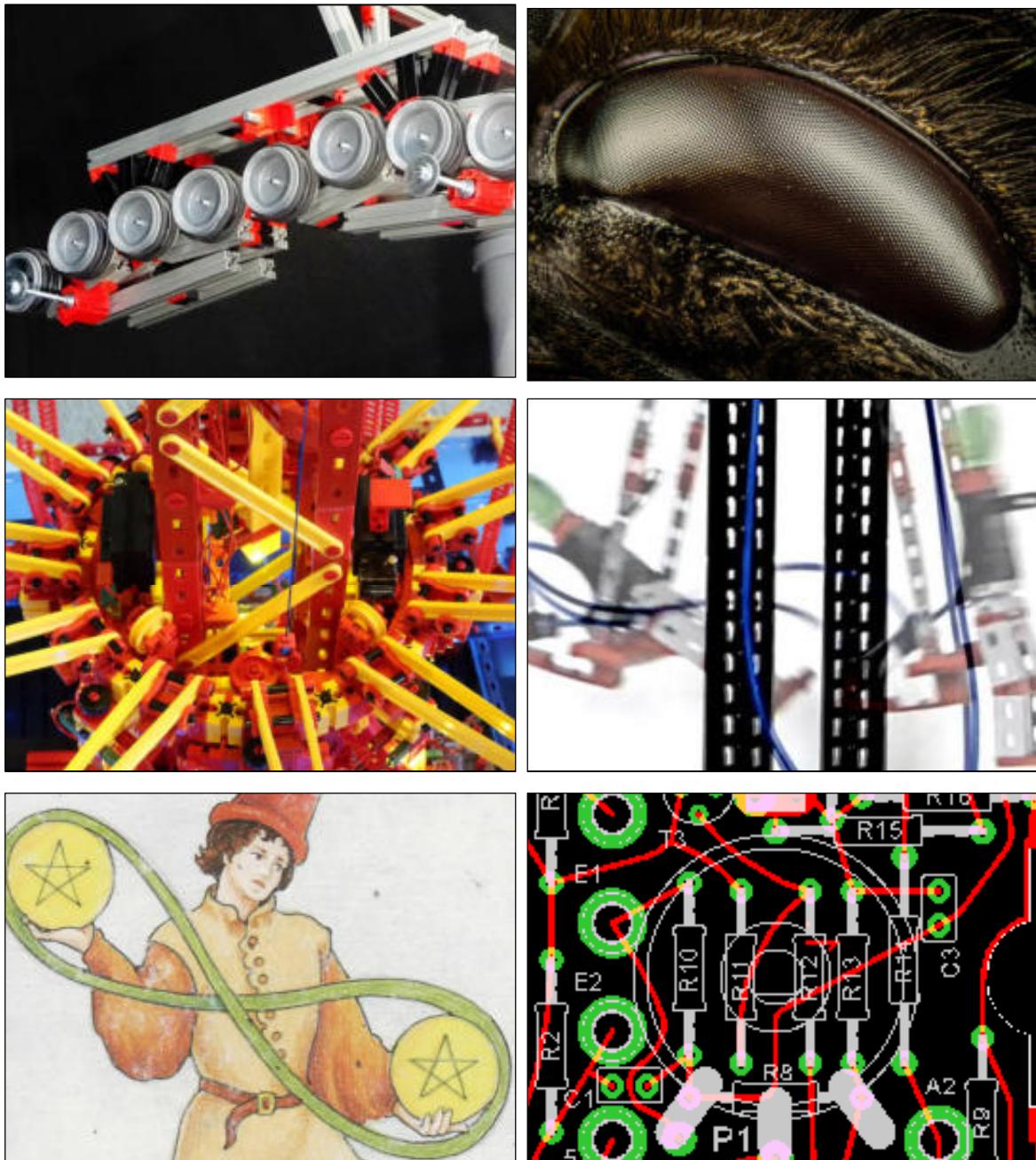


ft:pedia

Heft 2/2021



Herausgegeben von
Dirk Fox und Stefan Falk

ISSN 2192-5879

Editorial

42

In Douglas Adams' (1952-2001) Science-Fiction-Klassiker aus dem Jahr 1979 ist 42 die Antwort auf die „ultimative Frage nach dem Leben, dem Universum und dem Rest“. Der Supercomputer Deep Thought berechnet diese Antwort, und als er sie nach 7,5 Millionen Jahren Rechenzeit ausspuckt, ist die ursprüngliche Frage in Vergessenheit geraten.

Heute erhält man diese Antwort in Millisekunden – z. B. bei [Wolfram Alpha](#), vom Sprachassistenten Siri, und selbst Googles Taschenrechner lässt sich nicht lumpen: Wer nach „[the answer to life, the universe and everything](#)“ sucht, wird sofort mit der richtigen Ausgabe beglückt: 42.

Nach eigener Aussage wählte Douglas Adams die 42 rein zufällig. Keine Rede davon, dass sie im Binärsystem mit 101010_2 so eine reine Gestalt annimmt. Oder dass Arthur Dent im zweiten Band der Roman-Trilogie die Frage „Wie viel ist neun multipliziert mit sechs“ scrabbelt – eine Rechenaufgabe, die im 13er Zahlensystem genau 42_{13} ergibt. Oder dass Adams selbst erstmalig in der 42. Folge von [Monty Python's Flying Circus](#) im Fernsehen auftrat.

Die ursprüngliche Frage bleibt aber auch im 42. Jahr nach Veröffentlichung der Antwort im Dunkel. Zwar könnte man argumentieren, dass wir ja inzwischen im Zeitalter der „Fake News“ angekommen sind, einer Kommunikationskultur, in der wir ohnehin daran gewöhnt sind, permanent mit Antworten auf nicht gestellte Fragen überschüttet zu werden. Warum also nicht 42?

Dirk Fox, Stefan Falk

Doch Douglas Adams, den wir leider nicht mehr selbst danach fragen können, könnte es auch genau umgekehrt gemeint haben. Vielleicht liegt der tiefere Sinn der Antwort ja gar nicht in der Frage – sondern die Antwort trägt ihn in sich selbst?

Kennen wir das als fischertechniker nicht zur Genüge? Haben wir jemals ein Modell konstruiert, das, wenn es fertig ist und funktioniert, exakt unserer ursprünglichen Aufgabenstellung entspricht? Oder ist es nicht vielmehr so, dass sich die Aufgabe im Verlauf der Konstruktion präzisiert, modifiziert, adaptiert? Dass sich die eigentlich zu lösende Herausforderung erst während der Konstruktion herausschält? Und dass das funktionsfähige Modell zum Schluss eine Lösung für Probleme darstellt, von denen wir vorher womöglich nicht einmal geahnt haben, dass sie existieren?

So gesehen ist jedes fischertechnik-Modell eine 42 – eine Antwort auf eine (oder auch mehrere) Frage(n), die wir vorher bestenfalls partiell formuliert haben – wenn überhaupt –, und die sich oft überraschend aus dem Nichts materialisieren.

Damit ist auch jeder Beitrag in der ft:pedia eine 42 – eine Antwort auf eine nie gestellte Frage. Doch eröffnet uns jede dieser Antworten neue Erkenntnisse und Einsichten. Wir sollten also vielleicht nochmal über den Titel der ft:pedia nachdenken.

Beste Grüße,
Euer ft:pedia-Team

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

Inhalt

42	2
Fotostudio für Makrofotografie	4
Kreuztisch	7
Stacking-Schlitten für Stapelbilder	11
Schmidt-Kupplung.....	17
Die beschwingte Schaukel	21
Wellenflug – Wellenflieger.....	28
Die unendliche Lemniskate	36
Großprojekt Seilbahn (Teil 2): Erste Elemente	45
Es blinkt	54
Echtzeitgenerator.....	58
Controller-Erweiterung	63
Tresor	66
Silberlinge: Original oder Nachbau (Teil 2).....	80
Silberlinge: Original oder Nachbau (Teil 3).....	90
Preisgünstige, autarke Stepper-Motorsteuerung	101
ftSwarm – Das Prinzip Heinzelmännchen	107
TXT 4.0 Controller – ein Blick unter die Haube	112
ROBO Pro Coding – ein Blick hinter die Kulissen.....	117

Impressum

<http://www.ftpedia.de>

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

Autoren: Florian Bauer, Christian Bergschneider, Arnoud van Delden, Stefan Falk, Dirk Fox, Hans-Christian Funke, Stefan Fuss, Fabian Haas, Till Harbaum, Peter Krijnen, Claus Ludwig, Franz Nachtsheim, Rüdiger Riedel, Tilo Rust.

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

Fotostudio für Makrofotografie

Fabian Haas

Die Makrofotografie wird immer beliebter. Dabei sind nicht nur Insekten in freier Wildbahn im Focus, sondern oft auch H0-Modeleisenbahnfiguren (Maßstab 1:87) und Szenen. Egal ob drinnen oder draußen, das Licht ist wichtig! In diesem Beitrag beschreibe ich ein simples und flexibles Fotostudio für diesen Zweck.

Licht

Die Makrofotografie verlangt nach viel Licht, schließlich braucht man eine kleine Blende (f8-16) und kurze Belichtungszeiten (1/100 oder kürzer), sollen die Aufnahmen nicht verwackeln. Dabei spielt es keine Rolle, ob man draußen ist und Insekten und Blumen nachjagt, oder drinnen, und Szenen von H0-Modeleisenbahnfiguren arrangiert hat und fotografieren will. Letzteres wird immer beliebter; z. B. widmete erst kürzlich die c't-Fotografie ein ganzes Heft diesem Thema [1]. Instagram tut ein Übriges.

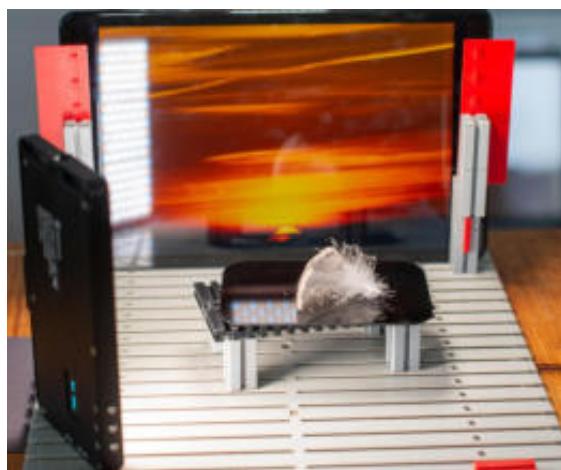


Abb. 1: Seitliches LED-Flächenlicht

Zum Glück haben sich LED-Flächenlichter in den letzten Jahren sehr weiterentwickelt, sodass man kein Blitzlicht mehr braucht, was v. a. den Vorteil hat, dass man die Lichtwirkung sofort sieht und entsprechend

regulieren kann (Abb. 1). Entsprechend den natürlichen Lichtverhältnissen wollte ich ein Licht von oben, das die Sonne oder Himmel simuliert, und kaum bewegt werden sollte. Ergänzt durch ein zweites, freies LED-Licht, mit dem ich die Figuren praktisch aus Augenhöhe anleuchten kann. Das ist mobil auf einem kleinen Fotostativ befestigt, oder steht ganz direkt auf dem Tisch.

fischertechnik bot sich als exzellente Lösung an, weil man hier alle Maße, Längen, und Gewichte im Griff hat und im Handumdrehen modifizieren kann. Aufgrund meiner 40-jährigen fischertechnik-Erfahrung, wählte ich die Grundplatte 1000 mit Boden und baute einen Galgen aus Statikteilen (Abb. 2).



Abb. 2: Der Galgen – Aufsicht auf den Ausleger. Metallachsen 120 stören die Beleuchtung nicht, und die LED-Leuchte lässt sich frei darüber bewegen

Es zeigt sich, dass die Höhe von zwei Winkelträgern 120 ausreicht; da es aber Hebelwirkungen durch den Ausleger, auf dem die LED-Leuchte liegt, geben wird, verbaute und verstrebte ich auf einer Seite jeweils drei mal zwei Winkelträger. Der Abstand zwischen diesen Pylonen hängt einfach von der breite der LED-Leuchte ab. In meinem Fall sind es 9 cm lichte Breite.

Der Ausleger ist drei Winkelträger 120 lang, zur Stabilisierung zweireihig und wird über lange Metallachsen 170 mit den Pylonen verbunden. Damit die LED-Leuchte nicht zwischen rechtem und linkem Ausleger hindurch fällt, wurden Metallachsen 120 eingesteckt und gesichert (Abb. 3). Darauf lässt sich das Licht vor und zurück schieben.



Abb. 3: Verbindung von Pylonen und Ausleger mit zwei langen Achsen

Aluprofile wären sicher noch etwas stabiler und fester, allerdings stört das verbleibende Spiel bzw. die Verformbarkeit im System nicht. Eine ganz präzise Ausrichtung der LED-Leuchte ist nicht notwendig; außerdem wird die Ausrichtung meistens nur einmal pro Fotosession geändert und dann nicht mehr angefasst.

Hintergrund

Ein Bild ist nur so gut wie sein Hintergrund! Insgesamt passt der Aufbau von den Dimensionen her sehr gut zu einem A3 Blatt, das es im Schreibwarenhandel in allen möglichen Farben gibt.

An den Pylonen wird das Blatt mit einer Gelenkklasche fixiert, die, in entsprechender Höhe mit einem S-Riegel fixiert, festgeklemmt wird. An der Grundplatte dienen Bausteine 15 x 30 x 5 oder Winkelsteine als Widerlager. Über die Entfernung des Widerlagers auf der Grundplatte und der Höhe der Gelenkklaschen lässt sich die Biegung im Papier (dort, wo es von der Grundplatte auf Pylonen übergeht) sehr gut regulieren und man erhält eine perfekte „Kuhle“, einen geschmeidigen Übergang, wie man das in der Fotografie liebt (Abb. 4).



Abb. 4: Eingespanntes A3 Papier bildet (regulierbare) Kuhle

Reicht ein Papier nicht als Hintergrund aus, weil man z. B. ein Foto, eine Szenerie zeigen will, kann man schlicht ein ausgedrucktes Foto oder ein Tablet unter den Gelenkklaschen platzieren.

Letzteres hat den Vorteil, dass es selber regulierbar leuchtet, also hat man nochmal eine Extralichtquelle. Über die Helligkeit des Displays lässt sich die Deutlichkeit des Hintergrundbildes festlegen. Ich konstruierte zwei einfache Halter aus jeweils vier Bausteinen 30 kombiniert mit Bauplatte 30 x 90 (Abb. 1). Damit ließ sich das Tablet erhöhen, über die Grundplatte stellen, was evtl. für den Bildaufbau wichtig ist, und über die Bauplatten wurde das Tablet etwas besser vor Umkippen geschützt als durch einfache Bausteine.

Die von mir mit meinem Makro-Fotostudio gemachten Bilder findet ihr auf Instagram (fabian_pos), und natürlich wird der Aufbau immer wieder modifiziert und verbessert. So bekam ich erst kürzlich einige Aluprofile, die sich zur Stabilisierung, Rahmung und auch Formgebung für das Papier einsetzen lassen. Es gibt immer etwas zu tun. Insgesamt hat sich der Aufbau aber sehr gut bewährt.

Referenzen

- [1] Peter Nonhoff-Arps: [*c't Fotografie 3/2021: Makrofotografie jenseits des Abbilungsmaßstabs 1:1*](#). Heise-News, 03.05.2021.

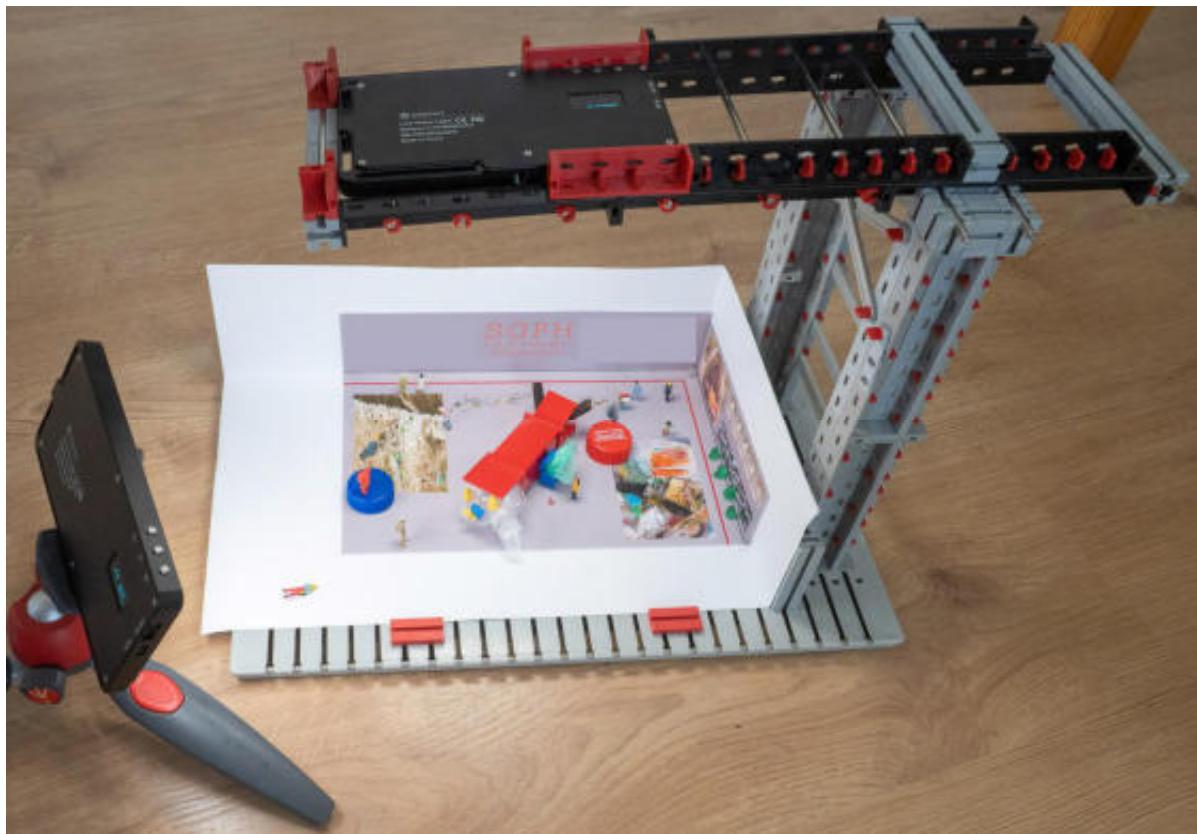


Abb. 5: Gesamtansicht des Fotostudios mit einem „Museumsaufbau“ über Müll im Maßstab 1:87.
Oben liegt die LED-Leuchte, die vor- und zurück geschoben werden kann.
Beide LED-Leuchten lassen sich unabhängig steuern.

Modell

Kreuztisch

Fabian Haas

Die Makrofotografie wird immer beliebter! Wer kleine Dinge groß herausbringen will, braucht ein kleines, aber feines Fotostudio. In der Makrofotografie spielt aber auch die Ausrichtung der Objekte eine große Rolle. Wenn die Kamera fest auf einem Stativ steht, muss das Objekt möglichst genau in (wenigstens) zwei Achsen bewegt werden. Dazu habe ich einen sogenannten Kreuztisch entwickelt, der genau dies erlaubt.

Die Idee eines Kreuztisches stammt natürlich nicht von mir. Ich kenne ihn aus der Mikroskopie, wo ein Objekträger vor dem sehr fest und vibrationsfrei installierten Objektiv in zwei Achsen hin und her bewegt wird, um genau an die Stelle zu kommen, die man betrachten möchte. Werkzeugmaschinen verschieben oft das Werkstück,

wodurch die großen und schweren Werkzeugteile (Bohrer, Fräsen) fest am Rahmen befestigt werden können. Allerdings: Bei fischertechnik habe ich so einen Kreuztisch noch nie gesehen, und ich dachte es wäre sicher schön, so einen zu bauen, um die Objekte punktgenau ins Bild zu bekommen.

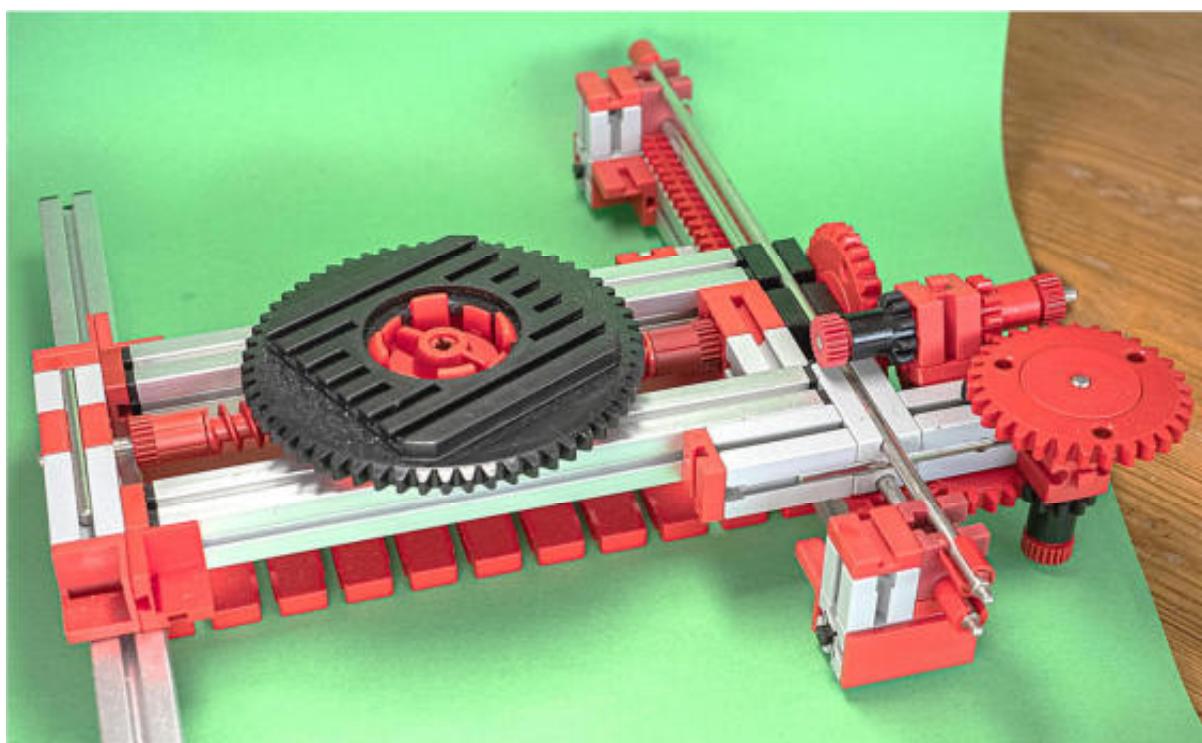


Abb. 1: Kreuztisch in der Gesamtansicht, mit der Antriebseinheit rechts im Bild.
Links gleiten die Aluprofile ohne Führung aufeinander.



Abb. 2: Antriebseinheit im Detail (1)

Dabei sollte er möglichst flach sein. Große Wege sind im Makrobereich nicht erforderlich, ist der Bildausschnitt doch nur wenige Zentimeter groß. Bei 1:1-Abbildung ist das Bildfeld gerade mal $24 \cdot 36$ mm groß bzw.

klein. Auf eine motorische Steuerung wollte ich verzichten, da ich nicht über Schrittmotoren und andere Hardware verfüge und auch nicht über die entsprechenden Programmierkenntnisse.

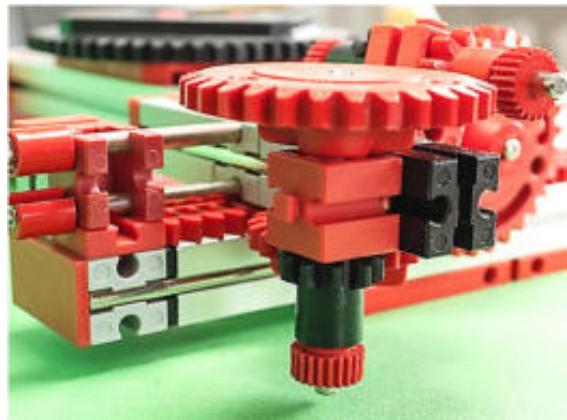


Abb. 3: Antriebseinheit im Detail (2)

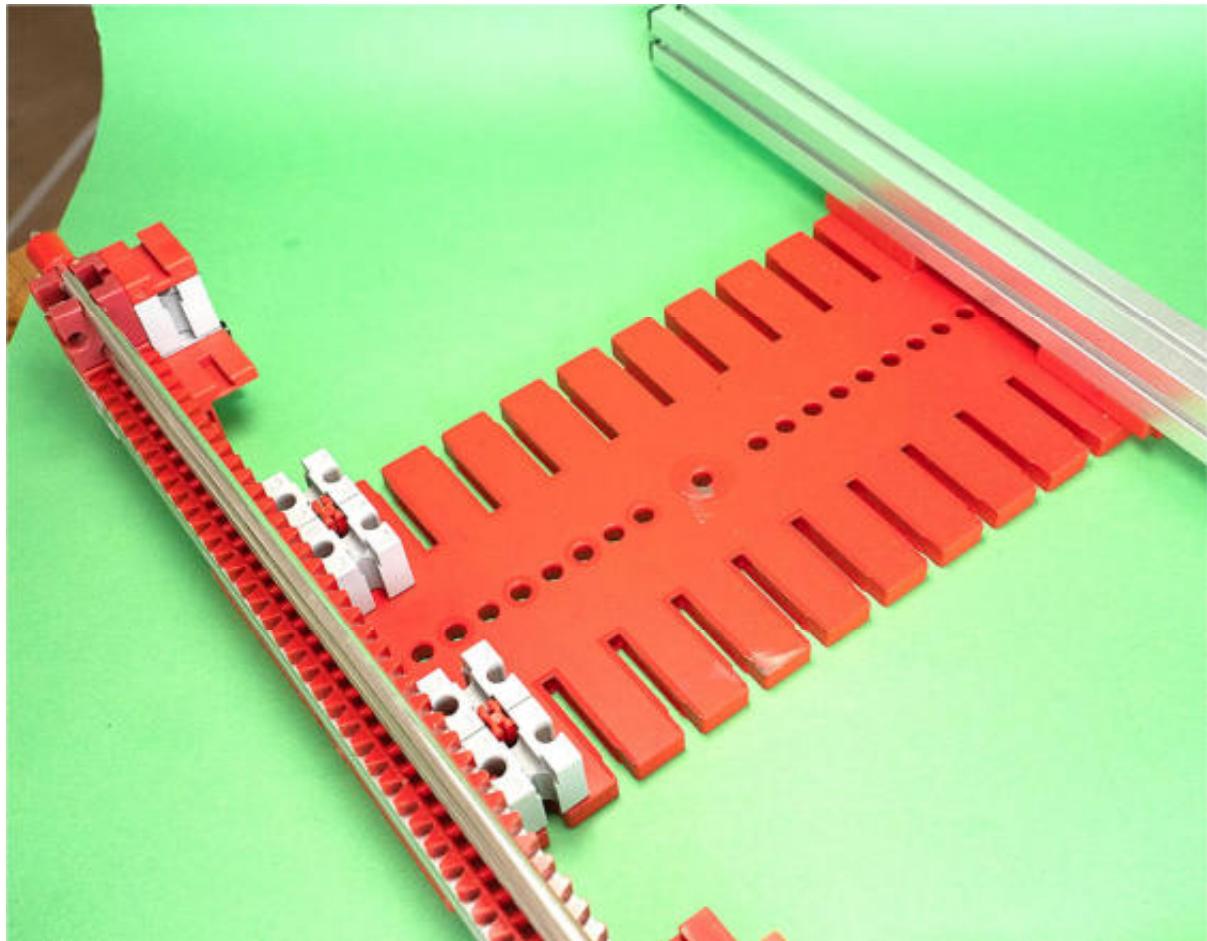


Abb. 4: Separate Basis, mit den Metallachsen als Führungen und zusätzlicher Stabilisierung durch Baustein 15

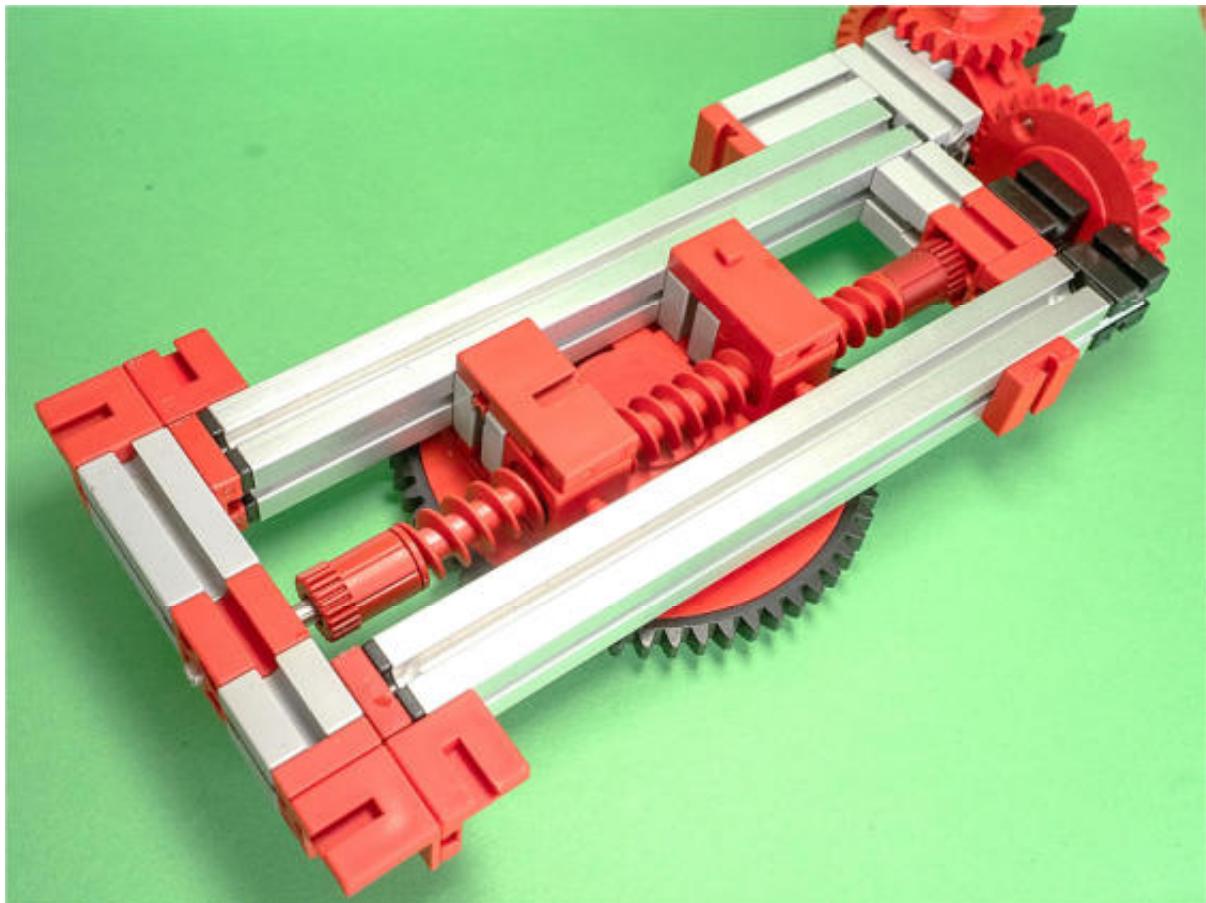


Abb. 5: Die Brücke von unten

Heraus kam eine Lösung mit vier Aluprofilen, die hier aufgrund der Stabilität absolut notwendig sind. Als Basisteil wurden zwei Aluprofile (225 und 175 mm waren verfügbar, wobei die Längen beliebig sind) jeweils am Rand der Grundplatte $180 \cdot 90$ mit Verbindern befestigt. Da Präzision und Spielfreiheit hier besonders wichtig sind, wurden sie mit zusätzlichen Bausteinen 15 gesichert. Ein Aluprofil wurde mit Zahnstangen besetzt und am Ende Rollenlager für zwei Metallachsen ausreichender Länge montiert. Dabei stehen die Metallstangen mittig über den Zahnstangen.

Die beiden Metallachsen dienen als Führung für die Brücke, die über Zahnräder, die in die Zahnstangen greifen, in x-Richtung verschoben wird. Ein Zahnrad Z30 dient als Griff. Es sitzt mit einem Ritzel Z10 auf einer Achse (geführt in einem Baustein 15 mit Bohrung), welche ein Zahnrad Z20 antreibt,

dass seitlich in die Zahnstangen auf dem Aluprofil der Basis eingreift. Ich versuchte so, noch eine zusätzliche Untersetzung einzubauen, damit der Vortrieb in x-Richtung noch etwas feiner reguliert werden kann. So wird die Brücke also über die auf der Grundplatte fest stehenden Aluprofile bewegt.

Die Brücke besteht aus zwei senkrecht zur Basis liegenden Aluprofilen, die am antriebsfernen Ende frei über das Aluprofil gleiten. Geführt wird die Brücke ausschließlich über die beiden Metallachsen über der Zahnstange. Zwischen den Aluprofilen der Brücke befindet sich eine Schnecke, auf der sich zwei Schneckenmuttern (m1,5) laufen. Im richtigen Abstand justiert, eliminieren zwei Schneckenmutter einiges an Spiel, das nur eine davon hätte, und erschweren den Vortrieb etwas, was in die-

sem Fall aber gewünscht ist. Auf der Metallachse der Schnecke ist ein Z30 befestigt (mit Baustein 15 mit Bohrung in der Brücke), als Griff dient aber wieder ein Ritzel Z10. Es liegt mit einem weiteren Ritzel Z10 auf einer Achse (über Baustein 15 mit Bohrung) und treibt ein Z30 an – eine 1:3-Untersetzung also.

Ich habe hier eine sehr enge Bauweise gewählt, um die beiden Griffe nahe beieinander zu haben. Andere Lösungen sind natürlich möglich.



Abb. 6: Details zu den Schneckenmuttern, auffüllen des Raumes mit Baustein 15 und Befestigung am Drehkranz Z58

Im abgebildeten Fall wurde ein Drehkranz Z58 an den Schneckenmuttern angebracht. Dadurch lassen sich die Schneckenmuttern spielarm justieren. Zusätzliche Bausteine 15 u. a. verringern das Spiel in der Brücke weiter. Der Drehkranz muss direkt ohne weitere Untersetzung, Rastung oder Mechanik angefasst werden, was in der Praxis aber kein Problem ist. Der Drehkranz wurde gewählt, da er präzise läuft und sich frei drehen lässt (die Zahnräder spielen keine Rolle). Wer seine Objekte nicht drehen möchte, kann eine Grundplatte 120 · 60 montieren.

Insgesamt wurde der Kreuztisch nur etwas höher als ein Baustein 30 – ohne den Drehkranz, der addiert nochmal einige Millimeter. Insgesamt ist der Kreuztisch schön flach und kompakt. Kompakter geht sicher noch, wenn man kürzere Aluprofile verwendet. Dann ist der Tisch aber vielleicht etwas unflexibler im Einsatz und man muss ihn immer neu verschieben. So lässt er sich z. B. mit der Grundplatte 1000 kombinieren.

Viel Spaß!

(Kontaktadresse des Autors:
fabian@fabianhaas.de)

Modell

Stacking-Schlitten für Stapelbilder

Fabian Haas

Die Makrofotografie wird immer beliebter! Ein Trend, der durch die gestiegene Rechenpower in den Kameras und Computern möglich wurde, ist das so genannte Image Stacking bzw. Stapelbilder. Dabei werden viele Aufnahmen gemacht, die nur einen schmalen, aber jeweils anderen Bereich des Motivs scharf abbilden, und dann zu einem einzigen durchgehend scharfen Bild zusammengesetzt. Hier möchte ich meinen Schlitten für Stapelbilder vorstellen, der den exakten Vortrieb zwischen den Bildern von nur wenigen 1/100 mm ermöglicht.

Die Methode des Stackings bzw. Stapelns, also das Zusammensetzen von Bildern, in denen jeweils nur ein schmaler Bereich scharf ist, zu einem durchgehend scharfen Bild, hat sich mittlerweile fest etabliert und

ist so beliebt geworden, dass sie in Kameras (z. B. Olympus und Panasonic) integriert wurde. Allerdings wird da nur eine recht geringe Anzahl von Bildern zusammenge rechnet, etwa 10 bis 15 Stück. Weiterhin

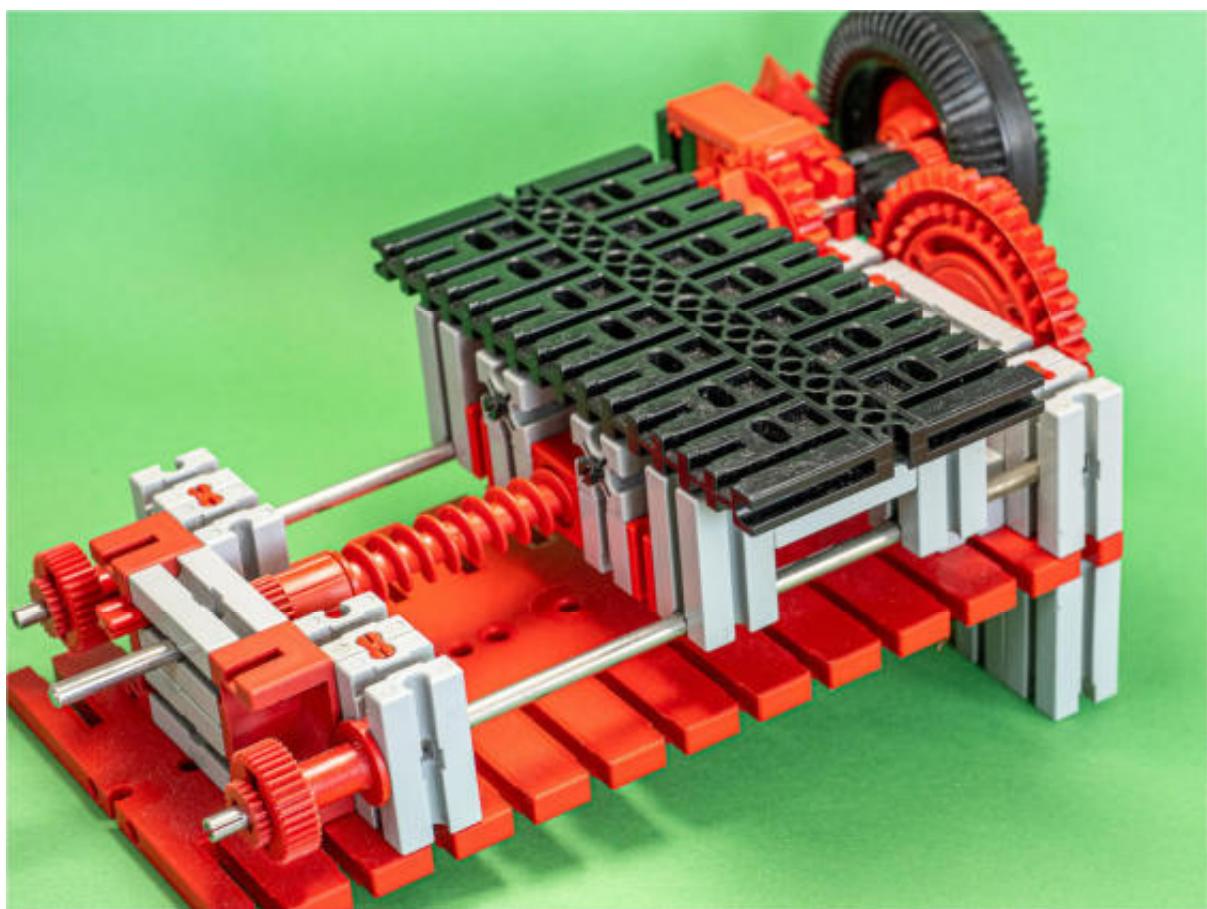


Abb. 1: Makroschlitten in der Gesamtansicht

wird ein geeignetes Objektiv mit Autofokus benötigt, da die Verstellung der Schärfenebene über die Autofokusmechanik von der Kamera gesteuert wird. Der Vorzug ist, dass man diese Funktion immer dabei hat und die Aufnahmen in sehr schneller Reihenfolge gemacht werden. Daher sind selbst lebende Insekten, die nur ein paar Sekunden stillsitzen, durchaus ein mögliches Motiv. Die Autofokus-gesteuerten Objektive haben typischerweise einen Abbildungsmaßstab von maximal 1:1.

Wer diese Grenzen sprengen will, muss ins Studio! Man benötigt ein anderes Objektiv, wie z. B. das Laowa 2,5-5x Ultramacro, das bis fünffach abbildet, aber keinen Autofokus besitzt. Und man benötigt mehr Aufnahmen, da der scharfe Bereich im Bild mit steigendem Abbildungsmaßstab immer kleiner wird. Um ein Objekt ganz abzudecken, benötigt man also mehr als die zehn bis 15 Bilder, die die Kamera anbietet. Die

Kamera kommt kurzerhand auf einen Schlitten, der eine schrittweise, exakte Verschiebung ermöglicht. So werden bis zu 100 Bilder möglich, die mit Helicon oder Zerene [3, 4] zu einem Bild zusammengerechnet werden. Kommerziell ist einiges verfügbar (siehe z. B [2]). Ich dachte jedoch, das sollte mit fischertechnik möglich sein, obwohl letzteres nicht unbedingt für allerhöchste Präzision – und die braucht es hier – bekannt ist.

Ich kam zu folgender Lösung: Als Basis dient die Grundplatte $180 \cdot 90$. Da der Arbeitsbereich fürs Makro-Stacking recht klein ist, reicht diese Grundplatte vollkommen aus. Als Gleitschienen wurden links und rechts zwei Metallachsen 180 mit je zwei Grundbausteinen 30 mit Bohrung angebracht und mit je einer Seiltrommel am Ende gesichert. In der Mitte, ebenfalls in Grundbausteinen 30 mit Bohrung gelagert, befindet sich eine Metallachse 200 mit einer



Abb. 2: Hummelauge mit fünffach vergrößerndem Objektiv aufgenommen und aus 50 Einzelbildern entstanden, mit Laowa Ultramakro und Helicon. Im 16 cm breiten Bild ergibt das Maßstab $\approx 80:1$.

durchgehenden Schnecke, die am oberen Ende ein Zahnrad Z40 mit Flachnabe trägt. Das Z40 liegt am Halteblock direkt auf, und je nachdem, wie eng alles festgezogen wurde, lässt sich die Reibung bzw. Schwer-gängigkeit einstellen. Da wir nur geringe Bewegung brauchen, die dafür mit hoher Präzision, sollten die Teile eher fest und schwer drehbar sein.



Abb. 3: Blick auf die Rastung (Reifen 60 und Federfuß) und das Getriebe für die minimal kleinen Schritte

Über ein Getriebe steht das Z40 mit einem Reifen 60 in Verbindung, wobei das Getriebe dessen Drehung untersetzt. Am Reifen 60 ist per Winkelstein 60° und Federfuß eine Rastung realisiert, die es erlaubt, den Schlitten genau um eine bestimmte Zahl von Rastungen voranzutreiben. Also genau das, was wir brauchen: Eine exakt definierte Schrittweite!



Abb. 4: Rastung mittels Federfuß

Der Reifen 60 wurde gewählt, weil er dank seines Profils die meisten Rastungen in der fischertechnik-Welt bereitstellt. Es sind 62 Lamellen, die das Profil erzeugt. Der Drehkranz hat nur 58 Zähne und das Z40 noch weniger, einzig das Drehschalteroberteil hat mit 80 mehr, ist aber Mikromechanik und dreht frei, da Noppen für die Nabenhügel fehlen.

Der Reifen 60 geht auf eine Z10:Z20 Unter-setzung, dann auf eine Z10:Z40 Unter-setzung, insgesamt also 1:8. Man benötigt daher acht Umdrehungen des Reifens 60, um eine Umdrehung des Z40 zu erzeugen. Oder: $8 \cdot 62$ Rastungen, also 496 Rastungen. Das sind sehr viele, erklärt aber nicht, welcher Vortrieb für den Schlitten erzeugt wird. Der Vortrieb ergibt sich aus der Höhe der Schnecke: eine Umdrehung der Schnecke treibt die Schneckenmutter um 5 mm voran.



Abb. 5: Detailblick aufs Getriebe

Es ergibt sich für eine Rastung ein Vortrieb von

$$\frac{5 \text{ mm}}{496 \text{ Rastungen}} \approx 0,010 \frac{\text{mm}}{\text{Rastung}}$$

Eine Rastung ergibt also ziemlich genau $1/100$ mm Vortrieb. Das ist genau der Bereich, den wir für hohe Vergrößerungen brauchen! In der Praxis nimmt man oft drei bis fünf Rastungen zwischen den Bildern. Die Bewegung der Schärfenebenen lässt sich direkt am Kameramonitor (Live View) beurteilen und die Zahl der Rastungen anpas-

sen. Die Schärfenebenen müssen sich überlappen und der Vortrieb gleichmäßig sein. Bei geringeren Maßstäben braucht man mehr Vortrieb und ggf. kann eine Unterstzung entfernt werden.

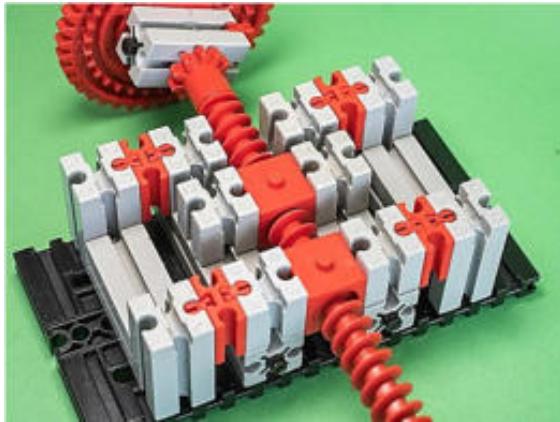


Abb. 6: Schlitten von unten mit den beiden Schneckenmuttern

Der Schlitten besteht aus einer Grundplatte $120 \cdot 60$, wobei sie mit zwei Schneckenmuttern in der Mitte und über Bausteine 30 in die seitlichen Führungsachsen eingreift. Das Gewicht von Kamera und Objektiv ist gering genug, dass die drei Metallachsen nicht gebogen werden. Die Führungen für die Metallachsen und die Schneckenmuttern sind mit Bausteinen 15, Baustein 7,5 und Verbindern zu einem Block verbunden. Schließlich sollen Verwindungen und Spiel, soweit es irgend geht, vermieden werden. Der Schlitten sollte „übermassiv“ gedacht werden.

Auf dem Schlitten ruht der Kameraadapter: wieder eine Grundplatte $120 \cdot 60$, die mit vier Achsen 30 und mehreren Bausteinen 7,5 mit dem Schlitten verbunden ist. Darauf

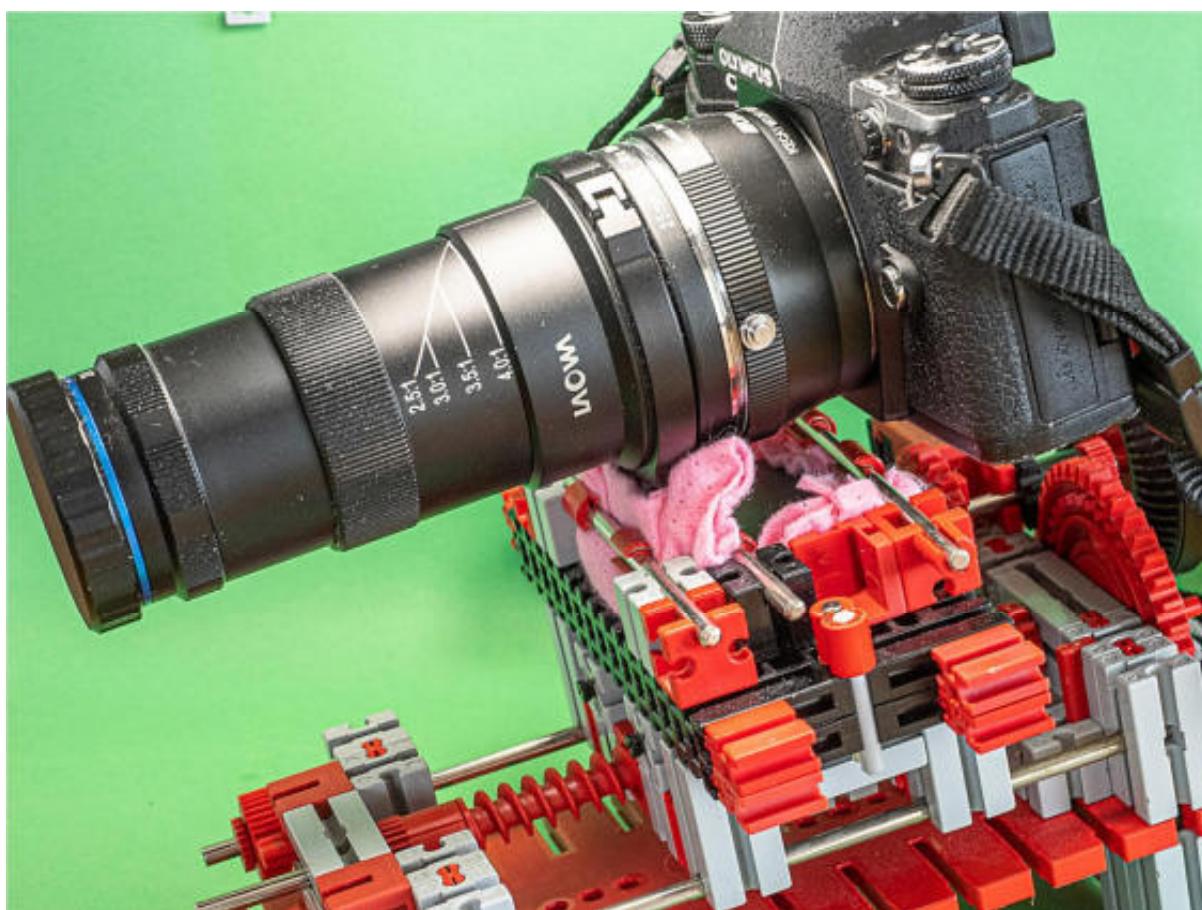


Abb. 7: Beispielhafter Aufbau mit Kamera und Makroobjektiv. Verbindung von Schlitten zu Kameradapter mit Achsen 30 und Bausteinen 7,5. Das Objektiv hat einen Stativring, der mit Metallachsen und Tuch festgeklemmt wurde.

wird die Halterung der Kamera mit dem Objektiv montiert. Der Kameraadapter muss je nach Kameramodell, Objektiv und Zubehörteilen individuell entworfen werden. Wichtig ist die feste und spielfreie Verbindung der Teile. Auch hier schadet mehr Material nicht. Über die zweite Grundplatte wird also auch ein Adaptersystem geschaffen und man kann ohne weiteres verschiedene Ausrüstungen auf mehreren Grundplatten bereithalten.

In der Praxis zeigte sich, dass die Rastung mit Federfuß und selbst das Drücken des Auslösers Vibrationen und kleine, aber merkliche Verschiebungen im Bild erzeugen können. Um verwackelte Aufnahmen zu verhindern, wurde der Vortrieb (3-5 Rastungen) mit dem Reifen 60 bewirkt und dann, ohne den Schlitten zu berühren, die

Kamera ferngesteuert ausgelöst. Da alle Kamerahersteller heute Smartphone-Apps anbieten, die über Bluetooth oder WLAN auf die Kamera zugreifen, ist diese Fernsteuerung kein Problem mehr. Das Live-Bild auf dem Handy gibt's ja gleich dazu.

Wichtig ist noch, dass ein Objekt immer in einer Richtung abgefahren wird. Das liegt am Spiel in der Schnecke; eine Änderung der Drehrichtung führt erstmal zu keiner Verschiebung des Schlittens. Das ist übrigens direkt auf dem Kameramonitor beobachtbar. Die Schärfenebene bewegt sich erstmal nicht, sondern erst, wenn das Spiel überwunden ist. So wird der gesamte Bereich des Objektes abgefahren, der scharf abgebildet werden soll.



Abb. 9: Erlenblüte mit Pollen und Staubgefäßen aus 120 Einzelbildern,
Laowa Ultramakro und Helicon

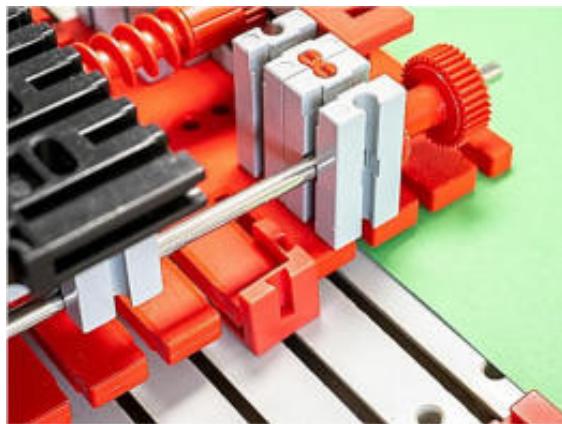


Abb. 8: Verbindung der Basis mit der Grundplatte 1000 des Fotostudios über V-BS 15

Der gesamte Makroschlitten sollte mit dem Objekthalter/-tisch in irgendeiner Weise fest verbunden sein. Ich verbinde die Grundplatte 1000 mit Winkelsteinen oder V-Bausteinen 15 oder anderen Methoden. Auch hier gilt es, die hohe notwendige Präzision und Konstanz des Aufbaus zu beach-

ten. Selbst kleine Verschiebungen und Veränderungen zerstören den Bildstapel und erfordern einen neuen Versuch.

Viel Spaß in der Makrowelt!

Quellen

- [1] Kontakt zum Autor: fabian@fabian-haas.de, auf Instagram [fabian_pos](#) und [fabians_celbodies](#).
- [2] Professionelles Makro- und Mikrofotozubehör z. B. bei [MJKZZ Europe](#)
- [3] Stacking Software: [HeliconSoft](#)
- [4] [Zerene Stacker](#)
- [5] Meisterliches Niveau bei Schichtbildern z. B. bei [Sagaoptics](#), Six FEET Art und auch viel bei Instagram
- [6] Rolf Meingast: *startIDE (4): Focus Stacking*. [ft:pedia 1/2018](#), S. 115–121.

Getriebe

Schmidt-Kupplung

Florian Bauer

Im innoffiziellen Lego-Technik-Buch [1] fand ich das Modell einer Schmidt-Kupplung, einer witzigen Konstruktion, um variable Achsversätze zu ermöglichen.

Die Schmidt-Kupplung ist ein Mechanismus, der die Drehbewegung einer Antriebswelle in zwei Dimensionen radial auf eine Ausgangswelle versetzen kann [2].

Sie wurde von *Richard Schmidt* und *Walter Haarmann* entwickelt [3] und wird unter anderem bei Walzenantrieben mit weiten Versätzen eingesetzt, um die Variation der Dicke der Werkstücke auszugleichen. In Wolfenbüttel gibt es sogar eine Firma SCHMIDT-KUPPLUNG GmbH, die diese Kupplungen anbietet. Auf deren Internetseite sind gute Abbildungen und interessante Informationen zu den Eigenschaften dieses Bauelements zu finden [4].

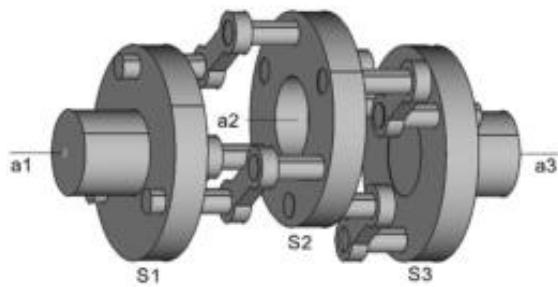


Abb. 1: Schmidt-Kupplung mit drei Scheiben

In Abb. 1 sind die Achsen a1 und a3 über Kurbelgestänge miteinander verbunden. Die Achsen a1 und a3 sind mechanisch vorgegeben. Die Achse a2 stellt sich entsprechend des Achsversatzes zwischen a1 und a3 automatisch ein.

Bei der Schmidt-Kupplung handelt sich um Parallelkurbelgetriebe bestehend aus drei Drehscheiben S1, S2 und S3, die über ein Kurbelgestänge miteinander verbunden sind. Die erste angetriebene Drehscheibe S1

überträgt das Drehmoment der Antriebsachse a1 über drei Kurbelstangen winkeltrau auf die mittlere Scheibe S2, die ihrerseits über drei Kurbelstangen das Drehmoment weiter an die Ausgangsscheibe S3 überträgt.

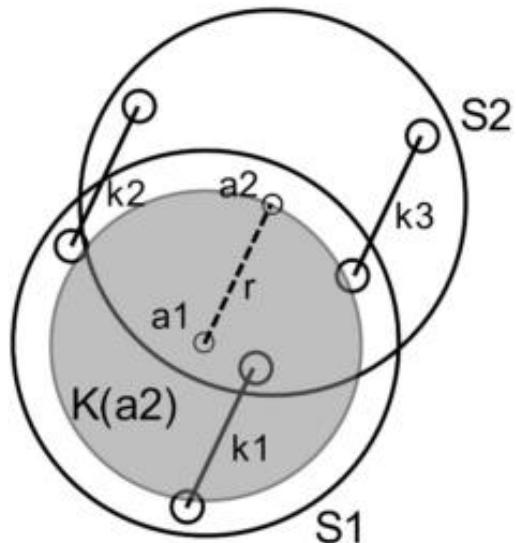


Abb. 2: Antriebs-Scheibe S1 und Mittelscheibe S2 sind durch Kurbeln k1, k2, k3 miteinander verbunden, die eine Drehung um Achse a1 winkeltreu auf Achse a2 übertragen

Wenn man zwei Scheiben mit drei gleich langen Kurbeln (Länge r) miteinander verbindet, kann man bei vorgegebenen Achspositionen eine Drehung 1:1 übertragen. Durch die Verwendung von drei Kurbeln ist sichergestellt, dass es keinen Totpunkt gibt und die Bewegung in allen Winkelstellungen gleichmäßig übertragen werden kann.

Die möglichen Achspositionen der Achse a2 der zweiten Scheibe liegen auf einem Kreis K(a2) mit Radius r der Kurbellänge

um die Achsposition der Antriebsscheibe. Damit lassen sich also Achsversätze der mittleren Scheibe realisieren, bei denen der Mittelpunkt dieser Scheibe auf diesem Kreis liegt.

Man kann dieses Übertragungsprinzip ein zweites Mal anwenden, um einen weiteren analogen Achsversatz der Ausgangsscheibe zu erhalten: Der Mittelpunkt der Ausgangsscheibe liegt auf einem Kreis mit Kurbelradius um den Mittelpunkt der Mittelscheibe.

Die Achsen a1, a2 und a3 können als Drehpunkte eines SCARA-Arms aufgefasst werden. Über die Winkelstellungen der „Arme“, an deren Drehachsen sich die Drehscheiben befinden, kann ein weiter Be-

reich von transversalen Achsversätzen eingestellt werden. Wichtig ist nur, dass die Kurbeln kurz genug sind, um sich ungehindert bewegen zu können.

Interessant ist der Fall ohne Versatz, bei dem die Achsen a1 und a3 zusammenfallen. In diesem Fall ist die Achsposition der mittleren Drehscheibe nicht mehr eindeutig bestimmt. Sie kann dann jeden Punkt auf dem Kreis K(a2) einnehmen.

Wikipedia entnehmen wir einen weiteren Vorteil: „... bei Versatzänderung unter Last und im Betrieb justiert sich das System über das Ausgleichselement Mittelscheibe selbst ein, ohne eine nach außen hin wirkende Reaktion. Radialschwingungen werden von einer Welle nicht auf die andere übertragen.“

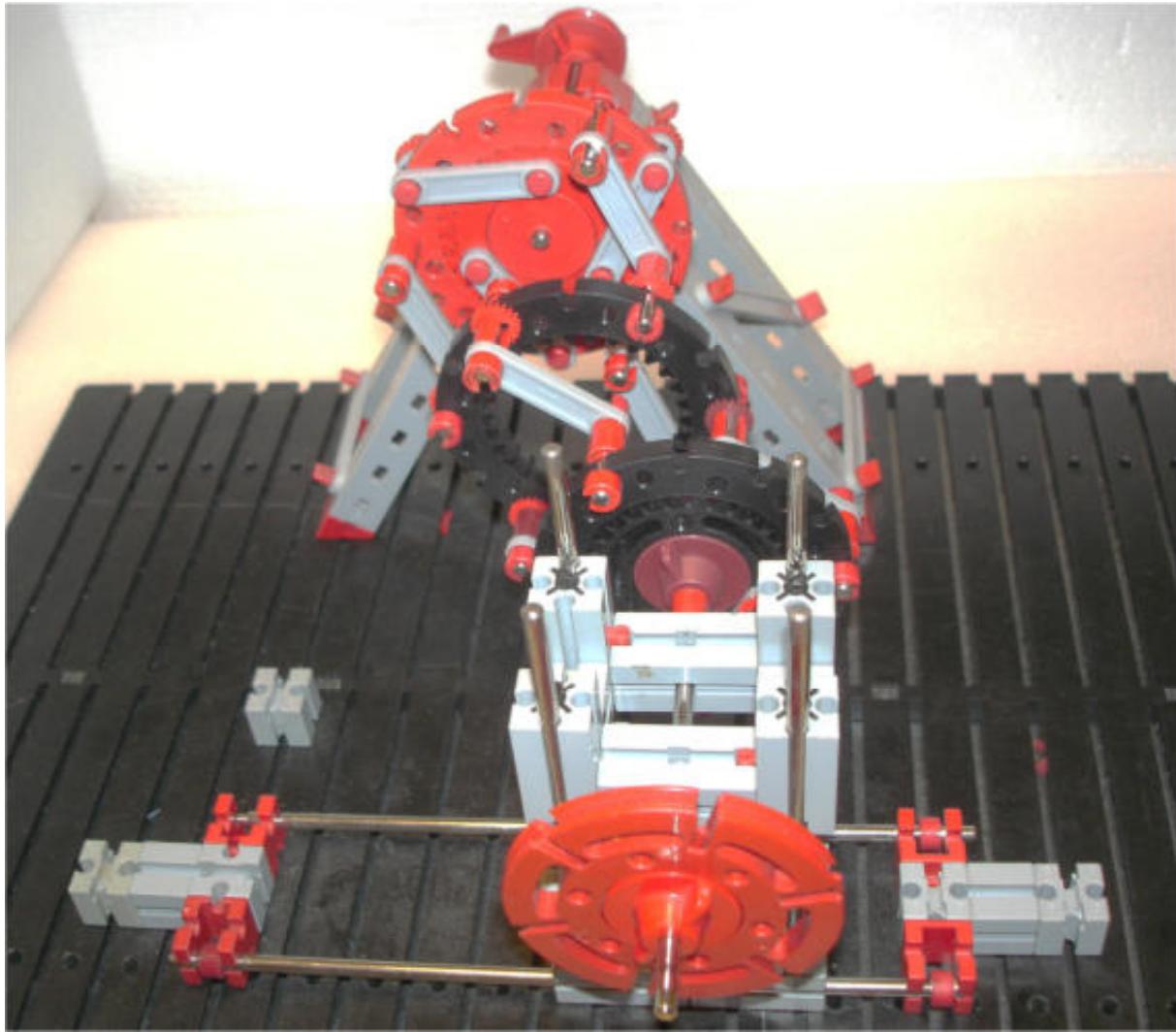


Abb. 3: Modell einer Schmidt-Kupplung mit fischertechnik

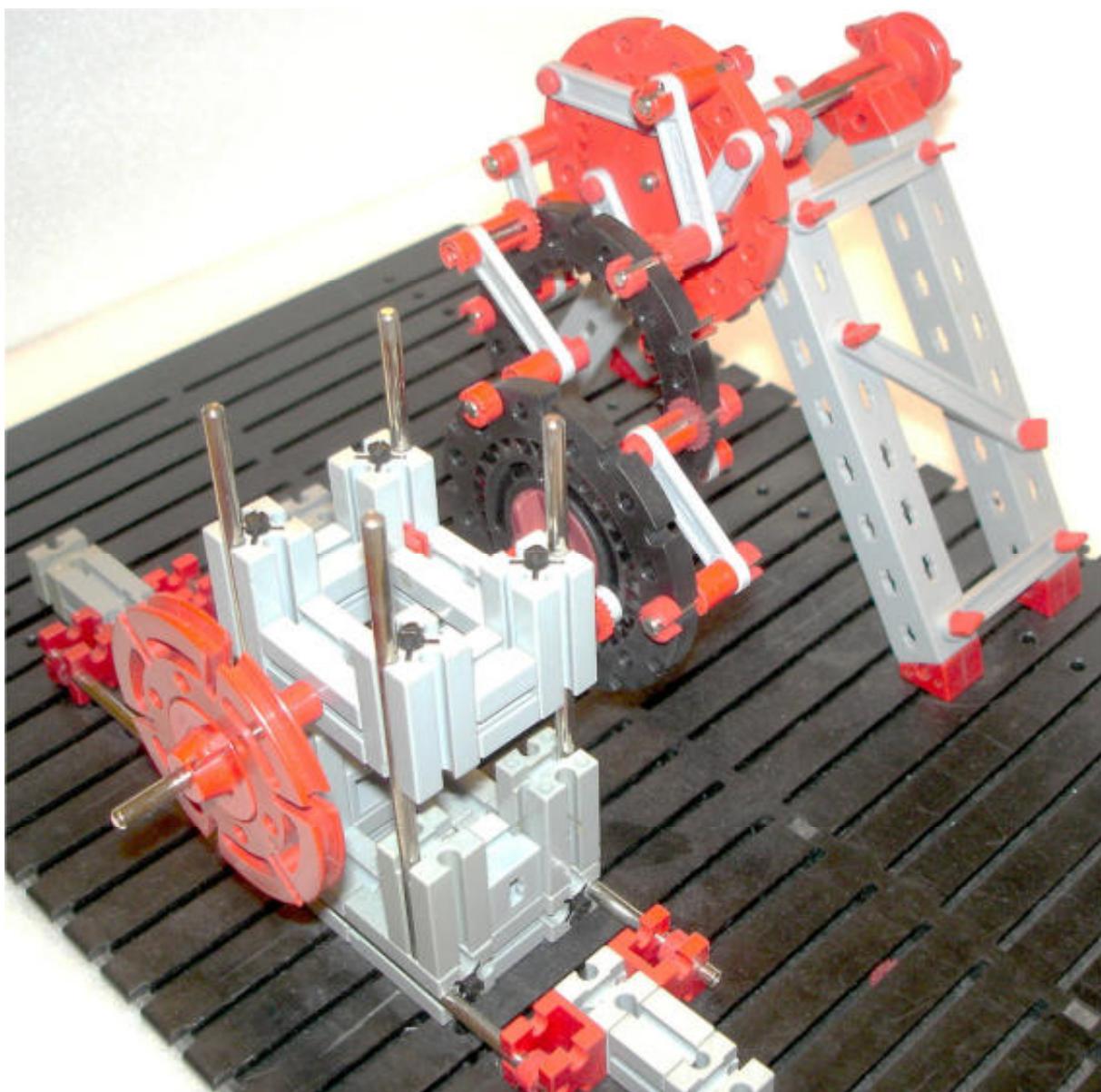


Abb. 4: Modell-Variante einer Schmidt-Kupplung aus fischertechnik mit Innenzahnradern. Um Achsaufnehmende Z40 fest mit den Innenzahnräder zu verbinden, kann man Nylonschnur (Tipp von Harald Steinhäus) oder Papier dazwischen klemmen.

Was mit Lego geht, ist auch mit fischertechnik möglich, wenn auch der Aufbau nicht ganz so kompakt ist.

Bei dem Modell in Abb. 3 lässt sich das Prinzip der Schmidt-Kupplung demonstrieren. Wichtig ist eine stabile und parallele Achsenlagerung. Am „Ausgang“ befindet sich ein Verschiebe-Schlitten, mit dem sich die Achse in der Höhe und Seite verschieben lässt.



Abb. 5: Um das Z40 mit dem Innenzahnrad zu verbinden, kann man Statikstreben einsetzen

Im Forum hat ein kleiner Wettbewerb um den kompaktesten Aufbau stattgefunden. Ein sehr elegantes Modell hat Thomas Püttmann (aka geometer) gepostet (Abb. 6).

Quellen

- [1] Paweł „Sariel“ Kmiec: [Das „inoffizielle“ LEGO-Technic-Buch: Kreative Bautechniken für realistische Modelle](#). dpunkt Verlag GmbH, 2013, S. 88 ff.
- [2] Wikipedia: [Schmidt-Kupplung](#).
- [3] Walter Haarmann: [Wellenkupplung](#). Patentschrift DE 2933722, 21.08.1979 (veröffentlicht am 02.02.1989).
- [4] Fa. SCHMIDT-KUPPLUNG GmbH: [Homepage](#).



Abb. 6: Sehr kompakte Schmidt-Kupplung von Thomas Püttmann (aka „geometer“)

Modell

Die beschwingte Schaukel

Rüdiger Riedel

Schaukeln ist kinderleicht, die physikalische Erklärung ist es nicht. Und dann gibt es noch die Schiffsschaukel...

Wie lässt sich die Bewegung eines schaukelnden Kindes modellmäßig imitieren? Am Saubersten durch Regelungstechnik mit Sensoren, die den Bewegungszustand ermitteln.

Mein Ansatz ist einfacher: Da eine Schaukel wie ein physikalisches Pendel einen einfachen Rhythmus zeigt, verwende ich einen einstellbaren Oszillatoren. Ich nehme dazu die Einstellung „Frequenzgenerator“ des Electronics-Moduls rt-tp ([152063](#)):

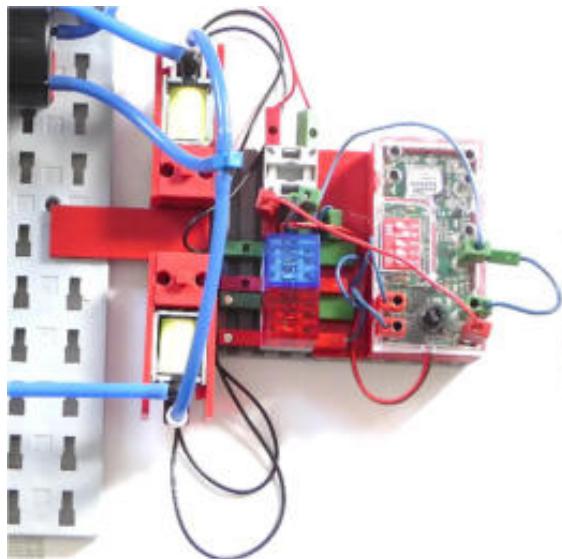


Abb. 1: Der Oszillator

Im Heft „Zusatzprogramme Electronics“ [3] ist im Kapitel 1.4.7 die Schaltung für zwei unabhängige Wechselblinker beschrieben. Wir übernehmen die DIP-Schalterstellungen laut Abb. 1. Die Taster an den Eingängen brauchen wir nicht: E1 wird an den Plus-Pol angeschlossen, E2 und E3

werden mit dem Minus-Pol verbunden. An den Ausgängen O1 und O2 habe ich jeweils die Plus-Seite einer LED angeschlossen und die Minus-Seiten an die Batterie. Parallel zu den LEDs werden zwei Magnetventile angeschlossen. Ich verwende die von fischerfriendsman [2].

Die Pneumatik benötigt noch den Kompressor ([121470](#)) und den Pneumatik-Tank ([156493](#)). Die Ausgänge der Ventile werden mit einem Pneumatik-Zylinder verbunden, der das jeweilige Modell antreibt. Im Falle des Pneumatik-Zylinders mit Feder-rückstellung ([133027](#)) benötigen wir nur einen Magnetventil-Anschluss, der zweite wird dann mit einem Stopfen ([31708](#)) verschlossen.

Schaukeln im Sitzen

Diese Art des Schaukelns sieht man am häufigsten.

Der Antrieb einer Schaukel auf einem Spielplatz, ohne den Schubser einer weiteren Person, beruht auf dem Austausch des Drehimpulses von der Bewegung des Schauklers zur Schaukelschwingung um den Punkt, an dem die Schaukel aufgehängt ist. Weitere Einflüsse wie die Verlagerung des Schwerpunktes kommen dazu.

Das Männlein ist mit einem S-Scharnier ([36329](#)) beweglich gelagert und wird vom Pneumatik-Zylinder 45 zu ruckartigen, alternierenden Drehbewegungen angetrieben.

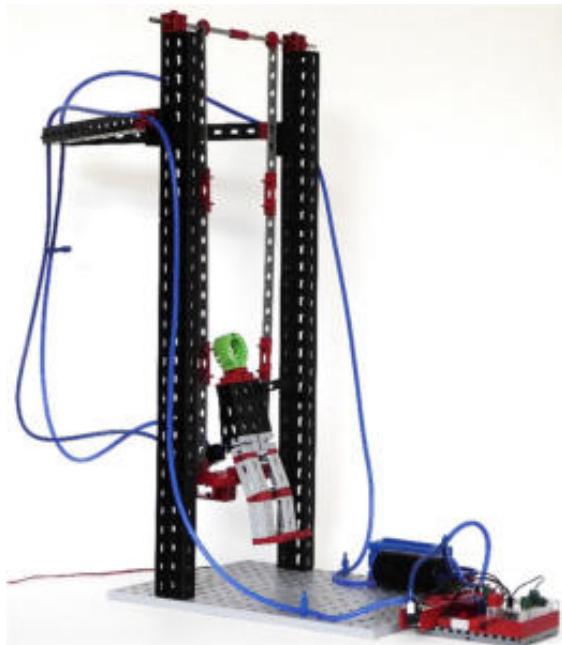


Abb. 2: Die Kinderschaukel

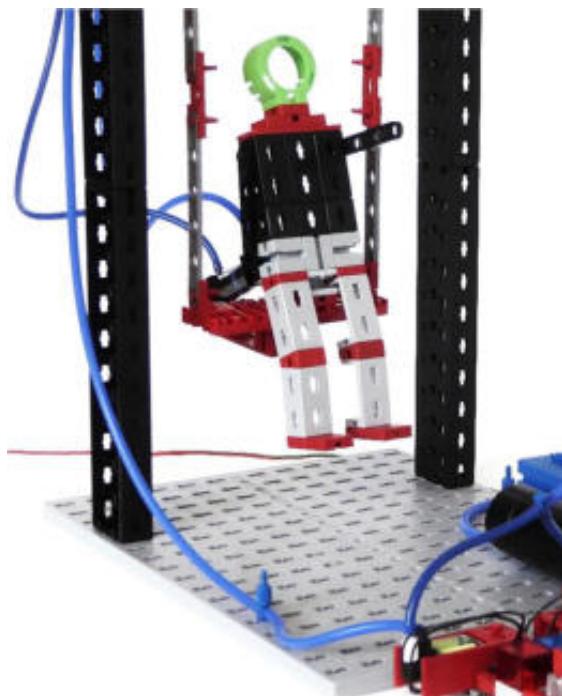


Abb. 3: Die Schaukel

Die Schaukelfrequenz und der Takt der Magnetventile müssen ungefähr übereinstimmen, dann beginnt das Männlein aus dem Ruhezustand heraus zu schwingen und zu schaukeln.

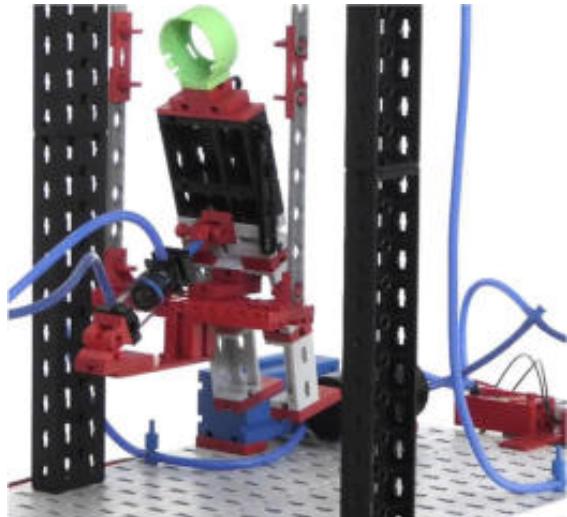


Abb. 4: Ansicht von hinten

Betrachten wir Abb. 5:

Am Ende der Schaukelbewegung nach links wird das Männlein von der Seite gesehen eine Drehung entgegen dem Uhrzeigersinn ausführen, es lehnt sich zurück. Dieser Drehimpuls wird auf die momentan ruhende Schaukel übertragen und treibt damit die Rechtsbewegung an. In dieser liegenden Position schwingt das Männlein nach rechts. Ist die Schaukel am rechten Höchstpunkt, erfolgt die Drehung des Männleins im Uhrzeigersinn und die Schaukel schwingt nach links.

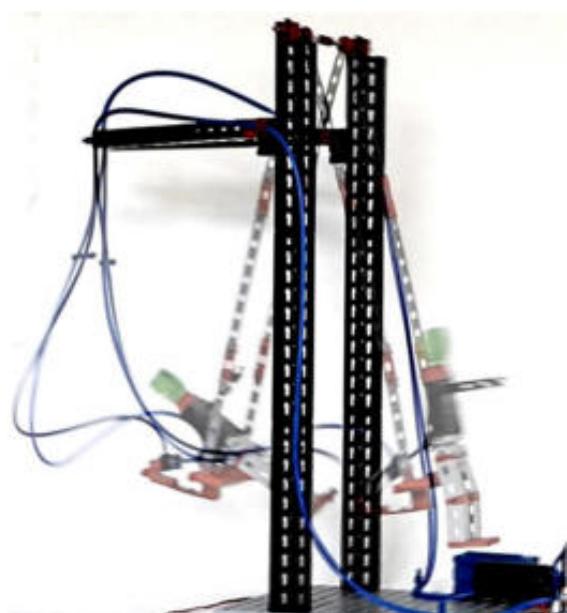


Abb. 5: Die Schaukelbewegungen

Mit der Rückkehr zum linken Höchstpunkt ist ein Takt der Schaukel und des Wechselblinkers abgeschlossen

Schaukeln im Stehen

Erfahrene Kinder können die Schaukel auch im Stehen antreiben. Sie halten sich mit den Händen an den Seilen oder den Ketten fest und bewegen ihren Körper abwechselnd nach vorne und zurück.

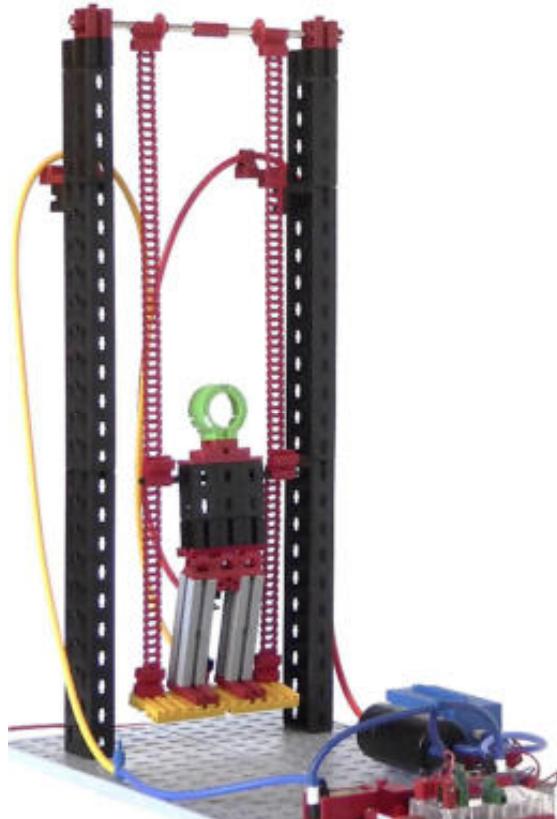


Abb. 6: Schaukeln im Stehen

Das Modell in Abb. 6 imitiert diese Bewegung durch Rotation des Oberkörpers um die Hüftachse. Kinder unterstützen dies mit ihren Händen, sie ziehen sich vor und drücken sich dann zurück.

Statt der Arme verwende ich eine V-Achse 80 ([35405](#)). Deren Enden werden in Bausteine 7,5 geschoben, die mit zwei Förderkettengliedern ([37192](#)) an der Kette befestigt werden.

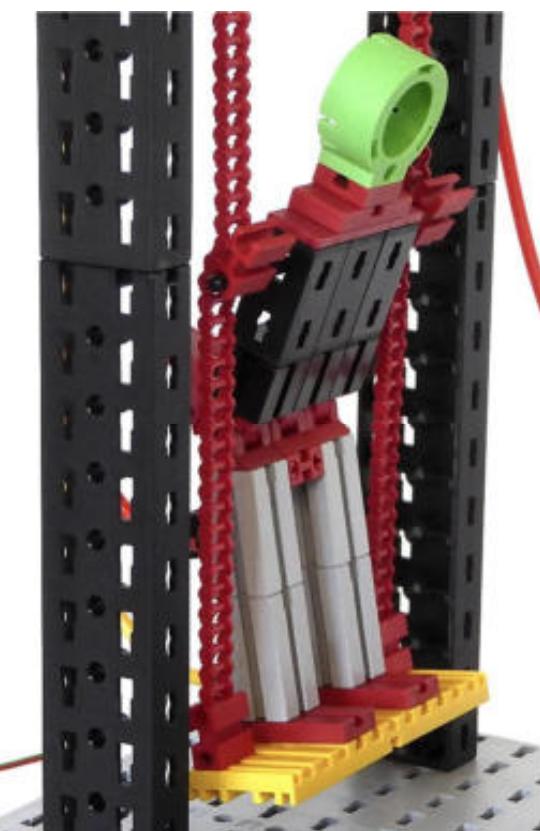


Abb. 7: Haltung für den Vorwärtsschwung

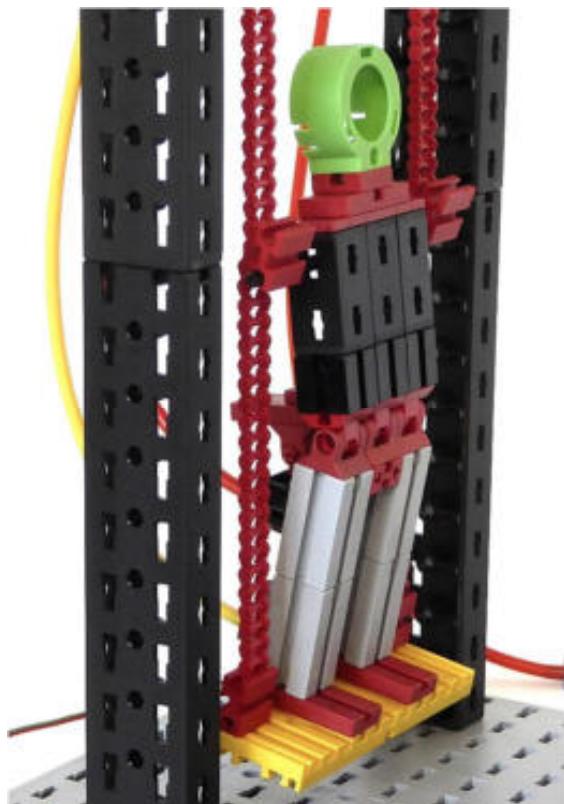


Abb. 8: So beginnt der Rückschwung

Die Füße sind zwei Bausteine $5\ 15 \times 30$ ([35049](#)) und fest mit dem Schaukelbrett „Bodenplatte $30 \times 90 \times 7,5$ “ ([32861](#)) verbunden.

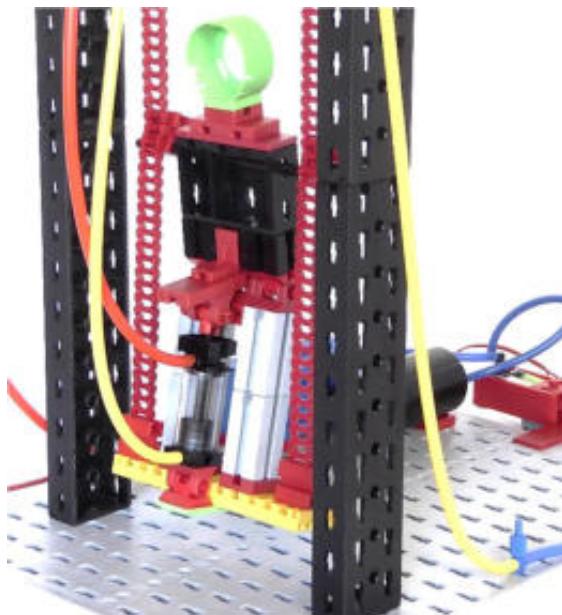


Abb. 9: Die Rückseite

Der Bewegungsablauf wird in Abb. 10 gezeigt.

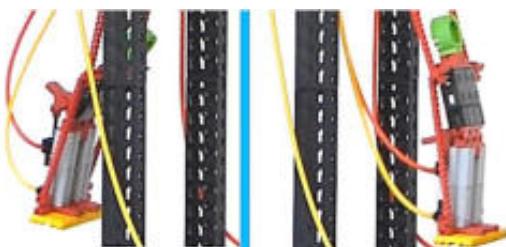


Abb. 10: Schwungholen

Bei der rückwärtigen Auslenkung: Po raus!

Bei der Vorwärtsauslenkung: Brust raus!

Die Schiffsschaukel

... oder das *Parametrische Pendel*.

Wer sich schon einmal in eine Schiffsschaukel gewagt hat, der weiß: Mit der Kinderschaukel hat der Antrieb nichts zu tun, hier wird eine andere Bewegungsabfolge benötigt.

Jetzt müssen nicht nur in den Extremstellungen Bewegungen ausgeführt werden, sondern auch beim Durchgang der Schaukel durch den tiefsten Punkt. In Abb. 11 wird das auf einer Kinderschaukel demonstriert.

Im linken Teil des Bildes geht man in die Hocke, in der Mitte muss man aufstehen um im rechten Teil des Bildes wieder in die Hocke zu gehen.

Eine Schaukelschwingung ist mit der Rückkehr zum linken Teil des Bildes abgeschlossen. In der gleichen Zeit muss der Schaukler zweimal in die Hocke gehen und zweimal wieder aufstehen.

Diese Art des Antriebs nennt man auch „Pumpen“ und dessen Frequenz muss gerade doppelt so hoch sein wie die Schaukelfrequenz. Es ist eine *Parametrische Anregung*. Ein Parameter der Schaukel wird rhythmisch verändert, hier ist es der Abstand des Schwerpunktes von der Aufhängung.



Abb. 11: Die Bewegungen auf einer Schiffsschaukel



Abb. 12: Das Fadenpendel

Mit einem Fadenpendel lässt sich das demonstrieren. Der Motor in Abb. 12 treibt eine Kurbel (bestehend aus einem V-Rad 23×10 ([36581](#)), in dessen Nut eine Klemmbuchse 10 ([31023](#)) mit einer Achse geschoben wird) und hebt und senkt das Männlein. Typisch für die parametrische Anregung ist die Erfordernis des ersten Schubses. Aus dem Stillstand heraus erfolgt kein Schwingungsaufbau, eine kleine Anfangsauslenkung ist nötig. Bei richtiger Motordrehzahl (doppelte Frequenz der Kurbel zur Pendelfrequenz) gerät das Männlein in heftige Schwingungen.



Abb. 13: Das Männlein schwingt hin und her

Hinweis: Das anregende Element muss recht genau auf die erforderliche Frequenz eingestellt werden, was etwas Geduld und eine fein verstellbare Stromversorgung erfordert.

Etwas näher an der Schiffsschaukel ist das Modell in Abb. 14:

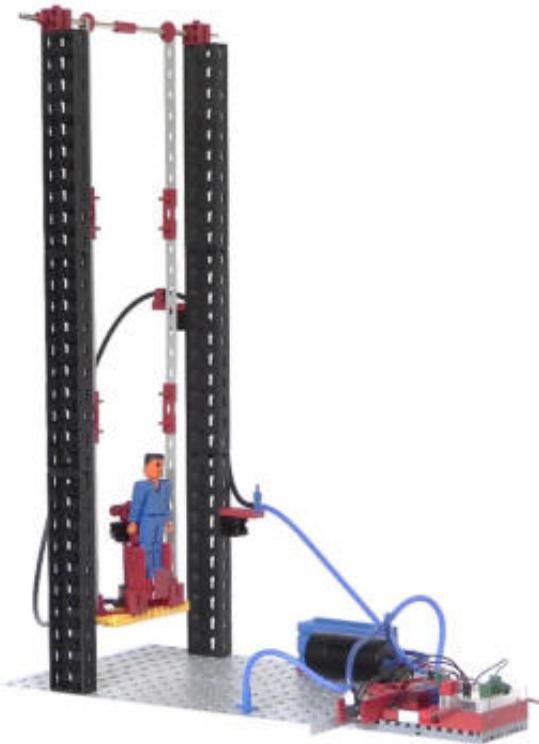


Abb. 14: Die Schiffsschaukel ohne Schiff

Die weiter oben beschriebene Steuerung benötigt jetzt nur ein Magnetventil, der Ausgang des zweiten wird durch einen Stopfen verschlossen. Das Heben und Senken des Schaukelschwerpunktes übernimmt ein Pneumatik-Zylinder mit Federrückstellung ([133027](#)).



Abb. 15: Das Männlein in der unteren Position

Das fischertechnik-Männchen steht auf einem BS15 mit zwei seitlichen BS5. Die Füße werden vorne gehalten von einer E-



Abb. 17: Der Bewegungsablauf

Magnet-Führungsplatte (32455), der Körper wird hinten gestützt durch zwei Bausteine $5\ 15 \times 30\ 3N$ (38428), die mit einer Bauplatte $15 \times 45\ 2Z$ (38242) verbunden sind. Die Anlenkung des Pneumatik-Zylinders erfolgt hier genauso wie bei den anderen Modellen mit einer Gelenkwürfel-Klaue 7,5 (31436) mit zwei hineingedrückten Abstandsringen 3 (31597) und einer V-Achse 20 Rastachse (31690).

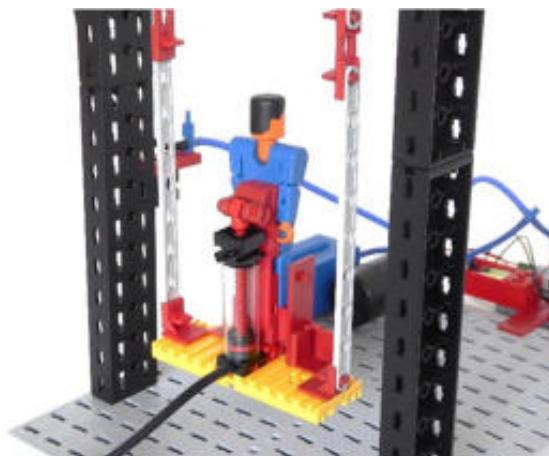


Abb. 16: Rückansicht

Besonders bei der parametrischen Anregung ist eine große Schaukelänge vorteilhaft bei der Verfolgung der Bewegungsabläufe. Die geringere Frequenz hilft auch der pneumatischen Anregung, die ja mit der doppelten Schaukfrequenz arbeiten muss.

Die drei Bilder der Abb. 17 sind Schnappschüsse aus einem Video [4]. In den beiden Endpositionen ist die Schaukel recht scharf und der Zylinder im eingefahrenen Zustand

zu sehen. Wegen der größten Pendelgeschwindigkeit in der unteren Position ist das mittlere Bild unscharf, der ausgefahrenen Zustand des Zylinders und damit die angehobene Position des Männleins sind aber gut zu erkennen.

Woher kommt der Energiegewinn, wenn beim unteren Durchgang der Schaukel ein Gewicht angehoben, aber in den äußeren Positionen wieder abgesenkt wird?



Abb. 18: Motorgetriebenes Pendel

Beim Hub wird nicht nur die Gewichtskraft überwunden, sondern zusätzlich die Zentralkraft, mit der die Schaukel auf der

Kreisbahn geführt wird. Demgegenüber wird bei der in den Umkehrpunkten zum Stillstand kommenden Schaukel nur die potenzielle Energie wieder frei.

Abb. 18 bis 20 zeigen drei weitere Beispiele der parametrischen Anregung.

Ein Merkmal einer parametrisch erzeugten Schwingung ist, dass sie im Unterschied zu einer erzwungenen Schwingung nicht ohne eine anfängliche Auslenkung aus der Ruhe-
lage entstehen kann.



Abb. 19: Pneumatisch betriebenes Pendel

Es gibt einen sehr schönen Artikel von Herbert Möller, in dem das Prinzip der Schiffs-
schaukel und der Spielplatz-Schaukel er-
klärt wird [1].

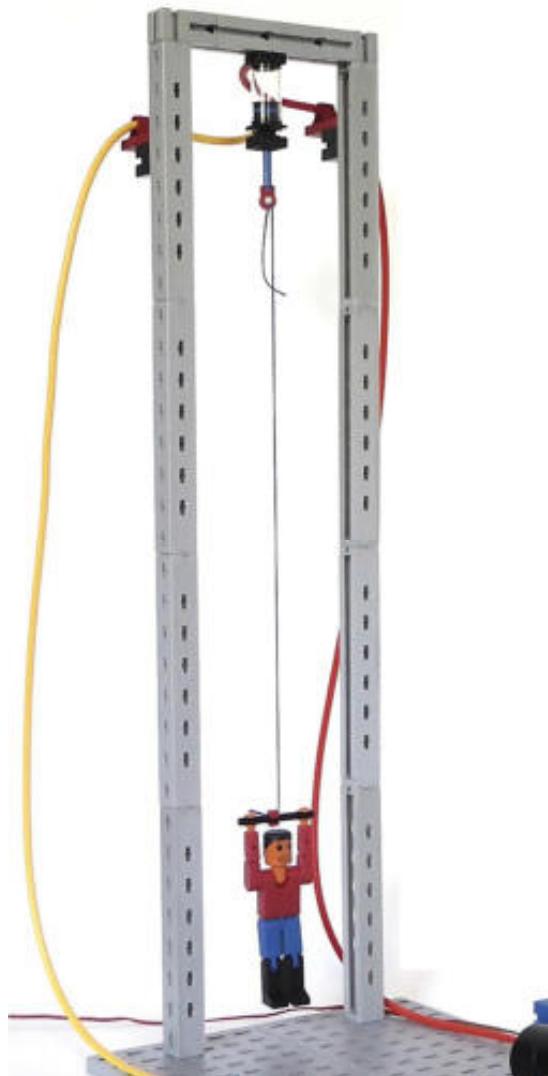


Abb. 20: Pneumatisch angetriebenes Faden-
pendel

Quellen

- [1] Herbert Möller: *Physik des Schaukelns und parametrische Verstärkung*. Auf solstice.de.
- [2] fischerfriendsman: [Magnetventil 11200s](#).
- [3] fischertechnik: [Ergänzendes Material zum Profi Electronics](#).
- [4] Rüdiger Riedel: *Rudis ft Projekt Die Schaukel*. Auf [youtube](#).

Modell

Wellenflug – Wellenflieger

Claus Ludwig

Nachdem ich im März 2020 mit meinem Flipper fertig wurde, der Artikel für die ft:pedia geschrieben und die Fotos gemacht waren, brauchte ich eine neue Idee. Die Modellideen, die ich gesammelt hatte, zogen mich gerade nicht an, sodass ich nach etwas Neuem suchte. Also begann ich alle Bilderpools und Archive zu fischertechnik systematisch durchzusehen. Und dabei stieß ich bei den Kollegen aus den Niederlanden [1] auf die Videos von und über die Modelle von Alfred Pettera.

Die Idee

Da gab es in einem der Videos einen „StarFlyer“, der mir sehr gut gefiel und etwas vorher eine eigene Konstruktion von Alfred Pettera, um einen Wellenflug darzustellen. Und da zündete meine neue Idee; auf der Grundlage des „Star Flyer“ einen Wellenflug zu bauen. Ich stellte fest, dass es bisher nur ein Modell dazu von Harald Steinhaus gab, das auf der Ausstellung in Erbes-Büdesheim 2011 gezeigt wurde (Abb. 1). Und das, so lernte ich später, wo doch der Wellenflug/-flieger eines der am häufigsten gebauten Karussells ist: über 200-mal, und in der ganzen Welt anzutreffen [2].



Abb. 1: Wellenflieger von Harald Steinhaus

Von Alfred Pettera entdeckte ich dann auch noch einen Artikel beim niederländischen fischertechnik-Club zu seinem „StarFlyer“ [3].



Abb. 2: Aufbau des Wellenfliegers
(Bild: Harald Steinhaus)

Der Wellenflug oder Wellenflieger wurde Anfang der 1970er Jahre entwickelt. Dabei handelt es sich um ein Kettenkarussell, bei dem der Gondelträger (an dem die Sitze hängen) an einem Mast nach oben fährt. Der Mast ist zum einen im oberen Teil leicht geknickt (ca. 14 Grad) und zum anderen dreht er sich langsam in die entgegengesetzte Richtung zum sich deutlich schneller drehenden Gondelträger. Durch die gegenläufigen Bewegungen und den abgeknickten

Mast entsteht dabei die Wellenbewegung [2].



*Abb. 3: Der Aufbau ist fast fertig
(Bild: Harald Steinhaus)*



*Abb. 4: Wellenflieger im Betrieb
(Bild: Harald Steinhaus)*

Die Säule

Die Säule ist als Skelett-Konstruktion gebaut. Besonders ist, dass die Querstreben bei zwei gegenüber liegenden Seiten außen und bei den anderen zwei Seiten innen liegen. Das geht nur, indem die zur Befesti-

gung verwendeten S-Riegel zum Teil so bearbeitet werden, dass sie einem flachen Niet entsprechen.



Abb. 5: Das Modell



Abb. 6: Der Fuß der Säule

Als Drehkranz wurde anstelle eines normalen fischertechnik-Drehkränzes ein Drehkranz von Andreas Tacke mit Kugellager verwendet. Um die im Oberteil insbesondere durch den Knick auftretenden Kräfte

abfangen zu können, wurde der Drehkranz zusätzlich mit vier Auslegern und Stützrädern versehen.

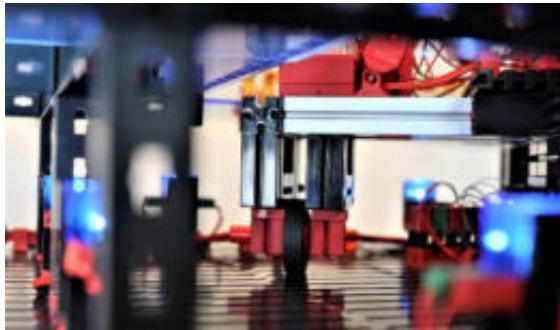


Abb. 7: Ausleger mit Stützrad

Zur Stromversorgung des Gondelträgers (Antrieb und Beleuchtung) gibt es an den Seiten der Säule, an denen die Querstreben innen liegen, jeweils zwei Messingschienen. Die Stromübertragung auf den Mast erfolgt über einen 12poligen Schleifring von fischerfriendsman.

Der Gondelträger

Der Gondelträger wurde in Anlehnung an die Vorlage von Alfred Pettera erstellt. Das Herzstück für den Antrieb des drehbaren oberen Teils bildet dabei ein Innenzahnrad aus Kettengliedern. Der untere Teil hat zum einen die Aufgabe der stabilen Führung des drehenden Innenkranzes und zum anderen der sicheren Führung des gesamten Gondelträgers am Mast.



Abb. 8: Ketten-Innenzahnrad



Abb. 9: Unterer Teil des Gondelträgers

Der Gondelträger ist in der Höhe etwas kompakter als die Vorlage, um den „Knick“ zu durchfahren. Dazu waren an einigen Stellen Modifikationen von Teilen erforderlich. So wurden sowohl die oberen Rollenlager als auch die Halter der Sitz-Ausleger (Baustein 15 · 30 · 5 mit 3 Nuten) etwas angeschrägt. Die einzelnen Teile des gesamten Gondelträgers, insbesondere des drehbaren Teils, wurden nach Abschluss der Tests mit Sekundenkleber fixiert. Damit soll dauerhaft ein ruhiger Lauf gewährleistet werden. Das gilt auch für die Kette, die als Innenzahnkranz fungiert.

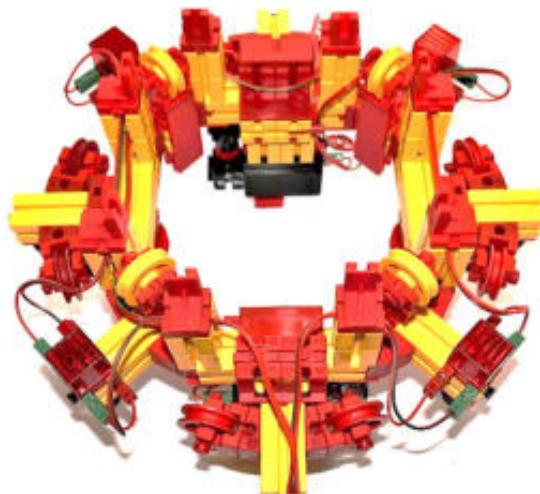


Abb. 10: Unterer Teil des Gondelträgers, umgedreht



Abb. 11: Detailblick auf die Rollen



Abb. 13: Blick auf die Kettenführung



Abb. 12: Ober- und Unterteil vereint

Zur Stromabnahme von den Messingschienen dienen Lokfederschleifer mit entsprechender Bodenplatte, die über Federgelenksteine am unteren Teil des Gondelträgers befestigt sind. Damit die Lokfederschleifer – die im Original leicht gewölbt sind – sicher arbeiten, wurden sie plan geschliffen. Bei den Federgelenksteinen wurde die graue Variante verwendet und mit mattroter Modellbaufarbe lackiert – die schwarzen Bausteine sind schwer zu bekommen.



Abb. 14: Blick von der Seite



Abb. 15: Stromabnehmer

Der Knick

Meine erste Idee war es, dass der Gondelträger gerade bis ans obere Ende des Mastes fährt und anschließend der obere Teil des Mastes motorisch gekippt wird. Das Ganze sollte über mehrere Taster und zwei Electronics-Module gesteuert werden.



Abb. 16: Der Knick

Leider zeigte sich in der Praxis, dass meine Konstruktion nicht die Stabilität hatte, die ich mir gewünscht hätte – was dazu führte, dass die Betätigung der Taster zum Stoppen der Aufwärtsfahrt, aber insbesondere zum Schalten des Abknickvorganges, mal funktionierte und mal nicht. Das Ganze war zu labil.

Dann probierte ich, wovon ich ursprünglich nicht geglaubt hatte, dass es funktioniert: die Säule mit einem festen Knick zu bauen. Und zu meinem Erstaunen schaffte der Gondelträger es, über den Knick zu fahren. Anfangs noch etwas hakelig, aber später, nach einigen Nacharbeiten, war ich sehr zufrieden.

Sitze und Aufhängung



Abb. 17: Sitz in Kettenaufhängung

Die Sitze und die Aufhängung wurden, soweit erkennbar, von der Idee von Alfred



Abb. 18: Aufhängung der Sitzgondeln am Ring

Pettera übernommen. Jeder Sitz hängt an zwei mal 47 Kettengliedern (ursprünglich waren es je 50). Nach dem Bau der Plattform, der Aufgänge und der Geländer zeigte es sich, dass hier noch etwas gekürzt werden musste. Da mir rote Sitze am besten gefielen, diese aber sehr schwer zu bekommen sind, wurden auch hier mit Hilfe mattroter Modellbaufarbe aus schwarzen Sitzen rote Sitze.



Abb. 19: Schrägstellung der I-Streben 120

Die Aufhängung der Sitze an I-Streben 120 und deren Befestigung an Rollenböcken führt dazu, dass die Streben leicht schräg nach oben geneigt sind. Das gibt dem Ganzen noch einen besonderen „Schick“.

Steuerung und Antrieb

Meine ursprüngliche Idee war ja, wie bereits beschrieben, dass der Gondelträger gerade bis ganz nach oben fährt und dann der obere Teil des Mastes motorisch geknickt wird. Dies war jetzt nicht mehr erforderlich. Es blieb das halbautomatische Hochfahren und das entsprechende Herunterfahren. Hierfür wird ein Electronics-Modul mit Grundprogramm verwendet, Endlagschalter sowie jeweils ein Taster zum Auslösen der Aufwärtsfahrt und der Abwärtsfahrt. Als Motor dient ein alter M-Motor mit entsprechendem Getriebe und Seiltrommel.

Für den Antrieb des oberen Teils des Gondelträgers haben die Tests gezeigt, dass zum

einen zwei statt eines Motors (Alfred Pettera) sinnvoll sind – insbesondere im oberen Bereich der Neigung. Zum anderen können die 9 V nach fischertechnik-Standard begrenzend wirken. Eine Reserve von bis zu 12 V ist sinnvoll. Daher suchte ich nach entsprechenden Reglern, die ich im Internet fand: PEMENOL PWM Motor Drehzahlregler, 5-30 V, max. 15 A, max. 150 W. Als Motoren dienen zwei S-Motoren.



Abb. 20: Steuerelektronik

Aus Symmetriegründen habe ich auch für die Steuerung der Mastdrehung den o. g. Regler verwendet, obwohl hier weniger als 9 V erforderlich sind. Der Mast dreht sich ja mit deutlich geringer Geschwindigkeit in entgegengesetzter Richtung zum Gondelträger. Als Antrieb dient ein alter Powermotor 8:1.

Beleuchtung

Die Beleuchtung der blauen Bodenplatten erfolgt mit zehn LEDs (breit abstrahlend) Marke fischerfriendsman, die mit blauen Rastkappen kombiniert sind. Die LEDs auf den unteren Maststabilisatoren sowie im inneren des Mastes werden über ein Electronics-Modul als Baustellen-Lauflicht gesteuert. Auch die Blicksteuerung der vier am Gondelträger angebrachten LEDs erfolgt über ein weiteres Electronics-Modul. Am oberen Ende des Mastes befinden sich noch vier Spot-LEDs im Dauerbetrieb mit roten Rastleuchtkappen. Den Abschluss bildet

eine Rainbow-LED in der kleinen Kugel oben auf der Mastspitze.

Kassenhäuschen



Abb. 21: Kassenhäuschen

Das Kassenhäuschen dient auf der Vorderseite als Kassenhäuschen und auf der Rückseite als Steuerzentrale für den Wellenflug. Die Flieger auf dem Dach und die Welle am oberen Teil der Vorderseite sollen den Wellenflug-/flieger symbolisieren. Angetrieben werden die Flieger durch einen auf ca. 3 V geregelten Minimotor. Der Regler (Niederspannungs-DC-Motordrehzahlregler) ist auch über das Internet bezogen. Beleuchtet werden die Flieger durch zwei Spot-LEDs.

Die Ein-/Ausschalter sind für jeweils einen der beiden großen Regler zuständig. Die beiden mittleren Taster dienen zum Ein- und Ausschalten der Lichtfunktionen und für die Versorgung der Aufzugsfunktion.

Die vorderen Taster sind für die Auf- bzw. Abfahrt zuständig. Drei weitere LEDs beleuchten den Innenraum und zwei Rainbow-LEDs befinden sich an den unteren Ecken der Vorderseite.

Stromversorgung

Die Stromversorgung erfolgt über drei Netzteile: jeweils eines für den Antrieb des Gondelträgers und den Antrieb des Mastes mit jeweils 12 V, das dritte für die Versorgung des Aufzuges und der Beleuchtung mit 9 V.

Abb. 22: *Im Flug* (1)Abb. 24: *Im Flug* (3)Abb. 23: *Im Flug* (2)

Quellen

- [1] Website des niederländischen fischertechnik-Clubs: <https://fischertechnikclub.nl>
- [2] Wellenflug auf [Wikipedia](#)
- [3] Petterra, Alfred: *Star Flyer*. In Ausgabe [April 2009](#) des niederländischen Clubheftes.
- [4] Steinhaus, Harald: *Wellenflug*. Im [Bilderpool](#) der ft:Community.

Modell

Die unendliche Lemniskate

Arnoud van Delden

Vor einiger Zeit dachte ich zusammen mit einem Clubmitglied über ein Modell nach, bei dem eine Acht-Form ausgeführt werden sollte. Die Herausforderung bestand darin, dass ein Element wie ein Karren oder eine Kugel einen Weg gemäß einer „Bogenform“ oder einer „Lemniskaten“-Kurve beschreibt. Dadurch entsteht eine Bewegung, die sich im Zentrum schneidet. Wenn das keine Herausforderung ist ...

Hintergrund

Der Begriff „Lemniskate“ leitet sich vom griechischen Wort für „hängende Schnur“ oder Girlande ab. Als geometrische Figur hat diese Form eine lange Geschichte und hat nicht nur mathematische, sondern auch symbolische Bedeutung erlangt. Die Form ist beispielsweise ein Vorbild für das Symbol für Unendlichkeit (∞) in der Mathematik und wird als Symbol für das ewige Leben und das Zusammenspiel gegensätzlicher Kräfte verwendet.

Beim Tarot-Kartenspiel balanciert eine Figur zwei Pentagramm-Münzen oder Talismane in einer Lemniskate, um Unendlichkeit und Ewigkeit darzustellen (Abb. 2). Indirekt kommt die Lemniskate auch in der Natur vor: Wer einen Stock in den Boden steckt und die Spitze des Schattens markiert, den die Sonne ein Jahr lang täglich auf den Boden wirft, wird irgendwann auch diese Acht erscheinen sehen.



Abb. 1: Arduino-Logo

Ein Symbol mit unendlichen Möglichkeiten und Anwendungen. Vielleicht hatte das italienische Designteam von Arduino das im Hinterkopf, als es nach einem geeigneten Logo für seine mittlerweile beliebte Familie von Single-Board-Mikrocontrollern suchte?



Abb. 2: Lemniskate im Tarot-Kartenspiel

Ursprung

Die mathematische Relevanz von Achterkurven scheint zuerst von *Proclus* erkannt worden zu sein, einem griechischen, neu-platonischen Philosophen und Mathematiker, der im 5. Jahrhundert n. Chr. lebte. Proclus betrachtete die Querschnitte eines Torus durch eine Ebene parallel zur Torusachse. Wie er bemerkte, besteht der Querschnitt der meisten dieser Abschnitte aus einem oder zwei Ovalen. Wenn die Ebene jedoch tangential zur Innenfläche des Torus verläuft, nimmt der Querschnitt die Form einer Acht an (Abb. 3).

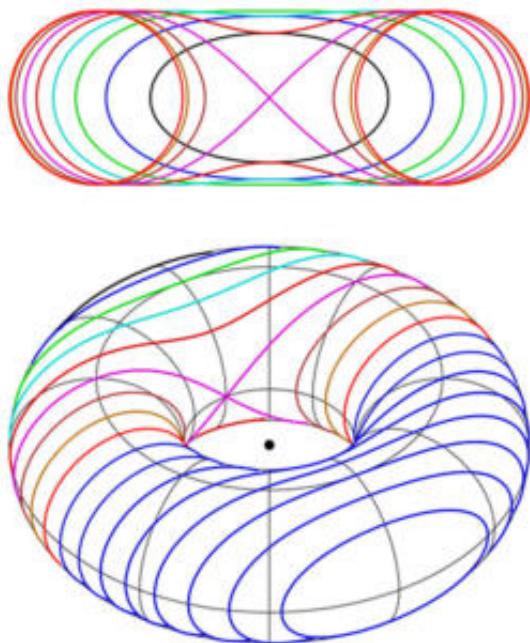


Abb. 3: Torus-Querschnitt

(Quelle: Ag2gaeh, [Wikimedia](#), CC BY-SA 4.0)

Diese Form erinnerte ihn an die Fesseln, mit denen zwei Pferdefüße beim Transport zusammengehalten werden können. Proclus nannte die Form daher „Pferdemanscheten“ oder „Hippopede“ (griech. ἵπποπέδη). Der Name Booths Lemniskate für diese Kurve geht auf die Arbeiten des Mathematikers *James Booth* (1810-1878) aus dem 19. Jahrhundert zurück [1].

Wenn ihr diese Form selbst weiter recherchieren werdet ihr schnell feststellen, dass es verschiedene Varianten gibt. Zum Beispiel Bernoullis Lemniskate (griech. λημνίσκος, Band), die Jakob Bernoulli 1694 in einem Artikel in seiner *Acta Eruditorum* vorgeschlagen hat. Oder die Lemniskate von Gerono, eine seitlich abgeflachte Variante des französischen Mathematikers Camille-Christophe Gerono (1799-1891) [2].

Mathematische Beschreibung

Eine Booth-Hippopede ist sowohl bezüglich der x-Achse als auch der y-Achse symmetrisch und kann mathematisch beschrieben werden als

$$(x^2 + y^2)^2 = (ax^2 - by^2)$$

mit Faktoren $a > 0$ und $b > 0$ (Abb. 4) [3].

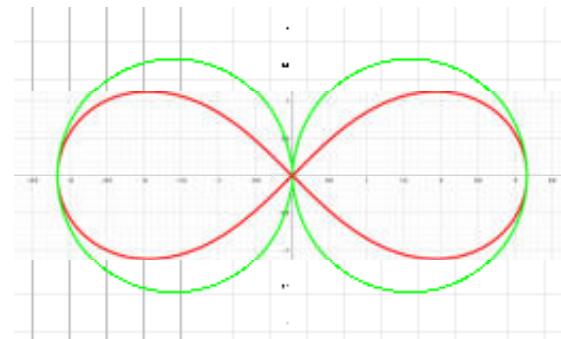


Abb. 4: Booth-Hippopede

Wenn die Faktoren weiter auseinanderliegend gewählt werden, wird die Lemniskate aus immer reineren Kreisen anstelle von gestreckten Ellipsen bestehen. Das lässt sich im praktischen Online-Visualisierungstool GeoGebra schön veranschaulichen (Abb. 4).¹

Für die rote Kurve wurden die gleichen Faktoren gewählt und wie ein globaler Skalierungsfaktor verwendet; die Formel lautet

$$(x^2 + y^2)^2 = (10x^2 - 10y^2)$$

¹ Ein umfangreiches Werkzeug zur Visualisierung mathematischer Formeln ist [GeoGebra](#).

Für die grüne Kurve wurden die Faktoren umgekehrt proportional gewählt:

$$(x^2 + y^2)^2 = \left(10x^2 - \left(\frac{1}{10}\right)y^2\right)$$

Die Bewegung

Ein logischer erster Schritt schien zu sein, mit einem fischertechnik-Modell eine möglichst reine Lemniskatenform mechanisch zu beschreiben. Inspiriert von dem Artikel über Jansens „Strandbeester“ in der letzten ft:pedia [4] suchte ich nach einer Variante der „Beinmechanik“, mit der die Acht mittels Gestänge und Getriebe erzeugt werden könnte. Einige Konstruktionsexperimente und eine Online-Suche führten schließlich zu einer interessanten Geschichte über einen Erfinder², der in den 1990er Jahren herausfand, dass seine Idee bereits 1957 von einem Clayton L. Vagneur patentiert wurde (Abb. 5).

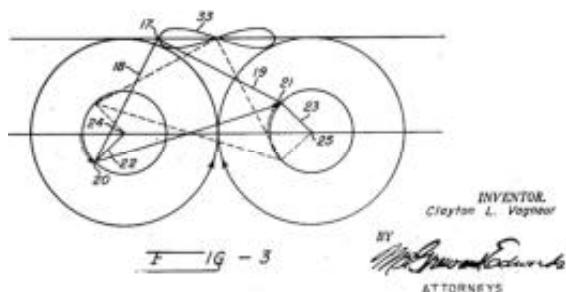


Abb. 5: US-Patent von Vagneur von 1957 [5]

Das vollständige Patent einschließlich der vollständigen Beschreibung dieser Mechanik ist online über Google verfügbar [5]. Vagneurs Ausgangspunkt war aber nicht, die Lemniskatenkurve mechanisch zu reproduzieren. Er wollte ein Getriebe, das möglichst wenig seitlichen Druck auf einen Kolben ausübt. Mit seinem Patent meldete er ein allgemeines Verfahren zur Umwandlung einer Rotation in eine hin- und hergehende (Gleit-) Bewegung für den Einsatz in Verbrennungsmotoren und Pumpen an. In seiner Originalskizze fällt jedoch sofort die Lemniskatenform (Kurve Nr. 33) ins Auge.

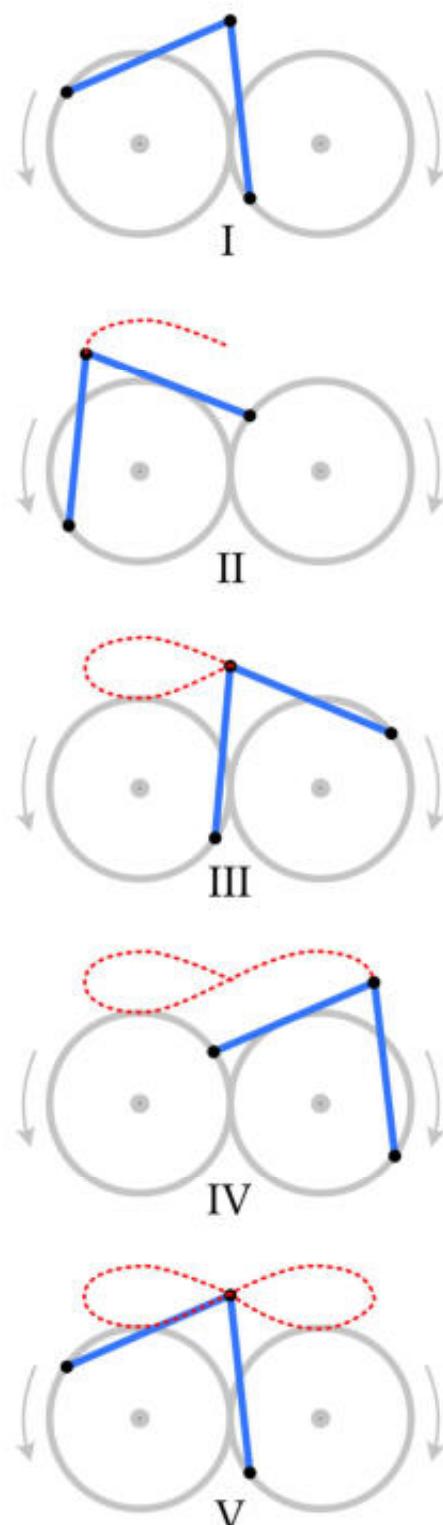


Abb. 6: Bewegungsverlauf der Antriebsräder

² Sansoy über seinen ‚Figure Eight Motion Drive‘.

Die Basis dieses Mechanismus bilden zwei Pleuel, die in einem Drehpunkt am Rand von zwei miteinander synchronisierten Rädern verbunden sind. Die Räder haben eine entgegengesetzte Drehrichtung und sind um eine Vierteldrehung (90°) Phasen verschoben. In Abb. 6 sind einige Stufen einer vollständigen Umdrehung dieser Antriebsräder dargestellt.



Abb. 6: Hippopede-Kurve nach Vagneur

Der Drehpunkt der Pleuel durchläuft damit eine Hippopede-Kurve.

Diese ist am größten, wenn die Drehpunkte möglichst an den Außenkanten der fischertechnik-Drehkränze ([31390](#), [31391](#)) platziert werden (Abb. 7, 8).

Auf diese Weise ist es auf jeden Fall möglich, eine Lemniskatenform aus dem alten amerikanischen Patent von Vagneur zu zeichnen.

Der Weg dieser Bewegung verläuft jedoch noch nicht entlang der Kurve. Als Grundlage für ein Modell, bei dem sich ein montiertes Element (z. B. eine fischertechnik-Figur) entlang der Bahn bewegen muss, erscheint diese Lösung noch nicht optimal. Da sich die gewünschte Bahn selbst schneidet, scheint eine Konstruktion mit einem fest montierten Element im Kurvenschnitt ein mechanisches Problem darzustellen. Eine mögliche Alternative besteht natürlich darin, das Element gar nicht zu montieren, sondern ihm nur die gewünschte Bewegung zu geben.



Abb. 8: Mechanische „Zeichnung“ der Hippopede-Kurve

Zum Beispiel muss es möglich sein, einen Wagen oder eine Kugel in einer sich bewegenden Bahn herumrollen zu lassen. Dies eröffnet auch die Möglichkeit, ggf. für den Vortrieb, das bewegte Element an der Kreuzung „aufzugeben“ und so den Kreuzungsknoten mechanisch komplett zu vermeiden.

Für den Bau einer solchen Fahrbahn scheinen die flexiblen fischertechnik-Kugelbahnen mehr Möglichkeiten zu bieten als die gebogenen Statik-Kurven mit festem Radius.

Folge der Kurve

Bevor ich mich an eine Art mechanisches „Transfer“-System des sich drehenden Elements in der Mitte wagte, versuchte ich zuerst, Booths Hippopede (die rote Kurve in der obigen Grafik) mit einem Kettenmechanismus anzunähern. Ein Magnet, mit einer Heißklebepistole an der Kette befestigt, beschreibt die Acht. Der kritische Punkt bei dieser Lösung bleibt der Schnittpunkt der Figur und in diesem Fall der Ketten. Um

den Flachmagneten passieren zu lassen mussten sogar einige Führungsrollen platziert werden (Abb. 9). Durch ein Kippen der Achsen der beiden Drehkränze gegeneinander um 15° konnte die Kette entlang der Kurve geführt werden (Abb. 10).



Abb. 9: Kreuzung der Kette

Ein seitlich auf ein Kettenglied aufgeklebter Neodym-Magnet kann den anderen Magneten im Sockel in einer nahezu perfekten Lemniskatenbewegung auf der Plexiglasplatte bewegen.

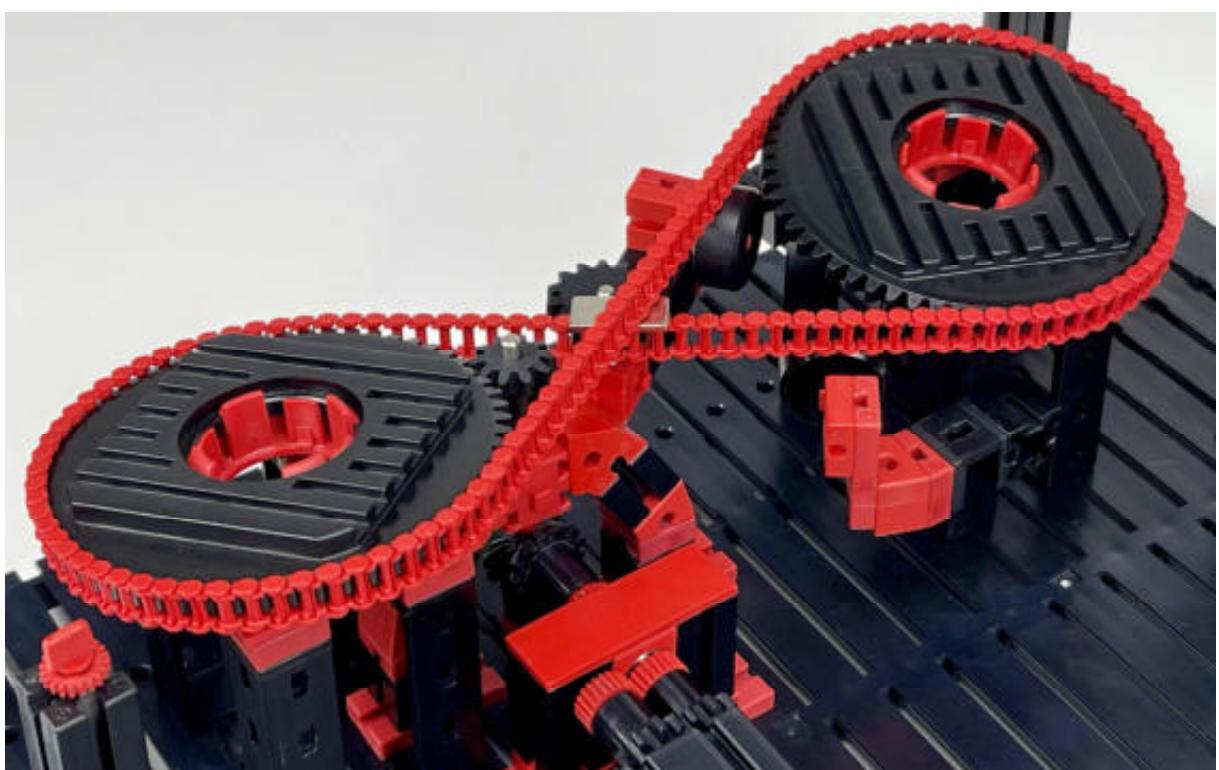


Abb. 10: Drehkränze mit gekippten Achsen

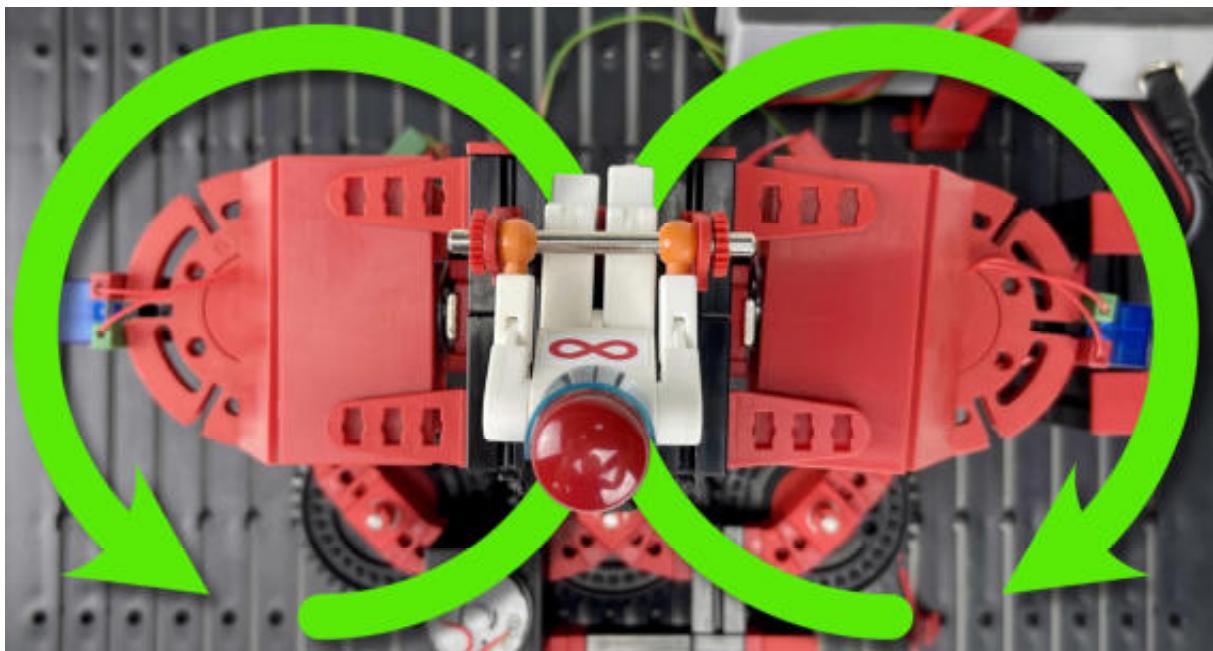


Abb. 11: Booth-Hippopede mit „Übergabe“ eines Testfahrers

Lorentz Lemniskate

In den bisherigen Modellen wird ein ziemlich genaues Booth-Hippopede mit gleichen Faktoren a und b nach der oben genannten Formel (rote Kurve) beschrieben. Auch beim zweiten Modell ist die Reibung der Bewegung in die Kurve erheblich. Eine genauere Spurbeschreibung könnte mit einem zweiten Magneten an der Kette erzwungen werden, aber auch dann ist das Gewicht des Umlaufelementes durch den Einsatz von Permanentmagneten sehr begrenzt. Selbst die kräftigen Neodym-Magnete überbrücken die maximale Distanz im Schnittpunkt der Ketten nur mühsam.

Deshalb suchte ich nach einer Konstruktion mit zwei gegeneinander rotierenden Rädern, die jeweils das bewegliche Element in der Mitte voneinander übernahmen. Auf diese Weise näherte ich mich Booths grün gezeichnetem Hippopede, bei dem die Faktoren a und b umgekehrt proportional sind.

Nach einigen Experimenten mit einer mechanischen Lösung (die es sicher geben

sollte) bin ich zum Prinzip des Magnetismus zurückgekehrt. Nur in diesem Fall und auch, weil sie steuerbar sein müssen, mit Elektromagneten.³



Abb. 12: Testfahrer Lorentz Lemniskate

Dies führte zu der folgenden Konstruktion, die es unserem Testfahrer Lorentz Lemniskate (Abb. 12) ermöglicht, seine Kreise ins Unendliche zu spinnen. Stuntman Lorentz,

³ Magnet fischertechnik (142504) oder ein chinesisches Äquivalent (ZYE1-P20/15).

dessen Name in diesem Zusammenhang natürlich eine Anspielung auf die Kraft ist, auf die er angewiesen ist, hängt immer an einem der Elektromagneten der Transporträder. Zurück in der Mitte wird er automatisch vom anderen Rad übernommen. Ein Impulsschalter an einem Flip-Flop sorgt für die Umschaltung.



Abb. 13: Modell in Aktion

Da ich anfangs viel mit der Drehzahl und der Stellung des Impulsschalters für das Flip-Flop experimentieren musste, wurde Lorentz' Wunsch nach einem Sturzhelm frühzeitig entsprochen. Und ich versichere euch, dass bei der Erstellung dieses Modells keine fischertechnik-Figuren zu Schaden kamen.

Dieses Verfahren erforderte ein Flip-Flop, dessen beide Ausgänge direkt abwechselnd einen Elektromagneten ansteuern konnten. Aus diesem Anlass habe ich mir dafür ein Flipflop gebaut⁴, das ich mit Treiberausgängen versehen habe⁵, damit keine zusätzlichen Relais (mit zusätzlichem Verschleiß, Rauschen und Verzögerung) benötigt wurden. Um das Experimentieren mit der Geschwindigkeit zu erleichtern, habe ich für

den verwendeten 12-Volt-Motor⁶ noch eine Motorsteuerung gebaut. Dafür reichte ein einfacher Schaltplan mit PWM-Steuerung.⁷ Beide Module wurden in grauen fischertechnik-Kassetten (35359) untergebracht und über ein 12-Volt-Netzteil gespeist (Abb. 14). Die Frontplatten wurden mit einem Laserdrucker auf Normalpapier gedruckt und anschließend laminiert und geschnitten.



Abb. 14: Flip-Flop und PWM-Steuerung

Da ich die Befestigung (und die Festigkeit der Stecker) der Steckverbindungen lieber direkt auf dem Deckel hatte, war diese Bas-

⁴ Flip-Flop von Donaupeter.

⁵ Flipflop driver: IRF520 Power MOSFET.

⁶ JGA25-370, 400 rpm, 12-Volt-Motor im selbst entworfenen und 3D gedruckten Motorhalter.

⁷ Geschwindigkeitsregelung mit NE555 und TIP120.

telei auch eine Untersuchung von Alternativen zu den bekannten Bundhülsen⁸, die normalerweise direkt auf die Leiterplatte gelötet werden. Neben Kupfer-M3-Abstandshaltern⁹ erweisen sich hierfür auch Hohlnieten¹⁰ als sehr geeignet. Letztere lassen sich nach dem Erhitzen ganz einfach in ein etwas zu klein gebohrtes Loch in der Plexiglas-

Front oder dem PLA-Kunststoff eines 3D-gedruckten Deckels schieben.

Nach dem Abkühlen sind die Rohre fest verankert. Bei Bedarf können die Rohre vorne mit einem 2,5-mm-Bohrer nachgebohrt werden, damit die fischertechnik-Stecker perfekt passen.

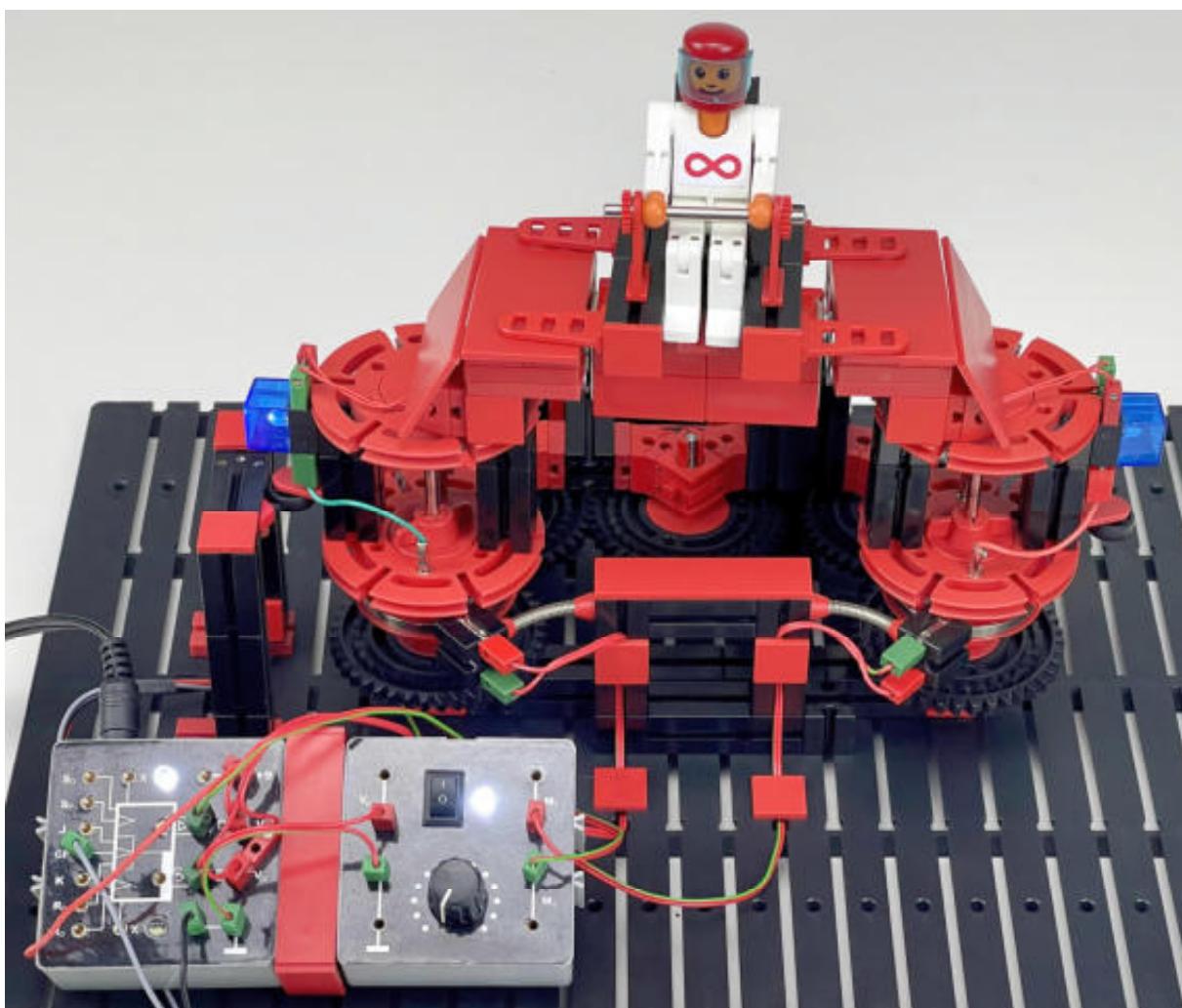


Abb. 15: Gesamtansicht des Modells

⁸ Bezug [Bundhülsen](#).

⁹ Alternative zum Anschließen der Stecker: Suche nach ,Hex Nut Spacing Screw Brass‘.

¹⁰ Kupfer-Verbindungsrohre, Suche nach ,Copper Hollow Tubular Rivets 3mm‘.

Fazit

Vielleicht können die Leser der ft:pedia mit den fischertechnik-Kugelbahnen der reinen mathematischen Lemniskate noch näher kommen oder durch Bewegung eine Kugel hindurchsteuern? Oder ist es vielleicht mit einer zusätzlichen Stangenkonstruktion in der ersten Lösung möglich, das bewegliche Element dem Kurvenverlauf folgen zu lassen?

Aber auch wenn ihr beim nächsten Anblick dieser wundersamen Gestalt (siehe z. B. das [Video](#) mit *Lorentz Lemniskate* in der Hauptrolle) nur lächelt, hat Lorentz Lemniskate sein Ziel bereits erreicht. Oder ihr testet einmal das auf diesem Mechanismus aufbauende „[Demolition Derby](#)“-Fahrgeschäft in einem Freizeitpark. Die Lemniskatenform wird die Menschen wohl noch bis ∞ faszinieren ...

Quellen

- [1] Wikipedia: [Lemniskate von Booth](#).
- [2] Wikipedia: [Lemniskate of Gerono](#).
- [3] Robert Ferréol: [Booth's Curve, Oval, Lemniscate](#). 2017.
- [4] Benjamin Klingebiel: *Strandbeester*. [ft:pedia 1/2021](#), S. 33–40.
- [5] Clayton L. Vagneur: [Mechanical Movement for Reciprocating Mechanisms](#). US-Patent No. 2.775.899, 01.01.1957.

Modell

Großprojekt Seilbahn (Teil 2): Erste Elemente

Tilo Rust

Diese Serie begleitet das Großprojekt „Kuppelbare Einseilumlaufbahn/Doppelmayr (10-MGD)“ im Fördertechnik-Museum Sinsheim von Anfang bis zur Fertigstellung und Ausstellung auf der BUGA 2023 in Mannheim. In dieser Folge zeigen wir die ersten Ergebnisse beim Bau.

Zu den Kernelementen einer Seilbahn gehören die Gondeln mit Klemme, die Rollenbatterien auf den Stützen und die Seilscheiben – das sind die großen Räder, um welche das Seil in den Stationen geführt ist. Diese Elemente sind nicht nur in der Form nachzubauen, sondern in der Dimension unseres 1:10-Modells besonders stabil zu gestalten. Hinzu kommt, dass sie präzise gefertigt werden müssen, damit der Betrieb ohne ständige Eingriffe funktioniert und die Zuverlässigkeit über Monate des täglichen Betriebes gewährleistet ist. Schließlich soll die

Seilbahn auf der BUGA23 jeden Tag von morgens bis abends durchgehend fahren – monatelang.

Die Station in Abb. 1 wird von Doppelmayr/Garaventa gebaut und ist eine kuppelbare Einseilumlaufbahn (10-MGD), die Station eine D-Line mit Medienfassade. Diese Seilbahn werden wir im Maßstab 1:10 aus fischertechnik im Fördertechnik-Museum Sinsheim bauen und hoffentlich dort ausstellen.



Abb. 1: So könnte die Station auf der BUGA2023 in Mannheim „Spinelli“ aussehen.
Quelle: Doppelmayr Seilbahnen GmbH [5], mit freundlicher Genehmigung.

Gondel

Schmuckstück einer jeden Seilbahn sind die Gondeln. Auf der BUGA23 in Mannheim werden diese von der Firma CWA [3] stammen und die Bezeichnung OMEGA V-10 tragen. Dankenswerterweise haben wir von CWA die exakten Pläne der Gondeln erhalten, so dass wir bis auf wenige Millimeter an den Maßstab 1:10 herankommen konnten. Unsere Modelle besitzen aktuell noch weder Boden, Seiten noch Türen – diese müssen erst aus Kunststoff gefertigt werden, um später dem edlen Look & Feel zu entsprechen.

Am Seil gehalten werden die Gondeln von den Klemmen. Wir haben diese nach dem Vorbild der Totpunkt-klemmen von Doppelmayr gebaut. Dabei stoßen wir an die Machbarkeitsgrenze mit fischertechnik. Sicher wäre es möglich, die Klemmen auch pur, also aus Originalteilen herzustellen, aber

nicht in diesem Maßstab, dieser Zuverlässigkeit und mit diesen Eigenschaften für einen Dauerbetrieb. Aus diesem Grund mussten wir hier Teile im 3D-Druck herstellen. Im endgültigen Modell werden sogar Komponenten aus Alu gefräst. Diese Klemmen werden mit dem rückwärtigen Hebel bei Einfahrt in der Station geöffnet, das Seil nach unten weggeführt und die Gondel abgebremst, sodass Passagiere sicher und komfortabel aus- und einsteigen könnten. Auch die Türen werden mit einem weiteren Hebel und einem Bowdenzug geöffnet und geschlossen. Hierbei setzt uns die Miniaturisierung mit fischertechnik abermals Grenzen. Auch diese Mechanik kann zwar mit Originalteilen gebaut werden, aber nicht so klein, kompakt und leicht wie mit 3D-Druckteilen. Schließlich steht nicht viel Raum für die Mechanik zur Verfügung und das Gewicht beeinflusst den Bau von Seilscheibe und Rollenbatterien.

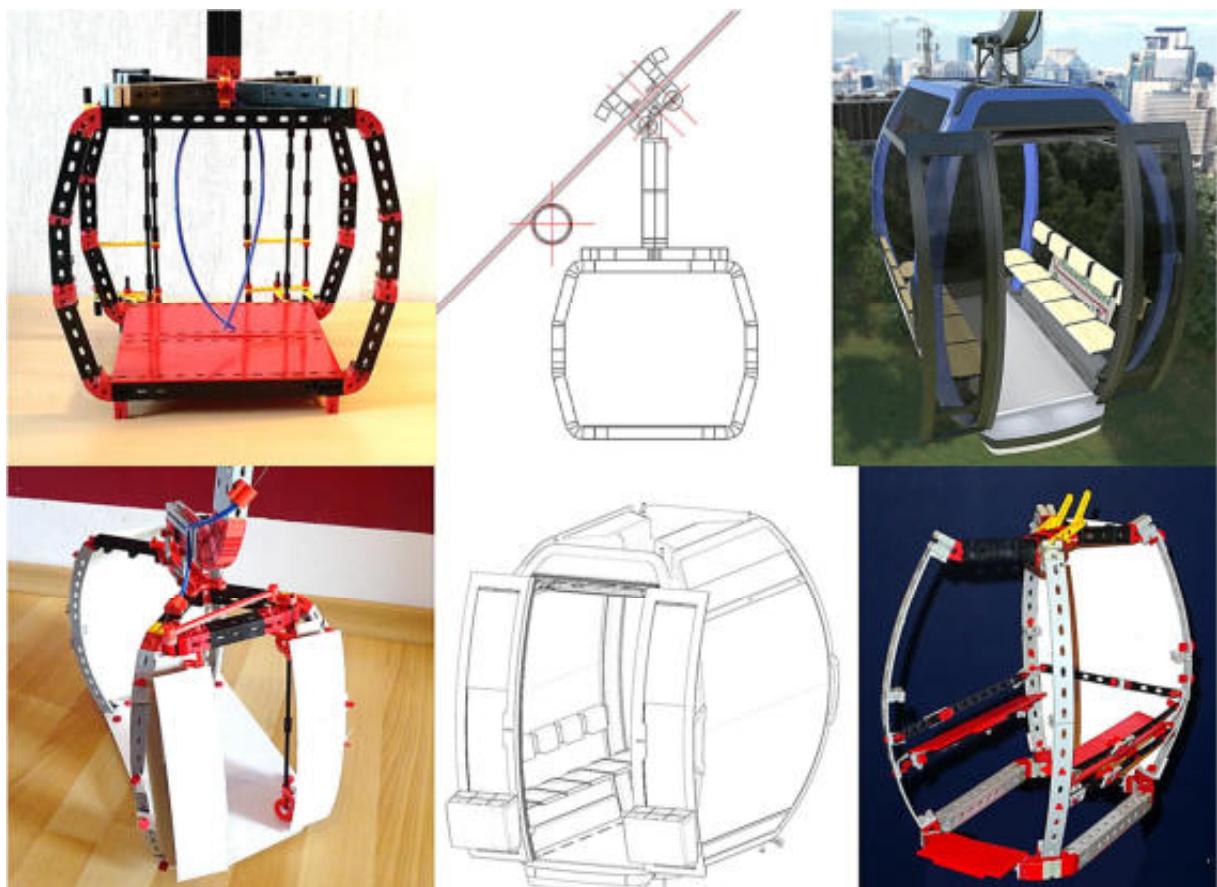


Abb. 2: Gondelvergleich

Die Bauanleitung und die ft-Designer-Datei stehen bei uns zum Download bereit. Wir suchen außerdem Helfer, die nach diesen weiteren 25 Gondeln fertigen können (Bauteile werden natürlich gestellt). Vom ersten Entwurf bis zur finalen Version zeichnet sich Michael Sengstschmid aus.

Abb. 2 zeigt die Entwicklung der Gondeln: Die erste Version (v1 links oben) war schon nicht schlecht und hatte bereits funktionierende Türen. Allerdings stimmten die Maße nicht mit dem Original von CWA überein. Dennoch konnten wir damit Bauplanungen durchführen (oben Mitte) und schließlich eine leichtere Version v2 (unten rechts) erstellen. Doch erst nachdem wir Originalbilder und Konstruktionszeichnungen erhalten (rechts oben und unten Mitte; Quelle: CWA, mit freundlicher Genehmigung) konnten wir die Version v3_1 (links unten) bauen. Sie weicht nur wenige Millimeter vom Maßstab 1:10 ab.

Seilscheibe

Das absolute Kernstück einer Seilbahn ist unscheinbar. *Seilscheibe* nennt der Fachmann das „große Rad“, um welches sowohl bei kuppelbaren Einseilumlaufbahnen (wie unserer 10-MGD) als auch bei Schleppliften, aber auch bei großen Pendelbahnen das Zugseil geschlungen ist. Sie muss zum einen das Seil spannen, zum anderen antreiben. Die Seilscheibe ist absolut sicherheitskritisch und wird bei den echten Anlagen mehrfach überdimensioniert, perfekt gefertigt und ständig geprüft. Auch in unserem 1:10 Modell ist sie kritisch.

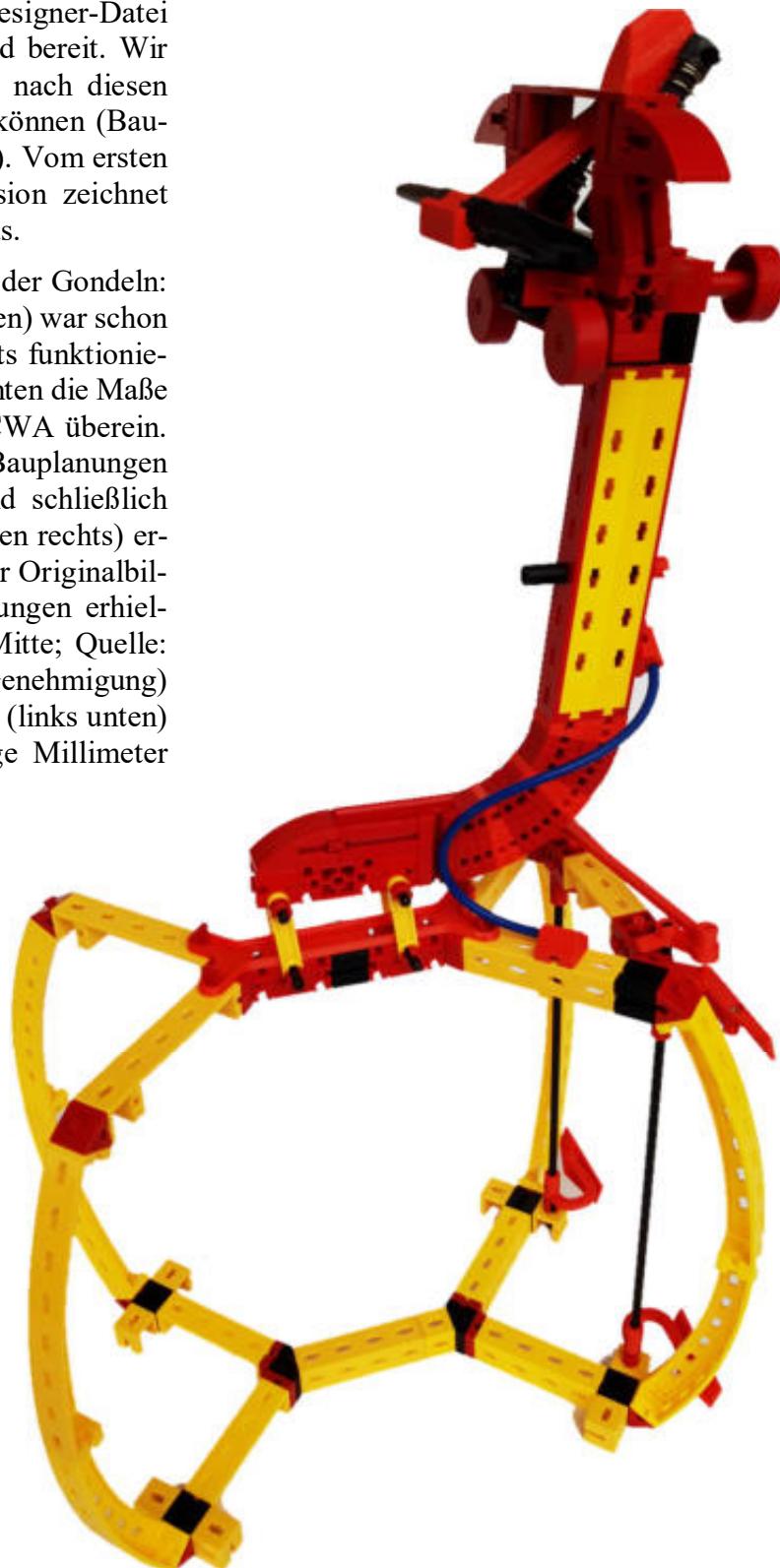
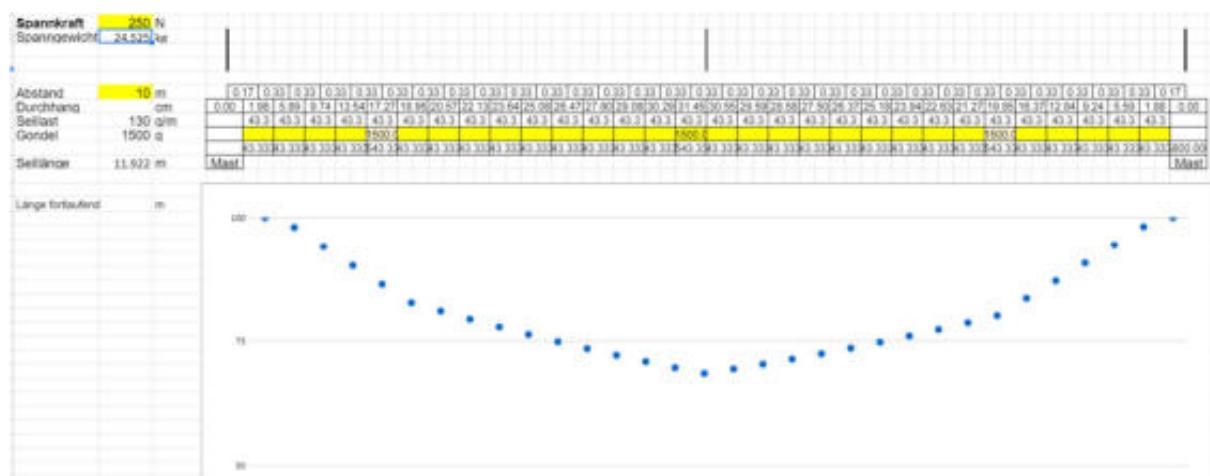


Abb. 3: Die Gondel Modell CWA OMEGA V-10 im Maßstab 1:10 hat später mit Türen und Fenstern aus Kunststoff ein Gewicht von ca. 1,5 kg. Gut erkennbar ist auch die Totpunktklemme oben. Etwa 25 Stück werden auf der Förderlänge von 30 m (72 m Seillänge) im Abstand von ca. 3 m fahren.



*Abb. 4: In dieser Tabelle kann die nötige Spannkraft des Seils abhängig von Seilgewicht, Abschnittslänge, Gondelanzahl und -gewicht, sowie maximal zulässigem Durchhang ermittelt werden
(Michael Sengstsched)*

Auf ihr lastet nicht nur eine statische Spannkraft von 500 N (entsprechend einem Spanngewicht von 50 kg), sie ist auch jeden Tag stundenlang in Betrieb und muss trotz der Stöße, die auf sie wirken, bis auf wenige Millimeter genau in der Spur laufen, ohne sich mit der Zeit zu verziehen. Ein Bruch wäre fatal. Nicht nur, weil das Modell Schaden nimmt, sondern auch, weil das Seil fallen und die Gondeln auf unsere Besucher stürzen könnten.

Aus diesem Grund haben wir die Seilscheibe so konstruiert, dass sich keine der Zapfenverbindungen verschieben kann oder gar Teile herausfallen könnten. Um die Stabilität und Fertigungsgenauigkeit nachzuweisen, wurden Test- und Messstände konstruiert und die Abweichungen dokumentiert. Ein Video darüber wird kurz nach Erscheinen dieser ft:pedia auf unserem YouTube-Kanal [8] zu sehen sein. Die Fertigungsgenauigkeit liegt bei ± 2 mm Höhenschlag und $\pm 2,5$ mm Seitenschlag (beim Fahrrad „Achter“ genannt). Selbst bei Vollast verzieht sich die Seilscheibe in diesen Tests auch nach zehn Tagen nicht mehr als ± 1 mm. Mit Hilfe einer Neoprenauflage erreichen wir eine Traktion von mindestens 100 N am Seil.

Alle diese Anforderungen zeigen: Wir beschreiten hier mit fischertechnik Dimensionen, die bei Erfolg zeigen werden, was dieses „Spielzeug“ alles kann.

Eine Herausforderung wartet allerdings noch auf uns: Wir haben die Seilscheibe in unserem Montage- und Teststand senkrecht stehen. Im Modell muss sie aber waagerecht unter einem Schlitten hängen und in der Antriebsstation über die Achse vom Motor gedreht werden. Sie wird also einseitig hängend gelagert. Hierzu wird die aktuelle Nabe mit 7-mm-Achse noch durch eine größere ersetzt. In den weiteren Ausgaben der ft:pedia werden wir darüber noch berichten.

Die Seilscheibe wurde von Michael Sengstschmid konstruiert, von Tilo Rust erweitert und von Ralf Geerken verriegelt. Sie besteht aus ca. 1.000 Einzelteilen und ist so kompliziert zu bauen, dass wir davon abgesehen haben, eine komplettene Bauanleitung zu erstellen. Die ft-Designer-Datei ist dennoch frei verfügbar. Das Video über die Seilscheibe auf unserem YouTube-Kanal zeigt nicht nur die Funktion und Konstruktion, sondern auch den genauen Test.



Abb. 5: Die fertige Seilscheibe auf dem Montagegestell

Die Seilscheibe wird später waagerecht in den Stationen laufen und das 6-mm-Stahlseil (in Abb. 5 bereits eingehängt) spannen und antreiben. Der innere Kranz ist mit Zahnstangen besetzt. Dieser dient in der Talstation später für den Notantrieb. Sein Höhenschlag darf daher nicht mehr als 1 mm betragen. Zur Stabilisierung trägt er nicht bei – ganz im Gegenteil, mit Hilfe von in den Profilen eingeschobenen Stangen stützt er sich am äußeren Kranz ab. Am unteren Rand ist die Messeinrichtung eingehängt, mit der wir die Spurgenauigkeit und Verformungen unter Belastung messen und dokumentieren können.

Der Aufbau der Scheibe ist also ganz schön kompliziert – oder raffiniert, wie man es betrachtet. Ralf Geerken hat die Teile der Seilscheibe so gegeneinander verriegelt, dass kein Bauteil herausfallen kann. Insgesamt sind die erforderlichen über 1.000 Teile ohne Modifikation oder Klebstoff verbaut (mit Ausnahme des Zahnkranzes; hier wurden acht Teile gekürzt).



Abb. 6: Die Seilscheibe im Bau

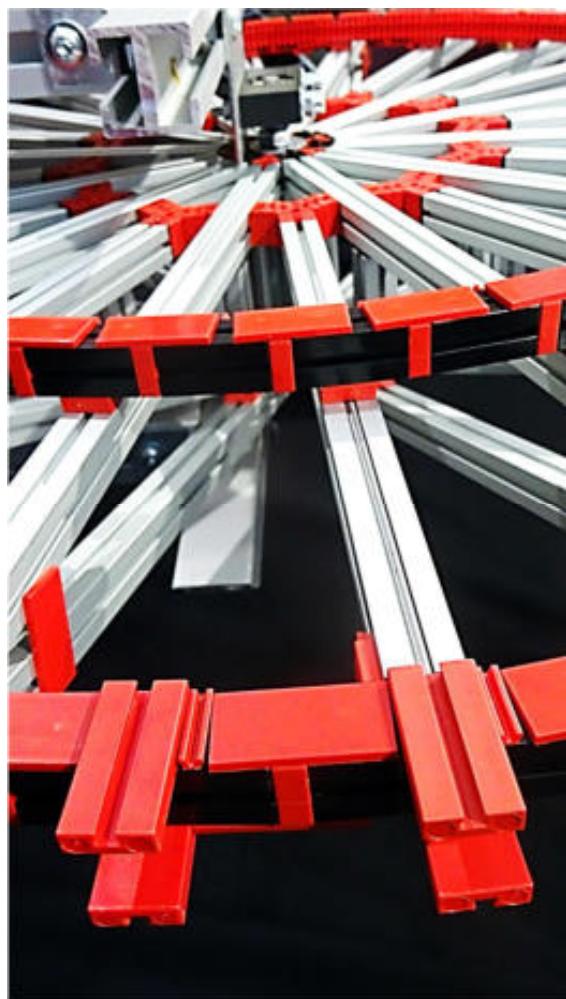


Abb. 7: Detailblick auf die Seilscheibe

Die Messeinheit zeigt Abb. 8: Rechts ist im Kardangelenk der Fixpunkt. Über die lange Stange, die genau in der Spur der Seilscheibe läuft und an der Kontaktstelle (nicht sichtbar) auf 6,1 mm erweitert wurde – das entspricht dem Seildurchmesser – wird am

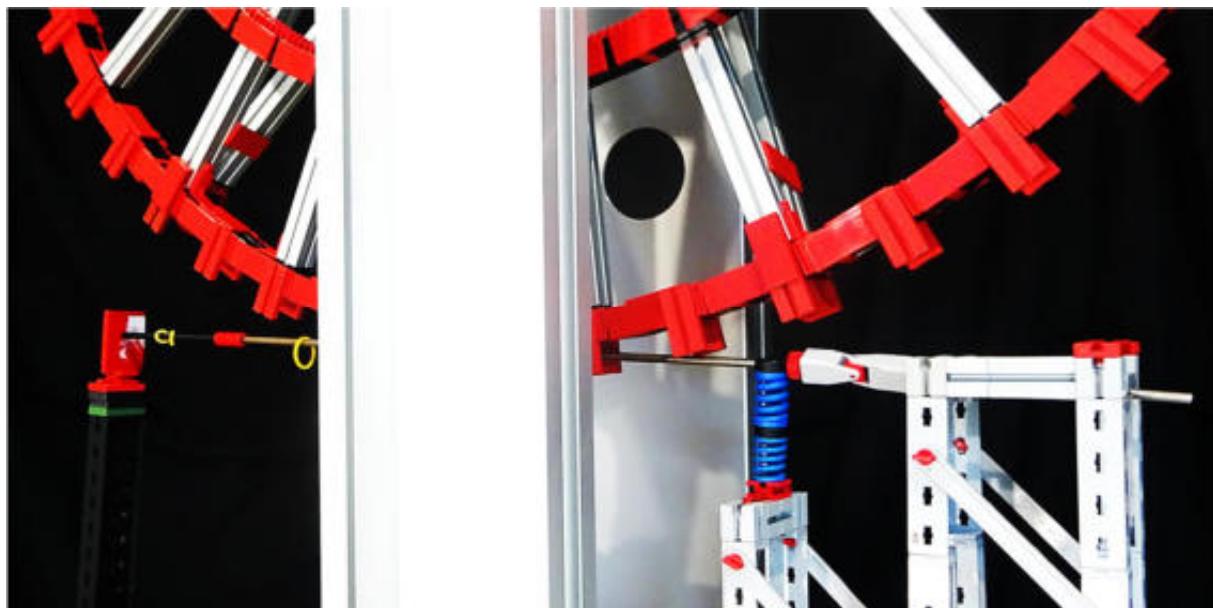


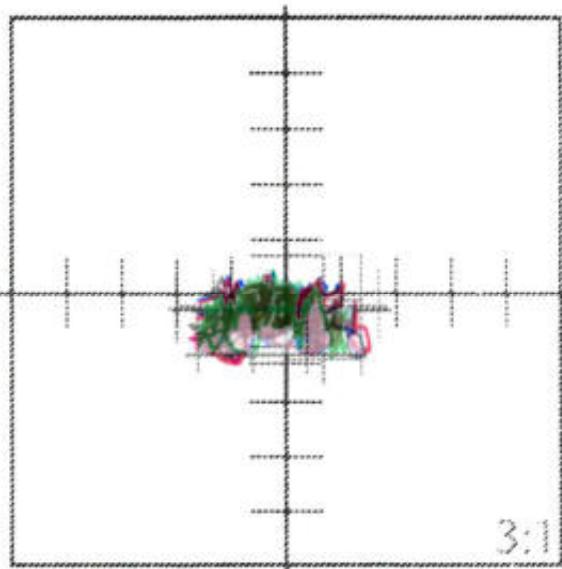
Abb. 8: Vermessung der Seilscheibe

anderen Ende mit einem Stift auf das Messprotokoll geschrieben. So können wir Höhenschlag und Spurlage ermitteln und dokumentieren.

Das Messprotokoll (Abb. 9) enthält die (nachträglich übereinandergelegten) Aufzeichnungen von vier Prüfungen:

1. In der Grundmessung wurde die Seilscheibe nach der Endmontage auf Fertigungsgenauigkeit geprüft.

2. Wiederholung der Messung, nachdem die Scheibe zehn Tage lang statisch mit der Spannlast von 500 N ohne Bewegung belastet wurde.
3. Danach wurde die Seilscheibe 100-mal um 90° und wieder zurück gedreht – das simuliert eine Belastung im Betrieb.
4. Anschließend wurden über zwei Stunden Vibrationen in unterschiedlichen Frequenzen und Stärken zur Last aufgebracht.



- 1. Grundmessung**
- 2. Statisch 50kg**
- 3. Dynamisch 90°**
- 4. Vibration**

Abb. 9: Messprotokoll

Die ausführliche Messung wurde per Video für unseren YouTube Kanal [8] dokumentiert.

Das Ergebnis: Die Abweichung bewegt sich bei $\pm 1,5$ mm in der Spurlage (horizontale Aufzeichnung) und ± 1 mm im Höhenschlag. Trotz aller Strapazen erkennen wir keine Verformung gegenüber der Grundmessung. Wir können also beweisen, dass man mit fischertechnik nicht nur stabil, sondern auch genau und haltbar bauen kann!

Rollenbatterie

Auf der Spitze einer Stütze sitzt bei den Seilbahnen eine Traverse, „Joch“ oder „Hauptjoch“ genannt, die seitlich in Spurweite die Rollenbatterien trägt. Üblich sind hier vier bis 16 Rollen, teilweise sogar mehr, in einer Reihe, welche das Seil und die Gondeln tragen und die Seilbahnstrecke der Geländeform anpassen. Unterschieden wird zwischen Traglast-, Niederhalte- (welche das Seil herunterdrücken, z. B. vor der Talstation) und Wechsellastrollenbatterien.

Unser Modell wird vier Stützen haben, ca. alle 10 m eine. Die Stützen 1 (vor der Tal-/Antriebsstation „Spinelli“) und 4 (vor der Berg-/Umkehrstation „Luisenpark“) werden Niederhalter sein. Die Stützen 2 und 3 werden in ca. 4,60 m Höhe heraufragen, da die Besucher unter unseren Gondeln durchlaufen dürfen, aber es nicht zu einer Berührung (z. B. Kinder auf Schultern der Väter) kommen darf.

Eine Stütze trägt jeweils die Hälfte aller Lasten der beiden Abschnitte vor und hinter ihr. Sie besteht aus dem Seilgewicht und dem Gewicht der maximalen Gondelzahl in dieser Strecke. Dazu übt die Seilspannung Kraft aus. Die Last, welche von unseren Rollenbatterien getragen werden muss, haben wir mit ca. 20 kg pro Fahrspur abgeschätzt. Hinzu kommen dynamische Lasten, z. B. vom Schwingen des Seils oder bei Überfahrt der Gondelklemmen an den Rollen. Mit acht Rollen pro Seite bricht sich das auf max. 3 kg Last pro Rolle herunter, entsprechend 1,5 kg pro Lager. Nach unserer



Abb. 10: Die Rollenbatterie

Meinung ist das mit fischertechnik machbar. Dennoch sind die Rollen mit je zwei Kugellagern und Alu-Profilen gelagert. Die Hauptachse ist eine 8 mm Stahlstange.

Befestigt wird die Rollenbatterie über eine Stahlplatte (in der Form wie eine fischertechnik-Bauplatte) durch die eine Gewindestange führt. Direkt daran ist ein Stahlseil befestigt, das durch das Kunststoffrohr in das Fundament auf der Europalette die Kraft einleitet. Mehr als ein Detail hierbei: Die Stütze muss später um einiges gekippt werden, damit der resultierende Kraftvektor senkrecht in die Stütze wirkt. Zudem muss alles leicht demontierbar sein und auf der Palette zum Transport gesichert werden können.

Nach einigen Prototypen und Lastversuchen haben wir uns für diese Version entschieden. Sie besteht aus ca. 900 Bauteilen. Seilabweiser und Seilfang sind bereits angebracht. Später wird noch eine elektronische Seilüberwachung hinzukommen und über dem Oberjoch ein Träger die Signalleitung führen.

Wer diese Rollenbatterie genauer anschauen möchte oder uns helfen will, die weiteren drei Exemplare zu bauen (alle Bauteile werden gestellt), der kann sowohl die Bauanleitung als auch die ft-Designer-Datei herunterladen und ein Video zur Montage anschauen. (Entwurf, Berechnung und Bau: Tilo Rust.)

Die fertige 8-fach-Tragbatterie zeigt Abb. 10 mit Joch auf einer Stütze montiert. Sie muss auf jeder Seite das Seil und die Gondeln tragen und im Dauerbetrieb fehlerfrei die Spur über 10 m Spannweite halten. Dieses Exemplar wird in 4,60 m Höhe dafür sorgen, dass die Besucher unter der Gondel durchlaufen können. Es fehlen noch die Wartungspodeste – das ist aber Dekoration und wird erst später hinzugefügt, wenn die Anlage funktioniert.

Details zeigt Abb. 11: Am jeweils ersten und letzten Rollenpaar sind Seilabweiser (grüner Pfeil: Winkelstein 60° mit abgesägtem Zapfen) und Seilfang (gelber Pfeil, M4-Schraube mit Unterlegscheibe) angebracht.

Das ist nicht nur eine Nachbildung des Originals, sondern es sind tatsächlich notwendige Sicherheitseinrichtungen in unserem Modell. Der Seilabweiser verhindert das Verklemmen des 6mm Stahlseils hinter den Rollen, während der Seilfänger einen Absturz des Seils nach einem etwaigen Entgleisen auffangen soll. Die Rollen selbst bestehen aus der Felge 43 mit aufgezogenen O- und X-Profil-Dichtringen als Lauffläche. Die Achsen sind Edelstahlschrauben, welche in der Schneckenmutter ([37925](#)) mit verpressten 4-mm-Kugellagern gelagert sind.

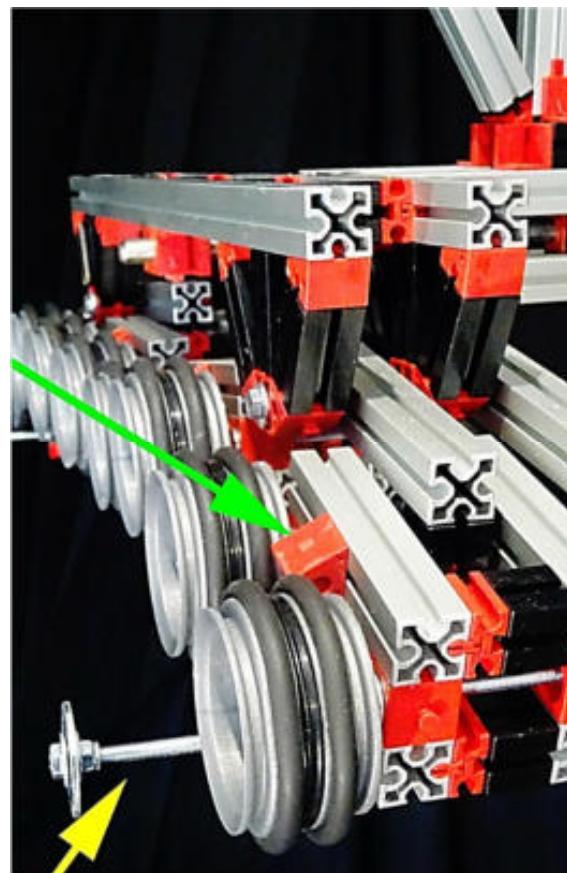


Abb. 11: Details der Rollenbatterie

Form follows Function

Im nächsten Schritt werden die Stationen um die Seilscheibe herum gebaut. Wir werden mit der Berg-/Umkehrstation „Luisenpark“ beginnen, da diese weder einen Spannschlitten noch einen Antrieb besitzt und somit einfacher ist (hierüber werden wir in der nächsten ft:pedia berichten). Auf den dort gewonnenen Erkenntnissen können dann die Tal-/Antriebsstation „Spinelli“ und nachfolgend die Garagierung gebaut werden. Herausforderungen hierbei sind vor allem die Kraftaufnahme (Seilspannkraft von 50 kg $\hat{=}$ 500 N) sowie die Anforderung, die Stationen in Module aufzuteilen und auf Euro-Paletten zu befestigen.

Quellen

- [1] ftc-Bilderpool: [Seilbahnen](#).
- [2] Alle Bauanleitungen und ft-Designer-Dateien zum Download auf [Google Drive](#).
- [3] Firma CWA (Gondelhersteller): [Homepage](#).
- [4] ftc-Forum: [Community-Projekt: Seilbahn](#).
- [5] Doppelmayr/Garaventa: [Homepage](#).
- [6] ftc-Bilderpool: *Kuppelbare Doppelmayr Seilbahn*. Projektwebsite im [Bilderpool](#).
- [7] Öffentlicher Media Pool des Seilbahnprojekts auf [Google Drive](#).
- [8] [YouTube-Kanal](#) des Projekts
- [9] Kontakt zur Projektgruppe „Seilbahn“: Projektleiter Tilo Rust, Schifferstadt, ft.seilbahn@gmail.com

Elektronik

Es blinkt

Peter Krijnen

Der Auslöser für diesen Beitrag war die Frage eines Bekannten von mir, ob ich noch ein paar Winkelsteine 30 habe und ob er sie ausleihen könne – für ein blinkendes Modell.

Ich antwortete ihm, dass ich nur noch 129 dunkelrote Winkelsteine 30 übrig habe. Die anderen 600 sind verbaut in vier Kranmodellen. Nach ein paar Wochen bekam ich meine Winkelsteine zurück. Später schickte er mir einige Bilder von dem, was er gemacht hatte. Eine Kerze. Jetzt fragt ihr euch natürlich, was das mit Blinken zu tun hat. Um das Flackern der Flamme zu simulieren, verwendete er eine Blinkelektronik von fischertechnik.

Ich selbst besitze beide Blinker von fischertechnik: die Blinkeinheit [36756](#) (9 V, 0,6 A, rot) und die Blinkelektronik [35604](#) (9 V, 0,8 A, schwarz).

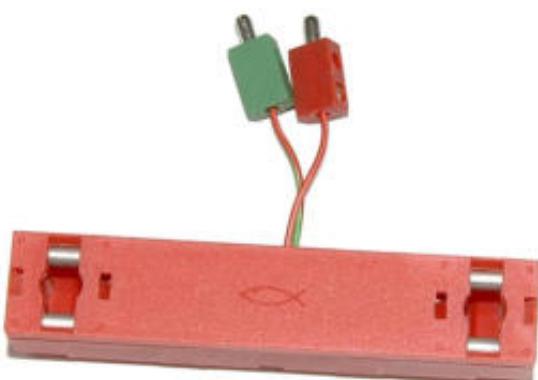


Abb. 1: Blinkeinheit [36756](#) (9 V, 0,6 A, rot)

Mit der [36756](#) (Abb. 1) ist sofort klar, wie sie angeschlossen werden soll: Zwei Leitungen mit Steckern für die Stromversorgung und zwei Positionen für zwei Lampen. An beiden Enden können dann weitere Lampen angeschlossen werden.

Bei der [35604](#) (Abb. 2) ist dies nicht klar, da sie nur zwei Leitungen hat, aber keinen Platz und keine Anschlüsse für Lampen.



Abb. 2: Blinkelektronik [35604](#) (9 V, 0,8 A)

Sieht man sich die Bedienungsanleitung der [35152](#) an, wird schnell klar, wie die Blinkelektronik angeschlossen werden soll. Das gelbe Kabel des Blinkers muss an „Plus“ der Stromversorgung angeschlossen werden, das blaue Kabel geht zur Lampe. Die Lampe wird dann wieder an „Minus“ der Stromversorgung angeschlossen.

Man wird verstehen, dass ich wirklich wissen wollte, wie das funktioniert. Genau wie bei den Silberlingen habe ich auch diese beiden Blinker geöffnet und mir angesehen, was drin ist.

Blinkelektronik 35604 (schwarz)

Eine kleine ($27 \cdot 12$ mm) doppelseitige Leiterplatte (Abb. 3) enthält einen NE555, drei Dioden, einen Elko, einen Kondensator, drei Widerstände und einen FET – alles in SMD-Technologie. Mit dem Ohmmeter habe ich alle Verbindungen zwischen den

Bauteilen gemessen und konnte daher das Layout bald auf Papier zeichnen. Auf der Grundlage dieses Layouts wurde auch das Schaltbild schnell gezeichnet (Abb. 4).



Abb. 3: Platine der [35604](#)

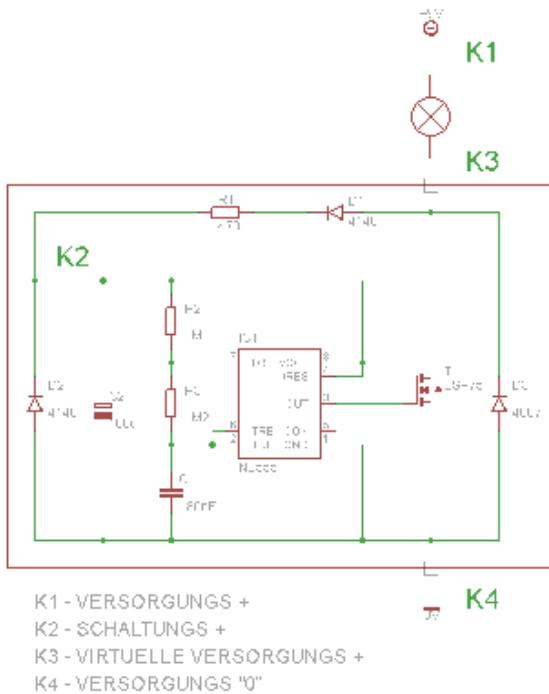


Abb. 4: Schaltbild der [35604](#)

Der NE555 wird über die Dioden D1, D2 und den Widerstand R1 mit Strom versorgt. Elko C1 fungiert hier als kleine Batterie und stabilisiert somit die Stromversorgung. Die Widerstände R2, R3 und der Kondensator C2 sind die frequenzbestimmenden Komponenten (2,18 Hz). Der Ausgang Pin 3 des NE555 steuert dann FET T1 an. Wenn dieser N-FET vom Typ BSP75 leitet, schließt sein niedriger RDS(ON)-Widerstand von

500 mΩ den Stromkreis kurz und die Lampe leuchtet auf. Der BSP75 ist eigentlich ein Lowside Power Switch, ein Schalter also, und eignet sich auch zum Schalten jeder Form von ohmscher, kapazitiver oder induktiver Last. Der BSP75 darf bis zu 0,7 A belastet werden. Wenn der Strom jedoch 1 A überschreitet, wird der BSP75 sofort durch seinen eingebauten Schutz ausgeschaltet.

Abb. 5 ist ein Screenshot meines Oszilloskops. Die Kanäle 1 bis 4 entsprechen den im Schaltbild angegebenen Messpunkten K1 bis K4.

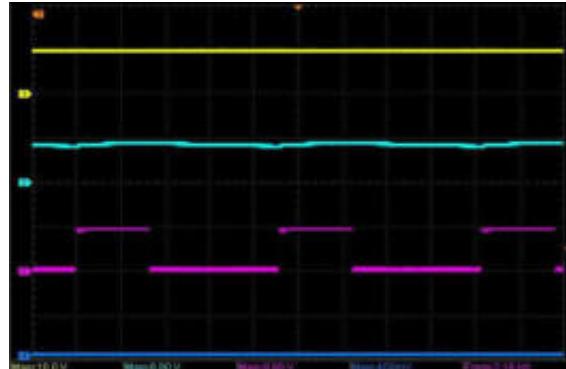


Abb. 5: Oszilloskop-Aufnahme

Kanal 3 (violett) ist die Darstellung des Ausgangssignals. Es ist auch ersichtlich, dass das Puls/Pause-Verhältnis ungefähr 1:2 beträgt. Da die Stromversorgung für den Stromkreis durch Schalten unterbrochen wird, muss diese durch C1 stabilisiert werden. Auf Kanal 2 (blaugrün) ist deutlich zu sehen, dass dies nicht ganz erfolgreich ist, aber es reicht aus, damit die Schaltung funktioniert.

Laut Bedienungsanleitung kann der Blinker auch verkehrt herum angeschlossen werden. Das Licht bleibt dann kontinuierlich an. In Abb. 6 sieht man die Pegel an den vier Messpunkten. Mir ist jedoch nicht ganz klar, was D3 in der Schaltung tun soll. Sie ist nicht erforderlich, um Spannungsspitzen zu unterdrücken, die durch geschaltete Lasten verursacht werden. Das ist schon im BSP75 eingebaut.

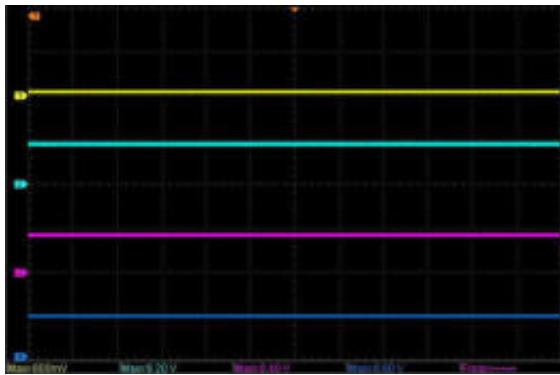


Abb. 6: Pegelaufnahme der Messpunkte

Die Spannungsdifferenz zwischen Kanal 3 und 4 beträgt in dieser Situation 1,2 V. Dies ist der Spannungsabfall an der Diode. Deshalb denke ich, dass D3 dafür in der Schaltung enthalten ist, um sie zu überbrücken. D2 macht das Gleiche, um C1 zu überbrücken. Trotzdem hat der GND-Anschluss (PIN 1) des NE555 eine um 0,4 V höhere Spannung als VCC (PIN 8).

Bei elektronischen Schaltkreisen gehe ich immer davon aus, dass der Schaltkreis an den „0“-Anschluss des Netzteils angeschlossen ist (Abb. 4). Das sollte jedoch laut Bedienungsanleitung umgekehrt sein.

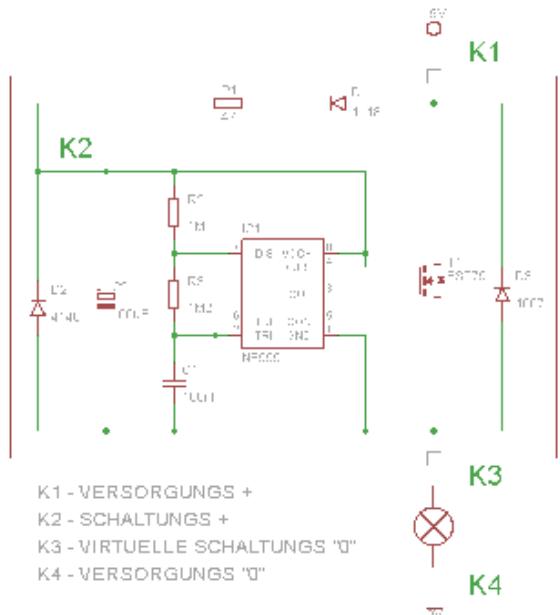


Abb. 7: Die zweite gemessene Schaltung

Deshalb habe ich auch diesen Aufbau gemessen (Abb. 7). Die Ergebnisse dieser Messung sind in Abb. 8 zu sehen. In der

vorherigen Situation schien das „Plus“ der Stromversorgung geschaltet zu werden. Im Aufbau nach Abb. 7 scheint es, als ob die „0“ geschaltet wird.

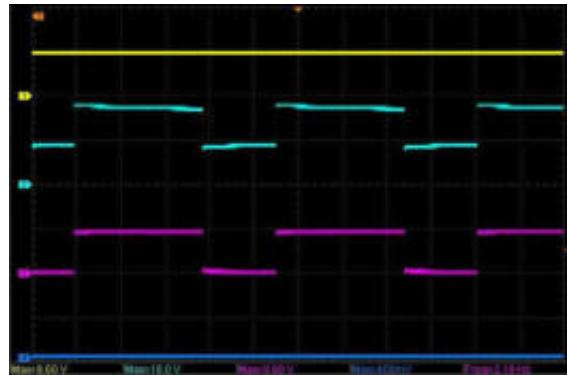


Abb. 8: Ergebnis der Messung von Abb. 7

Dies ist deutlich sichtbar. Am Messpunkt K3 schaltet die Spannung zwischen „0“ und „+“ der Stromversorgung um. Das hat zur Folge, dass die Spannung an Pin 8 des NE555 jetzt 18 V beträgt.

Wenn T1 leitet, liegt Messpunkt K3 an der Versorgungsspannung. Da K3 auch der Nullpunkt der Schaltung ist, befinden sich „Minus“ von C1 und Pin 1 des NE555 jetzt auch auf „Plus“ der Stromversorgung. Da C1 an seinem Minus-Anschluss keine „0“ mehr hat, addiert sich der Pegel über C1 zur Versorgungsspannung an seinem „Minus“-Anschluss: $9 \text{ V} + 9 \text{ V} = 18 \text{ V}$. Für den NE555 spielt es keine große Rolle, ob er nun 0 und 9 V an den Pins 1 und 8 hat oder ob es 9 und 18 V sind. Es bleibt eine Differenz von 9 V. Meiner Meinung nach ist das aber unpraktisch.

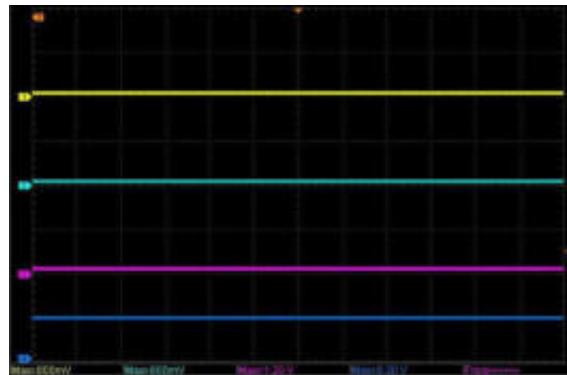


Abb. 9: Pegel bei umgekehrter Stromversorgung

Was auch sichtbar ist, ist, dass das Puls/Pause-Verhältnis jetzt 2:1 beträgt.

Wenn wir auch in diesem Aufbau die Versorgungsspannung umkehren sehen wir die Pegel wie in Abb. 9. Im ersten Aufbau sind die Pegel auf K2 und K3 auf „High“, im zweiten Aufbau sind diese jetzt „Low“.

Blinkelektronik 36756 (rot)

Mit Hilfe des weltberühmten NE555 oder eines Operationsverstärkers, einiger Widerstände und Kondensatoren kann man einen Impulsgenerator bauen. Wir haben gerade gesehen, dass das funktioniert.

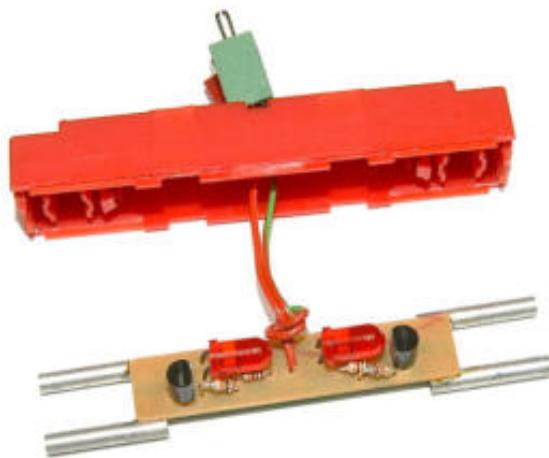


Abb. 10: Das Innere der [36756](#)

Wir werden jetzt sehen, dass dies auch viel einfacher gemacht werden kann. Abb. 10 zeigt, dass man nicht viel braucht: zwei Widerstände, ein Transistor und eine LED. In diesem Fall aber keine gewöhnliche LED, sondern eine blinkende. Die [36756](#) enthält gleich zwei dieser Schaltkreise auf ihrer $48 \cdot 10$ mm großen Platine. In Abb. 11 sieht man, wie es funktioniert.

Mit Hilfe eines Widerstands und der blinkenden LED wird eine Rechteck-Welle aufgebaut. Diese wird über einen Widerstand auf die Basis eines PNP-Transistors vom Typ BC876 gelegt. Die Lampen werden dann zwischen dem Kollektor und der „0“ angeschlossen. Der BC876 ist ein Darlington und darf bis 1 A belastet werden. So können pro Ausgang mehrere Lämpchen

parallelgeschaltet werden. Die Blinkfrequenz für beide LEDs beträgt etwa 2,8 Hz. Aufgrund der Toleranzen aller Komponenten beträgt die Frequenz einer LED 2,78 Hz und der anderen 2,85 Hz. Das Puls/Pause-Verhältnis liegt bei 1:3. Wie aus dem Schaltbild ersichtlich ist es nicht sinnvoll, die Versorgungsspannung umzukehren.

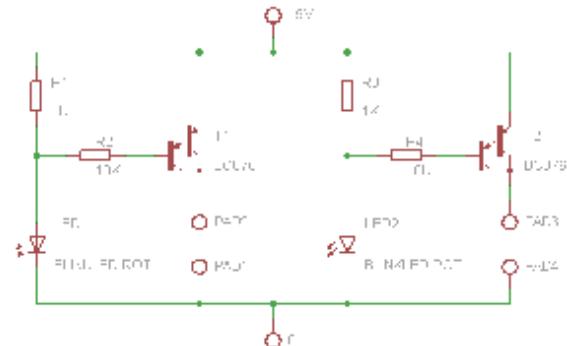


Abb. 11: Schaltbild der [36756](#)



Abb. 12: Oszilloskopbild der [36756](#)

Conrad hat nur eine 3 mm LED im Angebot. Sie hat eine Frequenz von 1,8 Hz und das Puls/Pause-Verhältnis lautet 1:2. Es gibt sie aber in mehreren Farben. Reichelt hat eine rote und eine grüne Blink-LED. Die Frequenz dieser 5-mm-LEDs hängt von der Versorgungsspannung ab und liegt zwischen 1,5 Hz (14 V) und 3 Hz (3,5 V).

Dass es also auch einfach geht, ist toll, hat aber auch einen Nachteil: Die Blinkfrequenz ist abhängig von der benutzten LED.

Elektronik

Echtzeitgenerator

Hans-Christian Funke

Zu der von mir vorgestellten digitalen Uhr [1] wurde ich mehrfach angefragt, ob man nicht statt der recht aufwendigen Justierung einen Sekunden-Taktgeber bauen und anschließen könnte. Das wäre tatsächlich sehr praktisch und auch für andere Anwendung sinnvoll, z. B. könnte man die echte Laufzeit eines Pakets durch die Sortierstation, die eines Autos durch die Waschanlage oder die einer Kugel durch die Kugelbahn ermitteln.

Wer liefert den Takt?

Zunächst musste ich Überlegungen anstellen, wie so ein genauer Taktgeber aussehen müsste und wie garantiert wird, dass dieser tatsächlich zuverlässig einen Sekunden-Takt liefert.

Für zuverlässige Taktgeberquellen wird in der digitalen Elektronik in der Regel auf einen Quarz zurückgegriffen, der äußerst genau eine Schwingung (Takt) mit konstanter Frequenz erzeugt. In der Praxis sieht es aber so aus, dass die Taktfrequenzen der Quarze im Megahertz- (MHz) oder Gigahertzbereich (GHz) gebraucht werden und somit im Wesentlichen auch nur solche zu finden sind. Allerdings gibt es einen Quarz im Elektronikbereich mit 32.768 kHz, der auch noch im Handel erhältlich ist – damit kommen wir unserem Ziel schon viel näher. Diese Art von Quarzen wurde schon in der Anfangszeit der Computertechnik (Commodore 64, Apple II, ATARI, ...) für die Uhrenkarten oder als Zeitgeber verwendet.

Für die weiteren Betrachtungen müssen wir uns einer Formel aus der Physik bedienen, die den Zusammenhang zwischen Frequenz (Hz) und Periodendauer (s) beschreibt:

$$\text{Periodendauer in } s = \frac{1}{\text{Frequenz in Hz}}$$

$$T = \frac{1}{f}$$

Wollen wir also einen Quarz mit einer Frequenz von 32.768 kHz einsetzen, dann erhalten wir folgende Taktdauer:

$$T = \frac{1}{f} = \frac{1}{32768 \text{ Hz}} \approx 30,518 \mu\text{s}$$

Wir brauchen aber einen Sekundentakt für unseren Echtzeitgenerator mit

$$f = \frac{1}{1 \text{ s}} = 1 \text{ Hz}$$

Die Frequenz muss somit kleiner werden, damit die Taktdauer größer wird. Hier hilft uns die digitale Technik mit ihrem binären Zählsystem weiter. Es gibt z. B. Zählerbausteine, die Takte nach dem binären Zahlen- system zählen. Ein 4-bit Binärzähler zählt dezimal von 0 – 15; in Tab. 1 kann man sehen, wie es im binären Zählsystem aussieht.

Daran kann man erkennen, dass sich immer nach zwei Takten eine Änderung am Bit 2 ergibt, nach vier Takten am Bit 3 und nach acht Takten am Bit 4. Das lässt sich beliebig fortsetzen, aber wir können hier schon erkennen, dass ein Binärzähler nichts anderes macht, als von Ausgang zu Ausgang (Bit zu Bit) den Takt zu verdoppeln bzw. die Frequenz zu halbieren.

Und genau dieses Prinzip machen wir uns zu nutze. Wir setzen einen Binärzähler ein,

der unsere Frequenz mehrmals halbiert (siehe Tab. 2).

Takt	Bit 1	Bit 2	Bit 3	Bit 4
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	0	1
11	1	1	0	1
12	0	0	1	1
13	1	0	1	1
14	0	1	1	1
15	1	1	1	1

Tab. 1: 4-bit Binärzähler

Frequenz (Hz)	Taktdauer (s)	Teilung (n)
32.768	0,000030517	0
16.384	0,000061035	1
8.192	0,000122070	2
4.096	0,000244140	3
2.048	0,000488281	4
1.024	0,000976562	5
512	0,001953125	6
256	0,00390625	7
128	0,0078125	8
64	0,015625	9
32	0,03125	10
16	0,0625	11
8	0,125	12

Frequenz (Hz)	Taktdauer (s)	Teilung (n)
4	0,25	13
2	0,5	14

Tab. 2: Teilungssequenz

Wir müssen die Frequenz von 32.768 kHz also 15-mal durch 2 teilen, damit wir eine Frequenz von 1 Hz erhalten.

Zusatzmodul 70051

Soweit die Theorie. In der Praxis ist es so, dass auch häufig ein Halbsekunden-Takt benötigt wird. Aus diesem Grund habe ich eine Schaltung entworfen, die einen 0,5-Sekunden-Takt liefert. Das Ergebnis ist das Zusatzmodul 70051, das mit drei Kabeln ausgestattet ist:



Abb. 1: Zusatzmodul 70051

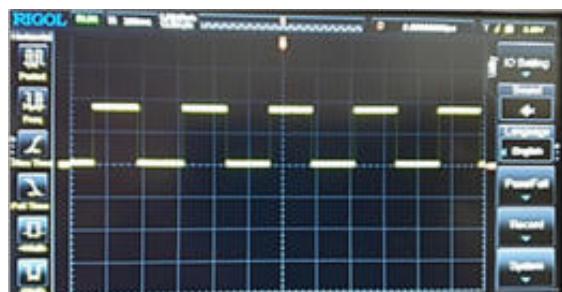


Abb. 2: Ausgangssignal als Rechteck-Takt

Die rote Leitung mit rotem Stecker wird mit „+“ (V_{dd} 3 V bis 15 V) verbunden, entsprechend die grüne Leitung mit grünem Stecker mit „-“ und kann so an jedem Elektronikmodul über „+“ und „-“ angesteckt werden. Das Zusatzmodul ist auch gegen Verpolung empfindlich, so dass auch hier ein

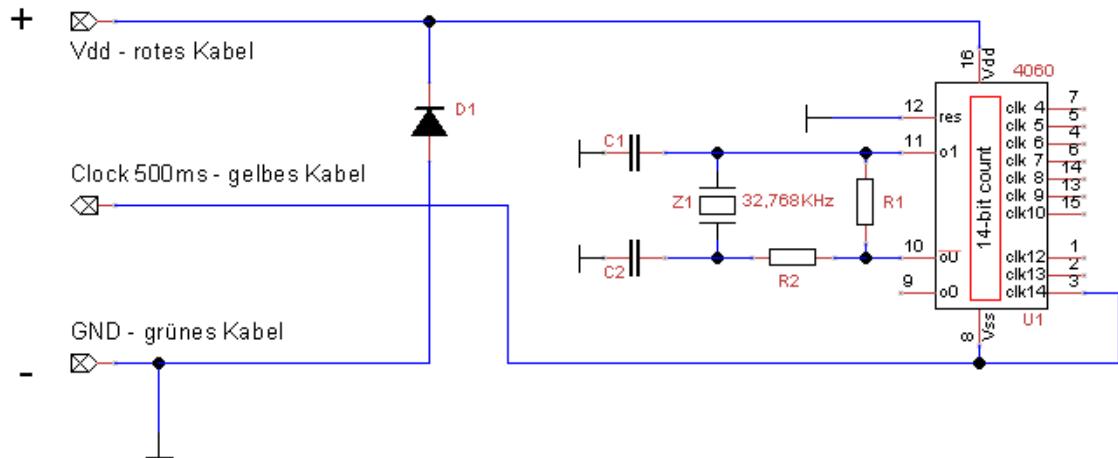


Abb. 3: Schaltung des Echtzeitgenerators

Verpolungsschutz integriert ist, der bis 2 A standhält. Die gelbe Leitung mit dem gelben Stecker liefert das Echtzeitsignal als Rechteck-Takt (Abb. 2).

Schaltung

Der Schaltungsaufbau ist sehr klein, um ein leicht handhabbares Zusatzmodul zu erhalten (Abb. 3).

Der verwendete CMOS-Baustein gibt nur 14 Teilungen her, aber damit wird gerade die Taktdauer von 0,5 Sekunden erreicht. Der Ausgang des Echtzeitgenerators (gelbe Leitung) verfügt über keinerlei Endstufe und kann maximal einen Eingang ansteuern, oder man schließt ihn zur Verstärkung

an ein Gatter- (AND-NAND, OR-NOR oder EXOR-EXNOR) oder ein Flip-Flop-Modul (D-FF oder JK-FF) an. Die Ausgänge der Gatter- und FF-Elektronikmodule (70010-14 und 70006-7) sind wesentlich höher belastbar und man erhält damit auch eine optische Anzeige, ob das Taktsignal anliegt bzw. funktioniert. Zur Verwendung des 0,5-Sekunden-Taktes ist ein Gatter zu verwenden und für einen 1-Sekunden-Takt ein Flip-Flop.

Nach der Inbetriebnahme des Echtzeitgenerators kann es passieren, dass dieser noch eine kurze Einschwingphase benötigt. Sollte sich dieses Verhalten störend auf das Modell auswirken, müsste diese Zeit mit einem Mono-Flop überbrückt werden.

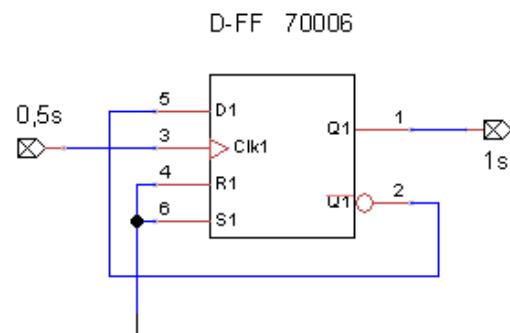
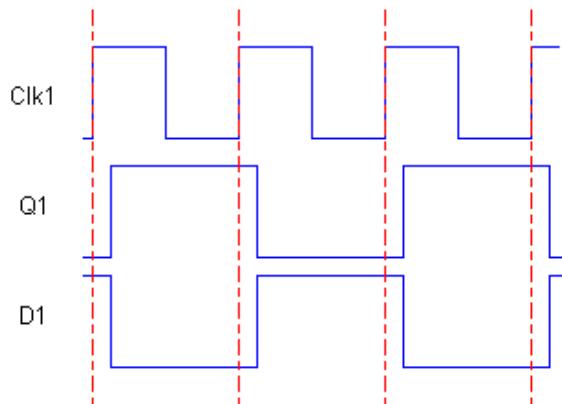


Abb. 4: Schaltung des Echtzeitgenerators

Wie erhält man einen Ein-Sekunden-Taktgenerator?

Wenn ein Sekundentakt benötigt wird, schließt man den Echtzeitgenerator an ein T-Flip-Flop (T-FF) an. Die eingehende Taktfrequenz wird durch das T-FF halbiert.

Ein T-FF lässt sich aus einem D-Flip-Flop (D-FF) oder ein JK-Flip-Flop (JK-FF) aufbauen. Bei einem D-FF erhält man durch Stecken einer Verbindung zwischen dem Ausgang \bar{Q} und dem Steuereingang D ein T-FF. Beim JK-FF müssen die beiden Steuereingänge J und K auf high gelegt werden, damit das JK-FF zum T-FF wird.

Im Diagramm in Abb. 4 ist zu erkennen, dass die Periodendauer am Ausgang Q doppelt so lang ist wie der Takt am Eingang Clk des D-FF.

Digitale Uhr mit Zusatzmodul 70051

Den Echtzeitgenerator habe ich gleich in die eingangs erwähnte Uhr eingebaut. Mit ein paar kleinen Änderungen erhält man eine Echtzeit-gesteuerte digitale Uhr (Abb. 5).

Der Echtzeitgenerator wird hier an eines der freien ODER-Gatter eines Moduls 70013 angeschlossen, sodass der Takt sichtbar und das Signal verstärkt wird. Der Taktgenerator, der vorher die Sekunden vorgegeben hat und dafür justiert werden musste, wird nun in einen Binärzähler umkonfiguriert, in dem der erste Jumper auf „ext.“ (externer Takt) und der zweite auf „D“ (digitales Signal, vom Echtzeitgenerator) gesetzt wird. Der Ausgang des ODER-Gatters wird auf den Takt-Eingang des Taktgenerators gelegt und gibt jetzt den Takt an. Die Schaltung mit den erforderlichen Änderungen ist in Abb. 6 zu sehen.

Der Echtzeitgenerator liefert aber einen 0,5-Sekunden-Takt; damit würde die Uhr doppelt so schnell laufen wie gewünscht. Dafür kann man entweder ein T-FF statt des ODER-Gatters einsetzen oder man muss alle Anschlüsse auf der Ausgangsseite des Taktgenerators um eine Position nach oben verschieben ($Ck8 \rightarrow Ck7$; $Ck7 \rightarrow Ck6$ usw.), wobei die Leitung von $Ck1$ jetzt direkt an den Ausgang des ODER-Gatter angeschlossen wird.

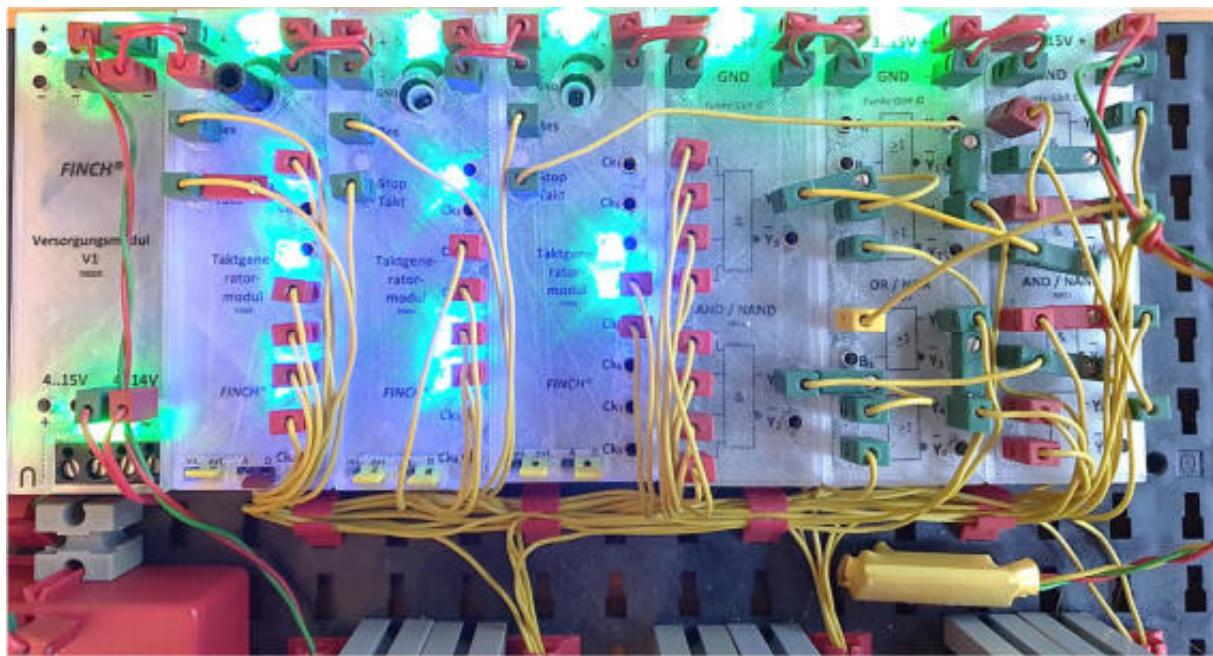


Abb. 5: Steuerung des Modells „Digitale Uhr“ mit eingegebautem Echtzeitgenerator

Verfügbarkeit

Das Zusatzmodul 70051 ist seit Anfang Juni bei Franz Santjohanser im Online-Shop erhältlich. Das Gleiche gilt für das Zusatzmodul 70050 Überspannungsschutz.

Schlusswort

Weitere Informationen zu den Elektronikmodulen findet ihr auf Instagram, Videos mit Modellen auf youtube, jeweils unter dem Stichwort „Elektronikmodule“. Einen Gesamtüberblick über alle verfügbaren Elektronikmodule findet ihr im Online-Shop von Franz Santjohanser [2].

Quellen

- [1] Hans-Christian Funke: *Digitale Uhr.* ft:pedia 3/2020, S. 78–82.
- [2] Online-Shop von Franz Santjohanser: <https://santjohanser.de/fischertechnik/einzelteile/elektronik/module/>

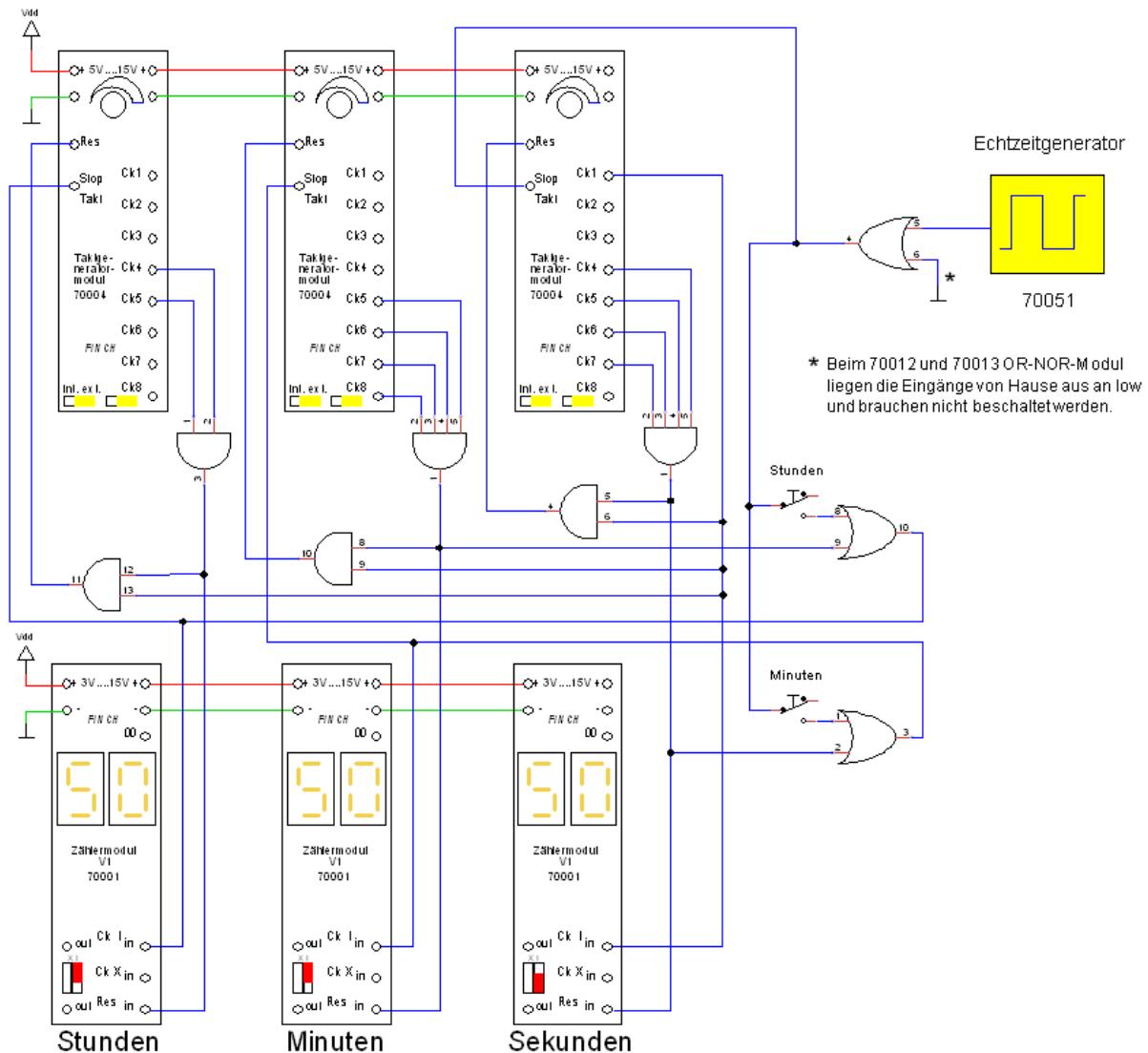


Abb. 6: Fertig geänderte Schaltung mit integriertem Echtzeitgenerator

Elektronik

Controller-Erweiterung

Hans-Christian Funke

Über eins sind wir uns alle sicherlich einig: Die Anzahl der Steuereingänge und Steuerausgänge am Controller sind häufig nicht ausreichend.

Welche Anforderung eine Erweiterung für die fischertechnik Robotics Controller erfüllen muss, stellt sich für jeden anders dar. Der eine benötigt mehr Eingänge, der andere wiederum mehr Ausgänge, ein Dritter benötigt keine analogen Eingänge, sondern nur die reine logische Information, ob sich

ein Schalter in Position „ein“ oder „aus“ befindet. Je nach Anforderung muss entweder das vom Hersteller angebotene System verwendet werden, oder man kann auf andere Lösungen ausweichen.

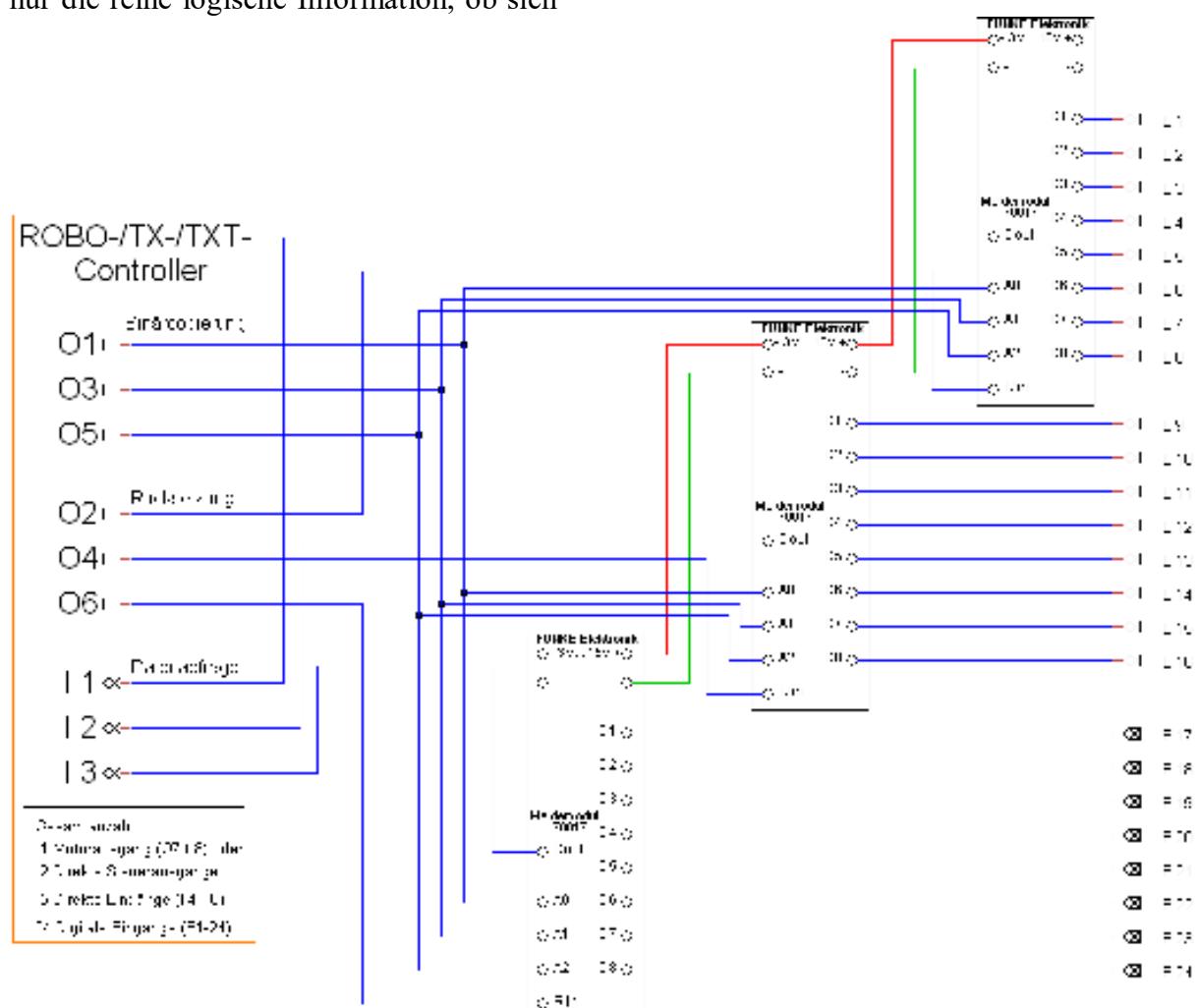


Abb. 1: Controller-Erweiterung mit 24 digitalen Speichereingängen

Auf die Herstellerlösung muss meistens zurückgegriffen werden, wenn es um die analogen Ausgänge zur Ansteuerung von Motoren mit unterschiedlicher Drehzahl und Belastung geht. Soll hingegen nur etwas ein- oder ausgeschaltet werden (digitaler Ausgang) vereinfachen sich die Anforderungen und es stehen eine Vielzahl von Alternativen zu Verfügung.

In diesem Beitrag zeige ich eine Möglichkeit, wie ein Controller um eine große Zahl von digitalen Eingängen bzw. digitalen Ausgängen erweitert werden kann. Als Basis verwende ich aus der Reihe der Elektronikmodule das Meldemodul 70017 für weitere Eingänge und das Steuermodul 70016 für weitere Ausgänge. Die grundsätzliche Ansteuerung dieser beiden Elektronikmodule findet ihr in der Ausgabe ft:pedia 4/2020 ausführlich beschrieben, und es gibt Programmbeispiele zum Herunterladen [1].

Erweiterung mit digitalen Steuereingängen

Jedes Meldemodul stellt acht digitale, mit Speichern hinterlegte Eingänge zur Verfügung. Die Speicher sind alle bei Inbetriebnahme zurückgesetzt. Die Speicher registrieren jeden noch so kurzen Impuls auf der Eingangsleitung. Die Abfrage der Speicherzustände erfolgt über einen Demultiplexer, der über drei Adresseingänge (A0-A2) per Binärcodierung angesteuert wird. Ebenso erfolgt die gezielte Rücksetzung jedes einzelnen Speichers.

In Abb. 1 ist ein Schaltungsbeispiel für eine Erweiterung mit 24 Eingängen zu sehen und in Abb. 2 ein Beispiel für 64 Eingänge; bei mehr als drei Meldemodulen müssen separate Demultiplexer (70024) eingesetzt werden – einer für die Datenauswahl und einer für die Rücksetzung der Speicher.

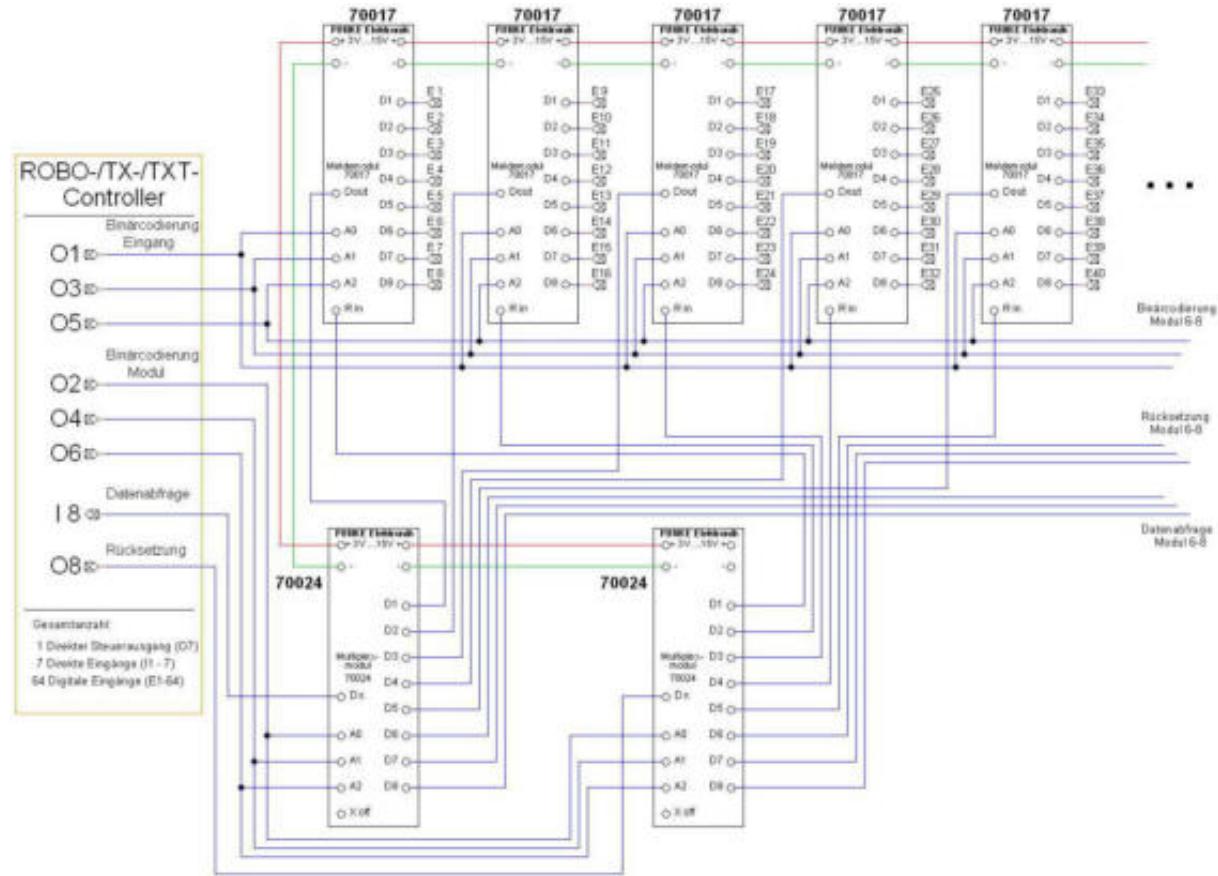


Abb. 2: Controller-Erweiterung mit 64 digitalen Speichereingängen

In der Schaltung (Abb. 2) werden über die Steuerleitungen O1, O3 und O5 die Melde-module adressiert. Über die Steuerleitungen O2, O4 und O6 erfolgt die Auswahl, von welchem Modul der Eingang an I8 bereitgestellt wird bzw. welcher Speicher über die Steuerleitung O8 zurückgesetzt werden kann. Mit Hilfe des Unterprogramms „Binärcodierung“ wird eine Dezimalzahl in einen Binärkode umgewandelt, um so recht einfach die Ansteuerung der Demultiplexer vorzunehmen.

Erweiterung mit digitalen Steuerausgängen

Beim Steuermodul wurde eine andere Technik zur Übermittlung der Daten angewendet: Die Daten werden seriell übergeben und über ein Schieberegister an die richtige Position befördert.

Diese Variante hat den Nachteil, dass mit steigender Anzahl an Ausgängen auch der zeitliche Aufwand für die Datenübertragung ansteigt. Vorteil ist, dass eine beliebige Anzahl Schieberegister aneinander gereiht werden kann, und zwar mit minimalem Aufwand an Steuerausgängen.

In Abb. 3 ist die Anschaltung der Steuermodule an den Controller zu sehen, wobei immer nur drei der analogen Ausgänge benötigt werden, um beliebig viele digitale Ausgänge zu generieren. Die Ausgänge besitzen nur eine kleine Endstufe, die je nach Versorgungsspannung bis zu 7 mA je Ausgang zu Verfügung stellen und somit Steuereingänge anderer Einheiten, Elektronikmodule, Relais mit Verstärker oder LEDs mit 1-2 mA (71031-38) Strombedarf betreiben können. Größere Verbraucher müssen über ein Relaismodul (70018) oder schnelle Leistungsschalter (70009, 70019) angeschlossen werden.

Fazit

Es gibt keine pauschale Antwort, welche Erweiterungsmöglichkeit die beste Variante ist (analoge oder digitale Erweiterung). Entscheidend sind die Anforderungen des Modells, wonach die Entscheidung gefällt werden muss, welche Erweiterung sich am besten eignet.

Quellen

- [1] Hans-Christian Funke: *Elektronik-Module (Teil 5)*. [ft:pedia 4/2020](#), S. 80–94.

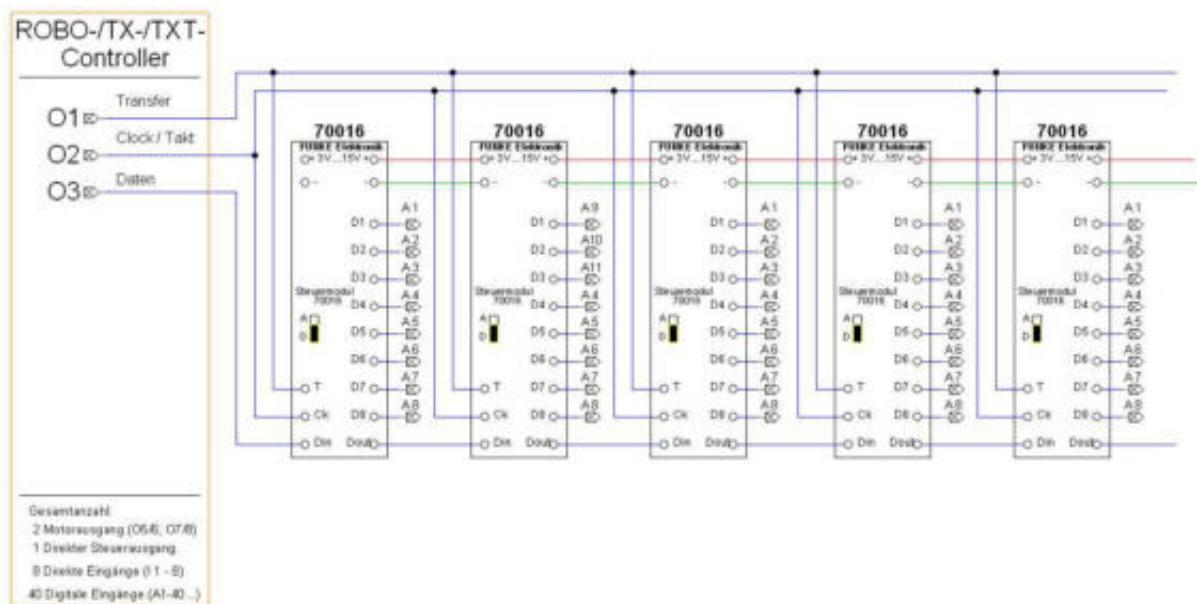


Abb. 3: Controller-Erweiterung für digitale Ausgänge

Modell

Tresor

Hans-Christian Funke

Eines meiner ersten elektronisch gesteuerten Modelle war ein Tresor. Diese Idee habe ich erneut aufgegriffen und ein Modell mit viel Raffinesse entworfen. Es gibt zwei Kodierungssysteme, die zum Öffnen geknackt werden müssen, eine Zeitverlängerung bei Fehlversuchen, eine doppelte Türverriegelung, und wer möchte, kann auch eine Sirene integrieren, die nach dem zweiten Fehlversuch aktiviert wird.

Funktion

Kernstück des gesamten Systems ist eine 4-Bit-Kodierung. Die Kodierung erfolgt im Innenraum des Tresors mittels Schalter oder hier einfach nur Drahtbrücken (Abb. 2). Mit jedem Öffnen des Tresors kann oder sollte die Kodierung geändert werden. So werden aufmerksame Beobachter davon abgehalten, durch bloßes Wiederholen der Kombination den Tresor zu öffnen.



Abb. 1: Der Tresor

Wer schon einmal nebenbei gerechnet hat, wird wissen, dass es bei 4 Bit nur 16 mögliche Kombinationen gibt. Bei der geringen Anzahl von Möglichkeiten könnte man diese tatsächlich schnell durchprobieren und hätte ein leichtes Spiel. Aus diesem Grund gibt es eine doppelte 4-Bit-Kodie-

rung, woraus sich $16 \cdot 16 = 256$ Möglichkeiten ergeben – nun wird es schon nicht mehr so einfach.



Abb. 2: Kodierung

Eine der beiden 4-Bit-Kodierungen muss über einen Drehschalter an der Tür eingestellt werden. Für die anderen vier Bit benötigt man ein Hilfsmittel. Ohne dieses Hilfsmittel wird man niemals in der Lage sein, den Tresor zu öffnen!

Im Dachbereich des Tresors sind vier Reedkontakte eingebaut, die eben nur mit Magneten aktiviert werden können. Auch müssen die Magnete an der richtigen Stelle platziert werden, ansonsten schließen die Reedkontakte nicht.

Wird mit den Magneten die korrekte 4-Bit-Struktur eingestellt, dann wird damit eine der beiden Türverriegelungen geöffnet. Über den Drehschalter müssen die anderen vier Bit korrekt eingegeben werden, damit sich die zweite Türverriegelung öffnen lässt. Erst wenn beide Türriegel geöffnet sind, lässt sich die Tür vom Tresor öffnen.

Ein Beispiel

Auf einem Experimentiermodul im Tresor werden die Drahtbrücken gesteckt. Drahtbrücken, die nach „+“ gesteckt sind, entsprechen einer logischen 1 (high) – hier Bit 1, 3 und 4. Die offenen oder an „-“ gesteckten Drahtbrücken entsprechen einer logischen 0 (low) – hier Bit 2. Damit ergibt sich für dieses Beispiel die Kodierung 1011.

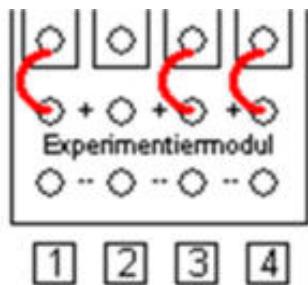


Abb. 3: Beispiel-Kodierung

Für die Reedkontakte gibt es eine definierte Reihenfolge und Position für die einzelnen Bits. Wem es zu kompliziert ist, sich zu merken, wo sich welches Bit befindet oder ein Magnet abgelegt werden muss, der kann weitere Hilfsmittel am Dach des Tresors anbringen, z. B. kleine Papierstücke, die die Position der Magnete kennzeichnen. Eine weitere Hilfe kann die Beschriftung der Papierstücke mit der Reihenfolge (1-4) sein. Welches Bit wohin kommt, entscheidet jeder für sich selbst bzw. wird bei der Verdrahtung des Tresors festgelegt.

An jeder Position, die auf high liegt, muss ein Magnet abgelegt werden. Wird ein Magnet falsch abgelegt, kann der Tresor in diesem Zyklus nicht mehr geöffnet werden. (Was ein Zyklus ist, erkläre ich später.) Liegen alle Magnete an der richtigen Position, öffnet sich ein Türriegel.

Es spielt keine Rolle, ob zuerst die Magnete in Position gebracht werden oder der Drehschalter bedient wird. Beim Drehschalter ist auch nicht die Reihenfolge entscheidend, sondern nur das Setzen der richtigen Bits. In unserem Beispiel müssen die Bits 1, 3 und 4 gesetzt werden. Somit muss der Drehschalter dreimal eingestellt werden. Dazu wird eine der drei Zahlen nach oben gestellt, sodass man diese lesen kann. Jetzt muss ein geheimer Bestätigungstaster gedrückt werden, der sich in der Tür befindet. Damit wird die erste Zahl gesetzt. Das Gleiche muss nun für die anderen beiden Zahlen wiederholt werden. Sind alle Bits korrekt gesetzt, öffnet sich der Türriegel. Auch hier gilt: Wird eine falsche Zahl gesetzt, kann die Tür in diesem Zyklus nicht mehr geöffnet werden.

Leuchtet das Licht an der Außenseite des Tresors, können keine Eingaben über den Drehschalter oder mit Hilfe der Magnete vorgenommen werden. Das Licht signalisiert eine Wartezeit. Diese Wartezeit von z. B. 5 s kann über ein Mono-Flop (MF) eingestellt werden. Erlischt das Licht, beginnt ein Zyklus, in dem die Tür geöffnet werden kann. Die Länge dieses Zyklus von z. B. 1 min könnt ihr über ein weiteres MF ebenfalls selbst festlegen. Innerhalb dieses Zyklus müssen die Magnete an der richtigen Position abgelegt, die Kodierung über den Drehschalter eingegeben und die Tür geöffnet werden.

Wird vergessen, die Tür zu öffnen, wird die Tür mit Ablauf des Zyklus wieder verriegelt. Wird die Tür rechtzeitig geöffnet, wird über einen Türschalter die Verriegelung überbrückt. So wird auch verhindert, dass

die Verriegelung aktiviert werden kann, so lange die Tür geöffnet ist.

Mit einer optionalen Erweiterung kann man die Wartezeit pro abgelaufenen Zyklus verlängern – so wie es bei manchen Zugangssicherungen passiert, wenn man das Passwort falsch eingegeben hat.

Eine weitere Möglichkeit – ebenfalls optional – ist es, ein Alarmsignal oder eine Sirene auszulösen, wenn der zweite Zyklus beendet wurde, ohne die Tür zu öffnen.

Ein spannendes Modell, das nicht nur viel Spaß beim Aufbau mit sich bringt, sondern auch dazu verleitet, weitere Hindernisse oder Türsicherungen zu erfinden und einzubauen.

Aufbauanleitung

Im Folgenden habe ich verschiedene Aufbauschritte abgebildet, in denen alle Details

hoffentlich gut ersichtlich sind. Am Ende findet ihr noch eine Stückliste zu den verbauten Teilen.

Hier einige Baugruppen und Details für die Türverriegelung:

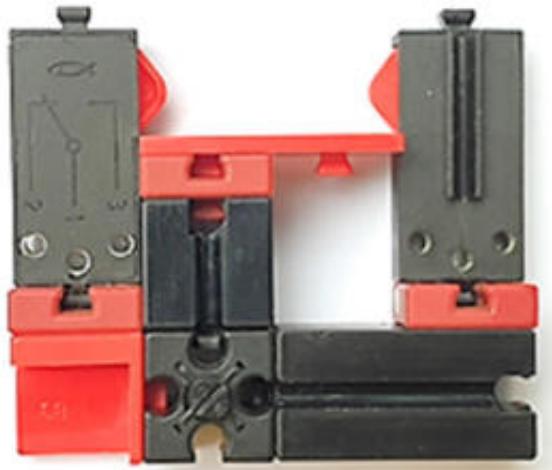


Abb. 5: Riegel Stopp-Schalter

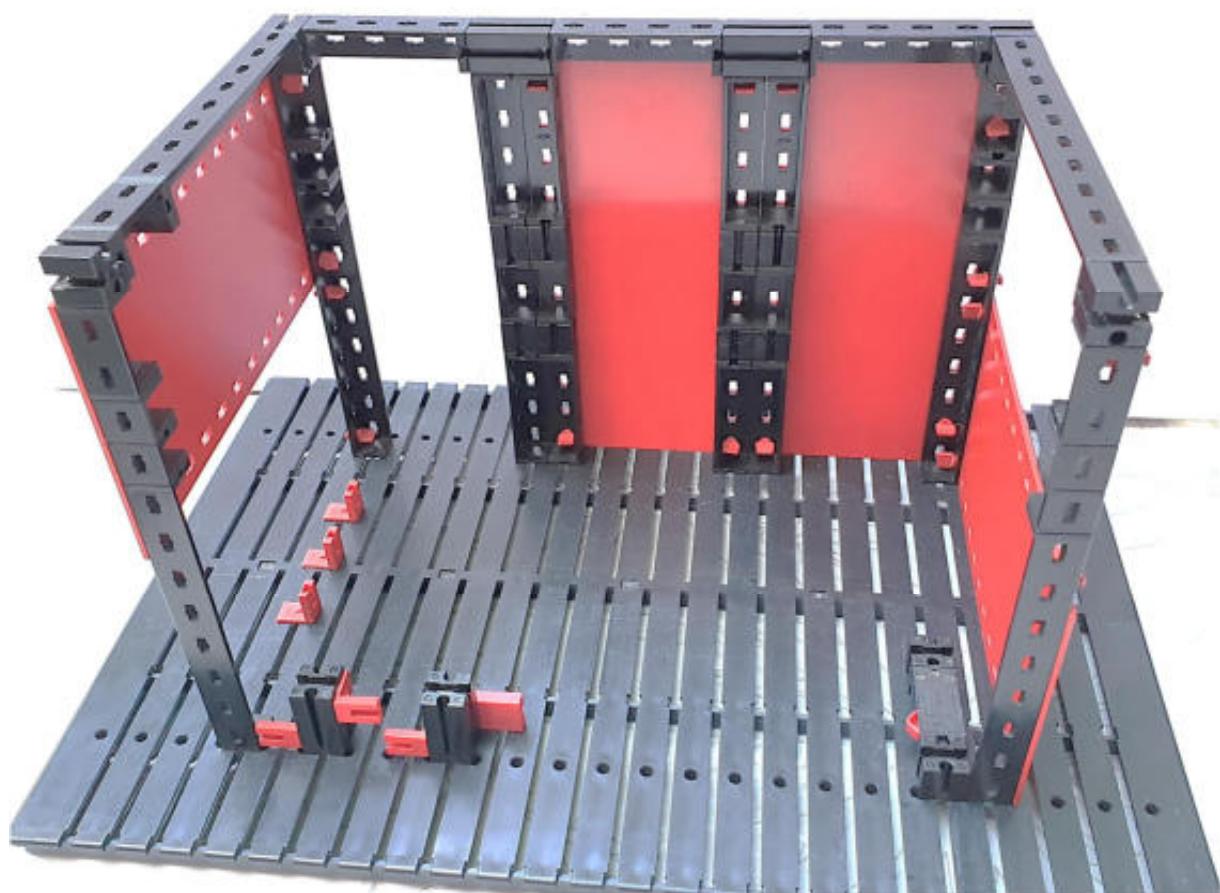


Abb. 4: Baustufe 1

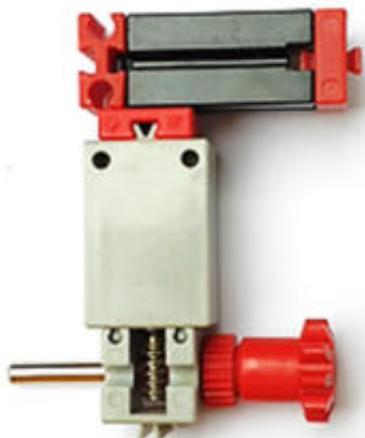


Abb. 7: Motor mit Getriebe (1)

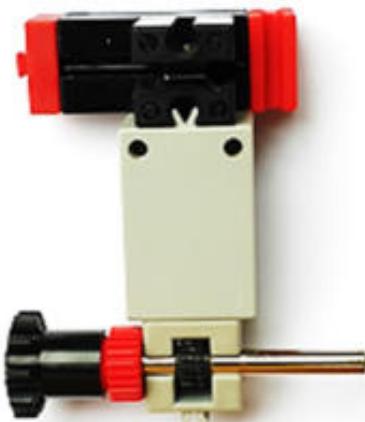


Abb. 8: Motor mit Getriebe (2)

Beim Aufbau hatte ich keine U-Getriebe zur Hand, sodass ich auf einen Ersatz zurückgreifen musste. Das U-Getriebe [31078](#) kann zusammen mit einer U-Achse 40 verwendet werden. In Abb. 9 ist ein U-Getriebe mit einer U-Achse 60 abgebildet. Die Höhe muss entsprechend angepasst werden, damit das Ritzel die Zahnstange führen kann.



Abb. 9: U-Getriebe

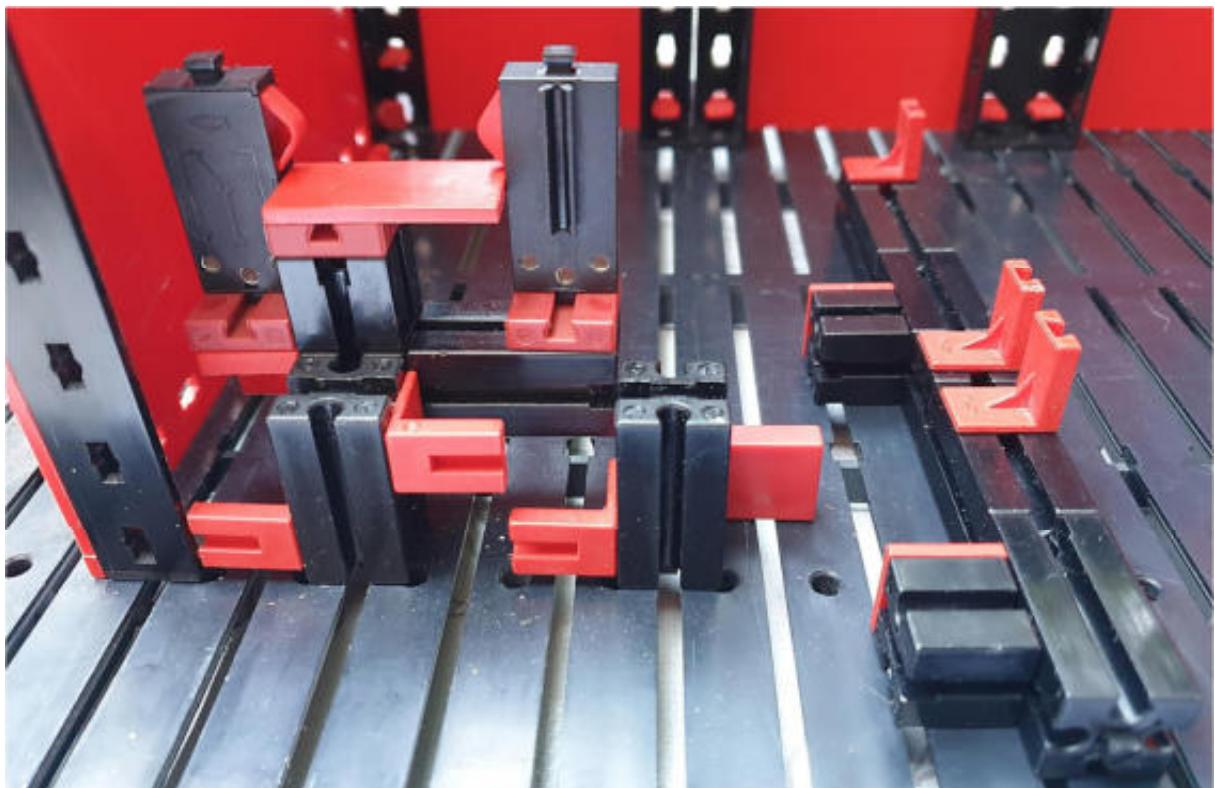


Abb. 6: Baustufe 2

Einige wenige Ergänzungen musste ich später noch vornehmen, auf die ich im Verlauf des Aufbaus noch eingehen werde. Zum Beispiel sind einige Reedkontaktehalter zur Kabelführung an Stellen zu sehen, die ich später dort nicht verwendet und dafür anderswo eingesetzt habe.

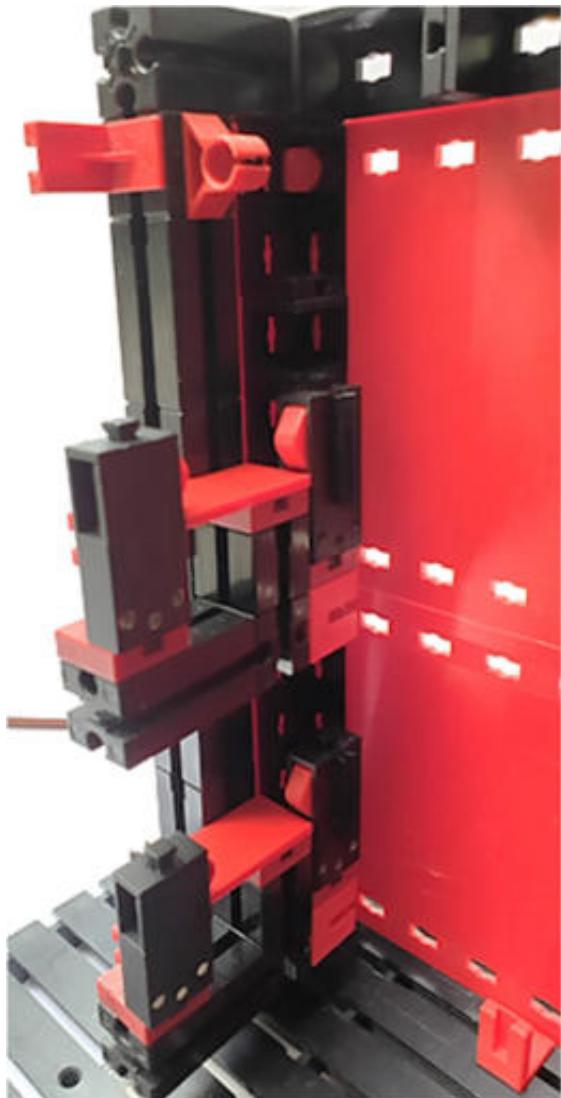


Abb. 10: Baustufe 3

In Abb. 10 wurden die Baugruppe aus Abb. 4 zweimal verbaut. Die liegende Baugruppe rechts in Abb. 6 wurde etappenweise verbaut, damit die äußeren Bauplatten besser eingesetzt werden können.

In Abb. 12 wurde ein Türriegel eingesetzt. Der Baustein 15 mit der Bauplatte $15 \cdot 15$

muss so ausgerichtet werden, dass diese minimal tiefer steht. Ansonsten hakt die Zahnstange an der Kante der Bauplatte $15 \cdot 15$.



Abb. 11: Baustufe 4



Abb. 12: Türriegel

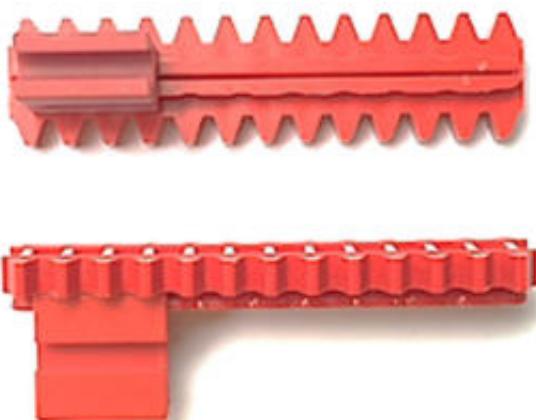


Abb. 13: Zahnstangen als Türriegel

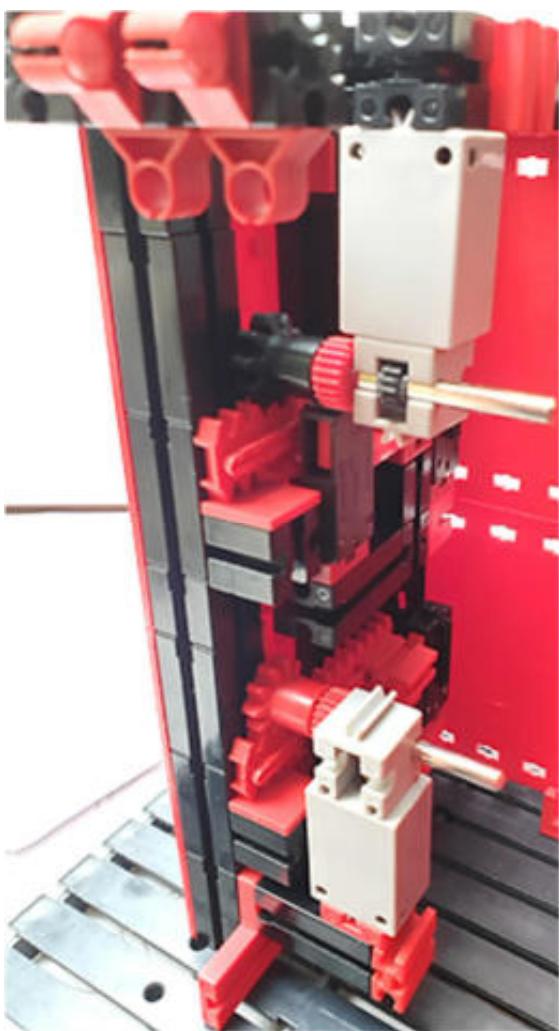


Abb. 14: Motoren aus Abb. 7 und 8 montiert

Die gezeigte Anordnung der Elektronikmodule hat sich bei der Verkabelung als sinnvoll erwiesen. Die Vierer-Baugruppen von Elektronikmodulen werden mit zwei senkrechten und zwei waagerechten Bauplatten

miteinander verbunden. Die Vierermodule können dann durch Einschieben von unten nach oben (siehe Abb. 18, 19) in die Nischen auf der Rückseite des Tresors eingesetzt werden. Es bedarf keiner weiteren Befestigung.



Abb. 15: Fertigstellung der Türverriegelung



Abb. 16: Verbindung der Elektronikmodule mit Bauplatten 15 · 30

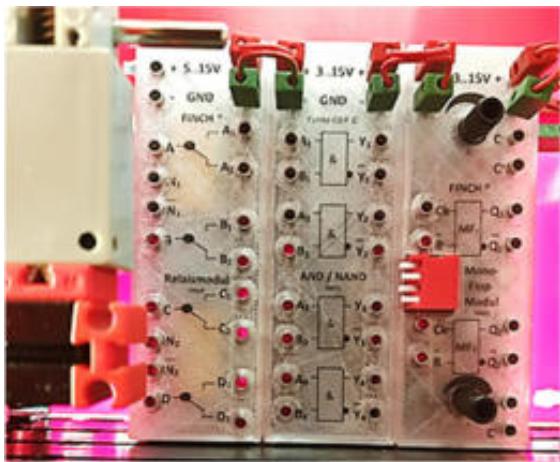


Abb. 17: Verbundene Elektronikmodule



Abb. 18: Elektronikmodule im Viererpack



Abb. 19: Von unten nach oben eingesetzt Baugruppen für die Tür

Jetzt ist der Tresorraum im Wesentlichen schon fertiggestellt. Was noch fehlt ist die Tür (Abb. 20).

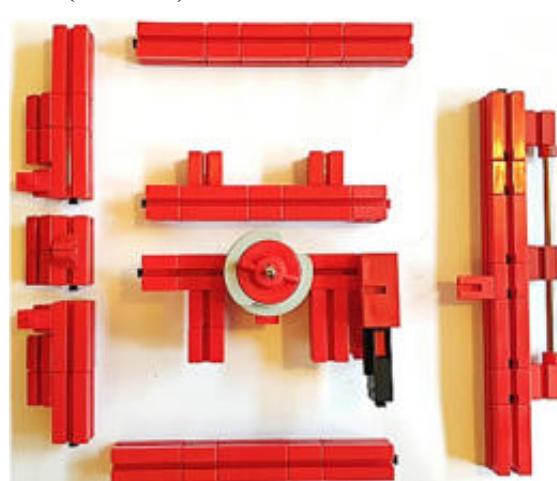


Abb. 20: Die Tür in einer Art Explosionsdarstellung



Abb. 21: Bestätigungstaster

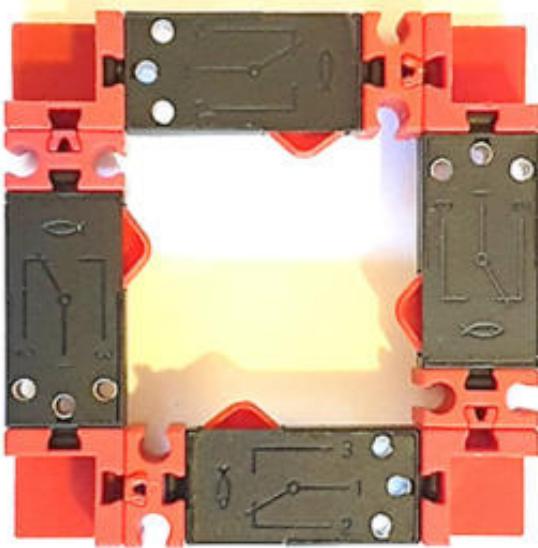


Abb. 22: Taster für den Drehschalter

Die Achse 110 an der Türinnenseite ist die Halterung für die Türriegel. Beim Zusammenbauen des Drehschalters sollte die Justierung gleich vorgenommen werden. Steht die „1“ oben, muss auf der Rückseite die Nut der Schalscheiben ebenfalls oben sein.



Abb. 23: Start mit dem Bestätigungstaster...

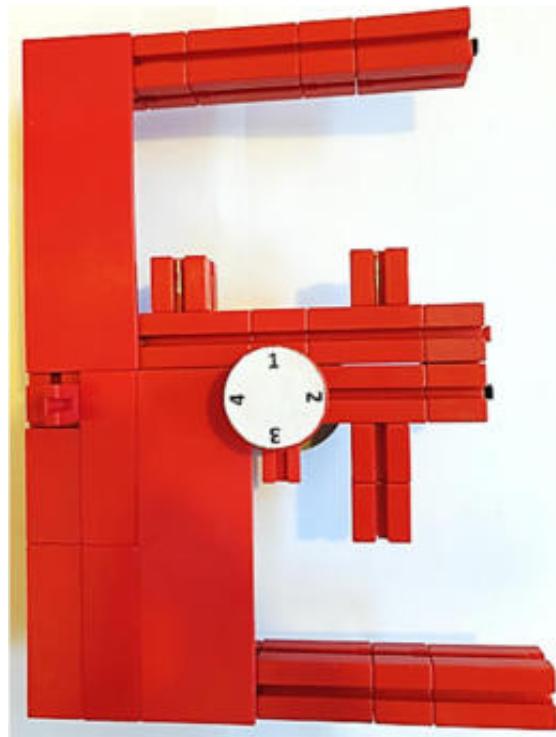


Abb. 24: ... und stückweiser Aufbau von links nach rechts

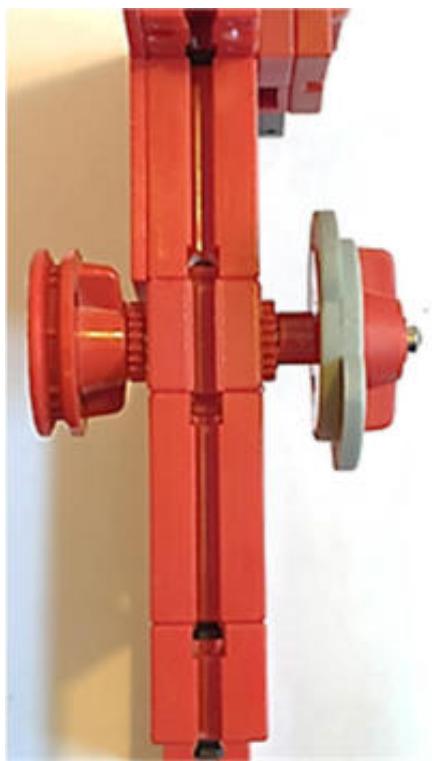


Abb. 25: Montierter Drehknopf des Drehschalters

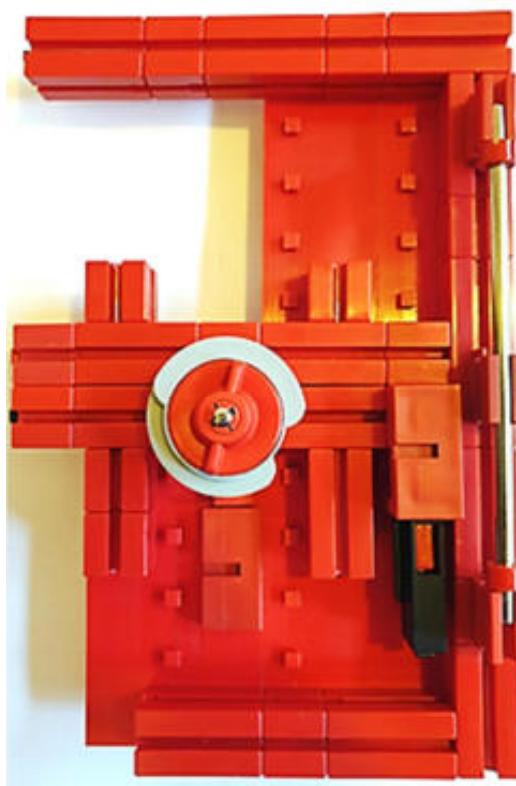


Abb. 27: Innenseite der Tür

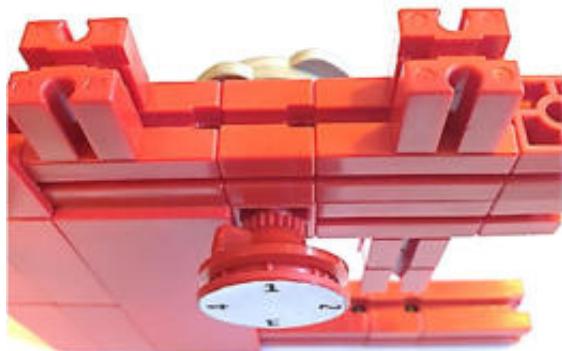


Abb. 26: Justierung mit der Nut oben bei „1“

Im letzten Schritt werden die Baugruppe für den Drehschalter montiert und Drehgelenke angebracht. Auch diese müssen Stück für Stück zusammen mit den Winkelsteinen 60° montiert werden.

Für den Türschalter war zuerst ein zweiter Winkelstein 60° vorgesehen (siehe die nächsten Abbildungen). Bei späteren Tests hat sich aber gezeigt, dass ein Winkelstein 30° mit Baustein 5 besser funktioniert.



Abb. 28: Tür mit Bauplatten versehen

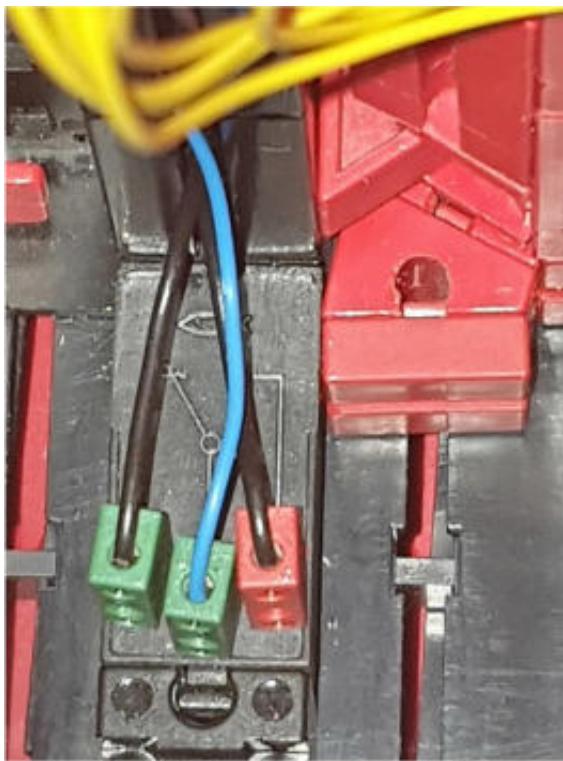


Abb. 29: Türschalter mit Winkelstein 30°

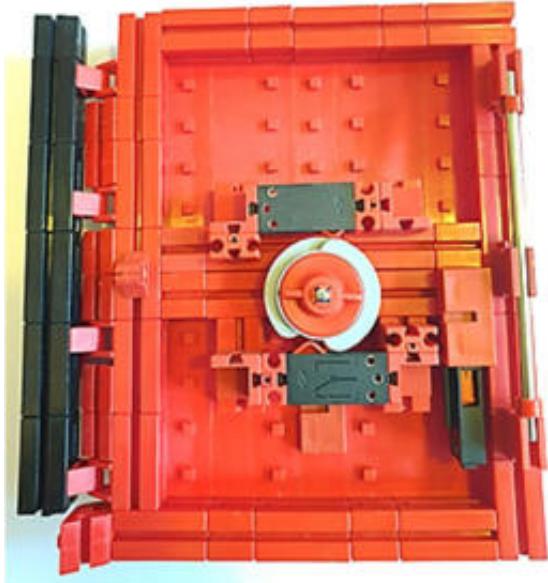


Abb. 30: Türinnenseite

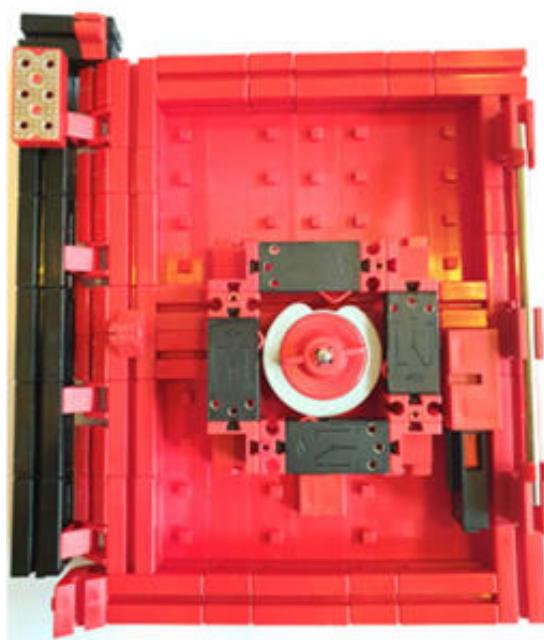


Abb. 31: Fertige Tür mit Rahmenbefestigung

Die fertige Tür kann nun eingesetzt werden und damit ist der Aufbau komplett. Bevor die Tür in den Tresor eingesetzt wird, sollten die vier Taster des Drehschalters justiert werden, sodass diese auch in jeder Drehrichtung schalten. Die Verteilerplatte (nicht mehr erhältlich) ist entbehrlich. Diese ist für die kurze Anbindung der Reedkontakte gedacht. Wer keine Verteilerplatte hat, muss Buchsen zur Verlängerung der Reedkontakt-Kabel verwenden.

Das vollständige Modell

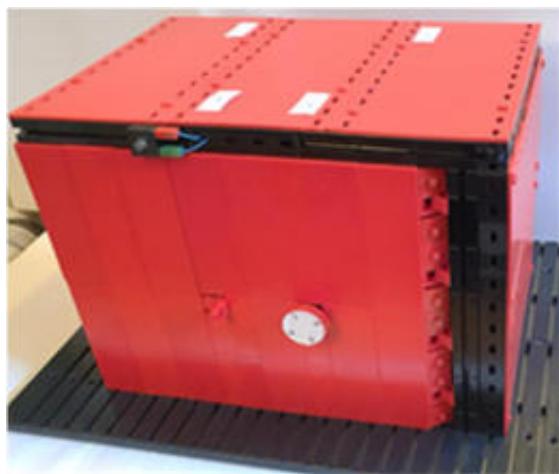
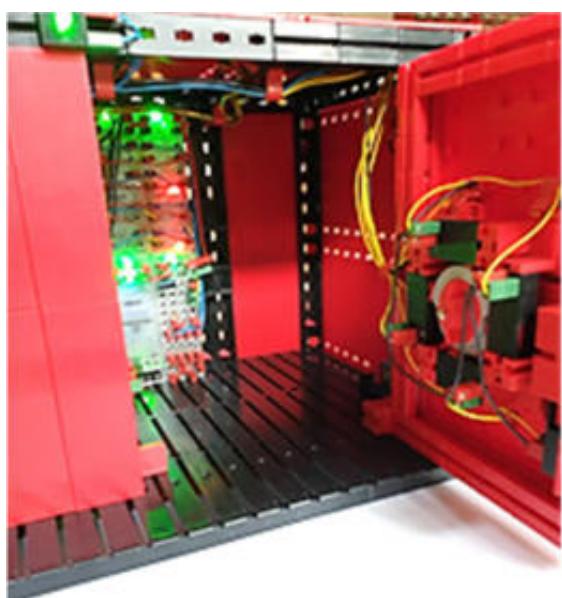


Abb. 32: Frontansicht mit Magnetmarkierungen



In Abb. 35 und 36 seht ihr den vollständigen Schaltplan, den ich auf zwei Seiten aufteilen musste.

Hinweis: Bei einigen Versionen des Mono-Flop-Moduls 70005 wurde die Beschriftung von den C-Anschlüssen am MF1 vertauscht. Der C⁺ Anschluss ist immer oben und der C⁻ Anschluss unten.

Wer Fragen zu dem Projekt hat, kann sich gerne an mich wenden unter elektronik@funke4you.de.

Diese Steuerung könnte man auch durch einen Controller ersetzen. Spannend wird es, wie die zehn Schaltkontakte dann verwaltet werden: entweder über eine Erweiterung (Extension), eine Pseudoverwaltung durch Doppelbelegung bzw. Kodierung der Schalter oder unter Zuhilfenahme von Ergänzungsmodulen (Multiplexmodul 70024 oder Meldemodul 70017).

Alle Teile bis auf die Platten 180 · 90, die Verteilerplatte und die Dauermagnete sind im Online-Shop der fischertechnik-Einzelteilhändler erhältlich. Als Ersatz für die Dauermagnete könnten auch die Dauermagnete der Kugelhalter verwendet werden ([119850](#)).

Artikel Stk. Beschreibung

31010	8	Winkelstein 60°
31011	1	Winkelstein 30°
31031	1	Achse 110
31053	2	Zahnstange 60
31058	2	Nabenmutter
31060	11	Verbindungsstück 15
31064	2	U-Achse 40 Z28
31078	2	U-Getriebe
31322¹	2	Dauermagnet, grün
31323¹	2	Dauermagnet, rot
31328¹	1	Verteilerplatte
31336	~ 90	Stecker rot
31337	58 ²	Stecker grün

¹ Diese Teile sind nicht mehr als Einzelteile erhältlich (nur über den Zweitmarkt).

Artikel	Stk.	Beschreibung
31426	4	Gelenkwürfel-Zunge
31436	4	Gelenkwürfel-Klaue
31928	3	Federnocken 15
32064	2	Baustein 15, rot, Loch, 2 Zapfen
32879	27	Baustein 30, schwarz
32881	23	Baustein 15, schwarz
32882	11	Baustein 15, schwarz, 2 Zapfen
35031	2	Flachnabenzange
35049	1	Baustein, rot, 15 · 30 · 5 Nut und Zapfen
35112	2	Ritzel Z10 M1,5
35113	2	Spannzange
35602	1	Grundplatte 390 · 270
35969	8	Reedkontaktehalter
36120	4	Reedkontakte
36293	10	Winkelträger 120, schwarz
36321¹	10	Platte 180 · 90, rot
36323	40	S-Riegel 4
36334	2	Riegelscheibe, rot
36920	12	Winkelträger 30, schwarz
36921	4	Winkelträger 60, schwarz
36922	4	Winkelträger 15, schwarz
36950	4	Winkelträger 15, schwarz, 2 Zapfen
71021	3	Doppellitze 50mm, 4 Stecker
71022	1	Doppellitze 105mm, 4 Stecker
37237	7	Baustein 5, rot, 1 Zapfen
37238	3	Baustein 5, rot, 2 Zapfen
37468	11	Baustein 7,5, rot
37679	3	Klemmbuchse 5
37727	2	Schalscheibe, schwarz
37783	10	Minitaster

Artikel	Stk.	Beschreibung
38240	7	V15 Eck, rot
38241	14	Bauplatte 15 · 30
38242	4	Bauplatte 15 · 45
38245	1	Bauplatte 15 · 90
38246	2	Bauplatte 15 · 15
38251	13	Bauplatte 30 · 90
38415	1	Achse 50, schwarz
38423	17	Winkelstein 10 · 15 · 15
38428	2	Baustein, rot, 15 · 30 · 5 3 Nuten
116251	21	Baustein 30, rot
116252	14	Baustein 15, rot
137096	2	Motor XS
	div.	Kabel versch. Länge
	1	Ausdruck/Beschriftung
71036	1	LED-Baustein blau, 1 mA
70003	1	Versorgungsmodul 3-14 V
70005	1	Mono-Flop-Modul
70007	2	JK-Flip-Flop-Modul
70008	1	RS-Flip-Flop-Modul
70010	1	AND-NAND-Modul (4 Eingänge)
70011	1	AND-NAND-Modul (2 Eingänge)
70014	2	EXOR-EXNOR-Modul
70018	1	Relaismodul
70021	1	Experimentiermodul
70006²	1	D-Flip-Flop-Modul
70015³	1	Multifunktionsmodul
70004³	1	Taktgenerator/Binärzähler
70024⁴	1	Multiplexmodul
71046	1	Kondensatorset
71047	1	Widerstands- und Dioden-Set

Tab. 1: Teileliste

² Optional: Zusatz für Alarmsirene nach zweitem Versuch.

³ Optional: Zusatz für Zeitverzögerung bei Fehlversuchen.

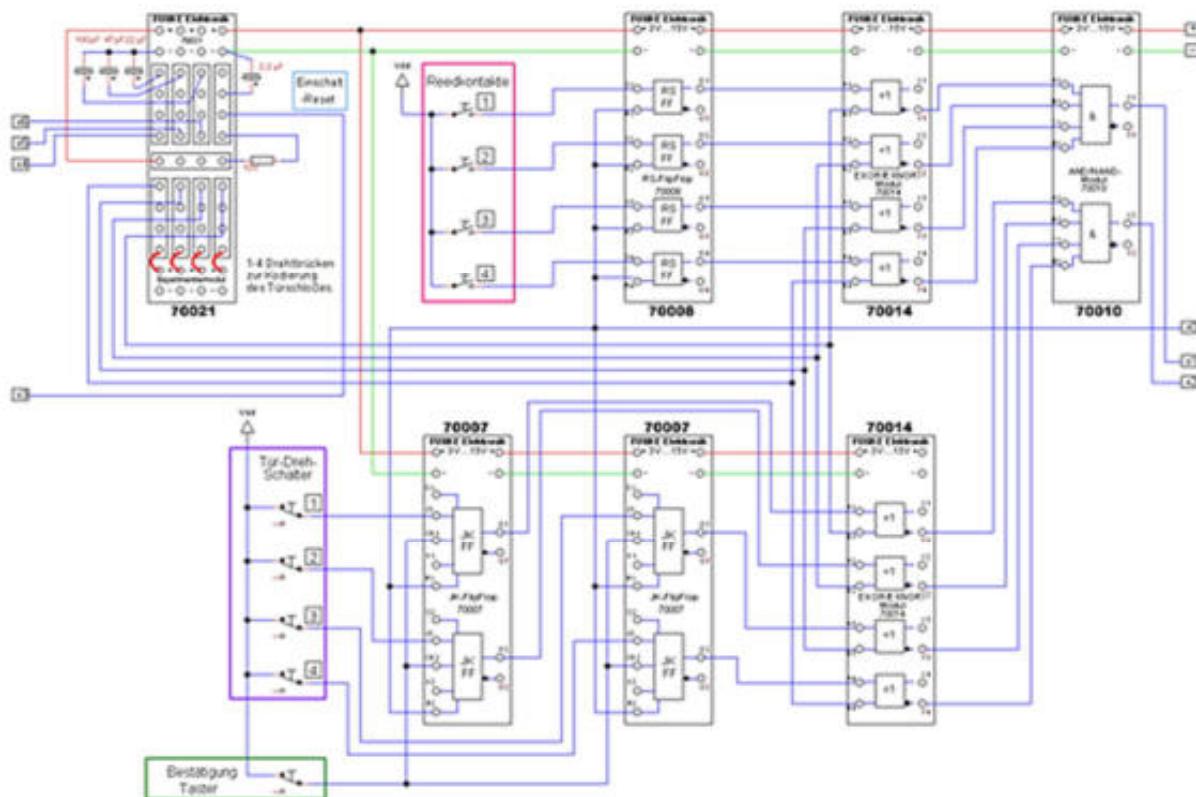


Abb. 35: Schaltplan Teil 1

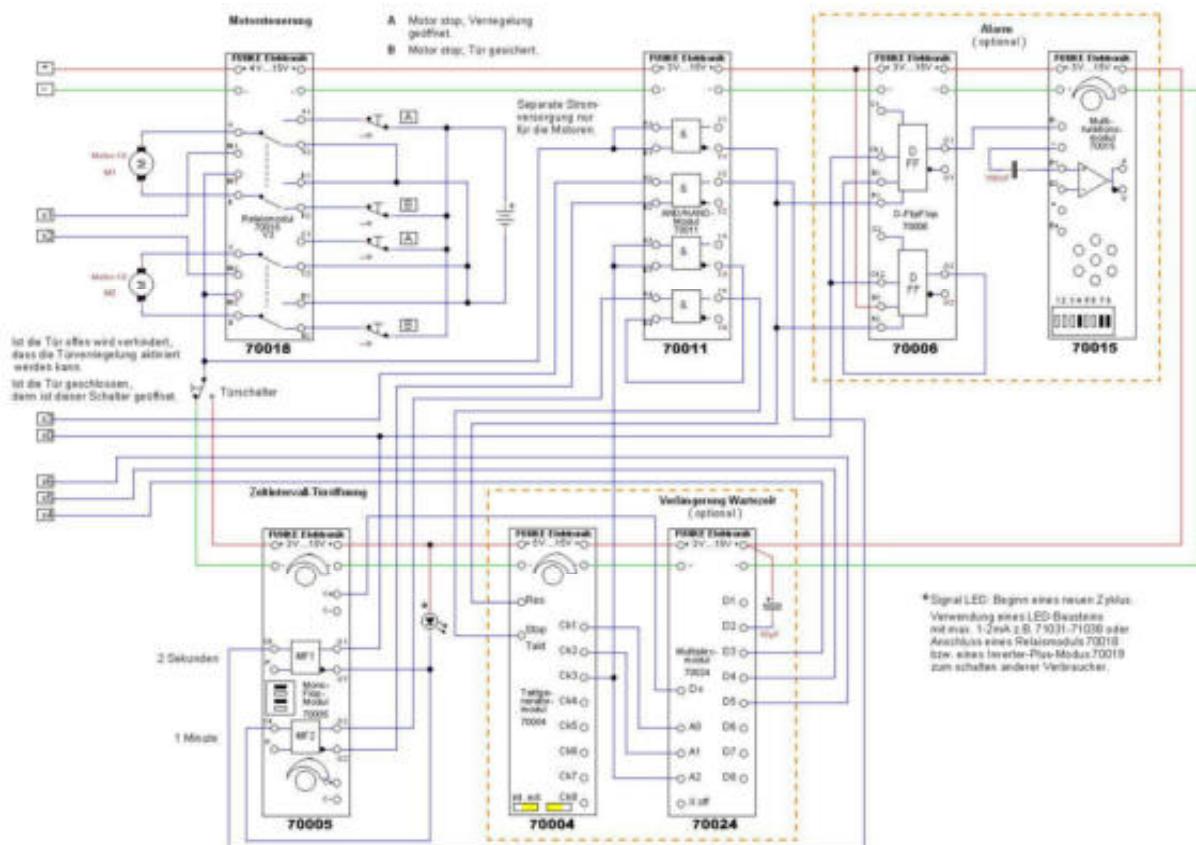


Abb. 36: Schaltplan Teil 2 – mit optionaler Erweiterung Alarm und Zeitverzögerung

Anhang

Eine PDF-Datei mit Beschriftung für die Flachnabe, Markierung für die Dauermagnete auf dem Dach und Zahlenstreifen für die Kennzeichnung auf dem Experimentiermodul steht zum Herunterladen zur Verfügung [1] (siehe Abb. 2).

Meine Empfehlung ist, die Papierstücke nach dem Ausscheiden mit einem Klebestift aufzukleben. Der Kleber ist in der Regel wasserlöslich und lässt sich später problemlos mit einem feuchten Lappen wieder von den Teilen entfernen.

Außerdem gibt es den Schaltplan ebenfalls als PDF zum Herunterladen zur besseren Ansicht.

Sommeraktion

Ein kleiner Hinweis zu den Elektronikmodulen: Vom 1. Juli bis 31. August 2021 wird es die Elektronikmodule 20% preisreduziert geben. Wer dieses Modell nachbauen möchte, sollte diese Aktion nutzen.

Quellen

- [1] Hans-Christian Funke: PDF-Downloads zu diesem Modell auf der [ft:pedia-Seite](#) der ft:Community.

Elektronik

Silberlinge: Original oder Nachbau (Teil 2)

Peter Krijnen

„*Volt, Watt, Ampère, Ohm – ohne mich gibt's keinen Strom*“. Wenn sich Herr Hertz die Mühe macht, diese vier Herren zum Swingen zu bringen, kann das nur zu einem führen: Elektrizität. In diesem zweiten Teil möchte ich den Gleichrichter h4 GB ([36393](#)) vorstellen.

Alternativ zum Gleichrichter ([36393](#)) könnte man auch die IC-Spannungsversorgung ([35732](#)) aus dem Kasten „Elektronik“ ([302539](#)) verwenden.

Ohne Netzteil funktioniert nichts

Am Anfang musste man sich mit einem Batteriestab ([31041](#), Abb. 32) zufriedengeben, der drei 1,5-V-Batterien benötigte, um zusammen bis zu 4,5 V zu liefern. Die noch recht schwachen Motoren waren jedoch für 6,8 V ausgelegt.



Abb. 32: Batteriestab ([31041](#))

fischertechnik nahm jedoch bald einen Transformator in sein Sortiment auf. Dieser Transformator, der nur 5 VA liefern konnte, wurde als „mot. 4“ ([31040](#), Abb. 33) vermarktet. Neben einem einstellbaren Gleichspannungsausgang bot er auch einen nicht regelbaren 6,8-V-Wechselspannungsausgang. Einige Jahre später wurde dieser

Transformator durch eine 7-VA-Version ersetzt. Noch als „mot. 4“ ([30173](#)) war die erste Generation noch mit einem Wechselspannungsausgang ausgestattet. Die zweite Generation erhielt jedoch einen festen Gleichspannungsausgang. Man kann allerdings nicht wirklich von einer Gleichspannung sprechen. Wie sie aussieht, zeigt Abb. 34, ein Screenshot meines Oszilloskops.



Abb. 33: Trafo „mot. 4“ ([31040](#))

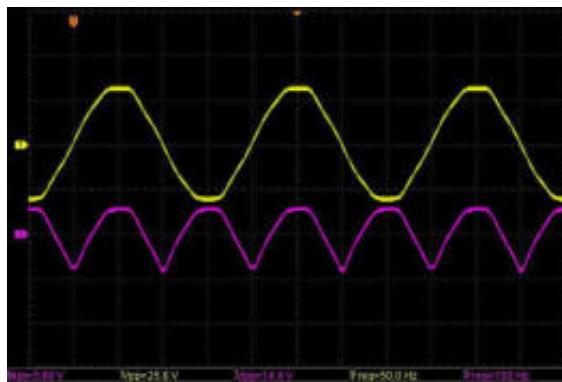


Abb. 34: Unverzerrte Ausgangsspannungen

Kanal 1 (gelb) repräsentiert den Ausgang der Wechselspannungsversion und Kanal 3 (violett) den Ausgang der Gleichspannungsversion. Beide sind hier noch ohne Last.

Es ist deutlich zu erkennen, dass die Wechselspannungsversion eine Frequenz von 50 Hz mit einer UTT von 25,6 V hat. Bei der Gleichspannungsversion ist zu erkennen, dass durch die Gleichrichtung der negative Teil der Wechselspannung umkippt und nun auch positiv wird. Daher beträgt die Frequenz jetzt 100 Hz und der UTT-Wert 14,4 V.

In Abb. 35 sind die Ausgänge mit einer Kombination aus 1 Gleichrichter, 1 Grundbaustein, 1 Relaisbaustein und 2 Lämpchen belastet. Wir sehen, dass sich die Form der Spannungen jetzt etwas geändert hat. Die UTT-Werte betragen jetzt 22,0 V und 11,2 V.

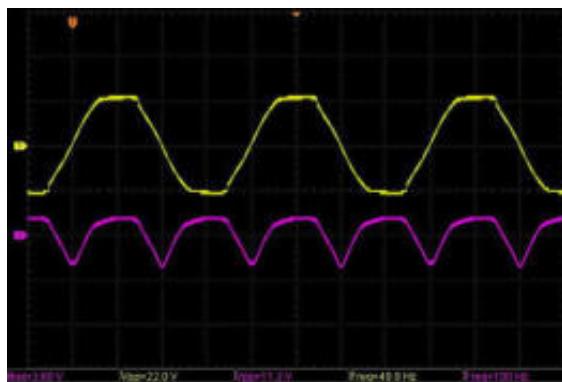


Abb. 35: Verzerrte Ausgangsspannungen durch Last

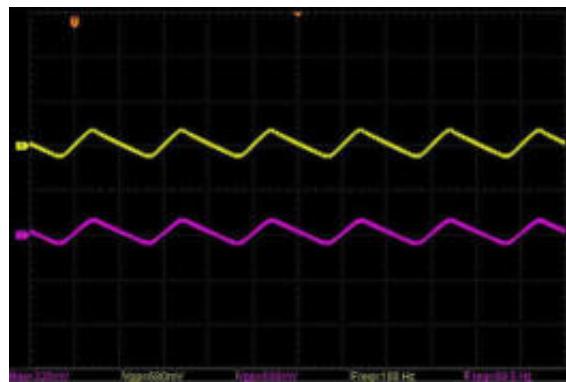


Abb. 36: Gleichrichter-Ausgangsspannung: Wellen, die durch die Last verursacht werden

Beide Kanäle sind auf 10 V/Div eingestellt. Aber wie sieht es an den Ausgängen des Gleichrichters aus? Das ist in Abb. 36 zu sehen.

Um einen besseren Überblick über das Geschehen zu erhalten, muss gezoomt werden. Ich habe jetzt beide Kanäle auf 1 V/Div eingestellt.

Die Ausgangsspannung beträgt ungefähr 10,5 V, aber die Last erzeugt ein welliges Signal. Trotz der Tatsache, dass ein großer Kondensator eingebaut ist, wird er zu schnell entladen, was zu diesen Wellen führt. Sie haben eine UTT von 680 und 600 mV. Eine schwerere Belastung führt zu einer größeren Welle.

Alternative Spannungsversorgung



Abb. 37: Steckernetzgerät ([505287](#)): 9 V, 2,5 A

Da die fischertechnik-Transformatoren viel zu schwach sind, ist es besser, nach einer Alternative zu suchen. Glücklicherweise hat fischertechnik seit einigen Jahren ein Steckernetzteil im Sortiment. Dieses 9 V, 2,5 A starke Steckernetzteil (Abb. 37) wird unter der Artikelnummer [505287](#) verkauft. Es hat jedoch im Laufe der Jahre einen anderen Stecker bekommen, allerdings nicht mit den bekannten 2,5-mm-Steckern.



Abb. 38: Stromadapter Duo 75152



Abb. 39: Power-Controller ([139778](#)) mit Festspannungsausgang

Knobloch bot daher „Stromadapter Duo“ ([75152](#), Abb. 38) an, mit dem von den 5,5-mm- und 3,5-mm-Steckern auf die 2,5-mm-Stecker gewechselt werden konnte.

Wenn man auch den Power-Controller ([139778](#), Abb. 39) aus Plus Power Set

([505283](#)) besitzt, kann man den auch verwenden. Der hat nämlich einen festen Ausgang mit den 2,5 mm-Steckern. Es ist dann weiterhin möglich, die Drehzahl eines Motors zu steuern.

Ich selbst benutze seit Jahren zwei Exemplare des Labor-Netzteils SPS 7330 von ELV (Abb. 40).



Abb. 40: ELV Labor-Netzteil SPS 7330

Wenn ihr ein starkes Netzteil verwendet, benötigt ihr keinen Gleichrichter mehr. Es muss jedoch ein Satz Verbindungskabel zur Verfügung stehen: von 4-mm-Bananensteckern zu 2,5-mm-Steckern.

h4 GB „Gleichrichter“

Wie ich bereits in der Übersicht der Module in Teil 1 [1] angegeben habe, verfügt das Gleichrichtermodul ([36393](#), Abb. 41) nicht über eine Leiterplatte, auf die die Komponenten gelötet werden. Dieses Modul besteht nur aus zwei Komponenten: dem eigentlichen Gleichrichter und einem großen Kondensator. Beide werden mit ihren Verbindungsdrähten direkt an die Anschlussbuchsen gelötet. Der Gleichrichter ist auf die Unterseite des Deckels geklebt.

Die zwei „+“-Anschlussbuchsen sind mit einem roten Draht mit dem Gleichrichter und die „-“-Anschlussbuchsen mit einem blauen Draht verbunden (Abb. 42). Die Kontaktstreifen an den Seiten des Gehäuses (Abb. 43) zur Stromversorgung anderer Module sind ebenfalls mit einem roten und einem blauen Draht mit dem Kondensator

verbunden. Der Deckel wird mit vier kleinen Schrauben am Gehäuse festgeschraubt. Abb. 44 zeigt den Schaltplan.

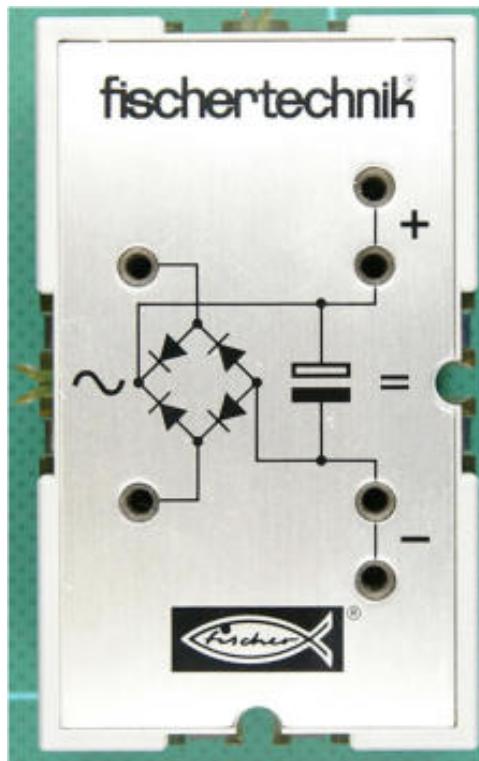


Abb. 41: Gleichrichterbaustein ([36393](#))

Nachbau

Ich habe selbst sechs Gleichrichter, also brauche ich keine neuen mehr. Deshalb habe ich sie nicht nachgebaut. Trotzdem habe ich vier neue Layouts entworfen: Typ 1 (Abb. 45) kann direkt in das Originalgehäuse eingelötet werden. Typ 2 (Abb. 46) ist für mein 4575-Gehäuse. Typ 3 (Abb. 47) ist für mein 6060-Gehäuse und für die 60er-Kassette ([32076](#)). Typ 4 (Abb. 48) ist für das Batteriegehäuse ([32263](#)). Für Typ 2, 3 und 4 verwende ich die 19,4-mm-Bundhülsen 75094.

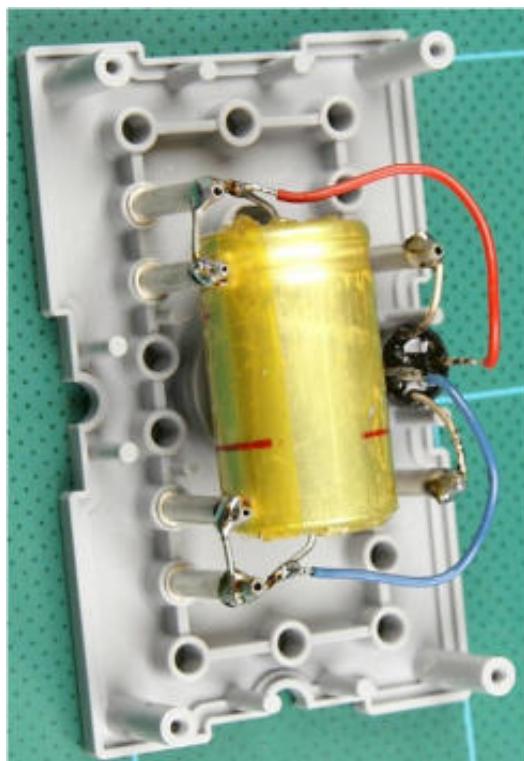


Abb. 42: Unterseite des Deckels mit Gleichrichter und Kondensator

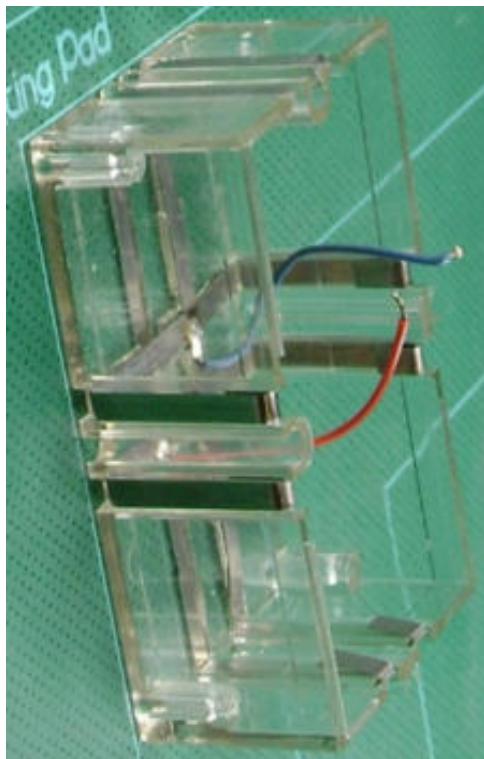


Abb. 43: Gehäuse mit den seitlichen Metallstreifen für die Stromversorgung mehrerer Silberlinge

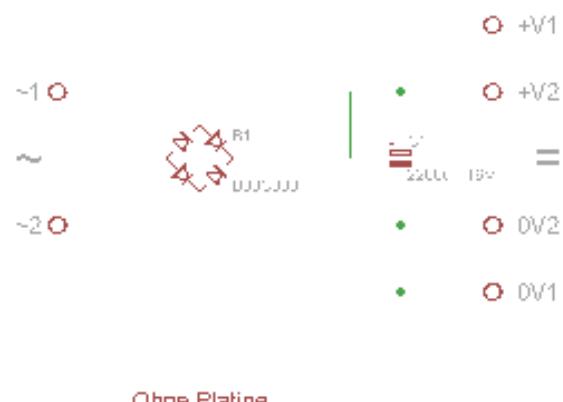


Abb. 44: Schaltbild des Gleichrichters

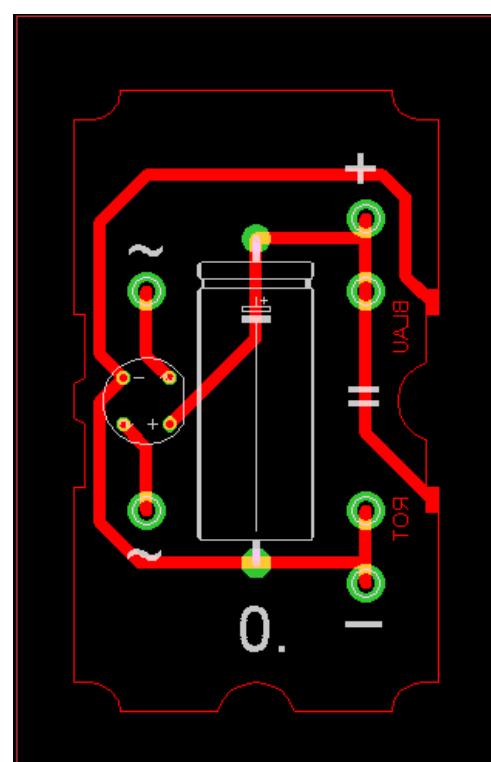


Abb. 45: Nachbau 1: Platinenentwurf für das Original-Silberling-Gehäuse

Für Typ 1 habe ich noch den Brückengleichrichter vom Original verwendet. Ich persönlich mag diese Brückengleichrichter nicht so sehr. Normalerweise liegen die Anschlüsse sich so gegenüber, dass ich Probleme beim Verlegen der Leiterbahnen habe. Daher verwende ich vorzugsweise die Dioden der Serien 1N4000 oder 1N5000. Der 1N4001 kann bis 1 A verwendet werden, ein 1N5401 bis 3 A und z. B. ein BY550 bis 5 A.

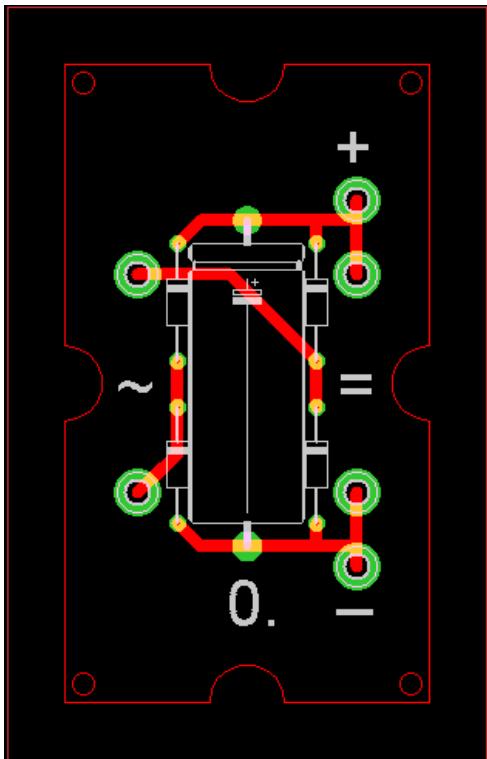


Abb. 46: Nachbau 2: Platinenentwurf für mein alternatives Gehäuse

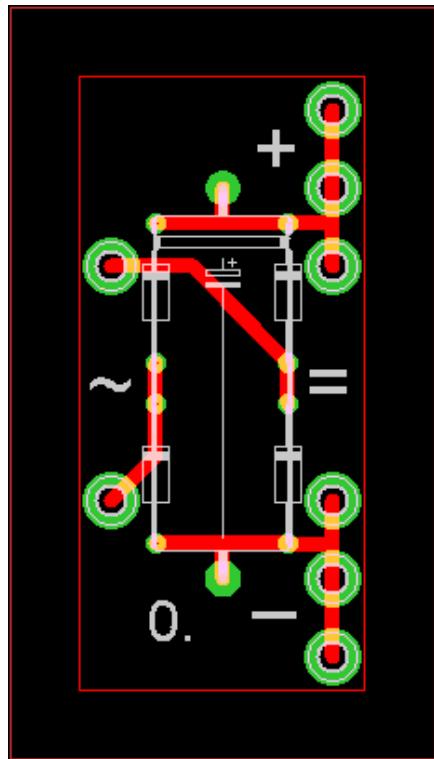


Abb. 48: Nachbau 4: Platinenentwurf für das Batteriegehäuse ([32263](#))

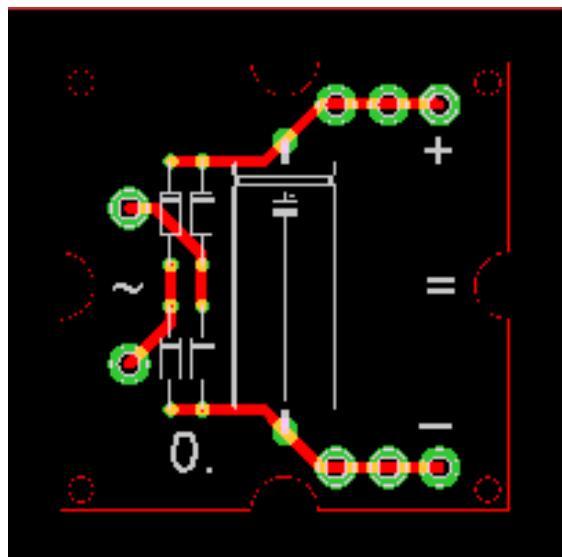


Abb. 47: Nachbau 3: Platinenentwurf für 60er Kassetten ([32076](#))

IC-Spannungsversorgung 35732 (Bild 49)

1981 stellten die Fischerwerke den neuen Elektronikkasten ([30253](#)) vor. Neben einer Leistungsstufe ([36296](#)) und einem Schwellenwertschalter ([36295](#)) gibt es auch eine IC-Spannungsversorgung ([35732](#), Abb. 49).

Die IC-Spannungsversorgung ist, wie der Name schon sagt, um ein IC herum aufgebaut: einen Spannungsstabilisator vom Typ 7805.

Die Schaltung besteht neben dem IC noch aus einer Diode, einem $1000\text{-}\mu\text{F}$ -Kondensator am Eingang des ICs und einem $47\text{-}\mu\text{F}$ -Kondensator am Ausgang des ICs. Die Anschlüsse sind 1,3-mm-Lötstifte. Normalerweise wird ein U-förmiger Kühlkörper zum Kühlen des Stabilisator-IC verwendet. Allerdings dient bei diesem Modul eine Kupferfläche auf der Unterseite der Leiterplatte als Kühlplatte (Abb. 50). Die Schraube und Mutter, mit der der IC auf die Leiterplatte geschraubt wird, sorgt zusätzlich für die

Ableitung der Wärme nach unten. Die Bodenwanne des Moduls ist in Abb. 51 zu sehen.



Abb. 49: IC-Spannungsversorgung ([35732](#))

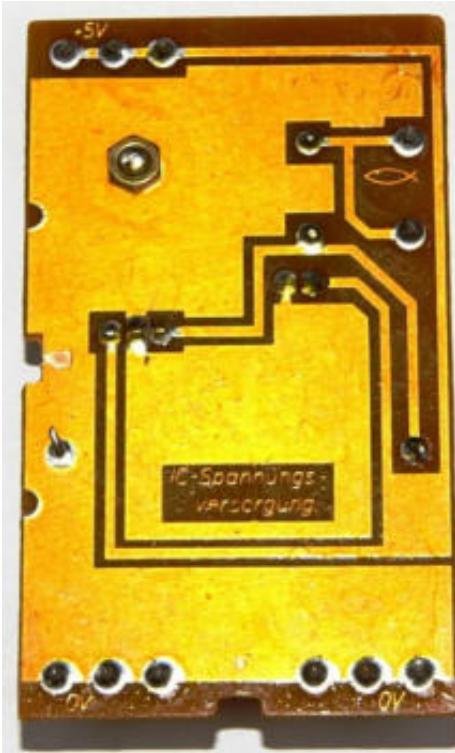


Abb. 50: Lötseite

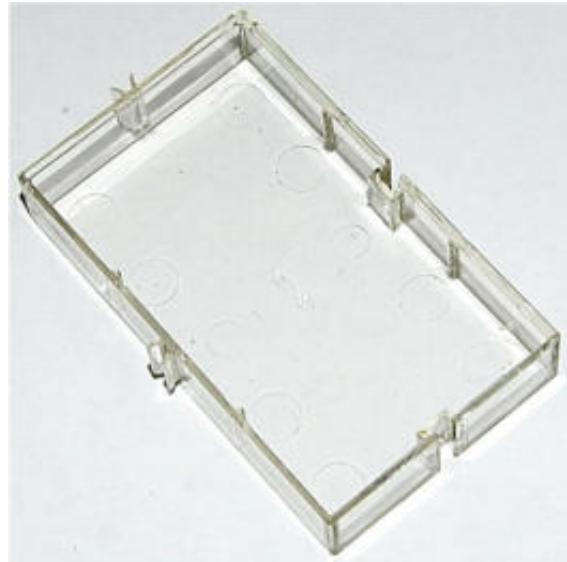


Abb. 51: Wanne

Die Silberlinge arbeiten auch mit 5 V, aber es ist natürlich möglich, das IC durch eine 9-V-Version zu ersetzen. Was möglich ist, seht ihr unter dem Schaltbild (Abb. 52).

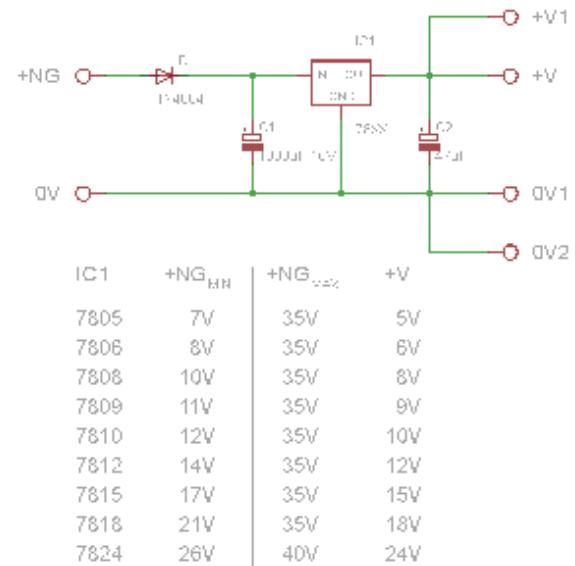


Abb. 52: Platinenlayout der IC-Spannungsversorgung ([35732](#))

Nachbau

Dieses Modul kann man wie das Original mit den 1,3-mm-Lötpins bauen (Layout in Abb. 53). Das wollte ich jedoch nicht. Ich wollte dieses Modul wie den Gleichrichterbaustein aufbauen (Abb. 54) – mit den 2,5-mm-Bundhülsen.

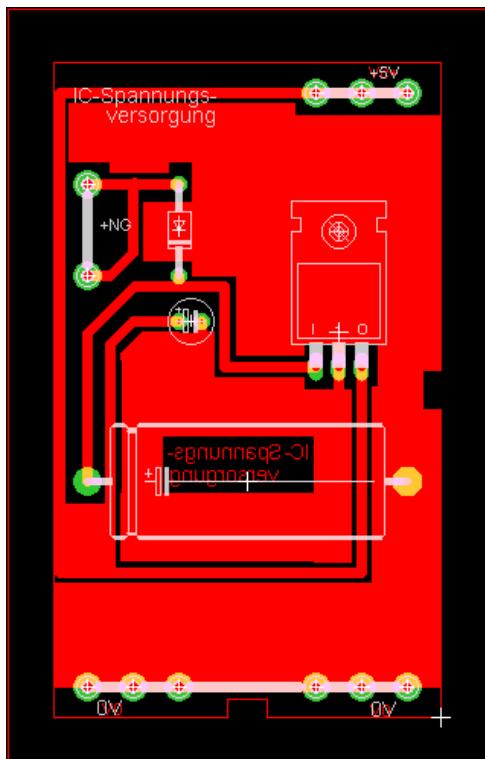


Abb. 53: Schaltbild

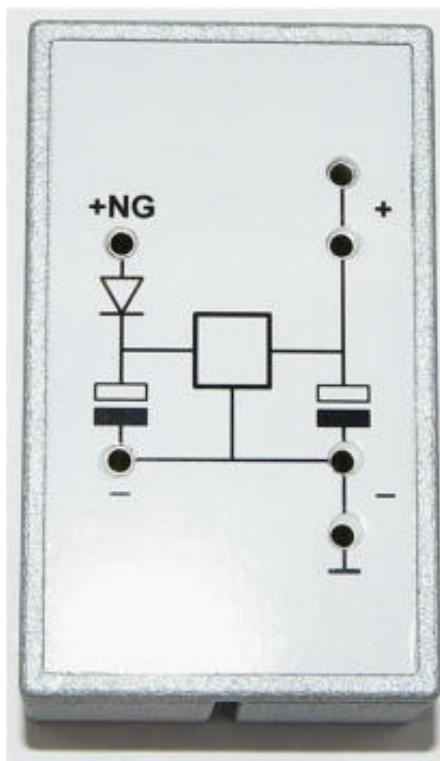
Abb. 54: Nachbau IC-Spannungsversorgung
([35732](#)) in meinem alternativen GehäuseAbb. 55: Nachbau IC-Spannungsversorgung
([35732](#)), ohne Deckel

Abb. 55 bietet dann einen Blick ins Innere.

Gegenüber dem Original musste ich die Bauteile verschieben und drehen. Daran sind die Bundhülsen schuld.

Wenn ich mir Abb. 56 anschau, denke ich, dass es besser gewesen wäre, wenn ich die Oberseite der Platine als Kühlplatte verwendet hätte. Mit etwas Wärmeleitpaste wird die Wärme des ICs besser abgeführt.

Unterhalb des Schaltplans (Abb. 52) befindet sich eine Tabelle, aus der die Minimal- und Maximalwerte ermittelt werden können, die am Eingang angeboten werden dürfen. Allerdings sollte man eine mögliche Welligkeit der Gleichspannung berücksichtigen. Beträgt die Welligkeit beispielsweise 1 V, müssen bei Verwendung des 7809 am Eingang 12 V angelegt werden.

Die Stabilisatoren der Serie 78xx gibt es in Versionen für 1 A, 1,5 A und 2 A. Ich persönlich finde eine 1-A-Version in der IC Spannungsversorgung mehr als ausreichend.



Abb. 56: Nachbau IC-Spannungsversorgung
([35732](#)) – Lötseite

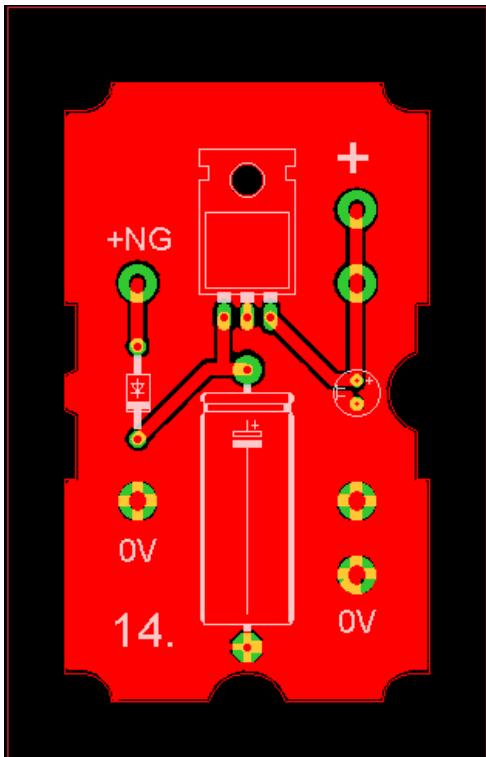


Abb. 57: Nachbau 1: Platinenentwurf
für das Original-Silberling-Gehäuse

Ich habe vier Alternativen für die IC-Spannungsversorgung entworfen:

Typ 1 (Abb. 57) kann direkt in das Originalgehäuse des Gleichrichters eingebaut werden.

Typ 2 (Abb. 58) ist für mein 4575-Gehäuse (Abb. 55).

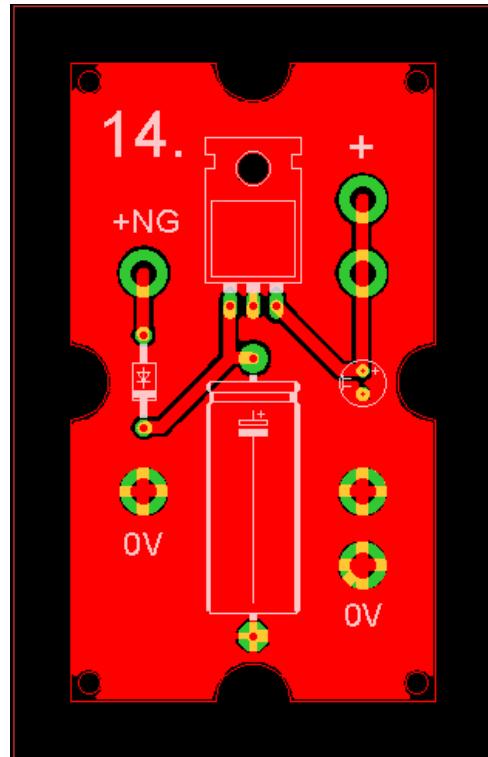


Abb. 58: Nachbau 2: Platinenentwurf
für mein alternatives Gehäuse

Typ 3 (Abb. 59) ist für die 60er-Kassette ([130961](#)).

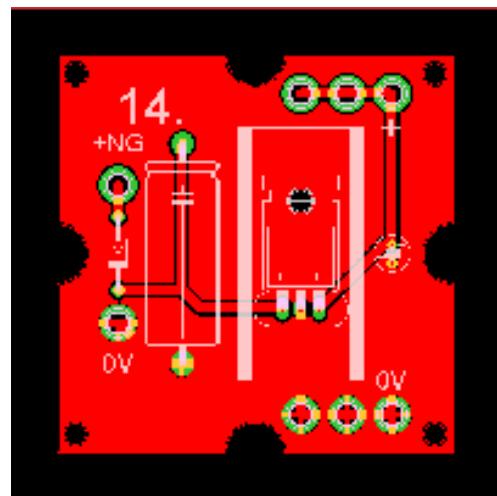


Abb. 59: Nachbau 3: Platinenentwurf für 60er-Kassette ([32076](#)) mit genügend Platz für einen Kühlkörper

Typ 4 (Abb. 60) ist für das Batteriegehäuse ([32263](#)).

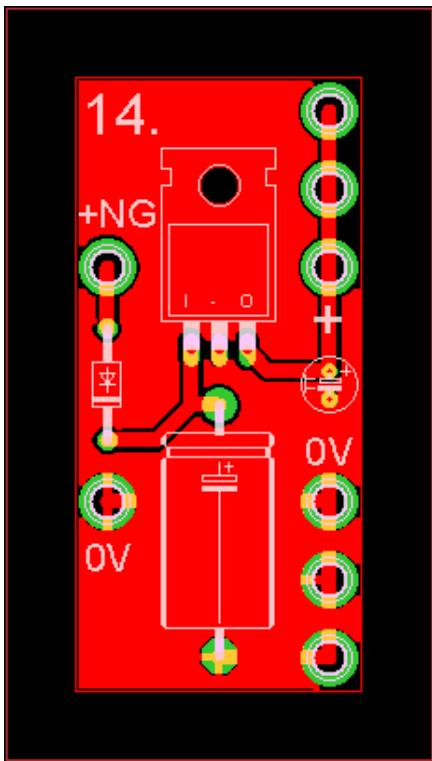


Abb. 60: Nachbau 4: Platinenentwurf für das Batteriegehäuse ([32263](#))

Es ist nicht sinnvoll, einen 7809 an die alten fischertechnik-Transformatoren anzuschließen: Diese bieten sehr wenig Leistung. Selbst bei einer Belastung von nur zwei Lampen sinkt die Ausgangsspannung des Stabilisators bereits unter 8 V, mit einer Welligkeit von fast 700 mV.

Wer noch einen Ringkerンtrafo von z. B. 12 V/120 VA (Abb. 61) herumliegen hat, ist damit besser bedient. In Kombination mit einem Brückengleichrichter oder mit vier separaten Dioden aufgebaut und einem großen 4700 μ F/35 V-Kondensator (Abb. 62) kann man davon ausgehen, dass man mit 2 · 5 A genügend Strom zur Verfügung hat. Allerdings muss der Stabilisator dann mit einem Kühlkörper ausgestattet werden, und einige Löcher im Deckel können auch nicht schaden.



Abb. 61: Ringkerンtrafo: 2 · 12 V, 2 · 5 A, 120 VA

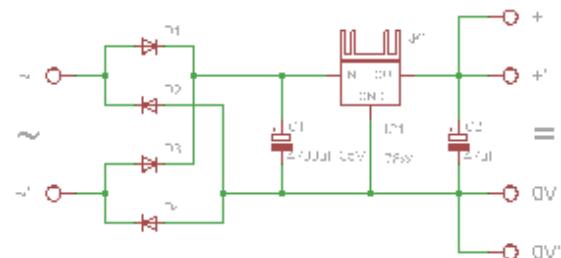


Abb. 62: Universelles 1-A-Netzteil

Das war's für dieses Mal. Im nächsten Teil geht es um den „Grundbaustein“ h4G.

Quellen

- [1] Peter Krijnen: *Silberlinge: Original oder Nachbau (Teil 1)*. [ft:pedia 1/2021](#), S. 80–93.
- [2] Peter Krijnen auf [Thingiverse](#)
- [3] AutoDesk Eagle [Download-Seite](#)
- [4] FrontDesigner 3.0 auf [Abacom](#)
- [5] 3D-Druck-Dienst [i.materialise](#)
- [6] Platinenfertiger [Eurocircuits](#)

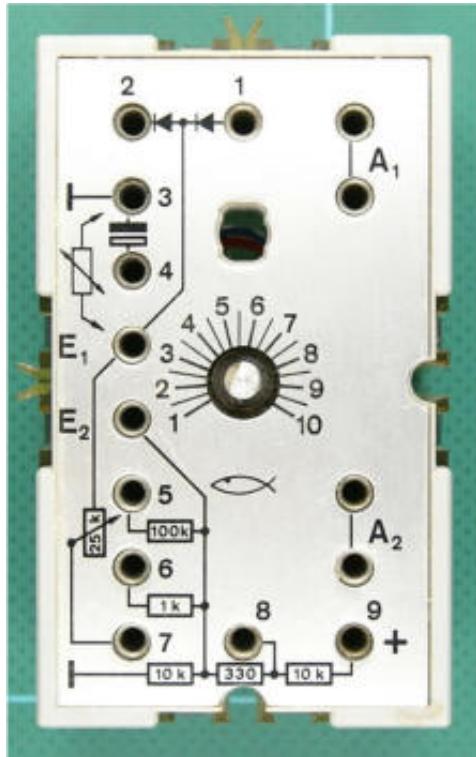
Elektronik

Silberlinge: Original oder Nachbau (Teil 3)

Peter Krijnen

Ist der Grundbaustein der Eckstein der Silberlinge? Der Mittelpunkt? Ich glaube schon. Außer als Verstärker (mit einem Eingang) kann er auch als Differenzverstärker (mit zwei Eingängen) verwendet werden, ebenso als Takt- oder Tongenerator. Er ist einstellbar und bietet einen normalen und einem invertierten Ausgang [1].

Es würde zu weit führen, hier eine vollständige Funktionsbeschreibung aufzunehmen. Hierzu verweise ich auf die Hobbybücher aus der Reihe „Experimente + Modelle“ [2]. Darin werden alle Möglichkeiten des Grundbausteins anhand von Versuchen und Modellen beschrieben.

Abb. 63: Grundbaustein [36391](#)

Ich selbst besitze zwei Versionen des Grundbausteins.

In Version 1 (Abb. 65) sehen wir die großen dicken braun-roten Widerstände der Typen 309/12 und 411/15.

In Version 2 (Abb. 67) sehen wir auch die gelblich gefärbten 207/12.



Abb. 64: Unterseite des Deckels

Hat das offene Potentiometer (Abb. 64) in Version 1 noch einen Wert von $50\text{ k}\Omega$, sind es in Version 2 nur noch $25\text{ k}\Omega$. Schaut man sich Version 1 (Abb. 66) und Version 2 (Abb. 68) unten an, gibt es Unterschiede.



Abb. 65: Komponentenanordnung
der ersten Version



Abb. 67: Komponentenanordnung
der zweiten Version



Abb. 66: Leiterplattenunterseite
der ersten Version



Abb. 68: Leiterplattenunterseite
der zweiten Version

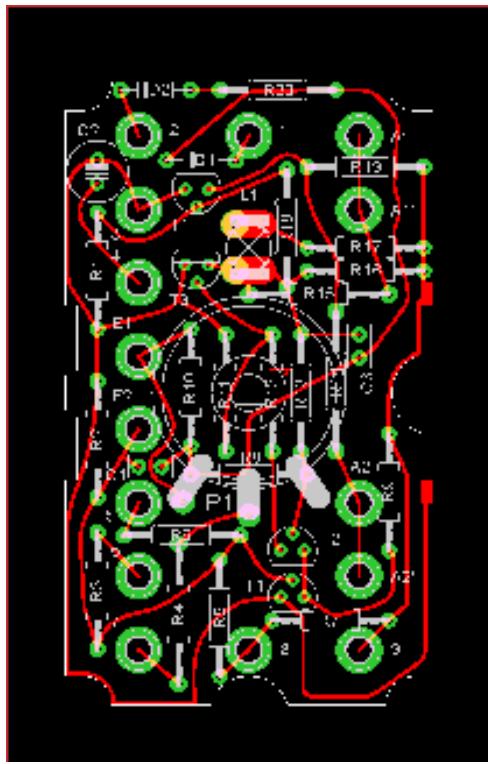
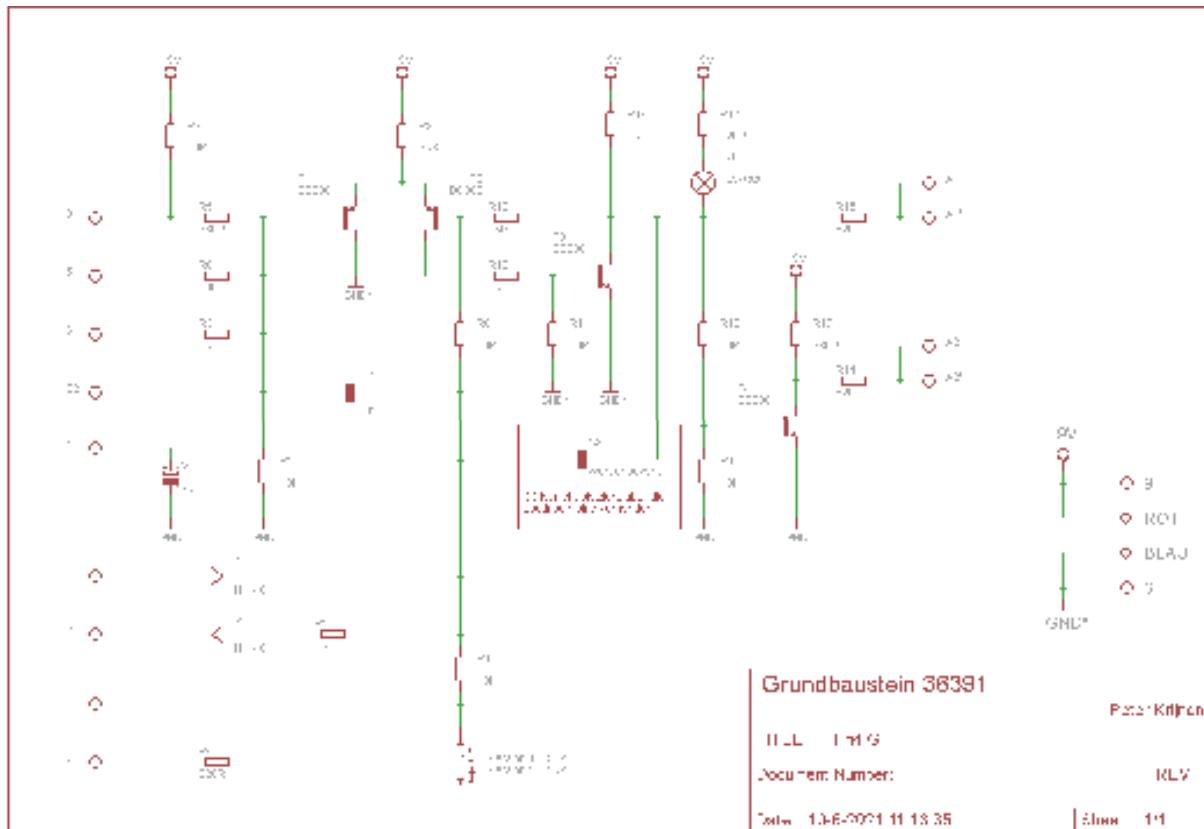


Abb. 69: Layout des Grundbausteins in der ersten Version

Auffällig bei beiden Versionen ist jedoch, dass noch vier Löcher übrig sind. Die beiden Löcher zwischen den Widerständen R6 ($100\text{ k}\Omega$) und R8 ($10\text{ k}\Omega$) sind für die Anschlussdrähte des Potentiometers. Dies ist im Layout auf Abb. 69 besser zu sehen. Wir sehen hier auch neben Widerstand R14 ($82\text{ }\Omega$) die anderen beiden Löcher.

Ich weiß nicht, was an dieser Stelle geplant war, aber da der Abstand zwischen den Löchern 2,5 mm beträgt, bin ich davon ausgegangen, dass auch hier ein kleiner Kondensator vorgesehen war. Es ist mir nicht klar, warum dieser später weggelassen wurde.

Um etwas Klarheit zu schaffen, habe ich an dieser Stelle einen 1nF -Kondensator platziert. Zuerst hatte ich gehofft, dass dieser Kondensator die Form des Ausgangssignals verbessern würde. Das war nicht der Fall. Das Einzige, was mir aufgefallen ist, war,



dass sich nach dem Einlöten des 1-nF-Kondensators die eingestellte Frequenz halbierte.

Das gesamte Schaltbild ist in Bild 70 zu sehen.

Die Basis

Wenn wir uns jedoch zuerst das zweite Schaltbild in Abb. 71 ansehen, sehen wir, dass fast alle Anschlüsse und die dazugehörigen Komponenten verschwunden sind. Dies ist das eigentliche Schaltbild, die Basis des Grundbausteins. Es werden nur die Eingänge E1 und E2 und die Ausgänge A1 und A2 benötigt (und natürlich die Stromanschlüsse 3 und 9).

Die Feedback-Widerstände an den Anschlüssen 5 und 6 sind intern auf der Platine mit E2 verbunden. Äußerlich muss man 5 oder 6 mit einem Draht mit A2 verbinden. Warum? Kann der Widerstand auch direkt

zwischen E2 und A2 angeschlossen werden? Gleichtes gilt auch für das Potentiometer an Anschluss 7. Dieser ist intern mit E1 verbunden und muss extern mit A2 oder 9 verbunden werden. Warum nicht ein separates Potentiometer zwischen E1 und A2 oder 9 anschließen? Und den Kondensator an Anschluss 4? Wird ein Kondensator mit einem anderen Wert benötigt, muss dieser auf jeden Fall extern zwischen 3 und E1 angeschlossen werden. Warum einen Kondensator ($47 \mu\text{F}$) mit eigenem Anschluss auf die Platine setzen?

Alle genannten Anschlüsse sind entscheidend für die Funktion des Grundbausteins. Die beiden Anschlüsse 1 und 2 sorgen jedoch dafür, dass der Grundbaustein nicht mehr funktioniert. Beide Anschlüsse dienen zum Sperren des Grundbausteins. Verbindet dazu „1“ mit „9“ oder „2“ mit „3“.

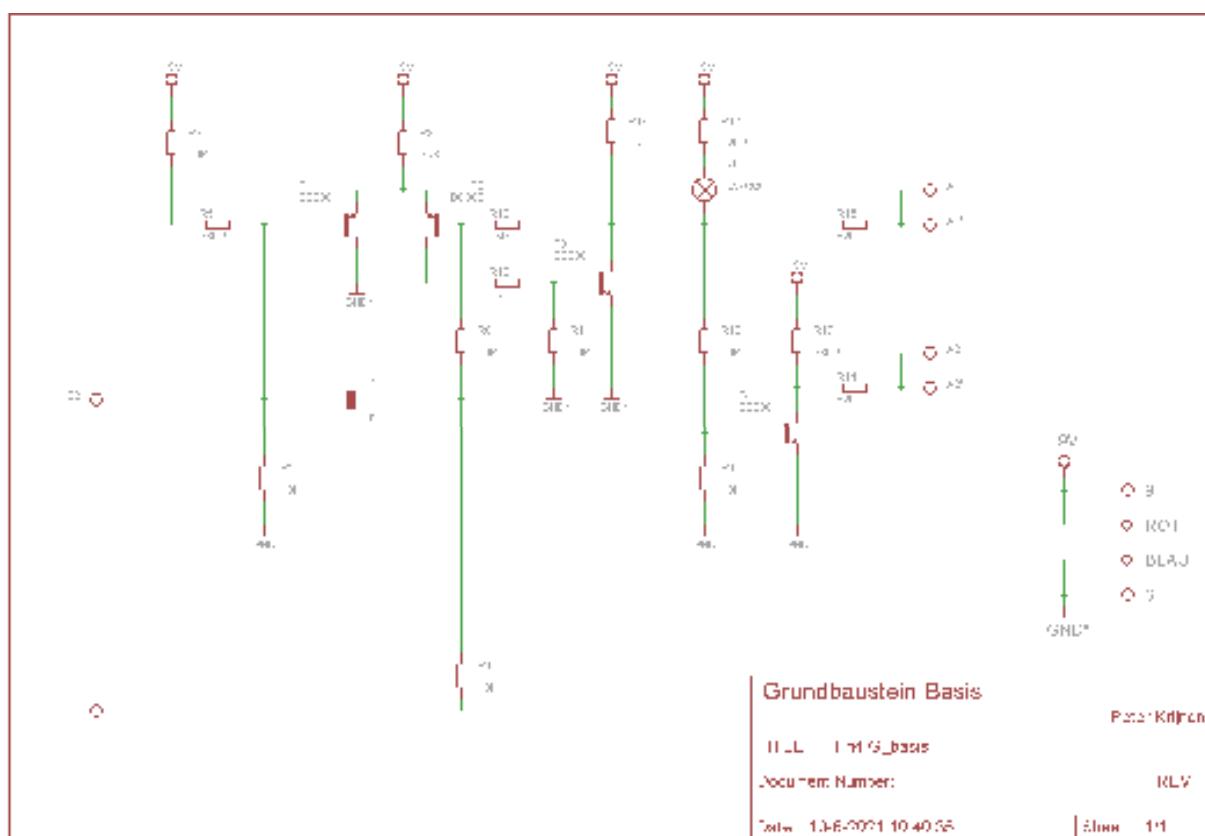


Abb. 71: Der eigentliche Schaltplan ohne die zusätzlichen Anschlüsse

Schließlich gibt es noch den Anschluss „8“. Der liegt fast in der Mitte eines Spannungssteilers, der mit zwei Widerständen von $10\text{ k}\Omega$ mit einem Widerstand von $330\text{ }\Omega$ dazwischen aufgebaut ist. Anschluss „8“ soll anstelle von Anschluss „9“ mit dem Potentiometer auf Anschluss „7“ verbunden werden.

Auf die obigen Fragen werde ich nie eine Antwort bekommen. Fest steht jedoch: Mit diesen zusätzlichen Anschlüssen ist der Grundbaustein zu einem universellen Baustein geworden. Wer mehr wissen möchte, den verweise ich auf die Buchreihe „Experimente + Modelle“ der Fischer-Werke [2].

Frontplatte

Die Frontplatte des Grundbausteins unterscheidet sich von der der anderen Silberlinge.

Vermutlich dachte man, dass dies eher den universellen Einsatz des Grundbausteins widerspiegeln würde. Betrachtet man Flip-Flop, Mono-Flop, And-Nand, Or-Nor und Dyn.-And, so haben sie alle ein DIN-Symbol auf ihrer Frontplatte, das die Funktionsweise des Moduls anzeigen.

Allerdings lässt sich der Grundbaustein durchaus mit DIN-Symbolen darstellen. Auf Seite 67 des Buches „hobby 4 Band 1“ aus der Reihe „Experimente + Modelle“ zeigt Bild 67.2, wie das aussieht.

Bei meinem ersten Versuch, den Grundbaustein nachzubauen, wollte ich das Potentiometer in der gleichen Position wie beim Mono-Flop haben. Allerdings war es dann nicht mehr möglich, die Widerstände auf der Frontplatte anzuzeigen. Da ich mich daran erinnerte, dass irgendwo in einem Buch ein Ersatzschaltbild des Grundbausteins stand, fing ich an, die hobby-Bücher durchzusehen. Dann habe ich eine Frontplatte mit den auf Seite 67 dargestellten Symbolen gezeichnet. Das Ergebnis ist in Abb. 72 zu sehen.

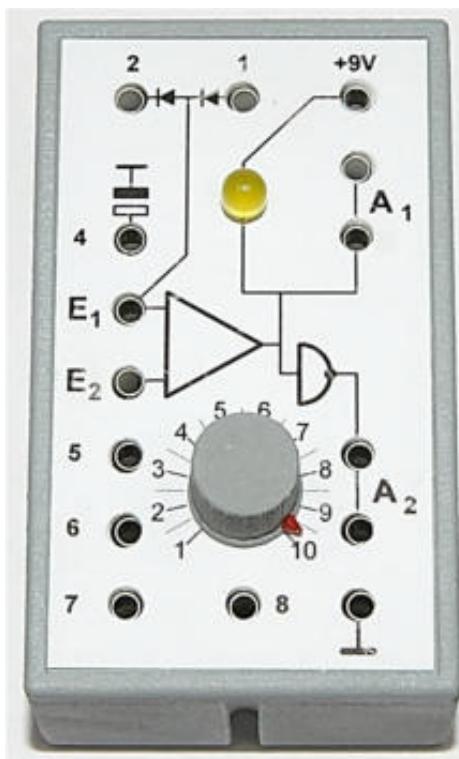


Abb. 72: Erste Version des Nachbaus:
Mit DIN-Symbolen

Am Ende bin ich mit diesem Aufbau nicht zufrieden. Warum nicht? Die Verbindungsdrähte zwischen den Anschlüssen 5 oder 6 und A2 sowie A2 oder 9 und 7 sind ziemlich im Weg, um an den Drehknopf des Potentiometers zu gelangen. Die Idee, Anschlussbuchse „3“ wegzulassen, stellte sich im Nachhinein als nicht die richtige Idee heraus. Auch die Platzierung von sechs Widerständen unter dem Potentiometer, das ich bei den Modulen aus dem „Elektronik“-Kasten 30253 gesehen hatte, erwies sich als keine so gute Idee: Die Steckachse des Potentiometers wird durch die Widerstände behindert. Um das zu beheben, musste ich das Potentiometer etwas höher einlöten als vorgesehen.

Also bleibe ich wieder beim ursprünglichen Aufbau.



Abb. 73: Alle zehn von mir gebaute Versionen

Nachbau

Auf Abb. 73 sehen wir alle zehn Versionen, die ich von 1983 bis heute hergestellt habe. Davon haben nur vier überlebt.

Nachbau 1 (Abb. 74)

Beim Auslöten der Platinen und späteren Einlöten stellte sich heraus, dass sich eine einzelne Lötinsel von der Platine gelöst hatte. Deshalb habe ich sofort eine neue Pla-

tine entworfen. Diese hat die gleichen Umrisse wie das Original und die Löcher haben einen Durchmesser von 2,2 mm. Bei diesem Design habe ich das Potentiometer ersetzt und das Lämpchen durch eine gelbe 5-mm-LED ersetzt. Wer jedoch das alte Potentiometer weiterverwenden will oder muss, braucht nur die Anschlussdrähte zu verlängern. Für die Transistoren habe ich die Kombination BC238 + BC308 oder BC547 + BC557 verwendet.

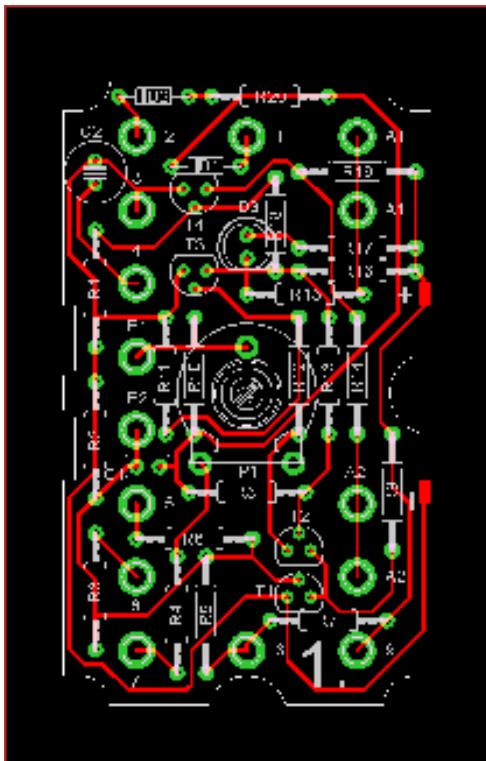


Abb. 74: Nachbau 1: Platinenentwurf für das Original-Silberling-Gehäuse

Nachbau 2 (Abb. 75)

Da ich selbst neue $45 \cdot 75 \cdot 30$ mm-Gehäuse entworfen und gedruckt hatte, konnte ich die Komponenten freier platzieren. Ich habe das $12 \cdot 8$ mm-Raster verwendet, in dem sich die Anschlussbuchsen im Original befinden. Auch die Position des Potentiometers und der gelben 5-mm-LED standen fest. Da ich für mein 4575-Gehäuse keine Zapfen verwende, sondern nur Nuten (Abb. 76) für die Verbindungen zu anderen Modulen, war weniger Platz auf der Platine, um die anderen Komponenten zu platzieren. Diese mussten nun alle in den Kreis der Anschlussbuchsen kommen. Das hat jedoch nicht ganz geklappt. Die Abmessungen der Platine betragen $40 \cdot 70$ mm.

Im Vergleich zum Original habe ich den „+“-Anschluss „9“ in die rechte obere Ecke verschoben. Dadurch mussten beide A1-Anschlüsse um eine Position nach unten verschoben werden. An der alten Stelle von „9“ habe ich jetzt einen zweiten Anschluss „0“ vorgesehen, um die Verbindung von

„+“ und „-“ zwischen den Modulen zu vereinfachen.



Abb. 75: Nachbau 2: Frontplatte für das 4575-Gehäuse



Abb. 76: Nachbau 2: Platine im 4575-Gehäuse

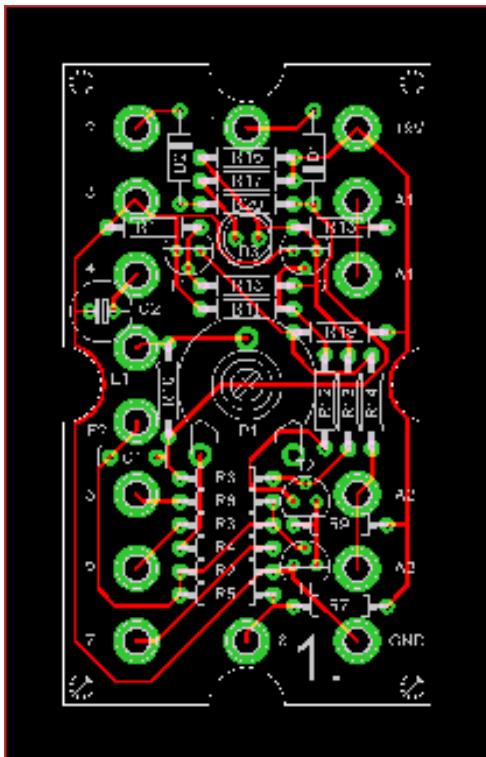


Abb. 77: Nachbau 2: Platinenentwurf für mein 4575-Gehäuse

Nachbau 3 (Abb. 78)



Abb. 78: Nachbau 3: Frontplatte für das 6060-Gehäuse

Nachbau 3 war ursprünglich für die graue 60er-Kassette [35359](#) gedacht. Natürlich kann man auch die gelbe Version [32076](#) oder die rote Version [130961](#) verwenden. Leider sind [35359](#) und [32076](#) „nml“ und

[130961](#) scheint „svr“ zu sein. Deshalb habe ich ein weiteres 60 · 60 mm-Gehäuse entworfen (Abb. 79), das ungefähr so aussieht wie das 4575. Allen gemeinsam ist, dass sie 28 mm hoch sind und der Deckel 2 mm hoch ist. Die Abmessungen der Platine betragen 56 · 56 mm und die Bundhülsen sind im ¼-Zoll-Raster: 6,35 · 6,35 mm. Alle Komponenten befinden sich nun innerhalb des Kreises der Anschlüsse.

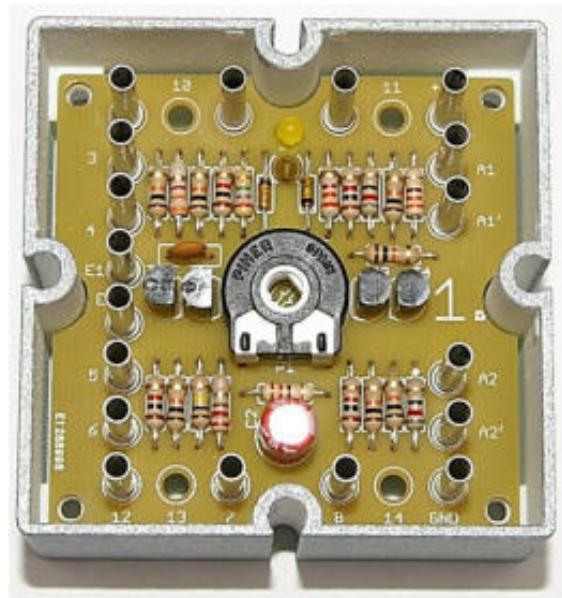


Abb. 79: Nachbau 3: Platinenentwurf für die 6060-Kassette

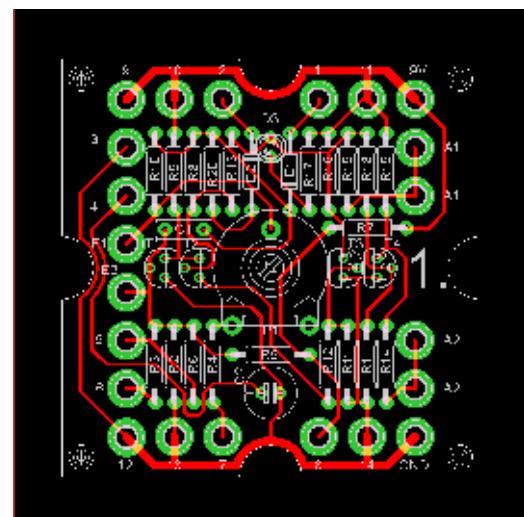


Abb. 80: Nachbau 3: Platinenentwurf für die 6060-Kassette ([32076](#))

Bei Nachbau 1 befinden sich noch 15 Hülser auf der Platine. In Nachbau 2 sind es wegen der zweiten „0“ 16. In Nachbau 3

sind es 22. In dieser Version habe ich jetzt vier „+“-Anschlüsse und auch vier „0“-Anschlüsse (tatsächlich fünf, denn Anschluss „3“ ist natürlich auch eine „0“).

Für die LED habe ich jetzt eine gelbe 3-mm-Version verwendet. Auch eine 5-mm-Version ist möglich. Man muss dann nur das Loch im Deckel auf 5 mm aufbohren.

Nachbau 4 (Abb. 81)

Eine noch kleinere Version des Silberlings sollte in das Batteriefach [32263](#) eingebaut werden. Das E-Tech- oder Elektronik-Modul diente als Vorbild. Dies bedeutete eine Halbierung der Platine-Oberfläche. Die Bundhülsen befinden sich am Rand der Platine (Abb. 82). Dadurch war es nicht mehr möglich, Bauteile und Leiterbahnen außerhalb des Kreises aus Anschläßen zu platzieren. Die 16 Bundhülsen liegen in einem $0,4 \cdot 0,28$ Inch-Raster: $10,16 \cdot 7,112$ mm.

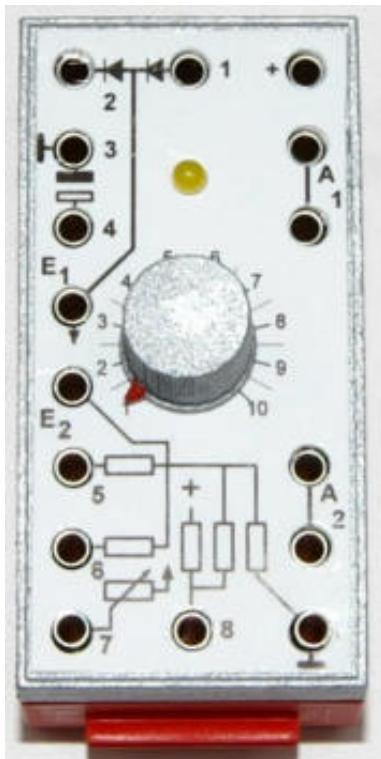


Abb. 81: Nachbau 4: Frontplatte für das Batteriegehäuse [32263](#)



Abb. 82: Nachbau 4: Eine sehr teure Platine im Batteriegehäuse [32263](#): Die Leiterbahnen haben eine Goldbeschichtung.

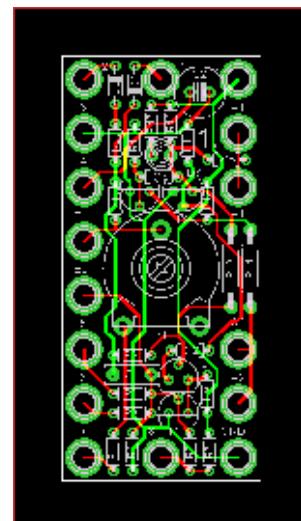


Abb. 83: Nachbau 4: Platinenentwurf für das Batteriegehäuse [32263](#)

Ich habe es geschafft, alle Komponenten auf der Platine zu platzieren. Für Nachbau 1, 2 und 3 konnte ich noch eine Kupferschicht verwenden, aber für Version 4 war es notwendig, auch Leiterbahnen auf die Oberseite der Platine zu verlegen (Abb. 83). Wie ich in Teil 1 [3] erwähnt habe, gibt es

in Version 4 kleine Widerstände vom Typ 204, deren Anschlussdrähte im Raster 0,2 Zoll (5,08 mm) gebogen sind. Leider konnte ich keine 204-Version für den $2.7\text{ M}\Omega$ finden. Deshalb musste ich dafür einen 207/10 nehmen.

Nachbau 5 (Abb. 84)

Darf es kleiner sein?

Ja, und das ist auch möglich. Von der Verwendung der Bundhülsen muss man sich jedoch verabschieden. Dies gilt auch für die verdrahteten Komponenten. Alles muss jetzt in SMD-Technik gebaut werden.

Es ist aber eine Kunst, mit SMD-Bauteilen zu arbeiten, besonders wenn sie sehr klein sind. In kommerziellen Produkten findet man hauptsächlich die kleinsten Versionen. Widerstände vom Typ 0603 sind nur $1,6 \cdot 0,8$ mm groß. Heute gibt es sogar noch viel kleinere. Der etwas größere Typ 0805 ist $2,0 \cdot 1,25$ mm groß. Um auch weniger Geübten den Nachbau einigermaßen zu ermöglichen, habe ich mich für den etwas größeren Typ 1206 entschieden. Mit Abmessungen von $3,2 \cdot 1,6$ mm sind diese noch überschaubar.

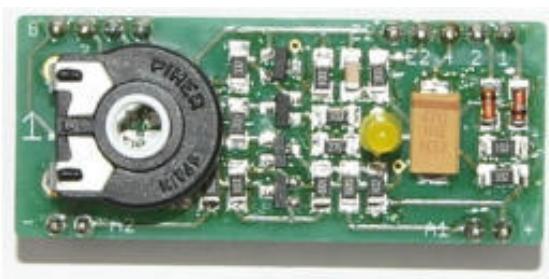


Abb. 84: Nachbau 5: für ein Breadboard: Grundbaustein als Shield in SMD Technik

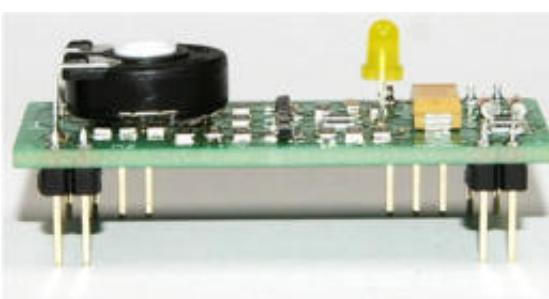


Abb. 85: Nachbau 5: von der Seite

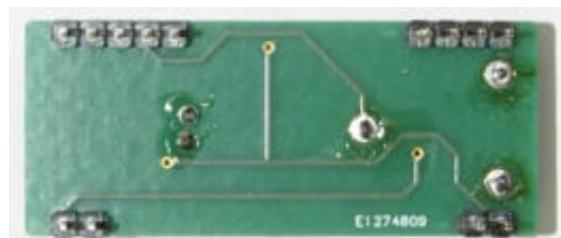


Abb. 86: Nachbau 5: Leiterplattenunterseite

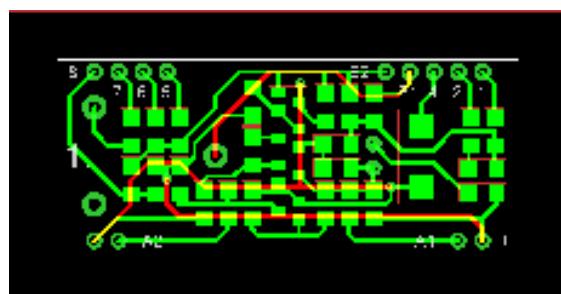


Abb. 87: Nachbau 5: Platinenentwurf des Shields

Da ich im fischertechnik-Sortiment kein passendes Gehäuse finden konnte, habe ich mich entschieden, auf Gehäuse zu verzichten. Für die Verbindungen habe ich angefangen, „Male Header“ zu verwenden.

Die Widerstände und der 1-nF-Kondensator sind vom Typ 1206. Als Transistoren eignen sich BC848 und BC858, beide im SOT23-Gehäuse. Für die LED habe ich allerdings noch eine 3-mm-Version verwendet. Da die SMD-Widerstände niedrig sind, konnten sechs Widerstände unter dem Potentiometer platziert werden.

Wie bei der Platine von Nachbau 4 war auch für Nachbau 5 eine doppelseitig durchmetallisierte Platine erforderlich (Abb. 87). Letztlich ist ein $20 \cdot 55$ mm-Steckmodul entstanden, das direkt auf ein Steckbrett gesteckt werden kann.

Darf es noch kleiner sein? Zweifellos. Allerdings darf man sich fragen, ob das noch sinnvoll ist. Das Ergebnis muss überschaubar, benutzerfreundlich und praktikabel sein.

Auf meiner Thingiverse-Seite [4] gibt es noch viele weitere Bilder zu sehen.

Das war's für Teil 3. In Teil 4 geht es um alle Relais.

Referenzen

- [1] Fischer-Werke: *fischertechnik h4 G Grundbaustein*. Auf docs.fischertechnikclub.nl.
- [2] Fischer-Werke: *fischertechnik hobby 4 Band 1*. Begleitbuch auf docs.fischertechnikclub.nl.
- [3] Peter Krijnen: *Silberlinge: Original oder Nachbau (Teil 1)*. [ft.pedia 1/2021](https://ft.pedia.1/2021), S. 80–93.
- [4] Peter Krijnen auf [Thingiverse](#).

Computing

Preisgünstige, autarke Stepper-Motorsteuerung

Franz Nachtsheim

Die beste Lösung für die präzise Drehung einer Achse sind Schrittmotoren. Sie werden allerdings von den fischertechnik-Controllern nicht unterstützt. Lösungen wie der ftPwrDrive-Controller [1] oder das Adafruit-Motor-Shield sind dagegen manchmal überdimensioniert. Gesucht war eine möglichst kompakte Einheit zur Ansteuerung eines einzelnen Stepper-Motors. Parameter sollten eingebbar und der aktuelle Status der Steuerung jederzeit sichtbar sein. Damit auch mehrere dieser Steuerungen in einem Modell verbaut werden können, sollte die Lösung außerdem kostengünstig sein.

Erste Versuche

Die erste Lösung für eine Schrittmotorsteuerung war schnell gebaut: Ein ftDuino [2] inklusive Motorsteuerung über den ftPwrDrive Controller [1]. Statusinformationen kann man auf dem integrierten Display des ftDuino darstellen. Die Eingabe zur Steuerung des Motors erfolgte über den seriellen Monitor der Arduino IDE (Abb. 1).

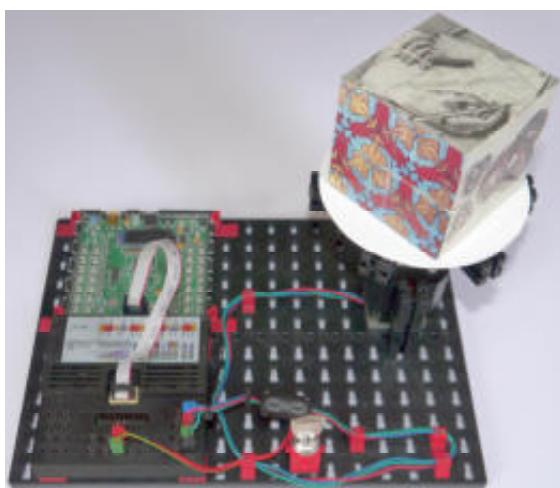


Abb. 1: Stepper-Motoransteuerung mit ftDuino und ftPwrDrive (Schlüsselschalter: Not-Aus)

Der Preis dieser Ansteuerelektronik liegt allerdings bei stolzen 140 €.¹ Das Ganze war aber für den Einsatzzweck ohnehin überdimensioniert: Der ftPwrDrive kann vier Steppermotoren und vier Servomotoren betreiben; benötigt wurde aber nur ein Steppermotor. Eine sehr hilfreiche Eigenschaft des ftPwrDrive ist, dass er Intelligenz mitbringt und seine Motoren direkt ansteuert. Nach der Übergabe eines Fahrbefehls an den ftPwrDrive beispielsweise

```
Drive.setRelDistance  
(FTPWRDRIVE_M1,1000);  
Drive.startMoving (FTPWRDrive_M1);
```

ist der ftDuino wieder frei und kann weiterarbeiten, während der Motor autark seine 1000 Schritte fährt.

Die zweite Lösung war ein Arduino Uno [3, 4] und ein Motor Shield von Adafruit² mit Darstellung der Statusinformationen auf einem OLED-Display (Abb. 2). Die Eingaben erfolgten mit einem Keypad. Dieses war aber nicht gut zu bedienen. Das Keypad musste beim Drücken der Knöpfe festgehalten oder alternativ die Bauplatte des gesamten Aufbaus fixiert werden. Außerdem belegt es viele Ressourcen in Form von acht

¹ Bezugsquelle: Gundermann Hard & Softwareberatung e.K., [Shop für ftDuino und Co.](#)

² [Adafruit Motor/Stepper/Servo Shield](#) for Arduino v2 KIT – v2.3.

Eingängen und war nicht immer ganz zuverlässig.³

Dieser Aufbau war schon etwas kompakter und günstiger, aber immer noch eine überdimensionierte Hardware. Wenn man originale Hardware einkauft und keine Clones aus China liegt man bei etwa 50 €.

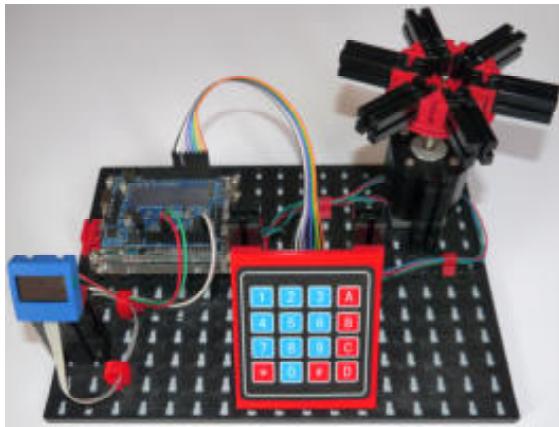


Abb. 2: Stepper-Motoransteuerung mit Arduino Uno und Adafruit Motor Shield v2.3

Kompakte Lösung

Um noch kompakter und preisgünstiger zu sein, wird die folgende Lösung vorgestellt. Die Intelligenz liefert ein Arduino Nano, die Anzeige erfolgt über ein 128x64-Pixel-OLED-Display⁴ und die Eingabe ist durch eine Infrarotfernbedienung⁵ ebenfalls vereinfacht. Diese ist trotz ihres geringen Preises zuverlässig und oft sehr praktisch. Bei Sichtkontakt zwischen Sensor und Fernbedienung ist ein Abstand von 5 m kein Problem. Da nur ein Motor angetrieben werden soll wird ein L9110 Motortreiber verwendet.⁶ Dieser ist sehr kompakt; dadurch kann das Gehäuse klein gehalten werden.

Damit ist der Aufbau klar. Bei Tests mit Arduino Nanos war der relativ neue und

preisgünstigere Nano Every gut geeignet. Selbst mit Originalteilen kostet diese Lösung nur noch etwa ca. 20 €. Der Nano Every hat etwas mehr Speicherplatz als die alte Nano-Version und ist deutlich robuster, was die Spannungsversorgung angeht, da er einen wesentlich leistungsstärkeren Spannungsregler besitzt. Ein weiterer Vorteil ist die größere Anzahl an Interrupts. Eine Übersicht der Unterschiede zeigt Tab. 1.

	Nano	Nano Every
Controller	ATmega328 (8-bit), 16 MHz	AT-mega4809 (8-bit), 20 MHz
Anschluss	Mini-USB	Micro-USB
Spannung	5 V	5 V
VIN min-max	6-9 V, empfohlen 7-9 V	7-21 V
Flash kByte	32	48
EEPROM Byte	1024	256
S-RAM kByte	2	6
Digitale Ein-/Ausgänge	14	12
davon PWM	7	5
analoge Ein-/Ausgänge	8/-	7/1
Interface	USB, I ² C, SPI	USB, I ² C, SPI

³ 4x4 Matrix Array Keypad, 8 Pin 16 Key Membrane Keyboard für Arduino. Danke an Peter Hambemehl, der kurzfristig nach einer Frage im Community-Forum eine einfache Möglichkeit konstruiert hat, um den Nummernblock in ein fischertechnik-Modell zu integrieren (Thingiverse Nr. [4336549](#)).

⁴ Es wurde ein 1,3" OLED Display SH1106 (128x64), I²C, gelb/blau verwendet.

⁵ Fernbedienungs-Modul: IR-Empfängermodul DIY Kit HX1838.

⁶ L9110 Dual Channel Stepper Motor Drive Board for Arduino.

	Nano	Nano Every
Interrupt Pins	2, 3	alle digitalen
Abmessungen	45 x 18 mm	45 x 18 mm
Gewicht (inkl. Header)	7 g	5 g

Tab. 1: Arduino Nano und Nano Every im Vergleich [3, 4]

Leider ist der neue Nano Every nicht 100% kompatibel, daher können Anpassungen notwendig sein: Die Prozessoren sind unterschiedlich, und die Kompatibilität wird durch eine Software-Zwischenschicht erreicht.⁷

Der Nano Every hat auf der Unterseite keine Bauteile, daher kann man ihn sehr gut direkt auflöten. Das ist auch der Grund, warum die Stiftleisten in der Regel nur beigelegt werden; man muss sie selber anlöten, was aber sehr einfach durchführbar ist. Von Nachteil ist, dass sich die Beschriftung auf der Unterseite befindet und somit praktisch nie sichtbar ist. Als Alternative habe ich mir eine kurze Übersicht in Folie laminiert, die immer griffbereit liegt. Sehr hilfreich ist ein Expansion Terminal Board für den Nano, auf das man ihn einfach aufstecken kann: Damit erhält man beschriftete Schraubklemmen, die sich gut nutzen lassen.⁸ Auch kann man den Nano jederzeit entfernen, ohne die Verdrahtung auflösen zu müssen.

Damit die Lösung mobil und robust ist, habe ich ein Gehäuse für einen 3D-Druck konstruiert, das alle Komponenten zusammenfasst.⁹

Erfreulicherweise lässt sich der Infrarotsensor durch transluzentes PLA¹⁰ nutzen, sodass das Gehäuse bis auf die Anschlusskabel des Motors geschlossen gehalten werden kann. Das Ganze wird direkt an einem USB-Port betrieben oder mit einem USB-Netzteil. Abb. 3 zeigt die fertig montierte Box.



Abb. 3: Bestückte Box

Rechts oben ist noch Platz für eine alternative Spannungsversorgung mittels eines Adapters von Hohlstecker auf Schraubklemme. Der Adapter wird durch das Gehäuse am Boden und durch die Durchführungsöffnung festgeklemmt. Das kann sinnvoll sein, wenn man für ein Bauteil eine höhere Versorgungsspannung anschließen möchte, die man dann auch für den V_{in}-Eingang des Nano Every verwenden kann, da er alles bis 21 Volt verträgt. Das runde Loch links (nur der abgenommene Deckel ist sichtbar) ist für Erweiterungen vorgesehen, um zusätzliche Kabel in das Gehäuse einführen zu können. Wenn man es nicht benötigt, wird es einfach durch den klemmenden Deckel verschlossen. In das PLA Gehäuse sind Gewindegürteln eingeschmolzen¹¹; damit lassen sich die Bauteile sicher montieren und der Deckel verschrauben.

⁷ Die zunächst sehr verunsichernde Fehlermeldung „Cannot locate „flash“ and boot“ memories in description“ kann man offenbar ignorieren.

⁸ Der HIMALAYA Basic Nano Expansion Terminal Board IO Shield Adapter for Arduino Nano ist sehr kompakt.

⁹ Gehäuse: Thingiverse Nr. [4831260](#).

¹⁰ Verwendet: PLA transluzent von Ultimaker.

¹¹ Einschmelzhülsen (beispielsweise von Ruthex) kann man, wenn sie am unteren Ende einen klei-

Für die Software gilt es, die passenden Bibliotheken zur verwendeten Hardware zu beschaffen. Folgende Bibliotheken habe ich verwendet:

- [IRremote.h](#) zur Nutzung der Infrarotfernbedienung
- [Stepper.h](#) zur Ansteuerung des Steppermotors am L9110
- [Adafruit SSD1306.h](#) zur Ansteuerung des OLED-Displays.

Da ich gelesen hatte, dass der Nano Every bei der Spannungsversorgung Vorteile hat, habe ich einen sehr günstigen Dreierpack gekauft und erlebte erst einmal eine Enttäuschung: Die von mir zunächst benutzte Display-Bibliothek für ASCII-Zeichen funktionierte nicht. Aber da der Flash-Speicher deutlich umfangreicher ist als beim Arduino Nano kann man ohne Probleme die größere Adafruit-Graphik-Bibliothek verwenden.

Die Spannungsversorgung erfolgt direkt über den Micro-USB Port. Die Verdrahtung zeigt Abb. 4.

Ein Hinweis zum Display: Ich hatte zunächst vier Bohrungen für alle vier Montagelöcher vorgesehen. Das Display ist aber im unteren Bereich sehr empfindlich, da dort das Verbindungskabel sitzt. Ich beschädigte zwei Displays, als ich auch die unteren Löcher zur Montage verwenden wollte. Mit den beiden oberen Löchern ist das Display ausreichend befestigt und man geht allen Risiken aus dem Weg.

Zur Programmierung: Das Auslesen der Tastencodes der Fernbedienung geht einfach, wenn man das Hilfsprogramm `IRrecvDemo.ino` benutzt. Man bekommt für jeden Tastendruck die übertragene Zahl angezeigt. Damit lassen sich auch die Fernbedienungen von Fernsehern etc. auslesen.

Noch ein Tipp zum Motortreiber und der Infrarotfernbedienung: Wenn der Motor für N-Pulse angesteuert wird, steht der Nano nicht für andere Aufgaben bereit. Man möchte aber in der Regel jederzeit schnell stoppen. Das kann man einfach mit einer While-Schleife bewerkstelligen:

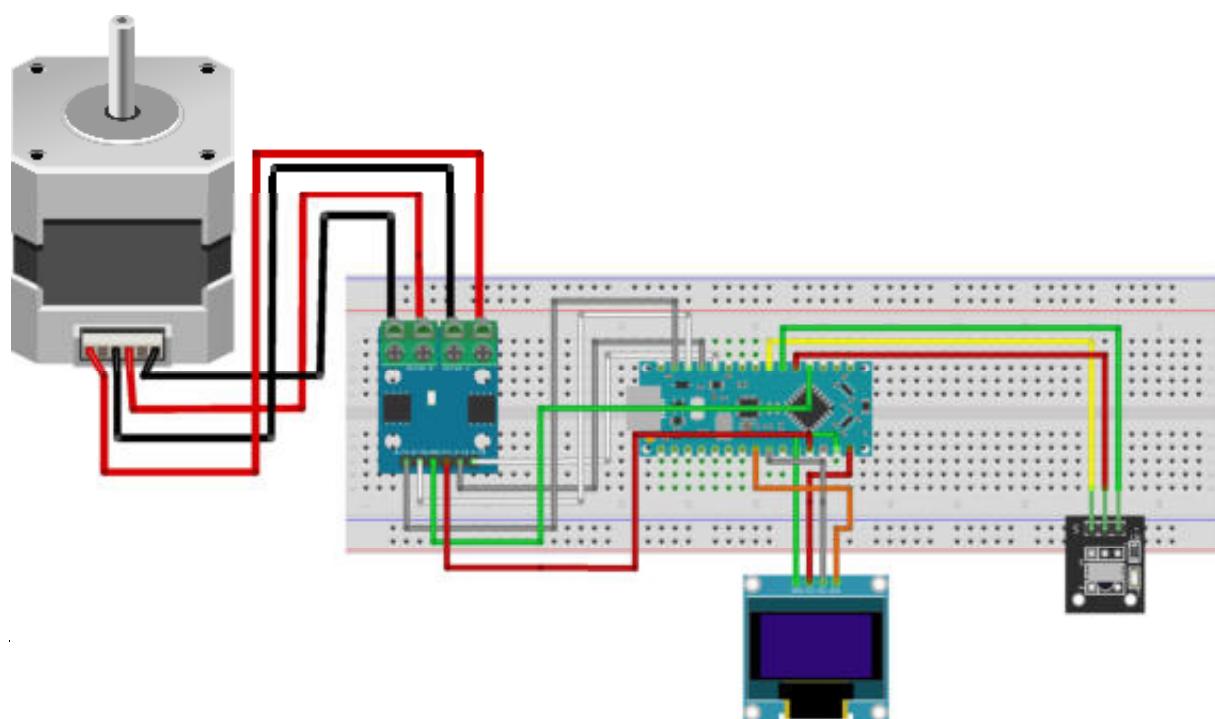


Abb. 4: Verdrahtungsplan (erstellt mit fritzing)

```

while (Mode != 'S') {
    irrecv.resume();
    // Der nächste Wert wird von der
    // Fernbedienung eingelesen.

    MyStepper.step(Nsteps);
    // Motor fährt N- Schritte

    if (irrecv.decode(&results)) {
        Mode = 'S';
    }
    // Bei irgendeinem Tastendruck
    // wird der
    // Operand „true“ und Mode wird
    // auf S (Stop) gesetzt
}

```

Der Motor läuft dann immer rund; die kurze Kontrolle der Fernbedienung ist nicht merklich. Durch Drücken einer beliebigen Taste auf der Fernbedienung stoppt man den Motor. Mit der Wahl der „Nsteps“ kann man festlegen, wie schnell der Stopp erfolgen soll.

Zum Schluss stelle ich noch zwei Anwendungen für die Stepper-Steuerung vor.

Anwendung 1: Lineartisch

Die erste Anwendung ist ein Lineartisch (siehe Abb. 5). Um das Spiel möglichst gering zu halten wird auch hier, wie schon öfters im Forum empfohlen, eine doppelte Schneckenmutter (unter dem Tisch) verwendet. Das zylindrische Bauteil aus Metall am Motor ist ein Wellenadapter¹², um von der 5-mm-Welle des Steppermotors auf die 4-mm-Welle von Fischertechnik zu adaptieren.

Vorne rechts ist ein Endlagentaster verbaut. Die weiße Platte senkrecht auf dem beweglichen Tisch ist zur Genauigkeitsmessung montiert. Dazu ist ein Infrarotsensor von Sharp [7] im gedruckten Gehäuse¹³ angebracht, der einen Messbereich zwischen 2 und 15 cm hat. Damit habe ich die Präzision der Positionierung vermessen. Der Sensor ist einfach zu handhaben; er verträgt alles von 3.3 bis 7 V und liefert die Entfernung als Analogwert. Um stabile Werte zu erhalten wird einfach über viele Messungen gemittelt.

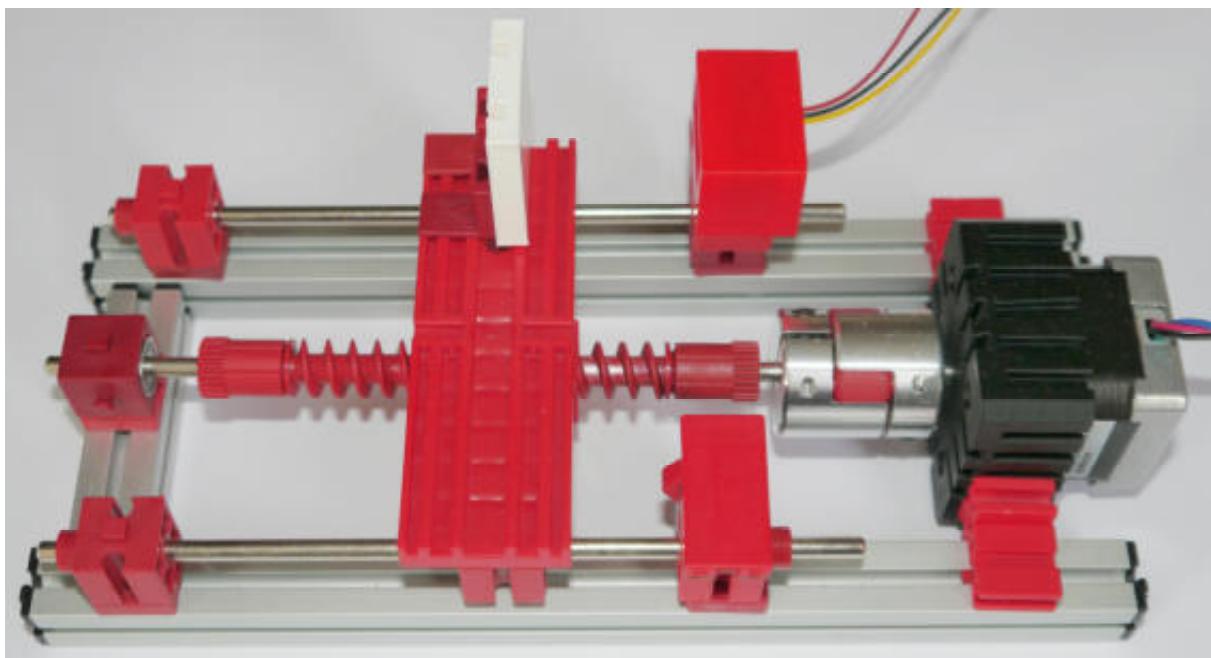


Abb. 5: Lineartisch

¹² Bezugsquelle: [Christians Technikshop](#).

¹³ Sharp-Gehäuse: Thingiverse Nr. [4831011](#).

Die Steigung der Schnecke liegt bei etwa 4,7 mm pro Umdrehung, d. h. bei 200 Schritten (Steps) in einer Umdrehung erhält man eine Auflösung von 0,0235 mm je Schritt. Wenn man nur zwischen zwei Positionen hin- und zurückfährt ist die Genauigkeit schon sehr gut. Ein Test über 10 Zyklen mit einer Hin- und Herfahrt über 1 cm erreicht die Ursprungsposition mit einer Abweichung von nur ca. 0,01 mm. Das sollte für viele Zwecke ausreichen.

Anwendung 2: Drehteller



Abb. 6: „Unterflur“-Drehteller

Den Drehteller in Abb. 6 kann man beispielsweise dazu benutzen, um 3D-Objekte reproduzierbar von allen Seiten zu fotografieren. Hier das besonders attraktive Objekt des Flötenmanns¹⁴. Der Drehteller wird mit einer Kette angetrieben; dadurch kann er sehr flach ausfallen. Das Antriebszahnrad und das eigentliche Drehtellerzahnrad sind selbst gedruckte Zahnräder.¹⁵ Wenn Antriebszahnrad und Drehtellerzahnrad identisch sind, erreicht man eine Auflösung von 1,8° mit gängigen Steppermotoren und die Reproduzierbarkeit ist gut. Bei zehn Runden über jeweils 36° Schritte liegt die Gesamtabweichung bei nur 2°. Eine größere Ungenauigkeit gibt es nur beim ersten Start bzw. bei Richtungswechsel, weil es da zu Schlupf kommt.

Um Stabilität zu erhalten sind Aluträger verbaut. Reflexe bei intensiver Beleuchtung

werden durch schwarzes Alu aus dem Open-Beam-Programm verhindert. Mit den Endstücken lassen sie sich sehr gut in das fischertechnik-System integrieren. Das Open-Beam-System bietet noch zusätzliche Montagemöglichkeiten.

Fazit

Die gezeigte Elektronikbox sollte für viele einfache Zwecke ausreichen. Die Bedienung über die Infrarotschnittstelle kann leicht an alle Anforderungen angepasst werden und ist auch einsetzbar, wenn das Modul nicht direkt zugänglich ist. Da die Infrarotschnittstelle nicht nur empfangen, sondern auch senden kann, kann man auch mit weiteren Modulen kommunizieren.

Quellen

- [1] Dirk Wölffel, Christian Bergschneider, Stefan Fuss, Björn Gundermann, Christian Lauff: *Der ftPwrDrive-Controller für Schrittmotoren und Servos – Teil 1.* [ft:pedia 4/2018](#), S. 67–70.
- [2] Till Harbaum: *ftDuino – Open-Source trifft Konstruktions-Baukasten.* [ft:pedia 1/2018](#), S. 85–91.
- [3] David Holtz: *Alternative Controller (1): Der Arduino.* [ft:pedia 2/2016](#), S. 56–59.
- [4] [Arduino Home Page](#).
- [5] Dirk Fox: *I²C mit dem TX(T) – Teil 18: Keypads und GPIOPort-Erweiterung.* [ft:pedia 2/2019](#), S. 46–51.
- [6] Wikipedia: [Liste von Arduino-Boards](#).
- [7] Sharp: [GP2Y0A51SK0F](#). Distance Measuring Sensor Unit. Measuring Distance: 2 to 15 cm. Analog output type. Sheet No.: OP13007EN.

¹⁴ Aus dem Triptychon „Das jüngste Gericht“ von Hieronymus Bosch (ca. 1450-1516).

¹⁵ Zahnräder für den Schrittmotor: Thingiverse Nr. [4889074](#).

Computing

ftSwarm – Das Prinzip Heinzelmännchen

Christian Bergschneider, Stefan Fuss

Bei der TX(T)-Controller-Serie von fischertechnik erscheint dieser Tage mit dem „TXT 4.0“ die nächste Generation. ftDuino [1] und ftPwrDrive [2] ergänzen die klassischen Controller mit ihren eigenen Anwendungsfällen. So unterschiedlich die Controller auch sind, ihre Grundflächen sind alle gleich. Der ftSwarm ergänzt das „90 · 90 mm-Axiom“ um das Prinzip Heinzelmännchen und ermöglicht so völlig neue Bauansätze.

Die Sage der Heinzelmännchen: wer kennt sie nicht? Kleine, fleißige Hausgeister, die nachts die Arbeit für die Großen erledigen. Man findet sie nicht nur in Köln, in England heißen sie Brownies, in Skandinavien helfen die Nisse in Haus und Hof. Den schwedischen Tomte hat Astrid Lindgren [3] über die Landesgrenzen bekannt gemacht.

In Terry Pratchetts Scheibenwelt-Märchen „Kleine freie Männer“ [4] nennen die Heinzelmännchen sich selbst „Wir-sind-die-Größten“. Sie sind das kleinste Volk der Scheibenwelt – aber mit Abstand das stärkste und effektivste. Obwohl nur eine Handspanne groß, ist jeder von ihnen ein unerschrockener Kämpfer. Zusammen sind sie unbesiegbar, jeder für sich und dennoch gemeinsam greifen sie ihre Feinde an. Gegen hunderte Gegner hat auch der größte Held der Scheibenwelt keine Chance.

Das „90 · 90 mm-Axiom“ führt bei fischertechnik-Modellen zu zwei grundlegenden Baugesetzen: Mindestgröße und Kabelbaum. Will man autonom fahrende Roboter bauen, ist man automatisch auf die Größenordnung des Fahrroboters aus dem Discovery Set festgelegt. Bei coolen Fahrzeugen wird deshalb der Controller schnell durch die Elektronik aus dem Control-Set ersetzt und wieder manuell gesteuert.

In größeren Modellen wird der Controller an einer zentralen Stelle mit ausreichend Platz verbaut; von dort aus wird jeder Sensor und Aktor verkabelt. Das Kabelbaum-Bauen hat sich so zu einer eigenen Disziplin entwickelt. Der Flachbandkabeladapter [5] erfreut sich deshalb einer großen Beliebtheit und scheint in keinem gut sortierten fischertechnik-Haushalt mehr zu fehlen.

Kleine, freie Controller

Der ftSwarm (Abb. 1) basiert auf dem Prinzip Heinzelmännchen: Kooperieren mehrere „Kleine“, so sind sie effektiver als ein „Großer“.



Abb. 1: Prototyp des ftSwarm – klein aber oho!

Mit nur der Hälfte an Ein- und Ausgängen ausgestattet ist der Controller mit dem Volumen von einem Achtel TXT ein echter Winzling.

Während man für die großen Controller noch nach einem geeigneten zentralen Platz im Modell sucht, ist der ftSwarm bereits am Plätzchen (Abb. 2).



Abb. 2: Der ftSwarm findet immer seinen Platz

Das Plätzchen ist nahe bei Motoren, Tastern und Lichtschranken, sodass nur kurze Kabelführungen benötigt werden (Abb. 3). Der Kabelbaum reduziert sich auf die Spannungsversorgung des ftSwarm.



Abb. 3: einfache Kabelführung

Werden an mehreren Stellen im Modell Aktoren und Sensoren benötigt, so werden ein-

fach mehrere ftSwarm-Controller an unterschiedlichen Stellen verbaut. Jeder Controller arbeitet autonom für sich und übernimmt seinen Teil der Steueraufgabe; miteinander kooperieren die Controller über Bluetooth.

Durch die geringe Größe lassen sich nun auch sehr kleine autonome Roboter realisieren. Als kooperierender Schwarm wird aus einer Gruppe von autonomen fischertechnik-Aufräumrobotern schnell ein aufgeräumtes Kinderzimmer.¹

Die Kelda

Die „kleinen freien Männer“ in *Terry Pratchett's* Roman sind ein chaotischer Haufen, da jeder für sich agiert. Den Überblick in diesem Volk hat das einzige weibliche Wesen des Clans, die Kelda. Sie sorgt für die Basis der Zusammenarbeit: Was die Kelda sagt, ist Gesetz und wird befolgt.

Mit dem ftSwarm wird aus dem zentralen Controller-Prinzip von fischertechnik ein verteiltes System. Wie im Roman ist es sehr anspruchsvoll, die ganzen kleinen, freien Controller programmiertechnisch unter einen Hut zu bekommen.

Hierfür hat der ftSwarm mit dem Kelda-Ansatz eine einfache Lösung: Da alle Controller über einen leistungsfähigen ESP32-Wrover-Prozessor verfügen, wird ein beliebiger Controller im Modell zur Kelda. Sie übernimmt die Leitungsaufgabe und sendet an die anderen Controller über BLE Steuerkommandos oder fragt den Status von Sensoren ab.

Auf der Kelda läuft zusätzlich zur normalen Firmware ein klassisches Programm in C oder MicroPython. Über eine API kann jede Funktion auf einem beliebigen Controller ausgeführt werden.

¹ Der Name *ftSwarm* kommt von der ursprünglichen Idee, viele preiswerte, kleine Roboter zu bauen. Das im Laufe der Entwicklung hinzugekommene Heinzelmann-Prinzip eröffnet sehr

viel mehr Möglichkeiten, als nur Schwärme zu bauen; mit *ftMite*, *ftNano* oder *ftDwarf* konnten wir uns aber nicht anfreunden.

Im Beispielprogramm (Abb. 4) werden ein Motor- und ein Taster-Element instanziert. Dabei wird die UUID des Controllers angegeben, auf dem der zugehörige Anschluss liegt. Anhand der UUID spricht die Firmware nun das richtige Gerät an, sei es lokal oder remote auf einem anderen Controller.

```
#include "kelda.h"

Motor Mx( UID_FTSWARM_XYTable, 1 );
Switch Tx( UID_FTSWARM_Joystick, 1 );

Tx.waitFor( PRESSED );
Mx.run(100);
Tx.waitFor( RELEASED );
Mx.stop();
```

Abb. 4: Die Firmware kümmert sich um die Probleme verteilter Systeme

ftSwarm

Obwohl der ftSwarm nur $45 \cdot 37,5 \cdot 18$ mm groß ist, stecken in ihm viele Funktionen:

- ESP32-Wrover-B Prozessor.
- Bluetooth, BLE & WiFi
- Zwei Aus- und vier Eingänge
- Zwei RGB- bzw. Neopixel-LEDs
- Servoanschluss
- SD-Kartenslot
- Optionaler Gyro

Um diese Größe zu erreichen, sind klassische fischertechnik- bzw. Märklin-Stecker viel zu groß. Am ftSwarm sind deshalb JST-PH-Stecker verbaut (Abb. 5).

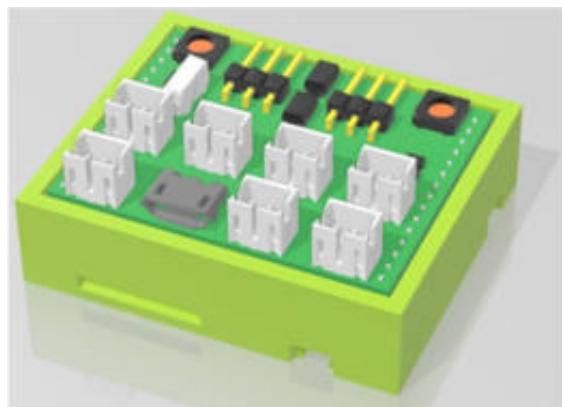


Abb. 5: Der ftSwarm ersetzt die Bundhülse durch kleine JST-PH-Stecker

Zum Anschluss der fischertechnik-Motoren und -Taster werden kurze Kabelstücke verwendet, „Pigtails“ genannt. Diese Pigtails werden in der Maker-Szene von einigen Plattformen genutzt.

An einem Ende des Pigtails ist bereits eine JST-PH-Buchse gecrimpt, an das offene Ende werden die normalen fischertechnik-Stecker angeschlossen.

Da der Controller in der Nähe von Aktoren und Sensoren eingebaut wird, werden nur wenige verschiedene Kabellängen benötigt.

ftSwarmControl

Zusätzlich zum ftSwarm gibt es den ftSwarmControl. Er kann im Modell fest als Steuerpult oder als Fernbedienung eingesetzt werden (Abb. 6).

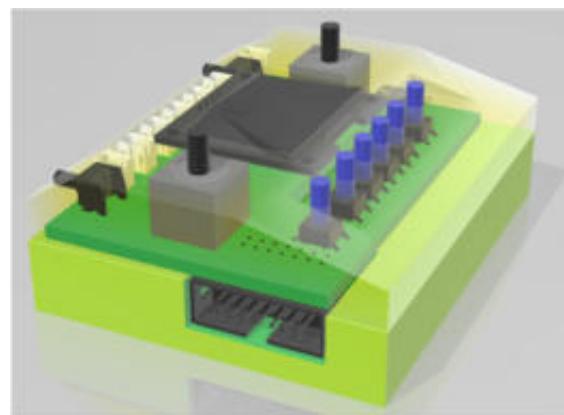


Abb. 6: Das Steuerpult ftSwarmControl

Die Größe des ftSwarmControl ist mit $85 \cdot 70 \cdot 37$ mm auch deutlich kleiner als das originale Control Set von fischertechnik. Im ftSwarmControl steckt jedoch ein fast vollwertiger ftSwarm. Zusätzliche Funktionen machen aus ihm ein perfektes Steuerpult:

- OLED Display
- Zwei Joysticks
- Sechs frei verwendbare Taster
- TX/ftDuino- und TXT-kompatible I²C-Schnittstelle
- Stromversorgung über 9 V Netzteil oder 9 V Blockbatterie

Wie beim ftSwarm lassen sich beliebig viele ftSwarmControl im Modell einsetzen.

	ftSwarm	ftSwarmControl
Prozessor	Esp32-Wrover-B mit 4MB Flash und 8 MB RAM	
Wireless	Bluetooth, BLE, wifi	
Motorausgänge	2 x DC Motor bis 0.5 A	2 x DC Motor bis 0.5 A
Eingänge	4 x analog	4 x analog
SD-Karte	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Display	<input type="checkbox"/>	0.96“ OLED gelb/blau
RGB- LEDs	2 x OnBoard, Erweiterungsstecker	<input type="checkbox"/>
Servoanschluß	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Joystick	<input type="checkbox"/>	2
frei nutzbare Taster	<input type="checkbox"/>	6
Reset-Taster	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
USB-Schnittstelle	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Stromversorgung	Extern 9V	Extern 9V und 9V Blockbatterie
I ² C-Interface	für MPU6040 optimiert	3.3V & 5V Logik
Gyro MPU6040	optional	optional

Abb. 7: Vergleich der technischen Daten von ftSwarm und ftSwarmControl

Er kann als Kelda verwendet werden, seine Funktionen lassen sich aber auch von jedem anderen Controller im Modell aufrufen.

Teamplayer

Durch die I²C-Schnittstelle mit beiden Logikpegeln kann der ftSwarmControl mit den kabelgebundenen Controllern TX und ftDuino kommunizieren. Über Bluetooth und Wi-Fi können beide ftSwarm-Varianten darüber hinaus mit TXT, TX-PI und ftDuino32 zusammenarbeiten.

Programmierung in C

Die Programmierung in C findet wie beim ftDuino über die Arduino-IDE² statt [6]. Für die Nutzung der Funktionen des ftSwarm steht eine umfangreiche Bibliothek zur Verfügung.

Beim Hochladen des Programms am Kelda-Controller wird der Inhalt des Flash-Speichers durch das eigene Programm ersetzt. Mit dem Neustart des Controllers startet dieses Programm automatisch.

Die normalen Controller erhalten eine Standardfirmware, sodass immer nur der Kelda-Controller geflashed wird.

Programmierung in MicroPython

MicroPython³ ist in der fischertechnik-Welt noch recht neu. Die Sprache ist aber einfach zu erlernen und bietet viele Vorteile.⁴ Die langsamere Programmausführung gleicht der leistungsfähige ESP32-Prozessor durch seine hohe Taktrate wieder aus.

Programmiert wird am PC am einfachsten in der Community-Version von PyCharm.⁵

² Download der Arduino-IDE:
<https://www.arduino.cc/en/software>

³ Homepage des MicroPython-Projekts:
<https://www.micropython.org/>

⁴ Python kennt keine Pointer und stellt bereits im normalen Sprachumfang komplexe Datentypen zur Verfügung. Die in C aus den Pointern häufig

entstehenden Speicherschutzverletzungen können in Python nicht auftreten; die Fehlerhäufigkeit wird geringer und die Fehlersuche erleichtert.

⁵ Download von PyCharm:
<https://www.jetbrains.com/de-de/pycharm/>

Der ftSwarm wird mit vorinstalliertem MicroPython-Interpreter ausgeliefert und startet automatisch die ftSwarm-Firmware.

Im Gegensatz zur C-Variante werden auf den Kelda-Controller die selbst geschriebenen Python-Programme zusätzlich hochgeladen; die Firmware wird nicht ersetzt.

Über das Display des ftSwarmControl kann nun wie am TXT zwischen verschiedenen Programmen umgeschaltet werden.

Der ftSwarm kann ebenfalls die Kelda-Funktion übernehmen. Da er aber über keine Bedienelemente verfügt, wird beim Upload der Python-Programme festgelegt, welches Programm beim Neustart des Controllers automatisch gestartet wird.

Verfügbarkeit

Im Laufe des Sommers werden die ersten Geräte verfügbar sein. Die erste Firmware wird die Grundfunktionen des ftSwarm unterstützen, die Kommunikation mit den „Großen“ ist für Release 2 geplant. Ab sofort sind beide Typen in Björns Shop⁶ vorbestellbar.

Quellen

- [1] Till Harbaum: *ftDuino – Open-Source trifft Konstruktions-Baukasten*. [ft:pedia 1/2018](#), S. 85–91.
- [2] Dirk Wölffel, Christian Bergschneider, Stefan Fuss, Björn Gundermann, Christian Lauff: *Der ftPwrDrive-Controller für Schrittmotoren und Servos – Teil 1*. [ft:pedia 4/2018](#), S. 67–70.
- [3] Astrid Lindgren: *Tomte Tummetott*. Oetinger Verlag, 1960.
- [4] Terry Pratchett: *Kleine Freie Männer, Ein Märchen von der Scheibenwelt*. Goldmann Verlag, 2003.
- [5] Stefan Fuss, Christian Bergschneider: *Verkabelung (Teil 2)*. [ft:pedia 4/2019](#), S. 19–21.
- [6] David Holtz: *Alternative Controller (1): Der Arduino*. [ft:pedia 2/2016](#), S. 56–59.

⁶ <https://www.gundermann.org>

Computing

TXT 4.0 Controller – ein Blick unter die Haube

Till Harbaum

In wenigen Wochen läutet fischertechnik die nächste Runde seiner Robotics Controller ein. Mit dem TXT 4.0 tritt nach acht Jahren ein Nachfolger in die Fußstapfen des 2013 auf den Markt gekommenen TXT. Im Schlepptau hat der TXT 4.0 die Software ROBO Pro Coding, die – anders als der Name vermuten lässt – mit der langen ROBO Pro-Tradition bricht. Grund genug, das bisher verbreitete Wissen um die neue Plattform zusammenzutragen. Hier zunächst der Blick auf die Hardware.

Als der TXT 2013 auf den Markt kam war sein Herz auf Basis eines 600-Mhz-Single-Core-ARM-Prozessors mit 256 MB RAM zeitgemäß und lag z. B. mit dem zur gleichen Zeit auf den Markt gebrachten ersten Raspberry Pi gleichauf. Als Software kam ROBO Pro zum Einsatz, wie damals gängig eine reine Windows-Lösung, die speziell für fischertechnik entwickelt wurde und in der Historie von Lucky Logic und Co. eine grafische Programmierung durch Flussdiagramme erlaubte [1].

Mit dem TXT wagte fischertechnik sich das erste Mal aus der bis dahin üblichen Close-Source-Ecke und bediente sich mit dem Linux des TXT bei quelloffener und frei kopierbarer Software.

Der TXT 4.0

Mitte 2021 wird der TXT 4.0 erscheinen. Optisch ist der Neuling leicht als fischertechnik-TXT zu erkennen: Nahezu alle Anschlüsse sind in gleicher Form an gleicher Stelle vorhanden und auch der zentrale Bildschirm wurde weitgehend übernommen. Das ganze Design ist flacher und sieht durch die glatte und glänzende Oberfläche wesentlich edler aus als der alte TXT (Abb. 1). Vor allem die Zwergstecker-Anschlüsse für die Taster, Motoren und anderen Sensoren und Aktoren scheinen weitgehend unverändert.



Abb. 1: TXT 4.0-Controller

Unter der Haube sind die Unterschiede schon etwas größer. Erste Hinweise zu seinem Innenleben gibt fischertechnik auf Seite 27 im aktuellen Katalog (Abb. 2). fischertechnik beschreibt die CPU im TXT 4.0 als „Arm Dual Cortex-A7 650MHz + Cortex M4“, der 512 MB RAM zur Seite stehen. Gegenüber dem TXT ist das eine recht exakte Verdoppelung der Rechenleistung und ein deutlicher Fortschritt.

Berücksichtigt man die acht Jahre, die seit dem Erscheinen des alten TXT vergangen sind, sieht der TXT 4.0 allerdings nicht mehr ganz so glanzvoll aus, wie der neuere Vergleich mit dem Raspberry Pi zeigt.

TXT 4.0 CONTROLLER

Der TXT 4.0 Controller bietet mit seinem Speicherplatz von 512 MB RAM und 4 GB eMMC, den drei Servo-Ausgängen sowie ein kapazitives Touch-Display, das Wischgesten unterstützt, jede Menge neue Features. Das verbesserte WLAN und Bluetooth Modul bietet für zahlreiche Anwendungen die passende, kabellose Schnittstelle. Eine weitere Schnittstelle ist der USB-Host-Anschluss, an den z. B. die fischertechnik USB-Kamera oder USB-Sticks angeschlossen werden können. An einen Controller können bis zu acht zusätzliche Controller als Erweiterungen angeschlossen werden. Durch das schmale Gehäuse lässt sich der Controller perfekt in die Modelle integrieren. Um immer auf dem neusten Stand zu sein, werden Firmware-Updates automatisch über die Cloud heruntergeladen, wobei eigene Programme erhalten bleiben. Mit der Software ROBO Pro Coding kann sowohl graphisch als auch mit Python programmiert werden! Sie ist Betriebssystem unabhängig und kann auch auf mobilen Endgeräten eingesetzt werden. Weitere Features des Controllers sind:

- › Prozessor: Arm® dual Cortex®-A7 650 MHz + Cortex®-M4
- › Speicherkapazität: 512 MB DDR3 RAM, 4 GB eMMC
- › Speichererweiterung: Micro SD-Karten-Slot
- › Farbiges Touch-Display: 2,4", 320x240 Pixel, kapazitiv (erlaubt Wischgesten)
- › Flache Bauform, Maße: 90x90x17,5mm
- › 8 Universaleingänge: Digital/Analog 0-9VDC, Analog 0-5 kΩ
- › 4 schnelle Zähleingänge: Digital, Frequenz bis 1kHz
- › 4 Motorausgänge 9V/250mA (max. 1 A), Geschwindigkeit stufenlos regelbar, kurzschlussfest, alternativ 8 Einzelausgänge z. B. für LEDs
- › 3 Servo-Ausgänge 5V (max. 2A), kurzschlussfest
- › Kombiniertes Bluetooth / WLAN-Funkmodul: Bluetooth 5.0 (BR, LE & EDR), WLAN Dual band 2.4 GHz und 5 GHz 802.11 a/b/g/n
- › USB 2.0 Client: Mini USB-Buchse zum Anschluss an den PC
- › USB Host-Schnittstelle: USB-A Buchse z.B. für fischertechnik USB-Kamera oder USB-Sticks
- › Kamera-Schnittstelle: über USB-Host, Linux Kamera Treiber im Betriebssystem integriert
- › 2x Stiftleiste 6-polig: zur Erweiterung der Ein- und Ausgänge (bis zu 9 TXT 4.0 Controller koppelbar) sowie PC-Schnittstelle
- › Integrierter Lautsprecher zum Abspielen von Sounds (WAV-Daten)
- › Linux basiertes Open-Source Betriebssystem, Firmware-Update über Cloud, USB Stick oder Micro SD-Karte
- › Programmierung mit ROBO Pro Coding (grafisch und Python), C/C++ Compiler (nicht enthalten) - Weitere Programmierungsmöglichkeiten über REST-Schnittstelle
- › Verfügbare Ausgangsspannungen 9V, 5V und 3,3V
- › Stromversorgung: 9V DC-Buchse 3,45 mm oder fischertechnik-Buchsen 2,5 mm

Abb. 2: Angaben zu den technischen Daten des TXT 4.0 im aktuellen fischertechnik-Katalog

Dessen aktuelles Modell 4B ist inzwischen seit einem Jahr auf dem Markt und verfügt über vier mit 1,5 GHz getaktete 64-Bit-Kerne und (je nach Modell) über 1 bis 8 GB RAM. Waren TXT und Rasp-Pi 2013 also noch gleichauf, so liegt zwischen ihnen heute – was Rechenleistung und Speicheranbau angeht - ein Faktor von mehr als 10. Das ist allerdings, wie jeder Arduino-Fan weiß, nicht zwangsläufig ein Nachteil.

Aber was genau steckt im TXT 4.0? Dazu verrät fischertechnik keine weiteren Details. Aber die recht auffälligen Eckdaten der CPU lassen genauere Rückschlüsse zu. Vor allem der zusätzliche Cortex-M4-Kern ist markant und passt z. B. exakt auf die 650 MHz-Variante des STM32MP157 von STMicroelectronics [1].

Dieser Chip besticht laut ST vor allem durch seinen 3D-Grafik-Prozessor. Zielanwendung für diesen Prozessor sind laut ST Linux-betriebene Industriesteuerungen, also recht genau das, was fischertechnik mit dem TXT 4.0 auch macht. Der integrierte kleine Cortex-M4 ist mit maximal 209 MHz getaktet und soll einfache Echtzeit-Aufgaben wie z. B. Motorsteuerungen übernehmen. Er übernimmt damit wohl die Aufgaben, für die im alten TXT noch ein separater Arm-Prozessor auf der sogenannten Motorplatine verbaut wurde.

Den STM32MP157 bietet ST in Varianten von 650 und 800 MHz an. Dass fischertechnik hier nicht die schnellere Version gewählt hat, überrascht zunächst, zumal sie im Zusammenhang mit dem alten TXT gerne auf dessen höhere Taktung gegenüber den Lego-Controllern verwiesen haben. Warum also hier die geringere Taktung?



Abb. 3: SOM-SL (Foto: Kontron)

Eine naheliegende Lösung dieses Rätsels ergibt sich, wenn man davon ausgeht, dass fischertechnik die CPU diesmal nicht separat einkauft, sondern im TXT 4.0 ein „rundum-sorglos“-Paket einsetzt. Der gerne betonte Fokus auf Made-in-Germany engt den Kreis der Verdächtigen schnell ein und man landet recht bald im bayerischen Ismaning bei der Firma Kontron. Mit dem SOM-SL

STM32MP157 hat Kontron ein kleines Modul im Angebot, das es allerdings nur in einer Variante mit der 650-MHz-CPU gibt [3] (Abb. 3).

Das klingt sehr nach unserem TXT 4.0. Zusätzlich ist das Modul in Varianten mit 256 oder 512 MB RAM erhältlich sowie mit 256 MB oder 512 MB NAND-Flash. Letzteres scheint nicht ganz zu den von fischertechnik angegebenen 4 GB eMMC-Flash zu passen. Allerdings werden die 256 bzw. 512 MB-NAND-Flash zum Laden des Linux-Kerns benötigt. Es ist also davon auszugehen, dass fischertechnik hier zusätzlich für das übrige Linux-System verbauten 4 GB eMMC-Speicher erwähnt.

eMMC kann man sich als eine fest im Gerät verlötete SD-Karte vorstellen. Der Zugriff auf eine eMMC erfordert in der Regel einen bereits laufenden Betriebssystemkern mit passenden Treibern, was eben wieder genau die Zweiteilung in 256/512 MB NAND-Flash für den Betriebssystem-Kern und 4 GB eMMC für das übrige Dateisystem erklärt.

Was steckt sonst noch im TXT 4.0? Hier müssen wir leider etwas mehr raten. Da das Kontron-Modul kein WLAN oder Bluetooth mitbringt (dafür aber Ethernet!) wird ein zusätzlicher Baustein dafür nötig. Die Tatsache, dass es sich für fischertechnik anscheinend finanziell lohnt, für den TXT-Smart-Controller auf eben diesen Baustein zu verzichten und stattdessen einen USB-WLAN-Dongle beilegt, sagt uns vor allem eines: Der von fischertechnik für WLAN und Bluetooth eingesetzte Baustein ist überraschend teuer. Aus dem Umfeld von fischertechnik kam der Hinweis auf das PAN9026-Modul von Panasonic [4]. Vor allem der recht hohe Einkaufspreis von knapp zehn Euro lässt das Modul plausibel erscheinen. Dieses sehr leistungsfähige Modul beherrscht alle gängigen und aktuellen Bluetooth- und WLAN-Varianten und wird mit passenden Linux-Treibern geliefert (Abb. 4).



Abb. 4: PAN9026-Modul (Foto: Panasonic)

Interessant ist auch das Display. Neben WLAN/Bluetooth und Lautsprecher ist das Display die dritte Komponente, auf die fischertechnik am TXT-Smart laut Katalog gegenüber dem TXT 4.0 verzichtet. Danach hat das Display nach wie vor eine Auflösung von 240x320 Bildpunkten bei einer Größe von 2,4 Zoll. Dessen im alten TXT recht unzuverlässig auf mechanischen Druck reagierender Touchsensor wurde laut Katalog durch ein auf reine Annäherung reagierenden kapazitiven Sensor ersetzt. Das ist interessant, da es überraschenderweise die einzige Komponente ist, zu der nichts weiter in Erfahrung zu bringen war.

Man darf gespannt sein, wie die Anbindung dieses Displays an den Hauptprozessor erfolgt. Ist die Verbindung wie beim TXT SPI/seriell und erlaubt damit nur den vom TXT bekannten langsamten Bildaufbau? Oder ist der Bildschirm des TXT 4.0 an die von ST beworbene leistungsfähige 3D-fähige Videohardware angeschlossen? Das ist bisher nicht bekannt.

Gewagt scheint der Einsatz dieses etwas exotischen Bildschirms. Ich habe es nicht geschafft, ein entsprechendes Modell im Handel zu finden. Angesichts der Tatsache, dass der alte TXT über acht Jahre unverändert in Produktion war, erscheint die Wahl eines exotischen Bildschirms mutig.

Damit sind die wesentlichen Komponenten des TXT 4.0 bekannt. Es bleibt die Beschaltung der Ein- und Ausgänge. Hier bestünde eigentlich kein Grund, von der schon im TX und eben auch im TXT nahezu identisch

eingesetzten Kombination aus 4051-Multiplexern an den Eingängen und den MC33879-Treibern für die Ausgänge abzuweichen – wenn es den MC33879 noch zu kaufen gäbe. Leider wurde er aber abgekündigt und wird offiziell nicht mehr hergestellt, was z. B. auch den ftDuino betrifft. Der MC33879 unterscheidet sich von nahezu allen z. B. im Arduino-Umfeld verwendeten Motortreibern dadurch, dass er per SPI-Bus angesprochen wird, was gerade bei den vielen Ausgängen am fischertechnik-Controller intern einiges an Verbindungen spart.

Der ftDuino wird in Zukunft einen Chip wie den TLE94108EL von Infineon oder den dazu Pin-kompatiblen DRV8908 von Texas Instruments einsetzen [5, 6]. Es ist davon auszugehen, dass auch fischertechnik einen vergleichbaren Ersatz einsetzen wird, zumal diese Chips billiger und leistungsfähiger als die beiden MC33879 des alten TXT sind.

Was kostet es?

Damit sind wir am Ende unserer kleinen Reise durch das vermutete Innenleben des TXT 4.0. Gegenüber dem alten TXT ist der TXT 4.0 vor allem einfacher und preisgünstiger aufgebaut. Die teuerste Komponente ist das CPU-Modul, das mit gut 30 € zu Buche schlägt, gefolgt vom WLAN/ Bluetooth-Modul mit knapp zehn Euro. Der gesamte Rest dürfte unter „Kleinkram“ abgehakt werden können und für weniger als 20 € zu haben sein. Inklusive des Gehäuses könnte der Einkaufspreis des TXT 4.0 damit bei ca. 65 € liegen. Hier hat fischertechnik in Zukunft durchaus noch Spielraum, wenn sie auf den Einsatz des CPU-Moduls verzichten und Prozessor und Speicher direkt beziehen – auch für das WLAN/Bluetooth-Modul sollte sich günstiger Ersatz finden lassen.

Während der TXT 4.0 laut fischertechnik für 549 € verkauft werden soll, ist der Kasten mit TXT-Smart-Controller schon für

249 € zu bekommen. Dass die Einsparungen am TXT-Smart gegenüber dem TXT 4.0 auch nur ansatzweise diesen Unterschied rechtfertigen darf bezweifelt werden. Der Einkaufspreis des TXT-Smart dürfte bei ca. 50 € liegen und damit deutlich näher an dem des TXT 4.0 als die Verkaufspreise der beiden Komplett-Kästen suggerieren.

Und Lego?

An dieser Stelle ist ein Blick über den Tellerrand in Richtung Lego interessant. Noch vor wenigen Jahren standen sich mit TXT und EV3 zwei sehr ähnliche Linux-Controller gegenüber, die allerdings auf sehr unterschiedliche Weise programmiert wurden. Interessanterweise geht die Entwicklung der Programmierumgebungen inzwischen in die gleiche Richtung, während nun die technische Entwicklung der Controller stark auseinander läuft.

Mit Blockly-basierter grafischer Programmierung nutzen fischertechnik und Lego in Zukunft eine identische Basis. Und auch wenn es um fortgeschrittene Programmierung gehen soll, setzen beide auf Python. Für den Anwender werden sich beide Systeme also in Zukunft sehr viel ähnlicher anfühlen als bisher.

Sehr unterschiedlich ist aber die in den Controllern verwendete Technik. fischertechnik setzt weiterhin auf Linux und ist gerade dabei, die Leistung seines Controllers zu verdoppeln. Lego geht den entgegengesetzten Weg und setzt kein Linux mehr ein. Stattdessen kommt bei Lego eine 100-Mhz-CPU (immerhin auch ein STM32 aus dem Hause ST) mit 320 kB RAM und 32 MB Flash zu Einsatz. Zwischen TXT 4.0 und Lego Spike liegt damit – was Rechenleistung und Speicher angeht – ein Faktor von deutlich mehr als 10.

Man kann davon ausgehen, dass das Innenleben des aktuellen Lego-Controllers damit bei großen Stückzahlen einen Einkaufspreis von unter 10 € hat. Lego verliert damit z. B. die Möglichkeit, einfache USB-Kameras

anzuschließen und wird sich, anders als fischertechnik, mit intelligenten Kameras à la Pixy begnügen müssen. Ob solche Möglichkeiten den ungleich höheren Hardwareaufwand auf fischertechnik-Seite rechtfertigen, muss sich zeigen.

Der Preisunterschied der Baukästen von Lego und fischertechnik spiegelt diese Unterschiede jedenfalls in keiner Weise wieder. Der Verdienst auf Lego-Seite dürfte ungleich höher sein.

Fazit

Der neue TXT 4.0 ist dem alten Gerät deutlich überlegen und dürfte auch günstiger herzustellen sein als sein Vorgänger. Aus Sicht des Bestands-fischertechnikers ist der TXT 4.0 also ein uneingeschränkter Gewinn.

Die Frage ist, ob das reicht. Lego macht vor, wie sich der Herstellungspreis deutlich senken lässt, wenn man die Hardware exakt auf

den Einsatzzweck zuschneidet. Die fischertechnik-Hardware lässt hier ungleich mehr Spielraum für zukünftige Erweiterungen. Ob fischertechnik dieses Potenzial nutzen wird oder ob sie es wie beim TXT der Community überlassen, zu zeigen, was in ihrem Produkt für verborgene Fähigkeiten schlummern, wird sich zeigen.

Referenzen

- [1] Till Harbaum: *Von Lucky Logic zu RoboPro Coding*. [ft:pedia 1/2021](#), S. 103–109.
- [2] STMicroelectronics: [STM32MP157](#).
- [3] Kontron: [SOM-SL STM32MP157](#).
- [4] Panasonic: [PAN9026](#).
- [5] Infineon: [TLE94108EL](#).
- [6] Texas Instruments: [DRV8908](#).

Computing

ROBO Pro Coding – ein Blick hinter die Kulissen

Till Harbaum

ROBO Pro ist seit vielen Jahren die Programmierumgebung der Wahl für die Robotics-Controller von fischertechnik. Mit ROBO Pro Coding ändert sich das trotz des ähnlichen Namens fundamental.

In der letzten Ausgabe der ft:pedia habe ich bereits über Blockly geschrieben [1]. Der Beitrag konzentrierte sich auf Blockly als Basis von ROBO Pro Coding und was von Blockly im Vergleich zum „alten“ ROBO Pro zu erwarten ist. Im vorliegenden Beitrag geht es dagegen um ROBO Pro Coding als Ganzes. Aus welchen Teilen besteht es? Wie funktioniert es?

Dass ich überhaupt den dafür nötigen tiefen Einblick nehmen kann ist der Tatsache geschuldet, dass ROBO Pro Coding keine klassische PC-Windows-Software mehr ist, sondern eine Browser-Anwendung, die zur Zeit über das Internet erreichbar ist.

Diese Anwendung war während des Erscheinens der ersten Werbevideos zu dem Thema Anfang des Jahres mit Hilfe gängiger Suchmaschinen zu finden. Inzwischen taucht sie nicht mehr als Suchergebnis auf, so dass man davon ausgehen kann, dass fischertechnik das Auffinden der Seite aktiv unterdrückt. Aus diesem Grund verweise ich an dieser Stelle auch nicht auf diese Seite.

Die Puzzleteile

Wie bereits in der letzten Ausgabe beschrieben basiert ROBO Pro Coding für den Benutzer sehr offensichtlich auf Googles Blockly.¹ Neben dieser grafischen Programmierung erlaubt ROBO Pro Coding aber auch eine klassische textbasierte Programmierung in Python.

Als dafür nötigen Code-Editor verwendet fischertechnik den Monaco Editor von Microsoft.² Viele fischertechnik-Benutzer hatten mit diesem Editor bereits Kontakt, da er auch die Basis vieler Online-Dienste von Microsoft darstellt und auch in Visual Studio Code eingesetzt wird.³ Blockly und der Monaco Editor sind zwei sehr beliebte, aktive und häufig genutzte Projekte und sicher keine schlecht Wahl für eine Neuentwicklung, die man noch einige Jahre betreiben möchte.

Die Editoren rahmt ROBO Pro Coding in eine so genannte Single-Page-Webapplikation ein (Abb. 1). Das bedeutet, dass man durch Interaktion mit den verschiedenen Bereichen der Webseite sich nicht wie üblich zwischen verschiedenen Seiten hin- und her bewegt.

¹ Google: [Blockly](#).

² Microsoft: [Monaco Editor](#), github.io.

³ Microsoft: [VS Code](#), github.com.

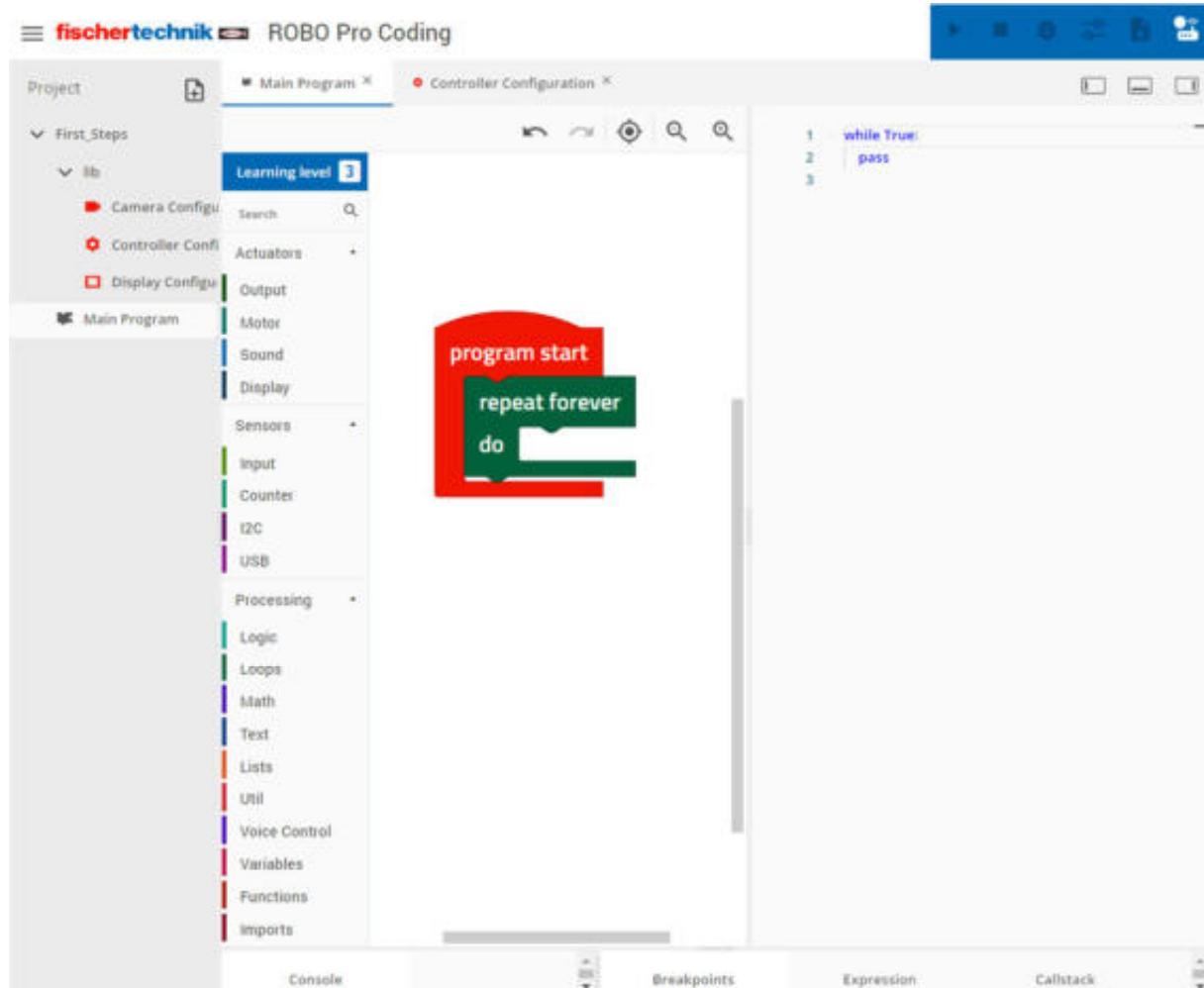


Abb. 1: ROBO Pro Coding IDE (Stand: Juni 2021)

Stattdessen wird beim Aufruf einer Single-Page-Anwendung einmal die gesamte Funktion geladen und der Benutzer agiert die ganze Zeit mit ein und derselben Seite, die sich allerdings permanent dynamisch dem anpasst, was der Nutzer gerade machen möchte.

ROBO Pro Coding basiert dafür auf Angular⁴, einem Google-Framework für Single-Page-Anwendungen. Angular bietet eine große Auswahl vorgefertigter Mechanismen, u. a. zur Anzeige von Dialogen und Hinweisen, für Menüs und für den fließenden Wechsel zwischen verschiedenen Ansichten (z. B. dem Blockly-Editor und dem Monaco-Editor). Zusätzlich passt sich eine

Angular-basierte Umgebung an das Endgerät an und die Anordnung der verschiedenen Elemente erfolgt z. B. auf dem kleinen Touchbildschirm eines Smartphones anders als auf einem mausgesteuerten PC-Bildschirm. Auf diese Weise erleichtert es Angular, Anwendungen zu schaffen, die unabhängig vom Endgerät nutzbar sind und die sich „wie aus einem Guss“ bedienen lassen. Auch Angular ist ein aktives und stabiles Produkt und dürfte für einige Jahre ROBO Pro Coding gut sein.

Die letzte Komponente, auf der ROBO Pro Coding basiert, ist die MXGraph-Bibliothek⁵, eine Bibliothek zur Darstellung von Graphen. ROBO Pro Coding nutzt diese

⁴ Google: [Angular](#).

⁵ JGraph: [MXGraph](#), github.com.

Bibliothek, um dem Benutzer die Konfiguration der Bildschirmdarstellung am TXT bzw. die Kamera-Konfiguration zu ermöglichen. In beiden Fällen kann der Benutzer grafische Objekte in ein Fenster ziehen dort anordnen und konfigurieren (Abb. 2)

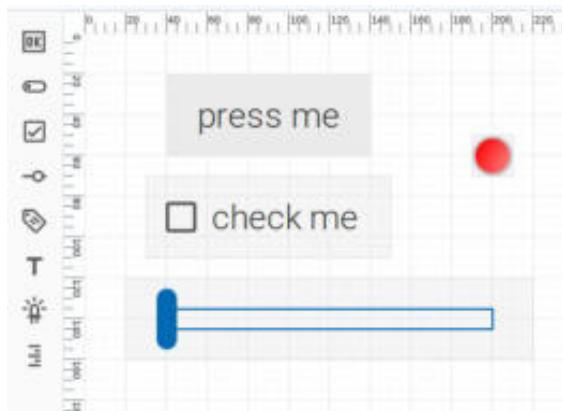


Abb. 2: Konfiguration des TXT-Displays

Leider wurde die Weiterentwicklung der MXGraph-Bibliothek bereits Ende 2020 eingestellt, und auch davor stockte die Entwicklung bereits sehr. Vor allem die nicht mehr zeitgemäße Code-Basis, die noch aus vor-Smartphone-Zeiten stammt, macht das Projekt schwer zu warten und ist auf Touchoberflächen kaum benutzbar.

Das erklärte Ziel von fischertechnik, ROBO Pro Coding auch auf aktuellen Smartphones und Tabletts laufen zu lassen, ist durch die Verwendung dieser Bibliothek zumindest gefährdet. Aber vor allem erscheint fragwürdig, dass ein frisches Projekt auf dieser Basis ein paar Jahre überleben soll. Ich habe fischertechnik bereits auf diese Problematik hingewiesen und sie sind zuversichtlich, das Problem im Griff zu haben.

TXT-4.0-Kontaktaufnahme

Mir steht kein TXT-4.0 zur Verfügung. Um mich dennoch ein wenig mit ROBO Pro Coding beschäftigen zu können und zu verstehen, wie ROBO Pro Coding und der TXT zusammenarbeiten, war also Kreativität gefragt.

Prinzipiell sind im Browser laufende Anwendungen dankbare Ziele für neugierige

User. HTML5-basierte Webanwendungen wie ROBO Pro Coding sind immer in mehr oder weniger menschenlesbarem Javascript-Code geschrieben und lassen sich mit den Browser-eigenen Werkzeugen bereits trefflich in die Karten schauen. Wer das selbst ausprobieren möchte öffne seine Lieblingswebseite auf dem PC im Browser und drücke die F12-Taste. Der sich dann öffnende Bereich erlaubt tiefe Einblicke in den Code und die Kommunikation einer Webseite.

Kommunikation ist in diesem Fall genau das, was wir suchen, denn irgendwie muss ROBO Pro Coding ja mit dem TXT-4.0 in Kontakt treten. Zusätzlich bieten sich Netzwerk-Analysetools wie Wireshark an.

Selbst wenn der TXT gar nicht existiert sind einseitige Kommunikationsversuche damit zu erkennen.

Die ersten Analysen dieser Art führten sofort zu unerwarteter Verwirrung: Es gab laut Netzwerkanalyse schlicht keinen Kommunikationsversuch des Browsers. Einen ersten Hinweis lieferte ein Blick in die Console des Browsers (eines der Dinge, die man über F12 erreicht). Dort erscheint die Meldung:

```
Blocked loading mixed active content  
"http://192.168.7.2/api/v1/ping"
```

Im Klartext: Die Webseite hat sehr wohl versucht, mit dem Gerät unter 192.168.7.2 Kontakt aufzunehmen. Aber der Browser erlaubt genau das nicht. Der Grund ist, dass viele Webseiten heute via „https“ verschlüsselt ausgeliefert werden. So auch ROBO Pro Coding. Die Nutzung von https verschlüsselt die Daten auf dem Transportweg zum User und verhindert Veränderungen auf diesem Weg. Das ist z. B. beim Homebanking offensichtlich wichtig. Findet die Kommunikation erstmal per https statt und fühlt sich der Nutzer entsprechend sicher, so darf natürlich nicht heimlich doch

noch irgendein Teil unverschlüsselt übertragen werden.

Aber genau das ist hier der Fall. ROBO Pro Coding selbst wurde verschlüsselt an den Benutzer ausgeliefert. Der Verbindungsversuch zum TXT erfolgt aber unverschlüsselt. Aus Sicht des Browsers kommuniziert der Benutzer also verschlüsselt. Die Webseite versucht aber weitere Daten unverschlüsselt nachzuladen. Das verbieten aktuelle Browser seit Kurzem per Voreinstellung. Der Browser kann schließlich nicht zwischen wichtigen Bankgeheimnissen und den (unwichtigen) Daten eines TXT-4.0 unterscheiden. Und wer weiß, vielleicht baut ja jemand aus einem TXT einmal ein Pin-Terminal zur Online-Banking-Geheimzahl-Eingabe.

Es ist daher anzunehmen, dass fischertechnik ROBO Pro Coding in anderer Form vertrieben wird und die aktuelle Webseite nur eine Übergangslösung ist.

Dem Ziel, der Kommunikation von ROBO Pro Coding auf den Zahn zu fühlen, war ich keinen Schritt näher gekommen.

Es gibt mehrere Wege, unverschlüsselten Inhalt zu erlauben. Einige Browser erlauben das, die Sicherheitseinstellungen entsprechend zu senken. Außerdem kann man ganz ROBO Pro unverschlüsselt übertragen. Dann ist auch die weitere Kommunikation unverschlüsselt erlaubt. Oder aber man kommuniziert nur mit dem eigenen PC. Das darf man jederzeit auch unverschlüsselt. Ich habe die letzteren Wege verfolgt.

Zunächst wurde ROBO Pro Coding auf einen lokalen PC heruntergeladen und dort über einen eigenen kleinen Webserver bereitgestellt. Das hat den zusätzlichen Vorteil, dass diese Experimente nicht ständig auf fischertechniks Web-Präsenz aufschlagen.

Zusätzlich habe ich die Kommunikation auf meinen eigenen PC umgeleitet. Das wäre mangels echtem TXT ja sowieso das Ziel

der Wahl, um die Kommunikation zu analysieren.

Ab hier ist es reine Fleißarbeit. Ein kleines Programm wurde geschrieben, das auf jede Kommunikation seitens ROBO Pro Coding mit einer passenden Antwort zu reagieren versucht. Recht wenig Aufwand ist nötig, bis ROBO Pro Coding z. B. versucht, ein vom Anwender in ROBO Pro Coding selbst geschriebenes Programm auf den TXT herunterzuladen.

Die Rolle des TXT übernimmt dabei mein kleines selbstgeschriebenes Programm: Statt die von ROBO Pro Coding empfangenen Daten auszuführen, wie es ein echter TXT machen würde, schreibt es die empfangenen Daten jedoch einfach zur weiteren Analyse auf die Festplatte.

ROBO Pro Coding schickt gleich mehrere Dateien an den TXT-4.0, wenn dort ein Programm ausgeführt werden soll. Die meisten davon enthalten Python-Programmcode. Eine Datei enthält sogenannten QML-Code:

```
// auto generated content from
display configuration
import QtQuick 2.2
import QtQuick.Window 2.0
import QtQuick.Controls 1.1
import QtQuick.Controls.Styles 1.1
import QtQuick.Extras 1.4

TXTWindow {
    Rectangle {
        id: rect
        color: "grey"
        anchors.fill: parent
    }
    TXTButton {
        id: txt_button
        text: "press me"
        font.pixelSize: 24
        enabled: true
        x: 40
        y: 20
        width: 100
        height: 40
    }
    ...
}
```

Das ist eine sogenannte QML-Beschreibung einer Benutzeroberfläche in QtQuick bzw. Qt. Qt ist die Grafikbibliothek, die

auch die Community-Firmware zur Darstellung der Benutzeroberfläche am Gerät verwendet. QtQuick bzw. QML sind lediglich eine etwas modernere Variante davon. Für uns bedeutet das zweierlei:

Erstens hat fischertechnik (oder der für die Entwicklung von ROBO Pro Coding verantwortliche Dienstleister) sich die Community-Firmware noch etwas genauer anschaut, als bisher angenommen.

Und zweitens nutzt fischertechnik damit Bibliotheken, die auch auf jedem Raspberry-Pi (und jedem Linux-PC generell) mitgeliefert werden.

Was liegt also näher, als die von ROBO Pro Coding heruntergeladenen Daten einfach den entsprechenden Grafikbibliotheken vorzulegen? Das Ergebnis ist in Abb. 3 zu sehen. Mein PC verhält sich wie ein TXT und zeigt die eigentlich für diesen bestimmte Benutzeroberfläche in einem kleinen Fenster selbst an.

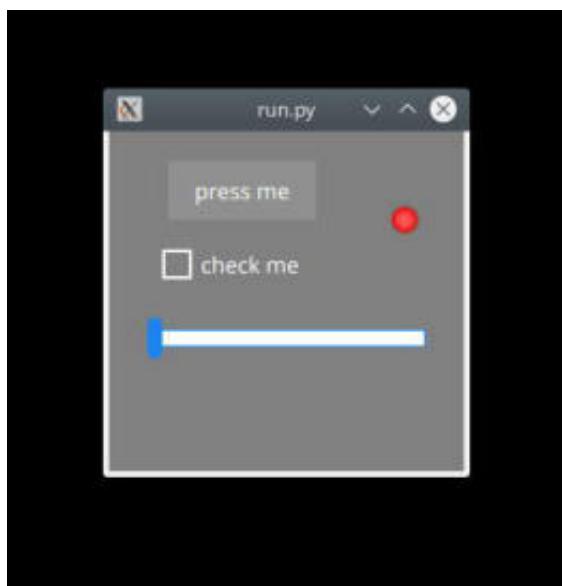


Abb. 3: TXT-Display im PC-Screen

In einem zweiten Schritt müssen nun nur noch die ebenfalls empfangenen Python-Programme mit den Grafikdaten kombiniert und ausgeführt werden, und schon können ROBO Pro Coding-Programme statt auf einem echten TXT-4.0 einfach auf einem

Linux-PC oder einem Raspberry-Pi ausgeführt werden [2]. Bindet man dann noch einen ftDuino per USB an den PC an, dann können sogar dessen Ein- und Ausgänge in ROBO Pro Coding genutzt werden. Und schließlich spricht nichts dagegen, diese Programme auch auf dem alten TXT (z. B. mit Hilfe der Community-Firmware) zu nutzen und so den alten TXT für ROBO Pro Coding fit zu machen. Die Grundlagen dafür sind nun alle vorhanden.

Kommunikation zur Laufzeit

Ist ein Programm einmal auf den TXT übertragen, dann öffnet ROBO Pro Coding einen Kommunikationskanal, um z. B. Textausgaben vom TXT zu empfangen und im Browser darzustellen. Der Anwender kann sich so Ausgaben aus dem laufenden Programm in den Browser schicken. Auch weitgehende Kommunikation z. B. zum Debugging scheint vorgesehen zu sein. Diesen Teil habe ich aber bisher nicht ausprobiert.

Generell nutzt ROBO Pro Coding an diesen Stellen eher altbackene Technik zur Kommunikation zwischen TXT-4.0 und Browser bzw. ROBO Pro Coding. Der Einsatz moderner Techniken wie Websockets hätte einige der eingangs erwähnten Probleme vermeiden können und würde eine wesentlich flüssigere Kommunikation in beide Richtungen ermöglichen.

Weitere Community-Projekte?

ROBO Pro Coding basiert wie der TXT-4.0 selbst auf Open-Source-Technik. Wie weit fischertechnik diese Philosophie selbst verfolgt ist bisher nicht bekannt.

Die Funktionen des TXT-4.0 sind aus ROBO Pro Coding-Sicht eher schlicht und lassen sich leicht nachbilden, wie sich gezeigt hat. Tatsächlich erschließt sich an dieser Stelle kaum der Hardwareaufwand, den fischertechnik beim TXT-4.0 treibt und der die Kosten für das Gerät in die Höhe treibt. Um durch ROBO Pro Coding steuerbar zu

sein bräuchte es nur einen Bruchteil der Technik, die im TXT-4.0 steckt.

Aber auch ROBO Pro Coding selbst könnte ein lohnendes Ziel für Veränderungen und Erweiterungen durch die Community sein. Fehlende Funktionen in ROBO Pro Coding selbst nachzurüsten ist zumindest theoretisch möglich. In der Praxis steht dem allerdings eine besondere Eigenart des Angular Frameworks entgegen: Um die zu übertragende Datenmenge gering zu halten, schrumpft Angular den Code so weit es geht ein. Aus eigentlich gut lesbarem Code wird dabei ein minimalistisches Kauderwelsch, das z. B. von allen erklärenden Kommentaren im Code bereinigt wurde und das keinerlei Text-Formatierungen mehr enthält.

Im Klartext: Das was z. B. von Blockly als Teil von ROBO Pro Coding beim Endanwender ankommt ist kaum lesbar und praktisch nicht durch den Endanwender erweiterbar. So wie es jetzt aussieht wäre der Aufwand, ROBO Pro Coding durch die Community zu erweitern, kaum zu rechtfertigen. Auch hier liegt es also primär in der Hand von fischertechnik, durch Veröffentlichung zusätzlicher Dokumentation und Quellcodes eine Mitentwicklung durch die Community zu ermöglichen.

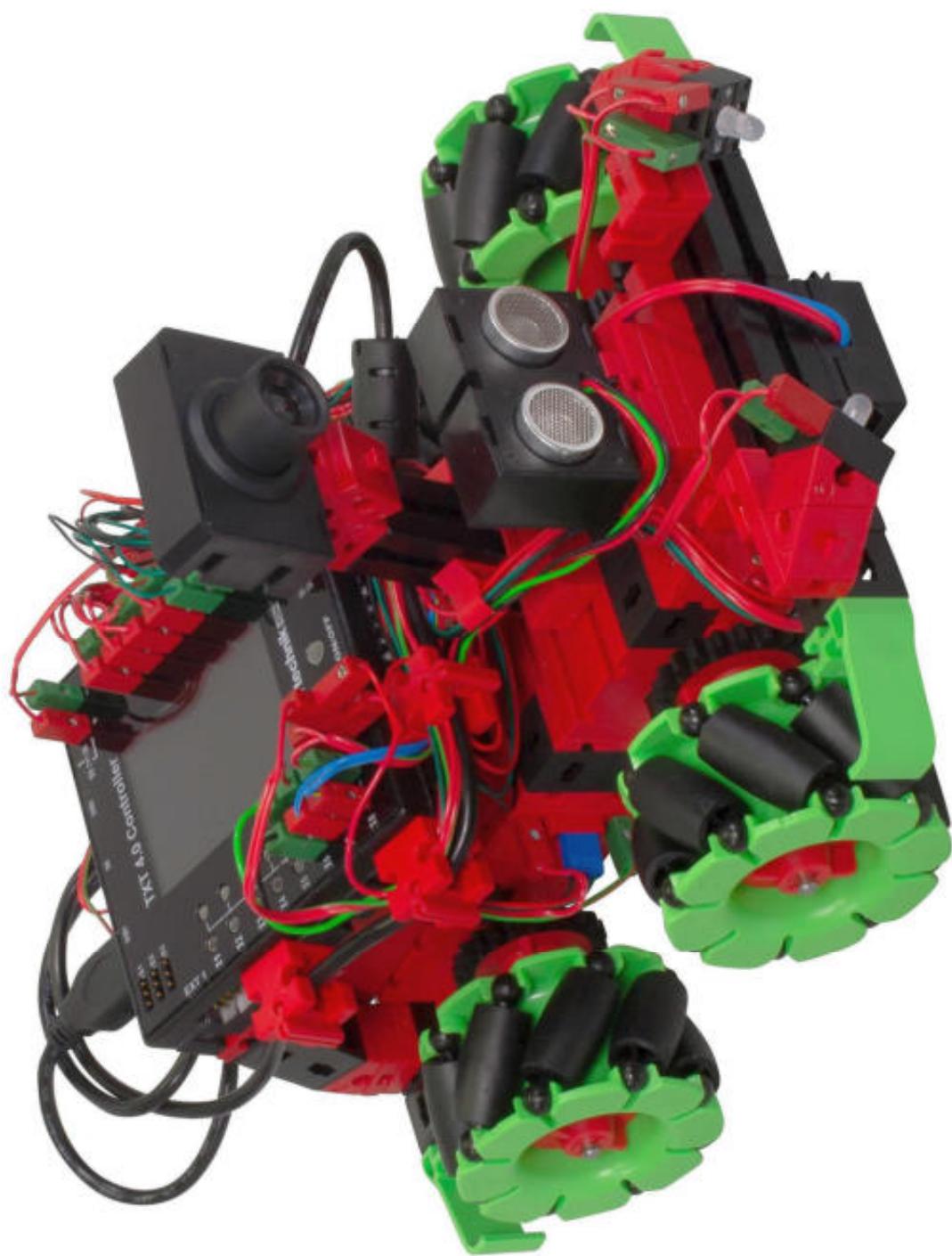
Fazit

Wie schon beim TXT-4.0 setzt fischertechnik auch bei ROBO Pro Coding überwiegend etablierte und robuste Technik ein. Wenn Lego das nahezu gleiche Ziel mit wesentlich einfacherer Technik anvisiert dürften die Aufwände im Projektmanagement und der Entwicklung entsprechend höher sein. Hier macht sich vor allem eine hohe Stückzahl bemerkbar, die Optimierungen der Hardware bei Lego lohnender macht.

Die Lösung von fischertechnik setzt an vielen Stellen auf etablierte und frei verfügbare fertige Komponenten. Den Preis dafür zahlt der Kunde mit der im Vergleich mit Lego um ein Vielfaches aufwändigeren und teureren Hardware des TXT-4.0. Zugleich bietet sich aber die Chance für zukünftige Erweiterungen, die die Möglichkeiten des TXT-4.0 gegenüber den aktuellen Lego-Controllern ausschöpfen. Vor allem hier könnte fischertechnik von einer Zusammenarbeit mit der Community profitieren.

Referenzen

- [1] Till Harbaum: [Von Lucky Logic zu RoboPro Coding](#). ft:pedia 1/2021, S. 103-109.
- [2] Till Harbaum: [SofTXT](#), github.com.



*Mecanum-Omniwheel-Roboter
aus dem neuen Baukasten "Robotics Hightech"*