

Editorial

Artur Fischer

Am 27. Januar 2016 ist Artur Fischer im Alter von 96 Jahren gestorben. Für seine Erfindungen wurden ihm etwa 6.000 eigene Patente und Gebrauchsmuster erteilt – nur ein kleiner Teil davon für fischertechnik und die Befestigungssysteme. Artur Fischer hatte das Erfinden zum Beruf gemacht wie vor ihm der Amerikaner [Thomas Alva Edison](#) (1847-1931). Ein Vergleich, den er im Übrigen gar nicht mochte, denn die wichtigste Edison zugeschriebene Erfindung, die Glühlampe, stammt gar nicht von ihm – er verbesserte sie allerdings in vielerlei Hinsicht.

Artur Fischer war ein Erfinder, kein Tüftler. Auf diesen feinen, aber wichtigen Unterschied legte er großen Wert. Anders als der Tüftler beseitigt ein Erfinder einen konkreten, praktischen Mangel, indem er in einem schöpferischen Prozess dafür eine zuvor nicht bekannte technische Lösung findet. Die Suche nach dieser Lösung folgt dabei einem systematischen, ingenieurmäßigen Vorgehen – und nicht blindem Versuch und Irrtum. Zudem hat der Erfinder von Anfang an auch die Vermarktung der gefundenen Lösung im Auge: „Die beste Lösungsidee ist letzten Endes bedeutungslos, wenn kein Markt dafür erschließbar oder ein Bedürfnis dafür nicht vorhanden ist.“

Aber Artur Fischer war ein besonderer Erfinder. Denn seine Erfindungen sollten auch nützlich sein. „Nur dann ist eine Erfindung eine nützliche Erfindung, wenn sie dem Menschen dient“, war einer seiner Leitsätze. Er war ein Philanthrop – und das äußerte sich nicht nur in seinen Erfindungen, sondern auch in seiner großzügigen

Dirk Fox, Stefan Falk

finanziellen Unterstützung zahlreicher sozialer Projekte.

Besonders am Herzen lagen ihm aber immer die Kinder. So, wie er selbst in seiner frühen Jugend vor allem von seiner Mutter in seinen Experimenten ermutigt worden war, wollte er Kinder zur Entfaltung ihrer Phantasie und Begabung ermutigen. Eine Geschichte, die er gerne erzählte, handelte von einem großen hölzernen Propeller, den er von einem befreundeten Schreiner geschenkt bekommen hatte. Als seine Flugzeugkonstruktion dann trotz Schweiß treibenden Kurbelns nicht abhob, tröstete ihn seine Mutter mit den Worten: „Jetzt hast Du etwas Wichtiges gelernt: So funktioniert es also nicht.“ Wie könnte man Frustrationstoleranz schöner vermitteln?

Bei allem Erfolg und den zahlreichen Ehrungen (darunter das Große Verdienstkreuz mit Stern und Schulterband und der Europäische Erfinderpriis für sein Lebenswerk) ist er immer ein bodenständiger und bescheidener Mensch geblieben, der am glücklichsten war, wenn er mit Feile und Schraubstock einen neuen Prototyp bearbeiten konnte.

Als fischertechniker können wir uns nun als glückliche Erben fühlen. Denn ein Erfinder hinterlässt der Menschheit seine Erfindungen. Also ran an die Kästen – das hätte Artur Fischer sich von uns gewünscht.

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

Artur Fischer	2
Mini-Modelle (Teil 9): Motorrad	5
Mini-Modelle (Teil 10): Jojo	6
fischertechnik-Nutprofile selbst herstellen	8
Kleine Statik-Hilfe	10
Magnetkupplung für die Stromversorgung	12
Mini-Modelle (Teil 11): Flugzeug	14
Neue ft-Teile selbst gemacht – 3D-Druck (4): Schleifringe (die Zweite), und überhaupt	15
Modellfotografie	21
Geradfürungen	24
Der Wohnzimmer-Dienstreisen-Urlaubs-Notfallkasten ..	31
Scherenhub	37
Schlauchquetschpumpe	40
Beschleunigung auf der schiefen Ebene	42
Radar mit Objektfokussierung	49
Laser-Anwendungen (3): Analoges CD-Spieler	58
TXT Controller – Tipps und Tricks (1): Das Root- Passwort	65
Digitalkamera mit Autofokus und Live-Video-Vorschau	69
I ² C mit dem TX(T) – Teil 13: Farbsensor	79

Termine

Was?	Wann?	Wo?
Clubtag	16.04.2016	Appeldoorn
Maker Faire	27.- 29.05.2016	Hannover
FanClubTag	24.07.2016	Waldachtal
fischertechnik Convention 2016	24.09.2016	Dreieich (bei Frankfurt)

Hinweise

Korrektur. Im Beitrag „Parallel-Interface durch Arduino gesteuert (1)“ von Jens Lemkamp ([ft:pedia 1/2014](#), S. 26) muss es heißen: „Man beginnt ganz rechts mit Motor 4 (M4), dann kommen M3, M2 und zum Schluss M1“. Danke an Holger Howey für den Hinweis.

Impressum

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

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

Autoren: Norbert Doetsch, Stefan Falk, Dirk Fox,
Andreas Gail, Werner Hasselberg, David Holtz, Raphael
Jacob, Thomas Püttmann, Harald Steinhaus, Torsten
Stuehn, Andreas Tacke, René Trapp, Dirk Wölfel.

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

Modell

Mini-Modelle (Teil 9): Motorrad

Norbert Doetsch

In dieser Folge gibt es ein winziges Zweirad mit einem kleinen Gummi statt Kettentrieb.

Unser Road-Runner ist üppig ausgestattet mit Lenkergriffen und sogar einem in Höhe und Neigung verstellbaren Sitz. Durch die Doppel-Hinterreifen geht er in der Fantasie natürlich ab wie ein Großer!



Abb. 1: Road-Runner von links vorne



Abb. 2: Road-Runner von links hinten



Abb. 3: Road-Runner von der Seite

Hier folgt die Stückliste.

St.	ft-Nr.	Bezeichnung
2	37468	Baustein 7,5
3	35797	Seilrolle
1	31690	V-Achse 20
1	32316	Verbindungsstopfen
1	36334	S-Riegelscheibe
1	37232	S-Verschlussriegel 4
1	38253	S-Kupplung 15
1	38260	S-Kupplung 22,5
2	31848	Strebenadapter
2	37679	Klemmbuchse 5
1	36914	S-Strebe 15
1	38413	Kunststoffachse 30
1	31597	Abstandsring 4 mm
1	31982	Federnocken
1		Loom-Gummiband

Tab. 1: Einzelteilliste

Modell

Mini-Modelle (Teil 10): Jojo

René Trapp

Auf und nieder. Immer wieder.

Es war einmal eine liebreizende Prinzessin, die kam zu ihrem altherwürdigen Hofmechanikus und verlangte: „Baue er mir ein Jojo. Eines, wie es sonst keine hat. Eines, um das mich alle anderen Prinzessinnen beneiden. Wenn er mir das fertige Jojo in einer Stunde bringt, überhäufe ich ihn mit Gold. Wenn nicht, lasse ich ihn in den Kerker werfen.“

Derart angespornt erdachte der Hofmechanikus ein Jojo. Eines, wie es sonst keine hatte. Eines, um das sie alle anderen Prinzessinnen beneiden würden. Allerdings hatte der Hofmechanikus kein fischertechnik. So musste er das Jojo mühsam von Hand anfertigen. Dafür reichte die vorgegebene Stunde natürlich nicht. Die Prinzessin schmolte und ließ den Hofmechanikus nach Ablauf einer Stunde unbarmherzig in den Kerker werfen. Und wenn er nicht gestorben ist, so sitzt der Hofmechanikus noch immer in seinem Verlies und feilt am Jojo – und die Prinzessin schmolzt.

Mittlerweile ist fischertechnik erfunden, und daher bereitet es uns keine größeren Schwierigkeiten auf die Schnelle ein Jojo zu bauen. Oder etwa doch?

An sich ist die Konstruktion klar. Man benötigt einen Faden, eine dünne Achse und zwei Schwungmassen (Abb. 1).

Als Schwungmassen eignen sich vorzüglich die Schwungscheiben (31718). Zwar sind diese leider rar und teuer. Drehscheiben (31019) sind jedoch wegen der Nuten auf dem Umfang unbrauchbar, und Speichenräder (19317 oder 36916) sehen irgendwie

seltsam aus und geben auch nicht die richtige Seilführung im Inneren.



Abb. 1: Das Jojo

Weil sich bei den gekauften Jojos immer die Schnur verheddert, wenn eine Weile nicht mit ihnen gespielt wurde, gibt es hier obendrein noch einen schicken magnetischen Handgriff.

Der Zusammenbau ist an sich nicht schwierig, dennoch scheinen ein paar Worte angebracht. Das Seil (35039) wird an einem Ende mit einer Klemmbuchse (37679) in der Mitte der Metallachse (31033) festgeklemmt. Eine Schlinge ist nicht zwingend erforderlich. Die mittige Position der Klemmbuchse ist wichtig für das Gleichgewicht. Die Schwungscheiben (31718) werden auf je eine Flachnabe (31015) gesetzt und mit der zugehörigen Nabenmutter auf die Achse geschoben – ganz dicht an die Klemmbuchse heran. Die Metallachse sollte an beiden Seiten gleich lang überstehen. Stimmt die Ausrichtung, werden die Nabenmutter festgezogen. Das andere

Ende des Seiles wird einmal um den Magnetbaustein (1081278) herumgeschlungen und mit dem Baustein 5 (37237) festgeklemmt – fertig ist das Jojo.

Zur Aufbewahrung wird das Seil aufgewickelt und der Magnetbaustein auf eine Seite der überstehenden Achse gesteckt (Abb. 2).



Abb. 2: Das Jojo in Ruhestellung

Diese Teile werden für das Jojo benötigt:

Stück	ft-Nr.	Bezeichnung
1	31033	Metallachse 50
1	37679	Klemmbuchse 5
2	31015	Flachnabe 25 (kompl.)
2	31718	Schwungscheibe 60x15
1	37237	Baustein 5
1	108278	Baustein 15 Magnet
1	35039	Seil 600

Tab. 1: Einzelteilliste

Anstelle des vorgeschlagenen Seiles mit 60 cm geht es auch länger. Die Körpergröße der Prinzessin sollte allerdings berücksichtigt werden, sonst schlägt das Jojo womöglich unsanft auf dem Boden auf.

Tipps & Tricks

fischertechnik-Nutprofile selbst herstellen

Andreas Tacke

Ohne Zweifel: „To mod or not to mod“ ist eine Grundsatzfrage. Auch wenn sich dem einen oder anderen die Nackenhaare bereits aufstellen, sobald in Hörweite des fischertechnik-Hobbyraums Werkzeuggeräusche zu vernehmen sind: Wenn man schon zum Bohrer greifen muss, dann soll das Ergebnis wenigstens manierlich aussehen. Wird eine funktionierende Nut gebraucht, ist guter Rat jedoch teuer. Wer kann da wohl besser helfen als TST?

In den letzten Jahren habe ich für das ein oder andere Modell einige spezielle Teile angefertigt. Eine Herausforderung war es dabei, eine fischertechnik-Nut zu fräsen.

Hier zunächst ein paar Bilder von einigen dieser speziellen Bauteile.



Abb. 1: Spezialhalter für den Powermotor

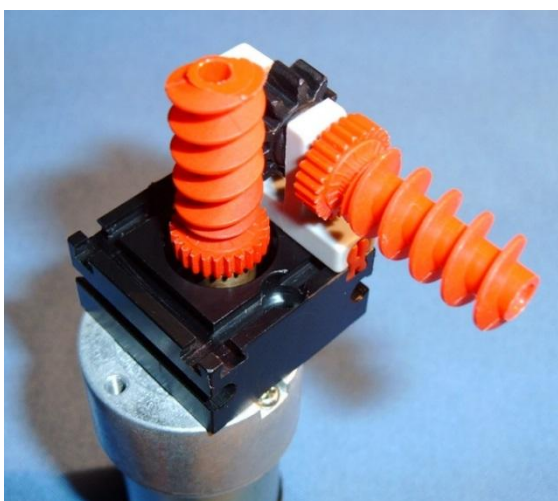


Abb. 2: Spezialhalter für den Schneckenantrieb

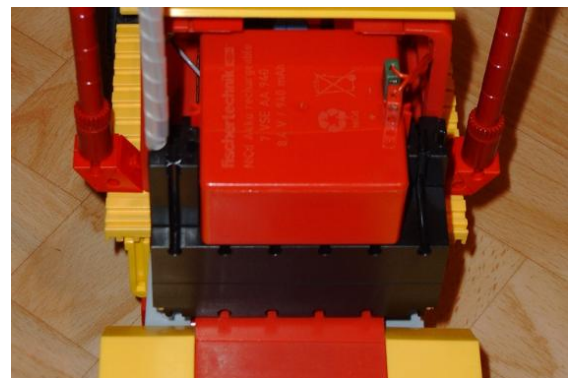


Abb. 3: Gehäuse für den Funkempfänger an meinem Truck

Zum Herstellen der Nuten werden folgende Teile benötigt:

- eine Fräsmaschine,
- ein 3 mm-Schaftfräser mit R1,5 und
- ein 4 mm-Kugelfräser.



Abb. 4: Fräswerkzeuge

Meine Bauteile habe ich aus POM (Polyoxymethylen) hergestellt, einem Material das sich sehr gut fräsen lässt. Dann ging es daran, die Maße der Nuten zu ermitteln.

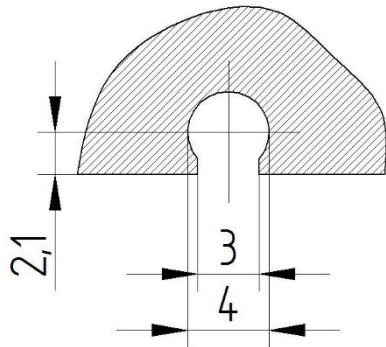


Abb. 5: Zeichnung der Nutgeometrie

Danach wird im ersten Arbeitsschritt eine 3 mm breite Nut mit einer Tiefe von 4,1 mm gefräst (Abb. 6).

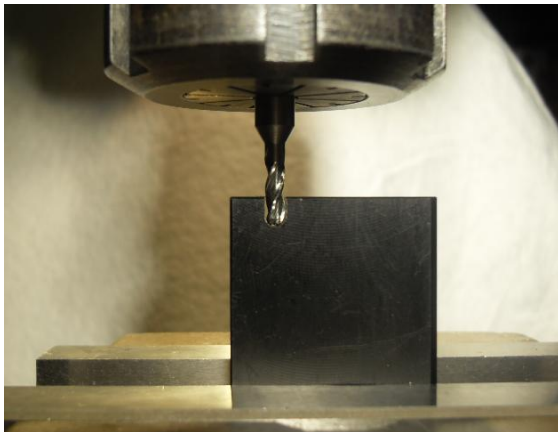


Abb. 6: Erster Schritt, Nut 3 mm, 4,1 mm tief

Im zweiten Schritt wird die 4 mm-Kugelform mit einem 4 mm-Kugelfräser gefräst. Die Tiefe beträgt hierbei ebenfalls 4,1 mm (Abb. 7).

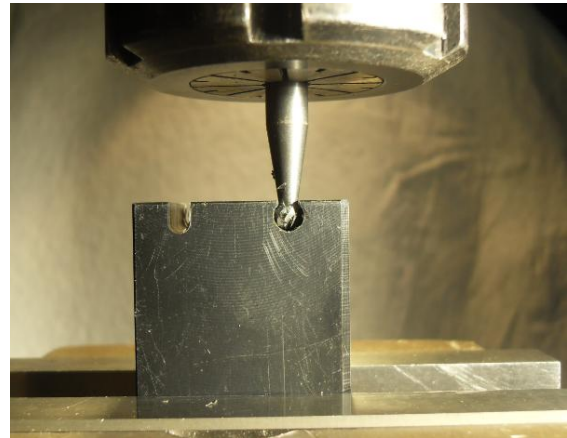


Abb. 7: Zweiter Schritt:
Kugelform 4 mm, 4,1 mm tief

So entsteht ein fischertechnik-Nutprofil, das sich mit allen fischertechnik-Teilen kombinieren lässt (Abb. 8).

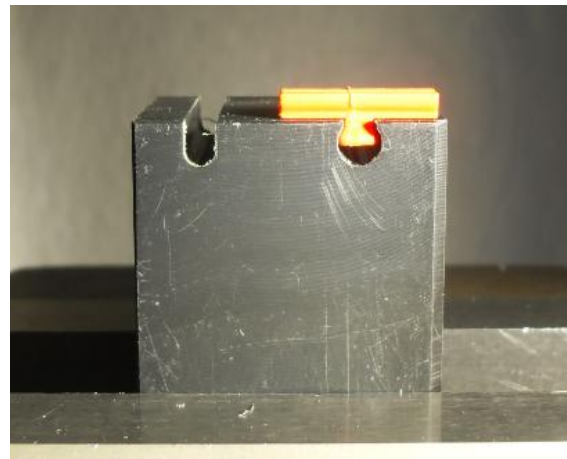


Abb. 8: Fertige Nut mit Federnocken

Auf diese Weise ist es möglich, sich das eine oder andere fischertechnik-kompatible Bauteil für seine Modelle selber herzustellen.

Schließlich gibt es noch die abgeflachte Nut. Wie ich die herstelle, werde ich zu einem späteren Zeitpunkt erläutern.

Wer über die oben angeführten technischen Hilfsmittel verfügt, kann sich nun seine ersten eigenen Bauteile herstellen...

Tipps & Tricks

Kleine Statik-Hilfe

René Trapp

Die Welt besteht nicht nur aus rechten Winkeln und 45°-Diagonalen. fischertechnik-Modelle auch nicht.

Das Erscheinen der fischertechnik-Statik-kästen 1970 erweiterte die Baumöglichkeiten erheblich. Wer sich nicht mit den gewohnten Konstruktionen mit Streben unter 90° und 45° zufrieden geben mag, kommt um eine rationelle Hilfe zur Strebenauswahl nicht herum – wenn er schiefe oder verzogene Modellkonstruktionen vermeiden möchte. Natürlich kann man auch immer wieder zum Taschenrechner oder Tabellenwerk [1] greifen.

Übersichtlicher geht es allerdings so wie es bereits im Handbuch zum 1971 erschienenen Kasten „Hobby S“ dargestellt ist [2]. Das Handbuch gibt es als Download im ft-Museum [3]. Darin gibt es auf Seite 20 ein Diagramm. Die Erklärung dazu ist denkbar kurz: *„Die mit Streben herstellbaren Verbindungen zeigt Diagramm (70).“* Sehr hilfreich – aber leider sind in diesem Diagramm ein paar Punkte eingezeichnet, die eben nicht so schön auf dem Raster liegen wie es uns das Diagramm glauben machen möchte. In Rastereinheiten mag auch nicht jeder denken, in der Einheit ‚mm‘ skalierte Achsen sind universeller in der Handhabung. Und so wird es Zeit, eine neue Version anzubieten (Abb. 1).

Ausgehend vom Koordinatenursprung ist auf der waagerechten Achse eine Skalierung im Raster 15 mm angebracht. Die Längen der I-Streben, die immer exakt im Raster liegen, sind als blaue Kreisbögen dargestellt, die Längen der X-Streben, die recht genaue Diagonalenmaße besitzen

(etwa $\sqrt{2} \cdot n \cdot 15$ mm), haben grüne Kreisbögen bekommen. Auf der senkrechten Achse ist ebenfalls die Skalierung zu 15 mm gewählt. Dazu gibt es noch ein zweites Gitternetz mit der Skalierung 21,2 mm ($\approx \sqrt{2} \cdot 15$ mm). Immer dort wo ein Kreisbogen eine Kreuzung aus waagerechter und senkrechter Gitterlinie schneidet, ergibt sich die passende Länge für eine Strebe.

Alle passenden Kreuzungspunkte sind mit einem roten Ring markiert, einige sind auch noch mit dem Ursprung verbunden, um die Richtung der Verbindung anzudeuten. Bei den Streben unter 45° Winkel wurde zu Gunsten der Übersicht darauf verzichtet.

Beispiel 1

Waagerechter Abstand 45 mm, senkrechter Abstand 60 mm ergibt einen Schnittpunkt mit einem blauen Kreisbogen. Dieser blaue Kreisbogen gehört zur Strebenlänge 75 mm (man findet das, indem man dem Kreisbogen bis an ein Ende folgt und die Länge an der Achsskalierung abliest).

Beispiel 2

Waagerechter Abstand 15 mm, senkrechter Abstand 42,4 mm ergibt auch einen Schnittpunkt mit einem blauen Kreisbogen, hier 45 mm.

Quellen

- [1] Thomas Habig: [Statikhilfe](#).
- [2] Fischer-Werke: [Hobby S \(39550\)](#).
- [3] [fischertechnik Museum.ch](#)

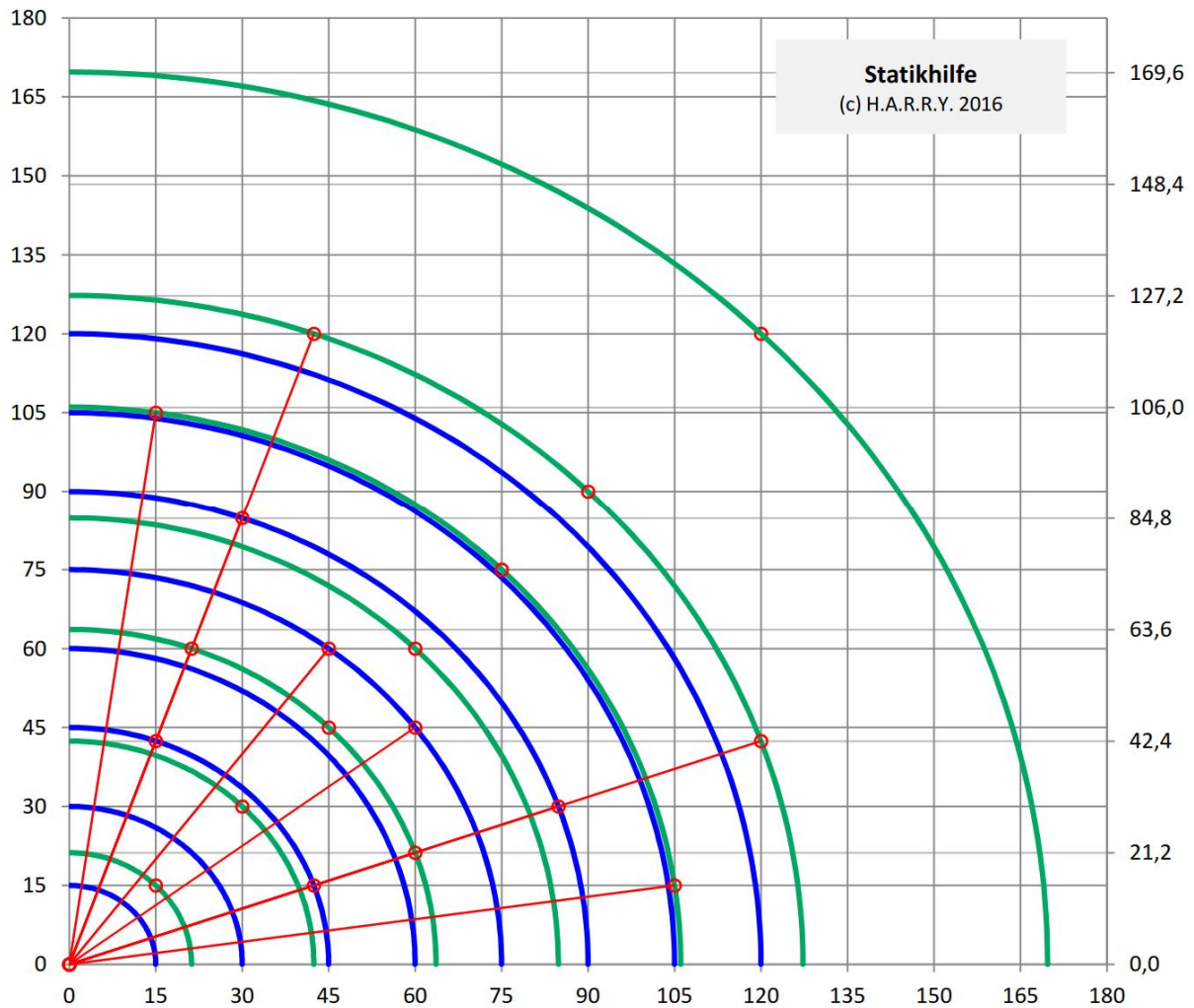


Abb. 1: Kleine Statik-Hilfe, alle Angaben in mm (nicht maßstäblich)

Legende

- I-Streben sind durch **blaue**, X-Streben durch **grüne** Kreisbögen dargestellt.
- Immer dort, wo ein Kreisbogen eine Kreuzung aus waagerechter und senkrechter Gitterlinie schneidet, ergibt sich die passende Länge für eine Strebe.
- Ausgehend vom Koordinatenursprung sind alle passenden Kreuzungspunkte mit einem **roten Ring** markiert.

Tipps & Tricks

Magnetkupplung für die Stromversorgung

Dirk Wölfel

Eine externe Stromversorgung der fischertechnik-Modelle mit einem zweiadrigen Kabel kann manchmal zu Problemen führen: Will man zum Beispiel die Stromquelle zwischen mehreren fertigen Modellen wechseln, muss man dabei auch immer auf die richtige Polung der Stecker achten. Bleibt man am Kabel hängen, können die Stecker oder das Kabel herausgerissen werden – im schlimmsten Fall wird sogar das Modell umgerissen. Dabei gibt es den Komfort, den Besitzer von Apple-Computern genießen, auch für fischertechnik-Modelle...

Die Magnetkupplung

Abhilfe schafft eine *Magnetkupplung* für die Stromversorgung oder zum Laden eines Akkus. Diese lässt sich einfach herstellen: Ihr braucht dazu nur wenige (zugegeben: teilweise „historische“) fischertechnik-Bauteile im Wert von ca. 5 € Fehlende Bauteile könnt ihr zum Beispiel bei [fischerfriendsman](#) bestellen.

Hier die Einzelteilliste:

Stk.	ft-Nr.	Bezeichnung
2	7051	Magnetbaustein + Magnet
2	31306	Metall Federkontakt
2	35329	Federkontaktgegenstück
2	32881	Baustein 15 mit 1 Zapfen

Tab. 1: Erforderliche Bauteile



Abb. 1: Bauteile für Magnetkupplung

Habt ihr alle Bauteile (Abb. 1) zusammen, so ist die Montage recht einfach. Zuerst müssen die Magnete montiert werden. Dabei ist darauf zu achten, dass *sich die beiden Magnete anziehen*, wenn diese montiert sind. Um sie herumzudrehen, könnt ihr die Magnete leicht mit einer 4 mm Metallachse demontieren (Abb. 2).



Abb. 2: Magnetbausteine und Achse (zur Demontage der Magnete)

Wenn ihr die Magnetbausteine fertig habt müsst ihr nur noch die restlichen Bauteile (Abb. 3) montieren. Die vier Federkontakte werden paarweise in die Nuten der Bausteine 15 geschoben. Zum Schluss nur noch die Stecker in die Federkontakte stecken. Hierbei ist auf die *richtige Polung* achten.

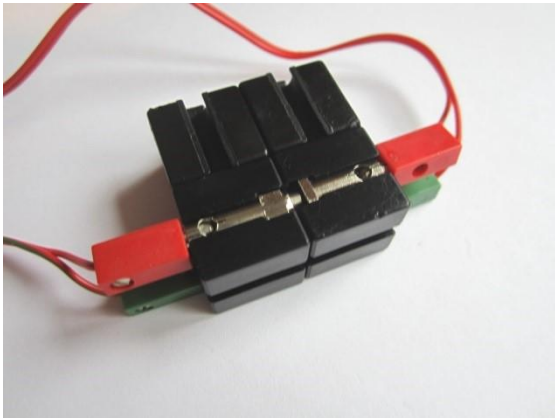


Abb. 3: Fertige Magnetkupplung

Dann befestigt ihr eine Seite der Magnetkupplung fest an eurem Modell. Die zweite Seite wird einfach angedockt und fertig ist die flexible Magnetkupplung für die Strom-

versorgung. Jetzt könnt ihr den Stromanschluss schnell und komfortabel zwischen den Modellen wechseln (Abb. 4).

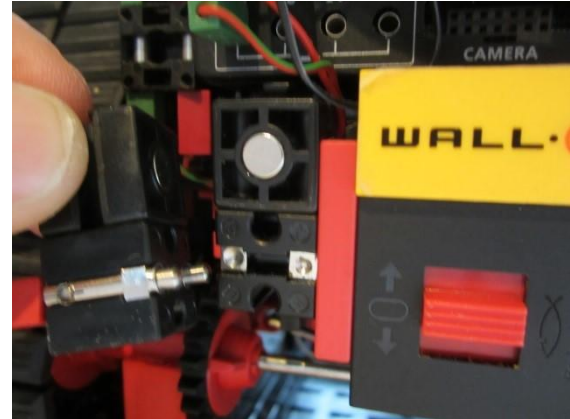


Abb. 4: Magnetkupplung am Wall-E

Man kann die Magnetkupplung auch zum Laden eines mobilen Roboters einsetzen, wie in Abb. 5 zu sehen ist.

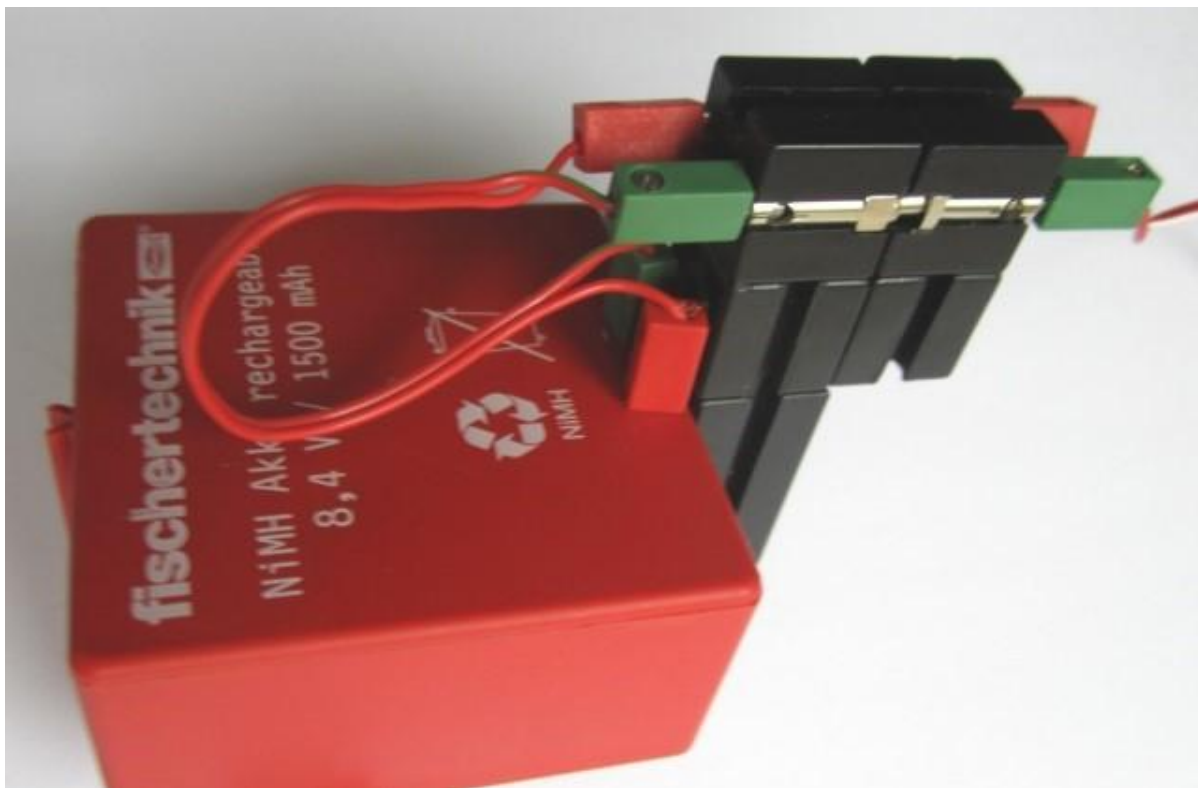


Abb. 5: Magnetkupplung zur Akku-Aufladung

Modell

Mini-Modelle (Teil 11): Flugzeug

René Trapp

Flieger, grüß' mir die Sonne...

Ein kleines Flugzeug aus weniger als zehn Teilen – da erübrigt sich sogar die Beschreibung des Zusammenbaus; Abb. 1 ist selbsterklärend.

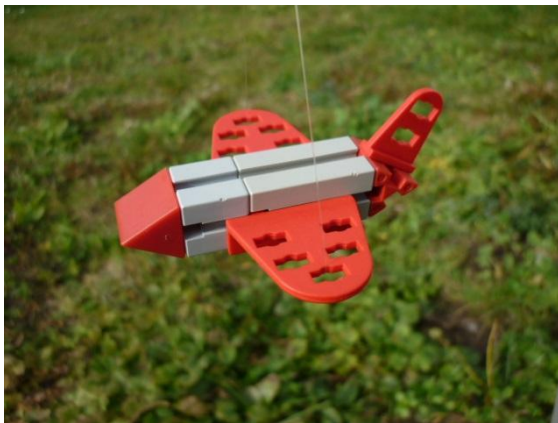


Abb. 1: Mini-Modell Flugzeug

Aber die Teileliste darf natürlich dennoch nicht fehlen.

St.	ft-Nr.	Bezeichnung
1	31003	Baustein 30, grau
1	31005	Baustein 15, grau
2	38252	Lagerbock
1	38253	Kupplungsstück 22,5
1	38309	V-Winkelstein 60°
1	31918	Winkelstein 60°, 3-Nut
1		Perlonfaden

Tab. 1: Einzelteilliste

Bisher erschienen:

- [1] René Trapp: *Mini-Modelle (Teil 1): Gabelstapler*. [ft:pedia 4/2013](#), S. 4-5.
- [2] Johann Fox: *Mini-Modelle (Teil 2): Panzer*. [ft:pedia 2/2014](#), S. 18-19.
- [3] René Trapp: *Mini-Modelle (Teil 3): Scheinwerfer*. [ft:pedia 3/2014](#), S. 11.
- [4] Johann Fox: *Mini-Modelle (Teil 4): Panzer*. [ft:pedia 3/2014](#), S. 12-13.
- [5] René Trapp: *Mini-Modelle (Teil 5): Traktor*. [ft:pedia 4/2014](#), S. 7.
- [6] Johann Fox: *Mini-Modelle (Teil 6): Bagger*. [ft:pedia 4/2014](#), S. 8-9.
- [7] Johann Fox: *Mini-Modelle (Teil 7): Hovercraft*. [ft:pedia 1/2015](#), S. 4-5.
- [8] René Trapp: *Mini-Modelle (Teil 8): Flugsaurier*. [ft:pedia 4/2015](#), S. 4.
- [9] Norbert Doetsch: *Mini-Modell (Teil 9): Motorrad*. [ft:pedia 1/2016](#), S. 6.
- [10] René Trapp: *Mini-Modell (Teil 10): Jojo*. [ft:pedia 1/2016](#), S. 4.

Tipps & Tricks

Neue ft-Teile selbst gemacht – 3D-Druck (4): Schleifringe (die Zweite), und überhaupt

Harald Steinhaus

Nachdem fischertechnik für diesen Sommer einen 3D-Drucker-Baukasten angekündigt hat, wird auch das Thema „Eigenbau von Teilen“ interessanter werden. Klar, man fängt an mit Schachfiguren und Nippes wie etwa Pokemon-Figuren, bei denen es auf Maßhaltigkeit und Kompatibilität nicht ankommt. Ich will hier nicht darüber diskutieren, ob ft-kompatible Teile aus einem ft-Drucker nun „ft-fremd“ sind oder nicht – dieser Beitrag behandelt einige Aspekte, auf die man beim 3D-Druck achten sollte.

Materialauswahl

Drucker im Hobby-Maßstab können mit ABS (Acrylnitril-Butadien-Styrol) oder mit PLA (Polylactide) arbeiten. ABS benötigt höhere Temperaturen und verzieht sich gerne, wenn man keine beheizte Grundplatte hat.

ABS ist spröde, und es scheint auch mit der Zeit zu schrumpfen. Das mag in vielen Fällen nicht viel ausmachen, aber wenn man z. B. ABS-Teile mit Kugellagern versehen hat, wird es ärgerlich. Das gelbe Hypozykloid-Getriebe von der Convention 2015 und die [Drehkränze von MisterWho](#) haben mittlerweile Risse in den Gehäusen. ABS reißt auch, wenn man sehr dünne

Wandstärken verwendet (Eigenbau-P-Betätiger in der Bildmitte).

Also heißt die Devise: bei ABS lieber einen Millimeter mehr Material und lieber 1/10 mm mehr Toleranz. Bei Kugellagern hilft auch eine deutlich zu große Aufnahme, die man mit elastischem Futter wieder auffüllt. Das geht prima mit Teflonband aus dem Sanitärhandel.

PLA wird bei geringeren Temperaturen verarbeitet und es ist ein wenig elastischer. Dafür ist PLA biologisch abbaubar, d. h. ein PLA-Bauteil kann unter tropischen Bedingungen innerhalb etlicher Wochen vergammeln.

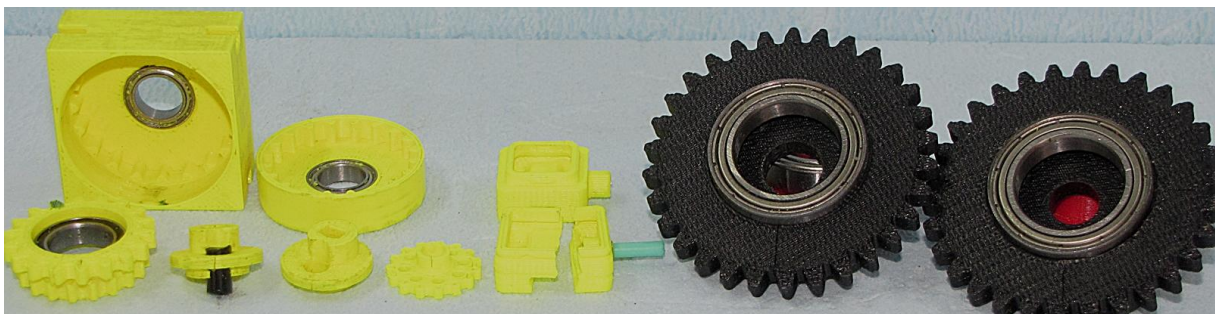


Abb. 1: Risse in Teilen aus ABS (Z30: auf 6 Uhr)

Düsen-Durchmesser

Drucker mit großer Düse können mehr Material pro Zeit durchschleusen und sind bei gegebenem Bauvolumen schneller fertig. Bei feinen Details erreichen sie aber schneller ihre Grenzen. Wände können nicht dünner sein als der Düsendurchmesser. Die nächstgrößere Wandstärke beträgt zwei Düsendurchmesser. Definiert man ein Maß dazwischen, dann hängt es von der Slicer-Software ab, was passiert (ausprobieren – siehe „Kalibrierung“).

Wählt man eine Wandstärke zwischen dem zweifachen und dem dreifachen Düsendurchmesser, kann man entweder zwei unabhängige Wände erhalten oder eine locker mit Material gefüllte Wand in der angegebenen Stärke. Beim Slic3r (c't 07/16) kann es passieren, dass Zwischenräume mit Material befüllt werden, obwohl dort der Platz nicht reicht. Das führt zu überschüssigem Material, das nach oben herausquillt und der Druckdüse in der nächsten Höhenlage im Weg ist.

Hat man dies alles berücksichtigt, kann es immer noch zu Überraschungen kommen, wenn Besitzer verschiedener Drucker A und B ihre CAD-Daten austauschen und die Drucker etwa mit Düsen von $A = 0,5 \text{ mm}$ und $B = 0,35 \text{ mm}$ bestückt sind, so dass ganz unterschiedliche Teile herauskommen.

Kalibrierung

Der Slicer tut sein Möglichstes, um mit der gegebenen Düse das Teil möglichst genau zu erzeugen. Bei großen Rechtecken geht das auch ohne Weiteres, aber beim 4-mm-Loch für eine ft-Achse fängt der „Spaß“ schon an. Üblicherweise rechnet ein Slicer damit, einen Materialstreifen zu erzeugen, der halb-und-halb links und rechts von der gefahrenen Bahn aufgetragen wird. Bei Kurven erhält aber die innere Seite mehr Material, da die gleiche Menge davon auf einem kürzeren Weg anfällt. In der Folge haben kleine Löcher immer Untermaß, und

das auch noch in Abhängigkeit vom Durchmesser.

Außerdem verlaufen ganz eng benachbarte Bahnen ineinander, sodass aus ganz schmalen Spalten (weit unter 1 mm) schnell Vollmaterial wird. Wenn der Slicer derartige Einflüsse nicht oder nicht richtig berücksichtigt, muss man schon im CAD-Programm Korrekturwerte für kleine Konturen einarbeiten. Hierbei hilft es, ein Musterteil mit gestuft breiten Schlitzten und Löchern auszudrucken und daraus die Korrekturwerte zu entnehmen. Weil der Drucker in der Z-Achse (Höhe) feiner auflösen kann als in X und Y, muss das Ganze zweimal geschehen. Im Einzelnen kann man Maße gebrauchen für:

- Bohrung für ft-Achsen und Kugellager;
- Langloch für S-Riegel;
- Langloch für Rastachse;
- ft-Nut;
- Aufnahmeloch für S-Strebe.

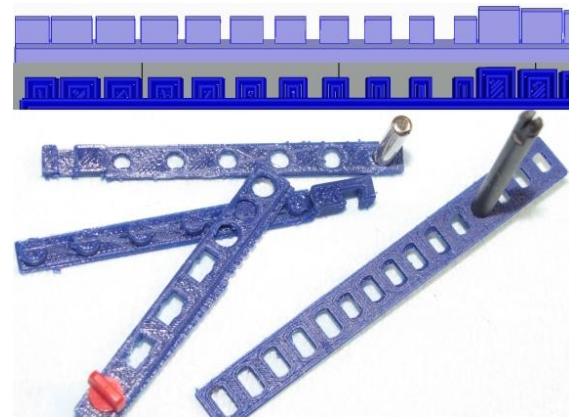


Abb. 2: Gestufte Lücken, Stifte und Löcher
(oben: CAD, Mitte: Slicer)

Nachbearbeitung

„Überflüssig“ gedrucktes Material fällt aus verschiedenen Gründen an:

- **Bett** (*brim*): Das Teil braucht ein Bett, d. h. es muss auf der Druckplattform mit einer flach aufliegenden Schicht aus Material umgeben werden, damit es sich beim Erkalten nicht verzieht und von der

Platte ablöst. Das ist schon bei ca. 3 cm Kantenlänge der Fall.

Der Slicer CURA hat hier einen Fehler: eine Schräge (Fase) an einer Seitenwand wird ignoriert, d. h. das Bett beginnt erst dort, wo die Projektion der Wand auf die Plattform trifft, und ist damit nutzlos (Abhilfe: Verzicht auf Fasen an Außenwänden; oder das Bett als Teil des Objekts konstruieren).

- **Stützmaterial:** Überhänge und Hohlräume werden mit einer waffelartigen, leicht zu zerbröselnden Struktur aus Material gefüllt, damit oberhalb davon nicht ins Leere gedruckt werden muss.
- **Lagersitze:** auch wenn man Aufnahmen für Kugellager mit reichlich Übermaß vorsieht, sollten die Randflächen geglättet werden.
- **Spinnfäden:** die Düse kann den Materialfluss nicht abrupt stoppen. Sie spinnst bei ihren Leerfahrten ein Muster aus dünnen Fäden kreuz und quer über das Bauteil. Die Fäden beginnen und enden in spürbar großen Fransen an den Bauteilen.

Zum Entfernen überschüssigen Materials braucht man die üblichen Geräte: Sandpapier, Feile, Taschenmesser, Bohrer, Reibahle. Bohrer sollten nur von Hand im Loch gedreht werden, damit man das Material nicht wieder aufschmilzt. Hierbei helfen einzeln erhältliche Bohrfutter als Handgriffe. Als Bauprojekt lohnt sich auch, den Bohrschaft quer mit einem Stift zu versehen und da drum herum einen Handgriff aus dem 3D-Drucker zu konstruieren.

Für gehobene Ansprüche empfiehlt sich eine Drehbank. Damit diese ihr Können ausspielen kann, muss das Teil aber gespannt werden können, was wiederum bei einem Dreibackenfutter am besten geht, wenn das Teil drei Ecken oder Vielfache davon hat. Bei einem Sechseck können drei Flächen zum Spannen in der Maschine dienen und die anderen drei stehen für

weitere Funktionen im Modell zur Verfügung.

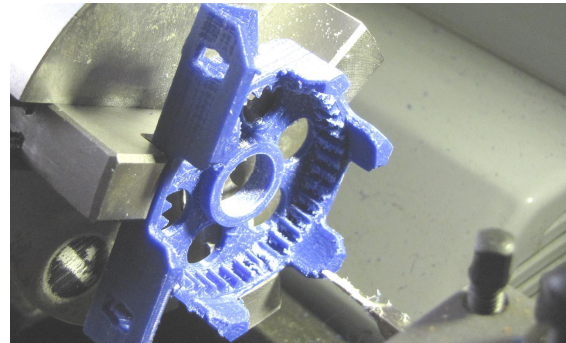


Abb. 3: Das Sechseckige Gehäuse passt ins dreieckige Spannfutter

Positiv oder Negativ

Hier geht es um komplexe Konstruktion mit Bohrungen oder Luftkanälen und dergleichen: Es kann einfacher sein, zuerst das Negativ zu konstruieren. Also, nicht einen Block aus dem Vollen, dem man Material wegnimmt, sondern ein Gerüst aus „Löchern“, das man im letzten Schritt vom Vollmaterial subtrahiert.

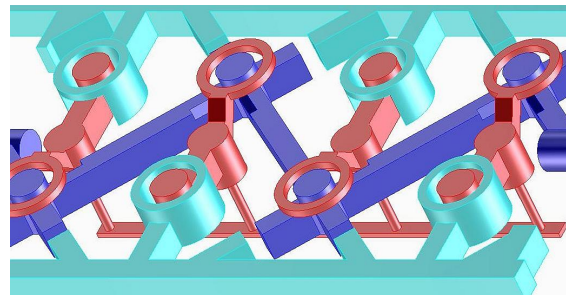


Abb. 4: Luftkanäle als Negativ (4-fach Ventil)

Verstopfte Düse

Eine verstopfte Düse kann man sich leicht „einfangen“: man definiere ein Modell, das entlang des Verfahrwegs sehr oft zwischen „drucken“ und „nicht drucken“ abwechselt, etwa ein großes Zahnrad mit sehr feiner Verzahnung. Beim Übergang von „drucken“ auf „nicht drucken“ zieht der Extruder den Materialfaden zurück, bei „drucken“ schiebt er ihn vor. Beim häufigen Wechsel verdunstet ein Bestandteil des Materialfadens, der für die elastischen Eigenschaften sorgt (so zumindest meine

unfachmännische Erklärung). Jedenfalls ist nach wenigen Zyklen dieser Art die Düse dicht, um den Drucker herum riecht es „chemisch-technisch“, und nichts geht mehr, obwohl die Maschine den Druckkopf fleißig weiter umherfährt.

Abhilfe: Um die Düse von dem darin fest gebackenen Material zu befreien, braucht man einen Stößel, Meißel oder Bohrer mit dem passenden Durchmesser von (je nach Typ) 0,35 mm oder 0,5 mm. Diese Werkzeuge sind durchaus zu haben, wenn man nur weiß, wo: Wenn Biologen eine Stahlnadel brauchen, um ein präpariertes Insekt an einer Tafel fest zu machen, ist nicht jede Nadel recht. Es gibt, jeweils im 100er Pack, Insektennadeln von 0,2 mm bis hinauf zu 1,2 mm mit Zwischenenwerten bei 0,3 mm, 0,35 mm, 0,4 mm und anderen. Dazu braucht man noch ein Spannfutter, das derart kleine Durchmesser auch greifen kann – passende Handwerkzeuge gibt es dort, wo es die Nadeln gibt. Wenn man neue Düsen für 30 Cent bekommen kann, ist allerdings die Frage eher, ob man sich das Gefummel mit den spitzen Nadeln antun mag.



Abb. 5: Bordwerkzeug für Düsenreinigung und Nachbearbeitung

Kühltürme

Die hier besprochenen 3D-Drucker für den Heimwerker beruhen auf dem Auftragen von schmelzbarem Material. Das Material der obersten Schicht muss hinreichend fest geworden sein, bevor die nächste Schicht darüber gedruckt werden kann. Der Druckkopf sollte seine eigene Bahn nie kreuzen und eine eben erst mit Material befüllte

Stelle so spät wie möglich wieder (in der nächsten Ebene) besuchen. Cura verwaltet eine Timer-Variable und schickt ggf. den Kopf „ins Freie“, um ihn erst nach einer kurzen Pause wieder an die Arbeit zurückzurufen. So weit, so gut. Cura versucht aber auch, die Verfahrswege kurz zu halten und fährt deshalb eine Ebene links herum und die nächste rechts herum. Genau dabei wird aber bei kleinen Teilen die frische Bahn auf das noch nicht ausgehärtete Material der vorigen Schicht aufgebracht. Die Folge: das Teil sackt zusammen und beult sich zur Seite aus.

Abhilfe: Schlanke Teile nie allein drucken, sondern mindestens als Paar. Oder, wenn man wirklich nur eins davon braucht und das Teil insgesamt viel Material benötigt, einen schlanken Turm mit hinreichend großem Sockel als zusätzliches Teil in den Druckjob geben und diesen Turm etwa 4 cm entfernt vom eigentlichen Teil ausrichten. Die nunmehr erzwungenen Leerwege sorgen fürs Auskühlen der alten Schicht, bevor die neue aufgetragen wird.

Bildergalerie

Kaum macht man etwas richtig (und spendiert noch zwei oder drei Probedrucke), schon gibt es schöne Teile: Hier sind ein paar davon versammelt.



Abb. 6: Betätiger, 3D und nicht-3D

Bild 6 zeigt zwei Pneumatik-Betätiger, davon einer aus dem 3D-Drucker (in blau und auf dem BS30 montiert); der andere besteht aus einem ft-Reifen 23, einer ft-Felge 23 und einer P-Düse als Luftzufuhr mit Schlauchanschluss. Die gelbe Membran

gibt es als Meterware überall, wo man Reha-Gerätschaften und Rollatoren kaufen kann. Das blaue Gummiband („Thera-Band“) gibt es gelegentlich bei Aldi und ist um Einiges straffer als das gelbe.

Bild 7 zeigt zwei Hypozykloidgetriebe. Das gelbe ist aus ABS, mit einer Übersetzung von 1:18, wurde auf der Convention 2015 ausgestellt und hat mittlerweile Risse in fast allen Teilen. Das blaue stammt aus dem eigenen Drucker, ist in PLA ausgeführt und hat eine Übersetzung von 1:420 ($1/20 * 1/21$). Beide verwenden einen Adapterstein, der die Antriebsachse frei drehend durchlässt (optional auch mit Kugellager), rundherum vier ft-Nuten aufweist, im Boden eine Ausnehmung für die Mitnahme von S-Streben hat und der mit zwei Scheiben auch als ft-Nabe (allerdings ohne Klemmwirkung) dienen kann.

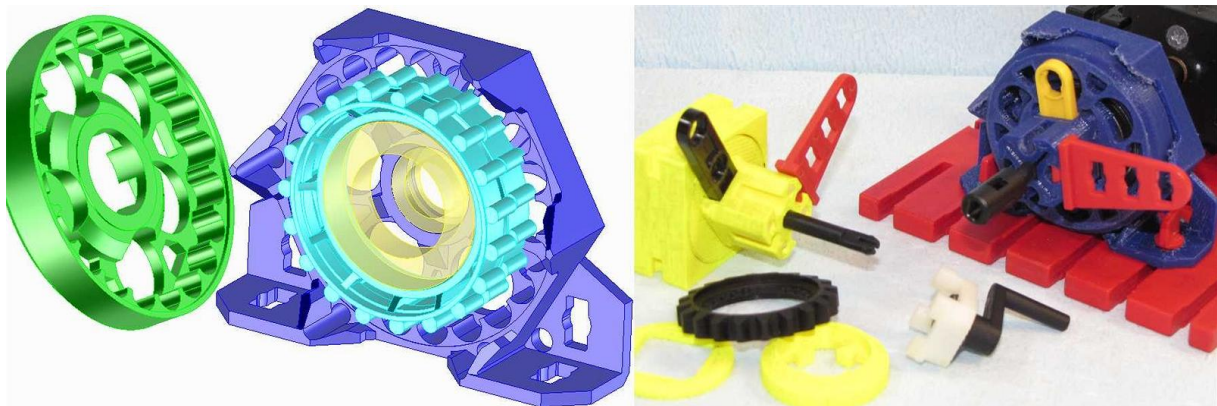


Abb. 7: Zwei Hypozykloidgetriebe

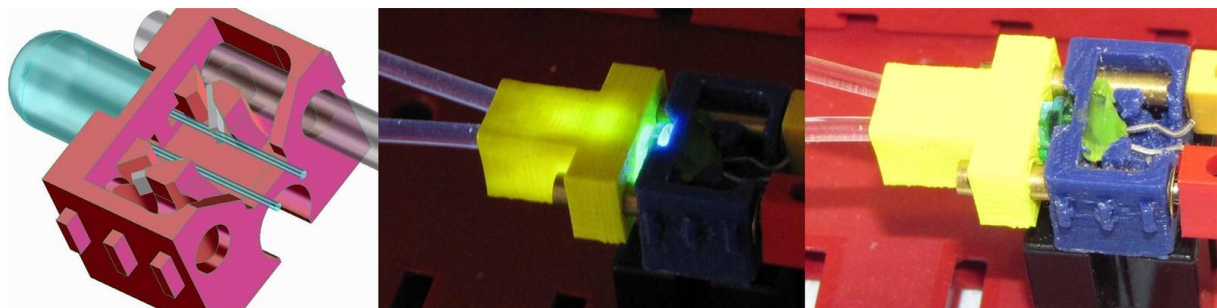


Abb. 8: LED-Stein für Draht-LED und SMD-Widerstände

Auf Bild 8 ist ein LED-Baustein zu sehen. Er sollte für meine vielen bedrahteten LEDs geeignet sein und im Gehäuse die Vorwiderstände enthalten. Die Vorwiderstände sind als SMD-Teile ausgeführt, und davon klemmt je einer zwischen den LED-Anschlussdrähten und einer der Steckerhülsen, über die der Stein angeschlossen wird. Ein bisschen grüne Knetmasse verhindert, dass die Widerstände eigene Wege gehen. Die seitlichen Rippen sind nötig, weil die dünnen Wände sonst während des Drucks zusammen sacken können. Der gelbe Vorsatz dient als Übergang auf Lichtleitfasern, die im Handel als „Bilderdraht“ (zum Aufhängen von Bildern in Galerien etc.) geführt werden.

Zum Abschluss zeigt Bild 9 den vor langer Zeit angekündigten Schleifring mit nunmehr neun Bahnen [2]. Die Kupferscheiben mit angelöteten Drähten werden mittels der gedruckten Scheiben zu einem Stapel verbunden und mittels vier Wattestäbchen-Röhrchen zusammen gehalten. Als Kontakte dienen die ft-Stecker ohne Hülle, die schwenkbar gelagert sind und mittels eines Streifens Moosgummi angeedrückt werden.

Was bisher erschien...

- [1] Harald Steinhaus: *Neue ft-Teile selbst gemacht – 3D-Druck*. [ft:pedia 2/2013](#), S. 4-10.
- [2] Harald Steinhaus: *Neue ft-Teile selbst gemacht – 3D-Druck (2): Schleifringe*. [ft:pedia 1/2015](#), S. 68-72.
- [3] Johannes Visser: *Neue ft-Teile selbst gemacht – 3D-Druck (3): Innenzahnkranz*. [ft:pedia 1/2015](#), S. 73-83.

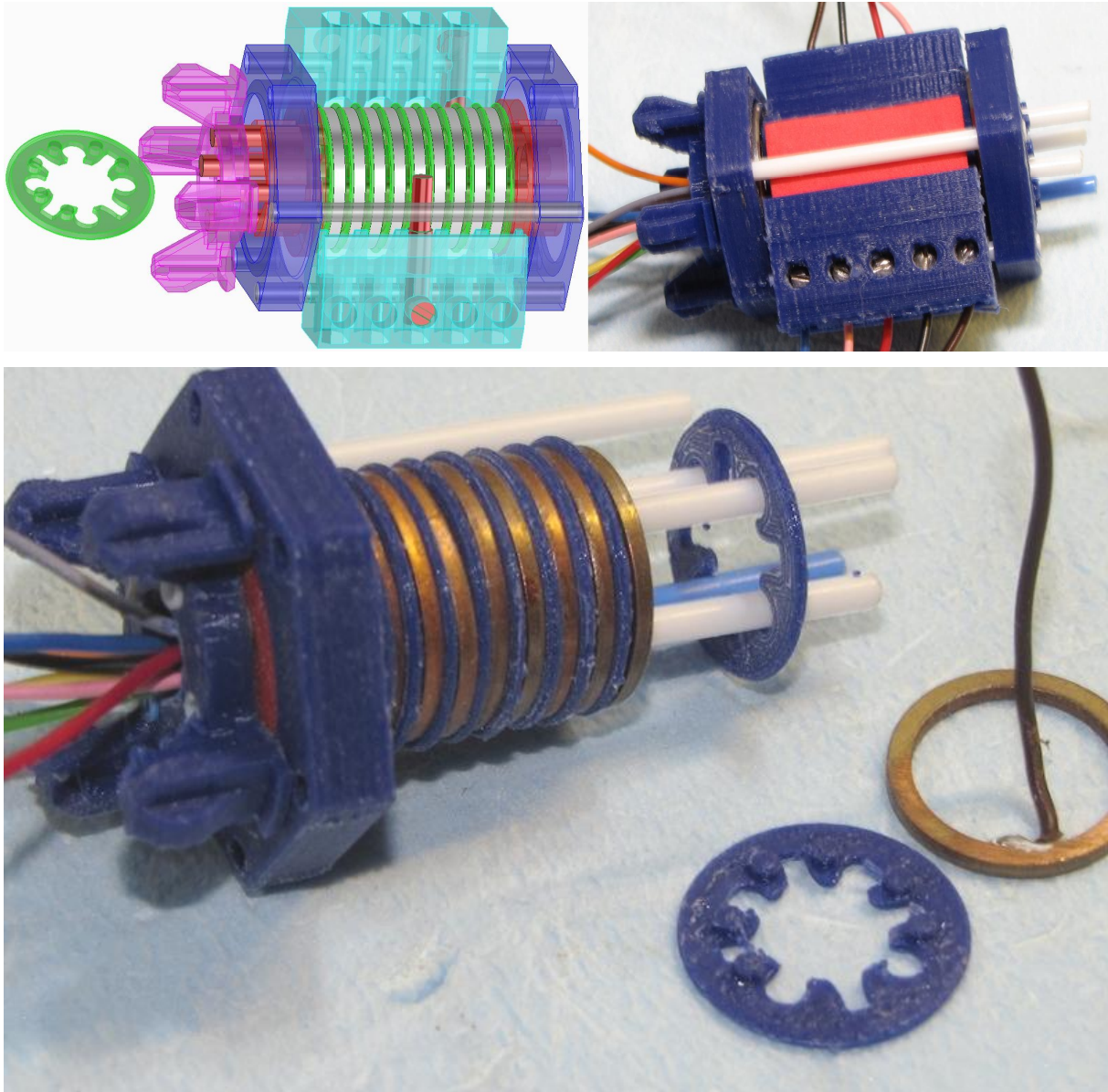


Abb. 9: Schleifring, 9-fach

Tipps & Tricks

Modellfotografie

Thomas Püttmann

Wer seine fischertechnik-Modelle anderen in der ft:pedia oder im Internet präsentieren möchte, muss sie fotografieren. In diesem Beitrag beschreibe ich, wie ich dabei vorgehe.

Ausrüstung

Gleich vorweg: Ich bin kein Profifotograf. Meine Kenntnisse sind beschränkt und meine Ausrüstung klein. Was ich hier wiedergebe, ist kein Expertenwissen. Ich hoffe trotzdem, dass der Artikel für den einen oder die andere nützlich sein kann, wenn er oder sie öfter fischertechnik-Modelle fotografiert.

Kamera

Ich arbeite mit der Kompaktkamera Lumix TZ10 von Panasonic, die vor ein paar Jahren etwas mehr als 200 € gekostet hat. Kompaktkameras haben in der Regel kleine Bildsensoren. Dadurch ist es sehr einfach, Fotos mit großer Tiefenschärfe zu bekommen. Für die technische Dokumentation von Modellen halte ich das für angebracht.

Von den vielfältigen Einstellungsmöglichkeiten der Kamera nutze ich meist nur wenige. Die wichtigsten sind für mich die manuelle Belichtungskorrektur und die zeitverzögerte Auslösung.

Stativ

Für die Modellfotografie ist es vorteilhaft, wenn die Kamera schnell in verschiedene, zum Teil extreme Positionen gebracht werden kann. Mit einem herkömmlichen Fotostativ ist es zum Beispiel unmöglich, ein auf einem Tisch stehendes Modell senkrecht von oben aus einem Abstand von einem Meter zu fotografieren.

Abhilfe schafft ein Mikrofonstativ, auf das ich den Kopf eines billigen Kamerastativs geschraubt habe, siehe Abb. 1. Durch die doppelte Ausziehmöglichkeit lassen sich Kamerahöhen von zwei Metern und Fotos aus allen möglichen Richtungen erzielen. Natürlich empfiehlt es sich dabei, zeitverzögert auszulösen, da die Kamera durch den Druck auf den Auslöser schwingt.



Abb. 1: Ein Mikrofonstativ, auf das der Kopf eines billigen Kamerastativs montiert wurde

Hintergrund

Meiner Meinung nach fotografiert man fischertechnik-Modelle am besten vor weißem Hintergrund. Damit verhindert man vom Modell ablenkende Objekte im Hintergrund, lässt die Bauteile in ihren natürlichen Farben zur Geltung kommen und erzielt maximal viel Licht von hinten und unten. Da die schwarzen Bausteine und Grundplatten sehr viel Licht schlucken und Details dadurch oft nur schwer abbildbar sind, ist das sehr wichtig.

Um einen homogenen Hintergrund zu bekommen, kann man auf sogenannte Lichtzelte zurückgreifen. Die deutlichen Nachteile dieser Zelte sind, dass die Kameraposition sehr eingeschränkt ist und man sehr viel Licht in das geschlossene Zelt hineinbringen muss. Die günstigste und bei weitem stärkste Lichtquelle – die Sonne – kann dabei gar nicht genutzt werden.

Nachdem ich verschiedene im Haus befindliche Textilien wie Bettlaken und Tischdecken ausprobiert hatte, war mir klar, dass ich mattes weißes Papier verwenden wollte. Bei [Winkler&Volkmann](#) in Bochum fand ich ein Angebot für eine 50-m-Rolle unbeschichteten Plotterpapiers in 91,4 cm Breite für nur ungefähr 10 €



Abb. 2: Papierrolle und U-Ständer

Für diese Papierrolle habe ich mir einen einfachen U-Ständer aus drei alten Brettern

zusammengeschraubt, auf dem oben eine alte, passend abgelängte Gardinenstange primitiv mit zwei Schrauben drehbar gelagert ist, siehe Abb. 2. So kann man das Papier extrem schnell auf- und abrollen, bei ausgedehnteren Modellen einen ganzen Tisch überdecken lassen oder bei hohen Modellen vom Tisch bis auf den Boden ziehen. Damit kann ich einen Hintergrund ohne Kanten und Knicke erzeugen.

Schließlich lässt sich der komplette Aufbau einfach in den Keller tragen und nimmt vergleichsweise wenig Platz weg.

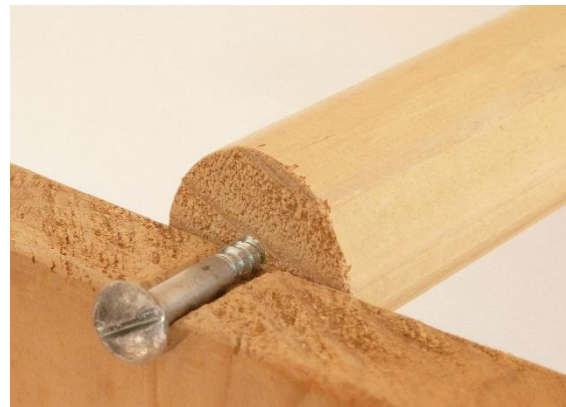


Abb. 3: Lagerung der Gardinenstange

Licht

Belichtungskorrektur

Wegen des weißen Hintergrunds muss die Belichtung manuell erhöht werden. Dadurch werden zum einen die Details in den Regionen, wo sich viele schwarze Bauteile befinden, sichtbar. Zum anderen verschwinden die Details des weißen Hintergrunds, der dadurch sehr homogen wird. Bei kleineren bis mittleren Baugruppen ohne Grundplatte liegt meine Belichtungskorrektur normalerweise zwischen 1 und $1\frac{2}{3}$. Etwas Vorsicht ist geboten, wenn gelbe Teile im Bild sind. Durch die Belichtungskorrektur können hier in ungünstigen Fällen Informationen unwiederbringlich verschwinden.

Blende

Verwendet man keine Kompaktkamera, so erhält man in automatischen Modi in der

Regel Fotos, die eine gewisse Tiefenunschärfe aufweisen. Dies ist bei der Personen- und Objektfotografie oft durchaus wünschenswert, da der Betrachter seine Aufmerksamkeit unmittelbar auf eine Tiefenebene konzentriert. Bei der technischen Dokumentation von Modellen möchte man in der Regel aber Details in allen Tiefen scharf abbilden. Um eine größere Tiefenschärfe herzustellen, kann man manuell die Blende weiter schließen, d. h. eine große Blendenzahl F wählen. Dadurch steigt natürlich die Belichtungszeit, was bei der Verwendung eines Stativs und der zeitverzögerten Auslösung allerdings kein Problem darstellt.

Lichtquellen

Wie oben schon geschrieben ist die beste und stärkste Lichtquelle die Sonne. Ich fotografiere am liebsten bei gutem Wetter in unserem Esszimmer, das zu zwei Seiten Fensterflächen und an den Wänden homogen weißen Kalkputz besitzt. Dadurch ist in der Regel keine weitere Lichtquelle erforderlich. Die meiste Zeit im Jahr steht die Sonne zu hoch, um direkt auf den Tisch zu scheinen, so dass es mit Schattenwürfen keinerlei Probleme gibt. Ansonsten schatte ich das Modell während des Fotografierens mit meinem Körper ab.

Bei schlechtem Wetter greife ich auf die Deckenleuchte, eine Energie sparende Leuchtstoffröhre mit 6000 Kelvin Farbtemperatur und einem Farbwiedergabeindex R_a von mehr als 90 zurück. Ein hoher Farbwiedergabeindex ist wichtig für natürlich aussehende Fotos. Bei Leuchten mit einem Farbwiedergabeindex von 80 und weniger besteht das Spektrum nur aus wenigen Linien bzw. Bereichen (siehe Abb. 4), während bei guten Leuchten Licht im gesamten Bereich von 400 nm bis 700 nm Wellenlänge möglichst gleich stark abgestrahlt wird. Sehr gut kontrollieren lässt sich

die Qualität von Leuchten mit einem Karton-Taschenspektroskop der Firma [Astromedia](#) für 7,80 € In einem anderen Raum unseres Hauses kommen übrigens hochwertige LED-Panels zum Einsatz, die eine phänomenal gute Farbwiedergabe besitzen und die ich inzwischen jeder Leuchtstoffröhre vorziehen würde.



Abb. 4: Spektrum einer billigen Leuchtstoffröhre (Foto: Anton, Quelle: Wikipedia GFDL 1.2)

Nachbearbeitung

In der Regel bearbeite ich Fotos nur sehr wenig nach. Die wichtigste Funktion ist sicherlich das Ausschneiden, um das Objekt ins Zentrum des Bildes zu bringen und überflüssige Ränder zu beseitigen. Diese wird von jedem Bildbearbeitungsprogramm bereitgestellt. Ab und zu benötige ich fortgeschrittene Funktionen wie stufenloses Drehen, das Anpassen des Weißpunkts und die Überlagerung mehrerer Aufnahmen.

Ich selbst nutze auf dem Mac den Graphic Converter 9 von Lemke Software – ein Programm, das für 34,95 € unglaublich viele Funktionen in exzellenter Qualität anbietet. Ein kostenloses Tool für Windows-Nutzer ist der [ft Community Publisher](#) von Stefan Falk, der außerdem ein komfortables Hochladen von Fotos in den Bilderpool der ft-Community ermöglicht.

Einen leicht grauen oder leicht schattigen Hintergrund zum Beispiel kann man mit diesen Programmen durch Anpassen des Weißpunkts beseitigen (GraphicConverter) oder durch Veränderung der Gamma-Korrektur bei anschließender Kontrast-Anpassung.

Grundlagen

Geradföhrungen

Dirk Fox

Die Geschichte der Geradföhrungen ist eng gekoppelt mit der Entwicklung der Dampfmaschine. Inzwischen gibt es zahlreiche weitere Anwendungen für Geradföhrungen. Der Beitrag stellt einige elementare Geradföhrungsgetriebe vor, die sich für unterschiedliche fischer-technik-Konstruktionen eignen.

Hintergrund

Eine der größeren Herausforderungen beim Bau der ersten Dampfmaschine, die die Kippbewegung des „Balancers“ in eine Rotation, also Drehbewegung umsetzte, war die exakt gerade Föhrung der Kolbenstange, die das auf einer Kreislinie verlaufende Ende des Balancers bewegen musste. Die Geradföhrung war erforderlich, um ein Verkanten oder Verbiegen der Kolbenstange zu verhindern.

In den frühen „Kolben-Wärmeleistungsmaschinen“ von [Thomas Newcomen](#) (1663–1729) von 1712 und auch den verbesserten Maschinen von [James Watt](#) (1736–1819), die in Bergwerken als Pumpen eingesetzt wurden, gab es dafür kein Erfordernis, da der Kolben lediglich eine Zugbewegung ausföhren musste; er wurde daher mit einer Kette am kreisförmig ausgebildeten Ende des Balancers befestigt. Das änderte sich, als Watt den Wirkungsgrad der Maschinen verdoppeln wollte, indem er den Kolben wechselseitig mit Dampf befüllte und damit die Kolbenstange auch eine Schubbewegung ausföhren ließ [1].

Wattsches Parallelogramm

Die Konstruktion der ersten Geradföhrung mit einem (Vier-) Gelenkgetriebe gelang James Watt im Jahr 1781. Damit konnte er die Auf- und Abbewegung des mit dem

Balancier verbunden oberen Endes der Kolbenstange dem Verlauf einer Geraden annähern. Nach eigenem Bekunden war er auf diese Erfindung besonders stolz: „I am more proud of the parallel motion than of any other invention I have ever made.“ Abb. 1 zeigt die entsprechende Abbildung aus der Patentschrift des Jahres 1784.

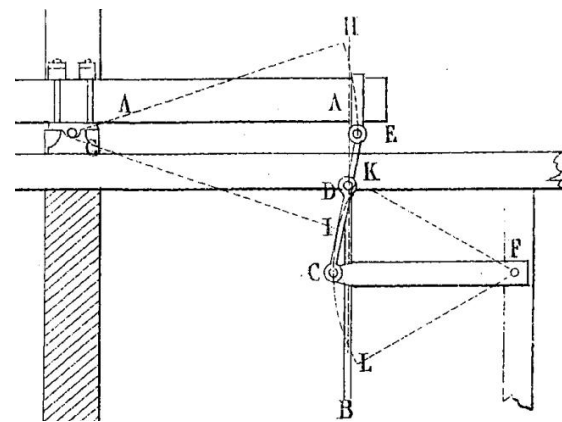


Abb. 1: Wattsche Geradföhrung,
GB Patent No. 1432 [2]

Oben im Bild erkennt man die rechte Hälfte des Balancers A, der um einen Drehpunkt (links oben) wippt. Das obere Ende der Kolbenstange B ist in Punkt D an einem durchgehenden Gestänge I befestigt, das auf der einen Seite mit dem Balancier (in Punkt E) und auf der anderen Seite mit einer Strebe (in Punkt C) verbunden ist, die sich um den Punkt F dreht. Kippt der Balancier, bewegt sich der Punkt D auf der senkrechten Geraden \overline{HL} .

Die Skizze in Abb. 2 zeigt die Geradföhrung und das zugehörige Gelenkgetriebe schematisch, mit etwas intuitiveren Bezeichnungen für die Gelenkpunkte.

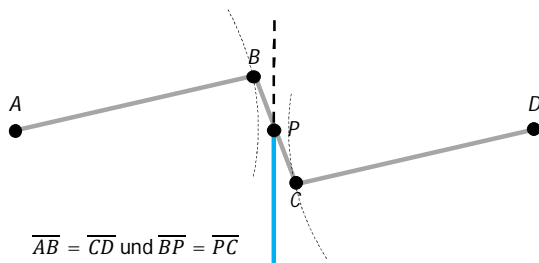


Abb. 2: Geradföhrung nach James Watt

Sind die in der Grafik angegebenen Bedingungen $\overline{AB} = \overline{CD}$ und $\overline{BP} = \overline{PC}$ erfüllt, bewegt sich der Punkt P näherungsweise auf einer Geraden. Dabei darf das Gelenkgetriebe allerdings nicht in die „Totlage“ geraten, da es dort sperrt. Die Totlage wird erreicht, wenn zwei benachbarte Elemente des Gestänges auf einer Geraden zu liegen kommen.

Im Übrigen funktioniert die Geradföhrung noch allgemeiner, wie man Watts Patentzeichnung entnehmen kann: P muss nicht in der Mitte von \overline{BC} liegen. Tatsächlich genügt es, dass gilt [3]:

$$\overline{AB} : \overline{CD} = \overline{CP} : \overline{CB}$$

Dabei liegt die Bewegungslinie des Punktes P umso näher an einer exakten Geraden, je größer das Verhältnis $\overline{AB} : \overline{BC}$ gewählt wird. Allerdings verkürzt sich dadurch zugleich die Länge des gerade verlaufenden Abschnitts der Bewegungslinie von P .

Die erste Wattsche Dampfmaschine mit Geradföhrung, die einen Rotationsantrieb realisierte, wurde 1782 in London in Betrieb genommen: Es war der erste Motor der Weltgeschichte. Entscheidend für diese Entwicklung war die Entdeckung des Geradföhrungsgetriebes durch James Watt. Abb. 3 zeigt eine Dampfmaschine dieser Bauart aus dem Jahr 1784.

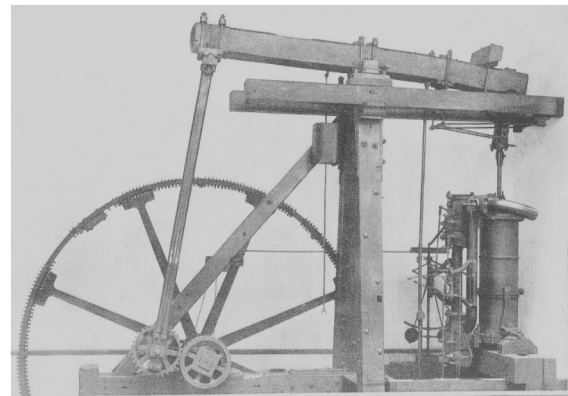


Abb. 3: Wattsche Dampfmaschine von 1784 (Abbildung: Artur Fürst [4])

Bei dieser Maschine verwendete Watt bereits eine Variante der einfachen Geradföhrung aus seinem Patent (im Bild vom Balken verdeckt): das so genannte „Wattsche Parallelogramm“. Abb. 4 zeigt diese Geradföhrung als Schemabild, in der man das ergänzte Gelenkparallelogramm \overline{BEFC} (in orange) gut erkennen kann. Der Punkt P bewegt sich nun auf einer zu der durch Punkt F verlaufenden, parallelen Geraden; an Punkt F wird das Ende der Pleuelstange befestigt.

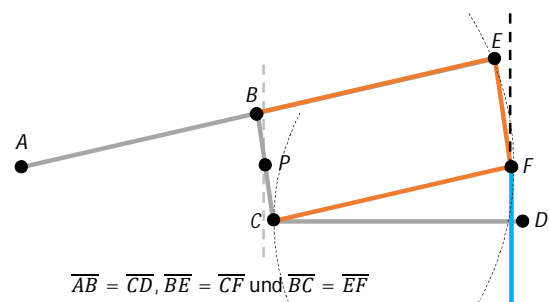


Abb. 4: Wattsches Parallelogramm

Wenn die in der Grafik angegebenen Bedingungen $\overline{AB} = \overline{CD}$, $\overline{BE} = \overline{CF}$ und $\overline{BC} = \overline{EF}$ erfüllt sind, bewegt sich das obere Ende F der Pleuelstange entlang einer senkrechten Geraden [3]. Mit dieser Konstruktion konnte Watt die Dampfmaschine kompakter bauen, denn sein Hebel war nun \overline{AE} (statt \overline{AB}).

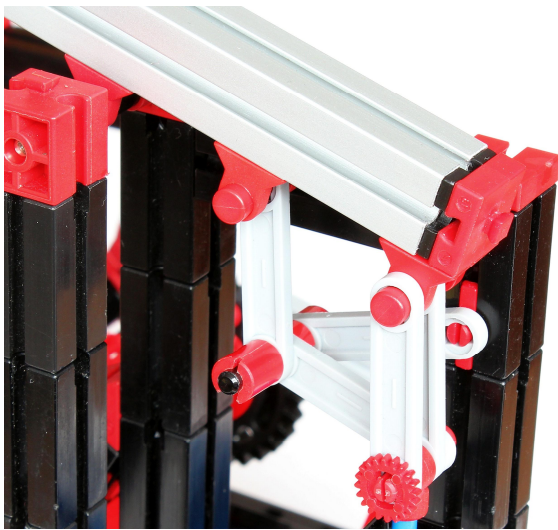


Abb. 5: Wattsche Geradföhrung mit fischertechnik

Abb. 5 zeigt die Konstruktion dieser Geradföhrung mit fischertechnik (siehe auch [1]). Inzwischen gibt es zahlreiche weitere Anwendungen für die Wattsche Geradföhrung, wie z. B. die Föhrung einer gefederten Rad-aufhangung. Abb. 6 zeigt eine solche Konstruktion mit fischertechnik.

Tschebyscheffsche Geradföhrung

Der russische Mathematiker [Pafnuti Lwowitsch Tschebyscheff](#) (1821-1894) fand 1850 eine weitere (Naherungs-) Losung fur eine Geradföhrung, ebenfalls mit einem Viereckenge triebe. Dazu verlegte er den Punkt D neben den Drehpunkt A und kreuzte die Verbindungen \overline{AB} und \overline{CD} . Diese Geradföhrung wurde daher auch als „Kreuzlenker“ bezeichnet [5].

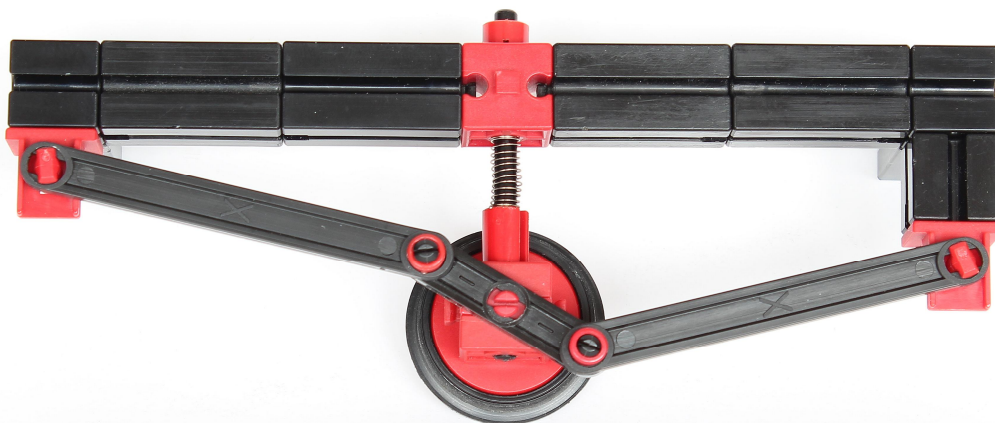


Abb. 6: Geradföhrung einer gefederten Achse

Anders als die Losung von James Watt unterliegt diese Konstruktion jedoch mehreren Einschrankungen. So mussen nicht nur die Verbindungen \overline{AB} und \overline{CD} gleich lang sein, sondern auch in einem festen Abstand und Langenverhaltnis zu \overline{BC} stehen: $\overline{AD} = 2 \cdot \overline{BC}$ und $\overline{AB} = \frac{5}{4} \cdot \overline{AD}$ (siehe Abb. 7).

Die Abweichung des Punktes P von einer exakt geraden Linie liegt – innerhalb des Bewegungsabschnittes, der von den Senkrechten durch die Punkte A und D begrenzt wird – bei maximal 0,5% der Lange \overline{BC} [3].

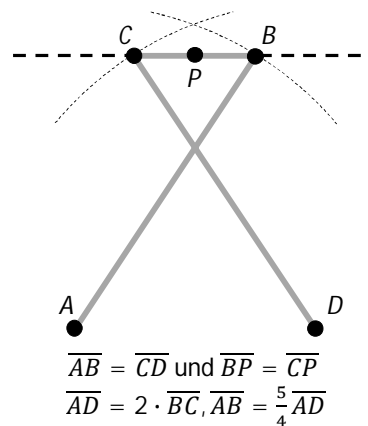


Abb. 7: Geradföhrung nach Tschebyscheff

Fur diese Geradföhrung gibt es auch eine elegante Anwendung: Mit einer kleinen Erweiterung um eine vierte Gelenkstrebe (siehe Abb. 8, 9) lasst sich eine geradlinige Horizontalverschiebung einer Flache durch die Punkte P und F realisieren [3, 6, 7].

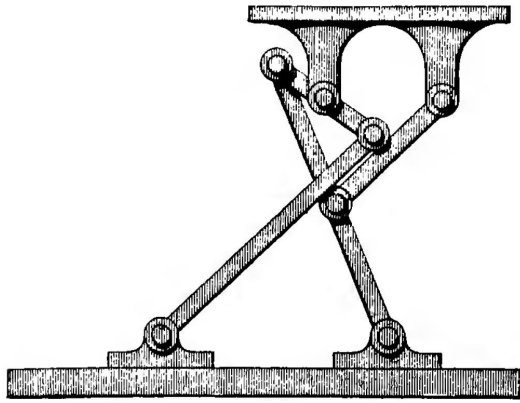


Abb. 8: Horizontalverschiebung [6]

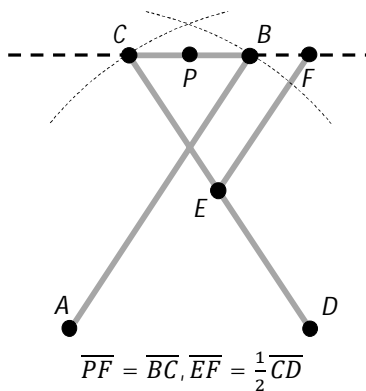


Abb. 9: Gelenkgetriebe zur geradlinigen Horizontalverschiebung einer Fläche

Abb. 10 zeigt ein prototypisches fischer-technik-Modell, mit dem sich die obere Grundplatte um knapp 6 cm horizontal verschieben lässt.

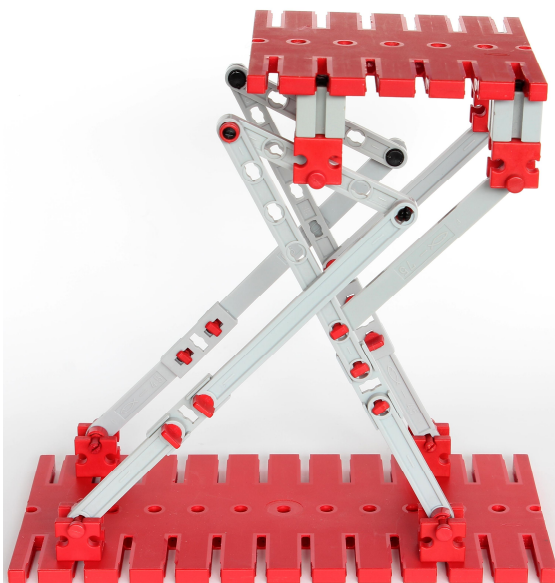


Abb. 10: Horizontalverschiebung

Geradführung von Roberts

Eine weitere Variante der näherungsweise Geradführung von James Watt stammt vom britischen Werkzeugmaschineningenieur und Unternehmer [Richard Roberts](#) (1789-1864). Roberts hob die Kreuzung der Verbindungen in Tschebyscheffs Geradführung auf und verlegte den Geradführungspunkt P nach unten (Abb. 11).

Die Geradführung von Roberts benötigt weniger Einschränkungen als die Tschebyscheffs: Es müssen $\overline{AB} = \overline{CD} = \overline{BP} = \overline{PC}$ und $\overline{AD} = 2 \cdot \overline{BC}$ sein, damit sich P näherungsweise auf einer Geraden bewegt.

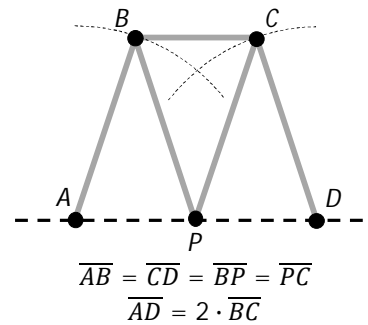


Abb. 11: Geradführung nach Roberts

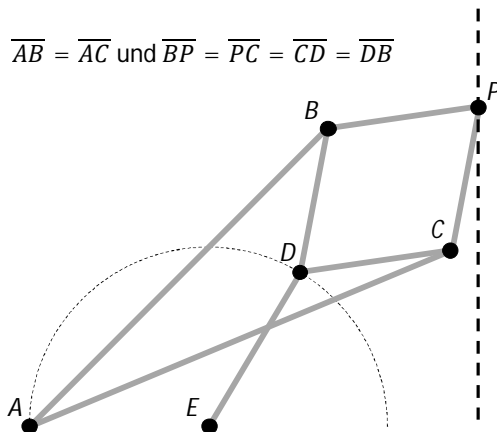
Die Annäherung an eine exakte Geradführung ist bei dieser Konstruktion sehr gut: Die Bewegung des Punktes P weicht auf der Verbindungslinie \overline{AD} maximal um 0,12% der Länge \overline{BC} von einer Geraden ab.

Peaucelliers exakte Geradführung

Alle bisher betrachteten Geradführungen waren lediglich (wenn auch z. T. sehr gute) Näherungen. Die erste exakte Geradführung (im zweidimensionalen Raum) gelang 1864 dem französischen Offizier [Charles Nicolas Peaucellier](#) (1832-1919) – dafür wurde er von der *Académie Française* mit dem *Prix Montyon* ausgezeichnet [6]. Auch wenn deren praktische Einsatzmöglichkeiten wegen der zahlreichen Gelenke (6) beschränkt sind, war sie eine Sensation – denn viele Mathematiker (unter ihnen Tscheby-

scheff) hielten die Konstruktion einer exakten Geradföhrung mit einem Gelenkgetriebe für unmöglich.

Die Skizze in Abb. 12 zeigt die Konstruktion. Die Punkte $BPCD$ bilden eine Raute, und die Strecken \overline{AB} und \overline{AC} müssen gleich lang gewählt werden. Dann bewegt sich der Punkt P exakt auf einer Geraden [3, 5, 8].

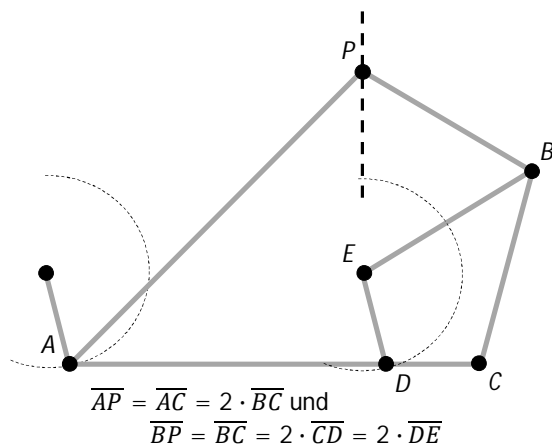


$$\overline{AB} = \overline{AC} \text{ und } \overline{BP} = \overline{PC} = \overline{CD} = \overline{DB}$$

Abb. 12: Exakte Geradföhrung von Peaucellier

Der Nachteil dieser Geradföhrung besteht in den zusätzlichen Gelenken und Streben, die für das Getriebe erforderlich sind. Sie reduzieren die Stabilität, wodurch es sich weniger für Mechanismen eignet, in denen größere Kräfte übertragen werden.

Eine Variante dieser Geradföhrung stammt vom englischen Mathematiker [James Joseph Sylvester](#) (1814-1897), siehe Abb. 13. Auch sie setzt eine Kreisbewegung in eine geradlinige Auf- und Abbewegung um.



$$\overline{AP} = \overline{AC} = 2 \cdot \overline{BC} \text{ und } \overline{BP} = \overline{BC} = 2 \cdot \overline{CD} = 2 \cdot \overline{DE}$$

Abb. 13: Geradföhrung von Sylvester

Ende des 19. Jahrhunderts wurden von Ingenieuren und Mathematikern zahlreiche weitere, sowohl exakte als auch genäherete Geradföhrungen entwickelt [6]. Für fischertechnik-Modelle ist jedoch noch eine weitere Klasse von Geradföhrungen von Bedeutung, die im Maschinenbau erst mit verfeinerten Verarbeitungsmethoden und geeigneteren Materialien stabil ausgeföhrt werden konnte.

Geföhrte Geradföhrung

Die gemeinhin [John Scott Russell](#) (1808-1882), dem Erbauer der Great Eastern (dem Schiff, das 1866 Europa und Amerika mit einem Telegrafenkabel verband [1] – aber das ist eine andere Geschichte) zugeschriebene Geradföhrung wurde bereits im Jahr 1801 von [James White](#) auf der Weltausstellung in Paris vorgestellt (und von Napoleon ausgezeichnet) sowie 1803 vom Uhrmacher [William Freemantle](#) in Großbritannien patentiert (British Patent No. 2741 vom 17.11.1803) [8, 9].

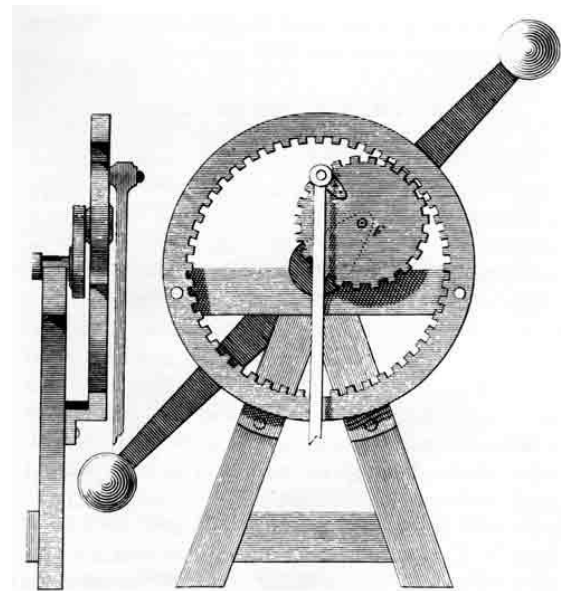


Abb. 14: Geradföhrung von James White [8, 9]

Diese Geradföhrung nutzt die Erkenntnis, dass jeder beliebige Punkt eines Innenkreises, der an einem Außenkreis mit doppeltem Durchmesser „abrollt“, sich auf einer

Geraden durch den Mittelpunkt des Außenkreises bewegt (siehe Punkt 5 in Abb. 15).

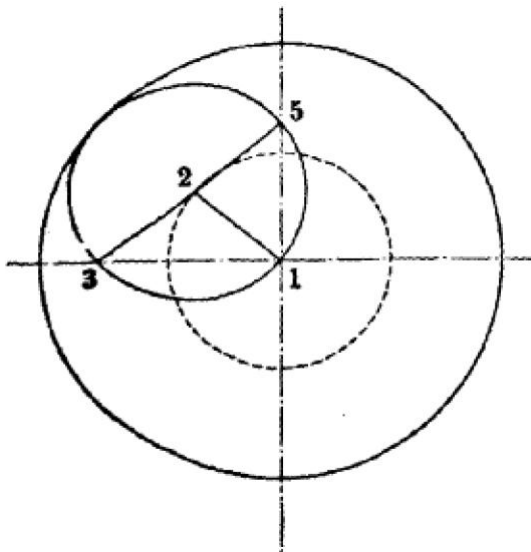


Abb. 15: Darstellung von Franz Reuleaux [5]

Aus den Verbindungen 3-5 und 2-1 lässt sich ein Getriebe für eine exakte Geradföhrung konstruieren, bei der in Punkt 3 eine „Schleife“, d. h. ein geföhrtes Gelenk eingebaut wird (Abb. 16).

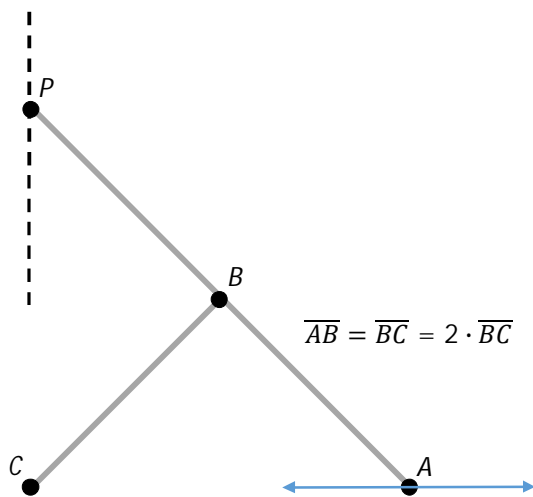


Abb. 16: Ellipsenlenker

Interessant an dieser von [Franz Reuleaux](#) (1829-1905) als „Ellipsenlenker“ [5] bezeichneten Konstruktion ist, dass Punkt *P* am Ende der Aufwärtsbewegung langsamer wird. Diese Eigenschaft machte sich z. B. der Kompressor von Owen (Abb. 17) mit einem doppelten Ellipsenlenker zu Nutze.

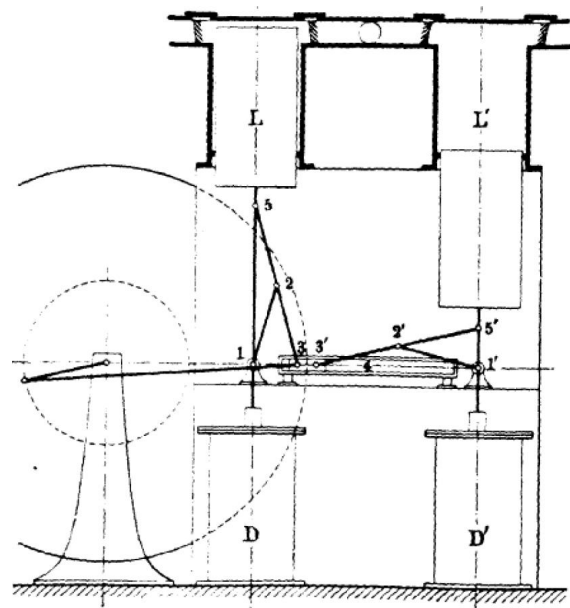


Abb. 17: Kompressor von Owen (1881) [5]

Der auch gelegentlich Scott-Russell-Getriebe genannte Ellipsenlenker ist übrighs der Grundmechanismus des Scherenhubs [10].



Abb. 18: fischertechnik-Ellipsenlenker

Mechanische Koppelgetriebe bieten zahlreiche Möglichkeiten, bestimmte Bewegungsabläufe mit sehr einfachen Mechanismen zu erzeugen – und lassen sich oft verblöffend leicht mit fischertechnik realisieren [11, 12, 13]. Wer mehr Mechanismen dieser Art kennenlernen möchte, dem sei das (nur noch antiquarisch erhältliche) Buch von Brian Bolt *Was hat der Bagger mit Mathematik zu tun?* [14] ans Herz gelegt.

Quellen

- [1] Dirk Fox, Thomas Püttmann: [Bauen, erleben, begreifen: Technikgeschichte mit fischertechnik](#). dpunkt-Verlag, Heidelberg 2015.
- [2] James Watt: *Fire and Steam Engines*. GP Patent No. 1432, 1784.
- [3] John Bryant, Chris Sangwin: *How Round is Your Circle? Where Engineering and Mathematics Meet*. Princeton University Press, 1934. Siehe auch die [Webseite](#) und das [Video zum Buch](#).
- [4] Artur Fürst: *Das Weltreich der Technik. Band 4: Lastenförderung, Kraftmaschinen, elektrischer Starkstrom*. Verlag Ullstein, 1927.
- [5] Franz Reuleaux: *Die praktischen Beziehungen der Kinematik zu Geometrie und Mechanik*. Vieweg Verlag, Braunschweig, 1900.
- [6] A. B. Kempe: [How to Draw a Straight Line – A Lecture on Linkages](#). Macmillan & Co., London 1877.
- [7] Friedrich Natalis: [Genaue Geradföhrungen](#). Julius Springer, Berlin 1930.
- [8] James White: [A New Century of Inventions](#). Leech and Cheetham, Manchester 1822.
- [9] Eugene S. Ferguson: [Kinematics of Mechanisms from the Time of James Watt](#). Contribution from the Museum of History and Technology, Paper 27, Bulletin 228, Smithsonian Institution, Washington 1962.
- [10] Stefan Falk: *Scherenhub*. ft:pedia 1/2016.
- [11] Harald Steinhaus: *Flugzeugfahrwerke*. [ft:pedia 2/2012](#), S. 4-10.
- [12] Ralf Geerken: *Die Geradföhrung einer Viergelenkkette im Einsatz bei einer kleinen Laufmaschine*. [ft:pedia 4/2012](#), S. 4-10.
- [13] [Koppelgetriebe](#). Webseite der Gesellschaft zur Förderung der Digitalen Mechanismen- und Getriebebibliothek e.V. ([DMG-Lib](#))
- [14] Brian Bolt: *Was hat der Bagger mit Mathematik zu tun?* Klett Verlag, Stuttgart 1995.

Baukasten

Der Wohnzimmer-Dienstreisen-Urlaubs-Notfallkasten

Stefan Falk

Was macht man, wenn man abseits von seinem fischertechnik-Reich sitzt und den bestimmt bekannten „ich muss jetzt sofort diese Idee ausprobieren“-Anfall bekommt? Was tut man, wenn man im Urlaub rumhängt und man muss jetzt sofort ganz dringend etwas bauen? Vorsorge tut Not – hier kommt ein praxisgetesteter Vorschlag für einen ordentlich bestückten Kasten für „das kleine Modell zwischendurch“, der in keinem Koffer fehlen sollte.

Der erste Anlauf

Die Mutter aller urlaubstauglichen Kästen ist meines Erachtens der hobby-1 von 1970 [1, 2]:

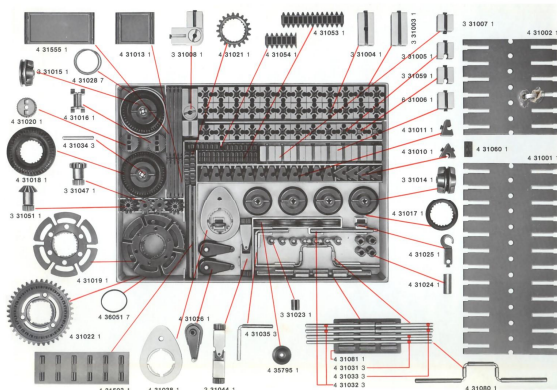


Abb. 1: hobby 1

Die Vorteile dieses Kastens für den Gebrauch unterwegs sind bestechend:

- Es handelt sich um eine Hartplastikwanne, die also einen Transport gut überstehen wird.
- Jedes Teil hat seinen Platz. Der Kasten kann daher mit einem Blick auf Vollständigkeit geprüft werden (wenn's wieder nach Hause geht nämlich).
- Er hatte einen brauchbaren Teilebestand, mit dem man viele mechanische Modelle herstellen konnte.

- Die Begleitbücher zur hobby-Serie geben eine perfekte Urlaubslektüre ab und bieten eine schier endlose Fülle von Modellanregungen.

Nach einigen Wohnzimmerabenden mit einem so bestückten Kasten ergaben sich aber auch Verbesserungswünsche:

- Einige heute geradezu unverzichtbare Teile fehlen, z. B. Bausteine 5 und 7,5.
- Dafür sind einige selten benötigte Teile enthalten, beispielsweise die beiden Kurvenscheiben und die Kurbelwelle.
- Der Kasten enthält noch relativ viel Luft – da könnten also durchaus noch Teile hinein.

Wie also bestückt man einen „hobby-1-Nachfolger“ heute? Dazu gibt es auch einen Forums-Thread unter [3].

Der Vorschlag

Von den Abmessungen her sind die hobby-Kästen identisch mit der fischertechnik Sortierbox 500. Als Deckel kann eine Bauplatte 500 dienen (das geht bei den hobby-Kästen übrigens auch). Also bestückte ich eine Box 500 mit Teilen und optimierte den Bestand über mehrere Wohnzimmerabende hinweg.

Hier ist das reichlich verwendete und bewährte Ergebnis:



Abb. 2: Der „Notfallkasten“, fertig zum Einpacken

Zum Transport sollte der Deckel vielleicht noch mit zwei starken Gummis, die um den Kasten gespannt werden, gesichert werden. Die tun auch bei manchen Antrieben gute Dienste.

Die Teileauswahl ist natürlich nur ein Vorschlag und sicherlich von meinen persönlichen Vorlieben geprägt. Jeder möge die Bestückung selbst nach Gusto verändern. Die Kriterien für die Stückliste waren folgende:

1. Jedes wesentliche mechanische Bauteil sollte in hinreichender Anzahl vertreten sein.
2. Die Stückzahlen sollten untereinander ausgewogen sein, sodass man nicht von einem Teil nutzlos viele eingepackt hat, während andere fehlen.
3. Der Kasten sollte trotz erweitertem Teilebestand immer noch möglichst rasch auf Vollständigkeit geprüft werden können. Deshalb gibt es nur die Stückzahlen 1 (Innenzahnrad, Differenzial, Schneckenmutter), 2, 3 (nur Klemm-Z10), 4 und ganzzahlige Vielfache von 4, nämlich 8, 12, 16 und 20.
4. Kein Teil darf beim Transport aus dem Kasten fallen, auch nicht, wenn man ihn auf dem Kopf hält und schüttelt.
5. Der Raum der Box 500 sollte gut ausgenutzt werden, um möglichst viele Teile unterzubringen.

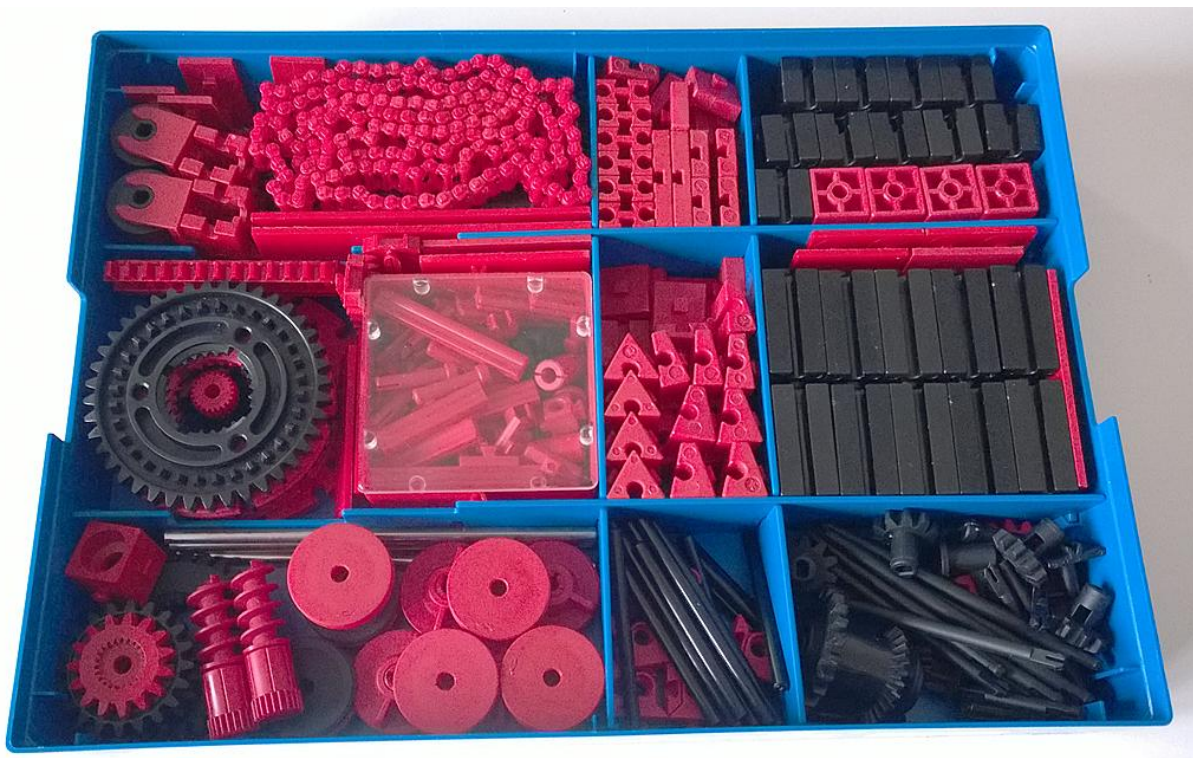


Abb. 3: Der eingeräumte Kasten

6. Im Zweifelsfall sollte leichten Teilen der Vorrang vor schweren gegeben werden, um Gewicht zu sparen.

Die Bestückung

Im Folgenden wird die Füllung der neun Fächer der Box 500 der Reihe nach dargestellt:

Rechts oben: Bausteine 15

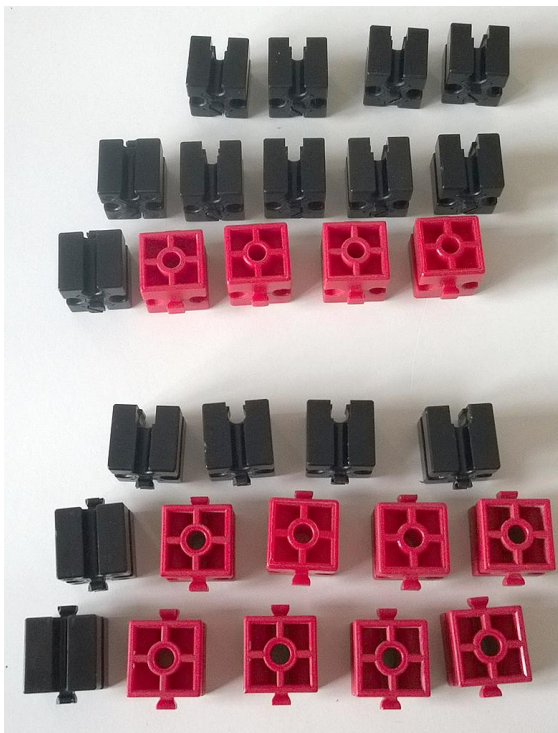


Abb. 4: Bausteine 15

Dieses Fach enthält je 12 Bausteine 15 mit einem Zapfen und Bausteine 15 mit Bohrung, sowie vier Bausteine 15 mit zwei Zapfen. In folgender Art kann das Fach in unterer und oberer Lage bestückt werden:



Abb. 5: Untere Lage

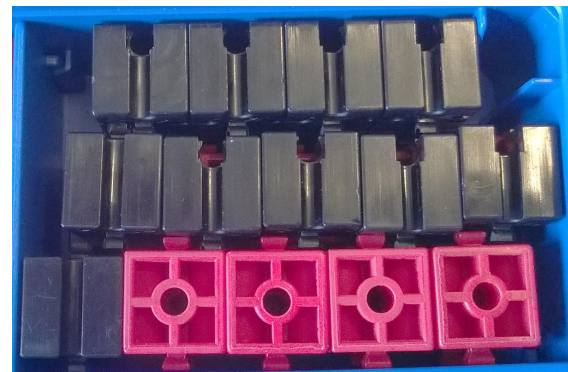


Abb. 6: Obere Lage

Rechts Mitte: Bausteine 30, Platten

Grundbausteine sind wichtig, vielseitiger einsetzbar als Statikträger 30 und trotz ihres Gewichts also reichlich vertreten:

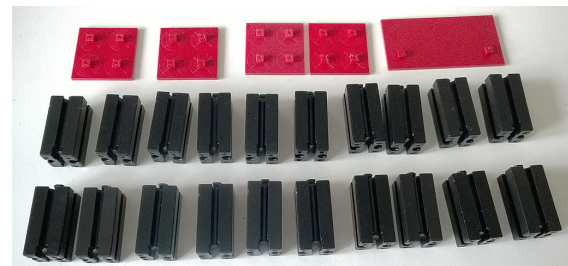


Abb. 7: 20 Stück Baustein 30, 4 Platten 30-30, 1 Platte 30-45

Rechts unten: Rastachsen und Rast-Getriebeteile

Von jeder Rastachsgröße bis 90 mm sind vier Stück vertreten, zusätzlich vier Rast-Z10, 2 Rast-Z20, 2 Rastkurbeln, ein Rast-Differenzial (in der Variante mit angebauem Z20) mit [zugehörigem Kegelzahnrad](#) sowie zwei [Rastaufnahmeachsen 130593](#).



Abb. 8: Rastachs-Teile

Mitte oben: Bausteine 5 und 7,5

Abb. 9: Bausteine mit 5 und 7,5 mm

Bausteine 7,5 (12 Stück) und Bausteine 5 (8 Stück) braucht man viele. Zusätzlich sind je vier [Bausteine 5 15:30](#) sowie [Bausteine 5 15:30 3N](#) vertreten. Dieses Fach muss sorgfältig eingeräumt werden, damit alles hineinpasst (siehe Abb. 3).



Abb. 10: Je 8 Winkelsteine

Mitte: Winkelsteine

Von den Winkelsteinen 7,5°, 15°, 30° und 60° sind je 8 Stück aufgenommen:

Mitte unten: Kunststoffachsen und Gelenke

Auch hier wurden von jeder Kunststoffachslänge 30, 45 und 60 vier Stück eingelegt. Acht Gelenksteine passen auch noch ins Fach:

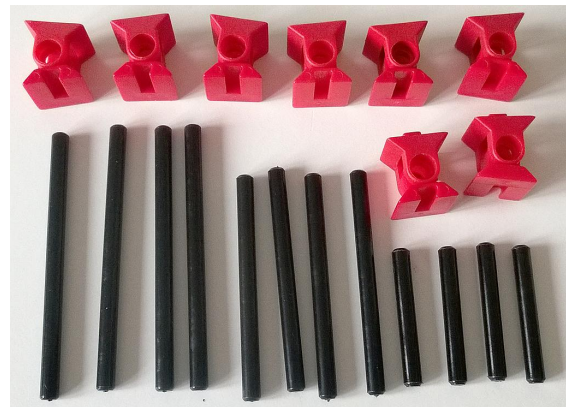


Abb. 11: K-Achsen, Gelenksteine

Links oben: Reifen, Seiltrommeln, Kette, Z10

Dieses Fach ist ein kleines Sammelsurium: Je vier [Reifen 45](#), [Rollenlager 15](#), Platten 30·15, je zwei Platten 30·90 und Seilwinden 30 komplett (bestehend aus [Trommel](#) und [Gestell](#)) füllen es. In die Reifen 45 passen noch zwei Zahnräder Z10, und obenauf eine Kette mit 160 Kettengliedern.

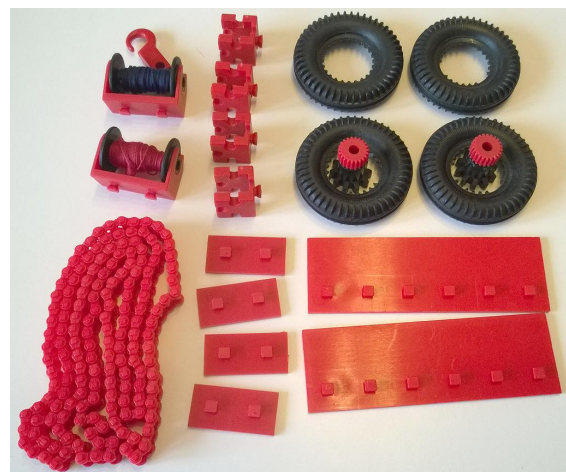


Abb. 12: Reifen, Seiltrommeln, Kette etc.

Die 160 Kettenglieder kann man leicht so abzählen:

1. Ein Kettenstück mit 10 Gliedern anfertigen.
2. Ein zweites durch Anlegen bilden.
3. Die beiden verbinden (20 Glieder lang).
4. Weitere 20 durch Anlegen bilden.
5. Verbinden (40 Glieder).
6. Usw. bis die Kette 160 Glieder lang ist.

Links Mitte: Große Räder, Zahnstangen, Platten, Kleinteile

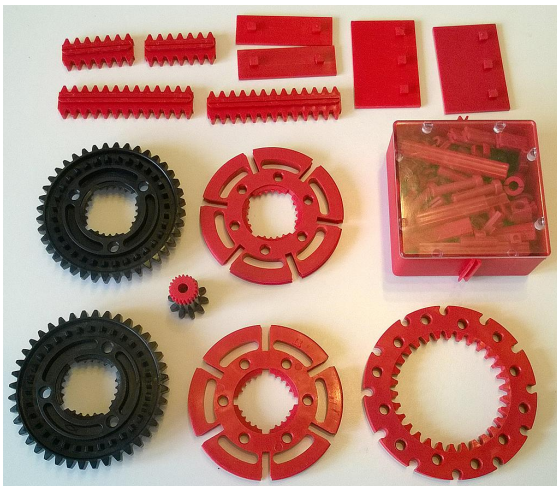


Abb. 13: Räder, Zahnstangen, Platten und eine Kasette

In diesem Fach stecken je zwei Drehscheiben, Zahnräder Z40, Zahnstangen 60, Zahnstangen 30 und Platten 15·30 sowie 30·45. Zuunterst liegt ein Innenzahnrad,

und in den zusammengesteckten Drehscheibe/Z40-Kombinationen wird ein drittes Zahnrad Z10 eingelegt. Damit hat man die drei Z10, die man für manche Innenzahnrad-Konstruktionen benötigt. Die Sortierkassette enthält alle Kleinteile:

- Zweifach:
 - [Klemmkupplung 20](#)
 - Kranhaken
 - Rast-Kardangeln
- Vierfach:
 - Verbinder 15, 30, 45
 - [Seilklemmstift 15](#)
 - [V-Achse 20](#)
 - [Abstandsring 3](#)
 - [S-Kupplung 15 2](#)
 - [Rastadapter 20](#)
 - [Lagerhülse 15](#)
 - [Rastachse 30 mit Kegelzahnrad](#)
 - [Rastkegelzahnrad](#)
 - [Seilrolle 12·5,5](#)
- Achtfach:
 - [Rastkupplung](#)
- Sechzehnfach:
 - Klemmbuchse 5
 - Federnocke

Links unten: Naben, Zahnräder, Metallachsen, Schnecke

Je vier Metallachsen 110, vollständige normale Naben und Flachnaben, je zwei vollständige [Freilaufnaben](#), Z15, Z20, Z30 sowie zwei [Schnecken](#) mit Klemmung und



Abb. 14: Inhalt der Sortierkassette

eine [Schneckenmutter](#) runden das Programm ab.



Abb. 15: Inhalt des letzten Fachs

Da keine hinreichend langen Achsen enthalten sind, ist es übrigens nicht sinnvoll, weitere [Schnecken 30](#) hinzuzufügen. Zur Vollständigkeitskontrolle empfiehlt es sich, alle Naben montiert im Fach zu haben. Zum schnelleren Bauen bevorzuge ich sie aber zerlegt – es kommt ja eh fast immer irgend-ein Rad drauf.

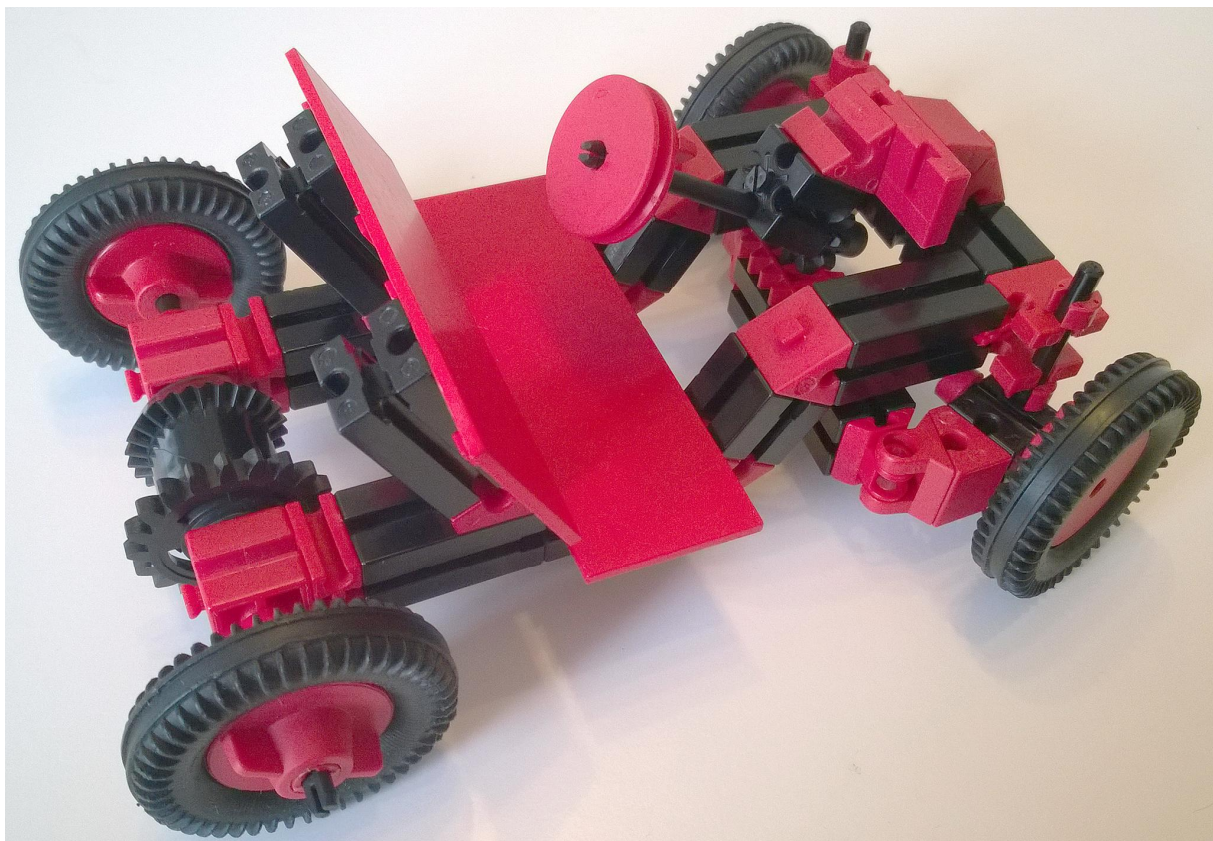
Bisherige Erfahrungen

Der Kasten macht süchtig J. Ich finde es sehr angenehm, ihn immer auf dem Wohnzimmertisch verfügbar zu haben und gemütlich etwas bauen zu können. Die Teileauswahl hat sich nun schon einige Baustunden lang sehr bewährt (sie erfuhr in dieser Zeit mehrmals Verfeinerungen).

Wer also, von akutem fischertechnik-Drang getrieben, schnell etwas zur Hand haben möchte, dem sei so ein Kasten wärmstens empfohlen. Modellvorschläge damit folgen.

Quellen

- [1] [hobby-1](#) (in der ft-Datenbank).
- [2] [hobby-1 Sortierplan](#) (in der ft-Datenbank).
- [3] [Wie würde ein „hobby-1“ heute bestückt sein?](#) im ft:c-Forum.



Modell

Scherenhub

Stefan Falk

Ausschließlich aus Teilen des in dieser ft:pedia-Ausgabe beschriebenen „Urlaubs-Kastens“ besteht diese Hebebühne mit Scherenhub-Mechanik.

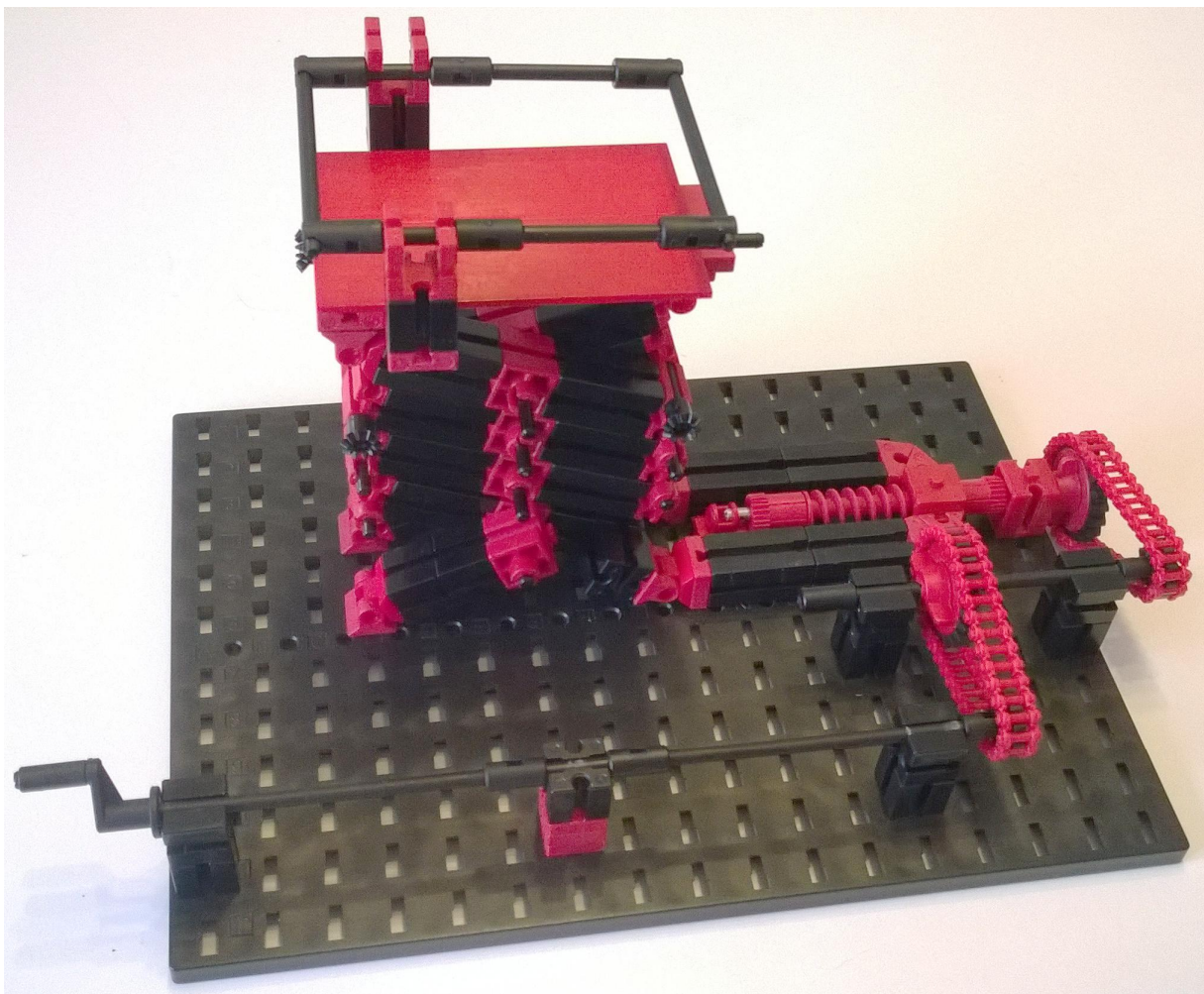


Abb. 1: Gesamtansicht des Modells

Das Modell ist auf einer Bauplatte 500 so aufgebaut: Per Kurbel wird nach zweimaliger Untersetzung eine Schnecke angetrieben. Die schiebt den auf der anderen Seite fest mit der Bauplatte verbundenen Scherenhub hinreichend kräftig zusammen, sodass der „Arbeitsbühne“ gar nichts

anderes übrigbleibt, als sich zu heben. Gerade wenn die Bühne sich ganz unten befindet, ist dazu eine erhebliche Schubkraft erforderlich, die aber über die beiden 2:1-Untersetzungen mit Z10, Z20 und Ketten leicht erreicht wird.

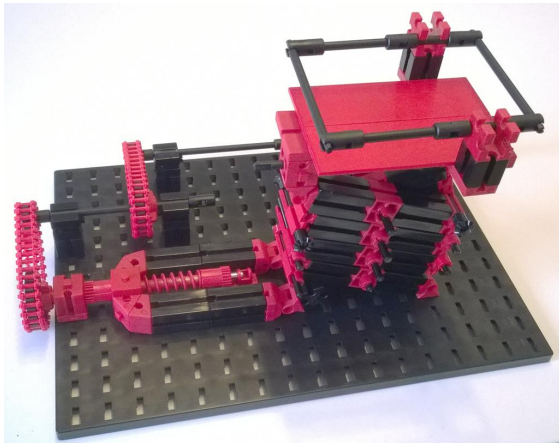


Abb. 2: Rückansicht



Abb. 3: Die Hebebühne ist ganz hochgefahren

Die beiden Schneckenteile sitzen auf einer Metallachse 100, die am drückenden Ende in einem BS15 mit Bohrung sitzt, der mit einem BS5 etwas erhöht auf der Platte sitzt.

Wichtig: Die Nut dieses BS5 muss quer zur Achse verlaufen, damit der BS15 nicht durch die erheblichen auftretenden Kräfte aus ihm herausgedrückt wird.

Auf der anderen Seite sitzt die eine Hälfte eines Gelenksteins per Federnocken und BS5 auf der Grundplatte. An der Schneckenmutter sind die beiden Schieberarme befestigt. Die bewegen sich also je nach Drehung der Schnecke.

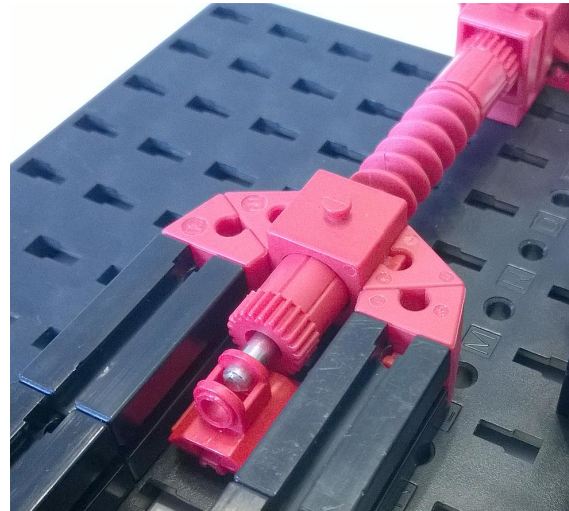


Abb. 4: Der Schneckenantrieb

Die Schubstangen kommen an den Gelenksteinen der einen Hälfte der Scherenhubmechanik an:

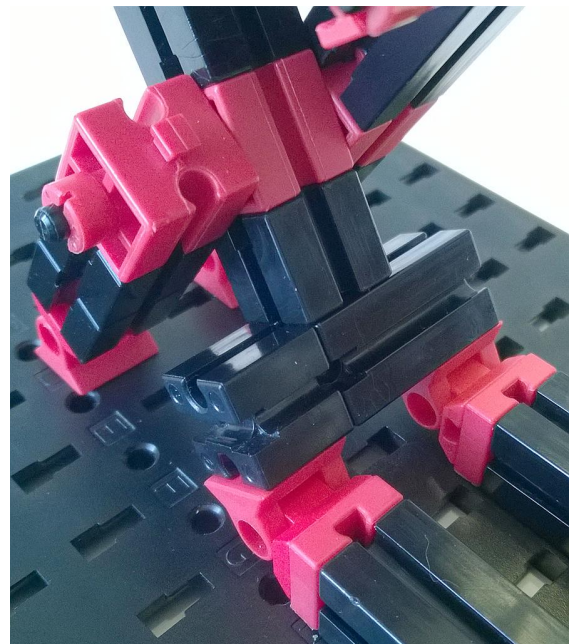


Abb. 5: Hubmechanik

Der Scherenhub besteht einfach aus jeder Menge gleicher Segmente aus BS7,5, Grundbaustein 30, BS15 mit Bohrung, Grundbaustein 30 und wiederum BS7,5. Solange Steine und Achsen reichen, kann weiter gebaut werden. Aus dem „Urlaubs-kasten“ werden hierfür neben Kunststoffachsen auch Rastachsen und dort anstatt Klemmbuchsen Rast-Winkelzahnäder zur Fixierung verwendet.

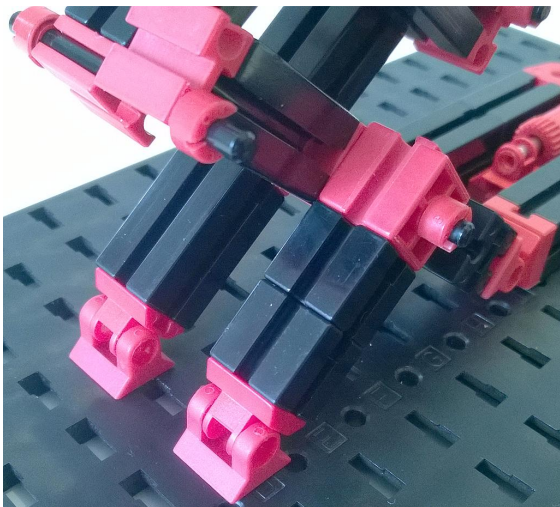


Abb. 6: Das feststehende Ende der Hubmechanik

Die äußeren Achsen an den Hubsegmenten sind so angebracht, dass in heruntergefahrenem Zustand alle Segmente plan aufeinanderliegen und die ganze Mechanik also so niedrig wie möglich baut.

Ganz oben schließlich kommt die anzuhebende Arbeitsbühne. Sie ist nur an einem Ende fest mit der Hubmechanik verbunden. Das andere gleitet in einer langen Rastachse, die in einer Kombination aus Rastachse mit Winkelzahnrad, BS15 mit Bohrung und Rastkupplung steckt.



Abb. 7: Aufbau der Hubsegmente

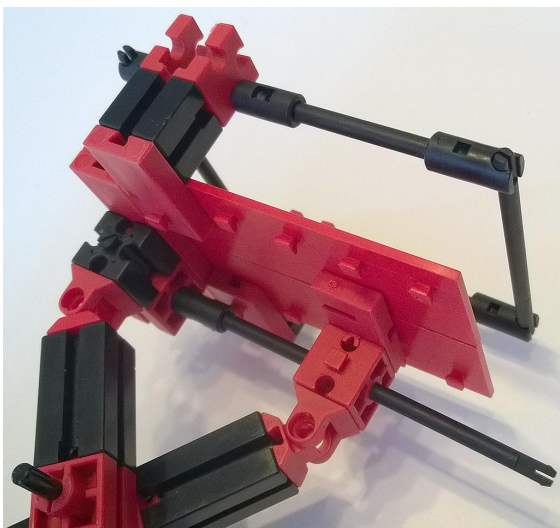


Abb. 8: Die Arbeitsbühne

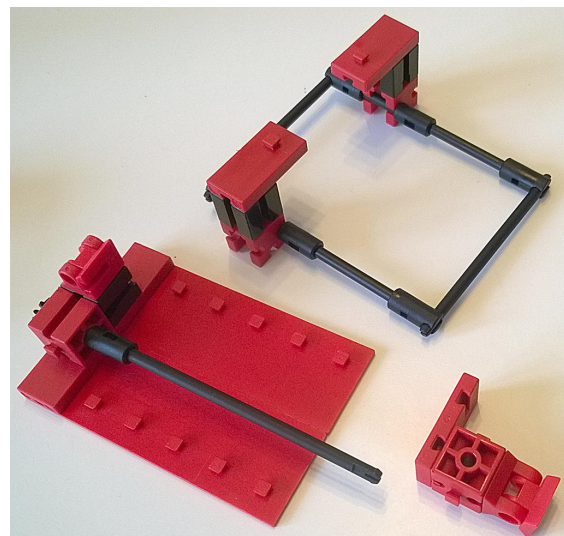


Abb. 9: Aufbau der Bühne

Modell

Schlauchquetschpumpe

Andreas Gail

In der Welt der Chemie- und Bioverfahrenstechnik sind Rohrleitungen das wichtigste Transportmittel. Die dazu erforderliche treibende Kraft wird häufig von Pumpen aufgebracht. Schlauchquetschpumpen sind hierbei eine gängige Pumpenart. Auch in Labor und Krankenhaus sind Schlauchquetschpumpen zu finden. Dieser Pumpentyp eignet sich besonders zum Aufbau mit fischertechnik.

Vorteile von Schlauchquetschpumpen

Eine Schlauchquetschpumpe kann vergleichsweise einfach aufgebaut werden. Der besondere Vorteil dieses Pumpentyps ist, dass die geförderten Flüssigkeiten keinen Kontakt zu den verbauten fischertechnik-Teilen haben (solange der Pumpschlauch nicht defekt ist). Die Pumpleistung einer Schlauchquetschpumpe, angegeben in z. B. Liter pro Stunde (L/h), ist eher gering. Stattdessen wird dieser Pumpentyp häufiger für Dosiervorgänge verwendet, bei denen es auf genaue Fördermengen ankommt.

Aufbau mit fischertechnik

Das Kernstück der Schlauchquetschpumpe ist in Abb. 1 gezeigt (frühere Modelle finden sich in [3] und [4]). Hier ist ein Schlauch (Silikon, Außendurchmesser 4 mm) zu sehen, der durch stets mindestens eines von drei kleinen Rädchenpaaren gegen das Schlauchgehäuse gequetscht wird. Die Rädchen sind drehbar zwischen zwei schwarzen Zahnrädern befestigt. Dieses Bauteil wird auch als Pumpenrotor bezeichnet und wird in Abb. 3 gesondert gezeigt. Die schwarzen Zahnräder wiederum werden motorisch angetrieben. Durch drehendes Vorwärtsrollen auf dem gequetschten Schlauch wird die davor

befindliche Flüssigkeitsmenge vorangetrieben. Insgesamt wird Wasser aus dem Wasserbehälter gemäß Abb. 2 über die Saugseite des Schlauchs angesaugt und über die Druckseite ausgestoßen.

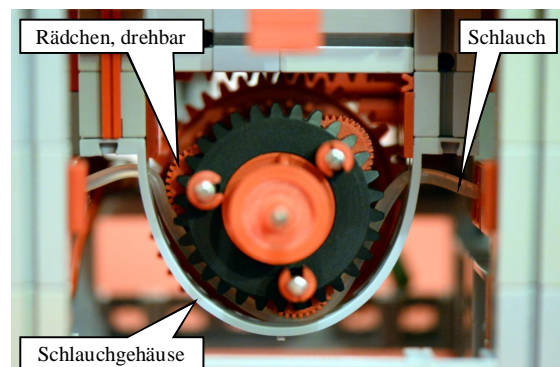


Abb. 1: Pumpenkopf mit Pumpenrotor, Kernstück einer Schlauchquetschpumpe; Lagerhalterung zur besseren Ansicht entfernt

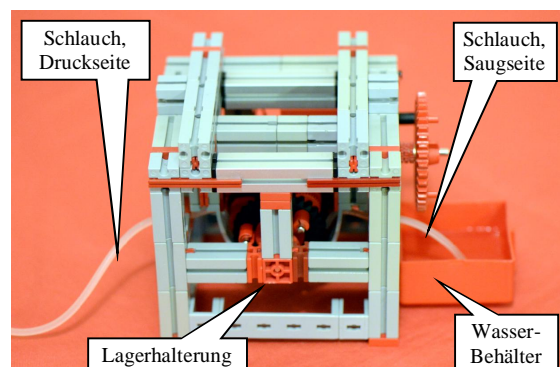


Abb. 2: Schlauchquetschpumpe Gesamtansicht

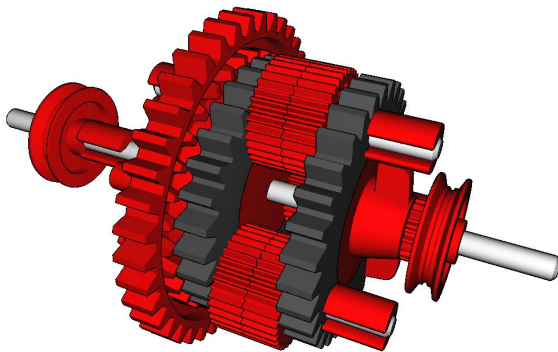


Abb. 3: Pumpenrotor

Der Antrieb

Der Antrieb erfolgt über einen Encodermotor, wie in Abb. 4 und Abb. 5 gezeigt. Das hat den Vorteil, dass eine Volumendosierung im Bedarfsfall möglich wird. Um das zu realisieren, muss die Pumpe eine Zeitlang laufen. Dabei werden die Encodermotorschritte gezählt und gleichzeitig das dabei geförderte Wasservolumen gemessen. Je größer das geförderte Wasservolumen, desto genauer ist die spätere Dosierung.

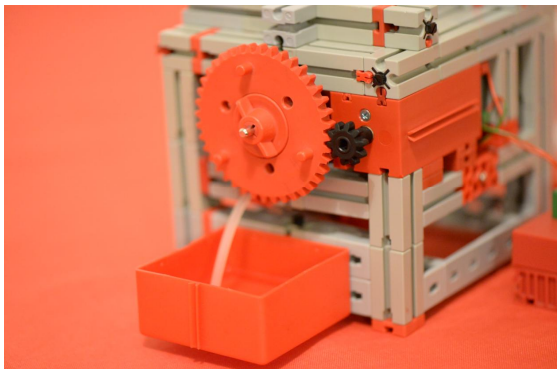


Abb. 4: Antrieb der Schlauchquetschpumpe

Da die Quetschung des Silikonschlauchs an mindestens einer Stelle für eine fischertechnik-Konstruktion schon eine merkliche Belastung darstellt, ist das Pumpengehäuse stark versteift aufgebaut. Zudem wird mit einer starken Untersetzung gearbeitet, um den Motor nicht zu stark zu belasten. Das bedeutet entsprechend eine geringere Pumpleistung. Für eine höhere Pumpleistung muss ein anderer Motorentyp eingesetzt werden, z. B. ein oder mehrere Power-Motoren. Weiterhin wäre es auch

denkbar, den Pumpenrotor so zu verbreitern, dass durch mehrere Schläuche gleichzeitig parallel gepumpt wird.

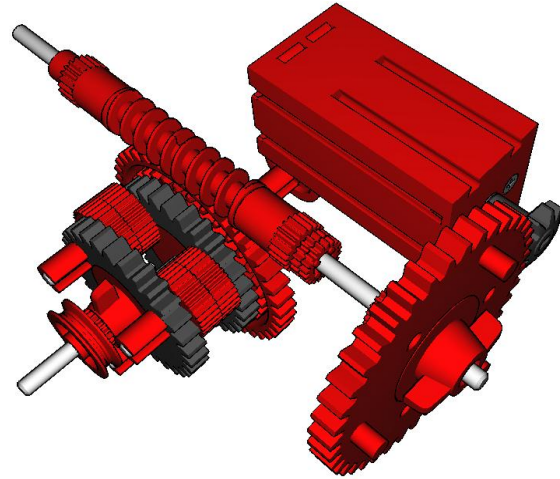


Abb. 5: Antrieb des Pumpenrotors

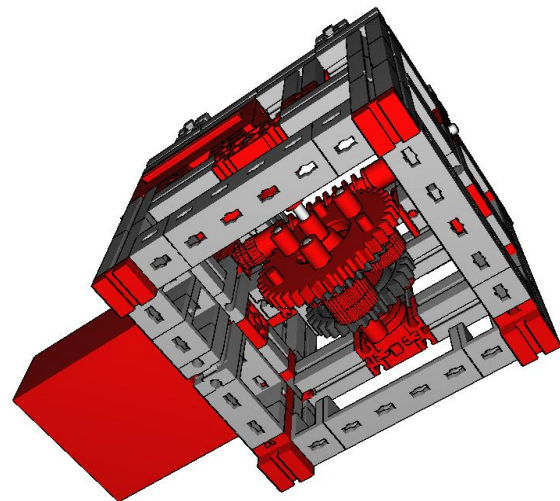


Abb. 6: Ansicht von unten, Darstellung ohne Schlauchgehäuse

Referenzen

- [1] Wikipedia: [Schlauchpumpe](#)
- [2] Gail, Andreas: [Schlauchquetschpumpe](#) (youtube-Video)
- [3] fischertechnik: *Neues vom fischertechnik-Club. Clubheft 1973-2*, S. 9-13. Fischer-Werke, 1973.
- [4] fischertechnik: *Niveauregelung bei bewegten Flüssigkeiten. Begleitbuch hobby 4 Band 5*, S. 42-45. Fischer-Werke, 1975.

Messtechnik

Beschleunigung auf der schiefen Ebene

David Holtz

Seit 2011 flitzen die Stahlkugeln auf den grünen Flexschienen durch Kugelbahnparcours. Der eine oder andere hat sich dabei sicherlich schon einmal gefragt, mit welcher Geschwindigkeit die Kugeln unterwegs sind oder wie stark sie beschleunigen. Ich habe einen Versuchsaufbau aus Fischertechnik entworfen, mit dem man die Beschleunigung auf der schiefen Ebene ermitteln kann.

Der Versuchsaufbau

An den Versuchsaufbau werden folgende Anforderungen gestellt:

- Flexschienen auf einer schiefen Ebene, damit die Beschleunigung konstant ist

- Frei einstellbarer Neigungswinkel der schiefen Ebene
- Starten der Kugel auf Knopfdruck

Der Versuchsaufbau (Abb. 1) setzt diese Anforderungen um. Die schiefe Ebene

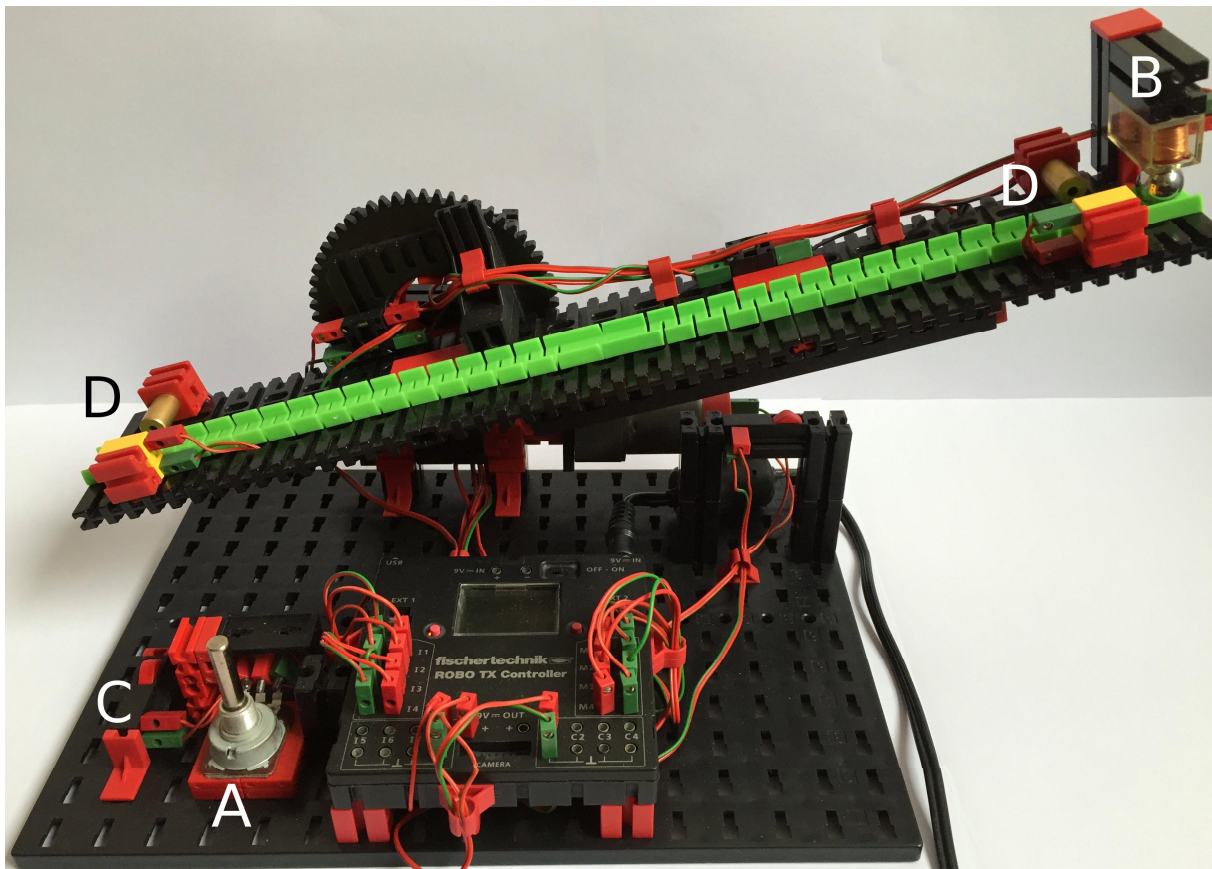


Abb. 1: Versuchsaufbau

besteht aus drei Grundplatten 120x60, auf denen zwei Flexschiene befestigt sind.

Dieser Aufbau ist an einem Drehkranz montiert, sodass man den Neigungswinkel α zwischen 1° und 45° einstellen kann. Dies funktioniert, indem man den Neigungswinkel am Potentiometer (A) vorgibt und ein Motor den Drehkranz über einen Soll-Ist-Abgleich in die gewünschte Position bewegt. Die Genauigkeit beträgt hierbei 1° . Der Elektromagnet (B) hält die Kugel, bis der Taster (C) betätigt wird. Beim Hinabrollen auf der schiefen Ebene unterbricht die Kugel die Lichtschranken (D), die jeweils aus einem Fototransistor und einer roten Punktlaserdiode bestehen.

Ein bisschen Physik...

Um zu verstehen, von welchen physikalischen Gesetzmäßigkeiten der Versuchsaufbau Gebrauch macht, sollte man einige Größen und deren Zusammenhänge kennen.

Geschwindigkeit

Spricht man von der Geschwindigkeit, muss man zunächst die Durchschnittsgeschwindigkeit von der Momentangeschwindigkeit unterscheiden.

Die Durchschnittsgeschwindigkeit \bar{v} (oder mathematisch ausgedrückt: mittlere Änderungsrate) beschreibt, welchen Weg s ein Körper im Zeitraum t zurückgelegt hat:

$$\bar{v} = \frac{\Delta s}{\Delta t}$$

Die Momentangeschwindigkeit (oder mathematisch ausgedrückt: momentane Änderungsrate) erhält man, wenn man in der obenstehenden Gleichung die Zeit t null annähert. Der Quotient aus dem Weg und der nach null angenäherten Zeit beschreibt die Geschwindigkeit v zum Zeitpunkt t .

$$v = \lim_{\Delta t \rightarrow 0} \frac{\Delta s}{\Delta t}$$

Während im Alltag die Einheit $1 \frac{km}{h}$ für die Geschwindigkeit am gebräuchlichsten ist, wird die Geschwindigkeit in der Physik meistens in der Einheit $1 \frac{m}{s}$ angegeben.

Beschleunigung

Die Beschleunigung a beschreibt die momentane Änderungsrate der Geschwindigkeit eines Körpers pro Zeit. Somit ist die Einheit der Beschleunigung

$$[a] = 1 \frac{\frac{m}{s}}{s} = 1 \frac{m}{s^2}$$

Bei konstanter Beschleunigung verändert sich die Momentangeschwindigkeit v proportional zur Zeit t mit dem Proportionalitätsfaktor a :

$$v(t) = a \cdot t + v_0$$

Schiefe Ebene

Unter einer schiefen Ebene versteht man eine ebene Fläche, die um den Neigungswinkel α geneigt ist [1].

Auf Körper, die sich auf der schiefen Ebene befinden, wirkt die sogenannte Hangabtriebskraft F_H . Diese Kraft ist durch die Gewichtskraft F_G bedingt.

Energie

Wenn Kräfte wirken, ändert sich das Energieniveau eines Körpers. Im Fall der schiefen Ebene wird die potenzielle Energie der Kugel in kinetische Energie umgewandelt, wodurch die Kugel die schiefe Ebene hinabrollt [2]. Allerdings entfällt ein Teil der potentiellen Energie auch auf das Rotieren der Kugel. Bei der sogenannten Rotationsenergie handelt es sich ebenfalls um kinetische Energie [3].

Die Rotationsenergie lässt sich aus dem Trägheitsmoment der Kugel und ihrer Winkelgeschwindigkeit berechnen:

$$E_{Rot} = 0,5 \cdot J \cdot \omega^2$$

Das Trägheitsmoment¹ der fischertechnik-Kugel ist so gering, dass die Rotationsenergie bei den nachfolgenden Berechnungen vernachlässigt werden kann. Wäre die Kugel größer und schwerer oder gar ein hohler Zylinder, müsste man die Rotationsenergie berücksichtigen.

Beschleunigung auf der schiefen Ebene

Die Berechnung der Beschleunigung beruht auf dem Zusammenhang zwischen Gewichtskraft und Hangabtriebskraft. Man kann die Hangabtriebskraft über die Formel

$$F_H = \sin \alpha \cdot F_G$$

berechnen [4]. Da in beiden Kräften die Masse des Körpers beziehungsweise die der Kugel enthalten ist ($F = m \cdot a$), kann man sie aus der Gleichung herauskürzen:

$$a = \sin \alpha \cdot g$$

g ist die Erdbeschleunigung, die in Europa etwa $9,81 \frac{m}{s^2}$ beträgt. Mit dieser Formel kann man die Beschleunigung eines Körpers auf der schiefen Ebene berechnen (vgl. Tab. 1).

Neigungswinkel α	Beschleunigung a in $\frac{m}{s^2}$
1°	0,17
10°	1,70
30°	4,91
45°	6,94
60°	8,50

Tab. 1: Beschleunigung auf der schiefen Ebene

Die Beschleunigung eines Körpers auf der schiefen Ebene ist konstant und unabhängig von dessen Masse.

Jetzt, wo man die Beschleunigung berechnen kann, scheint ein Versuchsaufbau doch überflüssig zu sein. Allerdings wird jeder

Körper durch Reibung gehemmt, weshalb die in Tab. 1 berechneten Beschleunigungen lediglich theoretisch maximal erreichbare Werte sind.

Funktionsprinzip

Die im Absatz *Beschleunigung* aufgestellte Formel $v(t) = a \cdot t + v_0$ würde genügen, um die Beschleunigung zu berechnen, wenn man sie nach a umstellt. In der Praxis gestaltet sich dieser Ansatz allerdings als schwer umsetzbar, da man die relativ hohe Momentangeschwindigkeit $v(t)$ ziemlich genau messen müsste, um ein sinnvolles Ergebnis zu erhalten.

Optimierung

Die Bewegungsgleichung

$$s(t) = \frac{1}{2} \cdot a \cdot t^2 + v_0 \cdot t$$

beschreibt den zurückgelegten Weg s während der Zeit t mit einer konstanten Beschleunigung a unter Berücksichtigung einer Anfangsgeschwindigkeit v_0 .

Um die Beschleunigung zu berechnen stellt man die oben genannte Formel nach a um:

$$a = \frac{-2 \cdot v_0}{t} + \frac{2 \cdot s}{t^2}$$

Folgende Vorteile ergeben sich, wenn man diese Formel anwendet:

- Die Zeit t ist relativ genau mit einfachen Mitteln bestimmbar.
- Der zurückgelegte Weg s ist eine konstante Größe, die man lediglich einmal ausmessen muss.
- Die Anfangsgeschwindigkeit v_0 ist vergleichsweise gering und somit einfacher messbar.

Die folgende Abbildung 2 veranschaulicht die zu messenden Größen.

¹ $J_{ft-Kugel} = \frac{2}{5} \cdot m \cdot r^2 \approx 1,25 \cdot 10^{-7} \text{ kg m}^2$

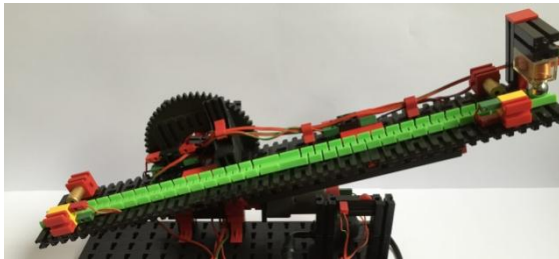


Abb. 2: Gemessene Größen

1. Der Weg zwischen den Lichtschranken (konstant)
2. Die Zeit, die die Kugel von Lichtschranke 1 zu Lichtschranke 2 benötigt
3. Die Geschwindigkeit der Kugel an der ersten Lichtschranke

Man könnte davon ausgehen, dass die Geschwindigkeit der Kugel beim Losrollen 0 m/s beträgt.

Erfahrungsgemäß wird die Kugel beim Losrollen durch Reibung gehemmt. Um daraus resultierende Messfehler zu vermeiden, beginnt die eigentliche Messung erst kurz nach dem Start der Kugel. Nachdem der Elektromagnet die Kugel losgelassen hat, rollt diese ein Stück, bevor sie die erste Lichtschranke unterbricht. Dort wird die Momentangeschwindigkeit gemessen. Anschließend wird die Zeit gemessen, die die Kugel benötigt, um den Weg s zurückzulegen.

Bei diesem Ansatz ergibt sich der Nachteil, dass man die Momentangeschwindigkeit kurz nach dem Losrollen ermitteln muss.

Momentangeschwindigkeit bestimmen

Der Messaufbau misst dazu die Verdunklungszeit an der ersten Lichtschranke (Abb. 3). Mit der im Absatz *Geschwindigkeit* aufgestellten Gleichung kann man die Momentangeschwindigkeit berechnen:

$$\bar{v} = \frac{\Delta s}{\Delta t}$$

Dabei ist der zurückgelegte Weg Δs ein konstanter Wert. Er entspricht dem Durchmesser der Kugel² (11 mm) unter der Voraussetzung, dass der Lichtstrahl an der breitesten Stelle auf die Kugel trifft. Die Verdunklungszeit wird für Δt in die Gleichung eingesetzt.

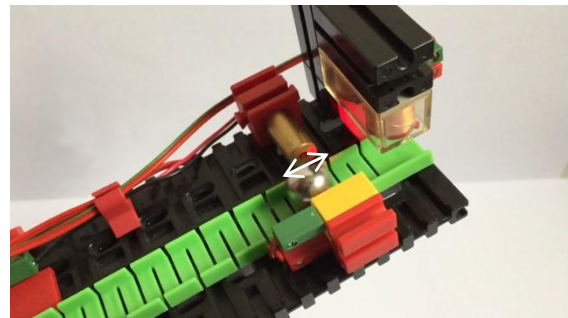


Abb. 3: Verdunklungszeit

Mathematisch korrekt ausgedrückt berechnet man mit dieser Gleichung die Durchschnittsgeschwindigkeit der Kugel in einem sehr kleinen Intervall. In der Praxis ist das Intervall aufgrund der geringen Verdunklungszeit ausreichen klein, um von einer Momentangeschwindigkeit sprechen zu können.

Mit den ermittelten Werten kann die Beschleunigung der Kugel nach der im Absatz *Optimierung* beschriebenen Gleichung berechnet werden. Man erhält ein ziemlich genaues Ergebnis, da alle zu bestimmenden Größen entweder konstant sind oder relativ präzise messbar.

Zudem kann man ausgehend von der Beschleunigung, der Zeit und der Anfangsgeschwindigkeit die Momentangeschwindigkeit der Kugel an der zweiten Lichtschranke berechnen:

$$v(t) = a \cdot t + v_0$$

² Die Stahlkugeln aus den Dynamic Baukästen von Fischertechnik haben laut Einzelteilpreisliste einen Durchmesser von 12,5 mm.

Beim Durchqueren der Lichtschranke wird der Lichtstrahl über eine Länge von 11 mm der Kugel unterbrochen.

Die Berechnungen führt der Mikrocontroller durch. Anschließend gibt er die Beschleunigung und die Endgeschwindigkeit am Display aus.

Implementierung

Zunächst habe ich die Berechnungen nur in RoboPro implementiert. Allerdings fällt bei den berechneten Werten auf, dass diese auffällig stark von den theoretischen Werten abweichen. Aus diesem Grund habe ich zusätzlich zu dem RoboPro-Programm ein Arduino-Programm geschrieben.

Das RoboPro-Programm

Der Vorteil bei RoboPro und dem TX(T) Controller liegt in der einfachen Bedienbarkeit. Man kann alle benötigten elektronischen Komponenten einfach an den TX(T) Controller anschließen und ein Programm ist schnell geschrieben. Der entscheidende Nachteil liegt in der Genauigkeit der Zeitbestimmung, die dafür verantwortlich ist, dass die gemessenen Werte nicht den realistischen Bedingungen entsprechen.

Das Arduino-Programm

Der Arduino UNO bietet eine genauere Möglichkeit, die Zeit zu messen. Dies bewirkt, dass die daraus errechnete Beschleunigung größer und in diesem Fall näher am theoretischen Wert liegt (vgl. Tab. 2). Für den Arduino UNO wird eine C-basierte Programmieroberfläche bereitgestellt, mit der man komfortabel programmieren kann. Nutzt man kein separates Display, kann man sich die errechneten Werte am PC anzeigen lassen.

Allerdings kann man die Sensoren nicht so einfach an den Arduino anschließen wie an den TX(T) Controller. Man benötigt für alle digitalen Sensoren einen Spannungsteiler und für Motoren einen Motortreiberchip beziehungsweise ein *Motor Shield*.

Neigungswinkel (°)	TX (m/s ²)	Arduino (m/s ²)	Theorie (m/s ²)
5	0,33	0,69	0,86
10	1,01	1,42	1,70
15	1,68	1,98	2,54
20	2,24	2,87	3,36
25	3,01	3,66	4,15
30	4,13	4,56	4,91

Tabelle 2: Vergleich gemessene / errechnete Beschleunigung auf Flexschienen

Letztlich ist die Wahl der Plattform eine Frage der Genauigkeit und des Komforts.

Reibung auf Flexschienen

Die gemessene Beschleunigung kann man nutzen, um die Reibung der Kugel beim Rollen zu berechnen. Als Vergleichspunkt zur Flexschiene dient eine Schienenkonstruktion bestehend aus zwei parallelen Edelstahlstangen.

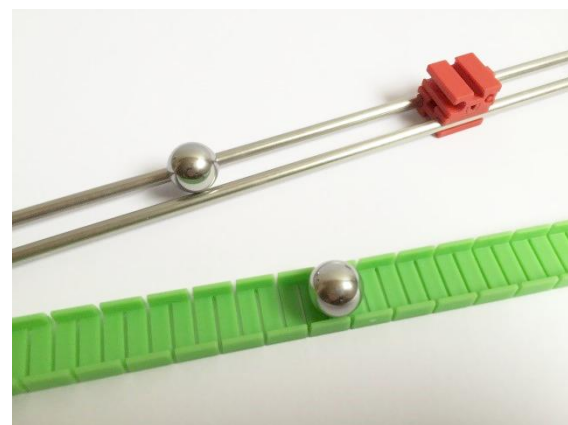


Abbildung 4: Flexschiene und Edelstahlschiene

Reibungskraft

Abbildung 5 veranschaulicht die Wirkung der Hangabtriebskraft und die der Reibungskraft. Die Reibungskraft hemmt die Kugel in ihrer Bewegung, da sie der Hangabtriebskraft entgegengerichtet ist.

Die Reibungskraft F_R ist das Produkt aus der Normalkraft und des Rollwiderstandskoeffizienten c_R . Beschreibt man sie in Abhängigkeit der Gewichtskraft und des

Neigungswinkels der schiefen Ebene, erhält man folgende Formel [3]:

$$F_R = c_R \cdot F_G \cdot \cos \alpha$$

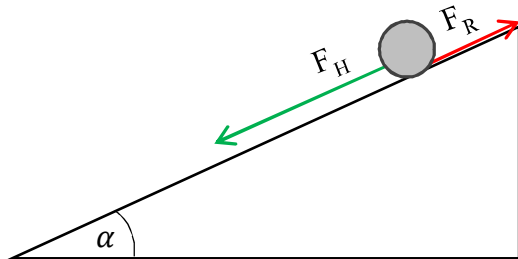


Abb. 5: Hangabtriebskraft und Reibungskraft

Die beschleunigende Kraft (beziehungsweise die gemessene Beschleunigung) ist gleich der Differenz zwischen der wirkenden Hangabtriebskraft (ohne Reibung) und der Reibungskraft, die der Hangabtriebskraft entgegengerichtet ist:

$$F = F_{H(max.)} - F_R$$

Da die Masse der Kugel in allen Kräften enthalten ($F = m \cdot a$) ist, kürzt sie sich raus:

$$a = g \cdot \sin \alpha - c_R \cdot g \cdot \cos \alpha$$

Für die Erdbeschleunigung wird der Wert $g = 9,81 \frac{m}{s^2}$ verwendet. Die Beschleunigung a ist bekannt, da man sie mit dem Versuchsaufbau messen kann. Mit diesem Wissen kann man den Rollwiderstandskoeffizienten berechnen, indem man die Formel nach c_R umstellt:

$$c_R = \frac{\sin \alpha}{\cos \alpha} - \frac{a}{g \cdot \cos \alpha}$$

Der Rollwiderstandskoeffizient hat keine Einheit. Da er indirekt über die Rollreibung bestimmt wurde, enthält er ungewollt auch eine Luftwiderstandskomponente und unvermeidbar Messtoleranzen. Da die Rotationsenergie in den Berechnungen vernachlässigt wurde, spiegelt der errechnete Wert nicht die reale Größenordnung wieder, sondern dient vielmehr dazu, die Reibung

auf unterschiedlichen Schienen zu vergleichen.

Verwendung

Tabelle 3 zeigt Rollwiderstandskoeffizienten für die Schienenkonstruktion und die Flexschienen. Im Idealfall ist der Rollwiderstandskoeffizient unabhängig vom Neigungswinkel. Allerdings ist dies aufgrund von Messtoleranzen nicht immer der Fall.

Neigungswinkel (°)	Edelstahlschiene	Flexschiene
15	0,037	0,059
20	0,032	0,053
25	0,030	0,055
30	0,032	0,041

Tabelle 3: Errechnete Rollwiderstandskoeffizienten

Die Tabelle zeigt, dass der Rollwiderstandskoeffizient und damit die Reibung auf den Flexschienen deutlich höher ist als auf den Edelstahlstangen. Dies liegt an der Mäanderstruktur der Flexschiene, die aufgrund ihrer kleinen Rillen für die höhere Reibung im Vergleich zur glatten Edelstahloberfläche verantwortlich ist. Der Rollwiderstandskoeffizient eignet sich somit gut zum Vergleichen der Reibung einer Kugel auf verschiedenen Schienen.

Ausblick

In der Praxis sind viele Streckenabschnitte leider keine schiefen Ebenen. In diesen Fällen wäre die Beschleunigung der Kugel nicht mehr konstant, sodass man die Bewegungsgleichungen nicht nutzen kann.

Nichtsdestotrotz gibt der Versuchsaufbau einen Einblick in das große Themengebiet der Mechanik. Man kann damit die Beschleunigung bestimmen und relativ genau die Endgeschwindigkeit der Kugel berechnen, ohne sehr kleine Zeiten messen zu müssen. Steigt man tiefer in die Materie ein, kann man sogar die Reibung auf unterschiedlichen Schienen vergleichen.

Quellen

- [1] Wikipedia: [Schiefe Ebene](#).
- [2] Leifi Physik: [Energieeinheiten](#).
- [3] Leifi Physik: [Drehbewegungen](#).
- [4] Wikipedia: [Hangabtriebskraft](#).
- [5] Wikipedia: [Rollwiderstand](#).

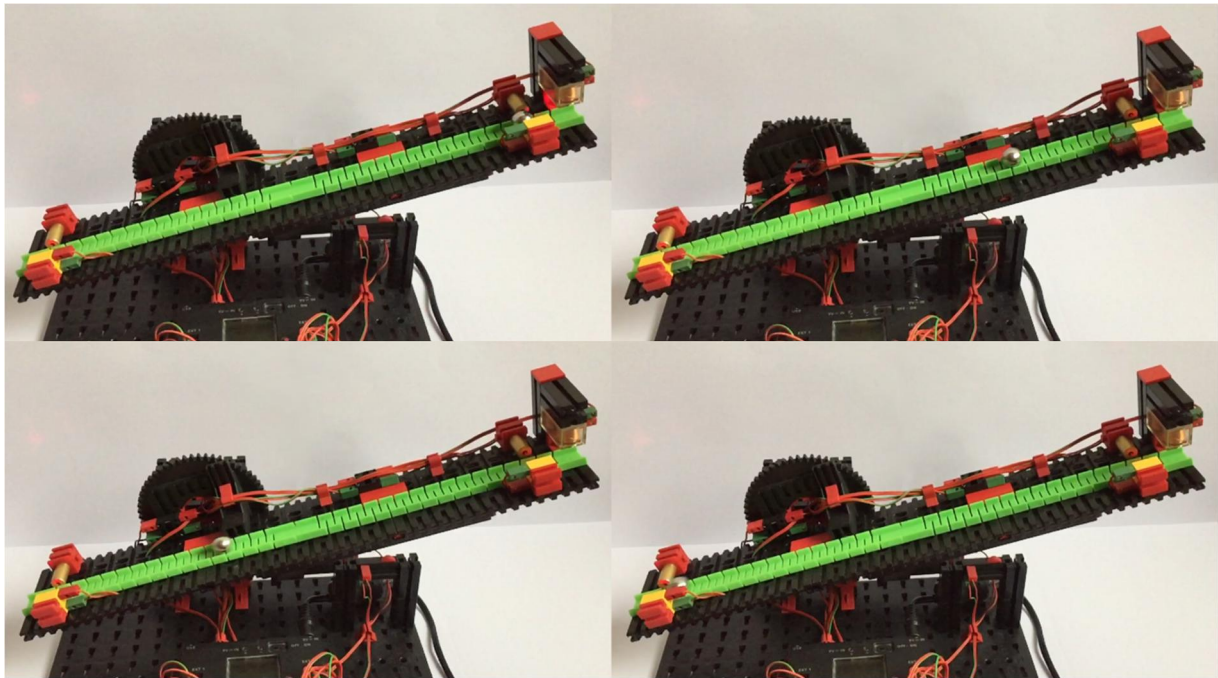


Abb. 6: Slow-Motion-Aufnahmen der Kugel beim Herabrollen

Elektronik

Radar mit Objektfokussierung

Werner Hasselberg

Dieser Beitrag stellt ein „Radar-Gerät“ zum Nachbauen vor. Es kommt gänzlich ohne Computer aus, verfügt aber trotzdem über eine anspruchsvolle Funktion: Ein erkanntes leuchtendes Objekt wird nicht nur angezeigt, sondern durch elektronisch gesteuertes Einpendeln der Radarantenne genau markiert.

Vorwort

Als es vor gut 30 Jahren erstmals möglich war, fischertechnik-Modelle über ein Interface und selbst geschriebene Programme zu steuern, war das eine echte Revolution. Plötzlich bewegten sich Maschinen wie von Geisterhand gesteuert, was vorher nur über diverse Schalter und Hebel möglich war. Heute nun durchdringen Computer so ziemlich alle Lebensbereiche und so gut wie alles wird von ihnen gesteuert. Ein Leben ohne sie – undenkbar. Gute Gründe also, möglichst früh damit anzufangen. Eine langjährige Studie in vielen Kindergärten der USA, alle vollständig mit Computern ausgestattet, ergab nun, dass damit weder das logische Denken noch die Kreativität noch das technische Verständnis, geschweige denn die Motorik der Kinder gesteigert werden konnten. Das Gegenteil war eher der Fall. Die Kosten für die Ausstattung gingen in die Millionen und waren, bis auf die satten Gewinne der Hersteller, somit völlig nutzlos.

Bereits Sechsjährige beherrschen es mittlerweile, Roboter zu programmieren. Nicht weil die viel klüger geworden sind, sondern weil der Computer samt Bildsprache die ganze kreative Arbeit übernommen hat. Zeit also, über Alternativen nachzudenken, die es wieder möglich machen, ganz ohne Computer und Bildchen-Wischerei zu basteln. Freilich sind komplexe

Steuerungen ohne Computer nicht denkbar. Wem aber die reine Freude am Basteln und ein gewisses Verständnis, wie Dinge funktionieren, am Herzen liegt, wird mit ft-Elektromechanik und -Elektronik viel Spaß haben. Denn hier gibt es viel zu tüfteln, wenn es gilt, Lösungen zu finden, wie z. B. das Umpolen eines Motors, ohne dass ein e-Tec-Modul oder der Computer das für uns erledigt. Es gibt so viele Dinge, die ganz ohne lösbar sind und dabei noch viele Grundlagen vermitteln. Deshalb möchte ich anhand eines Modells zeigen, dass es sehr interessant sein kann, elektronisch und mechanisch zu steuern. fischertechnik ist damit bis heute ohne jede Konkurrenz, und könnten sich wieder mehr Jugendliche dafür begeistern, stünde nichts dagegen, die ganze Sparte wieder in altem Glanz erstrahlen zu lassen.

Was soll unser Radar leisten?

Ein Radar dreht sich normalerweise im Kreis und sucht den Himmel ab. Bei Kontakt erscheint auf einer Konsole ein leuchtender Punkt in der entsprechenden Himmelsrichtung. Damit die Elektronik aber etwas mehr zu tun bekommt, bauen wir diese Grundfunktion noch ein wenig aus. Das Radar soll zusätzlich bei Kontakt einige Sekunden in einen Pendelmodus wechseln und so die Quelle länger fixieren. Dadurch wird der Leuchtpunkt auf der Konsole mehrfach hintereinander aufblinken. Nach

der erfolgreichen Entdeckung der Quelle dreht sich das Radar normal weiter – bis es erneut eine Lichtquelle ausmacht. Dann beginnt das Spiel von vorne.

Fangen wir gleich mit dem Bauen an.

Elektromechanischer Zeitschalter

Ihn brauchen wir, damit das Radar nur eine bestimmte Zeit in den Pendelmodus umschaltet und sich danach wieder ganz normal weiterdrehen kann.

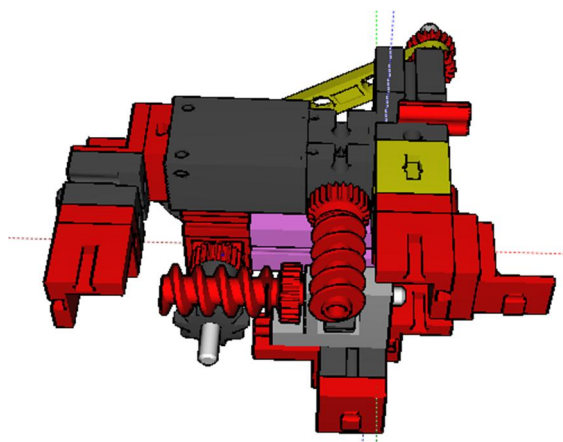


Abb. 1: Zeitschalter von unten

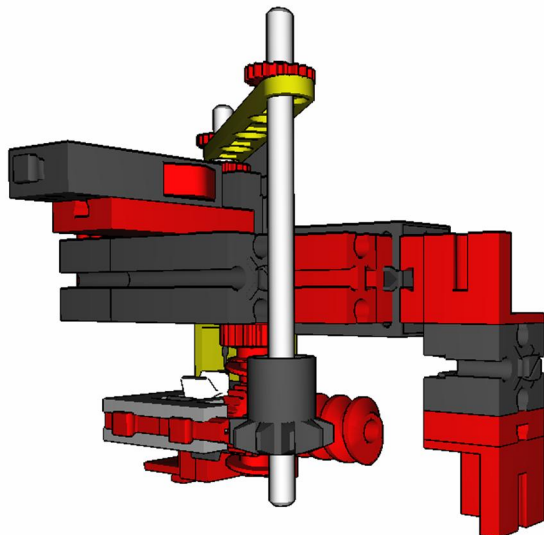


Abb. 2: Gegenüberliegende Seite

Der Taster wird auf eine Bauplatte 15·30·5 mit Nut und Zapfen geschoben und die Platte mit der Nut auf den darunterliegenden Baustein 30. So kann sie genau justiert werden.

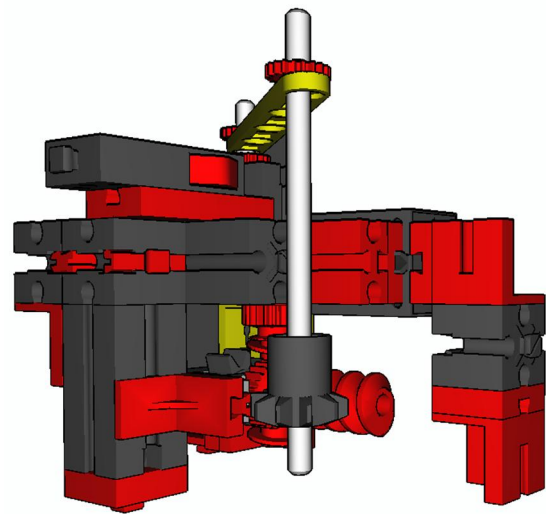


Abb. 3: Schnecke montiert

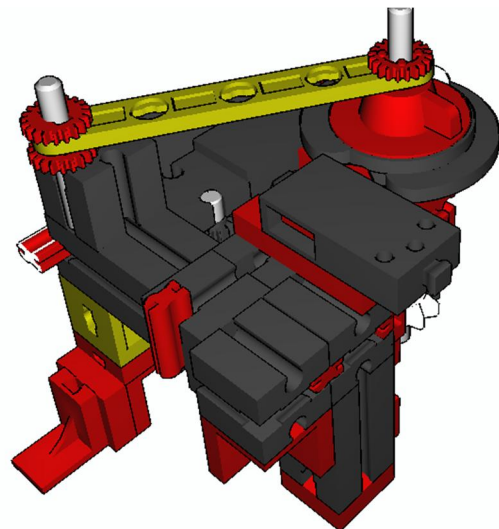


Abb. 4: Montierte Schaltscheiben

Der Aufbau des Radars

Als kleiner Anreiz vorab schonmal ein Foto des fertigen Modells samt Techniker im Einsatz. Die Radarantenne, soviel darf bereits verraten werden, besteht aus ganz gewöhnlicher Alufolie, die sich hervorragend als Reflektor eignet. Eingefangene Lichtsignale werden auf diese Art – ganz wie in der Wirklichkeit – auf den „Empfänger“ reflektiert (in unserem Fall ein Fotowiderstand, der zur Radarantenne hin ausgerichtet ist).

Wer dieses Modell bereits aus den früheren Clubheften der 1970er Jahre kennt, dem wird auffallen, dass die Elektronik hier

nicht von den guten alten Silberlingen aus dieser Zeit vertreten wird, sondern durch die IC-Elektronik der Nachfolgeneration. Diese sind tatsächlich leistungsfähiger, allein schon deshalb, weil in der bekannten Dreierkombination von Grundbausteinen im Gegenzug zu den Silberlingen gleich zwei völlig eigenständige Schaltungen untergebracht sind. Um dieses Modell also mit Silberlingen zu steuern, wären zwei Grundbausteine erforderlich, insgesamt also sechs Stück. Die IC-Elektronik schafft das gleiche mit drei – das em3-Relais im Hintergrund ausgenommen. Es dient uns als Polwendeschalter, dazu aber später mehr. Kümmern wir uns zuerst um das Modell.



Abb. 5: Das fertige Modell

Das Fundament

Die Bodenplatten werden in den weiteren Abbildungen noch miteinander verbunden. Das oben nur noch kurz sichtbare Kardangelen verbinden wir nachher mittels einer Achse 110 mit dem Radar. Der fehlende Zahnkranz in der Nabenmutter ist hier kein Fehler, sondern anstelle einer Kette (was eigentlich besser wäre) wird der Antrieb der

Konsole durch eine elastische Schnur bewerkstelligt. Das ist fast genauso gut wie eine Kette, falls nicht genügend Kettenglieder zur Hand sein sollten.

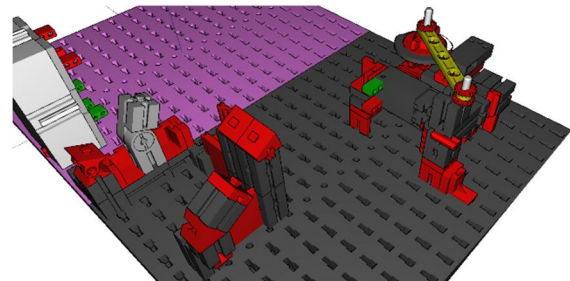


Abb. 6: Zeitschalter platziert und Sockel für den Radarmotor

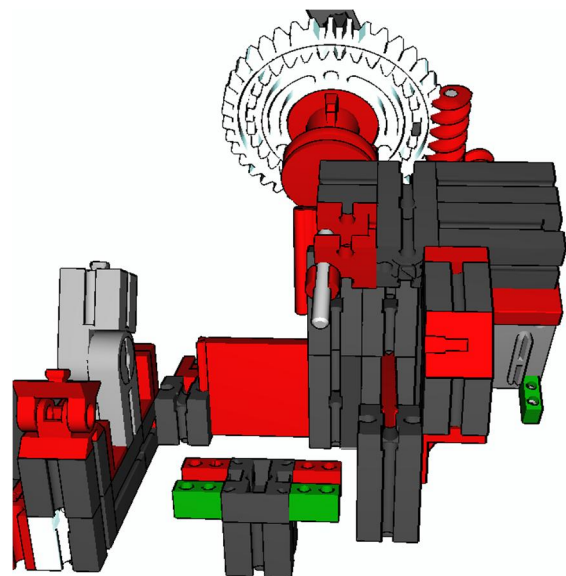


Abb. 7: Aufbau des Radar-Motors

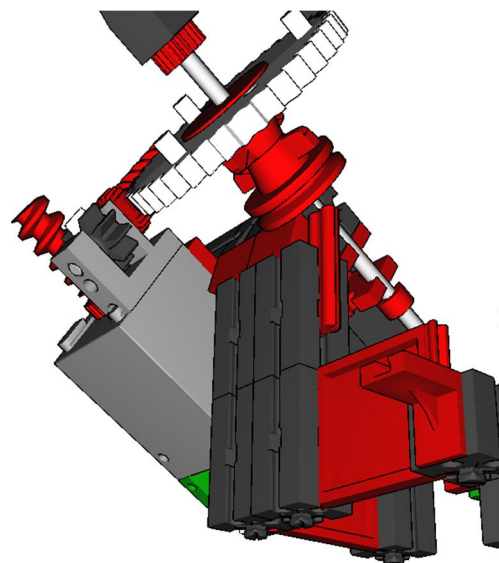


Abb. 8: Aufbau des Radar-Motors

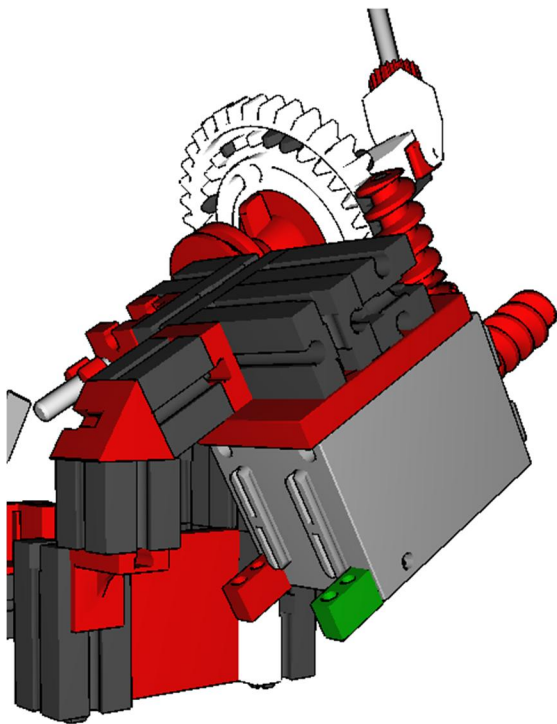


Abb. 9: Antriebsmotor

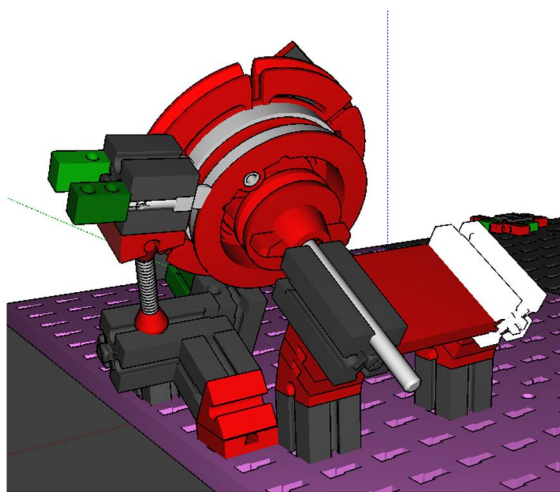


Abb. 10: Aufbau der Konsole; die elastische Schnur wird hinter dem Schleifring angebracht

Leider verfügt der Designer aktuell nicht über die IC-Elektronik, deshalb sind im fertigen Modell an deren Stelle die Silberlinge angebracht. Auch das em3-Relais gibt es derzeit nicht im Designer, deshalb ist hier das ec1-Relais mit Verstärker dargestellt (welches aber nicht so einfach an die IC-Bausteine angeschlossen werden kann).

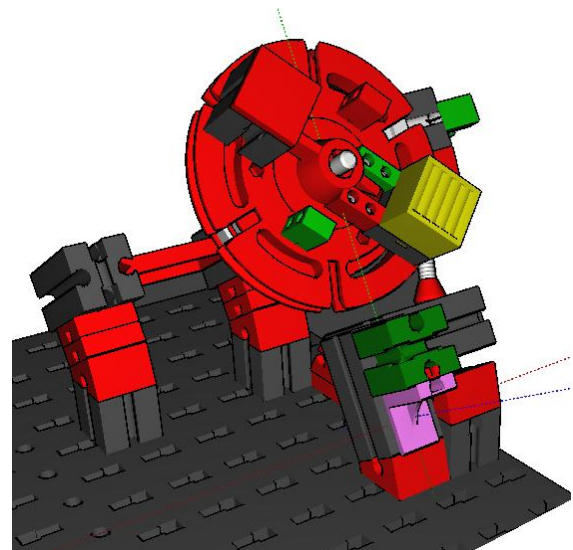


Abb. 11: Konsole von vorne

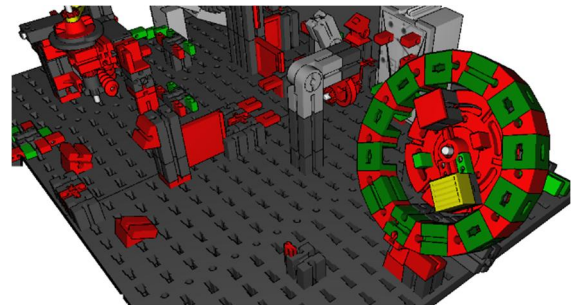


Abb. 12: Fertige Konsole, Kabelführungen und Stützen für die Elektronik-Bausteine.

Die Winkelträger an den Bausteinen 15 sowie das Verbindungsstück 15, wie im Vordergrund auf der Bodenplatte angebracht, dienen nur dazu, später die Kabel ein wenig zu justieren, indem sie z. B. zwischen der Bodenplatte und dem Verbindungsstück 15 leicht eingeklemmt oder von den Winkelträgern unten gehalten werden. Weitere sind je nach Wunsch möglich.

Die Antenne

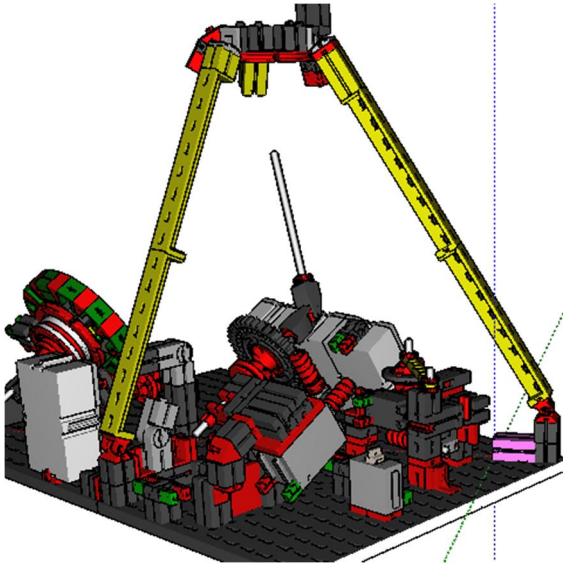


Abb. 13: Gelbe Streben für die Antenne und ein Polwendeschalter zum Einschalten der Beleuchtung.

Wichtig ist hierbei, dass die Bausteine 15 (direkt an dem markierten Baustein 30) einen roten Zapfen zum Drehen haben. Falls keine schwarzen verfügbar sind, gibt es die in der guten alten grauen Ausführung sogar mit roten Nocken beidseitig. Für unsere Zwecke ist sie auf nur einer Seite völlig ausreichend. Am besten baut man alle drei gelben Streben zuerst auf, und dann die rote Drehscheibe 60.

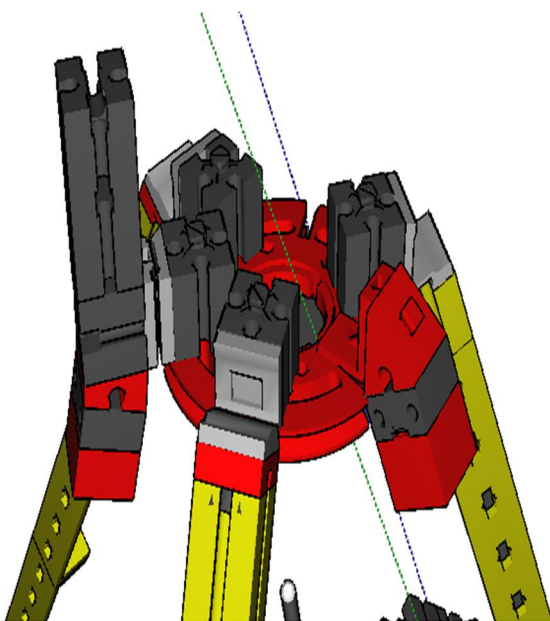


Abb. 14: Montage der Drehscheibe

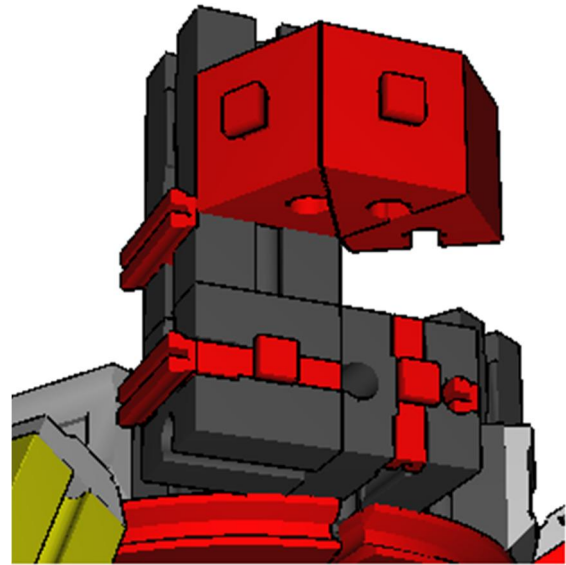


Abb. 15: Federnocken-Halterung

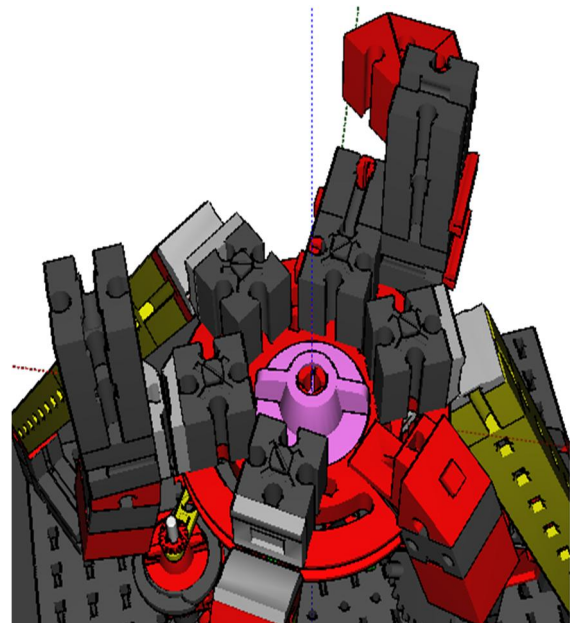


Abb. 16: Montage der Drehscheibe und Federnocken-Halterung

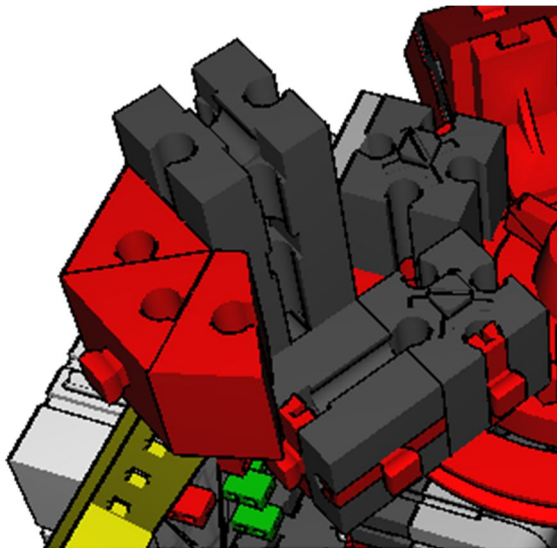


Abb. 17: Montage der Federnocken

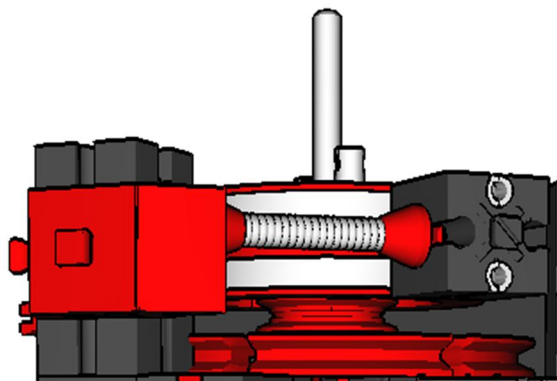


Abb. 18: Schleifring und Federkontakt

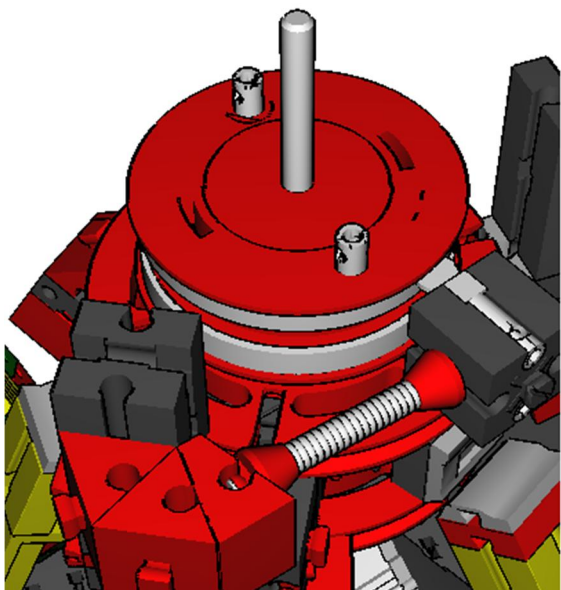


Abb. 19: Montage des Federnockens und des Schleifrings

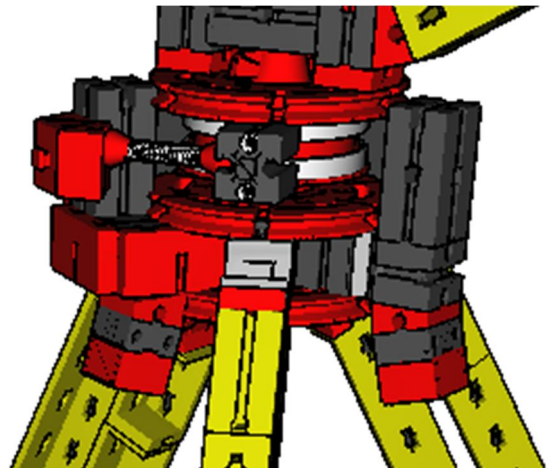


Abb. 20: Aufbau der Antennenmechanik

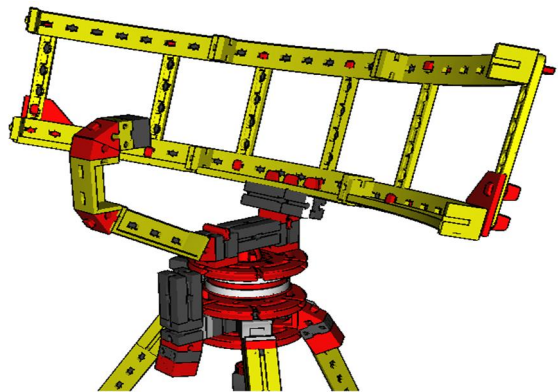


Abb. 21: Aufbau der Antenne

Die Achse 110 im Schleifring wird durch die Drehscheiben geführt und über zwei Kardangelenke mit dem Motor verbunden (siehe Abb. 22/23).

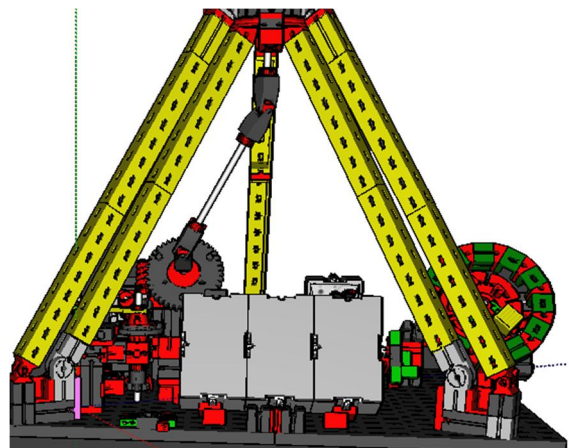


Abb. 22: Grundkonstruktion

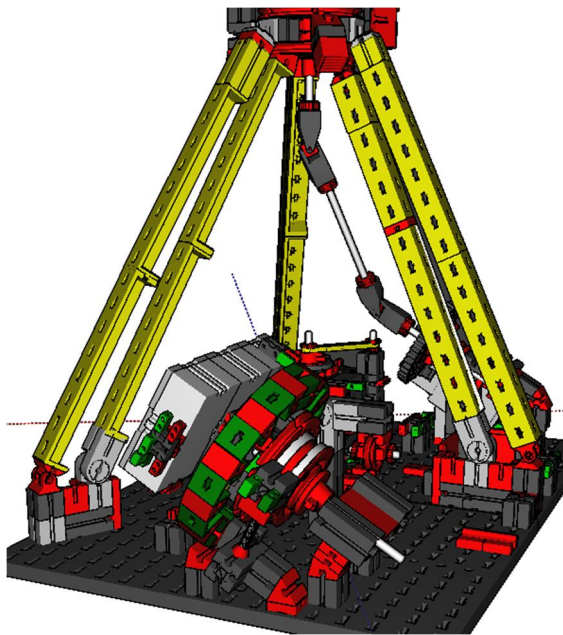


Abb. 23: Radarkonsole und hinterer Aufbau des Grundgerüsts

Die parallelen dreibeinigen Stützen der Grundkonstruktion können durch weitere Streben an der Seite noch stabilisiert werden, wie das Originalfoto in Abb. 5 zeigt.

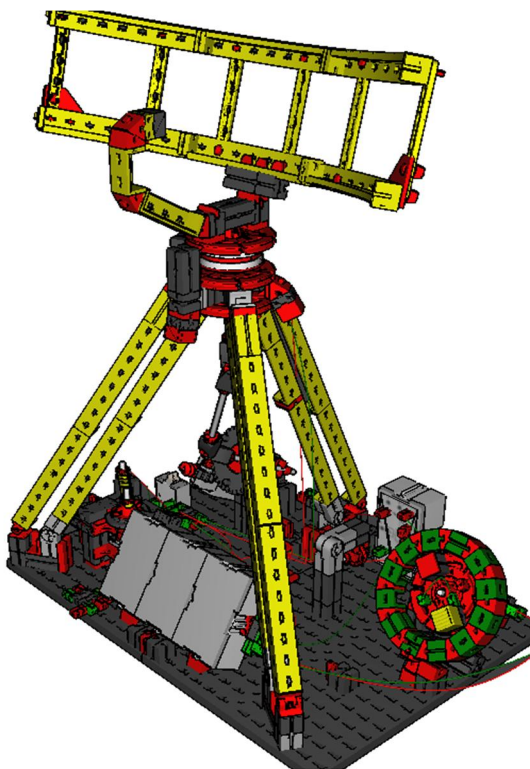


Abb. 24: Das fertige Modell

Wie gesagt sind die drei Silberlinge an der Front stellvertretend für die IC-Elektronikbausteine eingefügt. Die Verkabelung und den Schaltplan des Modells zeigt Abb. 25.

Der Schaltplan

Wie arbeitet die Steuerung? Der in der Abb. 25 rechts dargestellte Taster befindet sich am Zeitschalter (u. a. Abb. 4) und ist die Basis des Ganzen. Wenn kein Licht auf das Radar fällt, ist er durch die Schaltscheiben gedrückt. Die Lücke in den Schaltscheiben hat sich dabei gerade soweit gedreht, dass der Klick erfolgt.

In diesem Ausgangszustand ist A1 des SWS1 (Schwellwertschalters) mit B1 der LST1 (Leistungsstufe) verbunden. Am SWS1 ist über die (+)-Schiene und EA ganz normal ein Fotowiderstand angebracht, der sich oben auf der Radarantenne befindet. Ohne Lichtkontakt ist A1 negativ und die über den Taster mit ihr verbundene LST1 ausgeschaltet.

An die LST1 ist über C1 und (+)-Schiene die Lampe auf der Steuerkonsole angeschlossen. Bei LST1 „aus“ ist auch sie aus. Kommt der Fotowiderstand ins Licht, schaltet sich LST1 ein und die Lampe signalisiert den Kontakt.

Soweit die Grundsteuerung des Radars. Wir möchten aber zusätzlich, dass es sich auf den Lichtkontakt einige Augenblicke lang „einpendelt“. Dazu benötigen wir den Zeitschalter. Was tut dieser nun genau?

Sobald sich bei Lichtkontakt LST1 einschaltet, startet der Zeitschalter-Motor, weil er ebenso wie die Lampe an C1 der LST1 angeschlossen ist. Die Schaltscheiben sind so angeordnet, dass der Taster so lange gedrückt bleibt, bis ihn die kleine Lücke in den Schaltscheiben nach einer ganzen Umdrehung freigibt.

Sobald das geschieht, ist B1 nicht mehr mit A1 verbunden, sondern mit A1-Nicht. A1-Nicht ist, wenn Licht auf den Fotowiderstand fällt, negativ. Damit geht die LST1

aus und der Motor-Zeitschalter bleibt stehen.

Jetzt kommt die Pendelbewegung ins Spiel. Mit dem Start des Zeitschalters durch LST1 wird auch eine Blockade von LST2 aufgehoben. LST2 ist mit SWS2 verbunden, der als Taktgeber geschaltet ist. Er wird über den Widerstand und die Diode, (zwischen B2 und B1 – auf richtige Polung der Diode achten; (+) in Richtung B2) solange blockiert, bis die LST1 einschaltet.

Wie geht das? Bei LST1 aus ist A1 (-). Da A1 mit B1 verbunden ist, liegt also auch an B1 Minus. Damit wird Plus von SWS2 bei jedem Takt über A1(-) sofort abgeleitet, sodass sich die LST2 gar nicht erst im Takt des SWS2 ein- und ausschalten kann. Er ist immer aus, wenn LST1 aus ist, obwohl der Takt in SWS2 ständig arbeitet.

Das Ende der Blockade leitet also den Takt an die LST2 weiter, die sich folglich im Takt des SWS2 ein- und ausschaltet. An C2 der LST2 ist ein em3-Relais angeschlossen, das als Polwendschalter arbeitet und über eine zweite Stromquelle (Trafo) den Radarmotor steuert. Der zweite Trafo ist deshalb erforderlich, weil sonst die Elektronikbausteine vom Relais-Takt gestört werden können.

Der Takt schaltet am Ende das Relais ein und aus und versetzt somit das Radar für die Dauer, bis der Zeitschalter eine Umdrehung der Schaltscheiben geschafft hat, in den Pendelzustand. Das Radar bleibt in dieser Zeit normalerweise im Lichtkontakt und hat ihn auch noch, wenn der Zeitschalter auf die Schaltscheiben-Lücke stößt, die den Taster wie gesagt löst. Jetzt ist A1-Nicht mit B1 verbunden und A1-Nicht führt (-). Das schaltet die Blockade von LST2 durch die Diode wieder ein. Sie geht aus und das Pendeln hört auf. Der Radarmotor dreht

sich jetzt wieder normal weiter – weg vom Licht.

Sobald aber der Fotowiderstand den Kontakt verliert wird A1-Nicht wieder positiv (+). Dies schaltet für einen Augenblick den Zeitschalter wieder ein. Er dreht sich weiter bis der Taster wieder gedrückt wird. Damit schaltet sich A1 wieder an B1. A1 führt aber ohne Licht (-) und die LST1 geht sofort wieder aus. Der Zeitschalter bleibt stehen und ist wieder „scharf“ gemacht für den nächsten Lichtkontakt. Während dieses Moments wird zwar auch der Takt wieder aktiv; da das aber nur einen Augenblick lang der Fall ist, kommt es höchstens zu einem kurzen Ruckeln der Antenne. Sie bleibt aber außerhalb des Lichtkontaktes, weshalb der Prozess in seinem geplanten Fortgang nicht weiter gestört wird. Fertig!

Der Taktgeber und die Schaltung des Fotowiderstandes stammen aus dem IC-Elektronik-Handbuch, welches in der ft-Datenbank zum Download bereitsteht [1].

Zwischen der Lampenfassung und den drei Lampen für die Beleuchtung des Modells kann noch ein Polwendschalter (wie in den Abbildungen sichtbar) eingebaut werden, mit dem man die Lampen ein- und ausschalten kann. Die übrige Verkabelung am Modell sollte so vorgenommen werden, dass die als Kabelführung montierten Elemente auf der Platte genutzt werden. Das verhindert ein Verheddern der Leitungen in die laufende Mechanik.

Nun bleibt mir an dieser Stelle nur noch, viel Spaß beim Basteln und Spielen zu wünschen.

Quellen

- [1] fischertechnik: [Elektronik](#). Fischer-Werke, Tümlingen, 1981.

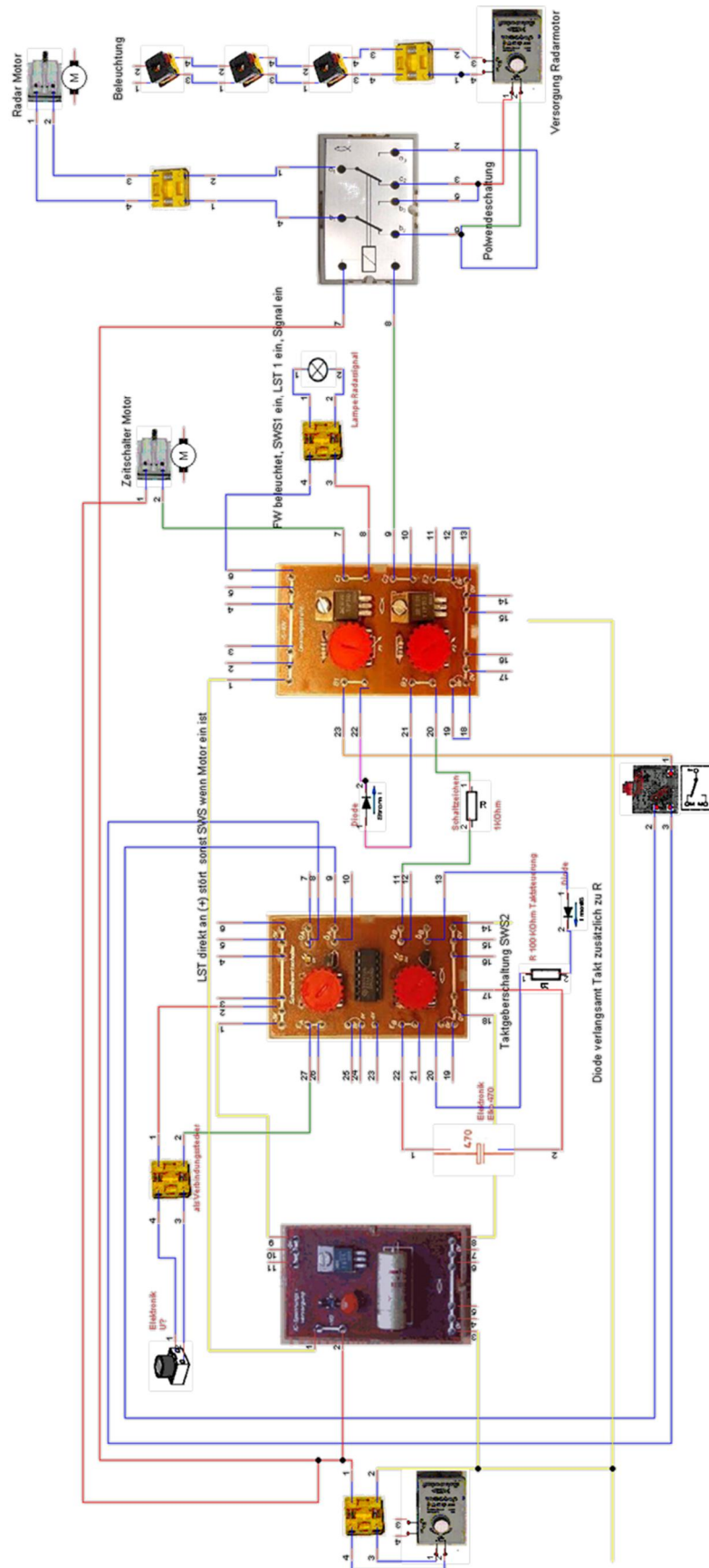


Abb. 25: Schaltplan der Radarsteuerung

Optik

Laser-Anwendungen (3): Analoger CD-Spieler

Andreas Gail

Verbaut in CD-Spielern kamen Laser in den 1980er Jahren massenhaft in die Haushalte. Derartiges mit fischertechnik nachzubauen ist sicherlich eine besondere Herausforderung, wenn echte CDs abgespielt werden sollen. Hierzu bedarf es einer ausgeklügelten Feinwerktechnik und Elektronik bzw. Software. Ein Modell jedoch ist durchaus möglich, wie nachfolgender Beitrag zeigt. Hierbei ist die Bezeichnung „Analoger CD-Spieler“ ein Wortspiel: Einerseits wird ein Modell gezeigt, welches prinzipiell ähnlich zu einem kommerziellen Gerät arbeitet, andererseits handelt es sich bei der Tonaufzeichnung im Grunde um ein analoges Signal, welches wiedergegeben wird.

Die Idee ist nicht neu

Im Jahr 1991 wurde bereits ein CD Spieler mit fischertechnik nachgebaut [1], um das Prinzip zu zeigen. Vermutlich gab es eine Reihe weiterer Modellvorschläge.

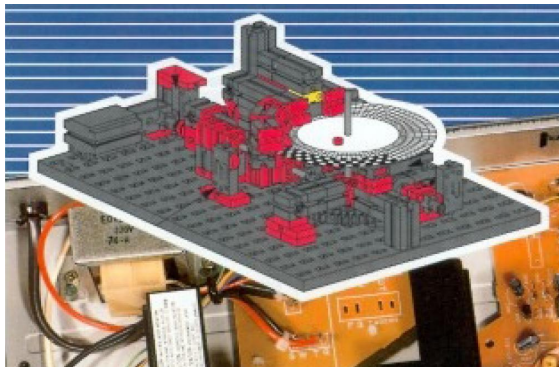


Abb. 1: CD Spieler Modell aus dem Experimentierbuch Profi Computing [1]

In der hier vorliegenden Beschreibung werden jedoch nicht einfach schwarze und weiße Kästchen ausgefüllt und später ausgelesen. Deshalb ist die Modellbeschreibung zweigeteilt. Zunächst wird eine PC-Software (PDD Creator, Paper-DataDisc Creator) [4] beschrieben, die aus einem einfachen einstimmigen Musikstück einen runden Datenträger erzeugt (PDD, Paper Data Disc), der einfach ausgedruckt und mit einer Schere ausgeschnitten werden

kann. Im zweiten Teil wird das fischertechnik-Modell gezeigt, welches diese PDD wieder auslesen kann. Zum Aufbau werden zwei Sonderbaugruppen benötigt, die in früheren Ausgaben der ft:pedia veröffentlicht wurden: ein Punkt-laser [2] und ein PWM-Tongenerator [3].

Um die Erwartungen gleich zu Beginn nicht zu hoch zu schrauben: bei dem hier Gezeigten handelt es sich um ein Modell, die Tonqualität ist durchaus noch verbesserungsfähig. Dazu am Ende mehr.

PDD-Datenträger erstellen

Zunächst muss die Software PDD-Creator [4] aus dem Internet beschafft werden. Sie wurde mit Microsoft Visual Basic Express 2010 in einer Windows XP SP3-Umgebung entwickelt. Zusätzlich getestet wurde diese Software unter Windows 7.

Nach dem erfolgreichen Programmstart erhält man ein Fenster gemäß Abb. 15. Allerdings muss zunächst ein passendes Musikstück eingegeben bzw. geladen werden. Mithilfe der Befehlsschaltflächen *Play All* oder *Play Partial* kann das Musikstück über den Systemlautsprecher des PCs überprüft werden. Unter Nutzung der Befehlsschaltfläche *Create* lässt sich

entsprechend der Parameter *Number of circles* (Abbildungsbreite der Datenspur) und *Single sound length* (Tondauer) die PDD auf der rechten Fensterhälfte erstellen. Die Befehlsschaltflächen *Print...* bzw. *Print Now* ermöglichen den Ausdruck über einen angeschlossenen Drucker. Wird der erhaltene kreisrunde Ausdruck ausgeschnitten ist die PDD fertig. Abb. 13 zeigt ein vollständiges Beispiel.

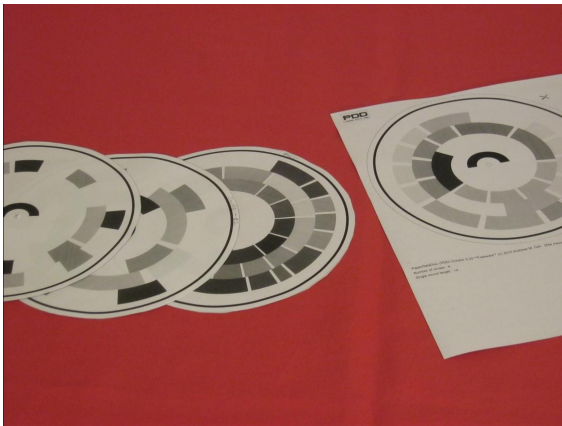


Abb. 2: mehrere PDDs zum Abspielen

Es sei angemerkt, dass beim späteren Abspielen die Drehgeschwindigkeit des Plattentellers konstant sein soll. Folglich müssen die einzelnen Töne bei gleicher Tonlänge am Außenrand der PDD eine längere Tonspurlänge haben als am Innenrand.

Der PDD Spieler

Die erstellte PDD wird auf den Plattenteller des PDD Spielers aufgelegt und die Leseinheit darüber geklappt. Das wird in Abb. 3 bis 5 gezeigt.

Der *Start*-Taster beginnt den Abspielvorgang, der *Stop*-Taster unterbricht ihn. Auch durch Öffnen der Leseinheit werden alle Vorgänge abgebrochen, d. h. die Motoren sowie der Laser werden abgestellt.

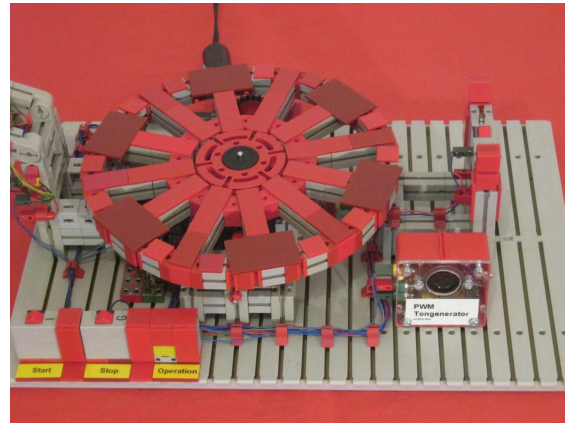


Abb. 3: PDD Spieler, Leseinheit offen

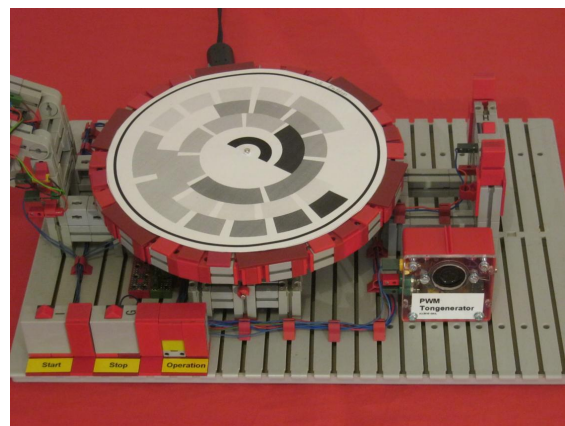


Abb. 4: PDD Spieler, Leseinheit offen, mit aufgelegter PDD

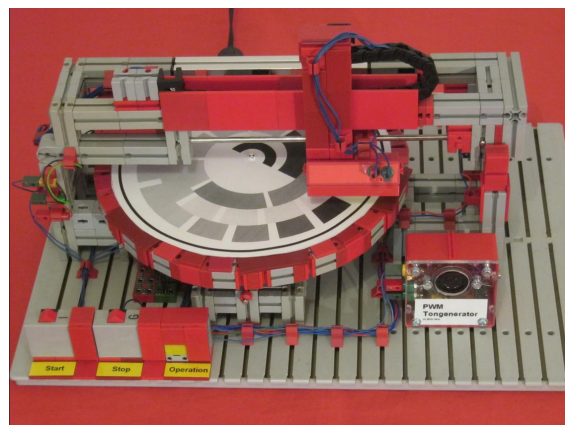


Abb. 5: PDD Spieler, mit aufgelegter PDD, abspielbereit

Detailvorgänge im PDD-Spieler

Radiale Positionierung

Mit radialer Positionierung ist die Bewegung der Leseinheit gemeint. Die beiden Endpunkte der Positionierung werden dabei durch zwei Taster begrenzt, wie in Abb. 6 violett dargestellt. Die Tasterbetätigung erfolgt durch den ebenfalls violett dargestellten Stift. Ist ein Endpunkt gefunden, wird die Positionseinstellung mithilfe des Encodermotors durchgeführt, d. h. es werden einfach die zuvor einmal bestimmten Schritte gezählt, um eine gewünschte Position anzufahren.

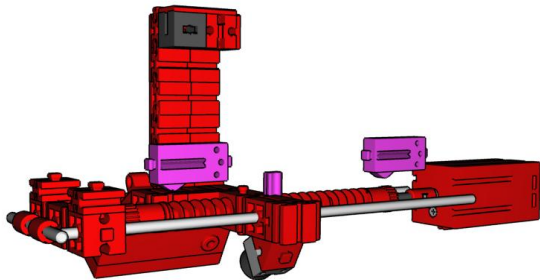


Abb. 6: Endpunkt-Positionierung der Leseinheit

Schwarz-weiß-Kalibrierung

Der Vorgang der schwarz-weiß-Kalibrierung wird in Abb. 7 gezeigt. Während sich der Plattenteller dreht, wird die PDD punktuell per Laser beleuchtet. Fällt dieses Licht auf eine schwarze Fläche (links), ist die Reflexion auf den Fotowiderstand geringer als bei der Reflexion von einer weißen Stelle (rechts). Um die unterschiedliche Helligkeit im RTXC auswerten zu können, muss der zugehörige Eingang auf die Betriebsart A5k (analog 5k Ω) eingestellt sein.

Die Kalibrierungsdauer wurde auf 10 s festgelegt. Während dieser Zeit wird der insgesamt hellste und auch der dunkelste Messwert festgehalten und in je einer Variablen für die spätere Verwendung gespeichert. Während der Kalibrierung und

dem nachfolgenden Abspielen einer PDD sollte sich die Raumhelligkeit nicht ändern.

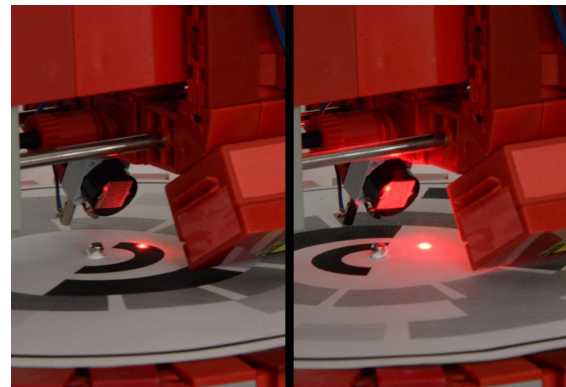


Abb. 7: schwarz-weiß-Kalibrierung und Positionierung des Plattentellers

Startpunkt-Suche

Die Suche des Startpunktes schließt sich direkt an die schwarz-weiß-Kalibrierung an. Die PDD wird dabei in Richtung des grünen Pfeils gemäß Abb. 8 gedreht; die Leseinheit ist weiterhin auf das schwarz-weiß-Kalibrierfeld eingestellt. Mit den zuvor ermittelten Werten für schwarz und weiß wird nun während der Drehung des Plattentellers der Übergang von schwarz nach weiß gesucht. Wenn dieser gefunden wurde, wird der Plattenteller angehalten und wieder etwas zurückgedreht. Die Leseinheit wird auf den Startpunkt gefahren. Nachfolgend kann mit dem Abspielen begonnen werden.

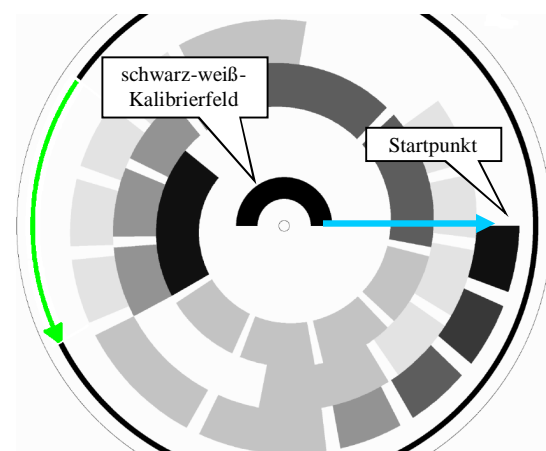


Abb. 8: Startpunkt Suche, ausgehend vom schwarz-weiß-Kalibrierfeld

Elektronisches Getriebe

Bei einem Getriebe wird (z. B.) eine Drehbewegung von einem Zahnrad auf ein anderes übertragen. Sind die Zahnräder unterschiedlich groß, ändert sich die Drehgeschwindigkeit. Im vorliegenden Fall haben sowohl der Plattenteller als auch die Leseinheit einen eigenen Encoder-Motor. Trotzdem soll eine aufeinander abgestimmte Bewegung beider Einheiten möglich sein. Insbesondere beim Abspielen soll die Leseinheit möglichst kontinuierlich von rechts nach links verfahren werden, während sich der Plattenteller kontinuierlich dreht. So soll sichergestellt werden, dass die Leseinheit den immer enger werdenden Tonspuren folgt. Um das zu erreichen, wird der Antrieb des Plattentellers eingeschaltet und der zugehörige Zählerstand des Encoder Motors beobachtet. Nach 20 Schritten wird die Leseinheit um 11 Schritte nach links bewegt. Ideal wäre es, wenn die Leseinheit nach z. B. zwei Schritten des Plattentellers um einen Schritt weiterlaufen würde. Dann würde sich ein deutlich kontinuierlicheres Laufverhalten ergeben. Hierbei traten bei der Verwendung des RTX-C jedoch Schrittfehler auf.

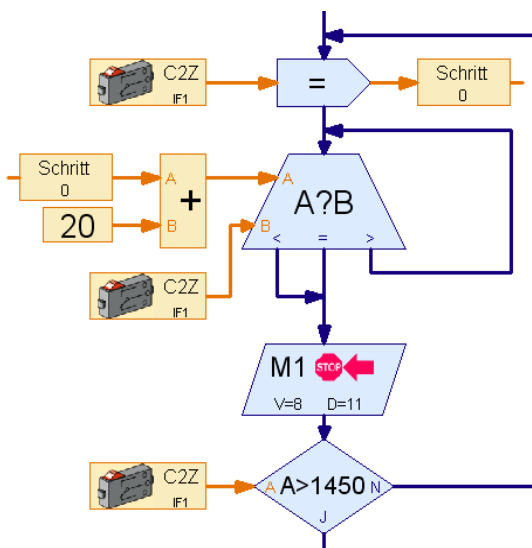


Abb. 9: Elektronisches Getriebe, M1 = Antrieb Leseinheit, M2 = Antrieb Plattenteller

Modellaufbau

Nachfolgend einige Bilder zum Modellaufbau:

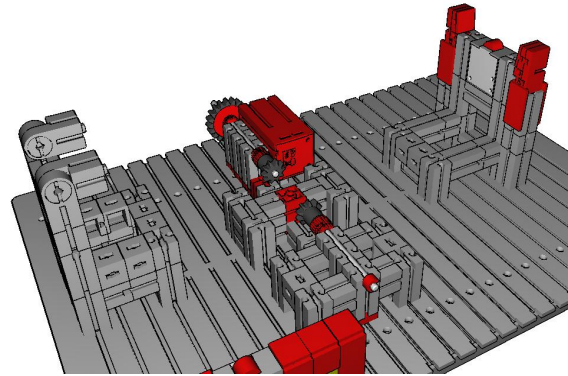


Abb. 10: Grundaufbau, Antrieb Plattenteller

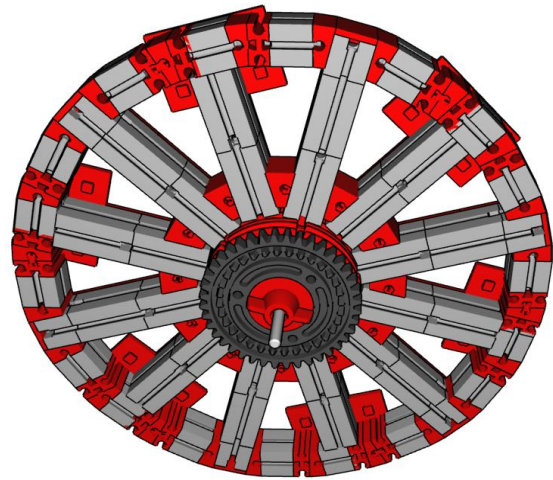


Abb. 11: Plattenteller

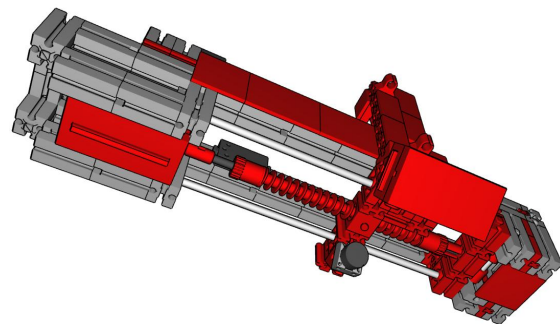


Abb. 12: Leseinheit

Verbesserungspotential

Dass das hier gezeigte Modell keine perfekte CD-Qualität liefern würde, war von Beginn an klar. Nachfolgend ein paar Anregungen, für ambitionierte fischartechniker. Erfindungen wurden und werden schrittweise verfeinert, wie z. B. das Auto oder der Computer. So ist es auch hier.

Plattentellerantrieb

Der Antrieb des Plattentellers über Zahnräder läuft etwas holprig. Über Gummiräder wäre vielleicht eine ruhigere Bewegung möglich. Auch die weit verbreitete Riemenlösung wäre denkbar. Allerdings darf es dabei keinen Schlupf geben, sonst läuft das elektronische Getriebe des Plattentellers zum Lesekopf nicht mehr synchron und der Lesekopf verlässt die vorgegebene Spur.

Spurbreiten auf der PDD

Die verwendeten Spurbreiten könnten mit der bereitgestellten Software PDD Creator deutlich schmaler erstellt werden, und auch der Lesekopf könnte feinere Konturen auslesen. Allerdings müsste das elektronische Getriebe dann deutlich feiner arbeiten und ein Nachführen des Lesekopfes müsste mit Einzelschritten erfolgen. Hierzu ist der verwendete RTXC unter Nutzung von RoboPro zu langsam; die Nachführung ist nur in Zehnersprüngen möglich.

Tonverfremdung an Übergangsstellen

Wird ein tiefer Ton bei einem schwarzen Feld ausgelesen und es folgt eine weiße Tonpause, so wird an der Grenze der beiden Zustände ein Mischton ausgelesen, d. h. die Tonhöhe erhöht sich. Dieses Problem könnte verbessert werden, indem der Laser stärker senkrecht die PDD beleuchtet und dadurch ein kleinerer Lesefleck entstünde. Dadurch wäre auch der unerwünschte Zwischenzustand kürzer. Weiterhin könnte eine höhere Drehgeschwindigkeit des Plattentellers das Phänomen vermindern.

Ein anderer Ansatz wäre, die ausgelesenen Töne zeitverzögert abzuspielen und den unerwünschten Effekt per Software im RTXC zu vermeiden.

Tonverfremdung durch Störlicht

Die Leseinheit ist relativ offen; Umgebungslicht kann somit den Lesevorgang beeinflussen. Um diesen Effekt zu vermindern, wird zu Beginn des Abspielens eine schwarz-weiß-Kalibrierung vorgenommen (siehe oben). Zur Verbesserung der Situation erscheint es sinnvoll, aus dünner schwarzer Pappe eine Störlichtabschottung zu ergänzen.

Tonfrequenz

Aus Gründen der Rechenkapazität des RTXC wurde darauf verzichtet, eine von der PDD ausgelesene Helligkeit korrekt in eine Note oder Tonfrequenz umzurechnen. Wie zuvor gezeigt [3] wäre dazu eine Gleitkomma-Funktion höherer Ordnung zu berechnen, und das mit maximaler Geschwindigkeit.

Tonerzeugung

Wenn für den Robo TXT Controller zukünftig die Möglichkeit bestünde, eigene Töne abzuspielen, könnte die Tonerzeugung insgesamt deutlich vereinfacht werden.

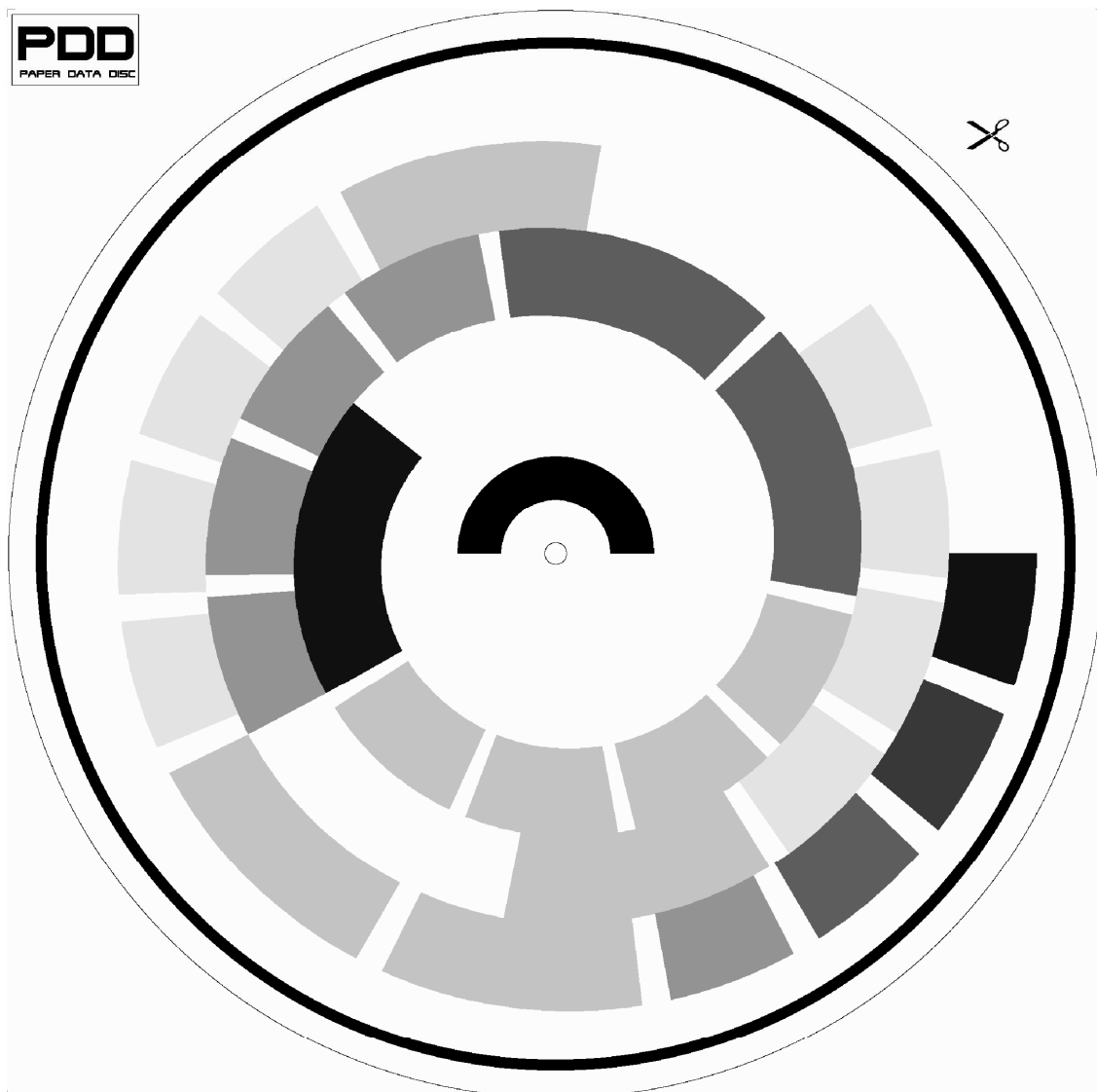
Höhere Rechengeschwindigkeit mit C

Grundsätzlich lässt sich der RTXC in C programmieren. Dieses könnte ein Weg sein, einige der oben genannten Probleme per Software zu lindern.

Referenzen

- [1] *Experimentierbuch Profi Computing*. Richard Pflaum Verlag GmbH & Co. KG München, 1991, ISBN 3-7905-0625-7. [Download vom fischertechnik-museum.ch](http://www.fischertechnik-museum.ch).

- [2] Gail, Andreas: *Einstieg in Experimente mit Lasern*. [ft:pedia 2014-2](#)
- [3] Gail, Andreas: *PWM-Tongenerator für Robo TX(T) Controller*. [ft:pedia 2015-4](#)
- [4] Gail, Andreas: *PaperDataDisc Creator*. [ft Community-Download](#)
- [5] Gail, Andreas: *Analog CD Spieler*. [youtube-Video](#)



PaperDataDisc (PDD) Creator 0.20 **Freeware** (c) 2015 Andreas M. Gail [Alle meine Entchen.PDD]
Number of circles : 4
Single sound length : 14

Abb. 13: PaperDataDisc (PDD), Kreisdurchmesser im Original 19,5cm

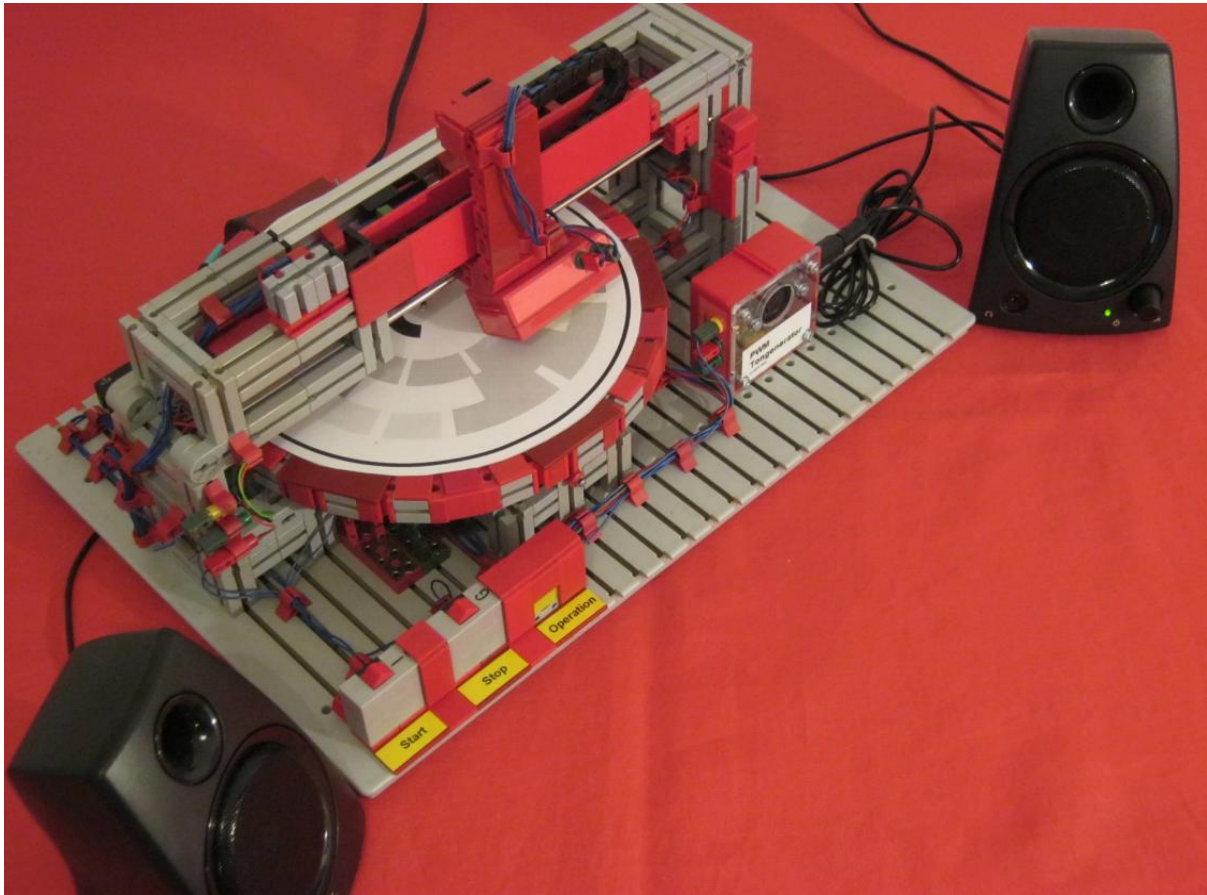


Abb. 14: PDD Spieler mit aufgelegter PDD und angeschlossenen Lautsprechern

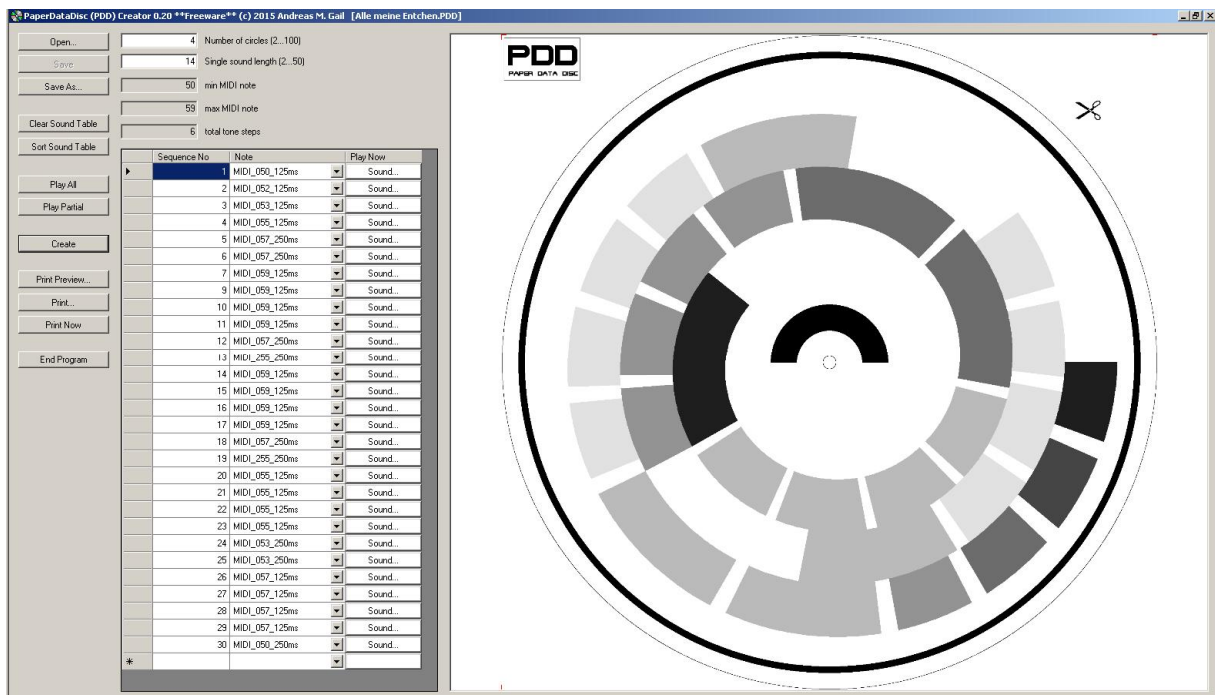


Abb. 15: PaperDataDisc (PDD) Creator, freeware [4]

Computing

TXT Controller – Tipps und Tricks (1): Das Root-Passwort

Raphael Jacob

Seit dem Erscheinen des ROBOTICS TXT Controllers (kurz: TXT) kurz vor Weihnachten 2014 fehlen leider noch immer einige der beworbenen Funktionen. Damit die Computerenthusiasten unter uns ihre eigenen Funktionen programmieren und testen können, benötigt man in den meisten Fällen einen „Root-Zugang“. Wie ihr diesen einrichten und auch sicher nutzen könnt zeige ich in diesem Beitrag.

Hintergrund

Um bei Linux systemkritische Dateiänderungen durchführen zu können, benötigt man wie bei Windows „Administratorrechte“. Diese können entweder über einen Rechteeeditor wie z. B. *visudo* verliehen werden, oder man loggt sich direkt mit dem Admin-Account ein. Bei unserer Vorgehensweise werden wir zunächst Letzteres tun, da dies beim TXT einfacher ist und Grundvoraussetzung für die Benutzung von *visudo*.

Durchführung

Der Weg zum Terminal

Um die Befehle durchzuführen, benötigt man Zugriff zum internen Terminal des TXT (ähnlich wie die Kommandozeile bei Windows). Dafür haben wir zwei Möglichkeiten:

- Man kann eine normale USB-Tastatur benutzen und am TXT anschließen. (Funksysteme habe ich nicht getestet und ich kann somit nicht bestätigen, ob diese funktionieren.) Um das Terminal zu öffnen drückt man die ALT-Taste (links, nicht ALT GR) und die Taste F2 gleichzeitig. Nun verschwindet die Bedienoberfläche und nur die Uhr bleibt.



Abb. 1: Loginmaske lokales Terminal

- Man kann sich auch per ssh (Secure Shell, also „Remote-Kommandozeile“) verbinden. Dafür benutze ich beispielweise die Software Putty ([Download](#)).



Abb. 2: Putty-Downloadseite

Putty muss nicht installiert werden und braucht nur z. B. auf dem Desktop abgelegt zu werden.

Nun öffnet man das Programm und gibt oben die IP-Adresse des TXTs ein. (USB: 192.168.7.2, W-LAN: 192.168.8.2, BT: 192.168.9.2) Dabei ist USB zu empfehlen.

Die Einstellung Port ist auf 22 zu belassen, so auch „Connection Type“ auf „SSH“. Bei Bedarf kann man diese Einstellungen noch abspeichern, indem man bei „Saved Sessions“ einen passenden Namen einträgt und auf „Save“ drückt. Diese können durch Doppelklick auf den entsprechenden Listeneintrag abgerufen werden. Ohne Abspeichern fährt man mit einem Klick auf „Open“ fort.

Einloggen



Abb. 3: Abfrage Benutzername

Bei „login“ schreibt man nun: „ROBOPro“ (ohne die Anführungszeichen; Groß- und Kleinschreibung beachten). Dies bestätigt man mit der Eingabetaste. Bei Passwort schreibt man nun auch „ROBOPro“, jedoch sollte man sich nicht davon irritieren lassen, dass man die Eingabe nicht sieht. Auch das Passwort bestätigen wir mit der Eingabetaste. Jetzt wird man eingeloggt und man wird von folgendem Bildschirm begrüßt:

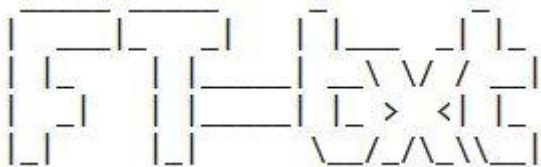


Abb. 4: Begrüßungsbildschirm

Auslesen des Zufallspasswortes

Um das schon ab Werk hinterlegte Root-Passwort abzurufen, muss man das folgende Kommando ausführen [1]:

```
echo "showroot=1" > .TxtAccess.ini
```

Um das Kommando auszuführen, tippt man es entweder ab oder man kopiert es und fügt

es mit einem Rechtsklick in Putty ein. Nun startet man den TXT wie gewohnt neu. Das Passwort erhält man nun, indem man auf „Einstellungen“ und dann „Info“ drückt. Dort ist es in der letzten Zeile der ersten Seite zu sehen. Falls man am TXT selbst arbeitet und nicht über SSH, sollte man sich das Passwort aufschreiben (es sind nur Großbuchstaben).

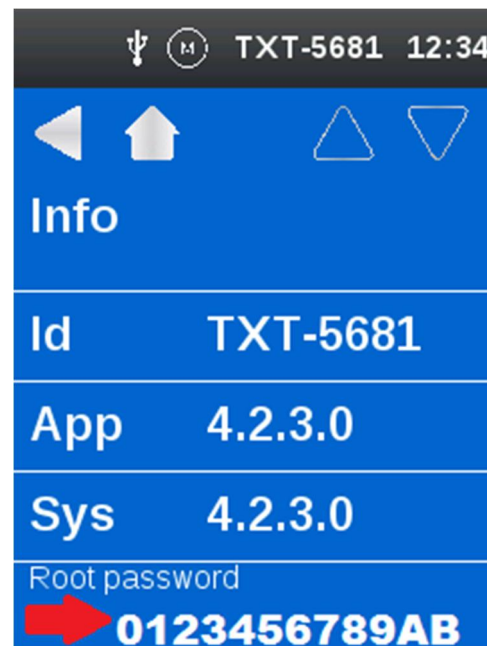


Abb. 5: Anzeige des TXT-Root-Passworts

Login als root

Um nun das neue Root-Passwort auch nutzen zu können muss man sich als Nutzer „root“ einloggen. Dies geschieht wie oben beschrieben, jedoch mit dem Benutzername „root“ und dem Passwort, welches man in den Einstellungen des TXTs findet (nur Großbuchstaben).

Setzen eines eigenen Passwortes

Auch wenn der TXT wohl nicht das Ziel von Hackern sein wird, sollte das Passwort trotzdem aus vielerlei Gründen geändert werden. Das geht wie folgt:

Nach dem Login als „root“ gibt man folgendes Kommando ein: „passwd root“ Dies wird mit „Enter“ bestätigt und man wird nach dem neuen „root“-Passwort gefragt, welches man auch mit „Enter“

quittiert. Danach wird noch einmal nach dem neuen Passwort gefragt, und man bestätigt auch dieses. Wenn die letzten beiden Eingaben identisch waren, kommt die folgende Nachricht:

```
# passwd root
Changing password for root
New password:
Retype password:
Password for root changed by root
```

Abb. 6: Ändern des „root“-Passwortes

Firmwareupdates

Da Firmwareupdates mit Root-Rechten installiert werden müssen, liest RoboPro vor einer Installation aus einer Datei, die auf dem TXT gespeichert ist, das „root“-Passwort aus. Wenn man dieses jedoch geändert hat, steht in dieser Datei noch das alte Zufallspasswort und RoboPro kann sich nicht authentifizieren. Dann erscheint die folgende Meldung:

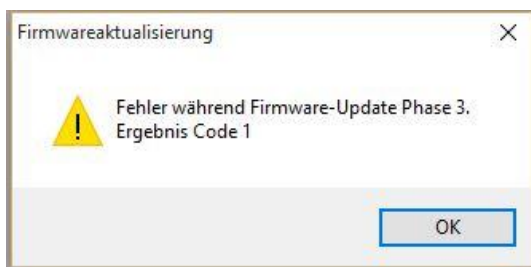


Abb. 7: RoboPro-Updatefehler bei eigenem „root“-Passwort

Um diesem Problem aus dem Weg zu gehen, gibt es zwei Möglichkeiten:

- Man setzt ein Zufallspasswort: Zunächst muss man natürlich als „root“ oder „ROBOPRO“ eingeloggt sein. Dann gibt man folgendes Kommando ein:

```
sudo /etc/init.d/rndpwd
```

Dies wird mit einer ähnlichen Meldung wie nach dem Setzen des eigenen Passwortes quittiert. Danach kann das Update wie gewohnt eingespielt werden.

- Man kann das Update auch manuell installieren: Dazu muss man die Datei „update.sh“ auf den TXT kopieren. Das

macht man mit einem SFTP-Client. Ich nutze beispielsweise „[Filezilla](#)“ (Abb. 8).



Abb. 8: Filezilla Downloadseite

Nun startet man Filezilla. Um einen neuen Server hinzuzufügen, drückt man auf den Knopf ganz links oben (1, Abb. 9). Jetzt öffnet sich ein neues Fenster „Servermanager“. Dort legt man mit Druck auf „Neuer Server“ (2) einen neuen Eintrag an. Nun erscheint links in der Liste der neue Eintrag (3), welchen man dann direkt umbenennen kann. Jetzt müssen rechts die wichtigen Informationen ausgefüllt werden. Bei „Server“ (schwarz 1) wird die IP-Adresse des TXT eingetragen (USB: 192.168.7.2, W-LAN: 192.168.8.2, BT: 192.168.9.2). Das Feld „Port“ (2) kann normalerweise leer bleiben oder man trägt „22“ ein. Bei „Verbindungsart“ (3) wählt man „SFTP – SSH...“ aus. Für die Authentifizierung (4) wählt man „Normal“ und die Felder „Benutzer“ und „Passwort“ (5, 6) werden mit den Zugangsdaten für den Nutzer „root“ ausgefüllt. Zum Schluss drückt man auf „Verbinden“ (rot 5).

Im linken oberen Kasten navigiert man nun zum Ordner `C:\Program Files (x86)\ROBOPRO\ROBO Interface Firmware\` und wechselt dann in den Unterordner `TXTController`. Dort sieht man die Datei `update.sh`. Rechts navigiert man zu `/opt/knobloch`. Jetzt macht man links einen Doppelklick auf die Datei `update.sh`, um diese zum TXT zu übertragen und bestätigt die Meldung mit „Enter“. Zum Schluss schließt man Filezilla und startet wieder Putty. Dort loggt man sich wieder mit dem Account „root“ ein und führt folgendes Kommando aus:

```
cd /opt/knobloch/ && ./update.sh
```

Jetzt startet das Firmwareupdate. Wenn dieses abgeschlossen ist, fährt der TXT herunter. Nun starten wir ihn wieder. Dabei ist zu beachten, dass jetzt das „root“-Passwort wieder gelöscht ist (Das neue Passwort ist in den Einstellungen zu finden) und bei Bedarf wie oben beschrieben wieder manuell gesetzt werden muss.

Quellen

- [1] Fischertechnik: [ROBOTICS TXT Controller Security Information](#), Firmware Version 4.1.4, 2015.

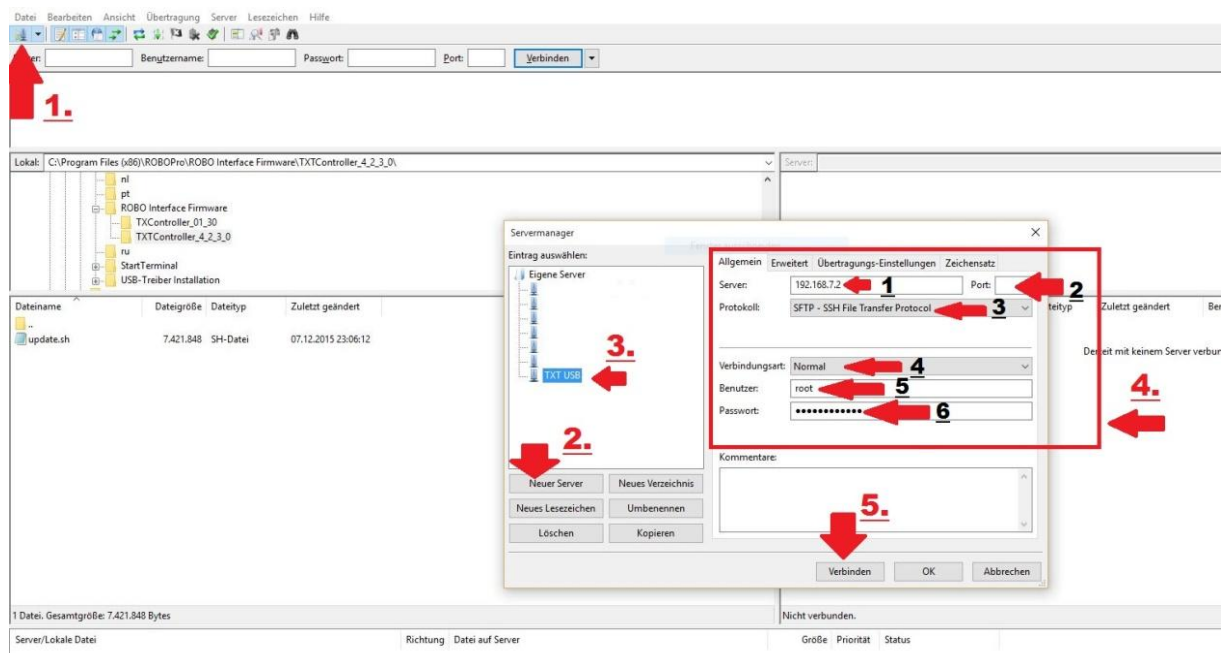


Abb. 9: Filezilla mit geöffnetem Servermanager

Computing

Digitalkamera mit Autofokus und Live-Video-Vorschau

Torsten Stuehn

Digitalkameras sind heute weit verbreitet und es gibt sie in den verschiedensten Spielarten – von der kompakten Systemkamera bis hin zur professionellen Spiegelreflex-Kamera (SLR). Mit dem Erscheinen des „TXT Discovery Sets“ findet man auch im fischertechnik-Sortiment alle benötigten Bauteile und Komponenten, um ein funktionierendes Digitalkamera-Modell mit Auto-Fokus und Live-Video-Vorschau bauen zu können.

Hintergrund

Beim Nachbau verschiedener Modelle des *TXT Discovery Sets* stößt man an einigen Stellen auf die Herausforderung, dass man den Fokus der mitgelieferten einfachen Webcam durch Drehen am Objektiv manuell so einstellen soll, dass ein Objekt in einem bestimmten Abstand scharf dargestellt wird. Die mitgelieferte ROBOPro Software bietet als Hilfe eine fluktuierende Balkenanzeige an, die ein ungefähres Maß für die Schärfe des aktuellen Kamerabildes liefert. Es stellt sich die Frage, ob man diesen Prozess des manuellen Scharfstellens nicht mit fischertechnik-Mitteln automatisieren kann.

Befestigung eines Zahnrads am Objektiv der Webcam

Die Webcam bietet zunächst keinen Ansatzpunkt um mit Hilfe eines Motors die Fokuseinstellung vorzunehmen. Es fehlt eine offizielle Möglichkeit, zum Beispiel ein Zahnrad am Objektiv zu befestigen.

Man kann sich jedoch mittels eines Streifens Papier oder dünner Pappe behelfen um das Zahnrad Z30 auf dem Objektiv der Webcam festzuklemmen. Der Streifen sollte knapp 1 cm breit und, je nach Dicke,

etwa 20 cm lang sein und wird fest innen in das Zahnrad gewickelt (siehe Abb. 1).



Abb. 1: Z30 mit „Klemmring“

Das Zahnrad sollte sich dann mit dem Pappstreifen über das Objektiv der Kamera schieben lassen. Der Sitz kann einfach über die Länge des Pappstreifens variiert werden und sollte nicht zu fest sein. Es sollte sich auch noch weiterdrehen lassen, wenn das Objektiv am Anschlag ist, da das Objektiv sonst beschädigt werden könnte, falls der Fokus-Motor einmal zu weit drehen sollte.

Als Fokus-Motor dient ein Encodermotor, der mit Hilfe einer Rastschnecke das Z30 antreibt und damit die Schärfenebene der Webcam verändern kann.

Das Modell

Das Kamera-Modell kann komplett mit Teilen aus dem fischertechnik *TXT Discovery Set* aufgebaut werden. Die Teileliste sieht folgendermaßen aus:

Stück	Bezeichnung
1	TXT Controller
1	Webcam
1	Encodermotor
1	Zahnrad Z30
1	Rastschnecke 20 m1,5
2	Zahnrad rot zum Aufstecken
1	Taster
5	BS 15
7	BS 30
1	S-Winkelträger 60
2	BS 15 rot mit Bohrung
1	Verbinder 15
4	Baustein 7,5 rot
2	Baustein 5 rot 15 x 30
2	Baustein 5
1	Rastachse 30
1	Rastachse 45
16	Federnocken
1	fischertechnik Akkupack
1	Pappstreifen 1 x 20 cm

Tab. 1: Teileliste

Da das Modell nicht sehr kompliziert ist, sollte es für geübte fischertechnik-Fans nicht sehr schwierig sein, es anhand der Abbildungen 1 bis 9 nachzubauen. Die mehr oder weniger ästhetische Kabelführung bleibt dabei jedem Nachbauer selbst überlassen. Es sollte allerdings darauf geachtet werden, dass sich keine Kabel, insbesondere nicht das USB-Kabel der Webcam, zwischen den beiden roten Zahnradern verfangen und diese blockieren können.



Abb. 2: Rückseite des TXT mit Taster

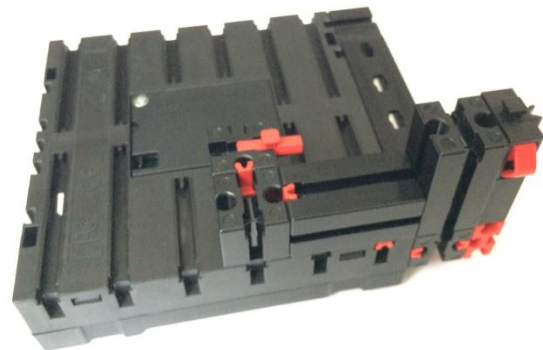


Abb. 3: Befestigung der Fokus-Einheit

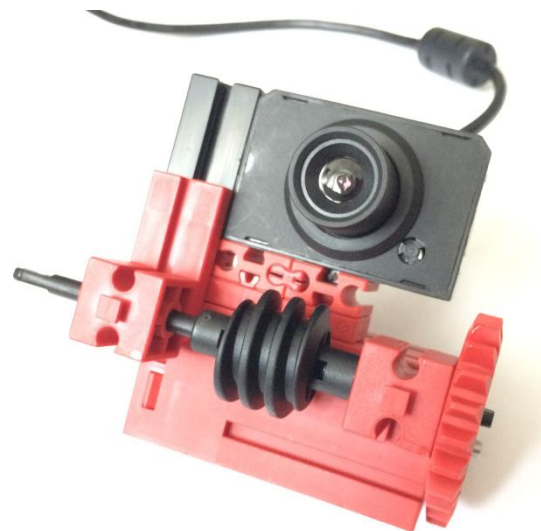


Abb. 4: Fokus-Einheit von vorne

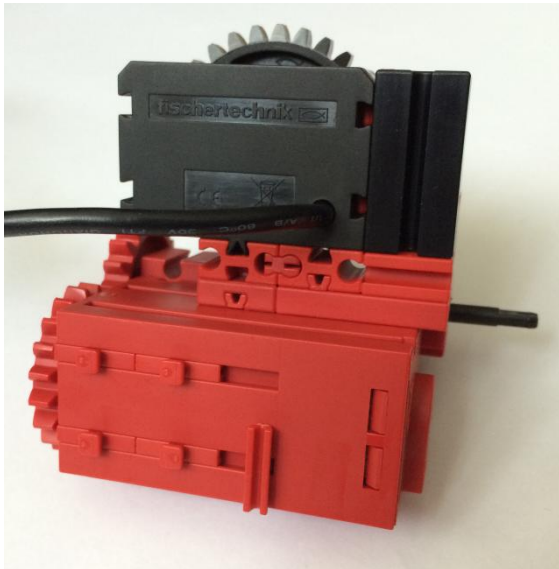


Abb. 5: Fokus-Einheit von hinten

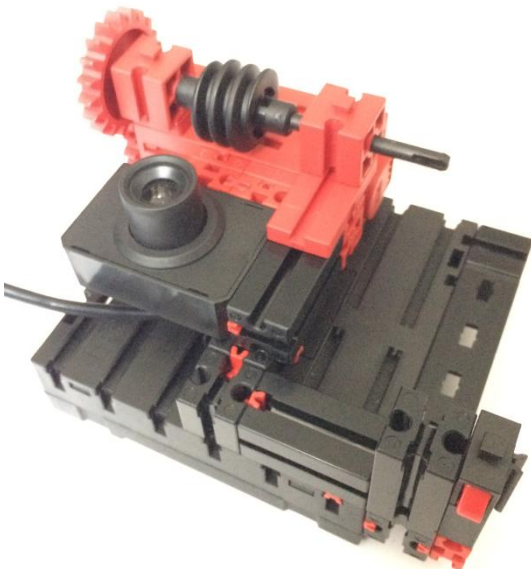


Abb. 6: TXT mit montierter Fokus-Einheit

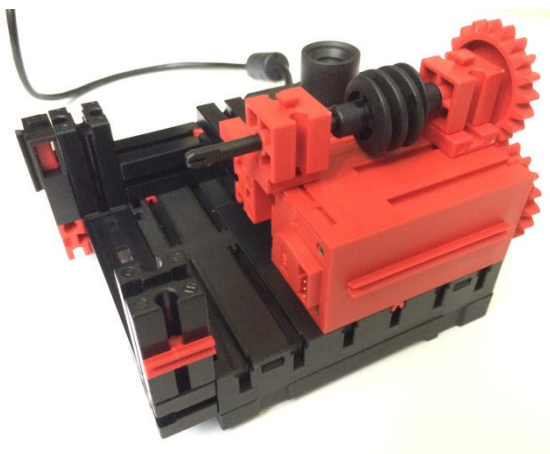


Abb. 7: Befestigung des Akkupacks

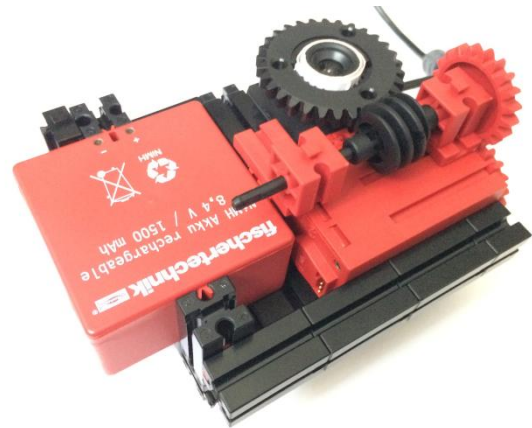


Abb. 8: Kamera mit Akkupack

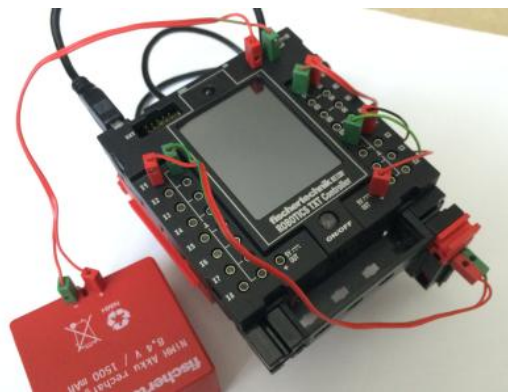


Abb. 9: Verkabelung

Wie wird die Schärfe gemessen?

Selbst preiswerte Digitalkameras bringen so gut wie immer einen brauchbaren Autofokus mit, um dem Verbraucher das Scharfstellen der Bilder zu erleichtern bzw. abzunehmen. Diese einfachen Kameras haben typischerweise keine zusätzlichen Sensoren zur Abstandsmessung eingebaut, sondern ermitteln die optimale Schärfeeinstellung ausschließlich aus der aktuellen Bildinformation. Als Messgröße für die Schärfe verwendet man häufig den durchschnittlichen Kontrast des gesamten Bildes oder eines Bildausschnittes (manche Digitalkameras verwenden auch mehrere verschiedene Bildausschnitte gleichzeitig oder erlauben die Auswahl eines bestimmten Bildausschnittes für die Schärfeeinstellung).

Der Kontrast ist dabei ein Maß dafür, wie stark sich die Helligkeit zwischen benachbarten Pixeln verändert. In unscharfen Bildbereichen ändert sich die Helligkeit nur

allmählich, während sie sich in scharfen Bildbereichen abrupter verändert. Die Farbinformation des Bildes wird dabei nicht unbedingt benötigt, ein schwarz/weiß-Bild reicht für die Kontrastmessung aus.

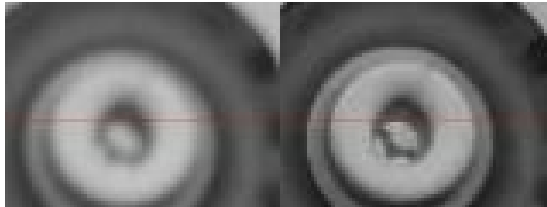


Abb. 10: unscharf / scharf

Die beiden Bilder in Abb. 10 zeigen jeweils einen (nach schwarz/weiß konvertierten) Ausschnitt eines Bildes, das mit der fischer-technik-Webcam aufgenommen wurde. Die Helligkeit der einzelnen Pixel entlang einer horizontalen Linie (als rote Linien in Abb. 10 gekennzeichnet) ist in Abb. 11 grafisch dargestellt. Die absolute Helligkeitsinformation reicht für die Schärfestimmung noch nicht aus. Erst die daraus gewonnene Information der Helligkeitsdifferenz (= Kontrast) zwischen benachbarten Pixeln, die in Abb. 12 grafisch dargestellt ist, lässt Rückschlüsse auf die Schärfe des Bildes zu.

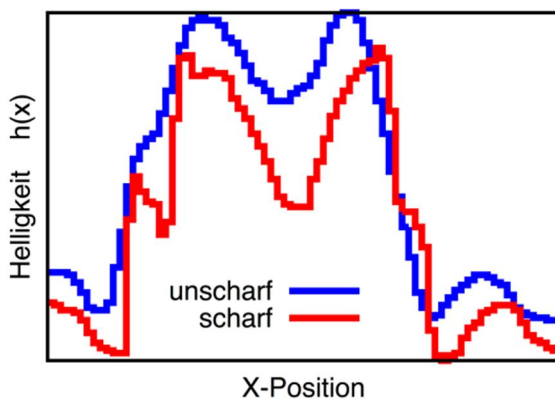


Abb. 11: Helligkeitsverlauf

Um die Unterscheidung zwischen scharfem und unscharfem Bild noch zu verstärken, ist nicht nur die einfache Differenz zwischen benachbarten Punkten aufgetragen, sondern das Quadrat der Differenz:

$$K(x) = (H(x) - H(x + 1))^2$$

Dabei ist K der Kontrast und H der Helligkeitswert, jeweils an der Stelle x , bzw. $x-1$.

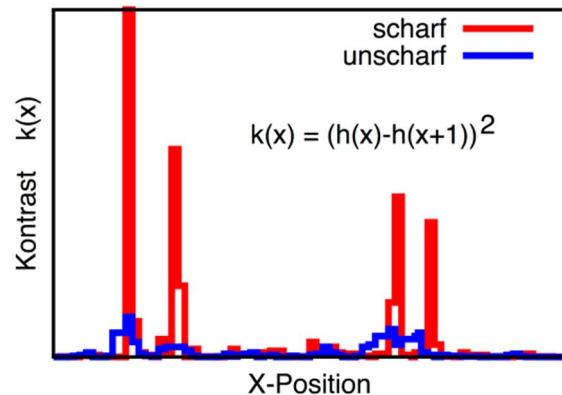


Abb. 12: Kontrastverlauf

In diesem einfachen Beispiel ist der Kontrast nur eindimensional entlang einer Linie aufgetragen. Im tatsächlichen zweidimensionalen Algorithmus zur Kontrastberechnung werden nicht nur die Punkte links und rechts, sondern auch die Punkte oberhalb und unterhalb des aktuellen Punktes durch Differenzbildung mit einbezogen. Der Gesamtkontrast des Bildausschnittes ergibt sich dann als Mittelwert aller Einzelkontraste und kann sehr gut für die Unterscheidung von scharfen und unscharfen Bildeinstellungen herangezogen werden. Der hier beschriebene Algorithmus kommt auch im Programm des fischer-technik-Digitalkamera-Modells zur Anwendung.

Ansteuerung der Webcam und des TXT-Displays

In der aktuellen ROBOPro-Version 4.2.3 kann die Webcam sowohl als Eingabesensor (Linien-erkenner, Baller-erkenner, Farbsensor oder Bewegungserkenner) als auch (im Online-Modus) zur Anzeige eines Live-Kamerabildes auf einem PC- oder Smartphone-Display eingesetzt werden. Eine Live-Anzeige auf dem im TXT eingebauten Display ist mit ROBOPro-Mitteln nicht möglich. Bisher gibt es auch keine Möglichkeit, innerhalb von ROBOPro direkt auf die Kamerabilder zuzugreifen, um eigene

Algorithmen zur Bildverarbeitung zu implementieren oder um die Bilder abzuspeichern. Die oben bereits erwähnte ROBOPro-Balkenanzeige als Hilfe zur manuellen Bildschärfe-Einstellung steht eigenen ROBOPro-Programmen nicht zur Verfügung. Aus diesen Gründen konnte die Software für das Kamera-Modell nicht mit ROBOPro programmiert werden. Stattdessen wurden alle benötigten Funktionen in C und Python implementiert.

Die hardwarenahen Funktionen zur Live-Anzeige auf dem TXT-Display und die rechenintensiven Funktionen zur Kontrast-Messung und JPEG-Dekodierung wurden rein in C implementiert.

Der Quelltext dieser Routinen ist in der Datei `ftrobopytools.c` zu finden [2]. Alle C-Funktionen wurden mit einer Python-Schnittstelle versehen, um die bereits für den TXT vorkompilierte Library `ftrobopytools.so` flexibel in Python-Programmen verwenden zu können.

Zusammen mit dem Python-Modul `ftrobopy.py` [3] zur Ansteuerung der Ein- und Ausgänge des TXTs kann damit die Software des Kamera-Modells übersichtlich in Python programmiert werden. Zu beachten ist, dass das komplette Programm auf dem TXT läuft (vergleichbar mit ROBOPro-Programmen im *Download-Modus*). Aufgrund der Funktionen zur direkten Ansteuerung des TXT-Displays kann das Programm nicht extern auf einem PC laufen (wie ROBOPro-Programme im *Online-Modus*).

Die Routinen in `ftrobopytools.c` zur direkten Ansteuerung der Webcam greifen auf die im TXT-Betriebssystem bereits mitgelieferte Video-for-Linux-Bibliothek V4L2 zu [4]. Die Anzeige des Video-Bildes auf dem TXT-Display erfolgt mit Hilfe der ebenfalls bereits auf dem TXT vorhandenen Standard-Display-Library (SDL) [5].

Da bei der Initialisierung der SDL der aktuelle Bildschirminhalt des TXT-Displays verloren geht, wird jeweils vor dem SDL-Start eine Backup-Kopie des TXT-Display-Inhalts (typischerweise ist dies das Bedienungsmenu des TXTs) über das direkte Auslesen des Linux Framebuffers (`/dev/fb0`) angelegt. Diese Kopie wird vor dem Beenden des Programms wieder in den Framebuffer zurückkopiert. Sollte das Programm unvorhergesehen beendet werden und der originale Displayinhalt des TXT dabei nicht wiederhergestellt werden, kann man den TXT evtl. nicht mehr richtig bedienen, weil das Menu nicht sichtbar ist (die Touch-Buttons des Menus bleiben jedoch nach wie vor aktiv). In diesem Fall hilft das einfache Aus- und wieder Einschalten des TXTs.

Durch den unmittelbaren V4L2-Zugriff auf die Webcam stehen, im Vergleich zum Zugriff über ROBOPro (320 x 240 Pixel), deutlich höhere Bild-Auflösungen zur Verfügung. Vor dem Abspeichern eines Bildes auf die SD-Karte wird die Kamera deshalb auf die maximal mögliche Auflösung von 1280 x 720 Pixeln umgeschaltet. Während der Anzeige des Live-Bildes wird die Kamera-Auflösung auf die TXT-Display-Auflösung 320 x 240 Pixel angepasst.

Das Farbdisplay des TXT hat eine RGB-Farbtiefe von 16 Bit pro Pixel (rot: 5 Bit, grün: 6 Bit, blau: 5 Bit) und kann damit nur 65535 verschiedene Farbtöne darstellen. Die Webcam liefert die Bilder im stark komprimierten JPEG-Format, die vor der Anzeige auf dem Display des TXT in dessen 16-Bit-Format umgewandelt und um 90 Grad gedreht werden. Diese Aufgabe wird aus Gründen der Geschwindigkeit ebenfalls von C-Routinen in `ftrobopytools.c` übernommen.

Das Python-Programm

Das Hauptprogramm zur Steuerung des Digitalkamera-Modells wurde weitestge-

hend als endlicher Automat in Python realisiert (Abb. 13) und findet sich in der Datei `ftDigiCam.py`.

Zur Ansteuerung des am M1-Ausgang und C1-Eingang des TXT angeschlossenen Fokus-Motors und zum Einlesen des am I1-Eingang angeschlossenen Auslöse-Tasters wird das Python-Modul `ftrobopy.py` verwendet [3].

Während der Darstellung des Live-Videobildes wird kontinuierlich der mittlere Kontrastwert bestimmt. Dabei wird nicht das gesamte Bild, sondern aus Geschwindigkeitsgründen nur ein Ausschnitt von 80×60 Pixeln in der Bildmitte zur Kontrastberechnung herangezogen (siehe z. B. Abb. 10).

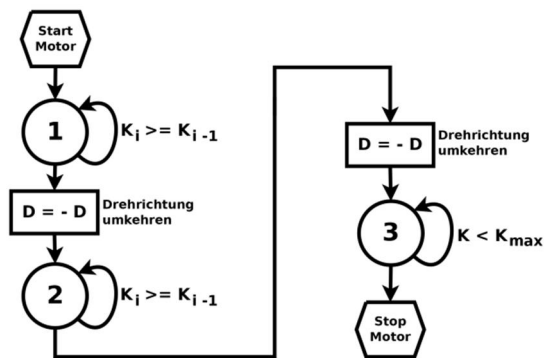


Abb. 13: Endlicher Automat

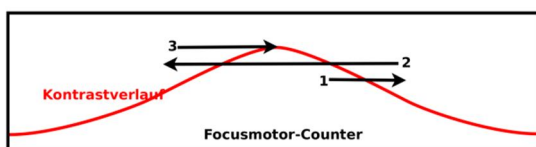


Abb. 14: Suche des maximalen Kontrasts

Der endliche Automat kann die folgenden fünf Zustände einnehmen:

- **Zustand 0:** Warten auf den Auslöse-Taster
- **Zustand 1:** Während der Fokus-Motor sich in die erste Richtung dreht: darauf warten, dass der Kontrast schlechter wird
- **Zustand 2:** Während der Fokus-Motor sich in die andere Richtung dreht: darauf warten, dass der Kontrast schlechter wird

- **Zustand 3:** Während der Fokus-Motor sich wieder in die erste Richtung dreht: darauf warten, dass die Position des in Zustand 1 und 2 ermittelten Kontrast-Maximums erreicht wird
- **Zustand 4:** Umschalten der Kamera-Auflösung auf 1280×720 Pixel, abspeichern des Bildes und zurückschalten der Auflösung auf 320×240 .

Die fünf Zustände werden der Reihe nach durchlaufen. Nach dem letzten Zustand springt der Automat wieder in den Zustand 0. In den Zuständen 1 und 2 wird eine Tabelle (als Python-Liste) aufgebaut, in der die Werte der Kontrast-Messung und die Werte des Zählereingangs C1 des Fokus-Motors abgespeichert werden. Aus dieser Tabelle lassen sich dann das Kontrast-Maximum und damit die beste Schärfereinstellung ermitteln, die im Zustand 3 angefahren wird.

Grundsätzlich gilt, dass die Schärfe umso genauer eingestellt werden kann, je langsamer sich der Fokus-Motor dreht. Man muss also einen Kompromiss aus Geschwindigkeit und Bildschärfe eingehen, wenn man nicht allzu lange auf den Auto-Fokus warten möchte. Die Geschwindigkeit des Fokus-Motors kann im Python-Programm über die Variable `focus_speed` eingestellt werden.

Wie immer beim Fotografieren, sollte man natürlich auch hier auf gute Lichtverhältnisse achten. Eine zu dunkle Umgebung führt zu verstärktem Bildrauschen und verhindert eine glatte Kontrastkurve, die für den hier beschriebenen Algorithmus notwendig ist. Außerdem sollte die Kamera möglichst ruhig gehalten werden, während der Auto-Fokus aktiv ist.

Installation der Python-Software auf dem TXT

Da der TXT ab Werk ohne Python ausgeliefert wird, muss zusätzlich zur Kamera-

Software auch ein Python-Interpreter installiert werden [7]. Das komplette Softwarepaket (Kamera-Software, Python-Interpreter und Installations-Scripte) kann unter [9] als ZIP-Datei heruntergeladen werden und kann sowohl mit Windows und Mac OS X als auch mit Linux auf dem TXT installiert werden.

Auf dem TXT sollte mindestens die Firmware-Version 4.1.5 installiert sein. Nach dem Entpacken der ZIP-Datei muss zuerst die Datei `ftDigiCam.tar` auf den TXT kopiert werden. Unter Linux und Mac OS X geschieht dies einfach über den folgenden Kopierbefehl auf der Kommandozeile (Terminal):

```
scp ftDigiCam.tar ROBOPro@192.168.7.2:
```

(den Doppelpunkt am Ende der Zeile nicht vergessen). Das Passwort lautet: `ROBOPro`.

Es wird hier davon ausgegangen, dass der TXT über das USB-Kabel mit dem PC verbunden ist. Die TAR-Datei landet dann im Verzeichnis `/opt/knobloch` auf dem TXT (dieses Verzeichnis ist das Home-Verzeichnis des `ROBOPro`-Users).

Unter Windows kann die TAR-Datei z.B. mit Hilfe des Programmes WinSCP, das unter [10] heruntergeladen werden kann, auf den TXT kopiert werden. Die WinSCP-Einstellungen können der Abb. 15 entnommen werden. Als Kennwort ist auch hier `ROBOPro` einzutragen.

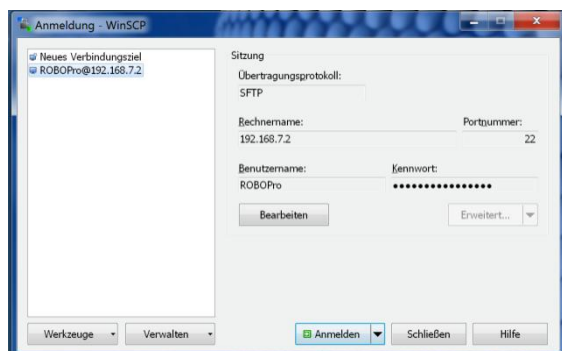


Abb. 15: WinSCP-Einstellungen

Als nächstes muss man sich auf dem TXT per `ssh` einloggen. Unter Linux und Max OS

X geschieht dies wieder auf der Kommandozeile durch Eingabe des Befehls:

```
ssh ROBOPro@192.168.7.2
```

Das Passwort ist: `ROBOPro`.

Unter Windows benötigt man dafür ein SSH-Terminal-Programm. Geeignet (und auch von fischertechnik empfohlen) ist z. B. das Programm *Putty*, welches unter [11] heruntergeladen werden kann. Die Putty-Einstellungen können der Abb. 16 entnommen werden.

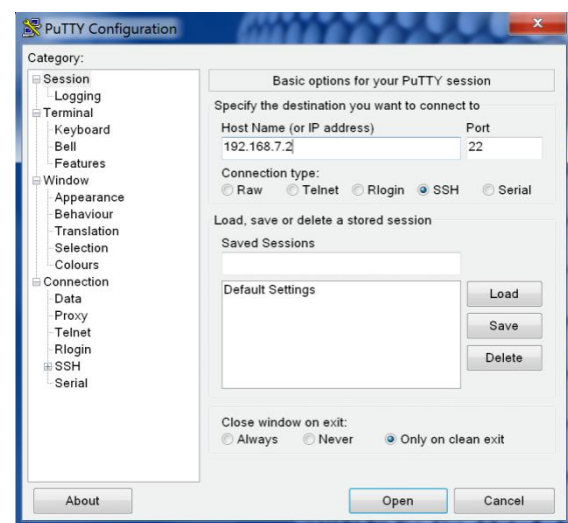


Abb. 16: Einstellungen für Putty

Nach dem Einloggen auf dem TXT wird nun mit dem Kommando:

```
tar -xf ftDigiCam.tar
```

das zuvor auf den TXT kopierte TAR-Archiv entpackt. Dabei werden die folgenden Verzeichnisse im Home-Verzeichnis des Users `ROBOPro` (`/opt/knobloch`) erzeugt:

```
ftrobopy/ (Kamera Software)
lib/      (Python2.7 Libraries)
bin/      (Python Executable)
```

Um Speicherplatz zu sparen kann nach dem Entpacken die TAR-Datei auf dem TXT mit folgendem Befehl wieder gelöscht werden:

```
rm ftDigiCam.tar
```

Im Verzeichnis `ftrobopy/` befindet sich das Python-Programm für die Steuerung

des Digitalkamera-Modells. Außerdem findet man hier das Python-Modul `ftrobopy.py` zur Ansteuerung der Motor-Ausgänge (M1-M4) und zum Einlesen der Eingänge (I1-I8, C1-C4) des TXT. Wer die Möglichkeiten dieses Moduls ausführlich kennenlernen möchte, der sei auf das `ftrobopy-Handbuch` (`manual.pdf`) verwiesen. Beim Modul `ftrobopytools.so` handelt es sich um ein kompiliertes C-Programm, das die schnellen Routinen für das Einlesen des Live Video-Streams und der Bilder und die Berechnung des mittleren Kontrastes für die Schärfe-Bestimmung übernimmt. Außerdem enthält es einen JPEG-Dekodierer (die Bilder der fischertechnik Webcam werden im JPEG-Format an den TXT übermittelt) und die Routinen zur direkten Ansteuerung des TXT-Displays zur Anzeige des Live Video-Streams. Die Funktionen dieses Moduls sind mit einer Python-Schnittstelle ausgestattet und können dadurch bequem auch in eigenen Python-Programmen verwendet werden.

Die Datei `ftrobopytools.c` enthält den kompletten C-Quellcode des Moduls zum Nachlesen, wird aber für die Funktion des Kamera-Modells nicht weiter benötigt.

Unterhalb des Verzeichnisses `ftrobopy/` wird außerdem das Verzeichnis `PICTURES/` angelegt. In diesem Verzeichnis werden die mit der Webcam angefertigten Bilder abgespeichert. Dieses Verzeichnis wird außerdem als Web-Seite exportiert, solange das Kamera-Programm ausgeführt wird. Man hat damit die Möglichkeit, die geschossenen Bilder jederzeit über die URL `http://192.168.7.2:8085` mit einem beliebigen Web-Browser anzuschauen, bzw. vom TXT herunterzuladen (diese Funktion kann durch das Setzen der Variablen `startWebServer=False` in Zeile 38 des Programms `ftDigiCam.py` deaktiviert werden).

Das Kamera-Programm kann jederzeit (nach dem Einloggen mit `ssh`) mit

folgendem Befehl auf dem TXT gestartet werden:

```
bin/python ftrobopy/ftDigiCam.py
```

Das Programm beendet sich automatisch nach 1000 Durchläufen. Diese Zahl kann in der Zeile 106 der Datei `ftDigiCam.py` beliebig verändert werden. Die Ausführung des Programms kann außerdem jederzeit durch Drücken der Tastenkombination `<Strg>-C` abgebrochen werden.

Die Software (und damit das Digital-kamera-Modell) ist auch ohne Verwendung einer Speicherkarte im microSD-Kartenslot des TXT uneingeschränkt verwendbar.

Speichern auf SD-Karte und Autostart

Da die microSD-Karte derzeit noch nicht automatisch beim Einschalten des TXT ins System eingebunden wird, muss diese Funktionalität vom Benutzer vorerst noch selbst aktiviert werden. Dazu ist es (einmalig) notwendig sich als `root` auf dem TXT einzuloggen und eine Systemdatei (`/etc/fstab`) um den Eintrag für das automatische Einbinden der SD-Karte zu erweitern.

Im Verzeichnis `ftrobopy/scripts` werden zu diesem Zweck und für die Einrichtung eines Autostart-Mechanismus für Python-Programme einige Hilfs-Skripte zur Verfügung gestellt (ausführlichere Informationen über die Einbindung der SD-Karte findet man auch unter [6]).

Wer das Root-Passwort seines TXTs noch nicht ermittelt hat, sollte jetzt zunächst den Beitrag von Raphael Jacob [14] lesen – und sich anschließend als `root` auf dem TXT mit `ssh` einloggen. Als User `root` wechselt man nun in das Verzeichnis `ftrobopy/scripts`:

```
cd /opt/knobloch/ftrobopy/scripts  
und führt dort das Shell-Skript zum  
Installieren der SD-Karte aus:
```

```
./install_sdcard.sh
```

Dieses Script verändert die Systemdateien `/etc/fstab`, zur automatischen Einbindung der SD-Karte beim Systemstart, und `/etc/init.d/rc.local` zum automatischen Starten der Kamera-Software nach dem Einschalten des TXT.

Im Detail wird von dem Script jeweils eine Zeile an das Ende der System-Dateien angefügt. An `/etc/fstab` wird die folgende Zeile angefügt:

```
/dev/mmcblk0p1 /mnt auto rw,noatime,user,umask=0022,gid=1000,uid=1000 0 0 1
```

und die folgende Zeile wird an das Ende der Datei `/etc/init.d/rc.local` angefügt:

```
[ -
f '/mnt/autorun.sh' ] && su ROBOPr
o -c screen -A -m -d -
S PYTHON /mnt/autorun.sh'
```

Das Script überprüft vor Veränderung der Dateien anhand der Berechnung einer md5-Prüfsumme, ob die Dateien noch im originalen Auslieferungszustand auf dem TXT vorliegen. Nur in diesem Falle werden die Dateien verändert. Falls die Prüfsummen nicht übereinstimmen, wird davon ausgegangen, dass ein erfahrener Anwender die Dateien bereits selbst verändert hat. Mit dem script `./restore_sysfiles.sh` können die Systemdateien `/etc/fstab` und `/etc/init.d/rc.local` auch wieder in den originalen Auslieferungszustand versetzt werden, falls sie vorher ausschließlich über das Skript `install_sdcard.sh` bearbeitet wurden (auch hier findet wieder ein md5-Prüfsummencheck statt).

Nachdem die Systemdateien angepasst wurden, wird eine SD-Karte im microSD-Slot des TXT beim nächsten Einschalten automatisch in das System miteingebunden (die SD-Karte sollte mit dem VFAT-Dateisystem formatiert sein, was typischerweise bei neu gekauften Karten der Fall ist). Die SD-Karte kann dann über das Verzeichnis `/mnt` auf dem TXT direkt angesprochen werden.

Um den Autostart-Mechanismus für die Kamera-Software zu aktivieren muss das Script `autorun.sh` aus dem Verzeichnis `/opt/knobloch/ftrobopy/scripts` ins Hauptverzeichnis der SD-Karte kopiert werden. Man muss sich dazu nicht mehr als User `root` auf dem TXT einloggen, eine Anmeldung als User `ROBOPro` reicht dafür aus:

```
ssh ROBOPro@192.168.7.2
cd ftrobopy/scripts
cp autorun.sh /mnt
```

Die Datei `autorun.sh` ist eine Shell-Datei, die nach Anpassung der `/etc/init.d/rc.local`-Datei automatisch nach jedem Einschalten des TXT einmal ausgeführt wird. Man kann in dieser Datei auch beliebige andere Prozesse beim Hochfahren des TXT automatisch starten. Das `autorun.sh` Skript startet in der aktuellen Form den Python-Interpreter mit dem `ftDigiCam.py` Programm. Der Parameter `/mnt/PICTURES` sorgt dafür, dass die Bilder auf der SD-Karte abgespeichert werden (der in Zeile 49 von `ftDigiCam.py` gesetzte Pfad `picture_dir=...` wird dann nicht verwendet).

Soll der Autostart beim Hochfahren nicht mehr durchgeführt werden, kann man die Datei `/mnt/autorun.sh` einfach wieder löschen, mit dem Befehl:

```
rm /mnt/autorun.sh
```

Die im `ftDigiCam`-Paket enthaltene Datei `autostart.sh` hat folgenden Inhalt:

```
#!/bin/sh
export
TSLIB_TSDEVICE=/dev/input/event0
export TSLIB_TSEVENTTYPE=INPUT
export TSLIB_CONFFILE=/etc/ts.conf
export
TSLIB_CALIBFILE=/etc/pointercal
export SDL_MOUSEDRV=TSLIB
export
SDL_MOUSEDEV=$TSLIB_TSDEVICE

# pictures will be written into
the directory /mnt/PICTURES

# create the directory if it does
not yet exists
```

```
[ -d "/mnt/PICTURES" ] || mkdir
/mnt/PICTURES

# start the software for the
camera model

cd /opt/knobloch/ftrobopy
/opt/knobloch/bin/python
ftDigiCam.py /mnt/PICTURES
```

Bei Fragen, Anregungen und Problemen rund um das ftDigiCam-Projekt steht auf der ftCommunity-Webseite das Forum unter [13] zur Verfügung.

Quellen und Links

- [1] Xin Xu, Yinglin Wang, Jinshan Tang, Xiaolong Zhang, and Xiaoming Liu: [Robust Automatic Focus Algorithm for Low Contrast Images Using a New Contrast Measure](#). Sensors 2011, Basel.
- [2] [C-Quelltext](#) der Live-Anzeige auf dem TXT-Display, der Funktionen zur Kontrast-Messung und JPEG-Dekodierung
- [3] [Python-Modul](#) zur Ansteuerung der Ein- und Ausgänge des TXTs
- [4] Video-for-Linux-Bibliothek ([V4L2](#))
- [5] Standard-Display-Library ([SDL](#))
- [6] ft:community-Forum: [Einbindung der SD-Karte](#)
- [7] [Python-Interpreter](#) für den TXT
- [8] [ftrobopy](#) (v0.6)
- [9] [ftDigiCam](#) (v0.83)
- [10] [WinSCP](#)
- [11] [Putty](#)
- [12] ft:community-Forum: [Community-Firmware für den TXT](#).
- [13] ft:community-Forum: [ftDigiCam-Projekt](#).
- [14] Raphael Jacob: *TXT Controller – Tipps & Tricks (1): Das Root-Passwort* (in dieser ft:pedia).



Abb. 17: Oster-Stillleben (aufgenommen mit der ft-Autofokus-Kamera)

Computing

I²C mit dem TX(T) – Teil 13: Farbsensor

Dirk Fox

Seit der ft:pedia-Ausgabe 3/2012 [1] stellen wir in loser Folge I²C-Sensoren sowie die zugehörigen Robo Pro-Treiber vor. Inzwischen beherrscht auch der TX-Nachfolger TXT das I²C-Protokoll. Die I²C-Anschlüsse an dessen Erweiterungsport (EXT) arbeiten aber nicht mit den beim TX anliegenden 5 V, sondern mit 3,3 V. In dieser Folge stellen wir daher zwei Farbsensoren vor – einen für den TX, den anderen für den TXT.

Vorbemerkung

Für die Bestimmung der Farbe eines Objekts bietet fischertechnik einen Farbsensor.

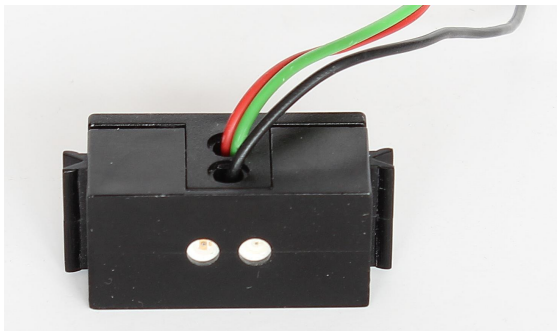


Abb. 1: fischertechnik-Farbsensor

Der Farbsensor verfügt über eine rote LED, deren Licht von verschiedenfarbigen Flächen unterschiedlich stark reflektiert wird. Genau genommen wird vom Sensor also nicht die Farbe erkannt, sondern die Helligkeit der Reflexion. Der Sensor liefert einen analogen (Spannungs-) Wert zwischen 0 und 10.000 mV zurück.

Zur Bestimmung der Farbe werden im Beispielprogramm zum Begleitheft des Baukastens *ROBO TX Explorer* für die Farben rot, grün und blau die folgenden Wertebereiche angegeben:

- rot: 1.075 bis 1.175
- grün: 1.500 bis 1.600
- blau: 1.700 bis 1.800

Das Ergebnis ist allerdings stark abhängig vom Abstand des Sensors von der Farbfläche – und vor allem vom Umgebungslicht. Daher sollten der Sensor möglichst dicht an das Objekt herangeführt und Störlicht abgeschirmt werden.

Eine Alternative zum fischertechnik-Sensor sind RGB-C-Farbsensoren, die die Farbe in die Farbbestandteile Rot, Grün, Blau (und Weiß) zerlegen und deren Anteil bestimmen. Sie sind wesentlich genauer und unabhängiger von Umgebungseinflüssen. Zwei sehr leistungsfähige und über das I²C-Protokoll ansprechbare Sensoren mit integriertem Infrarot-Filter stellen wir im Folgenden vor. Zunächst aber vorab ein paar grundsätzliche Erläuterungen zum Thema Farbe und Licht.

Was ist Licht?

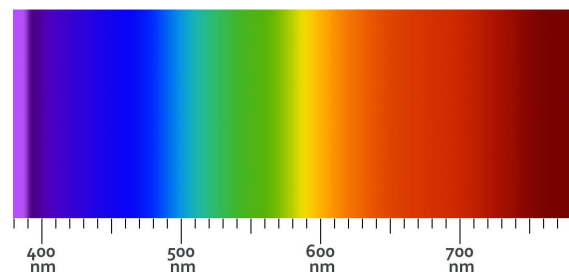


Abb. 2: Spektrum des sichtbaren Lichts

Als Licht bezeichnen wir das Spektrum der für Menschen sichtbaren Wellenlängen elektromagnetischer Strahlung. Es umfasst

die Wellenlängen von 380 nm (violett) bis 780 nm (dunkelrot), siehe Abb. 2. Die maximale Empfindlichkeit des Auges liegt bei 550 nm (gelbgrün). An den Rändern des Spektrums geht das Licht in UV- (Ultraviolett, 100-380 nm) und IR-Strahlung (Infrarot, 780-1000 nm) über.

Weißes Licht ist eine Mischung der Frequenzen aus dem gesamten Spektrum. Aber weiß ist nicht gleich weiß: je nach Intensität der verschiedenen Wellenlängen wirkt das weiß „wärmer“ oder „kälter“. Abb. 3 bis 6 zeigen die Farbanteile von Tageslicht, dem Licht einer Leuchtstofflampe und dem Licht einer (weißen) LED. Man spricht in diesem Zusammenhang auch von der *Lichtfarbe* oder, noch intuitiver, von der *Farbtemperatur*. Sie wird in Kelvin (K) angegeben: Tageslicht erreicht etwa 6500 K, eine weiße LED ca. 3000-4500 K und Kerzenlicht etwa 1000 K.

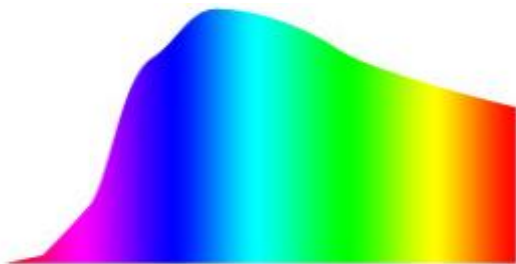


Abb. 3: Spektrum des Tageslichts

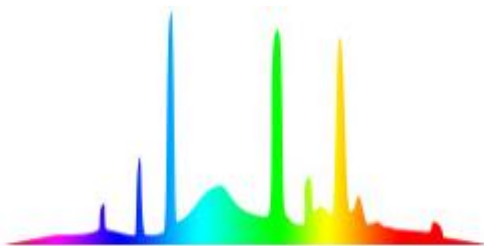


Abb. 4: Spektrum des Lichts einer Leuchtstofflampe



Abb. 5: Spektrum des Glühlampenlichts

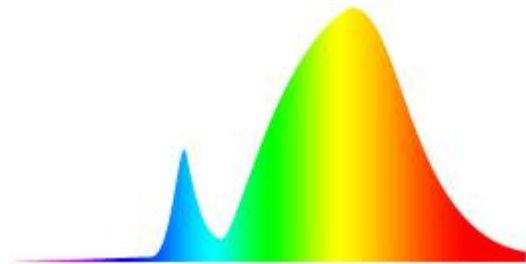


Abb. 6: Spektrum des Lichts einer (warm)weißen LED

Eigentlich nimmt das Auge jedoch gar nicht das gesamte Spektrum wahr, sondern im Wesentlichen die Farben Rot, Grün und Blau, weil es über drei verschiedene, für genau diese drei Farben empfindliche Rezeptoren (Zapfen) verfügt. Licht wird daher bereits dann als „weiß“ empfunden, wenn es – wie bei der Leuchtstofflampe – intensive Rot-, Grün- und Blauanteile besitzt, die die Zapfen anregen.

Was sind Farben?

Wie aber sehen wir Farben? Farben entstehen, wenn wir nur einen Teil des Lichtspektrums sehen – z. B. durch Filterung. Ein Objekt erscheint farbig, wenn seine Oberfläche nur einen ausgewählten Frequenzbereich des sichtbaren Lichts reflektiert. Ein Farbsensor misst also die Frequenzbereiche des von einem Objekt reflektierten Lichts. Dazu ist er oft mit einer eigenen Lichtquelle (LED) ausgestattet, um eine einheitliche Farbtemperatur des auf das Objekt treffenden Lichts sicherzustellen.

Der TCS34725

Von der Firma Taos (Texas Advanced Optoelectronic Solutions Inc., inzwischen Teil von AMS) stammt der I²C-Farbsensor TCS34725, der bei einer Betriebsspannung von 3,3-5 V eingesetzt werden kann – damit ist ein Betrieb sowohl am EXT2-Port des TX als auch am TXT möglich. Der Sensor ist auf einer Platine von Adafruit z. B. bei exp-tech für 7-9 € erhältlich (Abb. 7).

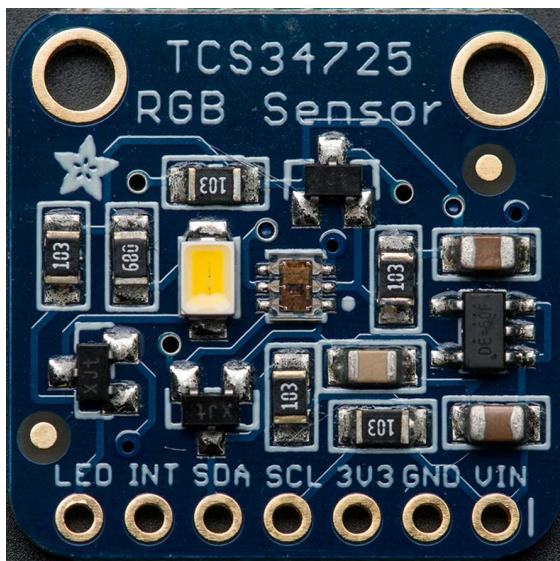


Abb. 7: Farbsensor TCS34725 (Adafruit)

Die Fotodioden zur Farbbestimmung sind mit einem IR-Filter ausgestattet. Die Platine ist außerdem mit einer LED bestückt, mit der das Objekt beleuchtet werden kann. Um den Sensor sowohl mit 3,3 V als auch mit 5 V zu nutzen, sollte der Anschluss `vin` (statt `3v3`) für die Stromzufuhr genutzt werden.

Technische Daten

- *Betriebsspannung:* 3,3-5 V
- *Stromaufnahme:* max. 0,235 mA
- *I²C-Adresse:* 0x29 (fest)
- *I²C-Geschwindigkeit:* 400 kbit/s
- *Auflösung:* 16 bit
- *A/D-Wandlung:* 2,4 bis 612 ms

Die LED des Sensors hat eine Farbtemperatur von 4150 K. Wenn man lediglich die Helligkeit oder die Farbtemperatur einer Lichtquelle messen möchte, kann man die LED abschalten, indem man den Kontakt `LED` (Abb. 7, links unten) mit einem GND-Anschluss des TX(T) verbindet.

Anschluss an den Controller

Für den Anschluss an den TX(T) habe ich in [2, 3] geeignete Verbindungskabel bzw. Adapter vorgestellt.

Inzwischen wurde von der Firma Knobloch [klargestellt](#), dass der 3,3 V-Pin des EXT-Anschlusses am TX – anders als der 5 V-Pin beim EXT2-Anschluss des TX – nicht für die Stromversorgung eines I²C-Sensors ausgelegt ist. Bei dem in [3] vorgestellten Temperatursensor als auch beim Farbsensor TCS34725 gelang das dennoch ohne Probleme, da der Stromverbrauch der Sensoren deutlich unter 1 mA liegt – zumindest dann, wenn die LED deaktiviert wird, indem der Pin `LED` mit einem GND-Anschluss des TX verbunden wird.

Wer auf Nummer sicher gehen oder I²C-Sensoren bzw. vor allem -Aktoren mit meist deutlich höherem Strombedarf anschließen will, sollte beim TX die Stromversorgung über einen Step-Down-Wandler aus dem 9 V-Ausgang des Controllers gewinnen. Mehr dazu in einem späteren Beitrag.

System-Status

Der Sensor kann vier verschiedene Zustände annehmen: Ruhezustand (*Sleep*), *Idle*-Zustand, Warte-Zustand (*Wait*) und Auswerte-Zustand (*RGBC*), siehe Abb. 8.

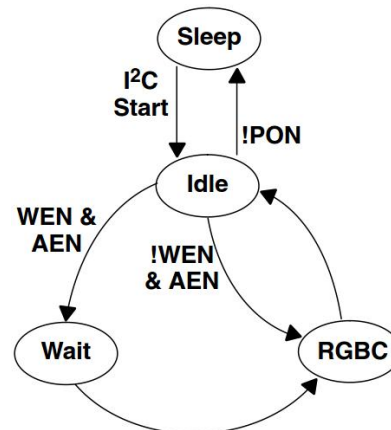


Abb. 8: Zustandsdiagramm des TCS34725

Nach dem Einschalten der Stromversorgung befindet sich der Sensor im *Sleep*-Zustand. Durch I²C-Befehle, die die in der Abbildung angegebenen Status-Flags setzen, kann er in die anderen Zustände überführt werden:

- In den *Idle*-Zustand geht er über, wenn *Power On* gewählt ist (PON-Flag).
- Der *Wait*-Zustand wird eingenommen, wenn der Energiesparmodus des Sensors (WEN-Flag) ausgewählt wird.
- Im *Wait*-Zustand verbraucht der Sensor nur 0,065 mA, gegenüber 0,235 mA im *RGBC*-Zustand.

Die Zeit, die der Sensor im *Wait*-Zustand verbleibt, bevor er automatisch in den *RGBC*-Zustand übergeht, kann in *WTIME* eingestellt werden. Ist außerdem das *WLONG*-Flag gesetzt, verlängert sich die eingestellte Wartezeit auf das 12-Fache. Bei maximaler Einstellung (Messung nur alle 7,37 s) sinkt der durchschnittliche Energieverbrauch auf 0,067 mA.

Tatsächlich dürfte die Energieersparnis für TX(T)-Anwendungen in der Regel nicht relevant sein. Dennoch wird die Nutzung dieser Option im Robo Pro-Treiber berücksichtigt. Außerdem unterstützt der Sensor Interrupts, die an einem eigenen Ausgang (*INT*, Abb. 6) abgefragt werden können. Für die Interrupt-Auslösung können eine untere und eine obere Schwelle für den Weiß-Sensorwert definiert werden. Wurde die Interrupt-Signalisierung aktiviert, wird das Anliegen eines Interrupts außerdem über ein Status-Bit angezeigt.

Command-Befehl

Ein Command-(Byte-)Befehl ist an einem gesetzten Bit 7 (0x80) erkennbar, gefolgt von der Selektion des Lese- oder Schreibmodus in Bit 5 und 6:

- *Repeated Byte* (0b00, es wird immer dasselbe Byte-Register ausgelesen bzw. geschrieben) oder
- *Auto Increment* (0b01, der Adresszähler wird nach jedem gelesenen bzw. geschriebenen Byte automatisch hochgezählt).

In den Bits 0-4 steht die Adresse des Registers, das nachfolgend (zuerst) gelesen bzw. beschrieben werden soll.

Mit der *Special Function* (0b11 in Bit 5/6) wird eine im *Status Register* (0x13) gesetzte Interrupt-Signalisierung wieder gelöscht (0b11100110 = 0xE6).

Register

Der TCS34725 verfügt über 20 zugängliche Byte-Datenregister (Tab. 1).

Register	Bedeutung
0x00	<i>Enable Register</i>
0x01	<i>RGBC Timing Register</i>
0x03	<i>Wait Time Register</i>
0x04-07	<i>Interrupt Threshold</i>
0x0C	<i>Persistence Register</i>
0x0D	<i>Configuration Register</i>
0x0F	<i>Control Register</i>
0x12	<i>ID Register</i>
0x13	<i>Status Register</i>
0x14-1B	<i>RGBC Channel Data</i>

Tab. 1: Register des TCS34725 [4]

Enable Register

Das *Enable Register* (0x00) dient zum Aktivieren und Deaktivieren des Sensors und einzelner Funktionen:

- **Bit 0:** „Power On“ (PON); wenn gesetzt wird der Sensor aus dem *Sleep*-Modus geweckt und geht in den *Idle*-Zustand über.
- **Bit 1:** „ADC Enable“ (AEN) aktiviert den A/D-Wandler zur Bestimmung der *RGBC*-Werte; der Sensor geht in den Zustand *RGBC* über. Vor der ersten Messung muss ein *Warm-up Delay* von 2,4 ms abgewartet werden.
- **Bit 3:** Aktiviert die Energiesparfunktion (WEN) über den *Wait*-Zustand. Die Dauer des Warte-Zyklus (*WTIME*) wird durch das *Wait Time Register* (0x03) und

das WLONG-Flag im *Configuration Register* (0x0D) festgelegt.

- **Bit 4:** Ist dieses Bit (AIEN) gesetzt, ist die Signalisierung eines Interrupts im *Status Register* (0x13) aktiviert.

RGBC Timing Register

Im *RGBC Timing Register* (0x01) kann die Integrationszeit (ATIME) der Analog-Digital-Wandlung (ADC) auf einen Wert zwischen 2,4 ms (0xFF, ein Integrationszyklus) und ca. 614 ms (0x00, 256 Integrationszyklen) festgelegt werden; sie wird als Zweierkomplement gespeichert.

Je länger die Integrationszeit, desto höher ist die Auflösung des Sensors und die Genauigkeit der Messung. Die Default-Einstellung ist 0xFF (ein Integrationszyklus).

Wait Time Register

Im *Wait Time Register* (0x03) wird die Länge der Wartezeit vor einer Messung (WTIME) eingestellt. Die Einstellung ist nur erforderlich, wenn die Energiesparfunktion mit dem WEN-Flag im *Enable Register* (0x01) gewählt ist. Sie wird als Byte-Zweierkomplement eingetragen und kann zwischen 2,4 ms (Wert 0xFF, default) und ca. 614 ms (Wert 0x00) gewählt werden.

Ist außerdem das WLONG-Flag im *Configuration Register* (0x0D) gesetzt, multipliziert sich die Wartezeit mit 12 (28,8 ms bis 7,37 s).

RGBC Interrupt Threshold

Für die Auslösung eines Interrupts (bzw. dessen Signalisierung im *Status Register*) können in den *RGBC Interrupt Threshold Registern* 0x04 bis 0x07 ein unterer (0x04-0x05) und ein oberer (0x06-0x07) 16-bit-Schwellenwert für den Weiß-Sensor eingetragen werden – niederstwertiges Byte (LSB) zuerst.

Liegt der vom Weiß-Sensor gemessene Wert unter dem unteren oder über dem oberen Schwellenwert, wird ein Interrupt

ausgelöst. Wie viele Messwerte hintereinander diese Bedingung erfüllen müssen, wird im *Persistence Register* (0x0C) eingestellt.

Persistence Register

Im *Persistence Register* (0x0C) wird die Zahl der Messwerte eingestellt, die in Folge die gewählten Schwellenwerte unter- bzw. überschreiten müssen, bevor ein Interrupt ausgelöst wird. Wählbar sind diskrete Werte zwischen eins (0x01) und 60 (0x0F); ab fünf (0x04) in Fünferschritten.

Ist der Wert 0x00 eingestellt, veranlasst jede RGBC-Bestimmung einen Interrupt.

Mit der Integrationszeit (ATIME) und der Wartezeit (WTIME) kann so auch eine Zeitspanne festgelegt werden, die ein Schwellenwert maximal unter- oder überschritten werden darf, bevor ein Interrupt erfolgt.

Configuration Register

Im *Configuration Register* (0x0D) kann lediglich das WLONG-Flag durch das Setzen von Bit 1 aktiviert werden. Damit wird die im *Wait Time Register* (0x03) festgelegte Dauer WTIME des Wartezyklus vor einer Messung um den Faktor 12 verlängert. Die Default-Einstellung ist 0x00.

Control Register

In den niederstwertigen beiden Bits vom *Control Register* (0x0F) wird der Verstärkungsfaktor (*Gain*) für die RGB-Fotodioden eingestellt. Wählbar sind:

- 0b00: einfach (default)
- 0b01: vierfache Verstärkung
- 0b10: 16fache Verstärkung
- 0b11: 60fache Verstärkung

Alle anderen Bits (2-7) sind ungenutzt.

ID Register

Beim TCS34725 ist im *ID Register* (0x12) der Wert 0x44 (= 68) fest eingetragen. Das Register kann daher zur Prüfung verwendet

werden, ob der (richtige) Sensor abgeschlossen ist.

Status Register

Das *Status Register* (0x13) zeigt an, wenn die RGBC-Bestimmung abgeschlossen ist und der Wert ausgelesen werden kann:

- **Bit 0:** Der Berechnungszyklus einer RGBC-Bestimmung ist abgeschlossen; die RGBC-Werte können in den Datenregistern gelesen werden.
- **Bit 4:** Ein Interrupt liegt vor, d. h. – je nach Einstellung im Persistence Register – erfolgte eine (mehrfache) Unter- oder Überschreitung der in den *Threshold Registern* eingestellten Schwellenwerte.

Die Abfrage des Status-Registers vermeidet einen Wartezyklus über die Dauer der Analog-Digital-Wandlung (ADC), die sich aus dem Wert des RGBC-Timing-Registers berechnen lässt.

RGBC Channel Data Registers

Die gemessenen RGBC-Werte werden als vier 16-bit-Werte in den Registern 0x14 bis 0x1B (mit dem niederwertigen Byte zuerst) bereitgestellt. Sie verteilen sich wie folgt:

- Weiß: 0x14-0x15
- Rot: 0x16-0x17
- Grün: 0x18-0x19
- Blau: 0x1A-0x1B

Da in jeder Oberflächenreflexion alle drei Farbanteile enthalten sind, ist der Farbanteil maßgeblich, der den größten Wert aufweist. Rote, grüne und blaue Oberflächen lassen sich so eindeutig identifizieren. Um möglichst unabhängig vom Umgebungslicht zu sein (wenn man die LED bei der Messung nicht nutzt), sollte man den Mittelwert der RGB-Werte bestimmen und für jede Farbe die (relative) Abweichung von diesem Wert.

Farbtemperatur und Helligkeit

Mit den vom Sensor gemessenen Werten lässt sich aber nicht nur die „gesehene“ Farbe identifizieren, sondern auch die Temperatur des (weißen) Lichts (in Kelvin) sowie die Helligkeit (in Lux) bestimmen.

Wie das geht ist in [5] beschrieben: Die Helligkeit H berechnet sich aus den (um IR-Anteile bereinigten) RGB-Werten (R' , G' und B') nach der folgenden Formel:

$$H = (0,136 \cdot R' + G' - 0,444 \cdot B') \cdot 129,17$$

Der trotz IR-Filter verbleibende IR-Anteil an den Messwerten kann wie folgt näherungsweise bestimmt und von den RGB-Werten R , G und B abgezogen werden:

$$IR = (R + G + B - C)/2$$

Der Parameter 129,17 ist passend zu einem ATIME-Wert von 2,4 ms und einem Verstärkungsfaktor (*Gain*) von eins gewählt (Default-Einstellungen).

Den Temperaturwert gewinnen wir nach der Gleichung von *C. McCamy* [5]:

$$T = 3810 \cdot \frac{B'}{R'} + 1391$$

Auch hier muss mit den IR-bereinigten RGB-Werten gerechnet werden.

Robo Pro-Treiber

Im [Downloadbereich der ft:c](#) findet ihr den Robo Pro-Treiber für den TCS34725 zum Herunterladen.

Er umfasst die folgenden Funktionen:

- Mit der Funktion `TCS34725_Init` wird der Sensor aus dem *Sleep*- in den *Idle*-Zustand versetzt und die RGBC-Erkennung aktiviert. Dabei wird die ID des Sensors geprüft (`TCS34725_GetID`); ist sie fehlerhaft, wird ein Fehler (NC) zurückgeliefert.
- Die Funktion `TCS34725_GetID` liefert die ID des Sensors (Register 0x12)

zurück; der Rückgabewert muss gleich 0x44 sein.

- Mit der Funktion `TCS34725_On` wird der Sensor aus dem *Sleep*- in den RGBC-Zustand versetzt; die komplementäre Funktion `TCS34725_Off` versetzt den Sensor in den *Sleep*-Zustand (*Power Off*).
- Die Funktion `TCS34725_GetRGBC` pollt den Status (Register 0x13), bis das Ergebnis einer RGBC-Messung vorliegt, und liefert die digitalen 16-Bit-RGBC-Werte *Clear*, *Red*, *Green* und *Blue* zurück.
- Mit der Funktion `TCS34725_SetGain` kann die Verstärkung des Sensors eingestellt werden; übergeben wird ein Wert von 0 (default) bis 3 (16fache Verstärkung).
- Mit der Funktion `TCS34725_SetITime` wird die Zahl der Integrationszyklen zwischen 1 (default) und 256 festgelegt.
- Der untere und der obere 16-bit-Schwellenwert für den Interrupt (*Clear Channel*) werden mit der Funktion `TCS34725_SetThresholds` eingestellt.
- Mit der folgenden Funktion wird festgelegt, wie viele aufeinander folgende Werte die Schwellenwerte übersteigen müssen, damit ein Interrupt ausgelöst wird: `TCS34725_SetPersistence` (0: Interrupt nach jeder Konvertierung, 15: 60 Werte außerhalb Schwellenwerte).
- Mit den Funktionen `TCS34725_CalcTemperature` und `TCS34725_CalcLux` wird aus den übergebenen RGB-Werten die Farbtemperatur (in Kelvin) und die Helligkeit (in Lux) berechnet.
- Die Funktion `TCS34725_CalcHue` berechnet aus den RGB-Werten den Farbton, bestimmt die Farbbrillanz und die Farbsättigung (siehe Abschnitt *Farbbe-stimmung*).

Der ISL29125

Von Intersil stammt ein recht junger Farbsensor, der ISL29125 [6]. Er arbeitet mit einer Betriebsspannung von 3,3 V und wird von [Sparkfun](#) auf einem Breadboard angeboten, das u. a. bei [exp-tech](#) für unter 9 € erhältlich ist (Abb. 9).

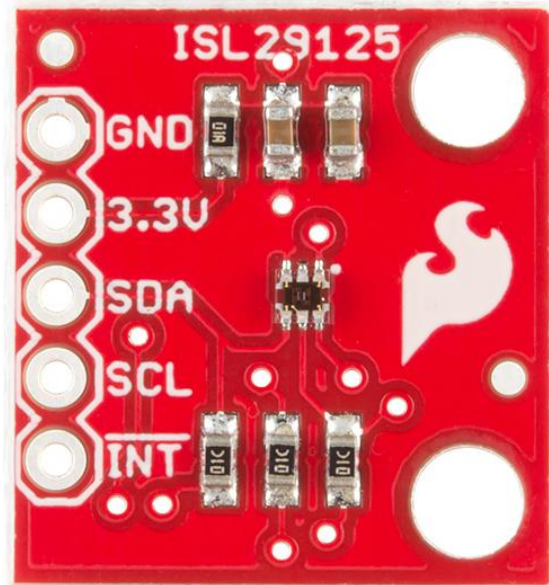


Abb. 9: ISL29125 (Sparkfun)

Technische Daten

- *Betriebsspannung*: 3,6 V
- *Stromaufnahme*: max. 0,085 mA
- *I²C-Adresse*: 0x44 (fest)
- *I²C-Geschwindigkeit*: bis 500 kbit/s
- *Auflösung*: 12-16 bit
- *Messbereich*: 0,0057-275 Lux (Bereich 0) bzw. 0,152-10.000 Lux (Bereich 1)
- *A/D-Wandlung*: 101 ms

Anschluss an den Controller

Der Farbsensor kann mit dem in [3] vorgestellten Adapter direkt am EXT-Ausgang des TXT Controllers angeschlossen werden. Da der Stromverbrauch deutlich unter 0,1 mA liegt, ist keine Überlastung des Anschlusses zu befürchten.

Der Betrieb am TX Controller, dessen I²C-Bus mit 5 V arbeitet, erfordert hingegen die Verwendung eines Step-Down-Spannungswandlers und eines Level-Shifters für die I²C-Signale. In einem späteren Beitrag werden wir dafür Lösungen vorstellen.

System-Status

Die A/D-Wandlung der Sensorwerte ist nach dem Einschalten und einem *Reset* (Rücksetzen der Registerinstellungen auf die Default-Werte) zunächst deaktiviert (*Power Down*). In diesem Zustand liegt der Stromverbrauch unter 0,5 μ A.

Zur Stromeinsparung kann der Sensor im Betrieb in den *Standby Mode* versetzt werden (keine A/D-Wandlung); der Stromverbrauch sinkt dabei auf maximal 37 μ A. Da der Verbrauch im Betrieb nach Datenblatt 85 μ A nicht übersteigt, lohnt das üblicherweise nicht.

Die Dauer einer A/D-Wandlung kann nicht beeinflusst werden; sie liegt fest bei 101 ms.

Register

Der ISL29125 verfügt über insgesamt 15 Datenregister, die über das I²C-Protokoll direkt adressiert und ausgelesen bzw. beschrieben werden können (siehe Tab. 2).

Register	Bedeutung
0x00	<i>Device ID / Reset</i>
0x01	<i>Configuration 1</i>
0x02	<i>Configuration 2</i>
0x03	<i>Configuration 3</i>
0x04-05	<i>Low Threshold, LSB first</i>
0x06-07	<i>High Threshold, LSB first</i>
0x08	<i>Status Register</i>
0x09-0A	<i>Green Data</i>
0x0B-0C	<i>Red Data</i>
0x0D-0E	<i>Blue Data</i>

Tab. 2: Register des ISL29125 [6]

Device ID/Reset Register

Das Register 0x00 enthält die *Device ID* des Sensors, den Wert 0x7D. Durch Auslesen des Registerinhalts kann überprüft werden, ob der (richtige) Sensor angeschlossen ist.

Wird das Device Register mit dem Wert 0x46 beschrieben, werden alle Registerinträge auf die Default-Werte zurückgesetzt (*Device Reset*).

Konfigurations-Register

Die Register 0x01-0x03 enthalten alle wesentlichen Einstellungen für die Arbeit des Farbsensors.

Im *Configuration Register 1* (0x01) können die folgenden Einstellungen vorgenommen werden:

- **Bit 0-2:** RGB Operating Mode; hier wird zwischen Power Down (default, 0b000), nur grün (0b001), nur rot (0b010), nur blau (0b011), Standby (0b100), rot/grün/blau (0b101), grün/blau (0b110) und grün/blau (0b111) gewählt werden.
- **Bit 3:** Sensing Range – Auswahl des Messbereichs (bis 375 Lux oder bis 10.000 Lux)
- **Bit 4:** Resolution – hier kann zwischen einer 16-bit-Auflösung (0) und einer 12-bit-Auflösung (1) gewählt werden.

Das *Configuration Register 2* (0x02) erlaubt, zusätzlich zu den passiven IR-Filtern in den Dioden, die Einstellung einer aktiven IR-Kompensation. Die Anpassung dieses Registers ist allerdings nur für sehr spezielle Anwendungen zu empfehlen (und wurde im RoboPro-Treiber daher nicht berücksichtigt).

Das *Configuration Register 3* (0x03) enthält alle Einstellungen zur Interrupt-Steuerung:

- **Bit 0-1:** Interrupt-Zuweisung; hier können Interrupts deaktiviert (default, 0b00) oder für die Farben Grün (0b01), Rot (0b10) oder Blau (0b11) aktiviert werden.

- **Bit 2-3:** Zahl der Integrationszyklen, nach denen ein Interrupt ausgelöst wird: 1 (default, 0b00), 2 (0b01), 4 (0b10) oder 8 (0b11).
- **Bit 4:** Aktivierung des INT-Pins (default: 0).

Schwellenwerte

Die 16-bit-Schwellenwerte für das Auslösen eines Interrupts werden in den Registern 0x04-0x07 gesetzt. Per Default sind die Schwellenwerte auf 0x00 (untere Grenze) und 0xFF (obere Grenze) gesetzt.

Status Register

Das Register 0x08 enthält mehrere Flag-Bits, die den Status der A/D-Wandlung anzeigen:

- **Bit 0:** Interrupt-Status-Bit; es ist gesetzt, wenn die Schwellenwerte des Interrupts überschritten wurden. Das Bit wird erst nach dem Auslesen zurückgesetzt.
- **Bit 1:** Das Konvertierungs-Status-Bit wird gesetzt, sobald eine A/D-Wandlung abgeschlossen ist.
- **Bit 2:** Das „Brownout“-Bit zeigt eine Störung an. Es ist nach dem Starten des Sensors gesetzt (0x04) und muss daher zunächst mit einem Schreibkommando auf 0x00 zurückgesetzt werden.
- **Bit 4-5:** Anzeigebits für die Sensorwerte (0b01 = grün, 0b10 = rot, 0b11 = blau), die gerade konvertiert werden.

Data Register

Die Register 0x09-0x0E enthalten die 12- bzw. 16-bit-Werte der Analog-Digital-Wandlung der Grün-, Rot- und Blau-Sensoren. Die Werte werden nach jeder A/D-Wandlung aktualisiert. Beim Einschalten enthalten die Register den Wert 0x00.

Robo Pro-Treiber

Im [Downloadbereich der ft:c](#) findet ihr den Robo Pro-Treiber für den ISL29125 zum Herunterladen.

Er umfasst die folgenden Funktionen:

- Die Funktion `ISL29125_Init` setzt alle Registerwerte auf die Default-Einträge und prüft die Device ID des Sensors (`ISL29125_GetID`); ist sie fehlerhaft, wird ein Fehler (NC) zurückgeliefert. Anderenfalls wird das Brownout-Bit im Status-Register gelöscht und die A/D-Wandlung für alle Farbsensoren aktiviert (`ISL29125_SetRGBOperation`).
- Die Funktion `ISL29125_GetID` liefert die Device ID des Sensors (Register 0x00) zurück; der Rückgabewert muss gleich 0x7D sein.
- Mit der Funktion `ISL29125_GetRGB` wird zunächst das Status-Register gepollt, bis die nächste A/D-Wandlung abgeschlossen ist; dann werden die digitalen 16-bit-Messwerte der Farbsensoren für *Green*, *Red* und *Blue* ausgelesen.
- Die Einstellung des *RGB Operating Mode* erfolgt mit der Funktion `ISL29125_SetRGBOperation`: Power down (0), nur grün (1), nur rot (2) nur blau (3), Standby (4), RGB (5), grün/rot (6) oder grün/blau (7).
- Der Messbereich wird mit der Funktion `ISL29125_SetSensingRange` ausgewählt (0: bis 375 Lux, 1: bis 10.000 Lux).
- Die Sensor-Auflösung kann mit der Funktion `ISL29125_SetResolution` geändert werden (0: 16 bit, 1: 12 bit).
- Zur Aktivierung der Interrupts wählt die Funktion `ISL29125_SetInterrupt` den Interrupt-Typ (0: keiner, 1: bei grün, 2: bei rot, 3: bei blau) und die Zahl der A/D-Wandlungen vor der Auslösung eines Interrupts (*Persistence*; 0: 1, 1: 2, 2: 4, 3: 8).

- Die 16-bit-Schwellenwerte für die Interrupt-Auslösung werden mit der Funktion `ISL29125_SetThresholds` festgelegt.
- Mit der Funktion `ISL291255_CalcHue` bestimmt man den Farbton, die Farbbrillanz und die Farbsättigung (siehe Abschnitt *Farbbestimmung*).

Vergleich der Sensoren

Der TAOS-Sensor ist zweifellos der leistungsfähigere: Er kann die Sensor-Messwerte verstärken, ermöglicht eine schnellere A/D-Wandlung, liefert den Wert des Weißlichts und erlaubt eine automatische Mittelwertbildung aus einer größeren Zahl von Messungen. Außerdem verfügt er über eine eigene LED mit einer dem Tageslicht ähnlichen Farbtemperatur von 4150 K.

Dafür punktet der Intersil-Sensor mit sehr niedrigem Stromverbrauch und einer einfacheren Schnittstelle. Allerdings müssen die Objekte sehr nah an den Sensor gehalten werden, um vernünftige Werte zu erhalten. Offenbar ist er auch anfälliger für Störlicht (schließlich verfügt er auch über keine eigene Lichtquelle wie der TAOS-Sensor). Schließlich werden die Messwerte schnell größer 32.767 – und werden damit in ROBO Pro negativ; da hilft nur eine Konvertierung in Fließkommazahlen.

Farbbestimmung

Üblicherweise wird eine Farbe durch den *Farbton*, die *Farbsättigung* und die *Farbbrillanz* (Helligkeit) beschrieben [7].

Die *Farbbrillanz* ist der größte Farbwert (das Maximum von R, G, B), und die *Farbsättigung* der Quotient aus der Helligkeitsdistanz und der Brillanz:

$$\text{Sat} = \frac{\text{Max}(R, G, B) - \text{Min}(R, G, B)}{\text{Max}(R, G, B)}$$

Die Bestimmung des *Farbtone*s ist etwas komplizierter und nicht als geschlossene Formel darstellbar: Er ist die Differenz der beiden kleineren Farbwerte geteilt durch die Helligkeitsdistanz. Falls $\text{Max}(R, G, B) = G$

wird zwei, falls $\text{Max}(R, G, B) = B$ wird vier hinzuaddiert; schließlich wird das Ergebnis durch sechs geteilt. Ist es kleiner 0, wird 1 hinzuaddiert [7].

Der Farbton hat die schöne Eigenschaft, auch bei Änderung des Umgebungslichts konstant zu bleiben. Somit lassen sich Farben sehr zuverlässig erkennen, indem man zunächst in einer Kalibrierungsphase die zu erkennenden Farbtöne „anlernt“ (tabelliert) und dann nach einer Farberkennung den Farbton mit den Werten dieser erlernten Farbtontabelle vergleicht.

Die folgende Grafik (Abb. 10) zeigt eine grobe Zuordnung der Farbtöne zu den Farbtonwerten:

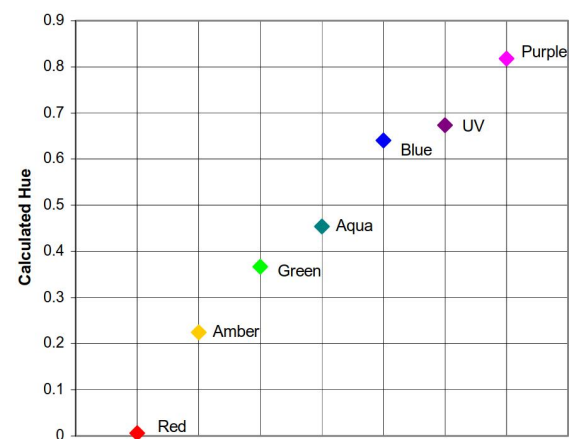


Abb. 10: Farbtöne [7]

Man sieht leicht, dass diese Farberkennung umso besser funktioniert, je größer die Farbsättigung der zu erkennenden Farbtöne ist – oder, anders ausgedrückt: je weiter der kleinste und der größte Farbwert auseinanderliegen.

Die Implementierung einer geeigneten Kalibrierung und der Farbtonerkennung überlasse ich euch als erste kleine Fingerübung mit den neuen Sensoren...

Referenzen

- [1] Dirk Fox: *I²C mit TX und Robo Pro – Teil 1: Grundlagen*. [ft.pedia 3/2012](http://ft.pedia.com/3/2012), S. 32-37.

- [2] Dirk Fox: *I²C mit dem TX – Teil 7: Real Time Clock (RTC)*. [ft:pedia 4/2013](#), S. 28-34.
- [3] Dirk Fox: *I²C mit dem TX(T) – Teil 12: Temperatursensor*. [ft:pedia 4/2015](#), S. 44-48.
- [4] ams (TAOS): *TCS3472 Color Light to digital Converter with IR Filter*. Datasheet, TAOS135, August 2012.
- [5] H. Burton: *Lux and CCT Calculations using ams Color Sensors*. Application Note DN 40, v1.0, ams, 26.08.2013.
- [6] Intersil: *Digital Red, Green and Blue Color Light Sensor with IR Blocking Filter*. Datasheet ISL29125, 15.01.2016.
- [7] Jack Berlien: *Color Classification with the TCS230: Identifying and Sorting Colors by Hue*. Designers Notebook Number 11, TAOS, 03.04.2004.

3D-Drucker von fischertechnik

