

Experimente mit Gleichstrommotoren

Gunther Zielosko

1. Grundlagen

Eine faszinierende Aufgabe für den Hobbyelektroniker ist die Verbindung von Mikrorechner und Mechanik. Dabei spielen Motoren und ihre Ansteuerung eine wichtige Rolle. In der Reihe der Applikationsschriften für den BASIC-Tiger wurden bereits Ansteuerungen von Servos und Schrittmotoren behandelt. Im vorliegenden Beitrag wird es um Versuche mit einem Motortyp gehen, der wohl am meisten verbreitet ist, dem Gleichstrommotor mit Permanentmagnet. Solche Motoren gibt es in vielen Ausführungen von extrem kleinen Exemplaren bis hin zum Antrieb kleiner Bohrmaschinen (Bild 1).



Bild 1 ein Sortiment von Gleichstrommotoren mit Permanentmagneten



Bild 2 Aufbau dieser Motoren, unten die beiden „Kohlebürsten“, rechts der Anker mit Kommutator, oben der ringförmige Permanentmagnet

Größere Motoren funktionieren zwar nach dem gleichen Prinzip, haben dann in der Regel aber keinen Permanentmagneten, sondern eine Statorwicklung. Ein Grund dafür sind höhere erreichbare Leistungen, aber auch der damit mögliche Betrieb mit Wechselspannung. Bei unseren Motoren mit Permanentmagneten ist der Aufbau immer identisch. Außen befindet sich meist ein ringförmiger Magnet mit entsprechenden Polführungen und innen ein Rotor mit typisch 3 bis 7 Spulen, die über einen Kommutator und sogenannte Kohlebürsten an Betriebsspannung gelegt werden (Bild 2). Rotor und Stator wirken magnetisch so aufeinander, daß sich der Rotor ein wenig dreht. Nach dieser Bewegung wird mit dem Kommutator die

Stromrichtung umgepolt und das neue Magnetfeld sorgt für eine weitere Drehung. Je höher die Anzahl der Wicklungen ist (und damit der Kommutatorsegmente), desto besser ist das Anlaufverhalten der Motoren und ihre Laufkultur.

2. Der computergesteuerte Motor

2.1. Wozu überhaupt eine Steuerung?

Man könnte nun fragen, welchen Sinn es macht, mit einem Computer (unserem BASIC-Tiger) einen simplen "Spielzeugmotor" zu steuern, ein Schalter genügt doch, um ihn laufen zu lassen. Im Laufe der Experimente werden wir aber sehen, daß es doch vielerlei Möglichkeiten gibt, solche Motoren in komplexen Systemen zu steuern:

- Im einfachsten Falle kann der Motor durch die Software ein- bzw. ausgeschaltet werden.
- Mit zwei Ausgängen des BASIC-Tigers können wir den Motor links- oder rechtsherum laufen lassen.
- Mit einer eleganten Ansteuerung läßt sich die Drehzahl einstellen.
- Durch einen Sensor wird es möglich, eine bestimmte Anzahl von Umdrehungen in beiden Richtungen vorzuprogrammieren, so daß sich der Antrieb ähnlich wie ein Schrittmotorantrieb verhält.

2.2. Die Schaltung des Interfaces

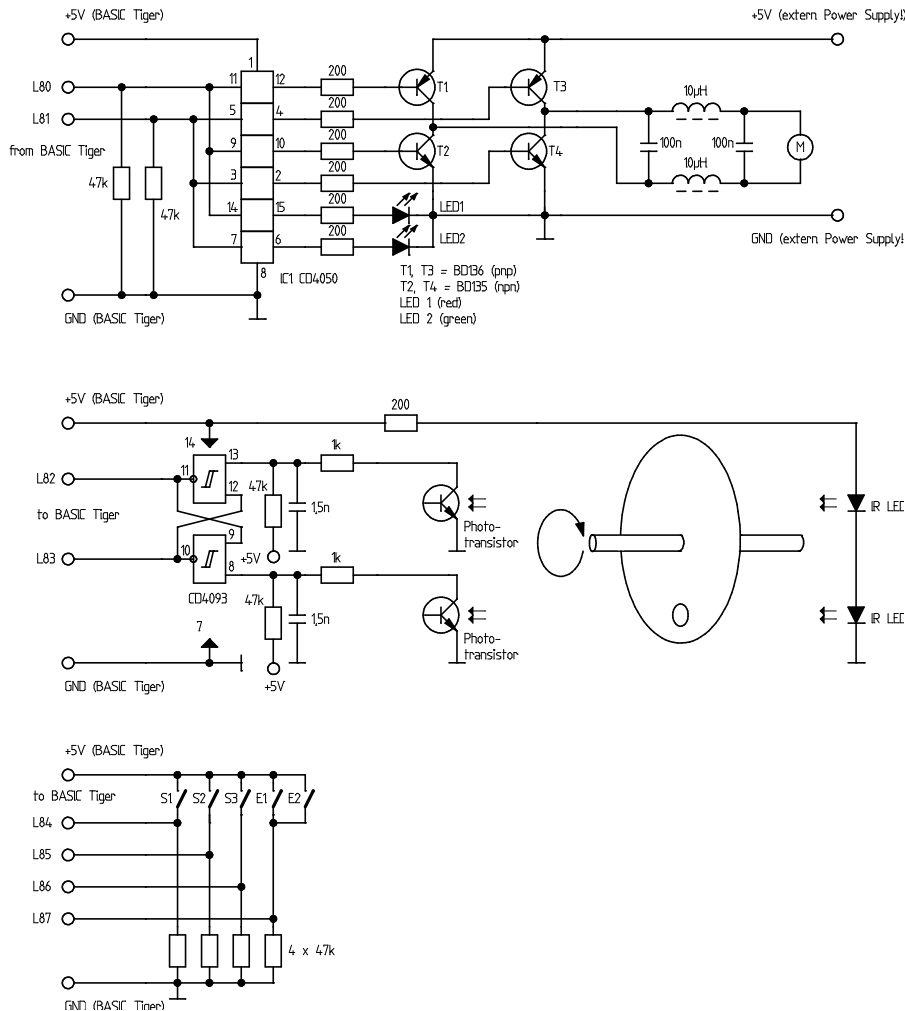


Bild 3 Die komplette Motoransteuerung

Die komplette Schaltung des Interfaces zeigt Bild 3. Dieses besteht aus 3 Teilen, der Motorbrücke (oben), der Lichtschrankenschaltung (mitte) und der Tastatur (unten). Für erste Experimente mit den Programmen MOTOR01.TIG...MOTOR04.TIG genügen der obere Teil, die eigentliche Motorbrücke und die Tastatur.

Für die Motoransteuerung allein benötigen wir zunächst nur zwei Ausgänge des BASIC-Tigers, hier L80 und L81. Diese werden jeweils an 3 Gatter eines CD4050 (6 nichtinvertierende Treiber) gelegt. Die beiden 47kΩ-Widerstände an den Eingängen dienen zur logischen Pegelfestlegung, falls der BASIC-Tiger fehlt oder diese Pins z.B. fälschlicherweise als Eingänge programmiert wurden. Die oben gezeichneten vier Ausgänge des CD4050 steuern die Transistoren für die Motorbrücke an, zwei weitere dienen zur optischen Signalisierung der Eingangsdaten (können ggf. auch weggelassen werden).

Die Funktion ist einfach. Die 4 Transistoren T1 bis T4 arbeiten als Brückenschaltung. Durch die Verkopplung der Eingänge des CD4050 wird gewährleistet, daß in jedem (senkrechten)

Brückenweig nur ein Transistor durchschalten kann. Bei Highpegel an L80 werden die Transistoren T1 und T2 an der Basis ebenfalls mit High beschaltet. Für den PNP-Transistor T1 bedeutet dies Sperrung (Basis praktisch auf Emitterpotential), für den NPN-Transistor T2 dagegen Durchschalten (Basis positiver als Emitter). Schaltet T2 durch, heißt das Verbindung mit Masse und damit, daß der untere Pol des Motors auf Masse liegt. Spielt man alle Kombinationen durch, ergeben sich folgende Möglichkeiten (Tabelle 1):

L80	L81	Unterer Motoranschluß	Oberer Motoranschluß	Resultat	LED1 rot	LED2 grün
0	0	Masse	Masse	Motor dreht sich nicht	aus	aus
0	1	Masse	+5V	Linkslauf	aus	an
1	0	+5V	Masse	Rechtslauf	an	aus
1	1	+5V	+5V	Motor dreht sich nicht	an	an

Tabelle 1 Motorfunktionen der Brückenschaltung

Die beiden Drosseln sowie die Kondensatoren dienen zur Funkentstörung und schützen auch ein wenig vor schädlichen Spannungen, die beim Betrieb des Motors auf die Elektronik einwirken könnten. Noch ein paar Hinweise für den praktischen Betrieb:

- Die Schaltung funktioniert grundsätzlich nur für Kleinmotoren!
- Die zusätzliche Spannung 5V für den Motor sollte nicht aus dem Plug-and-Play-Lab genommen werden, um Störungen oder Schäden an der empfindlichen Rechnerumgebung zu vermeiden! Benutzen Sie ein separates Netzteil oder Batterien! Verbinden Sie die Minuspole (GND) der beiden Spannungsquellen miteinander!
- Diese Zusatzspannung für die Motorbrücke darf nicht größer als 5V sein!
- Bei Inbetriebnahme ist zuerst die Rechnerspannung und dann die Motorspannung einzuschalten, da es sonst zu Fehlfunktionen oder Zerstörungen am CD4050 oder gar am Plug-and-Play-Lab kommen kann. Beim Abschalten zuerst die Motorspannung und dann die Rechnerspannung abschalten!

Den unteren Schaltungsteil brauchen wir für die Bedienung. Es gibt 3 Tasten (S1...S3), mit denen wir den Motor direkt steuern werden. Die beiden Kontakte E1 und E2 sind Endschalter, die für richtige Antriebe gedacht sind, bei denen der Motor beim Erreichen einer der beiden Endstellen abschalten soll. Dazu später mehr.

2.3. Er dreht sich...

Wenn die Interfaceschaltung mit dem Plug-and-Play-Lab (oder mit einem separaten BASIC-Tiger-Aufbau) verbunden wird, kann es schon losgehen. Gemäß Tabelle 1 muß sich der Motor bereits drehen, wenn L80 und L81 auf unterschiedlichem Logikpegel liegen, also z.B. L80 = 1, L81 = 0. Das ist nicht besonders spektakulär, mit dem kleinen Programm **MOTOR01.TIG** wird es schon besser, nach Drücken der Taste S1 (auf dem Interface!) soll er linksherum, beim Betätigen der Taste S3 rechtsherum laufen, bei Drücken der Taste S2 soll er stehenbleiben.

Für Antriebe, bei denen es auf gute Dosierbarkeit ankommt, reicht dies nicht. Hier wäre es besser, wenn der Motor beim Loslassen der beiden Tasten S1 bzw. S3 allein stoppt. Mit dem Programm **MOTOR02.TIG** läßt sich z.B. ein Kranmodell recht gut steuern.

Als nächstes wollen wir die Drehzahl des Motors variabel gestalten. Prinzipiell kann man einen Gleichstrommotor in der Drehzahl u.a. dadurch steuern, daß man ihn impulsförmig an- und ausschaltet. Je nach dem Verhältnis von "Anzeit" und "Auszeit" läuft er schneller oder langsamer. Das Programm **MOTOR03.TIG** steigert die Drehzahl mit der Taste S3 und senkt sie mit der Taste S1, gut für eine kleine Bohrmaschine.

Das nächste Programm **MOTOR04.TIG** treibt es noch weiter. Jetzt kann man mit der Taste S3 die Drehzahl rechtsherum steigern. Mit S1 wird der Motor langsamer, kommt zum Stillstand und dreht sich dann erst langsam, später schneller linksherum, ideal für ein Fahrmodell mit "Gas", "Bremse" und "Rückwärtsgang". Spezialisten können die Maximalgeschwindigkeit für Vorwärts- und Rückwärtsfahrt auch separat einstellen...

2.4. Ein zusätzlicher Sensor

Mit einem laufenden Motor kann man noch relativ wenig anfangen, auch wenn er noch so schön per Mikrocomputer gesteuert wird wie unserer. Interessanter sind richtige Antriebe mit Getriebe, die dann zum Beispiel eine Spindel (Gewindestange) antreiben. Auf der Spindel wird eine passende Mutter hin und hergeschoben, das ergibt einen einfachen "Linearantrieb". Die Bilder 4 und 5 zeigen beispielhaft solch einen Aufbau, er könnte z.B. als Antrieb für einen Fahrstuhl (-modell!) benutzt werden.

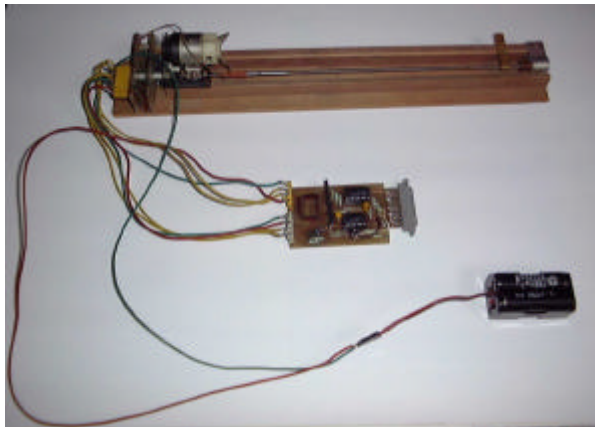


Bild 4 Modell eines Linearantriebes mit Gleichstrommotor, Getriebe, Spindel, Batterien und Interface

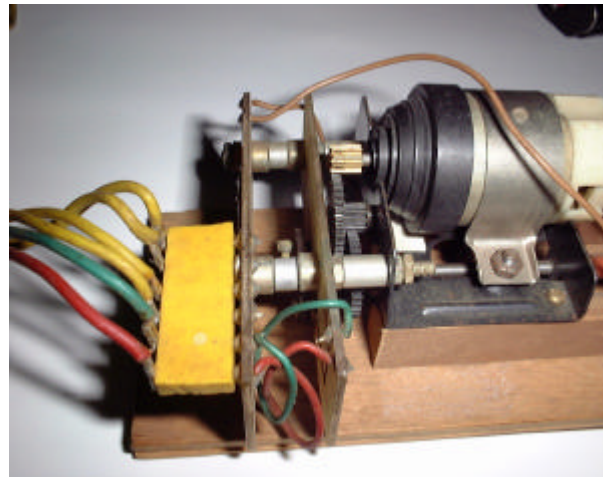


Bild 5 Detail mit Motor, Getriebe, Lichtschranken (auf den beiden senkrechten Leiterplatten)

Dazu wäre es wichtig, eine Positionsmeldung einzubauen. Wenn man beispielsweise die Umdrehungen der relativ langsamen Spindel zählen könnte, wüßte der Rechner sofort, wo sich der "Fahrstuhl" befindet. Genau das wollen wir jetzt ausprobieren. Mit dem mittleren Teil der Schaltung nach Bild 3 bekommen wir bei jeder Umdrehung einen Impuls, den der BASIC-Tiger auswerten kann.

Kernstück sind zwei Infrarot-Lichtschranken bestehend aus IR-LED und Phototransistor, die wechselseitig durch eine auf der Antriebswelle montierte Lochscheibe mit einem einzigen Loch aktiviert werden. Jedesmal, wenn das Loch einen Lichtstrahl freigibt, wird der entsprechende Phototransistor leitend und zieht den Pegel an dem dazugehörigen Gattereingang auf Masse. Damit kippt das aus zwei NAND-Gattern mit Schmitt-Trigger-Eingang gebildete Flip-Flop in einen neuen Zustand. Kommt das Loch an der um 180° versetzten anderen Lichtschranke vorbei, kippt es zurück. Mit dieser Schaltung ist gewährleistet, daß es keine undefinierten Zustände z.B. beim langsamen Durchgang des Loches geben kann. Das Schmitt-Trigger-Verhalten der Gatter hat im übrigen denselben Zweck, mit normalen Gattern könnte es Prellerscheinungen geben, die die Zählfunktion stören könnten.

Die Lichtschrankenelemente LED und Phototransistor sind normale Infrarottypen, die Auswahl richtet sich unter anderem nach den mechanischen Vorgaben. Die Widerstände 47kΩ sind ggf. anzupassen. Der Pegel an den Kollektoren der Phototransistoren müssen optimiert werden, im Beleuchtungsfall nahe GND und ohne LED-Beleuchtung nahe +5V. Fremdlicht sollte weitgehend abgeschirmt werden. Es funktioniert alles ordentlich, wenn bei Drehung der abgehenden Getriebeachse Achse durch den Motor am Ausgang des Flip-Flops etwa zeitgleich abwechselnd High- und Lowpegel gemessen wird. Das kann man übrigens sehr gut testen, wenn man einen Flip-Flop-Ausgang direkt mit dem Beep-Anschluß des Plug-and-Play-Labs verbindet. Beim Drehen der Getriebeachse sollte jetzt ein intermittierender Ton zu hören sein, auch die LED des Pins L82 muß blinken.

Mit dem Programm **MOTOR05.TIG** kann die Interfaceerweiterung sofort getestet werden. Mit den Tasten S1 (erniedrigen) und S3 (erhöhen) wird ausgehend von "0" eine Anzahl von Umdrehungen der abgehenden Getriebewelle vorgegeben und auf dem üblichen BASIC-Tiger-Display angezeigt. Nach Drücken der Taste S2 (sozusagen ENTER) läuft der Motor in der entsprechenden Richtung los und stoppt nach dem Erreichen der vorgegebenen Schrittzahl. Die Vorgabe sowie die jeweils erreichte Position werden auf dem Display angezeigt.

Noch ein paar Hinweise zum Aufbau und zum Betrieb. Wichtig ist zunächst, daß der Antrieb im gesamten Bereich "klemmfrei" arbeiten kann. Für erste Experimente sollte der Spindelantrieb zunächst so eingestellt werden, daß die Mutter etwa in der Mitte positioniert ist, damit es genügend Platz für Bewegung in beiden Richtungen gibt. Bei der Vorgabe der Umdrehungen über die Tastatur sollten Sie vorerst sehr "sparsam" sein und kleine Zahlen eingeben, damit der Antrieb nicht über das Ende hinaus läuft.

Als Abschluß folgt das Programm **MOTOR06.TIG**, das einen Positionierantrieb simuliert, wie er z.B. für Fahrstühle oder numerische Maschinen gebraucht wird. Wieder über die Tasten S1 und S3 erfolgt die Vorgabe der Position, mit S2 die Übernahme in den Rechner. Im Programm können Maxima für die Position festgelegt werden, so daß unsinnige Werte vermieden werden (der Fahrstuhl fährt über das oberste Stockwerk hinaus...). Hier errechnet der BASIC-Tiger die Anzahl der erforderlichen Schritte für die neue Position aus der alten Position. Voraussetzung ist eine "Initialisierung", d.h. alles ist nur richtig, wenn von Position 0 aus gestartet wird. Für eine solche Initialisierung gibt es mehrere Wege:

- Wir bringen den Antrieb von Hand (oder durch kurzes Anlegen der Motorspannung direkt an den Motor, Rechner vorher abziehen!!!) einmal in die gewünschte Ausgangsposition. Im Betrieb fahren wir den Antrieb dann "jeden Abend" durch Eingeben der Position "0" in diese Endstellung. Am "nächsten Morgen" geht der Rechner bei Einschalten von der Position "0" aus. Diese Arbeit könnte natürlich auch ein eigenes Programmstück übernehmen, mit dem Sie vor dem Aktivieren des Hauptprogramms zunächst wie im Programm MOTOR02.TIG den Motor frei bewegen und damit in die Position 0 fahren können.
- Das Programm MOTOR06.TIG wird so geändert, daß der Motor zuerst mit S1 und S3 analog zum Programm MOTOR02.TIG in die gewünschte 0-Position gefahren werden kann und dann erst das eigentliche Positionierprogramm folgt.
- Wir bauen einen zusätzlichen Endschalter ein, im Bild 3 sind zwei Endschalter (E1 und E2) vorgesehen, die mit L87 ausgewertet werden können. Die Software wird dann so umgestellt, daß der Antrieb beim Einschalten solange in die Richtung des Endschalters fährt, bis dieser betätigt wird, das Signal wird mit dem BASIC-Tiger-Eingang L87 ausgewertet. Der BASIC-Tiger stellt den Positionszähler auf "0" und initialisiert damit das System. Diese Lösung ist eleganter, entscheiden Sie selbst...

3. Software

Die Funktion der 6 BASIC-Tiger-Programme wurde bereits im Text erklärt. Im Anhang befinden sich die kompletten Listings aller Programme. Viel Spaß beim Experimentieren mit computergesteuerten Gleichstrommotoren.