Anfänger geeignet.

Vorteile sind:

- tollerant gegen Fehler im μ C Programm
- hohe Strombelastbarkeit
- sehr geringer Spannungsabfall (< 1 V)
- niedrige Kosten, auch bei hoher Leistung
- das Relais kann stromlos geschaltet werden (wenig Verschleiß an Kontakten)

Nachteile sind:

- hörbares "Klacken" beim Laufrichtungswechsel
- zusätzlicher Stromverbrauch, wenn das Relais geschaltet ist
- keine einfache Motorbremse

Ansteuerung mit MOS-FET

Neufassung

Dieser Artikel ist noch lange nicht vollständig. Der Auto/Initiator hofft das sich weitere User am Ausbau des Artikels beteiligen.

Das Ergänzen ist also ausdrücklich gewünscht! Besonders folgende Dinge würden noch fehlen:

Die Neufassung kann im momentanen Stand nur als Anregung verstanden werden.

Insbesondere fehlen noch angepasste Schaltpläne.

Einige der Anmerkungen aus dem Abschnitt "Kritik an der Schaltung" sind in der Neufassung noch nicht enthalten, die Beschreibung von Verzögerungsgliedern müsste auch eine Erklärung enthalten, welche Verzögerung mit welchen Werten erreicht wird.

Wenn die Neufassung komplett fertig ist, sollten die alten Texte weg.

Grundprinzip

Wenn man die Steuerung rein mit Halbleitern realisieren will, bietet sich die H Brückenschaltung (Vollbrücke) an. An jeden Motoranschluß kommt eine Halbbrücke, die den Anschluß mit GND oder der Versorgungsspannung verbinden kann. Die fertigen Motortreiber ICs sind auch entsprechend aufgebaut. Der Aufbau aus Einzelteilen wird vor allem gebraucht, wenn man kein passendes IC findet, z.B. bei hoher Leistung.

Bei den üblichen Batteriespannungen sind als Schaltelemente MOS-FETs die beste Wahl. Also (unipolaren) Transistoren, die quasi leistungslos nur durch Spannungen geschaltet werden. Zu diesem Thema findet man unzählige Beiträge im Roboternetz, denn die Schaltung ist nicht ganz einfach.

Die Grundschialtung besteht aus 2 N-Kanal-MOS-FET und zwei P-Kanal-MOS-FET, mit denen die Motoranschlüsse nach GND bzw. UB geschaltet werden. Auf die extra Freilaufdioden kann bei einer Vollbrücke mit modernen MOSFETs meistens verzichtet werden, da die MOSFETs bereits hinreichend gute Dioden enthalten. Teilweise werden trotzdem zusätzliche Schottky-Dioden benutzt, um die Verluste etwas weiter zu reduzieren. Die Dioden werden in Sperrichtung von den beiden Motoranschlüssen nach GND bzw. UB geschaltet. Die Transistortypen werden entsprechend der benötigten Ströme und Spannungen gewählt. Zu große Transistoren haben den Nachteil einer großen Gatekapazität und benötigen daher aufwendigere Treiber.

Grundsätzlich muss die Beschaltung der Gate-Eingänge so erfolgen, dass zum Starten des Motors die Transistoren einer Diagonale auf "leitend" und die der anderen auf "sperrend" geschaltet werden. Zum Anhalten des Motors werden z.B. die unteren auf leitend und die oberen auf sperrend geschaltet (Der Motor wird aufgrund des nicht unerheblichen Bremsstroms von Transistor und Freilaufdiode stark gebremst). Alternativ kann man auch alle Transistoren auf sperrend schalten, der Motor läuft dann im Leerlauf aus.

Die Transistoren einer Seite dürfen nicht gleichzeitig auf "leitend" geschaltet werden, da dies einen Kurzschluss verursacht.

Elektrische Ansteuerung der Gates

Beim Umschalten der Transistoren entstehen Verluste - der Widerstand wechselt von sperrend (sehr großer Widerstand) nach leitend (sehr kleiner Widerstand) und umgekehrt. Zwischen beiden Enden des Schaltvorgangs befindet sich der Transistor in der "Widerstandsphase" (weder ganz leitend, noch ganz sperrend), die eine erhebliche Verlustleistung und damit Erwärmung mit sich bringt.

Deshalb sollte der Schaltvorgang schnell erfolgen, insbesondere wenn für eine PWM-Regelung sehr oft geschaltet wird. Da zum Umschalten die durch den Aufbau des MOSFETs bedingte Gatekapazität umgeladen werden muss, ist hierzu (sehr kurzzeitig) ein hoher Stromfluss nötig. Zu schnell (<100 ns) sollte man wegen der Funkstörungen aber auch nicht schalten.

Für die P-Mosfets wird eine Spannung von 0 V bzw. etwa -10 V relativ zur positiven Versorgung gebraucht. Für die N-Mosfets reicht eine Spannung von 0 V bzw. ca. 10 V, bei logic level Mosfets reichen 5 V.

Schaltung direkt am Port

Eine Ansteuerung unmittelbar über den Ausgang eines Mikrocontrollers (oder eines TTL-IC) ist schon wegen der benötigten Spannungen oft nur für die N-MOSFETs möglich, und das auch nur bei Logic Level Fets. Bei den P-MOSFET bemisst sich die Schaltspannung gegen U_B und nur für ein U_B von etwa 5-6 V könnte die Spannung passen.

Dass ein Portpin nur einen geringen Stromfluss (ca. 20 mA) zulässt, ist ein weiteres Problem und verhindert einen schnellen Schaltvorgang. Diese Schaltungsversion ist für eher kleine Logic Level MOSFETs (bis etwa 2 nF Gate Kapazität), bevorzugt ohne PWM, möglich.

Schaltung mit einem Transistor

Eine einfache Steuerschaltung besteht aus einem Transistor-Schalter (Emitter nach GND) und einem Pull-Up-Widerstand.

Der Vorteil liegt in der Einfachheit, der Nachteil liegt darin, dass die Schaltung nicht sonderlich schnell vonstatten geht: Für die Schaltung des FET muss die Gate-Kapazität transferiert werden, was insbesondere beim Wechsel von 0 V nach U_B einen Moment dauert, da die Ladung durch den Widerstand hindurch muss. Ein kleinerer Widerstand könnte hier helfen, verursacht allerdings einen nicht unerheblichen Stromfluss im geschalteten Zustand.

Sinnvoll ist diese Version damit nur für die P-MOSFETs, wenn diese nicht mit dem PWM Signal versorgt werden. Für die N Kanal Fets ist es in der Regel besser direkt den Logicpegel und Logic Level Fets zu nehmen.

Schaltung mit zwei Transistoren

Eine bessere Ansteuerung sollte sich ergeben, wenn man zwei Transistoren zur Gate-Ansteuerung verwendet: Einer schaltet gegen 0 V, der andere gegen U_C . So kann man erreichen, dass in beide Richtungen rasch umgeschaltet werden kann.

Integrierte Gate-Treiber

Statt der Beschaltung mit zwei Transistoren kann man auch einen integrierten MOSFET-Treiber verwenden. Derartige ICs (z.B. ICL7667) sind darauf ausgelegt, kurzfristig den für eine Umschaltung benötigten hohen Stromfluss zu ermöglichen.

Es gibt auch Treiber-ICs (z.B. IR2111), die zur Ansteuerung der oberen Transistoren eine Spannung oberhalb von U_B erzeugen. Mit solchen Highside-Treibern kann die H-Brücke auch aus 4 N-Kanal-Transistoren gebaut werden, was den Vorteil hat, dass N-Kanal-FETs mit geringerem Schaltwiderstand zu bekommen sind. Bei den Treibern für highside N-Kanal-Fets hat man aber oft eine Begrenzung des Tastverhältnisses.

Logische Ansteuerung der Gates

Nach der elektrischen Ansteuerung ist die logische Ansteuerung durch den Mikrocontroller zu überlegen. In der H-Brücke sind zwei problematische Zustände zu berücksichtigen:

1. Das offensichtliche Problem ist ein Kurzschluss: Wenn beide Transistoren einer Seite auf leitend geschaltet werden ist das ein Kurzschluss. Falls vorhanden brennt die Sicherung durch oder es wird sogar etwas beschädigt.
2. Ein nicht so offensichtliches Problem rührt daher, dass der Schaltvorgang nur mit endlicher Geschwindigkeit abläuft und der Transistor auch schon vor Erreichen des Zielwertes der Gate-Spannung erheblichen Strom leitet. Hierdurch kommt es zu einem Kurzschluss durch den nicht mehr richtig gesperrten und den noch nicht richtig gesperrten Transistor. Es reicht also nicht die beiden Transistoren gleichzeitig zu schalten, sondern es wird eine kleine Verzögerung benötigt.

Die Stromspitze selbst liegt oft noch im Rahmen der Spezifikationen der Transistoren. Allerdings hat die Stromspitze drei Auswirkungen:

1. Die Transistoren erwärmen sich
2. Es kann an einem Messwiderstand zu einer Spannungsspitze kommen
3. Die Versorgungsspannung bricht kurzzeitig zusammen

Die Störung der Versorgungsspannung lässt sich ohne Oszilloskop kaum messen. Insbesondere in umfangreicheren Schaltungen können Störungen an ganz anderen Stellen (z.B. bei Sensoren) auftreten, die man dann nicht gleich mit der Motorsteuerung in Verbindung bringt.

softwaremäßige Ansteuerung

Die einfachste Art der Ansteuerung besteht darin, die 4 Gates mit 4 Ausgängen des Controllers zu steuern.

Die korrekte Ansteuerung muss dann durch die Software garantiert werden. Eine Fehlschaltung der Software z.B. bei einem Absturz kann allerdings zur Zerstörung der Schaltung führen, die Schaltung sollte also in jedem Fall eine Sicherung haben.

Einfache, gemeinsame Ansteuerung

Bei passender Spannung, können die Gates einer Seite jeweils gemeinsam geschaltet werden. Bei einer Gate-Spannung von 0V sperrt der N-MOSFET ($U_{GS}=0V$), der P-MOSFET leitet bei dieser Gate-Spannung, da hier $U_{GS}=-U_B$ ist.

Bei einer Gate-Spannung von U_B verhält es sich genau umgekehrt: Der N-MOSFET leitet ($U_{GS}=U_B$) und der P-MOSFET sperrt ($U_{GS}=0V$)

Diese einfache Ansteuerung benötigt nur 2 Port-Pins, hat jedoch den Nachteil, dass während des Umschaltens die oben beschriebenen temporären Querströme auftreten können und deshalb eher schnell geschaltet werden sollte. Die Ansteuerung mit nur einem Transistor ist hier also nicht angebracht. Je höher die Spannung, desto schlimmer wird das Problem beim Umschalten. Ohne extra Verzögerungen sind damit nur Spannungen bis ca. 8 V möglich.

Mit einer Parallelschaltung aus einem Widerstand und einer Diode vor den Gates, kann man erreichen, dass die Transistoren schneller auf sperrend als auf leitend schalten. Beim N-FET zeigt die Anode zum FET, beim P-FET die Kathode. Damit sind dann auch etwas höhere Spannungen möglich, aber spätestens bei ca. 20 V wird die maximale Gatespannung erreicht.

Eine mögliche Ausführung für etwa 8-15 V, ist es ein Gate Treiber IC (z.B. ICL7667) zu nehmen, und durch Dioden und Widerstände zu den Gates für eine Verzögerung beim Einschalten zu sorgen.

Ansteuerung mit logischer Verknüpfung

Der Zustand eines dauerhaften Querstroms sollte besser Hardwaremäßig vermieden werden. Zusätzlich kann die Logic die Zahl der IO Pins auf 2 oder 3 reduziert werden. Sinnvollerweise sollte dabei nur eine Leitung die PWM Steuerung übernehmen. Mittels Logik-Bausteinen werden zwei Ausgänge des Controllers so umgesetzt, dass die folgenden Zustände geschaltet werden können:

1. Motor-Stop: untere Transistoren leitend, obere sperrend (oder umgekehrt)
2. Motor-Vorlauf: 1. Diagonale leitend
3. Motor-Rücklauf: 2. Diagonale leitend
4. Motor-Leerlauf: Alle Transistoren sperrend

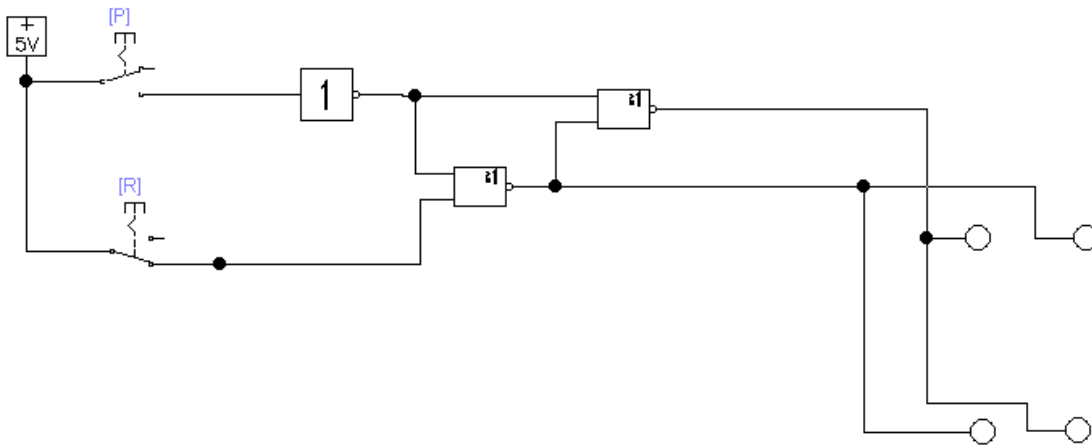
Der Motor Stop Zustand ist dabei nicht unbedingt nötig. Wie die Logic genau aussieht hängt davon ab ob die Gate Treiber die Signale noch mal invertieren. In der wohl einfachsten Form, kann man die P MOSFETs über einfache Transistortreiber langsam steuern. Die PWM Steuerung geht dann nur mit den N-FETs. Hier kann man Logic Level Typen nehmen, die man direkt von Logic Gattern (z.B. 74HC...) ansteuern kann, oder bei größeren FETs mit Gate Treibern.

Die Querstrom-Problematik beim Schalten kann hierbei softwaremäßig gelöst werden, indem beim Umschalten jeweils kurz der Leerlauf geschaltet wird, damit alle Transistoren sperren. Damit auch Temporäre Querströme sicher vermieden werden können auch hier Verzögerungselemente sinnvoll sein.

Heidingscher Encoder (Beispiel für Logic)

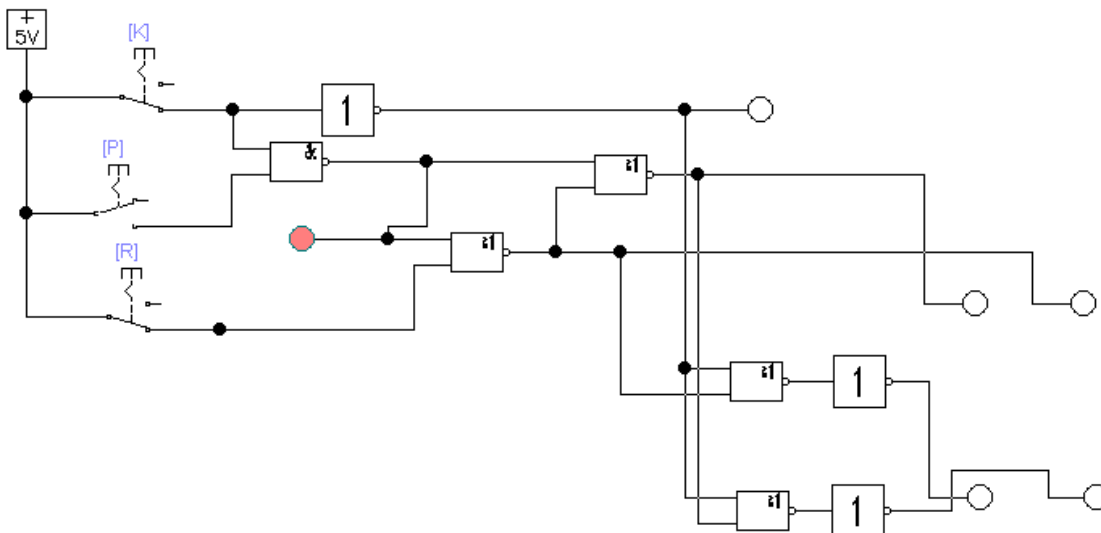
Aufgrund der Kritik an der alten Schaltung(siehe **Alte Schaltungsbeschreibung**) entschied ich mich einen neuen, einfachen sowie preisgünstigen Encoder zu entwickeln, den man mit der 74series aufbauen kann. Die 74series sind mit 20 mA Treiberstrom natürlich kaum zum direkten Treiben geeignet. Die Verzögerung müsste hier in die Treiber integriert sein. Die Logikanzeigen in den Grafiken sollen je einen Transistor bzw. Mosfet darstellen.

PR Encoder



Der PR Encoder(PWM-Richtung Encoder) ist mit zwei NOR Gates sowie eines Inverters aufgebaut, d.h. man kann ihn mit nur einem IC(74HC(T)02) aufbauen. Will man 2 Encoder verwenden, empfiehlt sich den Inverter auszulagern (74HC(T)04 oder 74HC(T)14. Sollte man keinen Schmitt-Trigger benötigen empfiehlt sich aus Schnelligkeitsgründen die 74HC(T)04 Variante). Alternativ ist es noch möglich sich den Inverter zu sparen, und die Ansteuerung für den P(WM) Kanal invertiert laufen zu lassen.

PRK Encoder



Der Prk-Encoder(PWM-Richtung-Kurzschluss-Encoder) ist eine Erweiterung des PR-Encoders. Er erlaubt es zusätzlich, die Motoren durch einen Kurzschluss zu stoppen. Aufgebaut werden kann dieser aus 1 NOR und 1 NAND Gate. Dabei werden 3 NAND-Gates als Inverter beschaltet (d.h. eine Leitung, die übrigen bleiben auf VCC).

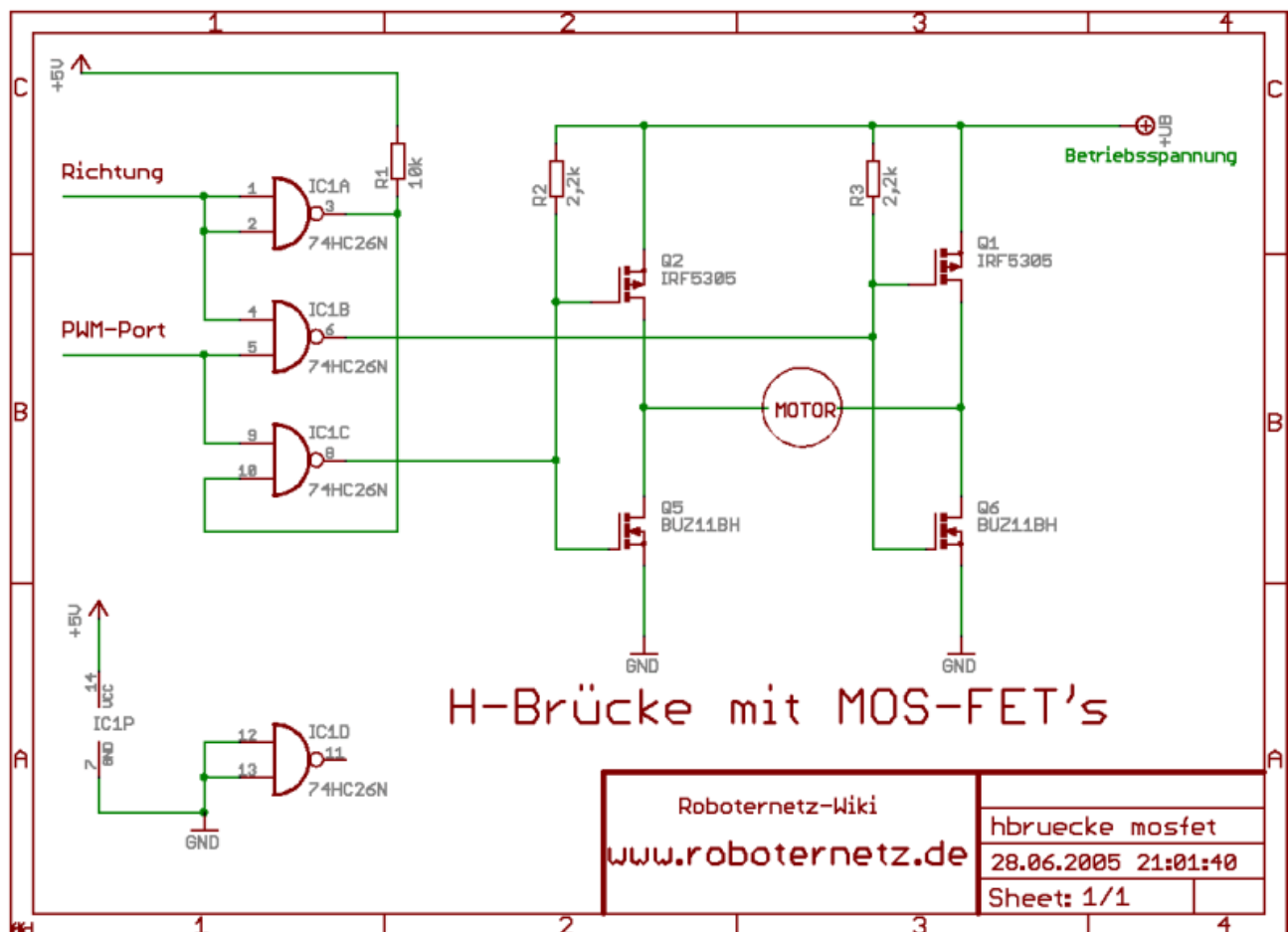
Support

Fragen einfach in das Forum posten, oder mich anschreiben (User s.o.)

Alte Schaltungsbeschreibung

Achtung! Die Schaltung ist so nicht zu verwenden! Siehe unter **Kritik an der Schaltung**

Wesentlich günstiger und auch beliebter ist die Ansteuerung von Motoren mit MOS-FETs. Also Transistoren, die quasi leistungslos nur durch Spannungen geschaltet werden und extrem hohe Ströme verkraften. Zu diesem Thema findet man unzählige Beiträge im Roboternetz. Eine der interessantesten Schaltungen, die im Roboternetz vorgestellt wurde, dürfte diese sein:

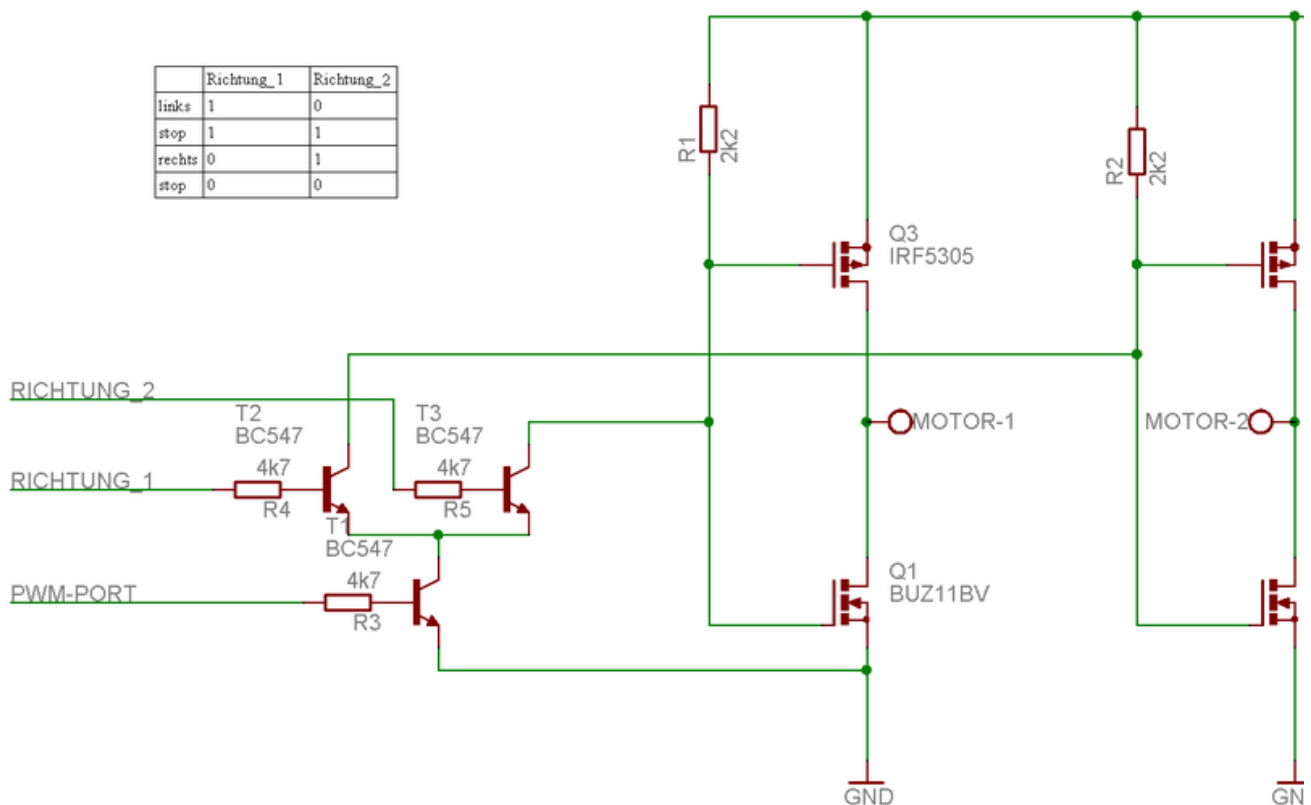


Anmerkung: Den 74HC26N gibts wohl nicht , nur die TTL-Version 74LS26, die HC-Mos-Version lautet 74HC03.

Eine sogenannte H-Brücke, die nur aus zwei MOSFETs und einem Logik-IC besteht. Mit zwei Controllerports kann diese Schaltung sowohl Geschwindigkeit als auch Drehrichtung regeln. Für die Geschwindigkeit ist ein sogenannter PWM-Port notwendig. Also ein Port, der durch ein gepulstes Signal den Motor etlichemal innerhalb einer Sekunde ein- und ausschaltet und somit quasi die Leistung regelt (siehe PWM). Die Schaltung ist so konstruiert, dass immer nur zwei Transistoren durchschalten. Auf diese Weise fließt einmal der Strom von links oben nach rechts unten und einmal von rechts oben nach links unten, der Motor wird also ähnlich wie bei der Relaischaltung umgepolt. Bei niedrigerer PWM_Frequenz sollte die Schaltung durchaus für einige Ampere geeignet sein, wobei ca. 8 bis 13 V ideal sein sollten.

Das Logic-IC wurde durch 3x BC547 Transistoren ersetzt.

	Richtung_1	Richtung_2
links	1	0
stop	1	1
rechts	0	1
stop	0	0



Kritik an der Schaltung

Diese Schaltung wird hier Forum von mehreren Leuten, u.A. Ratber, shaun, massiv kritisiert (<http://www.roboternetz.de/phpBB2/viewtopic.php?t=33067&sid=892e80c6cd2d003292e607bcd7a7123aim>) .

Ich habe leider nicht das Wissen eine bessere Lösung zu liefern. Das Problem sind die Querströme: Im Umschaltmoment sind highside und lowside switch für kurze Zeit beide leitend. Das entspricht einem Kurzschluss der Versorgungsleitung bei jedem Umschalten mit allen damit verbundenen Problemen (Erwärmung, Schwingung, Sicherungen, Reset). Aus diesem Grund sind NAND-Gatter nicht geeignet!

Zum Umschalten muss man

- den bisher leitenden MOSFET ausschalten,
- dann etwas warten,
- und dann den nächsten MOSFET schließen.

Dies kann man erreichen, indem man den Einschaltmoment

- durch RC-Glieder verzögert und
- den Ausschaltmoment durch Dioden beschleunigt.

Die entstehenden Schaltungen sind komplex, benötigen viele Bauteile und sind fehleranfällig.

Am sinnvollsten sind integrierte MOSFET-Treiber. Diese haben oft eine einstellbare Tot-Zeit (*Dead-Time*), können hohe Leistungen zum Umladen der Gatekapazität liefern und beachten andere Effekte wie *Propagation-Time*, Veränderung des Source Potentials beim Highside-switch, etc.

Erklärung von shaun:

Die gezeigte Schaltung taugt auch nur als Prinzipschaltung bedingt, da man einen variierenden Aufwand in eine sichere Verriegelung investieren müsste. Bei fixer und nicht zu hoher Betriebsspannung könnte man mit RD-Kombinationen in den Gateleitungen das Ausschalten beschleunigen und das Einschalten verzögern, so dass die Querleitung wegfällt. Allerdings bewirkt diese simple R-Cg-Verzögerung größere Schaltverluste.

Setzt man die Verzögerung vor dedizierte Treiber, müssen diese wiederum leistungsfähig genug zum Umladen der Gates sein und bis an die Versorgung heranreichen, weshalb man an diesem Punkt überlegen sollte, ob 95% duty cycle nicht auch reichen würden (*Anmerkung: Bezug auf bootstrapping?*) und man integrierte Treiber für reine N-Kanal-Bestückung einsetzen sollte - spart dann wieder

etwas Verluste am Highside-Switch, weil N-Kanäle gleicher Generation und Leistungsklasse einen niedrigeren $R_{ds(on)}$ als ihre P-Pendants haben.

Dieser Artikel ist noch lange nicht vollständig. Der Auto/Initiator hofft das sich weitere User am Ausbau des Artikels beteiligen.

Das Ergänzen ist also ausdrücklich gewünscht! Besonders folgende Dinge würden noch fehlen:

Eine vernünftige diskrete Ansteuerung für eine H-Brücke. Korrektur auf Rechtschreibfehler, Zeichensetzung, Schreibstil, Fachausdrücke. Gibt's denn keinen, der etwas verbessern möchte?

Ansteuerung mit einem LeistungsOPV

Die wohl preislich und platztechnisch günstigste Alternative zur Ansteuerung von Motoren in H-Brücken ist die Verwendung von (Leistungs-)Operationsverstärkern oder Audioverstärker ICs.

Heidingscher Motortreiber

Bei dem Heidingschen Motortreiber wird jeder Anschluss des Motors mit einem Ausgang des Operationsverstärker verbunden, die Eingänge des OPVs selbst als Komparator. Diese Methode ist für kleine und mittlere Motoren ($\leq 1A$) und mäßige PWM-Frequenzen ($< 10kHz$) gut geeignet. Ein RB35 von Conrad kann damit problemlos angesteuert werden. Bei zu hohen PWM-Frequenzen ist die SlewRate von vielen OPVs zu gering und die entsprechenden OPVs sind dann viel zu teuer und man sollte eine Ansteuerung per MOSFET (s.o.) bevorzugen. Momentan hat sich der TCA 0372 DP1 (0,57€ bei Reichelt) Verwendung gefunden. Weitere Informationen sind in diesem Forumsbeitrag (<http://www.roboternetz.de/phpBB2/viewtopic.php?t=38412>) zu finden.

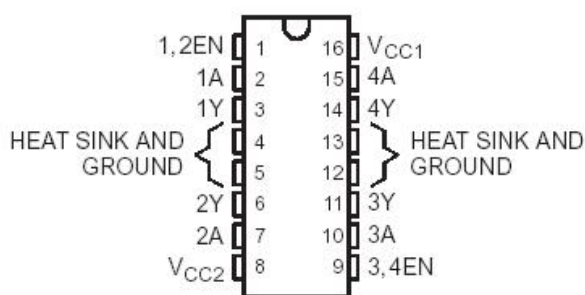
Gleichstromansteuerung

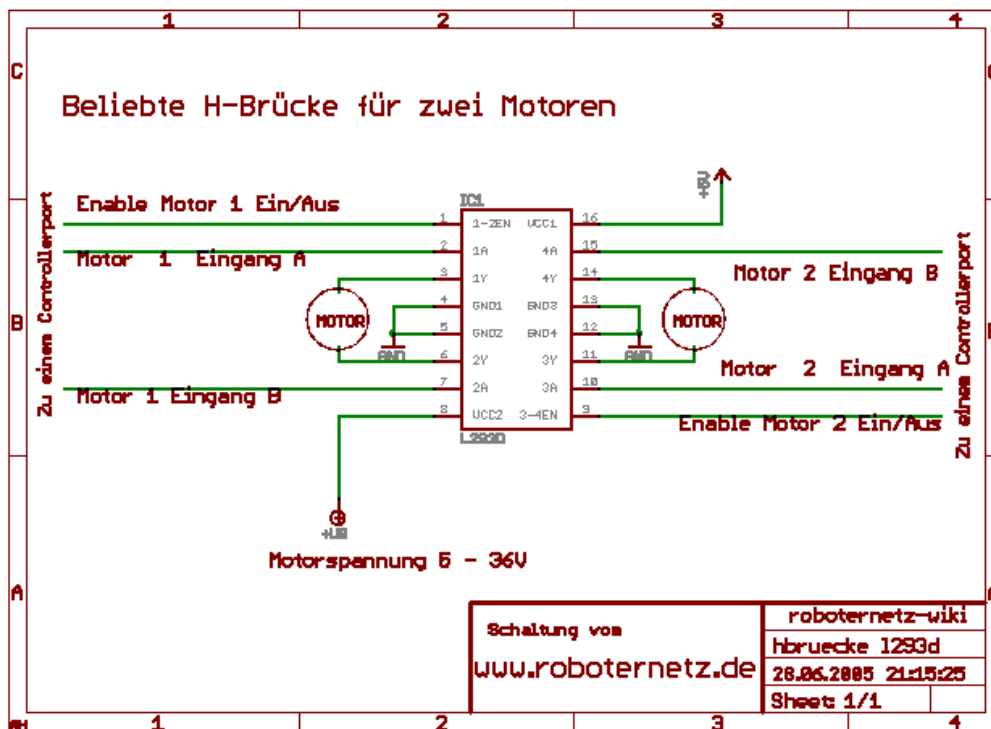
Eine weitere Möglichkeit ist es, dem Operationsverstärker eine Gleichspannung vorzugeben, mit denen der OPV die Motoren ansteuert. Diese Methode eignet sich sehr gut für analoge Schaltkreise. Der Wirkungsgrad ist allerdings nicht der beste, da die restliche Spannung in Wärme umgesetzt wird. Zudem werden viele weitere Komponenten benötigt, was den Aufbau teurer und fehleranfälliger macht. Der entsprechende Schaltkreis ist im Datenblatt des TCA 0372 DP1 zu finden.

Ansteuerung mit Treiber ICs

Ansteuerung mit Treiber IC L293 D

Dies ist ohne Zweifel die am häufigsten genutzte Ansteuerung bei Roboter-Bastlern: Man nimmt einfach das IC L293D, denn darin sind sogar zwei H-Brücken enthalten. Also mit einem IC lassen sich ohne weitere externe Bauteile gleich zwei Motoren ansteuern. Zwar nur bis ca. 600mA, aber das reicht oft schon für kleinere bis mittlere Roboteranwendungen aus.

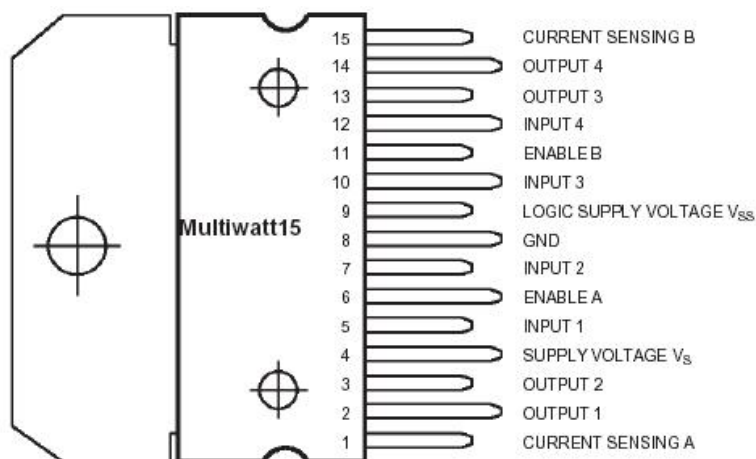




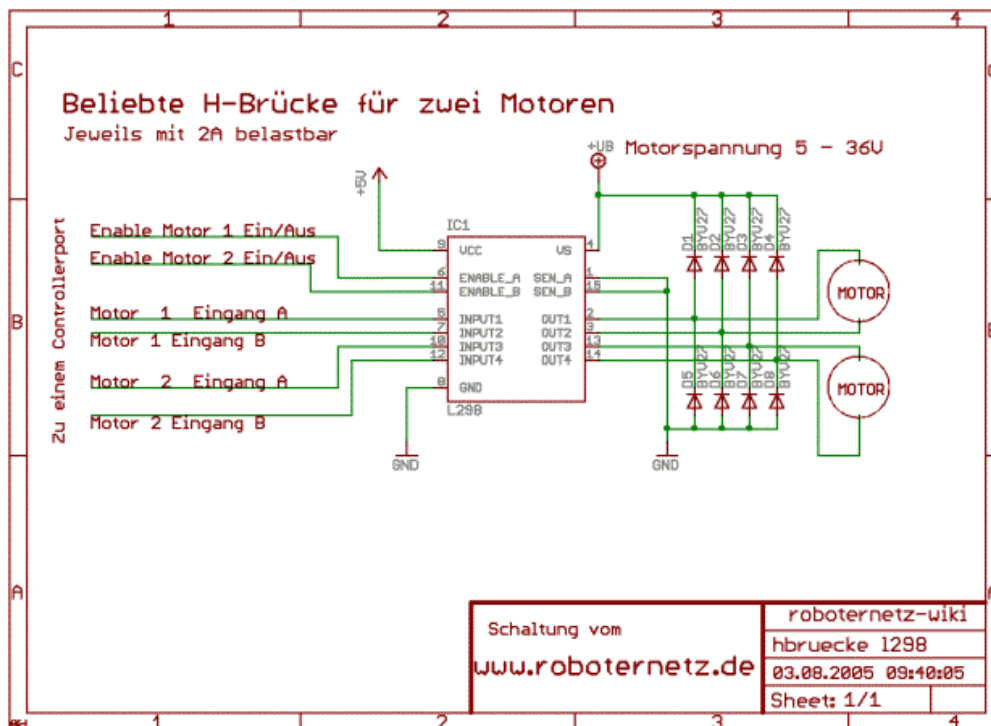
Wie aus dem Schaltbild zu ersehen ist, werden für die Ansteuerung jedes Motors 3 Ports benötigt. Die Enable-Leitung führt man oft auf einen PWM-Port welcher wie bei der MOS-FET Schaltung oben die Geschwindigkeit regelt. Die beiden anderen Ports geben die Drehrichtung an. Immer wenn die Ports unterschiedliche Polarität haben dreht der Motor in eine bestimmte Richtung, je nachdem wo Low und High anliegt. Das Besondere ist, dass wenn an beiden Ports der gleiche Pegel anliegt, also zweimal Low oder High, dann wird nämlich der Motor kurzgeschlossen, das dann als Bremse fungiert. Das Bremsen kann bei Robotern durchaus nützlich sein. Zudem ist bei schnellen Richtungswechseln immer zu empfehlen zuerst kurz zu bremsen, um nicht den Motor oder Motortreiber zu stark zu belasten.

Ansteuerung mit dem Schaltkreis L298

Der Schaltkreis L298 ist quasi der große Bruder des L293D. Er beinhaltet auch zwei komplette H-Brücken, kann also auch zwei Motoren ansteuern. Die Pinbelegung ist ebenfalls dem L293D sehr ähnlich, jedoch verfügt er über eine andere Bauform:



Der wichtigste Unterschied besteht jedoch darin, dass jede H-Brücke bei L298 bis zu 2 A belastet werden kann. Damit lassen sich also schon wesentlich größere Motoren ansteuern. Ein weiterer Vorzug sind die sogenannten SENSE-Ausgänge, über die der komplette Strom fließt. Oft wird hier ein Hochlastwiderstand angeschlossen, um aus der abfallenden Spannung den Strom berechnen zu können. Dies machen sich Steuerungen wie RN-Motor (<http://www.roboternetz.de/phpBB2/viewtopic.php?t=2741>) oder aber andere Schrittmotoransteuerungen zunutze, um den Strom zu regeln. Benötigt man keine Strommessung, so müssen die Sense-Ausgänge direkt mit GND verbunden werden. Anders als beim L293D werden hier externe Freilaufdioden gebraucht.



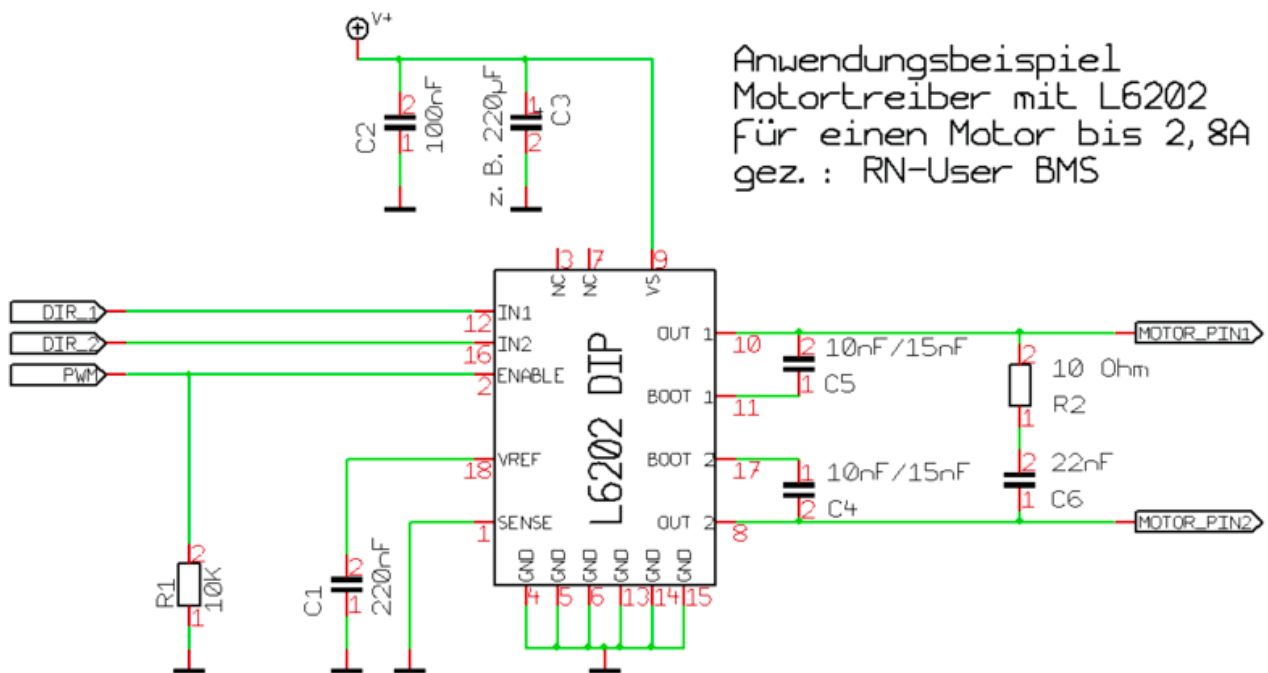
Ein weiteres Schaltbild gibt es hier [1] (<http://www.roboternetz.de/bilder/schaltung298getriebe.gif>)

Ansteuerung mit L620x

Der Schaltkreis L620x stellt eine etwas modernere Alternative zur L298-Ansteuerung dar. Die Ansteuerung ist praktisch identisch, der Schaltungsaufbau in etwa gleich. Beim L620x sind keine externen Freilaufdioden mehr notwendig, dafür ist ein Kondensator und zwei Dioden für die interne Spannungserzeugung notwendig. Ein großer Vorteil ist auch, dass kein extra Kühlkörper mehr notwendig ist. Durch die Verwendung von niederohmigen FETs im L620x fällt am Motortreiber weniger Spannung ab und die Verlustleistung des Motortreiber-ICs wird dadurch deutlich geringer. Durch die erhältliche DIP-Bauform lassen sich diese Motortreiber bei Defekt auch schnell und einfach austauschen. Viele integrierte Schutzfunktionen im L620x sorgen dafür, dass der Chip nicht so einfach zerstört werden kann.

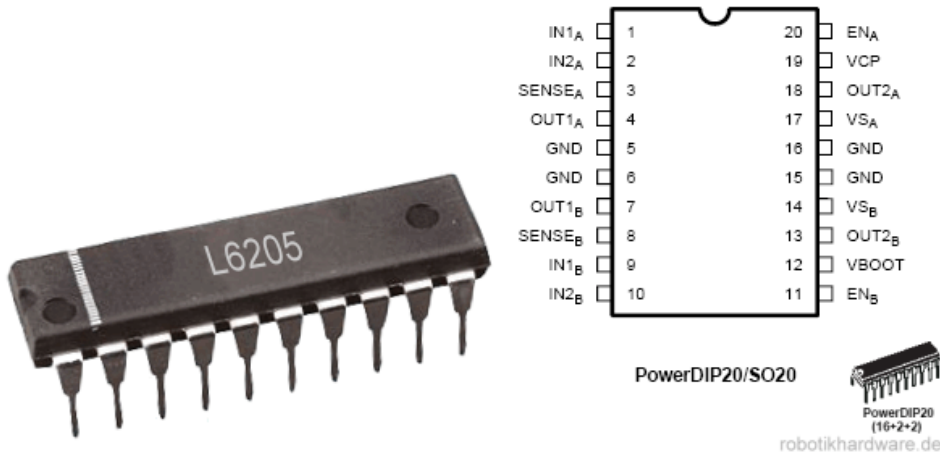
In der Praxis sind die angegebenen 2,8 A (für den L6205) jedoch mit Vorsicht zu genießen. Bei Testlayouts erwärmten sich die ICs bereits bei 2 A Dauerstrom (PWM) oft so stark, dass die automatische Temperaturabschaltung reagierte, somit wird man vermutlich ohne Kühlung oder großes Kühllayout auf der Platine auch nicht mehr als beim L298 erreichen.

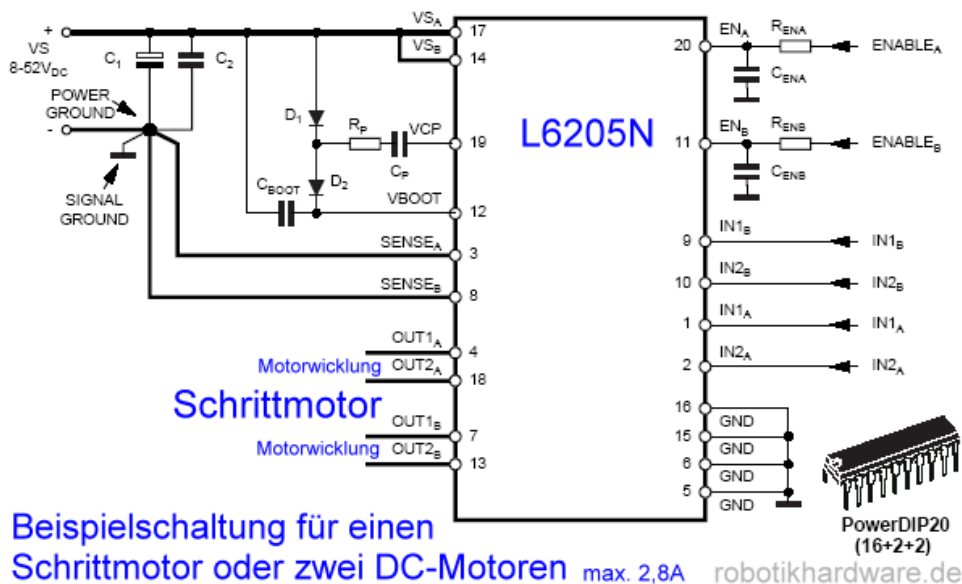
L6202 für 1 Motor



(R1 schaltet die H-Brücke bei fehlendem PWM-Signal ab und ist entbehrlich. R2 und C6 bilden ein sog. Snubber-Netzwerk, welches zur Entstörung dient.)

L6205 für 2 Motoren





weitere Treiber-ICs

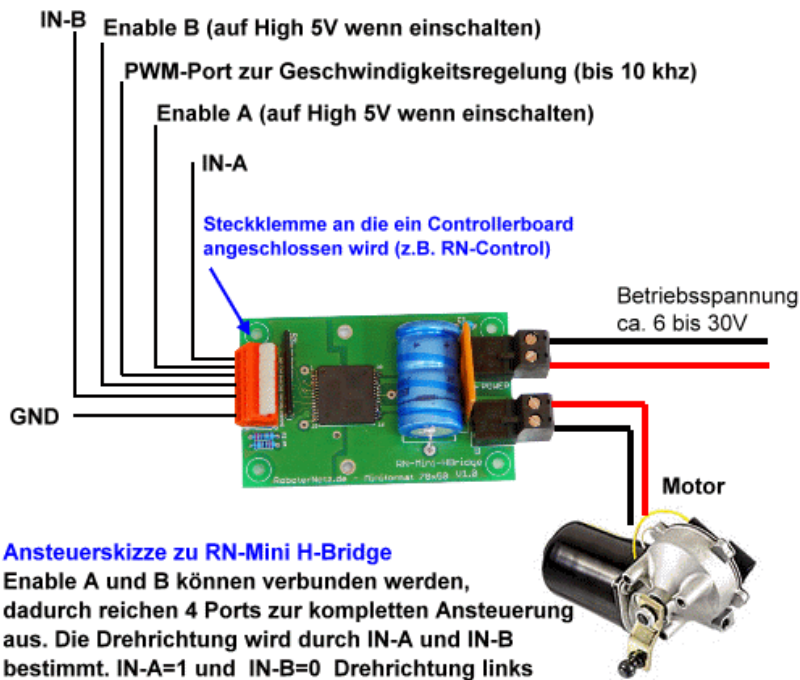
Es gibt noch mehr ähnliche Treiber ICs, die hier nur kurz aufgezählt werden sollten. Für Details sollte man ohnehin ins Datenblatt schauen.

- L293B, L293E ähnlich dem L293 D, aber max. 1 A, braucht externe Freilaufdioden
- TLE5205 max. 5 A, nur eine Brücke

Noch mehr Power gewünscht?

Obwohl der L298 schon einiges abdeckt, so kommt er spätestens bei den Scheibenwischemotoren langsam an seine Grenzen. Bei starker Belastung können solche Motoren kurzzeitig bis ca. 10 A und mehr ziehen. Für solch starke Motoren gibt es jetzt einen ganz interessanten Motorchip aus dem Kfz-Bereich: vnh3sp (Datenblatt im Roboternetz Download-Bereich). Mit ihm lassen sich sogar recht große Motoren ansteuern; vorausgesetzt, man kühlt ihn entsprechend, so verträgt der Chip bis zu 30 A. Aber selbst ohne Kühlung bietet er bedeutet mehr Leistung als der L298. Das Schöne, die Ansteuerung ist kaum anders als beim L298 und L293D.

Kleiner Nachteil: Da es ein SMD-Chip mit 1mm Kontaktabstand ist, muss man schon eine geeignete Platine (spezielles Layout mit Kühlflächen) und etwas Löterfahrung besitzen. Inzwischen gibt es aber schon verschiedene RN-Projekte mit dem Chip (RN-Power, RN-Mini H-Bridge), Platinen und Chip können über den Robotikhardware.de-Platinenservice (<http://www.shop.robotikhardware.de/shop/catalog/index.php>) bezogen werden.



Ansteuerskizze zu RN-Mini H-Bridge

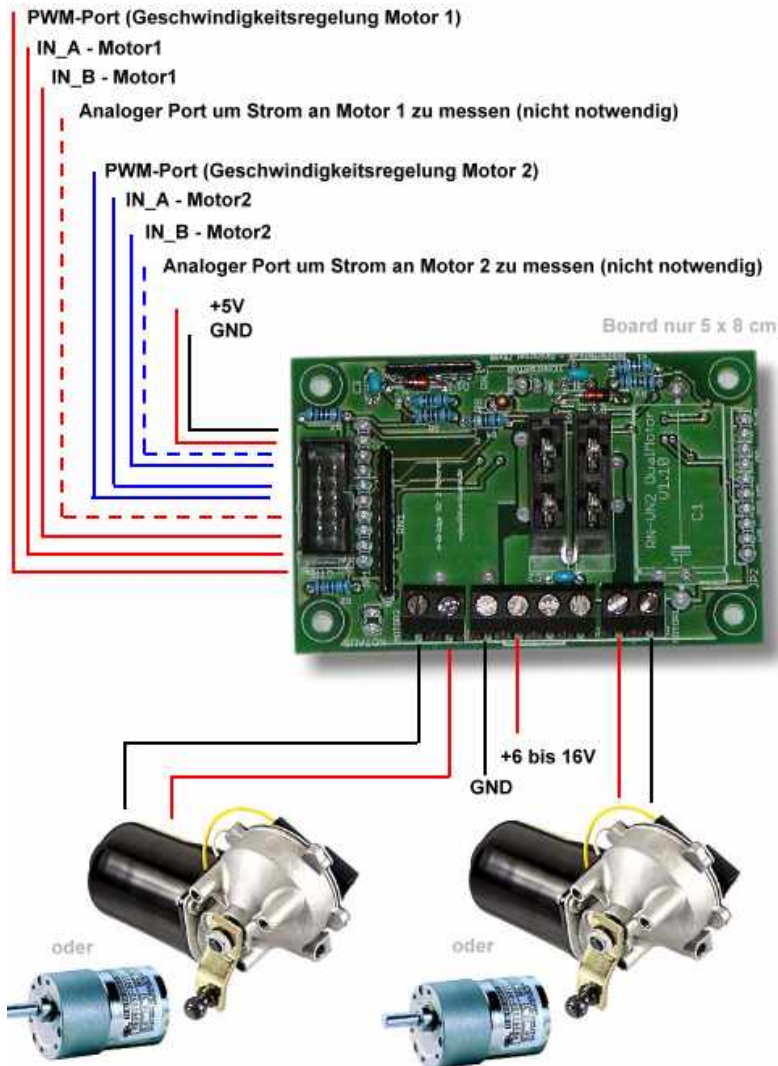
Enable A und B können verbunden werden, dadurch reichen 4 Ports zur kompletten Ansteuerung aus. Die Drehrichtung wird durch IN-A und IN-B bestimmt. IN-A=1 und IN-B=0 Drehrichtung links
IN-A=0 und IN-B=1 Drehrichtung rechts
Ist IN-A und IN-B logisch identisch dann wird gebremst

robotikhardware.de

Alternativ zum SMD-Chip VN3SP30 gibt es noch die Alternative VN2SP30. Dieser Chip erlaubt zwar nur Motorspannungen zwischen 6 und 16 Volt, jedoch hat er einen wesentlich geringeren Innenwiderstand und wird deshalb nur etwa halb so warm wie der VN3SP30. Aus diesem Grund eignet er sich auch ideal für kleine Doppelmotorsteuerungen. Natürlich gibt's auch hierfür ein Projekt und eine Bauanleitung, siehe RN-VN2Dualmotor

Das Schöne ist, dass es dieses RN-Projekt jetzt auch als Fertigmodul gibt!

Die Motorsteuerung RNVNH2 benötigt pro Motor nur 3 Ports zur Ansteuerung

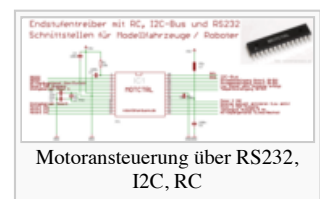


Die Drehrichtung wird durch IN-A und IN-B bestimmt. IN-A=1 und IN-B=0 Drehrichtung links
IN-A=0 und IN-B=1 Drehrichtung rechts
Ist IN-A und IN-B logisch identisch dann wird gebremst

robotikhardware.de

Endstufen mit I2C-Bus / RS232

Verwendet man eine der oben vorgestellten H-Bridges, so wird immer vorausgesetzt, dass ein PWM-Signal und mehrere Ports zur Ansteuerung bereitstehen. Ist dies nicht der Fall, weil diese vielleicht schon belegt sind, so lassen sich Endstufen durch einen zusätzlichen Microcontroller auch um beliebige Schnittstellen erweitern. Insbesondere über I2C lassen sich dadurch mehrere Motorboards über einen einzigen Bus ansteuern. Ein Beispiel ist der programmierte Controller MOTCTRL, der speziell für diese Aufgabe gedacht ist. Die Grundschialtung sieht man in der rechten Skizze.

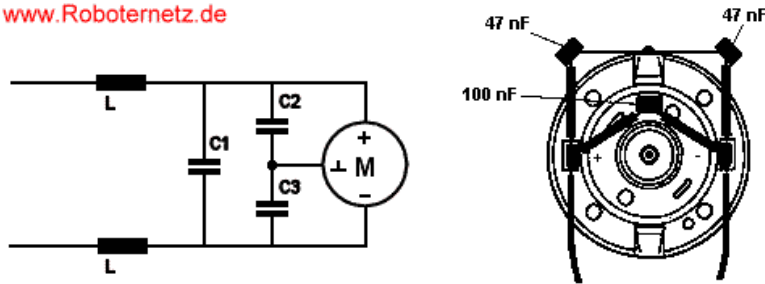


Und nie vergessen Motoren zu entstören

Das Entstören dient dazu, eine Ausbreitung von Funkstörungen durch das sogenannte "Bürstenfeuer" zu verhindern. Und so wird's gemacht:

Entstören von Elektromotoren

für www.Roboternetz.de



C1=100nf

C2 und C3 = 47nF

L = Ferritring der mit Zuleitung 6-8 mal umwickelt wird

Bei Steuerung des Motors per PWM sind die Induktivitäten hilfreich. Sie verhindern das bei den steilen Flanken des PWM Signals große Ströme durch die Kondensatoren fließen. Alternativ zu den 2 Kondensatoren (C2,C3) gegen das Gehäuse kann das Gehäuse auch direkt an Masse angeschlossen werden. Bei einer PWM Steuerung ist das die bessere Lösung.

Siehe auch

- Getriebemotoren
- RN-VNH2Dualmotor

Weblinks

- Drehmoment Berechnung (<http://www.roboternetz.de/phpBB2/motordrehmoment.php>)
- Motoren Übersicht (<http://www.roboternetz.de/motoruebersicht.html>)
- Bauteilesets (<http://www.shop.robotikhardware.de/shop/catalog/index.php?cPath=65>)
- Roboter-Tutorial (<http://www.roboternetz.de/robotertutorial.html>)
- Roboternetz Forum (<http://www.roboternetz.de>)

Von "http://www.m-wissen.de/index.php/Getriebemotoren_Ansteuerung"

Seitenkategorien: Artikelausbau | Robotikeinstieg | Praxis | Grundlagen | Motoren | Elektronik

-
- Diese Seite wurde zuletzt geändert um 18:43, 25. Nov 2010.