

Editorial

Überraschung

Wie jedes Jahr ist am 24. Dezember – ganz überraschend – Weihnachten. Dabei sollten wenigstens die Adventskerzen die Überraschung ein wenig mildern: Spätestens wenn die vierte Kerze stark heruntergebrannt ist, kann es eng werden mit den Weihnachtseinkäufen.

Vor 42 Jahren half auch fischertechnik, die Überraschung zu mildern: durch eine „Wunschzettel“-Einlage im Clubheft. Ein beidseitig eng bedrucktes DIN-A5-Blatt mit einer Liste aller damals erhältlichen 85 (!) fischertechnik-Produkte – von Baukästen für das Vorschulalter (mit den missverständlichen Titeln „25 V“, „50 V“, „100 V“, „1000 V“) über Motor und Getriebe, Statik, Elektromechanik und Elektronik bis zu den Hobby-Bänden.

Es kam, was kommen musste: Wir kreuzten alle 85 Kästchen an und überreichten die Übersicht unseren Eltern – zur freundlichen Weiterleitung an den zuständigen Weihnachtsmann.

Zwar kam für uns dadurch Weihnachten nicht ganz überraschend, wohl aber unsere Päckchen – denn zu unserer großen Verwunderung waren es deutlich weniger als 85. Immerhin steckte in dem einen oder anderen ein fischertechnik-Baukasten – und die Teile daraus bereichern bis heute unsere Materialsammlung. Intensiv bespielt und doch, nach über 40 Jahren, noch voll funktionsfähig – Polwendeschalter, Minimots, Silberlinge, Elektromagneten und 6 V-Motoren inklusive.

Dirk Fox, Stefan Falk

Das bringt uns zu einer anderen Liste – die Liste der Spielzeuge, die die Kindheit überleben. Keine Frage: fischertechnik ist wahrscheinlich das Spielzeug mit der höchsten „Speicherquote“ – dem Verhältnis der sorgsam auf dem Speicher verwahrten zu den entsorgten, verschenkten oder verkauften Kästen.

Betrachtet aus der Perspektive aktueller „Pay per Use“-Geschäftsmodelle erscheinen da die Preise eines fischertechnik-Kastens gleich in einem anderen Licht: Geteilt durch die Zahl der mit ihnen glücklich verbrachten Spielminuten dürfte fischertechnik zu den mit Abstand günstigsten Spielzeugen überhaupt zählen – noch vor Lego, denn dort mutieren gelegentlich einige der besonders ausgefeilten Modelle nach dem ersten Aufbau zur Regaldeko (vulgo: Staubfängern).

Sicher ist es auch vielen von euch vergönnt, ein paar fischertechnik-Kästen unter dem Weihnachtsbaum zu entdecken – die hoffentlich ebenso intensiv bespielt und lange gehegt werden wie unsere aus dem Jahr 1973.

Ein paar Ideen, was ihr damit so alles anstellen könnt, findet ihr in dieser ft:pedia.

Schöne fischertechnische Weihnachten wünscht euch
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	Überraschung	2
	Mini-Modelle (Teil 8): Flugsaurier.....	4
	Die Geheimnisse der Turmbergbahn	5
	Die fischertechnik-Werkzeug-Wanne	12
	Kaulquappenperlentauchen	15
	fischertechnik-Aufbewahrung	19
	Druckluftsteuerungen (Teil 4).....	24
	PWM-Tongenerator für Robo TX(T) Controller	31
	Nikon-Kamera-Ansteuerung über IR.....	40
	I ² C mit dem TX(T) – Teil 12: Temperatursensor.....	44
	PWM-Motorsteuerung am fischertechnik-Universal-Interface	49

Termine	Was?	Wann?	Wo?
	Modellpräsentation „ Technikgeschichte mit fischertechnik “	15.02.2016	Technoseum Mannheim
	Maker Faire	04.03.2016	Mathematikum Gießen
		27.- 29.05.2016	HCC Hannover

Hinweise Ehre, wem Ehre gebührt: Urheber des im Artikel „Kaulquappen (Teil 6)“ der ft:pedia 3/2015 unter der Kapitelüberschrift „Guest Star: Z58“ vorgestellten, via ft-Spurkranz auf die Achse verschraubten Drehkranzes ist Ludger Mäsing.

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

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

Autoren: Gerhard Birkenstock, Stefan Falk, Dirk Fox,
Andreas Gail, Ralf Geerken, Jörg-Peter Rau, René Trapp,
Dirk Uffmann.

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

Modell

Mini-Modelle (Teil 8): Flugsaurier

René Trapp

Und wieder ein zauberhaftes Mini-Modell – diesmal aus der Kategorie „Paläonthologie mit fischertechnik“.

Eigentlich sollte es ein kleines Vögelchen werden. Und dann schlüpfte der hier aus dem Ei.

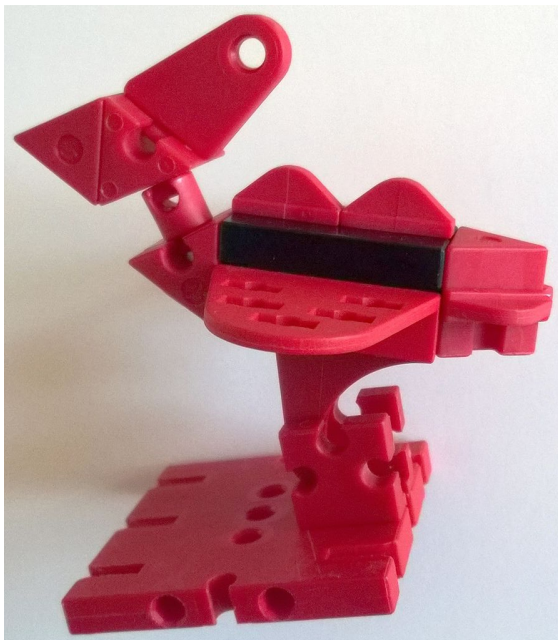


Abb. 1: Flugsaurier

Natürlich darf die Stückliste nicht fehlen:

Stück	ft-Nr.	Bezeichnung
2	31010	Winkel 60°
2	31918	Winkelstein 60°, 3-Nut
1	32879	Baustein 30 schwarz

Stück	ft-Nr.	Bezeichnung
2	38252	Lagerbock
1	38473	Lenkhebel
1	35980	Klemmhülse
4	31602	Kufe
1	31592	Kesselhalter
1	36093	Grundplatte 45x22
1	37636	Rollenlager

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.

Modell

Die Geheimnisse der Turmbergbahn

Ralf Geerken

Hier sollen nicht nur die Geheimnisse der echten Turmbergbahn aus Karlsruhe/Durlach gelüftet werden, sondern auch die Geheimnisse meines fischertechnik-Modells. Für die Realisierung eines einwandfrei laufenden Modells waren nämlich doch einige Tricks, Kniffe und fast nicht sichtbare Modifikationen an Bauteilen notwendig.

Geheimnis Nummer 1: Was mich mit dieser Bahn verbindet

Mit dieser Bahn verbindet mich meine Oma, sie sagte übrigens immer liebevoll „Bähnle“ zu ihr und wohnte nur ein paar hundert Meter vom „Bähnle“ entfernt. Als Kind bin ich oft mit ihr (also mit der Turmbergbahn – nicht mit meiner Oma) gefahren und habe mich damals schon immer gefragt, wie die in der Mitte liegende

Weiche der Bahn denn wohl funktioniert. Nicht einmal mein technisch sehr versierter Vater, seines Zeichens Maschinenbau-Ingenieur, konnte mir darauf eine Antwort geben. Ich habe damals geschaut, beobachtet, gequengelt – aber es bewegte sich weder etwas an der Weiche, noch nützte es etwas. Ich bekam es damals einfach nicht heraus. Das Prinzip der Abt'schen (Aus-) Weiche hatte ich dann vor ein paar Jahren zwar gegoogelt und das Prinzip war mir dann auch schnell klar, wie aber die Seilführung, also die Lage des Seils in der Weiche und schlussendlich die Befestigung an dem Wagen funktioniert, habe ich erst durch Ausprobieren bzw. durch dieses Modell herausgefunden.

Geheimnis Nummer 2: Der Antrieb

Die beiden Wagen sind mit einem Seil verbunden. Im Original wird ein Stahlseil mit 26 mm Durchmesser verwendet, bei mir tut es ein fischertechnik-Seil. Dieses Seil wird an der Bergstation mehrfach um eine Walze geführt, wobei diese dann von einem Elektromotor bzw. im Original von einem Leonardsatz angetrieben wird. Das Gewicht beider Wagen hängt also an der



Abb. 1: Die Turmbergbahn in Karlsruhe (Quelle: Wikipedia)



Abb. 2: Das Turmbergbahn-Modell

Antriebswalze. Da sich die Gewichte gegenseitig aufheben, muss der Antrieb nicht das Gewicht eines einzelnen Wagens nach oben ziehen, sondern nur den Gewichtsunterschied beider Wagen. Und auch nur dann, wenn der untere Wagen schwerer ist. Ist der obere Wagen schwerer, muss der Motor sogar als Bremse dienen. Der Leonardsatz kann die Bremsenergie dann auch wieder ins Stromnetz einspeisen. Ich habe zwar keinen Beweis dafür gefunden, dass dieses Prinzip auch bei der Turmbergbahn verwendet wird, meines Erachtens nach kann es aber gar nicht anders sein, weil die entstehende Energie bei schwererem, abwärts fahrenden Wagen ja irgendwo hin muss.

Anfangs habe ich das Seil auch mehrfach um eine Drehscheibe gelegt, um durch die höhere Reibung mehr Energie auf die Wagen zu bringen. Das führte aber sehr oft zu einem Hakeln oder gar zu einem Verheddern des Seils. Jetzt ist das Seil wieder nur halb um die Drehscheibe gelegt, in die Drehscheibe ist allerdings ein einfacher Haushaltsgummiring eingelegt, der verhindert, dass das Seil durchrutscht. Für eine ordentliche Untersetzung habe ich

seinerzeit den Drehkranz ([31390](#)) auseinandergenommen und zu einem Z58 umfunktioniert. Diese Kaulquappe hat Harald Steinhaus in der [ft:pedia 3/2015](#) beschrieben, wobei es eine Richtigstellung meinerseits gibt: Da Ludgers Monster-LKW [2] wesentlich älter ist, stammt diese Kaulquappe von ihm [3] (*Anm. d. Red.: Siehe auch den Hinweis beim Impressum*). Das zweite Z58 ist ein Überbleibsel aus der Testphase und hat einen Freilauf. Die Zahnradkette kann während der Fahrt auf dieses Z58 umgehängt werden, der Motor läuft dann weiter, aber das Bähnle kann von jetzt an per Hand geschoben werden. Dadurch ist es auf einfache Art und Weise möglich, Fehlerursachen zu finden, ohne die Bahn jedes Mal umzubauen.

Geheimnis Nummer 3: Seilführung und -befestigung

Zuerst muss man wissen, dass die beiden Wagen niemals die Seite wechseln. Das heißt, der eine Wagen fährt von oben gesehen an der Ausweichstelle immer rechts und der andere von oben gesehen immer links. Das heißt aber auch, dass die Seile sich nicht kreuzen. Bis 1965 war es

sogar so, dass die Strecke der Turmbergbahn aus 3 einzelnen Schienen bestand, wobei die mittlere Schiene von beiden Wagen befahren wurde.



Abb. 3: Seilaufhängung unterhalb des Schienenniveaus

Nur in der Abt'schen Ausweiche befahren und befahren sie logischerweise jeweils zwei Schienenstränge – die Wagen müssen ja aneinander vorbeifahren. Die Seile sind unter dem Wagen im oberen Drittel befestigt. Das ist jedenfalls bei meinem Modell so. Ob das beim Original auch der Fall ist,

weiß ich leider nicht, weil ich dann doch nicht so dreist war, unter den Wagen zu klettern, als ich im Sommer 2015 dort war. Der Befestigungspunkt liegt unterhalb der Schiene (siehe Abb. 3), das heißt er liegt tiefer als die Oberkante der Schiene.



Abb. 4: Die Wagen begegnen sich

Das muss auch so sein, damit das Seil bei Links-/Rechtsfahrt in der Abt'schen Ausweiche mit dem Wagen zusammen in die Kurve geht. Würde das Seil oberhalb der Schiene am Wagen befestigt werden, dann

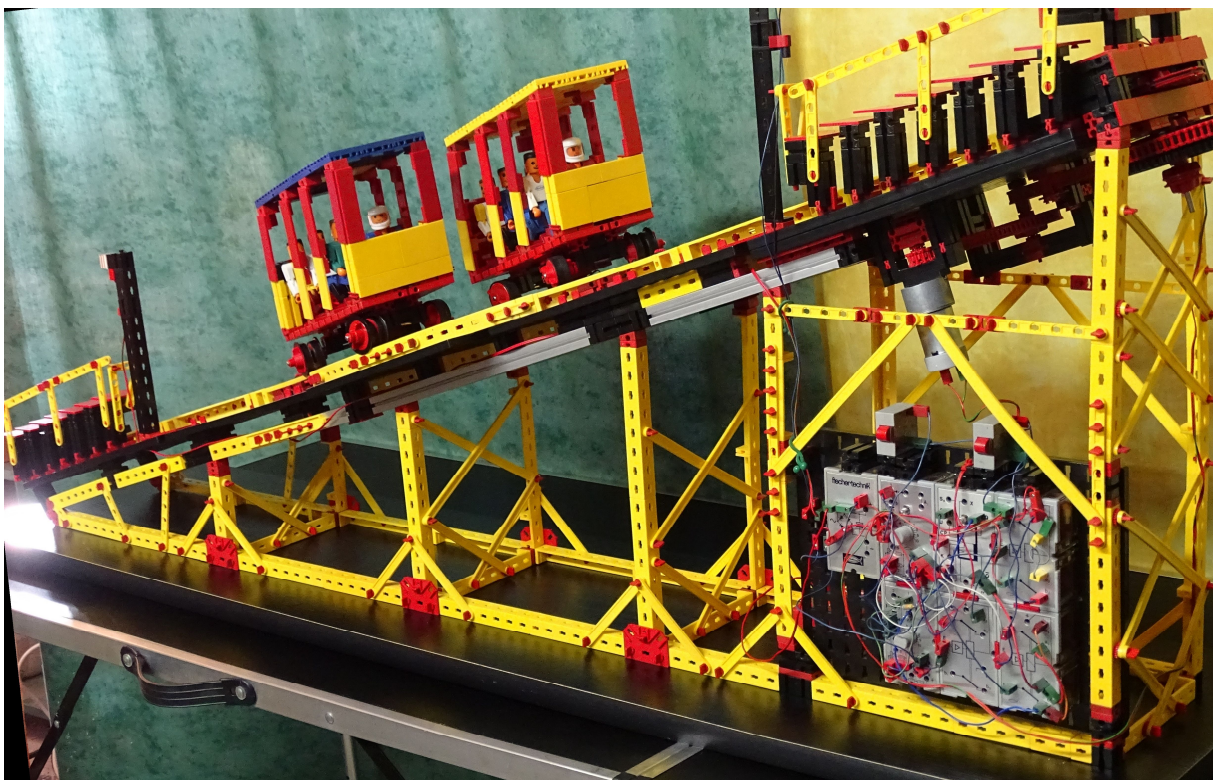


Abb. 5: Das Modell in Schrägansicht

würde es geradewegs über die Ausweiche hinweggehen und der entgegen kommende Wagen müsste das Seil überfahren, was dem Seil gar nicht gut tun würde. Den klein gehaltenen Befestigungspunkt habe ich mit einer kleinen Clipsachse (32870) realisiert, bei der ich den Clipsteil der Achse kürzen musste, weil er sonst zu weit ins Gleisbett hineingeragt hätte. Im Original wird das schwere Drahtseil über die ganze Strecke von schräg und gerade liegenden Rollen geführt, damit es nicht einfach im Gleisbett auf dem Boden schleift. Beim Nachbau habe ich in den Kurven auf die Seilrollen verzichten müssen, weil hier einfach kein Platz war. Nichtsdestotrotz habe ich aber mehrere Rollen (38258) auf den geraden Strecken eingebaut, weil das Seil dann wesentlich besser dageblieben ist, wo es sein muss, nämlich unterhalb der oberen Schienenkante.

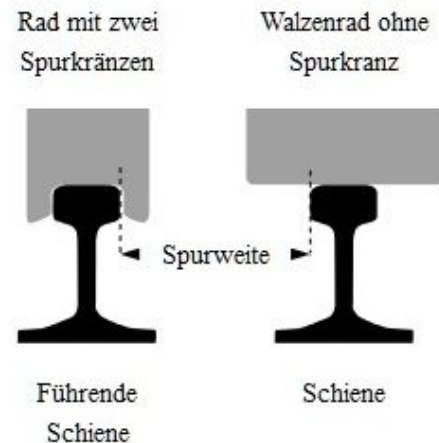
Geheimnis Nummer 4: Die Räder

Im Original und im Nachbau haben die Führungsräder einen doppelten Spurkranz. Das heißt sie umschließen die Schiene wie ein auf dem Kopf stehendes U. Die Führungsräder bleiben immer auf der äußeren Schiene, wodurch der linke Wagen auf der Abt'schen Ausweiche auch links bleibt und der rechte Wagen immer rechts (siehe Abb. 6). Die anderen Räder sind etwas breitere Walzenräder und haben keinen Spurkranz. Den dürfen sie auch nicht haben, weil der Spurkranz dann bei der Einfahrt in die Ausweiche ja irgendwie auf die andere Seite der Schiene (und wieder zurück) gelangen müsste.

Die ft-Spurkranze (36331) hatten sich zwar als brauchbar erwiesen, ich musste sie aber trotzdem modden, weil sie oft nicht in der Spur blieben. Das hinderliche Gummi habe ich einfach nur vom Rad heruntergenommen. Den sehr kleinen Ring, der normalerweise das Gummi auf der Lauffläche hält, habe ich abgeschliffen. Schließlich habe ich noch die Innenseite des

Spurkranzes ein wenig abgeschragt, damit das Rad besser von der Schiene in die richtige Richtung gedrückt werden kann.

linker Wagen



rechter Wagen

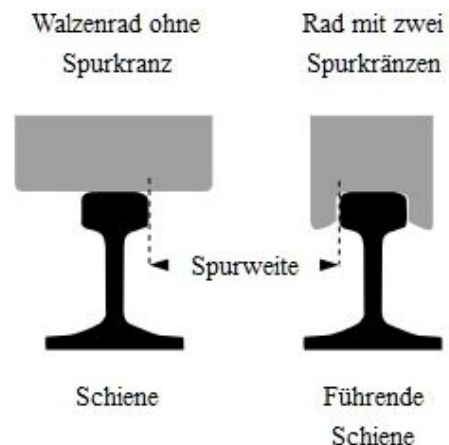


Abb. 6: Das Prinzip der Räder
(Quelle: Wikipedia [4])

Bei den Walzenrädern habe ich die alten Reifen (31017) genommen, wobei ich diese aber soweit herunter geschliffen habe, dass sie denselben Durchmesser wie die Spurkranzräder haben (siehe Abb. 7). Als die Durchmesser noch (wenn auch nur sehr geringfügig) unterschiedlich waren, drückten die Räder schon auf der geraden Strecke immer in eine Richtung. Inzwischen haben die Führungsräder aber durch Freilaufnaben (68535) quasi Einzelradaufhängungen

erhalten, womit sie jetzt richtig gut in der Spur bleiben.



Abb. 7: Gemoddete Räder in Detailansicht

Geheimnis Nummer 5: Die Steuerung

Das Bähnle sollte von Anfang an immer einige Zeit an den Bahnhöfen warten und dann (logischerweise) in entgegengesetzter Richtung wieder automatisch weiterfahren. Am oberen Punkt sitzt dafür ein Taster, der vom ankommenden Wagen betätigt wird.

Anfänglich hatte ich einen TX-Controller zur Steuerung eingesetzt. Das Programm war auch schnell geschrieben; es musste ja nicht viel tun: Wenn Taster nicht gedrückt,

dann Motor in einer Richtung einschalten und solange eingeschaltet lassen, bis der Taster gedrückt wird. Dann einige Zeit warten und den Motor in entgegengesetzter Richtung wieder einschalten, und so fort.

Dann fragte mich jemand auf der Convention 2014 in Erbes-Büdesheim, ob ein TX für eine solch einfache Steuerung nicht „etwas“ überdimensioniert sei; das müsste doch auch mit Silberlingen gehen. Wieder zu Hause angekommen habe ich dann etwa die dreifache Zeit, die ich bisher für die Realisierung der gesamten Bahn aufgewendet hatte, in die Steuerung mit Silberlingen gesteckt. Die Lampen an den Bahnsteigen sind LEDs und werden direkt über A2 des unteren Grundbausteins angesteuert. Dieser sorgt dafür, dass die LEDs während der Fahrt blinken, und während sie steht dienen sie der Bahnsteigbeleuchtung. Die Silberling-Steuerung kann aber noch einiges mehr: Durch einen zusätzlichen Taster kann die Bahn während der Wartezeit frühzeitig gestartet werden. Mit einem weiteren Taster kann während der Fahrt die Richtung gewechselt werden, dadurch kann man die Bahn bspw. in der Mitte auf der Weiche ‚tanzen‘ lassen. Falls mich meine Erinnerung nicht täuscht hat der Schaffner das

im Original bei irgendwelchen Anlässen auch manchmal getan, dafür habe ich aber weder Belege oder Beweise, noch können sich meine Eltern daran erinnern – vielleicht habe ich das ja damals auch nur geträumt.

Kein Geheimnis...

... sondern der Wikipedia entnommen: Die Turmbergbahn ist eine Standseilbahn, die von Karlsruhe-Durlach aus auf den Turmberg führt. Sie wurde 1888 eröffnet und ist damit

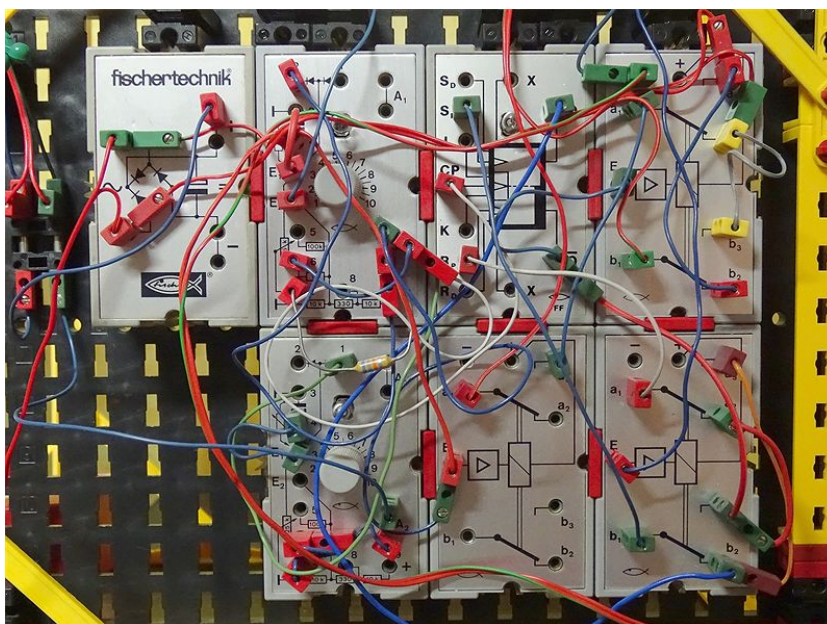


Abb. 8: Elektronische Steuerung

die drittälteste Standseilbahn in Deutschland, und die älteste, die noch in Betrieb ist. Auf einer Streckenlänge von 315 m überwindet die Bahn eine Höhendifferenz von 100 m bei einer maximalen Steigung von 36,2 %.

Die Turmbergbahn wurde am 1. Mai 1888 dem Betrieb übergeben. Der Antrieb erfolgte durch Schwerkraft mit Wasserballast, d. h. ein Tank im talwärts fahrenden Wagen wurde mit der erforderlichen Menge Wasser befüllt und zog den bergwärts fahrenden nach oben. Die Strecke war ursprünglich zweigleisig. (*Anm.: Nach Erzählungen meiner Eltern bin ich mit dieser wasserbetriebenen Variante auch schon gefahren, allerdings nur im Kinderwagen.*)

Die Bahn wurde 1966 umfassend umgebaut. Dabei wurden der Wasserballast-Antrieb durch einen elektrischen Antrieb ersetzt, neue Wagen beschafft und die zweigleisige Strecke zu einer eingleisigen mit in der Mitte liegender Abt'scher Ausweiche umgebaut. Die Bahnhofsgebäude an der Tal- und Bergstation wurden durch Neubauten ersetzt.

Auch kein Geheimnis...

... sondern Wikipedia entnommen, aber um einige Textstellen gekürzt: Die Abt'sche Weiche:

Die abtsche Weiche (auch Abt'sche Weiche) ist eine vom Schweizer Eisenbahn-Konstrukteur *Carl Roman Abt* (1850-1933) als „passive Weiche“ erfundene Gleiskonstruktion für Standseilbahnen, die eine selbsttätige Vorbeifahrt der Fahrzeuge an einer Ausweichstelle eingleisiger Strecken ermöglicht.

Diese Weichenkonstruktion ohne Weichenzunge und sonstige bewegliche Teile findet an den in der Streckenmitte nötigen Ausweichstellen eingleisiger Standseilbahnen Anwendung, um die Vorbeifahrt

entgegenkommender Fahrzeuge beziehungsweise Bahnkabinen selbsttätig („automatisch“) zu bewerkstelligen. Die Abt'sche Weiche ermöglicht auch eine problemlose Führung des Antriebsseils, welches mit schräg gestellten Rollen zwischen den Fahrschienen korrekt in das jeweilige Ausweichgleis geführt wird.

Jeder Radsatz eines Wagens ist mit zwei unterschiedlichen Rädern ausgerüstet: auf einer Seite mit einem Doppelspurkranzrad mit beidseitigen Spurkränzen (auch Doppelflanschrad genannt) und auf der anderen Seite mit einem breiten Walzenrad ohne Spurkränze.

Die Weiche selbst ist starr, muss also aufgrund systembedingt fehlender beweglicher Teile nie gestellt werden. Damit ist sie sowohl weitestgehend wartungsfrei als auch völlig ausfallsicher, was bei Bahnen dieser Art mit ihren zum Teil sehr steilen Strecken einen wichtigen Sicherheitsfaktor darstellt.

Quellen

- [1] Wikipedia: [Turmbergbahn](#).
- [2] Mäsig, Ludger: [Riesen-LKW](#).
- [3] Mäsig, Ludger: [Selbstgebautes Differenzial](#).
- [4] Wikipedia: [Abtsche Weiche](#).

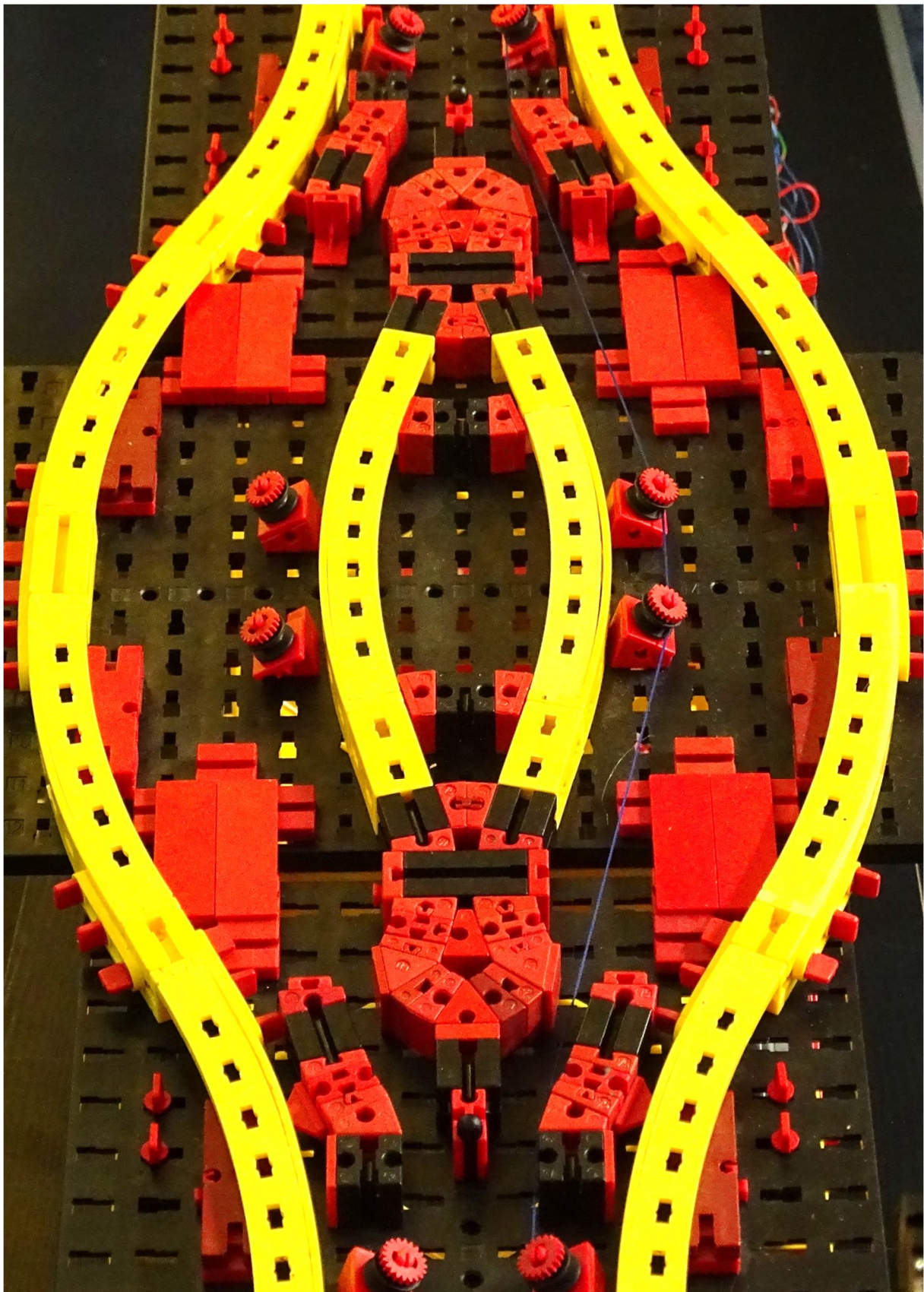


Abb. 9: Die Abt'sche Weiche

Tipps & Tricks

Die fischertechnik-Werkzeug-Wanne

Gerhard Birkenstock

Das Spielen („Arbeiten“) mit fischertechnik wird an vielen Stellen durch Werkzeug erleichtert. Wenn Werkzeuge sich – wohl sortiert – in einem Kasten befinden, nennt man das einen Werkzeugkasten. Damit die Werkzeugsammlung in das fischertechnik-System passt, kommt sie in die Sortierwanne.

Auflistung der Werkzeuge

In meiner ft-Werkzeug-Wanne befinden sich einige Schlüssel, die mir das Hantieren mit fischertechnik um vieles erleichtern, ja sogar Spaß bereiten. Im Einzelnen sind das die folgenden:

Nabenschlüssel



Abb. 1: Nabenschlüssel zwei Varianten

Die beiden Nabenschlüssel in Abb. 1 stammen von Andreas Tacke (TST) [1]. Mit ihnen kann man prima die Nabenmutter (31058) und die Zangenmutter (31915) fest ziehen und lösen.

Schraubendreher



Abb. 2: fischertechnik-Schlitzschraubendreher

Der originale Schraubendreher 36443 von fischertechnik. Er ist zum Anschließen der Kabel an die kleinen 2,5 mm Miniatur-Bananenstecker gedacht. Er wird bei jeder Kabelveränderung und Reparatur benötigt.

S-Riegel-Schlüssel

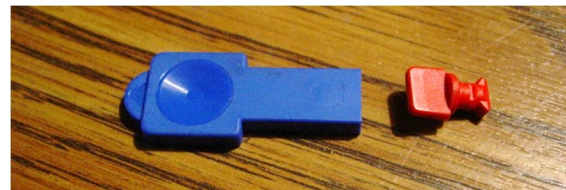


Abb. 3: fischertechnik S-Riegel-Schlüssel

Der S-Riegel-Schlüssel (35799) ist die einzige Montierhilfe von fischertechnik. Leider kommt man mit diesem kurzen Schlüssel sehr oft nicht an die Stellen heran, an denen der Riegel gesetzt werden soll. Es ist immer das Geschick der Hände gefragt.

Abhilfe habe ich mir hiermit geschaffen:

S-Riegel-Bit-Einsatz



Abb. 4: Bit-Einsatz mit zurück gezogener Zentrierhülse

Auf der linken Bildseite sind die Flächen für den Bit-Antrieb zu sehen. Nach der Normung haben die sechs Flächen einen

Abstand von 6,35 mm. Mit Teilkopf und Fräsmaschine habe ich diese Flächen aufgebracht.

Rechts ist der Schlitz für den S-Riegel eingesägt. Der Durchmesser der Metallachse beträgt 8 mm. Das Zentrierrohr hat außen 10 mm Durchmesser. Um an engen Stellen mit dem Zentrierrohr arbeiten zu können, ist das Rohr auf der Drehmaschine noch abgedreht. Mit 9 mm Durchmesser komme ich bisher an jedem Einsatzort gut zurecht.

Der Drehknopf ist von Conrad (824605) zugekauft. Er ist schön klein gehalten und doch mit dem 6,35 mm System bestückt. Da passt alles prima zusammen.



Abb. 5: Explosionsdarstellung



Abb. 6: ...einsatzbereit

Dank der vielen heutigen Möglichkeiten bei dem Bit-System (Verlängerung, Winkelgetriebe, ...) kommt man nun an die entlegensten Stellen in seinen Modellen

heran. Es macht richtig Spaß, damit die Riegel einzusetzen.



Abb. 7: Detaildarstellung

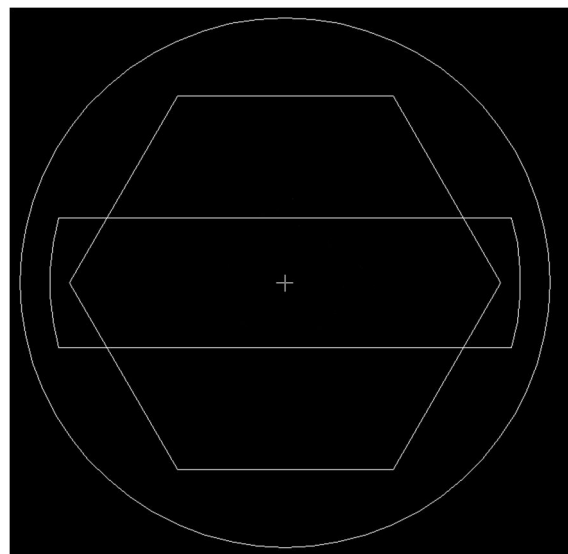


Abb. 8: Detailzeichnung

Da der Schlüssel keine großen Kräfte übertragen muss, würde auch ein Kunststoffstab mit 10 mm Durchmesser mit innen liegendem Schlitz und Bit-Sechskant ausreichen.

Zusammenführen der Werkzeuge

In der fischertechnik-Wanne (36587) kommt nun alles zusammen. Und dank eines ausgeschnittenen Schaumstoffs bleibt es auch schön griffbereit an seinem Platz.

Selbst das Stapeln der Wannen funktioniert dank des dünnen Schraubendreher-Heftes noch immer.



Abb. 9: fischertechnik-Werkzeug-Wanne

Beim Anwenden der Werkzeug-Wanne ist mir nun aufgefallen, dass ich doch ein Werkzeug vergessen habe. Es lag hinter einer Sortierkiste: Die abgewinkelte Pinzette!

Referenzen

- [1] Andras Tacke: *ft-Spezialzeile made by TST (Teil 4)*, [ftpedia 2/2013](#), S. 11-12.

Tipps & Tricks

Kaulquappenperlentauchen

René Trapp

Von „Kaulquappen“ und vom „Perlentauchen“ war in der ft:pedia bereits zu lesen ([1] bis [12]). Zieht man anstelle der Taucherausrüstung allerdings die Zwangsjacke an und springt ins falsche Gewässer, dann macht man seltsame Fänge.

Wer Ende Juli 2015 auf dem „50-Jahre-fischertechnik-Fanclubtag“ die Zeit fand der Glasmurmelnbahn des Autors eine Weile zuzusehen (Abb. 1), hat möglicherweise mitbekommen, dass unter anderem am Schleplift (Abb. 2) ein kleines, aber gemeines Problem auftrat.



Abb. 1: Die Glasmurmelnbahn

Die Statiklaschen 21,2 (31668) hielten nicht wirklich fest in den Schleppschlitten, und abends lagen einige der Laschen auf den Grundplatten unter dem Schleplift herum.

Die Konstruktion der Schleppschlitten (Abb. 3) ist ja schon abenteuerlich genug: I-Streben 45 (36913) verbinden den Mitnehmer an der Kette mit einem U-Träger-Adapter (35979). Innen auf den Streben sitzen die besagten Laschen. Sie passen *zufälligerweise* leicht klemmend zwischen die Klemmbuchsen (37679) und den U-Träger-Adapter.

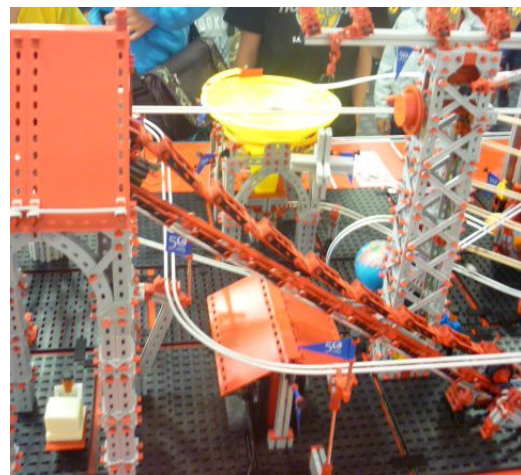


Abb. 2: Der Schleplift



Abb. 3: Ein Schleppschlitten

Der Zweck der Laschen ist es, den U-Träger-Adapter daran zu hindern, sich zu verdrehen. Mit verdrehten U-Träger-Adapttern werden die Murmeln unten am Einlauf nicht sauber vereinzelt und abgeholt oder oben am Ausstieg danebengeworfen.

Nun gab es also dieses Problem, dass die Klemmwirkung alleine die Laschen nicht dauerhaft festhalten konnte. Bedingt durch die Stöße beim Umlauf rutschten sie immer wieder von den *leicht konischen* Seiten der

Klemmbuchsen ab, und damit dann aus den Schlitten heraus.

Der Effekt war schon bei den Testläufen vor der Ausstellung aufgetreten, wurde aber als „ausreichend unbedeutend“ eingestuft. Die nicht gezeigte zickige Hebekunst [13] hatte bekanntlich richtig derben Ärger verursacht.

Langfristig sollte nun allerdings doch eine Lösung her – **ohne Modding** und **ohne Klebstoffe**. Doppelseitiges Klebeband zwischen I-Strebe und Lasche ist zwar möglich, naheliegend und einfach aber nur als äußerste Notlösung akzeptabel, wenn wirklich nichts aus reinem fischertechnik zu finden sein sollte, um die U-Träger-Adapter am Verdrehen zu hindern.

Wer sich jetzt fragt, warum es unbedingt der U-Träger-Adapter sein muss, der da das untere Ende des Schleppschlittens bildet: Nur dieses eine fischertechnik-Teil hat in etwa die benötigte Form und hinreichend geringe Dicke, um die Murneln unten am Einstieg zu vereinzeln.

Also hat der Autor nun in wochenlanger Kleinarbeit seinen Teilebestand gesichtet und einige Kombinationen ausprobiert, die den U-Träger-Adapter dauerhaft gegen Verdrehen sichern können. Dabei kommt es darauf an, dass ein in ein Statikloch des U-Träger-Adapter eingesetztes Teil mit seiner Kante exakt an einer Kante des U-Träger-Adapters anliegt. Einige Möglichkeiten fanden sich dann auch (Abb. 4), haben aber so ihre Nachteile: Innen zu schmal, dezent gewölbt (= krumm), zu lang, zu kurz, zu klobig, ...

Und so bleibt schlussendlich doch nur das doppelseitige Klebeband. Erfreulicherweise gibt es das Klebeband auch in 8 mm Breite, passend zur Breite der Statikstreben. Das vereinfacht die Verarbeitung.

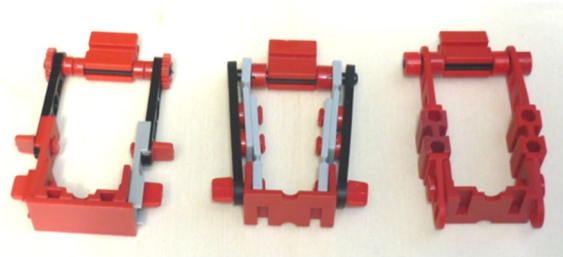


Abb. 4: Einige Ideen für verbesserte Schleppschlitten

Ein paar Tage später...

... grübelt der Autor immer noch über dieses Problem. Das Klebeband wäre zwar von außen kaum zu sehen, aber die Maxime lautet bislang „unverklebt“ und müsste nun aufgegeben werden. Nein, das geht auf keinen Fall. Einer der Kandidaten aus Abb. 4 könnte eventuell vielleicht doch geeignet sein, dafür sind dann aber gleich wieder reichlich Teile zur Beschaffung fällig. Immerhin sind 19 Schleppschlitten in Umlauf. Nein, das ist zu unsicher, denn hier liegen die Nuten sogar längs zu den Stößen beim Herunterklappen (Abb. 5). Und zu klobig ist das obendrein, es sieht nicht wirklich gut aus.



Abb. 5: Das Herunterklappen

Abb. 5 zeigt die alte Version der Schleppschlitten beim Herunterklappen. Das geschieht so schnell, dass der gerade heruntergeklappte Schlitten im Bild völlig verwischt ist.

Plötzlich: der Geistesblitz!

Bislang unbeachtete Details der Bauteile treten auf einmal in den Fokus. Unglaublich banale und bislang völlig unbeachtete Details. Solche Details, dass man sie schon

fast mit der Lupe suchen muss. Solche alltäglichen Details, dass man sie geflissentlich ignoriert, weil sie so allgegenwärtig sind. Und dann sind es nicht einmal wichtige Details, sondern teilweise nur Nebenprodukte des Produktionsprozesses. Also: Licht aus, Spot an.

Jede der Laschen hat auf einer Stirnseite so einen kleinen Pickel – den „Nabel“ vom Spritzguss (Abb. 6).

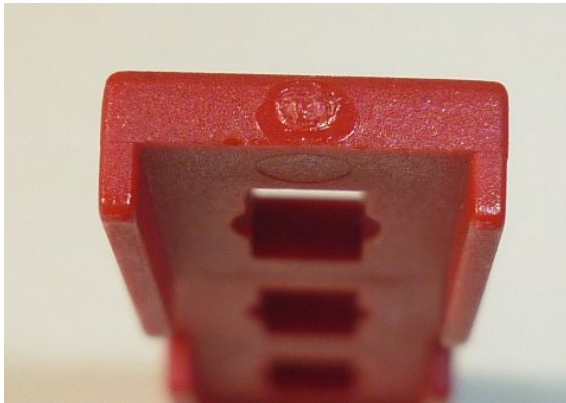


Abb. 6: Stirnseite einer Lasche 31668

Jede Klemmbuchse hat auf einer Seite so eine kleine Kerbe. Nein, nicht den großen durchgehenden Schlitz, sondern genau gegenüber, nur außen und nur ein kleines Stück ins Material hinein (Abb. 7).

Ob es sich einrichten lässt, diesen „Nabel“ mit der Kerbe zu verhaken? Dafür müssen „nur“ an allen Schlitten je zwei Klemmbuchsen umgedreht und alle Laschen einheitlich ausgerichtet werden. Die Erwartungshaltung des Autors bezüglich der Haltbarkeit dieser Verbindung war nicht besonders hoch.

Aber in der Not macht man verrückte Dinge, also wurde ein Schlitten umgebaut (Abb. 8). Natürlich ist es nicht einfach den Effekt auszunutzen, denn beide Klemmbuchsen müssen exakt gleich ausgerichtet sein, um die Laschen einrasten zu lassen. Natürlich kann diese Verbindung keine hohen Zugkräfte aushalten. Aber die Laschen rasten tatsächlich fühlbar ein!

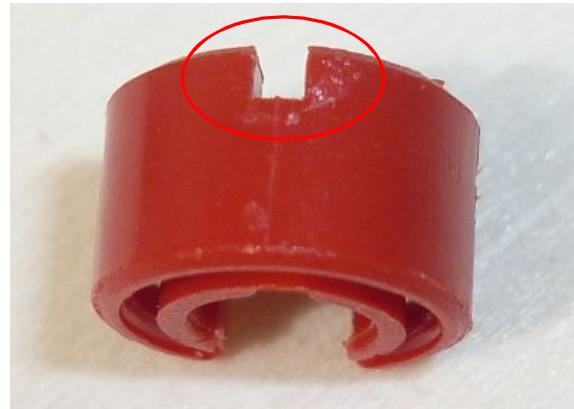


Abb. 7: Kerbe in der Klemmbuchse 37679



Abb. 8: Ein umgebauter Schleppschlitten

Ein *Zufall* oder doch etwas anderes? Wir werden wohl nie erfahren, warum die kleinen Kerben in den Klemmbuchsen sind. Die Abmessungen der Kerben unterscheiden sich sogar; es gibt längere und kürzere Kerben. Wir werden auch nie erfahren, warum die Laschen genau diese Außenmaße haben. Aber für das Problem des Autors liegt hier tatsächlich die Lösung. Es sind die kleinen „Nabel“ der Laschen, die einrasten. Auf geniale Weise findet der „Nabel“ dort auch gleich genügend Platz, und so ist die andere Seite sehr glatt und liegt flächig am U-Träger-Adapter an.

Am hinteren Ende sorgt die Geometrie mit der Klemmwirkung zusammen dafür, dass die Laschen nicht abrutschen, vorne sorgt die neu gefundene Rastmöglichkeit für ausreichend festen Halt. Mit nur einem Schlitten im Umlauf kann man nicht viel testen, also hat der Autor die anderen 18 Schlitten auch noch alle umgebaut und auf eine sehr lange Rundreise geschickt.

Seit dem Umbau sind mittlerweile mehr als 50 Betriebsstunden vergangen und zu Beginn hat nur eine Lasche ihren Platz mehrfach verlassen: Lasche und Klemmbuchse passen anscheinend im letzten Detail nicht richtig, sodass hier beide Teile auszutauschen waren. Alle anderen 37 Paare halten fest zusammen. Dass das so erstaunlich gut funktioniert ist schier unglaublich! Ja es kommt noch besser: die gewohnte Optik ist unverändert, es müssen keine zusätzlichen Teile beschafft werden, und die strenge Vorgabe (nicht verklebt, nicht gemoddet) bleibt erfüllt.

Zusammenfassung

Abb. 9 zeigt abschließend den direkten Vergleich. Eine kleine Ursache, die eine erstaunlich große Wirkung hat. Und so stellt sich die letzte Frage: „Muss man etwas an der Murmel haben um auf eine derart wahnwitzige Idee zu kommen?“ Der Autor meint: Vermutlich nicht. Aber mir hat es geholfen.

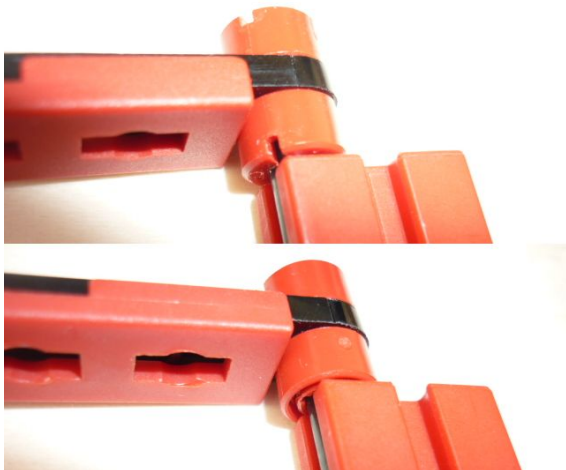


Abb. 9: Vorher (oben) – Nachher (unten)

Referenzen

- [1] Harald Steinhaus: *Kaulquappen (Teil 1)*, [ft:pedia 1/2011](#), S. 22-28.
- [2] Harald Steinhaus: *Kaulquappen (Teil 2)*, [ft:pedia 2/2011](#), S. 9-13.
- [3] Stefan Falk: *Perlentauchen (Teil 1)*, [ft:pedia 3/2012](#), S. 11-18.
- [4] Harald Steinhaus: *Kaulquappen (Teil 3)*, [ft:pedia 3/2012](#), S. 24-26.
- [5] Stefan Falk: *Perlentauchen (Teil 2)*, [ft:pedia 4/2012](#), S. 14-21.
- [6] Stefan Falk: *Perlentauchen (Teil 3)*, [ft:pedia 1/2013](#), S. 22-31.
- [7] Stefan Falk: *Perlentauchen (Teil 4)*, [ft:pedia 2/2013](#), S. 18-30.
- [8] Stefan Falk: *Perlentauchen (Teil 5)*, [ft:pedia 4/2013](#), S. 6-15.
- [9] Harald Steinhaus: *Kaulquappen (Teil 4)*, [ft:pedia 1/2014](#), S. 21-23.
- [10] Harald Steinhaus: *Kaulquappen (Teil 5)*, [ft:pedia 3/2014](#), S. 14-16.
- [11] Harald Steinhaus: *Kaulquappen (Teil 6)*, [ft:pedia 3/2015](#), S. 4-7.
- [12] Ralf Geerken: *Kaulquappen (Teil 7)*, [ft:pedia 3/2015](#), S. 8-9.
- [13] ft-community: [Die zickige Hebekunst](#).

Tipps & Tricks

fischertechnik-Aufbewahrung

Jörg-Peter Rau

Wohin mit all den schönen Sachen? Diese Frage stellt sich wohl jedem Fischertechniker. Bei der Wahl eines Systems zur Aufbewahrung sind verschiedene Aspekte zu beachten. Hier wird eine Lösung für kleine ebenso wie große Sammlungen vorgestellt, die in ein modular anpassbares Möbelsystem münden kann. Umfassende Anleitungen zum Nachbau sind [auf der Community-Webseite](#) zu finden. Wer in das Thema Aufbewahrung investiert, wird auf jeden Fall mit viel mehr Spielspaß belohnt!

Mit der Idee des Baukasten-Systems, bei dem alle Kästen aufeinander aufbauen, starb bei fischertechnik auch das in den 70er-Jahren verfolgte Prinzip, die Bausteine in dezidierten, stabilen Kästen auszuliefern. Dort hatte jedes Element seinen festen Platz, war sofort zu finden, und durch die Beschränkung auf nur zwei stapelbare Standardgrößen und eine universelle Box drumherum war die Frage der Aufbewahrung geklärt. Jedenfalls solange keine Zusatzpackungen und Einzelteile erworben wurden (Abb. 1).

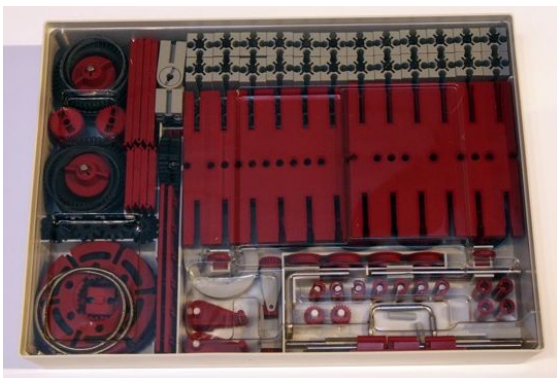


Abb. 1: Sortierwanne der 70er Jahre

Seit den 80er Jahren kommt fischertechnik (wie schon in den Anfangstagen nach 1965) fast ausschließlich in Kartons mit leichten Tiefziehteilen, die für eine dauerhafte Aufbewahrung kaum eine Lösung sind und viel zu wenige Fächer für ein echtes Sortieren der Teile haben (Abb. 2).



Abb. 2: fischertechnik in Tiefziehteilen

Welche Fragen sind zu klären?

Im Kern stellen sich also verschiedene Fragen, wenn es um die Aufbewahrung und Organisation einer fischertechnik-Sammlung geht. Einige Vorschläge sind in der fischertechnik-Community dazu [bereits gemacht worden](#). Zu klären ist in jeden Fall Folgendes: Soll die Sammlung stilecht in originalen fischertechnik-Produkten, also Boxen 500, Boxen 1000 und den dazugehörigen Sortierwannen aufbewahrt werden? Oder ist eine Lösung mit Sortiersystemen von Fremdanbietern akzeptabel? Ist die Sammlung abgeschlossen oder wird sie weiter wachsen? Wie schnell sind benötigte

Teile zu finden und wie komfortabel sind sie zu entnehmen? Gibt es einen sinnvollen Ansatz für die Beschriftung? Wie gut sind die Teile vor Staub und Licht geschützt? Wie platzsparend ist die Lösung? Soll Gleiches nutzerfreundlich zu Gleichem – oder soll in Anlehnung an die 70er Jahre die Integrität der einzelnen Baukästen gewahrt bleiben? Und was passiert, soweit vorhanden, mit den schönen alten Sortierkästen?

Was bietet fischertechnik selbst?

fischertechnik selbst hat für das Thema Aufbewahrung die Boxen 500 und 1000 im Angebot. Die Boxen 500 sind hochwertig, gut stapelbar und stabil, als Deckel kann die Bauplatte 500 verwendet werden. Die Inneneinteilung ist relativ flexibel. In den Kästen der Profi-Linie sind überdies Sortiervorschläge enthalten, die ggfs. kopiert ausgeschnitten und direkt auf die Böden gelegt werden können. Dies bietet sich auch für die (leider weit weniger soliden) Einlegewannen an, die mit der Box 1000 geliefert werden. Auch hier bieten die Trennstegte viel Flexibilität. Als Umgebände leistet die Box 1000 sehr gute Dienste, auch sie ist stabil.

Die Vorteile einer fischertechnik-Aufbewahrungslösung sind also eine gute Materialqualität, eine ordentliche Flexibilität und die Integration ins System mit den Bauplatten als Deckel. Und es kommt von fischertechnik. Die Nachteile sind die nicht immer gute Entnehmbarkeit gerade kleiner Teile, die Gefahr des Einstaubens durch die schwarzen Deckel-Bauplatten ohne Boden und der Umstand, dass Umbauten der Boxen 500 recht umständlich sind. Die Kosten liegen bei vergleichbarem Aufnahmevolumen bei knapp 30 € für entweder eine Box 1000 mit Sortierwannen oder vier Boxen 500 mit Trennstegen.

Was gibt es sonst am Markt?

Vor allem in Baumärkten oder gelegentlich auch bei Discountern sind andere Aufbewahrungssysteme für Kleinteile im Angebot. Die günstigen unter ihnen haben feste Einteilungen der Fächer, die teureren bieten mehr Flexibilität. Bei vielen Systemen haben die Kästen selbst einheitliche Größen, aber unterschiedliche Inneneinteilungen. Das erleichtert die Aufbewahrung und ermöglicht ein sinnvolles Stapeln. Als besonders praktisch erweisen sich die Boxen der Firma Allit¹, die wahlweise mit oder ohne lose Einsätze in einem festen Rastermaß angeboten werden (Abb. 3).



Abb. 3: Boxen der Fa. Allit

Sie können für die Entnahme oder das Herausschütten gerade kleiner Teile aus dem Umgebände entnommen werden. Die kleinsten Einsätze sind für eine Handvoll seltener Kleinteile nicht allzu groß, also platzsparend. Die größten Einsätze fassen auch größere Stückzahlen, so lange die Teile selbst nicht zu groß sind. Schließlich lassen sich Leer-Boxen ohne jede Inneneinteilung mit besonders häufigen oder sperrigen Teilen füllen, so lange Menge und Volumen nicht zu groß werden. Andererseits gibt es die Einsätze in allen Größen auch lose und einzeln zu kaufen (Abb. 4).

¹ Hier verwendet: Boxen der Serie Europlus 37, z. B. [EAN 4005187572360](https://www.europlus37.com/)



Abb. 4: Einsätze

Durch dicht schließende, durchsichtige Deckel ist der Inhalt ohne Öffnen sofort zu erkennen und vor Staub geschützt. Das System hat allerdings auch Nachteile. So ist der Preis relativ hoch – je Box sind rund zehn Euro fällig. Vier von ihnen bieten etwas mehr als das Volumen einer fischertechnik-Box 1000. Und: es kommt nicht von fischertechnik.



Abb. 5: Sortierung in Allit-Boxen

Welche Lösung bewährt sich bei mir?

Nach vielen Überlegungen habe ich mich für ein System mit Ausnahmen entschieden. Der weitaus größte Teil der Sammlung wird in Allit-Boxen mit losen Einsätzen aufbewahrt. Das ist sehr komfortabel und sicher. Wenn ein Einsatz an seine Kapazitätsgrenze stößt, wird der Inhalt in einen nächstgrößeren umgeschüttet; es muss nicht die gesamte Box entleert werden. Innerhalb der Boxen werden funktionell verwandte Teile zusammengeführt, zum Beispiel Zahnräder,

Statik-Teile, Motoren und Getriebe usw. Im Prinzip gibt es für jedes Einzelteil ein eigenes Fach, die Farben bleiben gemischt (Abb. 5).

Für einige wenige Teile, die für die Allit-Kästen zu sperrig (Alu-Profile, Netzteile usw.) oder die voluminös und zugleich in großer Zahl vorhanden sind (Statikträger 120 oder auch Bausteine 30), nutze ich derzeit fünf Boxen 1000, zumeist ohne jede Inneneinteilung. Ein Etikett auf der Außenseite verrät, was innen drin ist. Für drei Boxen habe ich staubdichte Deckel aus Holz angefertigt (Birke Multiplex, umlaufenden Falz eingefräst, lasiert).

Wächst die Sammlung weiter, müssen nicht alle Teile umsortiert werden. In vielen Einsätzen ist noch Platz. Und wenn ich einen neuen Kasten habe, wird er zunächst in zwei bis drei Allit-Boxen sortiert, kräftig ausprobiert und dann integriert. Im Prinzip könnte man sogar wie früher die Kästen getrennt lassen, was aber sehr viel Platz und Boxen benötigt und für das Bauen jenseits der Anleitungen auch nicht sinnvoll ist (Abb. 6).



Abb. 6: Ablage nach Baukästen getrennt

Übrigens: Auch ich habe es zunächst als fischertechnik-Purist versucht und ganz auf Boxen 1000 mit originalen Einsätzen sowie zusätzliche Boxen 500 gesetzt. Als sich dann 40 kleine Sortierwannen und nochmals fast so viele Boxen 500 um mich herum sammelten und ich vor lauter Ordnung gar nichts mehr fand bzw. die Fläche

auf dem Fußboden zu klein wurde, habe ich das Thema nochmals aufgegriffen.

Kriterium	Kartons o. ä.	Boxen 500, 1000	System Allit
leicht stapelbar	-	+	+
einfach zu erweitern	+	+	+
flexibel	-	o	+
stabil	o	+	+
staubgeschützt	o	o	+
gut zu beschriften	o	-	+
ins System integriert	-	+	-
gut einsehbar	o	-	+
Teile leicht entnehmbar	o	+	+
Kosten	+	o	-
Platz für Großmengen	+	+	-

Tab. 1: Gegenüberstellung der Lösungen

Die von mir gefundene Lösung passt für mich ganz gut, das einzige wirkliche Manko sehe ich in den Kosten. Allerdings kann man sie auch ins Verhältnis zum Wert des Inhalts setzen und bedenken, dass es nach dem Einstieg günstiger wird. Mit zehn Duo-Boxen inklusive Einsätzen (ergibt auseinander genommen 20 einzelne Boxen) ist ein prima Grundstock geschaffen. Die historischen Kästen (von 50 bis em-3) habe ich übrigens aufgehoben (was dann wiederum aber gar nicht platzsparend ist).

Wohin kommen all die Sortierkästen?

Im zweiten Schritt lohnt es sich darüber nachzudenken, wo und wie die Boxen untergebracht werden. Lediglich auf oder in einem Schrank/Regal gestapelt sind die

unteren Boxen nicht sehr gut zugänglich. Die Konstruktion² der von mir gewählten Sortierboxen macht es aber möglich, sie sehr einfach und ohne einzelne Fachböden in Korpusse von Regalen oder Schränken zu integrieren. Die Boxen werden dazu nur einhängt. Elegant geschieht dies mit einer ins Holz eingefrästen Nut, einfach mit ausreichend weit überstehenden Schrauben oder aufgesetzten Leisten. Den handwerklichen Ambitionen sind dabei keine Grenzen gesetzt. Die Höhe der Boxen von knapp 60 mm legt es überdies nahe, das im Möbelbau weit verbreitete Lochreihen-Raster von 32 mm zu nutzen; es gibt dafür verschiedene Bohrschablonen.

Muss man für den Bau eines fischertechnik-Möbels ein Profi sein?

In jeden Fall ist es sinnvoll, einen Korpus selbst zu bauen, da schon viel Glück dazu gehört, ein fertiges Regal oder einen Schrank von exakt passender Breite und Tiefe im Handel zu finden. Außerdem bietet dies die Chance, den Teilevorrat platzsparend und geschickt in die eigene Wohnsituation einzupassen. Für ein solches Projekt braucht es auch gar nicht unbedingt eine umfassend ausgestattete Werkstatt.



Abb. 7: fischertechnik-Regal in Einfachbauweise

Um die Bandbreite des Möglichen aufzuzeigen, habe ich einerseits mit einfachsten

² Siehe [genaue, bemaßte Zeichnung](#) bei den Unterlagen zum Nachbau von fischertechnik-Möbeln.

Mitteln einen Regalaufsatz für 14 einzeln entnehmbare Sortierboxen (Abb. 7) und andererseits ein ganzes fischertechnik-Möbelstück in Form eines Rollcontainers gebaut (Abb. 8).



Abb. 8: fischertechnik-„Rollcontainer“

Der Container ist im Bau durchaus komplex und erfordert einige Erfahrung im Möbelbau sowie diverse Maschinen. Dafür bietet es nicht nur Stauraum, sondern auch eine Arbeitsfläche. Wird ausnahmsweise gerade nicht mit fischertechnik gebaut, lässt sich der Container einfach in eine Ecke schieben.

Wie kann ich das Aufbewahrungssystem nachbauen?

Da die ft:pedia keine Heimwerker-Zeitschrift ist, verzichte ich an dieser Stelle auf umfassende Angaben zum Bau, Stück- und Werkzeuglisten. All diese Informationen sind aber in [Galerien](#) und [pdf-Dokumenten](#) auf der community-Seite zu finden. Dort finden sich auch Angaben, wie man sich den Bau mit einfachen Mitteln erleichtern kann.

Dass und wie sich die Baupläne für den eigenen Bedarf adaptieren lassen, erkennen fischertechnik-Fans mit ihrem gut geschulten räumlichen Anschauungsvermögen selbst. Vor allem die in Materialaufwand, Bauzeit und Maschinen- und Werkzeugbedarf sehr einfache, aber enorm zweckmäßige Variante hat in meinen Augen das

Zeug, eine weite Verbreitung zu finden. Zumal in einem solchen Möbel nicht nur fischertechnik einen guten Platz findet und schnell zugänglich ist.

Zusammenfassend ist festzuhalten, dass es sicherlich mehr Spaß macht, im Teilevorrat *kein* Chaos zu haben. Dafür bietet fischertechnik selbst gute Lösungen, es gibt aber auch sehr sinnvolle systemfremde Möglichkeiten. In jedem Fall hat eine gute Aufbewahrung ihren Preis. Eine schöne fischertechnik-Sammlung, die nur in losen Schachteln herumliegt und bei der Bauprojekte wegen der Angst vor ewiger Sucherei erst gar nicht in Angriff genommen werden, ist allemal die teuerste Variante.

Daher lohnt es auch, nicht nur über Sortierboxen, sondern auch über den Bau eines eigenen Möbelstücks nachzudenken. Die Arbeit daran macht Spaß, und wenn alles fertig ist, wird das Bauen mit fischertechnik attraktiver als es zuvor je war. Schließlich ist es eine Überlegung wert, ob sich nicht einige Fischertechniker unter der Leitung eines im Holzwerken erfahrenen Gleichgesinnten zusammenschließen. Sie könnten dann für ein Wochenende eine der vielerorts angebotenen Mietwerkstätten nutzen und gemeinsam ein neues Zuhause bauen für all die kleinen Bausteine, die unser Leben so sehr bereichern.

Referenzen

- [1] Henn, Guido (2013): *Handbuch Elektrowerkzeuge*. Hannover: Vincentz.
- [2] Rae, Andy (2013): *Möbelbau: Grundlagen, Konstruktionen, Tricks & Kniffe*. Hannover: Vincenz.

Sehr lesenswerter Blog von Heiko Rech zum Holzwerken: www.holzwerkerblog.de

Ein einfacher Rollcontainer für eine andere Art von Sortiment wird hier vorgestellt: www.holzundleim.de

Pneumatik

Druckluftsteuerungen (Teil 4)

Stefan Falk

In dieser Folge der Druckluftsteuerungen geht es um mit Druckluft betriebene Motoren. Wir stellen bereits gebaute Motoren kurz vor und bauen dann selbst einen kompakten Druckluftmotor, der ausschließlich mit aktuellen fischertechnik-Teilen auskommt.

Die Mutter aller Motoren

Die ersten Motoren, die ihre Kraft aus Druckluft schöpften, waren zweifellos die Dampfmaschinen. Schon Anfang des 18. Jahrhunderts gab es die ersten praktisch verwendbaren Exemplare davon [1, 2, 6].

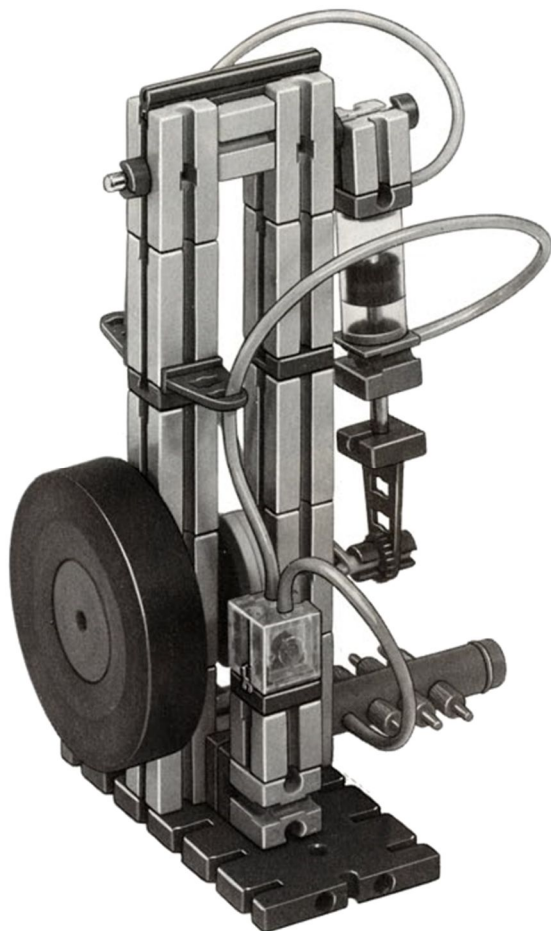


Abb. 1: Druckluftmotor aus der Anleitung zum Baukasten „Pneumatik“ von 1981 [3]

Auch im fischertechnik-System durften druckluftbetriebene Motoren natürlich nicht fehlen, als 1981 der erste Pneumatik-Kasten herauskam. Eines der durchaus sehr verschiedenen Modelle aus dessen Bauanleitung zeigt Abb. 1 (die Bauanleitung ist unter [3] verfügbar).

Es handelt sich um einen einzylindrigen, aber doppelwirkenden Motor. Das bedeutet, dass vom Pneumatikzylinder beide Anschlüsse verwendet werden. Er stößt also das Exzenter sowohl aktiv nach unten, zieht es aber auch aktiv wieder hoch. Kenner entdecken sofort das Festo-Pneumatikventil [4], von dem es in diesem Modell zwei gibt (eines ist auf der Rückseite angebracht). Wie wir sehen, konnte man so einen funktionierenden Druckluftmotor wirklich kompakt und mit nur wenigen Bauteilen herstellen.

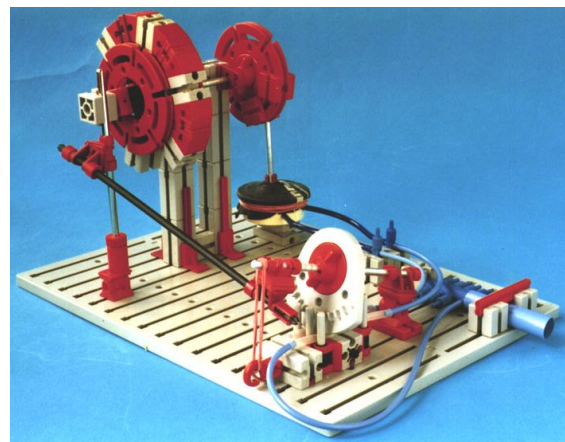


Abb. 2: [Druckluftmotor](#) von Alfred Pettera

Einige weitere interessante Konstruktionen zeigt auch die Kategorie *Membranpumpen und Luftmotoren* auf der Webseite der ft:Community [5]. Zwei Modelle möchte ich hervorheben, weil sie gänzlich ohne fischertechnik-Ventile auskommen. Das erste ist von Alfred Pettera aus dem Jahr 2003, verwendet einen Selbstbau-Zylinder und steuert die Druckluft über das Abquetschen von Schläuchen (Abb. 2).

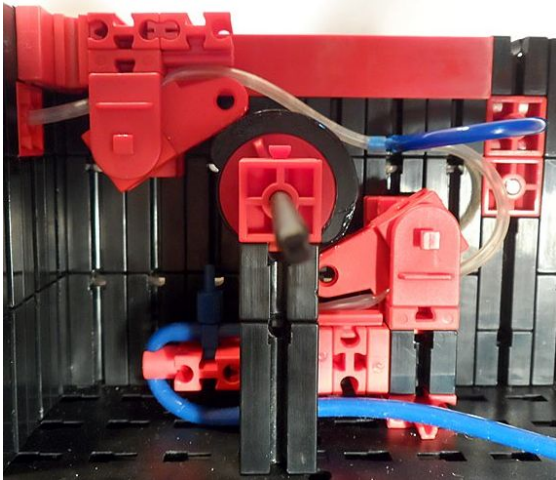


Abb. 3: Schlauch-Quetsch-Steuerung von Stefan Reinmüller

Der zweite (Abb. 3) verwendet ebenfalls das Zuklemmen von Schläuchen als Ventilersatz und wurde von Stefan Reinmüller in 2015 gebaut. Zu diesem Motor ist dort übrigens auch ein toll gemachtes Video verlinkt.

Das Modell

Da wir in der ft:pedia schon so gute Erfahrungen mit dem *Abknicken* von Schläuchen gemacht haben, stellen wir heute einen Druckluftmotor vor, der sich dieser Technik bedient. Eine Einführung in diese Ventilart zeigen die vorangegangenen Beiträge dieser Artikelserie in den ft:pedia-Ausgaben 1/2014 bis 3/2014.

Unser Motor hat folgende Eigenschaften:

- Er besitzt einen Zylinder.
- Der ist doppelwirkend betrieben, drückt und zieht also aktiv.
- Er verwendet ein selbstgebautes 4/2-Wegeventil mit Schlauchabknick-Technik.

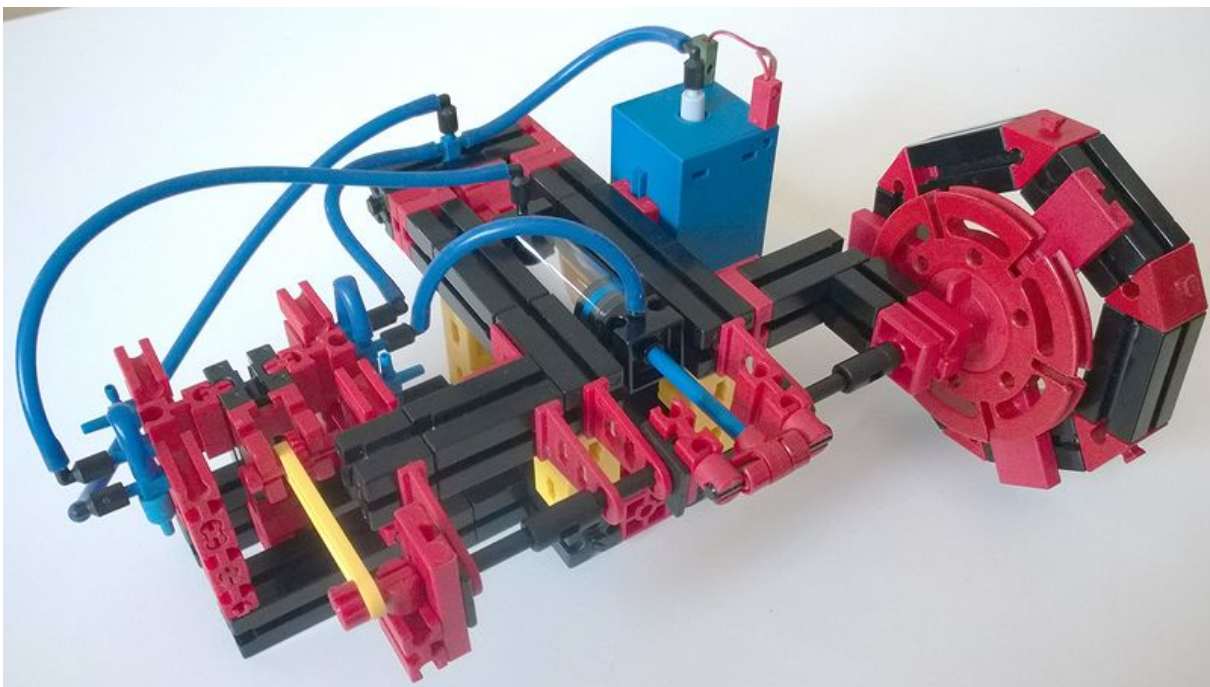


Abb. 4: *Doppeltwirkender Druckluftmotor mit Abknickventilen*

Das Knicken hat den Charme, dass das Ventil recht leichtgängig bleibt. Im Gegensatz zum Abquetschen eines Schlauches benötigt das Abknicken nur geringe Kraft. Über den zum Umschalten benötigten Weg hinweg gesehen ergibt das eine ordentliche Leistung, die durch die Leichtgängigkeit des Ventils gespart wird und also dem Abtriebsausgang des Motors zur Verfügung steht.

Das Modell besteht aus folgenden Komponenten:

- Ein aus Standardbauteilen hergestellter Rahmen trägt sämtliche Mechanik. Das gesamte Modell kommt ausschließlich mit unveränderten, aktuell hergestellten Bauteilen aus.
- Ein waagrecht liegender Pneumatikzylinder treibt eine Kurbelwelle an.
- Auf derselben Welle sitzt der Exzenter für die periodische Umschaltung des 4/2-Wegeventils, welches den Zylinder ansteuert.
- Schließlich sorgt ein kleines Schwungrad für den gleichmäßigen Lauf der Maschine.

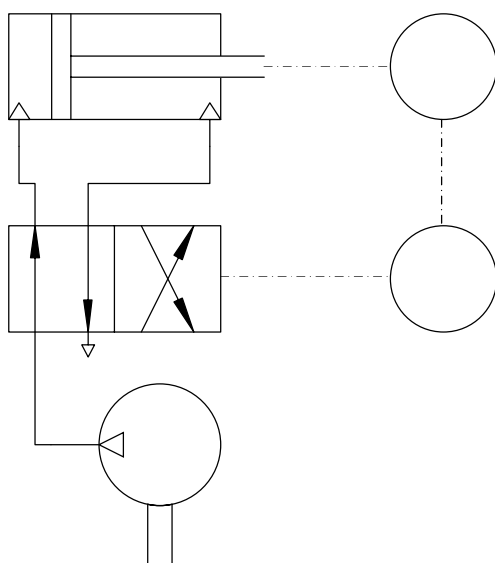


Abb. 5: Schaltplan

Die Wirkungsweise der Steuerung

Das Schaltbild in Abb. 5 macht deutlich, wie die Druckluftsteuerung realisiert ist.

Die beiden Kreise rechts stehen für Kurbelwelle und Steuer-Exzenter. Durch die Drehung der Kurbelwelle wird der Zylinder einfach immer zwischen Aus- und Einfahren umgesteuert.

Da uns ein 4/2-Wege-Ventil direkt nicht zur Verfügung steht, wird unser Selbstbauventil aus zwei identisch gebauten 3/2-Wegeventilen gebildet.

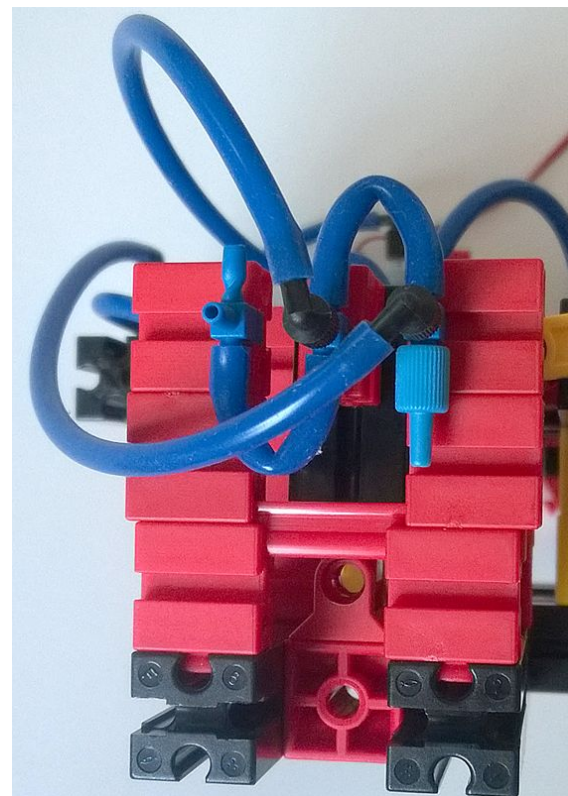


Abb. 6: Draufsicht auf das äußere 3/2-Wegeventil

Das bewegliche Element des Ventils ist ein Baustein 30 auf einem Gelenkstein. An dem ist für jedes der beiden 3/2-Wegeventile ein Baustein 7,5 angebracht, der ein Pneumatik-T-Stück trägt. Dessen mittlerer Anschluss ist per Schlauch mit einem der Zylinderanschlüsse verbunden.

Die beiden anderen Anschlüsse des beweglichen T-Stücks gehen mit je einem 30 mm langen Schlauchstück wie in den Abbildungen gezeigt zu je einem unbeweglich angebrachten T-Stück. Eines davon bleibt ansonsten einfach offen (das ist der Abluftausgang), das andere wird mit Druckluft versorgt und der dritte Anschluss seines T-Stücks mit einem P-Stopfen verschlossen. Somit haben wir eine Anordnung, die den Zylinder wahlweise mit Druckluft oder mit Abluft verbinden kann – das jeweils andere Schlauchstückchen ist durch Abknicken verschlossen.

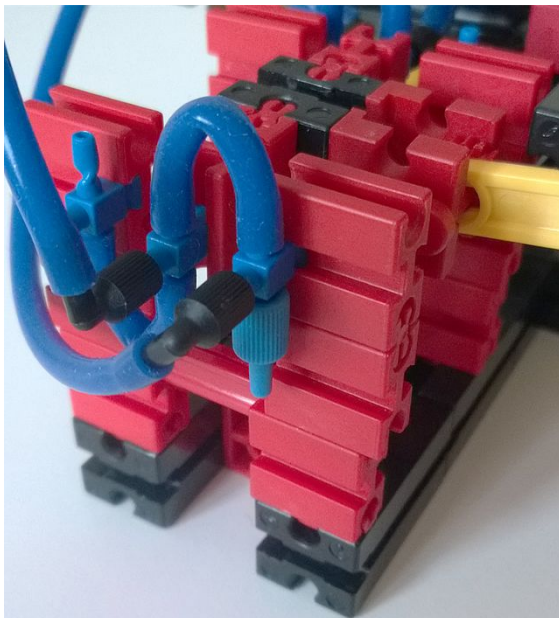


Abb. 7: Schrägansicht aufs äußere Ventil

Genau dieselbe Anordnung gibt es auch auf der anderen Ventilseite, nur – das ist der springende Punkt – sind hier die Zu- und Abluftanschlüsse vertauscht. Man beachte, dass auf den Aufnahmen beider Ventile die Zuluft von rechts kommt. Von der Gesamtansicht in Abb. 4 aus gesehen führt also beim linken Ventil der vordere und beim rechten Ventil der hintere Anschluss Druckluft. Die jeweils anderen Anschlüsse sind die offenen T-Stücke für den Abluftausgang.

Die „Türme“ von BS7,5 sind mit einem Verbinder 45 stabilisiert, die oberen BS7,5-Paare mit Verbindern 15 verbunden. So

können die Ventile genau justiert werden – dazu später mehr.

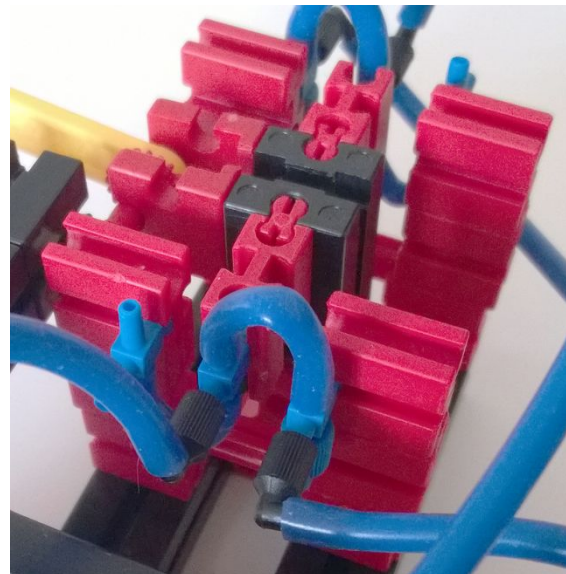


Abb. 8: Das innere 3/2-Wegeventil

Tatsächlich haben wir damit also folgende, zu Abb. 5 äquivalente Schaltung aufgebaut:

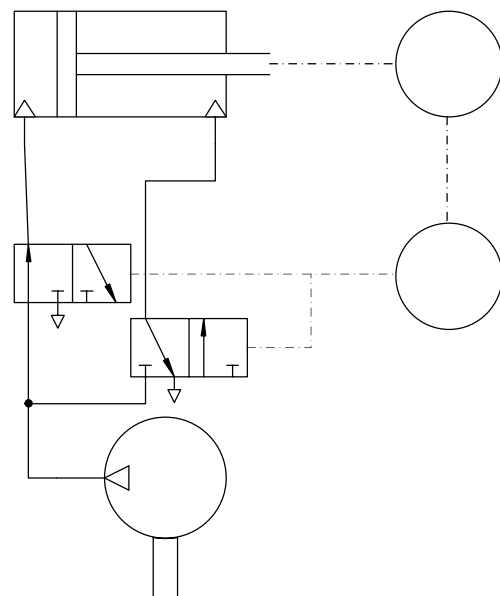


Abb. 9: Die tatsächlich realisierte Schaltung

Hinweise zum Bau des Modells

Rahmen

Wie Abb. 10 zeigt, hat der Zylinder in der Breite durch die BS5 hinreichend Platz, damit er nicht an den benachbarten

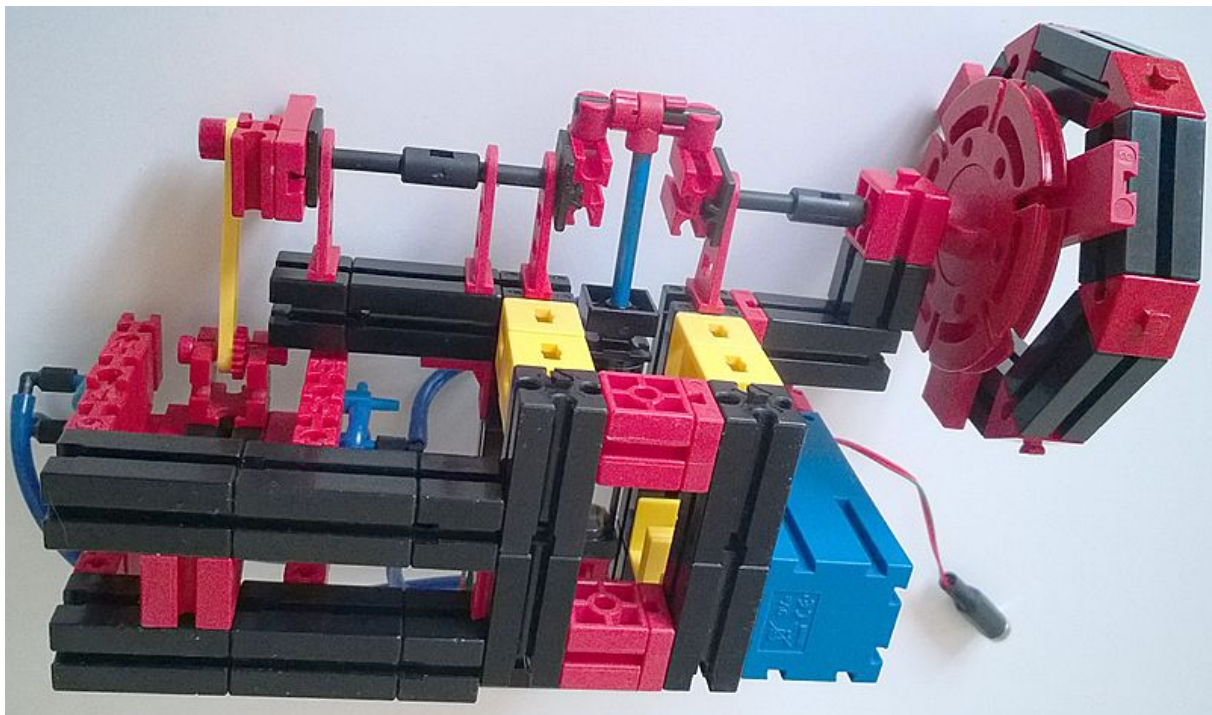


Abb. 10: Blick von unten auf das Modell

Bausteinen reibt. Nach links geht es weiter zur Befestigung des Ventils. Zwischen den beiden BS15 dort ist ein Verbindler 15 eingeschoben. Der erleichtert das Justieren der Ventilgruppe in der Nut des Rahmens unterhalb des Zylinders, weil er für einen konstanten Abstand der BS15 sorgt.

Die vertikale Ausrichtung der Nuten der beiden BS15 ist auch nützlich, um die gesamte Ventilgruppe anzusetzen oder auch wieder abzutrennen.

In Abb. 13 sieht man, wie der Rahmen dort, wo die Bausteine beim Zylinder nach links und rechts abgehen, mit [Winkelsteinen 10x15x15](#) (und Federnocken darin) gegen Verdrehen ihrer Zapfen verstärkt wurden. Der Kompressor kann einfach mit einem einzelnen BS5 am Rahmen befestigt werden.

Kurbelwelle

Der Zylinder greift an einer Achse 30 an und ist mittels zweier Klemmringe gegen Verrutschen gesichert. Die Kurbelwelle selbst besteht aus [Rastaufnahmeachsen](#)

[22.5](#). Sie tragen je einen BS7,5, und darauf [Klemmhülsen 7.5](#). Diese Konstruktion erlaubt das Justieren der Auslenkung so, dass der komplette Hub des Zylinders gerade vollständig ausgenutzt wird. Beide Seiten der Kurbelwelle müssen natürlich gleich justiert sein, damit die nach beiden Seiten abgehenden Achsen sauber fluchten.

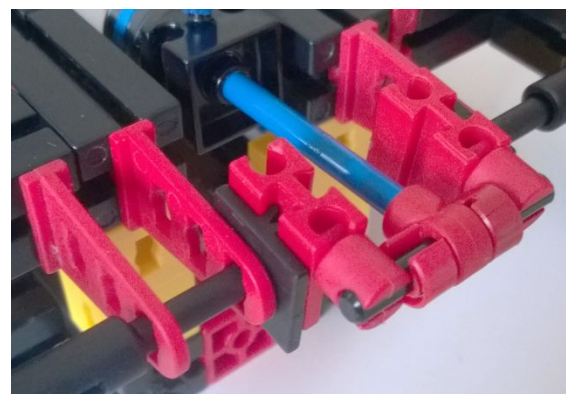


Abb. 11: Kurbelwelle

Exzenter für die Ventilsteuerung

In Richtung Ventil geht es auf der Achse weiter mit einer Rastkupplung und einer weiteren Rastaufnahmeachse. Auf der sitzt eine Kombination von [Baustein 5 15x30](#),

BS7,5 und wiederum einer Klemmhülse 7,5. Eine X-Strebe 42,4 sitzt auf einem [Seilklemmstift](#), der ganz von alleine fest in der Klemmhülse steckt. Auf der anderen Seite ist die Strebe mittels Klemmring gegen Herausrutschen geschützt.

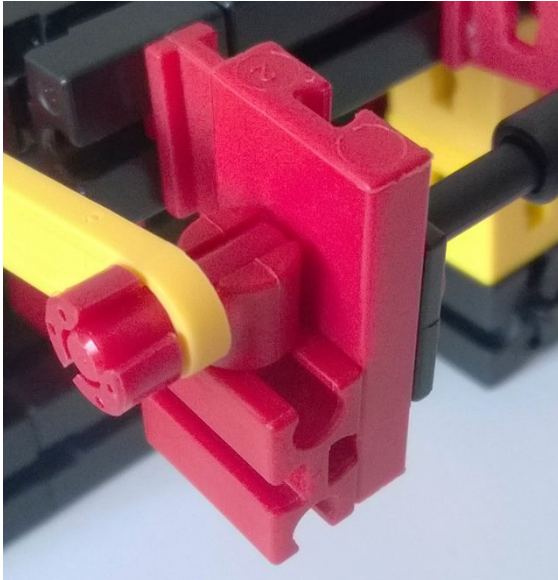


Abb. 12: Ventilsteuerungs-Exzenter

Schwungrad

Ein Ring von Bausteinen 30 und Winkelsteinen 60° dient als Schwunngmasse. In drei der Bausteine 30 wird je ein Federnocken so eingeschoben, dass der ganze Ring einfach mit drei Bausteinen 5 15x30, die zuletzt eingeschoben werden, an der Drehscheibe befestigt werden kann.

Justierung des Ventils

Damit das Modell gut läuft, empfehle ich, die Kurbelwelle in dieselbe Richtung zu justieren wie in Abb. 11 dargestellt. Kurbelwelle und Exzenter müssen so angeordnet werden wie in Abb. 4 gezeigt. Dann kann die Justage des Ventils beginnen:

- Zunächst wird die gesamte Ventil-anordnung durch Verschieben der BS15 (siehe Abb. 10) so eingestellt, dass der gelenkig gelagerte BS30 des Ventils senkrecht steht, wenn der Exzenter sich in Mittellage befindet.

- Der Exzenter wird durch Verschieben des Baustein 30 5x15 auf der Rast-aufnahmeachse so justiert, dass insgesamt etwa 7,5 mm (eine halbe Bausteinlänge) Auslenkung des Ventils erreicht werden.
- Anschließend werden die vier Anordnungen von BS7,5 und T-Stücken so justiert, dass beim Drehen der Kurbelwelle von Hand die beweglichen Teile des Ventils gerade nicht anstoßen. Würden sie das, würde der Motor dadurch an Kraft verlieren. Ist der Abstand zu groß, werden die Schlauchstückchen nicht scharf genug abgeknickt.

Wenn alles justiert ist, kann's losgehen! Der Motor dreht sehr ordentlich und läuft in vielen Fällen sogar selbst an. Für alle Fälle steht ein [Video](#) bereit. Das ersetzt aber selbstverständlich keinesfalls den Selbstbau des Motors und das Experimentieren damit.

Quellen

- [1] Fox, Dirk: *Die Dampfmaschine*. [ft:pedia 4/2012](#), S. 38-45.
- [2] Wikipedia: [Dampfmaschine](#).
- [3] fischertechnik: *Pneumatik. Baukasten und Anleitung*. fischerwerke, 1981.
- [4] Falk, Stefan: *Perlentauchen (Teil 5)*. [ft:pedia 4/2013](#), S.
- [5] ft:Community: [Membranpumpen und Luftmotoren](#).
- [6] Fox, Dirk und Püttmann, Thomas: [Bauen, erleben, begreifen: Technikgeschichte mit fischertechnik. 16 Meilensteine zum Nachbauen](#). dpunkt-Verlag, 2015.

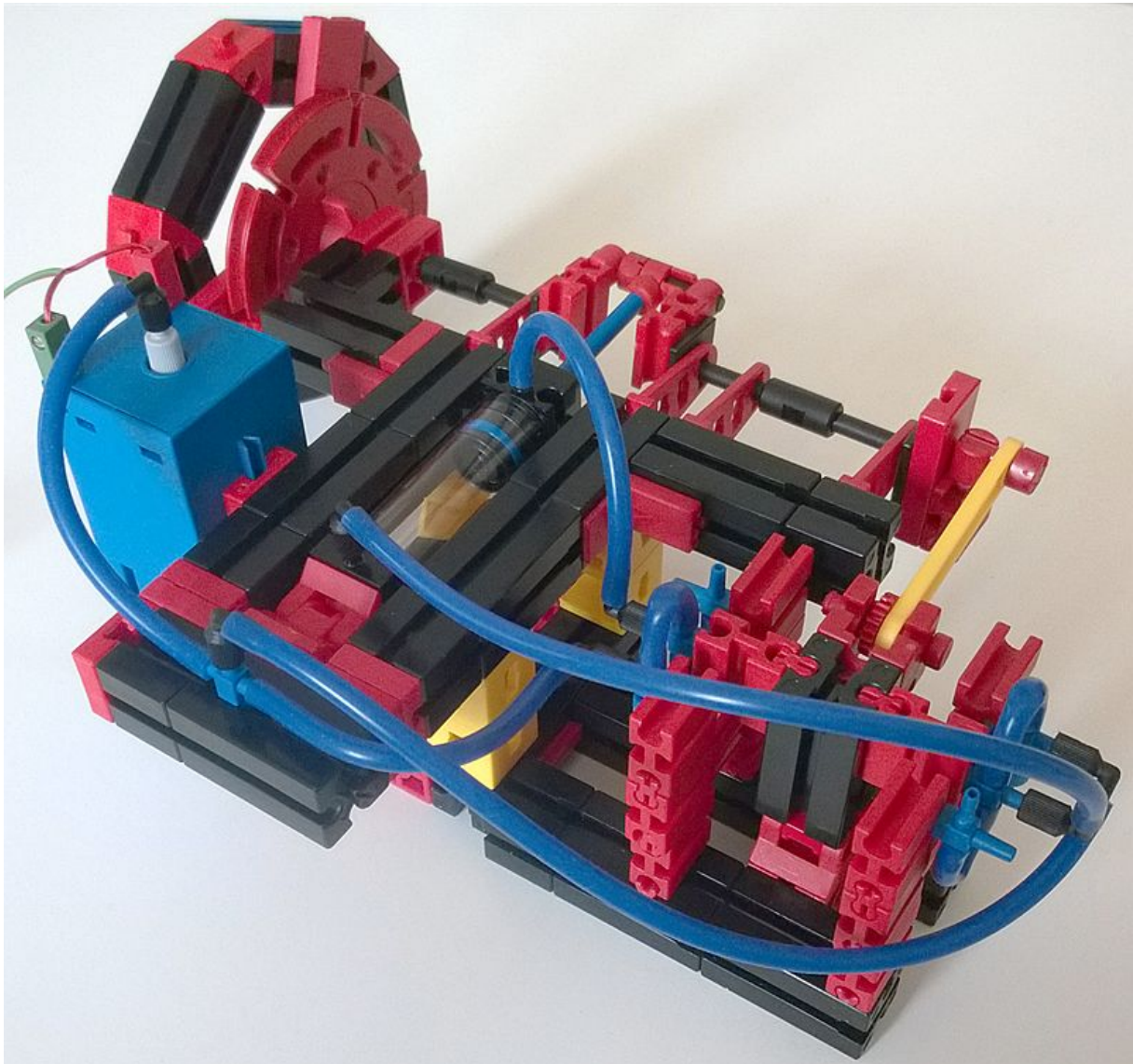


Abb. 13: Rückansicht des Modells

Elektronik

PWM-Tongenerator für Robo TX(T) Controller

Andreas Gail

Einfach nur Töne zu erzeugen, ist mit den guten alten „Silberlingen“ kein Problem. Aber in der Welt der Microcontroller, die unendlich erscheinende Möglichkeiten bietet, haben wir mit der Erzeugung einfacher Töne Probleme. Das gilt auch für den neuen fischertechnik Robo TXT Controller.

Die Welt der „Silberlinge“

Die „Silberlinge“ wie in Abb. 1 gezeigt waren gemäß Internet [1] ab 1971 auf dem Markt und zuerst im Baukasten hobby 4 enthalten. Damit konnte man Töne in einem weiten Frequenzbereich erzeugen.

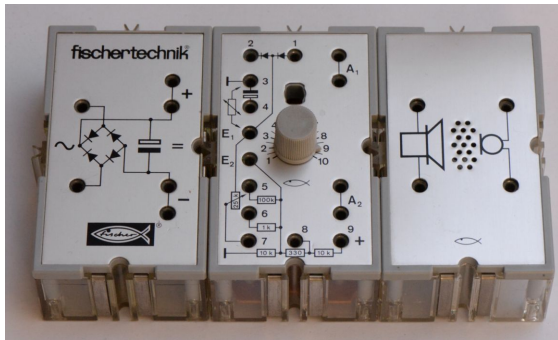


Abb. 1: „Silberlinge“ zum Aufbau eines Tongenerators

Die Microcontroller

Mit Beginn der Microcontroller schien es, als hätten die Silberlinge ausgedient. Die Tonerzeugung mit dem Robo TX Controller alleine war aber nicht möglich: Der TX hatte keinen Lautsprecher. Der Nachfolger Robo TXT Controller wurde zwar mit einem Lautsprecher ausgestattet, aber die Möglichkeiten der Tonerzeugung sind stark eingeschränkt. Der TXT kann zwar einige wav-Sounds abspielen, aber ohne Weiteres nur die von fischertechnik vorgegebenen. Obwohl es möglich ist, eigene wav-Dateien in den TXT einzuspielen, ist es zumindest

mit der fischertechnik Robo Pro-Software bis zur Version 4.1.8 nicht gelungen, diese auch abzuspielen (Stand 09/2015).

Anforderungskatalog

Für eine mögliche Zusatzhardware gibt es folgende allgemeine Anforderungen (Lastenheft):

- Die Nutzung der „Silberlinge“ kommt nicht in Betracht, da diese nur noch schwer zu bekommen sind.
- Der Nachbau soll möglichst einfach und kostengünstig möglich sein.
- Die Ansteuerung soll über Robo Pro möglich sein.
- Sounds sollen mit maximaler Geschwindigkeit begonnen oder auch abgebrochen werden können.

Anschluss an den TX/TXT

Zum Anschluss der neuen Zusatzhardware an den TX oder TXT bieten sich der I²C-Bus und die normalen Ausgänge für Motoren, Lampen usw. an. Nach einigen Vorversuchen wurden die normalen Ausgänge O1...8 zur Ansteuerung der neuen Zusatzhardware ausgewählt. Um zumindest diesen Punkt vorweg zu nehmen: Zur Erzeugung von vier Oktaven werden beim nachfolgenden Aufbau nur zwei Ausgänge (z. B. O1 und O2) benötigt.

Leistungseinstellung per PWM

Gemäß Abb. 2 hat der TX acht Ausgänge (O1...O8). Als Beispiel können hier angeschlossene Lampen in ihrer Helligkeit mithilfe der Robo Pro-Software gemäß Abb. 3 eingestellt werden. Die Einstellung bietet dabei maximal 512 Stufen.



Abb. 2: fischertechnik Robo TX Controller

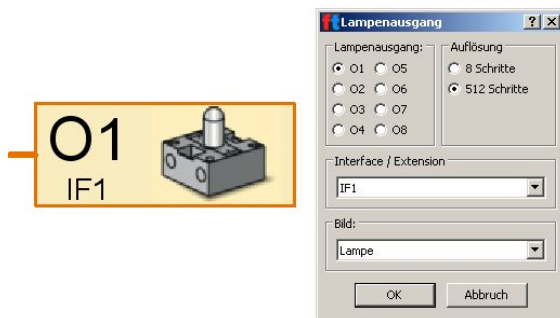


Abb. 3: Leistungseinstellung eines Ausgangs beim TX/TXT mithilfe der Robo Pro-Software von fischertechnik

Nun könnte man auf die Idee kommen, dass an einem solchen Ausgang eine Gleichspannung anliegt. Doch so einfach ist das leider nicht. In Wirklichkeit ist dieser Ausgang gepulst, und das stets mit der annähernd vollen Betriebsspannung des Controllers. Das Oszilloskop bringt es ans Licht (Abb. 4 und 5).

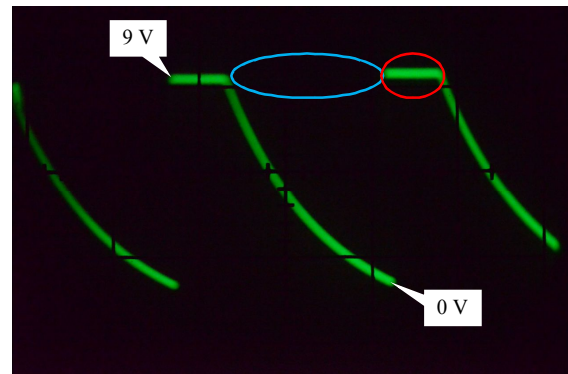


Abb. 4: Pulsweitenmoduliertes Ausgangssignal (PWM) des TX, Leistungseinstellung 252

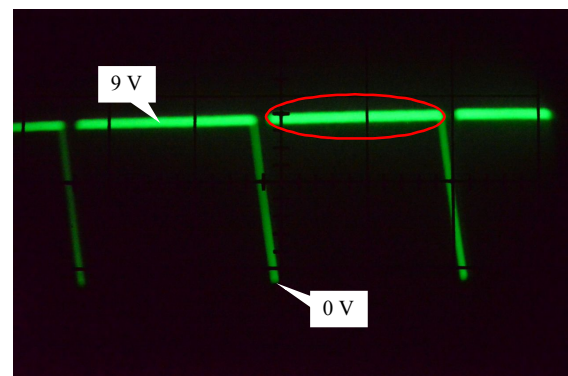


Abb. 5: Pulsweitenmoduliertes Ausgangssignal (PWM) des TX, Leistungseinstellung 498

Die oben genannte Abkürzung PWM steht für Pulsweitenmodulation. Die Grundidee dabei ist, dass bei gleichbleibender Frequenz am Ausgang abwechselnd 0 V oder z. B. 9 V anliegen. Mit der vorhandenen Messtechnik wurden 210 Hz gemessen. Die Leistungseinstellung erfolgt durch das Verhältnis der Dauer von anliegenden 0 V und 9 V. Gemäß Abb. 5 ist die 9 V-Dauer relativ lang (rot markiert) im Vergleich zur 0 V-Dauer. Damit ist die am Ausgang vorhandene Leistung relativ hoch und eine angeschlossene Glühlampe würde relativ hell leuchten. Wird im Gegensatz dazu Abb. 4 betrachtet, ist die 9 V-Dauer relativ kurz (rot markiert) und die 0 V-Dauer relativ lang (blau markiert). Deshalb würde eine angeschlossene Glühlampe dunkler leuchten als bei Abb. 5.

Aus dem PWM-Signal werden Töne

Die verwendete Schaltung wird in Abb. 16 gezeigt. Die Spannungsversorgung der Schaltung erfolgt über die Anschlüsse JP1 und JP2. Der Brückengleichrichter B1 dient als Verpolungsschutz der Schaltung. Um aus dem über JP3 eingespeisten PWM-Signal Töne werden zu lassen, wird dieses Signal zunächst in eine variable Spannung umgewandelt, dessen Spannungshöhe abhängig von der Leistungseinstellung des Controllers ist. Die Bauteile R1...3, C2...3 und IC2A (Conrad #174440) sind für die Gleichspannungserzeugung zuständig. In Abhängigkeit von dieser Spannung wiederum wird eine variable Frequenz erzeugt. Dies erledigt IC1 (Conrad #152184), als *Voltage Controlled Oscillator (VCO)* betrieben [2, 3]. An JP4 und JP5 kann direkt ein Lautsprecher mit 8 Ω , 0,2 W angeschlossen werden.

Schaltungsaufbau

Für vergleichsweise einfache Schaltungen bietet sich die Verwendung einer Lochrasterplatine an. Diese muss dann nur noch auf die passende Größe zugeschnitten werden. Im vorliegenden Fall ist das jedoch bereits ziemlich grenzwertig. Die einzelnen Schritte werden nachfolgend gezeigt.

Zunächst werden die Bauteile zusammengestellt. Daraus ergibt sich der Montageplatzbedarf auf der Platine. Die Bauteile sind in Abb. 6 grün markiert. Die Leiterbahnen sind durch Striche gekennzeichnet und die gelb markierten Flächen sind für Bohrlöcher reserviert. Damit wäre die Anordnung geklärt. Das ist einfach gesagt, es kann jedoch durchaus eine Zeit lang dauern, bis alles mit möglichst wenig Leiterbahnenkreuzungen auf die Platine passt. Um die Anordnung schneller hinzubekommen, ist das Programm EAGLE von CadSoft [7] als Freeware für nicht-kommerzielle Anwendungen sehr zu empfehlen.

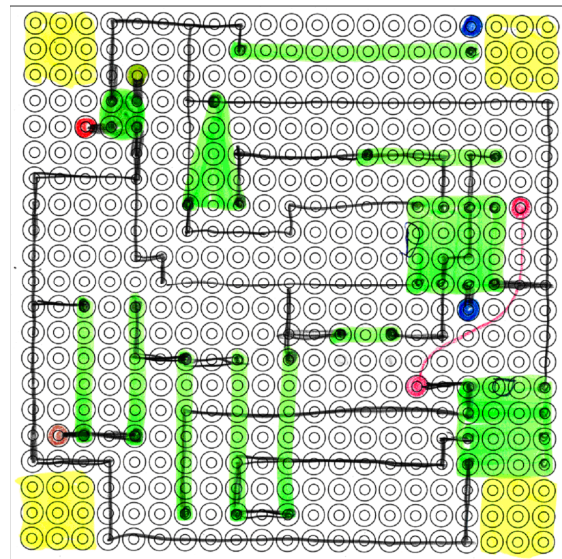


Abb. 6: Planung der Anordnung der Bauteile und Leiterbahnenführung (Ansicht von oben)

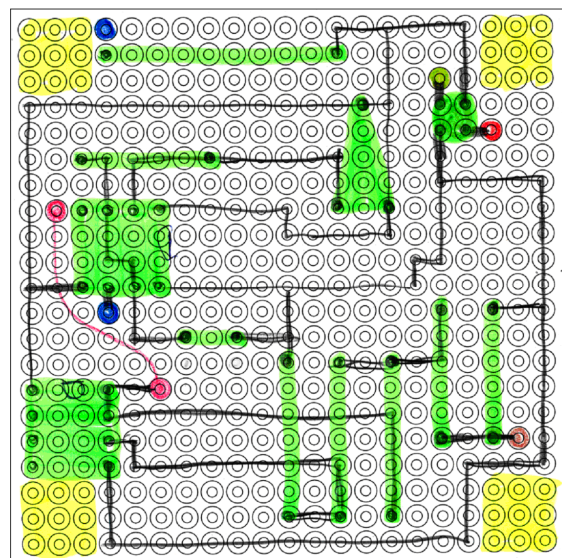


Abb. 7: Planung der Anordnung der Bauteile und Leiterbahnenführung (Ansicht von unten)

Der Unterschied zwischen Abb. 6 und Abb. 7 ist nur der spiegelverkehrte Ausdruck. Das ist ein kleiner, aber wichtiger Unterschied; die Leiterbahnen müssen unbedingt gemäß Abb. 7 hergestellt werden. Um die Leiterbahnen auf die Platine zu bekommen, wurde im vorliegenden Fall gemäß Abb. 8 ein 8 W-LötKolben verwendet.

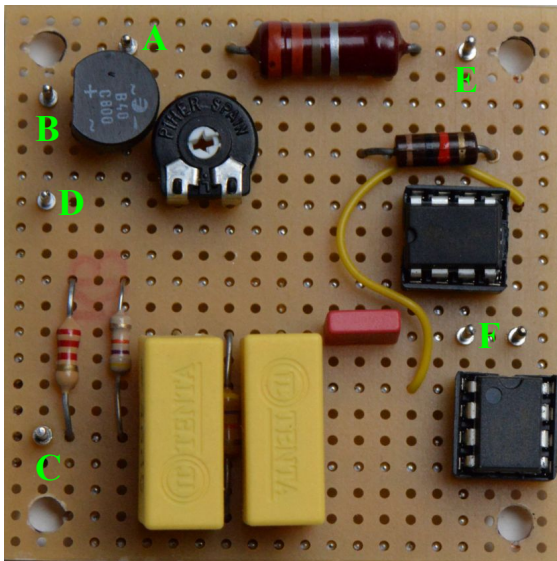


Abb. 8: fertig aufgebaute Schaltung, A: JP1 DC 9 V, B: JP2 DC 9 V, C: JP3 PWM Signaleingang, D: JP4 0 V, E: JP4 Lautsprecher, F: JP5 Lautsprecher

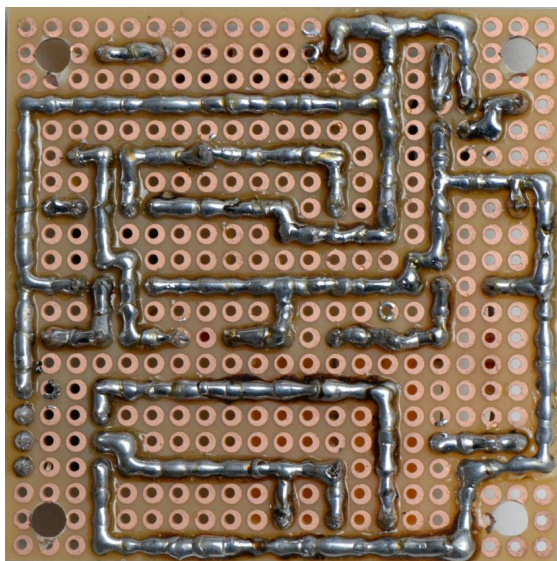


Abb. 9: Leiterbahnen auf der Lochrasterplatine

Von Robo Pro zum gewünschten Ton

Wie können mit der aufgebauten Schaltung gezielt gewünschte Töne erzeugt werden?

Zunächst muss über das Potentiometer der Arbeitsbereich eingestellt werden, in dem die Töne erzeugt werden sollen. Hierbei hilft Abb. 18. Nachfolgend erfolgt eine Einstellung im Bereich von 65...988 Hz (MIDI 36...83). Hierzu muss eine minimale

und maximale Leistungseinstellung am Ausgang des TX vorgenommen werden, bei gleichzeitiger Frequenzmessung am Lautsprecher. Danach darf der Poti nicht mehr verstellt werden.

Sind diese Vorarbeiten erledigt, wird eine Wertetabelle aufgenommen. Hier wird die Leistung am TX so eingestellt, dass man eine gegebene Tonfrequenz im gesuchten Bereich gemäß Abb. 15 erhält. Der zugehörige MIDI-Code [4] und PWM-Wert werden notiert. Schließlich erhalten wir Tab. 1.

Mathematik spart Arbeit

Wer es nicht glaubt, dem sei Nachfolgendes besonders empfohlen. Die Sache selbst ist eigentlich recht schnell getan, seitdem es Computer oder das Internet gibt.

Die Aufgabe ist es, eine Möglichkeit zu finden, einen gegebenen MIDI-Code in eine PWM-Einstellung umzurechnen [5]. Dazu werden die Daten aus Tab. 1 gemäß der Darstellung in Abb. 17 als rote Punkte ins Diagramm eingetragen. Die Verwendung von Microsoft Excel bietet die Möglichkeit, eine Ausgleichskurve (schwarz) automatisch zu berechnen. Alternativ kann die Berechnung auch über die Internetseite [6] durchgeführt werden. Schließlich erhält man eine Funktionsgleichung:

$$\begin{aligned}
 y = & 0,000010957678355 \cdot x^5 \\
 & - 0,002954323848880 \cdot x^4 \\
 & + 0,311229974344280 \cdot x^3 \\
 & - 16,257777205721100 \cdot x^2 \\
 & + 420,858828728566000 \cdot x \\
 & - 3838,610375204770000
 \end{aligned}$$

Diese Gleichung sieht kompliziert aus. Sie ist es aber nicht, solange man sie nicht mit Papier und Bleistift berechnen muss. Die vielen Kommastellen sind übrigens durchaus von Bedeutung, auch wenn die Anzahl der Stellen nicht für die Anwendung optimiert wurde. Der Vorteil dieser Gleichung ist, dass nun für jeden MIDI-Code eine PWM-Einstellung errechnet werden kann.

Bei den bisherigen Messungen wurden die MIDI-Codes der Halbtöne weglassen.

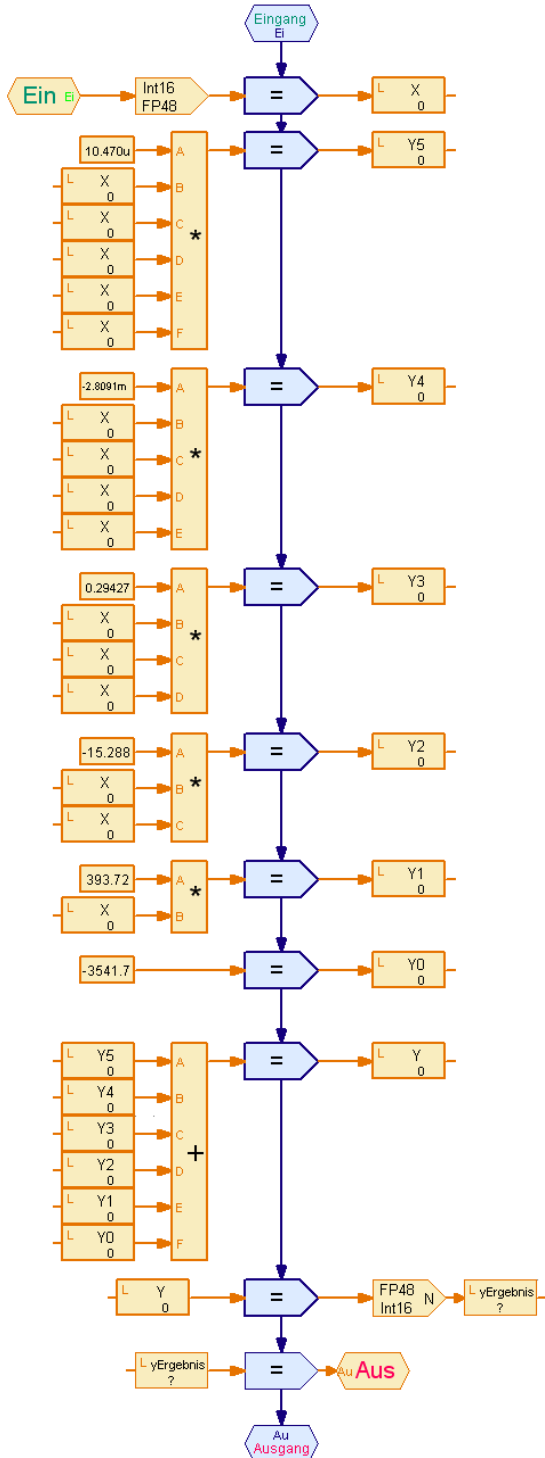


Abb. 10: Unterprogramm zur Berechnung der PWM-Einstellung aus einem gegebenen MIDI-Code

Funktionsgleichung in Robo Pro

Robo Pro ist eigentlich keine Programmierumgebung mit speziellem Fokus auf die Lösung mathematischer Aufgaben. Deshalb ist die Funktionsberechnung etwas aufwändig. Aber es klappt – siehe Abb. 10.

Nun ist es möglich, den TX frei programmierbare, einstimmige Musikstücke spielen zu lassen. Dazu können z. B. Listenfelder gemäß Abb. 11 verwendet werden. Für jeden zu spielenden Ton muss die Variable *Zaehler* um 1 erhöht und die Listenfelder ausgelesen werden, die PWR-Umrechnung gemäß Abb. 10 erfolgen und schließlich das Setzen eines Ausgangs, z. B. O1, auf den errechneten Wert durchgeführt werden.

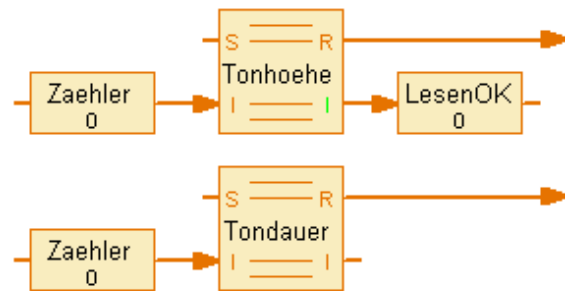


Abb. 11: Speicherung von Musikstücken innerhalb von RoboPro

Das fertige Modul

Die oben gezeigte Platine hat gemäß Abb. 8 eine Reihe an Steckerstiften. Über die zugehörigen Steckschuhe erfolgt für die Gehäusemontage die Verkabelung an die folgenden nach außen geführten Anschlüsse:

- 2 × Eingang Versorgungsspannung 9 V DC mit Verpolungsschutz
- 1 × Eingang PWM Signal
- 1 × Ausgang 3,5 mm Audio-Klinke (mono)

Abb. 14 zeigt das geöffnete Gehäuse. An den Ecken ist jeweils eine von unten durchgesteckte Schraube mit einem aufgeschraubten Abstandsbolzen (Langmutter)

zu sehen. Hierüber erfolgt schließlich die Montage des Deckels.



Abb. 12: PWM-Tongenerator,
Gehäuse von oben



Abb. 13: PWM-Tongenerator,
Gehäuse von unten

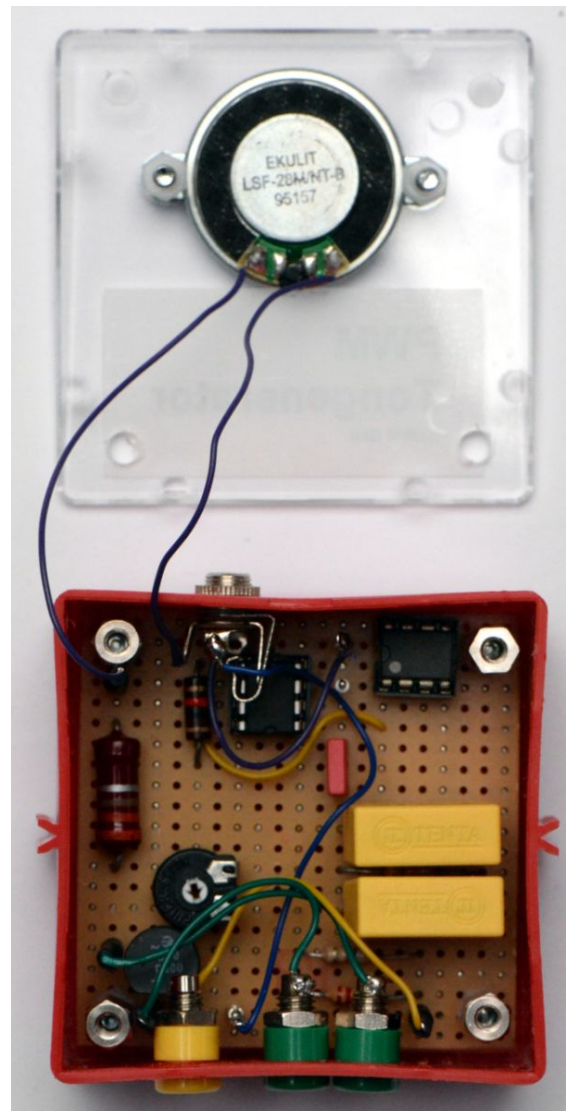


Abb. 14: PWM-Tongenerator,
Gehäuse geöffnet

Anschluss und Verwendung

Der gelbe Signaleingang wird an einen Ausgang des TX angeschlossen. Über diesen Anschluss wird die Tonhöhe eingestellt. Ein grüner Eingang zur Spannungsversorgung wird mit einem anderen Ausgang des TX verbunden. Somit wird die Spannungsversorgung des PWM-Tongenerators einfach unterbrochen, wenn eine Tonpause erfolgen soll. Über den Audio-Ausgang kann ein Verstärker angesteuert werden.

Einen Eindruck von der Funktionsweise vermittelt ein kleines [Youtube-Video](#).

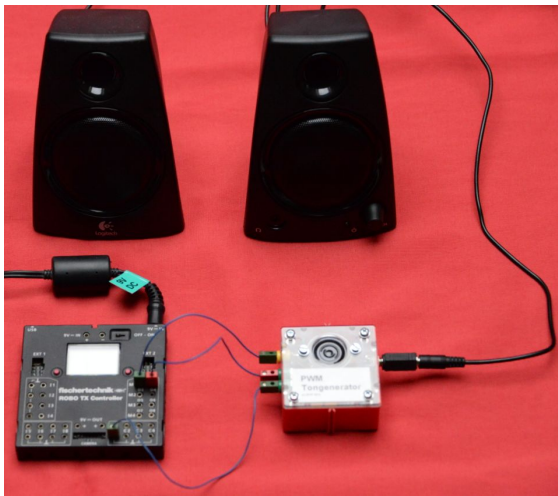


Abb. 15: PWM-Tongenerator, Gesamtaufbau im Verbund

Quellen

- [1] fischertechnik: [Baukasten hobby 4](#).
- [2] Analogsignal aus PWM: <http://om.dharlos.de/doku/>
- [3] Spannungsabhängiger Frequenz-generator (vco): <http://tymkrs.tumblr.com/post/4496507556/19fm-vco-555timer-i-ham-lesson-o-de-day>
- [4] MIDI, Töne und Frequenzen: <http://www.sengpielaudio.com/Rechner-notennamen.htm>
- [5] Dirk Fox: „Einmessen“ eines digitalen Messgeräts. [ft:pedia 1/2013](#), S. 39-48.
- [6] Regressionsberechnung im Internet: <http://www.xuru.org/rt/PR.asp>
- [7] CadSoft: [EAGLE Platinenlayout](#).

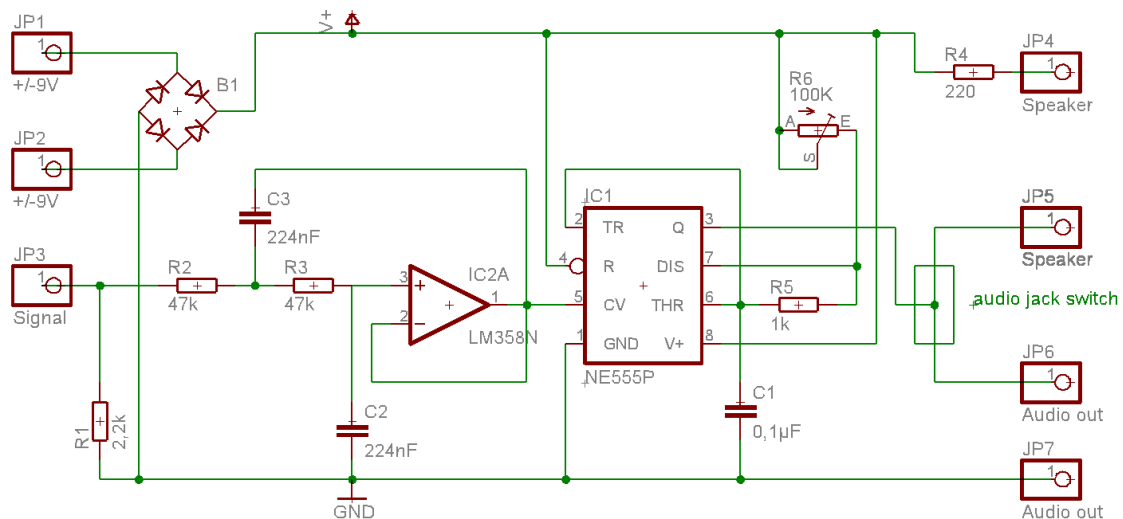


Abb. 16: Schaltplan zur Umwandlung eines veränderlichen PWM-Signals in eine zugehörige Tonfrequenz im Bereich von vier Oktaven

MIDI	Soll-Frequenz	Ist-Frequenz	PWM mittel
36	65	ohne	464
38	73	ohne	463
40	82	82	461
41	87	87	458
43	98	99	452
45	110	110	452
47	123	123	436
48	131	131	432
50	147	147	423
52	165	165	412
53	175	174	405
55	196	197	390
57	220	221	374
59	247	247	359
60	262	261	345
62	294	293	317
64	330	330	288
65	349	350	273
67	392	393	248
69	440	441	213
71	494	498	174
72	523	528	156
74	587	584	125
76	659	663	82
77	698	693	66
79	784	780	38
81	880	880	25
83	988	986	3

Tab. 1: Gemessener Zusammenhang zwischen MIDI-Code, Frequenz und PWM

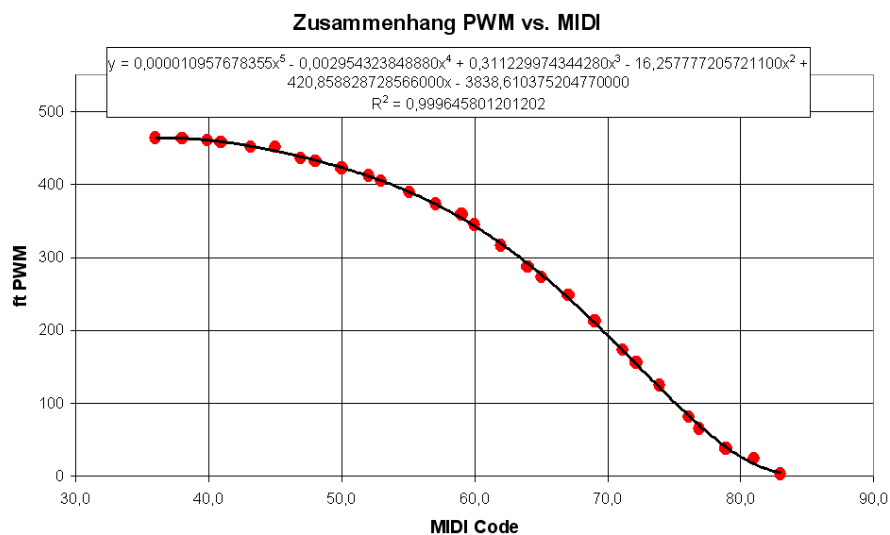


Abb. 17: Regressionsberechnung zur Umrechnung eines gegebenen MIDI-Codes in PWM

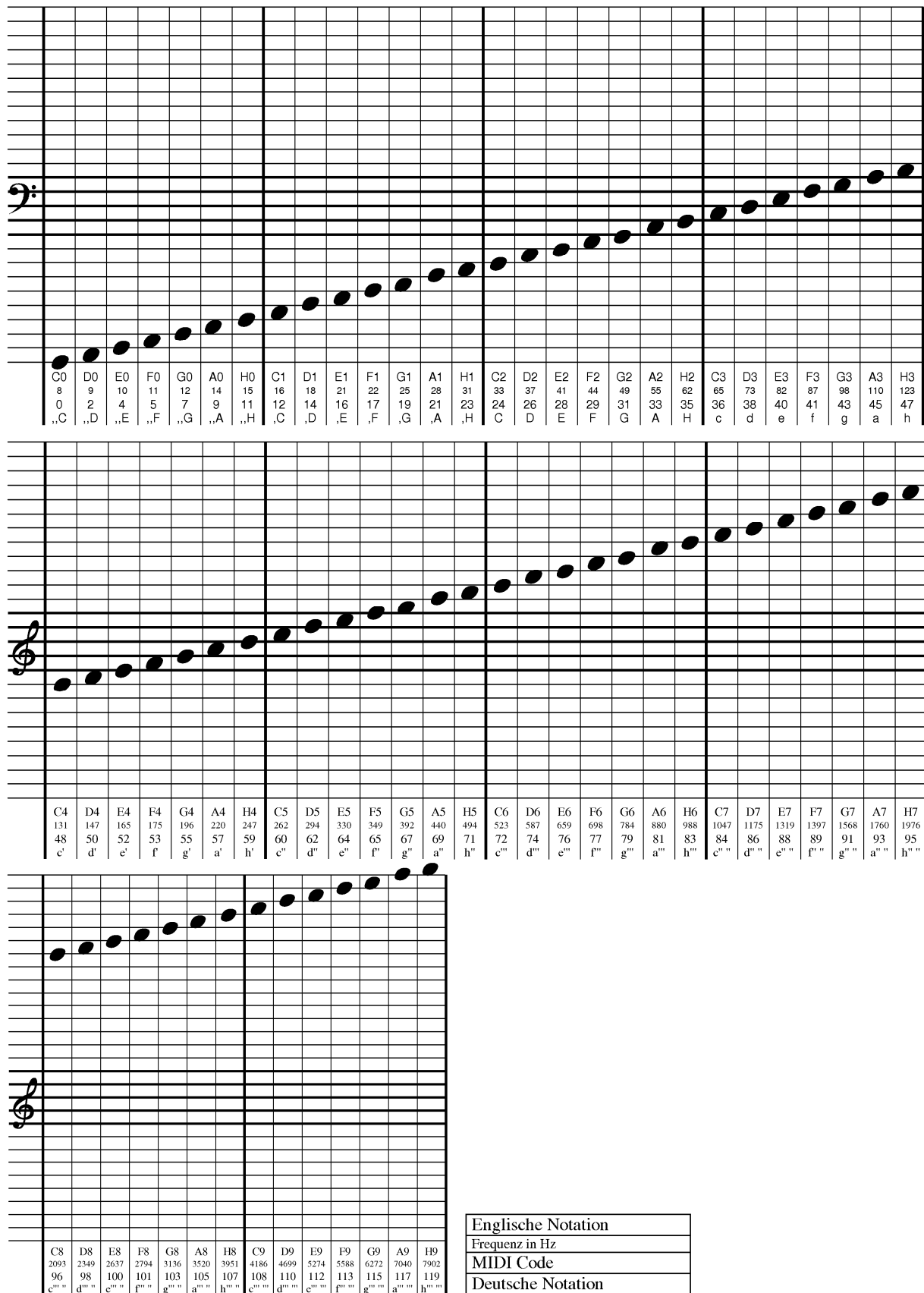


Abb. 18: Töne und Frequenzen, ohne Halbtonschritte

Elektronik

Nikon-Kamera-Ansteuerung über IR

Andreas Gail

Was hat dieses Thema mit Fischertechnik zu tun? Zunächst sollte erst einmal die Möglichkeit geschaffen werden, Aufnahmen aus der Fischertechnik-Welt heraus anstoßen zu können. Der nachfolgende Artikel zeigt, wie das über einen einfachen Taster, die guten alten „Silberlinge“ oder auch einen Robo TX/TXT Controller ermöglicht werden kann. In einer späteren Ausgabe der ft:pedia wird eine Anwendung gezeigt.

Die Kamera darf nicht zerstört werden

Bei aller Experimentierfreude ist es nach der persönlichen Sicherheit oberstes Gebot, dass die vergleichsweise teure Kamera-Ausrüstung nicht durch irrtümliche Verschaltungen zerstört wird. Deshalb wurde zur Ansteuerung die durch Nikon bereitgestellte Infrarot-Fernbedienung (IR) als Grundlage verwendet; siehe Abb. 1. Diese Fernbedienung arbeitet gemäß Hersteller mit einer ganzen Reihe von Nikon-Kameras zusammen.



Abb. 1: Nikon IR-Fernbedienung Typ ML-L3

Wenn es im Rahmen der Verwendung zu einem Fehler kommen sollte, wird höchstens die Fernbedienung zerstört. Wegen der fehlenden Kabelverbindung ist die Kamera selbst jedoch geschützt. Es soll nicht verschwiegen werden, dass die Möglichkeiten über die IR-Fernbedienung recht bescheiden sind: Damit kann nur ein Bild ausgelöst werden, solange die Kamera selbst in Bereitschaft ist. Parametereinstellungen oder sonstiges sind nicht möglich.

Die IR-Fernbedienung

Die Fernbedienung kann recht einfach geöffnet werden. Hierzu wird die obere Verschlussfolie entfernt:



Abb. 2: Geöffnetes Gehäuse der Fernbedienung

Das eigentliche Innenleben der IR Fernbedienung besteht aus einer kleinen Platine und einer 3 V-Knopfzelle zur Stromversorgung. An die Platine der IR-Fernbedienung müssen drei Kabel angelötet

werden: ein Anschluss als Eingangssignal (gelb) zur Kamera-Ansteuerung und zwei Anschlüsse zur Stromversorgung (+3 V = rot; Masse = blau). Abb. 3 und 4 zeigen das. Der Hinweis zum Anlöten des Signaleingangs wurde bereits zuvor im Internet in abgewandelter Form veröffentlicht [1, 2].

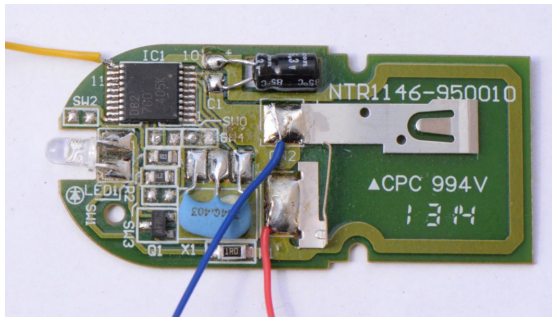


Abb. 3: Platine der Fernbedienung mit angelöteten Kabeln

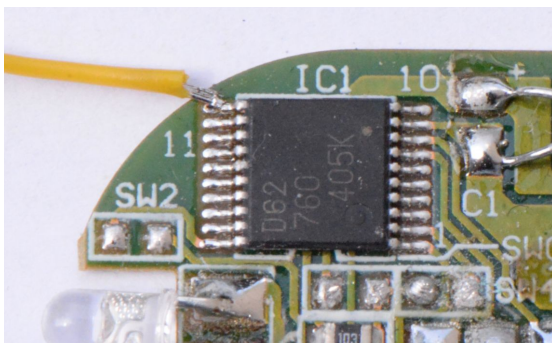


Abb. 4: Platine mit angelötetem Signalkabel zur externen Ansteuerung

Diese Lötarbeit gemäß Abb. 4 bedarf neben einer ruhigen Hand auch gute Sicht auf die Dinge. Gelötet wurde mit einem 8 W-Lötkolben.

Anforderungen an das zu erstellende IR Modul

- Die Verwendung soll einfach sein.
- Ein fehlerhafter Anschluss z. B. an den Robo TX Controller (RTXC) darf das Modul nicht zerstören.
- Die Spannungsversorgung soll mit Hilfe von 9 V möglich sein.
- Das Modul soll in einem eigenen Gehäuse eingebaut sein.

Verbindung der Welten

Mit dem Anlöten von drei Kabeln an die Fernbedienung ist es leider noch nicht getan, denn die IR-Fernbedienung arbeitet mit 3 V und fischertechnik mit 9 V.

Stromversorgung

Um die Handhabung des späteren IR-Moduls möglichst einfach zu gestalten, soll die 3 V-Stromversorgung aus fischertechnik heraus erfolgen. Nun könnte man auf die Idee kommen, dass hierzu ein Motorengang eines Robo TX Controllers einfach so parametrisiert wird, dass hier eine Spannung von 3 V dauerhaft ansteht. Das klappt jedoch leider nicht! Bei Mikrocontrollern ist es üblich, dass die Leistungseinstellung an „analogen“ Ausgängen über das PWM-Verfahren (Pulsweitenmodulation) erfolgt. Hierbei wird die volle bereitstehende Spannung (9 V) in sehr schneller Folge als Rechtecksignal ein- und ausgeschaltet. Die variable Leistungseinstellung ergibt sich dabei aus dem Verhältnis der ein- bzw. ausgeschalteten Pulslänge.

Um eine Gleichspannung von 3 V zu erzeugen, wird in der Schaltung gemäß Abb. 5 ein Spannungsregel-IC verwendet, Typ Texas Instruments LP2950 (Conrad #1014267). Die Spannungsstabilisierung erfolgt mit zwei Elektrolytkondensatoren. Als Verpolungsschutz wird zusätzlich ein Brückengleichrichter eingesetzt.

Signalweitergabe

Das Signal aus der fischertechnik-Welt, welches an die IR-Fernbedienung weitergegeben werden soll, hat wiederum eine Spannung von 9 V. Zur Verbindung dieser beiden Systeme wird ein Optokoppler eingesetzt, Typ Everlight EL817 (Conrad #156422). Die im Optokoppler befindliche IR-Diode wird durch einen 510 Ω -Vorwiderstand strombegrenzt.

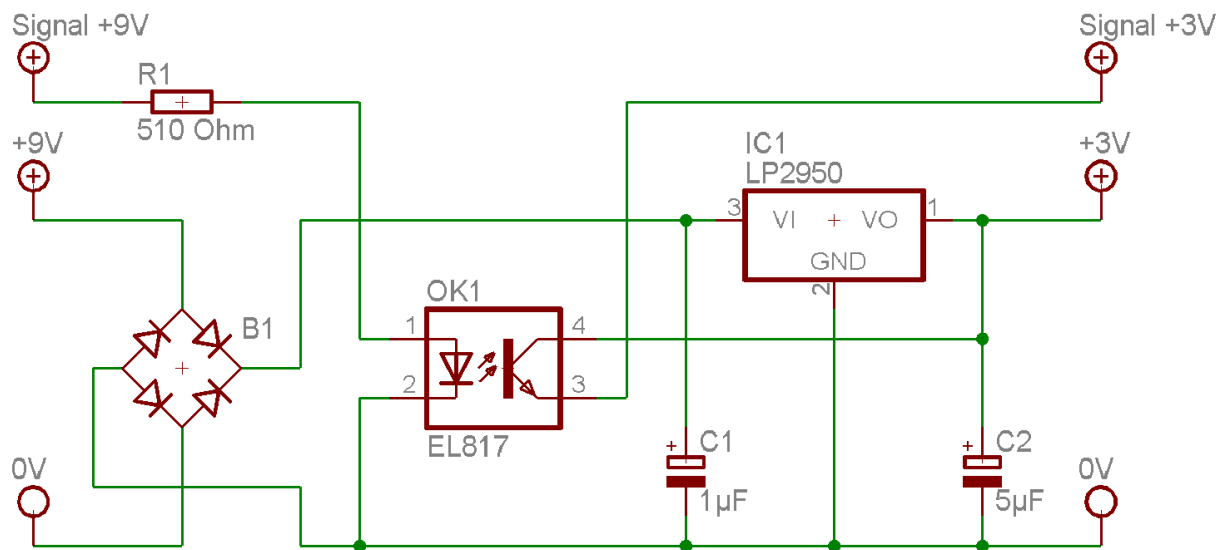


Abb. 5: Schaltplan zur Verbindung von fischertechnik mit der Nikon IR-Fernbedienung Typ ML-L3

Schaltungsaufbau

Schaltplan

Die oben beschriebene Schaltung ist in Abb. 5 zur Übersicht grafisch dargestellt.

Erster Test auf dem Steckbrett

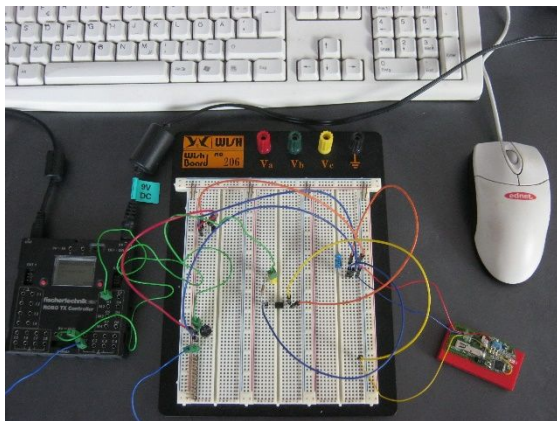


Abb. 6: Testaufbau zur Verbindung von fischertechnik mit der Nikon-Fernbedienung

Bei allen Schaltungsplanungen kann die Überlegung so gut sein wie sie will: Ob am Ende wirklich alles zusammen wunschgemäß funktioniert zeigt nur der Versuch.

Derartiges lässt sich vergleichsweise einfach auf einem Steckbrett gemäß Abb. 6 durchführen.

Fertiger Aufbau des IR Moduls

In Abb. 7 und Abb. 8 ist der vollständige Aufbau des neuen IR Moduls gezeigt. Alles passt zusammen in ein fischertechnik-Batteriegehäuse.



Abb. 7: Fertiges IR Modul

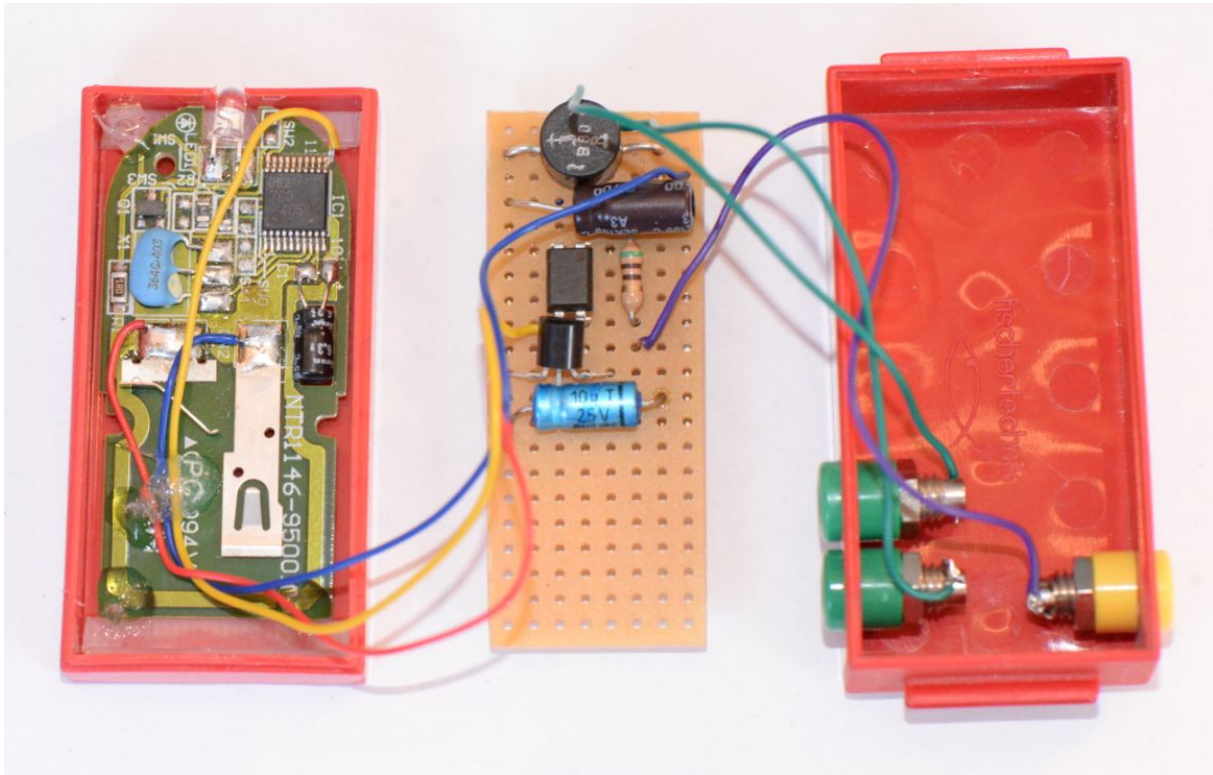


Abb. 8: Geöffnetes IR Modul

Hinweis zur Bewaffnung der Augen

Als sehr hilfreich bei kleinsten Lötarbeiten hat sich die Verwendung einer Lupenbrille (Abb. 9) erwiesen: vierfache Vergrößerung, Arbeitsabstand 25 cm.



Abb. 9: Lupenbrille mit vierfacher Vergrößerung, Eschenbach Typ rido-med 16364

Quellen

- [1] Modifikation Nikon ML-L3, [Variante 1](#).
- [2] Modifikation Nikon ML-L3, [Variante 2](#).

Computing

I²C mit dem TX(T) – Teil 12: Temperatursensor

Dirk Fox

Seit der ft:pedia-Ausgabe 3/2012 [2] stellen wir in loser Folge I²C-Sensoren vor, die sich an den TX anschließen lassen – mit passenden Robo Pro-Treibern. Inzwischen ist der TX-Nachfolger TXT erschienen und unterstützt seit dem 14.12.2015 auch das I²C-Protokoll (Firmware-Version 4.3.2). Die I²C-Anschlüsse am Erweiterungsport (EXT) des TXT entsprechen allerdings nicht denen des TX – es liegen nicht 5 V, sondern nur 3,3 V an. In dieser Folge stellen wir einen I²C-Universal-Adapter für den TXT vor – und einen Temperatursensor, der mit beiden Spannungen und damit auch an beiden Controllern betrieben werden kann.

Vorbemerkung

Temperaturmessungen sind mit fischertechnik seit der Einführung der NTC-Sensoren (*Negative Temperature Coefficient Thermistors*) möglich [1] – allerdings mit sehr begrenzter Genauigkeit. Auch wenn man eine saubere Kalibrierung seiner Messeinrichtung vornimmt [3].

Ganz strenge Betrachter gehen noch weiter zurück – denn schon das Bimetall im Elektromechanik-Kasten em-3 von 1970 erlaubte die Konstruktion eines einfachen Temperaturmessgeräts – wenn auch eines sehr ungenauen [5].

Mit IC-gestützten Sensoren kommt man zu deutlich besseren Werten. In einer früheren Folge der I²C-Reihe haben wir eine RTC (*Real Time Clock*) vorgestellt, die auch eine Temperaturmessung vornimmt – allerdings mit einer noch immer wenig überzeugenden Genauigkeit von $\pm 3^{\circ}\text{C}$ [4].

Hervorragende Resultate erzielt man mit I²C-Temperatursensoren. Sie erreichen eine – für die meisten Anwendungen ausreichende – Genauigkeit von $\pm 0,25^{\circ}\text{C}$ und liefern auf $0,0625^{\circ}\text{C}$ aufgelöste Messergebnisse.

Anschluss von I²C-Sensoren

In [4] habe ich gezeigt, wie sich in wenigen Minuten (und ohne LötKolben) aus F/F-Jumpfern und einer 6poligen Pfostenbuchse ein I²C-Universalkabel für den Anschluss an den TX konfektionieren lässt. Ganz ähnlich gelingt das mit F/F-Jumpfern und einer 10poligen Pfostenbuchse. Die I²C-Anschlüsse am EXT-Ausgang des TXT sind wie folgt belegt (Abb. 1):

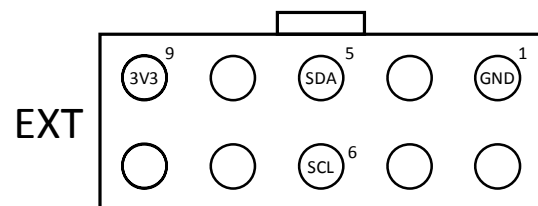


Abb. 1: Belegung des EXT-Anschlusses am TXT

Eine elegante Möglichkeit, um nicht für jeden Sensor ein (neues) I²C-Anschlusskabel konfektionieren zu müssen, wenn man schon TX-kompatible besitzt, ist die Verwendung eines Adapters, der die Anschlüsse von einem 2x5- auf einen 2x3-Pfostenstecker „umleitet“ (Abb. 2). Anschließend können die Sensoren mit den von mir vorgestellten TX-I²C-Universalkabeln über einen Bus-Expander (Conrad Best.-Nr. 192659-62, Abb. 3) mit dem TXT verbunden werden.

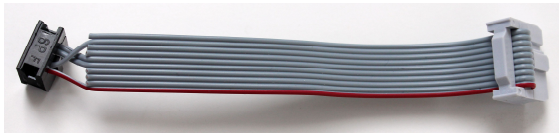


Abb. 2: I²C-Adapterkabel für den TXT

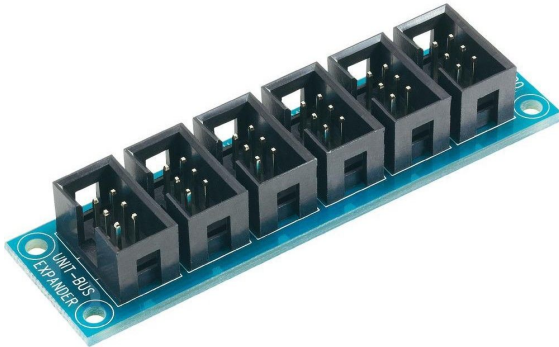


Abb. 3: I²C-Bus-Expander (Foto: Conrad)

Der MCP9808

Von Microchip Technology Inc. stammt der Temperatursensor MCP9808 aus dem Jahr 2011 (Abb. 4, [6]). Er besitzt eine I²C-Schnittstelle und arbeitet mit einer Betriebsspannung von 2,7-5,5V, daher lässt er sich sowohl am TX als auch am TXT betreiben. Adafruit hat ein [Breakout Board](#) mit dem MCP9808 entwickelt, das in Deutschland bei [exp tech](#) für weniger als 5 € erhältlich ist.

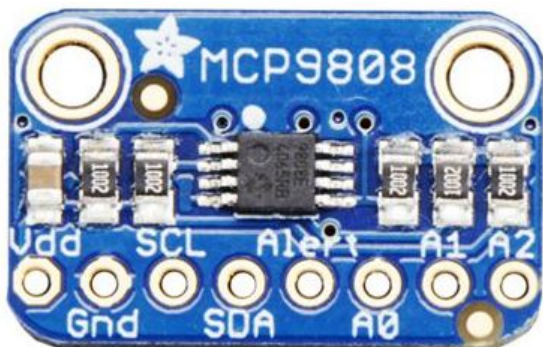


Abb. 4: Temperatursensor MCP9808 (Breakout Board von Adafruit)

Technische Daten

- **Betriebsspannung:** 2,7 bis 5,5 V
- **Stromaufnahme:** 200-400 μ A; im Standby-Modus 0,1-2 μ A.
- **I²C-Adresse:** Die Default-Adresse des Device ist 0x18; zur Vermeidung einer Adresskollision können durch Setzen der Eingänge A0, A1 oder A2 direkt auf dem Board alternativ die Adressen 0x19-0x1F gewählt werden.
- **Geschwindigkeit:** 400 kHz (Fast Mode).
- **Genauigkeit:** Im Temperaturbereich von -20°C bis 100°C hat der Sensor eine maximale Abweichung von 0,5°C (typisch: 0,25°C).

Die Berechnung der Temperatur kann in vier Auflösungen erfolgen: 0,5°C, 0,25°C, 0,125°C oder 0,0625°C. Der Zeitbedarf für die AD-Wandlung – und damit die Zahl der Temperaturbestimmungen pro Sekunde – unterscheidet sich dabei jedoch erheblich.

Register

Der MCP9808 verwendet für Konfiguration und Datenaustausch acht Register, die beschrieben (W) oder ausgelesen (R) werden können.³ Dabei sind die ersten sieben (0x01 bis 0x07) 16-Bit-Register; lediglich Register 0x08 ist ein 8-Bit-Register.

Register	Bedeutung
0x01	Configuration (R/W)
0x02	Alert Temperature (Upper Bound, R/W)
0x03	Alert Temperature (Lower Bound, R/W)
0x04	Critical Temp. (R/W)
0x05	Temperature (R)

³ Es sind insgesamt 16 Register adressierbar. Einige enthalten jedoch Kalibrierungsdaten und sollten daher nicht verändert werden.

Register	Bedeutung
0x06	Manufacturer ID (R)
0x07	Device ID, Revision (R)
0x08	Resolution (R/W)

Tab. 1: Register des MCP9808 [4]

Das Auslesen und Beschreiben der Register erfolgt mit einem I²C-Lese- bzw. Schreibbefehl, bei dem die Adresse des Registers als 8-Bit-Unteradresse angegeben wird. Wurde zuvor bereits aus derselben Adresse gelesen oder an dieselbe Adresse geschrieben, kann die Unteradresse weggelassen werden.

Configuration

Im *Configuration Register* (0x01) werden mehrere zentrale Einstellungen vorgenommen:

- *Hysteresis*: In den Bits 9 und 10 kann eine *Hysterese* (Verzögerung) eingestellt werden, die bei fallenden Temperaturen für eine verzögerte Auslösung eines Alarms sorgt: keine, wenn beide Bits 0 sind (00_b, default), Verzögerung von 1,5°C mit 01_b, 3°C mit 10_b und 6°C mit 11_b.
- *Shutdown Mode*: Im Shutdown-Modus (Bit 8 = 1) verbraucht der Chip lediglich 0,1 µA gegenüber ca. 400 µA im *Power Up Mode* (kontinuierliche Messungen; Default-Einstellung)
- *Lock Bits*: Durch Setzen der Bits 6 und 7 lassen sich T_{Upper} und T_{Lower} (die Grenzen des Messintervalls, Register 0x02 und 0x03) bzw. $T_{Critical}$ (Register 0x04) gegen Veränderung sperren.

Upper, Lower, Critical Temperature

In den drei folgenden 16-Bit-Registern (0x02-0x04) können drei Temperaturwerte eingestellt werden:

- Mit der Festlegung eines *Temperature Upper Bound* (Register 0x02), T_{Upper} , und eines *Temperature Lower Bound*

(Register 0x03), T_{Lower} , kann der Messbereich begrenzt werden.

- Durch die Einstellung einer *Critical Temperature* (Register 0x04), $T_{Critical}$, kann eine Alarmierung eingestellt werden.

Über- oder unterschreitet die Temperatur die Grenzen des eingestellten Messbereichs, T_{Upper} bzw. T_{Lower} , wird das durch die Bits 13 und 14 des Messwerts und – falls im *Configuration Register* ausgewählt – über das Setzen des Alert-Pins signalisiert. Ein Überschreiten von $T_{Critical}$ wird in Bit 15 des Messwerts signalisiert (siehe unten).

Die Darstellung des Temperaturwerts erfolgt als 13-bit-Zweierkomplement und entspricht der Darstellung der Messwerte.

Temperaturmesswert

Der aktuelle Temperaturmesswert kann aus Register 0x05 als 13-bit-Zweierkomplement in 16tel Grad (Celsius) ausgelesen werden. Der Wert wird – abhängig von der Auflösung (siehe unten) – mindestens vier und bis zu 33 Mal je Sekunde aktualisiert.

Die oberen drei Bit signalisieren ein Überschreiten der eingestellten $T_{Critical}$ (Bit 15) bzw. eine Temperatur t außerhalb des vorgegebenen Messbereichs, d. h. $t < T_{Lower}$ oder $t > T_{Upper}$ (Bit 13 und 14).

Manufacturer & Device ID

Die Hersteller- und Device-ID des Sensors ist in den Registern 0x06 und 0x07 gespeichert (*read only*). Die Register sollten die Werte 0x0054 (*Manufacturer ID*, dezimal: 84) und 0x0400 (*Device ID*, dezimal: 1024) enthalten. Mit dem Auslesen dieser beiden Werte kann geprüft werden, ob der Sensor korrekt antwortet.

Resolution

Im *Resolution Register* (0x08) wird die Auflösung der Temperaturangaben festgelegt. Je größer die Auflösung, desto zeitintensiver die A/D-Wandlung und geringer

die Zahl der je Sekunde durchführbaren Messungen:

- 0x00: 0,5°C/ca. 30 ms (33 Messungen/s)
- 0x01: 0,25°C/ca. 65 ms (15 Mess./s)
- 0x02: 0,125°C/ca. 130 ms (7 Mess./s)
- 0x03: 0,0625°C/ca. 250 ms (4 Mess./s), Default-Einstellung

Beim Einschalten des Devices (Power-Up) werden alle Registerinstellungen auf die Default-Werte gesetzt.

RoboPro-Treiber

Im [Downloadbereich der ft:c](#) findet ihr einen RoboPro-Treiber für den MCP9808 zum Herunterladen.

Die Treiber-Kommandos folgen im Wesentlichen der Registerstruktur – lediglich für die unterschiedlichen Einstellmöglichkeiten des Config-Registers wurden jeweils eigene Befehle integriert:

- Die Funktionen `MCP_GetManufacturerID` und `MCP_GetDeviceID` ermöglichen einen Test des Sensors: Sie müssen die Werte 84 bzw. 1024 zurückliefern.
- Mit den Kommandos `MCP_ShutDown` und `MCP_PowerUp` lässt sich der Sensor in den Sleep-Modus (Stromaufnahme < 0,2 µA) versetzen und wieder aufwecken.
- Die Auflösung des Sensors kann mit den Funktionen `MCP_GetResolution` und `MCP_SetResolution` ausgelesen bzw. eingestellt werden. Als Parameter wird ein ganzzahliger Wert von 0-3 übergeben.
- Mit den Befehlen `MCP_GetLowerBound` und `MCP_GetUpperBound` können die eingestellten Grenztemperaturen ausgelesen und mit den komplementären Kommandos `MCP_SetLowerBound` und `MCP_SetUpperBound` gewählt werden. Bei letzteren wird im Config-Register

das eventuell gesetzte Schreibsperre-Bit zurückgesetzt. Die Temperaturwerte werden als Fließkommazahl übergeben.

- Die Kommandos `MCP_GetCriticalTemp` und `MCP_SetCriticalTemp` lesen bzw. setzen die kritische Temperatur, ab der ein Alarm erfolgen soll. Dabei wird im Config-Register das eventuell gesetzte Schreibsperre-Bit zurückgesetzt. Auch hier werden die Temperaturwerte als Fließkommazahl übergeben.
- Die Hysterese wird mit `MCP_GetHysteresis` gelesen (Ausgabe: Gradangabe als Fließkommazahl) und mit `MCP_SetHysteresis` gesetzt (Input: Index 0-3).
- Mit `MCP_GetTemp` erhält man die aktuelle Umgebungstemperatur in Grad Celsius (als Fließkommazahl).

Anwendungsbeispiele

Eine simple Anwendung, die im Treiber-Download enthalten ist, ist ein Thermometer, das die aktuelle Umgebungstemperatur anzeigt.

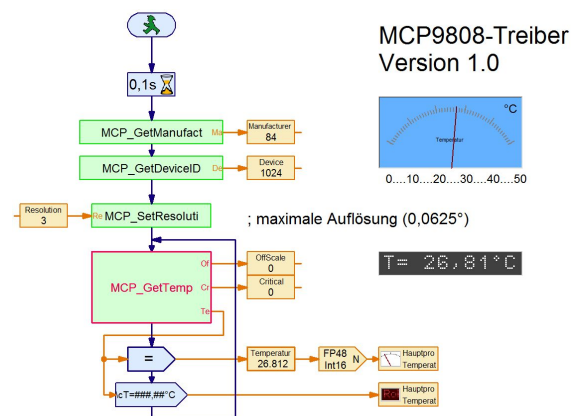


Abb. Einfaches I²C-Thermometer

Aber es lassen sich – auch in Kombination mit anderen Sensoren – einige weitere interessante Anwendungen verwirklichen.

Wetterstation

Ein derart genauer Sensor erlaubt auch professionellere Anwendungen, wie z. B. die Realisierung einer kleinen „Wettersta-

tion“, die die Außentemperatur (Tagesverlauf alle 15 Minuten, Tagesdurchschnitts-, Höchst- und Niedrigsttemperatur) protokolliert – zusammen mit dem aktuellen Luftdruck [7]. Mit einer *Real Time Clock* lassen sich die Messzeitpunkte präzise festlegen und protokollieren [4]. Die Messdaten können in einem Listenelement zwischengespeichert, als .csv-Datei exportiert und schließlich in einer Tabellenkalkulation anschaulich ausgewertet werden.

Temperaturregelung

Auch in einem Regler lässt sich der Sensor einsetzen: Sinkt die gemessene Temperatur unter eine voreingestellte Temperatur *LowerBound*, muss gewärmt werden; steigt sie über *UpperBound*, wird die Erwärmung gestoppt (oder, im Falle eines Kühlschranks: muss gekühlt werden).

Alarmierung

Schließlich kann man den Sensor für eine Alarmschaltung bei Überschreitung einer zuvor konfigurierten kritischen Temperatur einsetzen – sogar gleichzeitig mit den vorgenannten Anwendungen.

Referenzen

- [1] Dirk Fox: *Der Robo TX Controller als Messgerät*. [ft:pedia 1/2011](#), S. 33-36.
- [2] Dirk Fox: *I²C mit TX und Robo Pro – Teil 1: Grundlagen*. [ft:pedia 3/2012](#), S. 32-37.
- [3] Dirk Fox: *Einmessen eines digitalen Messgeräts*. [ft:pedia 1/2013](#), S. 39-48.
- [4] Dirk Fox: *I²C mit dem TX – Teil 7: Real Time Clock (RTC)*. [ft:pedia 4/2013](#), S. 28-34.
- [5] fischertechnik: [em3 – Elektro-mechanik](#). Handbuch zum Baukasten, 1970, S. 62.
- [6] Microchip Technology Inc.: [MCP9808](#). Datasheet, 02.08.2011
- [7] Georg Stiegler: *I²C mit dem TX – Teil 3: Luftdruckmessung*. [ft:pedia 1/2013](#), S. 32-38.

Computing

PWM-Motorsteuerung am fischertechnik-Universal-Interface

Dirk Uffmann

In der ft:pedia 2/2014 [2] habe ich vorgestellt, wie man das fischertechnik-Universal-Interface an einem AVR-Mikrocontroller betreiben kann – allerdings ohne PWM-Steuerung der Motoren. Das Interface, das für den Betrieb an der Parallel-Schnittstelle des PC vorgesehen war, bietet mit seinen Motortreibern TLE4201 keine explizite Hardware-Unterstützung für eine PWM-Steuerung. Eine Software von Ulrich Müller für den PC ermöglichte schließlich eine PWM-Steuerung [1]. Der vorliegende Beitrag stellt zwei Varianten vor, wie man mit einem AVR ATMEGA2560 an bis zu zwei Universal-Interfaces im Master-Slave-Betrieb alle acht Motorausgänge getrennt voneinander mit PWM steuern kann.

Die Idee

Die Idee von Ulrich Müller war, zur Umsetzung einer Geschwindigkeitsvorgabe die Motoren schnell genug (mit Zeiten im Bereich von 12-36 ms) jeweils ein- und dann zeitlich kontrolliert wieder auszuschalten. Genau diese Idee habe ich hier auch verfolgt. Ulrich Müller gibt an, dass man das parallele Interface (Abb. 1) sogar mit noch geringeren Zeiten ansteuern kann.

Wie ich in meinem Beitrag in ft:pedia 2/2014 [2] gezeigt habe, benötigt man für das parallele Interface weniger als 12 μ s für die Datenübertragung eines neuen Ausgangszustandes. Bei zwei fischertechnik-Interfaces im Master-Slave-Betrieb und 16 MHz Takt des AVR sind es ca. 32 μ s. Selbst mit zwei fischertechnik-Interfaces könnte man bei einer Auflösung bzw. Granularität von 100 PWM-Stufen also PWM-Frequenzen von bis zu 300 Hz realisieren ($1/(100 \cdot 32 \mu\text{s})$).

Das ist aber gar nicht nötig: Eine Frequenz zwischen 70 und 100 Hz ist vollkommen ausreichend und sogar besser für die Motoren geeignet. Auch bei Lampen am Ausgang sieht man so noch kein Flackern. Für den benötigten Zweck kann man sehr

geschickt einen Timer-Overflow-Interrupt und – bei der zweiten Variante – verschiedene Output-Compare-Interrupts des Mikrocontrollers nutzen.

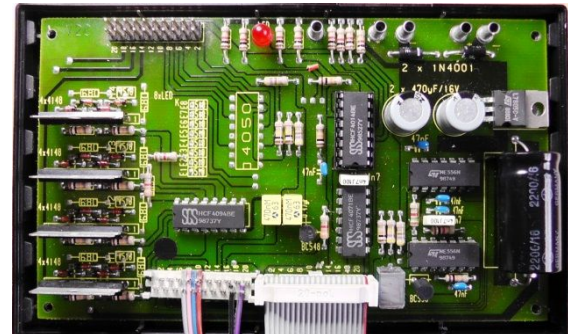


Abb. 1: Geöffnetes fischertechnik-Universal-Interface 30520 für die parallele PC-Schnittstelle. Links unten am Pfostenstecker wird ein PORT des Mega 2560 angeschlossen.

Die Hardware

Derzeit sind fertige Boards mit großzügig ausgestatteten Atmel AVR Mega-Controllern sehr günstig erhältlich. Einen Arduino Klon des MEGA 2560R3 (Abb. 2) erhält man in China bei Aliexpress für weniger als zehn Euro inklusive Versand. Ein solches Board habe ich hier genutzt und am PORT K das fischertechnik-Universal-Interface mit der in Tab. 1 dargestellten Pin-Zuordnung angeschlossen.



Abb. 2: Arduino-MEGA 2560R3. PORT K liegt unten rechts, bezeichnet als A8-A15.

```
MEGA2560-----FTUI(20-pol.Pfostenstecker)
-----
PK1-----LoadIn-(20P Pin 7)
PK2-----LoadOut-(20P Pin 8)
PK3-----Clock-(20P Pin 9)
PK4-----DataOut-(20P Pin 10)
PK5-----EY-(20P Pin 13)
PK6-----EX-(20P Pin 14)
PK7-----DataIn*-(20P Pin 17)
GND-----GND-(20P Pin 19)
```

Tab. 1: Pin-Zuordnung zwischen ATMEGA 2560 und fischertechnik-Universal-Interface

Eigentlich liegt *DataIn* auf Pin 18. Der Ausgang des Eingangs-Schieberegisters wird auf dem Weg zu Pin 18 über ein Oder-Gatter mit dem Ausgangssignal der NE555 Timer der beiden Analogeingänge verknüpft und dann über einen Treibertransistor BC548 invertiert. Da wir die Timer-Signale nicht brauchen und diese sogar stören, wenn die Trigger-Signale dafür nicht auf *high* gelegt und die analogen Eingänge nicht auf den Timer durchgeschleift werden, verwenden wir lieber Pin 17 als direkten Ausgang des Schieberegisters. Hier hat das Ausgangssignal auch noch eine geringere Verzögerungszeit zur positiven Takt-Flanke und daher können wir dann mit einer höheren Übertragungsrate arbeiten.

Eine Besonderheit der TLE4201-Motortreiber im fischertechnik-Universal-Interface ist, dass diese deutlich weniger Strom verbrauchen, wenn beide Ausgänge auf V+ geschaltet sind, als im umgekehrten Fall (beide Ausgänge auf GND). Bei Beschaltung mit Motoren empfiehlt es sich daher, diese dadurch abzuschalten, dass beide Ausgänge auf V+ geschaltet werden. Dies

wird in der Header-Datei *FTUI_PWM.h* definiert, hier am Beispiel M1:

```
#define M1 1
```

→ im Ruhezustand sind beide Ausgänge auf +V.

Verbraucher, die einphasig gegen Masse angeschlossen sind (Lampen, Magnetventile etc.), werden im Ruhezustand auf GND gelegt. Hier am Beispiel der beiden Ausgänge an M1:

```
#define M1 0
```

→ im Ruhezustand sind beide Ausgänge auf GND.

Ebenfalls wird in der Header-Datei definiert, ob nur ein fischertechnik-Interface angeschlossen ist oder noch ein zweites im Slave-Betrieb. Im letzteren Fall muss der Parameter *slave* mit dem Wert 1 definiert werden.

```
#define slave 1
```

Die Software

Die Software wurde mit Atmel-Studio in C geschrieben und mit XLoader auf den AVR geladen (damit umgehe ich die Arduino-Tools). Ich stelle zwei Varianten vor, die sich hinsichtlich Rechenlast und Speicherbedarf unterscheiden. Die erste Variante verursacht eine höhere Rechenlast im Mikrocontroller als die zweite, benötigt aber nur einen 8-bit-Timer mit Overflow-Interrupt und belegt deutlich weniger Programmspeicher als die zweite Variante.

Variante 1 ist daher auch für kleinere Mikrocontroller geeignet (wie z. B. den ATMEGA 328). Die zweite Variante benötigt mehrere 16-bit-Timer mit Overflow-Interrupt und zusätzlich bis zu acht Output Compare-Match-Interrupts (vier je angeschlossenes fischertechnik-Interface). Für Variante 2 ist daher ein großzügig ausgestatteter Mikrocontroller erforderlich (wie z. B. ein ATMEGA 2560).

Variante 1

Nicht jeder Mikrocontroller ist so üppig ausgestattet wie der MEGA 2560. Häufig hat man nur zwei oder gar keine PWM-Kanäle zur Verfügung. Hier bietet sich zur Implementierung der PWM-Funktionalität für alle acht Motoren folgende Methode an:

- Es wird ein Timer-Overflow-Interrupt mit 128 µs Periode genutzt.
- In dieser Interrupt-Routine wird ein PWM-Zähler inkrementiert und bei Erreichen des Wertes 100 auf 0 zurückgesetzt und mit diesem Ereignis werden dann alle aktiven Motoren eingeschaltet.
- Weiterhin wird geprüft, ob die in einem Array `uint8_t pwm[8]` gespeicherten PWM-Werte der acht Motoren mit dem Zähler übereinstimmen. Sofern dies

zutrifft, wird der jeweilige Motor abgeschaltet.

Auf diese Art kommt man auf eine PWM-Frequenz von 78 Hz und eine Granularität von 1 % (PWM-Steuerung mit 100 Stufen). Tab. 2 zeigt den Code zur Initialisierung.

Mit dem Timer-Overflow-Interrupt muss jeweils ein Datentransfer zum fischertechnik-Interface erfolgen. Da ich einen Funktionsaufruf mit Stack-Nutzung aus der Interrupt-Routine heraus vermeiden wollte, habe ich die Funktion `FTUI_update()` als *inline*-Funktion vor der Interrupt-Service-Routine definiert (Tab. 3). Das bewirkt, dass der entsprechende Code beim Kompilieren direkt in der Interrupt-Service-Routine (Tab. 4) platziert und dort ohne weitere Stack-Last ausgeführt wird.

```

/*****
    FTUI Init Routine
*****/
void FTUI_Init (void)
{
    // internen Pull-up Widerstand für DataIn einschalten
    FTUI_Port_Write = (1<<FTUI_DataIn);
    // Ausgänge auf low setzen
    FTUI_Port_Write &= ~((1<<FTUI_LoadIn)|(1<<FTUI_DataOut)|(1<<FTUI_Clock)|(1<<FTUI_LoadOut));
    // Ausgänge definieren bzw. aktiv schalten
    FTUI_Port_DDR = (1<<FTUI_LoadOut)|(1<<FTUI_LoadIn)|(1<<FTUI_DataOut)|(1<<FTUI_Clock);
    // system clock 16 MHz is used (external quartz oscillator)
    // start timer2 with prescaler clock/8
    TCCR2B = _BV(CS21);
    // activate 8-bit Timer0 overflow interrupt with prescaler divide by 8
    // -> 8 * 256 / (16 MHz) = 128 µs or ca. 7,8 kHz
    TIMSK2 = _BV(TOIE2);
    sei();

    // Motorausgänge initial high oder low schalten entsprechend der gewählten Beschaltung
    #if slave
        motorcontrol = output;
    #else
        motorcontrol = output1;
    #endif
}

```

Tab. 2: C-Code zur Initialisierung (Variante 1)

```

/*****
FTUI_update überträgt das motorword und liest parallel das inputword ein.
Diese Funktion muss mindestens alle 300 ms aufgerufen werden, um den
Watchdog im FT-Interface zurückzusetzen (durch das Clocksignal). Das
passiert hier automatisch über den Timer1-Overflow-Interrupt alle 12,8 ms.
Bei einem Systemtakt von 16 MHz braucht die Routine mit einem Interface
nur ca. 12 µs, mit zwei Interfaces sind es ca. 32 µs.
*****/
inline void FTUI_update (void)
{
#ifdef slave
    uint16_t mask_motor=(1<<15);
    uint16_t mask_input=(1<<7);
#else
    uint8_t mask_motor=(1<<7);
    uint8_t mask_input=(1<<7);
#endif

    FTUI_Port_Write &= ~(1<<FTUI_Clock); //Clock low-Phase einleiten

//    Zunächst werden alle Eingänge als 0 angenommen.
    inputword = 0;

//    CMOS4014: Nun werden die Eingangszustände in das Schieberegister übernommen.
//    Dazu muß LoadIn gesetzt werden und danach eine positive Clockflanke erzeugt werden.
    FTUI_Port_Write |= (1<<FTUI_LoadIn);
    FTUI_Port_Write |= (1<<FTUI_Clock); //Clock high-Phase einleiten
//    mit dieser positiven Clockflanke steht das erste Bit auf der
//    DataIn-Leitung an, jedoch mit einer Verzögerung. Daher wird
//    es erst in der nächsten Clock-Low-Phase eingelesen

//    CMOS4014: Dann werden die Eingänge wieder vom Schieberegister entkoppelt
    FTUI_Port_Write &= ~(1<<FTUI_LoadIn);

//    Dann werden die Bits 7 bis 0 (ein FT-Interface) bzw. 15 bis 0 (zwei
//    FT-Interfaces) gesendet (motorword) und parallel die Bits 7 bis 0 und
//    dann 15 bis 8 (zwei FT-Interfaces) eingelesen (inputword)
    while (mask_motor)
    {
//        Wenn das jeweilige Sendebit high sein soll, dann wird DataOut = 1
//        gesetzt, sonst DataOut = 0
        if(motorword & mask_motor) FTUI_Port_Write |= (1<<FTUI_DataOut);
        else FTUI_Port_Write &= ~(1<<FTUI_DataOut);
        FTUI_Port_Write &= ~(1<<FTUI_Clock); //Clock low-Phase einleiten
        mask_motor = (mask_motor>>1); //Schiebe das Maskierungsbit eins nach rechts

//        Nun wird das jeweilige Bit im Datenstream auf der Leitung DataIn
//        daraufhin getestet, ob der zugehörige Eingang "high" ist,
//        und dann wird inputword an der jeweiligen Bitposition auf 1 korrigiert
        if (FTUI_Port_Read & (1<<FTUI_DataIn)) inputword |= mask_input;
        mask_input = (mask_input>>1); //Maskierungsbit nach rechts weiterschieben
#ifdef slave
        if (mask_input==0) mask_input = (1<<15); //Maskierungsbit auf Position 15 setzen
#endif
        FTUI_Port_Write |= (1<<FTUI_Clock); //Clock high-Phase einleiten
    }

//    CMOS4094: Durchschalten der Daten aus dem Schieberegister ins Speicherregister
//    für die Motorausgänge
    FTUI_Port_Write |= (1<<FTUI_LoadOut);
    FTUI_Port_Write |= (1<<FTUI_LoadOut);
    FTUI_Port_Write |= (1<<FTUI_LoadOut);

//    wieder Entkoppeln von Speicherregister und Schieberegister
    FTUI_Port_Write &= ~(1<<FTUI_LoadOut);
}

```

Tab. 3: C-Code mit der in der Interrupt-Service-Routine genutzten inline-Funktion `FTUI_update()`. Diese Funktion ist für beide Varianten identisch.

```

ISR (TIMER2_OVF_vect) //findet alle 128µs statt
{
    uint8_t i;
    uint8_t update_flag = 0;
    pwm_counter++;
    if (pwm_counter == 100)
    {
        pwm_counter = 0;
//   Ausgänge gemäß der Variable motorcontrol einschalten
        motorword = motorcontrol;
        update_flag = 1;
    }

    i = 4 + (4*slave);
    do
    {
        i--;
        if (pwm_table[i] == pwm_counter)
        {
//   Ausgänge gemäß der PWM Tabelle pwm_table[] abschalten
            if (output & (0x0003 << (2*i))) motorword |= (0x0003 << (2*i));
            else motorword &= ~(0x0003 << (2*i));
            update_flag = 1;
        }
    } while (i);

    if (update_flag) FTUI_update();
}

```

Tab. 4: C-Code für die Interrupt-Service-Routine (Variante 1)

```

/*****
M () stellt die Funktion zur Manipulation von motorcontrol zur Steuerung
einzelner Motoren bereit. Der pwm Wert gilt auch bei Einzelbeschaltung
der Ausgänge immer für beide Ausgänge.
*****/
void M (uint8_t motor_number, uint8_t state, uint8_t pwm)
{
//   Abfangen unsinniger Werte
    if (motor_number == 0) return;
    if (motor_number > (4 + (4*slave))) return;
    if (state > 3) return;

    uint8_t position = (motor_number-1)*2;
    pwm_table[motor_number-1] = pwm;

//   Zunächst wird "state" in "motorcontrol" an der gewünschten Position hinzugefügt
//   dazu werden zunächst die jeweiligen 2 bits an den Bit-Positionen für motor_number
//   auf 0 gesetzt
    motorcontrol &= ~(3 << position);
//   und dann die 2 neuen Bits dem motorcontrol hinzugefügt
    motorcontrol |= state << position;
}

```

Tab. 5: C-Code mit der Motor-Funktion (Variante 1)

Variante 2

Der AVR ATMEGA 2560 bietet zu dem benötigten Zweck vier verschiedene 16-bit Timer (1, 3, 4, 5), die jeder mit jeweils drei Output-Compare-Registern kombiniert sind. Damit können bis zu 12 PWM-Ausgänge gesteuert werden. Über einen Timer-Interrupt werden jeweils alle Motoren eingeschaltet, die laufen sollen. Über die

jeweiligen Output-Compare-Interrupts wird dann je nach gewünschtem PWM-Wert der zugeordnete Motor – bis zum erneuten Einschalten durch den Timer-Interrupt – abgeschaltet.

Die 16-bit Timer werden auf eine Periodendauer von 12,8 ms bzw. eine PWM-Frequenz von 78 Hz eingestellt. Tab. 2 zeigt die Initialisierung. Die Timer zählen von 0

bis 25599 (Top-Wert im Register ICR). Für die Compare-Matches werden nur die oberen acht Bits der 16-Bit-OCR-Register genutzt, wodurch eine Granularität der PWM-Steuerung von genau einem Prozent erreicht wird (100 Stufen – dies ergibt sich aus den 25600 Timer-Schritten geteilt durch die 256 Timer-Schritte für die unteren acht Bits).

Sowohl mit dem Timer-Overflow-Interrupt als auch mit den jeweiligen Output-Compare-Match-Interrupts muss jeweils ein Datentransfer zum fischertechnik-Interface erfolgen. Dafür wird dieselbe *inline*-Funktion `FTUI_update()` wie bei Variante 1 verwendet.

Um einen Motor mit PWM-Wert einzuschalten, muss der PWM-Wert in das entsprechende OCR-Register übertragen werden und im Register TIMSK der Output-Compare-Match-Interrupt aktiviert werden.

Das Video

Ich habe beide Varianten des Codes ausgiebig mit meinem Modell eines Industrieroboters getestet und keine Fehler mehr gefunden. In einem [Youtube-Video](#) kann die Wirkung der PWM-Steuerung beobachtet werden. Sie wird beim Anfahren der verschiedenen Positionen genutzt. Vor dem Erreichen der jeweiligen Zielposition wird der PWM-Wert linear heruntergefahren (langsamer werdendes Anfahren des Ziels).

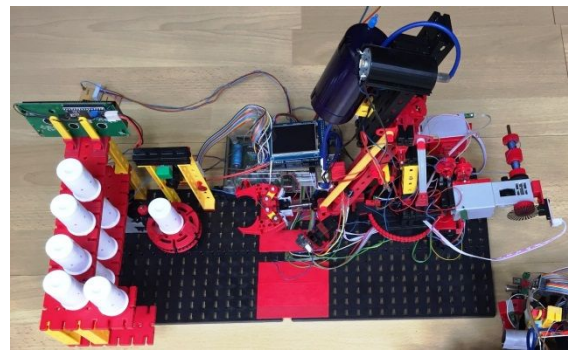


Abb. 3: Industrieroboter mit PWM-Steuerung der Motoren am fischertechnik-Universal-Interface, gesteuert durch einen ATMEGA2560

Download

Die benötigten Funktionen für den Programmcode des Mikrocontrollers stehen in einem [Downloadpaket](#) zur Verfügung. Für jede Variante ist eine eigene C-Datei vorgesehen. Bitte beim Kompilieren nur die eine jeweils ausgewählte Variante bzw. nur eine der beiden C-Dateien verlinken. Die Header-Datei `FTUI_PWM.h` ist für beide Varianten identisch.

Ich freue mich auf die Wiederbelebung eurer alten fischertechnik-Universal-Interfaces mit PWM-Funktionalität.

Quellen

- [1] Müller, Ulrich: [Drehzahlsteuerung von Motoren am Interface \(ftSlow.ZIP\)](#). ftcommunity.de
- [2] Uffmann, Dirk: *Nutzung des Universal-Interfaces 30520 als Port-Erweiterung an einem Mikrocontroller*. [ft:pedia 2/2014](#).



„[Das Buch](#)“ – seit dem 4. November 2015 im Handel.