

Editorial

Wozu fischertechnik?

Ein fischertechniker wird diese Frage für etwa ebenso überflüssig halten wie ein Fisch die Frage nach dem Zweck von Wasser – beides ist selbstvident. Dennoch schadet es sicherlich nicht, der Sache mit Unterstützung eines – zweifellos äußerst belesenen – Large Language Models (LLM) auf den Grund zu gehen. Und tatsächlich haben LLMs sehr genaue Vorstellungen davon, was fischertechnik leistet: für uns, unsere Zukunft und die unserer Kinder. Ein kleiner Überblick:

- Technische und naturwissenschaftliche Kompetenzen

Klar: Grundlagen der Mechanik und Statik, Elektrotechnik, Elektronik, Pneumatik und Informatik, Regelungstechnik und Sensorik – alles das lernt man quasi „on the fly“.

- Kognitive, methodische Kompetenzen

Diese Kompetenzen sind schon nicht mehr so offensichtlich, aber bei kurzer Betrachtung eindeutig: Problemlösungskompetenz, logisches und algorithmisches Denken, Modellbildung, Abstraktion, Planung, Strukturierung. Hinzu kommt das iterative Arbeiten – das kontinuierliche Verbessern durch Prototyping, Testen und Optimieren. Schließlich Fehleranalyse und -beseitigung – wichtige Fertigkeiten, die jeder Modellentwurf trainiert.

- Handwerkliche Kompetenzen

Natürlich gehören auch Feinmotorik – die Auge-Hand-Koordination – und räumliches Vorstellungsvermögen dazu; letzteres lässt sich mit dem großartigen fischertechnik-Designer von Michael Samek unterstützen.

Dirk Fox, Stefan Falk

Wir lernen, Baupläne, Schaltpläne und (technische) Zeichnungen zu verstehen und umzusetzen – vom abstrakten Abbild in die konkrete Wirklichkeit. Und schließlich Präzision: Je ausgefuchster das Modell, desto mehr kommt es auf das genaue Arbeiten im Millimeterbereich an. Da ist das Maßsystem von fischertechnik ein großes Plus: Wer es verletzt, baut „schief“.

- Soziale und personale Kompetenzen

Durchhaltevermögen und Frustrationstoleranz sind vielleicht die beiden wichtigsten in der Liste, meist hoffentlich gefolgt vom Erlebnis der Selbstwirksamkeit (sobald das Modell funktioniert). Wenn wir anderen unsere Entwürfe und Entwicklungen vorstellen, kommen Kommunikationsfähigkeit und Teamarbeit dazu – so z. B. im Forum, den fischertechnik-AGs oder gemeinsamen Entwicklungsprojekten.

Und nicht zuletzt ist da eine Kompetenz, die häufig nicht mit Technik und technischen Berufen in Verbindung gebracht wird:

- Kreativität

Dabei ist jedem fischertechniker klar: Genau das ist der Schlüssel – von der Modellidee bis zum Lösungsdetail strotzt jede Entwicklung von kreativen Ideen. Wer also irgendwann beruflich „irgendwas Kreatives mit Menschen“ machen will – der sollte es mal als Ingenieur probieren.

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

Wozu fischertechnik?	2
Kleinmodelle zum Nachbauen.....	4
Le Grand Éléphant.....	9
Die fischertechnik-Leiter.....	15
Federn und Dämpfer für fischertechnik-Modelle.....	21
Große Reifen für große Modelle.....	32
Großprojekt Seilbahn (Teil 12): Sie lebt!	55
Schwertransporte.....	61
Gegenläufige Propeller	65
Zyklisch ungleichförmige Übersetzung mittels Kardanfehler ...	70
Zahnradgetriebe für beliebige Übersetzungsverhältnisse.....	80
Servopneumatik mit fischertechnik.....	88
Synchronmotoren – revisited.....	106
Verbindungsstecker für Silberlinge selbst herstellen	116
Es kracht.....	119
Der COMPUTING Baukasten – eine Retrospektive	129
Die FTlegacy- und FTmodule-Software-Bibliotheken	137
Robocar mit Pixy2-Kamera	149
fischertechnik-Modelle mit der Oxocard steuern – Teil 3: Der I2C-Bus.....	157

Termine

Was?	Wann?	Wo?
Bescherung	24.12.2025	Weihnachtsbaum
Nordconvention	18.01.2026	Münster
Clubdag	14.02.2026	De Bilt

Impressum

<http://www.ftpedia.de>

Herausgeber: Dirk Fox, Ettlinger Straße 12-14,
76137 Karlsruhe und Stefan Falk, Ulmenweg 9, 76275 Ettlingen

Autoren: Florian Bauer, Axel Chobe, Arnoud van Delden,
Matthias Dettmer, Stefan Falk, Dirk Fox, Stefan Fuss, Fabian
Haas, Peter Krijnen, Ludger Mäsing, Volker Paelke, Jeroen
Regtien, Tilo Rust.

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

Modell

Kleinmodelle zum Nachbauen

Ludger Mäsing

Als ich 2008 irgendwann einmal den Baukasten „Service Station“ in Händen hielt, fiel mir auf, dass dieser vielleicht noch ergänzt werden könnte. Dazu stelle ich einige weitere Modelle vor.

Der Schweißer

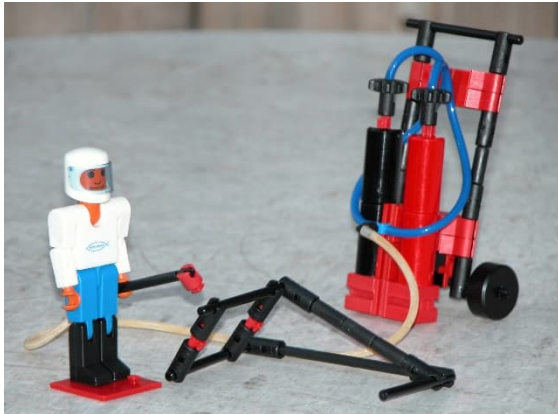


Abb. 1: Der Schweißer bei der Arbeit

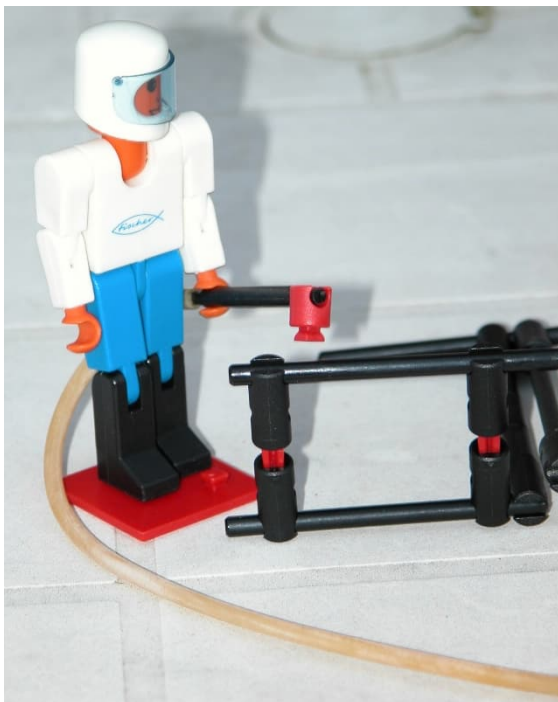


Abb. 2: Das rote Teil an der schwarzen Achse stellt die Flamme dar

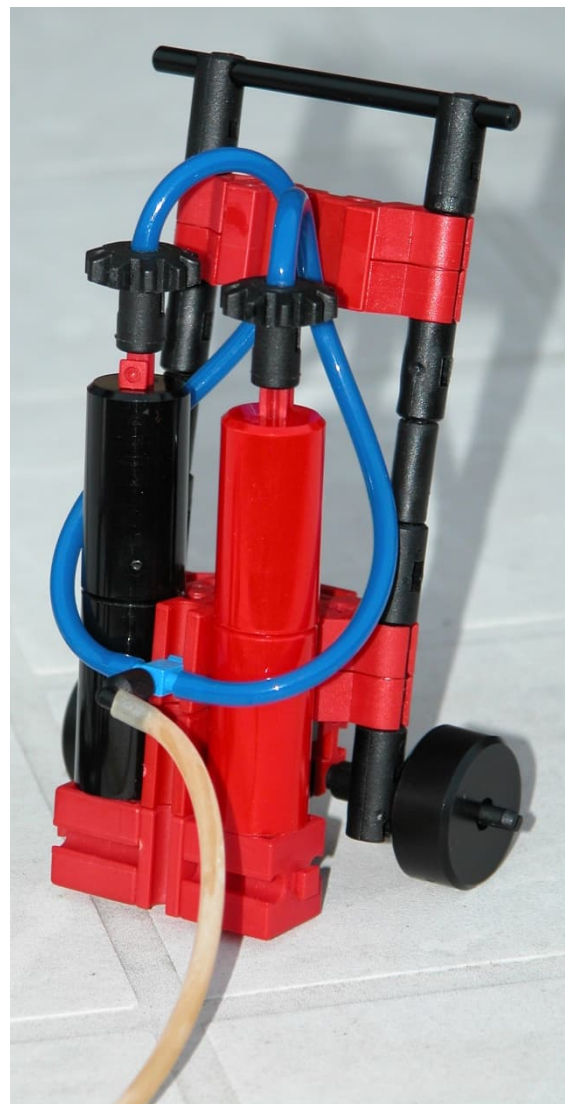


Abb. 3: Die Schweißstation

In der Schweißstation gibt es die beiden Flaschen für Sauerstoff und Acetylen. Für dieses Modell gibt es eine fischertechnik-Designer-Bauanleitung [1].

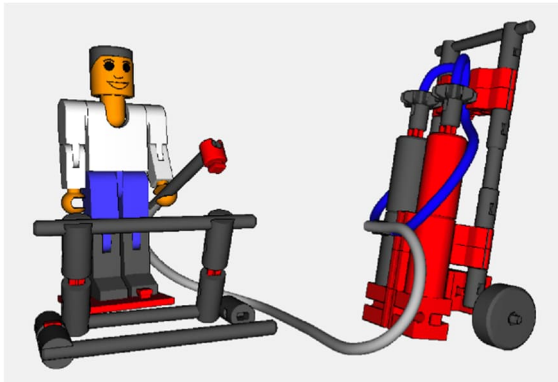


Abb. 4: Das Designer-Modell des Schweißers

Ich danke Stephan Kallauch dafür, dass er es ermöglicht hat, dass in dem Modell auch die Schläuche dargestellt sind.

Die blauen Schläuche sind die bekannten fischertechnik Pneumatik Schläuche, der helle Schlauch ist ein Gummischlauch wie er früher mit den Fahrradventilen genutzt wurde. Dieser ist hochflexibel. Es gibt aber auch welche, die nach einiger Zeit verhärten und brüchig werden. Leider kann ich nicht sagen wie man den Unterschied vorher erkennen kann.

Da die Stellplatten für die Figuren recht rar sind, habe ich nach einer anderen Lösung gesucht. Eine Bauplatte 30×30 mit zwei Zapfen funktioniert genauso gut. Lediglich ein Zapfen hält einen Stiefel. Das reicht vollkommen aus.

Die Rüttelplatte

Damit sich auf einer Baustelle nicht nur Fahrzeuge befinden, kommt eine Baumaschine dazu. Es soll der Nachbau einer Rüttelplatte sein. Sie rüttelt zwar nicht, aber sie sieht genauso aus.¹

Eine Bauanleitung für den fischertechnik Designer findet sich ebenfalls unter [1].



Abb. 5: Die Rüttelplatte im Einsatz

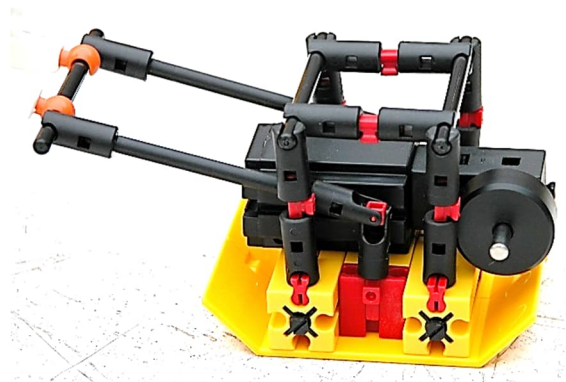


Abb. 6: Größere Ansicht

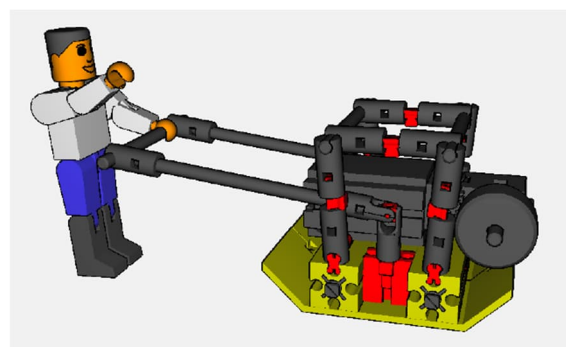


Abb. 7: Designer-Ansicht

¹ Es gäbe vielleicht eine Möglichkeit, diese Funktion einzubauen: Bei den großen Elektronik-Versendern gibt es kleine 3V-Vibrationsmotoren. Die passen evtl. noch in das Modell hinein.

Für die Stromversorgung könnte eine 3V-Knopfzelle genügen. Oder man löst es wie Rüdiger Riedel mit einem Minimotor und Batteriekasten [2].

Ausruhen

Ein kühles Getränk in der Sonne gefällig?

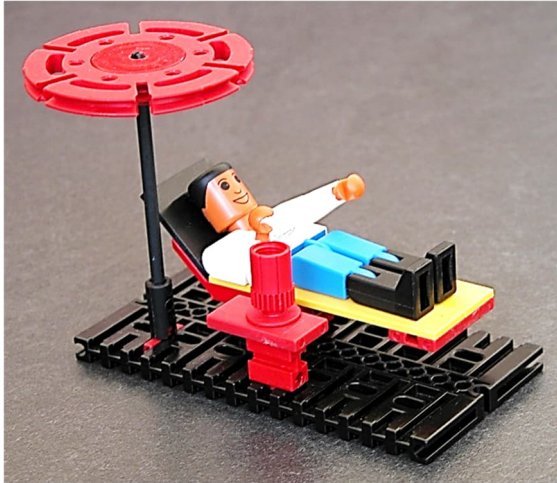


Abb. 8: „Mann“ gönnt sich ja sonst nichts

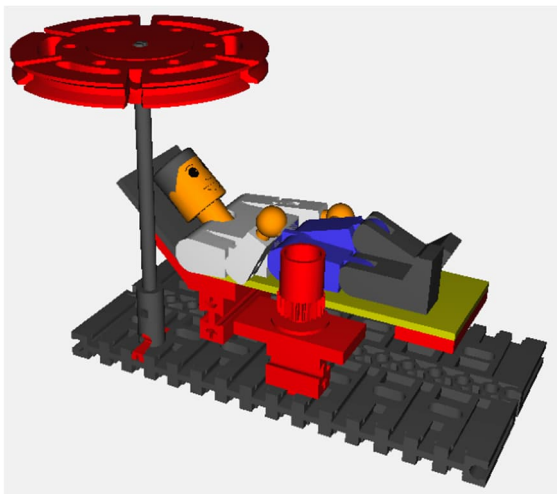


Abb. 9: Ansicht im
fischertechnik-Designer [1]

Paddelboot

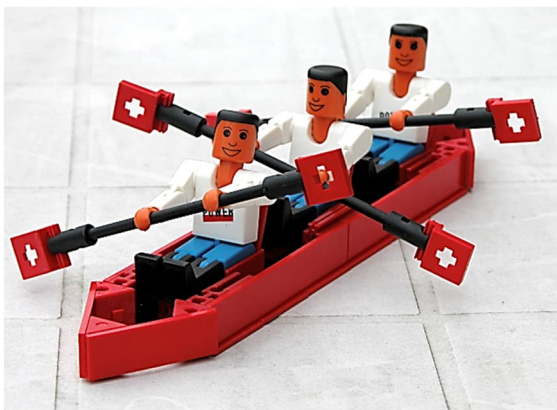


Abb. 10: Ein Dreier-Paddelboot

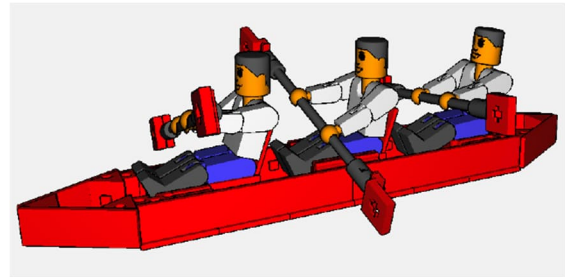


Abb. 11: Ansicht im
fischertechnik-Designer [1]

Drehkranz

2014 habe ich ein wenig mit einigen Bauteilen getüftelt. Durch Zufall ist ein sehr leichtgängiger Drehkranz entstanden. Viele Bauteile benötigt man nicht. Mit ein paar zusätzlichen lässt er sich noch erweitern.



Abb. 12: Ein leichtgängiger Drehkranz

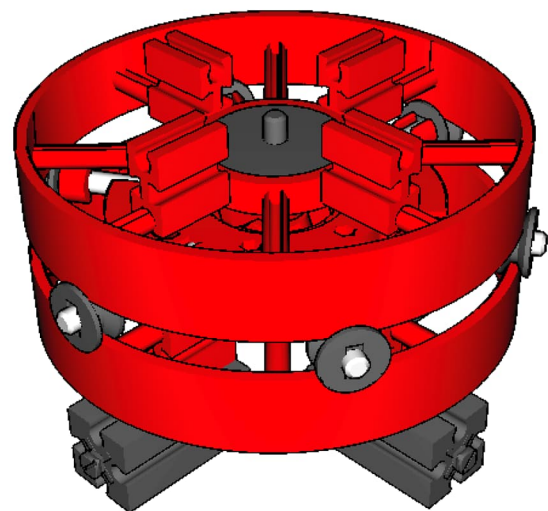
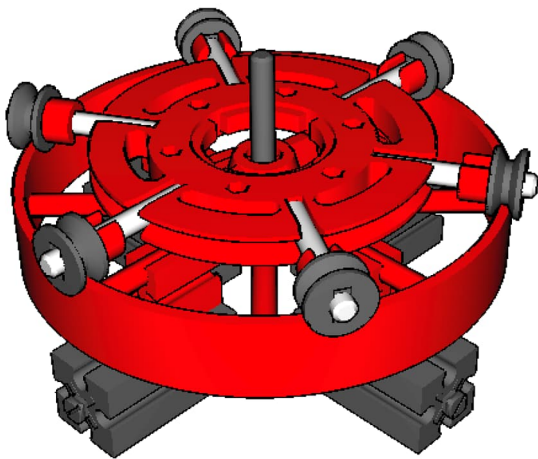


Abb. 13: Ansicht im
fischertechnik-Designer [1]



*Abb. 14: Innenansicht im
fischertechnik-Designer*

Fahrradfahrer

Den fischertechnik-Einrad-Fahrer kennen wahrscheinlich viele von euch. Da solche aber selten anzutreffen sind, habe ich mir gedacht, „bau' doch mal einen richtigen Fahrradfahrer!“ Hier ist er, solarbetrieben, und bei schönem Wetter wollte er unbedingt ins Grüne.

Auch hier danke ich Stephan Kallauch. Er hat dafür gesorgt, dass auf den Designer-Bildern der Gummiring und die Kabel zu sehen sind.

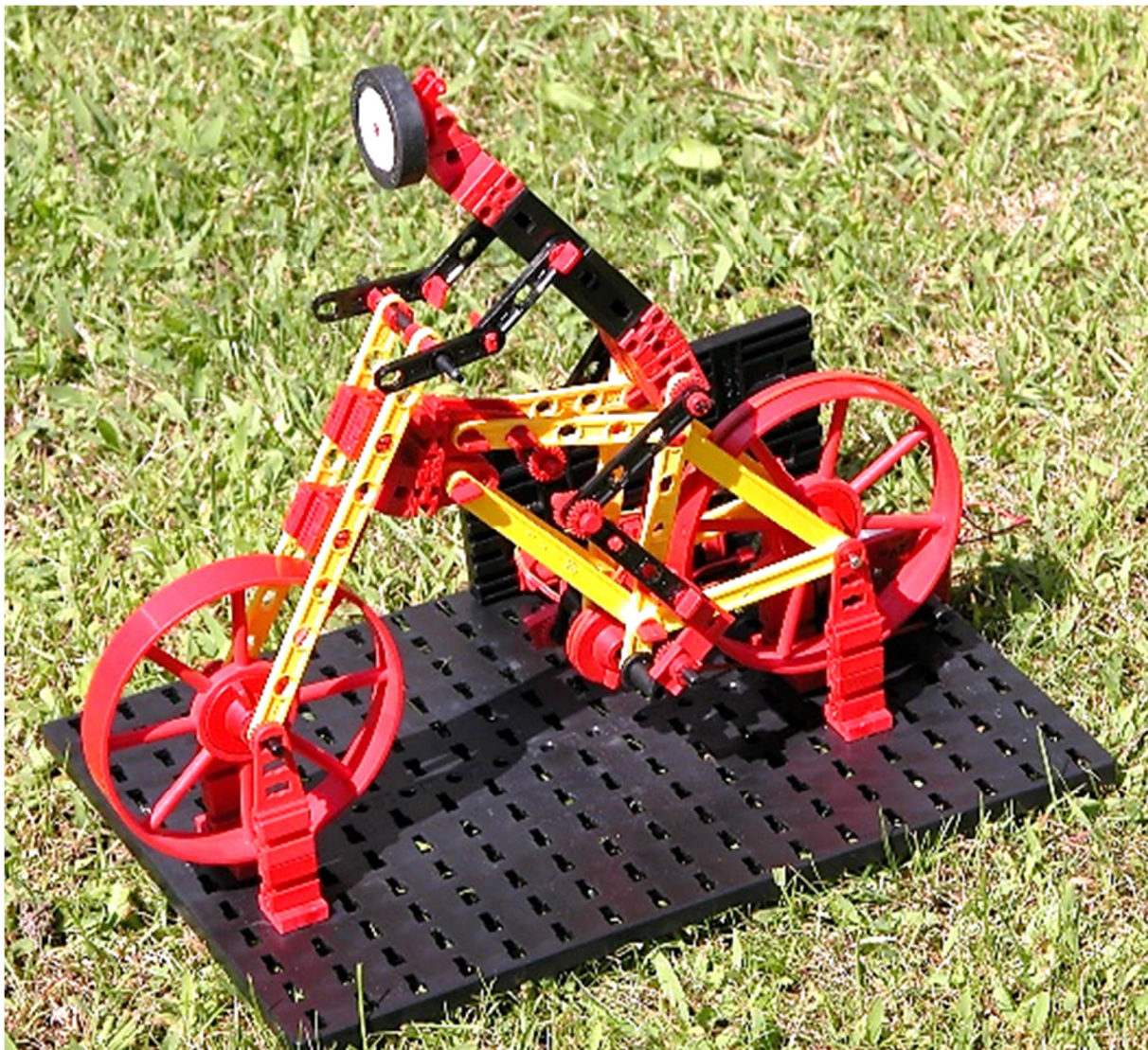


Abb. 15: Der Fahrradfahrer in seinem Element

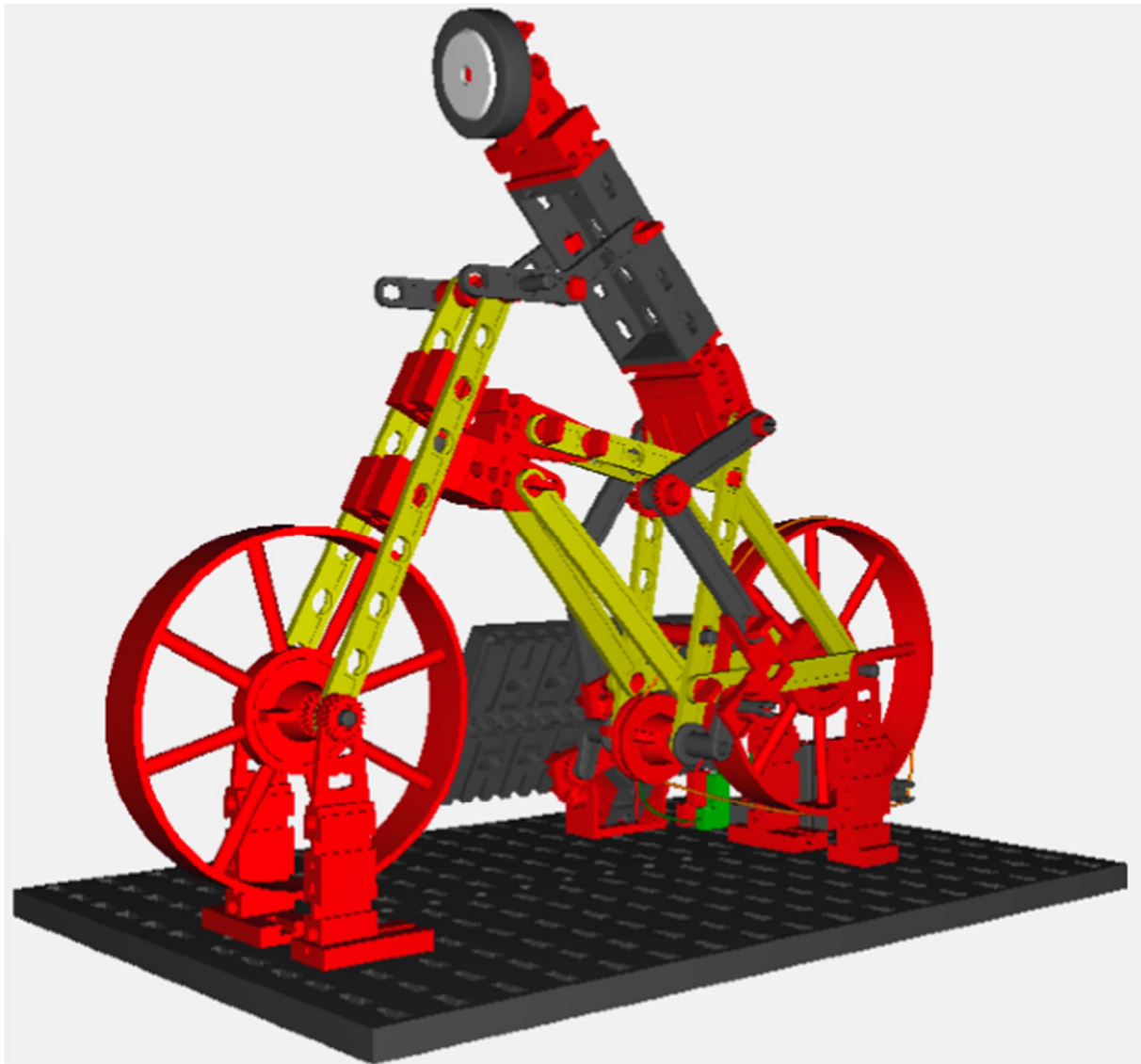


Abb. 16: fischertechnik-Designer-Ansicht [1]

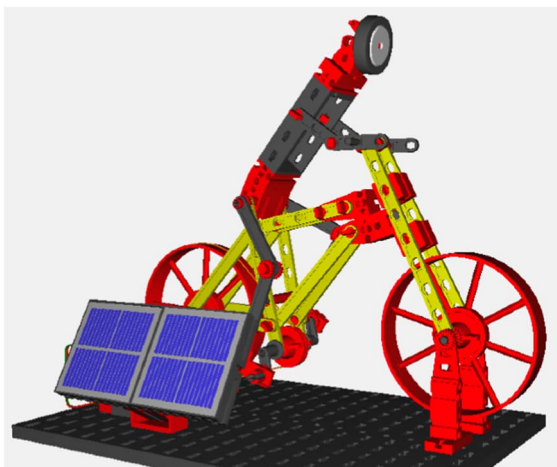


Abb. 17: Die Seite mit der Solarzelle

Quellen

- [1] Ludger Mäsing: *fischertechnik-Designer-Dateien zu den Kleinmodellen*. Auf der [Download-Seite zu dieser ft:pedia](#), 2025.
- [2] Rüdiger Riedel: *Rüttelplatte*. [ft:pedia 3/2019](#), S. 13–17.

Modell

Le Grand Éléphant

Stefan Fuss

Auf einem alten Werftgelände in Nantes tüftelt seit 20 Jahren die Künstlergruppe „Les machines de l'Île“ an den ausgefallensten Maschinen. Optisch sind sie aus den Romanen von Jules Verne entsprungen, gemischt mit da Vincis technischen Visionen und getoppt von einem Hauch von Steam Punk. Die meisten Objekte sind mechanische Tiere aus Holz und Stahl. Der Besuch der Machines ist ein Highlight für Groß und Klein; für fischertechniker sind sie außerdem ein El Dorado an ungewöhnlichen Modellideen.

Ob dreistöckiges Karussell mit Meerstieren, eine gigantische Spinne oder ein fliegender Archaeopteryx: Alle Kunstwerke in Nantes sind überlebensgroße Maschinen, motorisiert oder vom Besucher bewegt und bieten häufig eine Mitfahrtgelegenheit in oder auf dem Objekt an.

Die Maschinen haben bereits weitere Habitate außerhalb von Nantes erobert: Neben festen Ausstellungen in Calais und Toulouse war die Künstlergruppe mit Ihren

Bestien in der Vergangenheit auch schon in vielen Städten der Welt auf Tournee.

Le Grand Éléphant

Der große Elefant (Abb. 1) ist das älteste und bekannteste Objekt. Auf Balkonen, dem Sonnendeck oder in seinem Bauch lädt er bis zu 50 Gäste zu einer Tour über das Gelände ein. Dabei bewegen sich Beine, Kopf und Ohren. Wie ein echter Elefant spritzt er aus seinem beweglichen Rüssel Wasser auf die staunenden Besucher.



Abb. 1: Le Grand Éléphant vor dem Carrousel des Mondes Marins [1]

Der Elefant beeindruckt mit einer Höhe von 12 Metern, einer Länge von 21 Metern und einem Gewicht von über 48 Tonnen.

Die Körperteile werden durch hydraulische und pneumatische Zylinder angetrieben. Obwohl sich die Beine bewegen, fährt der Elefant von Elektromotoren angetrieben auf vier Rädern über das Gelände. Gesteuert wird er in einer Kabine zwischen den Vorderbeinen, ein Nachläufer beherbergt allerlei notwendige Antriebstechnik.

Die Idee

Auf der Convention 2024 packte fischertechnik viele untypische Teile in die Ausstellertüte: weiße Bausteine, Seitenteile und abgerundete Elemente in Weiß und Orange.

Bauteile, die nicht nur dazu geeignet sind die Modellfunktion in den Vordergrund stellen, sondern auch optisch ansprechende Modelle ermöglichen.

Le Grant Éléphant verbindet Stahlkonstruktionen mit mechanischen Antrieben und geschwungenen Elementen – also genau die richtige Mischung für ein Modell, das Form und Funktion vereint.

Das Modell soll eigenständig den Charme des Originals im fischertechnik-Stil wiedergeben und keine Kopie aus dem 3D-Drucker werden. Es wurde deshalb auf gemodelte Verkleidungen verzichtet. In einigen ausgewählten Bereichen musste dennoch etwas nachgeholfen werden (Abb. 2).



Abb. 2: Le Grant Éléphant en miniature

Verfügbarkeit treibt Design

Für die Rundungen am Kopf des Elefanten bietet sich der Einsatz von Winkelsteinen an. Trotz der erweiterten fischertechnik Farbpalette gibt es beim klassischen Rot die beste Bauteilauswahl. Der rote Grundton des Elefanten entstand so ganz von selbst.

Bei den Statikelementen an Vorderrad und Nachläufer wurde die Farbe an die Verfügbarkeit angepasst. Auch wenn große Teile des Nachläufers sich wie im Original in schwarz realisieren lassen – gebogene Statikteile gibt es bislang nur in Gelb und Grau.

Die Stoßzähne erstrahlen in passendem Weiß, da fischertechnik in den letzten Jahren eine große Menge verschiedener weißer Bauteile aufgelegt hat.

Orangefarbene Ohren und weiße Beine folgen ebenfalls dem Prinzip der verfügbaren Elemente.

Der Maßstab des Modells richtet sich im Großen und Ganzen nach der Größe der fischertechnik Männchen. Im Modell sollen sie die Balkone und das Sonnendeck bevölkern und bestimmen so die ungefähre Größe des Modells.

Die Rahmenbauweise von Kopf und Körper deutet die Silhouette des Elefanten an; sie lässt dem Betrachter aber gute Einblicke in das mechanische Innenleben des Elefanten.

Der Rüssel

Im Rüssel eingebaute pneumatische Zylinder ermöglichen beim Original die Bewegung in allen Richtungen. Der Nachbau war in diesem Modell die schwierigste Aufgabe. Schnell war klar, dass sich im verwendeten Maßstab leider keine Bewegung in alle Raumrichtungen umsetzen lässt.

Die wichtigste Funktion ist das Heben und Senken des Rüssels. Ein Mix aus Gelenkwürfeln, abgerundeten Bausteinen und zweier Pneumatikschläuche lassen die Auf- und Abbewegung zu. Über ein an der Oberseite verlaufendes Nylonband kann der

Rüssel durch einen Spezialservo angehoben werden (Abb. 3).

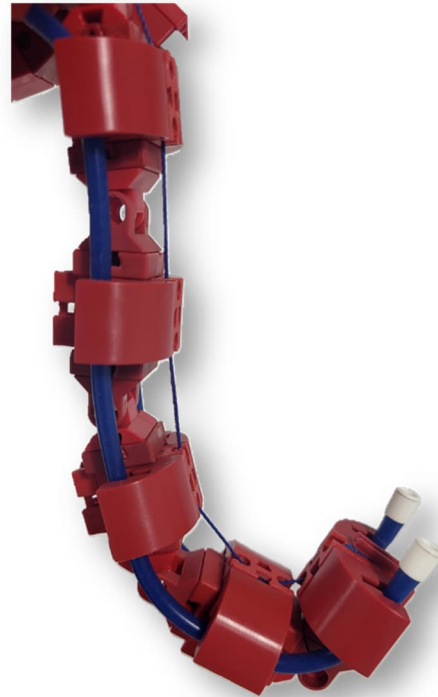


Abb. 3: Rüsselmechanik

Die Pneumatikschläuche unterstützen die stabile Bewegung des Rüssels. Sie können aber auch Rauch „spucken“: Wegen der verbauten Motoren und elektrischen Bauteile wurde das Wasserspritzen des Originals durch Bühnennebel ersetzt.

Der dafür notwendige Nebelgenerator sitzt auf dem Nachläufer. Er ist eine an fischertechnik angepasste Mini-Nebelmaschine von Castronica.

Die Beine

Die Beine bestehen im Wesentlichen aus Gelenkwürfeln und den Seitenteilen aus der Ausstellertüte 2024. Die Füße sind aus weißen und orangen abgewinkelten Bauteilen zusammengesetzt (Abb. 4).



Abb. 4: Ein Bein

Angetrieben werden die Beine über eine Pleuelstange im Bauch des Elefanten. Für eine flüssige Bewegung hebt ein zusätzlicher Seilzug den Unterschenkel mit an. Bei jedem Bein treibt ein Minimotor mit Getriebe beide Funktionen an.

Für einen realistischen Bewegungsablauf muss sich der Elefant im Passgang bewegen. Dabei hebt er abwechselnd die Beine

einer Seite an, sodass immer mindestens zwei Beine am Boden sind.

Zusätzlich sollen im Stillstand alle vier Beine den Boden berühren. Dazu muss jedes Bein durch einen eigenen Motor angetrieben werden. Die Ruheposition „Bein auf dem Boden“ wird durch einen Minitaster an die Steuerung übertragen.

Seilzug an Maul, Ohren und Kopf

Alle drei Funktionen werden durch einfache Seilzüge realisiert. Dabei werden die verwendeten Nylonfäden vom Kopf durch den Körper des Elefanten in den Nachläufer geführt, wo sie von Servos gesteuert werden. Da das Maul sich durch die Schwerkraft öffnet, genügt hier ein einzelner Seilzug zum Schließen.

Die Ohren werden über zwei Hebel von einem Gummiband an den Kopf gedrückt. Der Seilzug zieht in die Gegenrichtung und stellt so die Ohren auf.

Der Kopf lässt sich von links nach rechts bewegen. Dabei zieht ein Haushaltsgummi den Kopf auf die linke Seite, ein Seilzug wirkt dem Gummi entgegen.

Die Servos

Die vielen Funktionen im Kopf des Elefanten benötigen vier Servos im Nachläufer. Mit klassischen fischertechnik-Servos lassen sich diese im Modell nicht realisieren. Der aufgesteckte Steuerhebel ist nicht stabil genug. Er springt bei Belastung von der Achse. Da diese Servos zudem kontinuierlich ihre Position korrigieren, haben sie einen hohen Stromverbrauch.

Servos setzen sich aus einem Motor, einem Potentiometer zur Winkelmessung und der Steuerelektronik zusammen. In der Regel sind alle drei Komponenten in einem Gehäuse verbaut. Die Stromversorgung und die Vorgabe des Stellwinkels erfolgt über einen dreipoligen Stecker.

In der RC-Serie Anfang der 1980er Jahre hatte fischertechnik Servos ([30275](#)) im Ein-

satz, bei denen nur das Potentiometer und der Motor im Gehäuse enthalten waren (Abb. 5). Die Steuerung erfolgte damals extern über den RC-Empfänger ([3246X](#)).

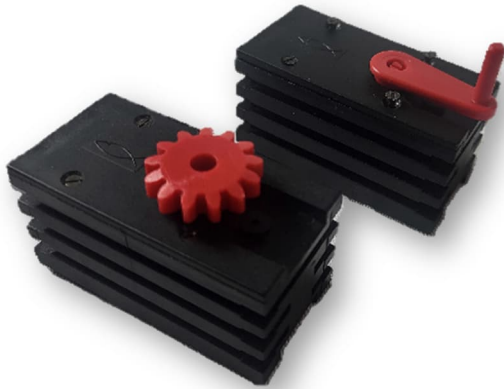


Abb. 5: RC-Servos von 1983

Im Modell wird das gleiche Prinzip verwendet. Ein S-Motor mit Getriebe treibt eine Spule an, auf der der Nylonfaden des Seilzuges aufgewickelt wird. An der Spulenchse wird zusätzlich ein externes, lineares Potentiometer befestigt. Es liefert der Regelung den Stellwinkel (Abb. 6). Ein softwarebasierter PID-Regler macht aus dem S-Motor einen Servo [2].

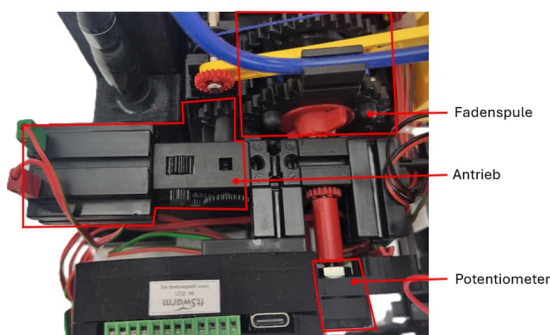


Abb. 6: Selbstbauservo

Da das Getriebe den Servo im Ruhezustand blockiert, wird die Regelung ausgeschaltet, sobald der S-Motor seine Sollposition erreicht hat. Der Stromverbrauch des Selbst-

bauservos beträgt so nur noch einen Bruchteil des fischertechnik-Bauteils.

Der Antrieb

Wie im Original fährt der Elefant auf Reifen. Für ein flüssiges Fahrgefühl werden die Hinterräder durch zwei Encodermotoren angetrieben.

Unter der Fahrerkabine steuert ein normaler fischertechnik-Servo die Stellung der beiden Vorderräder.

Die Stromversorgung

Für das Modell wären zwei originale Akku-Sets von fischertechnik notwendig gewesen. Aus Platzgründen wurden diese durch einen handelsüblichen 7,4 V-RC-Akku mit 3000 mAh ersetzt.

Im Gegensatz zum fischertechnik-Akku-Set hat der RC-Akku keine Schutzschaltung. Bei einem Kurzschluss liefert der Akku bis zu 45 A im Peak – mehr als genug, um die Kabel in Rauch aufgehen zu lassen. Eine PPTC-Sicherung limitiert den Strom und schaltet bei Überlast ab.

Die Steuerung

Zehn Motoren, ein klassischer Servo, die Nebelmaschine, vier Taster und vier Potentiometer – im Elefanten gibt es einige Steueraufgaben.

Die Steuerung übernimmt ein Swarm aus vier Controllern. Ein ftSwarmRS im Hals des Elefanten steuert den klassischen Servo an der Vorderachse [3]. Im Nachläufer kümmert sich ein ftSwarmXL um Beine und Rauchgenerator, ein ftSwarmRC ist für die Selbstbauservos zuständig.¹

Mit einem ftSwarmControl lässt sich das Modell bequem per Fernsteuerung bedienen. Die Programmierung der vier Controller ist sehr einfach. Sie sind über WLAN miteinander verbunden, die ftSwarm-Firm-

¹ Der ftSwarmRC ist derzeit noch in Entwicklung. Er kann sowohl alte RC-Servos als auch den

Radantrieb von 1983 steuern. Daher kann er als Ersatz für den RC-Empfänger eingesetzt werden.

ware kümmert sich um die Kommunikation zwischen den Controllern.

Rüssel, Rauchgenerator, Ohren und Hals werden über Taster am ftSwarmControl gesteuert. Eine klassische Programmierung ist für deren Funktion nicht notwendig. In der ftSwarm-Firmware können einfache Funktionen – z. B. ein Taster steuert einen Motor – im Bereich „Remote Control“ konfiguriert werden (Listing 1).

***** Remote Control #1 *****

```
( 1) F1.TriggerUp -> Ruessel.setPosition(75)
( 2) F1.TriggerDown -> Ruessel.setPosition(-35)
( 3) F2.TriggerUp -> Rauch.setSpeed(100)
( 4) F2.TriggerDown -> Rauch.setSpeed(0)
( 5) S1.TriggerUp -> Ohren.setPosition(50)
( 6) S1.TriggerDown -> Ohren.setPosition(0)
( 7) S2.TriggerUp -> Hals.setPosition(-60)
( 8) S3.TriggerUp -> Hals.setPosition(0)
( 9) S4.TriggerUp -> Hals.setPosition(60)
(10) JOY1FB.ChangeValue -> Antrieb1.setSpeed()
(11) JOY1FB.ChangeValue -> Antrieb2.setSpeed()
(11) JOY1LR.ChangeValue -> Lenkung.setPosition()
```

Listing 1: Die meisten Funktionen lassen sich ohne Programmierung konfigurieren²

Wird z. B. Taster F1 gedrückt, so setzt die Firmware automatisch beim Servo „Ruessel“ die Sollposition 75 und der Rüssel hebt sich an. Lässt man den Taster F1 wieder los, so wird die Servoposition -35 gesetzt (Zeile 1 und 2 in Listing 1).

Ähnlich einfach lässt sich der Antrieb lösen. Hier reagiert die Firmware auf die Veränderung der Joystickposition und nutzt den Sensorwert für die Ansteuerung der Motoren und Servos (drei letzte Zeilen in Listing 1).

Die Steuerung der Beine ist etwas komplexer. Fährt der Elefant los, sollen die Beine sich abwechselnd im Passgang bewe-

gen. Dazu müssen die Motoren zeitversetzt eingeschaltet werden (Listing 2).

```
void loop(void) {

    // warte auf das Fahrsignal
    while (joy1fb->getValue() <= 0) delay(25);

    // nun die Beine zeitversetzt starten
    for (uint8_t i=0; i<3; i++) {
        bein[i]->setSpeed(33);
        delay(250);
    }

    // warte jetzt auf das Stoppsignal
    while (joy1fb->getValue() > 0) delay(25);

    // nun die Beine anhalten
    uint8_t gestoppt = 0;
    while (gestoppt < 4) {
        gestoppt = 0;
        for (uint8_t i=0; i<3; i++) {
            // Bein stoppen?
            if (taster[i]->triggerUp())
                bein[i]->setSpeed(0);
            // Zählen, ob alle Beine gestoppt wurden
            gestoppt = gestoppt +
                taster[i]->getValue();
        }
        delay(25);
    }
}
```

Listing 2: Steuerung der Beine

Bleibt der Elefant stehen, dürfen die Motoren erst abgeschaltet werden, wenn der zugehörige Taster die passende Position meldet.

Quellen

- [1] [Les Machines de L'île](#), Parc des Chantiers, Boulevard Léon Bureau, 44200 Nantes
- [2] Dirk Fox: *PID-Regler – eine experimentelle Einführung*. [ft:pedia 4/2021](#), S. 62–72.
- [3] Christian Bergschneider, Stefan Fuss: *ftSwarm – Das Prinzip Heinkelmannchen*. [ft:pedia 2/2021](#), S. 107–111.

² Die hier dargestellte Syntax entspricht dem aktuellen Entwicklungszweig der ftSwarm-Firmware. Sie wird voraussichtlich in Q1/26 released.

Tipps & Tricks

Die fischertechnik-Leiter

Ludger Mäsing

Die fischertechnik-Leiter ist schon ein besonders interessantes Bauteil. Sie wird so gut wie gar nicht in irgendwelchen Modellen verbaut.

Überblick

Die Leiter (fischerform [3115X](#) bzw. fischertechnik [130925](#)) gibt bzw. gab es in drei verschiedenen Varianten:

1. Seit 1983 gab es laut der ft-Datenbank bereits eine rote Leiter von fischerform.
2. Dann kam die graue (in Abb. 1 im Hintergrund).
3. Später gab es noch eine in Grau/Silber (in Abb. 1 im Vordergrund).

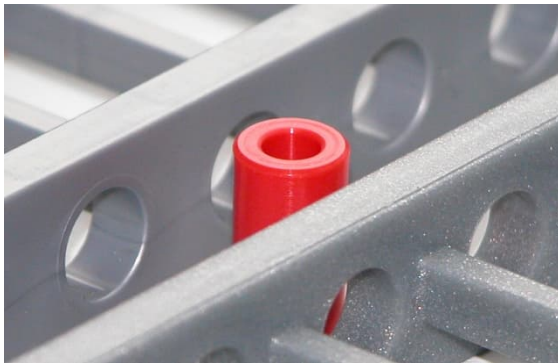


Abb. 1: Vergleich zweier der Leitern

Die vordere hat einen Glitzereffekt, während die hintere einfarbig grau ist. Die rote Hülse habe ich dort nur hingestellt, damit die Kamera einen Fokus hat. Es war nicht einfach, ohne die Hülse ein gutes Foto zu machen.

Verbindung von Leitern

Starre Verbindung

Wie verbindet man nun die Leiter sinnvoll mit den „normalen“ fischertechnik-Bausteinen? Ganz einfach: Zwei Bausteine 15 und

ein Baustein 5 passen genau zwischen die Seitenwände. Dann verbaut man nur noch drei Federnocken, schiebt diese auf einen oder auf beide Bausteine und klemmt die Leiter darauf. Die Federnocken passen genau zwischen die Leitersprossen, sodass diese einen sehr guten Halt hat.



Abb. 2: Verbindung mit Bausteinen

Möchte man zwei Leitern miteinander verbinden, so ist dieses auch recht einfach möglich. Die Sprossen der Leitern sind breiter als die fischertechnik-Achsen. Zwei I-Streben, ein paar Achsen und für jede Bohrung der Strebe vier Klemmbuchsen 5 sind alles, was man benötigt. Das ergibt eine solide Verbindung.



Abb. 3: Verbindung zweier Leitern

Gelenkige Verbindung



Abb. 4: Leitern mit Gelenk, gestreckt

Manchmal benötigt man vielleicht keine starre, sondern eine bewegliche Verbindung. Auch das ist möglich. Es genügen jeweils zwei Verbindungsstücke 15, zwei Klemmhülsen und einen Baustein $5 \times 15 \times 30$ mit drei Nuten. Die Klemmhülsen passen von der Länge her gerade so durch die Löcher der Leiterseitenwand, sodass man noch ein Verbindungsstück 15 aufschieben kann. Das hält super!

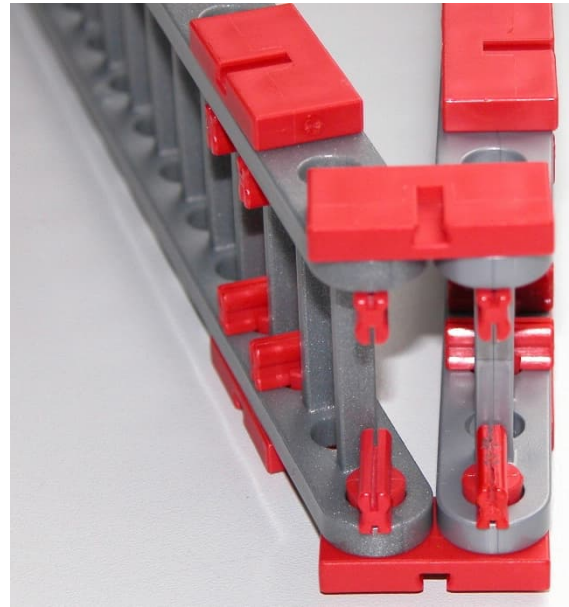


Abb. 5: Leitern mit Gelenk, abgelenkt

Ein erster Versuch: Die mittlere rote Baugruppe bildet das Gelenk, das kann man hier gut erkennen. An den beiden äußeren, die

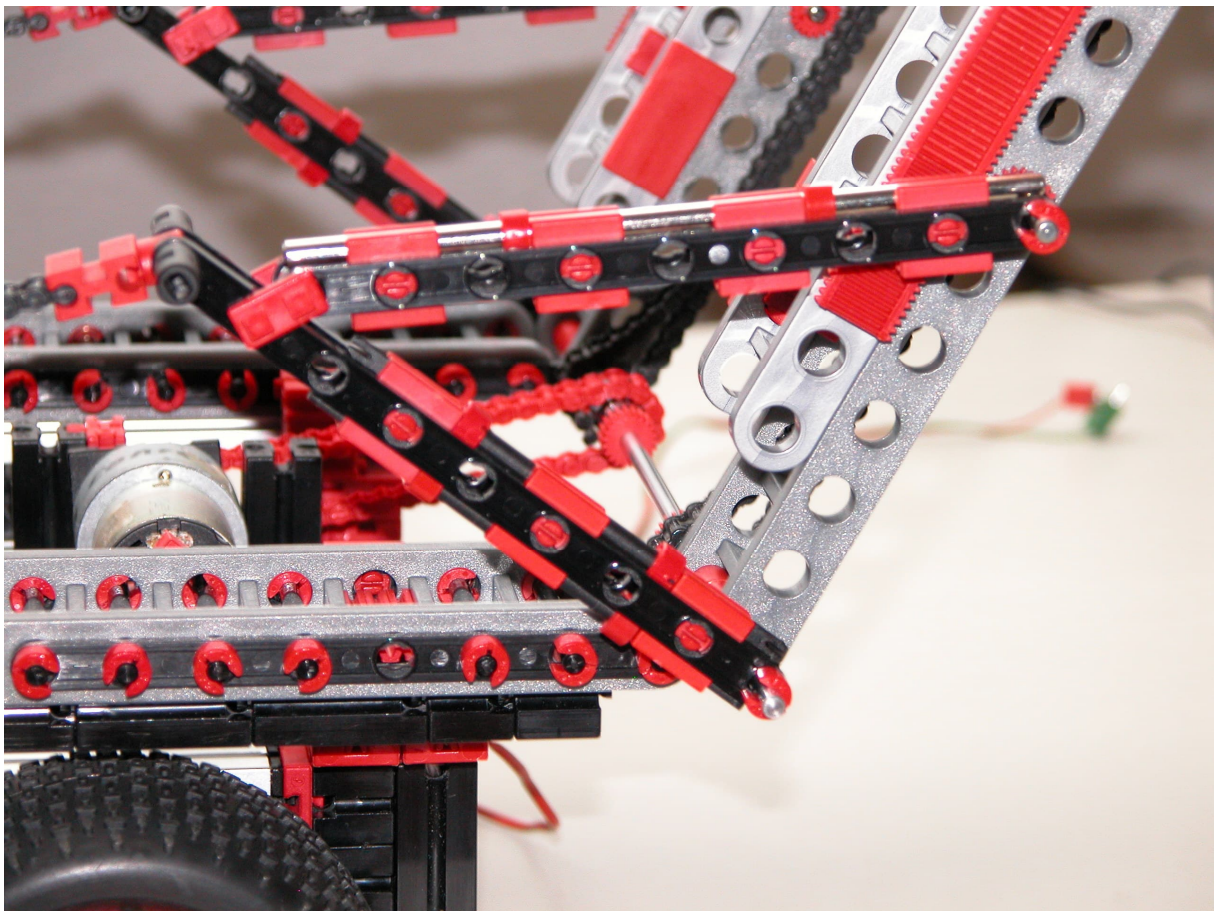


Abb. 6: Gelenk an einem Anhänger

hier nicht unbedingt erforderlich sind, können Anbauten vorgenommen werden. Man kann aber auch eine I-Strebe an einem Leiterende überstehen lassen und dort die zweite Leiter anbringen.

Leitern aufeinander gleiten lassen



Abb. 7: Leitern gleiten aufeinander (1)

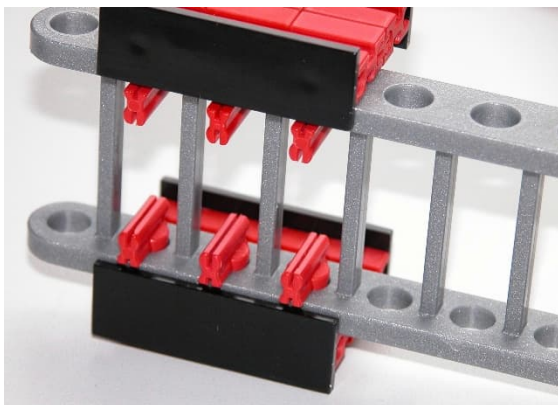


Abb. 8: Leitern gleiten aufeinander (2)

Das Gleiten zweier Leitern aufeinander funktioniert auch. Dazu benötigt man zwölf Bausteine 7,5, sechs Klemmhülsen, sechs

Verbindungsstücke 15, vier Bauplatten 15×45 und zwei Verbindungsstücke 45. Der Aufbau ist in Abb. 7 gut zu erkennen. Diese Baugruppe wird an der unteren Leiter befestigt. Die obere gleitet dann hin und her.

Drehbewegungen übertragen

Wem das an Bewegung noch nicht ausreicht, dem kann geholfen werden. Es ist z. B. auch möglich, eine Mechanik an der Seite der Leiter anzubringen (Abb. 9).

Von rechts oben vom Kardangelenck kommt eine Drehbewegung an. Diese wird zuerst an der oberen Seite entlanggeführt, dann durch die Leiter hindurch und letztendlich ein Stück an der unteren Seite entlang. Die Befestigung der Bausteine 15 mit Bohrung geschieht ebenfalls mit Klemmhülsen und Verbindungsstücken 15. Die Durchführung durch die Leiter geht mit Hilfe von zwei Rastachsen mit Kegelzahnrad, zwei Hülsen 15 und einer Rastkupplung.

Abb. 10 und 11 zeigen eine geänderte Version der Durchführung. Die Klemmbuchse 5, die sich in der Leiterseitenwand befindet, sollte sich hier leicht drehen lassen. Es gibt bei den Klemmbuchsen kleine Unterschiede. Manche machen ihrem Namen alle Ehre und klemmen wirklich. Das soll aber hier nicht sein. Eine Rastachse mit Kegelzahnrad, zwei Riegelscheiben und zwei Klemmbuchsen 5, das Ganze zwei Mal, die Rastkupplung dazu werden verbaut.

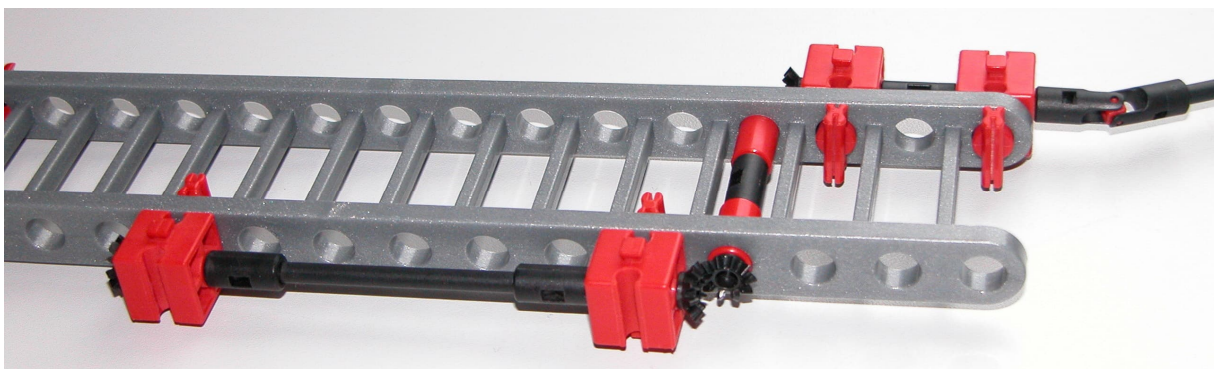


Abb. 9: Drehmoment-Übertragung

Die vier Riegelscheiben und die sichtbaren Klemmbuchsen 5 sollten bei diesem Beispiel sehr fest sitzen.

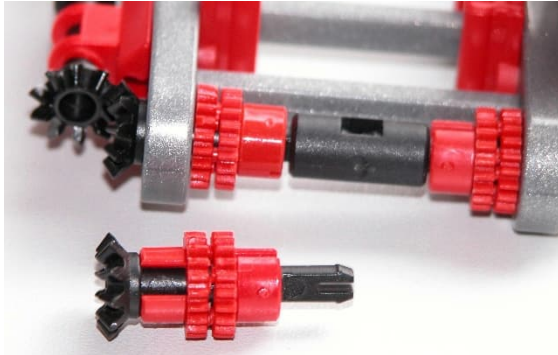


Abb. 10: Durchführung durch die Leiter

Der Aufbau kann noch dezenter gestaltet werden: Die Bausteine 15 mit Bohrung wurden durch Gelenkwürfel-Zungen ersetzt. Das sieht auch etwas besser aus. Weil die Zungen so nicht gut ins Raster passen, muss man hier Abstandsringe und Klemmbuchsen einbauen.

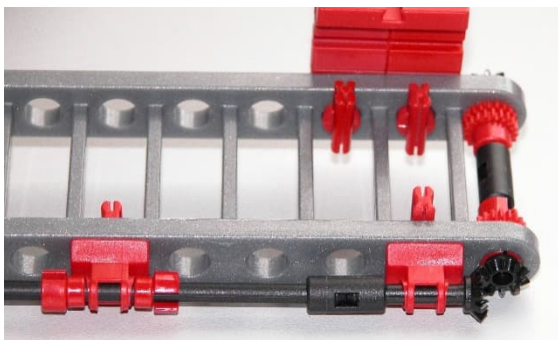


Abb. 11: Variation der Drehübertragung

Gleiten durch Drehbewegungen

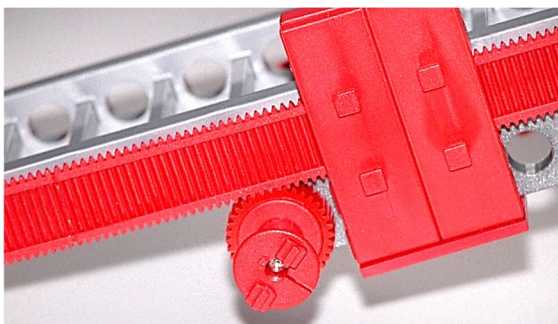


Abb. 12: Gleiten mit Zahnrad

Abb. 12 zeigt eine Kombination aus den vorher beschriebenen Baugruppen. Zwei

Leitern gleiten übereinander. Mit Hilfe des Zahnrades der Seiltrommel schiebt sich die obere Leiter vor und zurück.

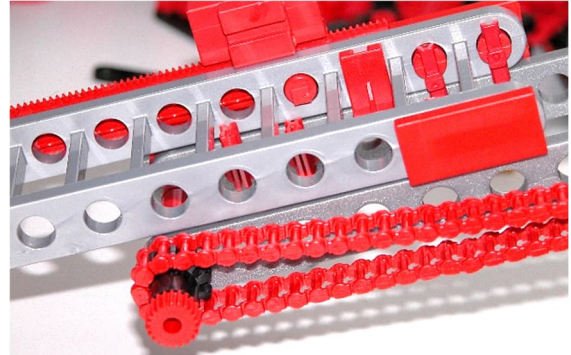


Abb. 13: Gleiten mit Kette

Anwendungen dafür gibt es natürlich auch:



Abb. 14: Anhänger

Dieses Modell war der Anhänger meines „Riesen-LKW“ [1].



Abb. 15: Riesen-LKW

Es geht aber auch noch anders: Abb. 17, 19 und 20 zeigen einen Bahnübergang mit Schranke unter Verwendung der Leitern.

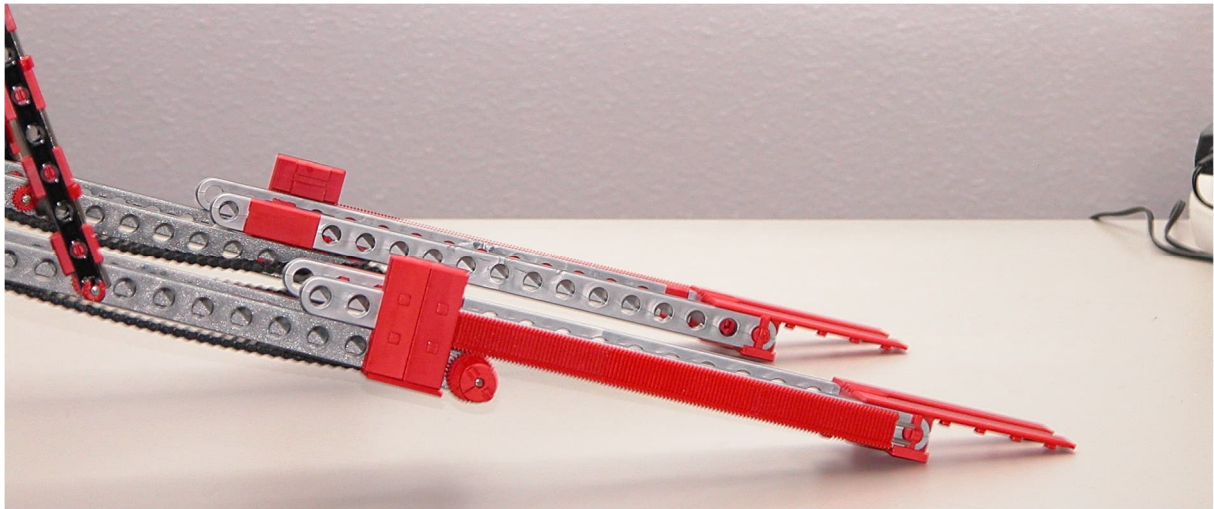


Abb. 16: Rampe des Anhängers



Abb. 17: Schranke geöffnet

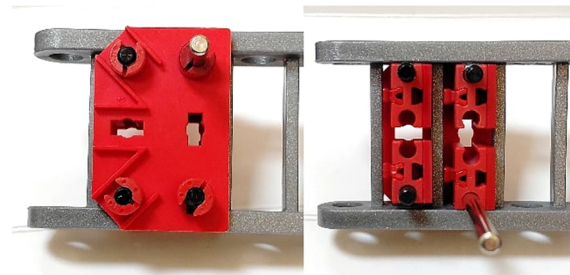


Abb. 18: Schrankenbefestigung

Die Halterung der Leiter ist schnell konstruiert. Dazu benötigt man lediglich zwei Knotenplatten, vier Federnocken, vier Bauplatten 15×15 , vier Bausteine 7,5, acht Klemmbuchsen 5, eine Metallachse 60 oder länger, drei K-Achsen 30.

Das waren nur einige Beispiele dafür, wie man die fischertechnik-Leiter nutzen kann. Euch fallen bestimmt noch weitere ein.

Quellen

- [1] Ludger Mäsing: *Riesen-LKW*. Im [Bilderpool der ft-Community](#), 2012.



Abb. 19: Schranke geschlossen

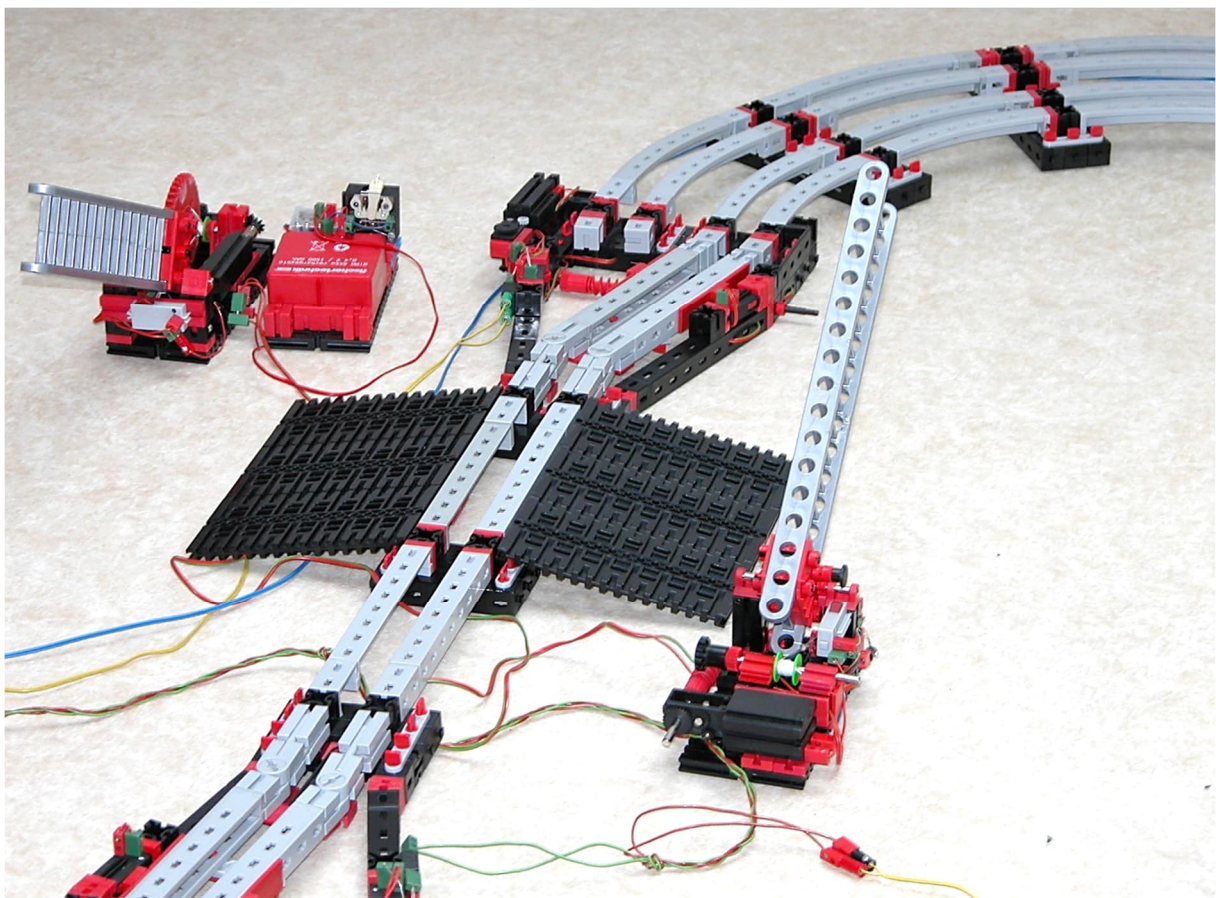


Abb. 20: Bahnübergang komplett mit Gleisen

Tipps & Tricks

Federn und Dämpfer für fischertechnik-Modelle

Ludger Mäsing

Wie kann man große und schwere Modelle gut federn? Verwendet man die Original-fischertechnik-Federn, Blattfedern oder Dämpfer? Dämpfer werden im RC Modellbau eingesetzt und leisten dort einiges. Es gibt sie als Reibungsdämpfer oder als Öldruckstoßdämpfer. Beide Arten werde ich vorstellen.

Zur Einführung

Bereits bei meinem „Riesen-LKW“ (Abb. 1 [1]) hatte ich das Problem, dass die blauen fischertechnik-Federn irgendwie nicht ausreichten. Damals hatte ich pro Seite zwei übereinander und das Ganze fünffach in Reihe doppelt aufgebaut. Ich benötigte also 40 Federn nur für die hinteren Achsen. Das kann man auf Abb. 2 gut erkennen. So hatte ich eine Federung, die relativ gut ausreichte.

Aber irgendwie war das doch noch nicht ganz so toll. Heute, nachdem der LKW leider wieder in seine Einzelteile zerlegt ist, ist mir zufällig mein altes RC-Auto (Tamiya

„Boomerang“) in die Hände gefallen. Dieses hatte ich mir im Januar 1989 zugelegt.

In dem, wie Tamiya schreibt, „Hochleistungsgeländefahrzeug mit Allradantrieb“ sind Öldruckstoßdämpfer eingebaut. Es hat eine allseitige Einzelradaufhängung über Doppelquerlenker mit einem vorderen, doppelwirkenden und hinten zwei einzelnen Öldruckstoßdämpfern. Diese Dämpfer sind leider nicht einzeln erhältlich. Ich habe sie einmal ausgebaut und etwas nachgemessen.



Abb. 1: Riesen-LKW [1]

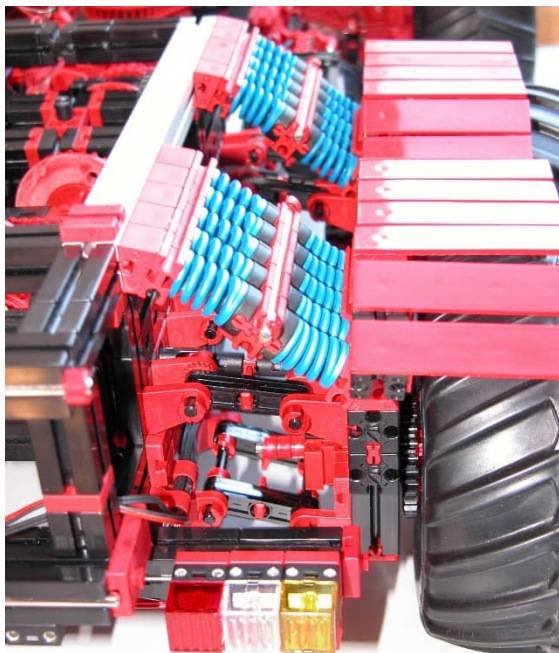


Abb. 2: Federung der Hinterachsen
des Riesen-LKW

Der vordere Dämpfer ist ca. 8,3 cm lang (Loch zu Loch) und hat einen Hub von 1,7 cm. Der Federweg kann durch entfernen von X3 und X4 auf 2,5 cm erweitert werden. Die hinteren sind ca. 10,3 cm lang (Loch zu Loch) und haben einen Hub von 2,4 cm. Der Federweg kann hier durch entfernen von T4 und T5 auf 3,6 cm erweitert werden. Somit braucht dann nicht so viel Kraft aufgewendet zu werden. Die Bohrung ist 4 mm weit, passt also gut ins fischer-technik-Raster.

Damit ihr Euch vorstellen könnt, wie diese Dämpfer funktionieren, habe ich vier Bilder (Abb. 3 bis 6) aus einer späteren Bauanleitung (aus dem Jahr 2008) angefügt. Die Firma Tamiya hat mir freundlicherweise erlaubt, die Bauschritte mit den Öldruckstoßdämpfern hier zu zeigen. Dafür möchte ich mich recht herzlich bedanken.

Wichtig ist: Das Öl muss oft separat dazu gekauft werden.

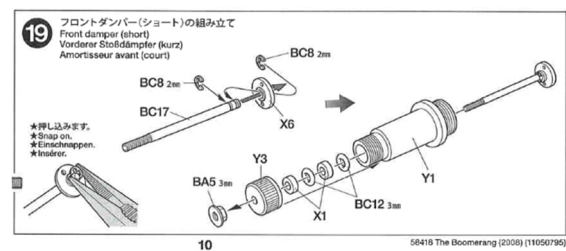


Abb. 3: Tamiya-Anleitung (Abb. 19)

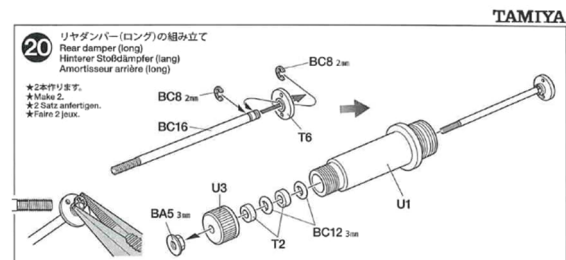


Abb. 4: Tamiya-Anleitung (Abb. 20)

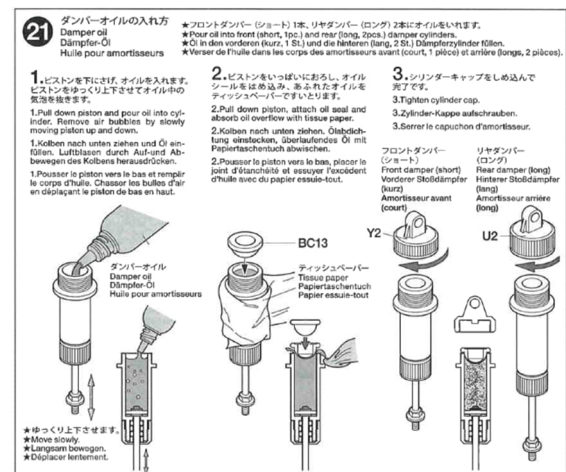


Abb. 5: Tamiya-Anleitung (Abb. 21)

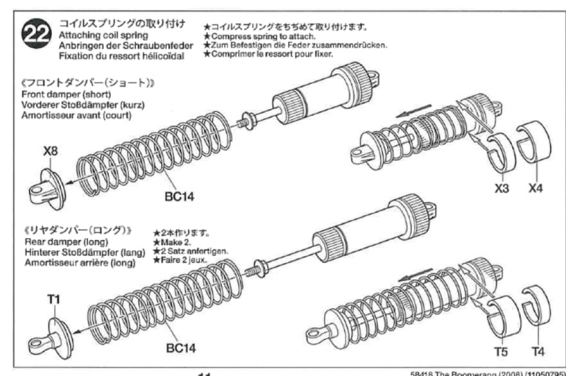


Abb. 6: Tamiya-Anleitung (Abb. 22)

Bei meinem RC-Modell „Boomerang“ habe ich damals „Tamiya Damper Oil Soft“ genutzt. Ein passendes Öl gibt es z. B. hier:

- Quelle:
<https://tamico.de/Tamiya-Stossdaempferoel-Soft-9ml-400>
- Preis:
20 ml = 2,99€

Bei Interesse kann man unter [2] die komplette Bauanleitung des Racers „Boomerang“ einsehen.

Verschiedene Dämpfer

Es folgen einige Beispiele für Aluminium-Dämpfer in diversen Längen. Sie machen optisch schon einen guten Eindruck. Ob sie gut funktionieren, kann ich nicht sagen, weil ich noch keinen davon getestet habe. Der Unterschied liegt nicht nur im Material. Beim ersten Beispiel ist die Feder im Kolben verbaut. Ich denke, dass so auch nur ein fester Wert für die Federung existiert.

Bei den Dämpfern von Tamiya konnte man den Federweg durch Entfernen der Ringe ein wenig variieren.

ZERO RC



Abb. 7: Dämpfer von ZERO RC



Abb. 8: Explosionsansicht

- Außendurchmesser:
9 mm
- Durchmesser Endloch:
2,5 mm
- Endbreite:
5,5 mm
- Preis für vier Stück (Lieferung ohne Öl!):
9,45 € – 10,19 €
- Länge (Loch zu Loch):
53, 59, 65, 71, 77 mm
- Hub:
12, 14, 17, 19, 22 mm
- Gewicht:
23, 25, 27, 28, 30 gr.
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008001245845.html>

Die Verbindung zum fischertechnik-System könnte mittels passender gerändelter Gewindeinsätze geschehen (wie sie z. B. bei den fischertechnik-Gewindebausteinen Anwendung finden). Firma GHW bietet diese auch in M 3, M 2,5 u. a. an.



Abb. 9: Dämpfer mit außenliegenden Federn

Bei der Variante in Abb. 9 liegt die Feder außen. Die Federvorspannung kann durch einen Stellring stufenlos eingestellt werden. Diese Dämpfer sind laut WWW schon fertig mit Öl befüllt.

Reely – Conrad

1/10 Aluminium Gewinde-Öldruckstoßdämpfer Rot-Metallic mit schwarzen Federn:

- Länge:
90 mm
- Hub:
11,3 mm
- Preis für vier Stück:
18,99 €
- Bezugsquelle:
<https://www.conrad.de/de/p/reely-1-10-aluminiumoeldruckstosssdaempfer-rot-metallic-mit-federn-schwarz-90-mm-4st-2469813.html>



Abb. 10: INJORA Aluminium Stoßdämpfer mit zwei Sätzen Federn unterschiedlicher Härte, Schrauben und Unterlegscheiben; Lieferung ohne Öl

- Außendurchmesser:
18,0 mm

- Durchmesser Endloch:
3,0 mm
- Preis zwei bzw. vier 4 Stück:
16,99 € bzw. 30,84 €
- Länge (Loch zu Loch):
90, 100, 110, 120 mm
- Hub:
17, 22, 27, 32 mm
- Bezugsquelle:
<https://de.aliexpress.com/item/1005004599082114.html>

JoYonDa/NEW ENRON



Abb. 11: JoYonDa/NEW ENRON-Ölstoßdämpfer

So wie es in Abb. 11 aussieht, werden die Kugelköpfe und Schrauben auch hier separat mitgeliefert. Auf Grund dessen könnte die Bohrung passend für fischertechnik-Achsen sein. Laut WWW sind die Dämpfer bereits mit Öl befüllt.

- Preis für zwei Stück:
ab 5,27 €
- Außendurchmesser:
12 mm
- Durchmesser Endloch:
vermutlich 4 mm
- Länge (Loch zu Loch):
75, 80, 90 mm
- Hub:
18, 20, 22 mm
- Bezugsquelle:
<https://de.aliexpress.com/item/1005006688951993.html>



Abb. 12: Kugelköpfe

Einige der Dämpfer haben auf einer Seite eine Aufnahme für einen Kugelkopf wie in Abb. 12. Für diese gibt es eine wunderbare Erweiterung. Diese Teile würden gut in einen Gewindebaustein passen. In manchen Fällen ist die Bohrung des Dämpfers aber so groß, dass eine fischertechnik-Achse direkt hineinpasst.

Man suche im WWW nach *4 mm Kugelkopf M4* oder der Tamiya-Artikelnnummer 319804831

- Preis 2 Stück:
ab ca. 3,50 €

ZuoLan



Abb. 13: ZuoLan Aluminium Öl-Dämpfer
(werden ohne Silicon-Öl geliefert!)

Auch bei den Dämpfern in Abb. 13 werden die Kugelköpfe und Schrauben separat mitgeliefert. Aus diesem Grund könnte die

Bohrung passend für fischertechnik-Achsen sein.

- Preis für vier Stück:
14,99 €
- Länge (Loch zu Loch):
55 mm
- Hub:
10 mm
- Durchmesser Endloch:
vermutlich 4 mm
- Bezugsquelle:
<https://www.amazon.de/ZuoLan-Aluminum-Sto%C3%9Fd%C3%A4mpfer-Absorber-D%C3%A4mpfer/dp/B0FBGN2J82/260-9610391-9957255>

INJORA



Abb. 14: INJORA Federstoßdämpfer mit
innenliegender Feder (1)

Abb. 14 und 15 zeigen ein Komplettsset mit sechs Federn und Befestigungsmaterial. Öl ist nicht enthalten.

- Preis für zwei Stück:
10,73 €
- Endloch mit Alu-Kugelkopf:
3,0 mm

- Außendurchmesser:
14,0 mm
- Länge (Loch zu Loch):
90, 100, 110, 120 mm
- Hub:
keine Angaben
- Bezugsquelle:
<https://de.aliexpress.com/item/4000308585052.html>

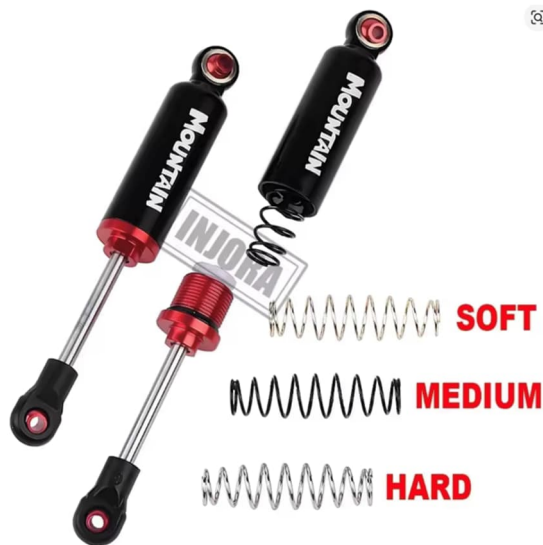


Abb. 15: INJORA Federstoßdämpfer mit innenliegender Feder (2)

JVRWSC

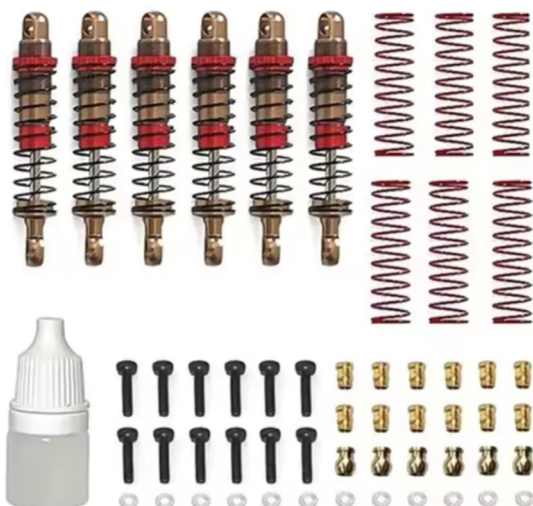


Abb. 16: Großes- Öl-Stoßdämpfer-Set

Abb. 16 zeigt ein Beispiel für ein großes Öl-Stoßdämpfer-Komplett-Set von JVRWSC (möglicherweise passen die mitgelieferten

Rändelmuttern in einen Baustein 15 mit Ansenkung):

- Preis für sechs Stück mit Federn, Befestigungsmaterial und Öl:
15,14 € – 52,76 €
- Länge (Loch zu Loch):
keine Angaben
- Hub:
keine Angaben
Bohrung: keine Angaben
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008682993548.html>

TAMIYA



Abb. 17: Tamiya Ersatzteil DT-01 Stoßdämpfer-Set

Tamiya bietet ein Komplett-Set mit je zwei kurzen und zwei längeren Dämpfern zu einem recht günstigen Preis an (Abb. 17). Zum Einstieg und Testen ist das, denke ich, eine gute Sache.

Bei diesen Dämpfern handelt es sich um Reibungsdämpfer, es wird also kein Öl benötigt. Das Material ist Kunststoff mit Metallfeder. Dieses Dämpfer-Set habe ich erhalten und kann deswegen nähere Angaben machen. Die Bohrungen auf beiden Seiten sind 4 mm weit und passen somit wirklich gut ins fischertechnik-Raster. Da die Metallstange recht dünn ist, wirken die Dämpfer etwas instabil. Nicht aber so, dass ich sagen müsste, dass diese nicht geeignet sind. Ich bin damit gut zufrieden.

- Durchmesser Endloch:
4,0 mm
- Außendurchmesser:
ca. 16 mm
- Länge (Loch zu Loch):
ca. 71 bzw. 92 mm
- Hub:
ca. 14 bzw. 20 mm
- Kraftaufwand bis Maximum:
ca. 900 bzw. 1.000 gr.
- Preis für zwei mal zwei Stück:
10,99 €
- Bezugsquelle:
<https://www.modellbau-seidel.de/index.php?firma=Tamiya&best=9415375>

Dämpfer in fischertechnik-Modellen

Aufbau mit Reibungsdämpfern

Abb. 18 zeigt einen ersten Test mit den gerade beschriebenen Tamiya-Reibungsdämpfern. Ein Dämpfer reicht bei dieser Konfiguration bei weitem nicht aus, obwohl

er 1.000 g „Kraftaufwand“ bis Maximum braucht. Der komplette Federweg wird in dieser Baugruppe nicht erreicht, weil die Hebel zu kurz sind. Möchte man eine gute Federung haben, sollten die Dämpfer schon außen angebracht werden (ähnlich wie auf dem zweiten Bild beim großen LKW), und das auf jeder Seite, oder man verwendet die Öldruckstoßdämpfer. Der Baustein 15 hat kein bauliches Erfordernis; er dient lediglich dazu, die Baugruppe etwas anzuheben. Die Räder würden außen angebaut werden.

Abb. 19 zeigt den Aufbau von der Seite. Es gibt bestimmt noch weitere Möglichkeiten, die ich aber hier nicht alle zeigen möchte.

Nebenbei passt der rote Dämpfer farblich sehr gut zu fischertechnik.

Pneumatische Federung

Porsche-Makus hat ein schönes Beispiel für eine pneumatische Federung entwickelt (Abb. 20). Näheres dazu findet man unter [3].

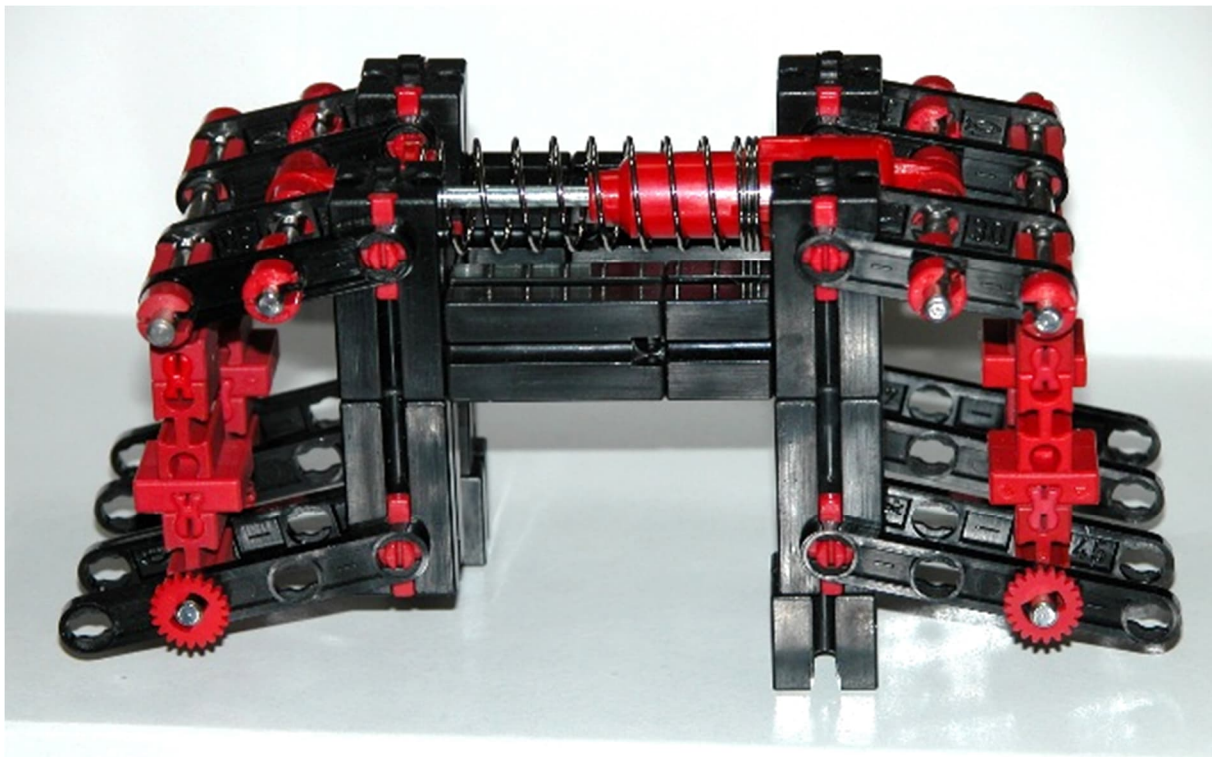


Abb. 18: Eine mögliche Konstruktion

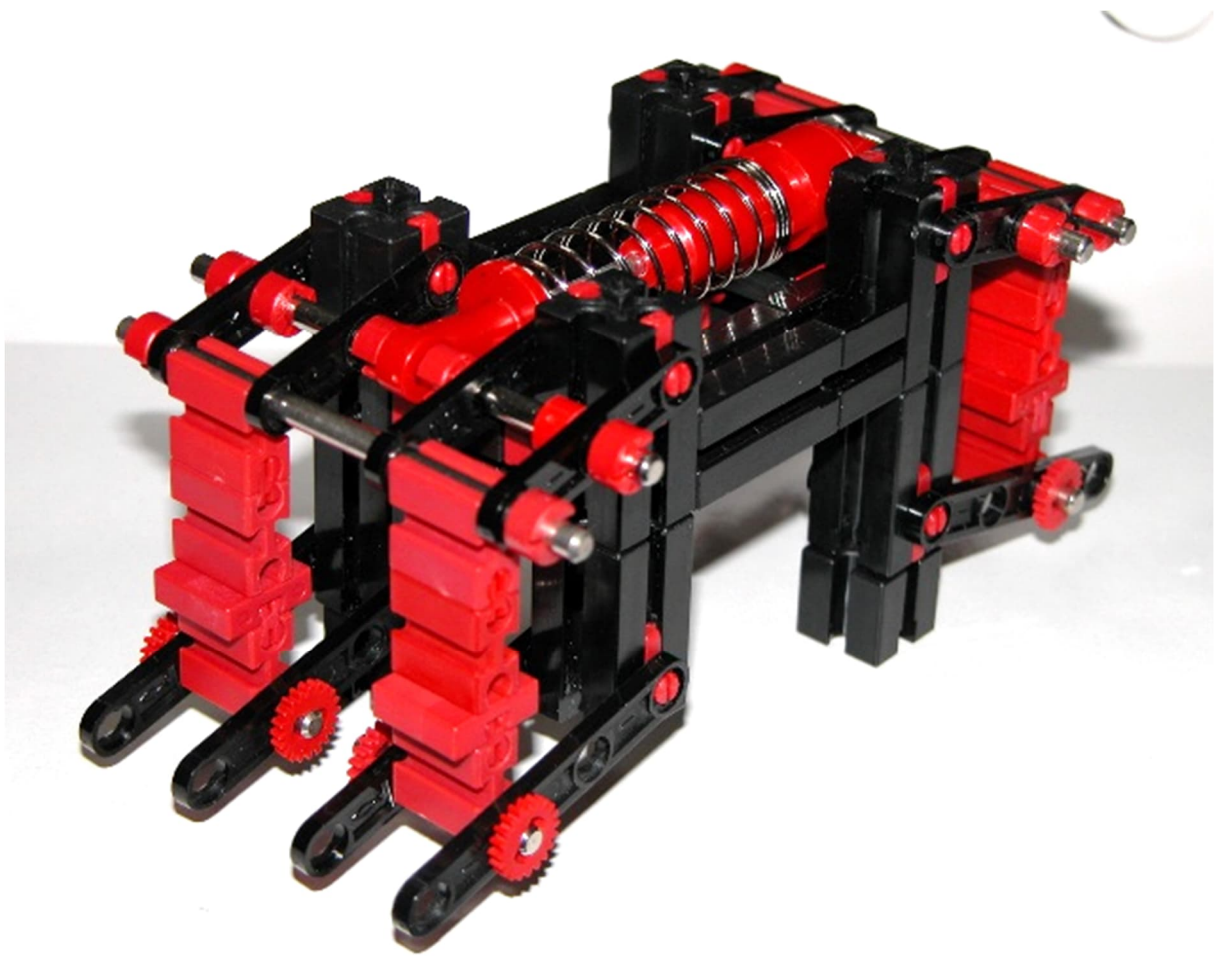


Abb. 19: 3/4-Ansicht

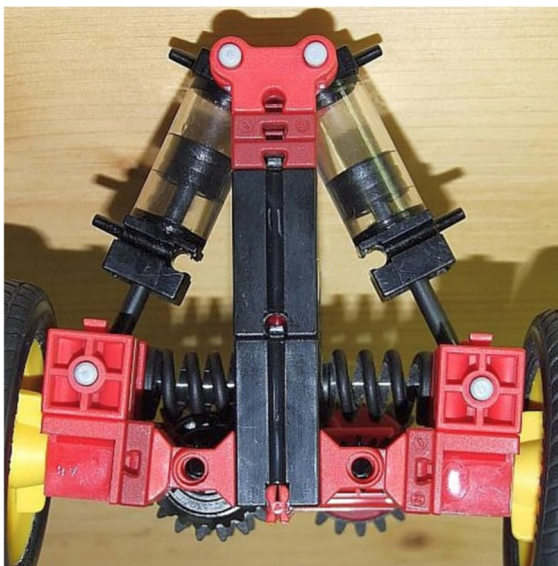


Abb. 20: Federung mit Pneumatik-Zylindern von Porsche-Makus

Blattfedern

Auch Blattfedern aufzubauen ist möglich. Abb. 21 zeigt einen Blick von oben auf einen ersten Versuch. Hier sind zwei Achsen mit jeweils zwei Blattfedern aufgebaut. Die Blattfedern sind hier nur am mittleren Rollenlager befestigt. Hier würden sich die neuen schwarzen ([208438](#)) optisch besser machen. Das andere Ende liegt lose im dazugehörigen äußeren Lager und wird nur durch den Verbinder 15 am Herausrutschen gehindert. Die Federung geschieht zusätzlich noch mit vier schwarzen fischertechnik-Federn, die zwischen die roten Bausteine 15 mit Bohrung und dem Aluträger eingebaut werden. Diese fehlen noch auf dem Bild.

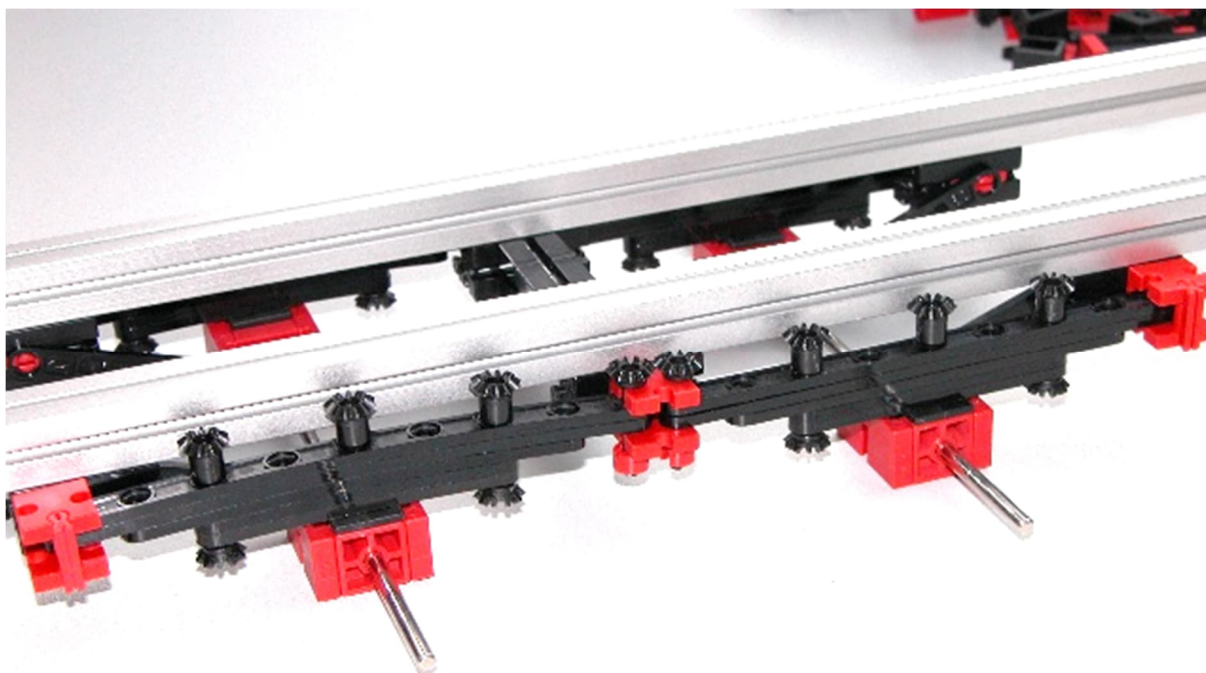


Abb. 21: Blattfedern aus fischertechnik

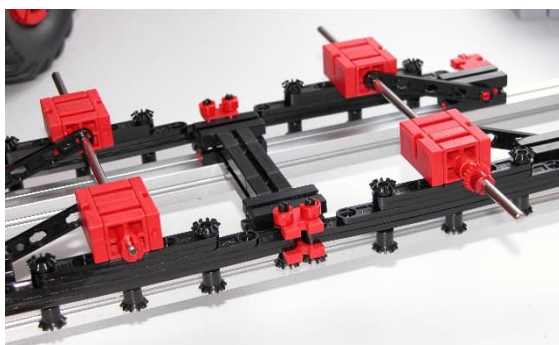


Abb. 22: Blick von unten

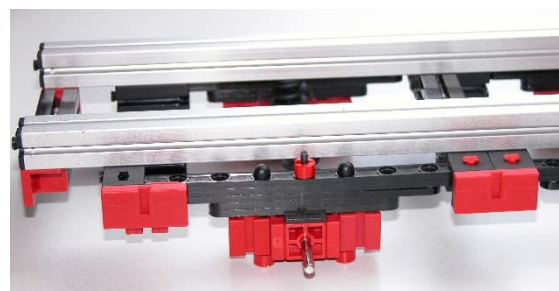


Abb. 24: Konstruktionsvariante

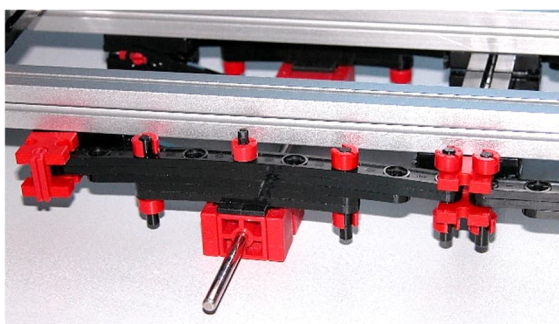


Abb. 23: Halterungen mit Klemmrings

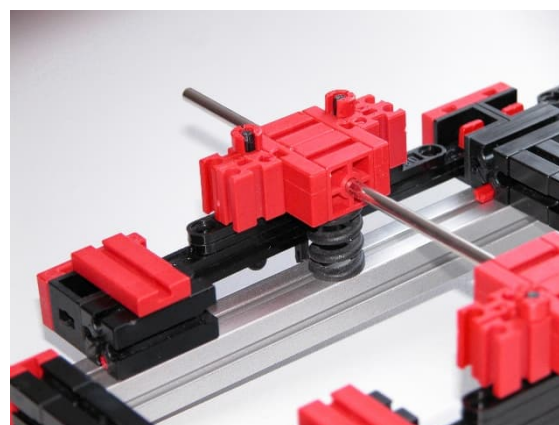


Abb. 25: Blick von unten unter den geänderten Aufbau

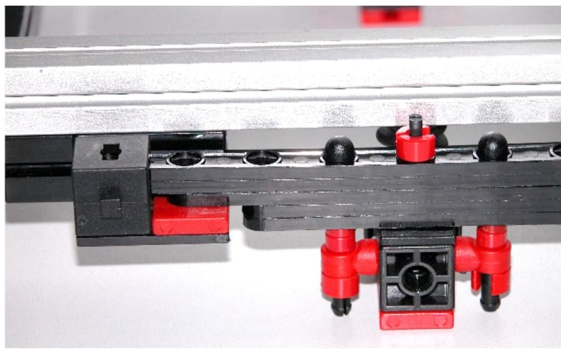


Abb. 26: Das endgültige Ergebnis

Abb. 22 zeigt den Aufbau von unten. Die mittlere Befestigung der Blattfedern ist zur Aufnahme des Bausteins 15 mit Bohrung unbedingt notwendig. Er wird durch eine Rastachse mit Bauplatte ([130593](#)) an der Blattfeder gehalten. Die beiden äußeren Verbindungen (die mit den kleinen Zahnradern) sollen ein Verdrehen der Streben verhindern.

Dadurch, dass die Blattfedern an so vielen, weit auseinander liegenden, Punkten gehalten wurden, war eine gute Federung nicht möglich. Deswegen mussten diese weiter in die Mitte, was man in Abb. 23 gut sehen kann. So federt das Ganze etwas besser.

Die Rollenlager wurden gegen Winkelträger 15 getauscht und die komplette Achsaufnahme ein weiteres Mal geändert. Die kleine rote Platte neben dem Winkelträger ist eine Bauplatte $15 \times 15 \times 3,75$ ([32315](#)) und dient als Höhenausgleich, damit die Blattfeder relativ waagrecht im Modell eingebaut werden kann. Auf der anderen Seite ist sie mit einem S-Riegel am Winkelträger 15 verbunden.

Zum Schluss kann das Ganze noch ein wenig verblendet werden und alles ist fertig.

Ergänzende Hinweise

Die Preise bei AliExpress ändern sich fast täglich, deswegen sind die in diesem Beitrag genannten nur ein Richtwert. Stand der Aufnahme in die Liste ist der 08.10.2025. Ich habe hier nur eine Auswahl einiger Dämpfer zusammengefasst. Über

Qualität und Funktionsweise der einzelnen Angebote kann ich keine Angaben machen.

Lediglich das Vierer-Set von Tamiya (rote Kunststoff-Reibungsdämpfer) und die Öldruckdämpfer des Boomerang habe ich hier getestet. Die Boomerang-Öldruckdämpfer haben eine starke Federung und Dämpfung. Um den vorderen Dämpfer komplett einzudrücken, braucht es 2 kg Kraftaufwand bis Maximum.

Und hier noch etwas zum Schmunzeln: Frage und Antwort von Käufern und Verkäufern aus einem Internet-Shop, der Öldruckstoßdämpfer anbietet:

Frage: *Wie füge ich Öl hinzu?*

Antwort: *Sparen Sie sich den Ärger und gießen Sie das Öl einfach über sich selbst, Ihre Schuhe und den Boden. Dort wird es sowieso enden.*



Abb. 27: „Kraftaufwand bis Maximum: ca. 1.000 g“

Das bedeutet, dass ich eine Gewichtskraft von 1.000 g aufbringen muss, bis der Kolben komplett eingefahren ist (gemessen mit einer Waage). Das hört sich nach viel an, ist aber je nach Hebelwirkung an den Befestigungspunkten mitunter gar nicht so viel.

Quellen

- [1] Ludger Mäsing: *Riesen-LKW*. Im [Bilderpool der ft-Community](#), 2012.
- [2] Tamiya: *The Boomerang – 1/10th Scale R/C High Performance 4WD Off Road Racer*. Aufbauanleitung auf [cloudfront.net](#), 2008.
- [3] Dirk Fox: *Wie man fischertechnik-Fahrzeuge federt*. auf [fischertechnik-blog.de](#), 2025.
- [4] Tamico: *Stoßdämpferöl Soft 9 ml*. Auf [tamico.de](#).
- [5] AliExpress: *METALL 53mm 59mm 65mm 71mm 77mm Integrierte Feder Stoßdämpfer Öl Dämpfer*. Auf [aliexpress.com](#).
- [6] Tamiya: *Stoßdämpfer-Beutel V/H (4St.) Mad Bull*. Auf [modellbau-seidel.de](#).
- [7] Conrad: *Reely 1:10 Aluminium-Öldruckstoßdämpfer Rot (metallic) mit Federn Schwarz 90 mm 4 St.* Auf [conrad.de](#).
- [8] Injora: *Aluminium-Stoßdämpfer 90/100/110/120 mm Dämpfer*. Auf [aliexpress.com](#).
- [9] Injora: *Eingebauter Federstoßdämpfer 90/100/110/120 mm*. Auf [aliexpress.com](#).
- [10] JoJonDa: *Aluminium 75mm 80mm 90mm Ölstoßdämpfer*. Auf [aliexpress.com](#).
- [11] ZuoLan: *Aluminium Stoßdämpfer 55mm*. Auf [amazon.de](#).
- [12] JVRWSC: *Metall Öl Stoßdämpfer Dämpfer*. Auf [aliexpress.com](#).

Tipps & Tricks

Große Reifen für große Modelle

Ludger Mäsing

Immer wieder kommt es vor, dass die fischertechnik-Reifen nicht ausreichen. Ganz besonders dann, wenn man größere Modelle in Planung oder gebaut hat, steht man vor dem Problem „Welche Reifen nutze ich und wo kann ich diese bekommen?“ Hier ist eine kleine Übersicht darüber, damit ihr eine Vorstellung davon bekommt, was alles möglich ist.



Abb. 1: AXSPEED 1,9“, 2,2“

- Material: Gummi
- Raddurchmesser: 122 mm
- Radbreite: 42 mm
- Innendurchmesser: 47 mm
- Preis für vier Stück mit Schaumstoff: 28,89 €
- Gewicht: 96 g/Stück

Die größere Variante:

- Raddurchmesser: 130 mm
- Radbreite: 54 mm
- Innendurchmesser: 54 mm
- Preis für vier Stück mit Schaumstoff: 39,60 €
- Gewicht: 146 g/Stück
- Bezugsquelle:
<https://www.ebay.de/itm/146242099502>



Abb. 2: JDM-123

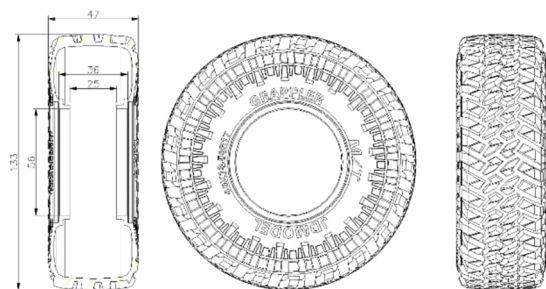


Abb. 3: Maßzeichnung

- Material: Gummi
- Raddurchmesser: 133 mm
- Radbreite: 47 mm
- Innendurchmesser: 56 mm
- Preis für zwei Stück mit Schaumstoff: 26,51 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005007453411700.html>



Abb. 4: Reifen mit „Spikes“

- Material: Gummi
- Raddurchmesser: 80 mm
- Radbreite: 35 mm
- Innendurchmesser: 60 mm
- Preis für 2 Stück: 6,50 €
- Bezugsquelle:
<https://arkai.de/produkt/vh36-hinterreifen-2x>



Abb. 5: AJRC 1,9“

- Material: Gummi
- Raddurchmesser: 110 mm
- Radbreite: 45 mm
- Innendurchmesser: 46 mm
- Preis für vier Stück: 51,22 €
- Gewicht: 80 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005005545101988.html>



Abb. 6: Dick und dicker

- Material: Gummi
- Raddurchmesser: 130 mm
- Radbreite: 60 mm
- Innendurchmesser: 56 mm
- Preis für vier Stück: 39,99 €
- Gewicht: 125 g/Stück
- Bezugsquelle (z. Zt. nicht verfügbar):
<https://www.amazon.de/hobbysoul-Badland-Monster-TraktorReifen/dp/B0743DHTVT>



Abb. 7: Kautschuk-Reifen

- Material: Butylkautschuk
- Raddurchmesser: 127 mm
- Radbreite: 54 mm
- Innendurchmesser: 56 mm
- Preis für zwei Stück: 25,50 €
- Bezugsquelle:
https://www.amazon.de/Generic-Crawler-5-0inch-PassendBeadlock/dp/B09XM9KPVR/ref=asc_df_B09XM9KPVR



Abb. 8: RS REDSPIDER

- Material: Gummi
- Raddurchmesser: 130 mm
- Radbreite: 55 mm
- Innendurchmesser: 53 mm
- Preis für vier Stück: 45,69 €
- Gewicht: 136 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008496835042.html>



Abb. 10: R-SPVH36104



Abb. 9: AXSPEED 2,2"

- Material: Gummi
- Raddurchmesser: 125 mm
- Radbreite: 50 mm
- Innendurchmesser: 54,5 mm
- Preis für vier Stück mit Schaumstoff: 38,98 €
- Gewicht: 116 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005007467338088.html>



Abb. 11: Noppen

- Material: unbekannt
- Raddurchmesser: 85 mm
- Radbreite: 36 mm
- Innendurchmesser: 59 mm
- Preis für 2 Stück: 12,50 €
- Bezugsquelle:
<https://arkai.de/produkt/noppenreifen-2-stk-breit-fuer-fast-alle-110er-rc-autos>



Abb. 12: R-SPVH36105



Abb. 13: Noppen

- Material: unbekannt
- Raddurchmesser: 85 mm
- Radbreite: 34 mm
- Innendurchmesser: 59 mm
- Preis für 2 Stück: 9,50 €
- Bezugsquelle:
<https://arkai.de/produkt/noppenreifen-2-stk-fuer-fast-alle-rc-autos>



Abb. 14: REDSPIDER

- Material: Gummi
- Raddurchmesser: 70 mm
- Radbreite: 26 mm
- Innendurchmesser: 33 mm
- Preis für vier Stück: 33,73 €
- Gewicht: 22 g/Stück
- Bezugsquelle (z. Zt. nicht verfügbar):
<https://de.aliexpress.com/item/1005009254538393.html>



Abb. 15: Tamiya 53059 Heckreifen

- Material: unbekannt
- Raddurchmesser: 80 mm
- Radbreite: 38 mm
- Innendurchmesser: 54 mm
- Preis für 2 Stück: 13,99 €
- Bezugsquelle:
<https://www.rc-kleinkram.de/products/17-53059>



Abb. 16: ZuoLan-Reifen

- Material: Gummi
- Raddurchmesser: 127 mm
- Radbreite: 40 mm
- Innendurchmesser: 55 mm
- Preis für vier Stück: 32,99 €
- Gewicht: 90 g/Stück
- Bezugsquelle:
<https://www.amazon.de/ZuoLan-Reifen-Radreifen-CrawlerWraith/dp/B01NB9EQND>



Abb. 17: Tamiya 50476

- Material: Gummi
- Raddurchmesser: 69 mm
- Radbreite: 26 mm
- Innendurchmesser: 47 mm
- Preis für 2 Stück: 11,99 €
- Bezugsquelle:
<https://www.rc-kleinkram.de/products/17-50476>



Abb. 18: REDSPIDER 1,9“

- Material: Gummi
- Raddurchmesser: 110 mm
- Radbreite: 38 mm
- Innendurchmesser: 55 mm
- Preis für vier Stück: 38,55 €
- Gewicht: 75 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005009349519224.html>



Abb. 19: RM12826

- Material: Gummi
- Raddurchmesser: 115 mm
- Radbreite: 45 mm
- Innendurchmesser: 55 mm
- Preis für vier Stück: 22,63 €
- Bezugsquelle (z. Zt. nicht verfügbar):
<https://de.aliexpress.com/item/1005009151883161.html>



Abb. 20: REDSPIDER

- Material: Hartgummi
- Raddurchmesser: 116 mm
- Radbreite: 42 mm
- Innendurchmesser: 43 mm
- Preis für vier Stück: 55,56 €
- Gewicht: 117 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008136286157.html>



Abb. 21: Chanmoo 2,2" Crawler Reifen

- Material: Gummi
- Raddurchmesser: 130 mm
- Radbreite: 50 mm
- Innendurchmesser: 55 mm
- Preis für vier Stück mit Schaumstoff: 40,34 €
- Gewicht: 100 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008176504971.html>



Abb. 22: Tamiya 54284 - DT-03 Material:

- Raddurchmesser: 80 mm
- Radbreite: 26 mm
- Innendurchmesser: 54 mm
- Preis für 2 Stück: 12,99 €
- Bezugsquelle:
<https://www.rc-kleinkram.de/products/17-54284>



Abb. 23: R-VH32055

- Material: unbekannt
- Raddurchmesser: 65 mm
- Radbreite: 26 mm
- Innendurchmesser: 45 mm
- Preis für 2 Stück: 10,50 €
- Bezugsquelle:
<https://arkai.de/produkt/profilreifen-fur-1-10-cars-2-stk>



Abb. 24: REDSPIDER

- Material: weicher Gummi
- Raddurchmesser: 55 mm
- Radbreite: 23 mm
- Innendurchmesser: 23 mm
- Preis für vier Stück: 6,34 €
- Gewicht: 11 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005009187203233.html>



Abb. 25: Tamiya 51427

- Material: unbekannt
- Raddurchmesser: 60 mm
- Radbreite: 25 mm
- Innendurchmesser: 42 mm
- Preis für 2 Stück: 7,99 €
- Gewicht: 70 g/Stück
- Bezugsquelle:
<https://www.rc-kleinkram.de/products/17-51427>



Abb. 26: Reifen mit starkem Profil

- Material: Gummi
- Raddurchmesser: 110 (Text) / 116 mm
- Radbreite: 45 mm
- Innendurchmesser: 42 mm
- Preis für vier Stück ??: 15,83 €
- Gewicht: 112 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008955112163.html>



Abb. 27: YEAHRUN

- Material: Gummi
- Raddurchmesser: 84,5 mm
- Radbreite: 30 mm
- Innendurchmesser: 44,5 mm
- Preis für vier Stück (evtl. mit Reifenabdeckung): 24,98 €
- Gewicht: 76 g/Stück
- Bezugsquelle:
<https://www.ebay.de/itm/334790183920>



Abb. 28: AXSPEED AX6658

- Material: Gummi
- Raddurchmesser: 120 mm
- Radbreite: 45,7 mm
- Innendurchmesser: 55,8 mm
- Preis für vier Stück mit Schaumstoff: 17,88 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 30: INJORA 1.0 RC Reifen – S5 Crawler

- Material: Weichgummi
- Raddurchmesser: 62 mm
- Radbreite: 20,5 mm
- Innendurchmesser: 24,2 mm
- Preis: ca. 12,98 €
- Gewicht: 55 g/Stück
- Bezugsquelle:
<https://www.amazon.de/dp/B09ZDBFB3Q>



Abb. 29: AXSPEED AE8500

- Material: Gummi
- Raddurchmesser: 125 mm
- Radbreite: 50 mm
- Innendurchmesser: 54,5 mm
- Preis für vier Stück mit Schaumstoff: 28,58 €
- Gewicht: 116 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 31: Austar 2,9“

- Material: Verbundmaterial
- Raddurchmesser: 180 mm
- Radbreite: 75 mm
- Innendurchmesser: 74 mm
- Preis für 2 Stück mit Schaumstoff: 55,72 €
- Preis für 4 Stück mit Schaumstoff: 102,20 €
- Gewicht: 82 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005005416854271.html>



Abb. 32: ARMA #AR520044 Front/Rear
3,0/2,2

- Material: unbekannt
- Raddurchmesser: 105 mm
- Radbreite: 47 mm
- Äußerer Wulstdurchmesser: 76 mm
- Innerer Wulstdurchmesser: 55 mm
- Preis für 2 Stück mit Schaumstoff: 14,85 €
- Bezugsquelle:
<https://www.rcfox.de/ARAC9431-Arrma-AR520044-Fortress-SCTire-30/22-Foam-Insert-2>



Abb. 33: AXSPEED AE8840

- Material: Gummi
- Raddurchmesser: 150 mm
- Radbreite: 67 mm
- Äußerer Wulstdurchmesser: 62 mm
- Innerer Wulstdurchmesser: 54 mm
- Preis für vier Stück mit Schaumstoff: 39,63 €
- Gewicht: 208 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 34: AXSPEED AE9328

- Material: Gummi
- Raddurchmesser: 125 mm
- Radbreite: 56 mm
- Innendurchmesser: 54 mm
- Preis für vier Stück mit Schaumstoff: 37,07 €
- Gewicht: 157 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 35: AXSPEED AE9326

- Material: Gummi
- Raddurchmesser: 120 mm
- Radbreite: 43 mm
- Innendurchmesser: 47 mm
- Preis für vier Stück mit Schaumstoff: 25,25 €
- Gewicht: 105 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 36: AXSPEED AE8557

- Material: Gummi
- Raddurchmesser: 120 mm
- Radbreite: 52 mm
- Äußerer Wulstdurchmesser: 54 mm
- Innerer Wulstdurchmesser: 48 mm
- Preis für vier Stück mit Schaumstoff: 22,89 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 37: AXSPEED AE4614 1,9"

- Material: Gummi
- Raddurchmesser: 107 mm
- Radbreite: 42 mm
- Innendurchmesser: 50 mm
- Preis für vier Stück mit Schaumstoff: 19,98 €
- Gewicht: 85 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005008971187667.html>



Abb. 38: AUSTAR AX-3020

- Material: Gummi
- Raddurchmesser: 102 mm
- Radbreite: 40 mm
- Innendurchmesser: 48,3 mm
- Preis für vier Stück mit Schaumstoff: 29,98 €
- Gewicht: 63 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/4000142188449.html>



Abb. 39: AKA13135GR

- Material: Soft-Reifen
- Raddurchmesser: 83 mm
- Radbreite: 39 mm
- Innendurchmesser: 55,9 mm
- Preis für 2 Stück mit Schaumstoff: 16,90 €
- Bezugsquelle:
<https://www.rcilinde.com/collections/aka/products/aka-110-rivetsoft-carpet-rear>



Abb. 40: Carson - Fulda EcoControl

- Material: Hohlkammerreifen
- Raddurchmesser: 82 mm
- Radbreite: 20 mm
- Innendurchmesser: 44 mm
- Preis für 2 Stück: 19,90 €
- Gewicht: 80 g/Stück
- Bezugsquellen:
<https://www.ebay.de/itm/387387057242>
oder
<https://www.testberichte.de/pdb/iHUqiKrQTpuf0GCjDBddiw.html>



Abb. 41: LKW-Reifen

- Material: Silikonkautschuk
- Raddurchmesser: 84 mm
- Radbreite: 23 mm
- Innendurchmesser: 44,45 mm
- Preis für vier Stück: 13,55 €
- Bezugsquelle:
<https://www.ebay.de/itm/388351611504>



Abb. 42: Austar AX-4020

- Material: Gummi
- Raddurchmesser: 110 mm
- Radbreite: 40 mm
- Innendurchmesser: 48,26 mm
- Preis für vier Stück: 22,40 €
- Gewicht: 70 g/Stück
- Bezugsquelle:
<https://www.ebay.de/itm/364351043879>



Abb. 43: KINGCOO

- Material: Gummi
- Raddurchmesser: 100 mm
- Radbreite: 40 mm
- Innendurchmesser: 48 mm
- Preis für vier Stück: 20,99 €
- Gewicht: 70 g/Stück
- Bezugsquelle:
<https://www.amazon.de/KINGCOO-Reifen-Crawler-FelgenRadreifen/dp/B0CR1CNPVN>



Abb. 44: KINGCOO

- Material: Gummi
- Raddurchmesser: 114 mm
- Radbreite: 42 mm
- Innendurchmesser: 48 mm
- Preis für vier Stück mit Schaumstoff: 25,99 €
- Bezugsquelle:
<https://www.amazon.de/KINGCOO-Reifen-Crawler-FelgenRadreifen/dp/B0CR1DVX5T>



Abb. 45: INJORA 2,2“



Abb. 46: Profilansicht

- Material: Weichgummi
- Raddurchmesser: 130 mm
- Radbreite: 46 mm
- Innendurchmesser: 50 mm
- Preis für vier Stück mit Schaumstoff: 39,83 €
- Gewicht: 114 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005001812950806.html>



Abb. 47: INJORA 1,9“ Rock Crawling Tire

- Material: Gummi
- Raddurchmesser: 108 mm
- Radbreite: 40 mm
- Innendurchmesser: 48 mm
- Preis für vier Stück mit Schaumstoff: 22,99 €
- Gewicht: 103 g/Stück
- Bezugsquelle:
<https://www.amazon.de/dp/B07167PS9V>



Abb. 48: INJORA 1,9“

- Material: Gummi
- Raddurchmesser: 114 mm
- Radbreite: 39 mm
- Innendurchmesser: 48 mm
- Preis für vier Stück mit Schaumstoff: 25,99 €
- Gewicht: 90 g/Stück
- Bezugsquelle:
<https://www.amazon.de/dp/B076MRY4PG>



weight: 180g/PCS

Abb. 49: 15 cm Durchmesser

- Material: Gummi
- Raddurchmesser: 150 mm
- Radbreite: 64 mm
- Innendurchmesser: 56 mm
- Preis für vier Stück mit Schaumstoff: 38,79 €
- Gewicht: 180 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005003674079728.html>



Abb. 50: DIRC 1,7 Tyre - ZC7SIWO

- Material: Gummi
- Raddurchmesser: 90 mm
- Radbreite: 26 mm
- Innendurchmesser: 43 mm
- Preis für 1 Stück mit Schaumstoff ???:
47,16 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005007444573456.html>



Abb. 51: Eine mittlere Größe

- Material: Gummi
- Raddurchmesser: 62 mm
- Radbreite: 22 mm
- Innendurchmesser: 24 mm
- Preis für vier Stück mit Schaumstoff:
5,70 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005006952659393.html>



Abb. 52: Fechtner

- Material: Vollgummi
- Raddurchmesser: 72 mm
- Radbreite: 25 mm
- Innendurchmesser: 40 mm
- Preis für 1 Stück: 17,40 €
- Bezugsquelle:
<https://fechtner-modellbau.de/1-SLT-Reifen-Vollmaterial-1-WDCDa-72mm-Di-40mm-25mm/3806>



Abb. 53: Veroma Tiefladerreifen

- Material: Vollgummi
- Raddurchmesser: 50 mm
- Radbreite: 14,5 mm
- Innendurchmesser: 28 mm
- Preis für 1 Stück: 4,62 €
- Bezugsquelle:
<https://fechtner-modellbau.de/Tiefladerreifen-1-Stueck-Vollgummi-50-x-28-x-14-5mm/1910>



Abb. 54: Fechtner

- Material: Vollgummi
- Raddurchmesser: 59 mm
- Radbreite: 16 mm
- Innendurchmesser: 29 mm
- Preis für vier Stück: 23,99 €
- Bezugsquelle:
<https://fechtner-modellbau.de/Tiefladerreifen-4-Stueck-Vollgummi59x29x16mm-aussen/10496>



Abb. 56: Fechtner

- Material: Vollgummi
- Raddurchmesser: 53 mm
- Radbreite: 16 mm
- Innendurchmesser: 29,5 mm
- Preis für 1 Stück: 11,98 €
- Bezugsquelle:
<https://fechtner-modellbau.de/1-Reifen-Strassenbreitreifen-fuerBruder-Sprinter.-Da-53mm/10433>



Abb. 55: Tiefladerreifen

- Material: Vollgummi
- Raddurchmesser: 47 mm
- Radbreite: 19 mm
- Innendurchmesser: 24 mm
- Preis für 1 Stück: 10,29 €
- Bezugsquelle:
<https://fechtner-modellbau.de/1-Reifen-Continental-315-55R16MPT81.-Vollmaterial/5056>



Abb. 57: EPINON

- Material: Gummi
- Raddurchmesser: 118 mm
- Radbreite: 45 mm
- Innendurchmesser: 46 mm

- Preis für vier Stück mit Schaumstoff: 25,40 €
- Gewicht: 105 g/Stück
- Bezugsquelle:
<https://www temu.com/de/1-9-kletterreifen-1-10-rc-reifen-gummireifen-118-45mm-fur1-10-t4--scx-10---8-4-stuck-g-601099552554898.html>



Abb. 58: Tamiya 56528

- Material: Gummi (hart)
- Raddurchmesser: 85,5 mm
- Radbreite: 30 mm
- Innendurchmesser: 44 mm
- Preis für 2 Stück: 14,99 €
- Bezugsquelle:
<https://www.conrad.de/de/p/tamiya-1-14-lkw-reifen-30-mmstrassenprofil-1-paar-1040429.html>



Abb. 59: AliExpress

- Material: Gummi/Kunststoff
- Raddurchmesser: 105 mm
- Radbreite: 38 mm
- Innendurchmesser: 43,5 mm
- Preis für 2 Stück: 33,98 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005004587586799.html>



Abb. 60: Reifen von Amazon

- Material: Gummi
- Raddurchmesser: 58 mm
- Radbreite: 20 mm

- Innendurchmesser: 24 mm
- Preis für 2 Stück mit Schaumstoff: 14,76 €
- Bezugsquelle: <https://www.amazon.de/dp/B0CGDJLQMX>



Abb. 61: YEAHRUN 2,2“

- Material: Gummi
- Raddurchmesser: 137 mm
- Radbreite: 49 mm
- Innendurchmesser: 58 mm
- Preis für vier Stück mit Schaumstoff: 22,79 €
- Gewicht: 131 g/Stück



Abb. 62: YEAHRUN 1,9“

- Material: Gummi
- Raddurchmesser: 107 mm
- Radbreite: 42 mm
- Innendurchmesser: 50 mm
- Preis für vier Stück mit Schaumstoff: 19,39 €
- Gewicht: 85 g/Stück

- Bezugsquelle: <https://de.aliexpress.com/item/1005003244582490.html>



Abb. 63: Steowsey 1,9“ Gotoger

- Material: Gummi
- Raddurchmesser: 90 mm
- Radbreite: 33 mm
- Innendurchmesser: 48 mm
- Preis für vier Stück mit Schaumstoff: 13,63 € /14,49 €
- Gewicht: 43 g/Stück
- Bezugsquellen: <https://www.amazon.de/dp/B0F6JKKK9X> oder <https://www.amazon.de/Gotoger-ST%C3%9CCKE-GummireifenRadreifen-Crawler/dp/B0DK3NTLJB>



Abb. 64: Ohulelks

- Material: Gummi
- Raddurchmesser: 128 mm
- Radbreite: 58 mm
- Innendurchmesser: 53 mm
- Preis für vier Stück: 20,50 €
- Gewicht: 182 g/Stück
- Bezugsquelle (z. Zt. nicht verfügbar):
<https://www.amazon.de/Ohulelks-St%C3%BCck-Felgen-TruckReifen-Upgrade-Teile/dp/B0DWSN9JN4>



Abb. 65: KINGCOO 1/10

- Material: Gummi
- Raddurchmesser: 65 mm
- Radbreite: 26 mm
- Innendurchmesser: 47 mm
- Preis für vier Stück mit Schaumstoff: 10,99 €

- Bezugsquelle:
<https://www.amazon.de/KINGCOO-Autoreifen-Stra%C3%9Fe-4St%C3%BCck-Gummireifen/dp/B0DL2N9X2L>



Abb. 66: Reifen mit Schaumstoffeinsatz

- Material: Gummi
- Raddurchmesser: 105 mm
- Radbreite: 40 mm
- Innendurchmesser: 44,45 mm
- Preis für 1 Stück mit Schaumstoff: 14,87 €
- Bezugsquelle (z. Zt. nicht verfügbar):
<https://de.aliexpress.com/item/1005003228474973.html>



Abb. 67: LESU RD-2004

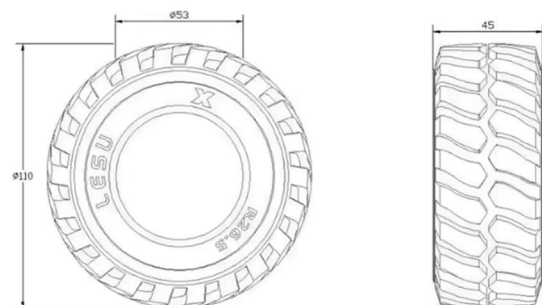


Abb. 68: Maßzeichnung

- Material: Gummi
- Raddurchmesser: 110 mm
- Radbreite: 45 mm
- Innendurchmesser: 53 mm
- Preis für 2 Stück: 43,65 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005007630826853.html>

weight:101g/3.56oz/pcs



Abb. 69: ARJC 1,9 KO2

- Material: Gummi
- Raddurchmesser: 115 mm
- Radbreite: 45 mm
- Innendurchmesser: 53 mm
- Preis für vier Stück: 53,00 €
- Gewicht: 101 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005007321552188.html>



Abb. 70: LESU RD-2022

LESU RD-2022-1:

- Material: Gummi
- Raddurchmesser: 89,1 mm
- Radbreite: 34,8 mm
- Innendurchmesser: 46,2 mm
- Preis für 2 Stück: 35,08 €

LESU RD-2023-1:

- Material: Gummi
- Raddurchmesser: 113,4 mm
- Radbreite: 43,9 mm
- Innendurchmesser: 58,2 mm
- Preis für 2 Stück: 36,25 €
- Bezugsquelle:
<https://de.aliexpress.com/item/1005005848326482.html>



Abb. 71: LESU

- Material: PU
- Raddurchmesser: 142 mm
- Radbreite: 60 mm
- Innendurchmesser: 53 mm

- Preis für 1 – 6 Stück mit Schaumstoff:
15,75 € - 85,39 €
- Gewicht: 385 g/Stück
- Bezugsquelle:
<https://de.aliexpress.com/item/1005006629003971.html>

Leider sind einige der Reifen zurzeit nicht bei dem angegebenen Anbieter erhältlich. Im WWW gibt es aber diverse andere Shops, bei denen man suchen könnte.

Felgen

Einige der Reifen-Innendurchmesser sind leider so gar nicht im fischertechnik-Raster. Mit ein wenig Bastelei kann man bei den größeren Rädern etwas zusammenstellen. Für diejenigen, die in der glücklichen Lage sind, einen 3D Drucker zu besitzen, dürfte es nicht schwer sein, eine passende Felge herzustellen. Vielleicht können diese dann zum Download angeboten werden.

Im fischertechnik-Programm gibt es diverse Felgen:

- Felge 20,5: [142252](#), [143231](#), [159782](#)
- Felge 23: [31986](#), [31987](#), [32241](#)
- Felge 30: [32627](#), [32883](#), [36700](#), [78727](#), [116913](#), [165364](#)
- Felge 40: [163205](#), [213850](#)
- Felge 43: [31878](#), [31879](#), [107359](#), [122646](#), [146245](#)
- Felge Omniwheel (Innendurchmesser ca. 31 mm, Außendurchmesser ca. 56 mm, Außenbreite ca. 33 mm ohne Kugeln und Rollen): [183025](#), [183026](#), [208535](#), [208536](#) (Alternativ: [208536](#) oder [208535](#))

Bei Thingiverse habe ich eine Felge entdeckt, die für 1/10-RC-Reifen passen soll. Genauer kann ich aber hierzu nicht sagen. Wer mag, suche dort nach „fischertechnik RC rim“.

Für die größeren Reifen würde sich eine Kombination aus Drehscheibe ([31019](#)) und ein paar Kleinteilen anbieten. Ich möchte hier einmal einige vorstellen:

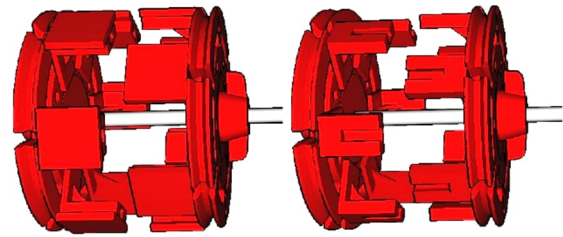


Abb. 72: Drehscheibe, Winkelstein, Platte

1. Drehscheibe ([31019](#)) und Winkelstein ([38423](#)) und aufgeschobener Bauplatte 15 × 15 ergeben eine Felge mit ca. 61,5 mm Durchmesser.
2. Drehscheibe ([31019](#)) und Winkelstein ([38423](#)) mit Nut nach außen ergeben eine Felge mit einem Durchmesser von ca. 57 mm.

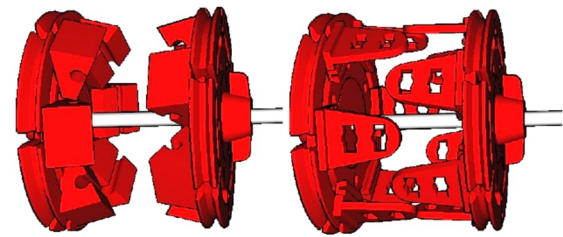


Abb. 73: Winkelsteine bzw. Kupplungsstücke

3. Drehscheibe ([31019](#)) und Winkelsteine ([31011](#)) ergeben eine Felge von ca. 57 mm bis ca. 52 mm. Weiche Reifen passen sich leicht an.
4. Drehscheibe ([31019](#)), Federnocken ([31982](#)) und Kupplungsstücke ([38253](#)) ergeben eine Felge von ca. 53 mm.

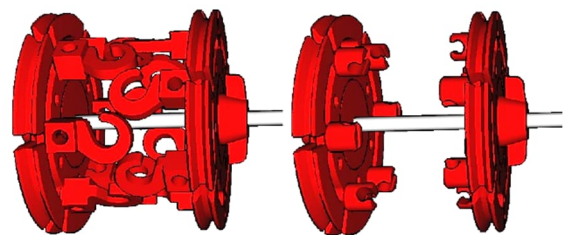


Abb. 74: Seilhaken, Klemmhülse

5. Drehscheibe und Seilhaken ([38225](#)) ergeben eine Felge von ca. 51 mm.
6. Drehscheibe ([31019](#)) und Klemmhülse 7,5 ([35980](#)) ergeben eine Felge von ca. 50 mm.

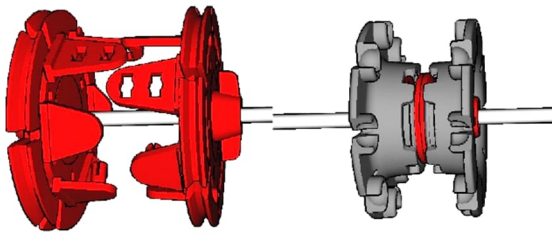


Abb. 75: Gemischt bestückt, Omniwheel

7. Drehscheibe ([31019](#)) und Kupplungsstück ([38253](#)), Radhalter ([35668](#)) oder Kufe ([31602](#)) ergeben eine Felge von ca. 45 mm.
8. Zwei OmniWheels-Felgen ([183025](#)) oder zwei [208535](#) ergeben eine Felge von ca. 31 mm innen und ca. 56 mm außen. Wenn man in die Vertiefungen der Felge noch einige Bauteile einsetzen könnte, wäre noch ein anderer Durchmesser zu erreichen.

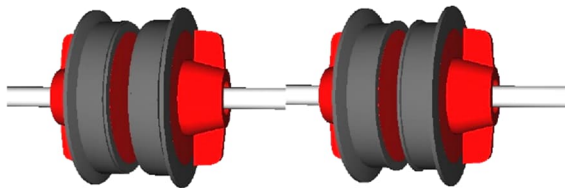


Abb. 76: Spurkränze

9. Spurkränze mit Gummiring ([35756](#)) ergeben eine Felge von ca. 30 mm. Eine Hälfte dieser Kombination passt auch wunderbar in das schwarze Drehkranz-Oberteil ([31390](#)).
10. Spurkränze ohne Gummiring ([36331](#)) ergeben eine Felge von ca. 28 mm Durchmesser.

Je nach Reifenbreite werden die beiden Hälften dann angepasst.

Designer-Datei

Diese zehn Beispiele gibt es als fischer-technik-Designer-Datei „Felgen“ zum Download [1]. Zur Darstellung der Felgen 1-7 bitte die Baugruppe „Grundgerüst“ und die entsprechende Baugruppe einer Felge auswählen. Zur Darstellung der Felge 8 bitte nur die Baugruppe „OmniWheels“ und

zur Darstellung der Felgen 9+10 bitte nur die entsprechende Felge auswählen.

In zwei Baustufen zeige ich verschiedene Antriebsmöglichkeiten. Dazu bitte „Grundgerüst“ und einen „Antrieb mit xxx“ und eventuell eine Felge auswählen. Die Baugruppen 8, 9 und 10 hier bitte nicht wählen.

Andere Maße

Für andere Maße könnte man versuchen, mit Hilfe eines 3D-Druckers passende Felgen zu entwickeln. So z. B. ein Bauteil wie die Drehscheibe, etwas verkleinert, vielleicht mit nur 5 Einschubmöglichkeiten. Dann können hier, genau wie vorhin beschrieben, diverse Kleinteile aufgeschoben werden, damit man verschiedene Reifengrößen bedienen kann. Die Langlöcher sind dann so weit zu reduzieren, dass noch ein Z40 aufgesteckt werden kann. Wenn dieses für die gewählten Reifen zu groß sein sollte, könnte es verkleinert oder mit einem 3D Drucker nachgebaut werden (Abb. 77).

Das habe ich in meinem Schwerlast-LKW verwendet und zum Antrieb noch ein Zahnrad Z40 angebaut, weil an einer Achse zu viel Schlupf entstanden wäre. Wie der Antrieb funktioniert, kann man auf Abb. 78 gut erkennen.



Abb. 77: Abgefrästes Z40

Jede Seite wird von einem Power-Motor angetrieben, beide unten im Bild halb zu sehen. Dadurch, dass die Kraft direkt über die liegenden Zahnräder durch die Seitenwand auf das Z40 übertragen wird, ist das Fahrzeug sehr stark. An den Achsen rutscht nichts, weil diese lediglich die Räder halten.

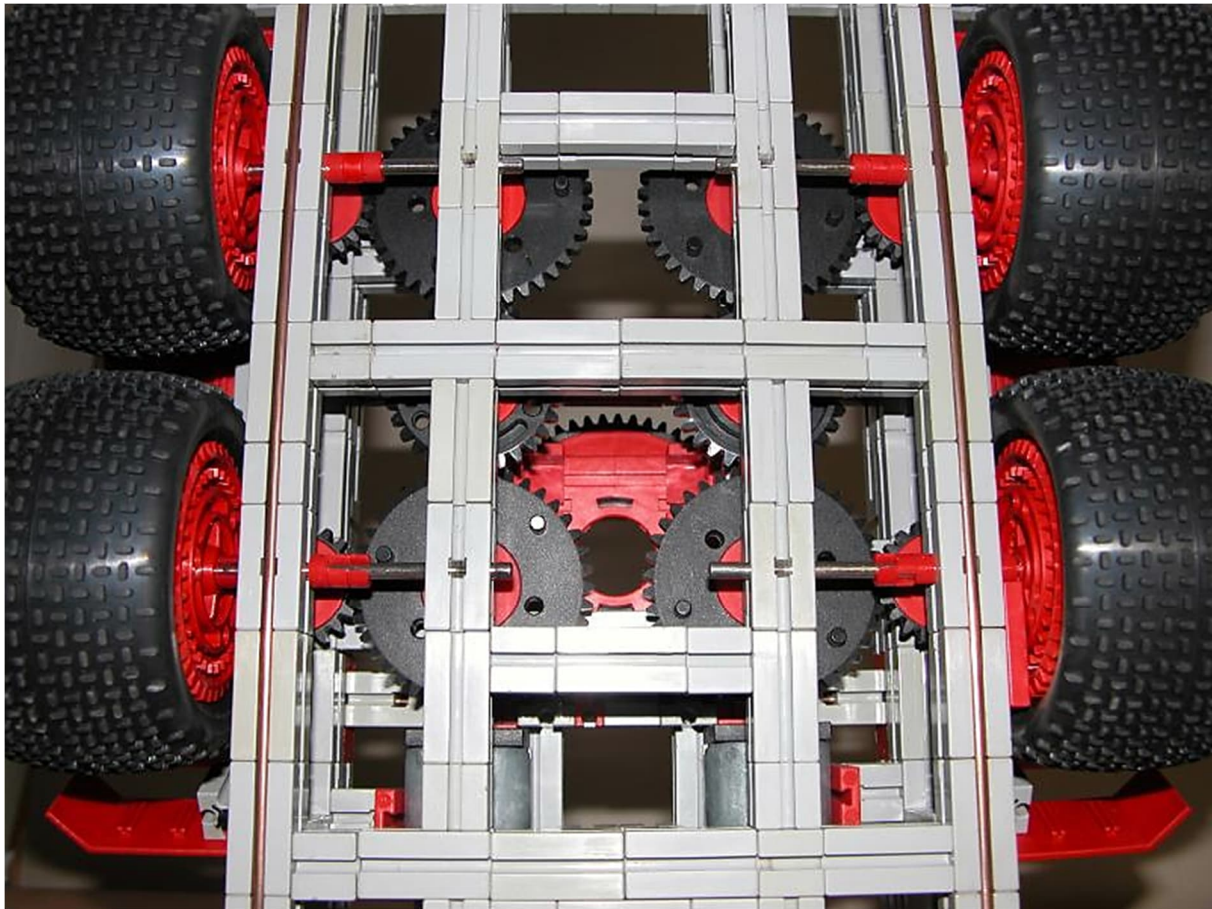


Abb. 78: Antrieb des Schwerlast-LKW

Der Stand der Reifen-Übersicht ist der 07.07.2025. Für Erstbesteller gibt es bei AliExpress besondere Konditionen. Das Copyright der Bilder liegt beim jeweiligen Hersteller.

Alle Angaben stammen aus dem WWW. Teilweise wurden die Texte aus dem Asiatischen ins Englische und dann ins Deutsche übersetzt. Verschiedene Anbieter haben bei gleichen Produkten bis zu 3 mm Abweichungen.

Für eventuelle Fehler kann ich keine Gewähr übernehmen.

Diese Anbieter habe ich gefunden:

- AJRC: <https://de.aliexpress.com/store/1102525096>
- AKA RC: über Suchmaschine
- Aliexpress: <https://de.aliexpress.com>
- Amazon: <https://www.amazon.de>
- AMK: über Amazon
- Arkai.de: <https://arkai.de>
- ARP: über Suchmaschine
- ARMA: <https://www.rrma-rc.com/de/teile/>
- Austar: über Aliexpress
- AXSPEED: über Suchmaschine
- Chanmoo: über Suchmaschine
- Dilwe: über Amazon
- Ebay: <https://www.ebay.de>
- fechtner-modellbau <https://fechtner-modellbau.de/>
- INJORA: <https://www.injora.com>
- JETKO: <http://jetkopower.com/>
- KINGCOO: über Amazon
- Kyosho: <https://www.kyoshoeurope.com/>
- LESU: über Suchmaschine

- RC-Kleinkram: <https://www.rc-kleinkram.de/>
- RhinoRC: über Suchmaschine
- Tamiya: <https://www.tamiya.de>
- Temu: <https://www temu.com>
- VeromaModellbau
<https://www.veroma-modellbau.eu>
- YEAHRUN: über Aliexpress
- Zero RC: über Aliexpress
- ZuoLan: über Suchmaschine

Wichtig: Das WWW schreibt, dass Warensendungen bis 150 € zollfrei sind, doch hinzu kommen 19 % Einfuhrumsatzsteuer. Zollgebühren können bis zu 12 % oder sogar mehr betragen. Die Einfuhrumsatzsteuer wird berechnet aus dem Warenwert + Porto.

Die EU plant, diesen 150 €-Freibetrag in Kürze aufzuheben. Ab dann müssen wahrscheinlich Zollgebühren gezahlt werden.

Bitte darauf achten!

Die Preise bei AliExpress ändern sich fast täglich. Deswegen kann es Abweichungen von den in der Liste angegebenen Preisen geben.

- Die Reifen von INJORA sind (so glaube ich) sehr weich. Siehe dazu Video <https://www.amazon.de/vdp/190bc5de30024dd8bfb6dc98f565d8c1>.

- Übersicht Reifen ZERO 1,9“: <https://de.aliexpress.com/item/32339499140.html>
- Übersicht Reifen ZERO 2,2“: <https://de.aliexpress.com/item/1005004285712308.html>
- Übersicht Reifen AXSPEED 1,0“: <https://de.aliexpress.com/item/1005006952659393.html>
- Übersicht Reifen AXSPEED 1,9“ + 2,2“: <https://de.aliexpress.com/item/1005009385717059.html>
- Übersicht Reifen YEAHRUN 1,0“: <https://de.aliexpress.com/item/1005006160882915.html>
- Übersicht Reifen YEAHRUN 1,9“: <https://de.aliexpress.com/item/1005007511158079.html>
- KINGCOO-Shop: https://www.amazon.de/stores/page/F1821FD9-CF9A-4EFE-A229-B8971D2F923E/search?lp_asin=B0DL2N9X2L&ref=ast_bln&terms=rc%20tire

Quellen

- [1] Ludger Mäsing: *Felgen*. Download der fischertechnik-Designer-Datei im [Download-Bereich zu dieser Ausgabe](#), 2025.

Modell

Großprojekt Seilbahn (Teil 12): Sie lebt!

Tilo Rust

Diese Serie begleitet das Großprojekt „Kuppelbare Einseilumlaufbahn / Doppelmayr (10-MGD)“ vom Anfang bis zur Fertigstellung und darüber hinaus. Sie berichtet von einigen Erlebnissen und Hindernissen sowie der Technik des Projektes und verweist auf unsere hilfreichen Sponsoren.

Totgesagte leben länger! Wie in unserem letzten ft:pedia-Artikel [1] beschrieben fürchteten wir um das Aus der Anlage. Unsere Hilferufe und intensive Arbeit vieler Seilbahn-begeisterter Profis tragen nun Früchte: Kurz vor Redaktionsschluss können wir mitteilen: Die Seilbahn lebt!

Das Aus

Auf der ftc-Süd-Convention 2024 teilte man uns seitens des Museums mit, dass ab Ende des Jahres kein Ausstellungsplatz mehr für das Modell zur Verfügung gestellt werde. Die Fläche sei bis spätestens Dezember 2024 zu räumen.

Nachdem wir diesen Schock eingesteckt hatten, galt es, das Team von den aktuellen Arbeiten am Projekt abzuziehen und alle Kräfte in den Bau der finalen Transportboxen und zugehörigen Transportsicherungen zu stecken. Leider zog die Nachricht auch die letzte Motivation aus den Teammitgliedern, und so blieb nur meine Frau als helfende Hand für die vielen kommenden Stunden harter Arbeit. Zwischenzeitlich halfen mir dankenswerterweise ein Elektriker, ein Schweißer und ein Tischler tatkräftig beim Bau, obwohl sie das Projekt nicht einmal im Museum gesehen hatten.

Doch alle Anstrengung nützte nichts. Selbst die zweite Deadline Ende Juni 2025 mussten wir überschreiten, bis die Boxen fertig waren.

Das Museum lagerte daraufhin die Anlage um. Seitdem standen die Boxen zwar unter einem Regendach, aber an der frischen Luft.

Mir war klar, dass, sobald der Frost einsetzen würde, mit technischen Schäden gerechnet werden muss und bei Starkregen die Anlage nur halbwegs geschützt ist.

Hilferufe

Parallel zu den Arbeiten an der Verpackung startete ich in allen Kanälen einen Hilferuf. Es konnte doch nicht sein, dass das größte fischertechnik-Projekt aller Zeiten mit einem Wert von über 135.000 € verschrottet werden würde. Sollten alle 6.500 Stunden ehrenamtlicher Arbeit, die unser Team bis dato geleistet hatte, nun vergessen sein? Und das, obwohl wir die Seilbahn in etwa 200 Stunden zum Laufen bringen könnten, wenn wir uns zusammenreißen und die Räumlichkeiten hätten?

Der Hilferuf machte die Runde. Nicht nur auf unserem YouTube-Kanal [2], der Mailingliste und unseren direkten Kontakten, sondern auch in der professionellen Szene. So veröffentlichte ihn das SI Magazin [3] und verteilte die Nachricht zusätzlich direkt an ihre Kanäle. Auch andere professionelle Stellen wie die Messeleitung der CCW oder Universitäten verbreiteten die Nachricht.

Strohhalme

Dann, nach dem erwähnten Artikel, meldeten sich die ersten Stellen. Einige Städte, die Seilbahnprojekte planen, ließen sich Informationen über Größe und Bedingungen zukommen, mussten aber letztlich meistens aufgrund politischer Hindernisse das Hilfsangebot zurücknehmen.

Zwei Universitäten meldeten sich. Eine davon ist immer noch im Kontakt mit uns – eine zukünftige Zusammenarbeit und Ausstellung wird gerade verhandelt.

Unser Fokus lag auf Museen. So kamen sogar die Kuratoren des Landesmuseums Technik aus Mannheim zu einer Besichtigung vor Ort, zeigten Interesse, konnten aber lediglich auf lange Sicht genug Platz schaffen, denn ein so großes Museum plant Ausstellungen viele Jahre im Voraus. So sind wir dankbar für das Angebot und bleiben weiter in Kontakt, denn vielleicht kann das Modell in ca. acht Jahren nach Mannheim überführt werden.

Andere Museen mussten leider absagen.

Eine kurzfristige Rettung fanden wir durch den Aufruf jedoch nicht. Und so zog sich die Schlinge immer weiter zu und eine Verschrottung kam näher.

Urbane Rettung



Abb. 1: Logo der Stadt Herne, der rettenden Sponsorin

Dann eine E-Mail aus einer Stadt im Ruhrgebiet: Herne! Wenige Minuten später ein Telefonat. Und Tränen in meinem Gesicht: Die Seilbahn ist gerettet!

Es brauchte dennoch mehrere Monate Verhandlungen und Detailklärungen, bis feststand, dass wir den Leihvertrag unterschreiben können.

Die Stadt Herne plant, eine urbane Seilbahn zu bauen und diese in den ÖPNV einzugliedern. Diese wird zwar aufgrund der Trassenführung eine Pendelbahn sein, vergleichbar mit der Zugspitzbahn, die Technik ist aber zum Teil mit unserem Typ MDG vergleichbar.



*Abb. 2: Logo unseres Sponsors
Ingenieurbüro Schweiger*

Unser Team ist anerkannt als Botschafter urbaner Seilbahnen und wird vor allem in direktem Kontakt mit den Bürgern tätig, um über technische Details von Seilbahnen und den städtischen Nutzen eines solchen Projektes zu informieren. Das Modell wird dafür bei Informationsveranstaltungen der Stadt Herne als zentraler Eyecatcher präsentiert.

Ab Januar 2026 wird also unser Modell in den Räumen einer Multifunktionshalle, der Sporthalle Eickel, eingelagert und zu mehreren Veranstaltungen der Stadt in den öffentlichen Raum transportiert, dort aufgebaut, von unserem Team betreut und nach der Veranstaltung wieder abgebaut und eingelagert. Der Aufwand mag zwar riesig sein, aber jetzt zahlt sich das neue System der Transportboxen aus, die erlauben, dass das Modell innerhalb kürzester Zeit auf- und abgebaut werden kann. Jede Stunde, die meine Frau und ich in das Ausstellungssystem investiert haben, wird sich nun mehrfach auszahlen. Ein ausführliches Video zu diesen Transportboxen veröffentlichen wir auf unserem YouTube-Kanal [2].

Die technische Zeichnung in Abb. 3 zeigt das Profil der geplanten Seilbahn in Herne. Mit 66 m Seilhöhe und der großen nötigen Spannweite wird hier eine Pendelbahn eingesetzt. Doch mit nur zwei Stützen werden mehrere Gleise und Straßen über-

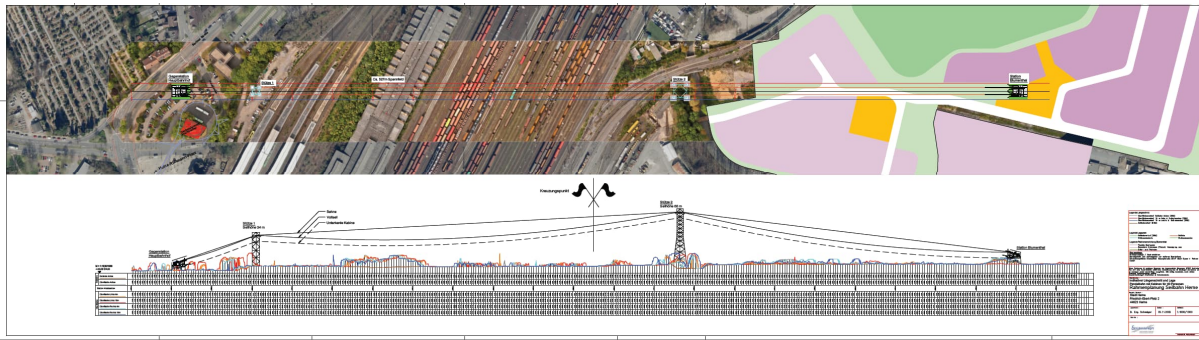


Abb. 3: Profil der geplanten Seilbahn der Stadt Herne

brückt. Damit werden der Hauptbahnhof und das geplante High- und Green-Tech-Quartier miteinander verbunden. Die reine Fahrzeit beträgt dann nur 3,5 Minuten und entlastet damit Pendler und den Verkehr erheblich.

Infostand

Mit diesem Trumpf in der Hand konnte nun auch die ft-Convention 2025 für mich stattfinden. Ich hatte einen kleinen Stand aufgebaut, an dem die zentrale Nachricht

deutlich zu sehen war: „Die Seilbahn lebt – sie braucht deine Hilfe!“

Zu sehen waren einzelne Prototypen von Elementen, die gebaut werden müssen, sowie Material, das wir noch benötigen. Einiges davon stellten mehrere Aussteller in Aussicht – dafür herzlichen Dank. Schon jetzt sind die Materialien teilweise eingegangen und werden von Projektmitgliedern zum Einbau vorbereitet.

Dennoch: Es gibt noch viel zu tun und zu beschaffen. Wer uns unterstützen kann,



Abb. 4: Auf der ft-Convention in Bad Hersfeld zeigte unser Stand außer den Hilfeauffrufen auch einige Prototypen, die später in der Seilbahn verbaut werden müssen. Wir brauchen tatkräftige helfende Hände!



Abb. 5: Verladung

sowohl finanziell, mit benötigten Materialien oder durch Übernahme von Bastelarbeiten (nicht nur fischertechnik, sondern auch Metall oder Holzarbeiten), möge sich bitte umgehend und direkt bei mir melden [4].

Die Gespräche auf der Convention waren durchgehend von wohlwollendem Interesse – auch wenn ich zu diesem Zeitpunkt den genauen Ort des zukünftigen Verbleibs nur als „eine nicht näher genannte Stadt im Ruhrgebiet“ bezeichnen konnte.

Schülerengagement

Kurz vor Leistung der Unterschrift im Vertrag mit der Stadt Herne schlug dann mein Herz noch einmal bis zum Anschlag. Eine Schule für Seilbahnbauer aus Österreich meldete sich. Schüler hatten das Projekt schon 2023 in Mannheim auf der BUGA gesehen und nun der Schule von unserem Hilferuf berichtet. In mehreren Referaten stellten diese dann die Möglichkeiten des Modells für die Schule dar.

Da aber der Abgabetermin für den Vertrag der Stadt Herne nur wenige Tage darauf lag, war Eile angebracht. Und so bemühte sich die Schulleitung in einem spätabendlichen

Telefonat um technische Details, musste dann am darauffolgenden Tag leider unter anderem aus Platzgründen ebenfalls absagen.

Dennoch danke ich auf diesem Wege ganz besonders den Schülern Oskar Leffler und Mika Glasmacher der LBS Hallein für ihr außergewöhnliches Engagement.

Transport

Von Ruhe im Karton war aber nun nicht die Rede. Der Transport musste organisiert werden. Bei einer Transportgröße von 20 Europaletten ist das nicht gerade ein Job für die Deutsche Bundespost. Mehrere Speditionen aber sagten ab oder unterbreiteten keine Angebote, da sie nur Aufträge von Unternehmen annehmen.



Abb. 6: Logo der Spedition, die uns unterstützt

Die Spedition Oswald aus Sinsheim war es schließlich, die uns nicht nur den Transport organisierte, sondern auch preislich entgegenkam. Letztlich tragen aber unsere Hauptsponsoren Doppelmayr und fischer-

technik zu gleichen Teilen die Frachtkosten, während die Stadt Herne Abladen und Einlagerung auch ohne Hilfe aus unserem Team übernimmt. An dieser Stelle Danke an die professionelle Zusammenarbeit aller Beteiligten, speziell Hrn. Fink und dem Team der Blumenthal-Entwicklungsgesellschaft Herne mbH & Co. KG.

Anfang Dezember 2025, also fast ein Jahr später als vom Museum gewünscht, konnten wir schließlich mit Sack und Pack umziehen (Abb. 6 und 7). Ein Video über den Abtransport veröffentlichen wir in den nächsten Tagen; die Einlagerung und die erste Ausstellung zeigen wir dann in einem weiteren Video im Frühjahr 2026.



Abb. 7: Transport

Alles gut verpackt und sicher im LKW verladen ging es nach Herne. Immerhin 20 EPAL-Palettenplätze wurden benötigt. Auch hier bewährte sich das ausgeklügelte und gut beschriftete Verpackungssystem, denn die Stationsmodule passen exakt quer auf einen LKW und können mit einem Gabelstapler einfach und zeitsparend eingehoben werden.

Im Keller verstaut (Abb. 8), aber vor der Verschrottung bewahrt, wartet die Anlage nun, bis wir sie im Januar 2026 auspacken und zur ersten Ausstellung in Herne aufstellen. Reparaturen und Erweiterungen können dabei noch angebracht werden. Hierfür sind viele Helfer nötig. Hat jemand Zeit und Lust? Bitte melden!

Neue Welt, neues Team

Nun also steht die Seilbahn in einer ganz neuen Welt. Zwar wird sie hauptsächlich in einem Keller gelagert und nur selten zu Ausstellungen ans Tageslicht und die Öffentlichkeit kommen – aber gibt es nicht den Spruch „Willst du gelten, mach dich selten“?

Seltenere, aber dafür fokussierte Bautage haben das Potential, unser Team konzentriert an den wichtigen Elementen arbeiten zu lassen. Der Baufortschritt wird natürlich auf Video dokumentiert.

Ein neues Zuhause deutlich weiter im Norden Deutschlands bringt auch einen Wechsel bei den Projektmitgliedern mit sich. Einige hatten sich schon mit dem Aus in Sinsheim zurückgezogen, andere verabschieden sich jetzt nach dem Umzug. Das bietet auch eine Chance, neue Tüftlerinnen und Tüftler und andere helfende Hände zu begrüßen.

Und so biete ich allen begeisterten fischer-technikerinnen und fischertechnikern an, neu in das Projekt einzusteigen. Es gibt viel zu tun, sowohl in Heimarbeit als auch vor Ort in Herne. Sowohl für Einsteigerinnen und Einsteiger als auch für richtige Freaks und für handwerklich begabte nicht-fischertechnik-Tüftler.

Konkret: Am Wochenende 10./11. Januar 2026 wird aufgebaut und am 17./18. wieder eingepackt. In dieser Zeit benötige ich Unterstützer, die nach Herne kommen und am Modell bauen. Wir sind den ganzen Tag am Bauen und verbringen abends gemeinsame Zeit miteinander.

Du hast Zeit, Lust, Interesse und Möglichkeiten? Auch nur einen einzigen Tag? Dann melde dich direkt bei mir und wir sprechen ab, wie wir das umsetzen können [4]. Wir werden doch wohl ein kleines Team zusammen bekommen und in 100 bis 200 Arbeitsstunden die Anlage zum Rundlaufen bringen, oder?



Abb. 8: Einlagerung in Herne

Quellen

- [1] Tilo Rust: *Großprojekt Seilbahn (Teil 11): Die Zukunft*. [ft:pedia 1/2025](#), S. 32–35.
- [2] Technik Fischer – fischertechnik-Seilbahn-Projekt: [YouTube-Kanal](#).
- [3] Seilbahnen International Verlag GmbH: *Seilbahn-Modell braucht dringend Obdach*. Im [SI-Magazin](#), 24.06.2025.
- [4] Kontakt zur Projektgruppe „Seilbahn“, Projektleiter Tilo Rust, Schifferstadt, E-Mail: ft.seilbahn@gmail.com

Modell

Schwertransporte

Ludger Mäsing

In einem meiner Bücher über Schwertransporte habe ich ein sehr interessantes Modell gesehen. Es handelt sich dabei um eine Skizze, die in dem Buch „Machines et Inventions“ in Frankreich vor mehr als 300 Jahren veröffentlicht wurde. Es zeigt eine Transportationalisierung durch ein Walzenfahrwerk. Dieses und ein weiteres Gerät aus dem Buch habe ich nachgebaut.

Modell 1: Walzen-Fahrwerk

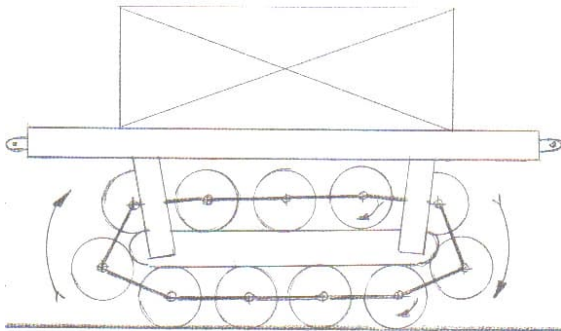


Abb. 1: Skizze im Buch

Abb. 1 zeigt das Prinzip eines interessanten Fahrwerks: Alle Walzen sind gelenkig miteinander verbunden. Sie bilden eine Kette, die unter der Transporteinrichtung durchläuft. Dieses Modell habe ich mit fischertechnik nachgebaut. Was mich dabei besonders überrascht hat, ist die Leichtigkeit der ganzen Konstruktion. Die kommt daher, dass das gesamte Fahrwerk auf den Rollen „abrollt“ und kaum nennenswerte Gleitreibung auftritt – genial!

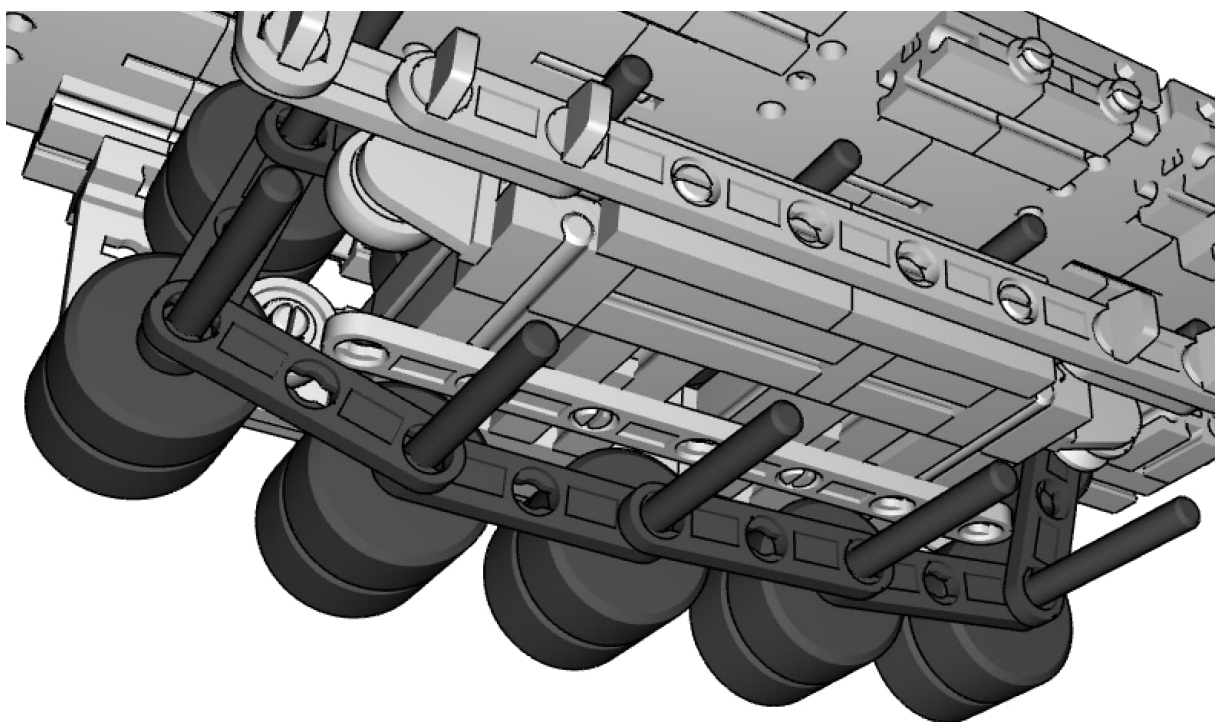


Abb. 2: Aufbau der „Kette“

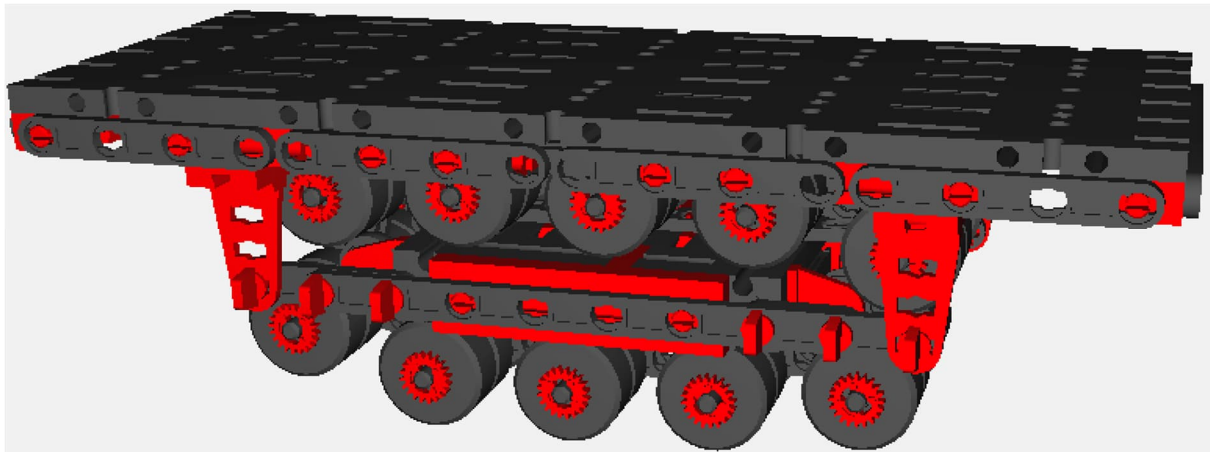


Abb. 3: Das fischertechnik-Modell des Walzenfahrwerks

Mein Modell hat eine Walze weniger als das Original. Es kann aber durch Versetzen der beiden Halterungen beliebig verlängert werden. Das Modell gibt es auch als Bauanleitung für den fischertechnik-Designer zum Download [1].

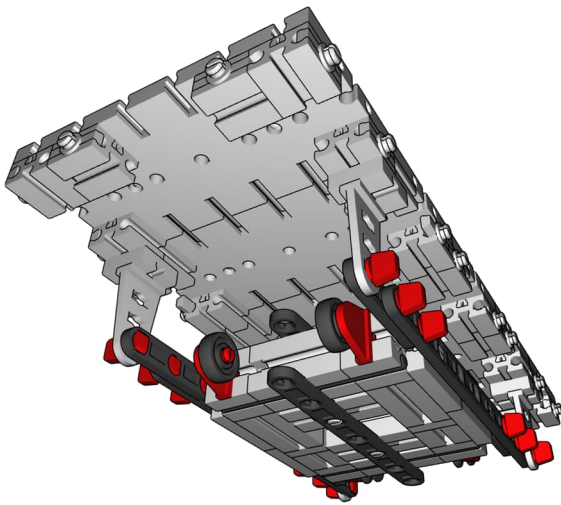


Abb. 4: Blick unters Fahrwerk ohne die Fahrkette

Modell 2: Schwertransport

Im gleichen Buch ist ein weiterer, sehr interessanter Schwertransport abgebildet. Er stammt von 1967 und wurde durch die holländische Firma „big lift“ im ägyptischen Wüstensand verwendet. Dieses Modell habe ich mit über 6.700 fischertechnik Bauteilen maßstabsgetreu nachgebaut und 2015 auf einer Modellschau aus-

gestellt. Das Bauen des Transportwagens war mittels der Zeichnung nicht schwer.

Ein größeres Problem waren die beiden Bulldozer. Nach langer Suche im Internet fand ich heraus, um welche Typen es sich dabei genau handelte. Genaue Maße konnte ich aber nicht finden, weil die Maschinen einfach schon zu alt waren.

Erst nach weiterer, langer Suche bin ich fündig geworden. Im Technik-Museum



Abb. 5: Das Buch „Schwertransporte“ [2]

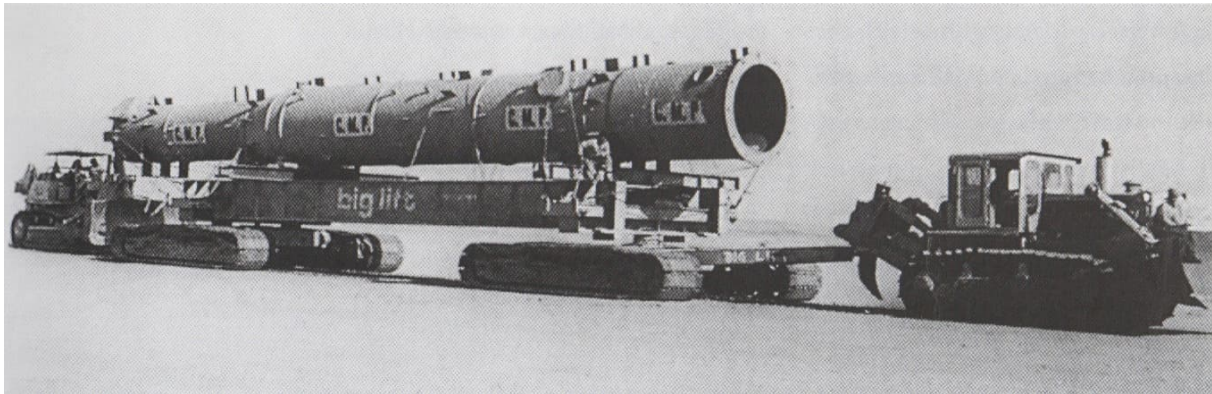


Abb. 6: Bild des Schwertransports im Buch

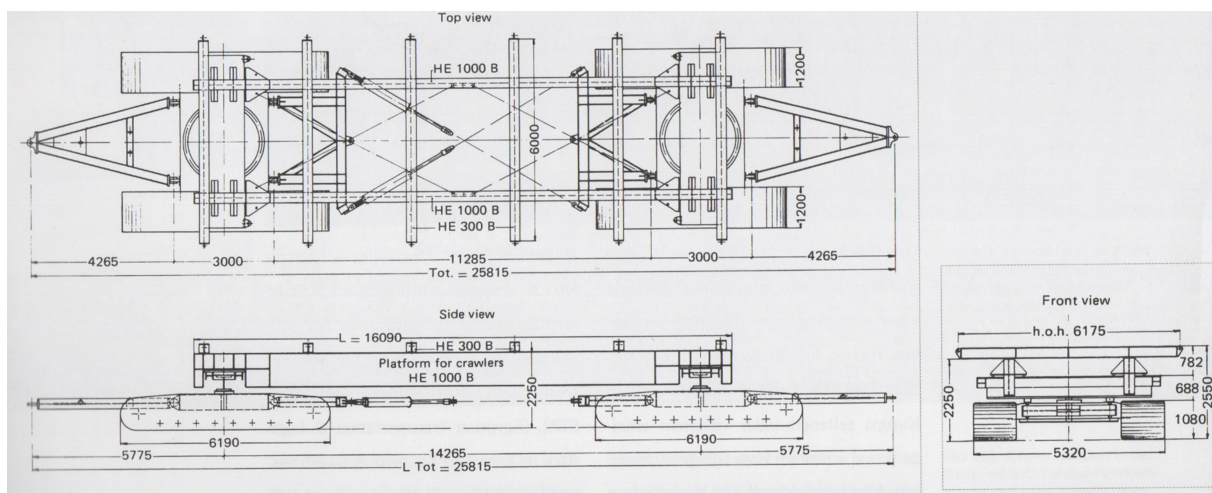


Abb. 7: Technische Zeichnung des Schwertransports aus dem Buch

Sinsheim stand ein solches Gerät wohl noch herum. Ein sehr netter Mitarbeiter war so freundlich und hat es für mich vermessen. Somit konnten die Bulldozer doch noch im richtigen Maßstab nachgebaut werden.

Von diesem Modell gibt es ebenfalls eine fischertechnik-Designer-Datei. Sie ist im fischertechnik-Designer unter *Datei*, *Online-Bibliothek* und dort unter *Fahrzeuge*, *Schwertransport* zu finden.

Mein Dank gilt Herrn Erich Hoepke dafür, dass ich die Abbildungen aus dem Buch verwenden durfte.

Quellen

- [1] Ludger Mäsing: *Designer-Datei zum Walzen-Fahrwerk*. Im [Download-Bereich der ft-Community](#), 2025.
- [2] Erich Hoepke, Gini Koster: *Schwertransporte auf dem Weg durch die Jahrhunderte – Faszination Baumaschinen*. Motorbuch-Verlag Stuttgart, 2003.

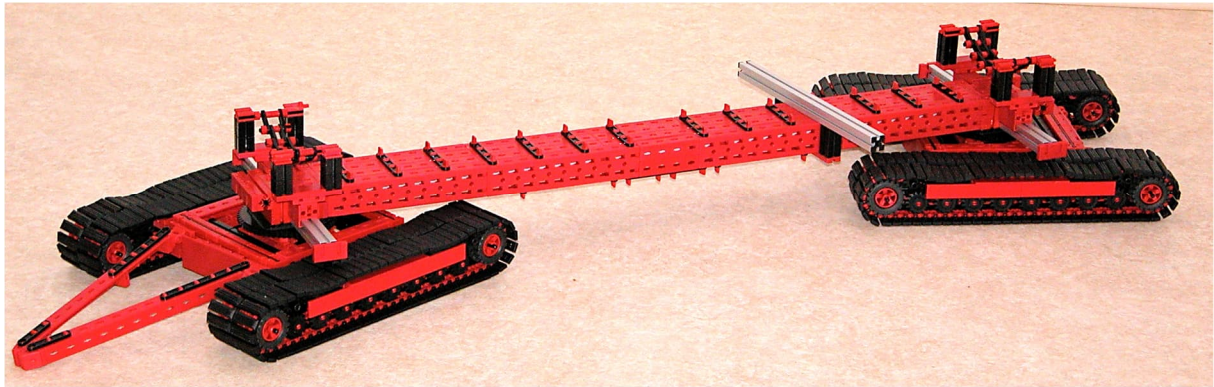


Abb. 8: Der Schwerlast-Anhänger



Abb. 9: Der Schwerlast-Zug



Abb. 10: fischertechnik-Designer-Ansicht des Modells

Getriebe

Gegenläufige Propeller

Fabian Haas

Gegenläufige Propeller finden sich immer wieder im Flugzeugbau, aber auch im Schiffsbau. Es sind allerdings Exoten, die nur ausnahmsweise realisiert werden. Dabei sind zwei eng hintereinander liegende, koaxiale Propeller über ein Getriebe miteinander gekoppelt. Obwohl sie von einem Motor angetrieben werden, laufen sie in umgekehrter Richtung. Ich stelle hier eine Lösung in fischertechnik für diese Konstruktion vor.

Es gibt verschiedene Vorteile dieser Konstruktion. Der erste ist, dass ein einzelner Propeller ein Drehmoment erzeugt, wie jedes rotierende Objekt. Dagegen muss mit der Flugsteuerung gearbeitet werden, was wiederum Energie verbraucht. Liegen die beiden Propeller hintereinander und rotieren sie gegeneinander, gleichen sich die beiden Drehmomente aus. Außerdem soll die aerodynamische Effizienz bis zu 16 % höher liegen als bei einzelnen Propellern [1].

Den Vorteilen stehen allerdings Nachteile gegenüber: Das Getriebe ist deutlich komplexer und schwerer als bei Einzelpropellern. Obwohl schon 1907 patentiert [1], gibt es bis heute nur wenige Flugzeugtypen, die diese Propeller verwenden: Die Antonov An-22 und die Tupolev Tu-95 sind die bekanntesten (wozu es Dank der Planespotter Community reichlich Material auf YouTube gibt). Daneben gab es kaum bekannte, kleinere Typen wie die Westland Wyvern,



Abb. 1: Die Antonov An-22 (UR-09307) bei einem Besuch in Leipzig, am 26 November 2016

Martin-Baker MB 4, Avro Shackleton und Fairey Gannet [2, 3, 4, 5].

Neuerdings wird dieses Konstruktionsprinzip vor allem wegen der Effizienzsteigerung durch weniger Kraftstoffverbrauch wieder intensiver untersucht [6]. Im Schiffbau zählen ähnliche Vorteile bei der Richtung der erzeugten Strömung und der Gesamteffizienz [7].

Inspiziert durch eine CAD-Arbeit auf YouTube (Mechanical CAD Designer, @learnEngineerings), habe ich mich an diese Konstruktion der gegenläufigen Propeller gesetzt und versucht, eine Lösung in fischertechnik zu finden. Nach einigen Fehlversuchen (Verbastelungen) hat es geklappt.

Mit 2×2 durch Federnocken verbundenen Innenzahnradern Z30 war die Basis für die Rotorblätter gelegt. Damit sind die Innen-

zahnräder Z30 zwar miteinander verbunden, aber noch nicht zentriert bzw. ausgerichtet. Dies gelingt für das vordere Paar der Innenzahnräder mit einer Drehscheibe 60 und Rastachsen 20 (mit Riegelscheiben), die die Drehscheibe und die Innenzahnräder zentrieren und fest verbinden. Die Drehscheibe auf der durchgehenden Metallachse stellt nicht nur die Zentrierung sicher, sondern hält den ganzen Mechanismus zusammen, in dem die Metallachse bis zur Basis durchgeht und mit einer Zangenmutter und Spannzange festgezogen wird.

Das hintere Paar der Innenzahnräder Z30 wird durch innen gelegene Ritzel Z10 und Seilrollen 12 in Position gehalten. Die Seilrollen laufen auf Schlauchadaptern D4 und sind mit Riegelscheiben gesichert. Anfangs versuchte ich, Ritzel Z10 in den verschiedenen Versionen als Rastritzel u. a. zu nutzen. Allerdings erhöhten Spannzangen oder

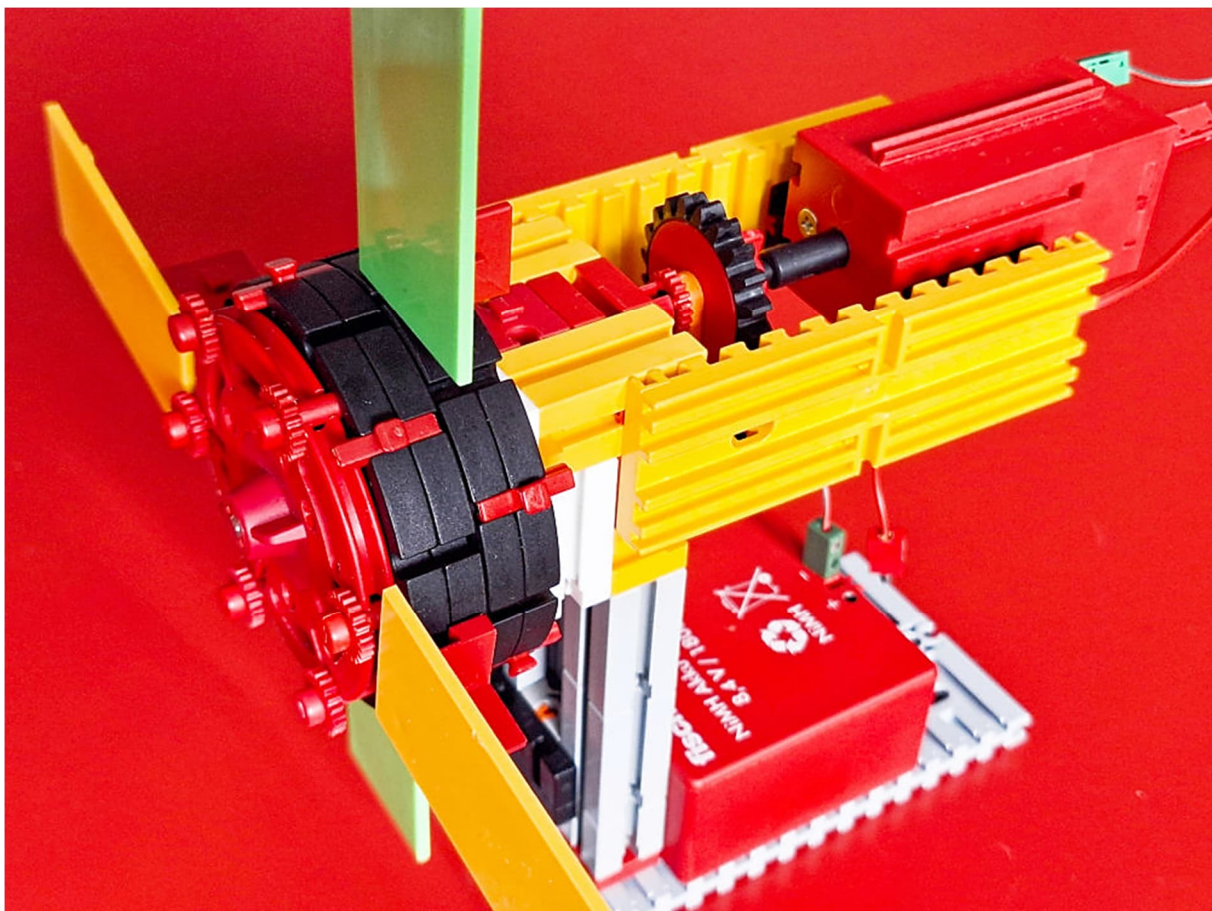


Abb. 2: Aufbau mit Innenzahnradern

Rastachsen den Platzbedarf und ich musste, um eine Zentrierung der Innenzahnräder zu erreichen, die Seilrollen nach unten schieben, sonst würde die Innenzahnräder zu sehr wackeln. fischertechnik weist an dieser Stelle zu viel Spiel auf.

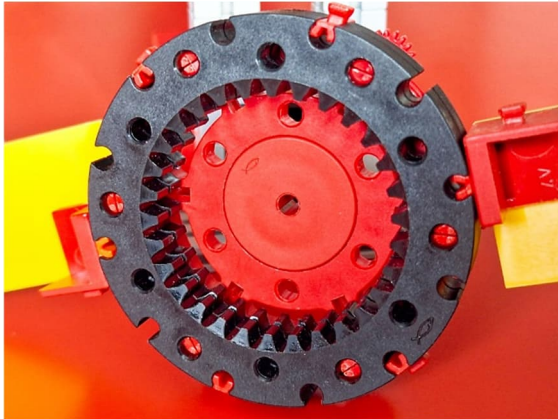


Abb. 3: Innenansicht

Aber nun zum Antrieb. Der passte nicht mehr in die Innenzahnräder, sondern musste von hinten zugeführt werden. Dazu boten sich Rastritzel Z10 auf Rastachsen an, die in zwei verschiedenen Höhen in die Innenzahnräder eingreifen. Die I-Strebe 30 (in Orange) und rote Abstandsringe sichern die Lage in den Innenzahnradern. Nach hinten gehen die Rastachsen – und die zentrale Metallachse – durch Bausteine 15 mit Bohrungen durch.

Die zentrale Metallachse ist hinten mit Spannzange und Zangenmutter fixiert; eine I-Strebe 30, Abstandsringe und Riegelscheibe sichern den Abstand und reduzieren das typische fischertechnik-Spiel in der Mechanik. Die Rastachsen bekamen ineinandergreifende Zahnräder Z20, und so wird die gegenläufige Bewegung erzeugt, wenn eine der Rastachsen angetrieben wird. Die vier Bauplatten 15×15 (in Weiß) sind die eigentlichen Kontaktflächen des hinteren Innenzahnrad mit der Auflage und reduzieren die Reibung.

Der Aufbau der Basis benötigt nur wenige Bausteine und ergibt sich aus den Bildern.

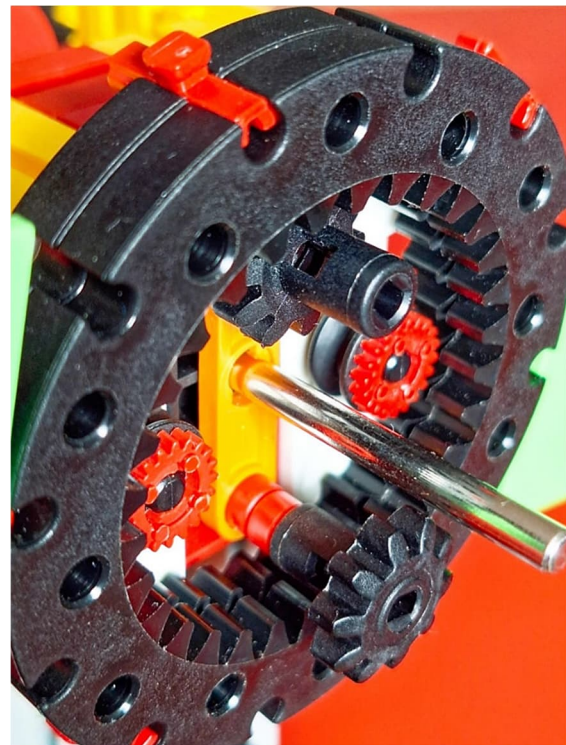


Abb. 4: Antrieb

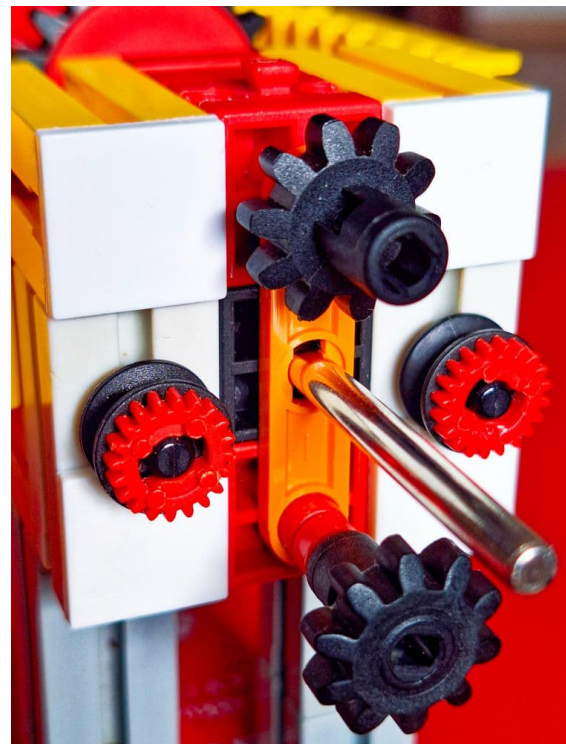


Abb. 5: Reibungsarme Lagerung des Innenzahnrad

Der Rest war einfach: ein Motor, über eine Rastkupplung an einer der Rastachsen angeflanscht, eine Motorhalterung über

eine Bodenplatte 30×90 und mit Bausteinen 7,5 (in Schwarz) konstruiert. Das Ganze wird über die (für das eigentliche Problem nicht essenziellen) grauen Bausteine 30 und auf der Grundplatte 120×60 hoch genug aufgestellt, damit die Rotoren ungehindert drehen können. Der Akku verleiht noch etwas zusätzliche Stabilität. Das Modell ist auf meinem YouTube-Kanal im Bewegtbild zu sehen [8].

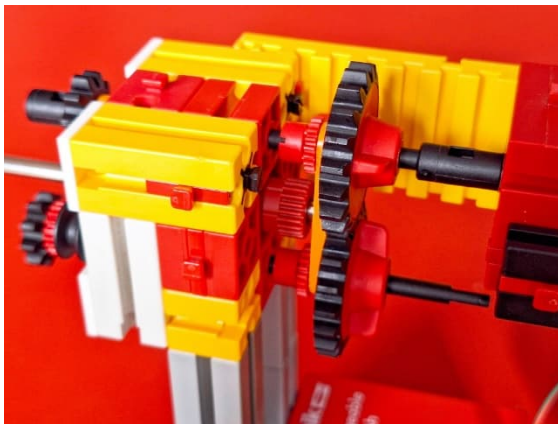


Abb. 6: Antrieb und Halterung des Getriebes

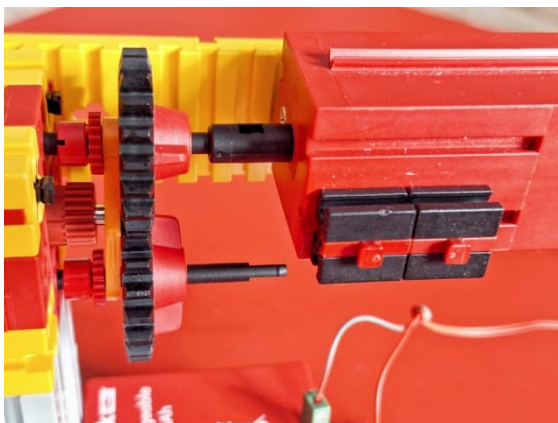


Abb. 7: Antriebsmotor

Mit der Positionierung des gesamten Antriebs kann die Luft frei strömen und wird nur wenig durch die Aufhängungen gestört. In weiteren Ausbaustufen könnte man zwei Motoren installieren, die über die beiden Zahnräder Z20 gekoppelt bleiben. Alternativ ließen sich die Rotoren unabhängig antreiben. Mit dem ROBO-Controller ließen sich dann zwei Schrittmotoren in Gleichtakt bringen oder aber mit ganz verschiedenen Geschwindigkeiten ansteu-

ern (dann natürlich auch in die gleiche Richtung).

Eine lebhafte und äußerst fruchtbare Diskussion zum Thema gibt es auch im [fischer-technik-Forum](#) [9].

Fotografie

Die Fotos und Videos sind alle mit einem Mittelklasse-Handy gemacht, diesmal ohne Stativ, weshalb die Videos etwas wackeln. Der rote Hintergrund ist ein kleiner roter Beistelltisch auf Rollen. Den schiebe ich dann immer in Richtung Wohnzimmerfenster, damit es schön hell wird (Vorhänge offen). Kunstlicht nutze ich nie. Eine einfache Bildbearbeitung nutze ich aber schon, um die Helligkeit besonders in ganz dunklen Bereichen (schwarze Zahnräder) anzuheben und den Farbeindruck zu korrigieren.

Bis man das sieht, was man zum Verständnis der Konstruktion wirklich braucht, wird probiert, die richtige Perspektive zu finden. Das bedeutet oft das teilweise Auseinandernehmen des Modells. Es sollen immer genug Details gezeigt werden, aber nicht so nah rangegangen werden, dass man nicht mehr versteht, wie das Detail im gesamten Modell liegt – ein Grundsatz in meiner Fotografie. Das kann schon mal knifflig werden, wobei meine Modelle eher klein und übersichtlich sind. Die verschiedenen Farben der Bausteine setze ich bewusst ein, um einzelne Teile und Baugruppen herauszuheben bzw. gegeneinander abzugrenzen (wie die grauen Bausteine 30, die ja eigentlich nichts mit der Konstruktion zu tun haben). Das hilft mir selbst sehr beim Verstehen der Bilder.

Viel Spaß beim Nachbau und der Weiterentwicklung!

Quellen

- [1] Wikipedia: [Contra-rotating propellers](#).
- [2] Wikipedia: [Westland Wyvern](#).
- [3] Wikipedia: [Martin-Baker MB 5](#).
- [4] Wikipedia: [Avro Shackleton](#).
- [5] Wikipedia: [Fairey Gannet](#).
- [6] Deutsches Zentrum für Luft und Raumfahrt: *Gegenläufiger Rotor soll Luftfahrtantriebe noch effizienter machen*. Auf [dlr.de](#), 2023.
- [7] Briag Merlegt: *Doppelpropeller an einem Schiff: Was bringt die Gegenrotation?* Auf [boote.com](#), 2024.
- [8] Fabian Haas: *Gegenläufiger Propeller -- Counter-rotating Propellers*. Auf [YouTube](#), 2025.
- [9] Fabian Haas: *Propellerverstellung, Diskussion im fischertechnik-Forum* der ft-Community, 2025.

Alle Fotos und Filme in diesem Beitrag stammen von Fabian Haas. Verwendung nur mit schriftlicher Zustimmung.

Getriebe

Zyklisch ungleichförmige Übersetzung mittels Kardanfehler

Stefan Falk

Getriebe, die eine gleichförmige Drehbewegung in eine periodisch ungleichförmige umsetzen, wurden schon öfters besprochen – seien es die Konstruktionen von Wilhelm Klopmeier, die ungleichförmigen 3D-Druck-Zahnräder von Esther Mietzsch oder meine Getriebe mit einem beweglichen Zwischenzahnrad. Dieser Beitrag zeigt eine Möglichkeit, den normalerweise unerwünschten Kardanfehler genau zu diesem Zweck zu nutzen.

Einige bisher bekannte Varianten

Klopmeier-Getriebe

Wilhelm Klopmeier erfand 2004 – mit fischertechnik! – einfach zu konstruierende zyklisch variabel übersetzende Getriebe, und sie waren in Ingenieurskreisen und unter fischertechnik-Fans in aller Munde [1, 2, 3]. Natürlich gab es mehrere Fans, die diese Getriebeart nachbauten und studierten.



Abb. 1: Ein Original-Klopmeier-Getriebe [1]

Elliptische Zahnräder

Esther Mietzsch stellte auf verschiedenen Conventions ihre im 3D-Druck entstandenen nicht-kreisförmigen Zahnräder vor.



Abb. 2: Esthers ungleichförmig übersetzende Zahnräder [4]

Bewegliches Zwischenzahnrad

An einem verregneten Samstag mit Freunden im September 2022 – natürlich mit dem Wohnzimmer-Dienstreisen-

Urlaubs-Notfallkasten [5] dabei – kam ich auf die Getriebevariante mit dem beweglichen Zwischenzahnrad, die in [6] ausführlich besprochen wurde (Abb. 3).

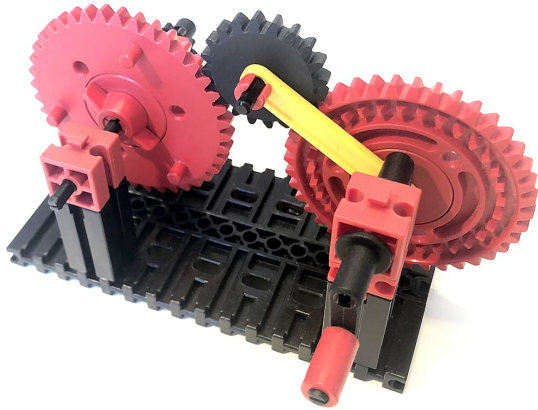


Abb. 3: Zyklisch variable Übersetzung mit Zwischenzahnrad

Der Kardanfehler

Kardangelenke [7, 8] wie das ältere fischer-technik-Kardan ([31044](#)) oder das jüngere Rast-Kardangelenk (bestehend aus zwei [35971](#) und einem [35972](#)) können Drehbewegungen um die Ecke leiten. Dabei tritt

allerdings ein unerwünschter *Kardanfehler* auf: Die Abtriebswelle dreht sich nicht mit konstanter Winkelgeschwindigkeit, wenn die Antriebswelle dies tut.

Kardanfehler vermeiden

Dieser Effekt kann neutralisiert werden, wenn man zwei solcher Kardangelenke – korrekt ausgerichtet – hintereinander betreibt. Die in diesem Sinne korrekte Ausrichtung erreicht man mit den fischer-technik-Rast-Kardangelenken automatisch, wenn man sie einfach mit einer Rastachse dazwischen verbindet. Man beachte, wie in Abb. 5 die beiden Kardangelenke jeweils sorgfältig so ausgerichtet wurden, dass sich die Kardanfehler gegenseitig aufheben. Abb. 6 zeigt dasselbe Prinzip mit Rast-kardangelenken aufgebaut.

Kardangelenke und der Kardanfehler werden auch ausführlich in den hobby-Begleitbüchern besprochen und am Modell demonstriert (siehe Abb. 4 und 5 sowie [9]).

Anordnungen wie in Abb. 5 und 6 gezeigt funktionieren auch, wenn man die beiden

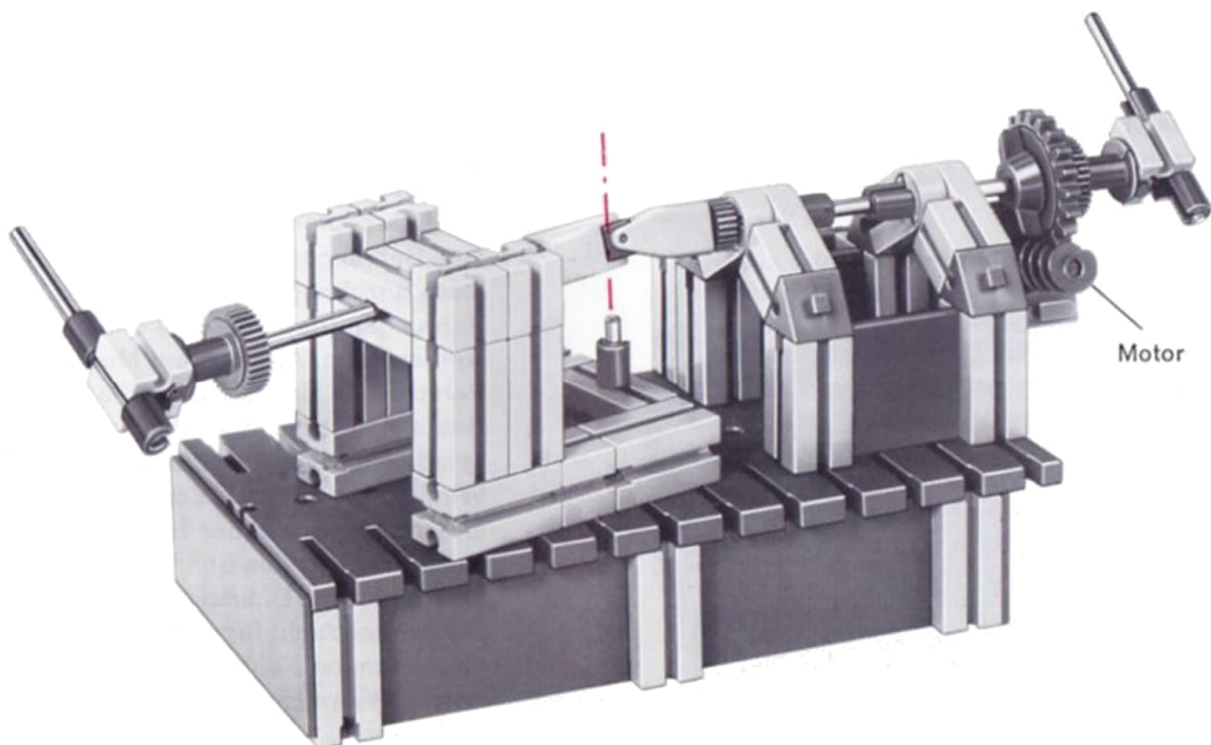


Abb. 4: Aufbau aus dem hobby-2-Begleitbuch Band 2 zum Sichtbarmachen des Kardanfehlers [8]

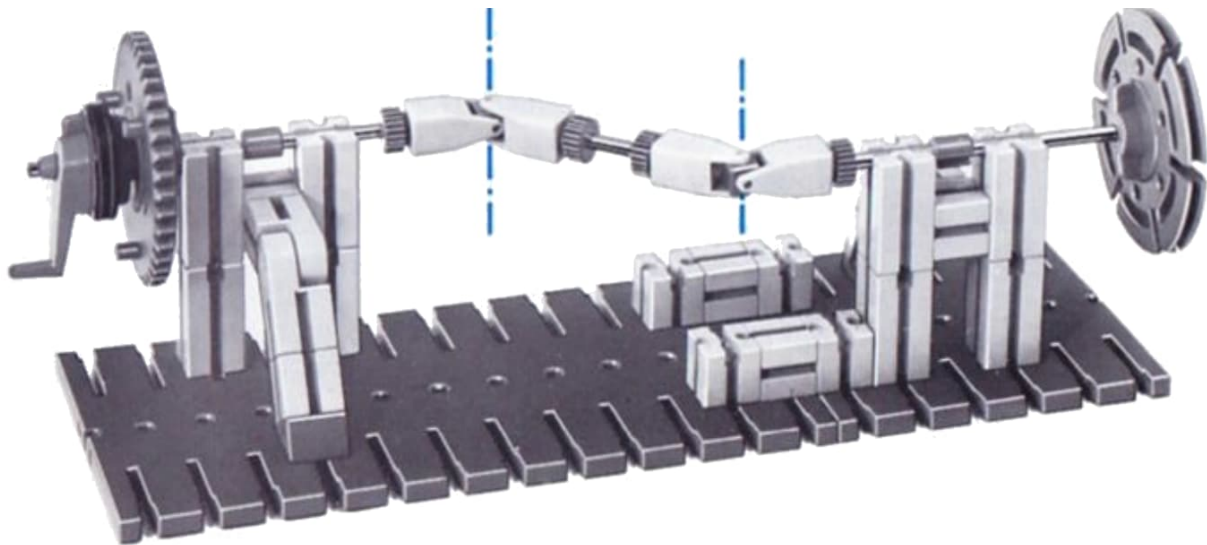


Abb. 5: Anordnung von zwei Kardangelenken so, dass sich ihr Kardanfehler gegenseitig aufhebt [8]

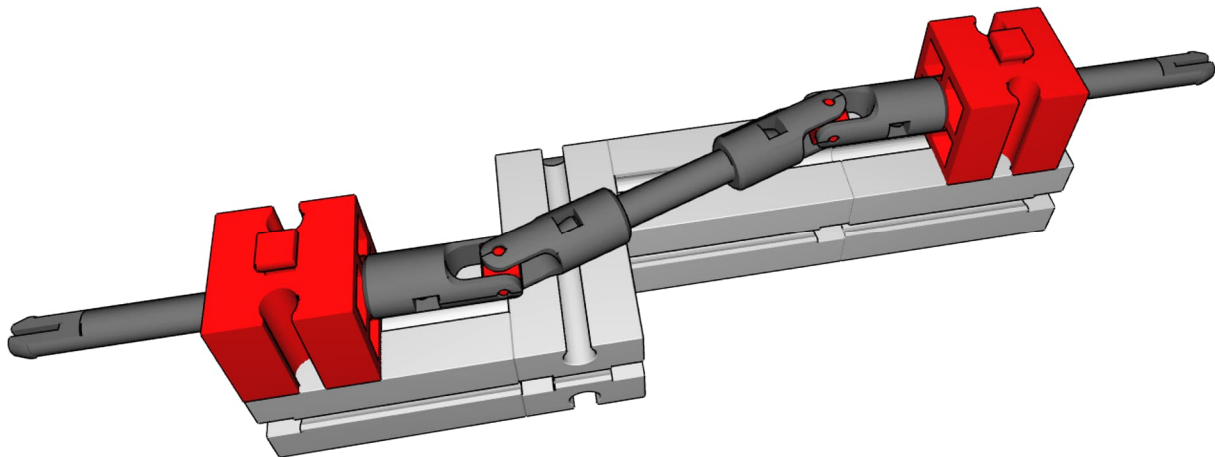


Abb. 6: Rastkardangelenke in der gleichen, getriebetechnisch korrekten Anordnung

Kardangelenke um den gleichen (!) Winkel in dieselbe Richtung anstatt gegenseitig schräg verlaufen lässt. Die Anordnung in Abb. 7 zeigt einen Aufbau mit Rastkardangelenken, in dem beide Kardangelenke um denselben Winkel – hier 45° - geneigt werden. Auch hier heben sich die Kardanfehler der beiden Gelenke gegenseitig auf.

Kardanfehler absichtlich verstärken

Wenn wir aber doch diese ungleichförmige Übertragung der Drehbewegung gerade wollen, können wir die beiden Kardangelenke doch genau „verkehrt“ zueinander

ausrichten. Damit verstärken wir den normalerweise unerwünschten Effekt sogar, um unser Ziel zu erreichen.

Bei den Rast-Kardangelenken brauchen wir dazu eine Anordnung, die die beiden Enden einer Rastachse um 90° gegeneinander verdreht. Das leistet ein Aufbau wie in Abb. 8.

Wenn wir an einer der Achsen drehen, dreht sich die abgehende Achse *sehr stark* ungleichförmig. Kein Wunder: Die ungleichförmige Übersetzung nach dem ersten Kardangelenk wird ja durch das zweite, absichtlich „falsch“ angeordnete, gerade nochmal wirksam und somit verstärkt.

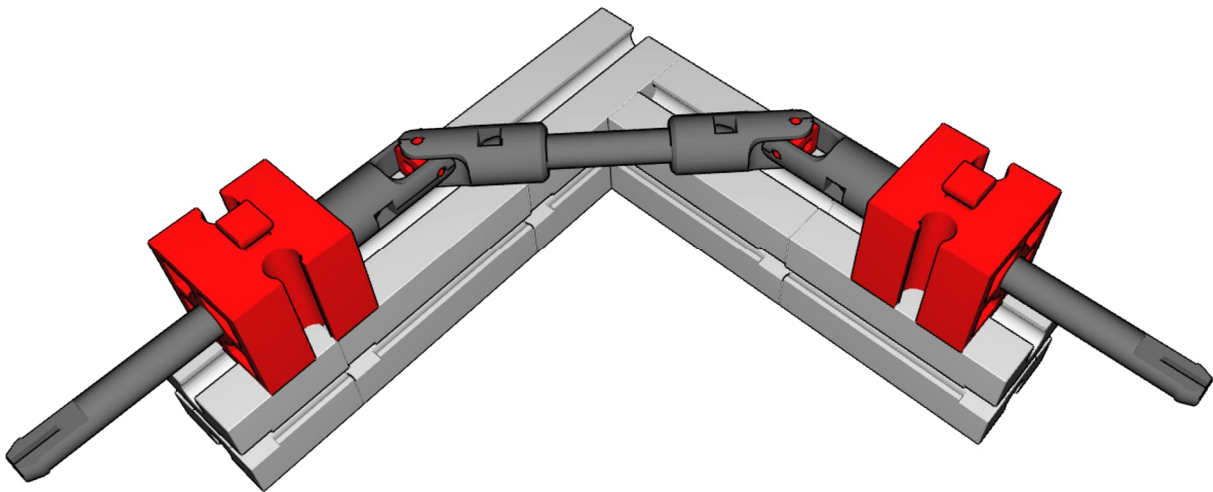


Abb. 7: Rastkardangelnke gewinkelt – so aufgebaut heben sich die Kardanfehler ebenfalls auf

Die beiden durch einen Federnocken verbundenen, aber um 90° gegeneinander verdrehten Bausteine 7,5 sorgen dafür, dass die daran angebauten „Rastachsen mit Platte“ ([130593](#)) die richtige Verdrehung der beiden darauf aufgesteckten Kardangelnke bewirken.

Solange die Anordnung wie in Abb. 8 dargestellt gerade durchgestreckt ist, verhält sie sich wie eine einzige durchgehende Achse. Sobald sie aber analog zu Abb. 6 oder 7 geneigt wird, verstärken sich die Kardanfehler gegenseitig. Das Resultat ist eine stark ungleichförmige, sich aber zyklisch wiederholende Drehbewegung an der (willkürlich gewählten) Abtriebswelle gegenüber der Antriebswelle.

Das Modell

Abb. 9 zeigt ein Demonstrationsmodell, das diesen Effekt ausnutzt. Der grundsätzliche Aufbau ist wie folgt:

- Ein Motor treibt, gänzlich unspektakulär, eine Achse an. Insbesondere dreht er *gleichmäßig* daran.
- An einer Seite befindet sich ein 90° -Versatz wie in Abb. 8. Wozu der im Modell dient, sehen wir später noch.
- An beiden so entstandenen Enden lassen wir je zwei 45° gewinkelte Rastkardangelnke folgen,
- danach jeweils wieder einen 90° -Versatz und schließlich
- an beiden Enden das abschließende Kardangelnke.

So haben wir zwei sich stark ungleichförmig drehende Abtriebswellen, die außerdem noch durch den mittleren 90° -Versatz (beim Motor, auf den wir zurückkommen wollten) genau gegensinnig ungleichförmig drehen: Wenn sich eine der Wellen gerade langsam dreht, dreht sich die andere gerade schnell, und umgekehrt.

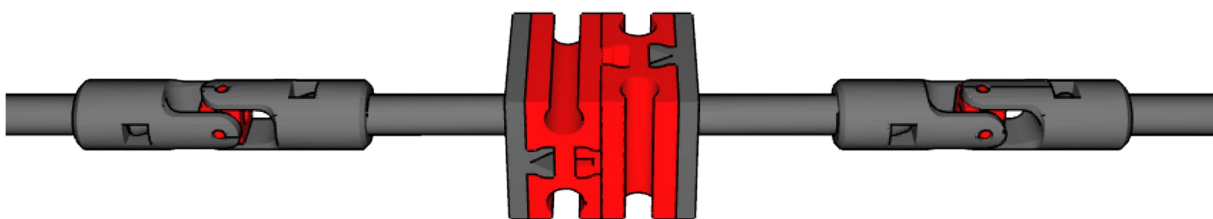


Abb. 8: Zwei Rastkardangelnke mit einem 90° -Versatz dazwischen

In Abb. 9 sehen wir den Aufbau:

- Auf der Bauplatte 500 sitzt hinten mittig der XS-Motor, von dem die Kardanstrecken links und rechts nach vorne laufen. Abb. 10 zeigt das in Großaufnahme.
- Rechts wird eine Drehscheibe mit sechs aufgesteckten Rollenlagern 15 ([37636](#)) durch die Übersetzung von einem Z10 auf ein Z30 bei jeder *halben* Umdrehung der Welle um $1/6$, also 60° , gedreht. Warum bei jeder halben Umdrehung? Weil die Kardangelenke jede halbe Umdrehung genau dieselbe Bewegung vollführen. Für eine ungleichförmige Periode braucht es also nur eine halbe

Achsdrehung. Eine ganze würde *zwei* ungleichförmige periodische Drehbewegungen ergeben. Abb. 12 zeigt das im Detail.

- Die andere Kardanstrecke macht eine Übersetzung durch Z20 auf Z10. Bei *jeder halben* Drehung des letzten Kardans links in Abb. 13 ergibt sich also eine *ganze*, ungleichförmig schnelle Drehung. Die wird auf einen Exzenter gegeben (siehe Abb. 14), das eine Schubstange periodisch ins Gebiet der Rollenböcke der Drehscheibe rechts führt und wieder herauszieht (Abb. 11).

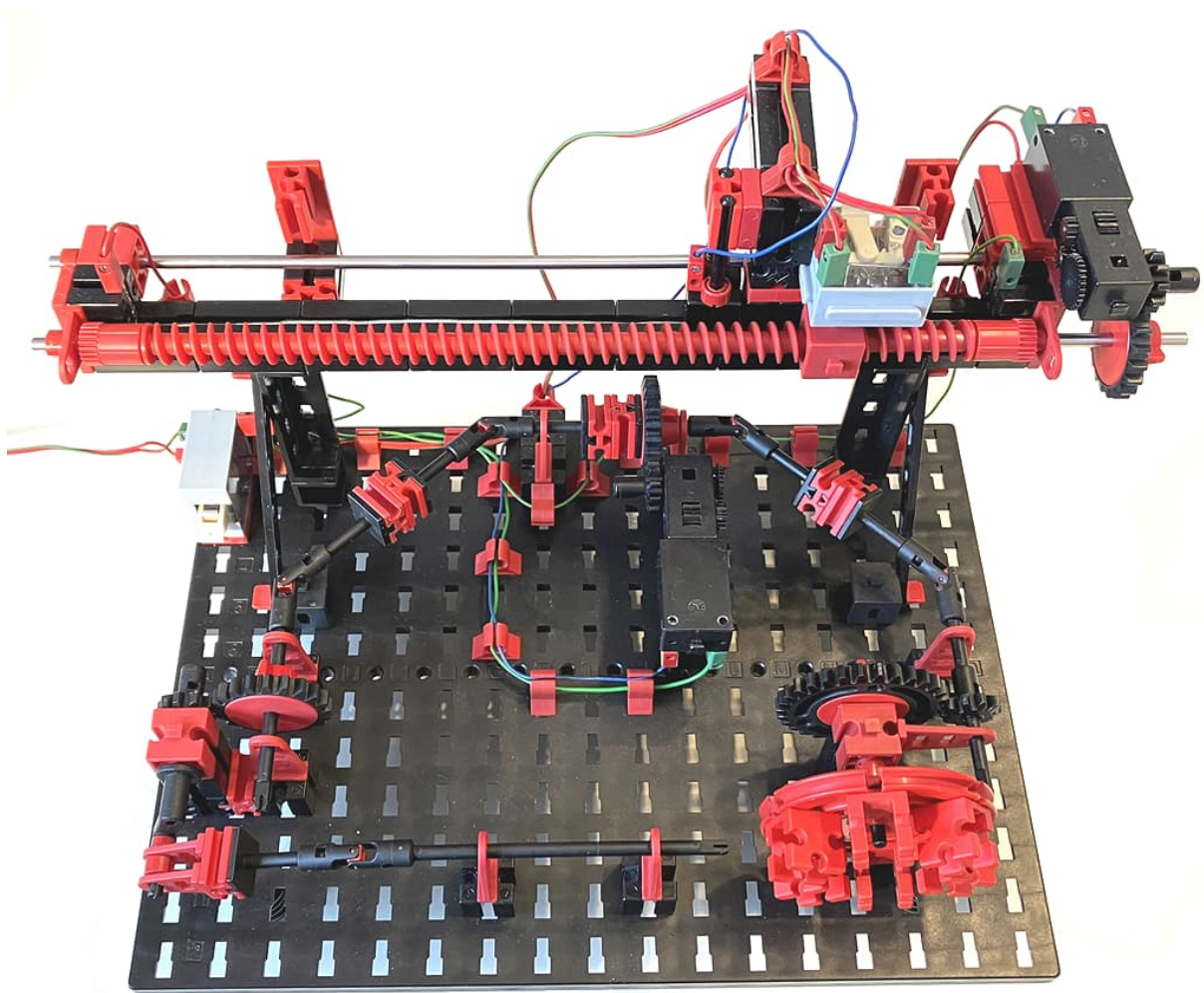
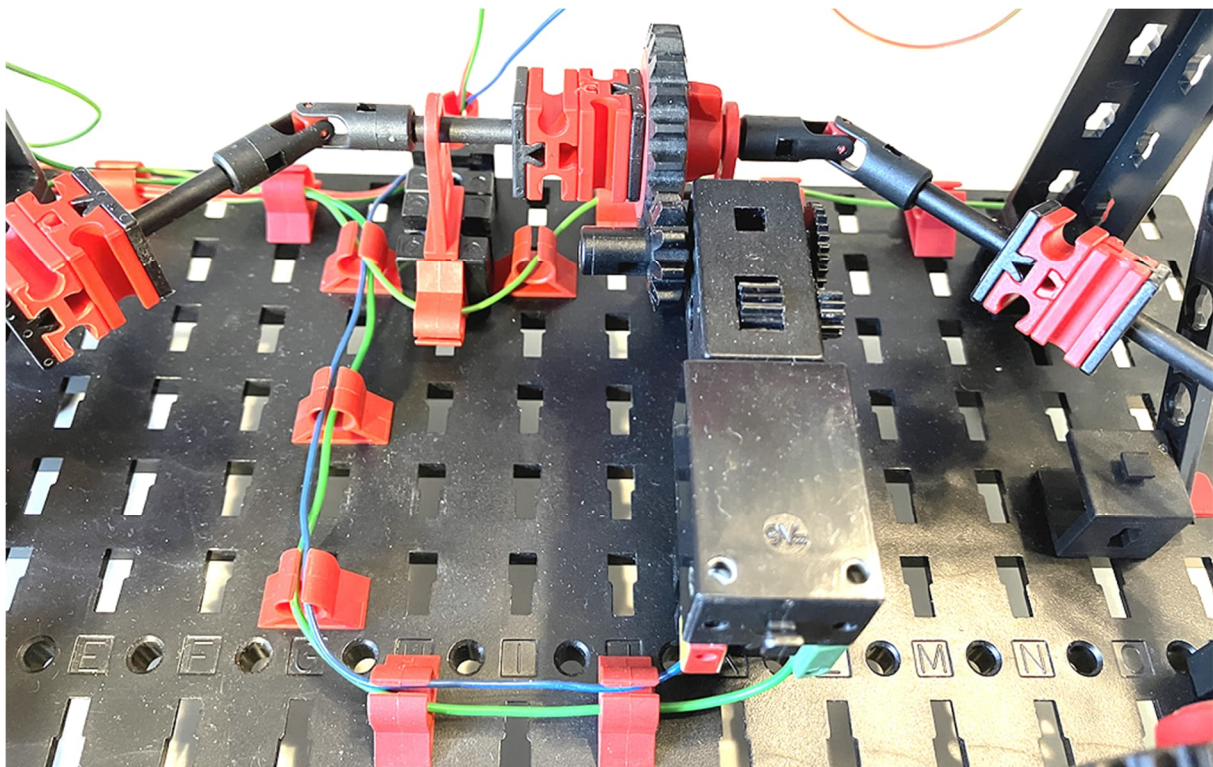


Abb. 9: Das komplette Modell. Der wesentliche Teil befindet sich auf der Bauplatte. Die darüber stehende Elektromechanik mit der langen Schnecke ist nur eine Ergänzung.



*Abb. 10: Der Antriebsmotor mit dem mittleren 90°-Versatz-Element.
Von dort gehen die Kardanstrecken ab.*

- Oberhalb des Modells schwebt eine kleine elektromechanische Steuerung, die später noch beschrieben wird. Sie dient nur einem interessanteren Dauerbetrieb des Modells, hat aber mit der Kardanfehler-Funktion nichts zu tun.

Die ungleichförmigen Drehbewegungen bewirken nun Folgendes:

- Die Drehscheibe dreht sich immer recht schnell um $1/6$ Umdrehung (60°) und bleibt dann relativ lange Zeit fast ganz still stehen – so stark haben sich die Kardanfehler der beiden Kardangelenke zwischen Motor und Drehscheibe verstärkt.
- Genau wenn die Drehscheibe stillsteht, dreht sich die Exzenter-Kurbel sehr schnell, und zwar um eine volle Umdrehung. Wenn man die Drehscheibe richtig justiert hat (dazu ist es nützlich, sie nicht ganz fest auf ihre Achse zu schrauben), trifft die Schubstange immer genau dann in den Hohlraum des gerade anstehenden

Drehscheiben-Rollenbocks, wenn dieser stillsteht.

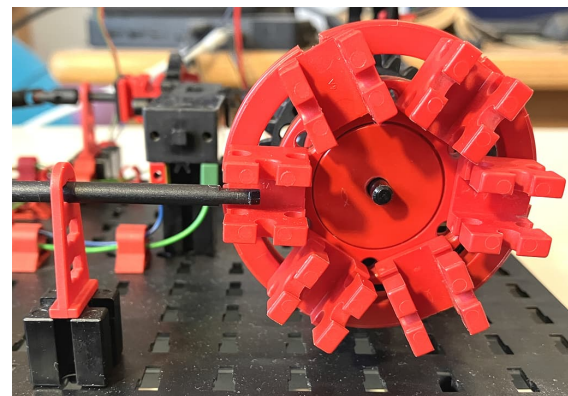


Abb. 11: Die Drehscheibe. Die von links kommende Schubstange soll immer genau in die Hohlräume der sechs Rollenböcke treffen.

Das funktioniert bei sehr langsam laufendem Motor – sodass man die Bewegungen wunderbar studieren kann – als auch bei recht schneller Drehung – bei der man staunen kann, wie präzise die Schubstange in die Rollenböcke eingreift und sich rechtzeitig vor der nächsten Drehung der Dreh-

scheibe zurückzieht. Auf dem Video unter [10] lässt sich das gut beobachten, aber natürlich noch viel besser, in dem man diese Mechanik nachbaut und selber studiert.

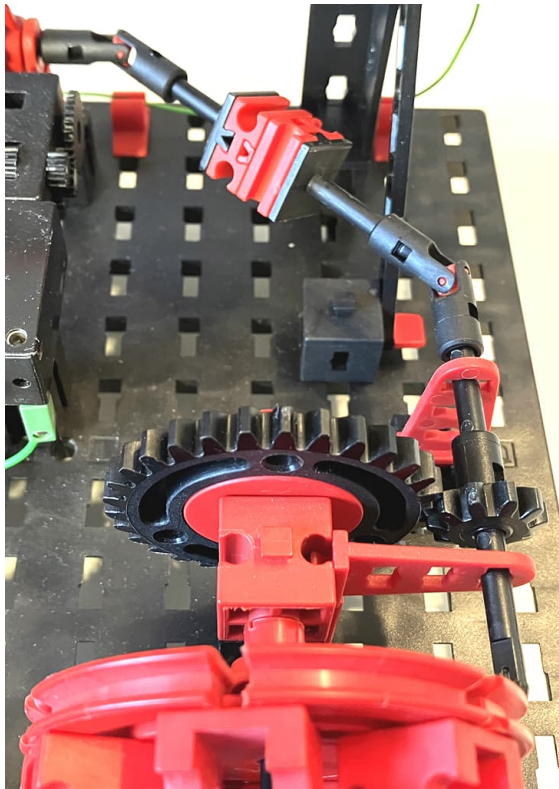


Abb. 12: Die rechte Doppelkardan-Strecke zur Drehscheibe mit den sechs „Werkstücken“

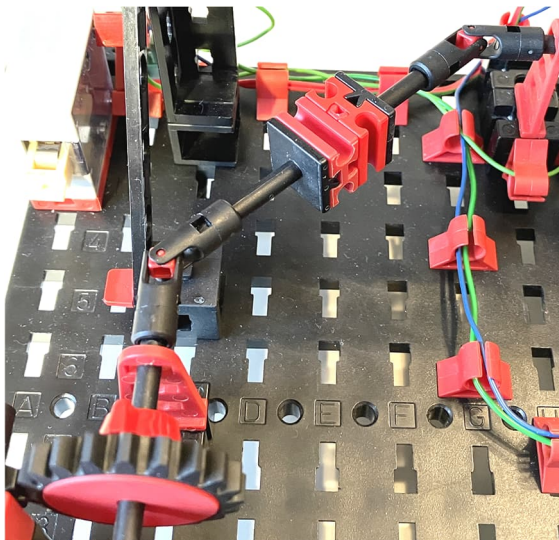


Abb. 13: Die linke Doppelkardan-Strecke vor zum Exzenter

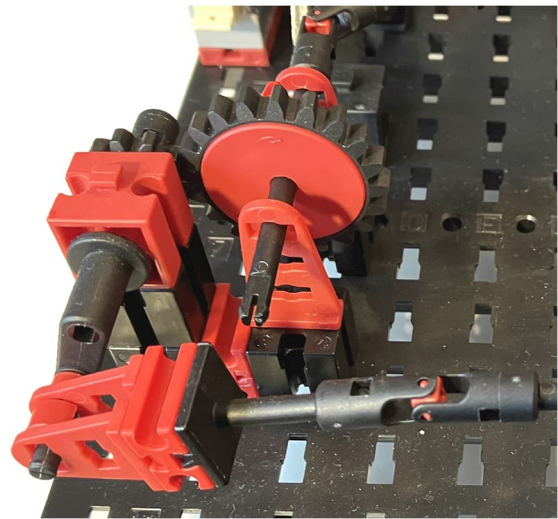


Abb. 14: Detailblick auf den Exzenter

Die Steuerung als Dreingabe

Damit man das Modell gut studieren und beobachten kann, habe ich ihm eine kleine Elektromechanik-Steuerung verpasst. Ich wollte, dass das Modell extrem langsam läuft (damit man die Wirkungsweise fein beobachten kann), immer schneller wird (damit man sehen kann, mit welcher Präzision das auch bei hoher Geschwindigkeit funktioniert), um wieder immer langsamer zu werden. Dieses Spiel sollte sich endlos wiederholen.

Nichts leichter als das! Bewerfen wir das Problem doch einfach mit ein paar Millionen Transistoren in Form eines Interfaces und programmieren ein bisschen...

Oder auch nicht: Ich wählte eine sehr klassische Lösung mit einem selbstgebauten Schiebe-Potentiometer nach der Grundidee aus dem fischertechnik-Clubheft 1972-3 (Abb. 15, [11]): Die ältere fischertechnik-Antriebsfeder ([31028](#)) wird in die Nut einer Reihe von Grundbausteinen eingelegt. An beiden Enden wird per Kontaktstift „+“ und „-“ der Versorgungsspannung angelegt (im Modell verwende ich drei Stück Federkontakt 18, [31306](#)). Ein per Motor und Schnecke verschieblicher dritter Kontakt kann den damit entstandenen Spannungsteiler an beliebiger Stelle abgreifen.

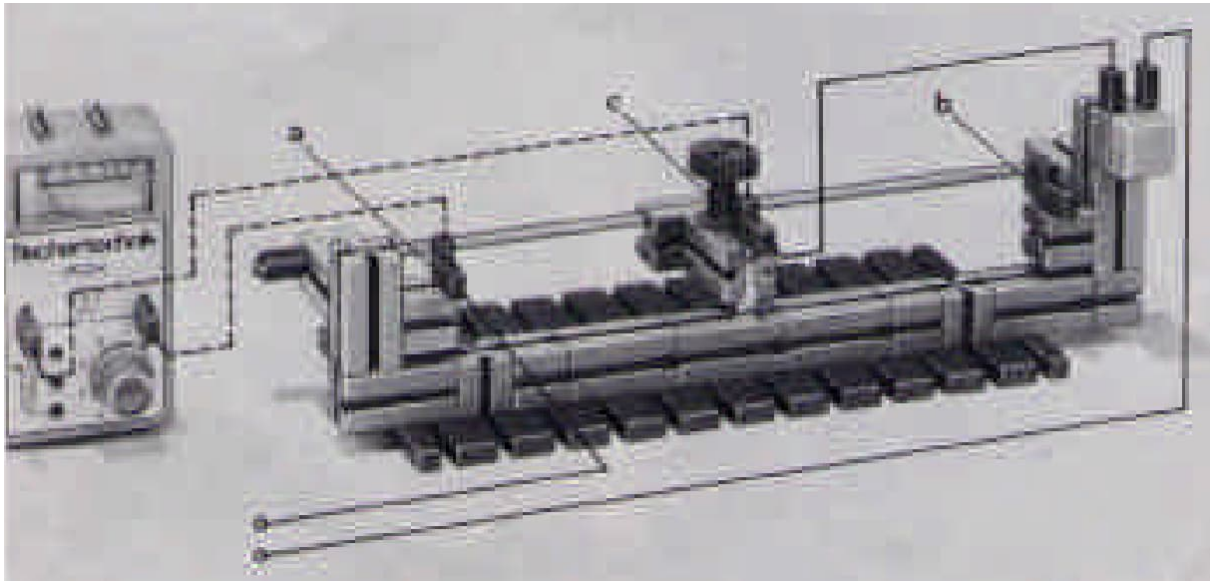


Abb. 15: Selbstbau-Schiebepotentiometer mit der fischertechnik-Antriebsfeder als Widerstandsdraht aus dem Clubheft 1972-3 [11]

Die so gebildete variable Spannung geht zum Antriebsmotor des eigentlichen Modells. Der Motor, der die Schnecke antreibt, wird immer an den Endlagen umgepolt, weil ein mitgeführter Polwendeschalter (31331) an den Endlagen mit seinem Kipphebel anstößt und so umgeschaltet wird. Damit der Schneckenantrieb noch langsamer läuft, habe ich ein fischertechnik-Lämpchen in Serie mit dem Schneckenmotor geschaltet.

Das Schiebepotentiometer wird also immer „voll aufgedreht“, dann wieder bis fast zum Stillstand des Modells heruntergeregelt, und das immer wiederholend. Abb. 16 bis 19 zeigen den Aufbau, falls das jemand nachbauen möchte.

Fazit

Periodisch ungleichförmige Getriebe kann man auch dadurch herstellen, dass man den normalerweise unerwünschten Kardanfehler ausnutzt oder sogar durch absichtlich „falsche“ Bauweise noch verstärkt. Damit lassen sich mechanisch Bewegungsabläufe realisieren, die sonst einer mehr oder weniger komplexen Steuerung bedürften. Die hier vorgestellte Lösung ist nur eine weitere in einer hoffentlich noch langen

Reihe von Ideen, wie man zyklisch ungleichförmige Übersetzungen realisieren kann.

Quellen

- [1] Wilhelm Klopmeier im [Bilderpool der ft-Community](#).
- [2] Dirk Fox: *Wie man mit fischertechnik zum Patent kommt (2)*. Auf [fischertechnik-blog.de](#), 2015.
- [3] Werner Dohmen: *Getriebe mit zyklisch variablem Übersetzungsverhältnis*. Würdigung der Arbeit von Wilhelm Klopmeier, auf [ftcommunity.de](#), 2007.
- [4] Esther Mietzsch: *Ungleichförmig übersetzende Zahnräder*. Foto von der ft-Convention Sinsheim 2024, im [Bilderpool der ft-Community](#).
- [5] Stefan Falk: *Der Wohnzimmer-Dienstreisen-Urlaubs-Notfallkasten*. [ft:pedia 1/2016](#), S. 31–36.
- [6] Stefan Falk: *Zyklisch ungleichförmig übersetzende Getriebe*. [ft:pedia 4/2022](#), S. 12–19.
- [7] Wikipedia: [Kreuzgelenk](#).

- [8] Stefanie Busch: *Kardanische Gelenkwelle*. [ft:pedia 2/2015](#), S. 8–14.
- [9] fischertechnik: *Winkelbewegliche Kupplungen*. In *hobby 2 Band 2* (auf [ft-datenbank.de](#)), S. 16–19, 1975.
- [10] Stefan Falk: *Zyklisch ungleichförmige Übersetzung mittels Kardanfehler*. Auf [YouTube](#), 2025.
- [11] fischertechnik: *Aktuelles zum Nachbauen*. Selbstgebaute Schiebe- und Dreh-Potentiometer im Clubheft 1972-3. Auf [ft-datenbank.de](#), S. 20-23.

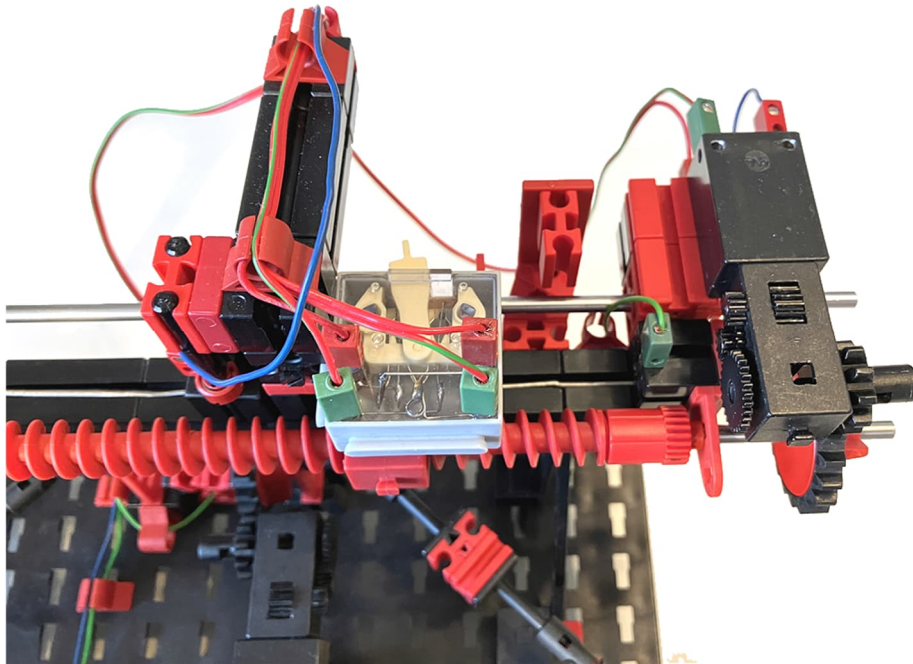


Abb. 16: Antrieb der Potentiometer-Schnecke mit Polwendeschalter zum Umpolen an den Endlagen

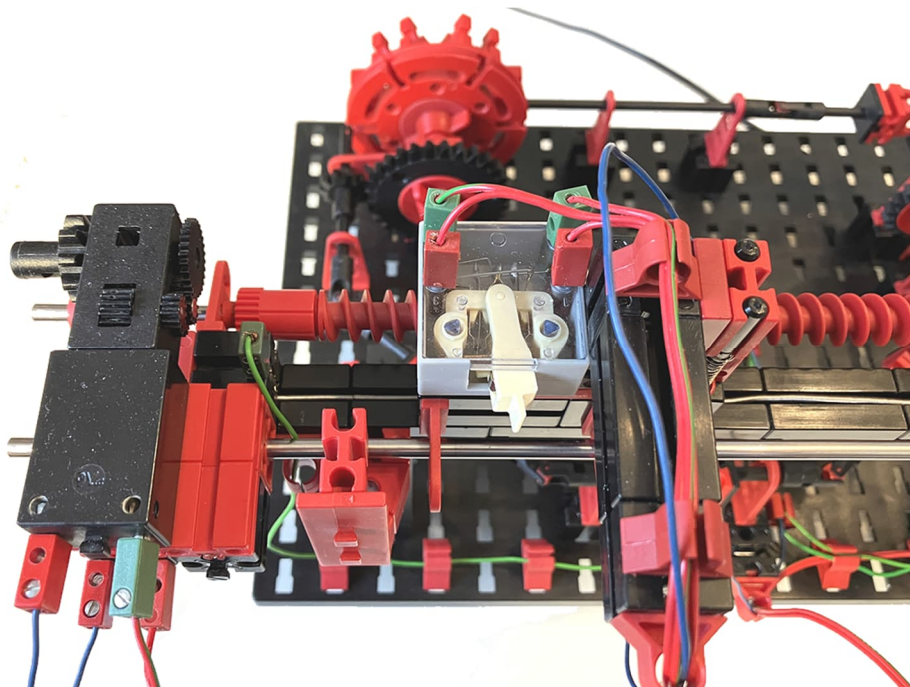


Abb. 17: Antrieb des Potentiometers von der Rückseite

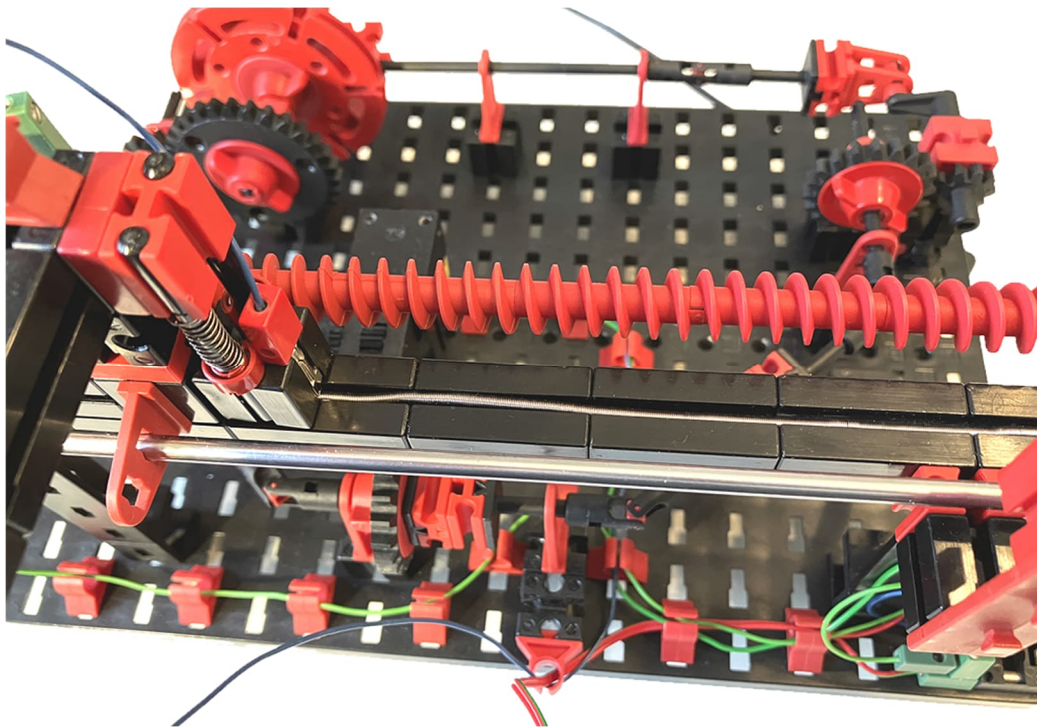


Abb. 18: Mittlerer, verschieblicher Spannungsabgriff. Die Druckfeder drückt den Baustein 15 mit der Kontaktfeder nach unten auf die in die Bausteinreihe eingelegte fischertechnik-Antriebsfeder, die links und rechts mit den Polen der Versorgungsspannung verbunden ist und von den dortigen Kontakten fixiert wird.

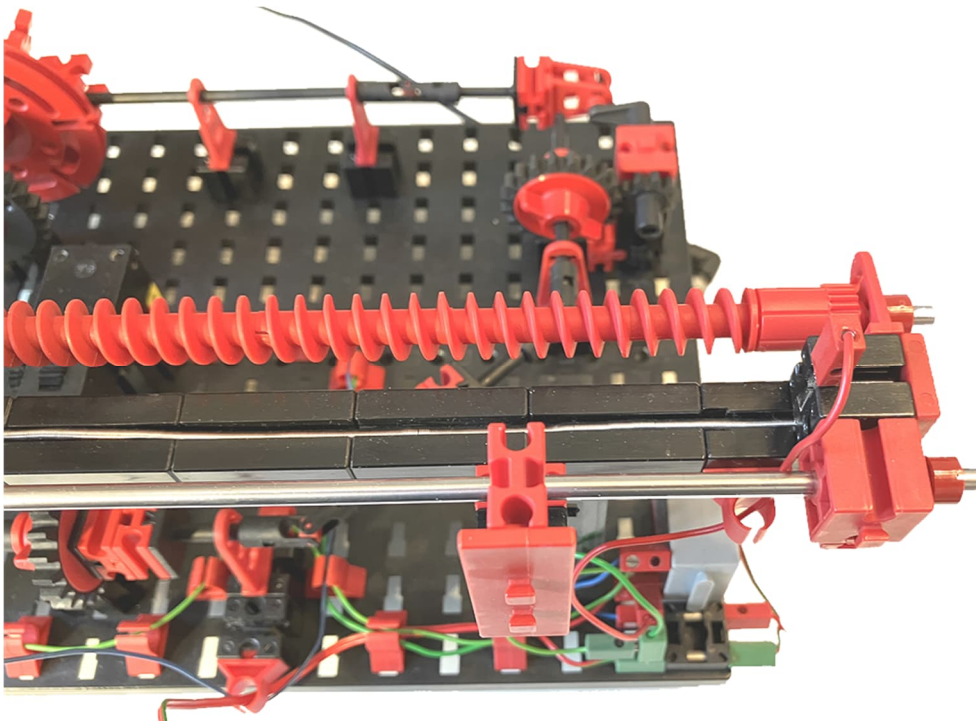


Abb. 19: Das zweite Ende des Schiebe-Potentiometers mit dem dortigen Kontakt und dem Baustein 7,5, an dem der Polwendeschalter anstößt und sich so selbst umschaltet.

Getriebe

Zahnradgetriebe für beliebige Übersetzungsverhältnisse

Dirk Fox

fischertechnik bietet nur eine kleine Auswahl an Zahnrädern. Damit sind die Möglichkeiten der Getriebekonstruktion jedoch keineswegs beschränkt: Tatsächlich kann man mit fischertechnik-Zahnrädern jedes rationale, also als Bruch darstellbare Übersetzungsverhältnis konstruieren. Das kann man sogar beweisen. Ein „magisches“ Getriebe macht das möglich: das Differential.

Hintergrund

Mit Z10, Z15, Z20, Z30 und Z40 sowie dem Zahnkranz Z85, dem Kegelzahnrad Z12 und dem Kronrad Z32 bietet fischertechnik im Modul $m = 1,5$ nur eine sehr beschränkte Vielfalt an Zahnrädern. Daher könnte man meinen, dass damit auch nur eine sehr begrenzte Zahl an Übersetzungen möglich ist – nämlich alle Vielfachen der damit erreichbaren Übersetzungen. Doch weit gefehlt: Tatsächlich lässt sich unter Hinzunahme des Differentials mit diesen wenigen Zahnrädern nicht nur jedes ganzzahlige, sondern auch jedes rationale Übersetzungsverhältnis konstruieren.

Das gilt nicht nur theoretisch, sondern auch praktisch: Hat man einmal die benötigten Teilübersetzungen bestimmt, so kann man das zur gesuchten Übersetzung passende Getriebe auch aus fischertechnik-Zahnrädern aufbauen. Bei einigen Getrieben benötigt man gegebenenfalls zusätzliche Zahnräder, die das Übersetzungsverhältnis nicht verändern, aber den Abstand zwischen zwei Getriebeachsen überbrücken. Manchmal kann man das resultierende Getriebe auch vereinfachen, indem man Ketten- und Riemenantriebe dafür verwendet.

In diesem Beitrag werde ich zunächst beweisen, dass jedes (rationale) Übersetzungsverhältnis mit den genannten Zahn-

rädern und einem Differential konstruiert werden kann. Das ist vielleicht mit „Kanonen auf Spatzen“ geschossen – für einen Mathematiker ist die Aussage vermutlich offensichtlich. Aber der Beweis ist konstruktiv und liefert damit zugleich eine Methode, wie man das zu einer gegebenen Übersetzung passende Getriebe findet. Schließlich demonstriere ich an einem ausgewählten Beispiel die Konstruktion eines Getriebes zu einer primen ganzzahligen Übersetzung.

Vorüberlegungen

Schon das Z15-Zahnrad, das eine 1:1,5-Übersetzung (Z10:Z15) ermöglicht, macht deutlich, dass es zumindest einige nicht ganzzahlige rationale (also als Bruch oder Dezimalzahl darstellbare) Übersetzungsverhältnisse gibt, die wir mit fischertechnik konstruieren können.

Tatsächlich erhalten wir zu allen ganzzahligen Übersetzungsverhältnissen, für die wir mit fischertechnik ein Getriebe konstruieren können, auch unmittelbar ein Getriebe zum Kehrwert des Übersetzungsverhältnisses, wenn wir An- und Abtrieb des Getriebes vertauschen. Denn es gilt (für alle natürlichen Zahlen $a \in \mathbb{N}$):

$$y = a \cdot x \rightarrow x = \frac{1}{a} \cdot y$$

Das funktioniert auch in der Praxis, solange wir in unserer Konstruktion kein Schneckengetriebe verwenden, denn das sperrt in eine Richtung und kann daher nicht umgekehrt werden.

Schließlich können wir mehrere Zahnradgetriebe koppeln: Zwei fischertechnik-Zahnradgetriebe mit den Übersetzungsverhältnissen $(a, b \in \mathbb{N})$

$$\begin{aligned} y &= a \cdot x \\ y &= b \cdot x \end{aligned}$$

können wir so miteinander verbinden, dass wir ein Zahnradgetriebe mit dem Übersetzungsverhältnis

$$y = \frac{a}{b} \cdot x$$

erhalten.

Außerdem können wir jedes Getriebe mit einer Übersetzung $y = a \cdot x$ über eine weitere Achse und zwei gleiche Zahnräder (z. B. Z10:Z10) mit einer Richtungsumkehr versehen. So erhalten wir das Übersetzungsverhältnis

$$y = -a \cdot x$$

Damit gelten unsere obigen Aussagen für alle ganzen Zahlen ohne die Null $(a, b \in \mathbb{Z} \setminus \{0\})$. Wenn wir nun beweisen wollen, dass wir zu *jedem* rationalen Übersetzungsverhältnis

$$y = \frac{a}{b} \cdot x$$

(für alle $a, b \in \mathbb{Z} \setminus \{0\}$) ein Getriebe aus fischertechnik konstruieren können, genügt es zu beweisen, dass uns das für *jedes ganzzahlige* Übersetzungsverhältnis gelingt. Das werden wir jetzt tun – unter ausschließlicher Verwendung der fischertechnik-Zahnräder und des Differentials.

Beweismethode

Aus dem Mathematikunterricht kennt ihr vielleicht noch die eine oder andere mathematische Beweismethode, wie den Beweis durch Widerspruch, durch Fallunterschei-

dung, durch Gegenbeispiel oder den indirekten Beweis, bei dem man die Gültigkeit der negierten Aussage beweist.

Zu den m. E. schönsten Beweisen gehören Beweise durch vollständige Induktion. Die erste Beschreibung dieses Beweisverfahrens wird dem Mathematiker [Blaise Pascal](#) (1623-1662) zugeschrieben, der es 1654 in *Traité du triangle arithmétique* verwendete. Damit lassen sich Aussagen über die natürlichen Zahlen $\mathbb{N} = \{1, 2, 3, \dots\}$ oder über Elemente einer Menge beweisen, die mit den natürlichen Zahlen aufgezählt werden können.

Die Beweismethode ist schnell erklärt: Eine Aussage $A(n)$ ist $\forall n \in \mathbb{N}$ (sprich: für alle n aus der Menge der natürlichen Zahlen) wahr, wenn

- $A(1)$ gilt (*Induktionsanfang* oder *Verankerung*) und
- $\forall n \in \mathbb{N}$ gilt: Aus der Gültigkeit von $A(n)$ kann die Gültigkeit von $A(n+1)$ abgeleitet werden (*Induktionsschritt*)

Jetzt müssen wir zunächst unsere zu beweisende Aussage geeignet als $A(n)$ formulieren.

Die Behauptung

Um unsere Behauptung, dass wir alle ganzzahligen Übersetzungsverhältnisse mit fischertechnik-Zahnrädern (inklusive Differential) konstruieren können, mit vollständiger Induktion beweisen zu können, müssen wir sie mathematisch für eine Menge von Elementen ausdrücken, die sich über die natürlichen Zahlen aufzählen lassen.

Dazu zerlegen wir zunächst den ganzzahligen Faktor a der Gleichung

$$y = a \cdot x$$

in seine Primfaktoren:

$$a = p_1^{e_1} \cdot p_2^{e_2} \cdot p_3^{e_3} \cdot \dots \cdot p_{n-1}^{e_{n-1}} \cdot p_n^{e_n}$$

Dabei bezeichnen p_1 bis p_n die ersten n Primzahlen. Die (ganzzahligen) Exponenten e_i können auch 0 sein. Nun können wir unsere Aussage $A(n) \forall n \in \mathbb{N}$ wie folgt formulieren:

Sei

$$a = p_1^{e_1} \cdot p_2^{e_2} \cdot p_3^{e_3} \cdot \dots \cdot p_{n-1}^{e_{n-1}} \cdot p_n^{e_n}$$

mit $e_i \in \mathbb{N}_0$, $n \in \mathbb{N}$ und p_i ($i \leq n$) die i -te Primzahl. Dann können wir unter ausschließlicher Verwendung der fischertechnik-Zahnräder inklusive Differential ein Getriebe konstruieren, das eine Übersetzung

$$y = a \cdot x$$

bewirkt.

Die Übersetzung $y = a \cdot x$ erhalten wir, indem wir für jeden Primfaktor p_i von a die entsprechende Übersetzung konstruieren und die resultierenden Übersetzungen (in beliebiger Reihenfolge) koppeln.

Der Beweis

Die Aussage $A(1)$ gilt ganz offensichtlich, denn $a = p_1^{e_1} = 1^{e_1} = 1$ ($\forall e_i \in \mathbb{N}_0$), also $y = x$. Damit ist $A(1)$ wahr.

Für unseren Beweis müssen wir noch den Sonderfall $n = 2$ betrachten, denn $p_2 = 2$ ist die einzige gerade Primzahl. Das Übersetzungsverhältnis $a = 2^e$ ($\forall e \in \mathbb{N}_0$) lässt sich leicht mit einer oder mehreren (falls $e > 1$) Übersetzungen Z10:Z20 (oder Z20:Z40 bzw. Z15:Z30) konstruieren. Ist $e = 0$ entspricht die Aussage $A(1)$. Damit ist auch $A(2)$ wahr.

Nun nehmen wir an, $A(n)$ sei wahr. Dann können wir jede Übersetzung

$$y = p_1^{e_1} \cdot p_2^{e_2} \cdot p_3^{e_3} \cdot \dots \cdot p_{n-1}^{e_{n-1}} \cdot p_n^{e_n} \cdot x$$

konstruieren. Wir müssen jetzt zeigen, dass wir aus $A(n)$ die Gültigkeit von $A(n+1)$ ableiten können. Dabei sei $n \geq 2$, denn die Gültigkeit von $A(1)$ und $A(2)$ haben wir bereits nachgewiesen.

Das ist trivial, falls $e_{n+1} = 0$, denn dann entspricht $A(n+1)$ der Aussage $A(n)$. Wir müssen also nur zeigen, dass $A(n+1)$ für $e_{n+1} = 1$ gilt – wir also unter der Voraussetzung, dass $A(n)$ gilt, auch

$$y = p_{n+1} \cdot x$$

konstruieren können. Denn damit gilt die Aussage $A(n+1)$ auch für alle $e_{n+1} > 1$.

Diese Konstruktion gelingt uns mit einem fischertechnik-Differential, dem „historischen“ mit Z15 als Differentialkäfig (31043) oder dem Rast-Differential mit Z20 (35983), siehe Abb. 1.

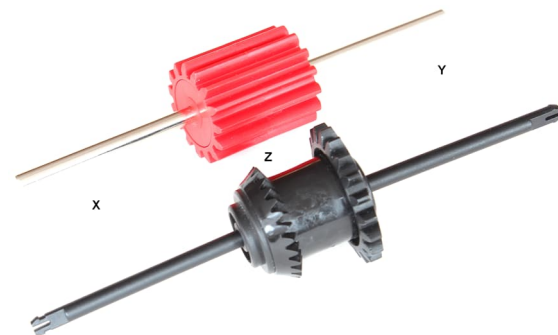


Abb. 1: fischertechnik-Differentiale, das „historische“ und das Rast-Differential mit Z20, mit Kennzeichnung der An- und Abtriebe

Nach der Grundgleichung des Differentials (im Folgenden kurz „Grundgleichung“), die Thomas Püttmann in [1, 2] erläutert und hergeleitet hat, gilt:

$$2z = y + x$$

Wenn wir die gewünschte Übersetzung

$$y = p_{n+1} \cdot x$$

in die Grundgleichung einsetzen, folgt

$$z = \frac{p_{n+1} + 1}{2} \cdot x$$

Da $p_{n+1} \geq 3$ eine (ungerade) Primzahl ist, ist $p_{n+1} + 1$ durch zwei teilbar. Das Ergebnis ist entweder eine Primzahl $< p_{n+1}$ oder lässt sich wiederum in Primzahlen $< p_{n+1}$ zerlegen. Für beide Fälle gilt die Aussage $A(n)$. Wenn wir ein solches Getriebe zwischen Antriebsachse und Differentialkäfig einbauen, erhalten wir unsere

gewünschte Übersetzung zwischen An- und Abtriebsachse. Damit gilt auch $A(n+1)$, was zu beweisen war.

Tatsächlich haben wir im Beweis neben dem Differential nur ein einziges Übersetzungsverhältnis verwendet: die 1:2-Übersetzung für die Konstruktion in $A(2)$. Neben den Differentialen benötigen wir (theoretisch) also nur Z_{10} und Z_{20} – allerdings wird das Getriebe in der Praxis viel kompakter, wenn man auf weitere Zahnräder zurückgreifen kann.

Die Praxis

Der theoretische Beweis, dass wir jedes Getriebe mit einem rationalen Übersetzungsverhältnis aus den fischertechnik-Zahnrädern (Z_{10} , Z_{20} , Z_{30} , Z_{40}) sowie den Differentialen konstruieren können, ist das eine – das andere ist die praktische Konstruktion des Getriebes selbst. Doch unser Beweis ist konstruktiv und gibt uns eine Methode an die Hand, ein gesuchtes Getriebe aus fischertechnik-Zahnrädern und dem Differential zu bilden.

An einem praktischen Beispiel will ich nun noch zeigen, wie das gelingt: Wir konstruieren ein Getriebe für die Mondphasenanzeige einer Uhr. Eine Lösung durch 3D-Drucken oder Fräsen eines passenden Zahnrads scheidet hier nämlich aus – der schiefen Größe wegen.

Mondphasenanzeige

Die Position des Mondes schneidet (aus der Perspektive eines Beobachters auf der Erde) im Schnitt alle 29 Tage, 12 Stunden, 44 Minuten und 2,882 Sekunden denselben Längengrad, also alle rund 29,530589 Tage. Diese Phase wird *Lunation* genannt, ihre Dauer *synodischer Monat* [3].

In Uhren mit Mondphasenanzeige wird diese meist mit einem 1:59-Getriebe angenähert, das eine Drehscheibe mit zwei Monden antreibt [3, 4] (Abb. 2) – also mit 29,5 Tagen. Damit geht die Anzeige mit jeder Lunation etwa 44 Minuten und 2,882

Sekunden (oder 0,0305883 Tage) vor. Nach knapp 33 synodischen Monaten, also rund 965 Tagen oder 2,64 Jahren, summiert sich der Fehler bereits auf einen ganzen Tag – und die Anzeige muss korrigiert werden.



Abb. 2: Mondphasenanzeige mit 1:59-Getriebe

Um den Faktor 50 genauer wäre die Anzeige, wenn sie den synodischen Monat mit 29,53 Tagen nähern würde. Dann ginge sie je Lunation nur knapp 51 Sekunden (oder 0,00059 Tage) vor; eine Korrektur um einen Tag wäre erst nach rund 137 Jahren erforderlich. Dafür benötigen wir eine Übersetzung von 1:29,53 – oder 100:2953.

Eine Übersetzung 1:100 habe ich in [5] vorgestellt: Sie lässt sich zusammensetzen aus einem Getriebe 1:25 und einer 1:4-Übersetzung ($Z_{10}:Z_{40}$).

Die Übersetzung 1:2953 ist etwas kniffliger, denn die Zahl 2953 ist eine Primzahl – die 425ste, um genau zu sein. Ein Zahnrad mit dieser Zähnezahl hätte im Modul $m = 1,5$ einen Durchmesser von 3,83 m. Es bleibt uns also nur, ein passendes Getriebe zu konstruieren.

Berechnung

Mit der im Beweis verwendeten Konstruktionsmethode können wir die gewünschte Übersetzung

$$y = 2953 \cdot x$$

durch die Verwendung eines Differentials weiter „zerlegen“. Dazu setzen wir sie in die Grundgleichung ein:

$$2z = x + y \Rightarrow z = 1477 \cdot x$$

Damit benötigen wir ein Übersetzungsverhältnis von 1:1477 zwischen Differentialkäfig z und Antriebsachse x . Wenn wir diesen Faktor in seine Primfaktoren zerlegen, erhalten wir:

$$1477 = 7 \cdot 211$$

Für die Konstruktion der Übersetzungsverhältnisse zu den Primzahlen 211 und 7 benötigen wir mindestens zwei weitere Differentialgetriebe.

Jetzt können wir einen kleinen Trick anwenden: Wir schieben in unserem Getriebe zwei Richtungsumkehrungen ein, die sich gegenseitig aufheben:

$$y = -(-2953 \cdot x)$$

Wenn wir nun das Übersetzungsverhältnis

$$y' = -y = -2953 \cdot x$$

in die Grundgleichung einsetzen, erhalten wir:

$$2z' = x + y' \Rightarrow z' = -1476 \cdot x$$

Zerlegen wir dieses Übersetzungsverhältnis zwischen Differentialkäfig und Antriebsachse nun in seine Primfaktoren, erkennen wir, dass die Konstruktion auf diesem Weg deutlich einfacher gelingt:

$$1476 = 2^2 \cdot 3^2 \cdot 41$$

Die ersten vier Primfaktoren konstruieren wir mit drei Stirnradgetrieben (Z10:Z40, Z10:Z30, Z10:Z30). Wenn wir den Richtungswechsel-Trick noch einmal auf den fünften Primfaktor 41 anwenden, erhalten wir die Übersetzung $y'' = -41 \cdot x''$ durch zwei einfache gekoppelte Differentialgetriebe:

(a) Für die Übersetzung zwischen Differentialkäfig und Antriebsachse – Herleitung wie oben – erhalten wir

$$2z'' = y'' + x'' \Rightarrow z'' = -20 \cdot x''$$

Diese Übersetzung können wir leicht durch zwei Teilgetriebe 1:4 und 1:5 ersetzen, über

die wir Differentialkäfig und Antriebsachse des Differentials miteinander verbinden.

(b) Die Übersetzung 1:5 ergänzen wir erneut um einen Richtungswechsel zu $y''' = -5 \cdot x'''$ und konstruieren sie mit einem Differential mit 1:2-Übersetzung zwischen Differentialkäfig und Antriebsachse:

$$2z''' = y''' + x''' \Rightarrow z''' = -2 \cdot x'''$$

Bei der Konstruktion des Gesamtgetriebes müssen wir an den An- oder Abtrieben, an denen wir den Richtungswechsel-Trick angewendet haben, eine Richtungsumkehr ergänzen.

Konstruktion

Das Getriebe setzt sich also aus den folgenden drei Teilgetrieben zusammen:

- *Getriebe A* mit dem Übersetzungsverhältnis 1:100, das wir aus den Übersetzungen 1:4 und 1:25 bilden, siehe die Konstruktion in [5].

Mit unserem „historischen“ Differential mit Z15 ([31043](#)) – das übrigens schon 1968 Einzug ins fischertechnik-System hielt –, zwei Z30, drei Z10 und einem Z40 ist das Getriebe schnell konstruiert (Abb. 3). Der Abtrieb ist die y -Achse des Differentials (im Bild rechts hinten).

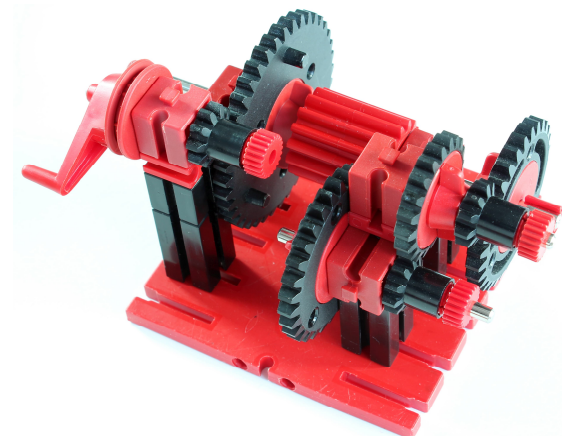


Abb. 3: Getriebe 1:100 mit „historischem“ Differential (Getriebe A)

Ähnlich leicht gelingt das mit einem Rast-Differential mit Z20 (Abb. 4).

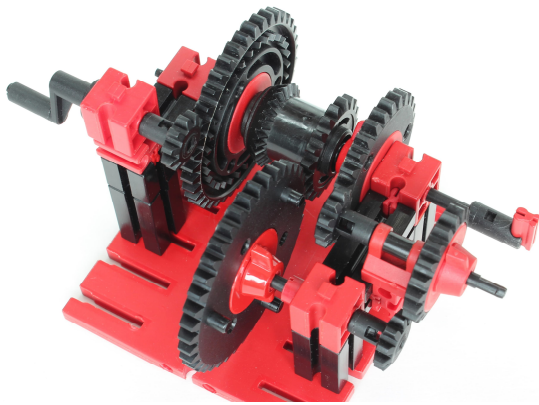


Abb. 4: Getriebe 1:100 mit Rast-Differential (Getriebe A)

- *Getriebe B* mit dem Übersetzungsverhältnis 1:41, das wir – wie im vorausgegangenen Abschnitt beschrieben – mit einem Differential und einer Übersetzung 1:20 (mit Richtungsumkehr) zwischen Differentialkäfig und Antriebsachse konstruieren.

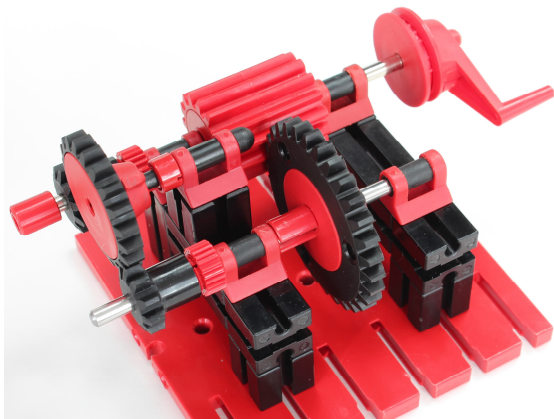


Abb. 5: Getriebe 1:5 mit „historischem“ fischertechnik-Differential

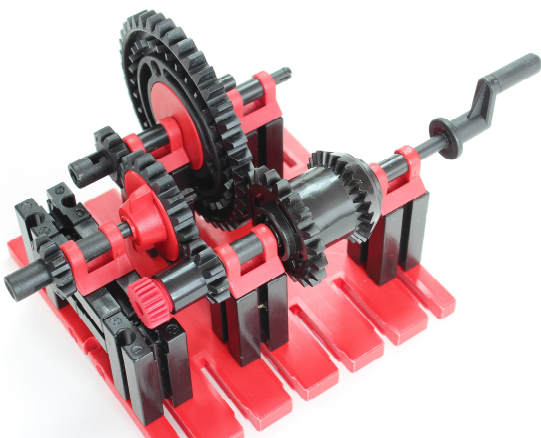


Abb. 6: Getriebe 1:5 mit Rast-Differential

Für das Getriebe B benötigen wir zunächst eine Übersetzung 1:5. Abb. 5 und 6 zeigen eine Beispielkonstruktion mit dem historischen und dem Rast-Differential.

Wir müssen das Getriebe nun noch um eine Richtungsumkehr und eine Übersetzung 1:4 erweitern. Dazu müssen wir in der Konstruktion lediglich das Z20 durch ein Z40 ersetzen und die Achse des Z40 als Abtrieb verwenden.

Das Getriebe B erhalten wir nun, indem wir den Antrieb des Getriebes 1:5 mit einem weiteren Differentialkäfig und den Antrieb mit der y-Achse des Differentials verbinden (Abb. 7, 8).

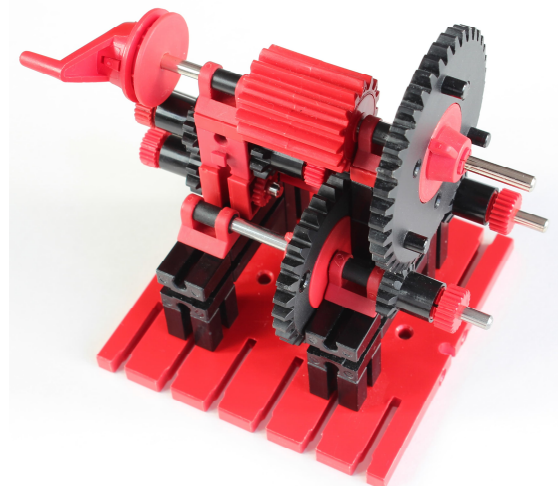


Abb. 7: Getriebe 1:41 mit „historischem“ Differential (Getriebe B)

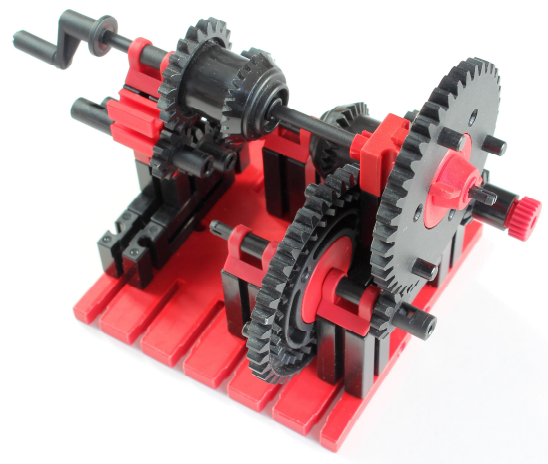


Abb. 8: Getriebe 1:41 mit Rast-Differential (Getriebe B)

- *Getriebe C* mit Differential, bei dem wir Differentialkäfig und Abtriebsachse über das Getriebe B und drei Übersetzungen (1:4, 1:3 und 1:3) mit Richtungsumkehr verbinden.

Das Übersetzungsverhältnis 1:36 ($y = 2^2 \cdot 3^2 \cdot x$) lässt sich direkt aus fischertechnik-Zahnradern als Stirnradgetriebe zusammensetzen ($3 \times Z10$, $2 \times Z30$ und $1 \times Z40$). Dabei können die Übersetzungen wahlweise vor der Antriebs- oder hinter der Abtriebsachse des Getriebes B ergänzt werden. Abb. 9 zeigt das resultierende Getriebe C mit einer Mischung aus historischen und Rast-Differentialen.

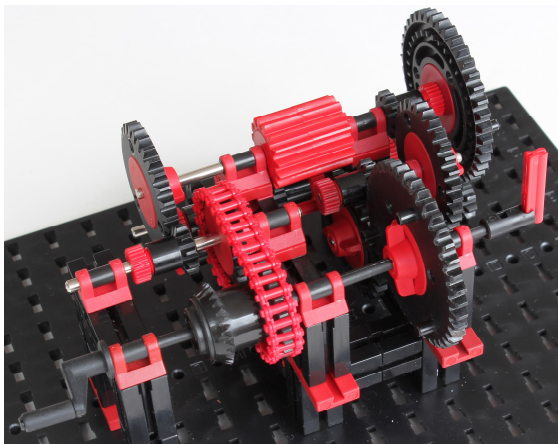


Abb. 9: Mondphasen-Getriebe, Übersetzung 1:2953 (Getriebe C)

Eine Herausforderung dieser Konstruktionsmethode habe ich verschwiegen: Jedes Zahnradgetriebe hat Verluste durch Reibung – sowohl in den Achslagern als auch an den ineinandergreifenden Zähnen. Im Hobby-Band 2-1 wird der Verlust je Zahnradpaar auf 10% geschätzt ([6], S. 41). Das ist erheblich – und summiert sich bei unserem Getriebe natürlich. Dadurch verringert sich das Drehmoment, das durch die Übersetzung ins Langsame dennoch steigt.

Kritisch ist daher die abschließende Übersetzung ins Schnelle: Wenn wir unser Getriebe 1:100 an das Ende unseres Mondphasengetriebes setzen, wird es mit dem Drehmoment eng. Bei einem vergleichsweise kleinen Gesamtübersetzungsverhält-

nis wie in unserem Mondphasengetriebe (näherungsweise 1:30) mit mehreren Nachkommastellen (hier: 2) kann es leicht passieren, dass das Drehmoment für die Übersetzung ins Schnelle nicht ausreicht.

Besser funktionieren daher Annäherungen an das gewünschte Übersetzungsverhältnis 1:29,53. Aber das Beispiel illustriert schön, wie sich ein primes ganzzahliges Übersetzungsverhältnis mit Differentialen konstruieren lässt.

Zusammenfassung und Fazit

Mit den fischertechnik-Differentialen und einer Anzahl Z10, Z15, Z20, Z30 und Z40 können wir also ein Zahnradgetriebe mit jedem beliebigen rationalen Übersetzungsverhältnis konstruieren. Das haben wir sogar bewiesen: Auch wenn die Tatsache vielleicht mathematisch offensichtlich ist, können wir mit der Konstruktionsmethode, die ich im Beweis verwendet habe, beliebige ganzzahlige Übersetzungen bauen. Beispielhaft habe ich das für ein Mondphasengetriebe mit dem (primen) Übersetzungsverhältnis 1:2953 gezeigt.

Bei kleinen Übersetzungsverhältnissen mit mehreren Nachkommastellen (hier 1:29,53) kann es bei einer abschließenden Übersetzung ins Schnelle allerdings mit dem Drehmoment eng werden, denn Zahnradgetriebe haben ganz erhebliche Reibungsverluste.

In der Praxis findet man solche Getriebe mit Differential selten. Die Reibung begrenzt die mögliche Komplexität eines solchen Getriebes; zudem sind Stirnradgetriebe flacher, mechanisch weniger anspruchsvoll und erlauben oft eine hinreichend gute Annäherung an ein gewünschtes Übersetzungsverhältnis. Auch das Spiel des Getriebes steigt mit der Anzahl der Zahnräder – auch wenn die Differentiale das konstruktionsbedingt begrenzen.

In hochwertigen Armbanduhren verwendet man daher für die Mondanzeige Stirnradgetriebe, die wenige Zahnräder benötigen und

das Übersetzungsverhältnis zwischen Tag und synodischem Monat einigermaßen gut annähern.

Mit dem fischertechnik-Differential sind allerdings sehr viele Übersetzungen konstruierbar, die angesichts der begrenzten Zahl an verfügbaren fischertechnik-Zahnradern auf den ersten Blick vielleicht unmöglich erscheinen. Einige schöne Beispiele für solche Getriebe findet ihr in Thomas Püttmanns Planetarium [4, 7].

Referenzen

- [1] Thomas Püttmann: *Zahnräder und Übersetzungen (Teil 2)*. [ft:pedia 3/2011](#), S. 25–28.
- [2] Thomas Püttmann: *Das Differentialgetriebe*. [ft:pedia 4/2014](#), S. 12–19.
- [3] Dirk Fox: *Kalenderwerk mit Mondphasenanzeige*. [ft:pedia 2/2022](#), S. 49–60.
- [4] Dirk Fox, Thomas Püttmann: *Technikgeschichte mit fischertechnik*. dpunkt Verlag, 2015 (Nachdruck 2016, 2021), Kapitel 5: Das Planetarium.
- [5] Dirk Fox: *Synchronuhr mit Stirnradgetriebe*. [ft:pedia 4/2021](#), S. 18–21.
- [6] Fischer Werke: *fischertechnikhobby – Experimente+Modelle*, [hobby 2 Band 1](#), 1972.
- [7] Thomas Püttmann: *Planetarium*. [ft:pedia 4/2011](#), S. 39–51.

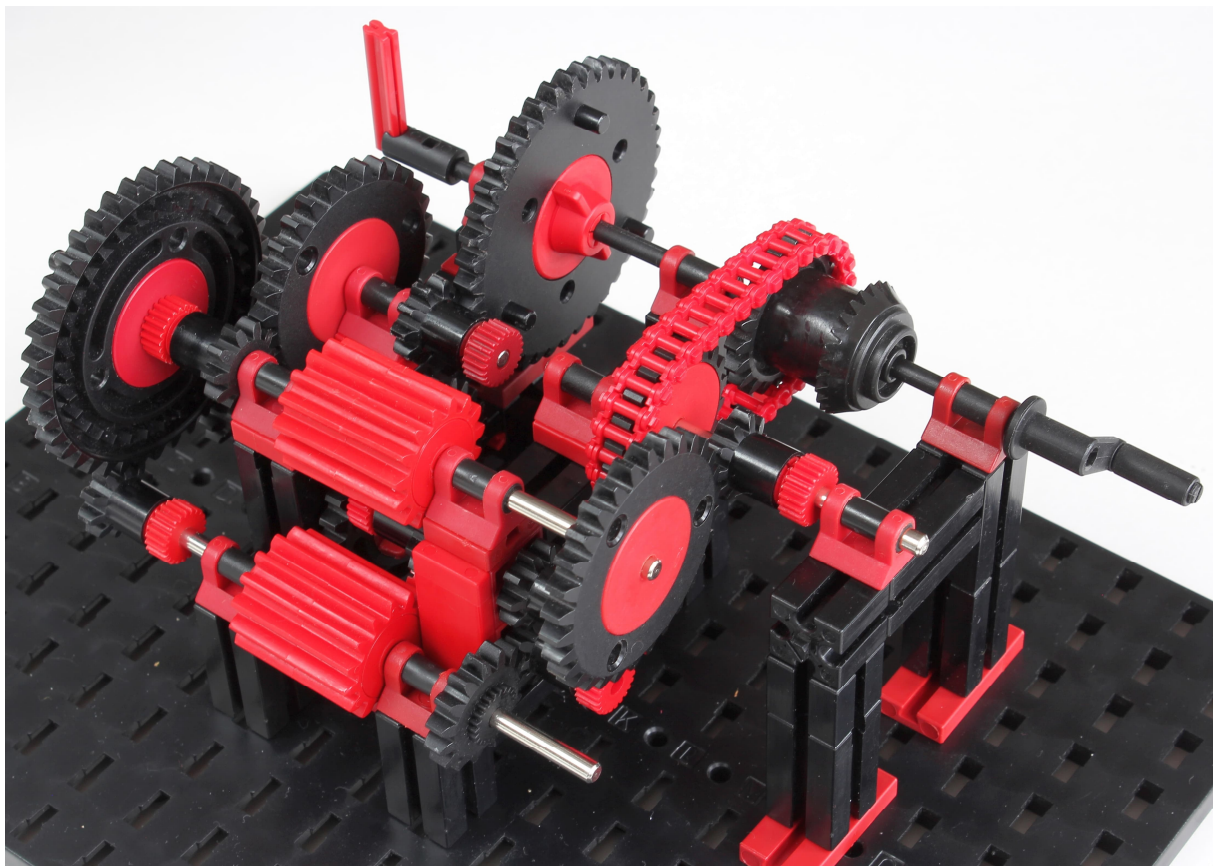


Abb. 10: Getriebe 1:2953 (Gesamtansicht)

Pneumatik

Servopneumatik mit fischertechnik

Florian Bauer

Dieser Beitrag zeigt, wie man fischertechnik-Pneumatik-Zylinder mit einem Microcontroller so ansteuert, dass gezielt auch Zwischenpositionen zwischen den End-Zuständen erreicht werden können. Neben Lösungen mit fischertechnik-Bauteilen werden auch einige Varianten mit Fremdmaterial zum Selbstbau und zur Weiterentwicklung vorgestellt.

Einleitung

fischertechnik hat seit längerem Pneumatik-Bauteile im Programm. Die legendären Pneumatik-Baukästen mit Festo-Bau-elementen sind vielleicht noch bei manchen in Erinnerung. Mit den äußerst kompakten Antriebszylindern und Ventilelementen kann man sehr interessante Modelle bauen, wie zum Beispiel den neuen Liebherr-Bagger L938 [1]. Die Zylinder dieses Modells können mit handbetriebenen Ventilen gesteuert oder mit fischertechnik-3/2-Wege-Magnetventilen ([35327](#)) angesteuert werden [17]. Auch komplexe pneumatische Funktionsmodelle wie Luftdruck-Motoren (z. B. [14, 7]) lassen sich realisieren.

Bei vielen Modellen werden die Zylinder entweder ganz aus- oder eingefahren. Um Zwischenstellungen zu erreichen, kann man den Luftstrom drosseln und mit einem On-Off-Ventil von Hand unterbrechen, wenn die Sollposition erreicht ist.

In diesem Beitrag will ich zeigen, wie es möglich ist, die Sollposition eines Stellzylinders elektronisch mit Hilfe von Servopneumatik kontinuierlich zu regeln.

Servopneumatik

Servopneumatik bezeichnet die Kombination aus pneumatischen Antrieben und elektronischer Regelungstechnik, mit der

sich Bewegungen präzise steuern lassen. Durch den Einsatz von Proportional- [10] oder Servoventilen [9] und Positionsmesssystemen können Stellung, Geschwindigkeit und Kraft eines Zylinders dynamisch und exakt geregelt werden – ähnlich wie bei elektrischen Servoantrieben, jedoch mit der Robustheit und Einfachheit der Pneumatik.

Was sind die Vorteile der Servopneumatik?

Energieeffizienz: Durch die bedarfsgerechte Regelung von Druck und Durchfluss wird weniger Energie verbraucht als bei konventioneller Pneumatik mit dauerhafter Druckversorgung.

Hohe Präzision: Positionen, Geschwindigkeiten und Kräfte lassen sich exakt steuern – ideal für Anwendungen mit variablen Bewegungsprofilen.

Flexibilität: Bewegungsabläufe können softwareseitig ohne mechanische Änderungen am System angepasst werden.

Kosteneffizienz bei bestimmten Anwendungen: Besonders bei mittleren Anforderungen an Dynamik und Präzision kann Servopneumatik günstiger sein als elektrische Antriebe.

Gute Integrationsfähigkeit: Da die Antriebsenergie über flexible Druckleitungen zugeführt wird, lassen sich servopneuma-

tische Steuerventile und Regler entfernt vom Wirkungsort platzieren.

Nachteile der Servopneumatik

Komplexere Steuerungstechnik: Neben den speziellen, oft recht teuren Proportional- oder Servoventilen werden Positionsensoren und entsprechende Regler benötigt. Die Regelung stellt wegen des dynamischen Verhaltens des kompressiblen Mediums Luft eine Herausforderung dar (siehe dazu [11]). Im Allgemeinen hat man mit Hysterese und Reibung zu kämpfen, was zum Teil nichtlineare Regelungssysteme erfordert.

Druckluftversorgung bleibt notwendig: Gegenüber voll elektrischen Systemen ist eine Druckluftversorgung notwendig.

Steuerung von doppelt-wirksamen Zylindern

Für eine Positions-Steuerung von pneumatischen Aktuatoren werden oft doppeltwirkende Pneumatik-Zylinder verwendet, die Bewegungen in beide Richtungen ausführen können – durch Druckluftzufuhr auf der Kolben- und der Stangenseite.

Mit einem Proportional- oder Servoventil werden der Luftstrom oder der Druck stufenlos geregelt, und es ermöglicht eine dynamische Anpassung der Bewegung.

Ein Wegmesssystem meldet die aktuelle Position. Die Steuerung vergleicht Ist- und Sollwert und regelt die Proportionalventile so, dass der Zylinder sich entsprechend bewegt. Optional kann der Druck in den Kammern zur zusätzlichen Regelung der Kraft erfasst werden.

Sobald die Sollposition erreicht ist, wird die Luftzufuhr gestoppt oder gehalten, je nach Anwendung.

Ventilarten

Für Servopneumatik werden Stetigventile [3] benötigt, die den Durchfluss oder Druck proportional zu einer Stellgröße einstellen

können. Dabei muss der Zusammenhang nicht unbedingt direkt proportional bzw. linear sein. Im Sprachgebrauch ist manchmal von Proportional- oder Servoventilen die Rede, die gerne auch synonym verwendet werden. Mit Servoventilen werden aber oft hoch genaue mehrstufige Steuerventile bezeichnet, die zusätzlich noch eine interne Closed-Loop-Regelung beinhalten.

Die in diesem Beitrag vorgestellten Ventilvarianten sind einfacher und eher in der Art eines Proportionalventils.

Im Folgenden soll das Prinzip eines 5/2-Wege-Proportionalventils auf Basis eines Kolbenschieber-Ventils erläutert werden.

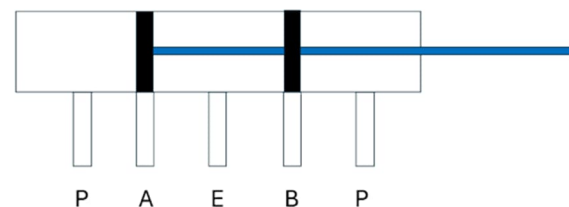


Abb. 1: 5/2-Proportionalventil geschlossen: Es fließt kein Luftstrom

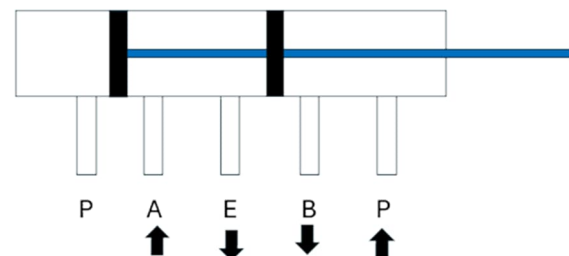


Abb. 2: 5/2-Wege-Proportionalventil offen: Ventilkammer an Anschluss A wird entlüftet, Ventilkammer B wird aufgepumpt

In Abb. 1 ist ein Kolben-Stangen-Proportionalventil mit fünf Anschlüssen und zwei Wirkanschlüssen gezeigt.

P bezeichnet den Anschluss für die Druckleitung (Pumpe), E steht für die Entlüftung, A und B sind jeweils die Anschlüsse für die Kammern des doppeltwirkenden Pneumatik-Zylinders. In Abb. 1 ist das Ventil geschlossen. In Abb. 2 ist die Ventilstellung so, dass Kammer A entlüftet und Kammer B mit dem Pumpendruck beaufschlagt wird. Wird der Kolben mit den Dichtungen nach rechts bewegt, kehrt sich die Wirkrichtung

um, weil dann Kammer A mit Pumpendruck P beaufschlagt und Kammer B entlüftet wird.



Abb. 3: Proportional-Wegeventil MPYE (© Festo SE & Co. KG) der Fa. Festo für die Verwendung mit einem externem Lageregler und Wegmesssystem zum Aufbau eines genauen pneumatischen Positioniersystems (mit freundlicher Genehmigung der Fa. Festo) [2]

Abb. 3 zeigt ein kommerzielles Proportional-Wegeventil der Firma Festo. Laut Herstellerangaben ist dieses Ventil für Drücke bis 10 bar ausgelegt und kann Durchflüsse von 100 bis 2.000 L/min regeln. Es eignet sich damit für Anwendungen mit variablen Durchflussanforderungen in pneumatischen Steuerungen in der Prozess- und Verpackungsindustrie.

Dieses gebräuchliche Prinzip eines Proportionalventils kommt auch bei Hydraulik-Steuerungen zum Einsatz. Anstelle der Entlüftungsöffnung E befindet sich eine als T bezeichnete Öffnung für die Rückführung der Hydraulik-Flüssigkeit zum Tank.

Herausforderungen bei Proportionalventilen

Beim Design eines Proportionalventils ist mit folgenden Herausforderungen umzugehen:

Dichtigkeit: Das Ventil sollte möglichst dicht sein. Dies wird durch glatte Oberflächen und flexible Dichtungen erreicht. Hier kommt es darauf an, ob die Dichtungen

in, entgegen oder senkrecht zur Druckrichtung wirken. In vielen Fällen verwendet man Dichtungen mit Dichtlippen, die durch den Druck an die Gehäusewand gedrückt werden.

Um das abdichtende Element zu verstellen, muss eine Kraftübertragung von außen in das Ventil erfolgen, was i. A. eine weitere Dichtung erfordert. Beispiel: Abdichtung der Durchführung einer Kolbenstange (z. B. mit einem Simmerring).

Reibung: Um eine ausreichende Dichtigkeit zu erreichen, wird das Dichtungsmaterial gequetscht bzw. an die Zylinderwand gedrückt. Dies führt zu einer erhöhten Reibung, die bei der Betätigung des Ventils überwunden werden muss.

Durchflusscharakteristik und Übergangszustände: Beim Proportionalventil wird jeweils eine Ventilseite geöffnet, während die andere geschlossen bleibt. Um einen Druckverlust zu vermeiden, sollte der Aufbau der Dichtung so gestaltet sein, dass immer nur eine Ventilseite P oder E mit dem Arbeitsanschluss verbunden ist, da sonst Luft entweichen kann. Bei einem Kolbenschieber-Proportionalventil kann dies dadurch erreicht werden, dass der Dichtkörper die Ventilöffnung überdeckt.

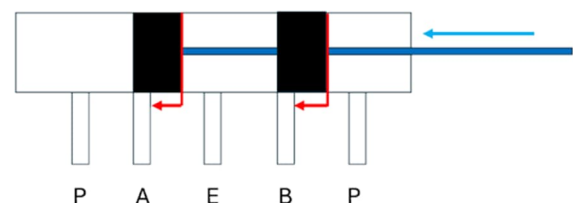


Abb. 4: Kolbenschieber-Proportionalventil mit Dichtungen breiter als Arbeitsöffnungen; Öffnung des Ventils (A-E und B-P) erst nach Überwindung der Tot-Zone (Verschiebung um Länge des roten Pfeils)

Ist der Dichtkörper dünner als die Öffnung und befindet er sich genau in der Mitte einer Öffnung, kommt es zu einem Übergangszustand, bei dem Druck von der Druckleitung direkt in die Abluftleitung gelangen kann. Solch ein Übergangszustand kann kurzzeitig in Kauf genommen werden,

wenn die mit Druck beaufschlagte Kammer im Ventil den Druck puffert, und er kann auch hilfreich sein, um die Regelbarkeit zu verbessern.

Ist der Dichtkörper breiter als die Öffnung, entsteht eine Tot-Zone, da der Ventilkolben so weit bewegt werden muss, bis er die Öffnungen wieder freigibt (Abb. 4).

In Abb. 5 sind Kurven der Durchflusscharakteristik für ein Ventil mit und ein Ventil ohne Tot-Zone dargestellt. Durch die Tot-Zone bekommt das Ventil eine Hysterese beim Umschalten.

Für eine gute Regelbarkeit sollte die Tot-Zone klein sein und es ist erforderlich, dass die Kurven der Durchflusscharakteristik (Abb. 5) nicht zu steil ansteigen. Dies erreicht man durch Dichtungsprofile, mit denen der Durchfluss feinfühlig eingestellt werden kann.

Ventildesign und Dichtungsschema

Es gibt zahlreiche Möglichkeiten, Ventile zu realisieren. Hier sind einige aufgelistet, die ich ausprobiert habe:

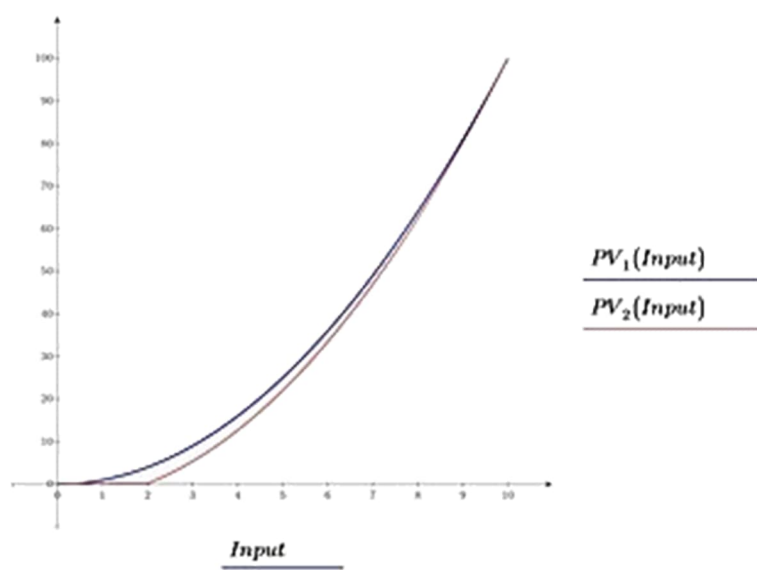


Abb. 5: Durchflusscharakteristik eines Proportionalventils. Die blaue Kurve zeigt die Durchflusscharakteristik eines Ventils ohne, die rote Kurve mit Tot-Zone (aus [3])

- **Kolbenventile**, die beim Verschieben eines Kolbens Ventilausgänge verschließen bzw. öffnen.
- **Drehrohrventile**, bei denen konzentrische Rohre mit Öffnungen gegeneinander verdreht werden können, um Luftstrom bei fluchtenden Öffnungen freizugeben.
- **Drehscheibenventile**, bei denen eine Scheibe mit Luftkanälen auf einer Dichtfläche mit Löchern gedreht oder verschoben wird.
- **Drosselventil-Kombination**: Mit vier Drosselventilen kann ein kontinuierliches Umschaltventil realisiert werden. Für jeden Arbeitsanschluss wird ein Ventilpaar verwendet: ein Ventil für die Einleitung von Luft in die Ventilkammer, eines für deren Entlüftung. Während der Betätigung dürfen wechselseitig nur zwei Drosselventile geöffnet sein: Während die Druckleitung auf den einen Arbeitsanschluss geschaltet ist, ist der andere Anschluss mit der Entlüftung verbunden. Die übrigen Ventile sind geschlossen. Die Kopplung der vier Ventile geschieht je nach Ventiltyp mechanisch oder elektrisch.

Je größer das Ventilvolumen ist, desto langsamer reagiert das Ventil, weil zusätzlich zu den Kammern im Pneumatik-Zylinder auch noch die Kammern im Proportionalventil gefüllt bzw. entlüftet werden müssen.

Bei Druckanlagen mit Druckverlust muss die Pumpe leistungsstark genug sein, um die Verluste auszugleichen.

Test-Aufbau und Steuerung

Für die im weiteren Verlauf vorgestellten Versuche zur Regelung der Position eines Pneumatik-Zylinders wurde der Aufbau in Abb. 6 verwendet.

Der Pneumatik-Zylinder bewegt einen Hebelarm, dessen Winkel zur horizontalen Ebene mit einem AS5600-Sensor [21] magnetisch gemessen wird. Der Sensor kann über den I²C-Bus ausgelesen werden. Der Sensor-Wert ist die Ist-Größe, die mit der Soll-Größe, die im Steuerprogramm eingegeben werden kann, verglichen wird. Die Regelung erfolgt mit einem Arduino mit Hilfe einer PID-Regler-Bibliothek.

Für Regelventile mit Servo-Ansteuerung wird die Stellgröße als PWM-Signal für den Servo ausgegeben. Digitale Ventile werden von Schaltausgängen über einen ULN2003-Darlington-Array-IC angesteuert [20].

Für proportionale Ventile werden PWM-Signale erzeugt, die ebenfalls über ein Darlington-Array verstärkt werden.

Die Druckerzeugung erfolgt mit einem fischertechnik-Kompressor, der an einem Tank angeschlossen ist. Damit der Kompressor nicht die ganze Zeit laufen muss, wird er bei Erreichen des maximalen Druckwertes abgeschaltet. Dies erfolgt mittels eines Betätigers, der gegen eine Stange mit Rückstellfeder drückt. Natürlich kann stattdessen auch ein Pneumatik-Zylinder mit Rückstellfeder verwendet werden. Beim Erreichen des Zieldrucks wird der Motor des Kompressors über einen Taster unterbrochen. Sinkt der Druck wieder, wird der Schalter geschlossen und der Kompres-

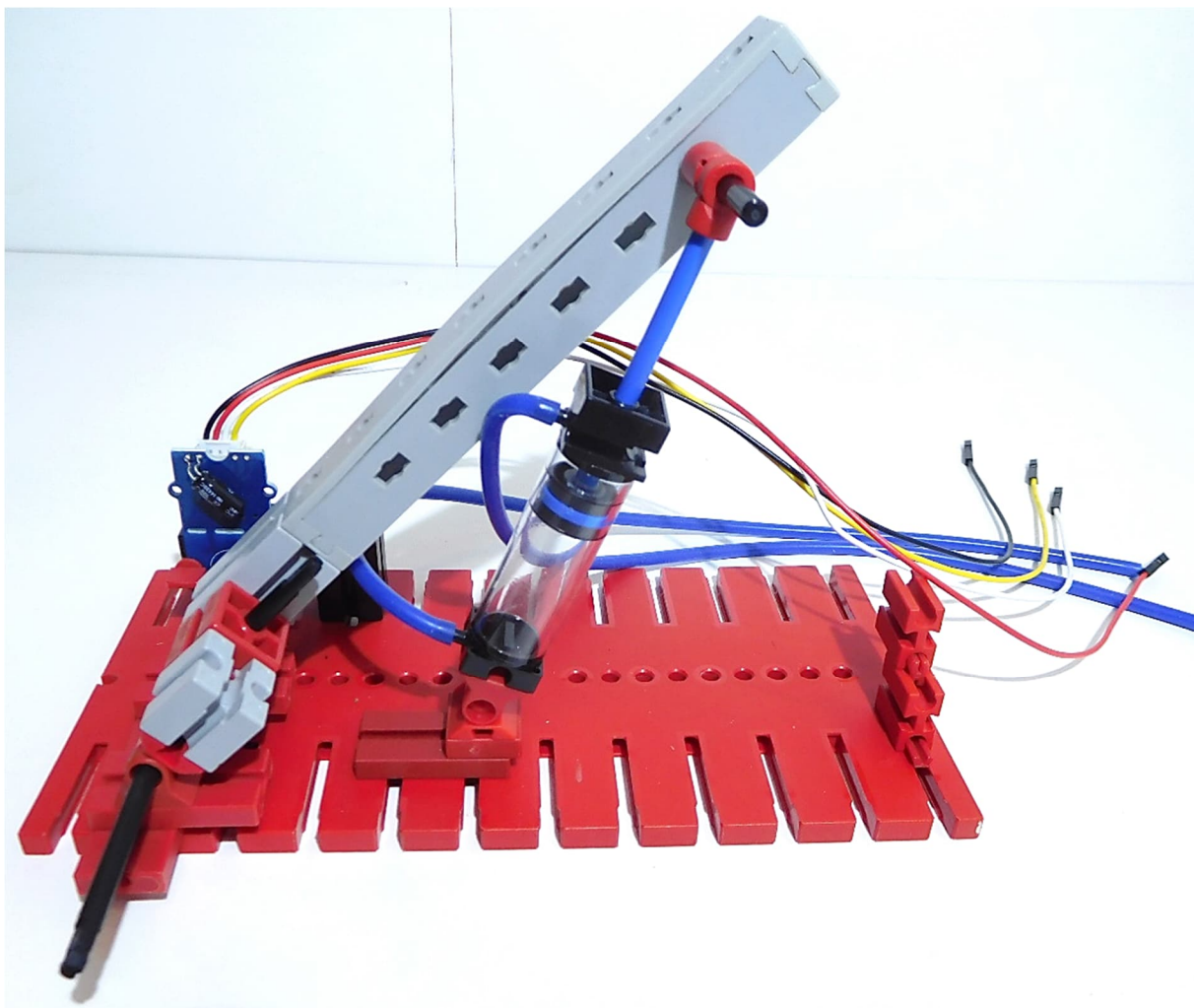


Abb. 6: Durch einen Pneumatik-Zylinder 60 (36938) bewegter Hebelarm mit magnetischem AS5600 Winkel-Encoder zur Messung des Anstellwinkels

sor läuft wieder an. Bei einem ausreichend abgedichteten Aufbau arbeitet der Kompressor nur dann, wenn ein größerer Druckverlust auftritt, wie er während der Betätigung der Ventile entstehen kann.

Regelung

Die Regelung der Position erfolgt mit einer PID-Regler-Library für den Arduino.

In diesem Abschnitt soll kurz erklärt werden, wie der Regler für eine Steuerung für den vorher beschriebenen Aufbau mit einem Servo-betriebenen Proportional-Ventil zu parametrisieren ist.

Zunächst müssen die Limit-Positionen für den Servo ermittelt und im Programm berücksichtigt werden. Diese Positionen sollte der Servo nicht überschreiten. Dann benötigt man die Servo-Steuerwerte, bei denen das Ventil für eine Abwärts- bzw. Aufwärtsbewegung eingestellt wird.

Darüber hinaus benötigt man die PID-Regler-Parameter kp , ki , und kd für die Berechnung des Ausgangswertes (Stellwert) aus der Abweichung zwischen Ist- und Sollwert. Diese Werte hängen vom verwendeten Drehwinkelgeber, dem Servo, dessen mechanischer Ankopplung an das Steuerventil, dessen Steuercharakteristik sowie dem Druck ab und werden empirisch ermittelt.

In der Regel beginnt man ohne Integrations- und Differentialanteil und setzt den Proportionalanteil kp so, dass keine Oszillationen auftreten [22]. Mit dem Integrations-Wert ki kann dann ein möglicher Offset ausgeglichen werden. Mit dem Differenzial-Wert kd kann man mögliches Überschießen über die Sollposition verhindern. Hier ist etwas Herumprobieren notwendig, was aber durchaus Spaß macht. Mit Drosselventilen – vor allem in den Abluftzweigen – kann man den Druck verringern und dadurch ebenfalls versuchen, Oszillationen entgegenzuwirken.

Hier eine mögliche Beispiel-Konfiguration:

```
double kp=0.35, ki=2, kd=0.0001;
```

Mit diesen Werten wird der Regler initialisiert:

```
PID servoPID = PID (&sinput,
&soutput, &ssetpoint, kp, ki, kd,
DIRECT);
```

Dabei bezeichnet *sinput* die Variable für den gemessenen Ist-Wert, *ssetpoint* die für den Soll-Wert und *soutput* die für den Ausgabe- oder Stellwert.

Die Nullposition für den Ausgabewert ist der Mittelwert der Servo-Extrem-Positionen. Damit werden dann die Regelgrenzen festgelegt:

```
servoPID.SetOutputLimits(
-(u_slimit - l_slimit) / 2,
(u_slimit - l_slimit) / 2)
```

In der Regelschleife wird dann nach Übergabe der gemessenen Ist-Position *sinput* ein neuer Output-Wert *soutput* berechnet:

```
servoPID.Compute();
```

Bei unserem Hebel-Aktuator liegt wegen der nach unten wirkenden Schwerkraft verständlicherweise eine Asymmetrie bei der Wirkrichtung vor, die wir empirisch kompensieren können: Da für die Abwärtsbewegung weniger Kraft benötigt wird, verringern wir die Stellgröße (hier beispielsweise auf 35 %):

```
if (soutput < 0) soutput *= 0.35;
```

Bei Ventilen mit einer Tot-Zone benötigen wir noch eine weitere Korrektur: Beim Anheben des Hebels müssen wir einen oberen Grenzwert angeben, ab dem sich der Kolben bewegt (hier 70):

```
if (soutput > 0)
    soutput = 70 + soutput ;
```

Beim Absenken müssen wir einen unteren Grenzwert (hier 20) einhalten.

```
if (soutput < 0)
    soutput = 20 + soutput;
```

Es ist offensichtlich, dass der Servo bei Schwankungen um die Sollposition herum immer zwischen diesen beiden Grenzwerten hin- und herfährt. Um diesen Effekt zu minimieren, schalten wir das Ventil dicht, wenn die Sollposition ungefähr erreicht ist, indem wir es auf die Mitte stellen:

```
if (abs(d - d_soll) < 4)
    soutput = 45;
```

Die Arduino-Steuerprogramme zu diesem Beitrag werden als Download zur Verfügung gestellt.

Diese Werte müssen für jeden Aufbau ermittelt werden.

Es sei angemerkt, dass jede Berechnung oder Ausgabe in der Regelschleife das Regelverhalten beeinflussen kann, da dadurch das Timing verändert wird. Wenn sich ein Regler partout nicht einstellen lässt oder erratisches Verhalten zeigt, kann es

hilfreich sein, die zusätzlichen Zeitfresser auszuschalten.

Realisierungsideen

Im Folgenden werden Realisierungsideen für Proportionalventile vorgestellt, die ich mit fischertechnik und zum Teil selbst gefertigten Bauteilen umgesetzt habe.

Proportionalventil mit Spritzen

Um die oben vorgestellten Kolbenventile zu realisieren, benötigt man Zylinder und passende Dichtungen, welche im Hobbybereich nicht ganz einfach zu bekommen sind. Eine Idee war, gewöhnliche Spritzen aus dem medizinischen Bedarf zu verwenden. Es gibt zwei Arten von Spritzen: mit und ohne Dichtgummis. Nur die mit Dichtungen sind geeignet, da sie einen geringeren Widerstand aufweisen als ihre dichtungslosen Kontrahenten.

Da sich die Entlüftungsöffnung bei einem Kolben-Proportionalventil in der Mitte

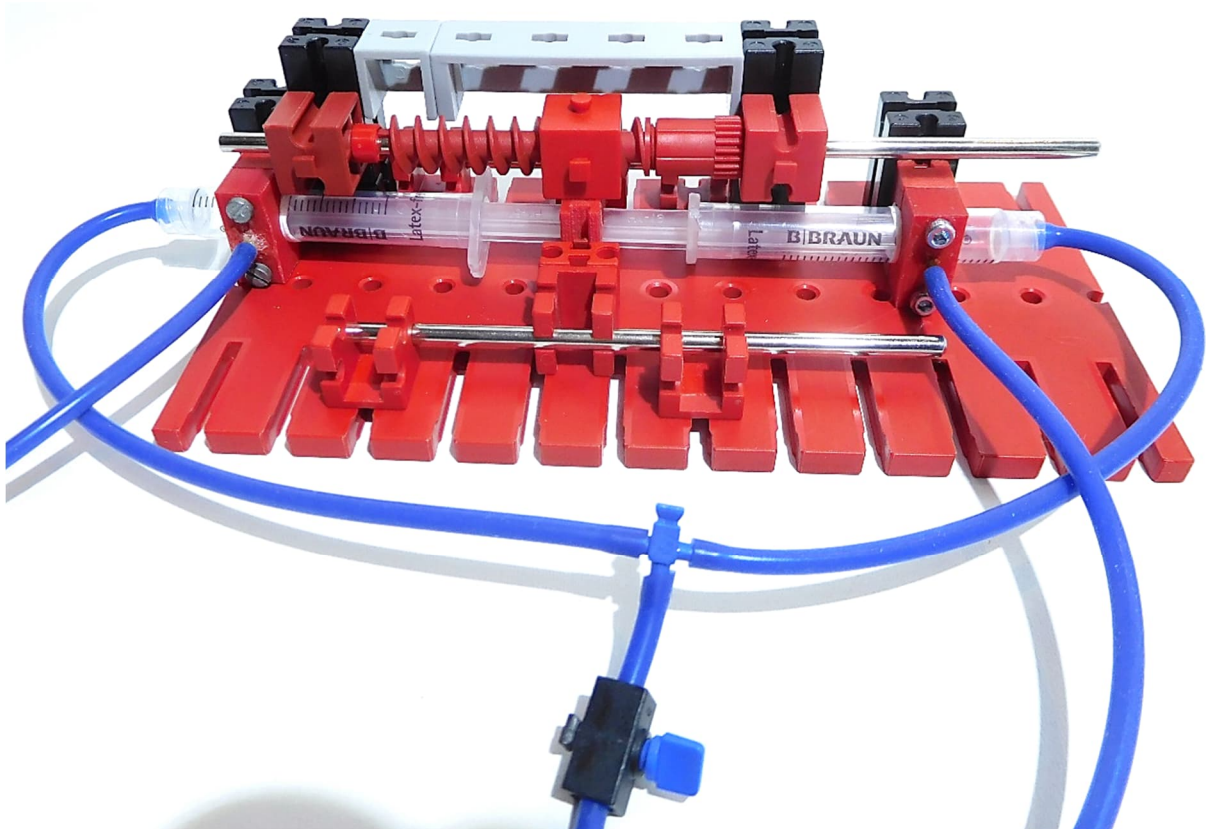


Abb. 7: Proportionalventil aus Spritzen mit Schneckenantrieb

befindet, kann man zwei Spritzen am Kolben miteinander verbinden. Eine druckdichte Durchföhrung für eine Kolbenstange wird nicht benötigt.

Der Druck wird durch die Austrittsöffnungen für die Nadeln der Spritzen eingeleitet. An der Seite sind jeweils Bohrungen für die Arbeitsanschlüsse zu setzen. Wenn sich der Dichtgummi zwischen Austrittsöffnung und Arbeitsanschluss befindet, ist dieser mit der Umgebung verbunden und die angeschlossene Zylinderkammer wird entlüftet.

Befindet sich der Dichtgummi auf der anderen Seite des Arbeitsanschlusses, gelangt Druckluft in die Kammer des Arbeitszylinders. Über das Verbindungs-Schloss wird die Spritze für den anderen Zylinder in die Gegenposition bewegt.

Um das Ganze für fischertechnik nutzbar zu machen, habe ich ein paar 3D-Druck-Teile entwickelt:

- Ein Joch als Halterung der Spritzen mit Arbeitsanschlüssen: Um den Anschluss dicht zu bekommen, wird eine Lage Teflon-Dichtband um die Spritze gewickelt und am Anschlussloch perforiert. Für den Anschluss der Pneumatik-Schläuche werden 2-mm-Kunststoffröhrchen in die Öffnungen der Joche eingeklebt.
- Ein Schloss, mit dem zwei Spritzen an der Drückerseite verbunden werden können: Das Schloss ist so gestaltet, dass ein Laufstein für einen Schneckenantrieb aufgesteckt werden kann

Abb. 7 zeigt den Aufbau mit einem Schneckenantrieb. Um eine geradlinige Bewegung der Spritzenkolben zu gewährleisten, sind Führungs-Stangen vorgesehen. Der Schneckenantrieb kann von einem XM-Motor angetrieben werden.

Mit diesem Modell kann man zwar das Prinzip eines 5/2-Wege-Proportionalventils demonstrieren, allerdings ist die Performance im Verhältnis zum Aufwand nicht überzeugend. Da die Dichtgummis eine

Breite von etwa 5 mm und die Arbeitsöffnungen 2 mm haben, hat dieses Ventil eine große Tot-Zone, was ein ordentliches Kurbeln für die Umschaltung erfordert. Eine Positionsregelung mit diesem Ventil ist schwierig und der Einsatz eines starken Motors zum Betätigen des Ventils lässt den Aufbau fragwürdig erscheinen. Daher will ich im Folgenden Alternativen zum Prinzip des Kolben-Proportionalventils vorstellen.

fischertechnik-Umschaltventile

Für die manuelle Ansteuerung von Pneumatik-Zylindern hat fischertechnik Umschaltventile entwickelt. Ich habe mich gefragt, ob mit solch einem Umschaltventil eine servopneumatische Steuerung zu realisieren ist.

Bei dem in Abb. 8 gezeigten Modell erhält der Begriff „Servo-Pneumatik“ eine zweite Bedeutung, denn das Ventil wird von einem fischertechnik-Servo bedient. Den mechanischen Anschluss des Handventils an den Servo übernimmt ein MiniMot-Umlenk-Getriebe, in das der Ventilhebel eingeklemmt wird.

Die Druckzuföhr ist am mittleren Ausgang angeschlossen, die beiden Zylinder-Anschlüsse an den äußeren Anschlüssen. Zusätzlich befindet sich ein Drosselventil in der Druckleitung.

Die fischertechnik-Umschaltventile haben praktischerweise eine interne Entlüftung: Wenn die eine Seite mit dem Druckanschluss verbunden ist, wird die andere entlüftet.

Das Umschaltventil hat eine große Tot-Zone und damit eine hohe Hysterese. Das bedeutet, dass man für die Umkehrung der Bewegung das Ventil recht weit in die Gegenrichtung bewegen muss. Im Regler-Programm wird dem durch eine Tot-Zonen-Korrektur Rechnung getragen. Dazu wird für jede Richtung auf die vom Regler berechnete Stellgröße ein entsprechender Offsetwert addiert, der das Ventil an die Grenze fährt, an der es öffnet.

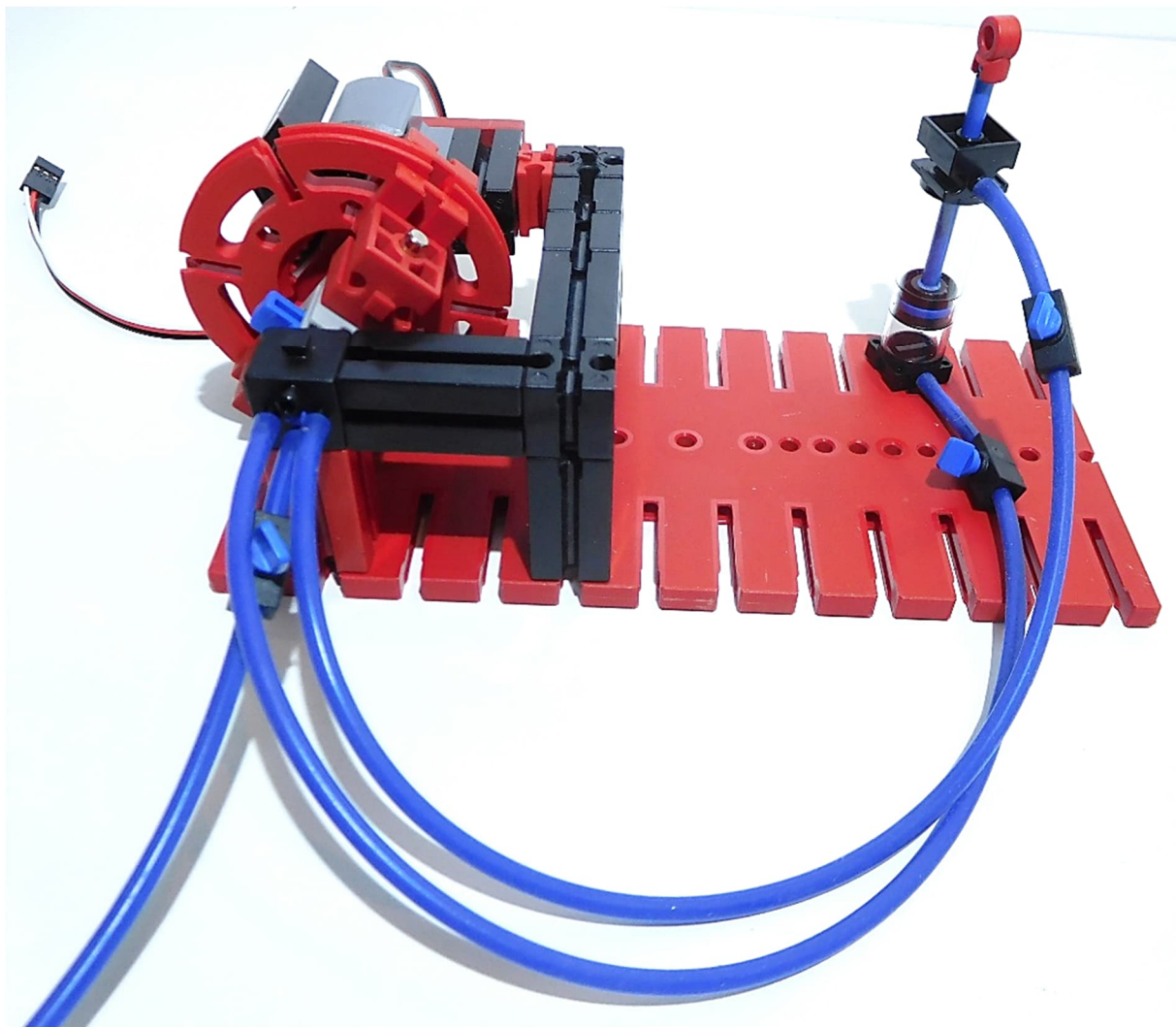


Abb. 8: fischertechnik-Umschaltventil mit Servoantrieb

Das Ventil ist in der mittleren Stellung komplett dicht. Dies kann man sich zu Nutze machen, indem man beim Erreichen der Sollposition das Ventil in diese Stellung bringt.

Hätte man keine Tot-Zonen-Korrektur, müsste man warten, bis der integrale Anteil des PID-Reglers hochläuft und der Regler-Output den Schwellenwert für eine Umschaltung überschreitet. Wegen der Trägheit der Integration kommt es dann zum Überschießen und zu Oszillationen. Diese bekommt man mit der Tot-Zonen-Korrektur nicht ganz weg. Um den Zylinder einigermaßen steuern zu können, muss man den Druck stark drosseln, was die Steuerung langsamer macht.

Um die Regelbarkeit dieses Aufbaus zu verbessern, kann man eine Abluft-Drossel in das Ventil einbauen, wie in [8] beschrieben.

Drehrohrventil (3D-Druck)

Ein sehr leistungsfähiges, relativ einfach zu realisierendes Ventil kann mit einem Rohr gebaut werden, das sich in einem Ventilkörper drehen kann. Ventilkörper und Rohr besitzen Öffnungen, die durch Drehung in Deckung gebracht werden und dadurch den Luftstrom leiten können.

Eine Idee für eine proportionale Drehkolben-Steuerung ist in Abb. 9 zu sehen. Das Röhrchen besitzt zwei Kammern, die in der Mitte getrennt sind, und jede Ventil-

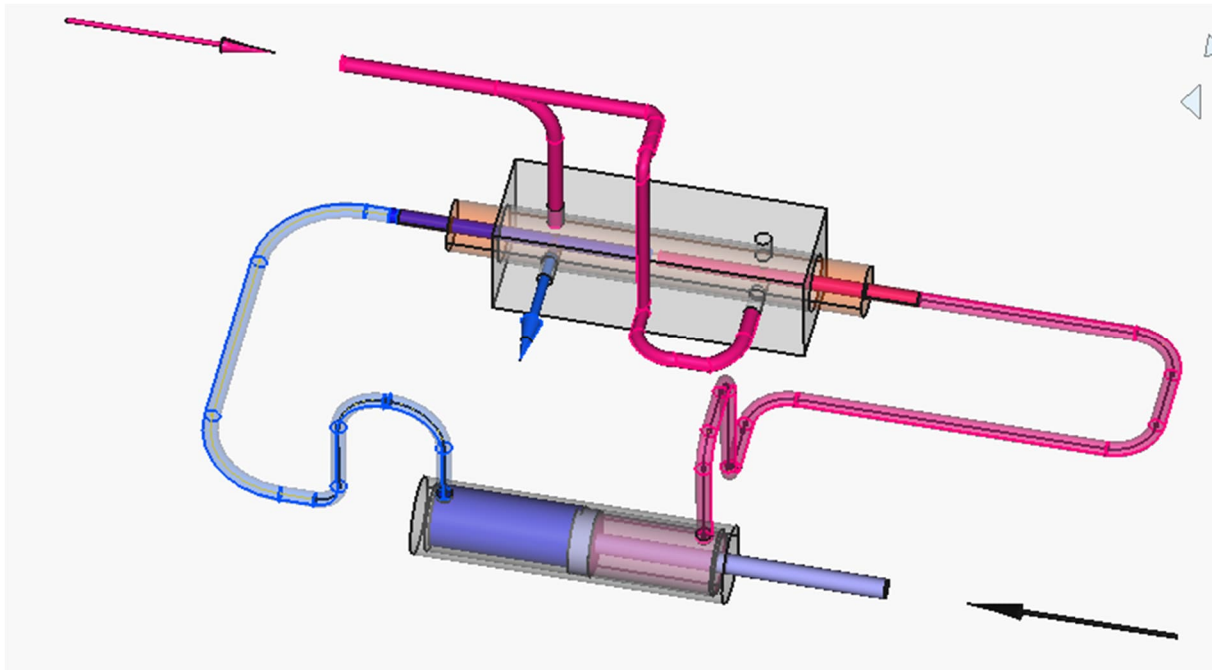


Abb. 9: Drehrohr-Proportionalventil: Ventilblock mit Drehrohr (orange) hinten, dargestellt bei Druckluftzufuhr der rechten und Entlüftung der linken Kammer

kammer besitzt eine Öffnung. Die beiden Enden des Röhrchens sind mit den Arbeitsanschlüssen eines doppelwirkenden Pneumatik-Zylinders verbunden. Im Ventil-

gehäuse befinden sich um 90° versetzte Bohrungen, die durch Drehen des Ventiltröhrchens mit dessen Öffnungen in Deckung gebracht werden können.

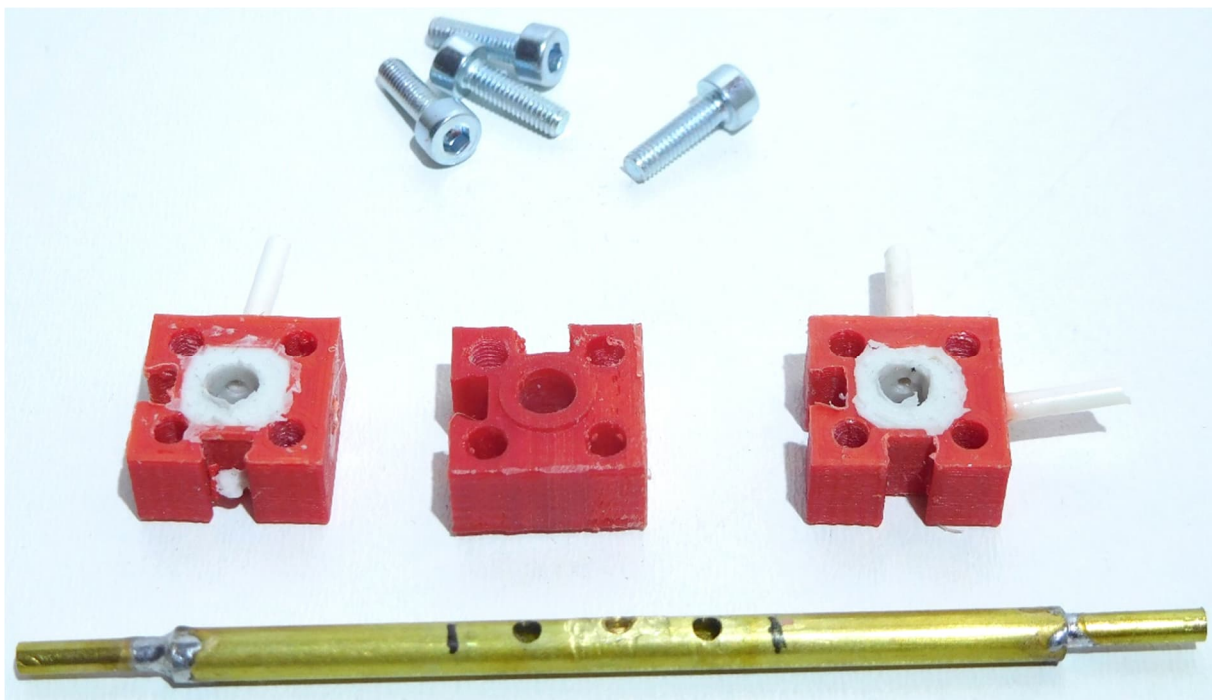
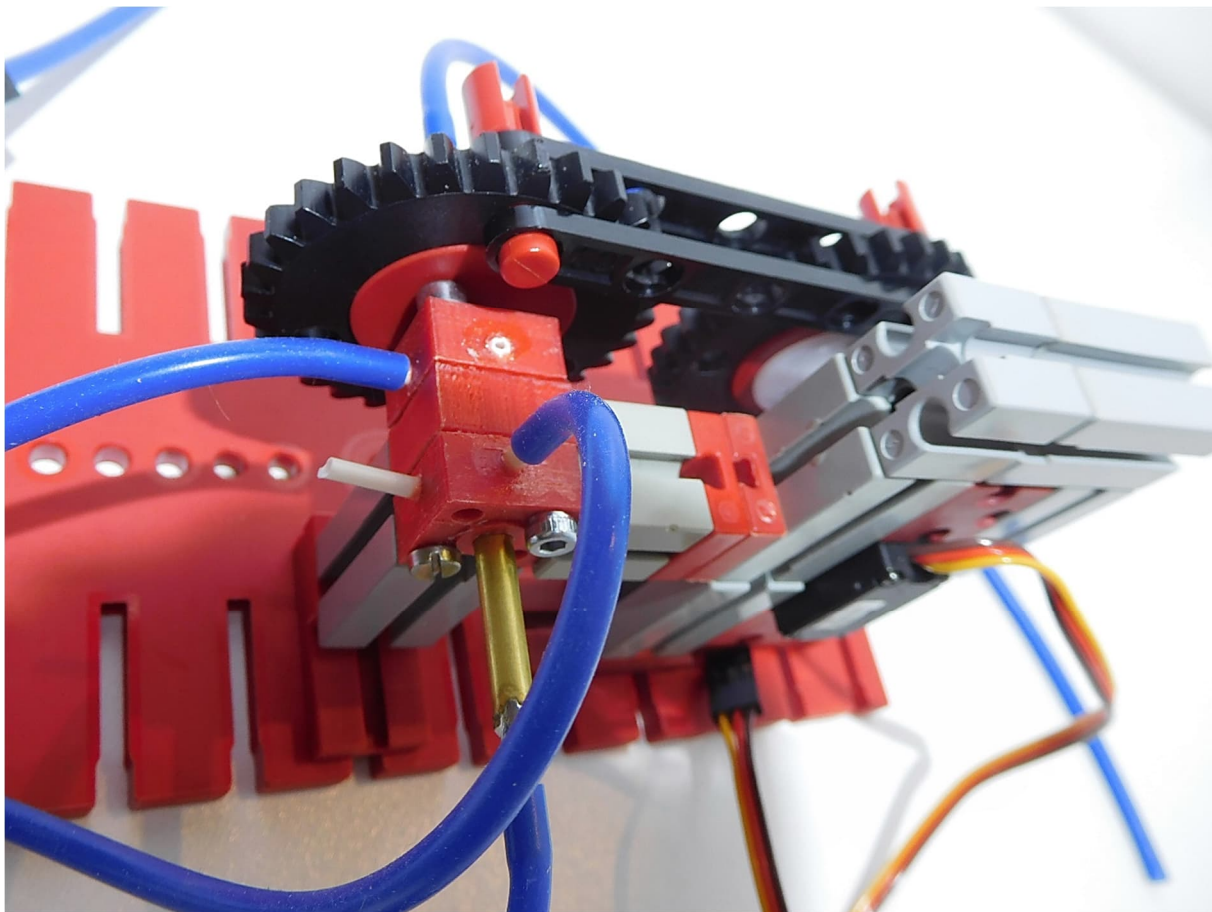


Abb. 10: Ventilelemente aus dem 3D-Drucker mit Silikondichtungen und Messingröhrchen zum Aufbau eines Drehrohr-Steuerventils



*Abb. 11: Drehrohr-Proportionalventil mit Ansteuerung durch Modellbau-Servo.
Die Druckleitung ist an den Anschlüssen links und oben angeschlossen*

In Abb. 9 ist dargestellt, wie die Pumpe mit der rechten Seite des Arbeitszylinders (rosa) und die linke Seite mit dem Entlüftungsausgang (blau) verbunden ist. Die rechte Seite wird mit Druck beaufschlagt, die linke Seite wird entlüftet. Dadurch bewegt sich der Kolben nach links. Dreht man das Ventilröhrchen im Uhrzeigersinn um 90°, kehrt sich die Wirkrichtung um.

Damit das gut funktioniert, sollte das Röhrchen im Ventilkörper luftdicht sitzen. Durch die Öffnungen im Röhrchen sollte keine Luft strömen, wenn sie sich nicht über einem Loch im Ventilkörper befinden.

Der erste Versuch bestand aus zwei ineinander gesteckten, gebohrten Messingröhrchen, die mit viel Armaturenfett eingeschmiert waren. Dies hat zwar funktioniert,

allerdings war die Dichtigkeit nicht gut, weil der Innendurchmesser des von mir verwendeten äußeren Röhrchens minimal größer als der Außendurchmesser des Inneren war. Deshalb habe ich spezielle 3D-Druck-Teile mit Aussparung für eine Silikon-Dichtung hergestellt.

In Abb. 10 sind die Ventilelemente und das präparierte Messingröhrchen zu sehen. Für die Herstellung der Endstücke mit den Ventilöffnungen werden in alle Öffnungen Messingröhrchen gesteckt und die Dichttasche in der Mitte mit Sanitärsilikon gefüllt.

Um eine glatte Fläche zu bekommen, wird noch eine Platte mit 4-mm-Loch daraufgesetzt. Nach einer Trocknungszeit von etwa zwei Tagen können die Platte und die Messingröhrchen entfernt werden.

Als Ventilröhrchen wird ein Messingröhrchen mit 4 mm Durchmesser verwendet, in das zur Trennung der Ventilkammern in der Mitte ein 5 mm langer Messingzylinder mit 3 mm Durchmesser eingelötet bzw. eingeklebt ist.

In die Öffnungen der Ventilblöckchen werden dann 2-mm-Plastik- oder Messingröhrchen für die Druckanschlüsse eingeklebt. Die beiden Endstücke werden an das Mittelstück geschraubt, in dessen Öffnungen ein M3-Gewinde geschnitten wird.

Zur besseren Befestigung des Zylinders wird das Messingröhrchen in der Mitte angebohrt. Durch das Loch kann Lötzinn oder Klebstoff eingebracht werden. Zusätzlich sind in das Röhrchen 2-mm-Löcher so gebohrt, dass sie mit den Öffnungen in den Ventilblöcken fluchten. Für die genaue Positionierung der Löcher ist es praktisch, das Röhrchen in die zusammengeschraubten Ventilblöcke zu stecken und deren Luftkanäle als Bohrschablone zu nutzen, bevor

die Plastikröhrchen für den Schlauchanschluss eingeklebt werden.

An den Endstücken sind für den Anschluss der Pneumatik-Schläuche 3-mm-Röhrchen angelötet.

Die Dichtigkeit der Ventilelemente ist sehr gut, vor allem, wenn man die Dichtung etwas quetscht. Dazu ist das Mittelstück beidseitig mit einer Erhöhung und vier Gewindelöchern versehen. Mit jeweils zwei oder vier Schrauben werden die Endstücke an das Mittelteil geschraubt und die Dichtungen an das Röhrchen angepresst.

Dieses Ventil ist leicht beweglich, und weil es eine moderate Tot-Zone aufweist, lässt sich damit eine gute Regelungs-Strecke realisieren. Der Antrieb erfolgt über einen Servo. Auch bei diesem Ventil kann eine Abluftdrosselung verwendet werden.

Drehschieberventil (3D-Druck)

Eine weitere elegante Variante von Jan Hanson (*juh* im ftc-Forum) möchte ich

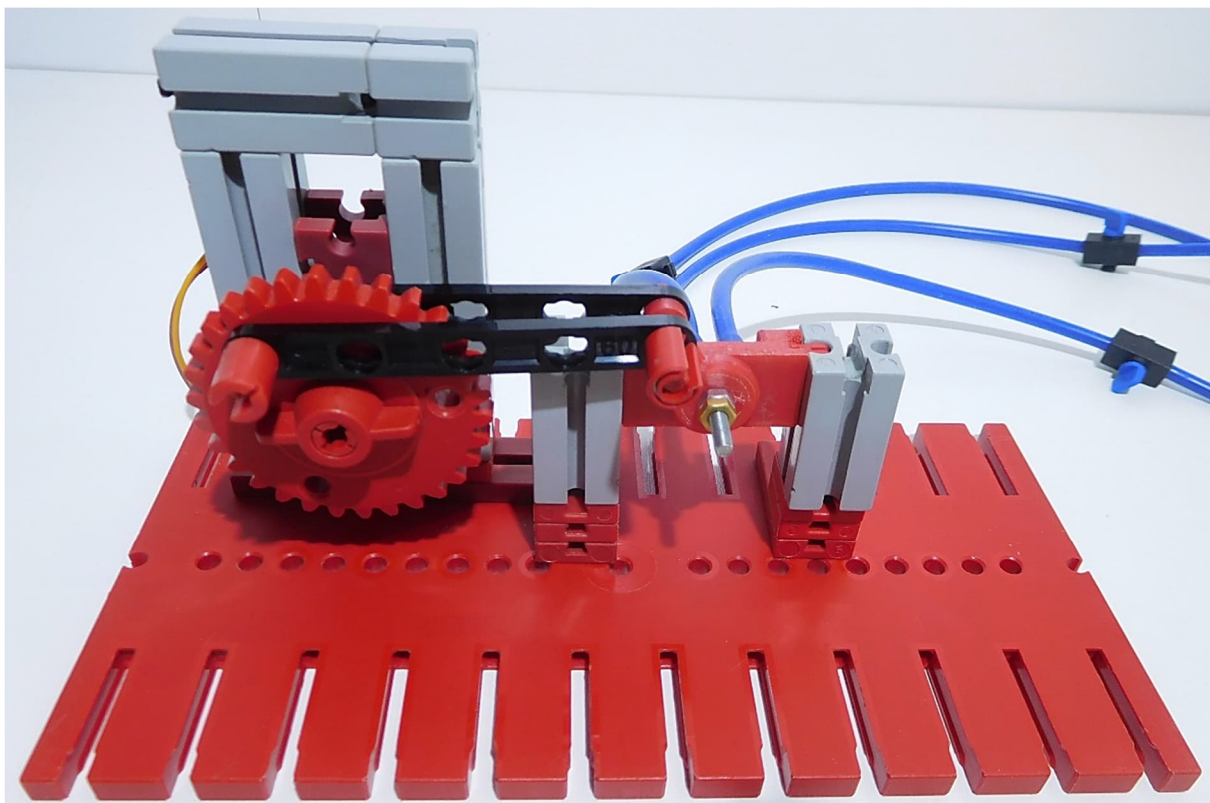


Abb. 12: Drehscheiben-Ventil mit Servo-Ansteuerung

ebenfalls vorstellen. Sie benötigt zwar auch zwei kleine, schnell druckbare 3D-Druck-Teile und Schrauben, allerdings kommt sie ohne Messingteile und Silikon aus. Die 3D-Druckdateien sind auf Thingiverse [13] veröffentlicht. Die von mir abgewandelte Variante stelle ich als Download zu diesem Beitrag zur Verfügung [19].

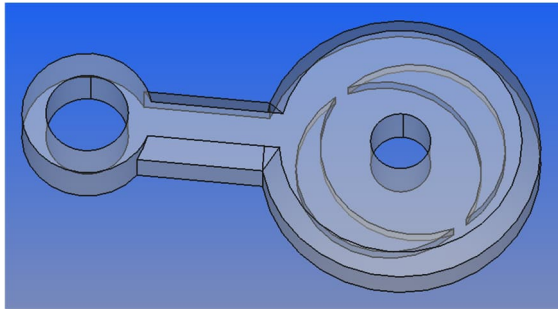


Abb. 13: Umschaltventilscheibe mit halbmondförmigen Luftkanälen und Servo-Hebel

Das Ventil besteht aus zwei Teilen: einer Halterung mit vier sehr feinen Löchern und einer Drehscheibe mit Luftkanälen und einem Hebel (siehe Abb. 13), an den ein Servo-Gestänge angeschlossen werden kann. Durch Drehen des Hebels werden jeweils zwei Löcher in der Halterung miteinander verbunden.

Um das Ventil dicht zu bekommen, sollte es mit 0,1 mm Auflösung gedruckt werden. Die Nachbearbeitung erfolgt mit Sandpapier bis zu einer 1200er Körnung. Anschließend werden die Dichtflächen der Teile auf einem Blatt Papier glatt gerieben, bis sie glänzen. Hebel und Halterung werden nach Einfettung mit Armaturenfett mit M3-Schraube,

Mutter und Kontermutter verbunden und dicht gezogen.

Das Ventil mit Servo-Ansteuerung ist in Abb. 12 gezeigt. Dieses Ventil ist etwas unstetig und benötigt wie das Umschaltventil eine Drossel in der Druckleitung und eine Tot-Zonen-Korrektur.

fischertechnik Schlauch-Knick-Proportionalventil

In der ft:pedia hat Stefan Falk eine Serie von Pneumatik-Modellen veröffentlicht (siehe [4, 5, 6]), die mich zu einer weiteren Variante inspiriert haben, die rein mit fischertechnik-Teilen realisiert werden kann: einem Schlauch-Knick-Ventil.

Durch kontrolliertes Abknicken der fischertechnik-Silikonschläuche kann man einen Luftstrom feinfühlig bis zur Unterbrechung einstellen. In Abb. 14 ist ein Ventil zu sehen, das sich dieses Prinzip zu Nutze macht. Es sind vier Schlauch-Schleifen vorhanden. Es gibt zwei Schlauchpaare. Jedes Paar ist auf der einen Seite mit dem Rahmen und in der Mitte mit einem beweglichen Ventil-Schlitten verbunden. Die Verbindungen sind auf der Rückseite des

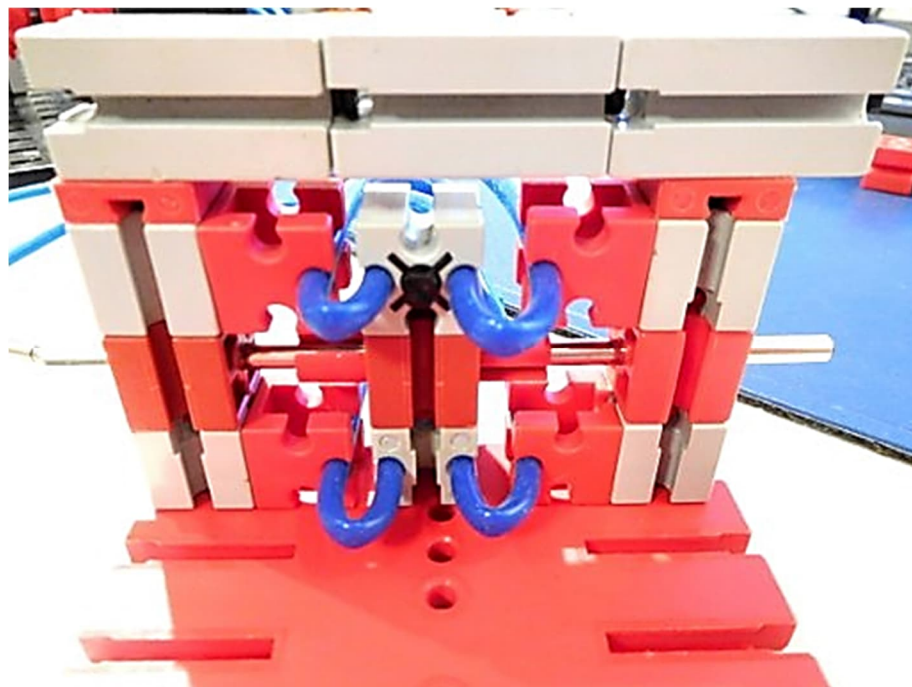


Abb. 14: Schlauch-Knick-Proportionalventil mit fischertechnik-Schläuchen

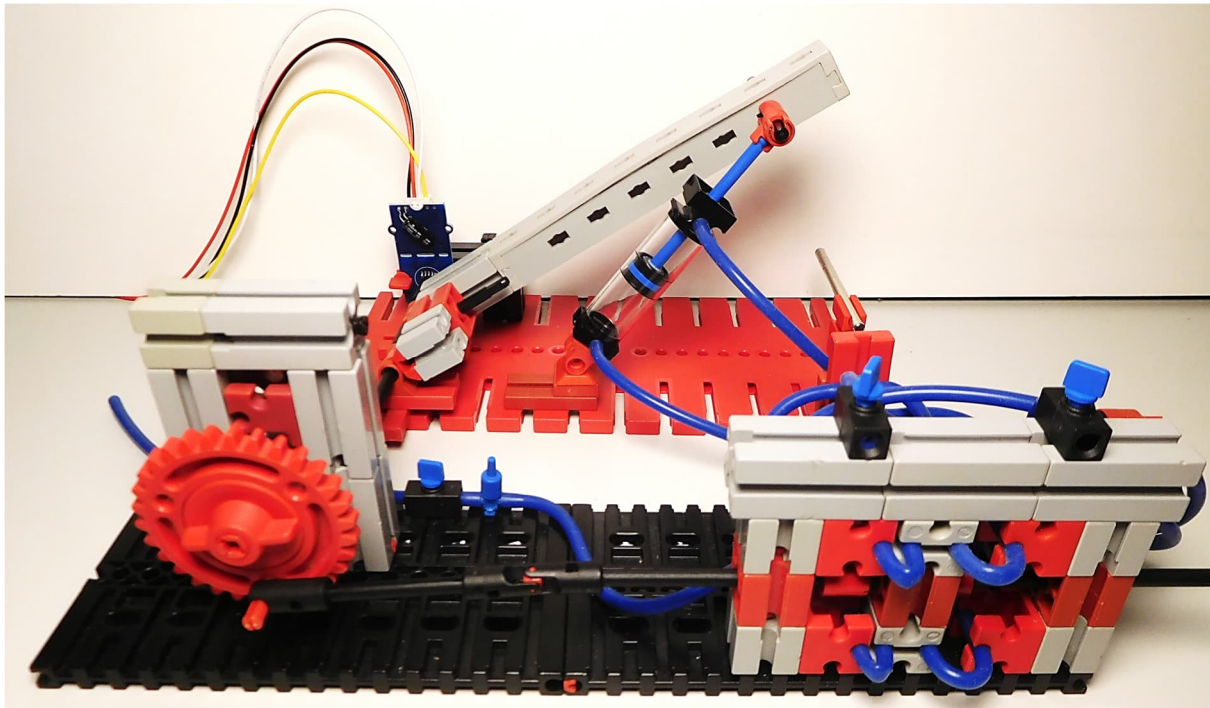


Abb. 15: Servo-Ansteuerung des Schlauch-Knick-Proportionalventils

Ventils an einer Säule mit T-Stücken gesteckt. Die Ansteuerung erfolgt wieder mit einem Servo (siehe Abb. 15).

Wird der Schlitten nach links gezogen, wird das linke Schlauch-Paar geknickt und der Luftstrom durch diese Schläuche unterbrochen, während das rechte Schlauch-Paar entlastet und der Luftstrom ermöglicht wird. Die Anschlüsse der mittleren Schlauchenden sind jeweils auf jeder Ebene (oben und unten) miteinander und zusätzlich jeweils mit den Arbeitsanschlüssen des fischertechnik-Zylinders verbunden. Zwei der am Rahmen befindlichen Schlauchenden sind diagonal und mit der Druckleitung verbunden. Die anderen beiden Schlauchenden sind offen (Entlüftung).

Der Ventil-Schlitten wird mit einer fischertechnik-Achse über eine Pleuelstange durch einen Servo angetrieben.

Wenn die Schlauch-Schleifen richtig eingestellt sind, erhält man ein sehr gut funktionierendes Proportional-Ventil.

Servopneumatik mit elektromechanischen 3/2-Wege-fischertechnik-Ventilen

Elektromagnetische fischertechnik-Ventile haben einen Druck-Eingang, einen Arbeits-Ausgang und eine Entlüftung. Ist das Ventil stromlos, ist der Arbeits-Ausgang entlüftet. Wird das Ventil unter Strom gesetzt, wird der Arbeits-Ausgang mit dem Druck-Eingang verbunden.

Mit zwei dieser Ventile lässt sich durch wechselseitige Aktivierung ein Kolben hin- und herbewegen. Das pneumatische Schaubild ist in Abb. 16 gezeigt.

Für die elektronische Ansteuerung dieser Ventile eignen sich Darlington-Transistoren. Ich habe dazu das Darlington-Array ULN 2003 verwendet, das ich mit einem Arduino-MEGA ansteuere.

Diese Ventile arbeiten nicht stetig. Man kann mit ihnen aber eine Zweipunkt-Regelung [18] realisieren, wenn man den Pumpendruck durch ein Drosselventil stark begrenzt und auch die Abluft drosselt. Wenn der Sollwert zu gering ist, schaltet

man ein Ventil ein, das andere aus. Ist der Sollwert zu groß, dreht man die Schalterverhältnisse um.

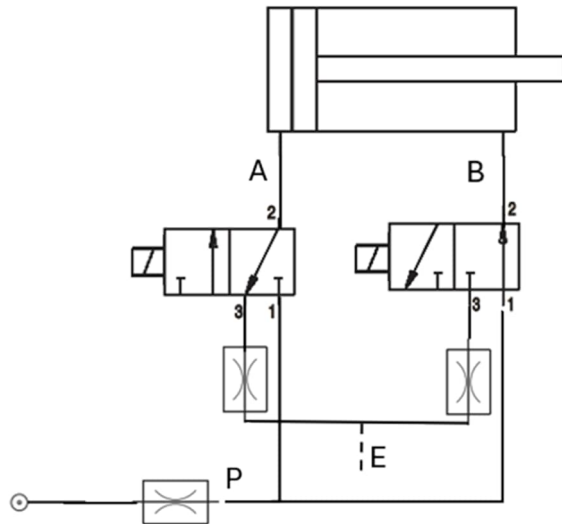


Abb. 16: Pneumatisches Schaltbild mit elektromagnetischen 3/2-Wege-Ventilen und Drosseln in Zu- und Ableitungen

Man kann auch eine Luftstoß-Regelung realisieren, bei der die Ventile nur für einen Bruchteil einer Sekunde geöffnet und dann wieder geschlossen werden. Man leitet dann für die Schaltdauer Luft ein.

Ich hätte es nicht erwartet, aber die Regelung funktioniert erstaunlich gut.

Servopneumatik mit Proportional-Drosselventilen

Es gibt auch elektromechanische proportional-Drossel-Ventile, mit denen man eine kontinuierliche Regelung aufbauen kann. Diese Ventile sind normalerweise recht teuer. Auf einer der bekannten Online-Plattformen habe ich günstig vier gebrauchte Ventile der Firma Parker [16] aus einer „Zahnarzt-Bohrinsel“ erstanden, mit denen einst die Turbinen von Bohrern angesteuert wurden.

Mit diesen Ventilen kann man den Luftstrom kontinuierlich steuern und damit eine genaue Positions-Regelung aufbauen. Da die von mir erstandenen Exemplare eine Schalt-Spannung von 20 V benötigen, musste ich sie wieder über einen ULN2003

ansteuern. Anstatt einer zuerst angedachten analogen Ansteuerung mit DA-Wandler und Leistungsverstärker habe ich mich nach Ermunterung im fischertechnik-Forum für eine PWM-Ansteuerung entschieden.

Abb. 18 zeigt den Ventilblock mit den vier Proportional-Drosseln. Jeweils zwei Ventile sind für die Druckluft-Zufuhr und die anderen beiden für die Entlüftung da. Soll sich der Kolben aufwärts bewegen, muss die dafür zuständige Zylinderkammer mit der Druckversorgung verbunden und das Entlüftungsventil für diese Kammer geschlossen werden. Die andere Zylinderkammer muss von der Druckluft-Zufuhr getrennt und dafür mit der Entlüftung verbunden werden.

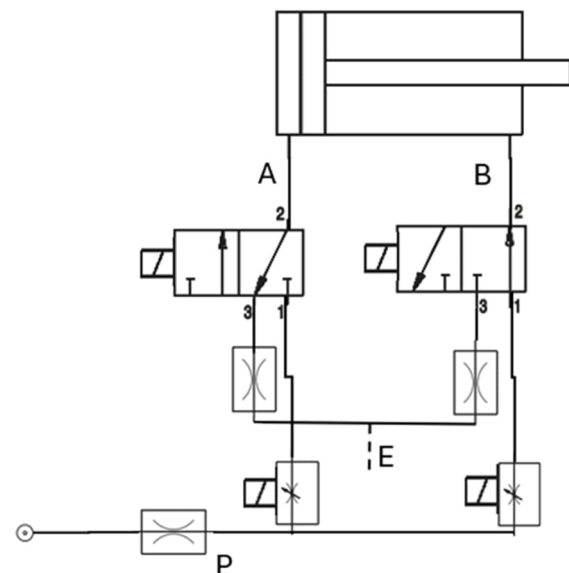


Abb. 17: Servopneumatische Ansteuerung eines doppelt-wirkenden Zylinder mit zwei elektromagnetisch schaltbaren 3/2-Wege-Ventilen und zwei elektrisch einstellbaren Proportional-Drosselventilen, über die der einströmende Luftstrom sehr fein eingestellt werden kann

Die Realisierung dieses Ventils erfordert vier teure Proportional-Ventile. Um Kosten zu sparen, kann man die zwei für die Entlüftung vorgesehenen proportional-Ventile wie in Abb. 17 dargestellt durch fischertechnik-Ventile ersetzen, an deren Entlüftungs-Ausgang jeweils ein Drosselventil angeschlossen ist.

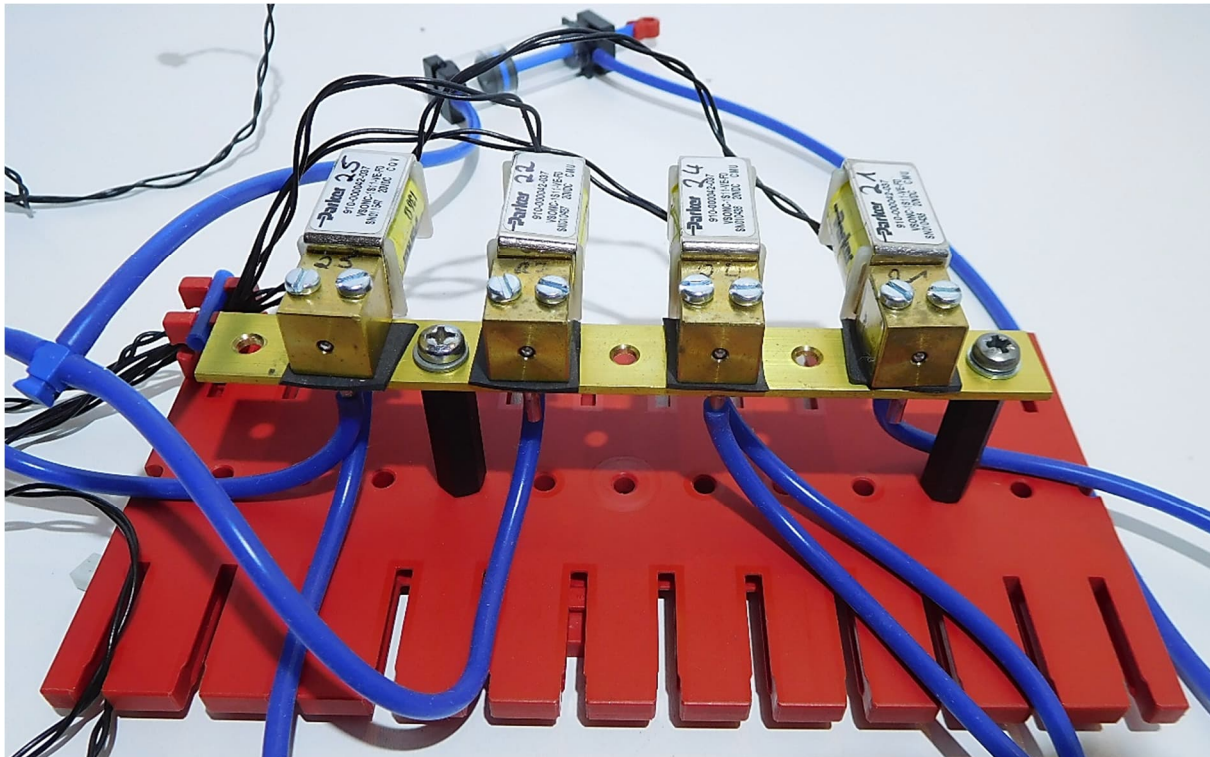


Abb. 18: Elektromagnetisch ansteuerbare Proportional-Drosselventile vom Typ Parker VSONC-1S11-VE-F0 auf einer gebohrten Messing-Leiste mit Stücken von gelochtem Fahrradschlauch als Dichtung. Für den Anschluss der Pneumatik-Schläuche sind auf der Unterseite der Leiste über die Löcher für die Ventil-Öffnungen 2,5-mm-Aderendhülsen angelötet.

Der Vorteil dieser Ventilsteuerung ist, dass man die Regelung bei unterschiedlichen Ventilverhältnissen – z. B. mit oder ohne Tot-Zone – gut untersuchen kann.

Vergleich

In der folgenden Tabelle habe ich versucht, einen Vergleich der vorgestellten Ventiltypen zu erstellen. In die Gesamtbeurteilung (Score) fließt ein, ob sich ein Ventiltyp mit reinen fischertechnik-Teilen realisieren lässt, welcher Aufwand bzw. welche Kosten entstehen, ob sich das Ventil komplett sperren lässt, welche Hysterese es hat und wie es um seine Regelbarkeit bestellt ist. Dabei ist die Beurteilung der einzelnen Kategorien subjektiv. Es ist gut möglich, dass jemand mit einem guten Händchen und einer tollen Idee zu einem anderen Ergebnis kommt.

Meiner Meinung nach stellen sich 3/2-Wege-fischertechnik-Magnetventile gemäß

dieser Bewertung als die praktikabelste Lösung dar. Am vielseitigsten finde ich den Ventiltyp mit zwei Proportionaldrosseln und zwei 3/2-Wege-Magnetventilen. Das Quetschventil ist die günstigste und intuitivste Variante. Die anderen Lösungen erfordern etwas Bastelarbeit.

Zusammenfassung und Ausblick

Servopneumatik kann auch mit fischertechnik umgesetzt werden. Sie ermöglicht präzise, elektronisch gesteuerte Bewegungen pneumatischer Modelle. In diesem Beitrag habe ich verschiedene funktionsfähige Ventilmodelle vorgestellt. Einige kommen mit reinen fischertechnik-Bau-elementen aus, andere kann man mit Teilen aus dem 3D-Drucker und dem Baumarkt selbst herstellen. Für höhere Ansprüche sind die Modelle mit den genauen proportional-Drosseln zu empfehlen. Mit diesem

Ventiltyp	Fischertechnik	Kosten /Aufwand 0=hoch 3=niedrig	Vollsperrung mgl. 1=ja 0=nein	Hysterese 0=hoch 3=niedrig	Druckverlust 0=hoch 3=niedrig	Regelbarkeit 3=sehr gut 0=sehr schlecht	Score	Bemerkung
2 x 3/2 Magnetventile + Drosseln	1	2	1	3	2	2	11	3-Punkt-Regelung, schnell
2 x Proportionaldrosseln + 2 x 3/2 Magnetventile + Abluftdrosseln	0	1	1	3	3	3	11	Genaue Regelung, Kompromiss
4 x Proportionaldrosseln	0	0	1	3	3	3	10	Genaue Regelung, teure Ventile
4-fach Quetsch- Servoventil + Drosseln	1	3	1	1	1	2	9	empfindliche Justage
Drehrohrventil	0	1	1	2	1	3	8	3D-Druck, Messing, Silikondichtung
Drehplattenventil	0	2	1	2	1	1	7	3D-Druck
ft Umschaltventil + Drosseln	1	3	1	0	1	0	6	schwergängig, hohe Hysterese

Abb. 19: Vergleichstabelle der getesteten Varianten

Beitrag werden die Arduino-Programme und 3D-Druckdateien zum Download zur Verfügung gestellt.

Es ist auf jeden Fall ein spannendes Thema, welches es lohnt, weiter verfolgt zu werden. Die Erschließung dieser Technik für fischertechnik steht noch am Anfang, bietet aber viele Möglichkeiten für kreative Weiterentwicklungen und praxisnahe Experimente. An der Entwicklung könnt ihr im fischertechnik-Forum teilhaben [15]. Vielleicht fällt euch zu diesem Thema eine elegante Lösung ein.

Für Grundlagen der Pneumatik und Schaltzeichen sei auf das Pneumatik-Handbuch [11] verwiesen. Wer sich mit der mathematischen Modellierung eines doppeltwirkenden Zylinders interessiert, findet in [12] gute Ansätze.

Quellen

- [1] fischertechnik: *Liebherr R 938 - Pneumatischer Bagger*. Auf fischertechnik.de.
- [2] Festo: *Proportional-Wegeventil MPYE*. Auf festo.com.
- [3] Wikipedia: [Stetigventil](#).
- [4] Stefan Falk: *Druckluftsteuerungen (Teil 1)*. [ft:pedia 1/2014](#), S. 58–72.
- [5] Stefan Falk: *Druckluftsteuerungen (Teil 2)*. [ft:pedia 2/2014](#), S. 20–29.
- [6] Stefan Falk: *Druckluftsteuerungen (Teil 3)*. [ft:pedia 3/2014](#), S. 21–31.
- [7] Stefan Falk: *Druckluftsteuerungen (Teil 4)*. [ft:pedia 4/2015](#), S. 24–30.
- [8] Stefan Falk: *Abluftdrosselung mit dem Pneumatik-Handventil*. [ft:pedia 2/2014](#), S. 11.
- [9] Wikipedia: [Servoventil](#).
- [10] Wikipedia: [Proportionalventil](#).
- [11] Luciano Zaghis: *Die moderne Pneumatik*. Auf pneumaxspa.com, 2025.
- [12] Christoph Hübner: *Modellgestützte Instandsetzung*. Diplomarbeit, Hochschule Mittwerda, auf monami.hs-mittweida.de, 2009.
- [13] juh: *4/2-way pneumatic valve for fischertechnik*. Auf thingiverse.com, 2025.
- [14] Elmar Brix: *3-Zylinder Luftdruckmotor*. Im [Bilderpool](#) der ft-Community, 2012.

- [15] Dr. Florian Bauer: *Servo-Pneumatik mit Fischertechnik?* [Diskussion im fischertechnik-Forum](#) der ft-Community.
- [16] Parker: *Miniatür-Proportional-ventile, Precision Fluidics*. Auf parker.com, 2012.
- [17] Ludger (jmn): *Liebherr*. Diskussion über den Liebherr R938 im [Forum der ft-Community](#).
- [18] fischertechnik: *Beleuchtungsstärke-Regelung*. Modell mit Zweipunktregelung im hobby-4-Begleitbuch Band 2, S. 48. Auf ft-datenbank.de.
- [19] Florian Bauer: *Servopneumatik.zip*. [Download mit 3D-Druck-Dateien und Arduino-Software](#) zu diesem Beitrag, 2025.
- [20] STMicroelectronics: *ULN2001, ULN2002 ULN2003, ULN2004 – Seven Darlington arrays*. Datenblatt auf st.com, 2021.
- [21] ams-OSRAM AG: *ams AS5600 Position Sensor*. Auf ams-osram.com.
- [22] Dirk Fox: *PID-Regler – eine experimentelle Einführung*. [ft:pedia 4/2021](#), S. 62–72.

Elektromechanik

Synchronmotoren – revisited

Matthias Dettmer

Vor fast zehn Jahren habe ich in einem Beitrag beschrieben, wie ich Synchronmotoren mit unterschiedlichen Polpaarzahlen aufgebaut habe [1]. Die Grundlagen zu den spezifischen Drehzahlen und eine Überprüfung der gewünschten Drehzahlen wurden auch beschrieben. Mittlerweile waren die drei Synchronmotoren auf einigen Fan-Club-Tagen, auf vier Conventions, auf drei Maker-Faires und einer Modell-Ausstellung zu sehen. Zwischen den Events wurden die Motoren behutsam weiterentwickelt, um Fehler abzustellen, aber natürlich auch, um auch andere der Polpaarzahlen aufzubauen und in Betrieb zu nehmen.

Problemchen

Auf jeder der Veranstaltungen ließen sich die Motoren anwerfen, und nach dem Anwerfen liefen sie zumeist die Stunden der jeweiligen Veranstaltung durch. Nur war das Anwerfen mitunter gar nicht leicht, und die niedrigen Polpaarzahlen 1 (3000 U/min) und 2 (1500 U/min) wollten nur schwer (PPZ2) oder auch gar nicht angeworfen werden. Ursprünglich sollten Lagerung und Einstellung sowohl der Elektromagneten als auch der Achslager für den Rotor so flexibel wie möglich gehalten werden.

Meine ursprüngliche Konstruktion erlaubte, die Rotoren-Achse in Höhe und Abstand zu verstellen. Das führte leicht dazu, dass der Rotor „schräg“ gelagert war. Kurz gesagt hatten sowohl die Elektromagnete als auch die Rotorlagerung jeweils zwei Freiheitsgrade: rechts-links und rauf-runter.

Am Rotor der Polpaarzahl 5 waren die Lagerungen der Neodym-Magnete wegen der Kette auf dem Zahnrad und dem recht schwachen Formschluss zwischen dem ‚tragenden‘ Kettenglied und der Nut des Bausteins 7,5 sehr wackelig. Daher wurden die Permanentmagneten sehr oft (und dann schnell und gründlich) von der Metallfläche der Pole eines der Elektromagneten an- und

aus ihrer Halterung gezogen – und der Motor blieb stehen. Mittlerweile gibt es Neodym-Magneten in Stäbchenform auch in „fischertechnik-freundlichen“ Maßen, mit einem Durchmesser von 4 mm und einer Länge von 15 mm.

Zur Weiterentwicklung soll nun vor dem Bau nachgedacht und geplant werden. Ein wichtiges Ziel ist dabei eine merkliche Vereinfachung der Konstruktion. Meine Kernfragen dazu:

- Kann die Motorenserie mit einem einzigen festen Grundgestell, also für alle Varianten gleich gestaltet werden?
- Geht das Ganze mit jeweils nur einem Freiheitsgrad in der Lagerung für Elektromagneten und Rotorachslager?

So viel vorab: Beide Fragen können mit „Ja“ beantwortet werden. Die Konstruktion in diesem Beitrag wird mit Bauabschnitten aus dem fischertechnik-Designer von Michael Samek beschrieben.

Grundaufbau

Betrachtet man die bestehenden Rotoren, so fällt auf, dass alle bis auf den Rotor mit Polpaarzahl 6 (der hat 17 mm) nur höchstens 15 mm breit sind. Also legen wir uns auf

einen Abstand von 30 mm für die Lager der Rotoren-Achse fest.

Die Aufnahme der Elektromagnete wird auf je einem Baustein 30 und einem Baustein 15 passieren. Die beiden Bausteine 30 folgen dem gerade festgelegten Grundabstand der Achslager von 30 mm.

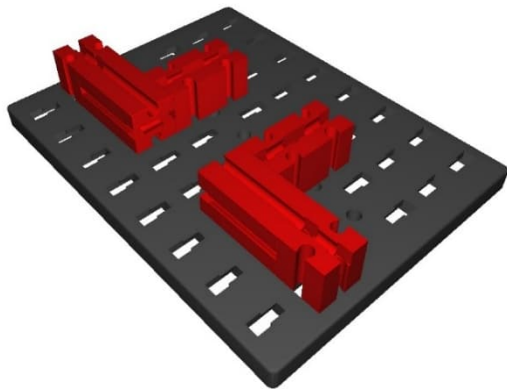


Abb. 1: Grundaufbau

Elektromagnete

Die Elektromagneten waren in den alten Modellen zuerst am Boden der Grundplatte angebracht – zwar gegeneinander beweglich, ansonsten aber für alle Modelle gleich hoch. Für den Rotor mit dem größten Durchmesser (Polpaarzahl 6 aus dem Buch „Technikgeschichte mit Fischertechnik“ [2]) ging das gut – bei kleineren Rotordurchmessern waren allerdings die Stecker im Weg. Tatsächlich ist dieser Rotor der einzige, bei dem die Spulen ganz unten liegen konnten - und schon benötigte ich für die Elektromagnete zwei Freiheitsgrade.

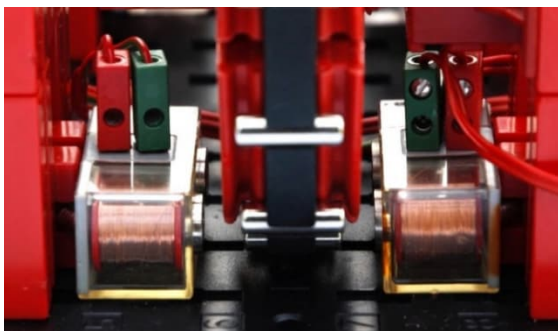


Abb. 2: Hier haben die Stecker noch genug Platz

Einfache und schnelle Lösung dieser Problemstellung: Die Elektromagnete müssen seitlich angebracht werden, am besten mit der immer gleichen und für alle Rotordurchmesser ausreichenden Höhe über Grund.

Der Rotor mit Polpaarzahl 6 hat einen Radius von etwa 35 mm. Die Neodym-Magneten in den Rotoren sollen möglichst genau zwischen den beiden Polen der jeweiligen Elektromagneten ([31324](#)) durchlaufen. Deswegen muss diese Elektromagneten-Mitte mindestens 35 mm plus Reserve vom Boden entfernt sein. Die gleiche Höhe muss später auch für die Achs-Mitte der Rotoren umgesetzt werden. Eine Höhe von 37,5 mm kann durch Verwendung eines Bausteins 7,5 sehr einfach umgesetzt werden.

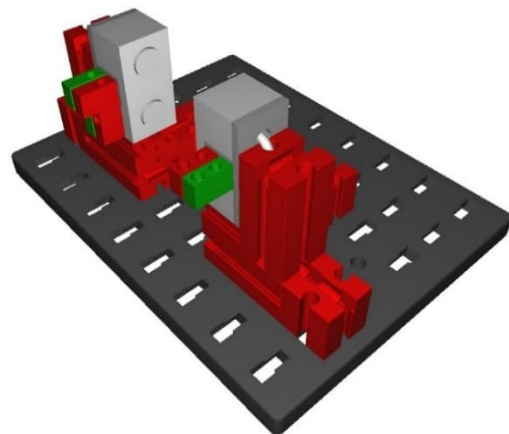


Abb. 3: Mitte der Spule bei 37,5 mm

(Kleiner Fun-Fact: durch die Verwendung des Bausteins 7,5 kann auch der etwas neuere Elektromagnet [32363](#) mit den seitlichen Federn verwendet werden.)

Hier ist der (einzige) Freiheitsgrad engerweiter auf der Längsnut eines Bausteines 30 umgesetzt: Jeder Magnet lässt sich um jeweils 15 mm vor- oder zurück verschieben – somit ergibt sich ein Abstand von 0 bis 30 mm zwischen den Elektromagneten.

Rotorachse

Der Freiheitsgrad rechts-links für die Rotoren-Achslager wird auch mit der (voll-

ständigen) Längsnut eines Bausteines 30 umgesetzt.

Die Lager selbst sind in der Designer-Konstruktion mit Bausteinen 15 mit Bohrung ([32064](#)) realisiert. Das wird auch funktionieren. Es steht aber zu befürchten, dass bei längerer Betriebsdauer diese Art der Lagerung zu Abrieb oder „Fressen“ der Achse führt. Deswegen sind alle Lager im echten Modell jeweils mit einer Schneckenmutter Modul 1 ([35973](#)) und einem Kugellager mit Außendurchmesser 9 mm und Innendurchmesser 4 mm umgesetzt. Nur der Vollständigkeit halber: 4 mm-Achsen kann man auch mit Schneckenmuttern Modul 1,5 ([37925](#)) und Kugellagern mit Außendurchmesser 10 mm lagern. Der fischertechnik-Designer kann das aber (noch?) nicht umsetzen; es gibt keine passenden Einzelteile.

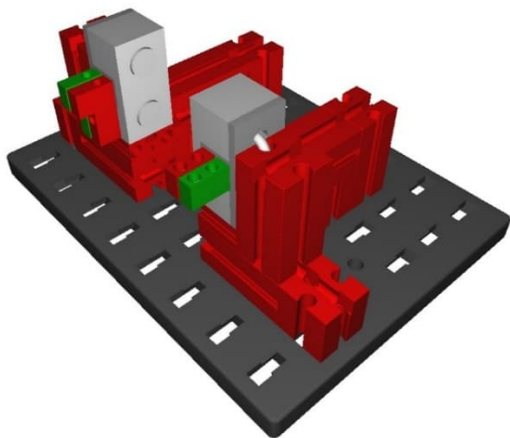


Abb. 4: Halterung der Rotorasche(n)

Für die Beschaffung der Kugellager möchte ich ungerne spezifischere Vorgaben machen, weil es hier sehr viele Lieferanten, Preise und Lieferbedingungen gibt. Bitte achtet bei einer Bestellung auf die genannten Durchmesser und eine Breite von 4 mm.

Die jeweiligen Achslager sind im Modell mit den jeweiligen Rotoren in den unterschiedlichen Polpaarzahlen dargestellt.

Jetzt sind die beiden eingangs gestellten Fragen beantwortet: Ja, es gibt ein einfaches Grundgestell für Rotoren mit Durchmessern von maximal 70 mm und beide (heute

schwer erhältlichen) Sorten Elektromagnet. Und sowohl Elektromagneten als auch Rotoren-Achslager lassen sich nur noch in einem Freiheitsgrad verstellen.

Aus Gründen der Übersichtlichkeit sind keine Kabel eingezeichnet, sondern nur die Stecker, um die richtige Verbindung anzudeuten.

Rotoren mit unterschiedlichen Polpaarzahlen

In den ersten drei Bauphasen haben wir „Grundplatte“, „Elektromagnete“ und „Rotorachsenhalter“ zusammengesetzt. Alle nun folgenden Rotoren können einzeln darauf aufgebaut und betrieben werden. Mehr als ein vollständiger Motor erfordert entsprechend mehr Grundaufbauten. Die Designer-Datei findet ihr im [Download-Bereich zur ft:pedia](#).

Mit der Polpaarzahl 6 aus dem Buch [2] fing es vor 10 Jahren an, und so beginnen wir auch hier. Aus den Bildern sollte der Grundaufbau eines jeden Rotors ersichtlich sein. Der richtige Abstand der Achslager zu den Elektromagneten wurde, wo möglich, mit jeweils gleichen Bausteinen auf jeder Seite „gleich gestaltet“.

Alle Rotorachsen bestehen aus Rastachsen, und jede Rotor-Variante hat am rechten Achsenende einen Rastadapter ([36227](#)). An dessen Ende steckt ein Verbindungsstück 30 ([31061](#)). Das Verbindungsstück soll je Umdrehung mit seiner „langen“ Seite einmal eine Lichtschranke unterbrechen, und steckt deswegen asymmetrisch im Adapter. Am „kurzen“ Ende steckt ein Achsadapter ([31422](#)), um das Achsende wenigstens rudimentär zu wuchten. Das Gebilde bezeichne ich später als „Drehzahlzählfinger“. Bei den Polpaarzahlen 2 und 4 muss das Verbindungsstück ein kleines Stück weiter herausragen, weil der Zählfinger sonst an das Gehäuse stößt. Den benötigten zusätzlichen Abstand erreichen wir mit jeweils einer Klemmhülse ([35980](#)) zwischen Adapter und Verbindungsstück.

Auf weitere spezifische Einzelheiten gehe ich in den Kapiteln zu den Rotoren ein.

Für alle Rotoren gilt auch weiterhin, dass selbstverständlich auf gleichmäßige und sich stets abwechselnde Polung der Neodym-Magnete geachtet werden muss – siehe im Buch auf Seite 218 [2].

Polpaarzahl 6 (500 U/min)

Zur Originalversion im Buch gibt es einige Unterschiede:

- Die Achse ist eine Rastachse 90 an Stelle einer Metallachse
- Hauptziel war im Buch, so einfach wie möglich zu bauen. Man sieht im „neuen“ Modell viel mehr Bausteine und eine größere Grundplatte.

Das Prinzip Einfachheit wurde jedoch nicht vollständig zu Gunsten der variablen Verwendung aufgegeben.

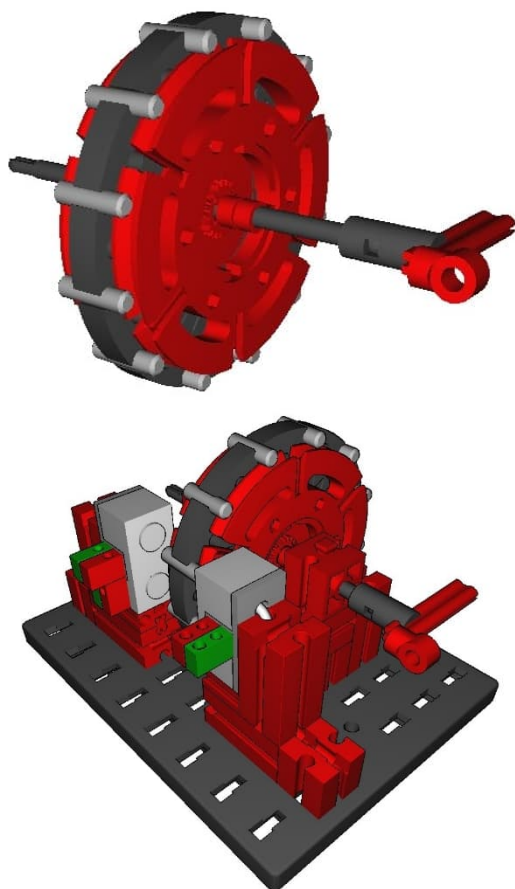


Abb. 5: Synchronmotor mit Polpaarzahl 6

Die Achslager-Bausteine können mit je einem Baustein 15 ([116252](#)) von den Elektromagneten „abgesetzt“ werden. Eine genaue Definition der Achslagerbausteine verhindert Anlauf- und allgemein auch Laufschwierigkeiten.

Polpaarzahl 5 (600 U/min)

Im Vergleich zu meiner ursprünglichen Variante mit Polpaarzahl 5 wird hier nicht mehr die Kette verwendet, und es wackelt auch nichts mehr.

Dazu bedurfte es aber einer kleinen Schummelei. Teilt man nämlich die 360° eines kompletten Rings durch 10 (Polpaarzahl mal 2), dann erhält man einen Winkel von 36° , wahlweise bei Verwendung eines Bausteines 7,5 wie im „alten“ Modell 72° . Beides sind Winkelmaße, die auf so kleinem Raum nicht direkt mit fischertechnik umgesetzt werden können.

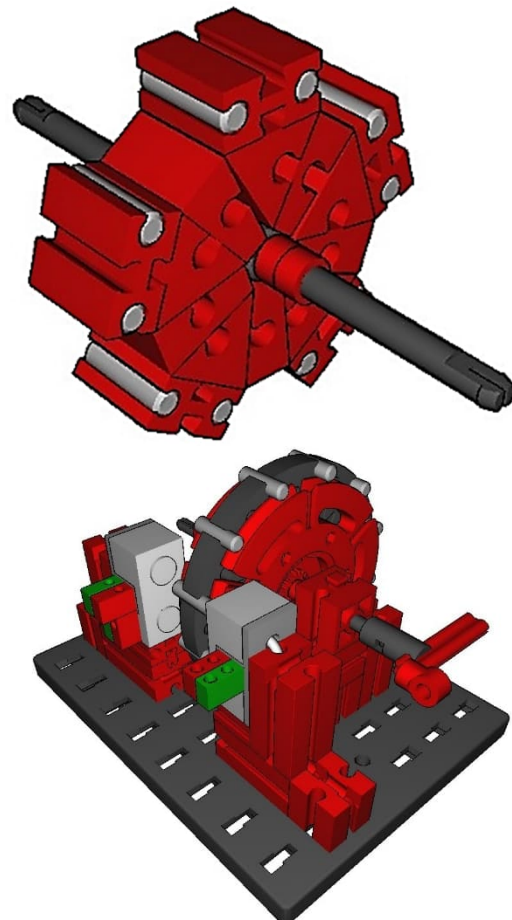


Abb. 6: Synchronmotor mit Polpaarzahl 5

Die Hauptachse wird bei dieser Variante von zwei Rastachsen 45 ([35064](#)) gebildet, die in der Mitte mit einer Rastkupplung ([35073](#)) zusammengehalten werden. Um diese Kupplung wird ein Ring aus fünf Paaren aus je einem Winkelstein 60° ([31010](#)) und einem Winkelstein 15° „aufgequetscht“. Rechnerisch ergibt so ein Ring also 375°, oder 3° je Segment. Das ist eine Abweichung, die die fischertechnikeile ohne bleibende Schäden zulassen.

Bei den Winkelsteinen 60° muss ein Zapfen immer von der „Radmitte“ wegweisen. Auf den Zapfen kommt dann ein Baustein 7,5 ([37468](#)), in dessen Nuten wiederum die Neodym-Magneten gesteckt werden.

Bei dieser Variante definiert man den Abstand der Rotorachse zu den Elektromagneten mit je einem Baustein 5 zwischen dem „Lagerbaustein“ und dem Elektromagneten.

Polpaarzahl 4 (750 U/min)

Dieser Rotor ist mal einfach zu bauen, ohne Tricks und Kniffe. Die Hauptachse setzt sich aus zwei Rastachsen mit Platte ([130593](#)) mit „Innenleben“ in 15×15 mm zusammen:

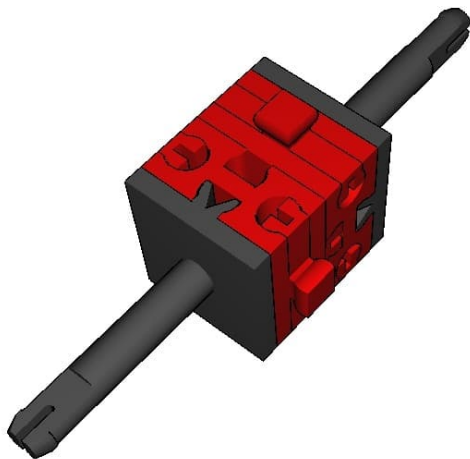


Abb. 7: „Innenleben“ des Rotors für die Polpaarzahl 4

Das „Innenleben“ besteht zwischen den Rastachsen mit Platte aus zwei gegeneinander um 90° gedrehten Bausteinen 7,5 ([37468](#)) und fünf Federnocken ([31982](#)).

Auf die vier Zapfen der Federnocken werden nun je ein Baustein 5 ([37237](#)) und als „Neodym-Magnet-Halter“ wieder ein Baustein 7,5 geschoben, und zwar so, dass das „Innenleben“ nicht mehr sichtbar ist. Die Nuten der Bausteine 5 und 7,5 und anschließend die Magnete müssen parallel zur Hauptachsrichtung ausgerichtet sein.

Die Achslagersteine sollten bis ganz an die Elektromagneten herangeschoben werden.

Polpaarzahl 3 (1000 U/min)

Auch dieser Rotor benötigt etwas Fingerspitzengefühl, auch wenn die Klemmkraft hier nicht so groß sind. Aus sechs Paaren Baustein 7,5 und Winkelbaustein 60° wird ein Ring zusammengesetzt. Die jeweils obere Nut des Bausteins 7,5 trägt den Neodym-Magneten, die untere Nut ein Verbindungsstück 15 ([31060](#)).

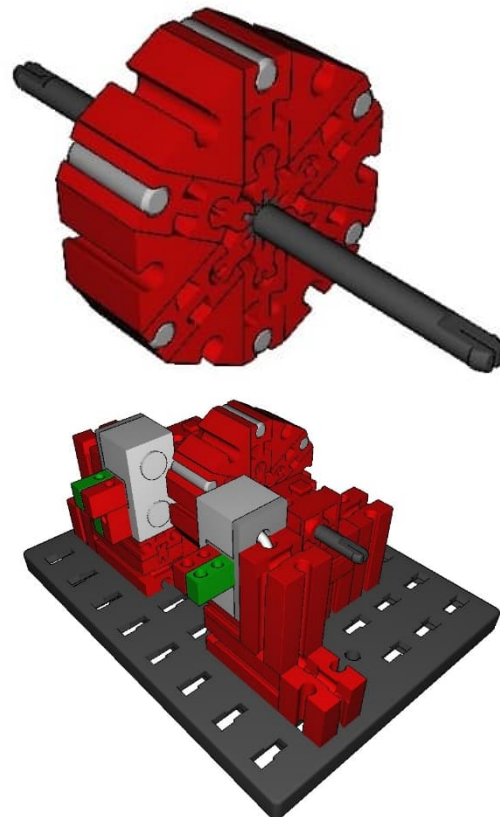


Abb. 8: Synchronmotor mit Polpaar 3

Die Hauptachse wird aus zwei Rastachsen 45 ([35064](#)) und einer Rastkupplung ([35073](#))

gebildet. Der Ring wird vorsichtig auf die Kupplung geschoben.

Hier kann es hilfreich sein, zwei gegenüberliegende Verbindungsstücke zuerst wegzulassen und zuletzt einzuschieben – so hat es bei mir jedenfalls prima funktioniert.

Polpaarzahl 2 (1500 U/min)

Dieser Rotor ist fast genauso aufgebaut wie der Rotor mit Polpaarzahl 4. Nur lassen wir hier den Baustein 5 weg, und lassen jeweils die gleiche Nut an den vier Bausteinen 7,5 „leer“. Polpaarzahl 2 darf schließlich nicht mehr als $2 \times 2 = 4$ Magnetpole besitzen.

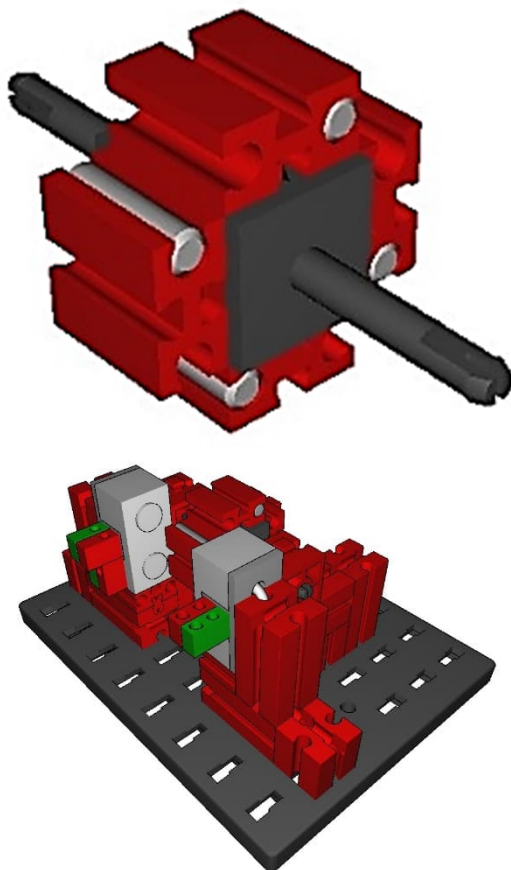


Abb. 9: Synchronmotor mit Polpaarzahl 2

Auch hier sollten die Achslagersteine bis ganz an die Elektromagneten heran geschoben werden.

Polpaarzahl 1 (3000 U/min)

Diese Polpaarzahl hat mir über mehrere Jahre doch so einiges Kopfzerbrechen bereitet. Das lag an zwei Fehlern:

- Mit der Idee, dass die Magneten immer „von oben nach unten“ (oder eben anders herum) durch das Magnetfeld wandern, wollte ich buchstäblich mit dem Kopf durch die Wand.
- Eine der wirksamsten Methoden beim Bau von Modellen ist es, alles Überflüssige wegzulassen, und mit so wenig Bauteilen wie möglich auszukommen.

Der erste Fehler führte zu „klobigen“ Konstruktionen, die wahrscheinlich aufgrund ihres Verhältnisses „Massenträgheit zu Magnetfeldstärke“ gar nicht ins Laufen kommen konnten. Der Rotordurchmesser war in einigen Versuchen größer als der von Polpaarzahl 4.

Diese Wand hat Rüdiger Riedel gründlichst (und auch schon früh – wer lesen kann ist klar im Vorteil) eingerissen. Und er hat sogar zwei Ansätze geliefert.

- In seinem ersten Ansatz hat er konsequent die Massenträgheit verringert [3].
- Sein zweiter Ansatz geht noch einen Schritt weiter: Durch seitliche Verlagerung der Drehachse liefert er einen „Einfachst-Synchronmotor“ mit Polpaarzahl 1 – aber nur einem Magneten. Weniger geht nicht [4].

Auf der Modellausstellung in Münster hat Rüdiger im Januar 2025 seinen Lavet-Motor ausgestellt. Aber erst mit der Idee, dass sich die beiden Magneten bei Polpaarzahl 1 direkt im Magnetfeld zwischen zwei gegensinnig, also nicht parallel geschaltet gepolten Elektromagneten drehen, ist (endlich 😊) auch bei mir der Knoten geplatzt.

Der Grundaufbau besteht aus der Hauptachse mit einer Rastachse mit Platte ([130593](#)), einer Rastkupplung ([35073](#)), einer Rastachse 45 und dem „Drehzahlzähl-

finger“ wie bei den anderen Polpaarzahlen. Wie bei allen anderen Rotoren sind die Achslagerbausteine als Bausteinen 15 mit Bohrung ([32064](#)) dargestellt – bitte gedanklich wie bei allen anderen Rotoren auch durch eine der beiden oben vorgestellten Kugellager ersetzen.

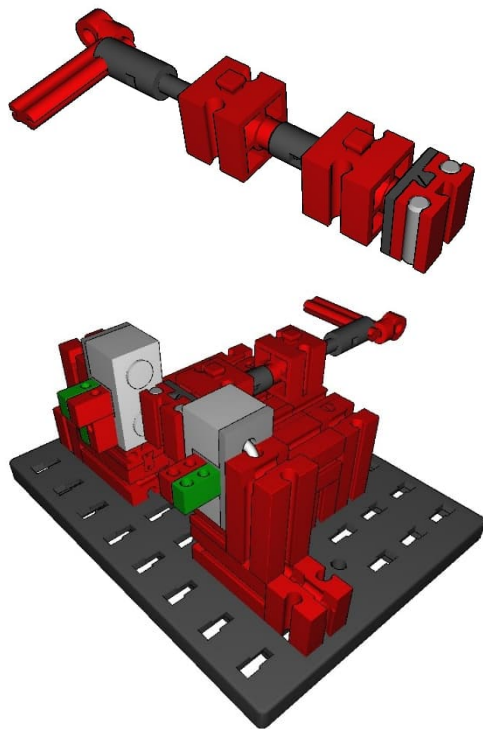


Abb. 10: Synchronmotor mit Polpaarzahl 1

Manöverkritik und Hinweise

Trotz gründlichen Überlegens und vielen Versuchen – *nobody (and nothing) is perfect*. Ein paar Gedanken dazu:

- Bei den Rotorachsen habe ich häufig Baustein-Paare verwendet, um den Abstand der Lager zu den Elektromagneten klar und gleichartig zu definieren.

Eine solche „Abstandsdefinition“ ist natürlich auch bei den Spulen möglich. Denkbar wären je ein Baustein 5 (sehr eng, nur 2,5 mm auf beiden Seiten „Luft“) oder je eine Bauplatte rot ([38246](#)). Bauplatten sind 2 mm dick, es blieben etwa 5,5 mm auf jeder Seite.

Je genauer man die Rotoren baut, desto dichter können die Elektromotoren an den

Rotor „heranrücken“, was zu besserer Lauf-ruhe und vermutlich zu einem höheren Drehmoment führt.

Die Lage des Rotors auf der Rotorachse (vorne-mittig-hinten oder links-mittig-rechts, je nach Betrachtungsweise) kann mit 2 + 2 Abstandsringen 4 mm ([31597](#)) oder in Kombination mit Riegelscheiben ([36334](#)) eingestellt werden. Anders als im fischertechnik-Designer dargestellt bildet ein eingesetztes Kugellager nicht die seitliche Oberfläche; es ist leicht im Loch des Bausteins 15 mit Bohrung ([32064](#)) „versenkt“.

Zur Erinnerung: die Elektromagneten sind für die Polpaarzahlen 2 bis 6 parallel, also „gleichsinnig“ geschaltet, für Polpaarzahl 1 aber „gegensinnig“.

Kleine Eselsbrücke: in der Konstruktion mit dem fischertechnik-Designer bilden die Stecker die gleichsinnige, parallele Schaltung ab: Die beiden roten Stecker liegen einander gegenüber (innen), die beiden grünen Stecker folglich außen.

Die Lichtschranke ist im fischertechnik-Designer-Modell nicht enthalten.

Nun noch ein paar Hinweise für den Fall, dass die Synchronmotoren nicht auf Anhieb „loslaufen“:

- Auch wenn es beim Betrieb mit Wechselstrom seltsam erscheint, aber die Polung der Elektromagnete im Verhältnis zueinander ist sehr wichtig (siehe oben).
- Moderne fischertechnik-Netzteile liefern einen Gleichstrom oder einen gepulsten Gleichstrom. Der Motor, egal welcher Polpaarzahl, wird damit nicht laufen. Bitte die „alten“ fischertechnik-Netzteile 812 oder 814 verwenden.
- Theoretisch sind auch alte „Eisenbahntrafos“ verwendbar. Hier besteht aber eine recht hohe Gefahr, die Elektromagneten von fischertechnik durch zu hohe Spannungen zu zerstören. Trafos von digitalen Spielzeugeisenbahnen sind

generell nicht verwendbar; sie liefern ebenfalls gepulste Gleichspannungen.

- Die Polung der Neodym-Magnete ist ebenso wichtig. Faustregeln:
 - Die Anzahl dieser Magneten ist immer gerade (2, 4, 6, ... 12).
 - Die Pole an der „Frontseite“ (also eine der Seiten, rechts oder links) wechseln sich immer ab. Das kann man mit einem Neodym-Magnet sehr einfach überprüfen.
- Läuft der Rotor ruhig und (fast) ohne Geräusch? Wenn es rattert, klemmt, oder steckenbleibt, bitte gängig machen. Ein leichtes Hakeln beim Passieren der Neodym-Magneten ist normal.

Drehzahlmessung

Oben wurde der „Drehzahlzählfinger“ schon erwähnt. Eine dazu passende Lichtschranke kann beliebig aufgebaut werden. Um die Schranke auch bei laufendem Motor flexibel einsetzen zu können, habe ich einen Halter aus einer Bauplatte 15×30 ([35049](#)), einem Federnocken, einer Bauplatte 15×30 mit drei Nuten ([38428](#)) und einer weiteren 35049 aufgebaut. So kann ich innerhalb 15×15 mm wahlweise 30 mm Weite Lichtschranke an den Finger heranbringen.

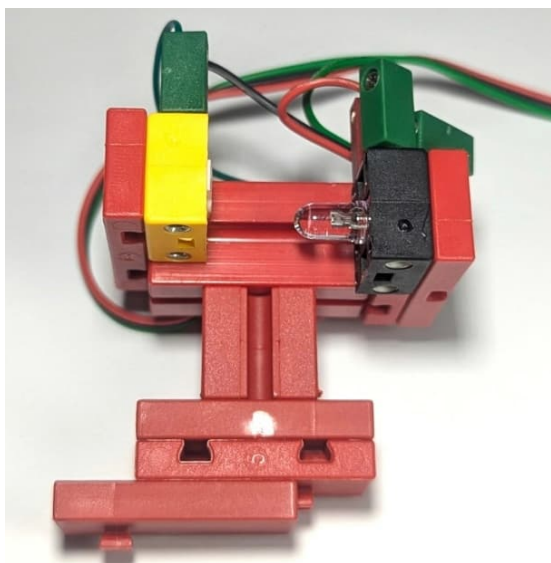


Abb. 11: Halter für Zähl-Lichtschranke

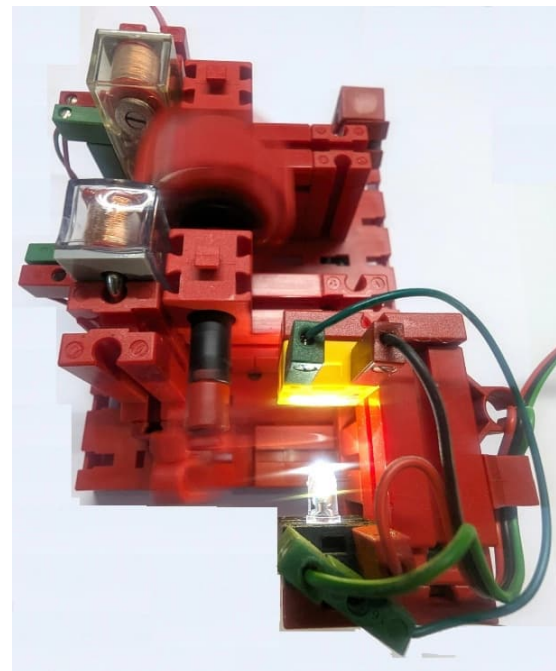


Abb. 12: Variante Polpaarzahl 2 mit Zählung

Im TX-Programm gab es ursprünglich drei Lichtschranken für die drei Polpaarzahlen. Der TX kann höchstens vier Lichtschranken an je einem Zähler bearbeiten; für die Polpaarzahl 6 fehlen also zwei „Zählmöglichkeiten“. Daher habe ich aus der Not eine Tugend gemacht und nur eine Lichtschranke und einen Zähler verwendet. Die Lichtschranke wird zum Betrieb an einem der sechs Varianten angebaut.

Das ROBOPRO-Programm (siehe Abb. 13) sollte ursprünglich für 10 Sekunden zählen und das Ergebnis auf der TX-Anzeige darstellen. Nun sind 10 Sekunden relativ ungenau: Für jede Periode konnte ich einen Fehler von etwa minus einem Prozent feststellen. Diesen Fehler habe ich recht pauschal glattgebügelt, indem ich die Zeitspanne auf 10,09 Sekunden eingestellt habe. Zugegeben, das ist keine exakte, aber eine schnell funktionierende Lösung.

Zuletzt wird die Polpaarzahl aus den gezählten Durchläufen errechnet und angezeigt (Abb. 14).

Das TX-Programm ist ebenfalls im [Download-Bereich der ft:pedia](#) zu finden.

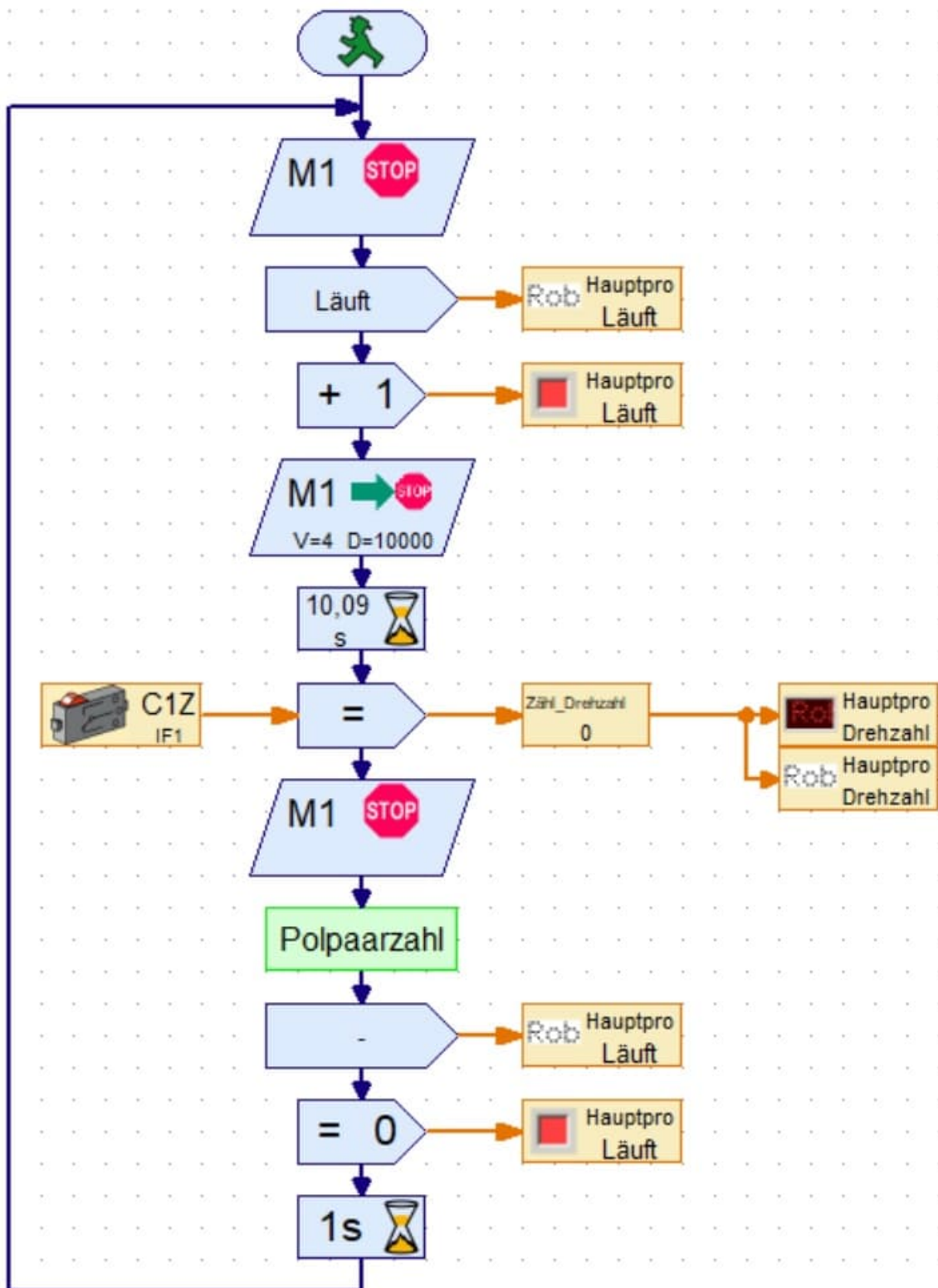


Abb. 13: ROBOPRO-Programm zur Drehzahlmessung

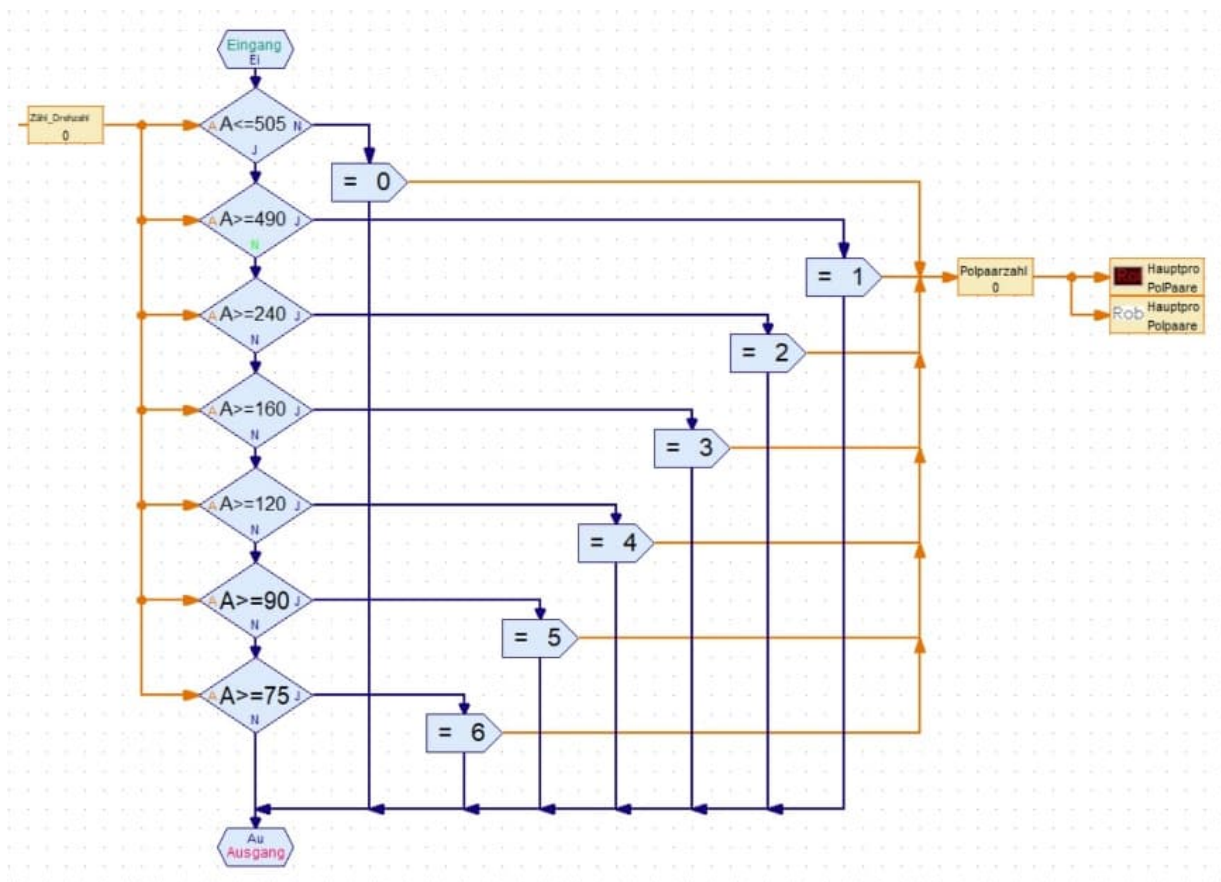


Abb. 14: Errechnung der Zahl der Polpaare

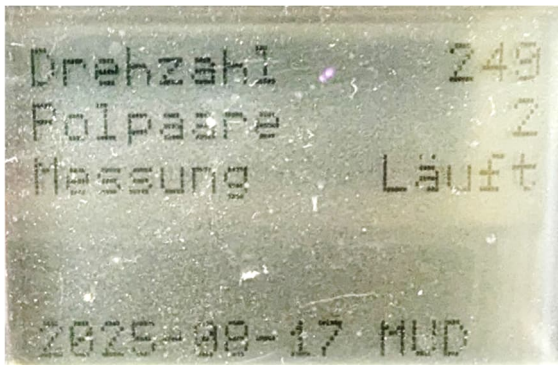


Abb. 15: TX-Ausgabe

Quellen

- [1] Matthias Dettmer: *Synchronmotoren*. [ft:pedia 2/2016](#), S. 48–52.
- [2] Dirk Fox, Thomas Püttmann: *Technikgeschichte mit fischertechnik*. dpunkt Verlag, 2015.
- [3] Rüdiger Riedel: *Der etwas andere Motor*. [ft:pedia 3/2016](#), S. 22–25.
- [4] Rüdiger Riedel: *Elektrische Uhren*. [ft:pedia 4/2017](#), S. 24–33.

Tipps & Tricks

Verbindungsstecker für Silberlinge selbst herstellen

Arnoud van Delden

Wer viele fischertechnik Silberlinge besitzt oder diese Elektronikmodule in 3D-gedruckten Gehäusen mit dem gleichen Formfaktor nachbaut, kommt irgendwann an den Punkt, dass alle verfügbaren Original-Baustein-Verbindungsstecker in Gebrauch sind. Über deren metallische Leiterstreifen lassen sich alle Module mit der Versorgungsspannung verbinden. Die 1971 eingeführten Verbindungsstecker haben die Teilenummer [36380](#) und werden immer seltener, da sie seit den 80er Jahren nicht mehr hergestellt werden. Glücklicherweise habe ich kürzlich eine Möglichkeit gefunden, um diese Stecker selbst herzustellen (oder nachzubilden).

Einführung

Auf meiner Website biete ich Repliken der originalen Silberlinge an – sowohl komplett gebaut und getestet als auch als Bausatz [1]. Zu jedem Exemplar gehört ein 3D-gedrucktes Gehäuse, das den klassischen Silberling-Gehäusen entspricht. Meine Kunden und ich haben jedoch festgestellt, dass die originalen Verbindungsstecker immer schwieriger zu bekommen sind. Aus diesem Grund suche ich seit einiger Zeit nach einer Möglichkeit, diese Verbindungsstecker selbst herzustellen.

Meine ersten Versuche bestanden darin, Metallstreifen in einen kleinen 3D-gedruckten Kunststoffbalken einzuschmelzen. Obwohl ich dafür ein spezielles hitzebeständiges Werkzeug angefertigt hatte, blieb dies eine sehr mühsame Angelegenheit.

Außerdem gelang es mir nicht, die Metallstreifen gerade und fest genug in dem nur 4,6 mm hohen Kunststoffbalken zu verankern. Selbst wenn ich diesen aus zwei zu verklebenden Teilen druckte, gelang es mir nicht. Experimente mit doppelt gefalteten Metallstreifen (für mehr Festigkeit und

Dicke) führten leider auch nicht zu einer einfachen Lösung.

Erst als ich einen nicht leitenden Isolierverbinder herstellte (der graue Clip rechts in Abb. 3), kam mir die Idee für eine viel einfachere Herstellungsmethode. Damit habe ich inzwischen schon viele brauchbare Verbindungsstecker nachbauen können.

In Abb. 1 sind die beiden kleinen 3D-gedruckten Teile zu sehen. Bevor diese zusammengedrückt werden können, werden zwei 3 mm breite Metallstreifen benötigt.



Abb. 1: Alle benötigten Teile

Diese Metallstreifen dürfen nicht zu dick sein; ich verwende eine Dicke von 0,1 mm, die leicht online erhältlich ist. Zunächst falte ich etwa 4 mm eines 6,5 cm langen Streifens um die Oberseite der Verbindungsstange. Dieser Streifen muss parallel zum gedruckten Stützstreifen verlaufen. Dann biege ich den Metallstreifen um die Unterseite und dann auf der Rückseite wieder nach oben. Das kleine verbleibende Ende biege ich um die Oberseite zurück nach vorne. Das wiederholt man für beide „Beine“ der Klammer.

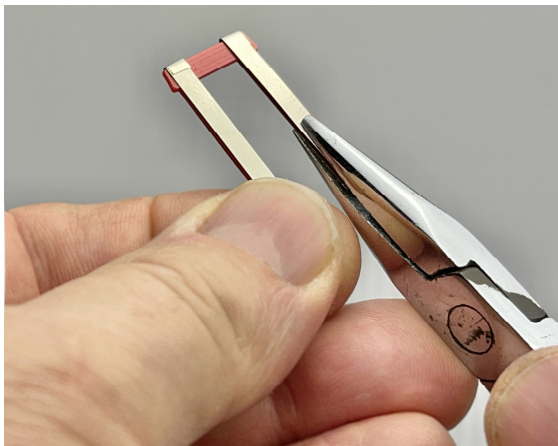


Abb. 2: Flachdrücken der Metallstreifen

Anschließend drücke ich die Metallstreifen an den Biegungspunkten mit einer Zange gut flach (Abb. 2) und schiebe diese „Gabel“ in den Schlitz des „Balkens“. Da sich an jeder Seite oben zwei Lagen Metallstreifen befinden, klemmen die Metallstreifen gut in den Aussparungen im Inneren des „Balkens“. Bisher habe ich noch nie Klebstoff benötigt, um ihn fest zu verankern.

Abb. 3 zeigt das Ergebnis. Wenn sich der Verbindungsstecker zu leicht zwischen den Modulen verschieben lässt und nicht richtig sitzt, empfiehlt es sich, beide Streifen an zwei Stellen mit einer Zange leicht zu biegen; damit wird der Verbindungsstecker dem Original ähnlicher und der elektrische Kontakt wird stabiler.

Die Druckdateien des Verbindungssteckers können bei Printables heruntergeladen wer-

den [2]. Ich empfehle, die Bauteile mit einer möglichst geringen Lagendicke zu drucken. Die Druckzeit spielt keine Rolle, und es erhöht die Stabilität des hohlen „Balkens“.

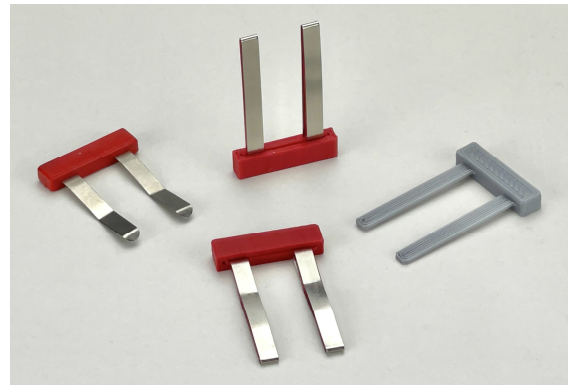


Abb. 3: Original-Verbindungsstecker und selbst gefertigte, gerade und mit gebogenen Streifen. Ganz rechts ein nicht leitender Isolierverbinder

Der nicht leitende Isolierverbinder besteht nur aus einem Teil [3]. Er ist praktisch, um bei Schaltungen, die aus mehreren Reihen Silberlingen bestehen, die leitenden Verbindungsstecker einzusparen, da nicht jede Verbindung zwischen zwei Modulen elektrisch hergestellt werden muss. Der Verbinder kann auch nützlich sein, wenn eine Reihe (oder Gruppe) von Silberlingen über eine getrennte Stromversorgung verfügt.

Ich verwende für diesen Isolierverbinder eine andere Farbe (grau). Wer Reihen von Silberlingen mit Strom versorgen, aber nur den Minuspol verbinden möchte, kann natürlich auch ganz einfach einen Verbindungsstecker mit einem leitenden und einem isolierenden Streifen herstellen, indem er nur ein „Bein“ mit einem Metallstreifen ummantelt.

Quellen

- [1] WhizzBizz.com: [Bauanleitungen für Silberlinge](#)
- [2] Printables: [Druckdateien der Verbindungsstecker](#)
- [3] Printables: [Druckdateien des Isolierverbinders](#).

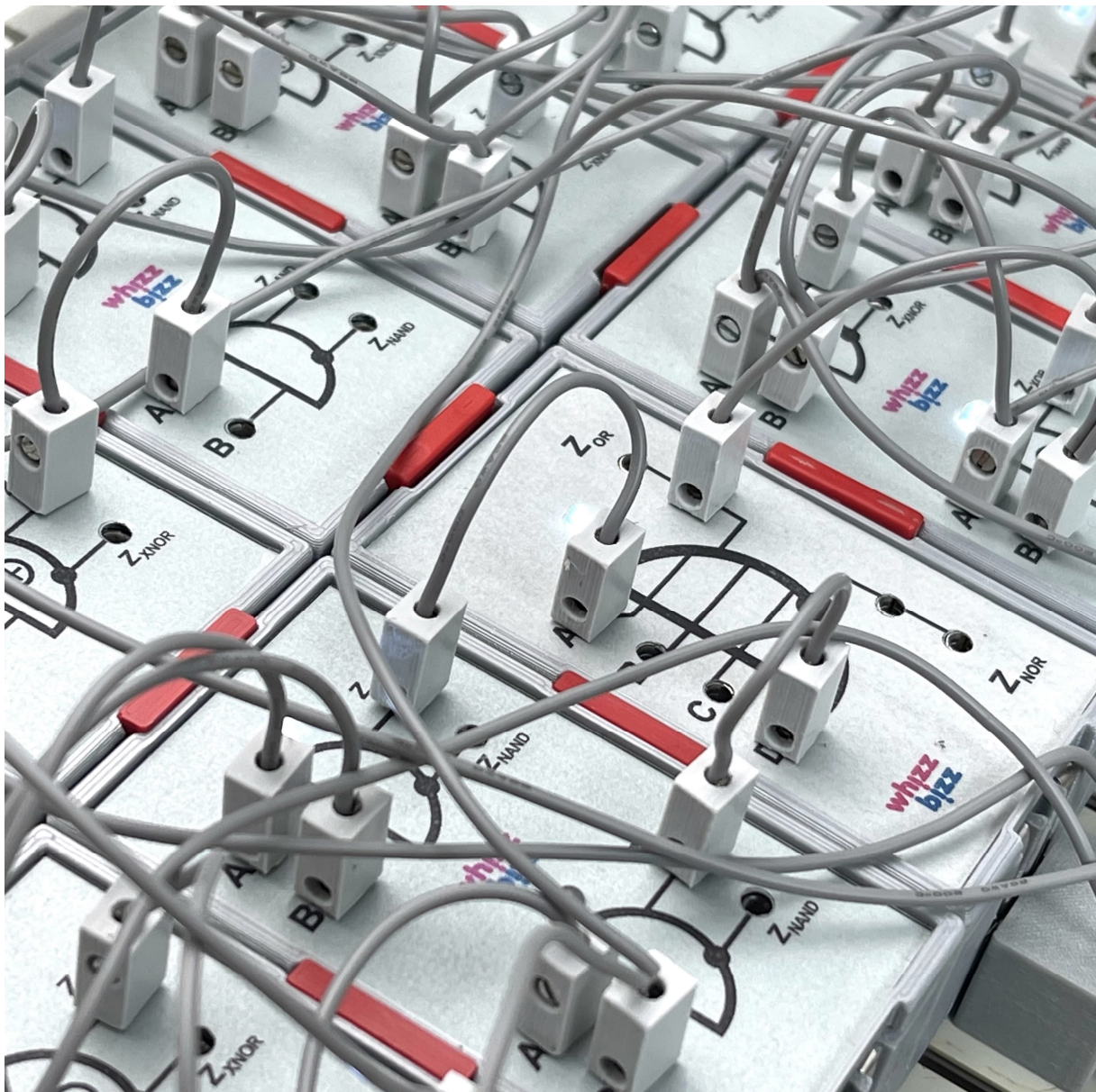


Abb. 4: Suchbild – welche Verbindungsstecker sind original und welche Nachbauten?

Elektronik

Es kracht

Peter Krijnen, Arnoud van Delden

Für mein Orgel-Projekt [1] benötigte ich mehrere Tongeneratoren. Zu diesem Zweck habe ich selbst zwölf Grundbausteine nachgebaut, alles wie gewohnt verdrahtet und auf die richtige Frequenz eingestellt. Leider klangen manche Frequenzen nicht ganz klar. Die Töne schienen zu vibrieren. Die Ursache? Unbekannt. Die Lösung? Die erfahrt ihr in diesem Beitrag.

Nachdem ich mein Projekt mechanisch aufgebaut und die gesamte Elektronik verdrahtet hatte, konnte ich die Generatoren auf die gewünschten Frequenzen einstellen. Eigentlich hatte ich das schon beim Bau gemacht, aber leider eine Oktave zu hoch. Also musste ich alles neu justieren. Dazu benutzte ich den Frequenzmesser meines Multimeters.

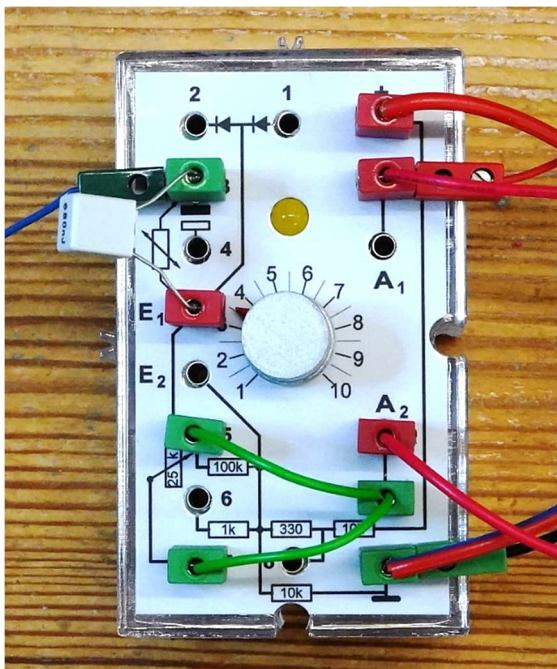


Abb. 1: Der übliche Aufbau eines Tonoszillators. Aufgrund der benötigten Frequenzen zwischen 500 und 1100 Hz habe ich einen 680-nF-Kondensator verwendet.

Ich schloss ihn an A1 an und stellte das Potentiometer am Grundbaustein so lange ein, bis ich 880 Hz auf dem Messgerät sah. Das ging nicht besonders gut. Die Frequenz schwankte stark. Diese Schwankung war kaum hörbar, es sei denn, man hatte das absolute Gehör. Die zuvor beschriebene hochfrequente Vibration war jedoch deutlicher hörbar. Beim Anschluss an A2 zeigte das Frequenzmessgerät jedoch eine Frequenz von 1760 Hz an.

Zuerst dachte ich, mit dem verwendeten Grundbaustein sei etwas nicht in Ordnung. Aber das gleiche Problem war auch bei allen anderen Grundbausteinen hörbar. Was sollte ich tun?

Also schloss ich das Oszilloskop an: A1 an Kanal 1 (gelb) und A2 an Kanal 2 (blau). Die horizontale Skala beträgt 1 ms/div.

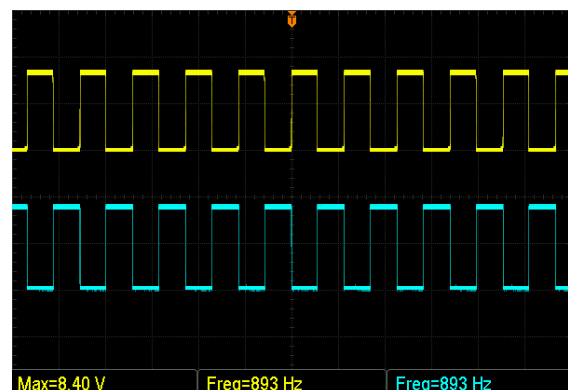


Abb. 2: Bei einer horizontalen Skala von 1 ms/div ist nicht viel zu sehen. Beide Kanäle messen 893 Hz.

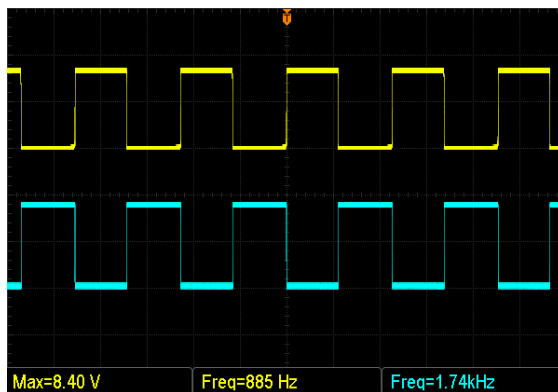


Abb. 4: Bei 500 µs/div sehen wir auch nichts Ungewöhnliches. Allerdings werden jetzt 1,74 kHz an A2 gemessen.

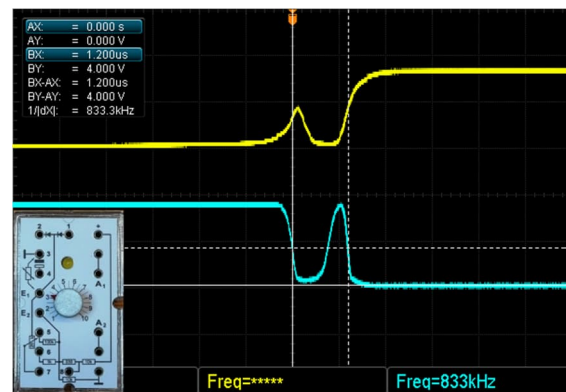


Abb. 3: Da ist der Übeltäter bzw. das, was der Übeltäter verursacht hat

Abb. 2 zeigt beide Signale. Für beide Kanäle werden 893 Hz gemessen. Es ist nichts zu erkennen, was auf eine mögliche Ursache für die vibrierenden Töne hindeutet. Bei 500 µs/div sehen wir auch nichts Ungewöhnliches. Allerdings werden jetzt 1,74 kHz an A2 gemessen. (Abb.3)

Als ich die Skala auf 1 µs/div einstellte, war sofort klar, dass etwas nicht stimmte (Abb. 4).

Es ist deutlich zu erkennen, dass die steigende Flanke an A1 beeinflusst wird. Die Ursache dafür ist mir nicht ganz klar, aber sobald ich das Potentiometer über 1 KΩ drehe, beginnen die Störungen. Es

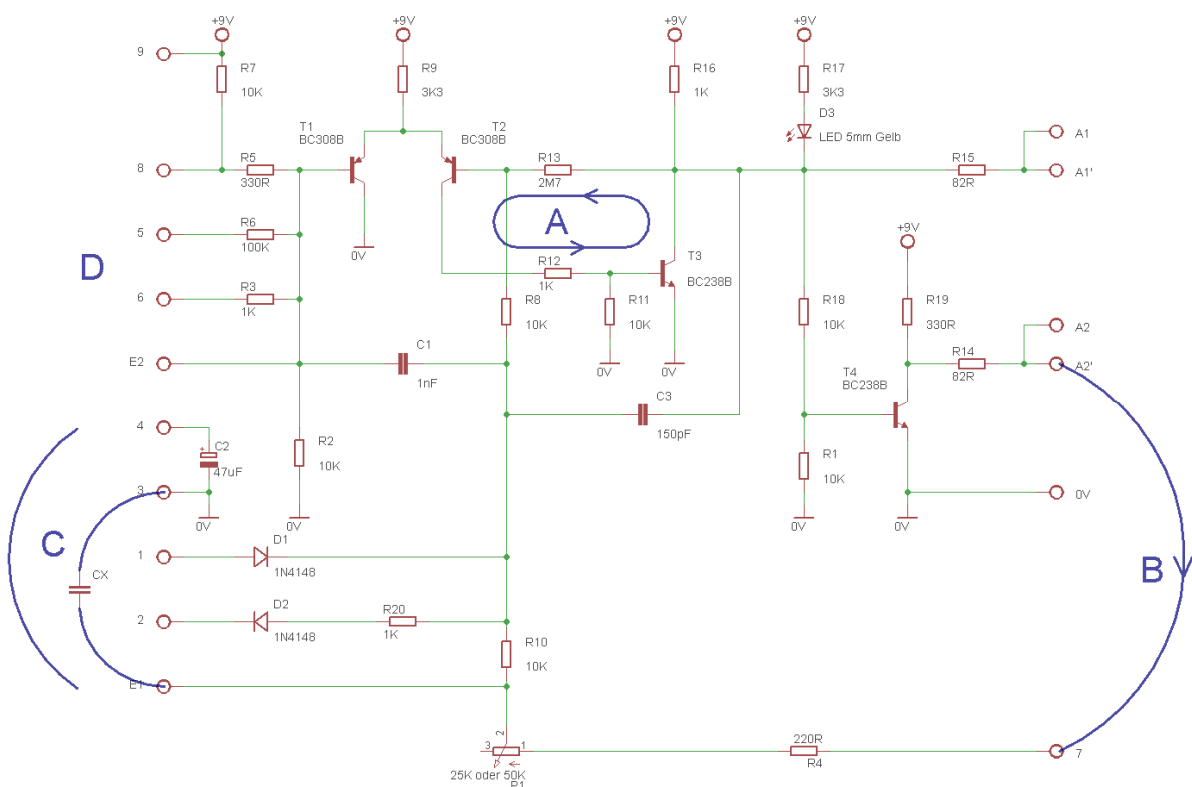


Abb. 5: Der Schaltplan meines Grundbaustein-Nachbaus

scheint, als würde das Potentiometer dann zu einem Saugkreis, der die steigende Flanke beeinflusst.

Anhand des Schaltplans in Abb. 5 erklären wir nun, was passiert, wenn wir den Grundbaustein als Oszillator verwenden.

A. Sobald die Versorgungsspannung an den Grundbaustein angelegt wird, leuchtet die Lampe (oder LED). Dies bedeutet, dass der NPN-Transistor T3 geöffnet ist, also leitet. Der Spannungspegel an A1 ist daher niedrig. Der PNP-Transistor T2 ist geöffnet, wodurch wiederum sichergestellt wird, dass T3 geöffnet ist. Dieser Zustand bleibt bestehen, bis sich der Spannungspegel an der Basis von T2 ändert. Wir können die Basis von T2 als negativen Eingang eines Differenzverstärkers (mit Rückkopplung) betrachten, der aus T1 und T2 besteht. Ohne externe Verbindungen zu den Sammelschienen 5, 6, 8 oder E2 hat der positive Eingang des Differenzverstärkers (die Basis von T1) einen festen Wert von etwa der Hälfte der Versorgungsspannung. Dieser Wert wird durch den Spannungsteiler aus R7 + R5 und R2 bestimmt.

B. Sobald wir A2 mit 7 verbinden, ändert sich die Basisspannung von T2. Da der Pegel an A2 hoch ist, steigt die Spannung an der Basis von T2 über das 25-k Ω -Potentiometer und die Widerstände R4, R10 und R8. Sobald sich die Basisspannung von T2 der Emitterspannung annähert, schließt T2 (schaltet ab). Dadurch wird auch T3 abgeschaltet, was zu einem hohen Ausgangsspannungsanstieg an A1, dem Erlöschen der Lampe bzw. LED und dem Abfallen des Pegels an A2 führt. Da A2 mit Bus 7 verbunden ist, öffnet die Basis von T2 nun wieder. T3 öffnet sich daraufhin erneut, und die Lampe bzw. LED leuchtet wieder auf. Dieser Vorgang wiederholt sich fortlaufend.

Dank der Rückkopplung arbeitet die Schaltung nun als freilaufender Oszillator mit hoher Frequenz. Diese kann zwar mit dem Potentiometer beeinflusst werden, wird aber hauptsächlich durch die maximalen Anstiegs- und Abfallzeiten der Schalttransistoren bestimmt. Abgesehen von den (üblicherweise unerwünschten) parasitären Kapazitäten, die durch die Verdrahtung und die Leiterbahnen auf der Leiterplatte entstehen, ist in der Rückkopplung noch keine frequenzbestimmende Kapazität vorhanden.

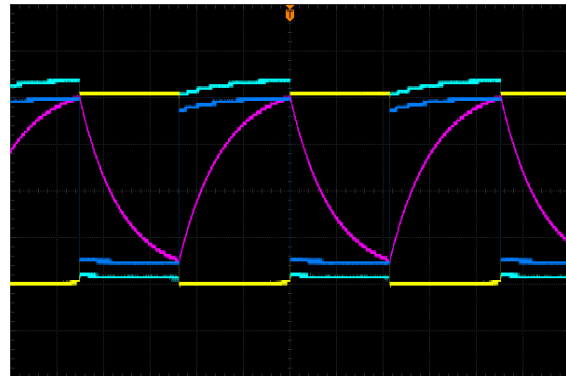


Abb. 6: A2 – 6: 1,11 Hz Lade- und Entladekurve des 47- μ F-Kondensators an E1 (lila).
Gelb = A1, Cyan = A2, Blau = E2

C. Wenn wir beispielsweise den 47- μ F-Kondensator von Bus 4 mit Bus E1 verbinden, lädt er sich auf, solange A2 auf hohem Pegel liegt (Abb. 6). Dieser Ladevorgang und der damit verbundene Spannungsanstieg nehmen zu, bis die an der Basis von T1 angelegte Spannung erreicht ist und der Differenzverstärker aus T1 und T2 umschaltet. A2 liegt dann wieder auf niedrigem Pegel und der Kondensator entlädt sich. Wird das Potentiometer ganz nach links gedreht (0 Ω), erfolgt der Ladevorgang schneller als bei ganz nach rechts gedrehtem Potentiometer (25 k Ω). Die Frequenz ist nun einfach einstellbar und somit umgekehrt proportional zum eingestellten Wert von P1.

D. Durch den Anschluss von A2 an Bus 5 (100 k Ω , Abb. 7) oder 6 (1 k Ω , Abb. 6)

lässt sich die Frequenz deutlich reduzieren. Dadurch entfällt der Vergleich mit einem festen Spannungspegel an der Basis von T1. Der positive Eingang E2 des Differenzverstärkers (Basis von T1), der zum Vergleich der variierenden Spannung am negativen Eingang (Basis von T2) dient, wird nun durch einen von R3 oder R6 bestimmten Teiler geteilt. Dies führt zu einem größeren Abstand zwischen den Schwingungsfrequenzen und somit zu einer geringeren Frequenz des Oszillators. Der Wert des Potentiometers und der Wert des angeschlossenen Kondensators bestimmen die Frequenz dieses sogenannten astabilen Multivibrators.

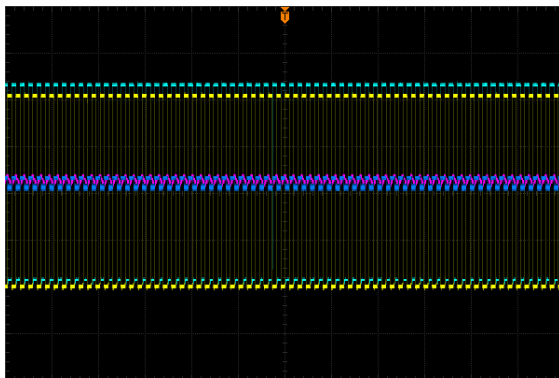


Abb. 7: A2 - 5: 29,1 Hz Lade- und Entladekurve des 47- μ F-Kondensators an E1

Beispielsweise entlädt sich ein 1-nF-Kondensator, der an E1 angeschlossen ist, deutlich schneller als beispielsweise ein 1000- μ F-Kondensator. Die Frequenz, die sich aus dem 1-nF-Kondensator ergibt, liegt bei etwa 128 kHz, während die Frequenz des 1000- μ F-Kondensators (selbst bei einer Verbindung zwischen den Pins A2 und 6 anstelle von A2 und 5) so niedrig ist, dass sie mit einem Oszilloskop nicht mehr messbar ist. Die Schaltung fungiert dann eher als Timer denn als Oszillator.

Operation Kondensator

Ich hatte bereits festgestellt, dass der originale Grundbaustein eine Aussparung für einen möglichen Kondensator (C3) hatte. Zumindest nahm ich das an. Ich hatte

damals einen 1-nF-Kondensator an dieser Stelle platziert. Ob dies die Funktion verbessert hätte, war mir damals allerdings nicht klar.

Ihr werdet verstehen, dass ich noch einmal untersuchen wollte, ob dieser mögliche Kondensator zumindest die Lösung sein könnte. Also öffnete ich einen der Grundbausteine, bohrte zwei Löcher in die Leiterplatte und lötete einen 1-nF-Kondensator darauf. Das Ergebnis ist in Abb. 8 zu sehen.

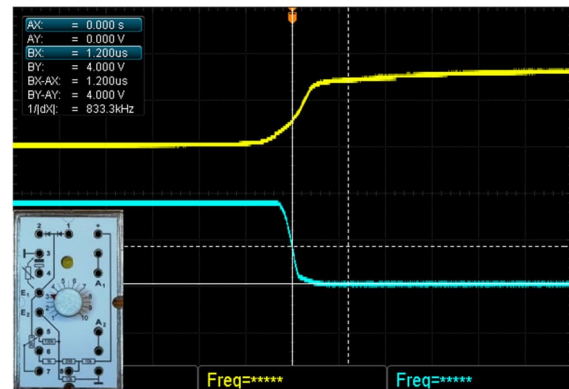


Abb. 8: Das Ergebnis der Operation ist deutlich sichtbar

Aber was ist mit den originalen Grundbausteinen? Und all meinen anderen selbstgebauten Modulen? Haben die auch diesen „Fehler“?

Neben den drei Originalen habe ich über 50 Nachbauten im Schrank. Ich habe sie alle überprüft.

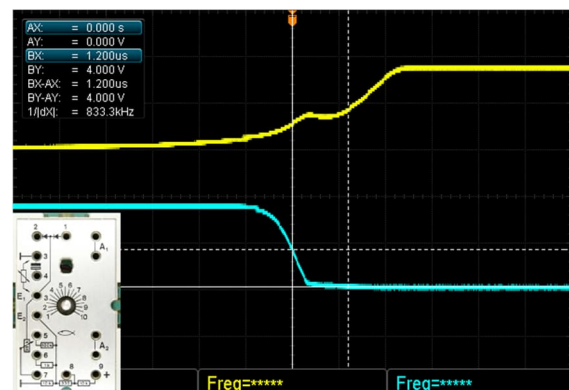


Abb. 9: Obwohl deutlich weniger gravierend, ist der Fehler dennoch vorhanden

Abb. 9 zeigt ein Original mit einem 25-k Ω -Potentiometer. Diese Version weist eben-

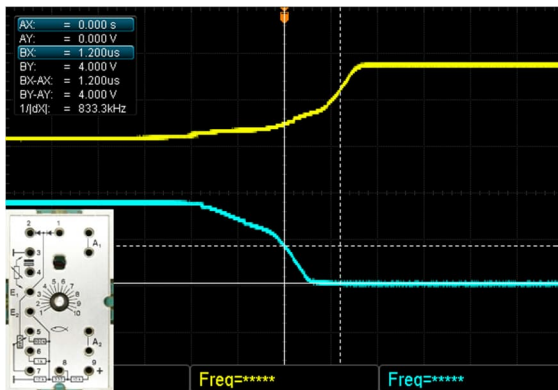


Abb. 10: In der Version mit dem 50-k Ω -Potentiometer ist der Fehler nicht mehr deutlich sichtbar. Auffällig ist, dass die Flanken weniger steil sind.

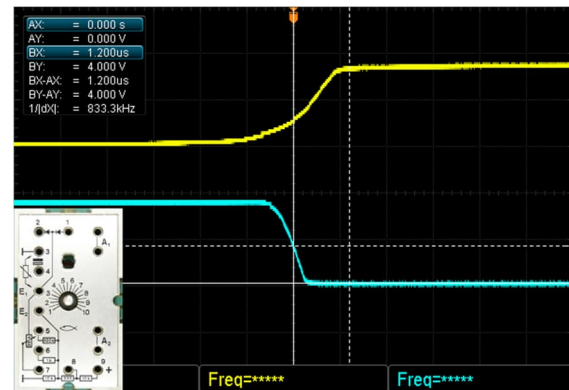


Abb. 11: Die 25-k Ω -Version, nun aber mit dem 1-nF-Kondensator C3. Die Flanken sind etwas steiler geworden.

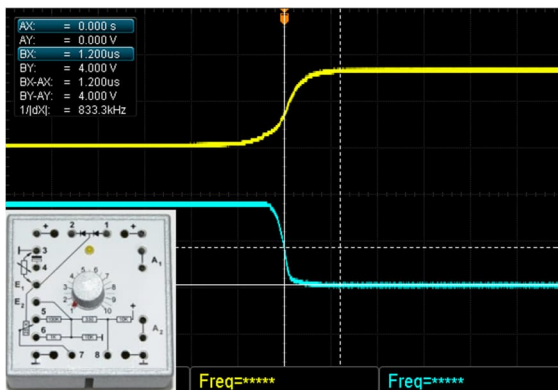


Abb. 12: Die Kassettenversion zeigt keine Störungen

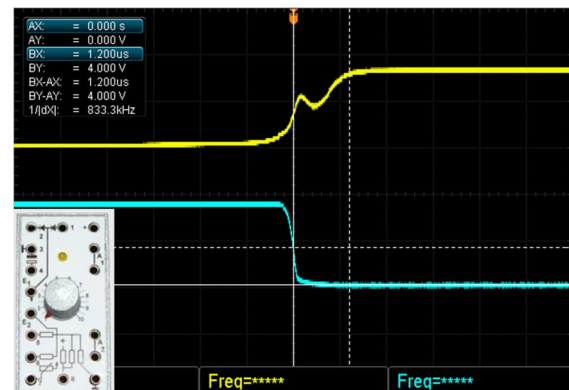


Abb. 13: Die Version für das 9-V-Batteriegehäuse weist nur Störungen an A1 auf

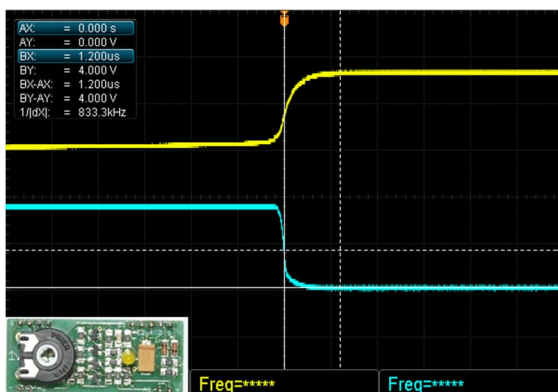


Abb. 14: Meine Shield-Version ist jedoch störungsfrei

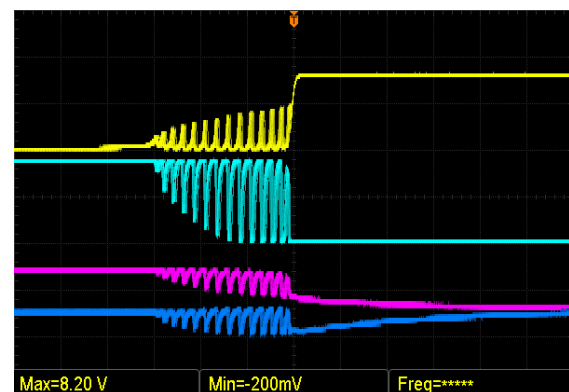


Abb. 15: Das Potentiometer ist nun vollständig im Uhrzeigersinn gedreht

falls den Fehler auf, allerdings in deutlich geringerem Ausmaß und nur auf A1. Die Version mit dem 50-k Ω -Potentiometer ist weniger betroffen. Auffällig ist hier, dass die Flanken deutlich flacher sind (Abb. 10). Ich habe zusätzlich einen 1-nF-Kondensator

zu einem meiner Originale hinzugefügt. Das Ergebnis ist in Abb. 11 zu sehen.

Ich habe auch Module in einem anderen Format hergestellt. In Abb. 12 sehen wir, dass die Version für die graue Kassette völlig sauber ist. Es gibt keinerlei Verzer-

rungen, obwohl die 1 nF vorhanden sind. Die Version für das Batteriegehäuse ist verzerrt (Abb. 13), die Shield-Version ist jedoch unverzerrt (Abb. 14).

Abb. 15 zeigt, dass es noch schlimmer kommen kann: Das Potentiometer ist nun vollständig im Uhrzeigersinn gedreht. Je höher der Widerstand von P1, desto stärker die Störungen.

Da ich es zu diesem Zeitpunkt selbst nicht herausfinden konnte, kontaktierte ich Arnoud van Delden. Er verstand es jedoch auch nicht und konnte mir nicht sagen, woher die Störungen kommen. Er gab mir jedoch einige Hinweise zum Überprüfen und Messen. Arnoud schrieb: *Der Grundbaustein fungiert hier als instabiler Multivibrator, dessen Rückkopplungs-RC-Netzwerk (bestehend aus dem Potentiometer und einem extern angeschlossenen Kondensator) die Frequenz bestimmt. Die Störungen treten an den Flanken des Rechtecksignals auf. Diese Kippunkte des Differenzverstärkers (aufgebaut um die PNP-Transistoren T1 und T2) werden durch die Lade- und Entladekurve des angeschlossenen Kondensators bestimmt. Da einige Designs weniger empfindlich zu sein scheinen als andere, gehe ich davon aus, dass die Störung durch parasitäre Kapazitäten und Induktionen aufgrund des Leiterplattendesigns verursacht wird. Eine Abschirmung mit Masseflächen auf der Leiterplatte könnte dies möglicherweise verbessern. Darüber hinaus spielen vermutlich auch Unterschiede in den Verstärkungsfaktoren der verwendeten Transistoren eine Rolle. Außerdem vermute ich, dass die LED dieses Verhalten verstärken kann, da sie dem Pfad, der zum Aufladen des Kondensators verfolgt wird, ein eigenes, nichtlineares Strommuster hinzufügt. Denn erst wenn sich die Spannung der Schwellenspannung der LED nähert (und die LED aufleuchtet), beginnt ein nennenswerter Strom zu fließen. Da die LED hier in die Rückkopplung eingebunden ist, kann*

dies sicherlich eine Rolle spielen, und möglicherweise sogar erklären, warum die Störung bei meinen nachgebauten Modulen mit Low-Current-LEDs (mit viel größerem Vorwiderstand) nicht auftritt. Um den möglicherweise störenden Einfluss der LED auszuschließen, wäre zu erwägen, diese zum A2-Ausgang zu verlegen (und dort für ein gleichmäßiges Anzeigeleuchten-Verhalten auf Masse zu schalten).

In einer späteren E-Mail gab er an, dass er mehrere Grundbausteine besitzt. Einer davon war jedoch anders als die anderen. Wie in Abb. 16 zu sehen besitzt er ein Modell aus der ersten Serie. Viel interessanter jedoch ist die Verwendung des erwähnten Kondensators C3 in dieser Version. Abb. 17 zeigt, dass es sich um einen „15“ handelt. Zumindest ist das so aufgedruckt. Wir können von 15 pF ausgehen, deutlich weniger als die von mir angenommenen 1 nF.

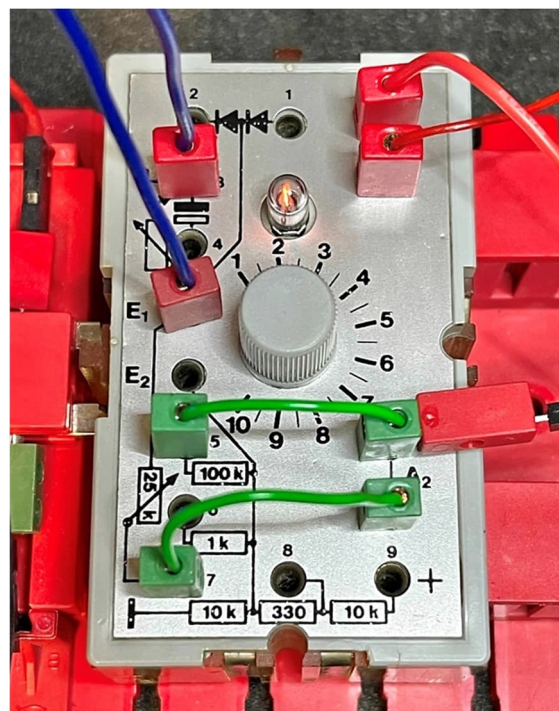


Abb. 16: Arnouds Ur-Version

Ich habe den 1 nF aus einem meiner Module entfernt und durch einen 15 pF ersetzt. Dies brachte jedoch nicht die erwarteten Ergebnisse. 100 bis 150 pF brachten jedoch ein besseres Ergebnis.

Warum ist C1 überhaupt da?

Das war auch eine von Arnouds Anmerkungen.

Deshalb habe ich C1 aus einem Grundbaustein entfernt und alle möglichen Schaltungen aus dem Handbuch von hobby 4 ([39540](#)) aufgebaut. Alle Schaltungen funktionierten einwandfrei.



Abb. 17: Ein 15-pF-Kondensator in der Ur-Version

Arnoud und ich vermuten, dass während der Entwicklung des Grundbausteins ein Fehler entdeckt wurde. Deshalb wurde C1 in die Schaltung eingebaut. Später stellte man fest, dass dies einen weiteren Fehler verursachte, der durch Hinzufügen von C3 behoben werden musste.

Es ist möglich, dass die Qualität der damals verfügbaren Bauteile dafür verantwortlich war. Bei Toleranzen von 20 % oder mehr kann eine Schaltung natürlich aus dem Ruder laufen. Heutzutage arbeite ich mit Toleranzen von 1, 2, 5 oder 10 %. Die Qualität der aktuellen Standardtransistoren vom Typ BC547 und BC557 ist ebenfalls besser als die der BC170/BC250 in Arnouds Ur-Version oder der späteren BC238B/BC308B.

Aber gibt es da nicht einen merklichen Unterschied? Nicht wirklich... Der einzige Unterschied besteht darin, dass die maximal erzeugbare Frequenz ohne C1 und ohne C3

200 kHz oder mehr beträgt. Mit $C1 = 1 \text{ nF}$ sinkt sie auf etwa 128 kHz, und mit $C3 = 1 \text{ nF}$ sinkt sie auf 62 kHz. Mit $C3 = 150 \text{ pF}$ erreicht die Frequenz etwa 90 kHz. Dies kann jedoch von Grundbaustein zu Grundbaustein variieren.

Da aber nur wenige Leute einen Grundbaustein zur Erzeugung so hoher Frequenzen verwenden, sollte das keine Probleme verursachen. Man kann C1 also getrost weglassen.

Arnouds Grundbausteine weisen jedoch nicht dieselbe Verzerrung auf wie meine. Weder sein Breadboard-Aufbau noch seine selbstgebaute Versionen zeigten diese Verzerrung.

Da weder Arnoud noch ich die Ursache finden konnten, vereinbarten wir, dass ich ihm einen meiner verzerrten Grundbausteine geben würde, damit er sich selbst ein Bild von der Sache machen und gegebenenfalls weitere Untersuchungen durchführen konnte.

Beim FTCN-Treffen am 18. Oktober war es so weit. Ich gab Arnoud einen verzerrenden Grundbaustein und zwei verzerrungsfreie. Außerdem gab ich ihm einige lose Bauteile und drei verschiedene Leiterplatten. Da ich auch Versionen ohne Verzerrung gebaut hatte, vermutete ich, dass es mit dem Layout zusammenhängen könnte. Später stellte sich jedoch heraus, dass dies nur teilweise der Fall ist.

Einige Tage später trafen seine ersten Untersuchungsergebnisse per E-Mail ein. Arnoud schrieb: *In den frühen Entwürfen wurde offenbar mit einem C3 im Rückkopplungsnetzwerk experimentiert. Ich denke, dass dies sicherlich einen Effekt haben kann, aber der Wert sollte nicht zu groß gewählt werden, da die Ecken der Rechteckwelle dann schnell weniger scharf (runder) werden. Vielleicht ist es daher sinnvoller, einfach den Wert des Rückkopplungswiderstands (ursprünglich $2,7 \text{ M}\Omega$) auf beispielsweise $1,5 \text{ M}\Omega$ zu reduzieren.*

Der Kondensator C1 (1 nF) erscheint mir für deine Anwendung (als Oszillator) wenig sinnvoll. Ich vermute, dass er aufgenommen wurde, um die Stabilität um den eingestellten Schwellenwert symmetrischer Eingangssignale zu verbessern. Etwas, das mir nur bei einem Mikrofon oder einem Elektromagneten sinnvoll erscheint. Wenn der Grundbaustein als Oszillator eingesetzt wird, scheint er mir die Störanfälligkeit nur unerwünscht zu erhöhen.

Doppelseitige Leiterplatte

Da Arnoud seine Leiterplatten nach dem Verlegen der Leiterbahnen beidseitig mit Masseflächen versieht, kam mir der Gedanke, dass dies einer der Gründe sein könnte, warum seinen Nachbauten der „Grünbausteine“ keine Störungen aufweisen. Also habe ich es genauso gemacht (siehe Abb. 18). Und ich muss sagen, dass bei dieser Version tatsächlich keine Störungen sichtbar waren. Nur beim Entfernen der Verbindung „A2 - 5“ und Drehen des Potentiometers bis zum rechten Anschlag zeigte sich eine leichte Delle an A1.

Meine Vermutung ist, dass durch das Verfüllen der Leiterplatte mit Masseflächen kleine Kondensatoren relativ zu den Leiterbahnen entstehen. Diese hätten dann dieselbe Wirkung wie C3.

Glühbirne oder LED?

Nachdem ich mehrere verschiedene Layouts entworfen und gebaut hatte, war ich immer noch nicht ganz zufrieden. Arnoud hatte mich wiederholt darauf hingewiesen, dass die verwendeten LEDs eine mögliche Ursache sein könnten. Anfangs hatte ich bereits festgestellt, dass die Störungen schon vorhanden waren, obwohl die LED noch nicht installiert war. Daher hatte ich mir über die LEDs keine weiteren Gedanken gemacht.

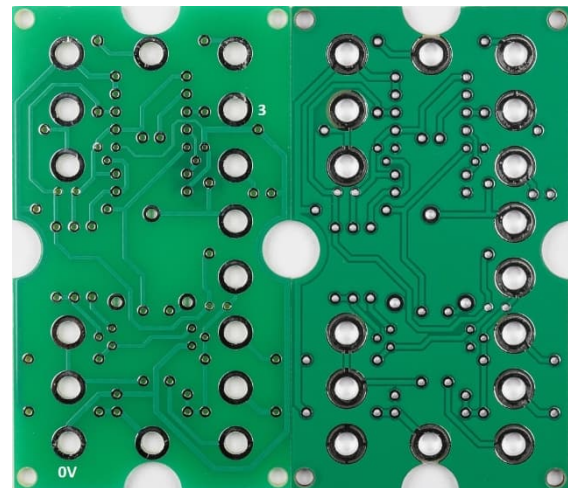


Abb. 18: Links die Unterseite der einseitigen Leiterplatte, rechts die doppelseitige, bei der die gesamte Oberfläche als Massefläche dient. Man erkennt deutlich, dass die Leiterbahnen von der Massefläche umschlossen sind. Die 0V-Leiterbahnen bilden nun eine Kupferfläche.

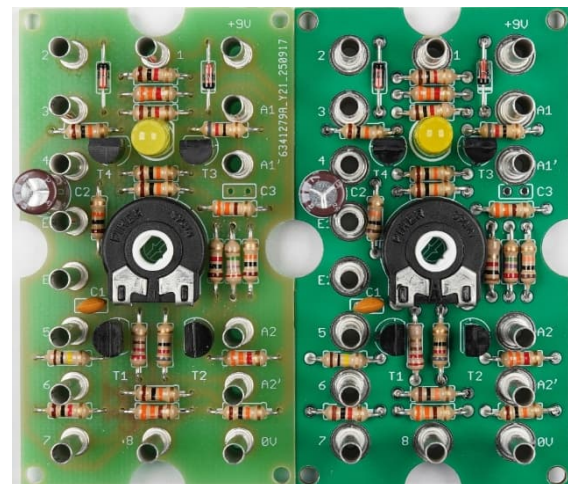


Abb. 19: Die Bauteilseite beider Leiterplatten: Betrachtet man die Lötstellen und die Bundhülsen, so erkennt man, dass sie durch einen schmalen Ring von der Kupferoberfläche getrennt sind. Nur die mit Masse (0V) verbundenen Pads und Leiterbahnen haben daher Kontakt zur Massefläche.

Später ersetzte ich die 20-mA-LED durch einen 2-mA-LED. Wie erwartet verbesserte dies die Situation nicht.

Doch dann dämmerte es mir plötzlich: Das Original hatte eine Glühbirne, keine LED. Also nahm ich die Originalglühbirne und den 220-Ω-Widerstand und schloss sie an

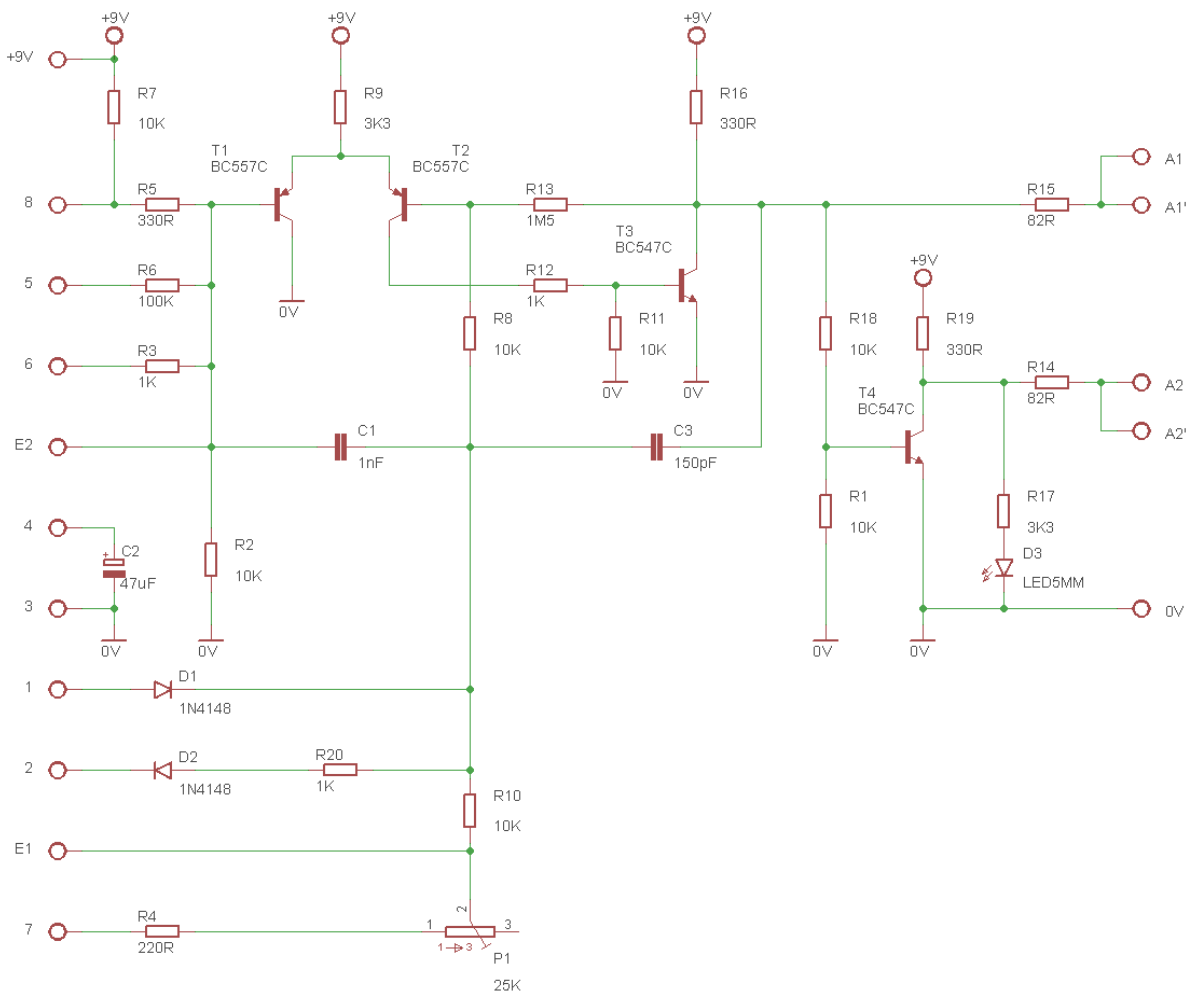


Abb. 20: Alternatives Schaltbild des Grundbausteins

meinen Nachbau an. Ergebnis: Keine Störungen mehr.

Ich maß dann die Glühbirne: Ihr Einzelwiderstand beträgt $35\ \Omega$. Wenn sie, in Reihe mit dem $220\text{-}\Omega$ -Widerstand R17, unter Spannung gesetzt wird (in diesem Fall bei $8,84\text{ V}$), fließen $18,12\text{ mA}$ hindurch, was einem Widerstand von etwa $490\ \Omega$ entspricht. Nach Abzug der $220\ \Omega$ des Widerstands R17 beträgt der Widerstand der Glühbirne $270\ \Omega$. Wenn ich nun die Reihenschaltung der Birne mit dem $220\text{-}\Omega$ -Widerstand mit dem $1\text{-k}\Omega$ -Widerstand parallelschalte (also LED + R17 und das Ganze parallel zu R16), erhalte ich einen Ersatzwiderstand von $330\ \Omega$. Das bedeutet, dass $8,84\text{ V} / 330\ \Omega = 27\text{ mA}$ durch dieses Netzwerk fließen.

Bei der LED sieht die Sache anders aus. Sie hat einen eigenen Widerstand von 17,79 k Ω . Mit dem in Reihe geschalteten 3,3-k Ω -Widerstand und einer Spannung von 8,84 V fließen 2,084 mA. Der Widerstand sollte also $8,84 \text{ V} / 2,084 \text{ mA} = 4,241 \Omega$ betragen. Davon muss man die 3,3 k Ω abziehen, was 941 Ω ergibt. Das ist fast das Vierfache des Widerstands der Birne von 270 Ω .

Um wieder den gleichen Strom von 27 mA zu erreichen, musste ich den 3,3-k Ω -Widerstand auf 1,8 k Ω reduzieren.

Nach dem Entfernen der Glühbirne zeigt jedoch auch der Original-Grundbaustein Störungen. Dieser muss daher an seinem Platz bleiben.

Alternativen?

Basierend auf meinen eigenen Erkenntnissen und Arnouds Hinweisen habe ich eine alternative Version entworfen.

Das Schaltbild ist in Abb. 20 dargestellt. Auffällig ist, dass die LED nun an Ausgang A2 angeschlossen ist. Um die gleiche Funktion wie im ursprünglichen Zustand zu gewährleisten, ist die LED nun mit 0 V verbunden. Außerdem wurde der Wert von R16 auf 330 Ω reduziert (entsprechend dem Wert von R19) und R13 auf 1,5 M Ω . Insbesondere die Reduzierung von R13 verbessert die Schaltung. Die Flanken sind nun deutlich steiler.

Des Weiteren wurden die Transistoren durch BC547C/557C ersetzt. Ich hatte die BC238/308 vor einigen Wochen bestellt, aber sie sind wahrscheinlich nicht mehr erhältlich.

Leider konnte die Störung dadurch nicht beseitigt werden. C1 verursacht einfach zu viele Probleme. Durch Hinzufügen von C3 mit 150 pF wird dem jedoch entgegen gewirkt.

C1 und C3 können aber auch weggelassen werden. Wie bereits erwähnt, hat dies keine negativen Auswirkungen auf die Funktion der Schaltung.

Da R16 nun ebenfalls 330 Ω beträgt, ist die Ausgangsspannung an A1 nun gleich der Spannung an A2 (siehe Abb. 21).

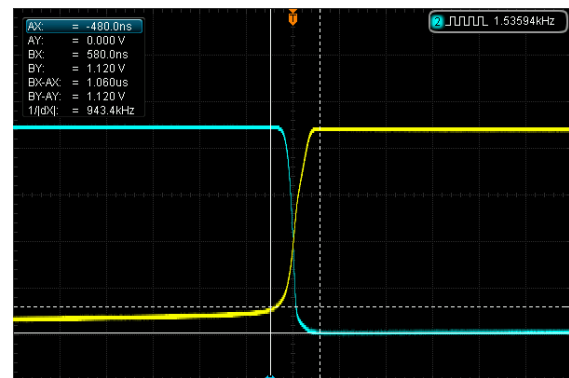


Abb. 21: $R16 = R19 = 330 \Omega$: Beide Ausgänge sind nun gleich.

Fazit

Es ist offensichtlich, dass keiner von uns beiden die beobachtete Störung wirklich erklären kann. Nach vielen Experimenten hat sich aber gezeigt, dass die Schaltung mit nur wenigen einfachen Anpassungen verbessert werden kann. Ob jemand dies für notwendig hält, überlassen wir euch.

(Die Abb. 16 und 17 stammen von Arnoud van Delden, die anderen Bilder von Peter Krijnen.)

Quellen

- [1] Peter Krijnen: *Es quietscht und pfeift: Wie man eine Orgel baut*. [ft:pedia 3/2025](#), S. 21–31.
- [2] Peter Krijnen: *Silberlinge: Original oder Nachbau (Teil 3)*. [ft:pedia 2/2021](#), S. 90–100.

Geschichte

Der COMPUTING Baukasten – eine Retrospektive

Volker Paelke

Im Oktober 2025 feierte fischertechnik in einem Social Media Post 40 Jahre fischertechnik Computing: „Schon 1985 brachte fischertechnik mit dem COMPUTING Baukasten die ersten computergesteuerten Roboter-Modelle auf den Markt“. Doch gab es diesen Baukasten nicht schon früher? Und warum ist auf dem Karton kein fischertechnik Computing Interface zu sehen? Zeit, den Computing Grundbaukasten einmal genauer zu betrachten.

Historischer Kontext

Anfang der 1980er-Jahre gab es Bemühungen, Computertechnologie und Mikroelektronik einer breiten Öffentlichkeit näher zu bringen. Dazu zählt das bekannte BBC Computer Literacy Project, aus dem der Acorn BBC Micro hervorging [1]. Eine der „Vorzeige-Anwendungen“ war der von der Firma Economatics aus fischertechnik-Teilen gebaute BBC Buggy [2], der schon in der ft:media vorgestellt wurde [3]. Auch in Deutschland entstand eine ähnliche Initiative: Der NDR produzierte 1984 unter der Leitung von Joachim Arendt im Schulfernsehen die Serie „Einführung in die Mikroelektronik“ und setzte dabei den modular aufgebauten, von Rolf-Dieter Klein entworfenen, NDR-Klein-Computer (NKC) ein [4].

Für diese Serie sollten praktische Anwendungen nicht nur am Bildschirm, sondern auch an realen technischen Modellen demonstriert werden. Dazu wurden von fischertechnik die Modelle Teach-In-Roboter und Grafiktablett entwickelt. Sie sind in Folge 0 (Vorstellung des NKC), sowie in Folge 9 (Roboter steuern) zu sehen. Abb. 1 zeigt Rolf-Dieter Klein in der Folge 0, mit dem Teach-In-Roboter auf dem

Monitor vorne links, und dem fischertechnik Grafiktablett vorne rechts.



Abb. 1: fischertechnik-Computing-Modelle in der NDR-Serie „Einführung in die Mikroelektronik“ (NDR-Schulfernsehen 1984)

Begleitend zur Serie erschien 1984 ein Computing Grundbaukasten von fischertechnik, aus dem diese Modelle nachgebaut werden konnten. Das Handbuch ([39495](#)) ist auf 5/1984 datiert und enthält Bauanleitungen und Programmierhinweise zu den beiden Modellen „Teach-In-Roboter“ und „Grafiktablett“. Anzeigen für diesen Baukasten fanden sich 1984 in der NKC-Zeitschrift Loop [5] (Abb. 2). In dieser Anzeige werden auch bereits zehn Modelle erwähnt.



Abb. 2: Anzeige für den fischertechnik-Computing-Baukasten zum NDR-Klein-Computer von 1984

Um eine breitere Nutzergruppe anzusprechen, stellte fischertechnik dann 1985 ein umfassendes Computing-System vor [6]. Der Grundbaukasten bekam einen neuen Karton, wobei das Originalbild nur etwas beschnitten wurde, weshalb das darauf sichtbare Interface weiterhin eine NKC-Platine abbildet – und nicht etwa eines der fischertechnik-Computing-Interfaces.



Abb. 3: Der Grundbaukasten im Computing-System von 1985

Außerdem gab es eine neue Bauanleitung (39496) und die zugehörige Programmieranleitung (39497) für insgesamt zehn Modelle. Gleichzeitig wurden Interfaces für Commodore, Apple II und den Acorn BBC Micro präsentiert. Abb. 3 zeigt den Computing-Bausatz in der „neuen“ Verpackung.

Der Baukasten und sein Inhalt

Da die Computing-Interfaces bereits mehrfach Thema in der ft:pedia waren und sowohl Nachbauten dieser Hardware als auch moderne Alternativen wie der ftDuino oder die TXT-Controller verfügbar sind, soll der Schwerpunkt in diesem Beitrag auf dem Grundbaukasten und seinen Modellen liegen.

Der Baukasten (Abb. 4) besteht aus etwa 260 Teilen, der Bauanleitung für zehn verschiedene Modelle sowie einer Programmieranleitung. Ein großer Vorteil des Grundbaukastens ist, dass die verwendeten Bauteile nicht speziell für diesen Baukasten entwickelt wurden, sondern aus dem regulären fischertechnik-Sortiment stammten. Die meisten Teile lassen sich daher leicht gebraucht oder auch heute noch neu beschaffen und alternativ durch heute gängige fischertechnik-Komponenten ersetzen. Wer nach sinnvollen Anwendungen für aktuelle oder klassische Controller sucht, kann die Modelle also auch ohne den Baukasten gut als Inspiration nutzen.



Abb. 4: Inhalt des Grundbaukastens

Modelle, Funktionen und Lerninhalte

Der Baukasten deckt dabei verschiedene technische Konzepte ab: Von Zustandsautomaten (Ampelanlage) über automatische Bewegungssteuerung (Aufzug, Werkzeugmaschine) bis hin zu rekursiven Algorithmen (Turm von Hanoi-Roboter) und sensorgesteuerten Systemen (Sortieranlage, Antennenrotor).

Die ersten vier Modelle in der Anleitung (Abb. 5) dienen in erster Linie dazu, die grundlegenden Techniken und damit die verschiedenen „Entwurfsmuster“, sowohl in der elektro-mechanischen Umsetzung in fischertechnik als auch in der Programmierung kennenzulernen.

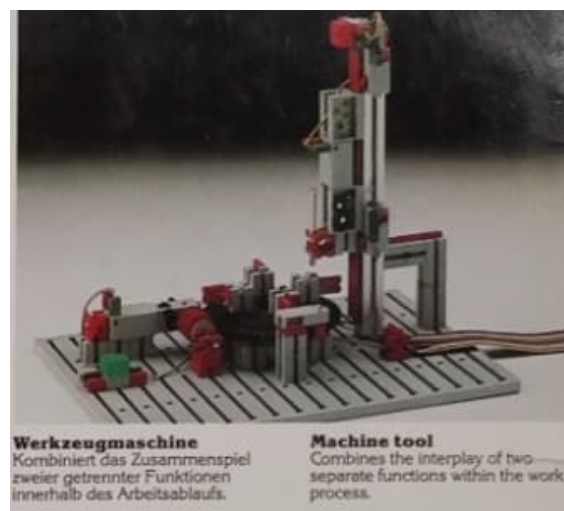
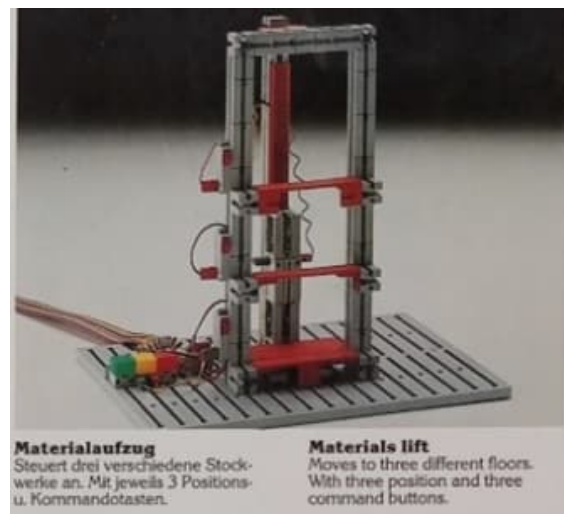
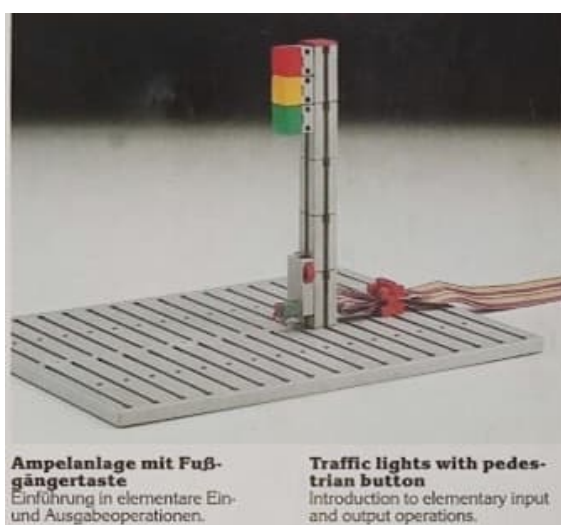
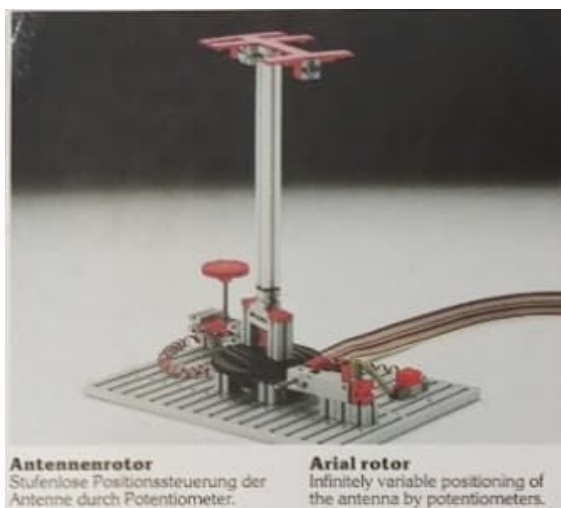


Abb. 5: Die ersten vier Modelle – Antennenrotor, Ampelanlage, Werkzeugmaschine und Aufzug

Die Ampelanlage mit Fußgänger-Taster führt in endlichen Automaten ein: Im normalen Betrieb folgen die Zustände „Grün“, „Gelb“, und „Rot“ einander in zeitlich definierten Abständen. Ein Tastendruck ändert den Ablauf und verkürzt die „Grün“-Phase.

Die Werkzeugmaschine vertieft dieses Prinzip und führt geschlossene Regelkreise mit Soll-Ist-Vergleich ein. Die Maschine durchläuft eine regelmäßige Abfolge von Zuständen, wobei ein Zustandswechsel entweder zeitgesteuert (Bohren) oder durch Ereignisse (Auslösen der Referenztaster bei Bewegungen) erfolgt. Hinsichtlich der elektro-mechanischen Umsetzung werden

die Grundlagen der Motorsteuerung, lineare Bewegungen (mit Führung durch ein leeres zweites Getriebe), rotatorische Bewegungen und die digitale Positionsbestimmung über Referenzaster eingeführt.

Der Aufzug greift die digitale Positionsbestimmung über einen Referenzaster und den Soll-Ist-Vergleich auf und motiviert unterschiedlich komplexe Steuerungen, etwa durch Erweiterungen des Beispielprogramms um die Anforderung des Aufzugs durch weitere Taster, die Optimierung und Priorisierung von Fahrwegen oder die Nutzung der Computer-Tastatur als alternative Eingabemöglichkeit.

Der Antennenrotor schließlich führt in analoge Techniken ein: Zwei Potentiometer dienen als Winkelgeber für die analoge Positionsbestimmung. Einmal als Eingabe des Benutzers (Steuerelement) und dann zur Positionsbestimmung des Rotors. An diesem Beispiel wird das Konzept des analogen Soll-Ist-Vergleichs erklärt. Die Steuersoftware führt die Position des Rotors der Eingabe des Nutzers nach. Dabei nutzt sie Puls-Weiten-Modulation (PWM) als „pseudo“-analoge Ausgabe, um die Geschwindigkeit so zu regeln, so dass der Rotor nicht über den Zielwinkel hinaus bewegt wird: Je kleiner die Differenz zwischen „Soll“- und „Ist“-Winkel, desto kürzer wird die Anschaltdauer des Motors gewählt.

Ein etwas größeres Modell ist dann die Sortieranlage. Eine lineare Bewegung wird hier zum Material-Transport genutzt, wobei Taster die Abmessungen des transportierten Bausteins messen, um lange und kurze Bausteine zu unterscheiden. In der Steuersoftware wird wieder ein endlicher Automat implementiert. Neu kommt hier eine Fall-Unterscheidung hinzu, so dass in Abhängigkeit von den gemessenen Abmessungen die kurzen Bausteine 15 nach links und die langen Bausteine 30 nach rechts transportiert und so sortiert werden.

Abb. 6 zeigt die Sortieranlage aus der Bauanleitung, Abb. 7 das zugehörige Diagramm für den endlichen Automaten aus der Programmieranleitung.

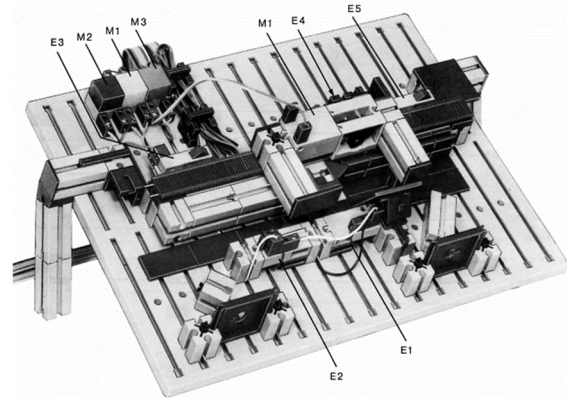


Abb. 6: Sortieranlage aus der Bauanleitung

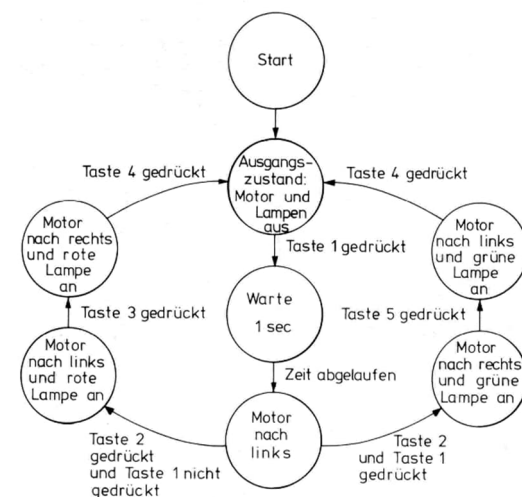


Abb. 7: Endlicher Automat aus der Programmieranleitung.

Als erster Roboter folgt der „Turm von Hanoi“-Roboter (Abb. 8): Dieser basiert auf der klassischen Denksportaufgabe, bei der Scheiben unter der Regel „keine größere auf eine kleinere“ zwischen drei Positionen umgestapelt werden. Mechanisch ist er ein zweiachsiger Roboter, der sich drehen und in der Höhe verfahren kann, ergänzt durch einen Elektromagneten als Aktor zum Greifen und Ablegen von Metallscheiben mit unterschiedlichem Durchmesser. Bemerkenswert ist, dass gemäß der Bauanleitung Eingabe- und Sensorelemente eingebaut werden (z. B. Potentiometer zur analoge

gen Höhenbestimmung), die das Beispielprogramm gar nicht nutzt (es beschränkt sich auf einfache Taster zur Höhenreferenz). Da die Belegung insbesondere der Potentiometer identisch zum folgenden „Teach-In“-Roboter ist, kann der Hanoi-Roboter damit alternativ auch als Teach-In-Roboter betrieben werden.

Während die verschiedenen im Roboter genutzten elektromechanischen Komponenten bereits aus vorhergehenden Modellen bekannt sind, führt die Software das algorithmische Konzept der Rekursion ein: Um einen Stapel aus N Scheiben umzusetzen, verschiebt man zunächst die $N-1$ oberen Scheiben rekursiv auf einen Hilfsstapel, dann die größte Scheibe auf den Zielstapel und anschließend die zuvor beiseite geräumten $N-1$ Scheiben erneut rekursiv auf den Zielstapel.

An dieser Stelle zeigen sich auch deutlich die Einschränkungen der verwendeten BASIC-Dialekte. Da die BASIC-Interpreter der damaligen Heimcomputer keine Rekursion unterstützten, musste diese umständlich und schwer verständlich simuliert werden. Es empfiehlt sich daher, an dieser Stelle nur die Mechanik und Idee zu übernehmen, die Software aber in einer rekursionstauglichen Sprache zu entwickeln.

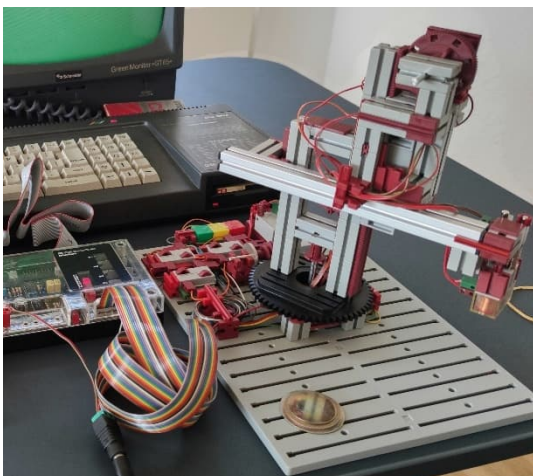


Abb. 8: Der Turm-von-Hanoi-Roboter am Schneider CPC 6128 (ebenfalls von 1985)

Der aus der Fernsehserie bekannte Teach-In-Roboter (Abb. 9) ist das wohl beliebteste Modell des Baukastens, da er fast alle Möglichkeiten des Interfaces nutzt und auch interaktiv genutzt werden kann.

Mechanisch greift er auf Elemente zurück, die bereits bei den vorherigen Modellen verwendet wurden, wobei sowohl die Orientierung, als auch die Arm Position analog über Potentiometer gemessen werden. Die Steuerung ist besonders interessant: Hier kommt das Konzept der Programmierung hinzu. Nutzer können den Roboter interaktiv über die acht Taster steuern (Arm auf/ab, Drehen links/rechts, Magnet ein/aus, Zustand lernen, Programm starten/stoppen) und neue Abläufe definieren und speichern, sodass neue Programme direkt am Roboter „gelernt“ werden können (daher „Teach-In“-Roboter). Wie bereits erwähnt kann man auch den Turm-von-Hanoi-Roboter als Teach-In-Roboter verwenden. Nach dem Teach-In kann der Roboter die Abläufe beliebig oft autonom wiederholen.

Damit bietet der Teach-In-Roboter eine anschauliche Einführung in die Konzepte der Programmierung und interaktiven Steuerung, während er gleichzeitig die volle Bandbreite der Interface-Möglichkeiten des Baukastens demonstriert (Abb. 9).

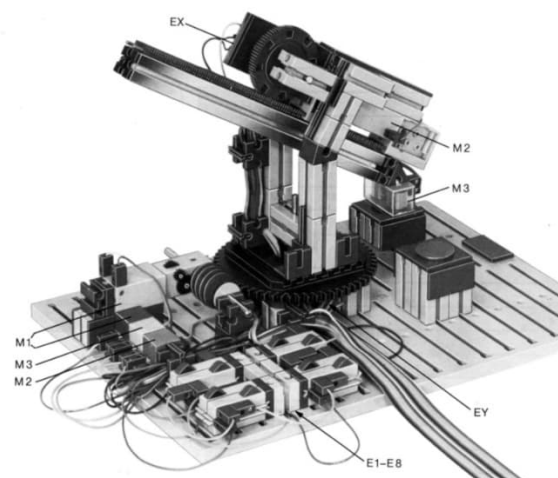


Abb. 9: Der Teach-In-Roboter

Das nächste Modell ist das ebenfalls in der Serie „Einführung in die Mikroelektronik“ vorgestellte Grafiktablett (Abb. 10).

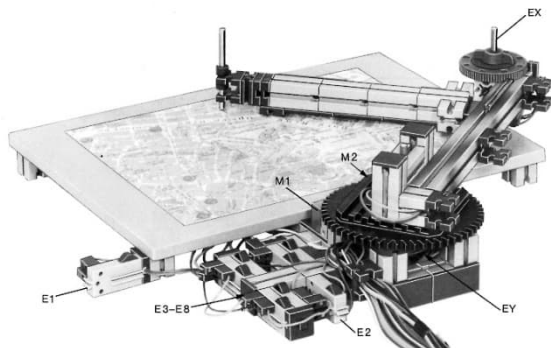


Abb. 10: Grafiktablett

Ein Grafiktablett als selbstgebautes Eingabegerät war zum damaligen Zeitpunkt besonders interessant, da es eine präzise 2D-Positions- und Eingabemöglichkeit bot, die mit den damals an Heimcomputern gängigen Eingabegeräten wie Tastatur oder digitalem Joystick nicht möglich war. So wurde z. B. die Digitalisierung von Karten oder Zeichnungen ermöglicht, etwas, das heute durch Kameras, Scanner, Touchscreens und moderne Tablets selbstverständlich ist.

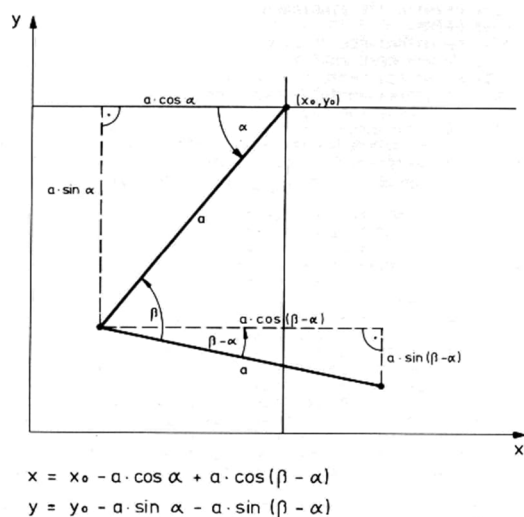


Abb. 11: Grafiktablett-Geometrie aus Programmieranleitung

Während der Anwendungsfall heute nicht mehr so motivierend ist, bietet das Modell eine interessante Einführung in die Vorwärtskinematik. Die Programmieranleitung

erklärt, wie aus den zwei mit den Potentiometern gemessenen Winkelwerten, die kartesischen X- und Y-Koordinaten der Position des Eingabestifts berechnet werden können (Abb. 11).

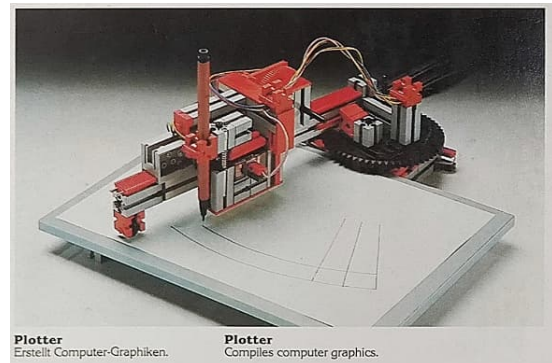


Abb. 12: Polarkoordinaten-Plotter

Der Polarkoordinaten-Plotter (Abb. 12) ist ein praktisch etwas eingeschränktes Modell. Die Software ist extrem limitiert, vermutlich wurde der Plotter erst spät im Entwicklungsprozess ergänzt. Inhaltlich wird in der Anleitung nachvollziehbar beschrieben, wie der Plotter arbeitet. Da es sich um einen Polarkoordinaten-Plotter handelt, müssen die gewünschten Zielpunkte (in kartesischen Koordinaten), zu denen der Stift geführt werden soll, zunächst in passende Winkel- und Entfernungsangaben umgerechnet werden. Auf komplexere Inhalte wie das Plotten von freien Formen, Buchstaben oder mehrteiligen Objekten geht die Anleitung hingegen nicht ein.

Dafür gab es im Computing-Programm ab Ende 1985 einen eigenen, deutlich leistungsfähigeren Plotter. Dies erklärt vermutlich auch, warum der Polarkoordinaten-Plotter heute nur selten aufgebaut zu sehen ist: Mit der Standardsoftware lassen sich ausschließlich „Kuchendiagramme“ zeichnen, wie sie in Abb. 12 zu sehen sind.

Obwohl die Entwickler davon ausgingen, dass Nutzerinnen und Nutzer eigene Programme schreiben würden, ist dies heute eher unwahrscheinlich. Meist wird nur ausprobiert, was ursprünglich mitgeliefert wurde. Beim Plotter verweist die Anleitung

explizit auf die Erweiterungs- und Umbaumöglichkeiten, etwa den Umbau zu einem Portalroboter für das klassische Spiel „Nim“.

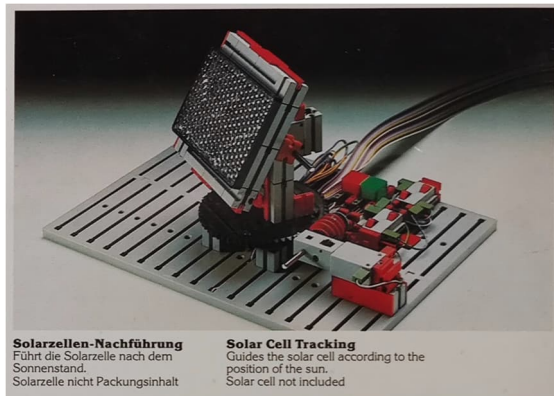


Abb. 13: Solarzellen-Nachführung

Das letzte Modell zur Solarzellen-Nachführung (Abb. 13) wirkt ebenfalls wie eine späte Ergänzung, die zwar technisch interessant ist, in der Dokumentation jedoch nur oberflächlich behandelt wird. Da dem Baukasten keine Solarzelle beiliegt, kann das Modell auch keine echte Funktion erfüllen.

In der Programmieranleitung wird lediglich grob umrissen, wie sich die optimale Ausrichtung einer Solarzelle in dreidimensionalen Winkelkoordinaten berechnen lässt, wobei Faktoren wie geografische Position, Datum und aktuelle Uhrzeit berücksichtigt werden müssen. Zwar ist die entsprechende Berechnung im abgedruckten BASIC-Programm implementiert, doch im Vergleich zu den anderen Modellen bleibt die Berechnung eine Black Box. Hier fehlte offensichtlich die Zeit oder der Platz für eine entsprechende didaktische Aufbereitung. Hinsichtlich der Programmierung wird ein interessanter Aspekt angesprochen: Die Kalibrierung der Potentiometer. Allerdings bleibt die Anleitung auch hier unbestimmt und verweist zur Umsetzung lediglich auf „die Literatur“.

Resumé

Der Computing-Baukasten bietet eine Vielzahl interessanter Modelle, die sich entweder direkt aus dem Baukasten nachbauen

lassen oder als Inspiration dienen können, um sie mit Bausteinen aus anderen Kästen funktionsähnlich nachzubilden. Die Konstruktionen lassen sich selbstverständlich mit den klassischen Interfaces ansteuern, eignen sich aber ebenso gut für den Einsatz an modernen Controllern. Die Programmieranleitung basiert auf dem damals üblichen BASIC (Commodore BASIC, mit Hinweisen, wie die Programme für andere Rechner anzupassen sind).

Aus heutiger Perspektive ist es nur begrenzt sinnvoll, diese Beispiele unverändert aufzugreifen, zumal Programmiersprachen mit strukturierten Datentypen, Rekursion, While-Schleifen etc. die Umsetzung erheblich vereinfachen und verständlicher machen. Mit den Hinweisen aus der Programmieranleitung ist eine Neuimplementierung gut machbar.

Auch wenn die Beschreibungen teilweise nicht besonders detailliert sind und sich auf Bau- und Programmieranleitung verteilen, lässt sich ein schlüssiges didaktisches Konzept erkennen: Die Funktionsprinzipien der Modelle bauen aufeinander auf und vermitteln sukzessive Grundprinzipien der Steuerungstechnik und Automatisierung, sowie deren Umsetzung in fischertechnik.

In der Anleitung wird deutlich, dass die Entwickler darauf abzielten, ein schrittweises Erlernen einzelner Basistechniken und Komponenten zu ermöglichen, die später in eigenen Modellen eingesetzt werden können. Der Baukasten dient also nicht primär dazu, einmalig ein vorgegebenes Modell nachzubauen, sondern die zugrundeliegenden Techniken kennenzulernen, auszuprobieren und in eigenen Konstruktionen weiterzuverwenden. Entsprechend schließt die Anleitung mit Hinweisen, wie z. B. der Teach-In-Roboter in komplexere Umgebungen integriert werden kann.

Beim Computing-Grundbaukasten wird deutlich, dass der Baukasten nicht nur seinerzeit hochaktuell und durchdacht war,

sondern auch heute noch viele spannende und lehrreiche Aspekte bietet, auch wenn die Erklärungen und Programmieranleitungen aus heutiger Sicht schon recht „retro“ sind.

Eine detaillierte, ebenfalls auf dem Computing Grundbaukasten basierende Darstellung der Robotik und Steuerungsprogrammierung finden sich in den Büchern „Experimente zur Robotik“ von Roland Schulé [7] und „Robotik mit dem Homecomputer“ von J.P.M. Steeman [8].

Quellen

- [1] BBC: [Computer Literacy Project \(1980-1989\)](#).
- [2] BBC: [Everything under Control. 8. Folge](#), Sendereihe „Making the most of the Micro“, ausgestrahlt am 28.02.1983.
- [3] Dirk Uffmann, Roland Enzenhofer: *Wiederbelebung eines fischertechnik-Buggy-Modells von 2002*. [ft:media 2/2016](#), S. 73–80.
- [4] Webseite zum [NDR-Klein-Computer](#)
- [5] Anzeige in Loop – Zeitung für Computer-Bauer, Anwender, Programmierer und Starter, Ausgabe 2/1984, Graf Elektronik Systeme, S. 10.
- [6] Presseankündigung zum fischertechnik-Computing-System: *Computing-Bauset*, HC – Mein Home-Computer, Ausgabe 4/1985, Vogel Verlag, S. 7.
- [7] Roland Schulé: *Experimente zur Robotik*. Franzis-Verlag, München, 1988.
- [8] J.P.M. Steeman: *Robotik mit dem Homecomputer*. Elektor Verlag, Aachen, 1987.

Computing

Die FTlegacy- und FTmodule-Software-Bibliotheken

Jeroen Regtien

Dieser Beitrag stellt zwei Softwarebibliotheken vor, die mit Arduino-Mikrocontrollern zur Steuerung von fischertechnik-Modellen verwendet werden können. Dabei kommen das Parallel- (Universal-), serielle (Intelligent-) oder ROBO-Interface sowie kommerzielle Shields wie die Didacta Shields, das Adafruit Motor Shield V2 oder das ftNano Shield zum Einsatz. Der ftDuino wird als Sonderfall ebenfalls behandelt. Die Bibliotheken minimieren die Hardware-abhängigkeit der Programme, sodass einfache wie auch komplexere Modelle mit maximaler Portabilität und minimalem Anpassungsaufwand programmiert werden können.

Einführung

Wenn ich Controller gesteuerte fischertechnik-Modelle baue, mag ich es nicht, die Software beim Wechsel von einer auf eine andere Hardwareplattform umschreiben zu müssen, sei es für einen anderen Mikrocontroller oder für ein Motor Shield bzw. Interface. Ich benutze auch gerne alte fischertechnik-Interfaces als I/O Shield für Mikrocontroller; in diesem Projekt wurden daher beide berücksichtigt.

In früheren ft:pedia-Beiträgen wurde die Hardware-Verbindung zwischen einem Arduino und den älteren fischertechnik-Interfaces ausführlich diskutiert [1, 2, 3, 4, 5, 6]. Arduino Unos, Nanos und Megas können auch mit kommerziell erhältlichen Motor Shields wie dem Adafruit Motor Shield V2, den Didakta Shields und dem ftNano [7] verwendet werden. Allerdings erfordert jede dieser Hardwarelösungen einen anderen und manchmal recht spezifischen Ansatz in der Software, was lästig sein kann, wenn man von einer Controller/Shield-Kombination auf eine andere wechseln möchte. Um dieses Portabilitätsproblem zu umgehen habe ich eine geschichtete Software-Bibliothek entwickelt, die die meisten Hardware-

spezifischen Befehle verbirgt und es dem Benutzer ermöglicht, Programme zu entwickeln, die so hardwareunabhängig sind wie möglich.

In diesem Beitrag liegt der Fokus auf der Software. Ich beschreibe die verwendeten Bibliotheken und gebe Beispiele für deren Nutzung. Die Implementierung erfolgt in C++ innerhalb des Arduino Integrated Development Environment (IDE). Die Wahl fiel auf die Arduino-IDE, weil sie weit verbreitet und benutzerfreundlich ist.

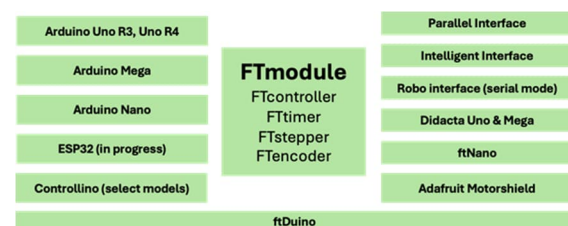


Abb. 1: Das Gesamtkonzept

Das Diagramm in Abb. 1 illustriert die Idee. Die Software-Bibliotheken können auf den in der linken Rubrik angegebenen Mikrocontrollern und auf den Interfaces oder Shields in der rechten Rubrik so laufen, dass bei einem Wechsel der Hardware nur ein Minimum an Änderungen am Programm erforderlich ist, um das fischertechnik-Modell zu steuern.

FTlegacy und FTmodule

Warum aber zwei Bibliotheken? Nun, viele verschiedene Optionen zu kombinieren kostet Platz, was für den kleinsten – aber am weitesten verbreiteten – Controller, den Arduino Uno R3, ein Speicherproblem darstellen kann. FTlegacy ist eine einfache Bibliothek, die sich ausschließlich auf die Steuerung der parallelen (Universal) und seriellen (Intelligent) Interfaces konzentriert. Als Controller kann entweder ein Arduino Uno R3/R4 oder ein Arduino Mega verwendet werden. Zusätzlich zu den grundlegenden Funktionen zur Steuerung des Interface verfügen beide Bibliotheken über Funktionen für Controller, Timer sowie Schritt- und Encodermotoren.

FTmodule ist eine erweiterte Version von FTlegacy. Sie kann neben Arduino-basierten Controllino-SPSen [8] (Abb. 2) auch die Didacta Shields (Mega und Uno) [9], das Adafruit Motor Shield und das ftNano Shield steuern. Sie läuft problemlos auf Arduino Mega und Uno R4, jedoch möglicherweise nicht immer auf dem Uno R3.

Ein Sonderfall ist der ftDuino [10]. Er ist eine Kombination aus Arduino und Input/Output Shield, die speziell für fischertechnik entwickelt wurde und eine hervorragende Lösung für alle darstellt, die mit fischertechnik in der Arduino-Umgebung arbeiten möchten. Der ftDuino verfügt über ein eigenes Ökosystem mit exzellenter Dokumentation und Beispielen und wurde

im Hinblick auf die Portabilität meiner Software in dieses Projekt integriert.

Der ftDuino bietet jedoch deutlich mehr Funktionen als FTmodule. Die Einbindung von EPS32, Arduino Due und Giga als alternative Controller ist in Arbeit.

Die Header-Dateien `FTlegacy.h` und `FTmodule.h` enthalten eine Reihe wichtiger Definitionen und Konventionen zur Standardisierung von Eingabe und Ausgabe und beschreiben außerdem die API, über die die Bibliothek verwendet wird. Zum Beispiel werden Eingangsschalter definiert als `ft_E1` zu `ft_E16`, Motoren als `ft_M1` zu `ft_M8` und einzelne Ausgangskanäle als `ft_O1` zu `ft_O16`.

Beide Bibliothekssammlungen sind auf GitHub verfügbar [11]. Die Software ist ausführlich in Referenzhandbüchern dokumentiert, die ebenfalls im GitHub-Repository zu finden sind. Da lange Listen von Softwaredefinitionen und Bibliotheken äußerst langweilig zu lesen sind, verweise ich auf diese Referenzhandbücher und behandle in diesem Beitrag nur die Grundlagen, Anwendungen und Beispiele. Ich hoffe, die Handbücher bieten ausreichend Informationen zum Programmieren. Es stehen außerdem einige Beispielskette als Grundlage zur Verfügung.

Nach der Installation der Bibliotheken in der Arduino-Umgebung, entweder im Ordner „libraries“ oder im Sketch-Ordner, kann die Programmierung beginnen.



Abb. 2: Taktstraße mit fünf modularen Einheiten (Speicher, Schweißen, Bohrer, Presse, Fräsen), die jeweils von einer Controllino-SPS angetrieben werden, die die FTmodule-Bibliothek verwendet.

Zunächst müssen ein Mikrocontroller („Controller“) und ein Shield/Interface („Legacy“ oder „Module“) definiert werden. Der Mikrocontroller-Typ wird von der Software automatisch über die Arduino-Board-Einstellungen ermittelt.

Die folgende Tabelle zeigt die Optionen für die FTLegacy- und FTmodule-Bibliotheken. Aufgrund der zahlreichen Anschlussmöglichkeiten des Arduino Uno und Mega können mehrere parallele oder serielle Schnittstellen an einen einzelnen Arduino angeschlossen werden. Die Zahlen in Tab. 1 geben die Anzahl der Schnittstellen an, die die Software für den Uno (U) oder den Mega (M) verarbeiten kann.

Verbindung der Schnittstellen

Die Didacta, Adafruit und ftNano Shields sind so konstruiert, dass sie direkt mit einem Arduino Uno, Mega oder Nano verbunden werden können. Für die parallelen (Universal) und seriellen (Intelligent) Interfaces

von fischertechnik muss ein maßgeschneidertes Shield angefertigt werden, damit diese Interfaces bequem mit dem Arduino verbunden werden können.

Vier grundlegende Shields habe ich bereits entworfen:

- Megashield 1: zwei parallele und zwei serielle Verbindungen (Abb. 3, links)
- Megashield 2: vier parallele Verbindungen
- Megashield 3: vier serielle Verbindungen (Abb. 3, rechts)
- Uno-Schild: eine parallele und eine serielle Verbindung

Es gibt etwa zehn verschiedene parallele Interfaces, die ursprünglich speziell für einen bestimmten Computer-Typ von fischertechnik entwickelt wurden [2]. Später wurden verallgemeinerte Versionen hergestellt, und die oben genannten Shields sind für diese Centronics-, Universal- oder CVK-Parallelschnittstellen konzipiert.

Interface oder Shield	Controller	Setting	FTLegacy	FTmodule
Parallel interface	Uno, Mega	PAR	U1, M4	U1, M4
Parallel interface with extension	Uno, Mega	PAREX	U1, M4	U1, M4
Serial Interface (30402)	Uno, Mega	SER	U1, M3	U1, M3
Serial Interface with extension	Uno, Mega	SEREX	U1, M3	U1, M3
Robo Interface (93293)	Uno, Mega	ROBO	U1, M3	U1, M3
Didacta Uno shield	Uno	DID_UNO	-	U1
Didacta Mega shield	Mega	DID_MEGA	-	M1
Adafruit Motorshield V2	Uno	ADA_UNO	-	U1
Controllino Mini	Uno	CONT_MINI	-	1
Controllino Maxi Automation	Mega	CONT_MAXOUT	-	1
Controllino Micro	RP2040	CONT_MICRO	-	1
Nano shield	Nano	FT_NANO	-	1
ftDuino	Leonardo	FT_DUINO	-	1

Tab. 1: Übersicht der Schnittstellenauswahl



Abb. 3: Megaschild 1 (mit/ohne Deckel) und 3

Ältere Interfaces können weiterverwendet werden, erfordern aber eine individuelle Lösung. Die Shields enthalten nicht nur die 20-poligen Header für die parallelen oder 10-poligen Header für die serielle Schnittstelle, sondern auch 6-polige Header für I2C. Zusätzlich gibt es für die serielle Schnittstelle einen RS232/TTL-Wandler, der den Arduino vor Eingangsspannungen über 5V schützt. Mehrere fischertechnik-Interfaces können wie folgt hinzugefügt werden, wobei der benutzerdefinierte Name der Schnittstelle rot ist:

```
// create interface objects. first argument
// = {PAR, PAREX, SER, SEREX, ROBO},
// second is interface number for the type.
FTlegacy interface_1 (PAR, 1);
FTlegacy interface_2 (PAR, 2);
```

oder:

```
FTlegacy taktstrasse (SER, 1);
// interface 1 for the belt and
// loading/unloading
FTlegacy frasen (SER, 2);
// interface 2 for the two milling stations
FTlegacy gantry (SEREX, 3);
// interface 3 plus extension for the
// gantry
```

oder im gemischten Modus:

```
FTmodule parallel (PAR, 1);
// first parallel interface
FTmodule serial (SER, 1);
// add a first serial interface
```

Ein Arduino Mega, verbunden mit vier parallelen Schnittstellen mit Erweiterungen, kann Eingänge von 64 digitalen Schaltern und 8 analogen Signalen aufnehmen sowie

32 bidirektionale Motoren oder 64 Ein-kanal-Ausgänge ansteuern.

Die PCB-Designs für die Shields habe ich in Gestalt von Gerber-Dateien auf Github verfügbar gemacht [11]. Ich verfüge über eine begrenzte Anzahl von PCB Shields für Interessierte unter euch. 3D-Druck-Dateien für fischertechnik-kompatible Gehäuse und Deckel zu den Mikrocontroller/Shield-Kombinationen gibt es bei Thingiverse [12]. Für fortgeschrittenere Anwendungen können die Shields mit zusätzlichen Headern ausgestattet werden, sodass alle Arduino-Pins zugänglich sind. So können zusätzliche Sensoren oder Peripheriegeräte zusätzlich zum seriellen oder parallelen Interface ergänzt werden.

Da die Entwicklungsplatinen der Mikrocontroller immer kleiner werden, können sie auch physisch in die fischertechnik Interfaces eingebettet werden. Ein frühes Beispiel findet sich in [5]. In Abb. 4 ist eine serielle Schnittstelle mit einem eingebetteten Arduino Uno Mini und einem TTL/RS232-Konverter zu sehen.

Der Arduino Uno Mini kann über den USB-C-Bus rechts mit dem Computer verbunden werden. Eine parallele Schnittstelle mit einem eingebetteten Arduino Uno wurde in einem früheren Beitrag beschrieben [6].

Das Einbetten eines Arduino Uno Mini, Arduino Pico, Raspberry Pi RP2040 oder einer kleinen ESP32-Platine in ein paralleles Interface steht weiterhin auf meiner To-do-Liste. Der Vorteil eingebetteter Mikrocontroller sind weniger Kabel und Boxen, der Nachteil ist weniger Flexibilität bei der Anzahl der angeschlossenen Schnittstellen, da nun eine Eins-zu-eins-Beziehung besteht, obwohl das ‚Extension‘-Interface weiterhin funktioniert.

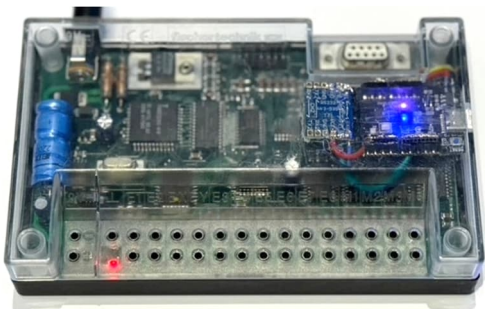


Abb. 4: Serielles Interface mit eingebettetem Arduino Uno mini

Verwendung von Displays

Es wurde ein standardisiertes LCD-Display gewählt, entweder vom Typ 2004 oder 1604, wobei die ersten beiden Ziffern die Anzahl der Spalten und die letzten beiden

die Anzahl der Zeilen angeben. 3D-Design-dateien (STL) für verschiedene fischertechnik-kompatible LCD-Gehäuse der Typen 2004 und 1604 sind auf Thingiverse [13] verfügbar. Die Displays werden über den I2C-Header mit dem Shield verbunden. Bei der Erstellung des Controller-Objekts wird das gewünschte Display ausgewählt.

```
// create controller and select optional  
// display {NONE or D1604, D2004}  
FTcontroller controller(D2004);
```

Bei Verwendung des ftDuino lassen sich die Displays problemlos über ein I2C-Kabel anschließen, und die zur Informationsanzeige genutzten FTmodule-Funktionen sind nutzbar. Abb. 5 zeigt ein Megashield 2, das an drei serielle Interfaces angeschlossen ist, darunter eine mit Extension und einem 2004-LCD, und das ein Taktstraßen-Modell steuert.

Basic Input/Output Commands

Im Folgenden gebe ich einen Überblick über einige der grundlegenden und fortgeschrittenen Befehle pro Schnittstelle. Ein vollständiger Überblick über die Programm-Bibliothek (API) findet sich im Handbuch.

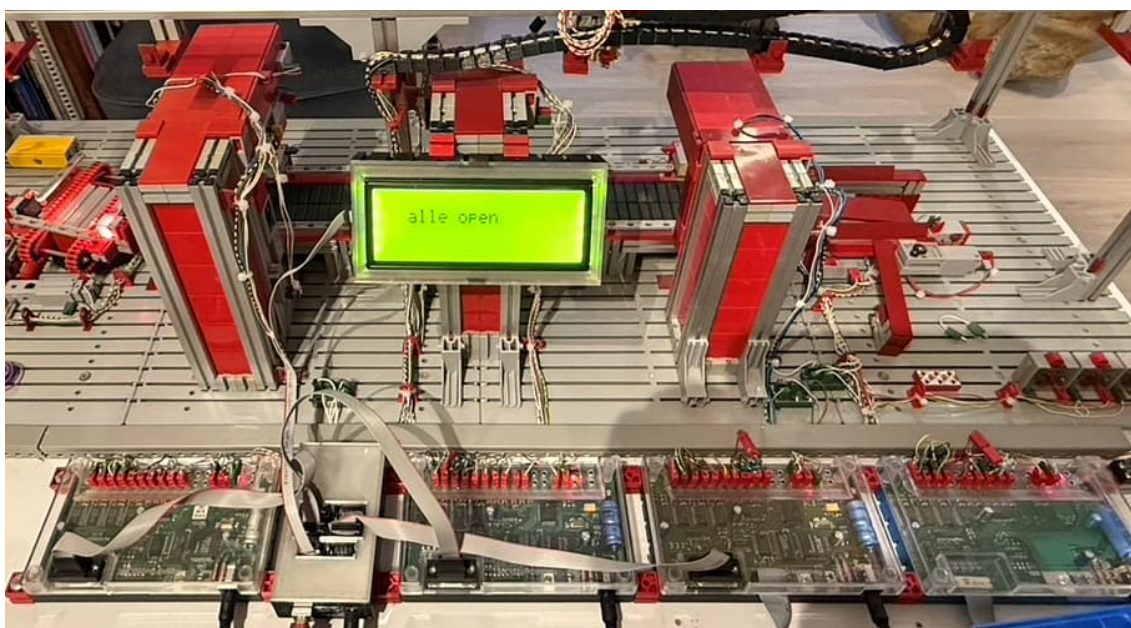


Abb. 5: LCD, verbunden mit Megashield 3

Methods - basic	Beschreibung
<code>void getInputs()</code>	Erhält alle digitalen Eingänge (E1-E16) und speichert sie im Puffer
<code>bool getInput(int E)</code>	Gibt an, ob Schalter E aktiviert wurde
<code>void getAnalogInputs()</code>	Erhält analoge Eingänge und speichert sie im Puffer
<code>int getAnalogX()</code>	Rückgabe des analogen X-Werts 0-255
<code>int getAnalogY()</code>	Rückkehr des analogen Y-Werts 0-255
<code>void setMotorCW(int M)</code>	Motor M im Uhrzeigersinn laufen lassen
<code>void setMotorCCW(int M)</code>	Motor M gegen den Uhrzeigersinn laufen lassen
<code>void setMotorSTOP(int M)</code>	Stoppe Motor M
<code>void setAllMotorsSTOP()</code>	Alle Motoren abstellen
<code>void setOutputON(int O)</code>	Aktiviere Port O (O1-O16) für unidirektionale Motoren oder Lampen
<code>void setOutputOFF(int O)</code>	Deaktiviere Port O (O1-O16) für unidirektionale Motoren oder Lampen

Tab. 2: Basic-Methoden

Methods - advanced	Description
<code>bool setMotorUntil (int M, int E, bool until, motorDirection dir)</code>	Motor M läuft in Richtung dir (CW oder CCW), bis Schalter E die Bedingung until (wahr oder falsch) zurückgibt
<code>bool setMotorUntilCount (int M, int E, bool until, motorDirection dir, int maxCount)</code>	Motor M läuft in Richtung dir (CW oder CCW), bis Schalter E maxCount Sekunden aktiv war
<code>bool setMotorUntilOrCount (int M, int E1, bool until1, int E2, bool until2, motorDirection dir, int maxCount)</code>	Motor M läuft in Richtung dir (CW oder CCW), bis Schalter E1 maxCount Sekunden aktiv war oder Schalter E2 betätigt wurde
<code>void setMotorSpeed(int M, int speed)</code>	Für Adafruit Motor Shield und ftDuino
<code>void ftMessageToDisplay(int x, int y, char *message, bool clear)</code>	Gibt message auf dem LCD an Position x,y aus; wenn clear ‚wahr‘ wird der Bildschirm vorher gelöscht
<code>void ftUpdateDisplay()</code>	Sendet Statusinformationen an das LCD für E1-E8, M1-M4 und analog-X und -Y

Tab. 3: Advanced-Methoden

Beim Adafruit Motor Shield und dem ftDuino kann man die Motordrehzahl einstellen; die genaue PWM-Einstellung hängt jedoch von der Plattform ab: Beim Adafruit Motor Shield liegt dieser Wert zwischen 0 und 255, beim ftDuino wird der Maximalwert durch die Variable `FtDuino::MAX` festgelegt, die in der von mir verwendeten Version den Wert 64 hatte.

Softwarebeispiele

Im folgenden Abschnitt werden einige Beispiele für Teile von Arduino-Sketchen vorgestellt. Das erste Beispiel bezieht sich auf das Modell „Antennenrotor“ aus dem Computing-Kit ([39499](#)), ein sehr einfaches Modell mit zwei analogen Eingängen und einem Motor. Nach der Erstellung und

Initialisierung der Objekte FTlegacy und FTcontroller werden die analogen Eingangsdaten erfasst. Basierend auf den analogen X- und Y-Werten wird der Motor in die richtige Richtung gesteuert, bis die beiden analogen Werte innerhalb einer voreingestellten Genauigkeit liegen.

```
// include the FT legacy controller classes
#include <FTlegacy.h>
// define Program name
char* programName = „Antennen Rotor“;

// create interface objects. first argument
// = {PAR, PAREX, SER, SEREX, ROBO},
// second is interface number for the type.
FTlegacy interface (PAR, 1);

// create controller and select optional
// display {NONE or D1604, D2004}
FTcontroller controller (D2004);

void setup() {
  // initiate serial output
  Serial.begin(9600);
  // start the controller and interface
  controller.begin(programName);
  interface.begin();
}

int iAX;
int iAY;

void loop() {
  int iDiff;
  // retrieve input from interfaces if any
  interface.getInputs();
  interface.getAnalogInputs();
  interface.printInputBuffer();

  iAX = interface.getAnalogX();
  iAY = interface.getAnalogY();

  iDiff = abs(iAX - iAY);
  if (iDiff > 5) {
    if (iAY > iAX)
      interface.setMotorCW(ft_M1);
    else if (iAX > iAY)
      interface.setMotorCCW(ft_M1);
    } else {
      delay(1);
      interface.setMotorSTOP(ft_M1);
    }
  interface.ftUpdateDisplay ();
}
```

Listing 1: Steuerung des Antennenrotors

Um die Flexibilität der Bibliotheken zu demonstrieren, könnte man für jeden Sketch Folgendes tun:

- Wechsle von einem Arduino Uno R3 zu einem Arduino Uno R4 oder einem Arduino Mega. Der Sketch funktioniert ohne jegliche Codeänderungen.

- Ersetze im Befehl `FTlegacy interface (PAR,1);` ‚PAR‘ durch ‚SER‘ und schließe ein serielles Interface an, wobei dieselben elektrischen Verbindungen für Schalter (E1-8) und Motoren (M1-4) erhalten bleiben. Der Sketch funktioniert ohne weitere Änderungen.
- Dasselbe gilt für das ROBO-Interface, das Didacta Uno/Megashield und das Adafruit Shield. Verwende die richtige Einstellung im FTmodule-Objekt und stelle sicher, dass Motoren und Schalter mit denselben Anschlüssen (E1-8 und M1-4) verbunden sind, dann sollte der Sketch funktionieren.

Ein zweites Beispiel ist ein Förderband mit einer Lichtschranke am Anfang, um das Band zu starten (`ft_E1`), und einer Lichtschranke am Ende, um das Band (`ft_E3`) bzw. den Motor (`ft_M3`) zu stoppen. Die Steuerung kann wie folgt programmiert werden:

```
void setup() {
  // start the controller and interface
  controller.begin(„rotating Belt“);
  interface.begin();
  // switch on lights at the beginning and
  // end of the belt
  interface.setOutputON(ft_O1);
  interface.setOutputON(ft_O3);
  delay(100);
}

void loop() {
  // retrieve input from interfaces if any
  interface.getInputs();
  // start belt by ft_E1 and stop belt by
  // ft_E3
  if (!interface.getInput(ft_E1)) {
    interface.setMotorUntil(ft_M3, ft_E3,
      false, CW);
  }
}
```

Listing 2: Steuerung eines Förderbands

Anwendungen und Modelle

Die Bibliotheken wurden in verschiedenen Modellen (Abb. 6) verwendet und getestet, darunter:

- Computing ([39499](#)). Dieses Set war das erste Computing-Set von fischertechnik und enthielt neben mehreren Basismodellen auch einen Teach-in-Roboterarm.

Die Steuerung erfolgte über die parallele Schnittstelle [14].

- Computing Experimental ([30573](#)). Dieses Set enthielt hauptsächlich fortgeschrittenere Software für den PC zur Steuerung des Modells sowie einer Schildkröte (einem Roboter mit Rädern).
- Plotter ([30571](#)) und Trainingsroboter ([30572](#)). Dies waren die damals fortschrittlichsten Modelle.
- Industrieroboter I und II ([30408](#), [96782](#)), bionische Roboter ([57486](#)) und pneumatische Roboter ([34948](#)). Diese Sets enthielten das serielle (Intelligent) Interface, die Computerschnittstelle der zweiten Generation von fischertechnik.
- ROBO Mobile Set ([93292](#))
- „Greifer“ aus dem Buch „fischertechnik-Roboter mit Arduino“ [15]
- Plan & Simulation „Taktstraße“ ([30851](#))
- Plan & Simulation „Teach-In Robot“ ([30822](#))

Die Programme für mehrere dieser Modelle stehen zur Unterstützung oder als Beispiele auf Github zur Verfügung.

Plotter

Für Schrittmotoren habe ich eine separate Klasse namens FTstepper entwickelt, die integraler Bestandteil der Bibliotheken FTlegacy und FTmodule ist. Der fischertechnik-Plotter ([30571](#)) ist ein interessantes Anwendungsbeispiel für die FTstepper-Bibliothek: Er verwendet zwei Schrittmotoren und ein Relais, die so angeschlossen sind, dass nur vier Motorausgänge benötigt werden. Ein einziges fischertechnik-Interface genügt.

Normalerweise belegt jede Spule eines Schrittmotors einen Motorausgang, sodass zwei Schrittmotoren vier Motorausgänge benötigen. Indem jeweils eine Spule jedes Schrittmotors an denselben Motorausgang der Schnittstelle angeschlossen wird, steht der vierte Motorausgang für das Relais zur Verfügung, mit dem der Plotterstift aktiviert wird.

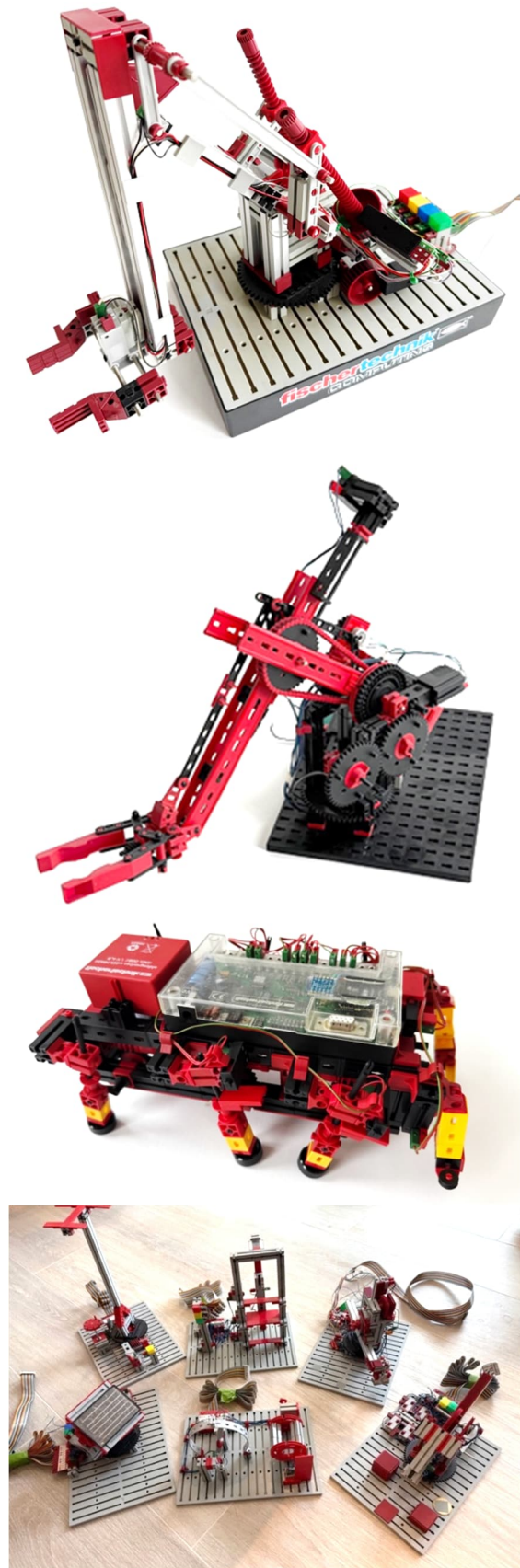


Abb. 6: Trainings-Roboter, Roboterarm, bionisches Modell und Computing Baukasten

Dies führt zwar zu leichten zusätzlichen Vibrationen, ist aber insgesamt eine clevere Lösung bei begrenzten Ressourcen. Weitere Informationen zum Anschluss eines Schrittmotors an ein fischertechnik-Interface finden sich hier [16].

In der FTstepper-Bibliothek stehen zwei Klassen zur Verfügung: eine FTstepper-Klasse für einzelne Schrittmotoren und die FTstepperXY-Klasse für Plotter oder Scanner. Die Funktionen wurden größtenteils aus der ursprünglichen Plotter-Software übernommen [17]. Es gibt Funktionen für absolute und relative Bewegungen, die Bewegung zum Startpunkt, den Schutz der Begrenzung des Zeichenfelds, das Plotten von Text und Zeichen sowie das Zeichnen von Objekten wie Linien, Rechtecken, Kreisen oder Ellipsen. Textzeichen können skaliert und in 90-Grad-Winkeln gedreht werden, genau wie in der Originalsoftware. Außerdem existiert eine Routine, um einen Graphen mit annotierten x- und y-Achsen zu zeichnen, sodass einfache xy-Graphen erstellt werden können.

Beispiel: Nachdem die Objekte des Interface (Shield) und der Controller (Arduino) erstellt wurden, wird ein Plotter-Objekt der Klasse FTstepperXY erstellt. Währenddessen sollten die Schnittstelle und die Verbindungen der Motorspulen zur Schnittstelle bereitgestellt werden. Wir sehen, dass `ft_M1` sowohl für Spule X1 als auch für Spule Y1 verwendet wird. Die Endschalter `ft_E7` und `ft_E8` sind ebenfalls vorhanden, ebenso wie die Motorverbindung (M4), die für den Plotter-Stift verwendet wird.

```
#include <FTlegacy.h>
#define PEN ft_M4
#define COILX1 ft_M1
#define COILX2 ft_M2
#define COILY1 ft_M1
#define COILY2 ft_M3
#define STOP_X ft_E7
#define STOP_Y ft_E8
// create the instances/objects
FTlegacy interface(PAR, 1);
FTcontroller controller(D2004);
FTstepperXY plotter(interface, COILX1,
COILX2, COILY1, COILY2, PEN);
```

Listing 3: Plotter-Initialisierung

Sobald der Plotter definiert ist, kann der Spaß mit dem Entwerfen eines beliebigen Diagramms beginnen. Grundlegende Plot-Anweisungen sind zum Beispiel:

```
// move to Position (250, 250)
plotter.moveToPosition(250, 250);

// draw ellips with center (400, 200)
// and radii 200 and 100
plotter.ellips(400, 200, 200, 100);
// draw circle with center (200, 200)
// and radius 100
plotter.circle(200, 200, 100);
// draw box with left lower corner
// (200, 200), length 75 and height 60
plotter.box(200, 200, 75, 60);
// draw line from current position
// to (200, 200)
plotter.line(200, 200);
// draw line from current position,
// 15 in X- and 25 in Y-direction
plotter.lineRelative(15, 25);
// plot the word „demonstration“ starting
// at 170,370 with size 4 and orientation 0
plotter.plotText(170, 370, 4, 0,
„demonstration“);
// draw an x- and y-axis with scale 0 to
// 100, in a box with plotter coordinates
// of size 400 by 300, left lower corner at
// 100, 120 and with annotated axis and 4
// tickmark intervals.
plotter.axis(0, 100, 0, 100, 100, 500, 120,
420, true, „Time(s)“, „Speed(m/s)“, „Sinus
example“, 4, 4, 7);
// curve of 90 points drawn from current
// position in plotter coordinates with
// curveX[90], an array of 90 x-coordinates
// and curveY[90], an array of 90 y-
// coordinates.
plotter.curve(90, curveX, curveY);
```

Listing 4: Beispielbefehle Plotter

Die Achsenfunktion erzeugt einen Graphen mit x- und y-Achsen. Im obigen Beispiel liegt die x-Achse von 0 bis 100, die y-Achse ebenfalls von 0 bis 100. Die Plotter-Koordinaten für diese Achsen werden durch die nächsten vier Parameter angegeben; in diesem Beispiel liegt der Ursprung des Graphen in den Plotter-Koordinaten (100, 120) und oben rechts bei (500, 420).

Achsenbezeichnungen werden angegeben, hier „Time [s]“ und „Speed [s]“. Die nächsten beiden Parameter sind die Abstände der Teilstriche auf der x- und y-Achse, in diesem Fall 4, daher erwarten wir Teilstriche bei 0, 25, 50, 75 und 100 für x-Achse und y-Achse. Zuletzt wird die Länge der Teilstriche in Plotter-Koordinaten angegeben. Das Ergebnis zeigen Abb. 7 und 8.

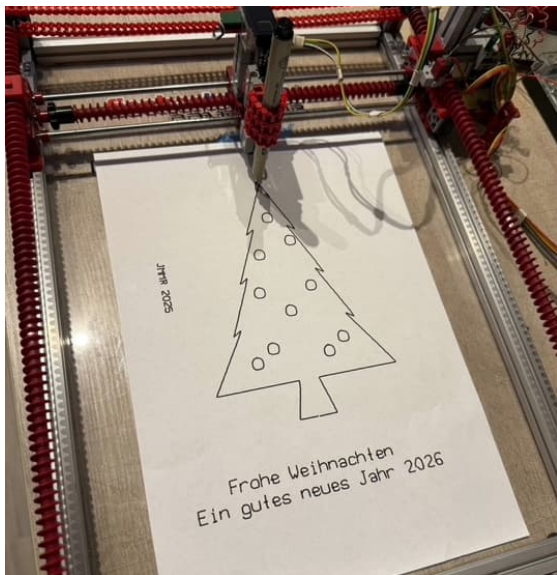


Abb. 7: Plotter-Ausgabe: Frohe Weihnachten

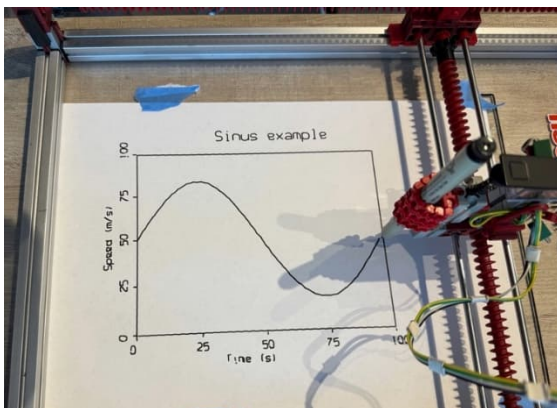


Abb. 8: Plotter-Ausgabe: Sinuskurve

Encodermotoren

Encodermotoren liefern je Achsumdrehung mehrere Impulse, die gezählt werden können, um sicherzustellen, dass der Motor an einer definierten Stelle stoppt. Auf diese Weise können Bewegungen mit großer Präzision ausgeführt werden, was für viele Modelle eine wichtige Voraussetzung ist. Um die Nutzung von Encodermotoren zu erleichtern, habe ich eine FTencoderMotor-Klasse entwickelt, die mit fischertechnik-(Encoder-) Motoren auf verschiedene Weise arbeiten kann:

- Das Encoder-Impulssignal eines Encodermotors wird mit einem der digitalen Eingänge ft_E1 bis ft_E16, Plus und Masse des Encoders mit 9V und GND

verbunden. Die Motorstromleitungen werden an den Motoranschluss am Interface oder Shield angeschlossen.

- Der Encoder-Impuls ist mit einem der Hardware-Interrupt-Pins des Arduino verbunden. Es ist wichtig sicherzustellen, dass der Encoder-Impuls niemals 5V übersteigt, da sonst der Arduino-Port beschädigt wird. Daher sollte die positive Spannung des Encoders nicht über 5V liegen.
- Ein Standard-fischertechnik-Motor, der ein Rast-Impulszahnrad antreibt, das einen Schalter auslöst, der wiederum mit einem Interface- oder Shield-Eingang verbunden ist: Das war die Methode, die in den älteren Computerrobotermodellen verwendet wurde, bevor Encodermotoren Teil des fischertechnik-Programms wurden.

Beim Erstellen eines Encoder-Objekts wird zwischen einem Standardeingang (E_STD) und der Verwendung des Hardware-Interrupt-Pin (E_INT) unterschieden. Die Software merkt sich den Ursprung und die Position des Motors. Beachte, dass die Hardware-Interrupt-Pins nicht beim ftDuino verwendet werden können, da sie dort nicht verfügbar sind.

```
int sensorPin = ft_E5;
FTencoderMotor verticalArm(&interface,
ft_M3, E_STD, sensorPin);

// Subsequently the motor can be moved in
// any direction:
verticalArm.setOrigin(0);
// 0 is set as the origin
verticalArm.moveToPosition(40);
// move to position 40
verticalArm.moveRelative(10);
// move ten extra steps
verticalArm.moveToPosition(20);
// move from position 50 to position 20
```

Listing 5: Roboterarmsteuerung

Abb. 9 zeigt ein Modell aus dem Baukasten Industry Robots II, in dem die Software vier Encodermotorobjekte verwendet, obwohl das Modell selbst keine Encodermotoren, sondern einen Leistungsmotor und drei S-Motoren mit Impulsrädern verwendet. Ein Modell mit Encodermotoren könnte dasselbe Programm verwenden.



Abb. 9: Industry-Robots-II-Modell mit simulierten Encodermotoren

Das Modell verwendet vier simulierte Encodermotoren, die wie folgt gesteuert werden:

```
#define ROTATE ft_M1
#define ROTATE_COUNT ft_E2
#define FORWARD ft_M2
#define FORWARD_COUNT ft_E4
#define UPDOWN ft_M3
#define UPDOWN_COUNT ft_E6
#define GRIPPER ft_M4
#define GRIPPER_COUNT ft_E8

char* programName = „Robot Arm“;
FTmodule interface(SER, 1);
FTcontroller controller(D2004);

FTencoderMotor verticalArm(&interface,
UPDOWN, E_STD, UPDOWN_COUNT);
FTencoderMotor horizontalArm(&interface,
FORWARD, E_STD, FORWARD_COUNT);
FTencoderMotor rotateArm(&interface,
ROTATE, E_STD, ROTATE_COUNT);
FTencoderMotor gripper(&interface, GRIPPER,
E_STD, GRIPPER_COUNT);
```

Listing 6: Steuerung der Encodermotoren

Der Vorteil der Nutzung der Encoder-Bibliothek ist deutlich weniger Aufwand, um den Motorzustand und -standort zu verfolgen. Die Tower-of-Hanoi-Beispielprogramme zeigen, wie dies möglich ist.

Timer

Die Arbeit mit Timern in einem Arduino kann lästig sein. Ich habe eine kleine FTtimer-Klasse hinzugefügt, um selbst den Überblick zu behalten und mir das Leben zu erleichtern.

```
FTtimer beltTimer(6000);
// create timer and set interval to 6
// seconds

While(go_on)
if (beltTimer.ready())
// returns true when timer interval lapses
```

```
interface.setAllMotorsSTOP();
go_on = false;
}

// Mit zusätzlichen Methoden:
beltTimer.reset(); // reset the timer
beltTimer.interval(); // set timer interval
```

Listing 7: Timer-Befehle

Einschränkungen

Der in diesem Beitrag beschriebene Ansatz unterliegt verschiedenen Einschränkungen; daher schlage ich ihn nicht als Alternative zur direkten Programmierung des ftDuino [10] oder der moderneren fischertechnik Controller (TX/TXT/TXT4/RX) vor. Diese Controller und die zugehörige ROBO Pro (Coding)-Sprache können mit einer Reihe von Sensoren, einer Kamera und Servomotoren umgehen. Bei den FTlegacy/FTmodule-Bibliotheken erfordert dies zusätzliche Programmierung und Verbindungen, zum Beispiel mit dem I2C-Schnittstellenprotokoll neben der seriellen oder parallelen Schnittstelle. Bei den Adafruit, Didakta und ftNano Shields ist das bereits der Fall.

Der ftDuino war nicht einfach zu integrieren, wahrscheinlich wegen seines bereits umfassenden „Ökosystems“. Je mehr Hardwareplattformen man berücksichtigen will, desto höher ist die Wahrscheinlichkeit einer Systemkollision. Im Fall des ftDuino betraf das drei Timer, die auch für Hardware-Interrupts und die SoftwareSerial-Bibliothek verwendet werden. Wird der ftDuino als Controller ausgewählt, werden die Hardware-Interrupts für die FTencoder-Klasse deaktiviert.

Es gibt keine grafische Programmieroberfläche. Um FTlegacy/FTmodule erfolgreich zu nutzen, muss der Benutzer grundlegende Programmierkenntnisse mit C++ in der Arduino Development Environment (IDE) haben. Für eine allgemeine Einführung empfehle ich das Buch „fischertechnik-Roboter mit Arduino“ von Fox und Püttman [15] oder die ftDuino-Dokumentation [18].

Abgesehen von diesen Einschränkungen sind die älteren seriellen und parallelen Interfaces weiterhin für eine große Modell-

palette äußerst nützlich und funktionieren weiterhin problemlos, auch mit modernen Mikrocontrollern und sogar nach 40 Jahren.

Fazit

Ich gebe nicht vor, ein professioneller Softwareentwickler zu sein, habe aber umfangreiche berufliche Erfahrungen mit dem Entwerfen, Schreiben und Testen von Anwendungssoftware gesammelt. Ein wichtiger Aspekt bei der Gestaltung und Umsetzung von Software ist ihre Lesbarkeit für diejenigen, die sie nicht geschrieben haben und sie nutzen möchten.

Man wird weder Intracte-Pointer- oder Dereferencing-Konstrukte noch Codezeilen finden, die versuchen, viele Dinge gleichzeitig zu tun. Das ist Absicht. Daher mag der (erfahrenere) Softwareentwickler unter euch den Code als umständlich oder ineffizient empfinden. Doch begrüße ich jeden Vorschlag zur Verbesserung, auch verbesserte Versionen oder Ergänzungen des Codes, damit das Produkt im Laufe der Zeit besser wird und von Vielen genutzt wird. Wenn ihr Verbesserungen vornehmt oder Funktionen hinzufügt, bitte teilt das mit der Community.

Zu den Verbesserungen, an denen ich gerade arbeite, gehören die Kompatibilität mit einer Reihe ausgewählter ESP32-Modelle, die Unterstützung für den Arduino Due und Giga, Fernsteuerungsoptionen, zusätzliche I2C-Sensor- oder Servofunktionen sowie eine Erweiterung der Liste der Beispiele.

Quellen

- [1] Jens Lemkamp: *Parallel-Interface durch Arduino gesteuert (1)*. [ft:pedia 1/2014](#), S. 24–30.
- [2] René Trapp: *V. I. P. – Ein I²C-nach-Computing-Interface-Umsetzer (Teil 1)*. [ft:pedia 2/2017](#), S. 63–73.
- [3] René Trapp: *V. I. P. – Ein I²C-nach-Computing-Interface-Umsetzer (Teil 2)*. [ft:pedia 3/2017](#), S. 57–68.
- [4] René Trapp: *V. I. P. – Ein I²C-nach-Computing-Interface-Umsetzer (Teil 3)*. [ft:pedia 4/2017](#), S. 36–49.
- [5] Dirk Uffmann: *Nutzung des Universal-Interfaces 30520 als Port-Erweiterung an einem Mikrocontroller*. [ft:pedia 2/2014](#), S. 30–35.
- [6] Jeroen Regtien: *Controlling parallel (universal) and serial (intelligent) interfaces with an Arduino*. [ft:pedia 4/2023](#), S. 69–80.
- [7] Jeroen Regtien: *Kommerzielle Arduino-Shields*. [ft:pedia 2/2025](#), S. 43–50.
- [8] [Controllino](#)
- [9] [Didacta advance](#)
- [10] Till Harbaum: *ftDuino – Open-Source trifft Konstruktions-Baukasten*. [ft:pedia 1/2018](#), S. 85–91.
- [11] Jeroen Regtien: [FTlegacy](#) und [FTmodule](#), Github.com
- [12] Jeroen Regtien: *Arduino House plus Lid for [Didacta](#), [Adafruit Shield](#) and [Megashields](#)*, Thingiverse
- [13] Jeroen Regtien: [Reference for LCD Housing and Lid](#), Thingiverse
- [14] Volker Paelke: *Der COMPUTING Baukasten – eine Retrospektive*, [ft:pedia 4/2025](#), in dieser Ausgabe.
- [15] Dirk Fox, Thomas Püttmann: *fischertechnik-Roboter mit Arduino*. dpunkt Verlag, 2020.
- [16] fischertechnik: [Hinweise zum fischertechnik-schrittmotor 32311](#).
- [17] fischertechnik: *Instructions Plotter/Scanner* ([39460](#) oder [39462](#))
- [18] Till Harbaum: [Bedienungsanleitung für den ftDuino](#). 12.01.2021.

Modell

Robocar mit Pixy2-Kamera

Arnoud van Delden

Am letzten Clubtag des niederländischen fischertechnikclubs wurde ein Wettbewerb für „Robocars“ veranstaltet: Die Fahrzeuge mussten selbstständig einer schwarzen Linie folgen. In diesem Beitrag beschreibe ich meine Experimente mit einem Robocar, den ich mit einer Kamera ausgestattet habe, die eine eigene Linienerkennungstechnik besitzt: der Pixy2-Kamera.

Einführung

Ich nutzte die Teilnahme an diesem Clubwettbewerb, um verschiedene Fahrzeuge mit unterschiedlichen Spurfolge-techniken zu bauen. Bei den Fahrzeugen mit einem Mikrocontroller konnte ich in der Software mit einer PID-Regelung experimentieren, damit das Fahrzeug so geschmeidig und schnell wie möglich auf Kurven reagiert. Bei einem Fahrzeug habe ich mehr als die üblichen zwei Infrarot-Liniensensoren eingesetzt: Mit fünf IR-Sensoren findet es sanft und schnell seinen Weg. Schließlich habe ich ein Modell mit einer Kamera mit Linienerkennung ausgestattet. Nach all diesen „Maker-Ideen“ war es an der Zeit, dem Aphorismus „In der Beschränkung zeigt sich erst der Meister“ gerecht zu werden. Deshalb bestand die letzte Herausforderung darin, einen vollständig analogen Robocar mit Silberlingen ohne Relais, aber mit einem „Kurven-Speicher“ zu bauen. Auch dieses Fahrzeug, das ausschließlich aus Komponenten aus dem (klassischen) fischertechnik-Programm besteht, ist intelligent genug, um die Linie wiederzufinden, wenn es sie verliert.

In diesem Beitrag konzentriere ich mich zunächst auf den gelben Robocar mit I2C-Kamera ganz rechts in Abb. 1; die beiden anderen werde ich in einem späteren Beitrag vorstellen.

Die Strecke

Die Strecke am Clubtag hatte keine Kreuzungen oder Abzweigungen, aber eine Herausforderung in Form eines „Kreisverkehrs“. Dass viele Fahrzeuge hier die Orientierung verlieren, ist verständlich: Die meisten verwenden wie bei den bekannten fischertechnik-Modellen nur zwei IR-Sensoren, um zu erkennen, dass das Fahrzeug von der Linie abkommt und die Fahrtrichtung korrigieren muss. Dabei ist es wichtig, dass so schnell wie möglich korrigiert wird, wenn der erste Sensor die Linie verlässt.

Verwendet man einen Mikrocontroller wie den TXT, genügt es zu wissen, welcher Sensor die Linie zuerst verlassen hat. Die Regelung in der Software kann dann versuchen, die Linie wiederzufinden, auch wenn inzwischen beide Sensoren die Linie verloren haben. Ohne einen Mikrocontroller mit einer Software-Regelung sollte möglichst vermieden werden, dass beide Sensoren die Linie verlieren. In der Praxis führt das bei scharfen Kurven dazu, dass die Geschwindigkeit des Fahrzeugs nicht sehr hoch sein darf.

Aber auch wenn mit einem Mikrocontroller eine intelligentere Regelung realisiert werden kann, bleibt ein sensibles Zusammenspiel zwischen der maximalen Fahrgeschwindigkeit und den noch korrigierbaren Kurvenradien bestehen.



Abb. 1: Spurfolger mit unterschiedlicher Technik

Auch der Abstand der Sensoren zum Drehpunkt des Fahrzeugs darf nicht zu groß sein, um ein Verlieren der Linie zu verhindern.

Die Technik

Als Mikrocontroller habe ich einen 30-poligen ESP32 WROOM32 in einem Entwicklungs- und Erweiterungsboard gewählt. Dieses Board hat den Vorteil, dass es seine 5-Volt-Stromversorgung aus dem Gleichstromeingang (5-16 Volt) selbst herstellt. Diese Versorgungsspannung steht auf der Platine an jedem GPIO-Pin zur Verfügung, sodass Sensoren, Servos usw. einfach mit einem dreipoligen Dupont-Stecker angeschlossen werden können.

Für mein Projekt war es praktisch, dass das Erweiterungsboard über einen kompakten 7,2-Volt-LiPo-Akku für den Modellbau mit Strom versorgt werden kann. Diesen Akku habe ich über einen XT30-Stecker direkt mit der gelben fischertechnik-Kassette ([32076](#)) verbunden, wie in Abb. 2 zu sehen ist. Darin sind neben einer Sicherung und einem Ein-/Aus-Schalter ein kleines Dis-

play und ein Step-up-Boost-Wandler enthalten, der die Motorspannung (10 Volt) für den L298N „Dual-H-Bridge“-Motortreiber für die beiden XM-Motoren ([135485](#)) liefert.

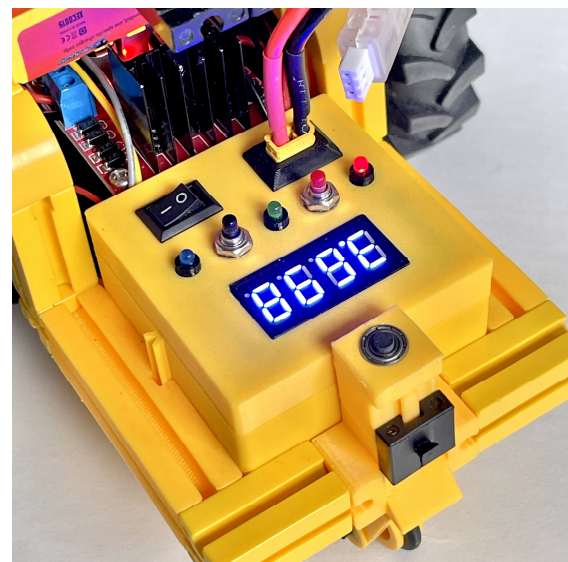


Abb. 2: Alle zusätzlichen Elektronikkomponenten sind in einer fischertechnik-Kassette untergebracht

Um einige generisch nutzbare Ein- und Ausgabemöglichkeiten zu schaffen, habe ich außerdem drei LEDs und zwei Drucktasten eingebaut. Auf einem vierstelligen 7-Segment-Display (TM1637) kann ein numerisches Feedback angezeigt werden. Während der Fahrt wird es beispielsweise verwendet, um den momentanen Fehler der PID-Regelung auf der Grundlage der gefundenen Richtungsvektoren anzuzeigen.

Auf der Rückseite des Fahrzeugs habe ich zwei kleine Potentiometer angebracht, mit denen die Motordrehzahl und der Kp-Faktor der PID-Regelung feinjustiert werden können (Abb. 3). Die eingestellten Werte werden ebenfalls auf dem kleinen Zifferndisplay angezeigt.

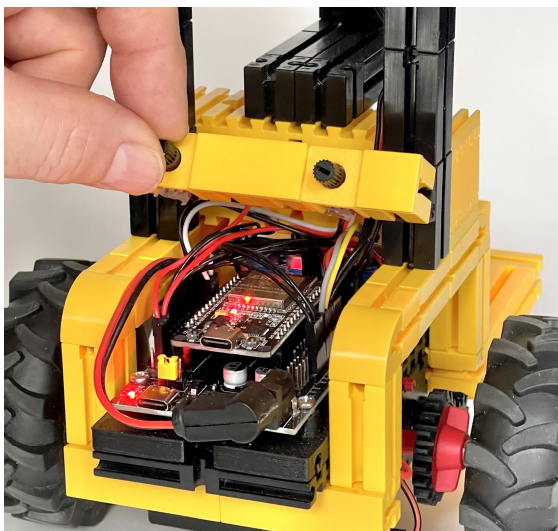


Abb. 3: Zwei Potentiometer zur direkten Feinabstimmung der Parameter

Die Pixy2-Kamera ist über die VSPI-SPI-Pins des ESP32 (MOSI=23, MISO=19, SCK=18, SS=5) an den I2C-Bus angeschlossen und wird direkt über die Erweiterungskarte mit 5 Volt versorgt. Ein weiterer Vorteil ist, dass die Pixy2 einen 3,3-Volt-I2C-Bus verwendet, sodass sie direkt an den ESP32 angeschlossen werden kann. Der I2C-Bus der Kamera ist übrigens 5-Volt-tolerant, sodass auch ein Arduino oder ein anderer 5-Volt-Mikrocontroller keine Probleme bereitet.

Die Pixy2-Kamera

Ich besaß bereits seit einigen Jahren einen „Pixy 2 CMUcam5 Smart Vision Sensor“, hatte ihn aber noch nie benutzt [1]. Der Linienfolgewettbewerb war ein guter Anlass, um zu untersuchen, ob die Richtungsvektorerkennung der Kamera schnell genug ist, um autonom den Weg über eine Linienstrecke zu finden.

In einem früheren Beitrag über einen Spurfolger mit PID-Regler in ROBOPro kam Dirk Fox 2021 zu dem Schluss, dass die Spurerkennung der fischertechnik USB-Kamera mit dem TXT bei sehr engen Kurven zu langsam reagiert [2]. Mit dem Arduino und der Pixy2 realisierte er 2020 einen schnellen Spurfolger [3].

Die Echtzeit-Bildgeschwindigkeit der Pixy2 ist bei einer I2C-Übertragung auf die Geschwindigkeit von maximal 400 kBit/s begrenzt. Die technischen Details zur ersten Version der Pixy-Kamera finden sich im ft:pedia-Beitrag von Dirk Wölffel und Dirk Fox aus 2014 [4].

Die Spezifikationen und eine Beschreibung der Funktionen des von mir verwendeten Nachfolgemodells, der Pixy2, finden sich auf der Webseite von PixyCam [1]. Diese Kamera ist kleiner, schneller und verfügt über mehr Funktionen als das Vorgängermodell, darunter einen Algorithmus zur Linienerkennung. Damit eignet sich die Pixy2 hervorragend für den Bau eines Linienfolgers [3].

Die Kamera kann auch direkt über ein USB-Kabel an den Computer angeschlossen werden. Mit der PixyMon-Software (verfügbar für Windows, MacOS und Linux) lassen sich die Kameraeinstellungen recht einfach durch Ausprobieren optimieren (Abb. 4).

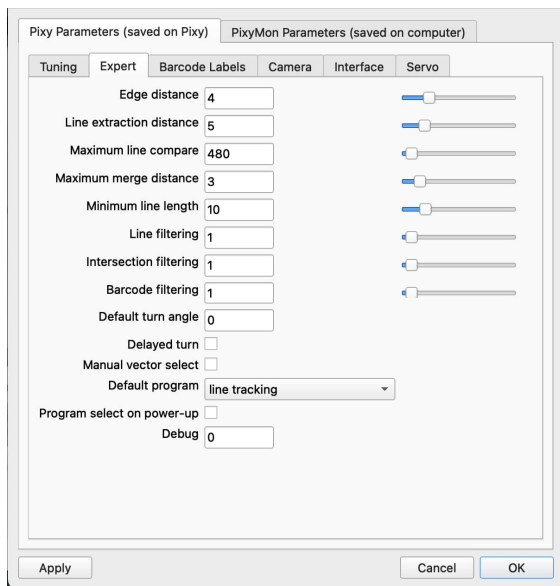


Abb. 4: Einstellung der Parameter mit der mitgelieferten PixyMon-Anwendung

Nach einigem Herumprobieren mit den Einstellungen gelang es mir schließlich, die Anzahl der fehlerhaften Erkennungen zu minimieren. Mit einer Geschwindigkeit von bis zu 60 fps wird der jeweils relevante Richtungsvektor über die Strecke ermittelt. Diese Daten können dann einfach über I2C abgefragt und im Regler verwendet werden.

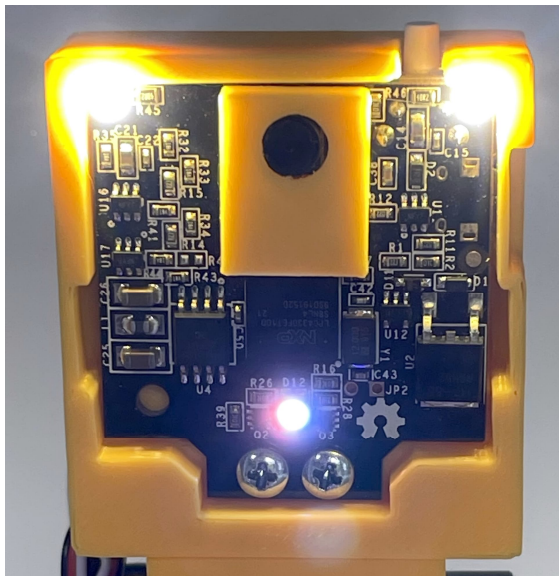


Abb. 5: Während der Fahrt wird die Beleuchtung eingeschaltet

Auf der Leiterplatte der Kamera befinden sich eine RGB-LED und zwei relativ helle Weißlicht-LEDs (Abb. 5). Durch Einschalten der gesamten Beleuchtung ist die Strecke gut ausgeleuchtet und ausreichend erkennbar.

Ich musste dafür jedoch mit dem Winkel experimentieren, unter dem die Kamera die Strecke „betrachtet“: Ein zu weiter Vorausblick führt schnell zu Vektorfehlern, wodurch das Fahrzeug zu einer benachbarten Spur „wechseln“ kann. Außerdem werden dabei die Kurven stark abgeschnitten.

Aber auch ein zu kurzer Vorausblick kann zu Problemen führen: Reagiert der Regler zu langsam, kann das Fahrzeug dabei erheblich von der Strecke abkommen. Daher habe ich einen Griff am Fahrzeug angebracht, mit dem das Fahrzeug schnell von der Strecke genommen werden kann, wenn es den kürzesten Weg zum Ausgang wählt oder Gefahr läuft, sich an einer Wand festzufahren.

Auch der Proportionalitätsfaktor (P-Faktor) der PID-Regelung musste genau abgestimmt werden. Da ich nicht jedes Mal die gesamte Firmware kompilieren und flashen wollte, habe ich an der Rückseite des Fahrzeugs zwei Potentiometer angebracht (Abb. 3): Mit dem einen lässt sich die Höchstgeschwindigkeit einstellen, mit dem anderen kann man den P-Faktor der Regelung verändern, damit das Fahrzeug so präzise wie möglich entlang der Linie fährt.

Der Prozess, auf den das Programm mit Anpassungen reagieren muss, wird in den folgenden Abb. 6 bis 10 veranschaulicht. Sie zeigen die verschiedenen Positionen des Robocars entlang einer Kurve.

Das Gehäuse habe ich auf der Grundlage des von Till Harbaum entworfenen Gehäuses für die Pixy2-Kamera entwickelt [5]. Ich habe die Seiten etwas höher gewählt und auch den Objektivhalter so entworfen, dass er in derselben Farbe wie das Gehäuse ausgeführt werden kann.

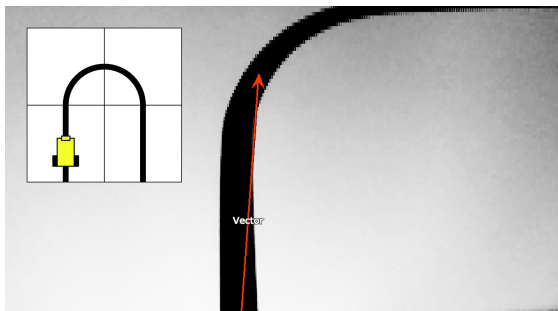


Abb. 6: Das Fahrzeug nähert sich einer Kurve, der Vektor zeigt noch geradeaus

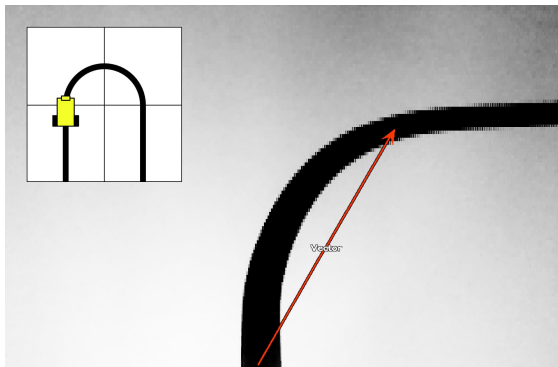


Abb. 7: Die Richtung des Vektors ist ein Maß für die proportionale Korrektur

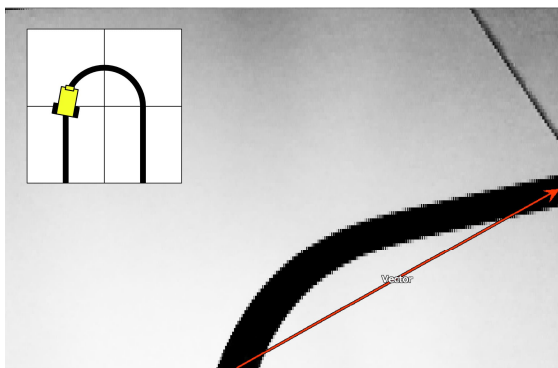


Abb. 8: Der Winkel wird steiler, die PID-Regelung verarbeitet die größere Abweichung

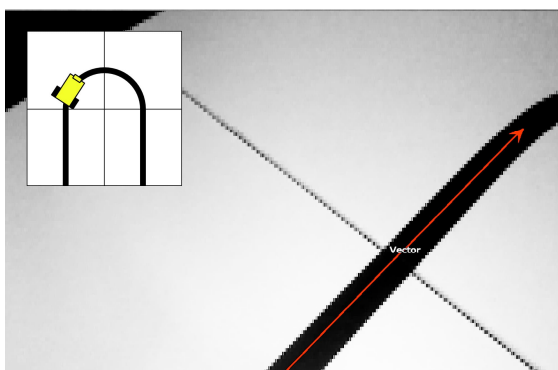


Abb. 9: Das Fahrzeug lenkt, um die Richtung des Vektors zu korrigieren

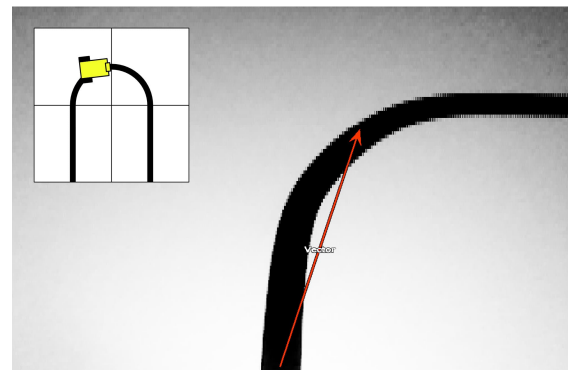


Abb. 10: Die Kurve ist genommen, der Vorgang wiederholt sich bei der nächsten Kurve

Obwohl das für die Spurfolge-Funktion nicht erforderlich ist, habe ich die Kamera mit einem Schwenk-Neige-Mechanismus mit zwei digitalen Servos aus dem Maker Kit Bionic ([571902](#)) ausgestattet, die direkt an die Pixy2-Kamera-Platine angeschlossen werden können (Abb. 11).

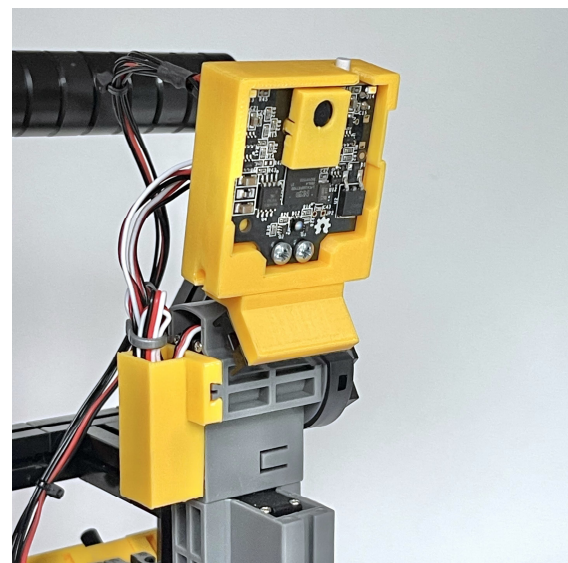


Abb. 11: Kamera auf „Tilt/Pan“-Mechanismus mit zwei Servos

Wenn die Pixy2 direkt über ein USB-Kabel an den Computer angeschlossen wird, kann man über die PixyMon-Software die Kamera mit dem Schwenk-Neige-Mechanismus ein farblich erkanntes Objekt verfolgen lassen. Für die Nutzung als Spurfolger würde eine unter einem festen Winkel fixierte Kamera natürlich ausreichen, aber für zukünftige Erweiterungen wie beispielsweise das Suchen von Objekten oder das

Ausweichen bei Hindernissen auf einer Strecke könnte der Schwenk-Neige-Mechanismus nützlich sein.

Die Software

Obwohl die Pixy2-Kamera sowohl eigenständig als auch über einen Computer verwendet werden kann, ist für mobile Anwendungen wie der Spurfolger ein Mikrocontroller erforderlich. Wie die Kamera über I2C mit einem TX-Controller, einem Arduino oder ftDuino gesteuert werden kann, wurde bereits beschrieben [3, 6]. Modernere Controller wie der TXT dürften keine Probleme bereiten, obwohl ich das nicht getestet habe. Ich habe mich für einen ESP32 entschieden, der sich einfach über die Arduino IDE in C++ programmieren lässt.

Um das Fahrzeug so sauber und schnell wie möglich entlang der Linie fahren zu lassen, habe ich eine PID-Regelung implementiert [2, 7]. Ein PID-Regler (Proportional-Integral-Differential) ist ein häufig verwendeter Algorithmus, um einen Prozesswert (in diesem Fall die Abweichung von der Linie) kontinuierlich auf einem gewünschten Sollwert (*setpoint*, die Mitte der schwarzen Linie) zu halten, indem die Abweichung (*error*) kontinuierlich berechnet wird. Es handelt sich dabei um ein rückgekoppeltes Regelsystem, das es ermöglicht, die Geschwindigkeit des Spurfolgers zu optimieren. Dabei beeinflussen der gegenwärtige Zustand (P), zurückliegende Zustände (I) und zu erwartende zukünftige Zustände (D) die Größe der vorzunehmenden Korrektur. Eine solche Regelung berücksichtigt, wenn sie mit den richtigen Parametern konfiguriert ist, die Lenkeigenschaften (Wendekreis und Trägheit) des Fahrzeugs.

Kurz gesagt besteht eine solche Regelung aus drei Teilen [2]:

- **P (proportional):** Reagiert auf die aktuelle Größe der Abweichung. Je größer die Differenz, desto stärker die Reaktion. Dieser Teil sorgt für die Hauptkorrektur,

kann jedoch eine geringe Restabweichung (statischer Fehler) hinterlassen.

- **I (integrierend):** Addiert die Abweichungen über die Zeit. Dieser Teil eliminiert den statischen Fehler, den die P-Regelung hinterlässt, sodass der Wert idealerweise schließlich genau auf den Sollwert korrigiert wird.
- **D (differenzierend):** Reagiert auf die Geschwindigkeit der Änderung der Abweichung. Dabei werden zukünftige Abweichungen berücksichtigt und der Regler so gedämpft, um ein Überschwingen zu verhindern, oder bei einer plötzlichen Änderung verstärkt.



Abb. 12: Robocar mit ESP32 und Pixy2

Bei meinen Experimenten wurde schnell klar, dass die P-Komponente den größten Einfluss hat. Obwohl ich auch die beiden anderen Komponenten in meine Regelungsregel implementierte, minimierte ich deren Einfluss, um so schnell wie möglich zu einem funktionalen Ergebnis zu gelangen, mit dem ich am Clubwettbewerb teilnehmen konnte. Die I-Komponente berücksichtigte ich nur zu einem sehr geringen Teil in

den Berechnungen, die D-Komponente ignorierte ich vorerst vollständig.

Mit ein paar Stichproben wurde deutlich, zwischen welchen Werten der Proportionalitätsfaktor K_p ungefähr liegen musste. Wenn dieser Wert zu niedrig gewählt wird, kann das Fahrzeug in scharfen Kurven die Spur verlieren; ist er zu groß, schießt das Fahrzeug ständig über die Spur hinaus und lenkt dann wieder zu scharf gegen. Auf YouTube finden sich viele Videos von Spurfolgern, die unter diesem Problem leiden.

Fazit

Die Pixy2-Kamera erwies sich als relativ einfach anzuschließen und zu konfigurieren – sie lädt zu weiteren Experimenten ein. Das Erkennen von Hindernissen auf der Strecke und ein automatisches Ausweichen ist nur eine von vielen Möglichkeiten.

Der von mir verwendete ESP32-Mikrocontroller, der mit WLAN und Bluetooth ausgestattet ist, bietet auch die Möglichkeit, den Robocar drahtlos zu überwachen oder zu steuern. Ich hatte bereits mit Python und dem GUI-Toolkit Tkinter eine kleine App für das iPhone entwickelt, mit der der Proportionalitätsfaktor K_p während der Fahrt konfiguriert werden konnte. Es stellte sich jedoch heraus, dass es einfacher war, diese einmalige Konfiguration mit einem Potentiometer am Fahrzeug einzustellen.

Die beiden anderen Spurfolger in Abb. 1 verwenden eine konventionellere Sensortechnik, um der Linie zu folgen. Bei diesen Fahrzeugen kommen Infrarotsensoren zum Einsatz. Wenn wir die verschiedenen Robocars nacheinander auf derselben Strecke fahren lassen, zeigt sich ein (erklärbarer) Unterschied zwischen diesen

Techniken: Der Robocar mit der Pixy2-Kamera antizipiert und schneidet die Kurven, während die Robocars mit den IR-Sensoren nur korrigieren, aber nicht „abkürzen“ können. Sie fahren weiträumige Außenkurven. In einem späteren Beitrag werde ich diese Fahrzeuge mit IR-Sensoren vorstellen.

Von den Robocars mit ESP32 gibt es mittlerweile ein kurzes Video auf YouTube [8]. Wer Interesse an den verschiedenen maßgeschneiderten Teilen hat, kann sich gerne bei mir melden; ich helfe dann mit den 3D-Druckdateien weiter.

Quellen

- [1] Carnegie Mellon University: [PixyCam](#).
- [2] Dirk Fox: *PID-Regler – eine experimentelle Einführung*. [ft:pedia 4/2021](#), S. 62–72.
- [3] Dirk Fox, Thomas Püttmann: *fischertechnik-Roboter mit Arduino*. dpunkt Verlag, 2021. Kapitel 3 (Spurfolger mit Kamera), S. 112 ff.
- [4] Dirk Wölffel, Dirk Fox: *I²C mit dem TX – Teil 11: Pixy-Kamera (1)*. [ft:pedia 4/2014](#), S. 43–51.
- [5] Till Harbaum: [fischertechnik case for Pixy2 camera](#). Thingiverse.com, 26.03.2020.
- [6] Helmut Jawtusich: *TX Controller und ftDuino im Vergleich*. [ft:pedia 1/2021](#), S. 114–118.
- [7] Wikipedia: [Prinzip der PID-Regelung](#).
- [8] Youtube: [Video der ESP32-Robocars](#).

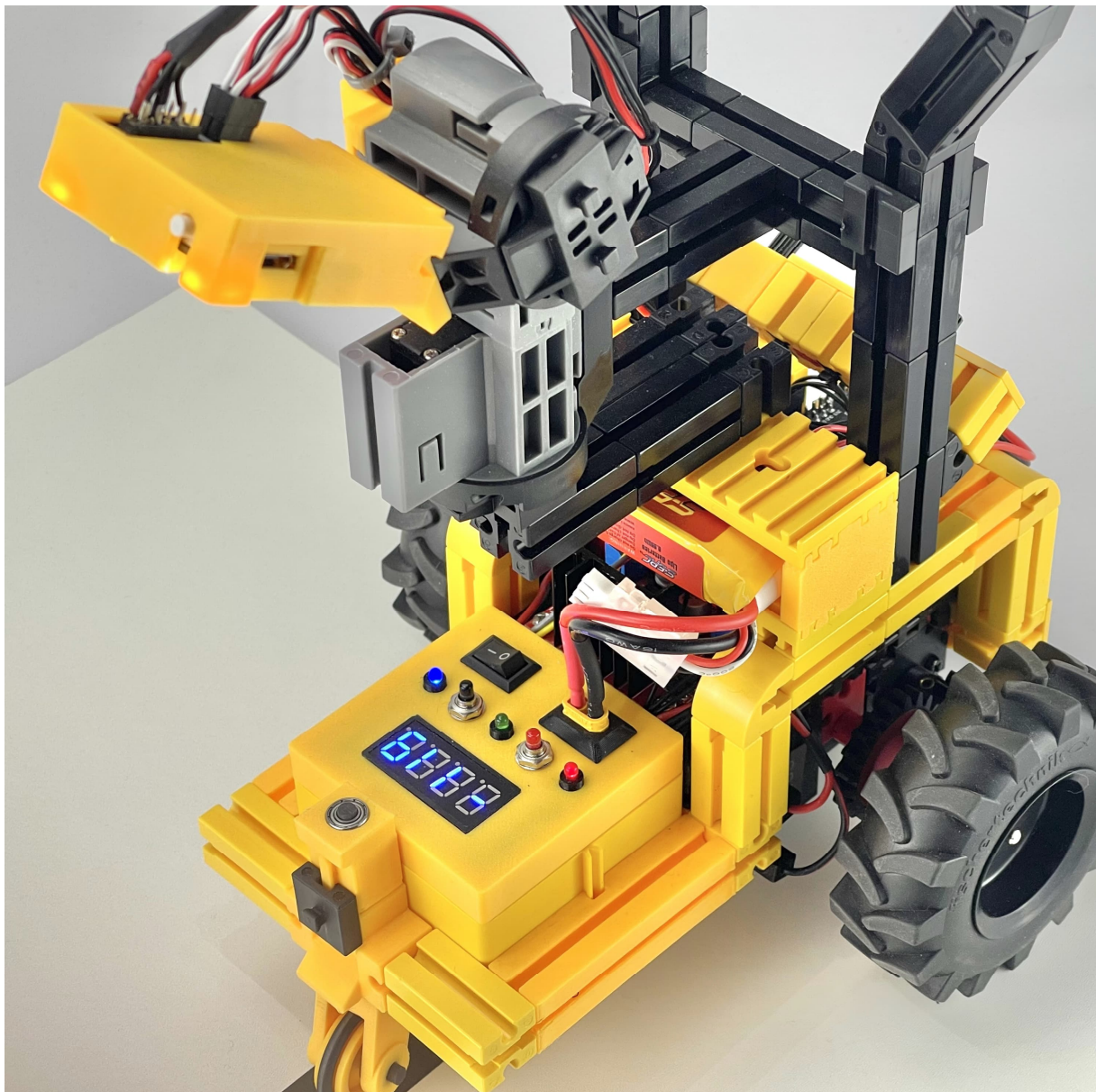


Abb. 13: Ein Blick auf den Neige-/Schwenkmechanismus und die Kassette mit der zusätzlichen Elektronik

Computing

fischertechnik-Modelle mit der Oxocard steuern – Teil 3: Der I2C-Bus

Axel Chobe

Im dritten Beitrag über die Oxocard geht es um das serielle Datenprotokoll I2C. Es wird schon seit dem TX Controller von fischertechnik unterstützt [1]. War es beim TX noch eher etwas für Tüftler, bot fischertechnik zum TXT bereits eigene I2C-Sensoren wie den Kombisensor ([158402](#)), den Umweltsensor ([167358](#)) und den Gestensensor ([186705](#)) an, mit 10poligem Stecker. Seit dem TXT 4.0 gibt es diese Sensoren auch für 6polige Stecker, und es kam der RGB-Farbsensor ([213965](#)) hinzu. Für alle diese Sensoren sind fertige Softwarelösungen erhältlich. Zusätzlich wurden von fischertechnik-Fans zu weiteren I2C-Komponenten wie Displays oder Sensoren ROBOPro-Treiber für den TXT entwickelt. Auch die Oxocard beherrscht das I2C-Protokoll, was an vier einfachen Beispielen demonstriert werden soll.

Hintergrund

Das I2C-Bus-System dient zum Aufbau und Betrieb von Geräten, für die die Anzahl der Steuerleitungen oder deren Belastbarkeit nicht ausreichen. Der I2C-Bus besteht aus vier Leitungen:

- der Vcc-Leitung (VCC)
- der Masseleitung (GND)

- der Datenleitung (SDA) und
- der Taktleitung (SCL).

Diese verbinden einen Steuercomputer (PC oder Mikroprozessor), den so genannten Master, mit einem oder mehreren Peripheriebausteinen, den Slaves [1]. Hier sind Datenspeicher, I/O-Portbausteine, AD- oder DA-Wandler, Uhrenbausteine und diverse Anzeigentreiber möglich.

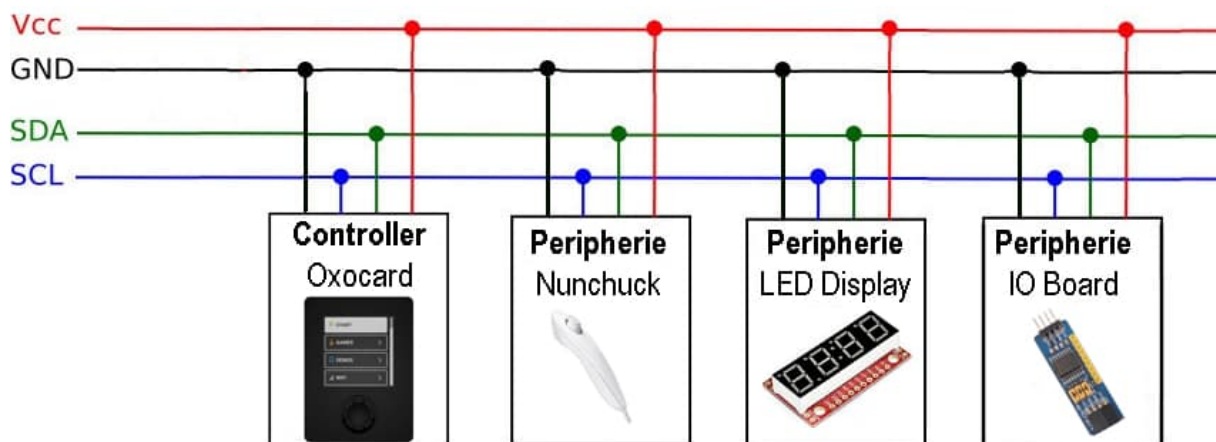


Abb. 1: I2C-Bus (schematisch)

Motivation

Seit dem TXT-Controller gab es die bereits genannten I2C-Bausteine von fischertechnik. Darüber hinaus gibt es in der Software ROBOPRO eine Vielzahl von I2C-Bibliotheken. So habe ich etliche Peripherie-Geräte ausprobiert und in fischertechnik-Modellen genutzt [2].

Da nun auch die Oxocard eine I2C-Schnittstelle besitzt, war es naheliegend, auch diese für fischertechnik-Modelle zu nutzen. Im Gegensatz zur Arduino-Programmierung gibt es bisher hier keine fertigen Bibliotheken, sondern es wird direkt mit den Registeradressen gearbeitet.

Der Umgang mit I2C ist nicht einfach. Die Datenblätter, sofern verfügbar, sind schwer verständlich. In diesem Beitrag werden deshalb nur die wichtigsten Funktionen der Sensoren vorgestellt. Darüber hinaus gibt es jeweils eine Vielzahl von Parameter, die ich hier vernachlässige.

Materielle Voraussetzungen

Als Grundlage habe ich die in meinem letzten Beitrag zu MQTT mit der Oxocard [3] vorgestellte Leiterplatte benutzt (Abb. 2).

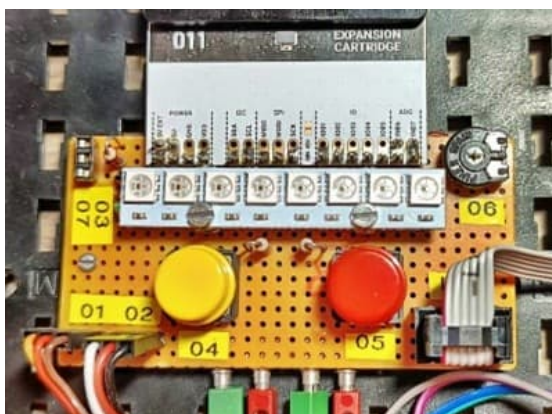


Abb. 2: Bestückte Leiterplatte aus [3]

Unten rechts befindet sich der Pfostenverbinder für die I2C-Peripherie. Sollten mehrere I2C-Elemente angeschlossen werden, kann ein Bus-Extender benutzt werden (Abb. 3).

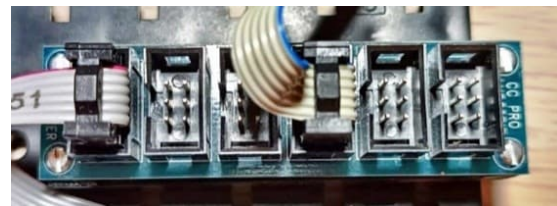


Abb. 3: Bus-Expander

Für fischertechnik gibt es eine Druckvorlage für eine Halterung dieser Erweiterung [4].

IO Board PCF8574

Mit diesem Modul kann man die sieben I/O-Ports der Oxocard um zusätzliche 8 erweitern [5]. Die auf 0x20 voreingestellte I2C-Adresse kann über die Jumper verändert werden, um weitere Module auf demselben Bus zu betreiben (Abb. 4).



Abb. 4: PCF8574

Bei maximal acht Modulen kann man so bis zu 64 zusätzliche I/O-Anschlüsse gewinnen.

Eingangszustand lesen:

```
print(checkI2CAddress(0x20))
while true:
    print(readI2CByte(0x20, 0x00))
    delay(1000)
```

Mit „checkI2CAddress“ wird geprüft, ob das Modul mit der nachfolgenden Adresse angeschlossen ist. Anschließend wird eine Dauerschleife programmiert, um die Eingänge ständig zu überwachen. Der folgende read-Befehl benötigt jeweils die Geräteadresse und die Adresse der zu überwachenden Pins.

Auf dem Terminal wird der hex-Wert des entsprechenden Einganges ausgegeben, wenn dieser Pin mit VCC verbunden wird. Durch Addition der Werte können auch mehrere I/O-Ports gleichzeitig ausgelesen werden (z. B. 33 für Port 0 und Port 5).

Ausgang ansprechen:

```
writeI2CByte(0x20, 0x00, 0x04)
```

Der write-Befehl benötigt jeweils die I2C-Geräteadresse, das Zielregister und den zu übertragenen Wert (im Beispiel Pin P2). An diesem Pin wird nun die Betriebsspannung bereitgestellt. Diese Spannung ist für fischertechnik aber zu gering, sodass z. B. ein Treiber eingesetzt werden muss.

INT	P7	P6	P5	P4	P3	P2	P1	P0
	128	64	32	16	8	4	2	1

Tab. 1: Adressierung der I/O-Pins

Der INT-PIN signalisiert eine Änderung am Eingangszustand. Ändert sich der Zustand eines Pins wird INT auf LOW gesetzt. Auf meiner Webseite habe ich ein [Beispielvideo](#) veröffentlicht.

Nunchuck

Der Nunchuk ist ein Zubehör für die Wii-Fernbedienung. Dieser besitzt einen Steuerknüppel, einen 3-Achsen-Beschleunigungsmesser sowie zwei Knöpfe (Abb. 5).



Abb. 5: Nunchuck

Normalerweise wird er an die Wii Remote angesteckt. Da das Übertragungsprotokoll I2C ist, kann es auch an die Oxocard angeschlossen werden [6, 7].

Position des Joysticks auslesen:

```
print(checkI2CAddress(0x52))
writeI2CByte(0x52, 0xf0, 0x55)
writeI2CByte(0x52, 0xfb, 0x00)
while true:
    writeI2CByte(0x52, 0x00, 0x00)
    delay(1)
    print(readI2CByte(0x52, 0x00))
    print(readI2CByte(0x52, 0x01))
    delay(1000)
```

Zuerst wird wieder geprüft, ob der Nunchuck angeschlossen ist. Die Adresse ist hier immer 0x52. Die nächsten beiden Zeilen dienen der Initialisierung des Nunchucks. Es wird nun wieder eine Endlosschleife programmiert, um die Zustände regelmäßig abzufragen. Zuvor wird die Speicheradresse (RAM-Pointer) zurückgesetzt und eine Pause von 1 ms festgelegt.

Die nächsten beiden Zeilen geben die X- bzw. Y-Position aus. Im Ruhezustand wäre das 128 und bei Vollausschlag 0 bzw. 256.

Zustand der Knöpfe abfragen:

```
print(readI2CByte(0x52, 0x05))
```

Wert 3	C und Z nicht gedrückt
Wert 2	Z gedrückt
Wert 1	C gedrückt
Wert 0	C und Z gedrückt

Tab. 2: Wert der „Knöpfe“

Diese Zeile wird in die Schleife des Programmes eingefügt. Die Auswertung erfolgt dann über die nebenstehende Tabelle.

Beschleunigungssensoren abfragen:

```
print(readI2CByte(0x52, 0x02))
print(readI2CByte(0x52, 0x03))
print(readI2CByte(0x52, 0x04))
```

Ausgegeben werden nun die Werte der X-Achse (AX), Y-Achse (AY) und Z-Achse (AZ). Diese Werte betragen im Test 75 bis 180.

Auch diese Zeilen können in die Schleife des Programmes eingefügt werden.

Wenn man den Stecker nicht abschneiden möchte, gibt es verschiedene kleine Adapterplatinen zu kaufen. Diese werden dann – wie in Abb. 6 zu sehen – mit der Oxocard verbunden. Ein [Beispielvideo](#) habe ich auf meiner Webseite veröffentlicht.

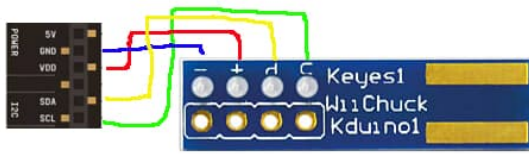


Abb. 6: Adapterplatine

LED-Display COM-11440

Das Sparkfun-Display ist eine vierstellige 7-Segmente-Anzeige, die häufig in Digitaluhren, Taschenrechnern oder Messgeräten verwendet wird [8] (Abb. 7).



Abb. 7: LED-Display

Ausgabe von Ziffern:

```
print(checkI2CAddress(0x71))
writeI2CByte(0x71, 0x00, 0x76)
writeI2CByte(0x71, 0x7a, 0xff)
writeI2CByte(0x71, 0x7b, 0x5b)
writeI2CByte(0x71, 0x7c, 0x3f)
writeI2CByte(0x71, 0x7d, 0x5b)
writeI2CByte(0x71, 0x7e, 0x6d)
```



Abb. 8: Resultierende Anzeige

Wiederum wird geprüft, ob das Display angeschlossen ist. Die Adresse ist hier 0x71. In der zweiten Zeile wird das Display gelöscht und in der dritten Zeile die Helligkeit festgelegt (0x00-0xff).

Mit den Zeilen 4 bis 8 werden die einzelnen 7-Segment-Elemente von links beginnend beschrieben. Dazu werden für die jeweiligen Registeradressen (0x7b bis 0x7e) die entsprechenden Hex-Werte eingetragen.

Im Folgenden sind die Werte der einzelnen Elemente beschrieben. Die Werte werden addiert und in Hex umgerechnet. Daraus ergibt sich folgende Tabelle für die Anzeige der Ziffern:

	Dez	Hex
Ziffer 1	06	0x06
Ziffer 2	91	0x5B
Ziffer 3	79	0x4F
Ziffer 4	102	0x66
Ziffer 5	109	0x6D
Ziffer 6	125	0x7D
Ziffer 7	07	0x07
Ziffer 8	127	0x7F
Ziffer 9	111	0x6F
Ziffer 0	63	0x3F

Tab. 3: Codierung der Ziffern

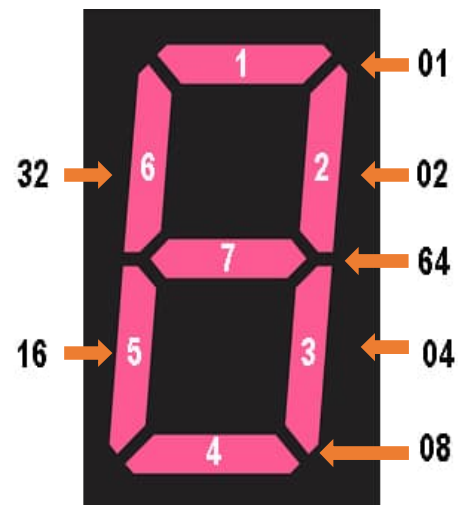


Abb. 9: Binärkodierung der Segmente

Ausgabe von Buchstaben:

Die Ausgabe von Buchstaben ist nur begrenzt möglich und erfordert etwas Phantasie. Zur Darstellung eines Zeichens müssen auch hier die Werte der einzelnen Elemente addiert und in Hex umgerechnet werden (Abb. 10).

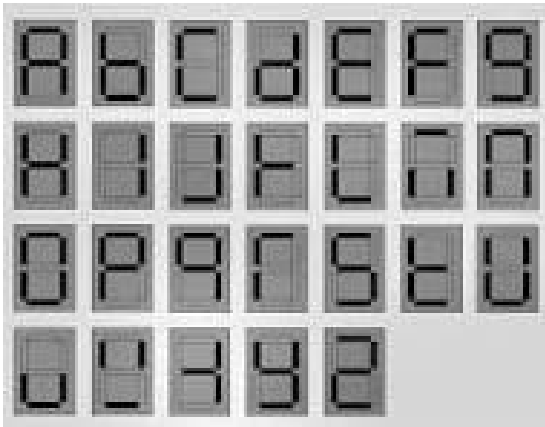


Abb. 10: Kodierung von Buchstaben

Beispiele:

S = 0x6d, t = 0x54, O = 0x3f, P = 0x50



Abb. 11: Resultierende Anzeige

Ausgabe von Dezimal- und Doppelpunkt:

```
writeI2CByte(0x71, 0x77, 0x10)
```

Im Beispiel wird der Doppelpunkt ausgegeben. Durch Addition verschiedener Hex-Werte können auch mehrere Anzeigen gesetzt werden.

Pos 0	0x01	Pos 1	0x02
Pos 2	0x04	Pos 3	0x08
Pos 4	0x10	Pos 5	0x20

Tab. 4: Kodierung der Anzeige



Abb. 12: Kodierung der Punkte

Ändern der I2C-Adresse:

```
writeI2CByte(0x71, 0x80, 0x01)
writeI2CByte(0x01, 0x81, 0x00)
```

In der ersten Zeile wird die Adresse von 0x71 auf 0x01 geändert (Adressraum von 0x01 bis 0x7e). In der zweiten Zeile wird die neue Adresse 0x01 mit einem Reset

wieder auf die Standardadresse geändert. Ein [Beispielvideo](#) habe ich auf meiner Webseite veröffentlicht.

Farbsensor TCS34725

Dieser RGB-Sensor bietet eine digitale Ausgabe von roten, grünen und blauen Werten [9] (Abb. 13). Er verfügt über einen Helligkeitssensor für die Umgebungslichtintensität und einen Interrupt.

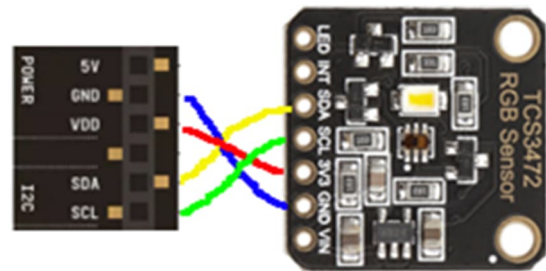


Abb. 13: Farbsensor

Eine zusätzliche LED ist standardmäßig eingeschaltet. Sie kann über den zusätzlichen Pin (LED) geschaltet werden

Lesen der Farbwerte:

```
writeI2CByte(0x29, 0x80, 0x00)
writeI2CByte(0x29, 0x80, 0x03)
while true:
    clear()
    drawText(10,20,"Rot Gruen Blau")
    drawText(10,60,(readI2CByte(0x29, 0x96)))
    drawText(80,60,(readI2CByte(0x29, 0x98)))
    drawText(150,60,(readI2CByte(0x29, 0x9a)))
    update()
    delay(1000)
```

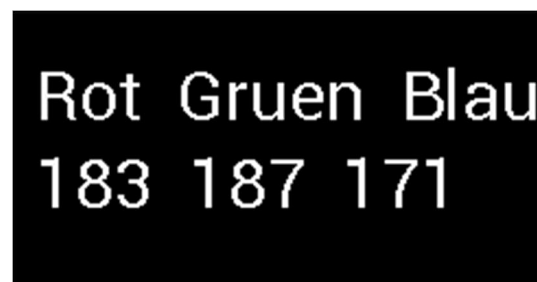


Abb. 14: Resultierende Anzeige

Die ersten beiden Zeilen dienen der Initialisierung. Dann folgt eine Dauerschleife. Der Bildschirm der Oxocard wird gelöscht und eine Kopfzeile geschrieben (Rot, Grün, Blau). In den folgenden drei Zeilen wird der jeweilige Wert der Farbe (zwischen 0 und 255) auf dem Display ausgegeben (0x96 = Rot, 0x98 = Grün, 0x9a = Blau).

Lesen des Helligkeitswertes:

```
print(readI2CByte(0x29, 0x94))
```

Mit diesem Befehl wird der Helligkeitswert auf der Konsole ausgegeben (Wert zwischen 0 und 255).

Alle gemessenen Werte sind stark abhängig vom Umgebungslicht. Deshalb muss die Auswertung jeweils individuell angepasst werden. Darum habe ich ein Gehäuse mit dem 3D-Drucker erstellt, das auf eine fischertechnik-Grundplatte geschoben werden kann (Abb. 15).

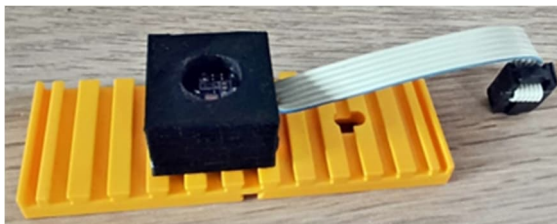


Abb. 15: Gehäuse für den Farbsensor

Die Druckvorlagen für alle vier beschriebenen Objekte stehen auf meiner Webseite zum [Download](#) zur Verfügung. Dort habe ich auch ein [Beispielvideo](#) veröffentlicht.

Fazit

Im Vordergrund steht hier der Spaß im Umgang mit den vorgestellten Sensoren, die die Möglichkeiten von fischertechnik erweitern. Es darf sich jeder aufgerufen fühlen, seine Erfahrungen mit weiteren Geräten für die I2C-Schnittstelle in der ft:pedia zu veröffentlichen.

Quellen

- [1] Dirk Fox: *I²C mit TX und Robo Pro – Teil 1: Grundlagen*. [ft:pedia 3/2012](#), S. 32–37.
- [2] <http://chobe.info/dokus/I2C.pdf>
- [3] Axel Chobe: *fischertechnik-Modelle über die Oxocard fernsteuern*. [ft:pedia 3/2025](#), S. 54–58.
- [4] Pirx99: *Fischertechnik I2C extender box*. Thingiverse.com, 06.06.2020.
- [5] Dirk Fox: *I²C mit dem TX(T) – Teil 18: Keypads und GPIO-Port-Erweiterung*. [ft:pedia 2/2019](#), S. 46–51.
- [6] Dirk Fox: *Wii-Nunchuck steuert fischertechnik-Modelle*. c't hacks, 1/2013, S. 64–69.
- [7] Dirk Fox: *I²C mit dem TX – Teil 4: Nunchuk-Fernsteuerung*. [ft:pedia 2/2013](#), S. 41–49.
- [8] Dirk Fox: *I²C mit dem TX(T) – Teil 14: LED-Display (2)*. [ft:pedia 4/2016](#), S. 84–89.
- [9] Dirk Fox: *I²C mit dem TX(T) – Teil 13: Farbsensor*. [ft:pedia 1/2016](#), S. 79–89.