

Editorial

Das Geheimnis des Lernens

Dirk Fox, Stefan Falk

Die Neurowissenschaften – vulgo: Hirnforschung – halten immer wieder für einen Laien überraschende Erkenntnisse bereit. Viele davon räumen mit verbreiteten Vorstellungen auf – beispielsweise der, dass das Gehirn ab dem jungen Erwachsenenalter „fest verdrahtet“ und nicht mehr „umprogrammierbar“ sei, oder dass es beim Lernen wie bei einem Muskel in erster Linie auf das „Training“ des Gehirns ankomme. Wusstet ihr, dass die Zahl der Neuronen im Gehirn in den ersten Monaten nach der Geburt am größten ist – und dann in nur zehn Jahren auf etwa die Hälfte sinkt? Dass die Konzentration auf ein konkretes Ziel das Denken praktisch „ausschaltet“ – und erst das Gegenteil, nämlich Ruhe und „Defokussierung“, Gehirntätigkeit überhaupt ermöglicht? Dass nicht das Lernen den Menschen mit zunehmendem Alter schwerer fällt, sondern vielmehr die Fähigkeit zur Begeisterung – durch die das Gehirn überhaupt erst in „Lernbereitschaft“ versetzt wird? Oder dass positive Lernerlebnisse (selbst der Glaube an deren Möglichkeit) bei Kindern nachweisbar zu signifikant besseren Lernergebnissen führen – während Angst eine „Lernlähmung“ bewirkt?

Kaum jemand vermittelt solche Erkenntnisse so zugespitzt und einprägsam wie [Prof. Dr. Hüther](#), Neurobiologe an der Universitätsklinik Göttingen, oder [Prof. Dr. Spitzer](#), Leiter der Psychiatrischen Universitätsklinik in Ulm.

Stimmen die Erkenntnisse der Hirnforschung, dann haben wir in Wirklichkeit kein Lern-, sondern ein *Begeisterungs-*

problem, wenn Kinder, Schüler und bejahrtere Kollegen oder Mitarbeiter sich mit der Aneignung neuen Wissens schwer tun, verstärkt durch einen häufig angeknacksten Glauben an die eigene Lern- und Lösungskompetenz. Zuversicht, Selbstvertrauen und Begeisterung stehen aber meist nicht im Zentrum der Lernwirklichkeit – dabei sind sie die wesentlichen Voraussetzungen für erfolgreiches Lernen.

Kennen wir das nicht alle auch von uns selbst? Wie leicht fällt uns das Lernen, wenn wir uns mit Hingabe und Begeisterung auf eine Problemlösung stürzen (wie war das gleich mit eurem letzten fischertechnik-Modell, das erst nicht so recht gelingen wollte?). Und wie unbesiegtbar fühlen wir uns, wenn es uns gelingt, für ein zunächst unlösbar erscheinendes Problem einen eleganten Lösungsweg zu finden (wie bei einer pffiffigen Teilerlösung in einem unserer ft-Modelle).

Verkneifen wir uns an dieser Stelle einen Seitenhieb auf die „Schule von heute“ – damit würden wir auch den einen oder anderen engagierten Pädagogen zu Unrecht kritisieren. Machen wir uns statt dessen (einmal mehr?) klar, welch unglaubliches Lernpotential in fischertechnik schlummert: Herausforderungen, Begeisterung und Erfolgserlebnisse ohne Ende. Schöne Weihnachten – und ran an die Kästen!

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

Das Geheimnis des Lernens	2
Die Geradführung einer Viergelenkkette im Einsatz bei einer kleinen Laufmaschine	4
fischertechnik-Eisenbahn (Teil 1).....	11
Perlentauchen (Teil 2)	14
Handaufzug-Mechanik.....	22
Die Ein- und Ausgänge des TX Controllers	24
I ² C mit dem TX – Teil 2: LED-Display	32
Die Dampfmaschine	38

Termine

Was?	Wann?	Wo?
Spielwarenmesse (neue fischertechnik-Kästen)	30.01.- 04.02.2013	Nürnberg
Fan Club Tag	14.07.2013	Waldachtal/ Tumlingen
fischertechnik Convention 2013	28.09.2013	Erbes-Büdesheim

Hinweise

Alle bisherigen Ausgaben der ft:pedia gibt es jetzt auch in Farbe und 3D: als [EDV-Nachdruck bei Knobloch](#) für 8,95 €.
Seit dem 17.12.2012 gibt es ein Update auf [Version 3.2.0](#) von Robo Pro.

Impressum

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

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

Autoren: Stefan Brunner, Stefan Falk (steffalk), Dirk Fox (Dirk Fox), Ralf Geerken (ThanksForTheFish), Walter-Mario Graf (bumpf), Thomas Püttmann (geometer).

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.

Kinematik

Die Geradföhrung einer Viergelenkkette im Einsatz bei einer kleinen Laufmaschine

Ralf Geerken

In vielen Fällen der „Praktischen Getriebelehre“ wird eine Kreisbewegung in eine geradlinige Bewegung umgesetzt. Hier wird nicht nur erklärt wie man mithilfe eines Koppelkurvetriebes einer Laufmaschine zu einem parallelen Gang verhilft, sondern auch wie man mithilfe einer Koppelkurvenscheibe die unterschiedlichsten Kurvenformen auf Papier bringen kann.

Am Anfang hatte mein Sohn Felix die Idee, einen Laufroboter zu bauen. Dieser sollte vier Beine haben. Jedes Bein sollte dabei von einem eigenen Motor angetrieben werden. Gar keine schlechte Idee, sagte ich ihm, und überlegte schon, wie ich die Mechanik dazu gestalten könnte. Wir diskutierten ein wenig und ich kam darauf, dass wir ja vielleicht nur zwei Motoren bräuchten, die jeweils zwei schräg gegenüber liegende Beine antreiben.

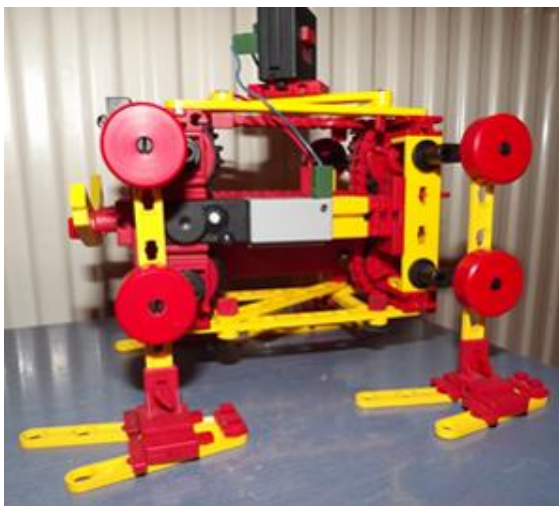


Abb. 1: Hugo

Nach ein wenig herumprobieren kam ich (ab hier war der Funke auf mich übersprungen – Felix hatte dann leider nicht mehr viel mit dem Projekt zu tun) darauf,

dass eventuell sogar ein einziger Motor genügen könnte. Gesagt, getan: Hugo entstand.

Hugo ist ein kleiner Laufroboter, der über die Jahre immer ein wenig verbessert wurde. Die erste Version hatte zwar funktioniert, aber die Zahnräder verrutschten entweder gegeneinander oder die Kette sprang ein paar Ritzel weiter, sodass Hugo dann nur noch vorwärts humpelte anstatt richtig zu laufen. Die letzte Version vom September 2012 hatte dann eine höhere Stabilität, weniger Zahnräder, eine größere Kettenumschlingung und obendrein wesentlich weniger Gewicht. Das Gewicht von Hugo spielt eine sehr große Rolle, weil seine Art, sich vorwärts zu bewegen, eine ganz besondere ist.

Der Mini-Motor muss nämlich immer das gesamte Gewicht von Hugo um 12 mm anheben. Das ist ein großer Nachteil – ich komme später darauf zurück. In einer seitlichen Skizze sieht die Art zu Laufen von Hugo so aus:



Abb. 2: Bewegungsablauf

Das sieht zwar sehr lustig aus – besonders ergonomisch ist das aber nicht. Übrigens: Der Gang eines Menschen ist zwar nicht ganz so extrem, er ist aber auch nicht besonders ergonomisch, d. h. im Klartext: Der Mensch braucht zu Laufen sehr viel Energie, weil er sein Eigengewicht bei jedem Schritt immer ein wenig anheben muss. Auch Hugo braucht ganz schön viel Energie beim Laufen, obwohl er sich ja eigentlich nur vorwärts und nicht auf und ab bewegen soll. Die Mechanik von Hugo zeigt Abb. 1.

Die Beine stehen immer fest verankert durch die beiden Rast-Kurbeln senkrecht zum Boden. Bewegen sich jetzt das rechte vordere und das linke hintere Bein nach oben und gleichzeitig das linke vordere und das rechte hintere Bein nach unten, dann hebt sich Hugo um den Rastkurbelradius nach oben, gleichzeitig aber auch um den Radius nach vorne.

Das linke vordere und das rechte hintere Bein bleiben dabei auf dem Boden, wie man in Abb. 3 sieht.

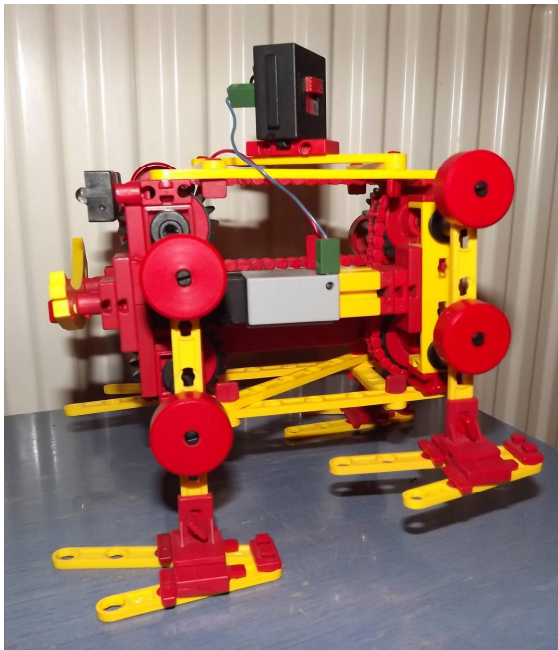


Abb. 3: Hugo macht einen Schritt

Das Ganze ist ein wirklich sehr einfacher Laufantrieb. Hugo kann damit auch auf

unebenem Gelände, auf Sand, auf Teppich und auf glattem Untergrund sehr gut laufen. Auch kann er kleinere Hindernisse wie z. B. Teppichkanten problemlos überwinden. Zwei Sachen kann er aber nicht:

1. Er kann nicht lenken bzw. gelenkt werden – jedenfalls ist mir hierzu noch keine einfache Lösung eingefallen.
2. Er kann kein zusätzliches Gewicht tragen. Hugo muss ja nicht nur sein Skelett tragen, sondern auch den Mini-Motor und die Batterie. Und da haben wir das Problem: Die Batterie hält nicht allzu lange und der Spielspaß ist relativ schnell vorbei. Legt man auch nur ein kleines Gewicht oben auf Hugo, bewegt er sich überhaupt nicht mehr.

Als bald kam mir aber die Idee etwas umzusetzen, was mir mein werter Herr Vater mal vor längerer Zeit erklärt hatte: Wie man mithilfe einer Viergelenkkette [2] eine Drehbewegung in eine geradlinige bzw. parallele Bewegung umwandeln kann. Er zeigte mir damals seine alte Koppelkurvenscheibe aus Astralon, mit der man mit einfachsten Mitteln die unterschiedlichsten Kurven auf Papier übertragen kann. Zur Erklärung: Eine Viergelenkkette hat nichts mit einer Zahnradkette zu tun, sondern es geht um die Verbindung (Verkettung) von vier Gelenken.

In der klassischen Getriebelehre werden diese Getriebe auch Koppelgetriebe oder Kurbelschwinge genannt. Ich will hier aber gar nicht auf die ganzen Begriffe oder gar die mathematische Herleitung oder vielfältigen Einsatzmöglichkeiten von Kurbelschwingen eingehen. Ich möchte hier einmal darstellen, wie einfach es sein kann, mit fischertechnik eine Idee in die Praxis umzusetzen – und zwar ohne vorher ein tief greifendes Maschinenbau-Studium absolviert zu haben.

Die Idee war ganz einfach: Wenn es auf eine mechanisch einfache Weise möglich ist, aus einer Drehbewegung eine Kurven-

form zu generieren, bei der es Teile mit geradliniger Bewegung gibt, dann muss es auch eine Möglichkeit geben, Hugo so umzubauen, dass er nicht mehr sein Gewicht heben muss, sondern dass er sich parallel (oder annähernd parallel) zum Boden bewegt. Dabei muss er dann nicht mehr sein Eigengewicht, sondern nur noch seine Beine heben und sich selbst nur nach vorne schieben.

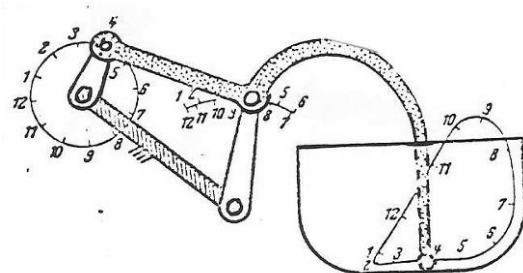


Abb. 4: Mechanik einer Teigknetmaschine

Als ein klassisches Beispiel sei hier einmal ein Bild vom Gelenkviereck einer Teigknetmaschine gezeigt (Abb. 4). Auf der dargestellten Bahnkurve gibt es gleich zwei Geradfürungen, und zwar zwischen den Strecken 2 und 5 sowie 10 und 1. Die Strecke zwischen 10 und 1 war für meine Zwecke die interessantere. Mit einem fischertechnik-Nachbau kann man die Bewegung sehr schön nachvollziehen (siehe Abb. 5 und 6).

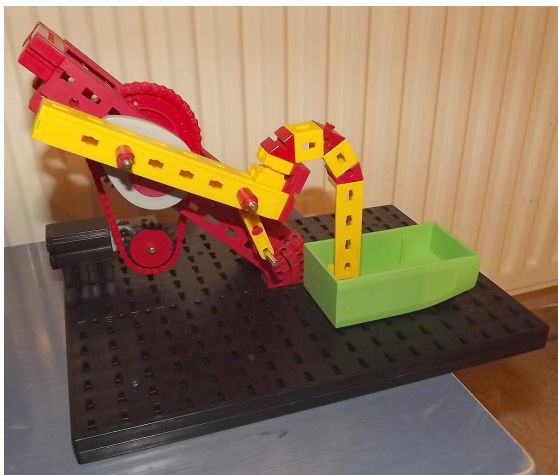


Abb. 5: Teigknetmaschine Stellung 1



Abb. 6: Teigknetmaschine Stellung 2

Für die Laufmaschine probierte ich also ein bisschen herum und fand dann schließlich ein gutes Längenverhältnis, was zur Rastkurbel passte und einen guten Vorschub sowie eine gute Anhebung der Füße hatte. Die Kurve dazu habe ich mal mit der alten Schablone meines Vaters aufgezeichnet.



Abb. 7: Entwurf von Viergelenkketten

Diese Schablone wird folgendermaßen bedient: Man schneidet sich zwei Plastik- oder auch Pappstreifen in der gewünschten Länge zurecht. Der kürzere Streifen wird mit einer Heftzwecke oder einem kleinen Nagel durch das Papier auf der Unterseite eines Kartons befestigt. Im richtigen Abstand, in unserem Fall 12 mm (das ist der Abstand der Achsmittelpunkte der Rastkurbel), steckt man eine Stecknadel so hindurch, dass die Spitze nach oben zeigt. Genauso verfährt man mit dem längeren Stück.

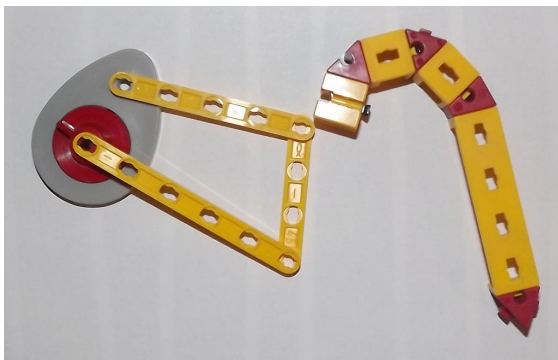
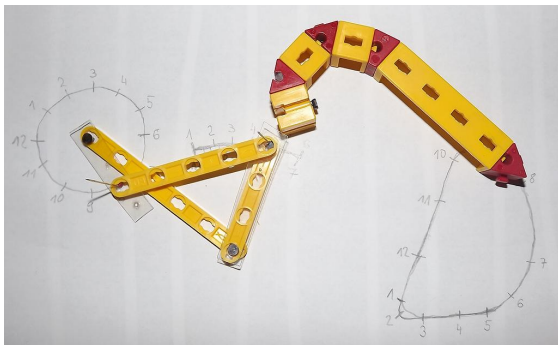
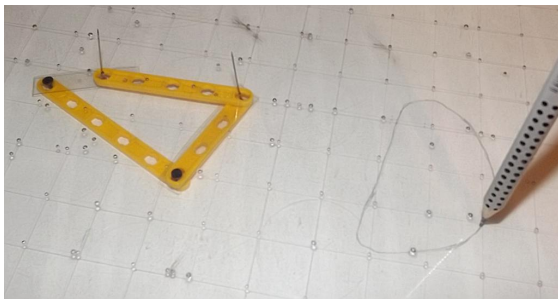


Abb. 8: Verwendung der Schablone für die Teigknetmaschine

Jetzt setzt man die in unserem Fall durchsichtige Schablone im richtigen Abstand auf die zwei nach oben stehenden Nägel. Wenn man nun an dem Nagel der kürzeren Seite dreht, macht die gesamte Schablone eine taumelnde Bewegung, und man kann durch die vielen Löcher in der Schablone erst mal beobachten, welche Kurven-

formen man denn bevorzugt betrachten möchte. Die vermeintlich besten kann man dann mit einem dünnen Stift auf das Papier übertragen (Abb. 7 und 8).

Die Markierung der Winkel ist nicht ganz einfach, aber für eine spätere Entscheidung, welches Getriebe man denn aufbauen möchte, absolut notwendig. Daran kann man mit einiger Übung ablesen, wie die Geschwindigkeiten auf der kurvigen Bahn verteilt sind und ob es evtl. zu höheren oder auch zu zu hohen Beschleunigungen kommt. Bei der Teigknetmaschine gibt es z. B. zwischen 12 und 1 sowie zwischen 1 und 2 einen sehr großen Geschwindigkeitsunterschied. Zwischen 1 und 2 ist der Knetarm sehr langsam, was aber auch gut so ist, weil er hier sehr viel Kraft benötigt, um den Teig wegzudrücken. Zwischen 2 und 3 beschleunigt er dann wieder.

Eine weitere Schwierigkeit waren die Füße. Hugo steht ja ziemlich ausgewogen und sicher auf zwei schräg gegenüber liegenden Rastkurbeln geben den Beinen jeweils den Halt, und die Form der Füße sorgt dafür, dass Hugo nicht schräg umkippt. Die Fußsohlen von Hugo sind dadurch immer parallel zum Boden.

Bei Herrmann war das nicht so einfach, weil die Fußsohlen beim Aufsetzen und beim Anheben leider noch keine Parallelität zum Fußboden haben. Die haben sie erst in der Strecke der Geradföhrung. Aber ganz so schlimm war das dann doch nicht, weil der Winkel und die Länge der Füße beim Aufsetzen sowie beim Anheben dafür sorgen, dass der Boden schon berührt wird, bevor die Füße parallel zum Boden stehen. Jetzt noch ein bisschen probieren, wie man die vier Gelenkvierecke auf kleinem Raum unterbringt, und dann lief Herrmann.

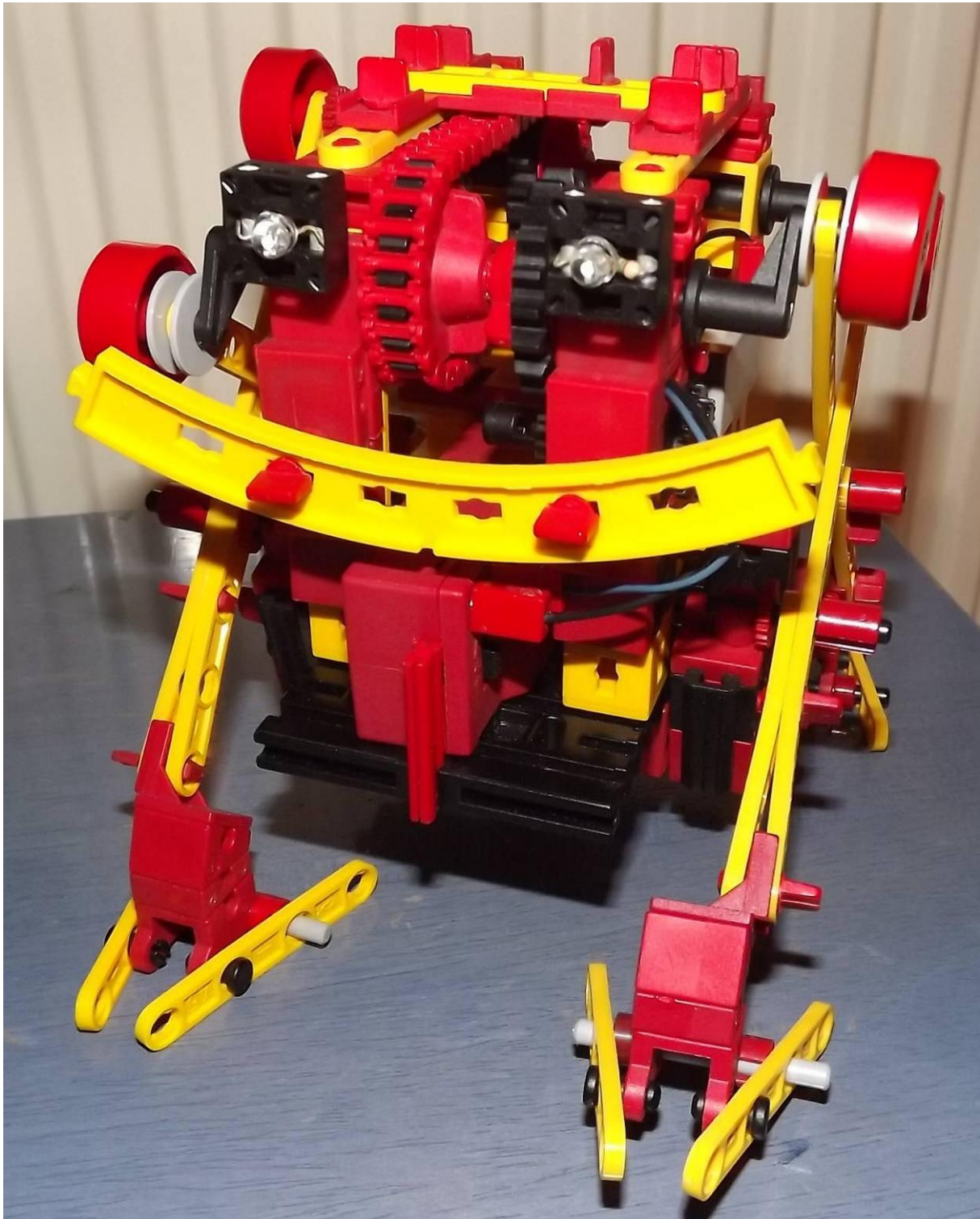


Abb. 9: Herrmann zum Nachbauen

Es macht sehr viel Spaß, die beiden kleinen Laufmaschinen nebeneinander laufen zu sehen. Sie laufen in etwa gleich schnell, aber unterschiedlich: Hugo mit

seinen etwas tapsigen Schritten und Herrmann mit seinen sehr weiten, fast schon ausladenden Schritten, die ihm aber zu einem sehr glatten Gang verhelfen.

Von Herrmann gibt es bereits Videos:

- [Herrmann von der Seite](#)
- [Herrmann frontal](#)
- [Herrmann erklimmt die Teppichkante](#)

Wer die verschiedenen Generationen der Laufmaschine nachbauen möchte, findet weitere Bilder auf der [ft-Community](#).

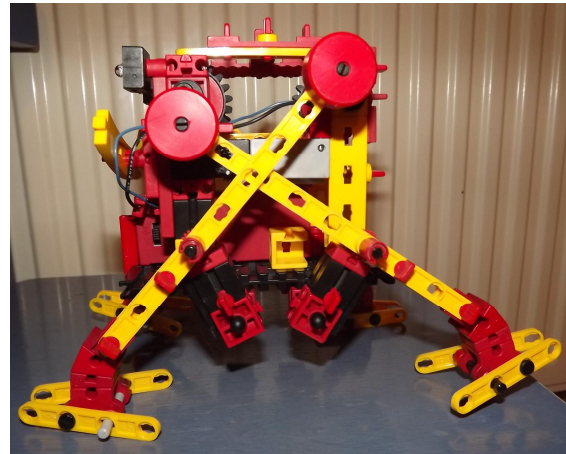


Abb. 11: Herrmann im Profil

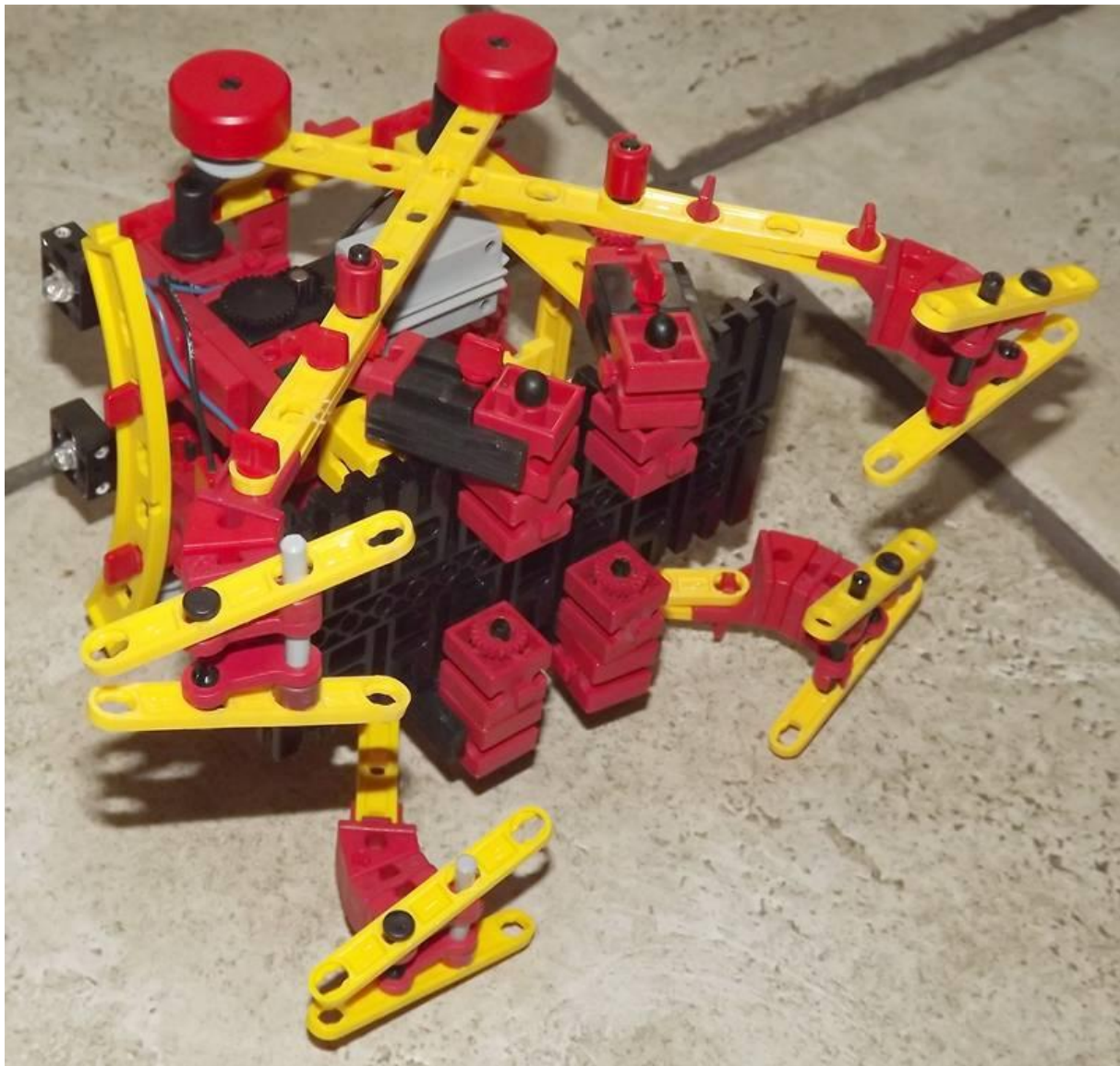


Abb. 10: Herrmann zum Nachbauen

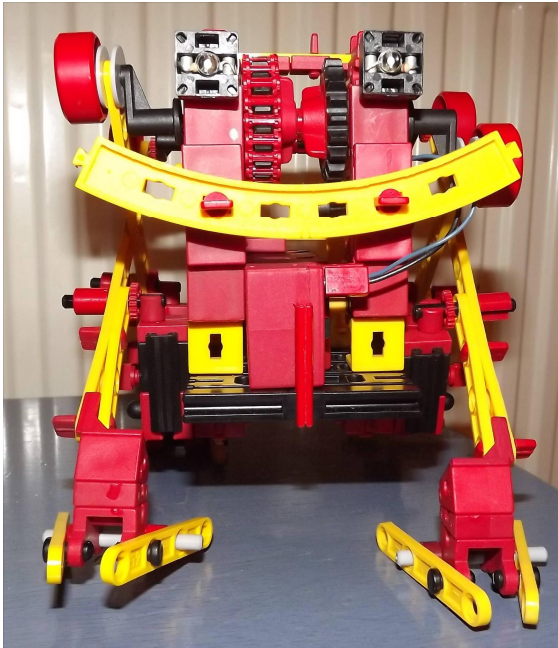


Abb. 12: Herrmann von vorne

Der Ausblick in die Zukunft: Fritz (die dritte Generation) mit vier Füßen, die aber während der gesamten Schrittbewegung parallel zum Boden verlaufen. Damit kann er dann z. B. höhere Teppichkanten überwinden. Hier ist Hugo nämlich ganz klar gegenüber Herrmann im Vorteil.

Lothar (die vierte Generation) soll viel größer und mit einem stärkeren Motor ausgestattet werden. Die fünfte Generation soll dann lenken können – aber dazu gibt es später mehr.

Quellen

- [1] Prof. Dr.-Ing. Kurt Rauh, Technische Hochschule Aachen: *Praktische Getriebetechnik* Heft 2, Kardanbewegung und Koppelbewegung, herausgegeben zur Jahreswende 1937/1938
- [2] Fischer-Werke: *Viergelenkgetriebe*. [hobby 2 Band 5](#), Tumlingen 1973, S. 4-17.

Schienenfahrzeuge

fischertechnik-Eisenbahn (Teil 1)

Walter-Mario Graf

Im ersten Teil möchte ich euch aufzeigen, wie ich vom Thema „fischertechnik-Eisenbahn“ infiziert wurde. Ich möchte euch, das heißt vor allem den Kindern, die Eisenbahn ein bisschen schmackhaft machen. Den Bericht schließe ich mit einer kleinen Bauanleitung. In Teil 2, der zu einem späteren Zeitpunkt erscheinen wird, geht es dann um das Thema fischertechnik-Gartenbahn.

Als vor fünf Jahren ein User im fischertechnik-Forum die Frage stellte: „Kann fischertechnik auf Duplo-Schienen fahren?“, brach bei mir der Eisenbahnvirus aus. Da ich noch irgendwo eine Kiste mit Duplo-Schienen herumstehen hatte, machte ich mich noch am selben Abend an die ersten Versuche – und es klappte auf Anhieb (Abb. 1).



Abb. 1: Duplo-Bahn-Chassis

Nun muss eine richtige Lokomotive her. Doch wie bringe ich die Stromversorgung unter? Denn von den Schienen kriege ich keine Energie. Wie kann ich die Lok steuern? Wie starte und stoppe ich sie? Kann sie nicht nur vorwärts, sondern auch rückwärts fahren? Fragen über Fragen.

Nach unzähligen Versuchen entschied ich mich für die [Lok mit dem alten IR Control Set](#) (Abb. 2). Dies, weil Kinder so auf einfache Weise den Zug steuern können.

Natürlich kann man auch das neue Control Set einbauen.



Abb. 2: Bahn mit IR Control-Set

Eine richtige Eisenbahn besteht natürlich nicht nur aus einer Lok, sondern auch aus vielen Waggons. Als ich mich an den Bau des ersten Wagens machte, kam mir mein Sohn in den Sinn, wie er damals mit der Duplo-Eisenbahn spielte. Selten nahm er die batteriebetriebene Lok zum Spielen, viel lieber rangierte er die Lok von Hand, hängte Wagen an Wagen und stieß die ganze Zugkomposition mit ohrenbetäubendem Lärm auf den Schienen umher. Da fasste ich den Entschluss, genau so eine Eisenbahn zu entwickeln. Sie sollte auf einem einfachen Konzept basieren, damit sie auch Kinder im Vorschulalter mühelos zusammen bauen können. An der Convention 2011 stellte ich die Eisenbahn zusammen mit meiner fischertechnik-Gartenbahn aus.

Da mich viele Besucher fragten, ob die Eisenbahn auch als Bausatz im Sortiment zu haben sei, stelle ich nun in einer Bauanleitung das Grundprinzip, auf dem die ganze Bahn aufgebaut ist, vor. Der Aufbau für alle Wagen und Loks ist gleich. Mit nur wenigen Teilen (Abb. 3) wird der Unterbau zusammengestellt.

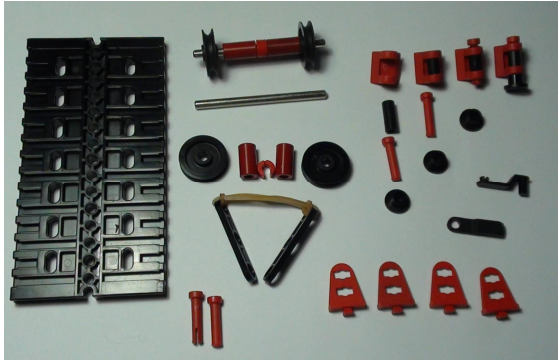


Abb. 3: Bauteile des Unterbaus

Dabei dient die Grundplatte 120x60 als Chassis. Zuerst werden die Kupplungen montiert, dazu stoßen wir zwei Rastachsen 20 durch die Platte. Danach steckt ihr die Streben 30, die zuerst mit einem Gummiband verbunden werden müssen, auf die Rastachsen auf und fixiert sie mit den Riegelscheiben. An den Streben werden noch ein Schalthebel und gegenüber ein Gelenkstück befestigt (Abb. 4).



Abb. 4: Die Kupplungen

Als nächstes werden die Puffer, die ihr vorher zusammenbaut, montiert. Dazu braucht ihr folgende Teile: Gelenkwürfelklaue, Lagerhülse schwarz und Achskappen (Puffer) schwarz, je vier Stück.

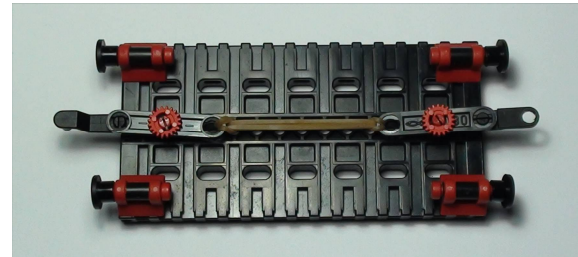


Abb. 5: Die Puffer

Damit das Ganze auf den Schienen fährt, fehlt uns nur noch die Achse mit den Rädern. An der Grundplatte machen wir zuerst auf der einen Seite die Kupplungsstücke 2 fest. An diesen werden die Achsen befestigt, bestehend aus einer Klemmbuchse 5 rot, die ihr zuerst mittig auf die Metallachse 60 aufklemt. Dann schiebt ihr noch beidseitig je eine Hülse 15 rot und eine Seilrolle schwarz über die Achse. Zum Schluss montiert ihr noch die übrig gebliebenen Kupplungsstücke 2 an die Grundplatte und fertig ist unser Unterbau (Abb. 6).

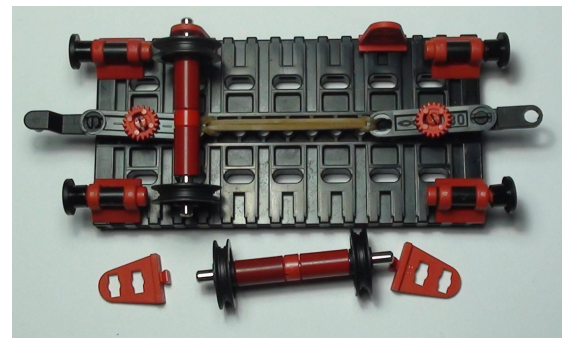


Abb. 6: Unterbau mit Achsen

Noch ein kleiner Hinweis: Da die Rastachsen 20 große Maßtoleranzen haben, müsst ihr die Puffer und die Riegelscheiben eventuell mit Heißkleber fixieren.

So, jetzt ist eure Fantasie gefragt. Denn wie die Lok oder die Waggons ausschauen sollen, ist jedem einzelnen überlassen. Hier noch einige Anregungen (Abb. 7). Oder schaut euch die Beispiele in der ft-Community an, wie die kleine [Dampflok mit Röhrenanhänger](#) oder die Waggons, die ich auf der [Convention 2011](#) gezeigt habe.



Abb. 7: Beispiele für die Waggon-Gestaltung

fischertechnik-Basiswissen

Perlentauchen (Teil 2)

Stefan Falk

In der letzten Ausgabe zeigten wir ein paar gebräuchliche und weniger gebräuchliche ältere fischertechnik-Teile. Heute setzen wir fort mit ein paar interessanten mechanischen Teilen und einem Querschnitt über die vielen früher verwendeten Motortypen und ihr Zubehör.

Der erste Motor

Fangen wir an mit dem Urahn aller fischertechnik-Motoren aus dem Kasten *mot 1*. Er ist für 6,8 V Gleichspannung ausgelegt und treibt eine direkt auf seiner Rotorachse fest angebrachte Schnecke an. Sein Gehäuse ist 30 mm breit und 45 mm lang – der Rest passt allerdings nicht gut ins Raster, denn seine Höhe hat ein etwas krummes Maß größer als 30 mm.



Abb. 1: Der Ur-Motor 31039

Das macht allerdings gar nichts, denn er besitzt eine rote Bodenplatte, die sich in Längsrichtung verschieben lässt, und diese wiederum trägt eine drehbare Platte mit zwei Stegen, die in die Nuten von Bausteinen passen. So ist dieser Motor gut verbaubar und kann obendrein in seiner Lage fein justiert werden (Abb. 1).

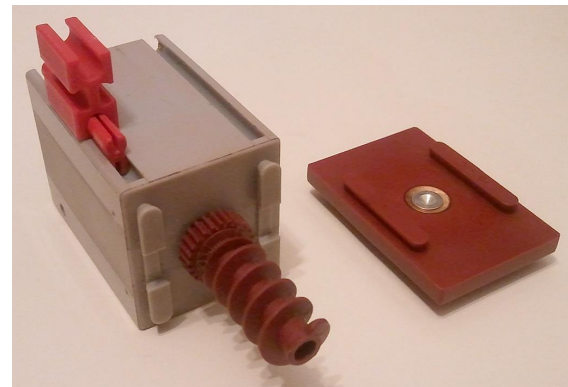


Abb. 2: Ur-Motor mit abnehmbarer Bodenplatte

Bei einer jüngeren Variante ist die Bodenplatte durch die nach hinten offenen Nuten abnehmbar. Wie Abb. 2 zeigt, passen die Nuten sogar mit einem Baustein 7,5 ins Raster – obwohl es damals noch gar keinen BS7,5 gab!

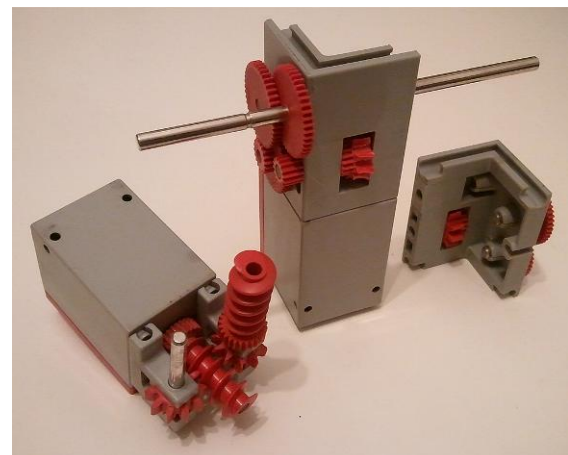


Abb. 3: Getriebeteile für den Ur-Motor

Die Frontseite des Ur-Motors verfügt ebenfalls über zwei Stege, in die allerlei Zubehör eingesteckt werden kann: Eine Kombination aus Z10 plus Achse, ein Z10 plus Schnecke und ein sehr flexibel verwendbares Stufengetriebe aus dem Ergänzungskasten *mot 2* (Abb. 3).

Das Stufengetriebe kann von zwei Seiten in allen Richtungen auf den Motor aufgesteckt werden und bietet durch das Verwenden von Kombinationen der mitgelieferten Untersetzungs Zahnräder eine variable Ausgangsdrehzahl (Abb. 4).

Allerdings liegt die abgehende Achse nirgends im Raster – mitunter muss man also etwas kreativ bei der Verwendung werden.



Abb. 4: Variable Untersetzungsverhältnisse beim Stufengetriebe

Auf eine Innenseite des Stufengetriebes kann noch ein Z10 gesteckt werden (Abb. 5), von dem ab mit grober und auch mit feiner Verzahnung weiter gebaut werden kann. Allerdings wurde dieses Zusatzzahnrad schon früher nur äußerst selten tatsächlich verwendet.

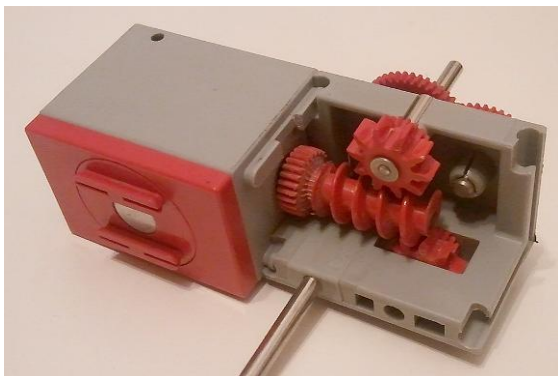


Abb. 5: Zusatzzahnrad im Stufengetriebe

Erst 1975 kam der Getriebeadapter 38203 hinzu, der es ermöglicht, die beiden Aufsteckteile auch um 90° verdreht anzubringen:

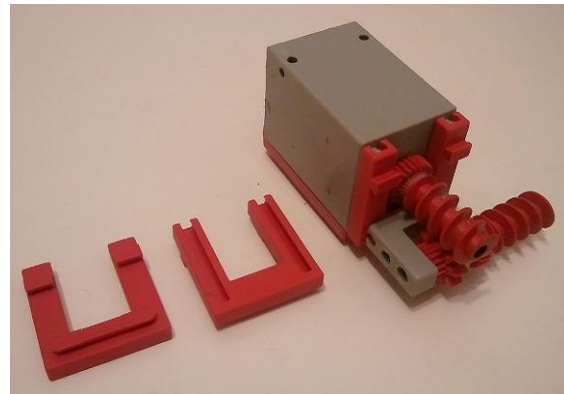


Abb. 6: Getriebeadapter 38203

Durch seine äußerst nützliche Bodenplatte kann dieser Motor in praktisch jeder Lebenslage so justiert werden, dass er perfekt in das gewünschte Abtriebszahnrad eingreift. Nur wenn die auftretenden Kräfte all zu groß werden, hält der Drehwiderstand der Bodenplatte dem nicht stand und der Motor dreht sich aus dem Zahnrad heraus, das er antreiben soll – ein einfacher Überlastschutz.

Den *mot 1* kann man etwa mit dem Ur-Differentialgetriebe 31043 von 1968 recht elegant für einfache Fahrzeugantriebe nutzen:

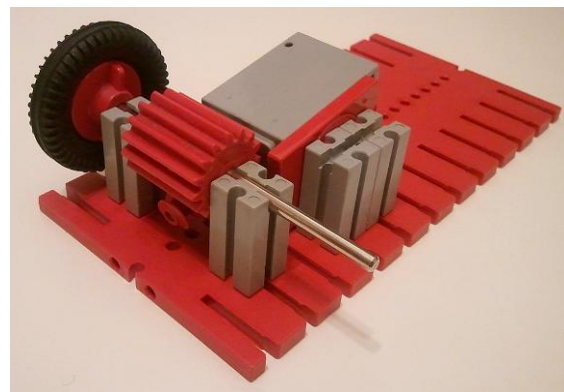


Abb. 7: Motor und Differential in einem Fahrzeugantrieb

Dieses Differential ist außen ein einziges 30 mm breites Z15. Das ist nützlich für die Konstruktion von Schaltgetrieben, bei

denen ein Zahnrad in Achsrichtung verschoben wird und dennoch ständig in Eingriff mit dem Differential bleibt. Abb. 8 zeigt ein einfaches Schaltgetriebe im „schnellen Gang“ (wenn man die untere Achse antreibt):

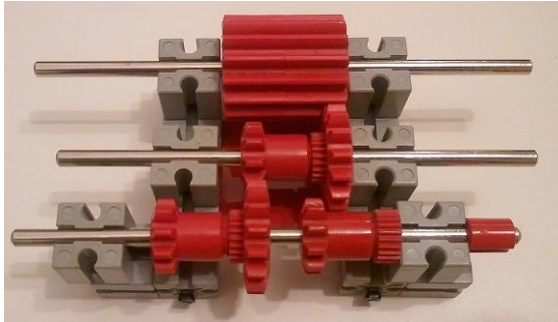


Abb. 8: Getriebe in hohem Gang

Verschiebt man die mittlere Achse, wird das Differential langsamer angetrieben:

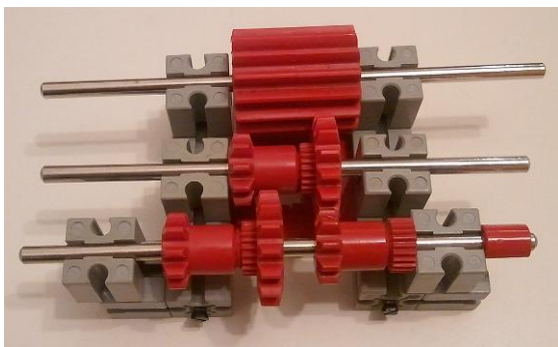


Abb. 9: Schaltgetriebe in niedrigem Gang

Schon kurz nach dem *mot 1* kamen mit dem Ergänzungskasten *ft 06* die Segment- und Kurvenscheiben auf den Markt:

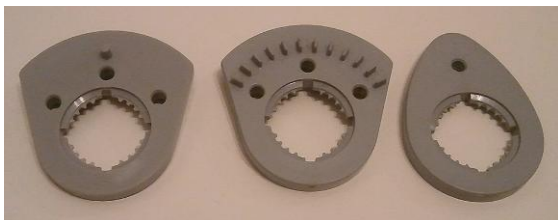


Abb. 10: Segmentscheibe, Kurvenscheibe

In deren exzentrisch angebrachten Bohrungen halten Achsen ohne weitere Mittel sehr fest.

Die Segmentscheibe kann man für Schrittschaltgetriebe oder, wie Abb. 11 zeigt, für ein Hammerwerk verwenden.

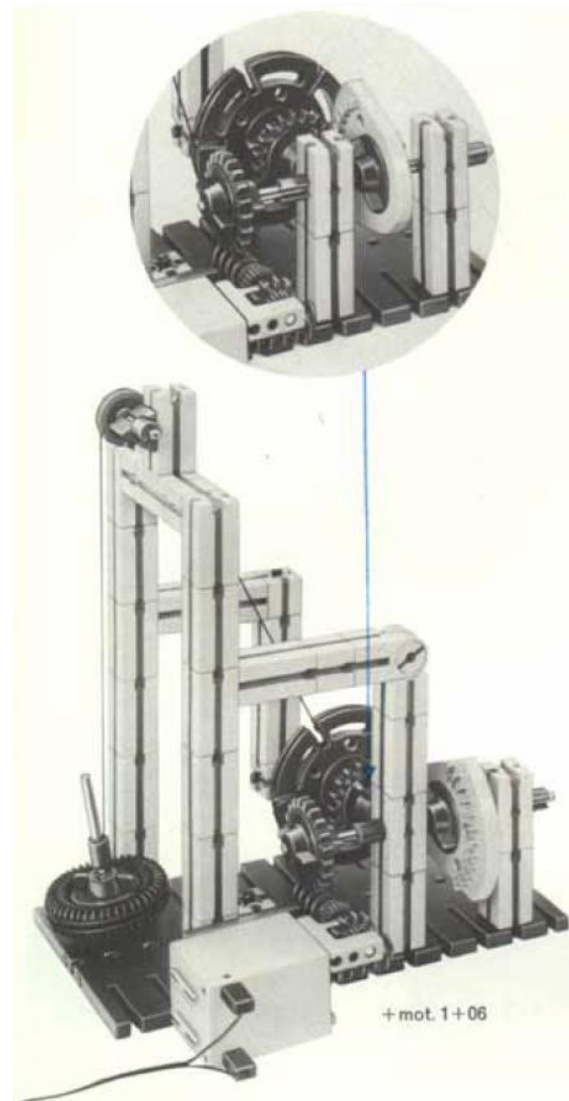


Abb. 11: Hammerfallwerk aus der Anleitung zum *ft 200 Grundkasten* [1]

Die Kurvenscheiben werden natürlich gerne für Modelle der Ventilsteuerung beim KFZ-Motor verwendet (Abb. 12).

Im Vorgriff auf die zum Elektromechanik-Programm gehörenden Teile sei kurz auf ein Schneckenkupplungs-Bauteil hingewiesen. In der Anordnung von Abb. 13 sind die beiden Schnecken der Motoren über das Kupplungsteil direkt miteinander verbunden. Führt man einem Motor Strom zu, arbeitet der andere als Generator – eine an ihm angeschlossene Lampe leuchtet.



Abb. 12: Ventilsteuerung eines KFZ-Motors aus der Anleitung zum ft 400 Grundkasten [1]

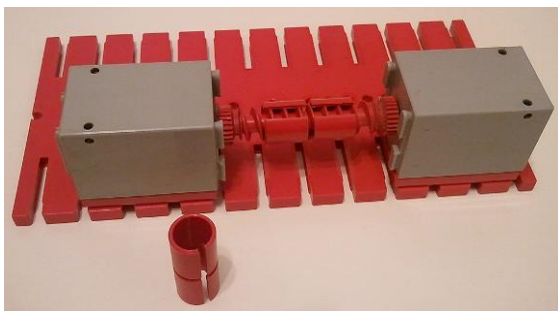


Abb. 13: Schneckenkupplung

Ein anderer Verwendungszweck ist ein direkter Abtrieb auf ein Zahnrad Z10 – solange nicht zu viel Drehmoment abverlangt wurde:

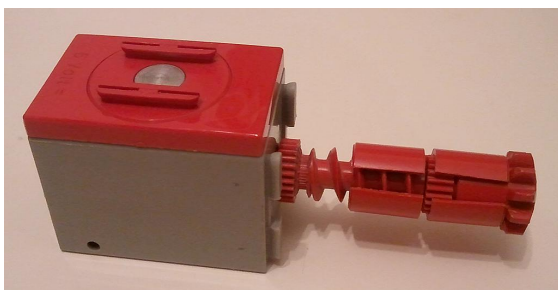


Abb. 14: Direktabtrieb von Ur-Motor auf Z10

Minimotoren

1969 kam schon der *mini-mot 1* heraus: Ein Kleinstmotor in genau der Größe eines Bausteins 30:



Abb. 15: *mini-mot 1* und Zubehör

In Abb. 15 sieht man in der Mitte den Minimotor neben einem Baustein 30 zum Größenvergleich.

Auch für den Minimotor existieren verschiedene Aufsteckgetriebe: Eine Untersetzung auf eine weitere kleine Schnecke, eine auf eine Schecke für normale Zahnräder mit großen Zähnen, eine beidseitig herausgeführte Achse sowie auch wieder ein aufsteckbarer Getriebeblock. Letzterer ähnelt dem *mot 2*-Aufsteckgetriebe für den *mot 1*, kann aber nur in zwei Richtungen aufgesetzt werden und hat keine variable Untersetzung. Für viele Zwecke passt das aber trotzdem, wie Abb. 16 zusammen mit den im *mot 2* bereits enthaltenen Raupenbändern zeigt:

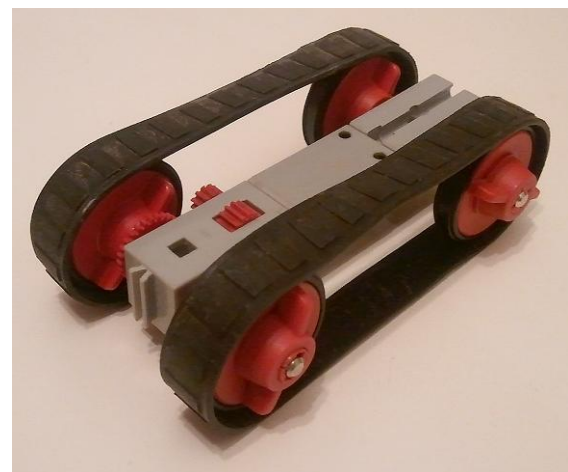


Abb. 16: Kleinst-Raupenfahrzeug

Außer dass das spätere Aufsteckgetriebe einen Zapfen anstatt Federn an der Stirnseite besaß, gibt es noch einen feinen Unterschied: Die erste Version führt die Abtriebsachse nicht genau rasterkonform heraus, wie Abb. 17 zeigt:

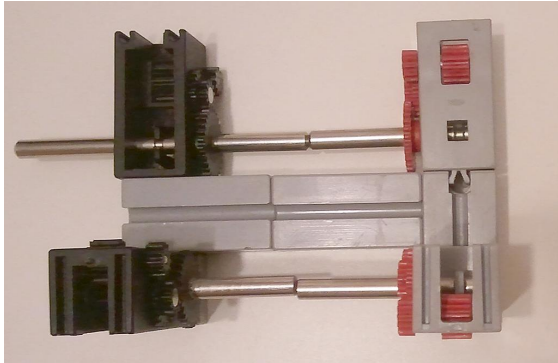


Abb. 17: Die Ur-Version des mini-mot 1 Aufsteckgetriebes (rechts) ist nicht genau rasterkonform, die neue (links) schon

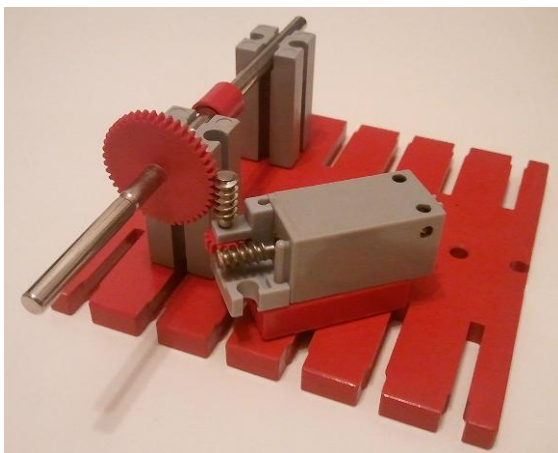


Abb. 18: Der mini-mot 1 ist fein justierbar

Da insbesondere die kleinen Schnecken aus Abb. 15 überhaupt nicht im ft-Raster liegen, lag auch eine separate rote Bodenplatte bei – ganz ähnlich der fest angebauten des *mot 1*. Damit kann man den *mini-mot 1* fein justieren, wie Abb. 18 zeigt. Übrigens ist diese Justierplatte genau 7,5 mm hoch. Das war also schon 1969 ein erster Baustein in halber Rasterhöhe – lange vor dem heute üblichen Baustein 7,5.

1974 kam das Hubgetriebe im Kasten *mini-mot 2* heraus (das war damals eine kleine Sensation). Im selben Kasten war auch ein in der Breite auf 20 mm gewach-

sener, dafür aber auch etwas kräftigerer Minimotor enthalten:

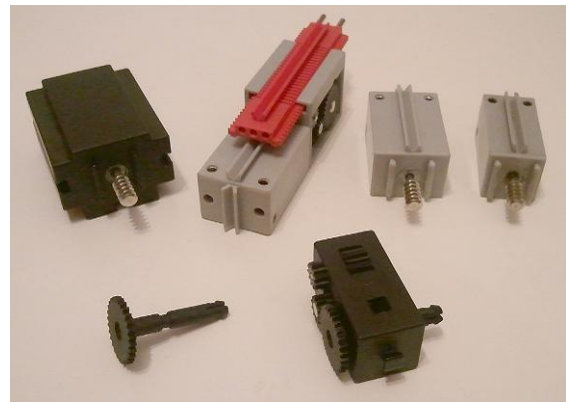


Abb. 19: S-Motor, mini-mot 2 und mini-mot 1 im Vergleich

Der spätere S-Motor ist schon wieder deutlich größer, aber dafür auch deutlich stärker, und er brauchte weniger Strom. Er war ab 1985 grau und ab 1990 schwarz. Sein Aufsteckgetriebe bietet einen Rastachsen-Abtrieb (Rastachsen gibt es erst seit 1990).

Wenn wir schon beim Hubgetriebe sind, schauen wir uns mal all die Zahnstangen an, die es gab bzw. gibt:



Abb. 20: Zahnstangen

Von links nach rechts zeigt Abb. 20 die kurzen und langen Zahnstangen mit grober Verzahnung (sie sind etwas länger als 30 bzw. 60 mm). Spätere Versionen trugen statt eines fast massiven Steges einen V-förmigen für die Nuten der Bausteine. Diese Zahnstangen gab es zusammen mit Varianten mit feiner Verzahnung bereits im *mot 2*-Kasten. Später kamen die bekannten Hubzahnstangen, ebenfalls mit feiner Verzahnung.

Ein etwas seltsam aussehendes, aber durchaus nützliches Bauteil ist der *Schneckenantrieb 32886*, der ins Aufsteck-

getriebe der kleinen Motoren passt und eine Schnecke für die grobe Verzahnung fest trägt:

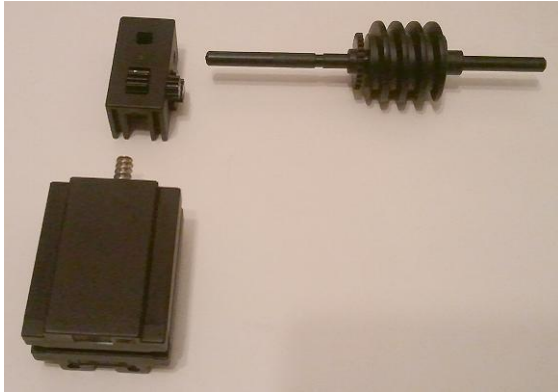


Abb. 21: Schneckenantrieb 32886

Das passt wunderbar zum Antrieb eines Drehkranzes 31393 – da sitzt nämlich alles perfekt im ft-Raster:

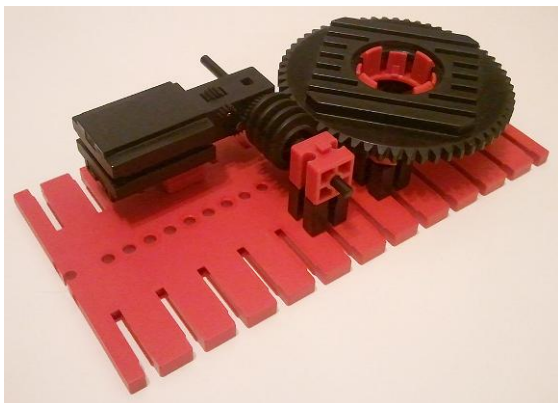


Abb. 22: Antrieb des Drehkranzes

Der Ur-Motor wird erneuert

Eine tolle Weiterentwicklung gab es 1987 mit der Einführung des *M-Motors* 32618:

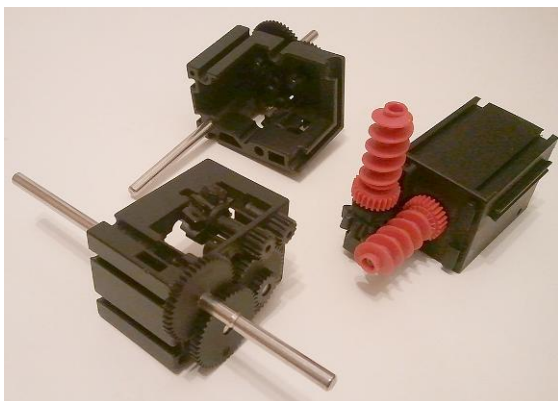


Abb. 23: M-Motor 32618 mit Zubehör

Gegenüber dem Ur-Motor hat der M-Motor eine ganze Reihe von Vorteilen:

- Sein Gehäuse passt mit 30 mm x 30 mm x 45 mm genau ins Raster.
- Er ist kräftiger.
- Er läuft ruhiger und noch gleichmäßiger, weil im Inneren mehr Magnete im Stator des Elektromotors sitzen. Das merkt man, wenn man am Ur-Motor und am M-Motor von Hand dreht: Man spürt, dass der M-Motor in kleineren Winkelabständen „anzieht“.

Die justierbare Bodenplatte von Abb. 2 passt auch auf den M-Motor. Wer genau hinschaut, sieht bei dem neuen Getriebeaufsatz mit Schnecke und beim neuen Stufengetriebeaufsatz in Abb. 23, dass die Zähne der Z10 schräg angeordnet sind. Das ist der Form der Motorschnecke besser angepasst und führt vor allem zu weniger Verschleiß am Zahnrad.

Zur Perfektion weiterentwickelt wurde das [Stufengetriebe des M-Motors](#):

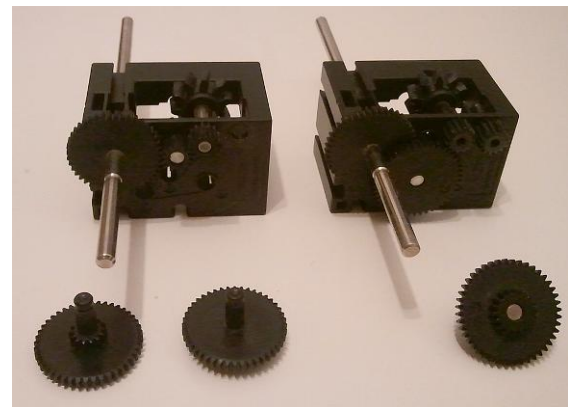


Abb. 24: Das verbesserte Stufengetriebe des M-Motors

Die Abtriebsachse sitzt jetzt im Raster (als ob da ein BS15 mit Bohrung verwendet wäre). Zudem kann man die verschiedenen Untersetzungsverhältnisse (1:240, 1:77 und 1:32) erreichen, ohne dass die Abtriebsachse an ungünstiger verwendbare Positionen wandern müsste. Wie Abb. 24 zeigt, kann man die Aufsteckzahnräder und ein neues, umsteckbares kleines Zahnrad

verschieden aufstecken. So erreicht man dieselben Untersetzungsverhältnisse wie beim Ur-Stufengetriebe (vgl. Abb. 4), nur dass die Abtriebsachse immer an derselben, rasterkonformen Position bleibt.

Das neue Stufengetriebe passt übrigens auch auf den Ur-Motor. Es kann zwar nur noch von einer Seite (aber immer noch in vier Richtungen) aufgesteckt werden, dafür erhält man an der Stirnseite und an weiteren Stellen Nuten – besser kann man so einen Getriebekblock wohl nicht entwerfen.

Powermotoren

1999 erschien wieder eine neue Generation Motoren, und diesmal eine richtig kräftige. Die *Power-Motoren* gibt es mit schwarzer, grauer und roter Kappe in ansonsten identischen Abmessungen. Gleich eingebaut sind nämlich Untersetzungsgetriebe 1:8 (schwarze Kappe), 1:20 (grau) bzw. 1:50 (rot):



Abb. 25: Die drei Varianten des Power-Motors und die drei Aufsteckteile

Man verwendet also je nach Situation den richtigen Motor und nicht mehr wie früher verschieden konfigurierte Aufsteckgetriebe. Deshalb gibt es als Abtriebs-Zubehör zum Aufstecken auch nur einen Adapter für Rastachsen, ein Z10 und ein Kegelfahrad, welches nur zum Antrieb des aktuellen Differentialgetriebes gedacht ist.

Wenngleich diese Motoren mit ihren hohen Drehmomenten für viele der heute üblichen größeren Modelle unverzichtbar

wurden, haben sie doch gegenüber ihren Vorgängern einen oft kritisierten Nachteil: Zum Anbau gibt es lediglich eine mit einer einzigen Schraube fixierte Platte mit vier Halbnuten. So mancher hat deswegen schon geschimpft, findige Leute fanden aber schon immer geeignete Lösungen [2] – und zum Glück hat fischertechnik diesen Missstand ja mit den aktuellen XM- und Encoder-Motoren korrigiert.

Spezielle Motoren

Ungewöhnlich geformt, selten anzutreffen und mitunter entsprechend begehrt ist der *RC-Radantrieb 31259* aus dem Jahre 1983:

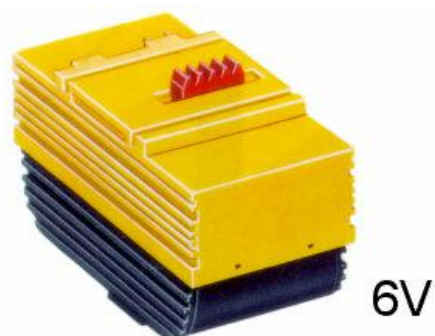


Abb. 26: RC-Radantrieb 31259

Er gehört zu einer Funkfernsteuerung, die mittlerweile von zwei Generationen von Infrarot-Fernsteuerungen abgelöst wurde. Immerhin gibt es dazu passend auch schon einen frühen Servomotor:

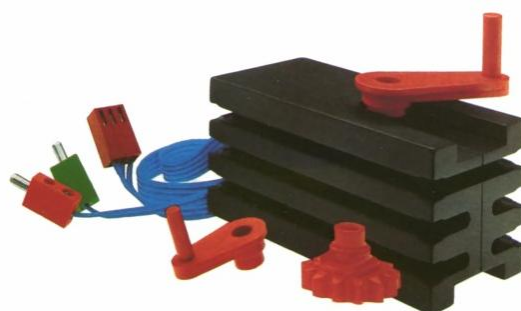


Abb. 27: RC-Servo 30275 mit Anbauteilen

Exoten

Ein ungewöhnliches Bauteil ist eine Kombination aus Schneckenaufsatz, Kontermutter und Spannkegel, mit dem man

einen Direktantrieb mit der Achse des M-Motors oder des Ur-Motors realisieren kann.



Abb. 28: Direktantrieb von einer Motorschnecke aus

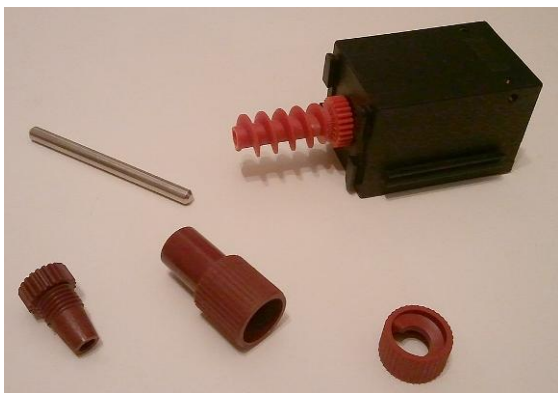


Abb. 29: Die Bestandteile des Direktantriebs

Ganz besonders exotisch ist wohl der *Schrittmotor 32311* aus dem Plotter-Baukasten von 1985.



Abb. 30: Schrittmotor nebst Zubehör

An diesem Motor passt so gar nichts zum fischertechnik-Raster: Sowohl damit man

ihn überhaupt verbauen konnte, als auch damit man eine normale ft-Achse verwenden konnte, wurden Spezialteile benötigt. Die Ansteuerung über seine vier Anschlüsse erfolgt sinnvoll nur in Computing-Modellen.

Schließlich werfen wir noch einen Blick auf den höchst seltenen *Kraftmesser* aus der Zusatzpackung *ft 025*:



Abb. 31: Kraftmesser aus dem Jahr 1970

Die Packung enthielt zwei austauschbare Federn – eine schwache und eine starke. Der Kraftmesser selbst verfügt über eine in Millimetern ausgeführte Skala und verschiedene Anbaumöglichkeiten. Neben interessanten Kräftebeobachtungen in den hobby-Begleitbüchern kann man ihn auch als langes federndes Element verwenden, wie man [hier](#) sehen kann.

Ausblick

In der nächsten Folge unserer kleinen Zeitreise wenden wir uns den vielen Teilen aus dem Bereich Elektromechanik zu.

Quellen

- [1] [Bauanleitung zu den Grundkästen 100-400](#), fischerwerke 1968-1972.
- [2] Andreas Tacke, *ft-Spezialteile made by TST (Teil 2)*, [ft:pedia 3/2012](#), S. 27-28.

Getriebe

Handaufzug-Mechanik

Thomas Püttmann

Eine mechanische Uhr kann man durch Drehen der Krone in beide Richtungen aufziehen. Wie die Gleichrichtung der Drehbewegung funktioniert, wird anhand eines fischertechnik-Modells erklärt.

Beim Aufziehen mechanischer Armbanduhren ist es gleichgültig, in welche Richtung man die Krone dreht. Die meisten Personen drehen die Krone sogar zwischen zwei Fingern periodisch hin und her. Die Spiralfeder des Uhrwerks wird aber natürlich nur durch Drehbewegungen in einer Richtung aufgezogen. Die Drehbewegungen werden also zwischen Krone und Feder gleichgerichtet. Eine solche Gleichrichtung lässt sich sehr schnell mit fischertechnik nachbauen.

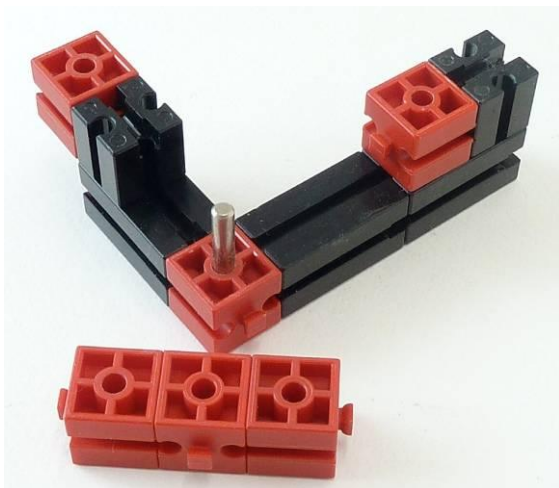


Abb. 1: Gerüst des Modells

In Abbildung 1 erkennt man den einfachen Aufbau des Gerüsts unseres Modells. Die drei Bausteine 15 mit Bohrung im Vordergrund bilden die für die Funktion wesentliche Wippe.

In Abbildung 2 erkennt man, was passiert, wenn die Eingangswelle (Z20 rechts) im

Uhrzeigersinn gedreht wird: Die Wippe kippt gegen den Uhrzeigersinn und das rechte Zahnrad Z20 der Wippe wird gegen das große Zahnrad Z40 gedrückt. Die Ausgangswelle dreht sich damit wie die Eingangswelle im Uhrzeigersinn.



Abb. 2: Eingangswelle im Uhrzeigersinn - Ausgangswelle im Uhrzeigersinn

Wenn die Eingangswelle rechts gegen den Uhrzeigersinn gedreht wird, kippt die Wippe im Uhrzeigersinn und das linke Zahnrad Z20 der Wippe wird gegen das Zahnrad Z40 gedrückt. Die Verlängerung der Drehbewegungskette von der Eingangswelle zur Ausgangswelle um ein Zahnrad sorgt für die Richtungsumkehr: Das Z40 dreht sich wiederum im Uhrzeigersinn.



Abb. 3: Eingangswelle gegen den Uhrzeigersinn – Ausgangswelle im Uhrzeigersinn

Das abgebildete Modell funktioniert schon so sehr gut. Verbessert werden kann es noch, indem man einen Abstandsring 2 mm zwischen Wippe und Gerüst setzt, um die Reibung beim Wippen zu verkleinern. Der restliche Aufbau muss natürlich dementsprechend angepasst werden.

Um zu erzwingen, dass sich das Z40 tatsächlich immer im Uhrzeigersinn bewegt, sollte eine Sperrklinke ergänzt werden. Hier wurde sie der Übersichtlichkeit wegen weggelassen.

Elektronik

Die Ein- und Ausgänge des TX Controllers

Stefan Brunner

Unter dem Weihnachtsbaum befindet sich dein neues fischertechnik ROBO Computing Set. So klein – der ROBO TX Controller – und so viele Anschlüsse: 10 V- oder 5k-Modus, Analog oder Digital – huh? M- und O-Ausgänge? Du weißt zwar nicht genau, was man mit dem ROBO TX Controller alles machen kann, aber eines erkennst du sofort: Es könnte kompliziert werden, da etwas anzuschließen...

Konsequente Weiterentwicklung der Silberlinge

Als Kind bin ich mit den Silberlingen aufgewachsen. Damals war mir nie ganz klar, was die ganzen Anschlüsse am h4-GB machen. Heute bin ich selbst der Papa und wollte wissen, was man so alles an den TX Controller anschließen kann – und habe den TX dazu etwas genauer unter die Lupe genommen (Abb. 1).

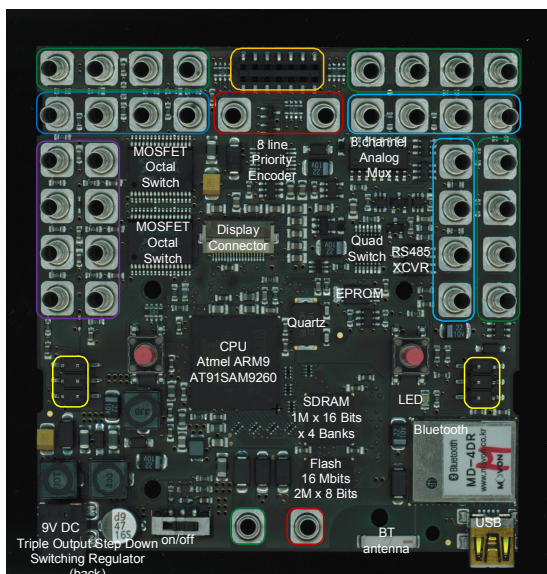


Abb. 1: Dem TX Controller ‚unter die Haube‘ geschaut

Im Prinzip machen alle Elektronikbausteine, der Ur-h4-GB der 70er Jahre, der Schwellenwertschalter der 80er, der Flip-Flop der 90er und der E-Tec-Baustein des Millenniums das gleiche wie der moderne TX Controller: Sie messen einen analogen Eingangswert von einem Sensor oder Schalter und geben einen Ausgangszustand weiter, mit dem sich Lampen oder Motoren steuern lassen.

Analoge Daten werden von Sensoren empfangen und in digitale *High-* und *Low-*Werte umgewandelt. Um ein Oszillieren, also das schnelle, ungewollte Umschalten zwischen ‚Ein‘ und ‚Aus‘ zu vermeiden, sind Ein- und Aus-Pegel einzeln veränderbar: früher via Poti, heute programmierbar. Man spricht von einem Schmitt-Trigger, der einen Bereich hat, in dem er nicht den Zustand ändert – dieser wird Hysterese genannt (Abb. 5).

Mit den Ausgängen konnte man Lampen und Motoren steuern, wobei die Ausgangsleistung über die Spannung, entweder über das Netzgerät oder bei der Leistungsendstufe über Transistoren, geregelt werden konnte – zusätzlich zur Polarität, die über die Beschaltung vorbestimmt war.

Das funktioniert beim TX Controller ähnlich. Die Ansteuerung der Ausgänge erfolgt nicht über ein Relais, sondern über

Transistoren. Die Leistung wird nicht über die Versorgungsspannung, sondern über das schnelle Ein-/Aus-Pulsen gesteuert. Der Effekt ist der gleiche: Es lässt sich eine Last ein- und ausschalten. Der Unterschied beim TX Controller ist, dass die Leistungsabgabe an die Last nicht nur voreingestellt werden kann, sondern auch während des Programmablaufes veränderbar und sogar die Polarität dort vertauschbar ist.

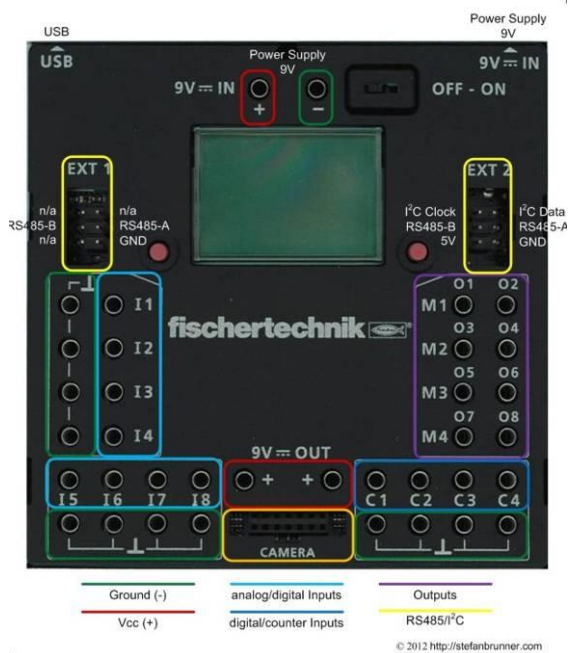


Abb. 2: Ein- und Ausgänge des TX Controllers

Wir können also einen analogen Wert messen, bestimmen, ob etwas ein- oder ausgeschaltet werden soll, und dann eine Last, deren Leistungsabgabe und die Polarität schalten.

Fehlt nur noch die Verknüpfung zwischen ‚Ein‘ und ‚Aus‘. Historisch waren das bei fischartechnik der Flip-Flop, der Mono-Flop und die Logikoperatoren AND, OR und NOT – die logischen Grundoperatoren.

Das alles kann der TX Controller auch. Der große Unterschied ist, dass der TX Controller acht Ein- und acht Ausgänge besitzt, die mit einer beliebigen Anzahl von Software-programmierbaren Opera-

toren verknüpft werden können. Man kann sicher viele ROBO Pro-Programme mit Silberlingen nachbauen, nur bräuchte man dafür sehr viele solcher Bausteine, was bei den Liebhaberpreisen nicht nur ein Vermögen kosten, sondern auch ein unsägliches Kabelgewirr nach sich ziehen würde. Ganz schön viele Logikoperatoren wären zum Beispiel erforderlich, um einfache arithmetische Funktionen durchführen zu können, und jedes Speicherbit benötigte ein Flip-Flop. Zusätzlich zu den Grundoperatoren unterstützt der TX Controller auch höhere Programmabläufe wie Bedingungen und Schleifen.

Als ich das erste Mal den TX Controller in der Hand hatte, dachte ich: fantastisch, eine unendliche Zahl von Silberlingen – was für Möglichkeiten. Aber diesmal wollte ich zuerst genau wissen, was man wie anschließen kann.

Eingänge

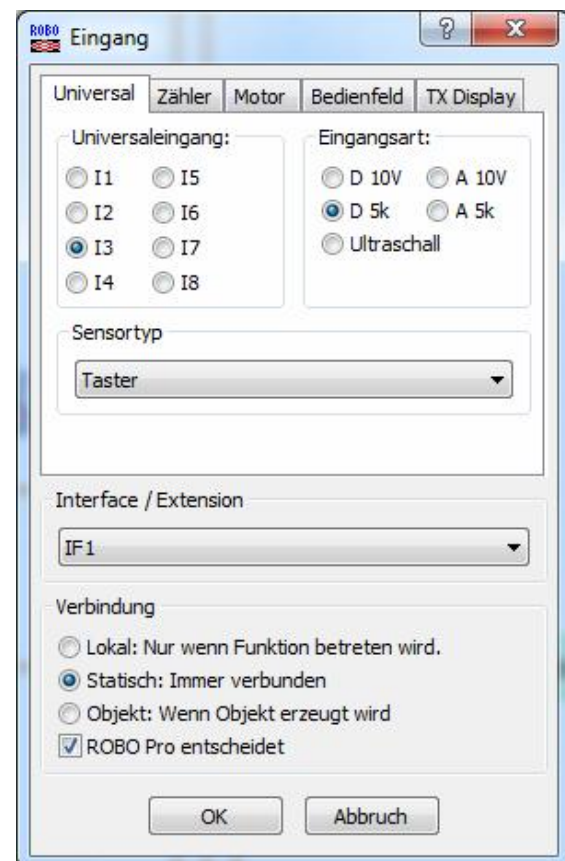


Abb. 3: 5k- und 10 V-Modus

Der TX Controller hat acht analoge Eingänge. Zusätzlich hat er auch noch vier digitale Eingänge, deren Hauptaufgabe es ist, die Achsenstellung der Encodermotoren zu erfassen.

Die Eingänge heißen Universaleingänge und messen die Spannung (mV) in Hardware wie ein Voltmeter. Das hört sich ziemlich einfach an. Aber warum gibt es in ROBO Pro so viel einzustellen? Diese Einstellungen haben nur eine Auswirkung auf die Software, nämlich wie diese die gemessene Spannung interpretiert. Und dies soll auch möglichst intuitiv zu den entsprechenden Sensoren passen, schließlich ist fischertechnik ja als Lernspielzeug gedacht.

Um es vorwegzunehmen: Die Einstellung auf *Digital* liefert einen booleschen Wert (1 oder 0) an die Software, wenn bestimmte Schwellenwerte für ‚Ein‘ bzw. ‚Aus‘ überschritten werden – genau wie bei den klassischen Bausteinen, wie oben beschrieben. Im Unterschied dazu liefert die Einstellung *Analog* einen variablen Wert, der entweder als Spannung ausgegeben oder in einen Widerstandswert umgerechnet wird. Da es keinen definierten Ein- oder Auszustand im *Analog*-Modus gibt, gibt es auch keine Hysterese, sondern nur einen diskreten Wert. Da die *Digital*-Einstellung nur eine Softwarefunktion ist, kann diese auch in ROBO Pro mit eigenen Schwellenwerten selbst zusammengebastelt werden. *Digital* hat also weniger mit der klassischen Verarbeitung von digitalen Signalen zu tun, sondern ist nur eine Programmierhilfe mit zu den Lichtschranken und Schaltern passenden, voreingestellten Schwellenwerten und Hysterese.

Bei beiden Einstellungen, *Analog* und *Digital*, sind die 10 V- und die 5k-Ohm-Einstellung nur verschiedene Interpretationen desselben Spannungswertes durch die Software. Und das funktioniert so (Abb. 4): Intern wird dem Eingang immer eine Spannung zugeführt. Aber über einen

internen Widerstand ist die Stromstärke stark begrenzt. Daher kann man den Wert mit einer extern angelegten Spannung unbedenklich bis zu 10 V selber überschreiben, oder über einen Widerstand auf Masse ziehen. Durch diese Dualität büßt der Eingang im unteren Bereich etwas Messgenauigkeit ein.

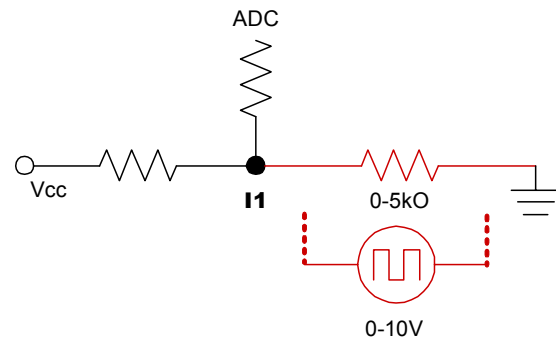


Abb. 4: 5k- und 10 V-Modus

In der 10V-Einstellung kann man Spannungen zwischen 75 mV und 10 V messen, mit einer Auflösung von 100 mV. Dabei wird ein Strom von nur 10 μ A gezogen. In der 5k-Einstellung liegt am Eingang ein Prüfstrom von 2,2 V und 60 μ A an, und es können Widerstandswerte von 50 Ω bis 5 k Ω mit einer Auflösung von 1 Ω gemessen werden. Diese Eingänge sind also nicht so empfindlich wie die des h4-GB oder des Schwellenwertschalters (Tab. 1). Für den Feuchtigkeitssensor z. B. würde es nicht reichen, aber man könnte natürlich den h4-GB, den Schwellenwertschalter oder die Leistungsendstufe als Messvorverstärker einsetzen.

Hysterese	Min	Max
Universal 10V	75 mV	10 V
Universal 5k	50 Ω	5 kΩ

Tab. 1: Min- und Max-Werte

Dasselbe gilt für den *Analog*- und den *Digital*-Modus. Nur im *Digital*-Modus werden die Analogwerte von der Software in digitale 1- (*High*) oder 0- (*Low*) Werte umgerechnet. Der Wert in der Software ist im *Analog*-Modus vom Typ Integer (nicht

Fließkomma), und im *Digital*-Modus ein boolescher Wert (1 oder 0). Liegen in der 10 V-Einstellung am Eingang weniger als 540 mV an, liefert das den Zustand *Low*, oberhalb 840 mV den Zustand *High*, mit einer Hysterese von 300 mV: Für Werte von 540 mV bis 840 mV ändert sich der Zustand nicht.

In der 5k-Einstellung wird der Eingang auf Masse gezogen, genauso wie im *Analog*-Modus. Ein Wert über 725 Ω ist *Low*, und ein Wert unter 435 Ω ist *High*. Dazwischen gibt es wieder die Hysterese, in der sich der Zustand nicht ändert (Tab. 2).

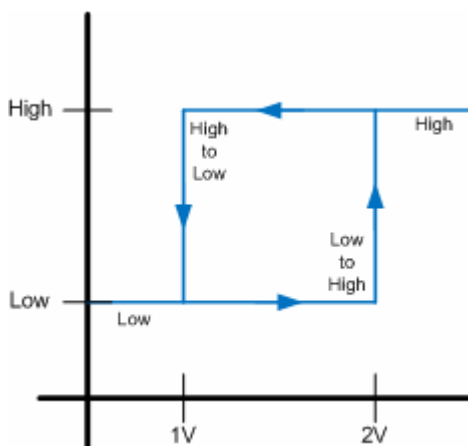


Abb. 5: Hysterese

Wie bereits erwähnt sind die Zähler-Eingänge getrennt zu betrachten. Dies sind echte digital Eingänge, die intern direkt am Prozessor angeschlossen sind. Daher sind hier auch die Schwellenwerte andere. Wenn man diese Eingänge nicht mit den Encodermotoren einsetzt, kann man sie auch als weitere Sensor-Eingänge nutzen. In der 10V-Einstellung ist in etwa alles über 2 V *High* und unter 1 V *Low*. In der 5k-Einstellung ist über 630 Ω *Low* und unter 1.520 Ω *High*. Der Prüfstrom ist mit 5,5 V und 1,8 mA höher als bei den Universaleingängen. Der Unterschied zwischen *Zähler*- und *Digital*-Einstellung liegt darin, dass der Software mitgeteilt wird, ob bei jedem *Low/High*-Übergang eine Eins addiert werden soll, oder nur der Zustand gewechselt wird. Damit ändert sich natürlich auch der Datentyp von Integer zu

Boolean. Im *Zähler*-Modus wird ein langes *High* als *Reset* interpretiert und der Wert auf Null zurückgesetzt.

Hysterese	Low	High
Universal 10V	0,54 V	0,84 V
Universal 5k	725 Ω	435 Ω
Zähler 10V	1 V	2 V
Zähler 5k	1.520 Ω	630 Ω

Tab. 2: Schwellenwerte im *Digital*-Modus

Im Normalfall schließt man an die Zähler-eingänge den Achsenwinkel-Sensor des Encodermotors an, der über ein Getriebe 75 Impulse pro Umdrehung zählt, also eine Auflösung von etwa vier Grad hat (Abb. 6). Das ist nicht ganz die Genauigkeit eines Schrittmotors – dafür braucht man auch keine bis zu acht Ausgänge, und Encodermotoren sind billiger.



Abb. 6: Hall-Sensor am Encodermotor

Nun stellt sich euch vielleicht die Frage: Kann ich mein Arduino-Board oder den IC-Digitalbaustein (oder Ähnliches) an diese universellen Digitaleingänge anschließen? Die Antwort: Im Prinzip ja. Aber das klappt nur in der *Digital*-Einstellung, wenn die Schwellenwerte kompatibel sind. Ansonsten muss man den Eingang auf *Analog* stellen und sich seine eigenen Schwellenwerte programmieren, oder mit ein paar Widerständen erden.

Besonders der relativ niedrige *High*-Wert könnte Probleme bereiten. Man sollte den Eingang in jedem Fall auf die 10 V-Ein-

stellung setzen. Die Schaltschwelle um 0,5 V herum sollte mit vielen TTL- und CMOS-Bausteinen funktionieren, selbst mit 3,3 V-High-Pegeln (Tab. 3). Das könnte etwas einfacher sein mit den Zähler-Eingängen im *Digital-Modus*.

Ausgang	Low	High
TTL	0,35 V	3,3 V
CMOS	0,2 V	4,7 V
LowV	0,4 V	2,4 V

Tab. 3: Schwellen gängiger Chip-Familien

Der Sensortyp in ROBO Pro ist nur eine Hilfe, um den Eingang für den jeweiligen Sensor und den typischen Gebrauch richtig einzustellen. Der fortgeschrittene Benutzer kann natürlich den Eingang selbst setzen, und je nach Beschaltung sind mehrere Kombinationen möglich.

Die Eingänge sollten recht robust gegen Fehlschaltung sein. Dennoch ist dringend angeraten, Optokoppler einzusetzen, wenn man andere Geräte als fischertechnik integrieren will, besonders wenn diese hohe interne Spannungen haben wie z. B. Radiosender (Amateurfunk oder WLAN), Verstärker, R/C-Steuerungen oder sogar Röhrengeräte – selbst die beliebten Nixie-Röhren arbeiten mit 180 V.

Man beachte dabei, dass Spannung immer ein relativer Wert ist – und sogar zwei Masseleitungen einen gewaltigen Potentialunterschied aufweisen können, wenn sie nicht an dieselbe Erdung angeschlossen sind, besonders bei den kleinen Strömen, die die Eingänge aufnehmen.

Ausgänge

Die Ausgänge sind keine Relais, sondern acht individuelle MOSFET-Leistungstreiber, die sich als vier H-Bridges verschalten lassen. Die H-Bridges ermöglichen das programmierbare Umschalten der Polarität (später mehr dazu). Man kann jedenfalls entweder acht Lasten mit vorbestimmter

Polarität anschließen, oder vier Lasten mit programmierbarer Polarität oder eine Kombination aus beidem.

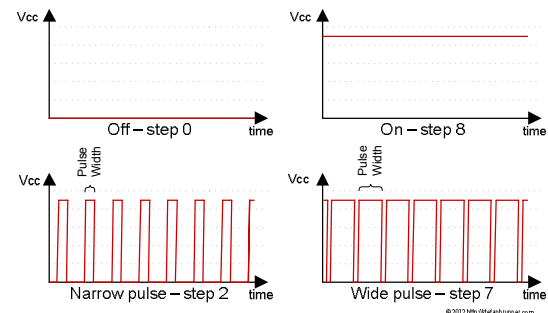


Abb. 7: Pulse Width Modulation (PWM)

Ein MOSFET ist ein Transistor, ähnlich wie die Transistoren in der Leistungsendstufe. Ein bedeutender Unterschied ist jedoch, dass die MOSFETs in Sättigung betrieben werden. Das heißt, der Transistor schaltet entweder durch – oder gar nicht. Bei den bipolaren TIPs in der Leistungsendstufe kann mit dem Eingangsstrom der Ausgangsstrom bestimmt werden. Das geht mit den MOSFETs im TX Controller allerdings nicht – aus gutem Grund: Die Leistungsaufnahme ist das Produkt aus Spannung und Strom, was wiederum vom Innenwiderstand des Verbrauchers abhängig ist. Der gleiche Strom, der zwischen Basis und Emitter fließt, kann ganz verschiedene Auswirkungen auf die Stromstärke der Last und daher der Leistungsaufnahme haben, abhängig von der Spannung, die an der Last anliegt, und dem Innenwiderstand.

Zudem sind die Fertigungstoleranzen bei Transistoren gewaltig. Dazu noch sind Transistorkennlinien nicht linear, abhängig von Spannung, Stromstärke und Temperatur. Das alles passt nicht in die digitale, deterministische Welt der *Highs* und *Lows*.

Daher hat man sich etwas ganz Geniales ausgedacht: die *Pulse Width Modulation* (PWM). Die Leistungsaufnahme der Last wird durch sehr schnelles Ein- und Ausschalten kontrolliert, wobei die Leistungsaufnahme direkt vom Verhältnis zwischen

An- und Auszeit abhängig ist. Das nennt sich der *Duty Factor* (Abb. 7), also der Teil des Signals, das als Leistung an dem Verbraucher zur Verfügung steht. Im Prinzip ist das eine Wechselspannung mit Rechtecksignal, die auf die Gleichspannung mit halber Amplitude aufmoduliert worden ist – mit anderen Worten: das Signal wechselt nicht von Plus nach Minus, sondern zwischen 0 V und Plus. Wenn wir daher den Ausgang auf 50 % Leistung programmieren, nimmt die Last auch nur 50 % Leistung auf, egal wie hoch die Versorgungsspannung oder deren Innenwiderstand ist.

PWM-Steuerungen sind Stand der Technik und auch der Grund, warum die Motoren in der U-Bahn summen, wenn sie anfährt. Die PWM-Frequenz liegt hier bei 210 Hz, daher ist das Summen der Motoren bei weniger als der vollen Leistung auch beim TX Controller deutlich zu hören.

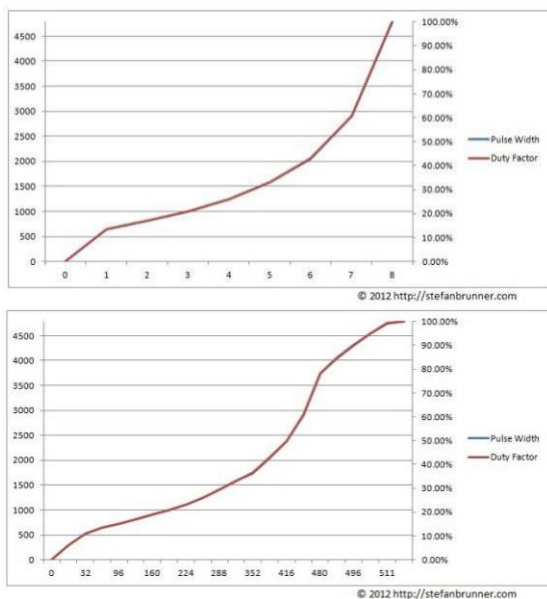


Abb. 8: 8-Schritte (oben), 512-Schritte (unten)

In ROBO Pro kann man nun die Leistung in acht oder 512 Stufen steuern (Abb. 8). Überraschend ist, dass zum einen die Stufeneinteilung nicht linear ist, und zum anderen der Kurvenverlauf der beiden Stufen sich unterscheidet. Ich stellte fest, dass die unteren Stufen im 8-Stufen-

Schema gar nicht mit den Motoren funktionieren. Im 512-Stufen-Schema gibt es im unteren und oberen Bereich mehr Granularität und mehr Steilheit in der Mitte; das ist im 8-Stufen-Schema genau umgekehrt.

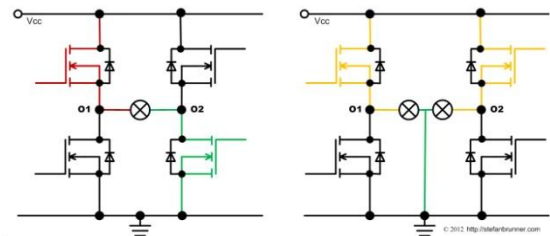


Abb. 9: Single Bridge

Oben habe ich die *H-Bridges* erwähnt. Ihr mögt euch vielleicht schon gewundert haben, warum man eine Last z. B. zwischen O1 und Masse, aber auch zwischen O1 und O2 anschließen kann. Der Grund: Eine *H-Bridge* hat vier MOSFET-Schalter, jeder Ausgang hat zwei. Zwei Schalter sind jeweils miteinander verbunden und an jeder Seite mit Plus und Minus (Masse) verbunden, sodass, wenn beide Schalter geschlossen würden, Minus und Plus überbrückt würden (daher *Bridge*), und es zu einem Kurzschluss käme. Im Nullzustand ist immer der Schalter, der mit Masse verbunden ist, geschlossen. Daher sind alle Ausgänge im Urzustand auf Masse gelegt. Wenn man jetzt den Ausgang einschaltet, sperrt der Schalter, der mit Masse verbunden ist, und der andere schließt (Abb. 9). Daher kann man eine Last genauso mit einem Pol an Masse als auch mit einem anderen Ausgang, der auf Null steht, verbinden, während der andere Pol an einem Ausgang hängt, der gesteuert wird.

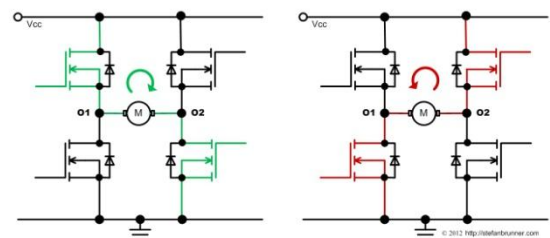


Abb. 10: H-Bridge

Beim Verdrahten sieht es vielleicht etwas schöner aus, wenn man nicht mehr als vier Ausgänge braucht. Genauso kann man aber auch die Polarität ändern, also die Drehrichtung eines Motors. In dem Fall schließt man den Motor zwischen zwei Ausgängen an und regelt dann entweder den einen oder den anderen. Man verdrahtet also zwei Brücken (Abb. 10) zu einem H (daher *H-Bridge*). Während in ROBO Pro alle Eingänge gleich sind und per Einstellung verändert werden können, gibt es zwei verschiedene Objekte für die Ausgänge: Das eine steuert die acht Ausgänge unabhängig voneinander, bei dem anderen werden immer zwei Ausgänge zusammen-gesteuert, paarweise O1 und O2, O3 und O4 usw., genannt M1, M2, etc. (Abb. 12).

Die Drehrichtung kann dann vom Programm gesteuert werden, wobei ROBO Pro automatisch entscheidet, welcher der beiden Ausgänge einzuschalten, also auf Plus zu schalten ist. Das gleiche lässt sich natürlich auch weniger elegant mit dem Objekt für acht Ausgänge steuern (Abb. 11), nur muss man dann per Software entweder den einen oder den anderen Ausgang einzeln schalten. Schaltet man beide, sind beide Pole mit Plus verbunden und es fließt kein Strom, dasselbe gilt, wenn beide auf Null sind. Man kann natürlich auch einen Motor zwischen einem Ausgang und Masse betreiben. Das geht genauso. Nur ist dann die Drehrichtung durch die Verdrahtung vorbestimmt.

Außerdem muss man darauf achten, dass man genug Strom hat für alle Motoren. Ein M-Motor kann unter Last locker 1 A ziehen, aber der TX Controller ist nur auf die Bedienung von maximal 2 A ausgelegt. Das wird mit vier Motoren schon knapp, und noch knapper mit acht. Zerstören sollte man den TX Dank des eingebauten Überhitzungsschutzes nicht können. Und natürlich ist die Stromaufnahme auch von der Belastung des Motors abhängig.

Das ‚Bild‘, das man in ROBO Pro unter den Ausgangsobjekten wählen kann (Abb. 11), hat keinen Einfluss darauf, wie der Ausgang funktioniert – ob in *H-Bridge* oder *Single Bridge* – und welchen Aktor man tatsächlich anschließt.

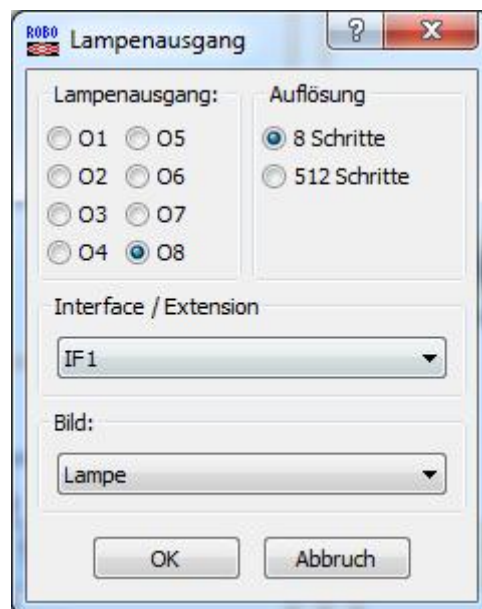


Abb. 11: Lampenausgang

Die MOSFETs sind sehr tolerant gegenüber induktiven Lasten wie den Pneumatikmagnetventilen, mögen jedoch keine Lasten mit Wechselstromfiltern, wie bestimmte Filter zur Funkentstörung, weil dies mit der PWM-Lastkontrolle im Konflikt steht. MOSFETs reagieren sehr empfindlich gegen Überspannung selbst bei sehr kleinen Strömen, dazu gibt es aber auf dem Treiber-IC eine Schutzschaltung.

Ansonsten sind die Treiber-ICs auch gegen Überlastung geschützt. Sollte der TX Controller plötzlich in den *Reset* oder gar ganz ausgehen, dann liegt das höchstwahrscheinlich daran, dass die Leistungsaufnahme der Last höher ist als die Stromquelle, die auch die CPU mit Strom versorgt, hergeben kann. Dann ist ein stärkeres Netzteil mit 2 A oder eine frisch aufgeladene Batterie zu empfehlen.

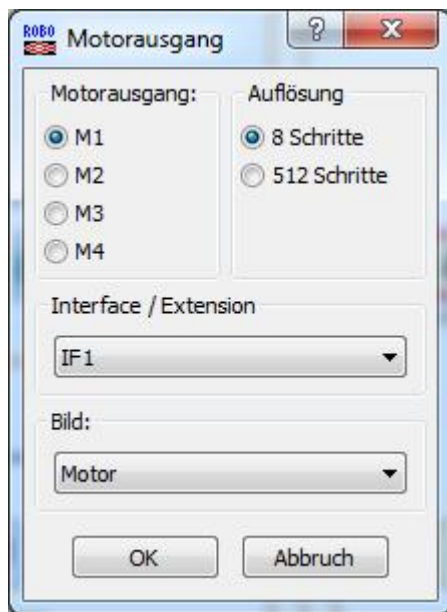


Abb. 12: Motorausgang

Schlusswort

Man kann natürlich noch so viel mehr machen mit dem TX Controller, und es gibt noch so viel mehr an der Hardware zu erkunden. Wie wird z.B. der Ultraschall-Sensor angesteuert? Oder wie funktioniert das mit dem I²C-Bus? Was genau muss

man beachten, um den TX Controller mit den alten Silberlingen zu verbinden (Stichwort ‚negative Logik‘)? Oder wie steuert man mehr als acht Lasten mit mehr als acht Sensoren, ohne dafür mehrere TX Controller zu verketteten?

Ziel dieses Beitrags war es jedoch, zunächst einen Überblick zu schaffen und darzustellen, dass man so ziemlich alles an die Ein- oder Ausgänge des TX Controllers anschließen kann, unabhängig davon, was im fischertechnik Computing-Programm dokumentiert ist. Ich wollte auch erklären wie ROBO Pro die Programmierung vereinfacht und die Hardware abstrahiert.

Alles Weitere ist Stoff für einen neuen Beitrag ...

Nun aber erstmal viel Spaß beim Experimentieren und Basteln.

Quellennachweis

- [1] Brunner, Stefan: [Hacking the fischertechnik ROBO TX Controller](#), 2011.

Computing

I²C mit dem TX – Teil 2: LED-Display

Dirk Fox

Seit dem Firmware-Update vom April 2012 auf Version 1.30 lassen sich am TX Controller I²C-Komponenten anschließen und mit Robo Pro Version 3.1.3 ansteuern. Nach der Einführung in die Grundlagen des I²C-Protokolls [1] wird in diesem Beitrag die Nutzung eines LED-Displays am TX vorgestellt.

Vorbemerkung

Die korrekte Ansteuerung von I²C-Komponenten – welche Befehle sie beherrschen, welche Daten sie übermitteln bzw. an sie übermittelt werden müssen und wie sie konfiguriert werden – kann den Datenblättern der darin verbauten I²C-fähigen ICs entnommen werden. Dort sind die elementaren Befehle meist tabellarisch dokumentiert, die in Robo Pro mit den Kommandos „I²C Schreiben“ und „I²C Lesen“ an die Komponente zu schicken sind.

Die in den Datenblättern beschriebenen Befehle steuern den IC auf „unterster“ Ebene. Um die Kernfunktionen der I²C-Komponente mit nur einem Aufruf ansteuern zu können, muss man aus diesen elementaren Befehlen zunächst komplexere Funktionen (Reset/Initialize, Set/Write, Get/Read, etc.) bilden. Für 18 verschiedene handelsübliche I²C-Komponenten wurden mit Version 3.1.3 von Robo Pro solche Bibliotheksfunktionen ausgeliefert. Viele davon verdanken wir [Rei Vilo](#); inzwischen wurden weitere im Download-Bereich der ft:c ([Rubrik „Robo Pro“](#)) veröffentlicht.

Zum besseren Verständnis der Funktionsweise erläutern wir in dieser und weiteren Folgen des Beitrags die Ansteuerung einiger ausgewählter Komponenten, die sich – aus verschiedenen Gründen – besonders für die Nutzung am TX eignen.

LED-Display

Von [Conrad Electronic](#) wird unter der [Bestellnummer 198344](#) ein vierstelliges LED-Display (*Light Emitting Diode*) mit einem I²C-Treiberbaustein vertrieben (Abb. 1), das seit Februar 1991 von Philips Semiconductors unter der Produktbezeichnung SAA1064 hergestellt wird [2]. Die vergleichsweise geringe Übertragungsgeschwindigkeit von 100 kHz (*Standard Mode*) ist dem Alter des Bausteins zuzuschreiben: Er erschien ein Jahr vor der Veröffentlichung von Version 1 des I²C-Standards, in dem der *Fast Mode* (400 kHz) eingeführt wurde.



Abb. 1: LED-Display SAA1064 mit Adress-Jumper (Bild: www.conrad.de)

Das Display erfordert eine Betriebsspannung zwischen 4,5 und 15 V (empfohlen: 5 V) und wird mit einem 6poligen Wan-

nenstecker mit der vom TX benötigten Anschlussbelegung geliefert. Es kann daher mit dem Flachbandkabel, das für den Anschluss weiterer TX Controller jedem TX beiliegt, ohne Zuhilfenahme eines Löt-kolbens an den EXP 2-Port des TX angeschlossen werden.

In der Bibliothek der Version 3.1.3 von Robo Pro finden sich bereits I²C-Treiber für dieses Display („LED-SAA1064“). Da es jedoch ein gutes Beispiel ist, um die I²C-Programmierung in Robo Pro zu verstehen und die fertigen Treiber außerdem nur die Ausgabe von vierstelligen Dezimalzahlen auf einem Display mit der Default-Adresse erlauben, erläutern wir im Folgenden die Ansteuerung im Detail.

Adressierung

Die höchstwertigen fünf Bit der *Slave*-Adresse des LED-Displays sind vom Hersteller festgelegt; zwei Bits können über einen Jumper am *Device* eingestellt werden (01110xx). Damit lassen sich bis zu vier LED-Displays kollisionsfrei über denselben I²C-Bus ansteuern – das Display kann so z. B. auf eine Anzeige mit 16 Stellen erweitert werden, oder die Displays werden für die Anzeige der Punktstände mehrerer Spieler (z. B. die Rundenzähler einer Carrera-Bahn oder die Gewinnpunkte am Flipper) verwendet.



Abb. 2: I²C-Verteilermodul
(Bild: www.conrad.de)

Wer über fünf der sechsadrigen TX-Flachbandkabel (bei Knobloch unter der [Artikelnummer 135465](#) für 2,50 € erhältlich) verfügt, kann die Displays sehr einfach mit Hilfe des (passiven) I²C-Verteilermoduls

(siehe Abb. 2), das Conrad Electronic unter der [Bestellnummer 191193-62](#) für knapp 7 € anbietet, mit dem TX verbinden.

Das LED-Display ist auf die Adresse 0111 000 (0x38) vorkonfiguriert. Durch Versetzen des Adress-Jumpers (Abb. 1) können die Adressen 0x39, 0x3a und 0x3b eingestellt werden.

Status-Abfrage

Das LED-Display besitzt ein auslesbares *Status Byte*, in dem das höchstwertige Bit (*Power Reset Flag*) gesetzt ist, wenn die Stromzufuhr unterbrochen wurde.

Zum Auslesen dieses *Status Byte* sendet der TX an das Display einen *Read*-Befehl – das Adress-Byte mit Read-Bit (Default: 0111 0001 = 0x71) – und erhält daraufhin vom *Device* das *Status Byte* übermittelt:

Byte	binär	hex.	Funktion
1	0111 0001	0x71	Adresse
2	?000 0000	0x?0	<i>Status</i>

Tab. 1: Auslesen des *Status Byte*

Nach dem Anstecken („Einschalten“) des LED-Displays oder einer Stromunterbrechung liefert das *Device* den Status-Wert 1000 0000 (= 0x80 oder 128). In Robo Pro sieht der Lesebefehl wie folgt aus (Abb. 3):



Abb. 3: Auslesen des *Status Byte* in Robo Pro

Nach dem Auslesen setzt das Display das *Status Byte* auf den Wert 0 zurück.

Anzeige

Die vierstellige Sieben-Segment-Anzeige (mit Dezimalpunkt) muss vom TX ziffernweise angesteuert werden. Dazu wird nach dem Adress-Byte mit *Write*-Bit (0111 0xx0 = 0x70, 0x72, 0x74 oder 0x76, je nach konfigurierter Adresse) ein *Instruction Byte* übermittelt, das einen 3-Bit-Befehlscode (*Subaddress*) enthält. Damit legt der *Master* fest, ob das folgende Datenbyte in die *Control Register* (C0 bis C6) oder in eine der vier Sieben-Segment-Anzeigen geschrieben werden soll.

Hat die Subadresse den Wert 0x00, dann können dem Byte-Wert für die *Control Register* vier weitere Daten-Bytes folgen, die dann automatisch (von rechts nach links, also beginnend mit der niederstwertigen Ziffer) in die vier Sieben-Segment-Anzeigen geschrieben werden.

Instruction Byte

Das *Instruction Byte* beginnt mit fünf führenden Nullen, gefolgt von einem 3-Bit-Befehlscode:

binär	hex.	Bedeutung
0000 0000	0x00	<i>Control Register</i>
0000 0001	0x01	1. Anzeige (links)
0000 0010	0x02	2. Anzeige
0000 0011	0x03	3. Anzeige
0000 0100	0x04	4. Anzeige

Tab. 2: *Instruction Byte*

Alle anderen möglichen Bitkombinationen sind nicht belegt. Das *Instruction Byte* legt damit fest, wohin die im nächsten Datenbyte folgenden Bits geschrieben werden sollen.

Control Register

In dem sieben Bit großen *Control Register* werden Eigenschaften des LED-Displays eingestellt: Über die Werte der Registerbits C4 bis C6 lässt sich die Helligkeit der 7-

Segment-Anzeige in acht Stufen von 0-21 mA regeln. Die Helligkeit wird dabei für alle Segmente zugleich eingestellt.

Reg.	Bedeutung
C0	Betriebsart <i>static</i> (= 0) bzw. <i>dynamic</i> (= 1): wechselnde Anzeige der Stellen 1+3 / 2+4.
C1	Löschen Anzeigen 1+3 (= 0)
C2	Löschen Anzeigen 2+4 (= 0)
C3	Test aller Segmente (= 1)
C4	+3 mA Anzeigestrom (= 1)
C5	+6 mA
C6	+12 mA
C7	<i>Nicht genutzt</i>

Tab. 3: *Inhalt des Control-Register*

Ein Test aller Segmente mit maximaler Anzegehelligkeit erfordert demnach die Übermittlung der folgenden drei Bytes an das Display (Tab. 4, Abb. 4):

Byte	Binär	Hex.	Funktion
1	0111 0000	0x70	Adresse, <i>Write</i> -Bit
2	0000 0000	0x00	<i>Cont. Reg.</i>
3	0111 1111	0x7F	Test

Tab. 4: *Funktionstest Display*



Abb. 4: *Funktionstest-Befehl in Robo Pro*

Segment-Codierung

Die vier Sieben-Segment-Anzeigen werden durch je einen Byte-Wert kodiert: Jedes Bit steht für ein Segment (Abb. 5); das höchstwertige Bit für den Dezimalpunkt. Ist ein Daten-Bit = 1, wird das entsprechende Segment aktiviert.

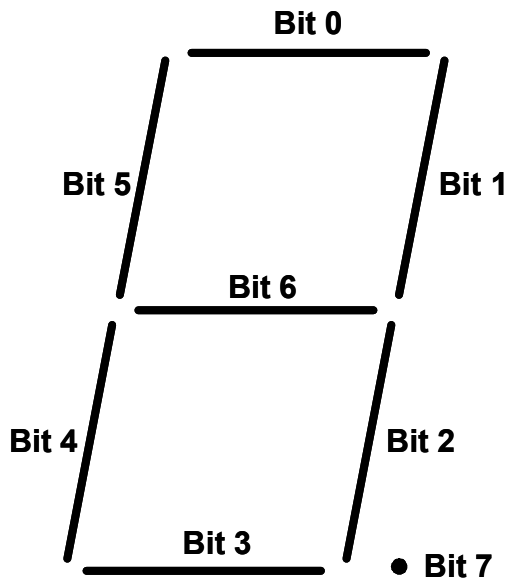


Abb. 5: Kodierung der Segmentanzeige

Damit lassen sich die Ziffern 0-9 elegant über eine Tabelle (Tab. 5) in eine LED-Kodierung überführen:

Ziffer	binär	hex.	dezimal
0	0011 1111	0x3f	63
1	0000 0110	0x06	6
2	0101 1011	0x5b	91
3	0100 1111	0x4f	79
4	0110 0110	0x66	102
5	0110 1101	0x6d	109
6	0111 1101	0x7d	125
7	0000 0111	0x07	7
8	0111 1111	0x7f	127
9	0110 1111	0x6f	111

Tab. 5: Ziffern-Kodierung

Soll der Dezimalpunkt gesetzt werden, muss der Wert vor der Übermittlung an das I²C-Device mit 0x80 bitweise ODER-verknüpft werden. Auch ein Minus-Zeichen lässt sich anzeigen – dafür ist nur Bit 6 zu setzen (0100 0000 = 0x40).

Treiber

Bei der Gestaltung der Treiber-Software ist zunächst festzulegen, wie viele Eigenschaften und Optionen des LED-Displays vor einem Programmierer „verborgen“ werden sollen.

Sicherlich sollte die Ziffernkodierung durch den Treiber erfolgen – das geht sehr einfach über ein mit den Werten aus Tab. 5 vorbelegtes Listen-Element (Abb. 6):

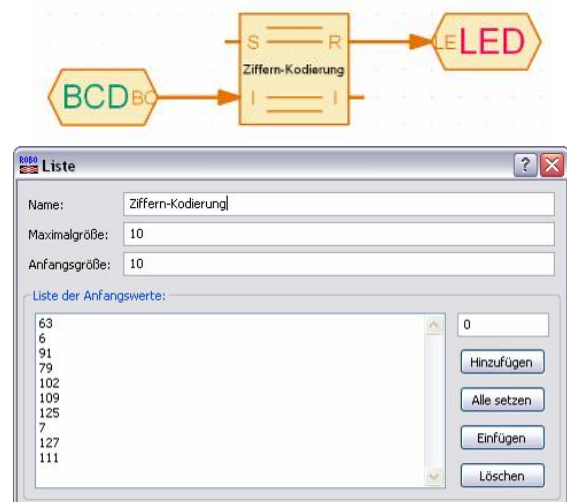


Abb. 6: Ziffern-Kodierung über Listen-Element (BCD2LED)

Für die Umrechnung einer Dezimalzahl in BCD-Darstellung kann ebenfalls eine einfache Funktion bemüht werden:

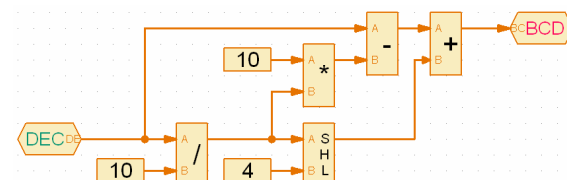


Abb. 7: Konvertierung einer Dezimalzahl in BCD-Darstellung (DEC2BCD)

Die Helligkeit der Anzeige sollte als Parameter übergeben werden.

Aber muss der Programmierer angeben, an welcher Stelle der Dezimalpunkt erscheinen soll, oder stattdessen man den Treiber besser mit verschiedenen Funktionen für unterschiedliche Zahlenarten (Ganzzahl, Fließkommazahl, Jahreszahl etc.) aus? Sollen führende Nullen angezeigt werden? Sind negative Zahlen zur Darstellung zugelassen?

Vergleichsweise leicht ist es, eine Jahreszahl darzustellen – kein Dezimalpunkt, keine führenden Nullen. Übergeben werden dazu beispielsweise das Jahr als zwei-stellige BCD-Zahl und die Helligkeit der Anzeige (0-7):

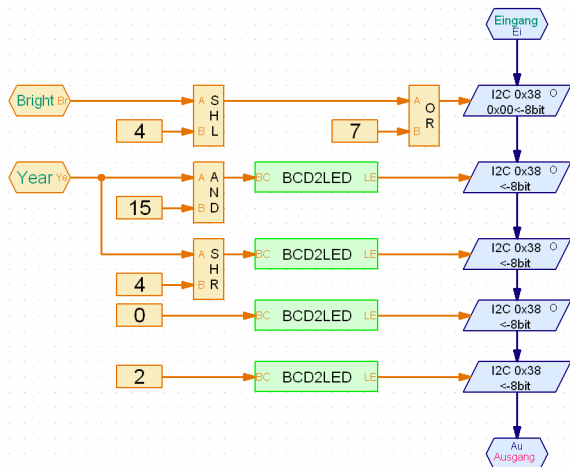


Abb. 8: Anzeige Jahreszahl

Mit drei LED-Displays und geringen Anpassungen des Unterprogramms aus Abb. 8 lässt sich so die ft-Funkuhr aus ft:pedia 3/2012 [3] um eine 12-stellige LED-Zeit-anzeige erweitern (Abb. 10).¹

Leider muss man die Geräteadresse in den Eigenschaften des I²C-Schreib- bzw. Lesebefehls von Robo Pro festlegen; sie kann nicht als Parameter an den Befehl übergeben werden. Bei der Ansteuerung mehrerer LED-Displays mit unterschiedlichen Adressen müssen die Befehle daher einzeln an die Adressen angepasst werden.

¹ Das erweiterte Robo Pro-Programm der ft-Funkuhr mit drei LED-Displays findet sich im [Downloadbereich der ft:community](#).

Herausfordernder ist es, eine (möglicherweise negative) Fließkommazahl ohne führende Nullen anzuzeigen. Dazu ist die Zahl zunächst auf ein passendes Format (ggf. Minuszeichen, Ziffern vor- und nach dem Dezimalpunkt) zu normieren – das sei euch als kleine „Fingerübung“ überlassen.

Leuchtschrift

In Grenzen kann man mit den 7-Segment-Displays auch Texte anzeigen. Allerdings lassen sich nur 14 Buchstaben des Alphabets gut erkennbar darstellen (Tab. 6).

	binär	hex.	dez.
A	0111 0110	0x76	119
B	0111 1111	0x7f	127
C	0011 1001	0x39	57
E	0111 1001	0x79	121
F	0111 0001	0x71	113
H	0111 0110	0x76	118
I	0011 0000	0x30	48
J	0111 1000	0x78	120
L	0011 1000	0x38	56
O	0011 1111	0x3f	63
P	0111 0011	0x73	115
S	0110 1101	0x6d	109
U	0011 1110	0x3e	62
Z	1101 1010	0xda	218

Tab. 6: Buchstaben-Kodierung

Einige simple Wörter lassen sich damit immerhin „schreiben“, wie ALLE, PAPI, BALL, LOCH, FACH, SAAL, HOPP, ELCH, BUCH, PASS, HOCH, ESEL, LAUF, BASS, HUF, BACH, PULS, FALL, HEFE, HAUS, HOHL, BOJE, FASS, HALS, LUPE, AFFE, LAUS, ZAHL oder PECH.

Damit lässt sich sogar eine einfache Laufschrift realisieren. Und wer in drei LED-Displays investiert, kann mit dem folgen-

den kleinen Programm immerhin versuchen, seinen Vater beim Marathon zu Höchstleistungen anzutreiben:

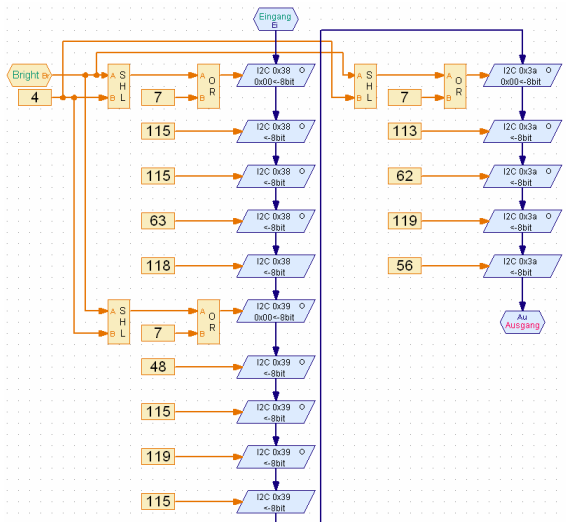


Abb. 9: Beispiel-Textausgabe

Noch zwei Tipps zum Schluss:

- Wenn ihr ein Programm schreibt, das auch durchlaufen soll, wenn kein LED-Display angeschlossen ist, dann müsst ihr in den I²C-Befehlen als Fehlerbehandlung „Sofort abbrechen“ wählen.
- Falls ihr mehrere LED-Displays ansteuert, solltet ihr darauf achten, dass sie nicht zu viel Strom ziehen – der TX ist auf insgesamt maximal 2 A ausgelegt. Gegebenenfalls müsst ihr die Helligkeit der Displays etwas reduzieren.

Quellen

- [1] Dirk Fox: *I²C mit dem TX – Teil 1: Grundlagen*. [ft.pedia 3/2012](https://ft.pedia.de/2012/03/32-37/), S. 32-37.
- [2] Philips Semiconductors: *SAA1064 – 4-digit LED-driver with I²C-Bus interface*. Data Sheet, Feb. 1991.
- [3] Dirk Fox, Dirk Ottensmeyer: *Bau einer ft-Funkuhr*. [ft.pedia 3/2012](https://ft.pedia.de/2012/03/4-10/), S. 4-10.

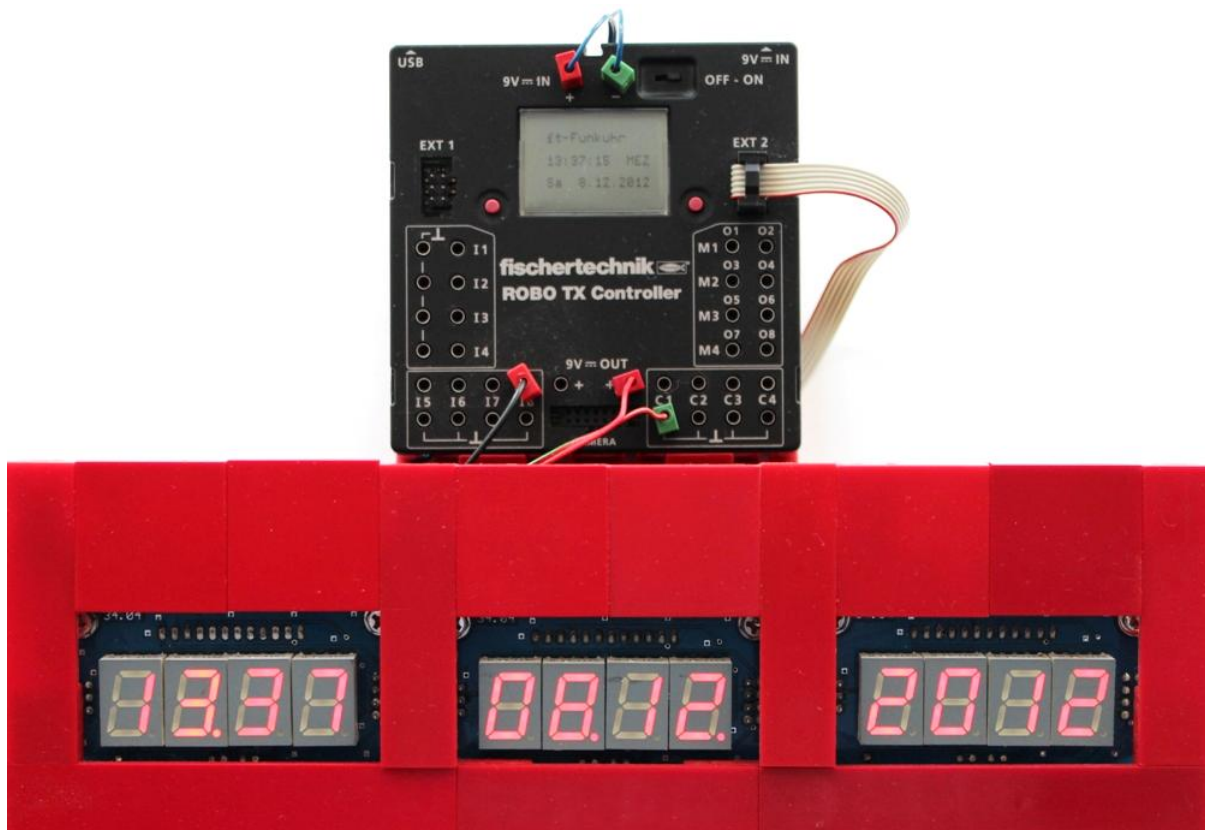


Abb. 10: LED-Anzeige der „ft-Funkuhr“ [3]

Antriebstechnik

Die Dampfmaschine

Dirk Fox

Die Entwicklung der Dampfmaschine vor genau 300 Jahren hat die Welt stärker verändert als jede andere technische Erfindung zuvor. Als Erfinder gilt allgemein James Watt – tatsächlich hat er sie lediglich (wenn auch in wichtigen Punkten) weiterentwickelt. Das Funktionsprinzip einer Dampfmaschine lässt sich unter Verwendung eines Pneumatik-Kolbens mit fischertechnik sehr anschaulich nachbilden.

Geschichte

Bis vor genau 300 Jahren waren Nutztiere praktisch die einzige bekannte Möglichkeit für den Menschen, Leistungen zu vollbringen, die die menschliche Arbeitskraft übersteigen. So finden sich unter den Skizzen von [Leonardo da Vinci](#) (1452-1519) einfache Maschinen, die die Kraft eines Pferds oder Rinds z. B. auf eine Eimerkette zum Wasserschöpfen oder einen primitiven Kran übertragen.

Lediglich für bestimmte Anwendungen wie das Mahlen von Getreide oder das Entwässern von Feldern (mittels [Archimedischer Schrauben](#)) wurden Wasser- und Windmühlen eingesetzt. Diese Antriebe waren aber an Wasserläufe und windreiche Gegenden gebunden. Damit war der technischen Entwicklung eine natürliche Grenze gesetzt: Die Menschheit verfügte über keine ortsunabhängig nutzbare Antriebskraft, die in der Lage gewesen wäre, kontinuierlich, zuverlässig und mit großer Kraft eine Maschine anzutreiben.

Das änderte sich grundlegend mit der Entwicklung der Dampfmaschine. Die erste funktionsfähige „Kolben-Wärme­kraftmaschine“ baute [Thomas Newcomen](#) (1663-1729) im Jahr 1712 für den Einsatz als Wasserpumpe in Bergwerken; damit wurden die Stollen frei von Grundwasser

gehalten. Genau genommen war die erste Dampfmaschine eine „Unterdruckmaschine“: Ein Zylinder wurde mit Wasserdampf gefüllt und dann durch die Zuführung von Wasser abgekühlt (Abb. 1, B).

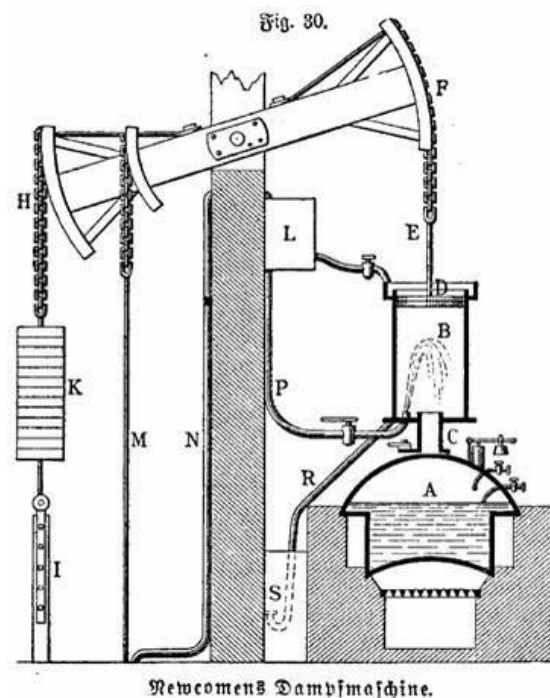


Abb. 1: Newcomens „Kolben-Wärme­kraftmaschine“

(Quelle: Meyers Konversationslexikon, 1890)

Der durch die Abkühlung des Dampfes entstehende Unterdruck zog einen Kolben (D) in den Zylinder. Anschließend zog ein Gegengewicht (K) den Kolben wieder hoch und mit ihm die nächste Dampf-

füllung in den Zylinder [1, 2]. Dieser Vorläufer der Dampfmaschine hatte lediglich einen energetischen Wirkungsgrad von etwa 0,5 % – wegen der Abkühlung des Zylinders musste man fast die 200fache Energie der Maschinenleistung für die Dampferzeugung aufwenden, eine wenig Ressourcen schonende Angelegenheit.

Der Schotte [James Watt](#) (1736-1819), der 1757 an der Universität von Glasgow eine Stelle als Instrumentenmacher angetreten hatte [3], erhielt 1764 ein Modell einer Dampfmaschine des von [Thomas Newcomen](#) entwickelten Typs zur Reparatur. Dabei kam ihm der Gedanke, den Kondensationsprozess auszulagern und den Kolben nicht durch Unterdruck, sondern direkt durch den Dampfdruck anzutreiben. Nachdem er in dem Eisenfabrikanten [John Roebuck](#) (1718-1794) einen Finanzier gefunden hatte, ließ er 1769 seine Dampfmaschine patentieren (Abb. 2) [4].

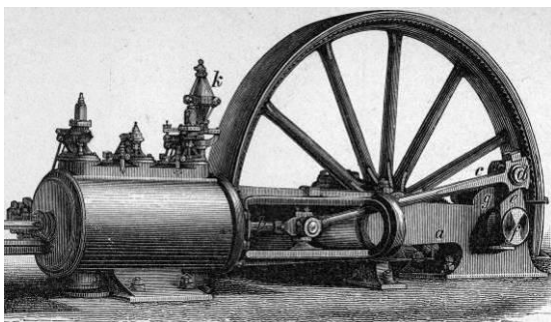


Abb. 2: Wattsche Dampfmaschine
(Quelle: Brockhaus' Konversations-Lexikon,
14. Auflage, Band 4, 1894)

Später gründete er mit dem Unternehmer [Matthew Boulton](#) (1728-1809) die Firma Boulton & Watt, deren von Watt entwickelten Dampfmaschinen die Industrialisierung der Welt einleiteten. Dieser Erfolg war mehreren Weiterentwicklungen Watts zu verdanken. 1781 führte er ein **Schubkurbelgetriebe** ein, mit dem er die Hubbewegung des Kolbens in eine Drehbewegung umwandelte. Damit war Newcomens Hebelarm („Balancier“) für die Rückführung des Kolbens nicht mehr erforderlich; er ersetzte ihn durch ein

großes Schwungrad. Um den Wirkungsgrad weiter zu erhöhen, leitete er 1782 den Dampf wechselseitig auf jeweils eine Seite des Zylinders.

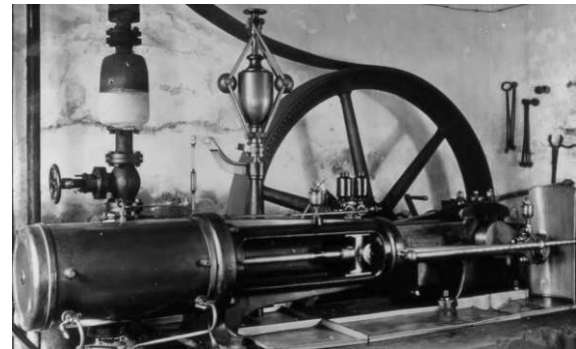


Abb. 3: Wattsche Dampfmaschine mit
Schwungrad und wechselseitiger Dampfzufuhr

Die Dampfzufuhr regelte er durch eine ausgeklügelte Mechanik, das **Wattsche Parallelogramm** (Abb. 3). Damit erreichte er einen Wirkungsgrad von 3 % – das Sechsfache der „Kolben-Wärme-Kraftmaschine“ Newcomens.

Für den maschinellen Antrieb von Fahrzeugen bedurfte es allerdings zahlreicher weiterer Verbesserungen, da Watts Dampfmaschinen dafür viel zu schwer waren.

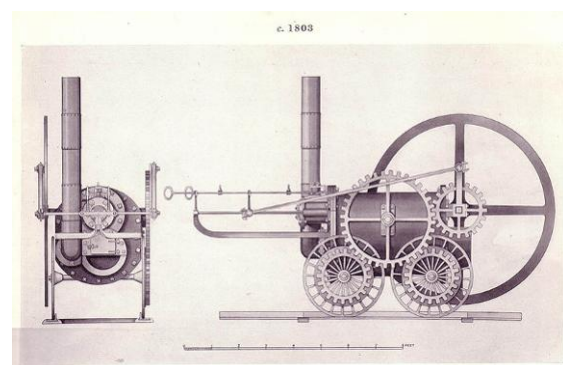


Abb. 4: Erste Dampflokomotive (Quelle:
The British Railway Locomotive, H.M.S.O.)

Die von [Richard Trevithick](#) (1771-1833) im Jahr 1802 entwickelten Hochdruck-Dampfmaschinen erhöhten den Wirkungsgrad weiter und verringerten zugleich das Gewicht der Maschine erheblich – Watt hatte Hochdruck-Dampfmaschinen Zeit seines Lebens für viel zu gefährlich gehalten. *Trevithick* gelang damit 1804 die

Konstruktion der ersten Lokomotive, die ohne Last eine Geschwindigkeit von ca. 25 km/h erreichte und fünf Tonnen Last über eine Strecke von 17,5 km zog (Abb. 4) [5].

Damit begann das Zeitalter der Eisenbahn – aber das ist eine andere Geschichte.

Ein „Druckluftmotor“

Zwar lässt sich mit fischertechnik keine echte Dampfmaschine konstruieren – klar, die Kunststoffteile würden eine Erhitzung auf mehr als 100°C nicht verzeihen. Trotzdem gibt es eine Möglichkeit, ein dem Prinzip der Dampfmaschine entsprechendes Funktionsmodell zu konstruieren: mit den Pneumatik-Zylindern 60 und Druckluft. Der Zylinder erlaubt eine beidseitige Einleitung von Druckluft; damit ließe sich ein der Wattschen Konstruktion ähnelnder „Druckluftmotor“ bauen.

Bereits im Festo-Pneumatik-Kasten von 1981 finden sich in der [Anleitung](#) mehrere Druckluftmotor-Modelle, darunter auch ein „liegender Luftmotor mit festem, doppelt wirkendem Zylinder“, der dem Prinzip einer Dampfmaschine sehr nahe kommt (Abb. 5).

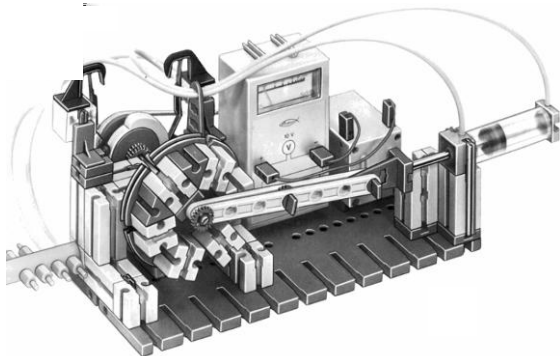


Abb. 5: Druckluftmotor aus der fischertechnik-Pneumatik-Anleitung (1981), S. 63

Ganz so einfach wollen wir es uns aber nicht machen – zumal einige der in diesem Modell verwendeten Teile in den aktuellen Pneumatik-Kästen nicht mehr enthalten sind. Denn die Wattsche Dampfmaschine enthält zahlreiche interessante und lehr-

reiche technische Details, die wir uns genauer ansehen wollen.

Versetzen wir uns dazu einmal 240 Jahre zurück in das Labor von *James Watt*. Was benötigen wir genau für eine Dampfmaschine? Im Wesentlichen sind es vier Komponenten:

- eine Energiequelle (in unserem Modell ist das ein Druckluftgenerator),
- ein Getriebe, das die Auf- und Abbewegung des Kolbens (eine „oszillierende“ Bewegung) in eine Drehbewegung (eine „Rotation“) umwandelt,
- ein Schwungrad (wozu das erforderlich ist, werden wir noch sehen), und
- einen Steuermechanismus, der – abhängig von der Position des Kolbens im Zylinder – den „Dampf“ (im Modell die Druckluft) wechselseitig im richtigen Moment auf die richtige Seite des Kolbens lenkt.

Die Konstruktion des Druckluftgenerators sparen wir uns hier – dazu findet ihr in euren Pneumatik-Kästen gute Anleitungen, und die Kästen [Robo TX ElectroPneumatik](#) und [Pneumatic 3](#) enthalten einen sehr leistungsfähigen fertigen ft-Kompressor. Auch die von Pollin Elektronik angebotene [Luftpumpe](#), die nach Entfernung des Halteblechs exakt in eine ft-Rohrhülse (36702) hineinpasst, eignet sich als fertiger Druckluftgenerator. Spannend ist hingegen die Konstruktion der drei anderen Elemente.

Das Schubkurbelgetriebe

Zur Umwandlung der oszillierenden (schwingenden) Schubbewegung des Kolbens im Zylinder in eine Drehbewegung verwendete *James Watt* ein **Schubkurbelgetriebe** (Abb. 6). Dabei wird die Kolbenstange *P* (auch „**Pleuelstange**“ genannt) mit einer auf der Antriebsachse („**Welle**“ genannt) montierten Kurbel *K* verbunden.

Dieses Prinzip war zu Lebzeiten Watts bereits seit Jahrhunderten bekannt – das

älteste bekannte Schubkurbelgetriebe fand man in einer Relief-Zeichnung einer römischen Sägemühle auf einem Sarkophag eines Müllers aus Hierapolis. Es datiert aus dem 3. Jahrhundert nach Christus – vor über 1.700 Jahren.

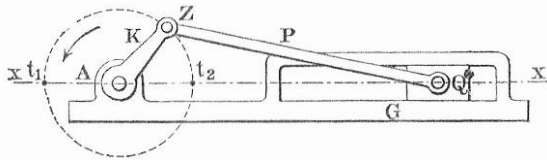


Abb. 6: Schubkurbelgetriebe
(Quelle: Meyers Großes Konversations-Lexikon, 6. Auflage, 1905-1909)

Mit fischertechnik gelingt eine ziemlich „naturgetreue“ Konstruktion eines solchen Schubkurbelgetriebes mit der Segmentscheibe (grau: 31037, schwarz: 110349): Die Pleuellstange wird dazu über einen B15 und ein Rollenlager (37636) mit einer Rastachse 20 (31690) an unserer Pleuellstange (aus zwei Statik-Streben X-127,2) befestigt. Damit die Pleuellstange sich nicht unter Last biegt und verkantet, wird der B15 über zwei Gelenkwürfel-Klauen (31436) mit Lagerhülse (36819) auf zwei parallelen Metallachsen 110 (31031) gerade geführt (Abb. 7). Diese müssen exakt ausgerichtet sein, damit die Lagerhülsen darauf reibungsfrei gleiten.

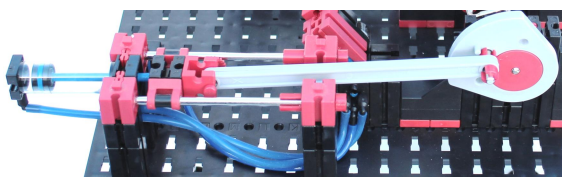


Abb. 7: fischertechnik-Schubkurbelgetriebe

Die „Pleuellstange“ wird auf der anderen Seite auf eine Metallachse 30 (31034) geschoben, die in das mittlere Loch der Segmentscheibe gesteckt wird. Mit drei Klemmbuchsen 5 (37679) werden Pleuell und Achse fixiert. Statt der Segmentscheibe kann man auch eine Drehscheibe 60 (31039), ein Z30 (36264) oder ein Z40 (31022) verwenden, denn die Löcher für die Befestigung der Metallachse 30 haben

alle dieselbe Position (1,5 cm vom Achsmittelpunkt).

Ein Schubkurbelgetriebe leidet jedoch unter einem prinzipiellen Problem: Die Umwandlung der Schub- in eine Rotationsbewegung erfolgt nicht gleichmäßig. Das zeigt uns eine kleine Schema-Zeichnung (Abb. 8): Sei x die Pleuellstange (rot), y der Radius der Segmentscheibe (blau), m der Mittelpunkt der Welle (Abb. 8a), A und B die äußersten Positionen des Endes S der Pleuellstange, und O deren Mittelpunkt.

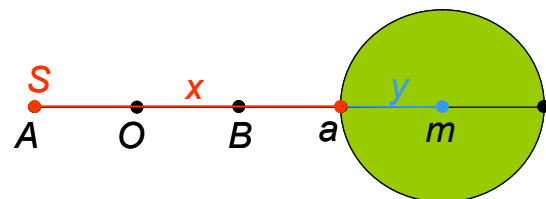


Abb. 8a: Schubkurbelgetriebe ...

Abb. 8b zeigt den Zustand, nachdem sich die Segmentscheibe um 45° (also $1/8$ des Kreises) gedreht hat. Tatsächlich hat sich der Punkt S dabei nicht um $y/2$, sondern lediglich um ca. $0,29 \cdot y$ bewegt² – und wird sich bei der nächsten Achteldrehung der Segmentscheibe um $0,71 \cdot y$ weiterbewegen.

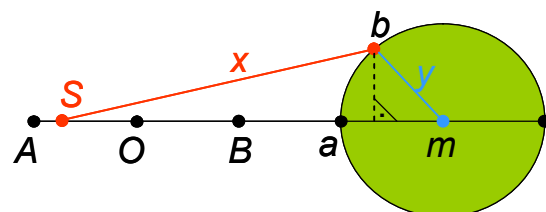


Abb. 8b: ... nach 45° ...

Was bedeutet das für unsere Dampfmaschine? Klar: Das Drehmoment eines Pleuellhubs ist bei der ersten und vierten Achteldrehung größer als bei der zweiten und dritten, Segmentscheibe und Welle werden daher mal stärker, mal schwächer angetrieben.

Und noch eine Ungleichmäßigkeit besitzt das Schubkurbelgetriebe: Nach einer $1/4$ -

² Genau sind es $(1 - 1/\sqrt{2}) \cdot y$.

Umdrehung steht die Segmentscheibe senkrecht im Punkt c (Abb. 8c). Nun sollte das linke Ende S der Pleuelstange genau auf O stehen – tatsächlich aber liegt S deutlich rechts davon. Wie kann das sein?

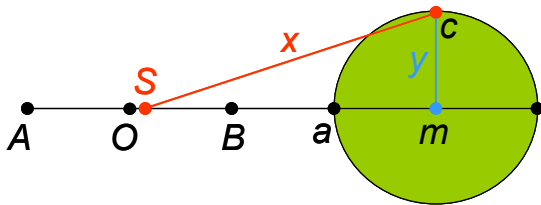


Abb. 8c: ... und nach 90°

Die Erklärung ist einfach: In dieser Position ist der Abstand zwischen dem einen Ende der Pleuelstange S und dem Achsenmittelpunkt m nicht gleich x – sondern (nach dem [Satz des Pythagoras](#)) gleich der Wurzel $\sqrt{x^2 - y^2}$. Ist also y größer als 0, dann ist der Abstand zwischen S und m kleiner als x . Das Drehmoment eines Kolbenhubes ist daher in der ersten Hälfte kleiner als in der zweiten.

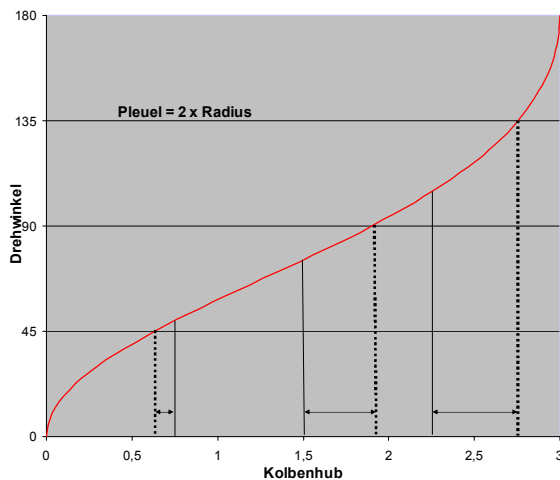


Abb. 9: Drehwinkel in Abhängigkeit vom Kolbenhub

Gut zu erkennen ist das, wenn man den Drehwinkel α , um den sich die Segmentscheibe bereits gedreht hat, in Abhängigkeit vom Kolbenhub $h = \overline{AS}$ ausdrückt. Das ist gar nicht so schwierig und erfordert nur ein wenig Trigonometrie [8]:

$$h = x + y - y \cdot \cos \alpha - \sqrt{x^2 - y^2} \cdot \sin^2 \alpha$$

Die Herleitung der Formel sei euch als kleine Fingerübung überlassen.

Das Ergebnis zeigt die Kurve in Abb. 9 für den Segmentscheibenradius von $y = 1,5$ cm und eine doppelt so lange Pleuelstange: Erst nach 63 % des Kolbenhubes hat sich die Segmentscheibe um 90° gedreht.

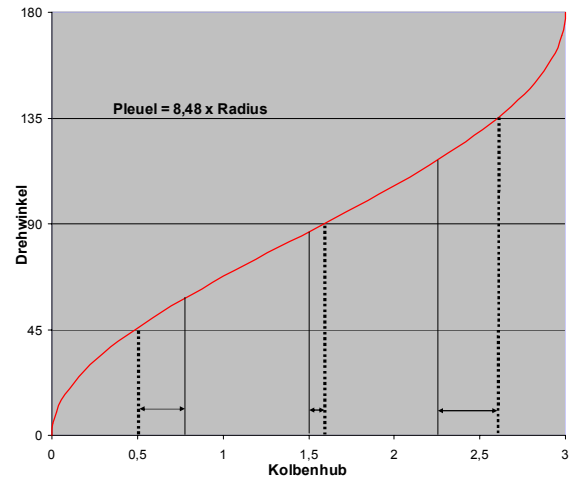


Abb. 10: Kolbenhub und Drehwinkel in unserem Modell

Da sich diese Ungleichmäßigkeit nicht vermeiden lässt, bleibt nur, sie durch eine geeignete Konstruktion zu verringern. Das gelingt, indem x (die Pleuelstange) deutlich größer gewählt wird als der Radius der Segmentscheibe. In unserem Modell hat die Pleuelstange (Statik-Strebe X-127,2) eine Länge von $x = 12,72$ cm, ist also 8,48 mal so lang wie der Radius der Segmentscheibe. Das Resultat sieht man in Abb. 10: Die Verlaufskurve ist deutlich „geglättet“, und die 90° werden bereits nach 53 % des Kolbenhubes erreicht.

Das Schwungrad

Ein Schubkurbelgetriebe hat jedoch noch einen Schwachpunkt: Beim Wechsel der Schubrichtung des Kolbens erreicht dieser einen „toten Punkt“, nämlich genau dann, wenn die Pleuelstange auf die Achse der Kurbelwelle „zeigt“. Bleibt der Kolben in exakt dieser Position stehen, kann ihn keine noch so starke Schubkraft mehr bewegen, da der Schub hier nicht in eine

Drehbewegung umgewandelt werden kann. Daher benötigt man einen Mechanismus, mit dem man erreicht, dass sich die Antriebswelle bei jeder Umdrehung über die beiden toten Punkte hinaus dreht.

Dazu machte *James Watt* sich ein Phänomen zu Nutze, das der Mathematiker [Leonhard Euler](#) (1707-1783) kurz zuvor, im Jahr 1740, erstmals systematisch untersucht hatte: die **Massenträgheit**. Darunter versteht man den Widerstand jedes Gegenstands gegen jede Änderung seines Bewegungszustands.

Das Prinzip kennt ihr alle vom Radfahren: Beim Anfahren braucht ihr viel Kraft, bis das Fahrrad die gewünschte Geschwindigkeit erreicht hat. Umgekehrt bleibt es aber keineswegs stehen, wenn ihr aufhört zu treten, sondern rollt weiter – zum Anhalten müsst ihr bremsen.

Watt nutzte die Massenträgheit bei seiner Dampfmaschine, indem er auf die Antriebswelle ein **Schwungrad** montierte: Erreicht der Kolben einen Endpunkt, sorgt die Trägheit des Schwungrads dafür, dass die Welle über den toten Punkt hinwegdreht. Ein Schwungrad ist also gewissermaßen ein kleiner Energiespeicher: Es „schluckt“ beim Starten Energie (und verlangsamt damit zunächst die Drehbewegung), und gibt diese Energie wieder ab, wenn die Kraftzufuhr abnimmt oder ausbleibt.

Das Schwungrad hat noch einen zweiten positiven Effekt: es gleicht die Ungleichmäßigkeiten des Schubkurbelgetriebes (s. o.) aus. Das ist besonders bei einer niedrigen Frequenz des Kolbens wie bei unserem Dampfmaschinen-Modell spürbar.

Das Schwungrad muss so konstruiert sein, dass die Kraft eines einzigen Kolbenhubs es so stark beschleunigt, dass es den „toten Punkt“ des Schubkurbelgetriebes überwindet. Dazu darf es einerseits nicht zu schwer sein, muss aber andererseits schwer genug sein, damit seine Massenträgheit ausreicht.

Auch ohne zu wissen, wie man die Trägheit berechnet, leuchtet ein, dass es bei der Konstruktion vor allem auf die Masse am äußeren Rand des Schwungrads ankommt – denn sie legt bei einer Umdrehung den längsten Weg zurück und bewegt sich damit am schnellsten. Der Rand des Schwungrads hat daher den größten Anteil an der Massenträgheit; sie wird im Wesentlichen vom Radius und dem Gewicht der Bausteine bestimmt.

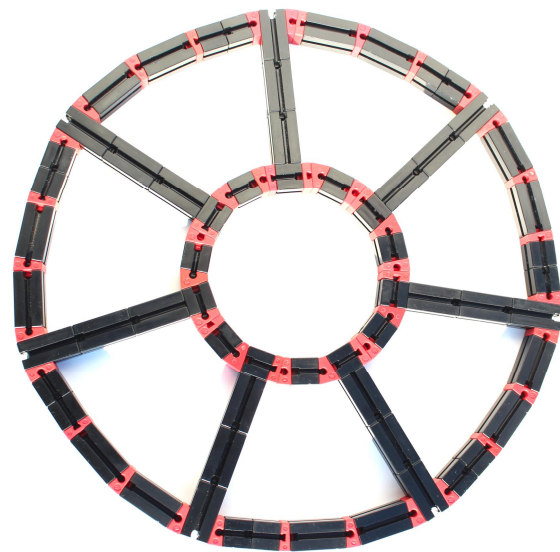


Abb. 11: Schwungrad aus Bausteinen 30

Die nahe liegende Lösung, das Schwungrad aus sechs Flachträgern 120 mit Bogenstücken 60° zu konstruieren, ist daher ungeeignet – das Schwungrad wäre viel zu leicht. Der äußere Ring sollte aus Bausteinen 30 und Winkelsteinen bestehen und einen ausreichenden Radius aufweisen, wie z. B. das aus dem [Modell von thomas004](#).

Die Druckluftzuführung

Bei den Druckluftmotoren aus der Pneumatik-Anleitung von 1981 (Abb. 5) und dem schönen [Dampfmaschinenmodell von Fredy](#) wird die Druckluftzuführung mit den Festo-Ventilen (36082) und einer auf der Welle montierten Schaltscheibe (37727) geregelt. Das ist eine elegante Lösung, da die Umschaltung der Druckzufuhr verzögerungsfrei erfolgt und das Modell daher mit einem deutlich kleineren Schwungrad

auskommt – aber die Ventile gibt es nicht mehr, und für unser „historisches“ Dampfmaschinenmodell wären sie zudem nicht zeitgemäß.

Watt verwendete einen Schieber, der, gesteuert von einem mit der Welle (W) verbundenen Gestänge und einer Exzenter-Scheibe (E) den Dampf abwechselnd in die eine oder die andere Kolbenseite einließ.

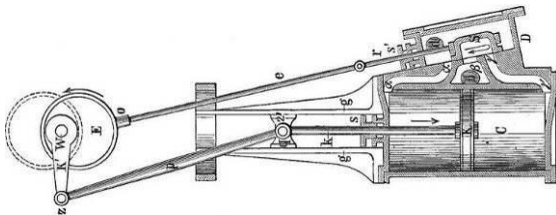


Abb. 12: Regelung der Dampfzufuhr
(Quelle: Meyers Großes Konversations-Lexikon, Band 4, Leipzig 1906)

Diesem Schieber kommt das Pneumatik-Handventil recht nahe, das in unserem Modell von einem Hebel bewegt wird (Abb. 13) – eine Variante der Idee aus dem [Dampfmaschinen-Modell von thomas004](#).

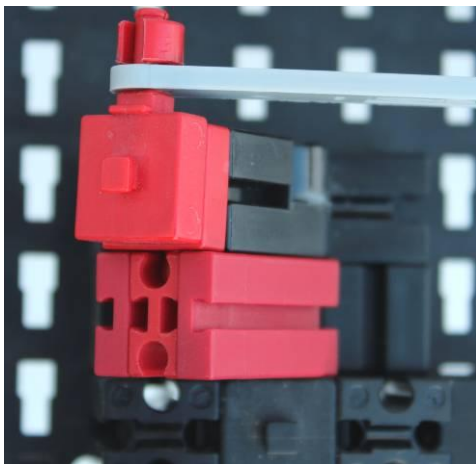


Abb. 13: Pneumatik-Handventil (unten, Mitte) mit Hebel (Gelenkbaustein 45)

Knifflig ist aber die Steuerung dieses Hebels. Er hat einen großen „Schaltweg“ – der 1,5-cm-Radius einer Segment- oder einer Nockenscheibe (31038) reicht dafür nicht. Also benötigen wir eine Kurbelwelle, die geeignet (z. B. mit einer Statik-Strebe) mit dem Hebel gekoppelt wird – ein zweites Schubkurbelgetriebe (Abb. 14).

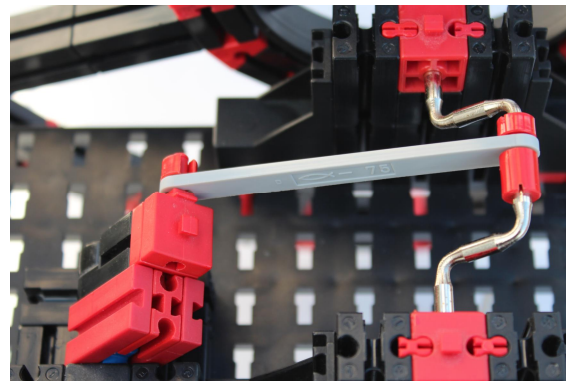


Abb. 14: Steuerung der Druckluftzufuhr

Sind Kurbelwelle und Segmentscheibe sauber justiert, dann legt unsere Dampfmaschine mit ca. 60 U/min los. Der relativ lange „Leerlauf“, den das Handventil verursacht und der durch das Schwungrad ausgeglichen werden muss, lässt sie ein wenig „stoßweise“ arbeiten. Das ließe sich durch ein größeres Schwungrad oder einen zweiten Kolben ausgleichen – oder doch durch die Verwendung verzögerungsfreier Umschalter wie den elektromagnetischen Ventilen (siehe bspw. das [Druckluftmotor-Modell von ixer](#)).

Solche Varianten seien eurem Einfallsreichtum überlassen ...

Quellen

- [1] Wikipedia: [Dampfmaschine](#).
- [2] Wikipedia: [Thomas Newcomen](#).
- [3] Wikipedia: [James Watt](#).
- [4] Steam Engines. [Watt's Specification](#). Patent No. 913, 1769
- [5] Wikipedia: [Richard Trevithick](#).
- [6] Achim Scheunert: [Sieg der Feuermaschine](#). Doku-Drama, 2008.
- [7] Werner Kiefer: [James Watt und die Dampfmaschine](#). Film (15 min.) aus der Sendereihe ‚Meilensteine der Naturwissenschaft und Technik‘ des Schulfernsehens der ARD, 2005.
- [8] Wikipedia: [Schubkurbel](#).

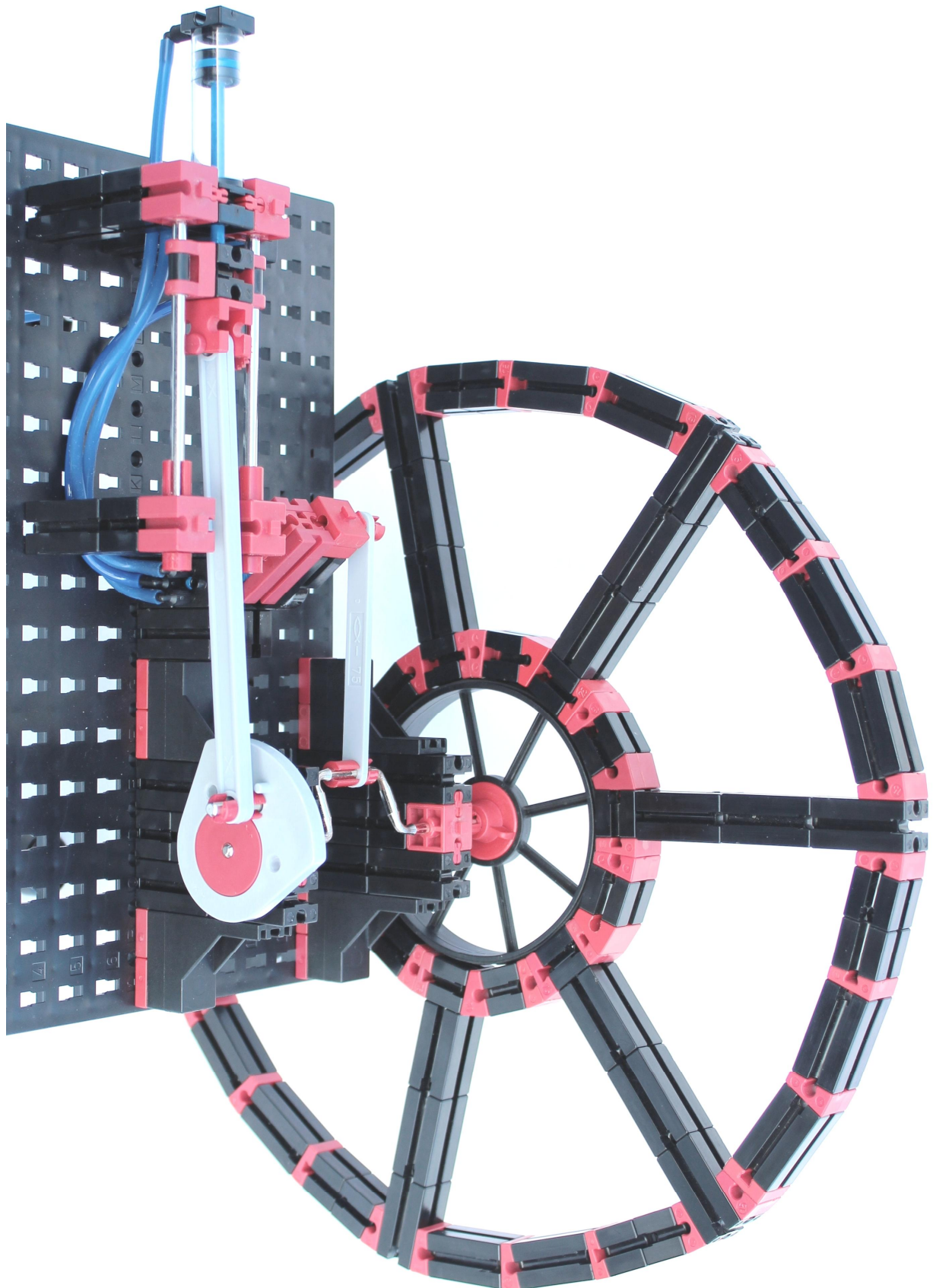


Abb. 15: Modell der Wattschen Dampfmaschine