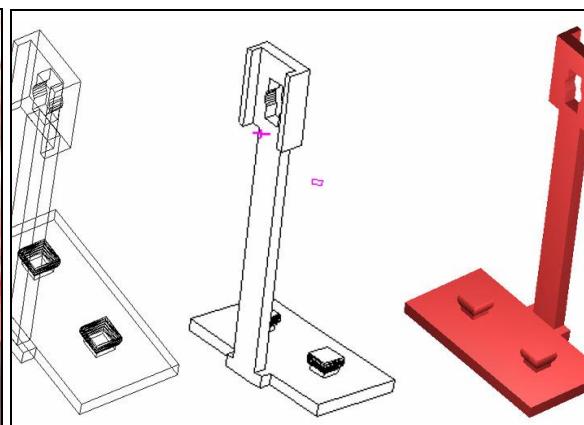
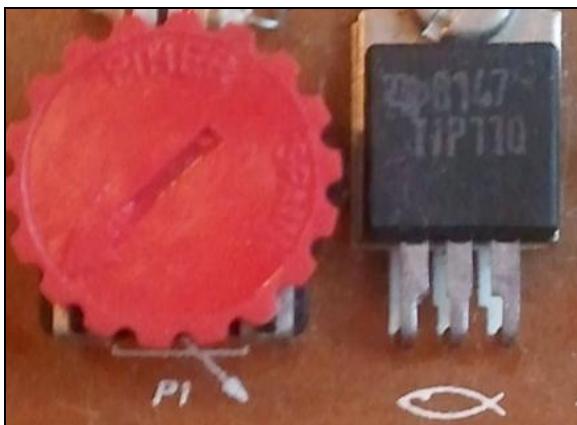
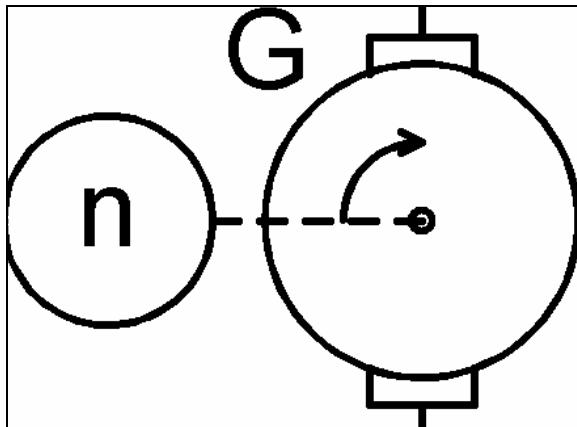


ft:pedia

Heft 3/2013



Herausgegeben von
Dirk Fox und Stefan Falk

ISSN 2192-5879

Editorial

Alarmsignale

Die Signale sind nicht neu – seit Jahren beklagen Unternehmen fehlenden Ingenieurnachwuchs – aber alarmierend. So entwickelt sich die [Altersstruktur von deutschen Ingenieuren](#) beängstigend: 21 % sind mindestens 55 Jahre alt, aber nur 18 % unter 35 Jahren. In Frankreich ist das Verhältnis umgekehrt – 42 % sind unter 35, 12 % über 54. Wird diese Tendenz nicht bald nachhaltig korrigiert, dürfte es in wenigen Jahrzehnten Essig sein mit „Made in Germany“.

Ursächlich für diese Entwicklung sind sicher nicht die Hochschulen – die Ingenieursausbildung genießt, den europäischen Vereinheitlichungen zum Trotz, nach wie vor einen guten Ruf. Hingegen gibt es zahlreiche Indizien dafür, dass das Fehlen von Technikbezügen in der Kindheit die Begeisterung für Ingenieurwissenschaftliche Berufe erst gar nicht entstehen lässt. So hat die Deutsche Akademie für Technikwissenschaften (akatech) bereits 2009 in einer Studie ([Nachwuchsbarometer Technikwissenschaften](#)) festgestellt, dass sich „klassische Spielbezüge wie Modellbahnen oder experimentelle Baukästen und manuelle Tätigkeiten (mit Ausnahme der bekannten Bausteine) (...) immer seltener in den Kinderzimmern (...) finden (...). Dabei vollzieht sich ein Wandel in den technischen Bezügen vom Konstruktiven zum Konsumtiven beziehungsweise vom Praktischen zum Abstrakten (z. B. Simulationen). Das hat zur Folge, dass Kinder sich heute nur noch selten mit Fragen der Funktionsweise von Technik auseinandersetzen.“ Fast 60 % der befragten Schüler

Dirk Fox, Stefan Falk

besaßen gar keine technischen Spielbezüge – hingegen hatten nur 8 % der befragten Ingenieure in ihrer Kindheit keinen Technikbezug. Diese Entwicklung hat offenbar einen größeren Einfluss auf das spätere Interesse an technischen Berufen als der Schulunterricht – denn über 35 % der heutigen Ingenieure hatten keinen Technikunterricht in der Schule, gegenüber nicht einmal 20 % aller heutigen Schüler. Im Fazit scheinen die Autoren der Studie zu kapitulieren: „*Ein Zeitalter der über Generationen erfolgreichen mechanischen, spielerischen Aneignung von Technik geht zu Ende.*“

Kann es denn wirklich sein, dass eine ganze Generation flächendeckend die Begeisterung für die Funktionsweise von Technik und das spielerische, kreative Lösen technischer Problemstellungen verliert? Wer seine eigenen Kinder beim Spielen mit fischertechnik beobachtet, kann das einfach nicht glauben.

Vielleicht sollte man die Autoren zur ft-Convention einladen, um ihnen zu zeigen, dass es „gallische Dörfer“ gibt, die sich diesem Trend wirksam widersetzen.

Und wenn der nächste Kindergeburtstag in eurer Familie ansteht: Ignoriert den Wunschzettel und schenkt ein wenig echte Faszination.

Beste Grüße,
Euer ft:pedia-Team

P.S.: Am einfachsten erreicht ihr uns unter ftpedia@ftcommunity.de oder über die Rubrik *ft:pedia* im [Forum](#) der ft-Community.

Inhalt

Alarmsignale	2
Der Elektromotor.....	4
Kenndaten der ft-Motoren	9
Automatik für weichen Motorstart und -stopp.....	30
ft-Spezialteile made by TST (Teil 5).....	36
Raffiniertes mit Achsen	38
Erstellung von Bauteilen für den ft-Designer.....	42
I ² C mit dem TX – Teil 6: GPS-Sensor	54
Gründung des ftc Modellbau e. V.....	63

Termine

Was?	Wann?	Wo?
fischertechnik Convention 2013	28.09.2013	Erbes-Büdesheim
Modellschau Münster	10.11.2013	HBZ Münster (Anmeldung: TST)

Hinweise

Fundsache: Auf den Webseiten des Deutschen Museums in München können die [Aufsätze der Zeitschrift „Kultur & Technik“](#) aus den Jahrgängen 1977 bis 2010 kostenlos als pdf-Datei heruntergeladen werden.

Impressum

<http://www.ftcommunity.de/ftpedia>

Herausgeber: Dirk Fox, Ettlinger Straße 12-14,
76137 Karlsruhe und Stefan Falk, Siemensstraße 20,
76275 Ettlingen

Autoren: Stefan Falk (steffalk), Dirk Fox (Dirk Fox), René Trapp (H.A.R.R.Y.); Werner Hasselberg, Andreas Tacke (TST), Johannes Visser (Daisayah).

Copyright: Jede unentgeltliche Verbreitung der unveränderten und vollständigen Ausgabe sowie einzelner Beiträge (mit vollständiger Quellenangabe: Autor, Ausgabe, Seitenangabe ft:pedia) ist nicht nur zulässig, sondern ausdrücklich erwünscht. Die Verwertungsrechte aller in ft:pedia veröffentlichten Beiträge liegen bei den jeweiligen Autoren.

Antriebstechnik

Der Elektromotor

Dirk Fox

Aus fischertechnik-Baukästen sind Elektromotoren nicht wegzudenken – schon in der „grauen Frühzeit“ ließen der mot.1 (1967) und der mini-mot.1 (1969) Kinderherzen höher schlagen. Aber weiß auch jeder, wie ein solcher Motor funktioniert? Dabei lässt sich ein Elektromotor sogar mit fischertechnik konstruieren.

Geschichte

Vor nicht einmal 200 Jahren, im Jahr 1819, beobachtet der dänische Physiker und Chemiker [Hans Christian Ørsted](#) (1777-1851), dass fließender Strom in einem Leiter eine Kompassnadel ablenkt. Diese Entdeckung „elektrisiert“ die europäischen Naturwissenschaftler; viele stellen das Experiment nach. Auch der französische Forscher [André-Marie Ampère](#) (1775-1836) hört von Ørsteds Versuchen und kann mit einer von ihm konstruierten ‚Stromwaage‘ zeigen, dass sich elektrische Leiter, die in derselben Richtung von Strom durchflossen werden, anziehen. Er weist nach, dass fließender Strom in einem Leiter ein kreisförmiges Magnetfeld erzeugt, dessen Ausrichtung sich mit der ‚Rechte-Hand-Regel‘ bestimmen lässt: Hält man den rechten Daumen in Richtung des Stromflusses (vom Plus- zum Minuspol), dann zeigen die gekrümmten anderen Finger der rechten Hand die Richtung des kreisförmigen Magnetfelds (vom Nord- zum Südpol) an [1].

Das Magnetfeld lässt sich erheblich verstärken, indem man den isolierten elektrischen Leiter um einen Eisenkern wickelt. Der Eisenkern bildet dann einen Stabmagneten mit einem Nordpol am einen und einem Südpol am anderen Ende – dessen magnetische Ausrichtung man

„umpolen“ kann, indem man die Stromflussrichtung wechselt. Diese Konstruktion nennt Ampère Elektromagnet [1].

Im darauf folgenden Jahr gelingt es dem Engländer [Michael Faraday](#) (1791-1867), einen von Strom durchflossenen Leiter um einen Dauermagneten kreisen zu lassen – das Prinzip des Elektromotors ist entdeckt [2]. 1822 gelingt es dem britischen Physiker [Peter Barlow](#) (1776-1862) ein Rad anzutreiben, das er innerhalb eines Magnetfelds radial (vom Mittelpunkt zum äußeren Rand) von Strom durchfließen lässt, indem er den Rand in eine leitende Flüssigkeit taucht – das nach ihm benannte ‚Barlow-Rad‘ (Abb. 1).

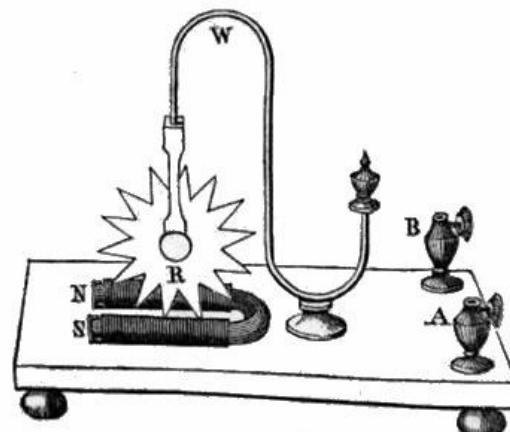


Abb. 1: Das Barlow-Rad von 1822
(Aus: *Manual of Magnetism*, 1842)

Im Jahr 1824 entwickelt [William Sturgeon](#) (1783-1850) den ersten Elektromagneten,

eine Spule mit Eisenkern zur Verstärkung des magnetischen Felds. Damit gelingt wenig später, im Jahr 1827, dem ungarischen Physiker [Ányos István Jedlik](#) (1800-1895) die Konstruktion eines ‚elektromagnetischen Rotors‘ (Abb. 2), der sich zwar noch nicht als Antrieb eignete, aber bereits alle Elemente eines (Gleichstrom-) Elektromotors enthielt: einen Stator, einen Rotor und einen Polwender (*Commutator*).



Abb. 2: Elektromagnetischer Rotor von Jedlik
(Quelle: Wikipedia [3])

Funktionsweise

Jedliks Rotor kommt der Funktionsweise heutiger Elektromotoren schon sehr nahe. Im Kern beruht ein Elektromotor auf der Anziehung und Abstoßung wechselnder (elektro-) magnetischer Felder, die an einem Stator und einem drehbaren Rotor erzeugt werden. Durch eine Änderung der Stromflussrichtung in den verwendeten Elektromagneten wird die Polung des Magnetfelds vertauscht – wenn das jeweils im richtigen Moment geschieht, veranlasst dies den Rotor zu einer gleichmäßigen Drehbewegung.

In dem Baukasten [Technical Revolutions](#) (2010) findet sich ein einfaches Experiment, das die Funktionsweise des Elektromotors veranschaulicht. Dazu wird eine Spule, genauer: eine Wicklung aus lackiertem Draht (Leiterdraht, 139250), an den Eingängen zweier Lampenfassungen lose ‚eingehängt‘ und unter der Spule ein starker (Neodym-) Dauermagnet montiert. Sobald Strom (in der richtigen Richtung)

durch die Spule fließt, stößt sie der Dauermagnet ab. Da an den seitlichen Drahtenden der Spule die Isolation nur auf der Unterseite des Drahts weggekratzt ist, wird die Stromzufuhr nach einer halben Drehung unterbrochen. Die Spule (ein einfacher Elektromagnet) dreht sich weiter, bis die Stromzufuhr erneut einsetzt und der Dauermagnet sie wieder abstoßt (Abb. 3).

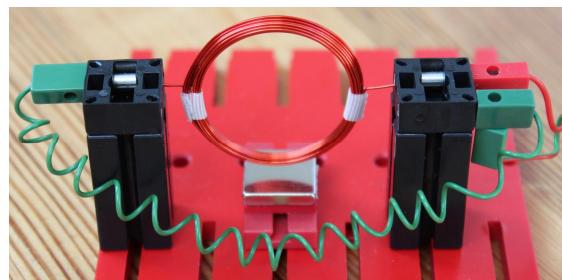


Abb. 3: Experiment aus dem Baukasten
[Technical Revolutions](#)

Das ganze ist eine etwas wackelige Angelegenheit, und das Drehmoment ist äußerst gering; daher kann die Spule (wie auch der elektromagnetische Rotor von Jedlik) nicht als Antrieb genutzt werden.

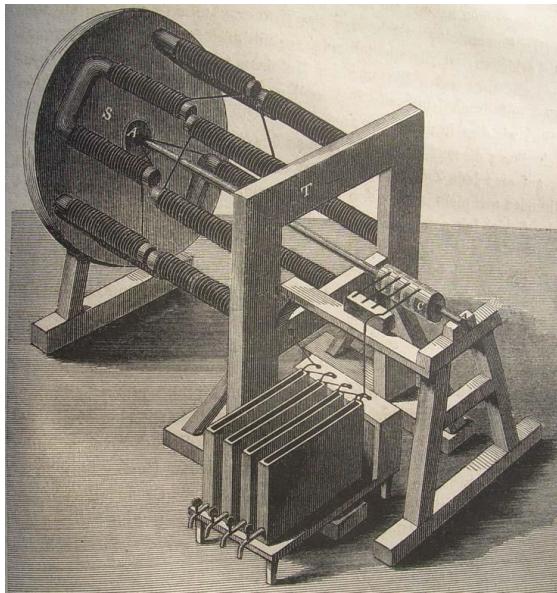
Aber das Prinzip des Elektromotors wird daran sehr schön deutlich, und die Spule erreichte in unserem Nachbau des Experiments immerhin 200-250 U/min (gemessen mit einem Fototransistor am Zählereingang des TX).

Jakobi-Motor

Der erste für die Verrichtung mechanischer Arbeit gedachte Elektromotor wurde 1834 von dem gebürtigen deutschen Ingenieur [Moritz Hermann von Jakobi](#) (1801-1874) in Petersburg konstruiert (Abb. 4) [4]. Vier Jahre später, im Jahr 1838, gelang Jakobi damit der Antrieb eines Passagierbootes auf der Neva; der von ihm konstruierte Motor erreichte ca. 200-300 W Leistung und das Boot eine Geschwindigkeit von 2,5 km/h.

Die Konstruktion litt allerdings unter den schweren, teuren, schwachen und unzuverlässigen Zink-Batterien – im Vergleich mit

den verbreiteten Dampfmaschinen war der Elektromotor daher zum damaligen Zeitpunkt unwirtschaftlich.



*Abb. 4: Jakobi-Motor von 1834
(Aus: Julius Dub, Die Anwendung des Elektromagnetismus, 1873)*

Funktionsfähige Modelle des Jakobi-Motors aus Abb. 4 können am Karlsruher Institut für Technologie (KIT, Abb. 5) und im Deutschen Museum in München bewundert werden (Abb. 6).



Abb. 5: Funktionsfähiger Nachbau des Jakobi-Motors am KIT: 160 U/min., 15 W [5]

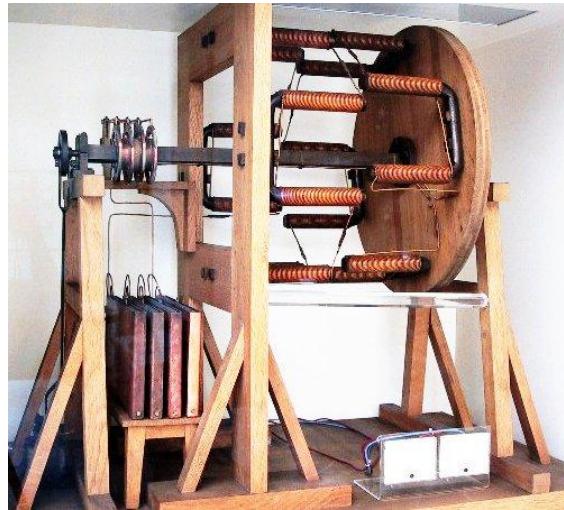


Abb. 6: Nachbau des Jakobi-Motors im Deutschen Museum in München

In den Folgejahren wurden zahlreiche Prototypen von Elektromotoren konstruiert. Das erste Patent für einen Elektromotor (US Patent No. 132) erhielt am 25.02.1837 der Amerikaner [Thomas Davenport](#) (1802-1854), ein Hufschmied – nach vier Jahren intensiver Experimente (Abb. 7). Der Motor erreichte bis zu 600 U/min.

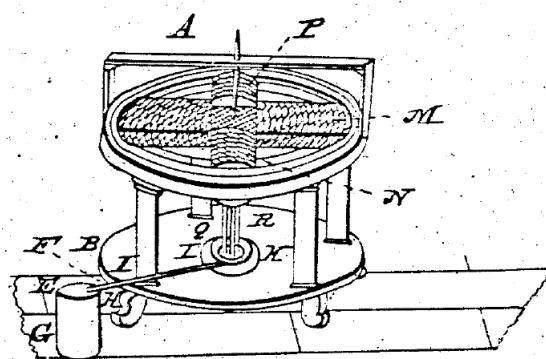


Abb. 7: Elektromotor von Thomas Davenport (US Patent No. 132)

Auch Davenport hatte wegen der hohen Batteriekosten mit seinem Motor keinen wirtschaftlichen Erfolg. Durchsetzen konnten sich Elektromotoren erst nach der Entdeckung des „dynamoelektrischen Prinzips“ durch [Werner von Siemens](#) (1816-1892) im Jahr 1867 [6]: Erst mit seiner „Dynamomaschine“ ließen sich kostengünstig starke Ströme erzeugen und damit leistungsstarke Elektromotoren betreiben.

Funktionsmodelle

Repulsionsmotor

Um einen maximalen Wirkungsgrad zu erzielen, wird der Elektromagnet in einem Elektromotor nicht nur ein- und ausgeschaltet, sondern – wie beim Jakobi-Motor – umgepolt. Dazu wird der Stator aus den Polschuhen von Permanentmagneten gebildet und der Rotor aus Elektromagneten, die über Bürsten mit Strom versorgt werden.

Durch die Drehbewegung wechselt die Stromrichtung an den Bürsten. Es kommt daher zwischen Rotor und Stator zu einem ständigen Anziehen und Abstoßen der Magnetfelder. Drehmoment und Umdrehungsgeschwindigkeit des Elektromotors lassen sich über ein Verdrehen der Bürsten – und damit die Veränderung des Umschaltzeitpunkts – beeinflussen. Ein solcher Motor wird auch [Repulsionsmotor](#) genannt.

Im Clubheft 1/1973 [7] erschien unter der Rubrik „Aktuelles zum Nachbauen“ die erste Anleitung für einen fischertechnik-Repulsionsmotor. Dabei wurden die Elektromagneten am Stator und die Permanentmagneten am Rotor befestigt – daher kann man auf die „Bürsten“ verzichten (der Rotor muss nicht mit Strom versorgt werden) und stattdessen mit einem Taster und einem Relais das Magnetfeld der Elektromagneten umpolen. Das Modell benötigt vier Dauermagnete, zwei Elektromagnete, einen Taster und ein Relais (Abb. 8).



Abb. 8: Elektromotor aus Clubheft 1/1973 [7]

Die Dauermagnete mit jeweils gleicher magnetischer Ausrichtung sorgen für acht wechselnde Magnetfelder rund um den Rotor. Die Unterbrecher müssen daher achtmal je Umdrehung die Stromflussrichtung ändern.

Ein Repulsionsmotor mit einer Leerlaufdrehzahl von ca. 600 U/min lässt sich mit den stärkeren und leichteren Permanentmagneten (Neodym-Magnet, 139251, aus dem fischertechnik-Baukasten Technical Revolutions) konstruieren (Abb. 9).

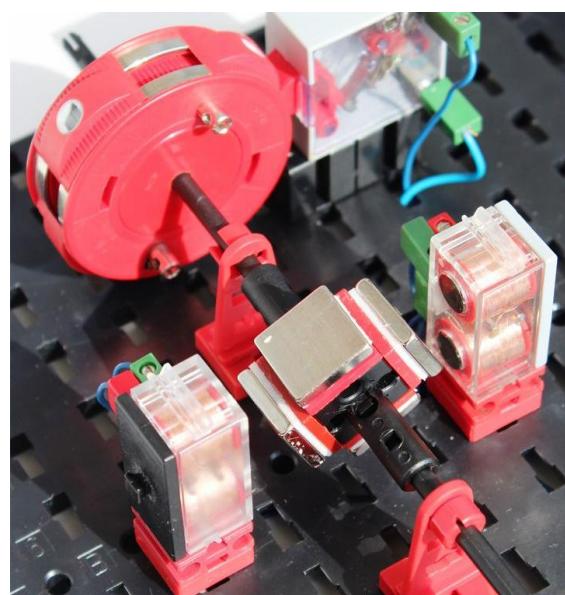


Abb. 9: Elektromotor mit Neodym-Magneten

Wechselstrommotor

Anstatt die Stromrichtung durch „Bürsten“ (im fischertechnik-Modell: Schleifring mit Unterbrecherkontakte) umzukehren, kann man den Motor auch gleich mit Wechselstrom betreiben: Schließt man die Elektromotoren an den Wechselstrom-Anschluss des guten alten grauen 6 V-fischertechnik-Netzteils, erhält man eine stabile 50-Hz-Frequenz aus dem Stromnetz.

Die Geschwindigkeit des Motors ist dabei durch die Netzfrequenz festgelegt. Er muss daher mit ein wenig Fingerspitzengefühl so angeworfen werden, dass die Richtungswechsel des Stromflusses im richtigen Moment erfolgen. Die Verwendung einer

festen Umschaltfrequenz ist insbesondere dann hilfreich, wenn man eine feste Geschwindigkeit möglichst exakt einhalten möchte – z. B. für den Antrieb einer elektrischen Uhr. Ausgehend von der Netzfrequenz kann die Rotationsgeschwindigkeit über die Zahl der verwendeten (Permanent-) Magneten, eine elektronische Frequenzanpassung oder eine nachgelagerte Übersetzung eingestellt werden.

In der ft:Community wurden mehrere 50-Hz-Motoren vorgestellt, unter anderem von Peter Damen und Thomas Püttmann (Abb. 10, 11).

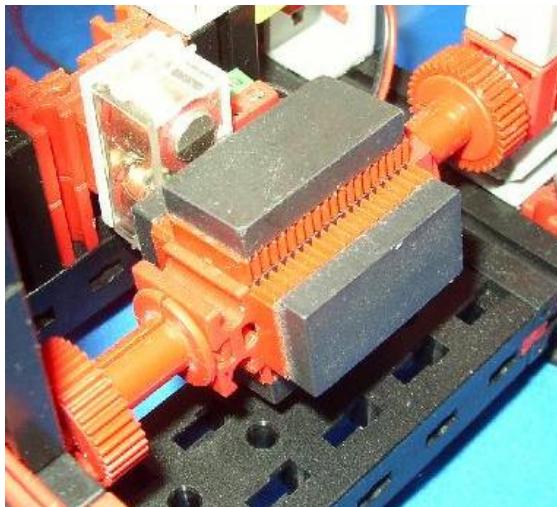


Abb. 10: 50-Hz-Motor von peterholland (2003)

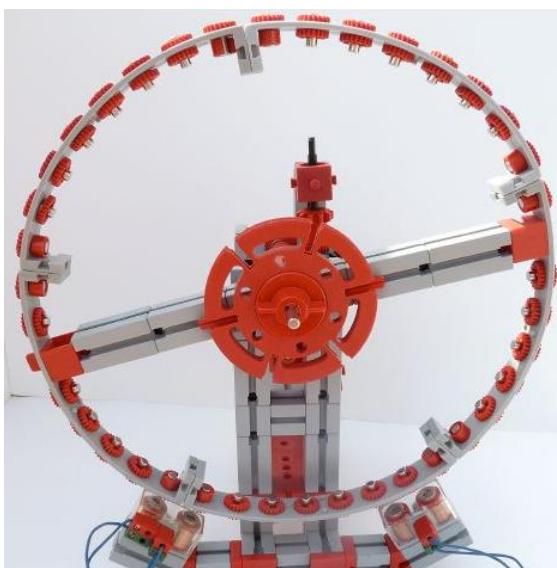


Abb. 11: 50-Hz-Motor von geometter (2011), mit 38 Neodym-Magneten und 2,5 U/min

Von Stefan Falk stammt ein 12,5-Hz-Motor, bei dem die Wechselstromfrequenz von Silberlingen und E-Tecs erzeugt wird (Abb. 12). Alle drei Motoren dienten einer mechanischen Uhr als elektrischer Antrieb.

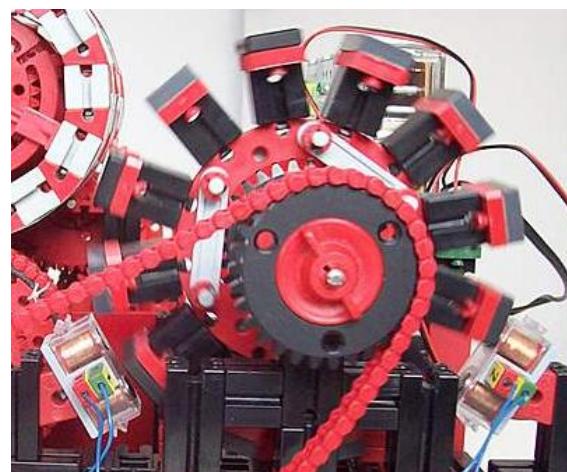


Abb. 12: 12,5-Hz-Motor von Steffalk (2010), für den Uhrenantrieb justiert auf 25 U/min

Quellen

- [1] Jörg Richter: [André-Marie Ampère und der Elektromagnetismus](#). Sendereihe ‚Meilensteine der Naturwissenschaft und Technik‘, ARD, 2006.
- [2] Werner Kiefer: [Michael Faraday: Strom aus Magneten](#). Sendereihe ‚Meilensteine der Naturwissenschaft und Technik‘, ARD, 2004.
- [3] Wikipedia: [Electric Motor](#).
- [4] Martin Doppelbauer: [Die Erfundung des Elektromotors \(1800-1854\)](#). Elektrotechnisches Institut am Karlsruher Institute of Technology (KIT).
- [5] [Der Jakobi-Motor](#). Elektrotechnisches Institut am KIT, 2012 ([Video](#)).
- [6] Susanne Päch, Herbert W. Franke: [Die dynamo-elektrische Maschine von Werner von Siemens](#). Sendereihe ‚Meilensteine der Naturwissenschaft und Technik‘, ARD, 1990.
- [7] [Aktuelles zum Nachbauen](#), fischertechnik Club-Heft 1/1973, S. 20-22.

Antriebstechnik

Kenndaten der ft-Motoren

René Trapp

Bei der Planung eines motorbetriebenen Funktions- oder Anschauungsmodells fällt es mangels entsprechender Motordaten oft schwer, den „richtigen“ Motor auszuwählen. Mit Kenntnis der Motordaten ließe sich vorab eine Aussage über die Eignung eines Motors für den geplanten Einsatz treffen, den voraussichtlichen Arbeitspunkt könnte man an Hand eines Rechenmodells ermitteln. Die dazu notwendigen Kenndaten eines Motors selbst zu bestimmen, ist gar nicht so schwer wie es vielleicht auf den ersten Blick scheint. Professionelle Motorenprüfstände sind dafür nicht erforderlich – einfache Aufbauten reichen aus. Der Umgang mit dem Rechenmodell ist auch kein Hexenwerk.

Jeder ft:pedia-Leser kennt (hoffentlich) das Prinzip des elektrischen Gleichstrommotors: Spannungsversorgung anschließen, einschalten, Strom fließt und schon dreht sich die Motorwelle. Je nach Last oder Änderungen der Versorgungsspannung variieren Stromaufnahme und Drehzahl.

Diese Verhältnisse, auch als Arbeitspunkte bezeichnet, werden üblicherweise vom Motorenhersteller mittels Kenndaten angegeben.



Abb. 1: Einige Motoren

Abb. 1 zeigt einige Motoren, die sich beim Autor angesammelt haben. Datenblätter mit den vollständigen Angaben gibt es zu diesen Motoren nicht. Leider trifft dies auch für die vergleichsweise teuren ft-

Motoren zu. Ohne Motordaten ist unklar, wie der Motor optimal eingesetzt werden kann. Häufig wird ein Motor daher am falschen Arbeitspunkt betrieben. Insbesondere bei Akkubetrieb aber sollte ein Motor die verfügbare Energie möglichst gut nutzen, also mit höchstmöglichen Wirkungsgrad arbeiten.

Fehlende Motordaten lassen sich über eine Modellbildung berechnen. Dazu ist die Kenntnis der physikalischen Zusammenhänge und der zu ermittelnden Parameter nötig.

Versuch 1: Der Motor als Generator

Beginnen wir einfach einmal mit einer unerwarteten Fragestellung: Was passiert bei einem Motor, an dessen Klemmen ein Digital Volt Meter (DVM) angeschlossen ist, wenn die Welle gedreht wird?

Um diese Frage zu beantworten ist bereits ein einfacher Versuchsaufbau geeignet. Abb. 2 zeigt ein Beispiel: Ein Hilfsantrieb (Mini-Motor 32293, rechts im Bild) dreht die Welle des Prüflings (Mini-Motor links im Bild) möglichst gleichmäßig und mit verschiedenen Drehzahlen.

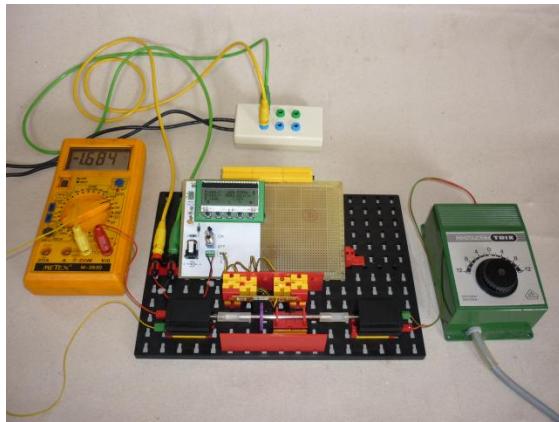


Abb. 2: Aufbau Generatorversuch

Die Motoren sind jeweils mit einem Stückchen Silikonschlauch an eine 4 mm-Welle angekoppelt. Die 4 mm-Welle trägt lediglich die Unterbrecherscheibe für die Drehzahlmessung und ist möglichst reibungsarm gelagert. Außer dem mechanischen Aufbau wird auch noch eine veränderbare Spannungsversorgung, hier der Modellbahntrofa, für den Hilfsmotor (U_{Mot}), ein Spannungsmessgerät mit hochohmigem Eingang (DVM) für U_{EMK} und ein Drehzahlmessgerät für die Drehzahl n (montiert auf der Platine im Hintergrund) benötigt. Der Schaltplan dazu ist nicht sehr umfangreich. Zur vollständigen Übersicht wurden in Abb. 3 auch noch die Drehzahlmessung und der Drehsinn bei positiver Spannung aufgenommen.

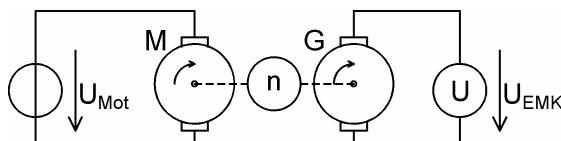


Abb. 3: Schaltplan Generatorversuch

M symbolisiert den antreibenden Motor, **G** symbolisiert den als Generator arbeitenden Prüfling. Die gestrichelte Wirklinie symbolisiert die mechanische Verbindung der Wellen mit der Drehzahlmesseinrichtung. EMK steht hier für Elektro-Motorische Kraft, englisch „EMF“, bei Motorbetrieb „back-EMF“.

Wird die gemeinsame Welle nicht bewegt, dann kommt an den Anschlussklemmen des Prüflings die Spannung $U_{EMK} = 0 \text{ V}$

heraus. Wird die Welle durch den Hilfsantrieb gedreht, dann kommt eine von Null verschiedene Spannung heraus. Dreht die Welle schneller, dann ist der Betrag der Spannung höher. Dreht die Welle anders herum, dann ändert sich das Vorzeichen der Spannung.

Eine negative Drehzahl bedeutet in dieser Anwendung einfach die andere Drehrichtung der Welle; also der mathematische Ausdruck für „rückwärts“ (oder auch „vorwärts“, das ist eine Frage der Philosophie). Der angetriebene Motor arbeitet nun also als Generator und erzeugt eine Spannung. Wegen des extrem geringen Stromflusses durch das DVM wird der Generator als unbelastet angesehen. Das DVM zeigt daher die sogenannte Leerlaufspannung des Generators an [8].

Misst man nun die Drehzahl der Welle und ordnet die bei dieser Drehzahl erzeugte Spannung U_{EMK} zu, so kann man daraus das Verhältnis Spannung zu Drehzahl bilden. Dieses Verhältnis wird als Motorkonstante k_U bezeichnet.

$$k_U = \frac{U_{EMK}}{n} \quad (1)$$

$$U_{EMK} = n \cdot k_U \quad (2)$$

Die am Versuchsaufbau aus Abb. 2 ermittelten Werte sind in Tab. 1 zusammengefasst. Die Motorkonstante k_U ist dabei entsprechend (1) errechnet.

Das Drehzahlmessgerät kennt natürlich kein Vorzeichen, daher ist von Hand ein Minus zu ergänzen, wenn die Welle anders herum dreht. Außerdem sind die Werte für k_U nicht alle identisch. Das kommt zum einen von Messfehlern, hauptsächlich aber vom unruhigen Lauf der gekoppelten Maschinen, die einen gewissen Ablesefehler der Anzeigen nach sich ziehen. Über die Messergebnisse wird k_U gemittelt zu $k_U = 0,0008428 \text{ V/min}^{-1}$.

Drehzahl n [min ⁻¹]	Generator- spannung U _{EMK} [V]	Konstante k _U [V/min ⁻¹]
-4.350	-3,67	0,0008437
-3.000	-2,54	0,0008467
-2.200	-1,87	0,0008500
-1.000	-0,852	0,0008520
0	0	k. A.
1.000	0,843	0,0008430
1.760	1,47	0,0008352
3.900	3,26	0,0008359
4.330	3,62	0,0008360

Tab. 1

Für eine elektrische Maschine (so heißen Motoren und Generatoren in der Fachsprache [6]) ist also schon das erste Element eines Modells gefunden: eine drehzahlproportionale Spannungsquelle (Abb. 4).

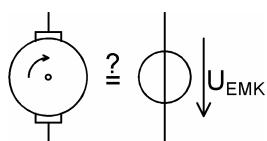


Abb. 4: Ein vorläufiges Ersatzschaltbild

Die erzeugte Spannung U_{EMK} wird durch die Induktionswirkung hervorgerufen [1]. Nun tritt aber die Induktionswirkung grundsätzlich immer auf wenn sich die Motorwelle dreht [7]. Es ist also völlig unerheblich ob der Motor als solcher oder, wie in diesem Versuch, als Generator betrieben wird.

Das Verhältnis k_U ist konstant, der Zusammenhang von Drehzahl n und induzierter Spannung U_{EMK} ist linear.

Versuch 2: der Motor treibt (fast) nichts an

Die nächste Fragestellung ist ebenfalls einfach: Was passiert auf der elektrischen Seite bei einem leer laufenden Motor?

Um diese Frage zu beantworten, wird der vorhandene Versuchsaufbau abgeändert. An den zu prüfenden Motor ist diesmal nur noch die Drehzahlmesseinrichtung angekoppelt – deswegen das „fast“ in der Überschrift. Der Hilfsantrieb (rechter Motor) ist abgekoppelt (Abb. 5).

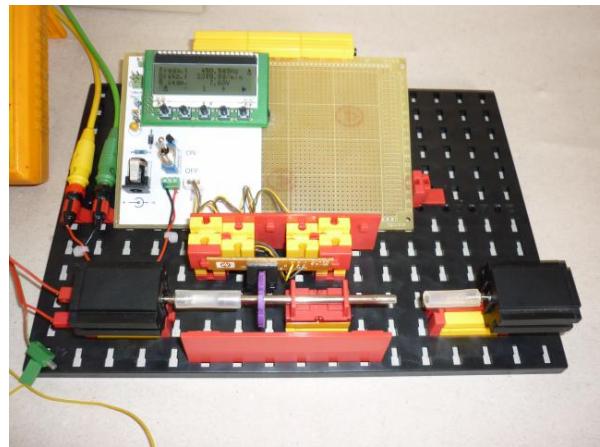


Abb. 5: Aufbau Leerlaufversuch 1

Außer dem mechanischen Aufbau wird auch noch eine veränderbare Spannungsversorgung für den Motor, ein Spannungsmessergerät, ein Strommessgerät und ein Drehzahlmessgerät benötigt. In Abb. 6 ist der Schaltplan angegeben.

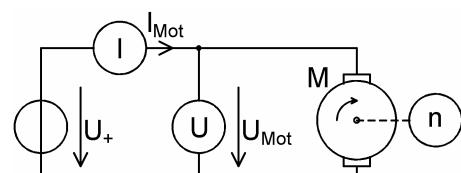


Abb. 6: Schaltplan Leerlaufversuch 1

M symbolisiert den antreibenden Prüfling. Damit das später ermittelte Motormodell (nicht das ft-Modell) auch richtig funktioniert, muss unbedingt der Motor als Motor verwendet werden, der vorher als Generator betrieben wurde.

Sobald nun der Motor mit der Spannungsquelle verbunden wird, beschleunigt er die Welle auf eine bestimmte Drehzahl. Die Drehzahl steigt mit zunehmender Klemmenspannung, der Drehsinn der Motorwelle ändert sich wenn die Spannung am Motor umgepolt wird. Unterhalb einer

bestimmten Mindestspannung dreht der Motor nicht.

Die am Versuchsaufbau ermittelten Werte sind in Tab. 2 zusammengefasst.

Drehzahl n [min $^{-1}$]	Motor- spannung U_{Mot} [V]	Motor- strom I_{Mot} [A]
-4.360	-5,26	-0,1296
-3.300	-3,97	-0,0996
-2.200	-2,81	-0,0796
0	0	0
2.000	2,73	0,0819
2.800	3,54	0,0977
4.000	4,85	0,1246

Tab. 2

Bekanntlich besteht der Rotor des Motors aus verschiedenen Wicklungsteilen. Der verwendete (Kupfer-)Draht hat einen bestimmten Durchmesser und eine bestimmte Länge. Daher ergibt sich ein bestimmter Widerstand solch einer Wicklung [2]. Völlig naiv kann man nun im Ersatzschaltbild einfach diesen Widerstand in Reihe zur drehzahlabhängigen Spannungsquelle anordnen. Er heißt hier, nicht ohne Hintersinn, R_{Dyn} .

Das im vorhergehenden Versuch ermittelte Ersatzschaltbild der idealen elektrischen Maschine (Abb. 4) wird also um einen Serienwiderstand R_{Dyn} ergänzt zu Abb. 7.

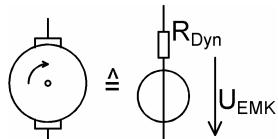


Abb. 7: Ersatzschaltbild

Wie schon ausgeführt, tritt die Induktionswirkung immer auf wenn die Welle sich dreht. Daher entsteht die drehzahlabhängige Spannung U_{EMK} auch, wenn der Motor als solcher arbeitet. Die ermittelten Messwerte aus der vorstehenden Tabelle

werden nun etwas ergänzt, nämlich um die mittels Gleichung (2) aus n und k_U errechnete Spannung U_{EMK} . Die Spannungsdifferenz zwischen der Spannung an den Motorklemmen U_{Mot} und der induzierten Spannung U_{EMK} treibt nun den Strom I_{Mot} durch den dynamischen Innenwiderstand R_{Dyn} . Und gemäß dem bekannten Ohm'schen Gesetz [13], benannt zu Ehren seines Entdeckers Georg Simon Ohm [10],

$$R = \frac{U}{I} \quad (3)$$

ist dieser R_{Dyn} damit bestimbar:

$$R_{Dyn} = \frac{U_{Mot} - U_{EMK}}{I_{Mot}} \quad (4)$$

$$R_{Dyn} = \frac{U_{Mot} - n \cdot k_U}{I_{Mot}} \quad (5)$$

Tab. 3 enthält nochmals die bereits gemessenen Werte vom Versuchsaufbau. Zusätzlich ist R_{Dyn} gemäß (5) errechnet. k_U ist bereits im vorhergehenden Versuch zu $k_U = 0,0008428 \text{ V/min}^{-1}$ bestimmt worden.

Drehzahl n [min $^{-1}$]	Motor- spannung U_{Mot} [V]	Motor- strom I_{Mot} [A]	Innen- wider- stand R_{Dyn} [Ω]
-4.360	-5,26	-0,1296	12,2
-3.300	-3,97	-0,0996	11,9
-2.200	-2,81	-0,0796	12,0
0	0	0	k.A.
2.000	2,73	0,0819	12,8
2.800	3,54	0,0977	12,1
4.000	4,85	0,1246	11,9

Tab. 3

Auch hier sind die errechneten Werte untereinander nicht identisch. Wiederum liegt dies am ungleichmäßigen Motorlauf und den Schwankungen der Anzeigen. Über die Messergebnisse wird R_{Dyn} zu $12,15 \Omega$ gemittelt.

Wer mit seinem Widerstandsmessgerät zum Vergleich den Widerstand eines stehenden(!) Motors messen möchte, sollte das bei verschiedenen Teildrehungen der Welle tun. Je nachdem wo der Stromwender gerade steht, ergeben sich sehr unterschiedliche Werte. Der kleinste so ermittelte Wert kann immerhin verwendet werden um die zu erwartende Einschaltstromspitze (oder den Blockadestrom) zu errechnen. Das ist wichtig für die Auswahl des Motortreibers.

Selbstverständlich taugen diese Werte aber nicht für das Motormodell, denn tatsächlich entsteht ein mittlerer Wert durch die Drehung der Welle, die Kommutierung und noch weitere dynamische Effekte. Daher wird der Widerstand als R_{Dyn} bezeichnet.

Versuch 3: der Motor bekommt Arbeit

Nun sind also schon zwei Teile eines brauchbaren Motormodells aufgefunden. Das ist aber noch nicht alles. Beim vorhergehenden Versuch zeigte sich eine (halbwegs) konstante Stromaufnahme des Motors, obwohl er mit unterschiedlichen Drehzahlen bei unterschiedlichen Betriebsspannungen arbeitete. Allerdings war die Welle immer nur mit dem Drehzahlgeber angekoppelt, mithin eine als konstant angenommene Last.

Machen wir also ein nächstes Experiment. Die Fragestellung lautet diesmal: Was passiert auf der elektrischen Seite eines Motors, der verschiedene, aber bekannte, Drehmomente aufbringen muss?

Mit unbekannten Drehmomenten lernt man zwar auch etwas, aber es lässt sich kein Parameter für das Motormodell errechnen. Eine Küchenwaage ist hier also sehr hilfreich.

Bekannte Drehmomente lassen sich auch ohne die Zuhilfenahme teurer, industriell gefertigter Drehmomentsensoren erzeugen.

Der Motor treibt hierfür eine möglichst reibungsarm gelagerte Seilwinde an, an die unterschiedliche Massestücke als Last angehängt werden. Je länger der mögliche Hubweg umso besser ist das für praktische Versuche. Abb. 8 zeigt die verwendete Windenmechanik: Ein Endschalter oben, der den Hubmotor am oberen Ende abschaltet, hilft Schäden an der Mechanik zu verhindern. Ein zweiter, hier nicht sichtbarer Endschalter am unteren Ende verhindert Seilsalat.



Abb. 8: Aufbau Seilwinde (von oben)

Das an der Motorwelle wirksame Drehmoment M lässt sich aus der am Seil angehängten Masse m , dem Seiltrommelradius r und der Erdgravitation g errechnen. Für g reicht der ungefähre Wert $9,81 \text{ m/s}^2$ völlig aus [3]. Wer es trotzdem präziser haben möchte schaut mal bei [4], allerdings rechtfertigen die auftretenden Messfehler diese Präzision keineswegs.

$$M_{Mot} = r \cdot m \cdot g \quad (6)$$

Außer dem mechanischen Aufbau wird auch noch eine veränderbare Spannungsversorgung für den Motor und ein Strommessgerät benötigt.

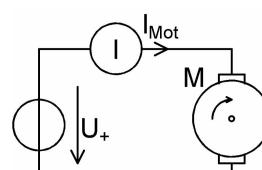


Abb. 9: Schaltplan Seilwinde

Auch hier wird wieder exakt der Motor genommen, der im ersten Experiment als Generator arbeiten und im vorherigen Experiment fast leer laufen durfte. Wegen der Ankopplung an die Welle ist allerdings eine Getriebebeanordnung eingefügt.

Ist kein Massestück angehängt, läuft der Motor während des Hubes (fast) leer. Der Motor nimmt dabei einen bestimmten Strom auf. Wird ein Massestück angehängt, nimmt der Motor einen höheren Strom auf. Wird ein anderes Massestück angehängt, nimmt der Motor einen anderen Strom auf.

Misst man nun die Masse m des jeweiligen Massestückes (zum Beispiel mit einer guten Küchenwaage), errechnet daraus mittels (6) das vom Motor aufzubringende Drehmoment M und ordnet diesem den aufgenommenen Strom I zu, so ergibt sich Tab. 4. Der Seiltrommelradius r beträgt 2 mm, g ist mit $9,81 \text{ m/s}^2$ eingesetzt.

Masse m [kg]	Dreh-moment M [Nm]	Motor-strom I_{Mot} [A]
0	0	0,071
0,109	0,0021	0,098
0,051	0,0010	0,082
0,238	0,0047	0,129

Tab. 4

Nun ist dieser Versuchsaufbau meilenweit von einem idealen Motor entfernt, der dynamische Widerstand R_{Dyn} macht sich bemerkbar und die diversen Lagerreibungen stellen eine Art Grundlast dar. Daher ist bei jedem Durchgang die Leerlaufstromaufnahme (mit der angekoppelten Seilwinde!) vom jeweiligen Laststrom abzuziehen.

$$I = I_{\text{Mot}} - I_{\text{Leer,Winde}} \quad (7)$$

So korrigiert lassen sich die Stromwerte zum jeweiligen vorgegebenen Drehmo-

ment ins Verhältnis setzen, zum Beispiel Drehmoment/Strom.

$$k_I = \frac{M}{I} \quad (8)$$

$$M = I \cdot k_I \quad (9)$$

k_I ist der Proportionalitätsfaktor, der Drehmoment und Strom verknüpft.

Masse m [kg]	Dreh-moment M [Nm]	Motor-strom I_{Mot} [A]	Konstante k_I [Nm/A]
0	0	0,071	k.A.
0,109	0,0021	0,098	0,07921
0,051	0,0010	0,082	0,09097
0,238	0,0047	0,129	0,08051

Tab. 5

Dieser Versuch ist leider mit mehr Messfehlern und Toleranzen behaftet als die beiden vorherigen Versuche. Schon ein geringer Messfehler reicht aus, um das Ergebnis deutlich abweichen zu lassen. Über die Messergebnisse wird k_I zu 0,08356 Nm/A gemittelt. Aber dieser Wert gilt hierbei mit dem Vorsatzgetriebe (siehe Versuchsaufbau Abb. 8). Ohne Vorsatzgetriebe, also für den Motor alleine, liegt k_I bei etwa 0,00805 Nm/A.

Wer allerdings erwartet, dass k_I nun um das Übersetzungsverhältnis des Getriebes (ca. 64,8:1) größer ist als k_I des Motors alleine, der sei noch auf die mechanischen Getriebeverluste insbesondere bei Schnellkettengetrieben aufmerksam gemacht. Im Abschnitt über den Zusammenhang von k_U mit k_I und dem Abschnitt über die Getriebemotoren wird hierauf nochmal näher eingegangen. Außerdem wurde in diesem Versuch mit Getriebe gemessen, in den anderen Versuchen wurde ohne Getriebe gemessen – dies ist ein erheblicher Unterschied!

Trotz der Probleme einer korrekten Messung zur Berechnung von k_I wird das

erzeugte Drehmoment durch die Lorentzkraft hervorgerufen, siehe [5]. So wie die Induktion immer auftritt sobald die Welle sich dreht, tritt die Lorentzkraft, und somit ein Drehmoment, immer auf, wenn Strom durch die elektrische Maschine fließt. Es ist an dieser Stelle fast unerheblich ob Motorbetrieb oder Generatorbetrieb vorliegt. Fast: Bei einem Generator ist die Stromrichtung genau andersherum als bei Motorbetrieb mit gleicher Drehrichtung der Welle.

Die Stromrichtung ist hierbei eine willkürliche Festlegung. Bei einem Verbraucher (Widerstand oder auch Motor) zeigen der Spannungszählpfeil und der Stromzählpfeil in die gleiche Richtung.

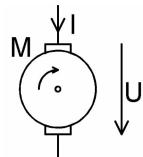


Abb. 10: Zählpfeile bei Motorbetrieb

Bei einem Erzeuger (Batterie oder auch Generator) zeigen die beiden Zählpfeile in entgegen gesetzte Richtungen (vgl. Abb. 10 und 11).

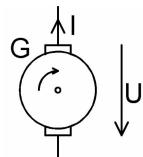


Abb. 11: Zählpfeile bei Generatorbetrieb

Multipliziert man die beiden Größen Strom und Spannung mit Vorzeichen, dann geht in einen Verbraucher eine positive Leistung, bei einem Generator eine negative Leistung hinein. Im letzteren Fall kommt faktisch eine positive Leistung heraus. Der Kniff mit dem Vorzeichen vereinfacht die Mathematik, wie bei den Drehrichtungen.

Jedenfalls ist jetzt auch der Zusammenhang von Strom und Drehmoment hergestellt. Das bisher zusammengestellte elektrische Ersatzschaltbild aus Abb. 7 behält weiter seine Gültigkeit.

Versuch 4: der Motor läuft wirklich leer

Wieso denn nochmal? Nun ja, beim letzten Mal (Versuch 2) war der Drehzahlnehmer angekoppelt. Der braucht selbst ein gewisses Drehmoment, das der Motor aufbringen muss. Bislang war das nicht weiter störend oder wurde per Rechnung unschädlich gemacht. Aber für einen echten Motor möchte man ja noch genau wissen, welches Drehmoment seine Lager und sein Kommutator erzeugen. Die sind nämlich für den Strom im Leerlauf verantwortlich. Diesmal ist der Versuchsaufbau beschämend einfach:

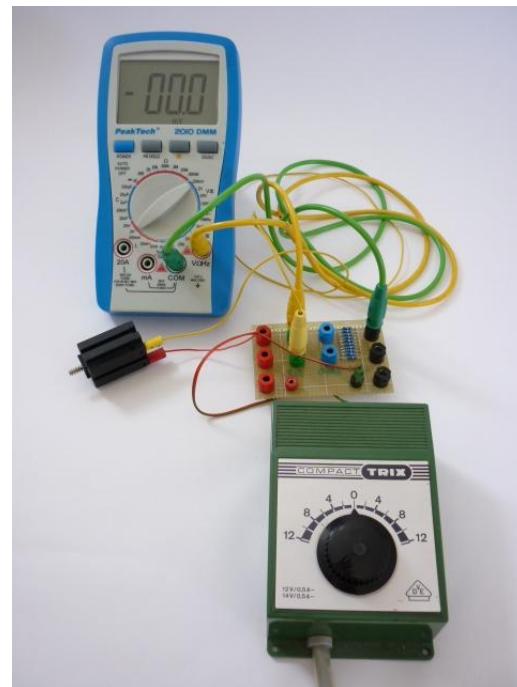


Abb. 12: Aufbau Leerlaufversuch 2

Außer dem mechanischen Aufbau – na ja, der Motor liegt einfach auf dem Tisch – wird auch noch eine veränderbare Spannungsversorgung für den Motor und ein Strommessgerät benötigt.

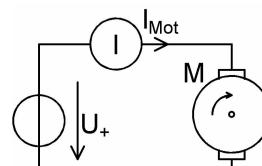


Abb. 13: Schaltplan Leerlaufversuch 2

Gemessen wird die Stromaufnahme des Motors. Mit der bereits vorliegenden Konstanten k_I und (9) ist das auftretende Verlustdrehmoment im Motor nun schnell ermittelt. Zur Vereinfachung seien diese Verluste alle dem Lager zugeschrieben.

$$M_{Lager} = I_{Leer} \cdot k_I \quad (10)$$

Die am Versuchsaufbau ermittelten und errechneten Werte sind in Tab. 6 zusammengefasst. Das Verlustmoment M_{Lager} ist mittels (10) errechnet.

Motorstrom I_{Mot} [A]	Verlustmoment M_{Lager} [Nm]
-0,0656	-0,000528
-0,0601	-0,000484
-0,0550	-0,000443
0	k.A.
0,0500	0,000402
0,0581	0,000468
0,0606	0,000488

Tab. 6

Über die Messergebnisse wird M_{Lager} zu 0,000469 Nm gemittelt.

Eine Anwendung für k_I – der Motor als Drehmomentsensor

Aus (9) ergibt sich eine interessante Anwendungsmöglichkeit für den Motor (oder auch einen Generator). Sie gilt ja nicht nur im Leerlauf sondern immer wenn Strom durch die elektrische Maschine fließt. Die dabei auftretenden Verluste im Motor müssen natürlich abgezogen werden um das nutzbare Drehmoment zu errechnen.

$$M_{Mot} = I \cdot k_I - M_{Lager} \quad (11)$$

Aus einer Strommessung kann mit Kenntnis von k_I und M_{Lager} das gerade vom Motor abgegebene Drehmoment unmittel-

bar errechnet werden (11). Wird der Motor als Generator betrieben, so stellt der Strom aus dem Generator das unmittelbare Maß für das an der Generatorwelle wirksame Drehmoment dar. In diesem Fall ist natürlich das Verlustmoment noch aufzuhaddieren.

$$M_{Gen} = I \cdot k_I + M_{Lager} \quad (12)$$

So kann das Drehmoment eines Motors oder auch eines Generators ohne teure Drehmomentsensoren errechnet werden. Im Falle eines Motors könnten damit die mechanischen Verluste in einem vorgesetzten Getriebe errechnet werden.

Die Sache mit dem Wirkungsgrad

Generatoren und Motoren sind Energie- bzw. Leistungswandler. Sie wandeln mechanische Energie in elektrische Energie und umgekehrt. Überall wo eine Energieform durch reale (nichtideale) Maschinen in eine andere umgewandelt wird, treten Verluste auf. In diesem Zusammenhang ist die Größe Wirkungsgrad definiert als Verhältnis von abgegebener Leistung zu eingespeister Leistung [15]. Im Falle des Generatorbetriebs ist der Wirkungsgrad selbstverständlich durch das Verhältnis von elektrischer Leistung (abgegeben) zu mechanischer Leistung (aufgenommen) definiert:

$$\eta_{Gen} = \frac{P_{el}}{P_{mech}} \quad (13)$$

Im Falle des Motors ist es also das Verhältnis von mechanischer Leistung zu elektrischer Leistung.

$$\eta_{Mot} = \frac{P_{mech}}{P_{el}} \quad (14)$$

Der Wirkungsgrad kann niemals größer als 1 sein! Das ist physikalisch unmöglich. Errechnet man hier einen Wert größer als 1, dann sollte man seine Rechnung und eventuell auch seine Überlegungen sehr genau nachprüfen.

Das kleine Geheimnis von k_U und k_I

In der elektrischen Maschine passieren immer zwei Dinge gleichzeitig. Einmal ist es die Induktionswirkung, die die Spannung U_{EMK} bestimmt und für deren komplexes Wirkungsgesetz k_U die vereinfachte Kurzfassung darstellt. Zum anderen ist es die Lorentzkraft, die durch den mechanischen Aufbau in ein Drehmoment umgewandelt wird. k_I dient hierfür als Kurzfassung. Beide Effekte werden faktisch durch das gleiche Magnetfeld im gleichen Motor hervorgerufen. Also erscheint es nur logisch, dass k_U und k_I , als zwei verschiedene Aspekte einer gemeinsamen Grundlage, irgendwie miteinander zusammenhängen.

Bildet man das Verhältnis k_U/k_I , so ergeben sich im Zähler und im Nenner formal Leistungseinheiten. Zur Erinnerung:

Elektrische Leistung [11]:

$$P_{el} = U \cdot I \quad [W = V \cdot A] \quad (15)$$

Mechanische Leistung [12]:

$$P_{mech} = 2\pi \cdot M \cdot n \quad [W = Nm \cdot s^{-1}] \quad (16)$$

ausgedrückt in „Newtonmeter pro Sekunde“ (nicht „pro Minute“!). Für die mechanische Leistung ist vorher immer die Drehzahl in die richtige Einheit s^{-1} umzurechnen. Oder aber, für n in den gewohnten „Umdrehungen pro Minute“, gilt die Größengleichung

$$P_{mech} = \frac{2\pi \cdot M \cdot n}{60} \quad [W = Nm \cdot s^{-1}] \quad (17)$$

Gemeinerweise ist (16) ohne die explizite Angabe der Einheiten dieser kleine Unterschied bei der korrekten Drehzahleinheit nicht direkt anzusehen und man kann ganz böse Überraschungen erleben, wenn die Einheit nicht stimmt!

(16) kann man auch noch so notieren:

$$\frac{P_{mech}}{2\pi} = M \cdot n \quad [W = Nm \cdot s^{-1}] \quad (18)$$

Dies ist wird gleich noch nützlich sein.

Doch zurück zu k_U/k_I : mit der gängigen Denkweise für n in „Umdrehungen pro Minute“ eingesetzt und mathematisch umgeformt, führt das schließlich auf einen konstanten Faktor:

$$\frac{\frac{U_{EMK}}{n}}{\frac{60}{M}} = \frac{U_{EMK} \cdot I}{60 \cdot M \cdot n} = \frac{2\pi}{60} \quad (19)$$

Da es sich hier um k_U und k_I einer idealen elektrischen Maschine handelt, sind die elektrische Leistung im Zähler des dritten Terms und die mechanische Leistung im Nenner des dritten Terms exakt gleich. Es gilt daher:

$$\frac{U_{EMK} \cdot I}{2\pi \cdot M \cdot n} = 1 \quad (20)$$

Beziehungsweise

$$\frac{U_{EMK} \cdot I}{M \cdot n} = 2\pi \quad (21)$$

k_U wurde bereits in Versuch 1 bestimmt. k_I kann nun ohne Versuch 3 wegen der konstanten Verknüpfung unmittelbar aus k_U errechnet werden. (19) umgeformt ergibt

$$k_I = \frac{60}{2\pi} k_U \quad (22)$$

$$k_U = \frac{2\pi}{60} k_I \quad (23)$$

In letzter Konsequenz bedeutet dies, dass nur drei der vier Versuche erforderlich sind. Sobald entweder k_U oder k_I bekannt ist, ist ihr Komplementär mittels (22) oder (23) ebenfalls gefunden. Für Getriebemotoren ist hierbei zu beachten, dass der meist unbekannte Getriebewirkungsgrad

die korrekte Ermittlung von k_I verhindert; damit wird dann auch k_U falsch berechnet.

Getriebemotoren

Nun ist es so, dass der Versuch 1 (Generatortversuch mit Hilfsantrieb) bei hohen Übersetzungen oder mit Schneckenge trieben nicht durchgeführt werden kann. Es muss also eine Möglichkeit gefunden werden, die nötigen Parameter des Motor modells auch für Getriebemotoren auszumessen.

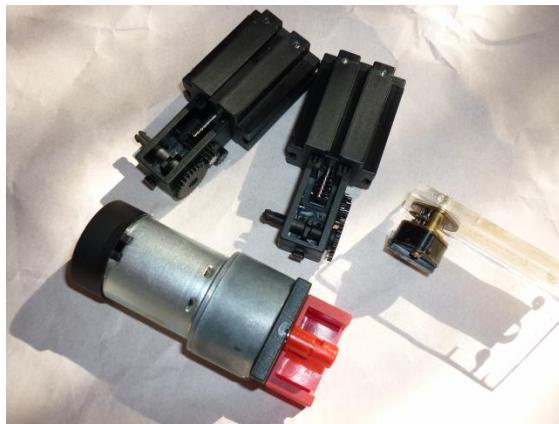


Abb. 14: Einige Getriebemotoren

Abb. 14 zeigt einige Getriebemotoren. Motoren mit angeflanschtem Getriebe werden prinzipiell nicht anders berechnet als Motoren ohne Getriebe. Das Getriebe wird dabei vereinfacht als integraler Bestandteil des Motors betrachtet. Das Motormodell funktioniert weiterhin und liefert in diesem Fall die Werte für die Einheit Motor + Getriebe.

Es gibt allerdings, wie bereits erwähnt, Einschränkungen bei der Ermittlung der Parameter für das Motormodell. Diese Einschränkungen muss man kennen und berücksichtigen.

Was ändert sich nun durch ein Getriebe? Bei jedem Getriebe gibt es ein Übersetzungsverhältnis. Dieses gibt an wie oft sich die Antriebswelle drehen muss, um der Abtriebswelle eine ganze Umdrehung zu erteilen. Bei Zahnradgetrieben, Zahnrinnengetrieben und Kettengetrieben stimmt

dieses Drehzahlverhältnis exakt wegen der Verkämmung der Zähne, bei Riemen getrieben dagegen kommt Schlupf hinzu, so dass die Drehzahlverhältnisse nur noch ungefähr stimmen. Weiter gibt es in jedem Getriebe bauartabhängig Drehmoment verluste. Vor diesem Hintergrund ist der Begriff Getriebewirkungsgrad nun sicher etwas verständlicher. Und eben dieser Getriebewirkungsgrad wurde bei unseren Messungen, z. B. bei Versuch 3, implizit mit gemessen.

Kennt man diesen Getriebewirkungsgrad, dann kann er mathematisch erfasst und die Ergebnisse können korrigiert werden. Typischerweise ist er aber zunächst unbekannt. Dadurch werden die Gleichungen (19) bis (23) zwar nicht unbrauchbar, aber sie können nicht eingesetzt werden.

Wird die Messung von k_I wie bei Versuch 3 beschrieben durchgeführt, so verfälscht der Getriebewirkungsgrad das Ergebnis. Trotzdem können, auch bei Getriebemotoren, durch entsprechende Messungen die Parameter des Motormodells ermittelt werden. Im Prinzip betreibt man den zu prüfenden Motor mit Getriebe an zwei verschiedenen bekannten Lasten bzw. Drehmomenten. Dabei notiert man außer dem Drehmoment noch Drehzahl, Motor spannung und Motorstrom. Aus den Mess werten lassen sich nun die benötigten Parameter errechnen.

Zunächst liefert (11) die Grundlage für k_I . Sie ist hier für die beiden Arbeitspunkte unterschiedlich indiziert angegeben:

$$M_1 = I_1 \cdot k_I - M_{Lager} \quad (24)$$

$$M_2 = I_2 \cdot k_I - M_{Lager} \quad (25)$$

Gleichung (25) wird von (24) abgezogen und durch diesen Kniff das (noch) unbekannte Verlustmoment M_{Lager} von Motor mitsamt Getriebe eliminiert. Übrig bleibt

$$M_1 - M_2 = I_1 \cdot k_I - I_2 \cdot k_I \quad (26)$$

und ergibt nach k_I umgeformt

$$k_I = \frac{M_1 - M_2}{I_1 - I_2} \quad (27)$$

Dabei kann man Gleichung (11) bzw. ihre Entsprechungen (24) oder (25) gleich noch heranziehen, um die Lagerverluste mit dem frisch gewonnenen k_I zu errechnen. Am Beispiel von (24) sieht das dann so aus:

$$M_{Lager} = I_1 \cdot k_I - M_1 \quad (28)$$

Für den Parameter k_U wird hier (5) als Grundlage genommen. Um hilfreich zu sein, wird sie nach U_{Mot} aufgelöst und ergibt gleich wieder für die zwei Arbeitspunkte angeschrieben das Gleichungssystem

$$U_1 = I_1 \cdot R_{Dyn} + n_1 \cdot k_U \quad (29)$$

$$U_2 = I_2 \cdot R_{Dyn} + n_2 \cdot k_U \quad (30)$$

Gleichung (30) wird von (29) subtrahiert, und es ergibt sich ein Ausdruck, der nach k_U umgestellt werden kann:

$$k_U = \frac{U_1 - U_2}{n_1 - n_2} - R_{Dyn} \frac{I_1 - I_2}{n_1 - n_2} \quad (31)$$

R_{Dyn} ist an diesem Punkt allerdings immer noch unbekannt.

Zweckmäßigerweise widmet man sich nun als nächstes dem noch fehlenden Parameter R_{Dyn} . Die Herkunft von (29) wurde bereits erklärt. (30) ist hier equivalent zu benutzen, nur die dargestellten Indizes würden sich ändern. Zurück zu Gleichung (29), die hier nochmals angegeben wird:

$$U_1 = I_1 \cdot R_{Dyn} + n_1 \cdot k_U \quad (32)$$

k_U und R_{Dyn} sind unbekannt. Für k_U wurde bereits (31) als Ersatz gewonnen. Diesen Ausdruck hier anstelle k_U einzusetzen ergibt auf den ersten Blick keinen Sinn, da ja auch dort noch R_{Dyn} enthalten ist. Allerdings erhält man einen unhandlichen Ausdruck

$$U_1 = I_1 \cdot R_{Dyn} + n_1 \cdot \left(\frac{U_1 - U_2}{n_1 - n_2} - R_{Dyn} \frac{I_1 - I_2}{n_1 - n_2} \right)$$

der außer R_{Dyn} nur noch bekannte Größen (Messwerte) enthält. Dieser Ausdruck kann nach R_{Dyn} umgeformt werden und es ergibt sich

$$R_{Dyn} = \frac{U_1 - \frac{n_1}{n_1 - n_2} \cdot (U_1 - U_2)}{I_1 - \frac{n_1}{n_1 - n_2} \cdot (I_1 - I_2)} \quad (33)$$

Nachdem R_{Dyn} nun aus den Messwerten errechnet werden kann, wird (31) angewendet, um k_U zu bestimmen. Damit liegen alle nötigen Parameter des Motormodells vor.

Versuch 5: Seilwinde, aber richtig

Basierend auf der Seilwinde aus Versuch 3 werden die Modellparameter des Prüflings mit Getriebe nochmals ermittelt. Den Aufbau zeigt Abb. 15.



Abb. 15: Messaufbau mit Seilwinde

Diesmal wird die eben betrachtete Vorgehensweise mit zwei verschiedenen Dreh-

momenten verwendet. Um alle Werte gleichzeitig zu ermitteln, wird die Verschaltung des Prüflings gemäß Abb. 6 vorgenommen. Die Messwerte sind aus Tab. 7 ersichtlich.

Masse m [kg]	Motor- spannung U [V]	Motor- strom I _{Mot} [A]	Drehzahl n [min ⁻¹]
0,143	8,38	0,118	127
0,039	8,67	0,081	141

Tab. 7

Mittels (27), (28), (31) und (33) errechnen sich hierfür die Parameter inklusive dem Vorsatzgetriebe gemäß Tab. 8.

k _U	0,054209	V/min ⁻¹
k _I	0,0551	Nm/A
R _i	12,67	Ω
M _{Lager}	0,0037	Nm

Tab. 8

Man vergleiche diese Werte einmal mit den Parametern des Motors ohne Getriebe sowie dem errechneten k_I aus Versuch 3.

Ist das Getriebe abnehmbar, wie in den Beispielen mit den ft-Mini-Motoren, so kann man die Daten des Motors alleine ebenfalls ermitteln. Mit angeflanschtem Getriebe ändert sich nun die Motorkonstante k_U um den Übersetzungs faktor (gerundet 64,8 für ft-Getriebe 31078), die Motorkonstante k_I um eine Verknüpfung aus Übersetzungs faktor und Getriebewirkungsgrad, die Leerlaufstromaufnahme erhöht sich um das vom Getriebe eingeforderte zusätzliche Drehmoment, der dynamische Innenwiderstand des Motors bleibt jedoch gleich.

Es ist sicherlich interessant auf diese Art die zusätzlichen Getriebeverluste bestimmen zu können. Beim vorliegenden Prüfling, Mini-Motor mit Getriebe (32293, 31078) und ft-Abtriebsachse (31082), liegt der Wirkungsgrad des Getriebes bei ~10%.

Wie kommt man auf den Getriebewirkungsgrad?

Beim verwendeten Getriebe ist kein Schlupf vorhanden, somit ist k_U mit Getriebe um exakt das Übersetzungsverhältnis \ddot{U} größer als k_{U_Motor} des Motors alleine. k_I mit Getriebe liegt allerdings nicht um eben diesen Faktor \ddot{U} höher als k_{I_Motor} ohne Getriebe, sondern um lediglich rund ein Zehntel dieses Wertes. Zum Vergleich siehe die Messdaten im Abschnitt „Kenndaten eines ft-Minimotors“.

(19) gilt für den Motor ohne Getriebe bzw. einen Getriebewirkungsgrad von 100 %, also, mit erweitertem Index,

$$\frac{k_{U_Motor}}{k_{I_Motor}} = \frac{2\pi}{60} \Leftrightarrow 100\% \quad (34)$$

Es wird nun angesetzt

$$k_{U_Motor} = \frac{k_U}{\ddot{U}} \quad (35)$$

und

$$k_{I_Motor} = \frac{k_I}{\ddot{U} \cdot \eta_{Getriebe}} \quad (36)$$

Ersetzen der jeweiligen Ausdrücke in (34) durch (35) und (36) ergibt nach entsprechender Umformung:

$$\eta_{Getriebe} = \frac{k_I}{k_U} \cdot \frac{2\pi}{60} \quad (37)$$

Man verwechsle bitte den Getriebewirkungsgrad nicht mit dem Wirkungsgrad des Motors. Der Gesamtwirkungsgrad eines Motors mit Getriebe ist übrigens

$$\eta_{Gesamt} = \eta_{Motor} \cdot \eta_{Getriebe} \quad (38)$$

Diese ganze Betrachtung für Getriebemotoren ist ohne Einschränkungen auch auf Motoren bzw. Generatoren ohne Getriebe zu übertragen; $\eta_{Getriebe}$ ist dann formal gleich 1.

Motordaten ohne Versuche

Die bisherigen Versuche waren nicht besonders aufwendig, aber sie stellen halt doch eine gewisse Arbeitsleistung dar. Wer bereits entsprechende Datenblätter seiner Motoren hat, kann die enthaltenen Angaben zur Ermittlung der dargestellten Kenndaten nutzen. Die beschriebenen Versuche sind dann nicht notwendig.

In jedem vernünftigen Motorendatenblatt sind die Leerlaufdaten eines Motors enthalten:

- Leerlaufdrehzahl n_{Leer}
- Leerlaufstromaufnahme I_{Leer}
- Motor-Nennspannung U_{Leer}

Diese Leerlaufdaten gelten für die so genannte Nennspannung. Für diese Spannung wurde der Motor gebaut und dort erreicht er die optimalen Betriebsdaten. Der Index „Leer“ soll hier bedeuten, dass es sich um die Angaben für den Leerlauf handelt.

Formal sind das alles die Angaben, die aus Versuch 2 vertraut sind. Allerdings mit dem Unterschied, dass keine extra Verluste durch den Drehzahlgeber entstanden sind. Was jedoch zur direkten Auswertung fehlt, ist k_U .

In jedem vernünftigen Motordatenblatt sind auch wenigstens die Daten eines Lastfalles am Motor enthalten. Üblicherweise gilt auch hier wieder die Nennspannung. Meist sind aufgeführt:

- Nenn-Drehzahl n_{Nenn}
- Nenn-Stromaufnahme I_{Nenn}
- mechanische Nenn-Leistungsabgabe P_{Nenn}

Formal entspricht das den aus Versuch 3 vertrauten Werten. Aber hier fehlt zur Auswertung k_I . Mit zwei Arbeitspunkten (hier: Leerlauf und Nenn-Lastfall) kann man die fehlenden Parameter für das Motormodell

berechnen (siehe auch den Abschnitt „Getriebemotoren“).

Zuerst das Nenn-Drehmoment M_{Nenn} : dazu wird (17) nach M umgestellt. Die Drehzahl n ist dabei bequem in min^{-1} einzusetzen:

$$M_{Nenn} = \frac{60 \cdot P_{Nenn}}{2\pi \cdot n} \quad (39)$$

Nun kümmern wir uns um den Strom. Im Leerlauf nimmt der Motor einen bestimmten Strom auf, um die Drehmomentverluste in den Lagern, dem Kommutator usw. aufzubringen. Dabei wird an der Welle kein Drehmoment abgegeben. Daher ist die Differenz zwischen Nenn-Strom und Leerlaufstrom für das an der Welle erzeugte Nenn-Moment verantwortlich. Auch in Versuch 3 wurde aus genau diesem Grund die Leerlaufstromaufnahme abgezogen um k_I zu ermitteln – siehe (7).

(8) entsprechend erweitert führt auf

$$k_I = \frac{M_{Nenn}}{I_{Nenn} - I_{Leer}} \quad (40)$$

Damit ist per Gleichung (23) auch sofort k_U bekannt – sofern kein Getriebe vorhanden ist. Mittels Kenntnis von k_I und (9) lässt sich aus dem Leerlaufstrom das Lagerverlustmoment M_{Lager} errechnen.

Gleichung (5) wird ein wenig modifiziert:

$$R_{Dyn} = \frac{U_{Leer} - n_{Leer} \cdot k_U}{I_{Leer}} \quad (41)$$

und liefert sofort den dynamischen Innenwiderstand.

Das war es auch schon. Ein seriöses Datenblatt des Motors, ein wenig „know-how“ und schon hat man sich die Versuche gespart.

Im Abschnitt „Ein Beispiel zur Abrundung“ wird diese theoretische Betrachtung aufgegriffen und mit echten Motordaten gerechnet.

Das einzige Problem mit den Angaben zu den Motoren: Man muss sich absolut

sicher sein können, dass die diversen Angaben auch alle vom gleichen Arbeitspunkt stammen. Bei Leerlauf ist das noch halbwegs so. Aber sobald es um Lastfälle geht, wird es meist inkonsistent.

Dann steht der I_{max} mal für maximal möglichen Strom (wenn die Welle blockiert), mal für Strom bei maximaler Leistungsabgabe und mal für Strom bei maximalem Wirkungsgrad. Entsprechend gilt das unabhängig für Drehzahlen und Leistungen. I_{max} und n_{max} haben also manchmal nur zufällig den gleichen Index. Im Zweifel muss dann doch nachgemessen werden.

Benutzen der aufgefundenen Parameter: das Motormodell

Aus den vorangegangenen Versuchen sind jetzt vier Parameter bekannt:

- Der Zusammenhang von Spannung und Drehzahl k_U .
- Der Zusammenhang von Drehmoment und Strom k_I .
- Der dynamische Innenwiderstand R_{Dyn} .
- Die internen Drehmomentverluste M_{Lager} .

Alle diese Parameter konnten bereits durch die beschriebenen, einfachen Versuche mit kostengünstigen Messmitteln aufgefunden werden:

- Versuch 1: $k_U \Rightarrow k_I$

bzw.

Versuch 3: $k_I \Rightarrow k_U$

- Versuch 2: R_{Dyn}

- Versuch 4: M_{Lager}

Weniger Arbeitsaufwand, umfangreicherer Versuchsaufbau und sowohl für Maschinen mit und ohne Getriebe geeignet:

- Versuch 5: $k_I, k_U, R_{Dyn}, M_{Lager}$

Auch ein elektrisches Ersatzschaltbild konnte aus den Versuchsergebnissen ermittelt werden, Abb. 7.

Zusammen mit dem Ersatzschaltbild einer realen Spannungsquelle (zum Beispiel ein Generator) ergibt sich ein Stromkreis nach Abb. 16.

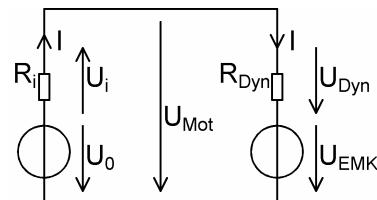


Abb. 16: Ersatzschaltung für Motor mit Speisespannung

Und nun lassen sich die Zusammenhänge auch ganz gut erklären.

Zuerst sei noch ein Ausdruck für den Strom I angegeben. Dieser folgt aus Gleichung (5), unter Zugabe von R_i der Spannungsquelle.

$$I = \frac{U_0 - n \cdot k_U}{R_{Dyn} + R_i} \quad (42)$$

Wir erinnern uns noch an Gleichung (2): $U_{EMK} = n \cdot k_U$. Zuerst ist der Stromkreis unterbrochen, der Motor dreht nicht. In dem Augenblick in dem der Stromkreis nun geschlossen wird, ist $U_{EMK}=0 \text{ V}$ da die Welle steht. (42) vereinfacht sich zu

$$I_{Stop} = \frac{U_0}{R_{Dyn} + R_i} \quad (43)$$

Das entstehende Drehmoment

$$M_{Stop} = I_{Stop} \cdot k_I - M_{Lager} \quad (44)$$

übersteigt normalerweise das von der Last abgeforderte Drehmoment. Dieser Drehmomentüberschuss führt nun zu einer Drehbeschleunigung der Motorwelle; die Drehzahl steigt an. Mit steigender Drehzahl sinkt nun aber gemäß (42) die Stromaufnahme I bis zu dem Punkt, an dem exakt das von der Last abgeforderte Drehmoment erzeugt wird – das Ganze ist eine Gleichgewichtsreaktion. (44) wird zur

Berechnung des Stromes an diesem Gleichgewichtspunkt umgestellt zu

$$I_{Last} = \frac{M_{Last} + M_{Lager}}{k_I} \quad (45)$$

(42) und (45) beschreiben nun den gleichen Strom im Stromkreis und somit dürfen sie auch gleichgesetzt werden. Das führt dann zuerst einmal zu

$$\frac{M_{Last} + M_{Lager}}{k_I} = \frac{U_0 - n_{Last} \cdot k_U}{R_{Dyn} + R_i} \quad (46)$$

Unter der Voraussetzung, dass alle bisher genannten Parameter des Motormodells (k_U , k_I , R_{Dyn} , M_{Lager}) sowie das Lastmoment (M_{Last}) und die Daten der Spannungsquelle (U_0 , R_i) bekannt sind, bleibt als einzige Unbekannte die sich einstellende Drehzahl n_{Last} übrig. (46) umgestellt nach n_{Last} ergibt schließlich

$$n_{Last} = \frac{U_0 - (M_{Last} + M_{Lager}) \cdot (R_{Dyn} + R_i)}{k_U \cdot k_I} \quad (47)$$

Mit der bisher benutzten Einheit von k_U in „V/min⁻¹“ erhält man die Drehzahl n in der gewohnten Einheit „min⁻¹“. Wird für k_U jedoch „V/s⁻¹“ angegeben, dann erhält man n auch in „s⁻¹“, und muss noch entsprechend umrechnen.

Das war ganz schön viel Arbeit um herauszufinden, wie schnell ein bestimmter Motor bei einer bestimmten Last an einer bestimmten Versorgungsspannung dreht. Anstelle nun aber alle diese Arbeitspunkte einzeln auszumessen, sind als Vorleistung gerade einmal die beschriebenen Versuche notwendig. Ersatzweise kann ein seriöses Motordatenblatt verwendet werden. Sind die vier Parameter des Motormodells bekannt, dann lässt sich das Motorverhalten beliebig berechnen. Man denke nur an die Kombination von Motor und Akku. Akkus haben bekanntlich eine nicht konstante Spannung, die mit fortschreitender Entladung absinkt. Der ft-Akku ist mit 8,4 V angegeben – vollgeladen. Der Entlade-

schluss (= Akku leer) ist bei 7 V erreicht. Aus den Motorenndaten, bei beispielsweise 9 V, lässt sich mittels Motormodell auf die Verhältnisse bei Akkubetrieb, z. B. bei 7,2 V, umrechnen.

Ohne es explizit hergeleitet zu haben, sind die beschriebenen Versuche und die abgeleiteten mathematischen Gleichungen, insbesondere (42), (45) und (47), der Kern des mathematischen Rechenmodells für die ft-Motoren mit und ohne Getriebe; natürlich auch für alle anderen baugleichen elektrischen Gleichstrommotoren mit Fremderregung.

Drei kleine Helferlein

Ohne weitere Herleitung oder Hintergrundinformationen seien noch drei Gleichungen angegeben, mit denen die Drehzahlen im Leerlauf, bei optimalem Wirkungsgrad und bei maximaler Leistungsabgabe für eine gegebene Motorspannung errechnet werden können. Mit vorgegebener Motorspannung, bekannter Drehzahl, bekannten Modellparametern und den bereits angegebenen Gleichungen sind die restlichen Betriebsdaten der Arbeitspunkte schnell errechnet.

Drehzahl im Leerlauf

$$n_{Leer} = \frac{U_{Mot}}{k_U} - \frac{R_{Dyn} \cdot M_{Lager}}{k_U \cdot k_I} \quad (48)$$

Drehzahl bei optimalem Wirkungsgrad

$$n_{Opt} = \frac{U_{Mot} - \sqrt{\frac{U_{Mot} \cdot R_{Dyn} \cdot M_{Lager}}{k_I}}}{k_U} \quad (49)$$

Drehzahl bei maximaler Leistungsabgabe (Leistungsanpassung)

$$n_{Max} = \frac{1}{2} \left(\frac{U_{Mot}}{k_U} - \frac{R_{Dyn} \cdot M_{Lager}}{k_U \cdot k_I} \right) \quad (50)$$

Die Gleichungen (48) bis (50) gelten allgemein für Motoren (nicht Generatoren) mit und ohne Getriebe.

Ein Beispiel zur Abrundung

Gegeben sei ein Motor mit den Angaben:

- Nennspannung $U_{Nenn} = 9 \text{ V}$
- Leerlaufdrehzahl $n_{Leer} = 10000 \text{ min}^{-1}$
- Leerlaufstrom $I_{Leer} = 80 \text{ mA}$
- Nenndrehzahl $n_{Nenn} = 6720 \text{ min}^{-1}$
- Nennleistung $P_{Nenn_mech} = 390 \text{ mW}$
- Wirkungsgrad $\eta = 26,2 \%$

Bei der Nennleistung ist man sich sicher, dass es sich um die abgegebene, also mechanische, Leistung handelt. Ebenso ist man sich sicher, dass der Wirkungsgrad für den Nennarbeitspunkt gegeben ist.

Zunächst wird mittels (39) das Nennmoment M_{Nenn} bestimmt:

$$M_{Nenn} = \frac{60 \cdot 0,5W}{2\pi \cdot 8500 \text{ min}^{-1}} = 554,2 \mu\text{Nm}$$

Weiter wird die Nennaufnahmleistung P_{Nenn_elektr} benötigt. Gleichung (14) wird entsprechend umgestellt und eingesetzt:

$$P_{Nenn_elektr} = \frac{0,5W}{0,47} = 1,489W$$

Gleichung (11) definiert die elektrische Leistung als Produkt von Strom und Spannung. Entsprechend errechnet sich der Nennstrom I_{Nenn} zu:

$$I_{Nenn} = \frac{1,064W}{9V} = 165,4mA$$

Weiter geht es mit (40) zu k_I :

$$k_I = \frac{561,7 \mu\text{Nm}}{118,2mA - 80mA} = 0,00649 \frac{\text{Nm}}{A}$$

Und mit (23) wird, weil kein Getriebemotor vorliegt, k_U , gleich in der gewohnten Einheit min^{-1} , daraus errechnet:

$$k_U = 0,00649 \frac{\text{Nm}}{A} \cdot \frac{2\pi}{60} = 679,6 \frac{\mu\text{V}}{\text{min}^{-1}}$$

(Anmerkung: von Nm/A auf V/s^{-1} umzurechnen sieht zuerst nach Pfusch aus.

Allerdings sehe man sich die Überlegungen an, die zu Gleichung (23) führten.)
 M_{Lager} ist durch (10) nun schnell errechnet:

$$M_{Lager} = 0,00649 \frac{\text{Nm}}{A} \cdot 0,08A = 0,519 \text{ mNm}$$

Auch R_{Dyn} ist, Dank (5), kein Geheimnis mehr:

$$R_{Dyn} = \frac{9V - \frac{10000}{\text{min}} \cdot 679,6 \mu\text{V min}}{0,08A} = 27,6 \Omega$$

Damit sind die Parameter des Motormodells bekannt. Der vorgenannte Motor wird aus einem Akkusatz mit Nennspannung $7,2 \text{ V}$ versorgt. Zur Vereinfachung sei angenommen, dass der Innenwiderstand R_i des Akkus 0Ω beträgt. Die angetriebene Mechanik benötigt vom Motor $400 \mu\text{Nm}$ ($= 0,0004 \text{ Nm}$) an Drehmoment.

Aufgabe 1: der Anlauf- bzw. Blockadestrom soll ermittelt werden. Welches Stillstandsdrhmoment erzeugt dieser Strom?

Für den Strom kennen wir bereits Gleichung (43) und benutzen sie:

$$I_{Stop} = \frac{7,2V}{27,6\Omega + 0\Omega} = 260,9mA$$

Mittels (44) errechnet sich das gesuchte Stillstandsdrhmoment:

$$M_{Stop} = 260,9mA \cdot 0,0649 \frac{\text{Nm}}{A} - 0,519 \text{ mNm}$$

$$M_{Stop} = 1174 \mu\text{Nm}$$

Aufgabe 2: Welche Betriebsdaten stellen sich mit dem vorgesehenen Drehmoment ein?

Angegeben sind $400 \mu\text{Nm}$. Durch (45) wird der Motorstrom errechnet:

$$I = \frac{0,4mNm + 0,519mNm}{6,49 \frac{mNm}{A}} = 141,6mA$$

Die aufgenommene elektrische Leistung ist gemäß (15):

$$P_{elektr} = 7,2V \cdot 0,1416A = 1,02W$$

Für U_{EMK} bleiben übrig:

$$U_{EMK} = 7,2V - 0,1416A \cdot 27,6\Omega = 3,292V$$

Der Stromkreis basiert auf Abb. 16. Und mit (2), nach n umgeformt, kann die Drehzahl errechnet werden:

$$n = \frac{3,292V}{679,6 \frac{\mu V}{min^{-1}}} = 4844,0 min^{-1}$$

(47) hätte natürlich auch direkt, anstelle des Umweges über U_{EMK} , benutzt werden können.

Nun kann man auch noch die abgegebene mechanische Leistung mittels (17) angeben:

$$P_{mech} = \frac{2\pi \cdot 0,4mNm \cdot 4844 min^{-1}}{60 \frac{min^{-1}}{s^{-1}}} = 202,9mW$$

Der Wirkungsgrad für diesen Arbeitspunkt errechnet sich mit (14) zu:

$$\eta_{Mot} = \frac{202,9mW}{1020mW} = 0,1989 \approx 20\%$$

Satte 80 % der eingespeisten elektrischen Energie bleiben in diesem Beispiel letztendlich als Verlustwärme im Motor und heizen diesen auf – obwohl der Motor in der Nähe des bei dieser Betriebsspannung erreichbaren optimalen Wirkungsgrades betrieben wird! Für Akkubetrieb sollte man sich jetzt nach Alternativen umsehen; entweder ein anderer Motor oder ein Akku mit höherer Nennspannung.

Aufgabe 3: Wie sehen die Kennlinien für Abgabeleistung und Wirkungsgrad aus?

In den Datenblättern von Motoren sind meist die Kennlinien bei Nennspannung

angegeben. Dabei wird das Lastmoment M als Variable verändert während alle anderen Parameter konstant bleiben. Die Rechnung ist für jeden Stützpunkt absolut identisch zum Rechenweg aus *Aufgabe 2*.

Um eine brauchbare Übersicht zu bekommen legt man beispielsweise 21 Stützpunkte fest. Das Stillstandsdrehmoment M_{Stop} gibt das maximal sinnvolle Lastmoment vor, Leerlauf stellt sich ein bei 0 Nm. Die Stützwerte dazwischen werden entsprechend errechnet.

Von Hand 21 mal die Berechnungen für jeden Arbeitspunkt durchzuführen artet in Fleißarbeit aus. Es bietet sich an alle diese Rechenschritte in ein Tabellenkalkulationsprogramm einzuspeisen. Mittels der meist vorhandenen Schaugrafiken lassen sich die entsprechenden Kennlinien durchaus ansprechend und für die gewünschte Betriebsspannung angepasst erzeugen.

Für den vorgegebenen Motor sind die Kennlinien in Abb. 17 angegeben.

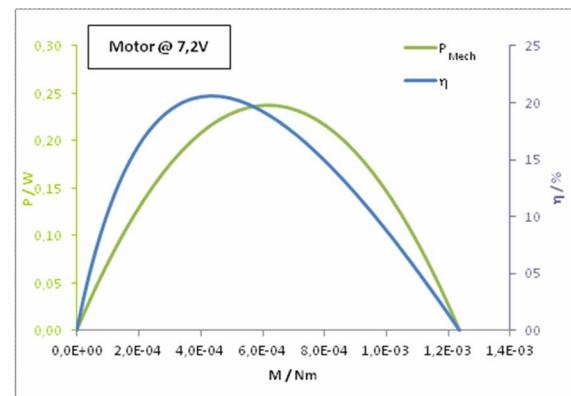


Abb. 17: Kennlinien für Leistung und Wirkungsgrad (Beispielmotor)

Mit Hilfe der Möglichkeiten gängiger Tabellenkalkulationsprogramme lassen sich auch die Auswertungen der Versuche dem Computer übertragen und die manuellen Berechnungen der Modellparameter können entfallen.

Kenndaten eines ft-Mini-Motors

Im Internet ist, mehr oder weniger bekannt, eine Sammlung von ft-Motordaten zu finden [21]. Darin sind zwar für die verschiedenen Motoren einige Werte angegeben, jedoch keine Daten für deren optimalen Einsatz. Es lassen sich auch keine vernünftigen Modellparameter aus diesen Angaben errechnen. Um nun alle interessierenden Größen kennen zu lernen, bleibt also nichts anderes übrig, als diese Motoren alle zu messen.

Für den Motor (32293) aus den beschriebenen Versuchen ergeben sich ohne Getriebe die Modellparameter nach Tab. 9.

k_U	0,000843	V/min^{-1}
k_I	0,00805	Nm/A
R_i	12,15	Ω
M_{Lager}	0,000463	Nm

Tab. 9: Modellparameter (Motor)

Für eine Betriebsspannung von 9 V ergeben sich die Motordaten nach Tab. 11 (siehe nächste Seite).

Die Kennlinien für Abgabeleistung und Wirkungsgrad sind in Abb. 18 dargestellt.

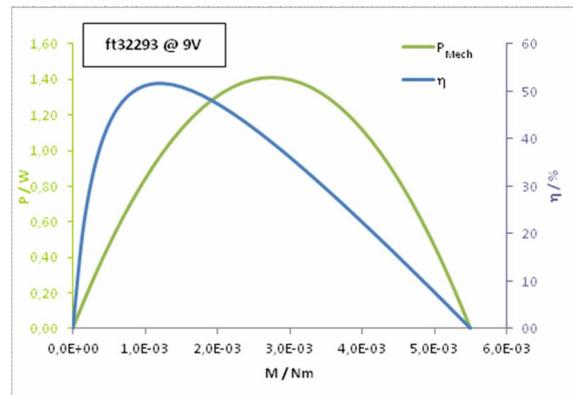


Abb. 18: Kennlinien für Leistung und Wirkungsgrad (Motor)

Für exakt den gleichen Mini-Motor mit Getriebeeinheit (32293, 31078 und 31082) ergeben sich die Modellparameter nach Tab. 10.

k_U	0,054209	V/min^{-1}
k_I	0,0551	Nm/A
R_i	12,67	Ω
M_{Lager}	0,0037	Nm

Tab. 10: Modellparameter (Getriebemotor)

Für eine Betriebsspannung von 9 V ergeben sich daraus die Motordaten nach Tab. 12 (siehe nächste Seite).

Die zugehörigen Kennlinien zeigt Abb. 19.

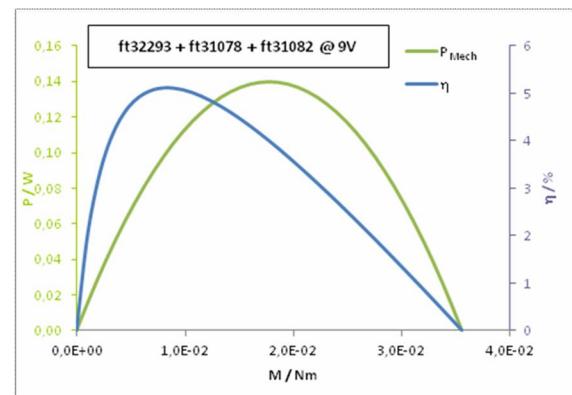


Abb. 19: Kennlinien für Leistung und Wirkungsgrad (Getriebemotor)

Man beachte die unterschiedlichen Skalierungen in den Diagrammen Abb. 18 und Abb. 19!

Bei diesen Beispielen ist zu bedenken, dass es sich jeweils um die Werte für ein bestimmtes Motor-Exemplar handelt. Ohne typische Werte oder umfangreiche Untersuchungen nach statistischen Methoden sind diese Werte nur bedingt zu verallgemeinern. Mit Abweichungen bis zu 30 % der typischen Daten dürfte von Motor zu Motor zu rechnen sein.

Leerlauf	
I_{Leer}	0,058 A
V_{EMK_Leer}	8,292 V
n_{Leer}	9.839 min^{-1}
P_{elek_Leer}	0,52 W
max. Effizienz	
I_{Opt}	0,21 A
V_{EMK_Opt}	6,48 V
n_{Opt}	7.684 min^{-1}
M_{Opt}	0,0012 Nm
P_{mech_Opt}	0,97 W
η_{Opt}	51,8 %
max. Leistung	
I_{max}	0,40 A
V_{EMK_Max}	4,15 V
n_{Max}	4.919 min^{-1}
M_{max}	0,0027 Nm
P_{Mech_Max}	1,41 W
η_{Max}	39,3 %
Stillstand	
M_{stop_AP}	0,0055 Nm
I_{stop_AP}	0,74074 A

Tab. 11: Kenndaten Motor bei 9 V

Grenzen des Motormodells

In der Praxis lässt sich beim Leerlaufversuch feststellen, dass die Stromaufnahme nicht wirklich konstant ist sondern mit steigender Drehzahl leicht ansteigt. Nicht allzu dramatisch, aber doch dezent sichtbar. Dafür ist das so genannte „viskose Reibmoment“ verantwortlich. Die die rotierenden Teile umgebende Luft wird mit verwirbelt und mit steigendem Tempo steigt der Luftwiderstand überproportional an [16]. Damit steigt auch das erforderliche Drehmoment in gleichem Maße. Für das Motormodell wird dieser Effekt ignoriert und so ergibt sich bei hohen Motordrehzahlen eine gewisse geringe Abweichung

Leerlauf	
I_{Leer}	0,067 A
V_{EMK_Leer}	8,149 V
n_{Leer}	150 min^{-1}
P_{elek_Leer}	0,60 W
max. Effizienz	
I_{Opt}	0,22 A
V_{EMK_Opt}	6,23 V
n_{Opt}	115 min^{-1}
M_{Opt}	0,0083 Nm
P_{mech_Opt}	0,10 W
η_{Opt}	5,1 %
max. Leistung	
I_{max}	0,39 A
V_{EMK_Max}	4,07 V
n_{Max}	75 min^{-1}
M_{max}	0,0177 Nm
P_{Mech_Max}	0,14 W
η_{Max}	4,0 %
Stillstand	
M_{stop_AP}	0,0355 Nm
I_{stop_AP}	0,71014 A

Tab. 12: Kenndaten Getriebemotor bei 9 V von der Realität. In der ft-Modellbaupraxis ist dieser Fehler meist vernachlässigbar.

Mit steigender Drehzahl nehmen die Wirbelstromverluste im magnetischen Material des Rotors zu [17]. Auch dies führt zu einem Anstieg der Stromaufnahme mit steigender Drehzahl. Dieser Effekt wird im Motormodell nicht berücksichtigt.

Völlig unterschlagen wird die Tatsache, dass der magnetische Kreis im Motor nichtlineare Eigenschaften besitzt. Je höher der Motorstrom liegt, umso stärker machen sich Sättigungseffekte bemerkbar [18]. Diese Effekte führen in der Realität mit steigendem Motorstrom zu einer Reduktion von k_U und k_I .

Dieser Effekt wird hier ebenfalls ignoriert. Weiter gibt es Rückwirkungen vom Magnetfeld des Rotors auf das Magnetfeld des Stators [19]. Der Effekt führt mit zu den bekannten Funken am Stromwender (Kommutator) sobald die Maschine größere Leistung übertragen muss.

Auch der dynamische Innenwiderstand ist Schwankungen unterworfen. Man denke nur an die thermische Beanspruchung der Wicklung durch den Stromfluss. Höhere Temperaturen bedingen auch höhere Wicklungswiderstände, die in R_{Dyn} eingehen.

Die Lagermomente sind sicherlich auch nicht ganz konstant. Je nach tatsächlich auftretender radialer und axialer Belastung variieren sie zum Teil erheblich. Insbesondere beim Anlaufen eines Motors aus dem Stillstand tritt noch der Unterschied zwischen Haftreibung und Gleitreibung am Kommutator und in allen Wellenlagern auf [20]. Der Strom muss einen bestimmten Wert überschreiten um die Welle „loszubrechen“, danach schießt die Drehzahl auf einen recht hohen Wert, die Stromaufnahme sinkt wieder ab. Auch diese Effekte sind im verwendeten vereinfachten Motormodell nicht berücksichtigt.

Zusammenfassung

Das hier benutzte Motormodell eignet sich zur Ermittlung von statischen Arbeitspunkten. Für dynamische Vorgänge (Beschleunigen, Bremsen) fehlt das Trägheitsmoment J_{Mot} (vom Motor und der Last auch noch). Für Untersuchungen zur Auslegung von Motortreibern fehlt zumindest die dynamische Induktivität L_{Dyn} .

In der Praxis werden solche idealisierten Motormodelle dennoch häufig angewendet. Sind die Parameter für das Motormodell ermittelt, lässt sich damit durchaus recht brauchbar das Verhalten des Motors voraussagen. Die Kennlinien lassen sich erzeugen. Liegen sie vor, kann der Arbeits-

punkt durch bloßes Ablesen im Handumdrehen abgeschätzt werden. Auch der Vergleich verschiedener Motoren wird so recht anschaulich ermöglicht.

Für einen Mini-Motor (32293) ohne und mit U-Getriebe (31078 + 31082) liegen durch die Untersuchungen erstmals die Modellparameter und die vollständigen Kenndaten für 9 V-Betrieb vor. Kenndaten für andere Betriebsspannungen lassen sich errechnen.

Service

Zur Hilfe für die Versuchsauswertung und auch zur Verwendung der Modellparameter sind im Download-Bereich der ft:community zwei vorbereitete Excel-Sheets bereitgestellt [22, 23]. Beide Dateien kommen ohne jegliche Makros aus. Die Inhalte sollten nach der Lektüre dieses Artikels selbsterklärend sein.

Quellen

- [1] Wikipedia: [Induktionsgesetz](#).
- [2] Wikipedia: [Spezifischer elektrischer Widerstand](#).
- [3] Wikipedia: [Erdbeschleunigung](#).
- [4] Physikalisch-Technische Bundesanstalt: [Erdbeschleunigung für beliebige Orte](#).
- [5] Wikipedia: [Lorentzkraft](#).
- [6] Wikipedia: [Elektrische Maschinen](#).
- [7] Wikipedia: [Elektrische Gleichstrommaschine](#).
- [8] Wikipedia: [Leerlaufspannung](#).
- [9] Walter Fendt: [Animiertes Schaubild Elektromotor](#).
- [10] Walter Fendt: [Animiertes Schaubild Generator](#).
- [11] Wikipedia: [Elektrische Leistung](#).
- [12] Wikipedia: [Mechanische Leistung](#).
- [13] Wikipedia: [Ohm'sches Gesetz](#).

- [14] Wikipedia: [Georg Simon Ohm](#).
- [15] Wikipedia: [Wirkungsgrad](#).
- [16] Wikipedia: [Strömungswiderstand](#).
- [17] Wikipedia: [Wirbelstrom](#).
- [18] Wikipedia: [Magnetische Sättigung](#).
- [19] Wikipedia: [Ankerrückwirkung](#).
- [20] Wikipedia: [Losbrechmoment](#).
- [21] fischertechnik: [Daten einiger ft-Motoren](#), fischertechnik.
- [22] Zum Download: [Spreadsheet zur Versuchsauswertung](#).
- [23] Zum Download: [Spreadsheet zur Motorberechnung](#)
- [24] Forumsbeitrag: [Daten von ft-Motoren](#), ftcommunity.de

Elektronik

Automatik für weichen Motorstart und -stopp

Werner Hasselberg

Getreu dem Grundsatz von fischertechnik, technische Bildung zu vermitteln, will dieser Beitrag zeigen, wie man mit den Elektronik-Bausteinen von Fischertechnik aus den 1980iger Jahren umgeht und was man mit ihnen machen kann. Der Beitrag beschreibt im Detail, wie ein Motor vollelektronisch gesteuert langsam zum Stehen kommt, eine gewisse Zeit gestoppt bleibt, um dann ebenso langsam wieder auf Touren zu kommen. Dazu existiert im Begleitheft [1] bereits ein Vorbild, der „weiche Start und Stopp“ eines Motors, das hier aber beträchtlich erweitert wird, um eine vollautomatische Funktionssteuerung zu erhalten.

Im Mittelpunkt stehen hier das grund-sätzliche Verständnis der Schaltungen und deren logischer Aufbau; tiefer gehende elektronische Theorie soll dabei keine größere Rolle spielen. Der Beitrag beschränkt sich auf prinzipielle Abläufe in den jeweiligen Bausteinen. Deren Bauteile sind immer noch im Handel erhältlich, und findige Bastler bieten vollständige Anleitungen zu deren Nachbau im Internet an. Natürlich können deren Möglichkeiten heute nicht mehr mit einem Interface mithalten. Aber dennoch ist es erstaunlich, was alles mit diesen Bausteinen möglich ist. Ein schönes Beispiel hierfür ist die „weiche Start- und Stopp-Automatik“ eines Motors. Damit kann beispielsweise ein Zug der Bau-Spiel-Bahn (BSB) langsam anfahren und stoppen, was bei automatischen Steuerungen sehr nützlich ist. Aber auch für Transportbandsteuerungen oder das vorsichtige Anheben von Lasten kann die Schaltung gute Dienste leisten.

Im Handbuch zum „Elektronik“-Baukasten ([39190](#)) befindet sich bereits die Basis unserer Schaltung, ein allein auf der Leistungsstufe aufgebauter weicher Start und Stopp eines Motors. Mittels eines manuellen Schalters wird ein Motor

langsam angefahren und ebenso gestoppt. Diese Grundlage entwickeln wir zu einer vollautomatischen Anlage weiter, so dass ein Zug beim Passieren einer Lichtschranke von selbst langsam anhält, (am Bahnhof) eine gewisse Zeit steht und dann langsam wieder anfährt.

Die Schaltung muss deshalb folgende Aufgaben erfüllen.

1. Das weiche Aus- und Anschalten des Motors,
2. die Steuerung einer Lichtschranke für einen Dunkelimpuls und
3. das Miteinbeziehen eines Zeitschalters für die Wartezeit am Bahnhof.

Alle drei Aufgaben sollen mit nur einem System, also mit einer Leistungsstufe (kurz LST) und einem Schwellwertschalter (SWS) erledigt werden, um möglichst wenig Bausteine zu benötigen. Je weniger man braucht, desto mehr kann man mit dem vorhandenen Vorrat basteln.

Zuerst aber ein Hinweis zum „Dunkelimpuls“: Ein Dunkelimpuls bedeutet, dass eine Unterbrechung der Beleuchtung eines Fotowiderstandes (FW), also ein Nachlassen der Lichtintensität auf dessen Oberfläche, ihn abdunkelt. Geschieht das nur

einen kurzen Augenblick handelt es sich um einen Dunkelimpuls. Der Dunkelimpuls ist im Unterschied zum Lichtimpuls auf den FW deshalb von Vorteil, weil dafür keine Lampe auf dem fahrenden Zug montiert werden muss. Lampe und FW befinden sich einfach gegenüber und die Zuggleise verlaufen dazwischen. Der Abstand zwischen Lampe und FW ist deshalb auch nicht besonders groß, so dass deren Licht für den FW in jedem Fall ausreicht, die Elektronik der Bausteine zu schalten. Passiert ein Zug die so erzeugte Lichtschranke, löst das den Dunkelimpuls aus und die Schaltung für das Anhalten kann loslegen.

Jetzt können wir uns ganz dem Aufbau der Schaltung widmen. Welche elektronischen Bauteile sind nötig?

- Ein komplettes System aus dem Elektronik-Baukasten, also Spannungsversorgung (SPV), Schwellwertschalter (SWS) und eine Leistungsstufe (LST),
 - ein Relais aus dem Baukasten em3 oder ein Relais mit gleichen Eigenschaften, was die Betriebsspannung und die Anzahl der Schalter betrifft (wenn keines verfügbar ist, kann man sich auch mit einem Eigenbau aus den fischertechnik Elektromechanik-Baukästen behelfen),
 - ein $22\ k\Omega$ -Widerstand,
 - drei Dioden sowie
 - zwei Elektrolytkondensatoren (Elkos) $470\ \mu F$.

Der weiche Start und Stopp

Im Begleitheft des Elektronik-Kastens [1] wird dazu nicht viel erklärt; das soll hier wenigstens im Ansatz nachgeholt werden. Im Kern arbeitet die Schaltung mit pulsierender Gleichspannung, also nicht mit der Spannung der SPV, sondern direkt vom Netzgerät. Reine Gleichspannung ist ungeeignet. Der Kondensator des Baukastens

lässt ihn nur sehr kurz hindurch, dann wird dessen Widerstand zu groß und es kann kein Strom mehr abfließen. Pulsierender Gleichstrom lädt den Kondensator (Elko) langsamer auf. Erst gegen Ende des Ladevorgangs fließt keine bzw. kaum noch Ladung durch den Elko. Genau diese Eigenschaft nutzen wir für den langsamten Start und Stopp eines Motors. Die folgende Abbildung zeigt den Aufbau der Schaltung aus dem Elektronik-Anleitungsbuch:

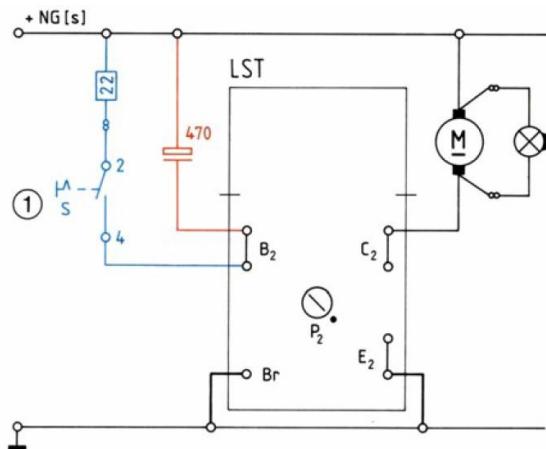


Abb. 1: Schaltung von Seite 44 des Elektronik-Begleitheftes [1]

Das blaue Symbol unter dem $22\text{ k}\Omega$ -Widerstand (das blaue, rechteckige Kästchen) stellt den Ein-/Ausschalter dar. Am Ausgang C2 der Leistungsstufe hängen der Motor und parallel dazu eine Lampe. Sie unterstützt das langsamere An- und Auslaufen des Motors, weil sich dadurch die Stärke des Stroms verringert, der direkt durch den Motor fließt. Falls Start und Stopp noch nicht gleichmäßig verlaufen, kann eine zweite Lampe parallel geschaltet werden. Kern der Schaltung ist der Elko $470\text{ }\mu\text{F}$ in Verbindung mit dem Transistor der Leistungsstufe.

So wie der Elko in der obigen Abbildung geschaltet ist, kann er sich über den Motor zwar entladen, allerdings ist dessen Leistung viel zu schwach und der Widerstand des Motors zu hoch, um direkten Einfluss zu nehmen und ihn zu starten. Das geht nur über den Transistoreffekt des

LST-Bausteins. Dessen Basis B2 wird mit der (-) Seite des Elkos verbunden. Parallel dazu liegt der Schalter mit einem $22\text{ k}\Omega$ -Widerstand (blaue Leitung).

Ist der Schalter geöffnet, lädt sich der Elko über die (+) Schiene vollständig auf. Sein Widerstandswert wird dadurch zunehmend größer, so dass immer weniger Strom in die Basis fließen kann. Der zunehmend geringere Steuerstrom bewirkt einen zunehmend geringeren Laststrom in C2, und der Motor läuft langsam aus.

Beim Schließen des Schalters kann sich der Elko nur über die blaue Leitung entladen, was dessen Widerstand sofort verringert. Der pulsierende Gleichstrom lädt ihn aber sofort wieder nach. Es erfolgt somit ein ständiges Ent- und Aufladen des

Elkos. Es fließt aber auch Strom durch die blaue Leitung nach B2. Durch den Widerstand ist diese Last alleine zu gering, um B2 anzusteuern. Der Elko entlädt sich aber auch über diese Leitung, weshalb das Entladen insgesamt schneller erfolgt als das Aufladen. Durch den damit nachlassenden Widerstand des Elkos steuert B2 mit langsam steigernder Leistung C2 an und fährt den Motor langsam wieder hoch.

Dieses Prinzip erweitern wir jetzt um eine automatische Steuerung. Das erfordert den Einsatz des em3-Relais als Schalthilfe.

Dabei machen wir uns zunutze, dass der Elko sich nur dann entladen kann, wenn dessen Strom von der Plusseite nach Minus abfließen kann (in Abb. 1 durch die blaue Leitung zu B2). Anstelle des Schalters

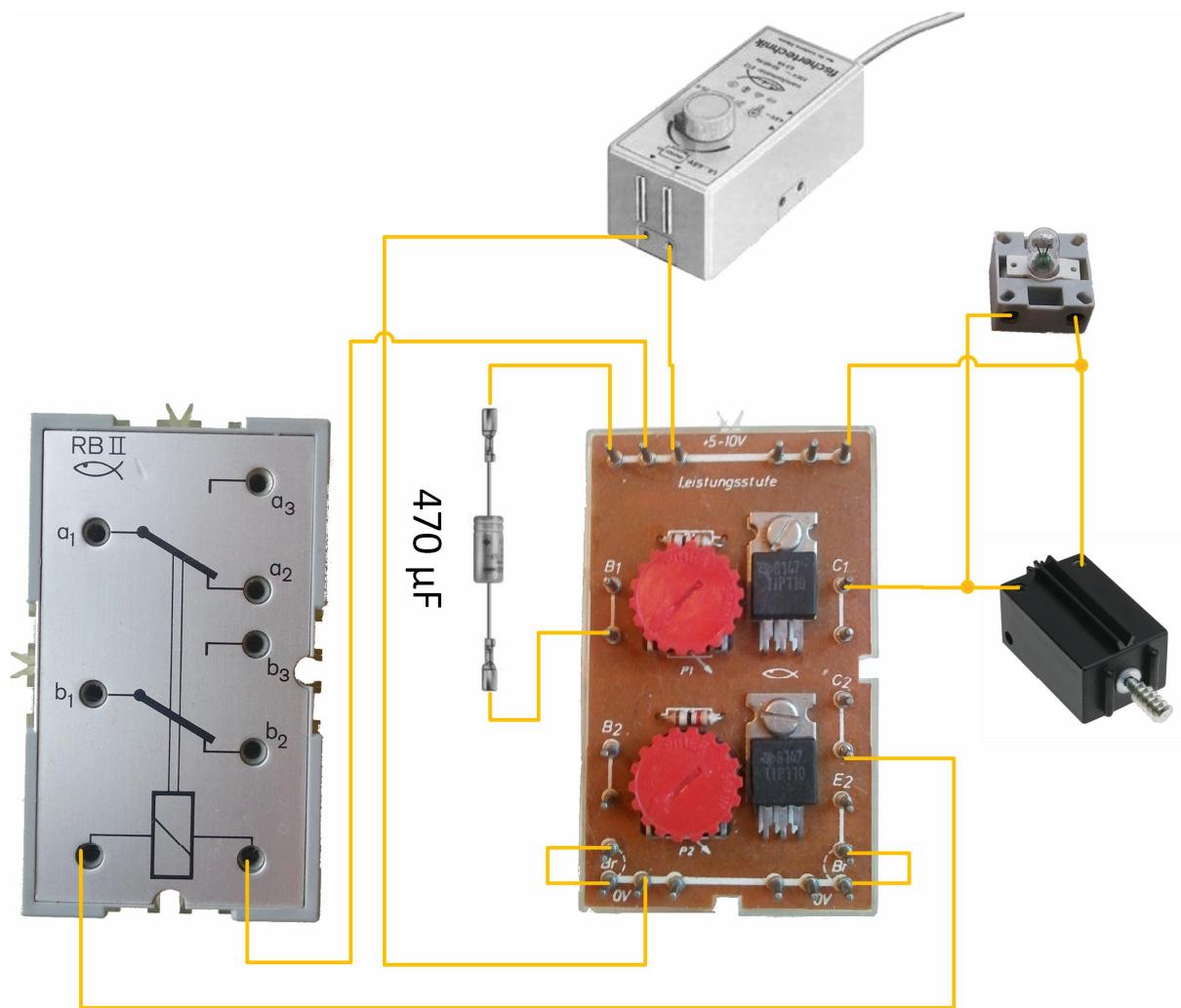


Abb. 2: Schaltung mit em3-Relais

verwenden wir jetzt das Relais und schalten es an die LST nach Abbildung 2. Allerdings wird es nicht an B2 angeschlossen, sondern an C2, und auf den Widerstand wird verzichtet. C2 wird negativ, sobald Strom an B2 liegt. Das schließt den Kreislauf des Relais und schaltet es zu gegebener Zeit ein.

Es fällt auf, dass keiner der Relaiskontakte den blauen Schalter ersetzt, sondern das Relais selbst als Schalter dient. Sobald es anzieht, kann es auch die Entladung des Elkos durch C2 bewirken und damit den Motor langsam anlaufen lassen, wie zuvor der Widerstand in Verbindung mit dem Schalter. Der Elko wird hier an B1 angeschlossen – nicht an B2 wie in Abb. 1 – denn dieser Basis-Pin ist für den Zeitschalter reserviert. Der Zug fährt also los, wenn das Relais anzieht. Sein Widerstand entspricht etwa den $22\text{ k}\Omega$ der vorherigen Schaltung. Die beiden Brücken unten an der LST sind nötig, um diese ebenfalls in Betrieb zu nehmen, an deren C2-Pin das Relais bereits geschaltet ist.

Im nächsten Abschnitt entwickeln wir den automatischen Schalter für das Starten und Stoppen des Zuges, die Lichtschranke. Abb. 4 zeigt die komplette Schaltung.

Die Lichtschranke

Wenn der Zug die Lichtschranke passiert, soll der Stoppvorgang eingeleitet werden.

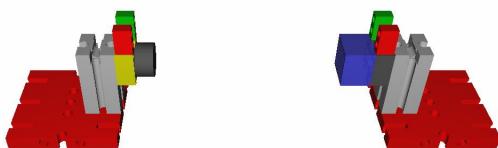


Abb. 3: Beispiel einer Lichtschranke

Der Fotowiderstand wird im SWS2 am unteren Brücken-Pin und (-) angeschlossen (siehe Abb. 4) und ein $22\text{ k}\Omega$ -Widerstand an dem (+5 V) Pin und EB gesteckt. Der SWS2 ist ausgeschaltet, solange der FW beleuchtet wird. Dessen Widerstandswert ist in diesem Zustand geringer als der des

SWS2-Poti (je nach Einstellung). Der Strom fließt dadurch direkt von (+) über den FW direkt zur (-) Schiene ab, ohne in die Steuerstufe (den SWS2) zu gelangen. Der SWS2 bleibt aus, die rote LED neben dem SWS2 leuchtet nicht.

Wird der FW dagegen abgedunkelt, erhöht sich dessen Widerstand erheblich. Sobald er höher wird als der des SWS2-Poti, fließt Strom in den SWS2 und schaltet ihn ein (rote LED leuchtet). Bekommt der FW wieder mehr Licht, sinkt sein Widerstand, der Strom fließt wieder direkt über die (-) Schiene ab und SWS2 schaltet sich wieder aus. Passiert also z. B. ein Zug die Lichtschranke, schaltet das den SWS2 ein. Er geht jedoch sofort wieder aus sobald der Zug vorbei ist, außer er besteht aus vielen Wagen und stoppt vorher.

Eine andere von der Zuglänge unabhängige Lösung ist, einen noch so kurzen Dunkelimpuls solange wie nötig zu speichern, indem wir den eingeschalteten SWS2 in eine so genannte Selbsthaltung versetzen. Einmal eingeschaltet, bleibt er durch diese Selbsthaltung einfach an, ganz egal ob der FW beleuchtet wird oder nicht. Das erreichen wir durch eine Diode, die wir zwischen QB und EB anbringen (die untere der drei Dioden in Abb. 4). Wichtig ist die richtige Polung: Ein positives Signal muss von QB nach EB gelangen können. Einmal eingeschaltet, kann sich der SWS2 jetzt nur ausschalten, wenn der in Abb. 4 rechte Anschluss der Diode direkt mit (-) verbunden würde oder gar nicht angeschlossen wäre.

Jetzt kommt das em3-Relais ins Spiel. Es ist eingeschaltet, wenn der Fotowiderstand beleuchtet ist, denn: C2 führt immer dann „-“, wenn \overline{QA} oder \overline{QB} des SWS durch die beiden Schutzdiode „+“ liefern. Die beiden Schutzdiode bewirken ein logisches „Oder“ zwischen $\overline{QA}/\overline{QB}$ und verhindern Kurzschlüsse zwischen den Q-Ausgängen. Das Beleuchten des Fotowiderstandes führt zu „+“ an \overline{QB} – also

führt letztlich das Beleuchten des FW zum Anziehen des Relais.

Durch einen Ruhekontakt des Relais verläuft die Selbstthalteodiode. Sobald eine Unterbrechung der Lichtschranke SWS2 einschaltet, schaltet sich das Relais ja sofort aus und die Selbsthaltung des SWS wird aktiv: Er bleibt aktiv, weil das „+“-Signal von QB bei abgefallenem Relais durch die Selbstthalteodiode dem Eingang EB zugeführt wird. Gleichzeitig startet über einen Arbeitskontakt des Relais der Zeitschalter des SWS1, der je nach Poti-Wert die Stopzeit des Motors vorgibt und am Ende das Relais und damit den Motor wieder einschaltet.

Zeitschalter für Stoppdauer und Wiederanfahren

Jetzt folgt der einfachste Teil der Schaltung. Der zweite Elko wird am SWS1

Brücken-Pin mit seiner (+) Seite angebracht und seine (-) Seite mit der (-) Schiene verbunden. Er wird über einen Relais-Arbeitskontakt ein- und ausgeschaltet. Das geht am besten mit der Krokodilklemme aus dem Baukasten. Ist das Relais eingeschaltet, wird der Elko voll aufgeladen. In diesem Zustand ist der SWS1 eingeschaltet – die obere rote LED leuchtet.

Schaltet sich das Relais aus (SWS2 ist wie zuvor beschrieben jetzt ein), wird der Arbeitskontakt unterbrochen. Es findet kein weiteres Aufladen statt, und die Zeitverzögerung beginnt. Der Elko beginnt sich je nach Poti-Stellung langsam über den Brücken-Pin zu entladen.

Jetzt sind beide SWS eingeschaltet, und weil beide über Q mit der LST2 verbunden sind, ist diese in jedem Fall ausgeschaltet – \overline{QA}/QB sind (-). Sobald die Wartezeit des

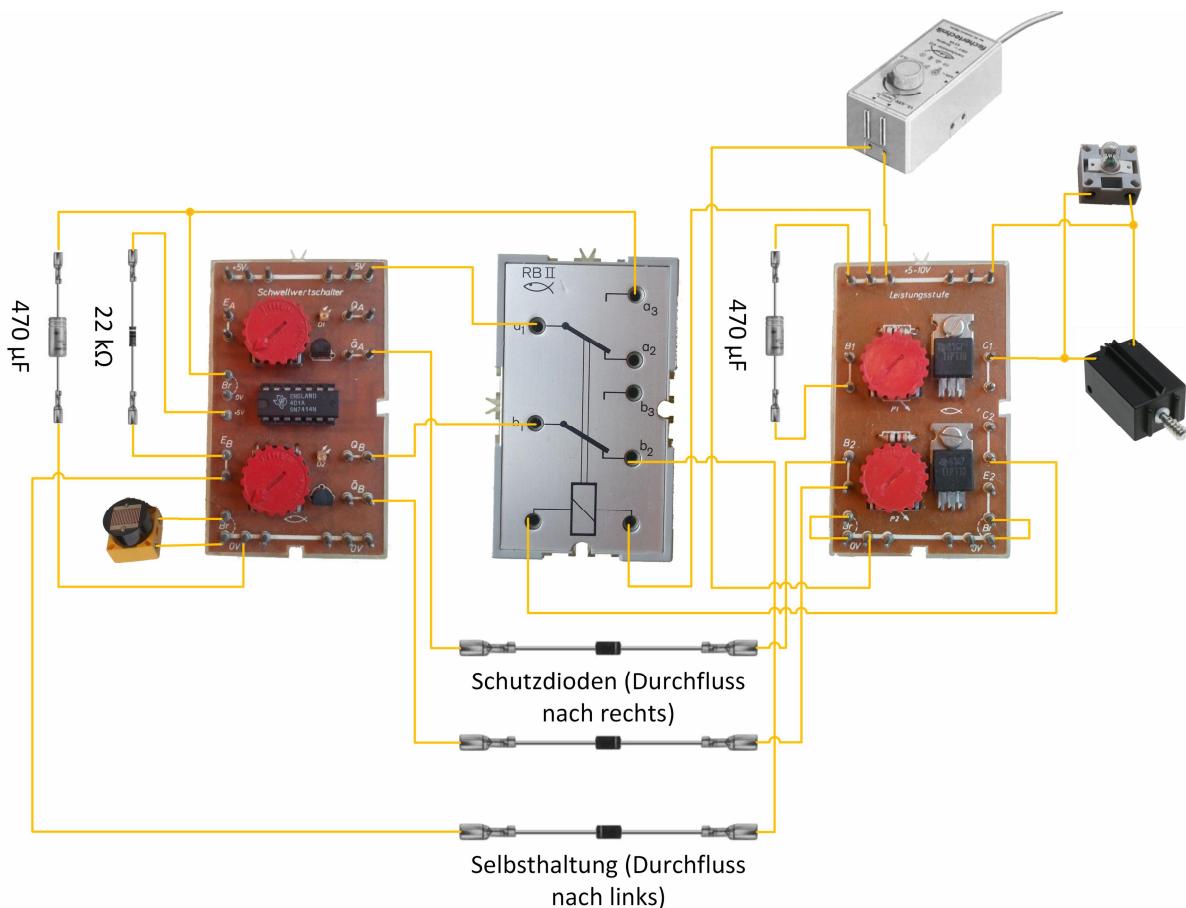


Abb. 4: Die komplette Schaltung

Zeitschalters abgelaufen ist, schaltet sich SWS1 aus. Sofort wechselt QA die Polung auf (+) und die LST2 schaltet sich ein. Damit schaltet sich das Relais wieder ein und der Zug fährt langsam wieder los.

Die komplette Schaltung zeigt Abb. 4 (ohne Spannungsversorgung und die einzelnen Verbindungen der Bausteine über die (+) und (-) Schienen). Die sollte aber kein Problem darstellen und wird im Elektronik-Begleitheft genau beschrieben [1], ebenso die Möglichkeiten, wie der Elko als Zeitschalter verwendet werden

kann. Wichtig ist nur, dass lediglich der SWS über den Spannungsversorgungs-Baustein betrieben, die LST aber direkt an der pulsierenden Gleichspannung des Netzteils angeschlossen wird.

Viel Erfolg und Spaß beim Basteln!

Quellen

- [1] Fischer-Werke: *Elektronik Bau-kasten-Begleitheft*, Tumlingen, 1981

Tipps & Tricks

ft-Spezialteile made by TST (Teil 5)

Andreas Tacke

In einer lockeren Reihe stellt TST einige von ihm entwickelte Spezialteile vor, die so manche Lücke beim Bauen mit fischertechnik schließen. Diesmal geht es um raffinierte Modifikationen des Hubgetriebes 37272.

In meinem heutigen Beitrag geht es um das Hubgetriebe. Vor einigen Jahren stand bei mir ein neues Projekt an: Ein Drei-Achsen-Roboter sollte es werden. Um sehr kompakt bauen zu können, sollte eine Achse per Zahnstange und Hubgetriebe bewegt werden.

Da stellte sich mir die Frage, wie ich die Positionsabfrage vornehmen sollte. Mit Schaltnocken und Taster auf dem Verfahrweg wollte ich nicht arbeiten, da dies nicht sehr komfortabel ist.

Also suchte ich nach einer Möglichkeit, Wegimpulse am Hubgetriebe abzufragen um diese dann an das Interface zu senden. Nach vielen vergeblichen Versuchen hatte ich schließlich die zündende Idee: Ein Impulsrad musste her, um die Schritte zählen zu können. So entstand der erste Prototyp (Abb.1).

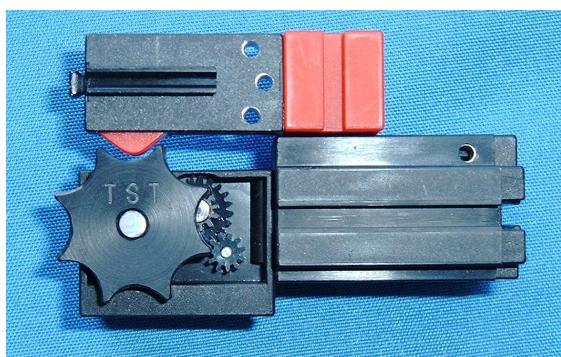


Abb. 1: Hubgetriebe mit Impulsrad, Version 1

Es zeigte sich allerdings, dass das Interface mit der Frequenz der eingehenden Impulse nicht zurecht kam – es waren zu viele Schaltvorgänge pro Umdrehung. Ein weiterer Nachteil dieser Version lag darin, dass die Achse, an der das Impulsrad montiert wurde, zusätzlich angefertigt werden musste. Also hieß es: weitertüfteln.

Dann kam Version 2, bei der ich das außen liegende Zahnrad Z11 durch ein Impulsrad mit Zahnrad Z11 ersetzte (Abb. 2, 3).

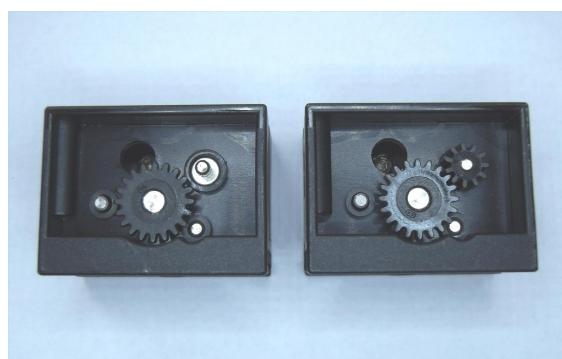


Abb. 2: Demontage des Z11

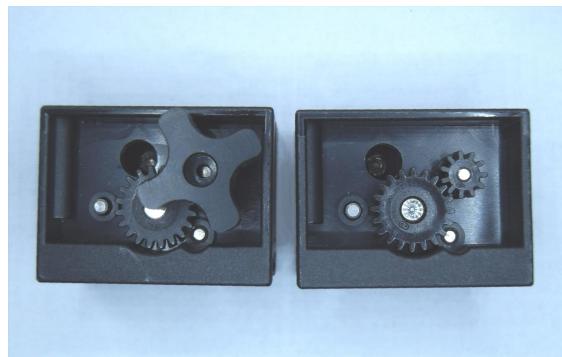
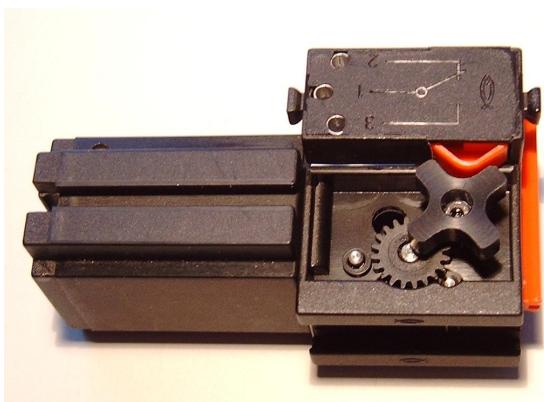


Abb. 3: Hubgetriebe mit montiertem Impulsrad



*Abb. 4: Hubgetriebe mit Impulsrad,
Version 2 mit Taster 37783*

Dieses Impulsrad besitzt entgegen der ersten Version nur noch vier Schaltnocken, genau wie das Impulsrad 37157.

Das Impulsrad besteht aus dem Zahnräder 75107 sowie dem Impulsrad, das ich aus Polyoxymethylene (POM) herstelle. Das originale Impulsrad 37157 konnte ich leider nicht verwenden, da dessen Außen-durchmesser zu klein ist. Also ran ans CAD-Programm und selber eines entworfen (Abb. 5).

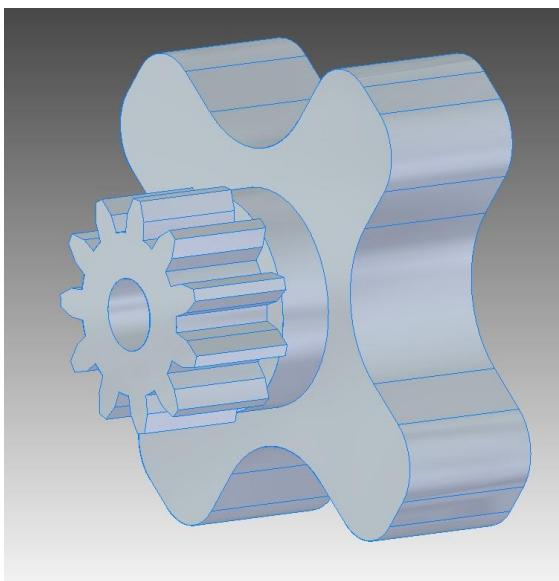
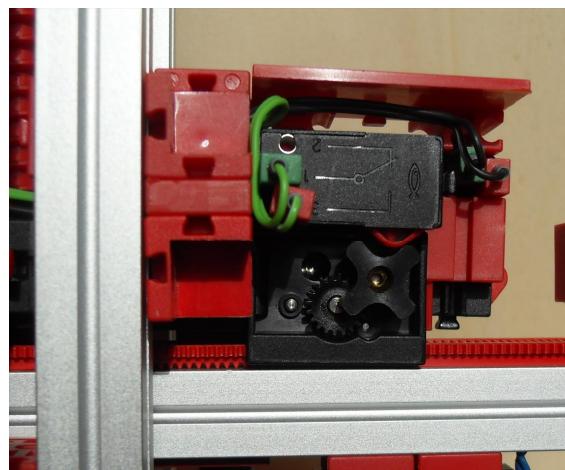


Abb.5 : Impulsrad mit Zahnräder Z11

Nun werdet ihr euch mit Recht fragen: „Und wie genau kann man damit nun die Position abfragen?“

Bei einer Umdrehung des Impulsrades legt die Zahnstage einen Weg von 9,44 mm zurück. Das entspricht je Impuls 2,36 mm (9,44 mm geteilt durch vier Impulse).

Nutzt man die Möglichkeit in Robo Pro, die steigende und die fallende Flanke zu zählen, so verdoppelt sich die Genauigkeit auf 1,18 mm je Flankenwechsel. Dies reicht für die meisten Anwendungen normalerweise vollkommen aus.



*Abb.6 : Hubgetriebe mit Impulsrad
im Modell verbaut*

Damit gelang es, für meinen Roboter die Bewegung der Achse recht präzise und ohne großen Aufwand abzufragen. Ein weiterer Vorteil dieser Ausführung ist, dass sich der Standard-Taster 37783 verwenden lässt. Es wird also kein zusätzlicher Sensor wie eine Lichtschranke oder Ähnliches benötigt. Und so passt das modifizierte Getriebe perfekt ins fischertechnik-System.

Und wieder mal gilt: Geht nicht, gibt's nicht. Man muss sich nur etwas einfallen lassen...

Tipps und Tricks

Raffiniertes mit Achsen

Stefan Falk

Kurz und knapp stellen wir zwei pfiffige Detaillösungen mit Achsen in der Hauptrolle vor.

Trick 1: Mehrfachausnutzung derselben Achse

Sowohl der Ur-fischertechnik-Gelenkstein als auch dessen jüngste Generation erlauben es, genau durch den Drehpunkt des Gelenks auch noch eine Achse hindurchzuführen:

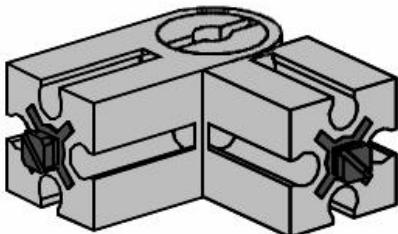


Abb. 1: Der Ur-[Gelenkstein](#)

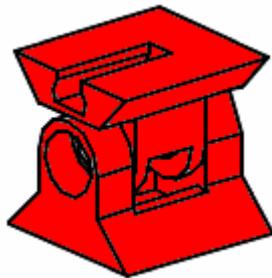


Abb. 2: Der aktuelle [Gelenkwürfel](#)

Das kann man wunderbar verwenden, um von einem fest angebrachten Antrieb Drehmoment auf eine Baugruppe zu übertragen, die mit eben diesen Gelenken schwenkbar aufgehängt ist. Zwei Beispiele:

Beispiel 1: Zahnradkupplung

Abb. 3 zeigt eine Kupplung: Das mit einem grünen Pfeil markierte Z40 sitzt auf einer Achse, die durch zwei der Ur-Gelenksteine verläuft. Es wird vom Motor über das Differential von oben angetrieben, dreht sich also um seine Achse.

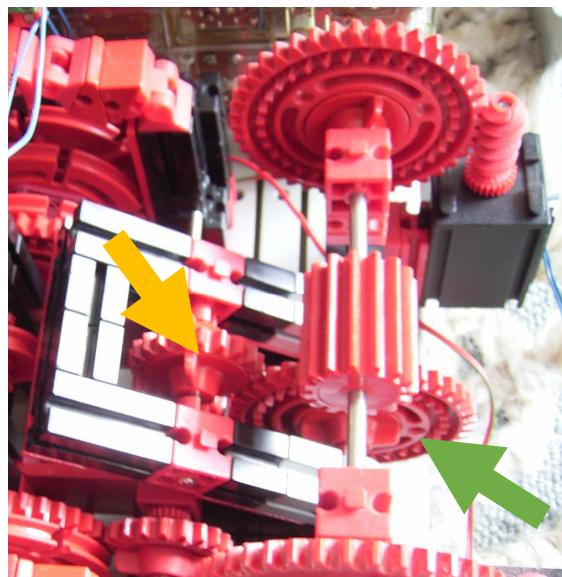


Abb. 3: Kupplung eines [Free-Fall-Towers](#)

An den Gelenksteinen aber sitzt ein (mit dem gelben Pfeil markiertes) Z20 im richtigen Abstand, um vom Z40 angetrieben zu werden. Das Z20 dreht sich also um seine eigene Achse, gleichzeitig kann es aber völlig problemlos um die Zentralachse geschwenkt werden, die durch das Z40 und die Gelenksteine verläuft. Das geht deshalb, weil sein Abstand vom Z40 immer gleich bleibt – es wird ja genau um das Z40 herum geschwenkt.

In einem Free-Fall-Tower-Modell wurde das wie folgt ausgenutzt: Unterhalb des Z20 saß wiederum ein Abtriebszahnrad (in diesem Fall wieder ein Differenzial, aber das tut nichts zur Sache). Wenn nun das Z20 nach unten auf das Abtriebszahnrad abgesenkt wurde, kuppelte es ein und die Antriebskraft wurde weitergeleitet. Im Modell wurden tatsächlich Federgelenksteine ([31308](#)) verwendet, um das Z20 nach unten zu drücken. Schwenkte man das Z20 aber mit einem geeigneten Antrieb nach oben, wurde elegant ausgekuppelt – die Drehung des Z20 gelangte nicht auf weitere Zahnräder.

Diese Art der Kupplung hat noch weitere Vorteile gegenüber solchen, bei denen ein Zahnrad quer verschoben in ein anderes eingreifen muss: a) ersetzt man einen komplizierten Schiebemechanismus durch eine einfache, elegante Anordnung, und b) kuppelt diese Form viel leichter ein, ohne dass sich zwei Zahnräder verklemmen oder verhaken können.

Beispiel 2: Schwenkbarer Antrieb mit stationär angebrachtem Motor

Abb. 4 zeigt die Hebemechanik für den Kranarm eines [Teleskop-Mobilkrans](#): Der Power-Motor treibt das Zentralrad an (grüner Pfeil), welches wieder auf einer Achse sitzt, die mitten durch Gelenke führt. Das Abtriebsrad (gelber Pfeil) ist wieder ein Z20, und so kann Drehmoment vom fest angebrachten Motor elegant auf die schwenkbare Mechanik gebracht werden.



Abb. 4: Kranarm-Hebemechanik eines [Autokrans](#)

Hier sieht man noch einen kleinen Konstruktions-Kniff: In diesem Modell muss auf die langen Schneckenachsen sehr viel Kraft gebracht werden. Um möglichst leicht bauen zu können, wird die erst spät aufgebaut: Ein 50:1-Power-Motor mit schwarzer Kappe dreht relativ schnell, also drehen sich auch das zentrale Z30 und das Abtriebs-Z20 sehr schnell. Über einen zweistufigen Schneckentrieb wird erst sehr spät innerhalb der Gesamtmechanik hohe Drehzahl gegen hohes Drehmoment „eingetauscht“. So gelingt es, mit nur wenigen Bauteilen eine enorm kräftige Schiebemechanik herzustellen.



Abb. 5: Spätes Eintauschen von hoher Drehzahl gegen hohes Drehmoment und hohe Schiebekraft

Trick 2: Kräftiger Seilantrieb auf kleinstem Raum

Wenn man die Aufgabe hat, ein Seil sehr kräftig per Motor anzuziehen, aber kein Platz für Untersteckgetriebe zur Verfügung steht, kann man das Seil auch per Klemmring einfach direkt auf einer Achse anbringen. Es wird dann nur auf der 4-mm-Achse, also mit nur 2 mm Radius, angetrieben. Das ergibt eine enorme Zugkraft ohne jegliches Zwischengetriebe und mit minimalem Platzbedarf. Auch hier zwei Beispiele:

Beispiel 1: Seilzug für einen Teleskop-Arm

In Abb. 6 werden zwei Achsen langsam, aber kräftig angetrieben:



Abb. 6: Seilzug eines Teleskop-Arms des oben schon erwähnten [Mobilkrans](#)

Die direkt auf diesen Achsen angebrachten Seile bringen eine hinreichend große Kraft auf, um einen Dreifach-Teleskoparm auszufahren:

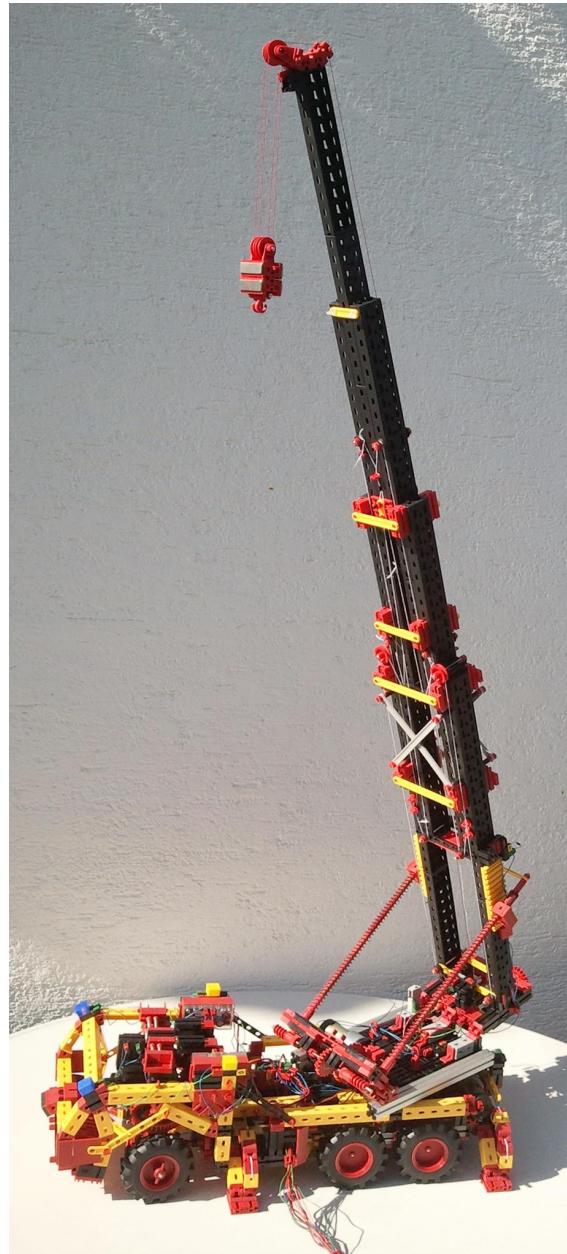


Abb. 7: Der Mobilkran mit Aufstellmechanik und Teleskoparm

Beispiel 2: Seilzug zum Heben eines Greifarms

Eine ähnliche Anwendung ist der folgende Seilzug, der zwei Greifarm-Segmente anheben kann. Dazu ist ebenfalls eine erhebliche Kraft erforderlich, die auf einfachste Art gebildet wird, indem die Seile direkt auf die Rast-Achs-Stummel der Mini-Aufsteckgetriebe geknotet werden. Der kleine Schlitz in den Rastachs-Enden bietet eine willkommene

Aufnahme für den Faden – so benötigen wir nicht einmal Klemmringe dazu:

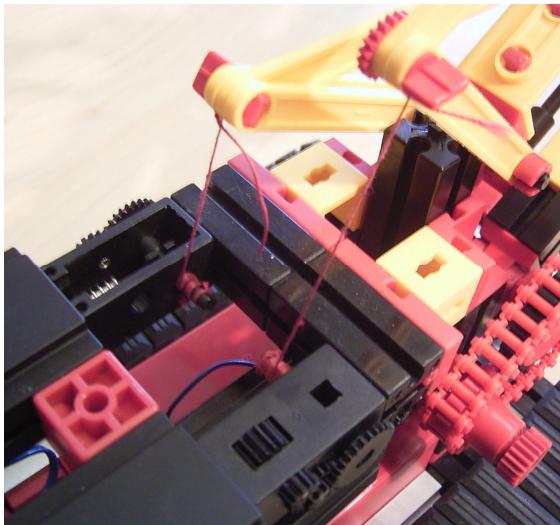


Abb. 8: Kräftige Seilzüge mit minimalem Aufwand

Durch diese Bauart können die beiden kleinen Motoren völlig ohne weitere Getriebe mit großer Kraft Hebelarme betätigen:

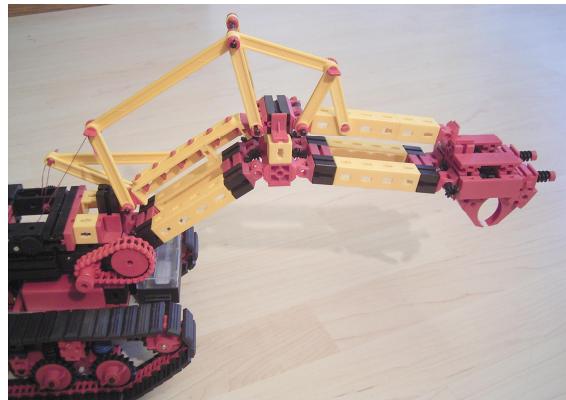


Abb. 9: Angehobene Armsegmente

3D-Modellentwurf

Erstellung von Bauteilen für den ft-Designer

Johannes Visser

Der [ft-Designer](#) von Michael Samek ([Demoversion](#)) erfreut sich großer Beliebtheit bei der Dokumentation eigener fischertechnik-Modelle. Der eine oder andere Nutzer ist vermutlich schon darüber gestolpert, dass er ein selteneres Bauteil für sein Modell benötigte, das in der (sehr umfangreichen) Bauteilbibliothek fehlt. Was tun, wenn auch ein Bauteilupdate nicht hilft? Entweder Michael Samek bitten, das Bauteil zu ergänzen – oder: Selber machen!

In diesem Beitrag wird vorgestellt, wie man die Bauteilbibliothek des ft-Designers selbst um ein benötigtes, aber in der Bauteilbibliothek nicht enthaltenes Bauteil ergänzen kann.

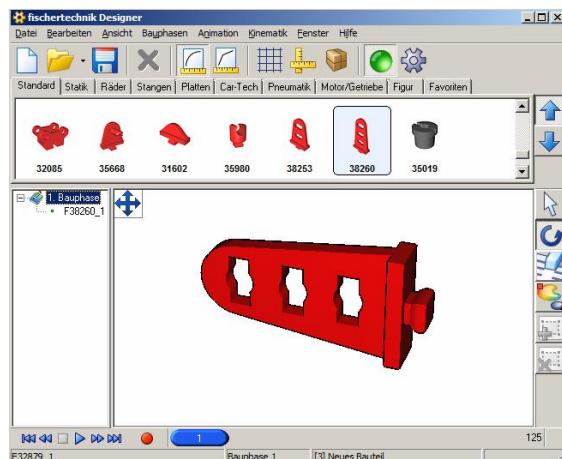


Abb. 1: ft-Designer von M. Samek (v1.1.0)

Dazu sind im Wesentlichen zwei Schritte erforderlich: Die Entwicklung der 2D- und 3D-Grafikdateien mit Hilfe einer geeigneten CAD-Konstruktions-Software und die Festlegung der Eigenschaften des Bauteils für die Nutzung im ft-Designer (Anbaustellen, Ursprungskoordinaten, Rotations-eigenschaften und Bauteilfarbe).

Der „Import“ in den ft-Designer erfolgt anschließend durch die Speicherung der erzeugten Dateien in der Verzeichnisstruktur des ft-Designers.

Struktur des ft-Designers

Der ft-Designer baut auf einer Orderstruktur auf, in der die Bauteile nach ihrer Artikelnummer abgelegt sind.



Abb. 2: Dateistruktur des FT-Designers

Schauen wir uns das an der Kupplung 38260 aus Abb. 1 an. Zuerst einmal befindet sich im Ordner icon ein 64x64-Pixel-Bild des Bauteils im Bitmap-Format. Das wird für den Katalog verwendet, aus dem man die Teile beim Konstruieren auswählt.

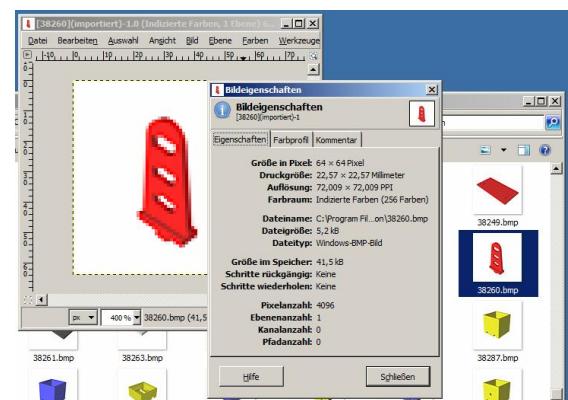


Abb. 3: Bild-Eigenschaften von „icon“, ermittelt mit Gnu Image Manipulation ([GIMP](#))

In dem Unterordner print ist das Bild nochmal in besserer Auflösung (400x400 Pixel) im JPG-Format abgelegt. Dieses Bild wird benötigt, wenn man für eine Anleitung eine Teile-Liste erzeugt.



Abb. 4: Bauteile einer Bauphase ausgeben

Im Ordner def sind spezielle Definitionsdateien abgelegt, aus denen das Programm die Informationen für die Bauteile gewinnt: Drehung, Koordinaten, Andockstellen etc. Auf die Struktur dieser Datei wird später noch im Detail eingegangen.

3D-Daten im STL-Format

In den Ordnern LoRes und HiRes befinden sich die 3D-Modelle von den Bauteilen vornehmlich im STL-Format ([Surface Tessellation Language](#)) [1], das sich als Standard im Bereich *Rapid Prototyping* (Stereolithografie/3D-Druck) etabliert hat. Hier werden dreidimensionale Objekte durch ihre Oberfläche definiert.

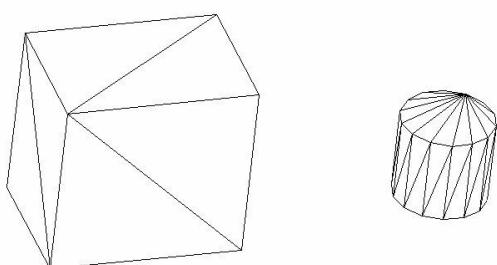


Abb. 5: Würfel und Zylinder in STL-Darstellung

Dabei wird von dem Ansatz ausgegangen, dass man alle Flächen (bei definierter

Granularität) durch eine Kombination von Dreiecken nachbilden kann. Jedes Dreieck wird durch seine drei Eckpunkte mit dessen Koordinaten im Raum beschrieben.

Die Reihenfolge der Punkte gibt vor, wo Außen und wo Innen ist. Umläuft man entlang der Punkte im Uhrzeigersinn, befindet man sich innen. Die Elektrotechniker unter uns werden es kennen: die Rechte-Hand-Regel.

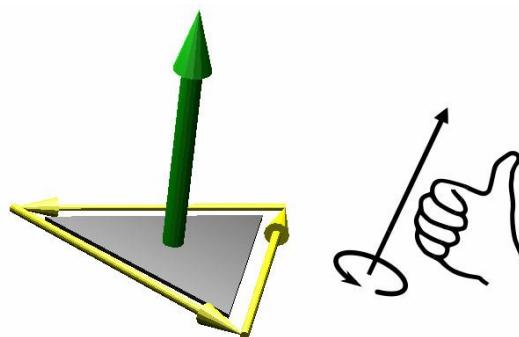


Abb. 6: Rechte-Faust-Regel.
Pfeil zeigt nach außen [2]

Aus historischen Gründen unterscheidet der ft-Designer noch zwischen Low- und High-Resolution mit zwei Sätzen von 3D-Daten in den gleichnamigen Ordnern. Bei begrenzter Rechenkapazität kann man auf eine niedrige Auflösung der Modelle zurückgreifen. Die Erfahrung mit modernen Rechnern zeigt, dass man auf die Low-Resolution-Variante verzichten kann: Die Rechenleistung ist heute in den meisten Fällen mehr als ausreichend. Aus diesem Grund kann man ein selbst erstelltes 3D-Modell sowohl in den LoRes- als auch in den HiRes-Ordner kopieren.

Im HiRes-Ordner befinden sich außerdem Daten mit der Endung *.NRM. Dahinter verbergen sich die so genannten Normalvektoren (Pfeile), die die Außenseite einer Facette definieren. Diese Datei ist nicht zwingend nötig – das Programm läuft auch ohne sie, wenn die STL-Dateien korrekt erzeugt worden sind.

An dieser Stelle sei angemerkt, dass manche 3D-Konstruktionsprogramme

etwas nachlässig mit der STL-Datei-Erzeugung umgehen. Wenn die Facetten nicht korrekt erzeugt sind, kann die Darstellung nicht erkennen, wo Innen und wo Außen ist, und das Objekt wird wie „von Innen nach Außen gekrempelt“ dargestellt. Auf dieses Problem wird später noch eingegangen.

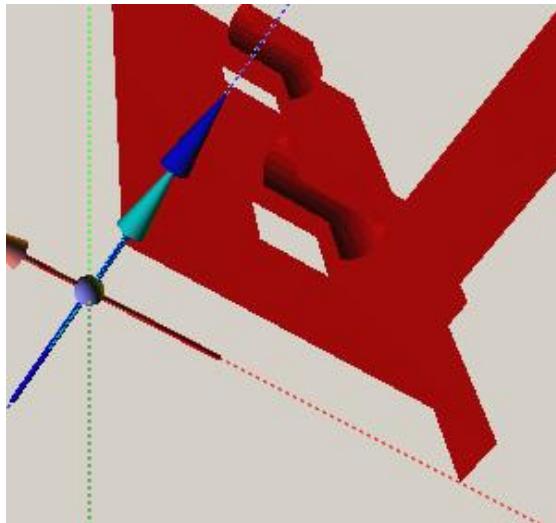


Abb. 7: fehlerhaftes STL

3D-Konstruktion des Bauteils

Im Folgenden werden wir darauf eingehen, wie man 3D-Objekte erstellen kann. Programme zur Erstellung solcher Objekte gibt es viele, allerdings nur wenige, die kostenlos (mit funktionalen Einschränkungen) verfügbar oder preiswert sind. Einige Programme hat Harald Steinhaus in [ft:pedia 2/2013](#) schon im Zusammenhang mit dem 3D-Druck von Bauteilen vorgestellt [3].

IMSI TurboCAD

[IMSI TurboCAD](#) ist eine für den Preis sehr leistungsfähige 2D-/3D-Konstruktionssoftware. Für die aktuelle Version muss man mit Kosten von ca. 750 € rechnen. Aber es gibt Anbieter, die ältere Releases anbieten; dort erhält man [TurboCAD Pro v16](#) auch schon für 100 € oder v17 für 160 €.

Innerhalb eines Release gibt es auch Fassungen mit reduziertem Funktions-

umfang. Die Einschränkungen im 3D-Bereich sind aber zu groß, daher empfiehlt sich der Kauf der Pro-Fassung.

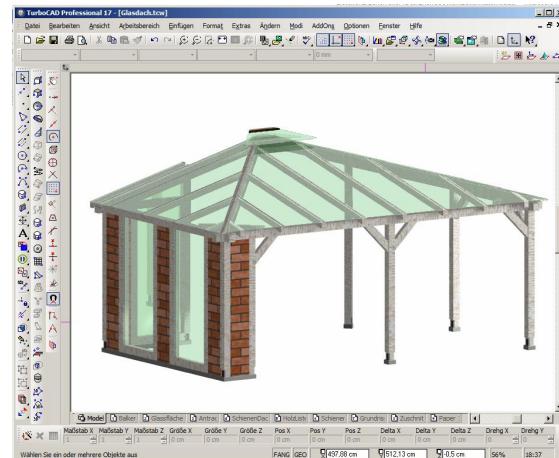


Abb. 8: TurboCAD am Beispiel der Terrassenüberdachung des Autors

TurboCAD bietet in diesem Zusammenhang den Vorteil, STL-Dateien laden zu können, so dass man existierende 3D-Objekte aus dem Fischertechnik-Designer als Vorlage verwenden kann.

PTC Creo Elements

[Creo Elements / Direct Modelling Express](#) gibt es für den Privatanwender kostenfrei, man muss allerdings seine Daten preisgeben und einen Account anlegen. Creo hat den entscheidenden Nachteil, dass keine STL-Objekte importiert werden können. Somit muss man bei der Konstruktion *from the scratch* beginnen.

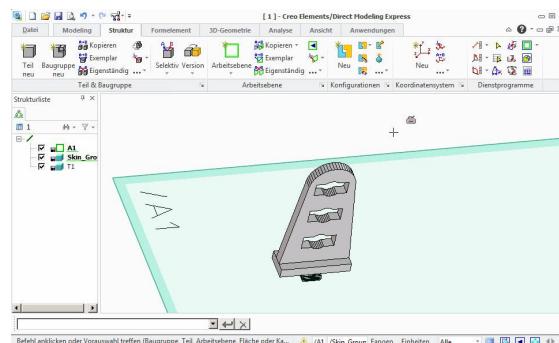


Abb. 9: PTC Creo

In diesem Beispiel wurde etwas gemogelt. Das fischertechnik-Bauteil wurde zuerst mit TurboCAD von STL geladen, in einen

Volumenkörper konvertiert, als *Initial Graphics Exchange Specification (IGES)* abgespeichert und dann mit Creo geladen. Grundsätzlich ist das ein komplexer Vorgang. STL ist eine Oberflächenbeschreibung; daraus einen eindeutigen Volumenkörper zu berechnen, ist schwierig, weshalb Creo diesen Import nicht bietet. Exportieren nach STL kann Creo aber sehr wohl.

Abgeleitetes Bauteil mit TurboCAD

Es sei als Beispiel ein neues Bauteil mit TurboCAD erstellt, das aus einem bestehenden Bauteil abgeleitet werden kann.

Als erstes muss man wissen, welche Artikelnummer das Bauteil hat, das man modellieren möchte. Hierbei hilft die Webseite von [fischerfriendswoman](#): Dieser Versandhandel hat eine ausgesprochen gut sortierte Liste aller fischertechnik-Teile mit Angabe der ft-Artikelnummer.



Abb. 10: Gondelhalter (31713)
bei FischerFriendsWoman

Um sich die Arbeit zu vereinfachen, sucht man zunächst ein Bauteil, das dem zu erstellenden am nächsten kommt. Hier bietet sich die Bauplatte mit zwei Zapfen 30x15 (38241) an. Sie entspricht im unteren Teil der Gondel. Dazu nimmt man die Lasche 15 (31667); davon kann man ein Stück „abschneiden“.

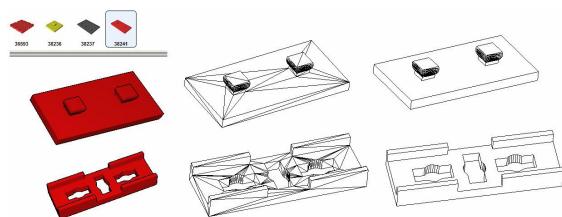


Abb. 11: Bauplatte (38241), Lasche (31667):
im ft-Designer, als STL und konvertiert zu
Volumenobjekt mit TurboCAD

Fangen wir mit der Lasche an. Durch Nachmessen an der Gondel wird ermittelt, dass die Aufhängung 13 mm lang ist. Um das aus der Lasche herausschneiden zu können, werden das mittlere Loch zugesetzt und die beiden Ränder aufgefüllt.

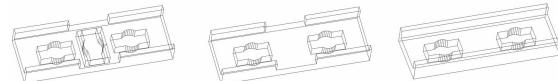


Abb. 12: Boolesche Operationen
an Lasche (31667)

TurboCAD bietet dazu so genannte Boolesche Operationen an; damit lassen sich Teile verschmelzen. So werden für den zweiten Schritt passende Quader in die Vertiefungen gelegt und mit dem Bauteil vereinigt.

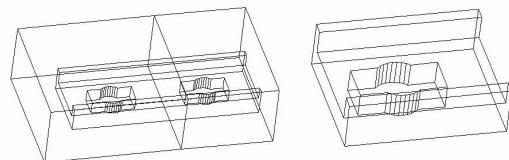


Abb. 13: Boolesche Operation ,Subtrahieren‘

Es wird nun ein umhüllender Quader definiert, der 13 mm der Lasche von rechts her umschließt. Der linke Teil des Quaders ist der Rest. Man kann nun eine Boolesche Subtraktion um den linken Quader machen oder eine Schnittmenge mit dem rechten Quader bilden. Übrig bleibt die benötigte Befestigungslasche.

Jetzt kann man die Gondel zusammenbauen. Die verbindende Strebe kann man leicht aus Quadern konstruieren. Die Lasche wird oben angefügt, die Teile werden verschmolzen, rot eingefärbt und dann gerendert.

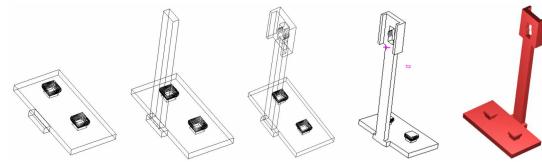


Abb. 14: Zusammenbau Gondel (31713)

Dateien für den ft-Designer

Aus dem gerenderten Bauteil erzeugt man nun ein Icon als [Bitmap](#) (64x64 Pixel), am besten mit nur 256 Farben, und eine JPEG-Datei (400x400 Pixel) [4]. Das Bitmap wird mit dem Dateinamen <Art.Nr.>.bmp in den Ordner icon kopiert, die JPEG-Datei als <Art.Nr.>.jpg in den Ordner icon/print. In unserem Beispiel sind das die Dateien 31713.bmp und 31713.jpg.

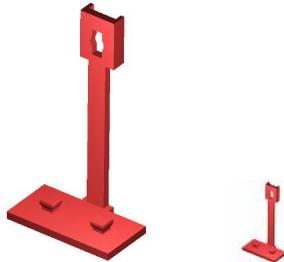


Abb. 15: Gondelhalter (31713) im 400x400 Pixel JPEG- und 64x64 Pixel BMP-Format

Die von dem 3D-Konstruktionsprogramm erzeugte Datei wird nach STL konvertiert und unter <Art.Nr.>.stl (in unserem Beispiel 31713.stl) in die Ordner HiRes und LoRes kopiert.

Fallstrick Artikelnummer

Der ft-Designer kann nur fünfstellige Artikelnummern verwalten; fischertechnik verwendet inzwischen aber schon sechsstellige. Will man ein solches neues Bauteil erzeugen, lässt man bei der Anlage des Dateinamens einfach die führende Ziffer weg – und hofft, dass die Nummer in der Bauteilbibliothek noch nicht vergeben ist.

Tatsächlich kann man sogar einen beliebigen, noch nicht verwendeten Dateinamen mit fünf Zeichen Länge wählen. Die echte Artikelnummer wird in der DEF-Datei in voller Länge angegeben (dazu mehr im folgenden Abschnitt).

Damit ist der erste Konstruktionsschritt abgeschlossen. Es fehlt nun noch die DEF-Datei; für deren Erstellung ist allerdings ein bisschen Fleißarbeit nötig.

Die DEF-Datei

Als zentrale Informationsdatei benötigt der ft-Designer eine DEF-Datei, in der Informationen abgelegt sind, wie

- Ursprungskoordinaten
- Andockstellen
- Farbe
- Rotationen

Michael Samek hat dafür das Tool DefMaker entwickelt, mit dem diese DEF-Datei erzeugt werden kann. Er hat es der ft-Community netterweise zur Verfügung gestellt: DefMaker kann im [Download-Bereich](#) herunter geladen werden.¹ Das Tool muss in den Programmordner des ft-Designers kopiert werden.

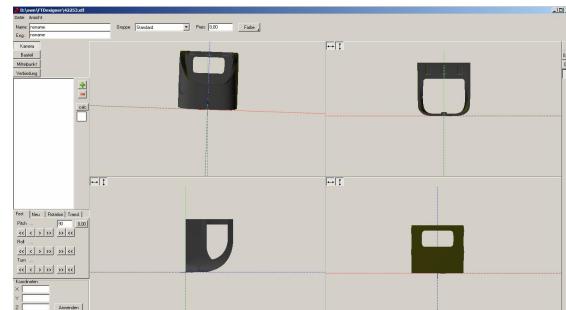


Abb. 16 DefMaker von Michael Samek

Erster Schritt: Datei/Neu

Nach dem Start von DefMaker wird man als erstes aufgefordert, die zugehörige STL-Datei anzugeben.

An dieser Stelle stolpert man bei Verwendung von TurboCAD häufig über die nachlässige Erzeugung der STL-Datei. In der 3D-Ansicht sieht es so aus, als ob das Objekt „umgerekempelt“ wäre: Man kann scheinbar durch jede Oberfläche hindurch sehen und blickt immer von Innen auf die

¹ Michael Samek weist ausdrücklich darauf hin, dass er dieses Tool „so wie es ist“ bereitstellt und keinerlei Garantie oder Haftung für eventuelle Schäden übernimmt, die bei der Anwendung entstehen.

dahinter liegende Oberfläche der anderen Seite.

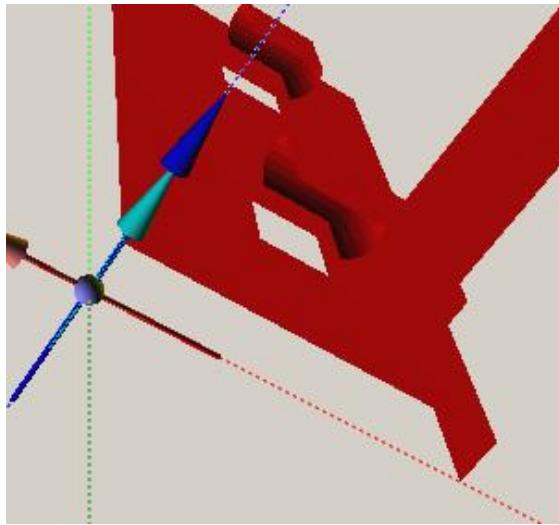


Abb. 17: STL-Fehler – Innen ist Außen

Nach Ansicht des Autors ist dieses auf das Fehlen der NRM-Datei (Normal-Vektor-Datei) zurückzuführen, die beschreibt, wo bei den Facetten Innen und wo Außen ist. Indem man die STL-Datei regelkonform erzeugt, kann man dieses Problem umgehen. TurboCAD bietet das allerdings nicht. Dazu muss man ein separates Tool verwenden.

Zweiter Schritt: Netfabb-Reparatur

[Netfabb-Studio](#) ist als Software für das *Rapid Prototyping* gedacht (3D-Drucken, Stereolithographie). Netfabb kann STL-Dateien korrigieren.

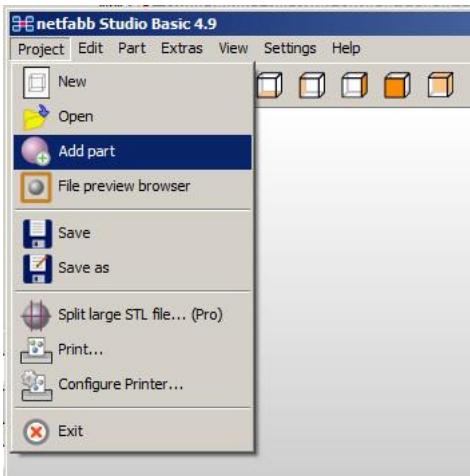


Abb. 18: NetFabb Studio Basic Einstieg

Mit eingeschränktem Funktionsumfang kann man das Programm [kostenfrei](#) herunterladen.

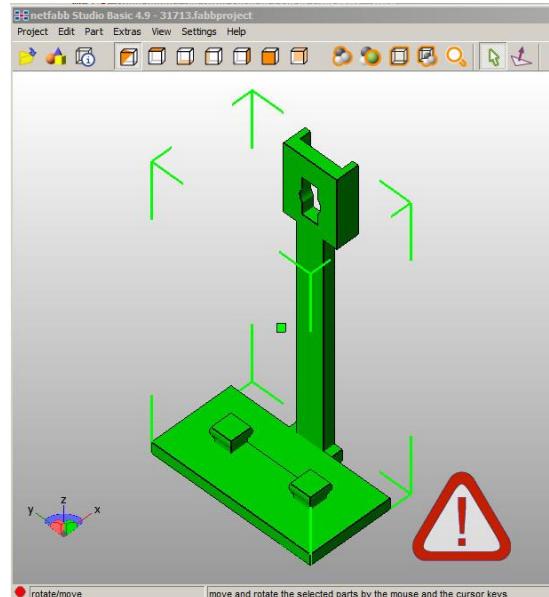


Abb. 19: Netfabb erkennt Fehler in STL-Datei

Nachdem die STL-Datei geladen ist, hat Netfabb den Fehler direkt erkannt. Über das Menü Extras/PartRepair wird das Bauteil automatisch repariert:

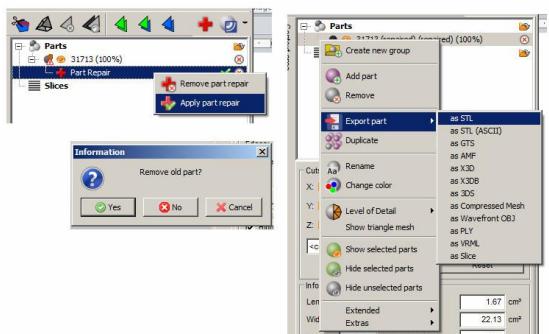


Abb. 20: Bauteil-Reparatur mit Netfabb

- Part/Apply Repair
- Remove old Part

Diesen Schritt muss man so lange wiederholen, bis alle Fehler beseitigt sind. Das können durchaus viele Zyklen sein. Anschließend kann die STL-Datei wieder exportiert werden. Nach erneutem Laden in DefMaker ist auch dieser glücklich.

- Part/Export to STL (binary)

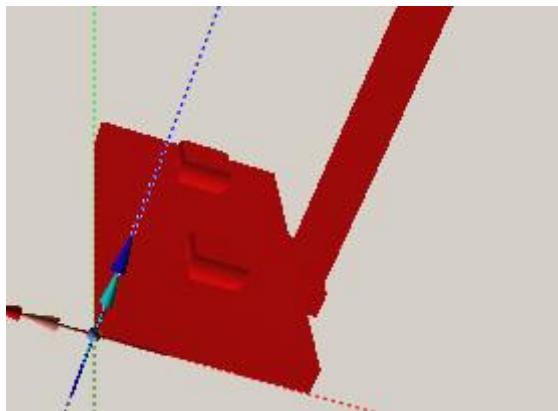


Abb. 21: Korrigierte STL-Datei im DefMaker

Dritter Schritt: DEF-Datei

Werfen wir einen Blick in eine DEF-Datei eines einfachen Bauteils am Beispiel der Bauplatte 15x30 mit zwei Zapfen (38241):

```
[Color]
Blue=0
Red=255
Green=0

[Common]
Name=Bauplatte 15x30, rot
Eng=Mounting Plate 15x30, red
Group=4
Prize=0,35
OrgNumber=38241
Printable=1

[Offset]
X=-7,50
Y=-7,50
Z=0,00

[Connections]
Count=4

[Conn1]
Name=Zapfen 1 Horizontal
Eng=Pin 1 horizontal
Typ=1
Thick=4,00
X=0
Y=0
Z=4
PF=90
RF=90
TF=0
PN=90
RN=0
TN=-90
```

Die Datei gliedert sich in folgende Sektionen (einfache Bauteile):

- [Color] Farbe des Bauteils
- [Common] Grundlegende Daten zum Bauteil
- [Offset] Wo ist im STL-Bild der Ursprung?
- [Connections] Zahl der Andockstellen
- [Conn1] Daten zum ersten Andock-Punkt; je nach Zahl der Andockstellen mehr davon

Grundsätzlich muss man zwei Arten von Bauteilen unterscheiden: Einfache und komplexe Bauteile. Einfache Bauteile sind aus einem Stück. Komplexere Teile bestehen aus einem „Mutter-Bauteil“ gefolgt von „Kindern“.

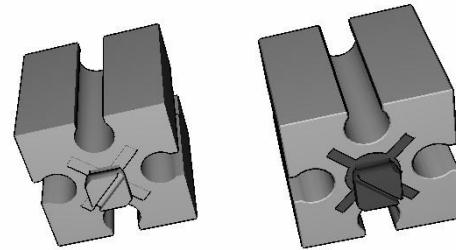


Abb. 22: Einfaches und komplexes Bauteil (mit „Kindern“)

Im rechten Beispiel ist das Mutter-Bauteil der Stein und der Zapfen das Kind. Komplexe Bauteile können auch unterschiedlich gefärbt sein. Im Falle des Bausteins 15 ist die Unterscheidung weniger interessant. Bei Bauteilen mit einem Gelenk ist das Konzept Mutter/Kind nötig, um z. B. den Knickwinkel ändern zu können.

Einfache Teile

Color

- Blue: Blauer Farbanteil des Bauteils im Wertebereich von 0..255
- Red: Roter Farbanteil des Bauteils im Wertebereich von 0..255

- Green: Grüner Farbanteil des Bauteils im Wertebereich von 0..255
- Texture: Dateiname der Textur des Bauteils (z. B. für Aufkleber). Dateiformat ist BMP und die Datei muss sich im Icon-Verzeichnis befinden. Der Dateiname ist dabei vollkommen frei zu vergeben.
- TxWidth: Breite der Textur in *mm*
- TxHeight: Höhe der Textur in *mm*

Common

- Name: Name, unter der das Bauteil erscheint
- Eng: Englischer Name (Sprachumschaltung)
- Group: Gruppe, in der das Bauteil erscheint
 - 0: Standard
 - 1: Statik
 - 2: Räder
 - 3: Stangen
 - 4: Platten
 - 5: CarTech
 - 6: Pneumatik
 - 7: Getriebe
 - 8: Figur
 - 9...18: entsprechende Gruppen der alten Teile
- Prize: Damit kann man eine Stückliste mit Preisen erzeugen
- OrgNumber: Hier wird die fischertechnik-Artikelnummer eingetragen. Das kann die ganze Nummer sein – wichtig insbesondere, wenn der Dateiname wegen der Beschränkung auf fünf Stellen gekürzt werden musste.
- Printable: Bauteile können so eingestellt werden, dass die Bauteile beim Erstellen der Stückliste nicht einzeln

erscheinen. Bei Mutter/Kind Teilen soll nur das Mutter-Teil erscheinen.

- Hide: Bauteile erscheinen nicht im Bauteil-Katalog (bei Kinder-Teilen, siehe Komplexe Teile)
- SortOrder: Sortier-Index innerhalb einer Gruppe. Die Bauteile innerhalb einer Gruppe werden anhand der SortOrder sortiert dargestellt. Ist die Eigenschaft nicht vorhanden, dann wird die 5-stellige Bauteilnummer als SortOrder verwendet.
- Kinerot: Vorgabe der Rotation im Kinematik-Modus. (0=X-Achse, 1=Y-Achse, 2=Z-Achse).
- Teeth: Bei Zahnrädern kann hier die Anzahl der Zähne angegeben werden. Dieser Wert wird bei der Anlage von Rotations-Bindungen verwendet.

Offset

Offset gibt an, wo in der STL-Datei der Nullpunkt ist.

Connections

Mit der Anzahl der Connections wird angegeben, wie viele Andock-Punkte das Bauteil hat. Im Folgenden muss für jeden Andock-Punkt eine neue Sektion [Connx] angelegt werden.

Connx

Beschreibung einer einzelnen Andock-Stelle:

- Name: Name des Andockpunkts
- Eng: Name in Englisch
- Typ: 0: Loch, 1: Zapfen
- Thick: Stärke (oder Dicke) der Verbundstelle. Hiermit wird sichergestellt, dass eine 4 *mm* Achse auch nur in ein 4 *mm* Loch gesetzt werden kann. Es können Bauteile nur dann verbunden werden, wenn die Stärke identisch ist.

- XYZ: Offset, Koordinate des Andock-Punkts in der STL-Datei
- PF, RF, TF, PN, RN, TN: geben an, wie ein andockendes Teil gedreht werden kann.

Komplexe Teile

Mutter/Kind-Bauteile bestehen aus mehreren Teilen, die unterschiedlich gefärbt sein können. In der DEF-Datei gibt es Verweise auf die Kinder-Teile:

```
[Childs]
Count=3

[Child1]
Number=C0041
[Child2]
Number=C0042
[Child3]
Number=C0043
```

Childs

- Count: Gibt an, wie viele Kinder es gibt. Es dürfen maximal 10 Kinder sein.

Childx

- Number: Hinweis auf den Dateinamen des Kind-Bauteils. Hier gilt wieder: maximal fünf Zeichen je Dateiname. Da das Kind-Bauteil keine eigene Artikelnummer hat, wählt man beliebige Namen.

Zu jedem Kind muss es eine STL-Datei und eine DEF-Datei geben. In den DEF-Dateien der Kinder wird

```
Printable = 0
```

gesetzt. Zusätzlich gibt es einen Eintrag

```
Hide = 1
```

der dafür sorgt, dass das Teil alleine nicht in dem Bauteilkatalog auftaucht, sondern nur in Kombination mit der Mutter.

Einstellung im DefMaker

Der DefMaker übernimmt die Erzeugung der DEF-Datei. Man kann prinzipiell die

DEF-Datei auch manuell editieren. Aber die Einstellung der Koordinaten und der Rotations-Eigenschaften ist dann sehr lästig.



Abb. 23: Grunddaten eingeben

Als erstes gibt man die Grunddaten ein: Name, englischer Name (tja, wie übersetzt man Gondelhalter?), Gruppe. Das Bauteil ordnen wir der Gruppe Statik zu.

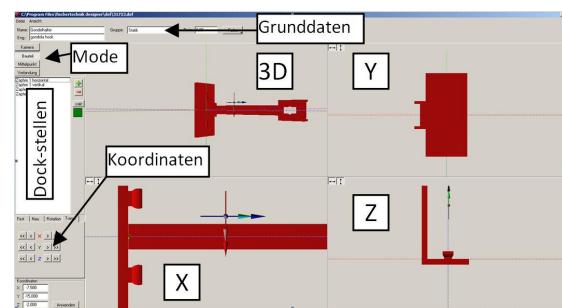


Abb. 24: Elemente des DefMaker

Im nächsten Schritt erfolgt die Einstellung diverser Koordinaten. Der DefMaker zeigt das Bauteil (wie im Schlauch-Manager des ft-Designers) in vier Fenstern: Im ersten wird das Bauteil in 3D angezeigt und kann gedreht werden. Die anderen Fenster zeigen je eine Ansicht (Seite/Oben/Vorne). Hier kann das Bauteil in einer Richtung verschoben werden.

Die Mode-Optionen bewirken, dass Schiebebewegungen in den Bauteil-Fenstern Unterschiedliches bewirken:

- Kamera: Man verschiebt das Bauteil so, dass man die Stelle sehen kann, die man bearbeiten möchte.
- Bauteil: Einstellung des Ursprungs (das Fadenkreuz aus den farbigen Linien)
- Mittelpunkt: Man verschiebt die kleine gelbe Kugel, die den Mittelpunkt des Bauteils markiert. Dieser Punkt wird beim Drehen des Bauteils im ft-Designer als Drehpunkt angeboten.

- Verbindungen: Damit verschiebt man das kleine Kreuz aus den sechs bunten Pfeilen. Dieses markiert den Arbeitspunkt für eine einzufügende Andock-Stelle.

Zur Festlegung des Ursprungs wird der Modus Bauteil gewählt und das Bauteil relativ zum Ursprung verschoben.

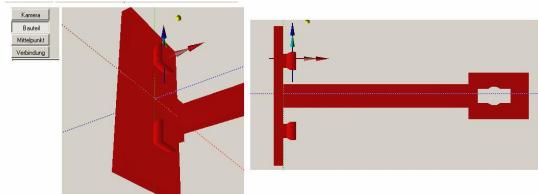


Abb. 25: Ursprung verschieben

Der Ursprung wird durch das Kreuz der dünnen Linien dargestellt. Im Beispiel wird der Ursprung zwischen die beiden Zapfen gesetzt. Diese Änderung hat Einfluss auf die Rubrik Offset in der DEF-Datei:

```
[Offset]
X=-7,5
Y=-15
Z=-2
```

Mit dem Modus Mittelpunkt kann dieser verschoben werden. Der Mittelpunkt wird als gelbe Kugel dargestellt. Der Mittelpunkt hat keinen Einfluss auf die DEF-Datei – er wird aber als Hilfsmittel zur Rotation von Verbindungen benötigt. Dies ist immer dann hilfreich, wenn Verbindungen kreisförmig angeordnet werden. Eine neue Verbindung kann dann um den Mittelpunkt rotieren. Gerade bei den Statik-Bögen ist ohne dieses Hilfsmittel eine manuelle Platzierung von Verbindungen kaum möglich.

Die kleinen Kontrollfelder (Abb. 26, rechts am Bildschirmrand) legen fest, in welchen Schritten das Bauteil, der Ursprung bzw. der Mittelpunkt mit der Maus verschoben wird.

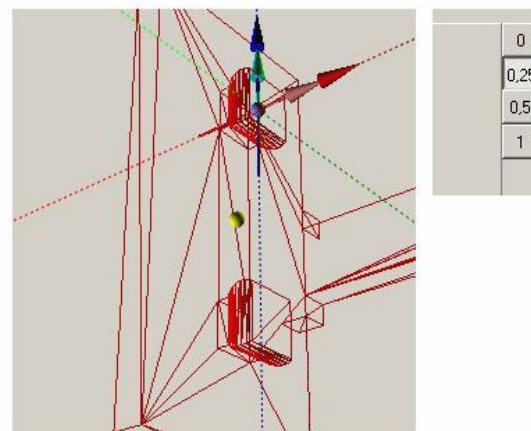


Abb. 26: Verschieben des Mittelpunkts, Schrittweiten-Einstellung

Bleibt die Festlegung der Andock-Stellen. Bezuglich der Verbindungen sind zu unterscheiden:

- Zapfen (Pins)
- Nuten (Groove) bzw. Löcher

Bei der Definition einer Andock-Stelle muss diese Eigenschaft angegeben werden. Für Verbindungen definiert man eine Dicke. Objekte passender Dicke können zusammengefügt werden, Zapfen in Nuten.

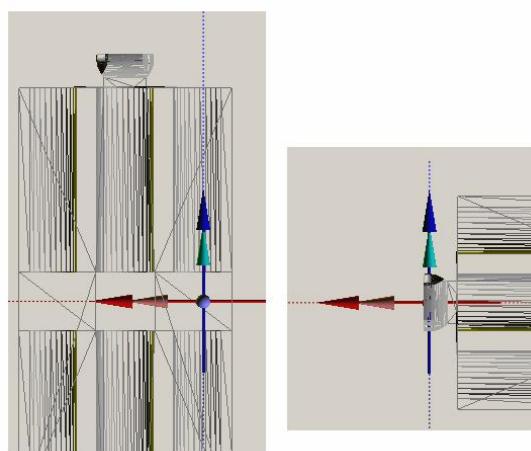


Abb. 27: Koordinaten, Drehung der Andock-Stellen

Die blaue Kugel in Abb. 27 ist die Koordinate der Andock-Stelle. Werden zwei Bauteile zusammengefügt, liegen die jeweiligen Kugeln übereinander. Im Beispiel ist für eine seitliche Nut des

Standard-Steins gewählt, dass die Koordinaten in der Mitte der Nut liegen. Da man in eine solche Nut auch z. B. eine Stange stecken kann, ist der Ort sinnvoll gewählt. Für den Standard-Stein liegt der Punkt also um die halbe Dicke einer Stange nach innen (2 mm). Demzufolge ist der Fügepunkt eines Zapfens 2 mm vom Rand des Steins definiert.

Die Richtungspfeile eines Fügepunkts bestehen aus zwei Gruppen von Pfeilen: roten und blauen bzw. pink- und türkisfarbenen. Die roten und blauen Pfeile beziehen sich auf das feste Bauteil, welches schon platziert ist; die beiden anderen Pfeile beziehen sich auf das neue Bauteil, das gerade platziert wird.

Der ft-Designer legt die Pfeil-Paare beim Einfügen übereinander. Wenn man sich bereits definierte Teile anguckt, erkennt man z. B. bei Nuten, dass der rote Pfeil immer in die Nut zeigt. Demzufolge müssen bei Zapfen die roten bzw. pink-farbenen Pfeile in Zapfenrichtung zeigen.

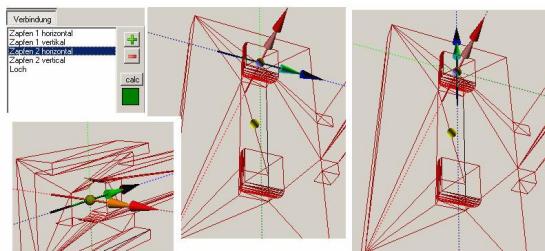


Abb. 28: Definition der Verbindungen

Für jedes Bauteil müssen die vorhandenen Verbindungen editiert werden. Für Zapfen erstellt man jeweils eine horizontale und eine vertikale. Wie oben dargestellt, unterscheiden sich die beiden durch die Ausrichtung der blauen bzw. türkisfarbenen Pfeile.

Beim Loch für den S-Riegel zeigt der rote Pfeil in die Richtung, in die der Riegel geschoben wird. Eine Unterscheidung zwischen vertikaler und horizontaler Verbindung ist hier nicht sinnvoll, da die Stelle über dem Kragen das Verdrehen verhindert.

Es muss immer nur die feste Verbindung festgelegt werden. Die andere Verbindung wird durch Druck auf die Calc-Schaltfläche automatisch berechnet. Wenn das darunter liegende Farbfeld grün erscheint, ist alles in Ordnung und die DEF-Datei kann gespeichert werden.

Manche Eigenschaften der DEF-Datei lassen sich (noch) nicht über den DefMaker editieren. Diese Eigenschaften müssen mit einem Text-Editor manuell nachgetragen werden:

- Prize
- Printable
- Hide
- Childs
- Childx

Ausprobieren

Nach Anlage eines neuen Bauteils muss der ft-Designer – wie nach jedem Bauteil-update – neu gestartet werden, damit die neuen Bauteile zur Verfügung stehen. So sollte nun auch der Gondelhalter in der Gruppe Statik erscheinen und sich platzieren und drehen lassen. Andere Bauteile müssen angedockt werden können, ohne dass Überschneidungen auftreten.

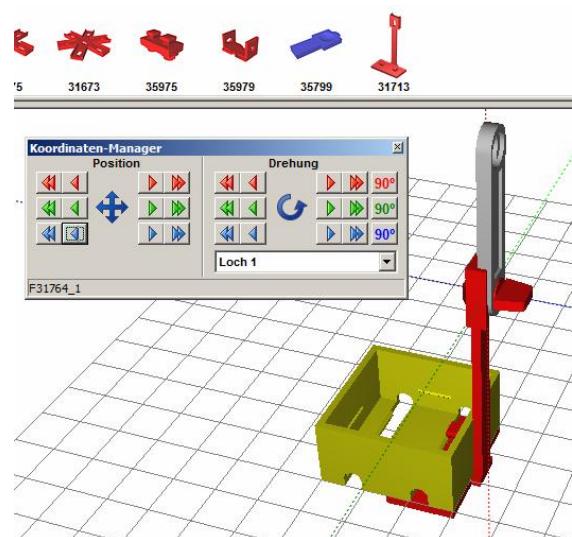


Abb. 29: Der Gondelhalter ist fertig

Ausblick

Ein eigenes Bauteil bringt Verantwortung: Man muss es allen zur Verfügung stellen, die ein mit diesem Bauteil erstelltes Modell im ft-Designer öffnen wollen, denn die Bauteile werden nicht mit den Modellen zusammen abgespeichert.

Wird das Bauteil irgendwann von Michael Samek für die offizielle Bibliothek bereit gestellt, lädt der ft-Designer es beim Teile-Update allerdings nicht, weil er der Meinung ist, er hätte es schon – man muss daher zuvor das selbst konstruierte Bauteil manuell löschen.

Damit die ft-Community bei der Erstellung von Bauteilen mitmachen kann, wäre Folgendes eine tolle Ergänzung des ft-Designer:

- Ein separater Speicherbereich für Custom-Parts, so dass sie sich nicht mit den offiziellen Teilen vermischen
- Ein Pack-And-Go-Assistent, um alle Dateien eines Teils zusammenzupacken

- Eine Upload-Möglichkeit auf dem ft-Designer-Server (das Bauteil wird dann geprüft und ggf. in die offizielle Bibliothek übernommen)

Bleibt noch der Wunsch an die Leser: Viel Spaß bei der Entwicklung eigener 3D-Bauteile für den ft-Designer!

Quellen

- [1] Wikipedia: [STL – Surface Tessellation Language](#).
- [2] Wikipedia: [Korkenzieherregel](#).
- [3] Harald Steinhaus: *Neue ft-Teile selbst gemacht: 3D-Druck*. [ft:pedia 2/2013](#), S. 4-10.
- [4] Joined Expert Picture Group: [JPEG File Interchange Format \(JFIF\)](#). Verlustbehaftete Grafik-Kompression nach ISO/IEC 10918-1.

Computing

I²C mit dem TX – Teil 6: GPS-Sensor

Dirk Fox

In unserer I²C-Serie haben wir bereits einige I²C-Komponenten vorgestellt, die sich seit dem Firmware-Update vom April 2012 an den TX anschließen und mit RoboPro nutzen lassen. In dieser Folge stellen wir einen etwas komplexeren Sensor vor – mit dem man in die faszinierende Welt der Navigation eintauchen kann.

Global Positioning System

Das *Global Positioning System* (GPS) ist ein weltweites Netzwerk von Navigationssatelliten, mit dessen Aufbau das US-amerikanische Verteidigungsministerium 1973 begann und das offiziell am 17.07.1995 in Betrieb genommen wurde.

Die mindestens 24 GPS-Satelliten, die in einer Höhe von etwa 20.200 km auf sechs unterschiedlichen Bahnen zweimal täglich die Erde umkreisen, senden dazu alle 30 Sekunden ihre Position und den Zeitpunkt der Positionsbestimmung auf zwei Frequenzen ($L_1 = 1.575,42 \text{ MHz}$ und $L_2 = 1.227,60 \text{ MHz}$) in einem festen Format aus (Abb. 1). Aus den Laufzeiten der Signale kann ein Empfänger seine genaue Position bestimmen (siehe nächster Abschnitt).

Das zivil nutzbare, unverschlüsselte Signal hat eine Datenrate von 50 bit/s. Ein zweites, nur für die militärische Nutzung vorgesehenes verschlüsseltes Signal wird mit der 10fachen Bitrate gesendet und

erlaubt eine entsprechend genauere Positionsbestimmung.

Anfänglich wurden die zivilen Signale (*Standard Positioning System*, SPS) mit einem Fehlerwert künstlich verschlechtert, sodass die Positionsbestimmung eine Ungenauigkeit von über 100 m besaß. Dieser *Selective Availability* genannte Mechanismus wurde am 02.05.2000 abgeschaltet. Seitdem liegt die Ungenauigkeit – abhängig von der Zahl der sichtbaren Satelliten und dem verwendeten Lokalisierungsverfahren – zu 95% unter 8 m [1]. Für einen etwaigen Krisenfall behalten sich die USA jedoch vor, das zivil nutzbare Signal zu stören. Auch deshalb haben die Sowjetunion (*Glonass*, nutzbar seit 2011), China (*Compass*, ab 2020) und die EU (*Galileo*, nicht vor 2020) eigene Satellitennavigationssysteme entwickelt. Galileo wird mit GPS kompatibel sein, für zivile Anwendungen eine höhere Bitrate verwenden und daher eine präzisere Positionsbestimmung (Ungenauigkeit < 4 m) ermöglichen [2].

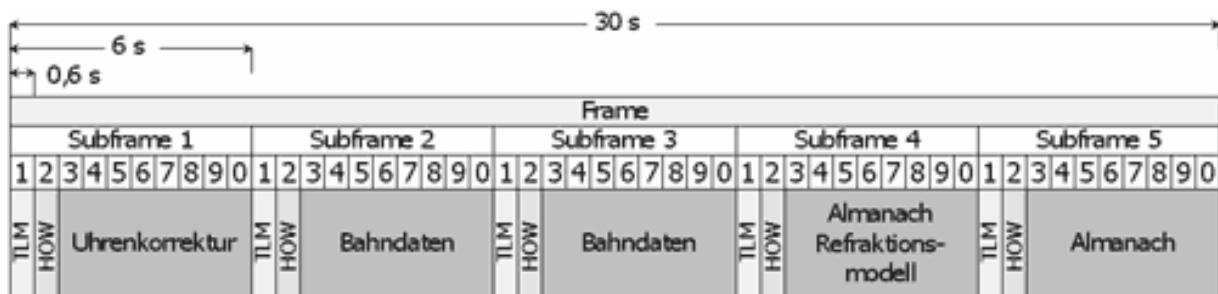


Abb. 1: Aufbau des GPS-Datensignals (Bild: <http://www.kowoma.de/gps> [3])

Positionsbestimmung

Ein GPS-Sensor kann aus empfangenen GPS-Signalen nicht nur seine Position (geografische Länge und Breite, Höhe) bestimmen, sondern auch seine Geschwindigkeit und Bewegungsrichtung und daraus den Abstand zu einem vorgegebenen Zielpunkt und dessen Richtung berechnen. Wie aber funktioniert das genau?

Die Position (Längengrad, Breitengrad und Höhe) bestimmt ein GPS-Sensor, indem er aus der Signallaufzeit den Abstand zur Position des sendenden Satelliten bestimmt. Dazu subtrahiert er die im GPS-Signal enthaltene Zeitangabe von der Signalempfangszeit. Das GPS-Signal eines einzelnen Satelliten liefert als möglichen Standort also die Punkte auf einer Kugel um den Satelliten – wenn die Höhe bekannt ist, auf einer Kreislinie, dem Schnitt der Kugel mit der Erdoberfläche.

Empfängt man das Signal eines zweiten Satelliten, reduziert sich die Lösungsmenge auf zwei Punkte (die Schnittpunkte der Kugeln um die beiden Satelliten), von denen nur einer auf der Erdoberfläche liegt. Somit genügen grundsätzlich zwei GPS-Signale zur Positionsbestimmung, sofern die Höhe der eigenen Position bekannt ist.

So weit die Theorie. In der Praxis ist das allerdings nicht ganz so einfach, denn die Ausbreitungsgeschwindigkeit des Signals erreicht fast Lichtgeschwindigkeit – knapp 300.000 km/s . Das Signal benötigt für die gut 20.000 km lange Strecke vom Satelliten bis zur Erde weniger als $0,07 \text{ s}$ – ein Messfehler von $0,001 \text{ s}$ führt so zu einem Fehler in der Positionsbestimmung von etwa 300 km . Damit die Zeitangabe des Satelliten hinreichend genau ist, verfügen diese über eine Atomuhr mit einer Abweichung von $1,2 \cdot 10^{-14}$ – das entspricht einem Fehler von einer Sekunde alle 2,64 Mio. Jahre [4].

GPS-Sensoren besitzen allerdings in der Regel keine ausreichend genaue und mit den GPS-Satelliten exakt synchronisierte Uhr. Für dieses Problem gibt es jedoch eine elegante Lösung: Man betrachtet die Empfangszeit als weitere Unbekannte – und wertet ein drittes Satellitensignal aus. Dann lassen sich Ort (geografische Länge und Breite) und Empfangszeitpunkt als Gleichungssystem mit drei Unbekannten ausdrücken. Als ein Nebenprodukt liefert der GPS-Sensor damit sogar die Atomzeit – ganz ohne DCF77-Zeitsignal [4].

Hat der GPS-Sensor drei Satellitensignale empfangen und daraus seine geographische Position bestimmt, zeigt er das als „2D-fix“ an. Dabei geht der GPS-Sensor von der Annahme aus, dass er sich auf Meereshöhe befindet. Will man auch die exakte Höhe (als vierte Unbekannte) mit Hilfe des GPS-Sensors bestimmen, muss dieser das Signal eines vierten Satelliten auswerten – und damit ein Gleichungssystem mit vier Unbekannten lösen. Eine solche Positionsbestimmung (Länge, Breite, Höhe) wird vom Sensor als „3D-fix“ angezeigt. Damit eine 3D-Positionsbestimmung möglichst überall gelingt, wurden die Umlaufbahnen der GPS-Satelliten so gewählt, dass an jeder Position der Erde Signale von mindestens vier verschiedenen Satelliten empfangen werden können.

In der Praxis wird die Genauigkeit der Positionsbestimmung außerdem dadurch beeinträchtigt, dass sich die Satelliten selbst mit einer großen Eigengeschwindigkeit (fast 14.000 km/h) fortbewegen. Auch ist die Ausbreitungsgeschwindigkeit der Signale in der Atmosphäre kleiner als im Vakuum, erreicht also nicht genau Lichtgeschwindigkeit. Schließlich verfälschen oder überlagern Reflektionen z. B. von Gebäuden einzelne Signale. Und nicht zuletzt hängt die Genauigkeit davon ab, wie exakt der Sensor einen Flankenwechsel in der Codefolge identifizieren kann.

Moderne Empfänger berücksichtigen diese Effekte und erreichen bei der Positionsbestimmung eine Genauigkeit von unter 3 m. Die Präzision kann – je nach Ort (Tal oder Berg) und Empfangsqualität (Stadt oder Land) durch die Berücksichtigung zusätzlicher Satellitensignale oder die Verwendung von *Differential GPS* (DGPS) auf unter 1 m verbessert werden. Weitere Optimierungen sind durch die Nutzung anderer Sensoren (Höhen- oder Geschwindigkeitsmesser) und einen Abgleich mit terrestrischen Referenzsignalen möglich (bis zu 1 cm).

Mit wiederholten Positionsbestimmungen kann ein bewegter GPS-Sensor auch seine Geschwindigkeit (*Ground Speed*) und -richtung (*Ground Course*) aus dem Abstand der jeweils letzten Positionen sowie der dazwischen verstrichenen Zeit berechnen.

GPS-Sensoren

Inzwischen gibt es kostengünstige GPS-Sensoren, mit denen sich GPS-Signale auch im Modellbau nutzen lassen. Ein solcher GPS-Sensor ist der GTPA010 von [GlobalTop Technology Inc.](#), der mit 5 V Betriebsspannung arbeitet und eine sehr hohe Empfindlichkeit von -165 dBm besitzt. Die Update-Rate liegt bei beachtlichen 10 Hz. Dabei ist der Sensor sparsam – er verbraucht 37 mA im *Tracking Mode* (Standortverfolgung) und 48 mA im *Acquisition Mode* (Standortbestimmung).

GPS-Sensoren übertragen die Daten in der Regel seriell (mit unterschiedlicher Baudrate) nach dem Standard NMEA 0183 [5] der *National Marine Electronics Association* (NMEA). Um einen solchen GPS-Sensor an unserem TX Controller nutzen zu können, ist eine Umsetzung des seriellen Signals in das I²C-Protokoll erforderlich. Derzeit sind mir drei *Breakout Boards* bekannt, die diese Umsetzung vornehmen und den GTPA010 um eine I²C-Schnittstelle erweitern:

- der [I²C GPS Stick](#) von DSS Cirquits,
- der [PA6C](#) von DroTec und
- der [Navigatron](#) von Flytron.

Die folgende Darstellung bezieht sich auf das Flytron-Board Navigatron v2 (Abb. 2), das für knapp 50 € erhältlich ist und die folgenden technischen Daten besitzt:

- 3,3-5 V Betriebsspannung (und daher ohne *Level Shifting* direkt am I²C-Ausgang des TX anschließbar),
- Genauigkeit der Positionsbestimmung (unter idealen Bedingungen) unter 3 m
- *Cold Start* (Einschalten) unter 35 s, *Warm Start* (Aktivierung des Sensors) unter 34 s und *Hot Start* unter 1 s
- Unterstützung von DGPS
- Unterstützung des I²C *Fast Mode* (400 kB/s) [6]

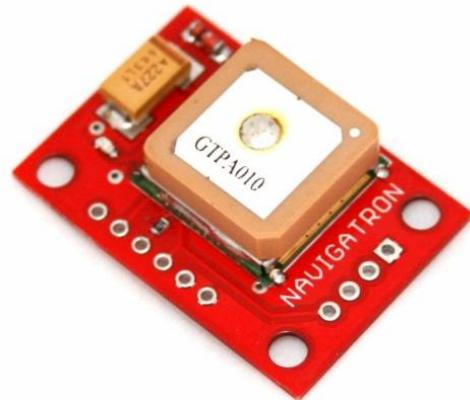


Abb. 2: Navigatron v2 (Flytron), ca. 50 €

Das 240 Byte große Register kann bis zu 16 Wegpunkte speichern und merkt sich dank eines Kondensators gespeicherte Daten auch im ausgeschalteten Zustand mindestens eine Woche lang.

Das Flytron-Board zeigt außerdem den Status an: Eine blinkende rote LED signalisiert den *Aquisition Mode* (unvollständige Positionsbestimmung). Leuchtet die rote LED, ist die Position bestimmt. Blinkt eine blaue LED dabei 2x, liegt ein 2D-fix vor, blinkt sie 3x, ist es ein 3D-fix.

Die I²C-Schnittstelle

Die I²C-Adresse des GPS-Boards ist 0x20 (= 32). Über diese Adresse können die Register des Sensors (Tab. 1) ausgelesen, einzelne auch beschrieben (*write*) werden.

Register	Inhalt
0x00	<i>Status Register</i>
0x01	<i>Command Register (write)</i>
0x02	<i>Waypoint Register</i>
0x03	<i>Firmware Version</i>
0x07	<i>Current Position</i>
0x13	<i>Distance to Waypoint [cm]</i>
0x17	<i>Bearing towards WP [1/1000 Grad]</i>
0x19	<i>Crosstrack Corrected Bearing towards WP [1/1000 Grad]</i>
0x1B	<i>Bearing from home [1/1000 Grad]</i>
0x1D	<i>Distance from home [m]</i>
0x1F	<i>Ground Speed * 100 [m/s]</i>
0x21	<i>Altitude [m]</i>
0x23	<i>Ground Course [1/10 Grad]</i>
0x27	<i>UTC Time (read)</i>
0x31	<i>Radius of WP [cm] (write)</i>
0x33	<i>Navigation Flags (write)</i>
0x3F	<i>Waypoint 0 (read/write)</i>
0x4A	<i>Waypoint 1 (read/write)</i>
...	...
0xE4	<i>Waypoint 15 (read/write)</i>

Tab. 1: Register

Da das Board für den Einsatz in Multicoptern entwickelt wurde, sind einige der vom Sensor berechneten und über die Register auslesbaren Werte sehr spezifisch und nur für die Navigation eines Flugmodells von Bedeutung. Sie fehlen in der Tabelle und

werden auch im Folgenden nicht erläutert, um die Darstellung nicht unnötig kompliziert zu machen – schließlich sind ft-Motoren zu schwach und zu schwer für ein Flugmodell.

Das Status-Register

Das *Status Register* 0x00 liefert einen Byte-Wert zurück, der in den unteren vier Bits vier Zustandswerte enthält (Tab. 2). Sie geben an, ob neue Daten empfangen wurden, ob eine 2D- (Längen- und Breitengrad, 2D-fix) oder 3D-Positionsbestimmung (Höhe, 3D-fix) gelungen ist und ob der im *Waypoint Register* 0x02 angegebene aktive Wegpunkt erreicht wurde.

Status	Bedeutung
0x01	Neue Daten empfangen
0x02	2D-fix (Positionsbestimmung mit 3 Satelliten)
0x04	3D-fix (Positionsbestimmung mit min. 4 Satelliten)
0x08	<i>Active Waypoint</i> erreicht

Tab. 2: Werte des Status Registers

Konnte eine Positionsbestimmung vorgenommen werden, gibt das Register in den vier höchstwertigen Bits die Zahl der für die Positionsbestimmung genutzten Satelliten an – bei einem 2D-fix müssen es mindestens drei, bei einem 3D-fix mindestens vier sein. Die blaue LED blinkt dabei entsprechend oft.

Über das Register 0x31 kann der Radius eines „Zielkreises“ um den im *Waypoint Register* 0x02 angegebenen aktiven Wegpunkt in cm festgelegt werden, bei dessen Erreichen das Erreichen des Wegpunkts in Bit 4 des *Status Registers* (0x08) signalisiert wird.

Als Default enthält das Register den Wert 200 – ein angesteuerter Wegpunkt gilt damit innerhalb eines Kreises von 4 m Durchmesser als erreicht.

Das Befehlsregister

Mit dem *Command Register* 0x01 wird die Navigation des GPS-Sensors gesteuert. So kann die Navigation zu einem angegebenen Wegpunkt gestartet oder angehalten werden; bei einem Navigations-Stopp werden alle aktuellen Positionsregister auf den Wert Null gesetzt. Außerdem kann die aktuelle Position als Wegpunkt (*Waypoint*) gespeichert werden.

Byte	Kommando
0x01	aktuelle Position halten
0x02	Navigation zu aktivem Wegpunkt starten
0x03	aktuelle Position in aktivem Wegpunkt kopieren
0x06	Navigation anhalten (Positionswerte := 0)

Tab. 3: Relevante Befehle des Command Registers

Bei den Kommandos 0x02 und 0x03 wird in den oberen vier Bits des Kommando-Bytes die Wegpunkt-Nummer (0-15) angegeben.

Das Versionsregister

Das Versionsregister 0x03 gibt den aktuellen Versionsstand der Firmware des I²C-GPS-Boards zurück (derzeit Version 21).

Die Positionsregister

Mit dem Register 0x07 beginnen die vom GPS-Modul berechneten Angaben zur aktuellen Position. Da sind zunächst die *GPS Location Register*, die die geografische Breite und Länge der aktuellen Position in je zwei 32 bit langen Integer-Werten angeben. Diese Werte liegen nur vor, wenn das *Status Register* wenigstens einen 2D-fix anzeigt. Mit dem Kommando *Stop Navigation* werden sie gelöscht.

Das *Distance to Waypoint Register* 0x13 liefert den Abstand zum aktiven Wegpunkt in cm als ganzzahligen 32 bit-Wert; es wird

gefolgt von einer 16 bit-Richtungsangabe zum aktiven Wegpunkt in 1/1000 Grad (*Bearing towards Waypoint*, 0x17). Eine erforderliche Richtungskorrektur (*Cross-track Corrected Bearing towards WP*) wird im Register 0x19 in 1/1000 Grad angegeben.

Die Register *Bearing from Home* (0x1B) und *Distance from Home* (0x1D) geben die Kompass-Richtung in 1/1000 Grad vom Startpunkt (Wegpunkt 0) zur aktuellen Position und den Abstand (Luftlinie) in m jeweils als 16 bit-Wert an.

Die Register *Ground Speed* (0x1F) und *Ground Course* (0x23) geben die Sensor-Geschwindigkeit in cm/s und die Kompass-Richtung des aktuellen Kurses in 1/10 Grad ebenfalls als 16 bit-Werte an.

Liegt ein 3D-fix vor, enthält Register 0x21 außerdem die Höhe in m (über Normal Null) als 16 bit-Wert; anderenfalls hat der Registerinhalt den Wert 0.

Die Satellitenzeit

Das Register *UTC Time* (0x27) liefert die ermittelte Satelliten-Atomuhrzeit – auf die 1/1000ste Sekunde genau als 32 bit-Wert.

Interpretiert als Dezimalzahl geben die letzten drei Stellen die Millisekunden an, die beiden Stellen davor die Sekunden, die nächsten beiden Stellen die Minuten und die beiden führenden Stellen die Stunden (0-23). Für die Ausgabe auf einem Display sind die Werte entsprechend umzurechnen.

Das Navigationsregister

In den *Navigation Flags* (Register 0x33) kann durch Setzen von Bit 6 (0x80) ein Filter aktiviert werden, der bewirkt, dass als Positionskoordinaten immer der Durchschnittswert der letzten fünf Positionsbestimmungen angezeigt wird (default). Damit werden Schwankungen bei der Positionsbestimmung abgemildert. Zugleich wird so allerdings auch die Aktualisierung der Zeitangabe verlangsamt (nur noch alle

ca. 17 Sekunden). Ist Bit 5 (0x40) gesetzt, wird zudem bei einer Positionsbestimmung im Ruhezustand das „Rauschen“ herausgefiltert (default).

Die Wegpunkt-Register

Das *Waypoint Register* 0x02 enthält einen Byte-Wert, dessen untere vier Bits die Nummer des aktuellen aktiven Wegpunkts enthalten. Die oberen vier Bits geben die Nummer des vorangegangenen Wegpunkts an.

Ab Register 0x3F können bis zu 16 Wegpunkte (*Waypoints*) gespeichert werden. Jeder Wegpunkt ist durch die in Tab. 4 aufgelisteten Angaben definiert: Breitengrad, Längengrad, Höhe und ein Flag-Byte, das die Festlegungen für das Register 0x33 (*Navigation Flags*) enthält. Insgesamt erfordert die Speicherung eines Wegpunkts damit also 11 Byte.

Byte	Inhalt
01-04	Breitengrad (int32)
05-08	Längengrad (int32)
09-10	Höhe (uint16)
11	<i>Navigation Flags</i>

Tab. 4: Parameter eines Wegpunkts

Die Wegpunktregister sind Schreib-/Leseregister. Sie können daher sowohl dafür genutzt werden, eine vorgegebene Strecke (z. B. für ein autonom steuerndes Fahrzeug) einzulesen und abzufahren, als auch, um Markierungspunkte einer abgelaufenen oder abgefahrener Strecke zu speichern (bspw. bei einer Wanderung oder Radtour).

In der Regel wird im Wegpunkt 0 der Startpunkt gespeichert. Damit kann ein Navigationsprogramm auch wieder zum Startpunkt zurückfinden (*Return to home*); der Sensor berechnet zudem ständig den Abstand zu diesem Ausgangspunkt.

Auswertung der GPS-Daten

Längen- und Breitengrad

Der GPS-Sender liefert die Positionsdaten nach dem globalen Referenzsystem WGS84 (*World Geodetic System 1984*). Zur Festlegung der geographischen Koordinaten wird die Erdoberfläche dabei in 360 Längen- und 180 Breitengrade² unterteilt, mit Greenwich (der Londoner Sternwarte) und dem Äquator als Nullmeridiane. Das WGS84 berücksichtigt zudem, dass die Erde keine Kugel ist, sondern ein Ellipsoid.

Der GPS-Sensor gibt die Geographische Länge und Breite als 32-bit-Wert mit einer Genauigkeit von 10^{-7} Grad zurück. In RoboPro müssen dazu die Werte byteweise ausgelesen und in eine Fließkommazahl (48 bit) umgewandelt werden.

Üblicherweise erfolgt die Angabe des Längen- und Breitengrads als Fließkommazahl in Grad. Diesen Wert erhält man, indem man den vom Sensor empfangenen Wert durch 10^7 teilt. Alternativ kann eine Position auch „klassisch“ in Grad, Minuten und Sekunden ausgedrückt werden. Für die Umrechnung in Minuten sind die Nachkommastellen der Gradangabe mit 60 zu multiplizieren; die Sekunden erhält man, indem man die verbleibenden Nachkommastellen mit 60 multipliziert.

Überprüfen lässt sich die vom GPS-Sensor empfangene Positionsangabe mit dem Dienst [Google Maps](#): Navigiert man seinen aktuellen Standort an, klickt mit der rechten Maustaste auf den Punkt und selektiert „Was gibt es hier?“, wird die Stelle mit einem grünen Pfeil markiert. Klickt man auf den Pfeil, zeigt Google Maps die Position sowohl in Fließkommadarstellung als auch in Grad, Minuten und Sekunden

² Die geographische Länge wird als „Lat“ (oder ϕ), die Breite als „Lon“ (oder λ) bezeichnet.

auf zwei Nachkommastellen genau an (siehe Abb. 3).

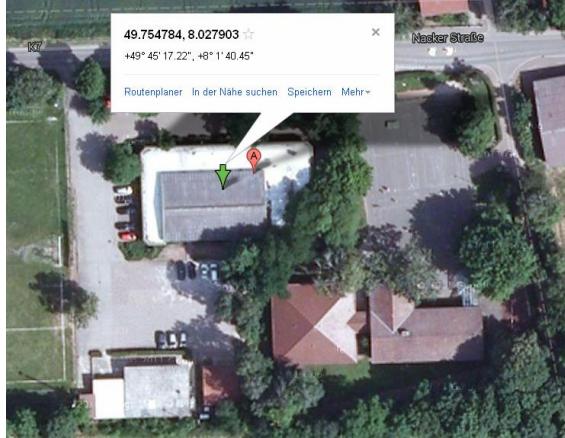


Abb. 3: GPS-Koordinaten der Mehrzweckhalle von Erbes-Büdesheim (Google Maps)

Stimmen diese Positionsangaben nicht mit den vom GPS-Sensor gelieferten Werten überein, kann man aus den Differenzen im Breiten- und Längengrad die Abweichung bestimmen. Ganz einfach geht das ebenfalls mit Google Maps: Man gibt die Koordinaten des Sensors in Grad, Minuten und Sekunden im Suchfeld ein und lässt sich den Abstand zum Standort anzeigen.

Aber es geht auch ohne Internet: Die Abweichung lässt sich berechnen. So entspricht der Abstand b zwischen zwei Breitengraden auf der Erdoberfläche je Breitenminute (näherungsweise, da die Erde keine exakte Kugel ist, sondern ein Elipsoid) etwa einer Seemeile³:

$$b = \frac{2\pi \cdot R}{360 \cdot 60} \approx 1,852 \text{ km}$$

Zwei Breitensekunden liegen damit auf der Erdoberfläche ca. 30,9 m voneinander entfernt.

Der Abstand zwischen zwei Längenminuten hängt vom Breitengrad ab: Am Äquator entspricht er ebenfalls etwa einer Seemeile (1,852 km), an den Polen ist er 0. Berechnen lässt sich der Abstand l nähe-

rungsweise aus dem Umfang des zum Breitengrad φ gehörigen Breitenkreises:

$$l \approx 1,852 \cdot \cos \varphi \text{ km}$$

In Deutschland liegt l zwischen 1,257 (Oberstdorf) und 1,061 km (Sylt) – und in Erbes-Büdesheim bei 1,1965 km (Längengrade: 19,94 m) [7].

Kann der Sensor die Signale von mindestens sechs Satelliten auswerten (selbst in der Stadt konnte ich bis zu acht Satellitensignale empfangen), lag der durchschnittliche Fehler bei meinen Tests unter 0,3 s im Breiten- und Längengrad, die mittlere Abweichung also unter 4 m.

GPS-Zeit

Die vom GPS-Board gelieferte Satellitenzeit wird aus (mindestens) drei empfangenen Satellitensignalen berechnet und entspricht der synchronisierten GPS-Satellitenzeit. Die Bestimmung erreicht eine Genauigkeit von bis zu 1 μsec ($= 10^{-6} \text{ sec}$).

Diese GPS-Zeit entsprach beim Start 1980 der koordinierten Weltzeit ([Coordinated Universal Time](#)) UTC [4, 8]. Seitdem weicht sie um einige Sekunden davon ab: Die Abweichung entsteht durch Schaltsekunden [9], die vom IERS ([International Earth Rotation and Reference Systems Service](#)) zum Ausgleich von Schwankungen der Erdrotationsgeschwindigkeit festgelegt werden. Derzeit (2013) beträgt die Abweichung 16 Sekunden; sie wird jedoch mit den Satellitensignalen (Abb. 1) übermittelt und vom GPS-Sensor bei der Bestimmung der Zeit berücksichtigt.

RoboPro-Treiber

Die Programmierung eines I²C-Treibers für den GPS-Sensor ist in RoboPro nun keine große Sache mehr. Auf eine Erläuterung der Implementierung verzichte ich daher an dieser Stelle – und lege euch stattdessen den [Treiber im Download-Bereich der ft:c](#) ans Herz.

³ In der Formel bezeichnet R den Erdradius, der im Mittel bei 6.368 km liegt.

Der Treiber umfasst ein kleines Programm, das den Standort (Länge, Breite, Höhe), die *Coordinated Universal Time* (UTC), die Geschwindigkeit und die Zahl der empfangenen Satellitensignale auf dem Display des TX anzeigt (Abb. 4).



Abb. 4: Positionsbestimmung mit dem TX (hier mit 7 Satellitensignalen)

Der TX als Navi

Mit dem I²C-Treiber lässt sich der TX nun mit wenigen RoboPro-Befehlen in ein Navi verwandeln. Die Programmierung überlasse ich euch als kleine ‚Fingerübung‘. Welche Programmschritte dafür benötigt werden, ist klar:

- Das RoboPro-Programm sollte zunächst eine Wegpunkte-Datei im csv-Format öffnen und die darin enthaltenen Punkte in die *Waypoint Register* 1-15 schreiben.
- Hat es dann eure Position bestimmt (3D-fix), sollte es diese Position als Startpunkt in *Waypoint Register* 0 abspeichern.
- Anschließend sollte es mit dem Befehl 0x02 die Navigation zum ersten Wegpunkt starten. Den Abstand zum Zielpunkt (in cm, Register 0x13) und die erforderliche Richtungskorrektur (in 1/1000 Grad, Register 0x19) solltet ihr euch auf dem TX-Display anzeigen lassen.
- Ist der Wegpunkt erreicht, sollte das auf dem Display angezeigt werden. Nun

solltet ihr zur Auswahl stellen, ob der nächste Wegpunkt angesteuert werden soll oder ihr zum Startpunkt zurückkehren möchtet.

Mit einem solchen kleinen Programm habt ihr ein ausgewachsenes Navi – das sich sogar bei Fahrradtouren oder Wanderungen einsetzen lässt.

Dafür benötigt ihr aber noch ein Verfahren, um ausgewählte Tracks in eine csv-Datei mit Wegpunkten zu überführen. GPS-Daten findet ihr zahlreich im Internet – so bietet der ADFC seinen Mitgliedern viele [Fahrradtouren als GPS-Track](#) an, und Regionen wie z. B. die Eifel publizieren auf einem eigenen [GPS-Portal](#) die GPS-Daten zu Wander- und Radtouren.

Die Tracks enthalten meist deutlich mehr als 15 Wegpunkte – das stellt aber kein Problem dar, wenn ihr die Wegpunkte abschnittsweise in die *Waypoint Register* laden lasst. Allerdings müsst ihr die aus dem Internet herunter geladenen GPS-Informationen zunächst noch in eine csv-Datei konvertieren.

Leider verwenden GPS-Dateien sehr unterschiedliche Formate. Ein verbreitetes GPS-Format ist GPX (*GPS Exchange Format*), das auf dem XML-Standard basiert [10]. Es ermöglicht die Angabe von Ortspunkten, Routen (Folge von Ortspunkten), die eine geplante Strecke beschreiben, und Tracks (Folge von Wegpunkten, die in Tracksegmente unterteilt werden können):

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<gpx version="1.1" creator="Ersteller der
Datei">
  <!-- Wegpunkte -->
  <wpt lat="xx.xxx" lon="yy.yyy">
    <!-- Attribute des Wegpunkts -->
  </wpt>
  <!-- weitere Wegpunkte -->
  <!-- Routenpunkte -->
  <rte>
    <rtept lat="xx.xxx" lon="yy.yyy">
      <!-- Attribute des Routenpunkts -->
    </rtept>
    <!-- weitere Routenpunkte -->
  </rte>
```

```

<!-- Tracks -->
<trk>
  <trkseg>
    <!-- Trackpunkte -->
    <trkpt lat="xx.xxx" lon="yy.yyy">
      <!-- Attribute des Trackpunkts -->
    </trkpt>
    <!-- weitere Trackpunkte -->
  </trkseg>
  <!-- weitere Track-Segmente -->
</trk>

</gpx>

```

Abb. 5: GPX-Darstellung von Wegpunkten

Aus einer solchen GPX-Datei lassen sich die Wegpunkt-Koordinaten mit einem kleinen Script leicht extrahieren (Suche nach „lat = ...“ und „lon = ...“) und in einer csv-Datei abspeichern. Liegen die Daten in einem anderen Format vor, hilft das kleine, kostenlose Konvertierungsprogramm [GPS-Babel](#): Es kann eine sehr große Zahl von Formaten in GPX „übersetzen“.

Natürlich geht es auch umgekehrt: Das „TX-Navi“ speichert während einer Wanderung oder Radtour auf Knopfdruck die jeweils aktuellen Positionsangaben; später werden sie ausgelesen, in eine GPX-Datei konvertiert und z. B. [in Google Earth importiert](#). Dafür muss man aber in der Regel mehr als 15 Wegpunkte speichern können – da führt kein Weg an einem größeren Speicher vorbei. Aber auch für dieses Problem gibt es eine I²C-Lösung – mehr in einem der nächsten I²C-Beiträge.

Jetzt aber viel Vergnügen mit eurem „Selbstbau-Navi“ – ich bin gespannt auf die ersten Touren-Berichte!

Quellen

- [1] Wikipedia: [Global Positioning System \(GPS\)](#) und [GPS-Technik](#).
- [2] Wikipedia: [Galileo](#).
- [3] Anja Köhne, Michael Wössner: [Bestimmung der Position beim GPS-System](#). www.kowoma.de
- [4] Dirk Fox, Dirk Ottensmeyer: *Bau einer ft-Funkuhr*. [ft.pedia 3/2012](#), S. 4-10.
- [5] Anja Köhne, Michael Wössner: [NMEA-0183 Daten](#). www.kowoma.de
- [6] Dirk Fox: *I²C mit TX und Robo Pro – Teil 1: Grundlagen*. [ft.pedia 3/2012](#), S. 32-37.
- [7] Wikipedia: [Geographische Koordinaten](#).
- [8] Wikipedia: [GPS-Zeit](#).
- [9] Wikipedia: [Schaltsekunde](#).
- [10] Wikipedia: [GPS Exchange Format](#).

In eigener Sache

Gründung des ftc Modellbau e. V.

Stefan Falk

Nach einiger Vorbereitungszeit wurde am 31.08.2013 offiziell der „ftc Modellbau e. V.“ gegründet, der „Modellbau mit fischertechnik“ fördern will.

Seit dem Aufbau des ft-Community-Servers vor zwölf Jahren (2001) durch Markus Mack und Sven Engelke hatte Sven sämtliche anfallenden Kosten für das Serverhosting übernommen. Ohne die beiden gäbe es keine „ft-Community“: keinen Bilderpool, kein Wiki, keinen Chat und auch kein Forum. Danke also an Sven und Markus!

Um die Finanzierung auf „offizielle Füße“ zu stellen, sodass nicht mehr eine einzelne Person alles bezahlt, beschlossen wir die Gründung eines „ftc Modellbau e. V.“. Nach langer Vorbereitung – das deutsche Vereinsrecht stellt da durchaus Anforderungen – und ausführlicher Diskussion über die Gestaltung, Größe und den genauen Zweck des Vereins war es dann am letzten August-Tag 2013 soweit: Die Gründungsmitglieder trafen sich bei Frankfurt/Main, wo ein Notar (an einem Samstag!) dazu stieß und die notwendigen Formalitäten erledigte. Im Vorfeld hatte sich Martin Westphal intensiv in das Prozedere zur Vereinsgründung eingearbeitet, eine Satzung ausgearbeitet, dem Finanzamt zur Prüfung vorgelegt und vieles mehr. Dafür sei Martin an dieser Stelle herzlich gedankt.

Da der zunächst einzige praktische Zweck des Vereins die Finanzierung des neuen ftc-Servers ist (den Christian Pütter und André Mohr bereits aufbauen), wurde bewusst nur ein Minimum an büro-

kratischem Aufwand getrieben. Nur die zur Eintragung eines Vereins notwendige Mindestanzahl an Mitgliedern wurde angestrebt. Alle anderen Ideen und Pläne können später folgen. Deshalb sieht der Verein derzeit personell so aus:

- Der Vorstand besteht aus dem ersten Vorsitzenden, Martin Westphal, dem Kassenwart, Frederik Vormann, sowie dem Schriftführer, Stefan Falk.
- Stimmberechtigte Beisitzer sind Sven Engelke und Christian Pütter.
- Als Kassenprüfer fungieren Severin Both und Thomas Brestrich.
- Weitere Mitglieder sind Thomas Kaiser, Bernhard Lehner, André Mohr (sobald volljährig) und Marius Seider.



Gruppenbild mit Laune: Die Gründungsmitglieder auf einen Blick

Über alles Weitere werden wir natürlich im [ftc-Forum](#) informieren.



Prater von Markus Wolf auf der Convention 2012 (Foto: Dirk Fox)