

Kinderleicht: Elektronik mit **PicoZero-Modul** steuern

## Vintage trifft High-Tech

- ▶ Retro-Audio-Player selber bauen
- ▶ Vinyl-Schätze über WLAN-Lautsprecher hören
- ▶ ESP32-CAM zeigt Filmnegative
- ▶ Amiga-Disketten retten mit Arduino



## Projekte

- ▶ 10-Cent gegen Audiostörungen
- ▶ Servo-Tester ohne Mikrocontroller
- ▶ Nachgelegt: Post-Eule mit ESP32

## Werkstatt

- ▶ Analoges Speicheroszi erklärt
- ▶ Fräswerkzeuge im Eigenbau
- ▶ SMD-Recycling: Reballing-Tricks

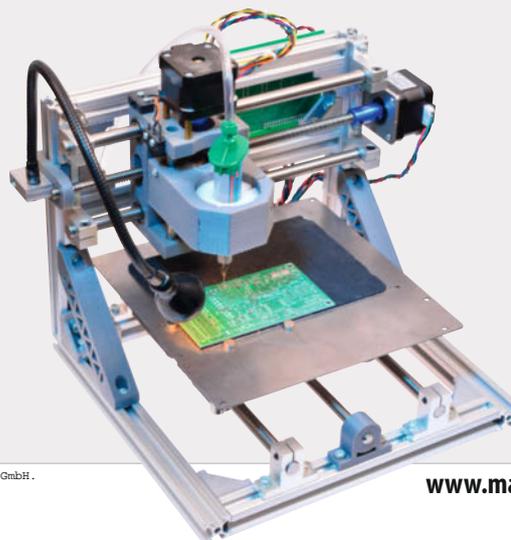
## Animatronische Wettermühle

- ▶ So pimpt man einen Modellbahn-Bausatz
- ▶ Mit ESP8266
- ▶ Zeigt Wind und Temperatur



## CNC-Fräsen-Hack

- ▶ Als Lötpastendispenser für SMD-Platinen
- ▶ Steuersoftware mit Gerber-Import



**5/22**  
 22.9.2022  
 CH CHF 25.80  
 AT 14,20  
 Benelux 15,20  
 € 12,90



# UNSER SORTIMENT VON TECHNIKERN FÜR TECHNIKER



The best part of your project: [www.reichelt.de](http://www.reichelt.de)

## Nur das Beste für Sie – von über 1.500 Markenherstellern

Unsere Produktmanager sind seit vielen Jahren bei reichelt tätig und kennen die Anforderungen unserer Kunden. Sie stellen ein breites Spektrum an Qualitätsprodukten zusammen, optimal auf den Bedarf in Forschung & Entwicklung, Instandhaltung, IT-Infrastruktur und Kleinserienproduktion sowie auf Maker zugeschnitten.



### Arduino Pro Nicla Vision – überwachen - analysieren - erkennen

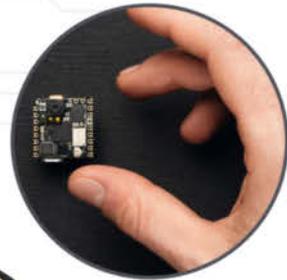
#### KI-Kameramodul mit Dual ARM CortexR M7 & M4

Nicla Vision ist eine einsatzbereite, eigenständige Kamera zur Analyse und Verarbeitung von Bildern on the Edge und eignet sich für Asset Tracking, Objekterkennung und vorausschauende Wartung.

- 2-MP-Farbkamera
- 6-Achsen-Bewegungssensor
- integriertes Mikrofon
- Abstandssensor
- mit Wi-Fi / BT
- KI und ML bei geringem Stromverbrauch

Bestell-Nr.:  
ARD NIC VISION

**98,50**



**reichelt**  
elektronik **MAGAZIN**

Arduino im reichelt Magazin:  
Arduino Projekte, Ratgeber und  
How-tos - legen Sie los!

Jetzt lesen ▶  
<https://rch.it/m-arduino>



### Arduino - Leistungsfähige Mikrocontroller für Schalt- und Steuerungsaufgaben

Ideal für alle, die interaktive Objekte  
oder Umgebungen programmieren wollen.

Gleich entdecken ▶  
[www.reichelt.de/arduino](http://www.reichelt.de/arduino)



■ Top Preis-Leistungs-Verhältnis

■ über 120.000 ausgesuchte Produkte

■ zuverlässige Lieferung – aus Deutschland in alle Welt

[www.reichelt.de](http://www.reichelt.de)

Bestellhotline: +49 (0)4422 955-333

**reichelt**  
elektronik – The best part of your project

Es gelten die gesetzlichen Widerrufsregelungen. Alle angegebenen Preise in € inklusive der gesetzlichen MwSt., zzgl. Versandkosten für den gesamten Warenkorb. Es gelten ausschließlich unsere AGB (unter [www.reichelt.de/agb](http://www.reichelt.de/agb), im Katalog oder auf Anforderung). Abbildungen ähnlich. Druckfehler, Irrtümer und Preisänderungen vorbehalten. reichelt elektronik GmbH & Co. KG, Elektronikring 1, 26452 Sande, Tel.: +49 (0)4422 955-333

TAGESPREISE! Preisstand: 29. 8. 2022

## Technik-Globuli

„Dass die Darbietung mit unmittelbarem Impetus und bis in die feinsten Details aufgefächerte, zugleich körperhaft-greifbar wie luftig geriet, war die eine Sache,“ las ich letzgens in einer einschlägigen Hifi-Zeitschrift. Weiter ging es mit: „Eine auffallend reizvolle musikalische Spannung und kribbelnde Atmosphäre ergänzten den hifidel ausgefeilten Auftritt.“ Ahnen Sie, worum es geht? Um einen Plattenspieler. Für 15.000 Euro!

Ich musste an meinen Plattenspieler für unter 100 Euro im Artikel *Schallplatten über Sonos hören* auf Seite 14 denken. Ok, bei Gleichlauf und Rumpeln (niederfrequente Störgeräusche durch den Antrieb) wird er mit dem Luxus-Modell schwer mithalten können (ich habe nicht nachgemessen). Und beim Vergleich der Tonabnehmer (2500 alleine beim Luxus-Teil) kann mein Modell mit *Audio Technica* nur die weiße Flagge schwenken. Gegen „Samarium-Cobalt-Magneten mit einer Kreuzspule aus reinen Kupferdrähten“ und einer „mikroelliptisch gebundenen Diamantstiftspitze, die für maximale Abtastung ausgelegt ist, ohne den Hochfrequenzgang oder das erhöhte Oberflächenrauschen zu beeinträchtigen,“ kommt man einfach nicht an. Oder?

Ehrlich gesagt sind aus meiner Sicht die Parallelen zwischen überdrehtem Hifi-Enthusiasmus, Homöopathie und Esoterik enorm groß. Fantastische Versprechungen, hochtrabend klingende Eigenschaften und verdreht erklärte Sachverhalte lassen so manchen halbwissenden Verbraucher Dinge hören, fühlen und sehen, die gar nicht da sind. Das betrifft neben dem Turntable sogar vermeintliche levitierte Lautsprecherkabel und geht weiter bis zum CD-Player für 8500 Euro mit impedanzoptimiertem Platinenlayout – was in der HF-Technik erst bei Frequenzen ab mehreren GHz sinnvoll ist.

Während bei Hifi-Themen der Glaube an Schlangenöl noch unter mildernde Umstände fällt, kann es für den Kauf eines USB-Sticks mit quantenholografischer Katalysator-technologie zur Wiederherstellung der Kohärenz der Geometrie der Atome (kein Scherz, gab's wirklich) wohl keine Nachsicht geben.

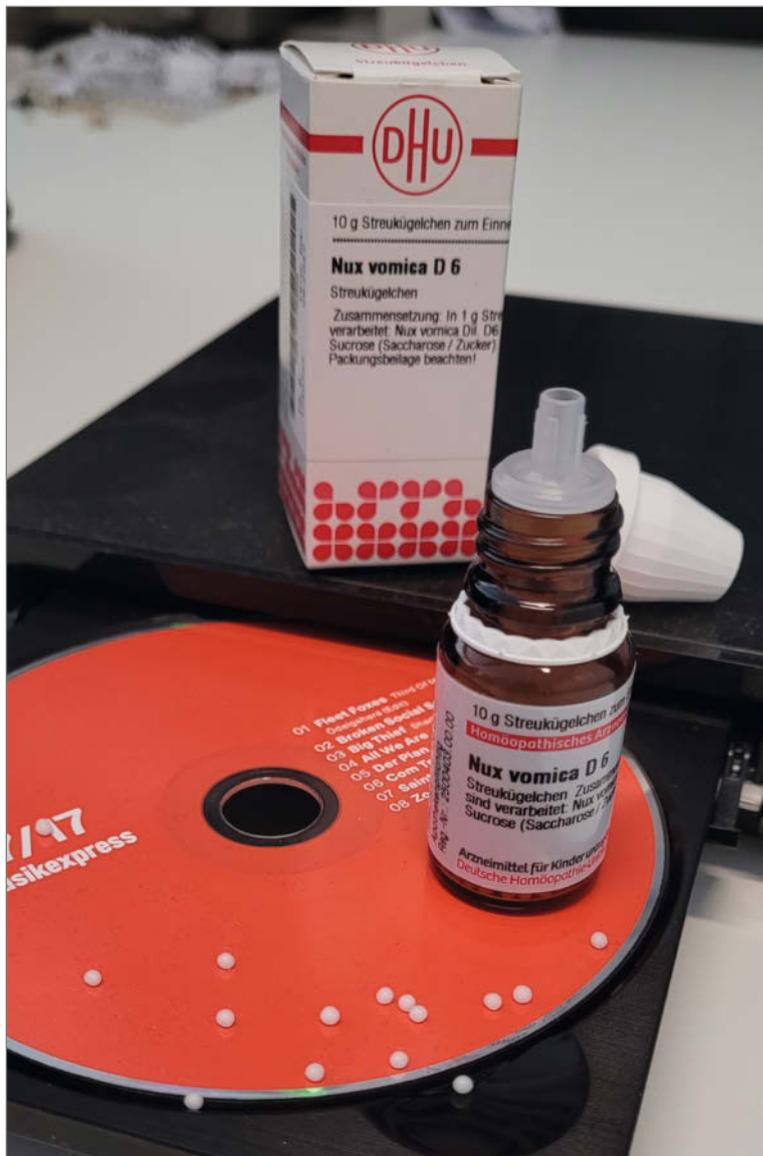
Damit Sie nicht das Nachsehen haben, versuchen wir Ihnen mit unseren Artikeln Praxis und Wissen an die Hand zu geben, um Technik besser zu verstehen und einschätzen zu können. Damit Sie beim Bullshit-Bingo zwischen Wirkstoff und Placebo besser unterscheiden können. Bleiben Sie aufmerksam!

*Daniel Bachfeld*

Daniel Bachfeld

► [make-magazin.de/xtnw](https://make-magazin.de/xtnw)

**Sagen Sie uns Ihre Meinung!**  
[mail@make-magazin.de](mailto:mail@make-magazin.de)

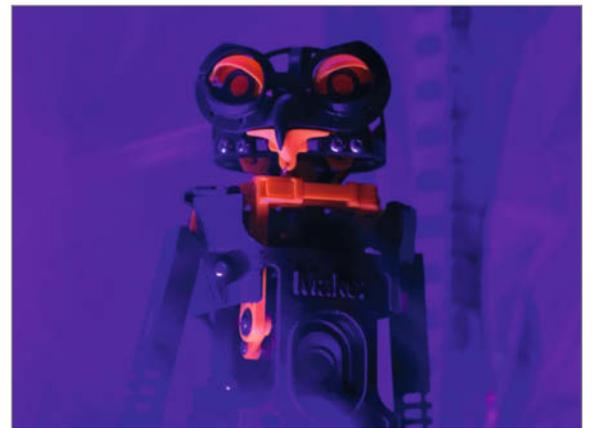


# Inhalt

## Projekte

Sie kommt wieder: Unsere animatronische Post-Eule bekam ein Update verpasst, womit die derzeitigen Lieferschwierigkeiten bei der neuronalen Cyberdyne-CPU umgangen werden. Kaum von der Chipknappheit betroffen sind dagegen unser Servo-Tester und der 10-Cent-Trick zur Störunterdrückung auf Audio-Kabeln.

- 46 „I'll be back“: Post-Eule jetzt mit ESP32
- 74 Gegen Audiostörungen: Quasi-symmetrische Signalübertragung
- 98 Servo-Tester mit Experimentierset bauen



## Vintage trifft High-Tech

Ist er nicht schick, unser MP3-Player in Gestalt eines Kassettenrekorders aus der Glam-Rock-Zeit? Statt *Sweet, Slade* und *Stones* spielt der involvierte ESP32 natürlich auch Hörbücher für die Kleinen ab. Der Espressif-Prozessor kommt ebenso in unserem Kleinbildfilm-Gucki zum Einsatz, der im Unterschied zum Original sogar Negative „entwickelt“ und in Farbe anzeigt. Weitere Vintage-Schätze können Sie mit dem Amiga-Datenretter und unserem Sonos-Plattenspieler heben.

- 10 Retro-Audio-Player selber bauen
- 16 Vinyl-Schätze über WLAN-Lautsprecher hören
- 22 ESP32-CAM zeigt Filmnegative
- 28 Amiga-Disketten retten mit Arduino

- 3 Editorial
- 6 Leserforum, Comic
- 10 Projekt: Retro-Audio-Player
- 16 Projekt: Schallplatten per Sonos hören
- 22 Projekt: ESP32-Cam zeigt Filmnegative
- 28 Projekt: Per Anhalter durch die Amiga-Disk-Galaxie
- 34 Projekt: Raspi-Super8, Teil 2
- 38 So war die Maker Faire Hannover 2022
- 40 Workshop: Elektronik steuern mit dem Pi Pico
- 46 Projekt: „I'll be back“ – Posteule mit ESP32
- 52 Community-Projekt: Mit dem Gameboy ins Internet
- 54 Community-Projekt: RC-Schlepper im Benchy-Design
- 56 Community-Projekt: Sortieranlage als Projektarbeit
- 58 Projekt: Solar-Wetterstation, Teil 2

## Werkstatt

Wenn das Budget nicht für ein neues Digital-Oszilloskop reicht, darf es durchaus auch etwas Gebrauchtes sein: Wir zeigen an einem Flohmarkt-Fund, was die Technik schon vor 50 Jahren leistete. Der Sparsamkeit hat sich auch unser Beitrag zum Selbstbau von Bohr- und Fräs Werkzeugen für die Mechanik-Werkstatt verpflichtet.

**88** Analoges Speicher-Oszilloskop erklärt

**114** Fräswerkzeuge selbst schärfen und bauen

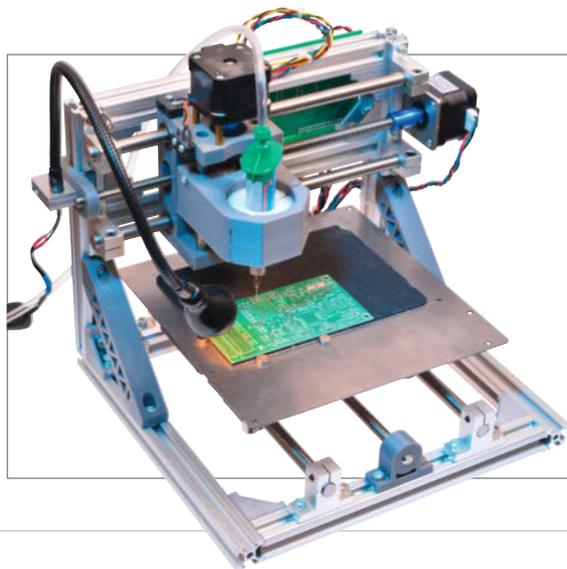


## SMD-Tricks

Das Bestücken von SMD-Platinen könnte so einfach sein – wenn doch nur nicht die Schmiererei mit Schablone und Lötpaste wäre, von den garstigen Chips im BGA-Gehäuse ganz abgesehen. Für beide Probleme haben wir praktische Lösungen parat: Damit gelingen auch Projekte mit den neuzeitlichen Miniatur-Bauteilen.

**102** Statt Schablone: Lötpasten-Dispenser

**108** Reballing und Löten von BGA-Chips



- 64** Projekt: Wetter- und Wind-Mühle
- 74** Know-how: Symmetrische Signalübertragung
- 80** Report: Elektromechanische Spielautomaten
- 88** Know-how: Analoges Speicher-Oszilloskop
- 92** Reingeschaut: Rasenmäher-Roboter
- 94** Tipps & Tricks: Lochband, Print-in-place
- 98** Know-how: Servo-Tester selber bauen
- 102** Projekt: Statt Schablone – Lötpasten-Dispenser
- 108** Workshop: Reballing und Löten von BGA-Chips
- 114** Projekt: Fräswerkzeuge selbst schärfen und bauen
- 118** Kurzvorstellungen: Power Station, 3D-Scanner, Labornetzteil, 3D-Drucker, CNC-Oberfräse und mehr
- 122** Impressum, Nachgefragt

Themen von der Titelseite sind rot gesetzt.

## Animatrische Wettermühle

Alles dreht sich, alles bewegt sich: Unsere Holländer-Windmühle zeigt Windrichtung und -stärke originalgetreu anhand der aktuellen Wetterdaten an – nicht nur auf einer Modellbahn, sondern vielleicht auch auf Ihrem Schreibtisch. Mit Hilfe eines ESP8622 und Getriebemotoren animieren wir den klassischen Fallerbausatz.

**64** Wetter- und Wind-Mühle



# Leserforum

## Falsche Firma

Pi Pico mit WLAN, Make 4/22, S. 9

Das Funkmodul CYW43439 im Pico W stammt von *Infinion* und nicht von *Intel*, wie im Artikel steht. Wir bitten um Nachsicht für dieses Versehen.

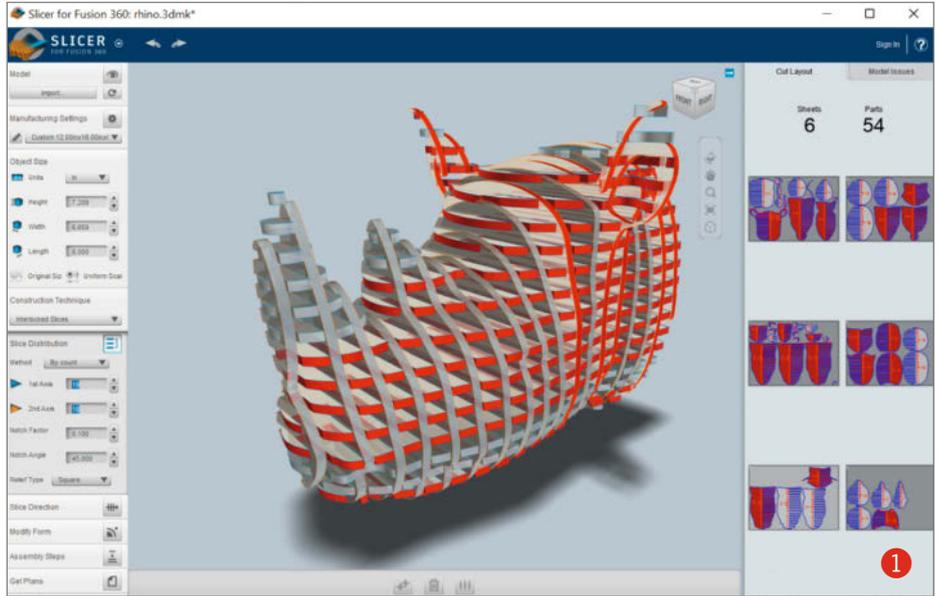
## Klare Worte

Editorial: Wolkiges Angebot, Make 4/22, S. 3

Danke für die klaren Worte bezüglich *123D* und den verwirrenden Angaben durch Autodesk, mit denen ich mich damals und auch aktuell wieder in der Freizeit konfrontiert sah und sehe.

Da hatte ich mich noch gefreut, auf einer Festplatten-Partition die Installationsdatei für *123D Make* in einem alten Download-Ordner gefunden zu haben. Nach der gewohnt einfachen Programm-Bedienung scheiterte der Export jedoch an der von Autodesk gekappten Serververbindung ... Mir geht es in erster Linie um die Funktion *Interlocked Slices*, wie es in *123D Make* benannt wurde: Durch ineinander gesteckte, in Kontur geschnittene Platten auf zwei kreuzenden Achsen ergibt sich eine dreidimensionale Figur. Habt ihr einen Tipp, mit welchen Programmen oder Kombinationen so etwas möglichst komfortabel möglich ist?

Christian Schönberg ▶ [make-magazin.de/xuz2](http://make-magazin.de/xuz2)



*Tatsächlich gab es für 123D Make mit „Slicer for Fusion“ ein Nachfolge-Tool. Das klang zwar so, als wäre es ein Modul, das man aus Fusion 360 benutzt, aber das war irreführend, es handelte sich um ein Stand-alone-Programm. Zwar wird auch das mittlerweile nicht mehr weiter entwickelt. Aber: Es gibt die Software noch bei Autodesk zum Download, und zwar hier:*

## Lieber grob informiert als unwissend

Leserbrief von Matthias Zwerschke in Make 4/22, S. 6, zu Eigenbau-Geigerzähler mit Alarm, Make 3/22, S. 8

Herr Zwerschke zweifelt die Nützlichkeit eines solchen Messgerätes in seinem Leserbrief ziemlich vehement an. Ich jedenfalls wäre froh gewesen, wenn ich damals zu Zeiten von Tschernobyl die mögliche Radioaktivität der Sandkiste unserer Kinder oder der Lebensmittel hätte messen können. Das Fleisch der Rentiere ist immer noch – wenn überhaupt – nur eingeschränkt essbar. Wenn also im Rahmen der Kriegshandlungen oder der Atomenergienutzung der nächste Meiler leck wird, werde ich froh sein, zumindest eine grobe Einschätzung der Strahlenbelastung vornehmen zu können und eine Handlungsgrundlage zu haben. Und das so etwas passieren wird, ist sicher – offen ist nur, wann.

Peter Danisch

## Offener Brief an FreeCAD

Gratis-3D-CAD für Maker, Make 4/22, S. 76

Erstmal danke an das FreeCAD-Team für das mächtige Tool, dass Sie uns kostenlos zur Verfügung stellen. Laut vielen Quellen kann man damit richtig coole Ergebnisse produzieren. Das macht Lust.

## Kontakt zur Redaktion

Leserbriefe bitte an:

[heise.de/make/kontakt/](http://heise.de/make/kontakt/)

Wir behalten uns vor, Zuschriften unter Umständen ohne weitere Nachfrage zu veröffentlichen; wenn Sie das nicht möchten, weisen Sie uns bitte in Ihrer Mail darauf hin.

Sie haben auch die Möglichkeit, in unseren Foren online über Themen und Artikel zu diskutieren:

[www.make-magazin/forum](http://www.make-magazin/forum)

 [www.facebook.com/MakeMagazinDE](http://www.facebook.com/MakeMagazinDE)

 [www.twitter.com/MakeMagazinDE](http://www.twitter.com/MakeMagazinDE)

 [instagram.com/MakeMagazinDE](https://www.instagram.com/MakeMagazinDE)

 [pinterest.com/MakeMagazinDE](https://www.pinterest.com/MakeMagazinDE)

 [youtube.com/MakeMagazinDE](https://www.youtube.com/MakeMagazinDE)

## Korrekturen

Manchmal unterläuft uns ein Fehler, der dringend korrigiert gehört. Solche Informationen drucken wir weiterhin auf den Leserbriefseiten im Heft, aber seit Ausgabe 1/17 finden Sie alle Ergänzungen und Berichtigungen zu einzelnen Heft-Artikeln auch zusätzlich über den Link in der Kurzinfo am Anfang des jeweiligen Artikels.

Ich habe ihr Tool probiert, habe viele historische YouTube-Videos zu historischen FreeCAD-Versionen und auch ein User-Lexikon gefunden. Habe dann versucht, damit ein einfaches 3D-Objekt zu erstellen – aber viele fundamentale Funktionen, die in historischen Dokumentationen beschrieben sind, funktionieren heute nicht mehr. Das macht Frust.

Details liefere ich Ihnen gerne. Hier kurz und anschaulich, wie ich FreeCAD erlebe.

Wenn FreeCAD Autos bauen würde: Dann wäre das Lenkrad mal rechts, mal links, mal ein Joystick in der Mitte und in Zukunft ... Die Bremse würde bei einem Versionsprung nicht mehr funktionieren, etc. Bei Autos ist das User Interface immer gleich: Lenkrad, Bremse, Gas, all die wichtigen Funktionen sind

1. da zu finden und

2. so zu bedienen, wie User es seit Generationen gewohnt sind.

Das Fahrzeug wird unter der Haube regelmäßig erneuert und der avancierte User/Instandhalter möchte da *up to date* sein. Der Fahrer nicht.

Zurück zu FreeCAD. Ich möchte Sie um zwei Dinge bitten:

**Erstens** eine abgespeckte Einsteigerversion / Einsteiger-Workbench (mit der Langlebigkeit von Ubuntu LTS). Diese

– ist über viel Jahre lang gleich: YouTube-Tutorials sind viele Jahre lang gültig.

– enthält ausschließlich die nötigsten Funktionen, um ein Teil zu zeichnen.

– enthält: Neues Projekt, neuer Quader, Taschen, Polster, Maße, PDF- und DWG-Export.

Ja, sie darf gerne Bugfixes und wenige wirklich wichtige neue Funktionen bekommen. Nein, alte Funktionen dürfen nicht entfernt werden.

**Zweitens** ein YouTube-Video, das den Workflow erklärt, wie ich mit der Einsteiger-

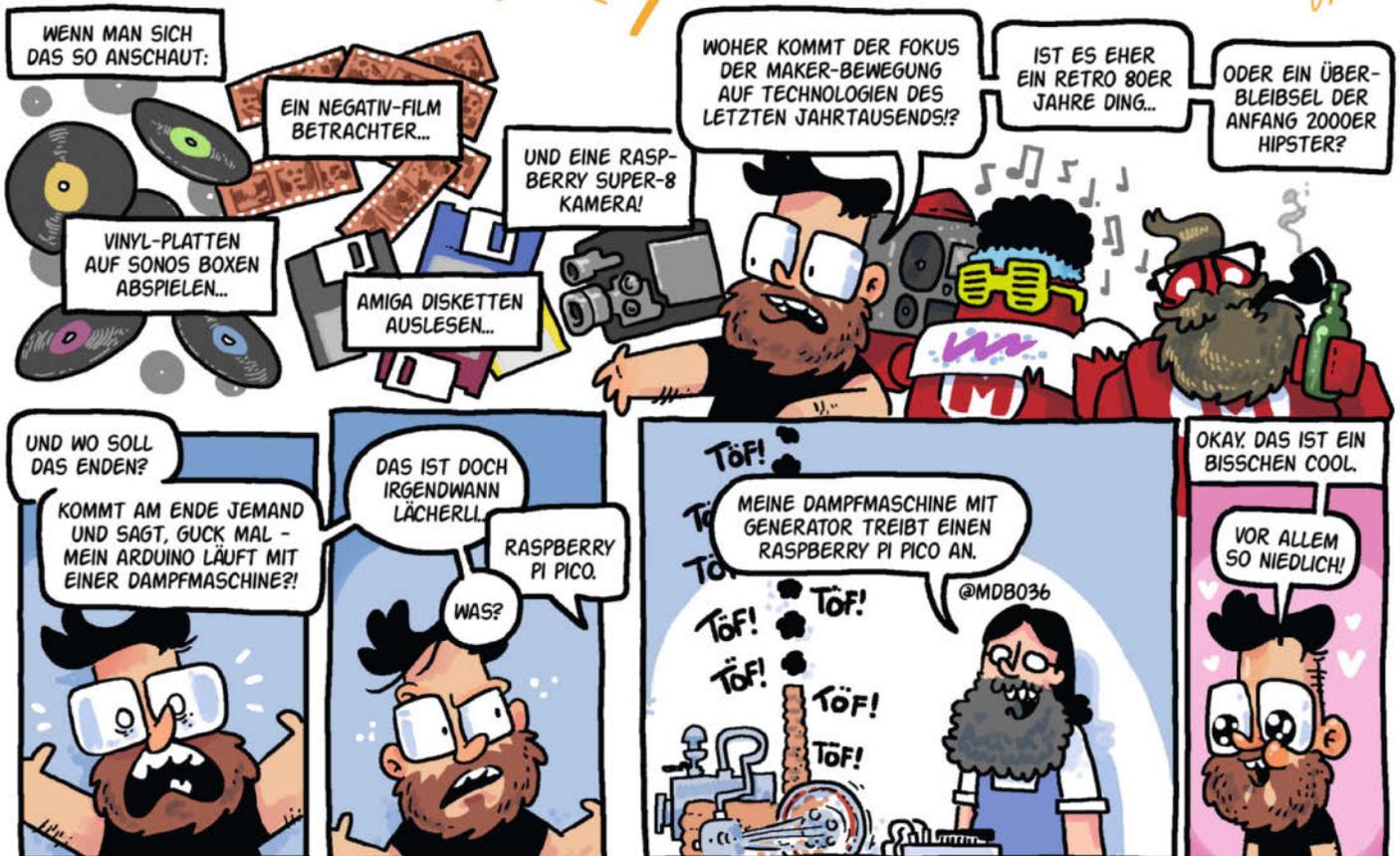
version ein einfaches Objekt erstelle. Dabei soll es alle wichtigen Funktionen behandeln, um Objekte vom 3D-Druck bis zu Häusern zu zeichnen. Es darf umständlich sein und darf mit den anderen Workbenches viel schneller gehen. Aber wichtig ist: Einsteiger möchten nachvollziehen können, wie sich Entwickler den Gebrauch ihrer Software vorstellen. Jeder noch so kleine Schritt ist nachvollziehbar für den User, speziell jene Schritte, die für Entwickler selbstverständlich sind und an denen ein Einsteiger zerschellt. Und bitte als Erstes die FreeCAD-Factory Settings resetten.

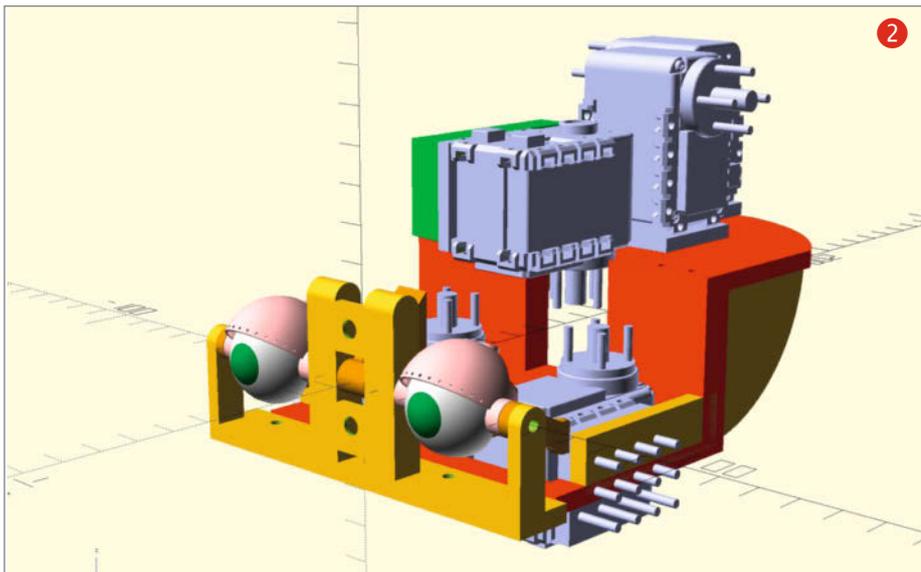
Die Einsteiger-Workbench hat die gleiche Aufgabe wie ein gutes Kinderspielzeug: Sie soll Lust auf mehr machen. Das avancierte FreeCAD-Werkzeug, bei dem bei jedem Versionsprung erlernte Funktionen nicht mehr funktionieren und welches mir ohne Tutorial

## 51 Retro Making

# Kolophonium

von und mit @beetlebum





## Warnhinweis für den Solartisch

Der Solartisch und die grüne Steckdose, Make 4/22, S. 10

**Achtung:** Befinden sich (besonders bei alten Modulen) keine *Bypass*-Dioden zwischen den Strängen der Solarzellen im Modul, so können Teilverschattungen zu Hotspots führen, die das Modul beschädigen können. Ist das bei Ihrem Modul der Fall, sollten Sie den Solartisch im Stromerzeugemodus komplett abräumen und, sobald er als Tisch benutzt wird und in der Sonne steht, mit einer lichtdichten Tischdecke schützen. Mehr dazu siehe:

► [make-magazin.de/xuz2](https://make-magazin.de/xuz2)

vor die Füße geschmissen wird – macht vor allem Frust.

Ich freue mich, wenn Sie Leute wie mich mit ins Boot nehmen.

Andreas Gurk

## Nicht gerecht

Dankeschön für die tolle Übersicht der kostenfreien CAD-Programme. Ich finde aber, dass nur „für einfache Objekte“ und ein Minus bei der CNC-Eignung in der Tabelle *OpenSCAD* nicht gerecht werden. Ich selbst konstruiere seit Jahren alle meine 3D-Druck- und CNC-Objekte damit und bin nach wie vor begeistert. Und da sind auch durchaus komplexere Modelle dabei 2.

Klar, man kann damit nicht durch visuelles Klicken konstruieren – aber gerade das finde ich wunderbar: Ich muss mir z.B. keine Gedanken machen, ob eine Fläche die andere be-

rührt – der Code und die Zahlen geben das eindeutig vor. Zudem kann man bei Bedarf auch organischer entstandene Formen über STL und andere Formate importieren.

Daniel Springwald

*Das Minus bei CNC trägt OpenSCAD – ebenso wie Blender –, weil ja gekrümmte Oberflächen aufgrund der unterstützten Exportformate nur facettiert ausgegeben werden.*

## Nicht frostfest

Solarstrom für Mikrocontroller, Teil 1, Make 4/22, S. 32

Sie verwenden in diesem Projekt einen MPPT-Laderegler mit Solarmodul, um einen LiPo-Akku zu laden. Der von Ihnen verwendete MPPT Laderegler verfügt gemäß *Waveshare* nicht über eine Temperatursicherung.

LiPo- sowie Lilon-Akkus-sollten generell unter 0°C nicht geladen werden – dies schädigt die Akkus unwiderruflich (leider habe ich kein Datenblatt zu dem von Ihnen verwendeten Akku gefunden). Obwohl dem Akku täglich ca. 1000mAh entnommen werden, behaupte ich, fällt in einem normalen Winter in D/A/CH die Temperatur des Akkus auf unter 0°C.

Es gibt eine einfache Abhilfe, indem man beispielsweise einen Bimetall-Temperaturschalter an den Akku anbringt. Dann kann beispielsweise darüber der Minuspol der Solarzelle unterbrochen werden, sollte die Temperatur zu tief sein. In diesem Fall ist der Akku geschützt, wird jedoch nicht geladen. Alternativ könnte man auch eine Heizung (Widerstand) an den Akku anbringen, welche den Akku durch das Solarpanel erwärmt, sollte dieser unter 0°C kalt sein.

Matias Meier

## Beach-Office statt Home-Office

Die Idee zu einem Solartisch hatte ich auch schon. Eine transportable Power Station mit 1,5 kWh habe ich bereits gebaut. Neben Notebook und Smartphone kann man damit auch noch eine Kühlbox betreiben. Ein klappbarer Solartisch wäre dann eine logische Erweiterung zum Beach-Office.

Ich habe das aber wieder verworfen, weil die Sache einen riesengroßen Nachteil hat. Man kann an dem Solartisch nicht sitzen und arbeiten, da er in der prallen Sonne steht. Mein Beach-Office besteht damit aus Power Station, Klappstuhl, Stuhl, Sonnenschirm und separat auslegbaren Solarpanels. Dazu Smartphone und Notebook als Arbeits-equipment und kühle Getränke fürs Wohlbefinden ...

Carsten Meyer

## Geht noch einfacher

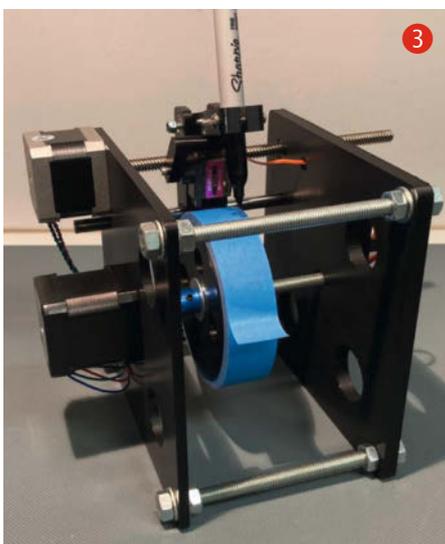
Community-Projekt: Masking-Tape-Etikettendrucker, Make 4/22, S. 98

Mit Interesse habe ich den Bericht über den *Masking-Tape-Etikettendrucker* gelesen. Es geht jedoch noch einfacher, mit einem Arduino Nano und einem CNC Shield V4 3. Alle Dateien und die Bauanleitung gibt es bei [Printables.com](https://printables.com) (siehe Link).

Klaus Hassheider

*Danke für den Hinweis auf Ihr Projekt und herzlichen Glückwunsch, dass Sie damit den Sharpie-Mods-Wettbewerb auf Printables gewonnen haben.*

► [make-magazin.de/xuz2](https://make-magazin.de/xuz2)





SHAPER

# STUDIO

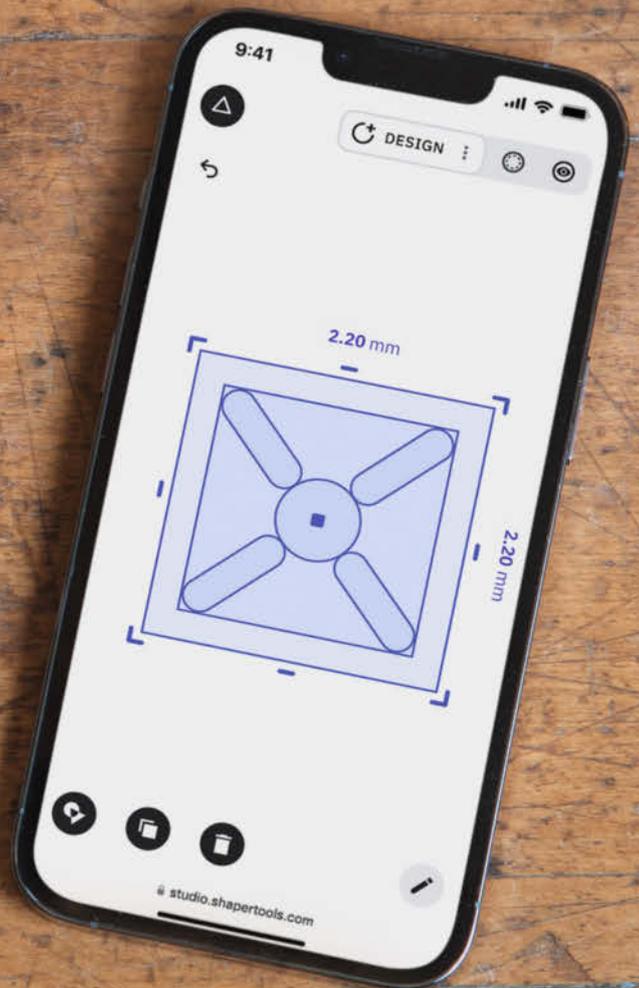
Die intuitive Zeichensoftware  
für dein Handwerk.

Zum fertigen Design in nur wenigen Klicks.  
Mit Studio erstellst du schnell fertige Projektdaten  
im SVG-Format – egal ob für Fräsarbeiten mit Origin,  
deinen Lasercutter oder Schneideplotter.



Setze deine Ideen mit einfach zu bedienenden  
Zeichen- und Layoutwerkzeugen in die Realität um.  
Kombiniere Grundformen, füge Text hinzu und  
skaliere in reellen Maßeinheiten.

[shapertools.com/studio](https://shapertools.com/studio)



Mehr erfahren:



# Retro-Audio-Player

Dieser selbstgebaute Kassettenrekorder kommt ganz ohne Kassetten aus: Hinter dem Retro-Design verbirgt sich ein ESP32, der Hörbücher und Musik von einer SD-Karte abspielt.

von Maximilian Kern



In Sachen intuitiver Bedienung war an den Kassettenspielern der 1970er bis 1990er wenig auszusetzen: Eine Taste zum Starten der Wiedergabe, eine zum Stoppen und zwei weitere zum Vor- und Zurückspulen. Selbst die Abspielposition in Hörbüchern blieb beim Ausschalten bestehen, ganz ohne Software. Heute sind solche Geräte natürlich keine Option mehr, sie sind zusammen mit ihren Kassetten längst aus den Geschäften verschwunden. Aber lässt sich die Optik und Haptik der Kassettenrekorder trotzdem auf moderne Hardware übertragen? Aus dieser Überlegung heraus ist ein ESP32-basierter MP3-Player entstanden, der an seine historischen Vorbilder angelehnt ist und dennoch einige moderne Funktionen zu bieten hat.

## Design und Aufbau

Zu Beginn des Projekts standen schon ein paar grundsätzliche Ideen fest, die umgesetzt werden sollten. Dazu gehörte in erster Linie ein möglichst einfaches Bedienkonzept mit mechanischen Tasten anstelle eines Touchscreens. Bei den Tasten entschied ich mich für *Cherry MX Blue*, die normalerweise in mechanischen Tastaturen eingesetzt werden. Mit ihrem großen Hub und dem schwarzen Kunststoffgehäuse kommen sie den mechanischen Schaltern in echten Kassettenspielern am nächsten. Die Tasten sind zudem standardisiert, sodass sich leicht passende Tastenkappen finden lassen. Als Lautstärkeregler kommt ein Dreh-Encoder zum Einsatz, der zusammen mit den drei Tasten das minimalistische Bedienkonzept vervollständigt.

In nahezu allen Kassettenspielern erlaubt eine durchsichtige Scheibe den Blick auf das sich drehende Band und zeigt so direkt an, in welchem Zustand sich der Spieler gerade befindet. Beim modernen Nachbau gibt es natürlich kein echtes Tape mehr. Stattdessen werden die Spulen virtuell auf einem Display angezeigt. Das ist funktional nicht zwingend erforderlich, trägt aber wesentlich zur authentischen Retro-Optik bei. Das hierzu nötige Display bildet ein 2,8-Zoll-Farb-LCD mit SPI-Schnittstelle. Die Auflösung von 320×240 Pixeln reicht völlig aus, um die Tape-Animation und einige Textblöcke anzuzeigen.

Damit auch die Stromversorgung dem 20. Jahrhundert treu bleibt, setzt das Projekt auf vier AA-Batterien bzw. Akkus, die vom Nutzer ausgewechselt werden können. Für die Tonausgabe wählte ich einen flachen Mono-Lautsprecher aus. Eine Audio-Qualität wie aus einem teuren Bluetooth-Lautsprecher kann man damit nicht erwarten. Für höhere Ansprüche kommt deshalb noch ein 3,5mm-Klinkenstecker für externe Lautsprecher mit in das Gehäuse.

Durch die Auswahl der Tasten, des Displays, der Batterien und des Lautsprechers ergaben

## Kurzinfo

- » Musik- und Hörbücher dank ESP32 in Retro-Optik genießen
- » Intuitive Bedienung mit mechanischen Tasten und Farbdisplay
- » Automatisches Speichern der Wiedergabeposition

## Checkliste

-  **Zeitaufwand:**  
6 bis 8 Stunden zzgl. 3D-Druck
-  **Kosten:**  
60 bis 100 Euro
-  **Löten:**  
SMD-Lötfahrung ist hilfreich
-  **3D-Druck:**  
180g Filament
-  **Programmieren:**  
Arduino IDE für ESP32
-  **Elektronik:**  
Grundkenntnisse

## Material

- » WaveShare 2,8" LCD (320 × 240 Pixel)
- » Batteriehalter für vier AA-Batterien
- » 3 Cherry-MX-Tastenkappen von Sparkfun
- » 4 M3 × 8mm Inbusschrauben
- » 7 M3 × 5mm Inbusschrauben
- » CUI CDS-40304 Mini-Lautsprecher
- » Zweilagige Leiterplatte von Aisler
- » Die vollständige Teileliste befindet sich im GitHub-Repository des Projekts.

## Werkzeug

- » 3D-Drucker Bauraum mindestens 18cm × 18cm × 18cm
- » Inbusschlüssel 1,5mm
- » LötKolben mit Lötzinn
- » Schlitzschraubendreher
- » Pinzette zum SMD-Bestücken

## Mehr zum Thema

- » Florian Schäffer, Lizenz zum Löten, Make Sonderheft 2017, S. 78
- » Gustav Wostrack, Der Weg zur Platine, Teil 1, Make 6/21, S. 104
- » Carsten Wartmann, Bartop Arcade mit Raspberry Pi, Make 4/20, S. 14

Alles zum Artikel  
im Web unter  
[make-magazin.de/x12s](https://make-magazin.de/x12s)



1 Das CAD-Modell des Gehäuses entstand in Autodesk Fusion 360.



2 Ein Schnapphaken verriegelt den Deckel des Batteriefachs.

sich die Größe des Gehäuses (ca. 18cm lang) und der innere Aufbau. Beim Design wollte ich den tragbaren, flachen Kassettenrekordern möglichst nahekommen, wie man sie zum Beispiel aus Interviews und Verhören im Fernsehen kennt. Der Bildschirm sitzt hierzu mittig im Gehäuse, darunter direkt das Gitter des Lautsprechers. Die Tasten sind darüber angeordnet, sodass sie bündig mit dem Display abschließen. Um die drei Tasten und den Dreh-Encoder herum verläuft ein Griff, an dem man den Rekorder tragen kann. Auf der Rückseite befindet sich schließlich eine Klappe zum Wechseln der Batterien.

Nachdem die Form grob feststand, musste das Design noch verfeinert werden. Mein erster Entwurf war noch etwas zu eckig. Nach ein wenig Recherche stieß ich auf den *Philips 113*, einen Plattenspieler aus den 70ern, dessen geschwungene Form ich für den Griff und für das Farbschema übernahm 1. Das daraus resultierende Gehäuse besteht aus zwei großen Halbschalen, die durch vier M3-Schrauben zusammengehalten werden. Damit die 3D-gedruckten Bauteile auch authentisch nach Spritzgussteilen aussehen, sind ein paar Details eingearbeitet: Wo beide Hälften aufeinanderstoßen, verläuft eine dicke Trennlinie.



3 Tasten und SMD-Bauteile passen zusammen auf eine Leiterplatte und so in das kompakte Gehäuse.

Dazu kommen Formschrägen an allen Seiten. Um das Gehäuse auch aufstellen zu können, befinden sich zwei kleine Rippen unterhalb des Lautsprechers. Der Deckel des Batterie-fachs wird mithilfe eines Schnapphakens verriegelt 2.

## Elektronik

Bedingt durch die Chipkrise ist die Auswahl an Mikrocontrollern noch immer stark eingeschränkt. Glücklicherweise ist mit dem *ESP32* von *Espressif* noch ein sehr leistungsfähiges Modul beschaffbar. Eigentlich bedarf der *ESP32* in der Maker-Community keiner Vorstellung mehr – dank WLAN, Bluetooth, Dual-Core und Arduino-Kompatibilität findet man das kostengünstige Modul in immer mehr Hobbyprojekten.

Ein besonderer Vorteil des *ESP32* für dieses Projekt ist dessen integrierte *I2S*-Schnittstelle. *I2S* steht für Inter-Controller-Sound (nicht zu verwechseln mit *I2C*) und ermöglicht es, Audiodaten digital vom Mikrocontroller zu einem externen Verstärker zu übertragen. Hierzu nutzt *I2S* drei Leitungen: Daten (*SD*), Takt (*SCK*) und Word Select (*WS*), wobei Letzteres nur zur Unterscheidung des linken und rechten Audiokanals dient. Um Töne auszugeben, fehlt also zwischen Lautsprecher und *ESP32* nur noch der Verstärker. Hier kommt der *MAX98357* des Herstellers *Maxim* zum Einsatz. Mit 3W hat er ausreichend Leistung, um den kleinen integrierten Lautsprecher zu betreiben. Einziger Nachteil: Der Verstärker hat nur einen Mono-Ausgang. Beim Anschließen von Kopfhörern an die 3,5mm Klinke muss man also auf Stereo-Sound verzichten. Auch diese Bauteilwahl ist leider der Knappheit von passenden Stereo-ICs geschuldet. Tatsächlich kann der *MAX98357* sogar zweimal verbaut werden, wobei ein IC den linken und ein anderes den rechten Audiokanal übernimmt. Die Treiber unterstützen jedoch nicht, dass die beiden angeschlossenen Lautsprecher denselben Massebezug haben, wie es bei den meisten Klinkensteckern der Fall ist. Wer das Projekt nachbauen möchte und auf Stereo nicht verzichten kann, muss also auf einen Stereo-Verstärker wie beispielsweise den *UDA1334A* von *NXP* wechseln.

Von Audiogeräten ist man gewohnt, dass die Wiedergabe automatisch zum Kopfhörer wechselt, sobald der Klinkenstecker angeschlossen wird. Wie aber kann der *ESP32* zwischen internem und externem Lautsprecher umschalten? Ich hatte schon ein paar komplizierte Lösungen im Kopf, bis mir auffiel, dass dieses Problem schon vor Jahrzehnten gelöst wurde. Die 3,5mm Buchse ist so aufgebaut, dass die Ringe im Stecker durch Federelemente kontaktiert werden. Ist kein Stecker angeschlossen, liegen die Federn auf zusätzlichen Kontakten auf, die mit dem internen Laut-

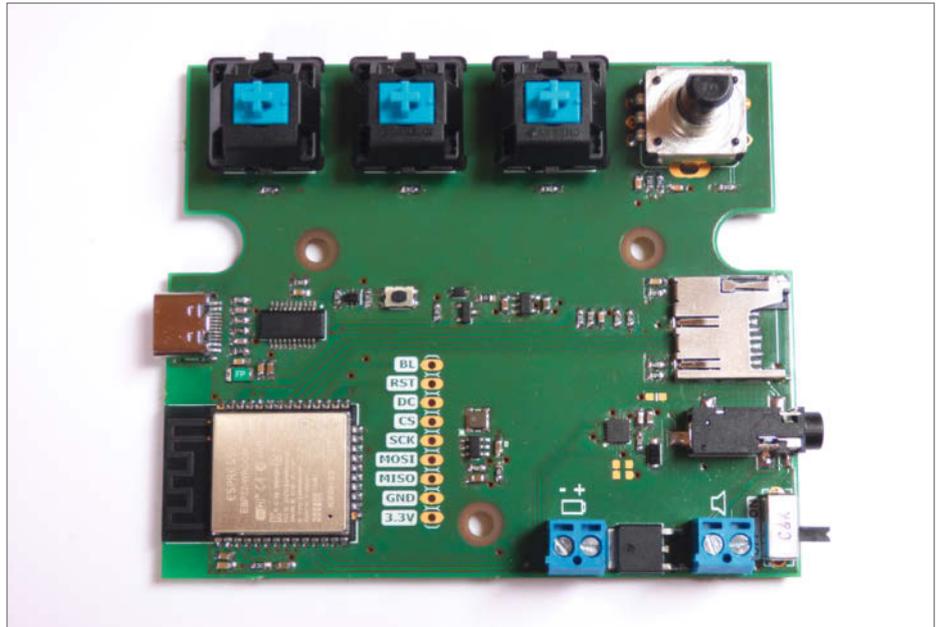
sprecher verbunden sind. Der Ton wechselt also ganz automatisch auf den richtigen Ausgang. Lediglich, um die Lautstärke individuell für Lautsprecher und Kopfhörer einstellen zu können, ist eine Steckererkennung nötig. Hierfür verwende ich einen Pull-Up Widerstand am Ausgang des internen Lautsprechers zusammen mit einem Kondensator und Widerstand als Tiefpass, um das analoge Audiosignal herauszufiltern. Das gefilterte Signal wird dann vom ESP32 eingelesen und zeigt an, ob ein Klinkenstecker angeschlossen ist.

Da der ESP32 nur einen 4MB Flash-Speicher besitzt, braucht es noch einen separaten Speicher für Audiodaten. Hierfür bieten sich SD-Karten an, da sie preisgünstig und leicht auszulesen sind. Denn obwohl SD-Karten ein eigenes SD-Bus-Protokoll nutzen, sind sie dennoch zum ganz gewöhnlichen SPI kompatibel. Um an die Daten zu kommen, braucht es also nur einen SD-Kartenslot, der an die SPI-Schnittstelle des ESP32 angeschlossen ist.

Zu diesem Zeitpunkt der Entwicklung wurde eines klar: Eine eigene Leiterplatte muss her, sonst passen die Einzelteile kaum in das Gehäuse hinein **3**. Außerdem sollten USB-Buchse, Klinkenstecker und SD-Karte von außen über ordentlich angeordnete Öffnungen im Kunststoff erreichbar sein. Damit für die von außen zugänglichen Schnittstellen und für die Tasten samt Dreh-Encoder nicht mehrere Leiterplatten nötig sind, habe ich all diese Bauteile auf gleicher Höhe angeordnet **4**. Dort, wo die Tasten befestigt werden, ragt die Leiterplatte aus dem Gehäuse heraus. Hinzu kommt noch ein kleiner Schiebeshalter zum Ein- und Ausschalten mit einer 3D-gedruckten Kappe, um ihn leicht bedienen zu können. Auf der gegenüberliegenden Seite der ESP32 und sein USB-C-Anschluss zur Programmierung. Es lohnt sich, die Antenne des Moduls nach außen auszurichten und von Kupfer auszusparen. Dann können später auch drahtlose Funktionen mit einer guten Reichweite genutzt werden.

Damit sich die SD-Karte und das Display nicht dieselbe Schnittstelle teilen müssen, ist das LCD an einen separaten SPI-Bus angeschlossen. Hinzu kommen noch mehrere Signale, die etwa für die Ansteuerung der Hintergrundbeleuchtung benötigt werden. Zusammen mit allen anderen Funktionen reizt das die verfügbaren Pins des verwendeten ESP32-WROOM-32 Moduls schon fast vollständig aus.

Da das Gerät über vier AA-Batterien mit Strom versorgt wird, akzeptiert es auch einen recht weiten Bereich als Eingangsspannung. Alkaline-Zellen sind zwischen ca. 1V bis 1,5V nutzbar, was zusammen 4V bis 6V entspricht. Da die verwendeten Komponenten unterschiedliche Spannungsbereiche haben, entschied ich mich für die folgende Lösung: Der ESP32



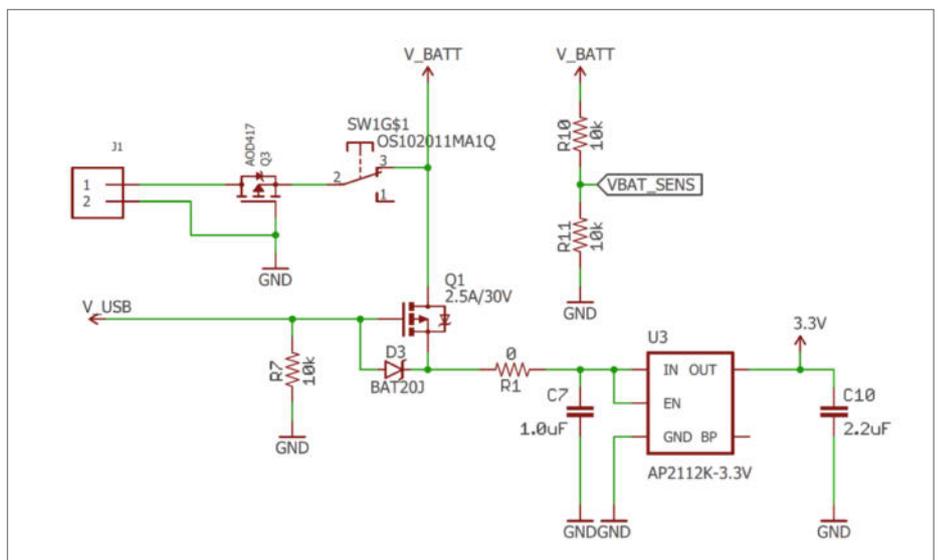
**4** Auf der zweilagigen Leiterplatte ist reichlich Platz für alle Bauteile.

wird über einen 3,3V Spannungsregler versorgt. Der Verstärker-IC bezieht seine Spannung jedoch direkt aus den Batterien. Eine Silizium-Diode in Reihe verhindert, dass bei frischen Batterien die maximale Eingangsspannung überschritten wird. Das Display wiederum wird über einen MCP1640-Step-Up-Konverter von Microchip mit stabilen 5V versorgt. Falls die Batterien doch einmal falsch herum eingelegt werden, dient ein MOSFET am Batterieanschluss als Verpolschutz **5**. Die Spannung der Batterien ist zur Überwachung auf einen Analogeingang des ESP32 gelegt. Ein Spannungsteiler sorgt dafür, dass der Messbereich des Analog-Digital-Konverters (ADC) nicht überschritten wird.

### Programmierung

Obwohl der ESP32 seine eigene Entwicklungsumgebung mit sich bringt, gestaltet sich das Programmieren über die beliebte Arduino IDE noch etwas einfacher. Der Controller muss lediglich über den Boardverwalter installiert und die Verbindung über USB zum Computer hergestellt werden. Schon kann die Software geschrieben und getestet werden.

Das Abspielen von MP3-Files als Hauptfunktion ist wiederum keine einfache Sache. Daten müssen von der SD-Karte geladen, dekodiert und über das I2S-Interface ausgegeben werden. Dank frei verfügbarer Bibliotheken muss man jedoch nicht komplett bei



**5** Geregelte Stromversorgung: Das P-MOSFET leitet nur, wenn V\_USB frei ist.

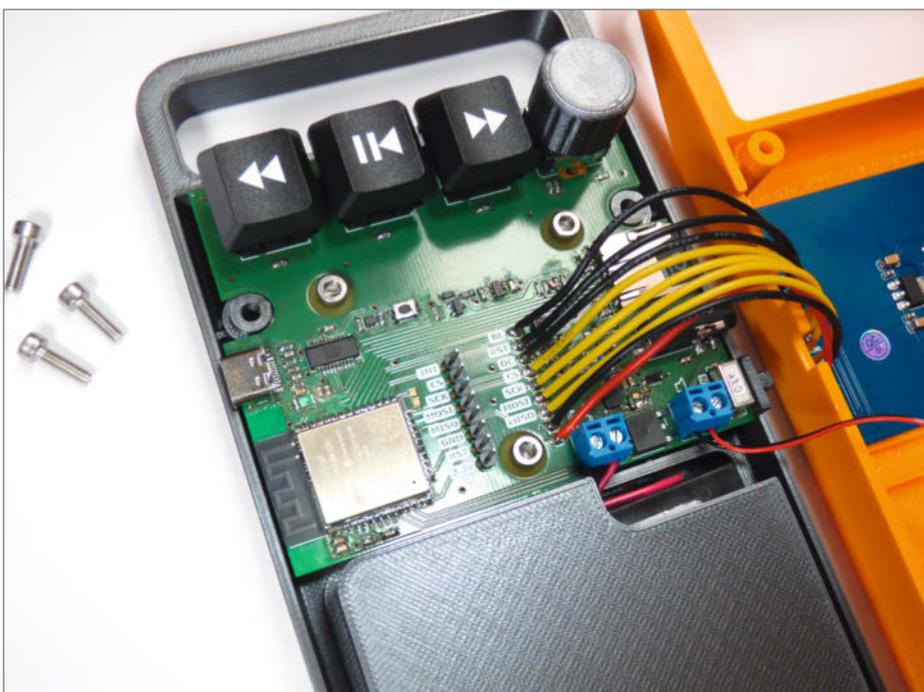


6 Das Display zeigt Titel, Interpret und ein animiertes Magnetband.

null anfangen. Mein Projekt habe ich auf der Library *ESP32-audioI2S* von GitHub-Nutzer *schreibfaul1* aufgebaut. Die Bibliothek kann bereits MP3-, M4A-, und WAV-Formate dekodieren und unterstützt noch zahlreiche nützliche Zusatzfunktionen. Unter anderem können die in den Dateien eingebetteten ID3-Informationen ausgelesen werden, wodurch sich Titel und Interpret ermitteln lassen. Außerdem kann durch längere Audiodateien

gesprungen werden, was gerade für Hörbücher ideal ist. Selbst die Wiedergabe von Netzwerk-Streams wird unterstützt.

Als zweiter Funktionsblock muss das Display angesteuert werden. Die Bibliothek *TFT\_eSPI* bietet alle nötigen Funktionen, um jede Art von SPI-Display zu betreiben <sup>6</sup>. Und das umfasst nicht nur das Ändern einzelner Pixel. Auch einfache Grafikobjekte, Bilder und Text brauchen nur noch abgerufen zu werden.



7 Das geöffnete Gehäuse mit den montierten Einzelteilen

Damit der ESP32 den Audioverstärker und das Display gleichzeitig sauber ansteuert, wird die Instanz der I2S-Bibliothek als eigener Task ausgeführt. Dies ist durch das Echtzeitbetriebssystem *FreeRTOS* möglich, das vom ESP32 unterstützt wird. Das Dekodieren der Audiodatei läuft somit als Hintergrundprozess mit hoher Priorität ab, während das Hauptprogramm mit niedrigerer Priorität bearbeitet wird. Dauert nun das Beschreiben des Displays im Hauptprogramm zu lange, läuft die Wiedergabe dennoch ohne Aussetzer weiter.

Als optischer Hingucker zeigt das Display ein sich bewegendes, angedeutetes Magnetband. Auch wenn die realistische Darstellung einer Kassette mit dem ESP32 möglich gewesen wäre, entschied ich mich bewusst dagegen. Stattdessen sollte ein stilisiertes Band erscheinen, ganz in Weiß auf schwarzem Hintergrund. Grob angelehnt ist diese Darstellung an das Interface der Synthesizer von *Teenage Engineering*. Es war nicht schwer, das Magnetband aus den Linien und Kreisen zusammenzubauen, die in der Displaybibliothek hinterlegt sind. Mithilfe von Antialiasing (Kantenglättung) lässt sich kaschieren, dass das LCD keine allzu hohe Auflösung hat. Ein bisschen schwieriger war es, die Spulen zu animieren. Um zu bestimmen, wo die Spulen zu jeder Zeit stehen, holte ich die alten Geometrie-Formeln wieder hervor.

Mein erster Test der Animation war ernüchternd: Die mühsam erstellte Spule flackerte beim Drehen. Die meisten SPI-Displaytreiber funktionieren so, dass Informationen, etwa eine Linie aus Pixeln, direkt an den Bildschirm gesendet werden. Das spart einen Puffer im Mikrocontroller, der bei 16-Bit Farbtiefe und 320x240 Pixeln schnell den Arbeitsspeicher sprengt. Als Nebeneffekt entsteht jedoch ein Flackern, da alle einzelnen Elemente nacheinander angezeigt werden, auch wenn sie direkt übereinander liegen. Eine Lösung für dieses Problem sind Sprites, also kleinere Bilder, die einfacher zwischengespeichert werden können. Der ESP32 zeichnet also erst alle Grafikelemente der Kassettenspule in das Sprite und setzt es zum Schluss in den Bildschirm ein. Dreht sich die Spule weiter, wird das alte Sprite mit einem neu berechneten überschrieben, sodass kein sichtbares Flackern mehr erscheint. Als Nebeneffekt kann Rechenzeit eingespart werden, indem dasselbe Sprite auf beide Spulen kopiert wird.

Neben der animierten Kassette braucht es nur noch wenige weitere Elemente für ein vollständiges Interface. Titel und Interpret werden als einfache Textblöcke angezeigt. Dazu kommen zwei Slider, jeweils für die Lautstärke und als Fortschrittsbalken für die aktuelle Audiodatei. Erkennt der ESP32 eine Batteriespannung von unter 4V am ADC-Eingang, erscheint zusätzlich ein kleines Batterie-Icon zur Warnung. Bis auf den Text sind alle Elemente aus einfachen

geometrischen Formen zusammengesetzt. Das Batterie-Icon besteht beispielsweise aus drei sich überlappenden Rechtecken. Werden diese Elemente kontinuierlich auf das Display geschrieben, kommt es auch hier zum Flackern. Um das zu optimieren, genügt es, die Elemente nur dann neu darzustellen, wenn sie sich geändert haben. Es wird also beispielsweise erst aktualisiert, sobald eine neue Audiodatei geladen oder die Lautstärke angepasst wurde.

### Automatisches Speichern

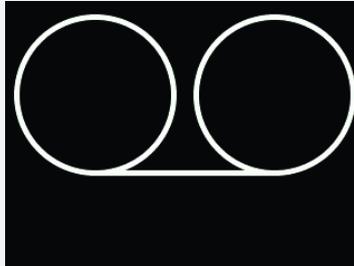
Damit das Gerät auch für Hörbücher geeignet ist, fehlt jetzt noch eine Funktion: das Speichern der Wiedergabeposition. Ohne sie fängt das Hörbuch sonst bei jedem Einschalten wieder von vorne an. Die Position muss also in einem nichtflüchtigen Speicher gesichert werden. Kontinuierlich auf die SD-Karte zu schreiben ist allerdings gefährlich. Die meisten Karten erlauben nur etwa 100.000 Schreibzyklen und verfügen nicht über *Wear-Leveling*, um den Flash-Speicher zu schonen. Schreibt man die Wiedergabeposition also sekundlich auf die gleiche Stelle, sind die Schreibzyklen schon nach einem Tag erreicht. Damit die SD-Karte länger durchhält, speichere ich die aktuelle Position nur einmal pro Minute. Die einzelnen Zeitstempel werden aneinandergehängt in einer Textdatei abgelegt. Pro Hörbuch entsteht so eine Datei, die während der Wiedergabe anwächst. Um die Position beim Start zu laden, muss der ESP32 nur die Größe der Datei ermitteln und zum letzten Zeitstempel springen. Die Lautstärke wird in einer zweiten Datei auf ähnliche Weise gespeichert. Auch hier kann die Karte geschont werden, indem erst gespeichert wird, nachdem der Lautstärkeregler gedreht wurde.

### Hinweise zum Nachbau

Ich versuche meine Projekte so zu gestalten, dass sie von anderen nachgebaut werden können. Leiterplatte, 3D-Modelle und Software stehen deshalb auf GitHub und Thingiverse (siehe Links in der Kurzinfor) zur Verfügung. Daneben braucht es nur noch den Zugang zu einem ordentlichen 3D-Drucker und einem feinen Lötkolben, um sich einen Retro-Audio-Player zusammenzubasteln.

Die Leiterplatte mit SMD-Bauteilen mag erstmal abschreckend wirken. In den letzten Jahren ist es für Privatleute jedoch immer einfacher und günstiger geworden, an Printed Circuit Boards (PCBs) und Bauteile zu kommen. Meine Leiterplatte habe ich beim deutschen Hersteller *Aisler* bezogen. Man muss beim Bestellvorgang lediglich die Daten aus dem *EAGLE*-Layoutprogramm hochladen. Ähnlich funktionieren auch internationale Anbieter wie *OSH-Park* und *JLPCB*. Einzelne Bauteile lassen sich zum Beispiel bei *Digikey*, *Mouser*

## Wie das User-Interface entsteht



### 1. Hintergrund

Der unbewegliche Teil der Spule muss nur einmal zu Beginn auf das Display geschrieben werden.

### 2. Dynamische Elemente

Titel und Interpret werden als Textblöcke dargestellt. Fortschritt und Lautstärke sind Slider aus einfachen Rechtecken. Diese Elemente werden nur bei Bedarf aktualisiert.

### 3. Animierte Spulen

Die rotierenden Spulen werden zuerst als Sprite erstellt und danach an beide Positionen im Display kopiert.

oder *LCSC* bestellen. Das Bestücken der Leiterplatte ist in diesem Fall mit einem guten Lötkolben möglich. Reflow-Löten mit Lötpaste und einem Ofen ist etwas komfortabler, allerdings habe auch ich diese Werkzeuge nicht zur Verfügung. Wer sich nicht an das SMD-Löten herantraut, kann die Leiterplatte z.B. bei *Aisler* auch bestückt bestellen oder versuchen, die Schaltung aus einzelnen Modulen zusammenzusetzen und zu verdrahten. Hierfür muss jedoch das Gehäuse angepasst werden.

Als Material für den 3D-Druck empfiehlt sich PETG-Filament, da es formstabil und temperaturfest ist. Wichtig ist ein Slicer, der gute Stützstrukturen erstellt, denn die Halbschalen werden liegend gedruckt. Ich habe den *PrusaSlicer* verwendet, auch weil ich dessen Bügel-Modus testen wollte. Dabei fährt die heiße Düse nach dem Druck erneut über die letzten Layer, um diese zu glätten.

Sind Leiterplatte und Gehäuse erstellt, kann das Gerät montiert werden **7**. Hierzu wird das Display in der oberen Gehäusehälfte verschraubt und über Kabel mit den Kontakten auf der Leiterplatte verlötet. Die Leiterplatte wird wiederum in der unteren Hälfte

verschraubt und die Kappen für Tasten, Dreh-Encoder und Einschalter aufgesteckt. Danach können Lautsprecher und Batteriehalter im Gehäuse befestigt und deren Kabel in die Schraubklemmen der Leiterplatte gesteckt werden. Schließlich werden die Gehäusehälften mit M3-Schrauben verbunden. Nachdem der Arduino-Sketch via USB aufgespielt ist, fehlen nur noch Batterien und eine SD-Karte mit Musik darauf – schon kann der Retro-Audio-Player in Betrieb genommen werden.

### Ausblick

Am Ende eines Projekts bleiben immer Ideen zurück, die man gerne noch realisieren würde. So bietet sich zum Beispiel die ungenutzte WLAN-Schnittstelle für ein Webradio an. Das animierte Magnetband könnte dabei durch eine Radioskala ersetzt werden, um beim Retro-Thema zu bleiben. Ebenso sind der Stromverbrauch und die Animationen optimierbar. Der aktuelle Funktionsumfang reicht allerdings schon völlig aus, um Hörbücher und Musik im Retro-Stil zu genießen. —*akf*

# Schallplatten per Sonos hören

Plattenspieler neuerer Generation haben oft einen USB-Anschluss, über den man seine Musiksammlung digitalisieren kann. Mit einem Raspberry Pi kann man damit leicht bestehende Multiroom-Speaker-Systeme wie Sonos füttern und spart sich so einen Hifi-Verstärker.

von Daniel Bachfeld

**F**rüher™ stand in jedem (Wohn)-Zimmer eine HiFi-Anlage mit Plattenspieler oben drauf. Heute reduziert sich die Musikanlage oft auf Lautsprecher mit WLAN-Streaming-Funktion, etwa die von *Sonos* oder von *Ikea* (steckt auch *Sonos* drin), um *Spotify*, *Tidal* und Konsorten darüber zu streamen. Aus Gründen (siehe Editorial Make 5/21) schwappt seit einiger Zeit die Retro-Welle durch meinen Haushalt und ich habe jetzt einen Plattenspieler von *Dual* (DT210) mit USB- und Audio-Ausgängen im Arbeitszimmer.

Dank des integrierten RIAA-Entzerrervorverstärkers kann man den Plattenspieler direkt an aktive Lautsprecher oder eine Soundbar anschließen. Allerdings habe ich nur ein Paar sonoskompatible Ikea-WLAN-Boxen im Zimmer, die sich leider nicht zum Abspielen eignen, da sie keinen Audio-Eingang haben. Mit einem Raspberry Pi lässt sich der Plattenspieler aber leicht Sonos-tauglich machen, um sich trotzdem Schallplatten über die Boxen anzuhören. Da ich alles parat hatte, hat mich das Projekt genau 0 Euro gekostet, wer alle Teile und Plattenspieler erst anschaffen muss, kommt auf rund 400 Euro.

Neben dem Pi brauchen wir noch zwei weitere (kostenlose) Zutaten, die eigentlich zum Betrieb von Internetradiostationen eingesetzt werden: Einen Streaming-Server, von dem aus Clients (Audioplayer) die Audiodaten im MP3-Format (oder sogar Ogg Vorbis) über das Übertragungsprotokoll *http* abholen und abspielen können. Dazu bietet sich der freie Server *Icecast* (in Version 2) an.

*Icecast* selbst kann sich die Audiodaten jedoch nicht über die Audioschnittstellen der Systemhardware holen, sondern erwartet fertige Audio-Streams von Source-Clients. Hier kommt *Darkice* ins Spiel: Diese Client-Software holt die Audiodaten von einer Soundkarte, wandelt sie ins MP3-Format und streamt sie zum *Icecast*-Server. Das kann lokal auf einem System geschehen, wie in diesem Fall, oder übers (W)LAN.

Als Soundkarte fungiert der Plattenspieler mit seinem integrierten Analog-Digital-Wandler, der mit vorgeschaltetem RIAA-Entzerrervorverstärker die analogen Signale des Tonabnehmers in digitale umwandelt und sie per USB rauschickt. Der Wandler agiert quasi

## Kurzinfo

- » Audio-Streaming-Systeme verstehen
- » Darkice und Icecast2 auf dem Pi installieren
- » Streamingquellen für Sonos einrichten

## Checkliste



**Zeitaufwand:**  
1 Stunde



**Kosten:**  
0-400 Euro

## Mehr zum Thema

- » Detlef Grell, Kurven kriegen (RIAA-Entzerrer), Make 3/17, S. 118
- » Jens Nohl, Nadel ab und Riemen schlapp, Make 3/17, S. 104

## Material

- » Raspberry Pi 3 oder 4
- » Plattenspieler Dual DT210 oder ähnliche mit USB
- » alternativ Behringer UFO 202 U-Phone für herkömmliche Plattenspieler ohne USB-Anschluss
- » WLAN-Boxen Sonos One oder IKEA Symfonisk
- » übliches Zubehör wie Netzteil, SD-Karte passend zum Raspberry

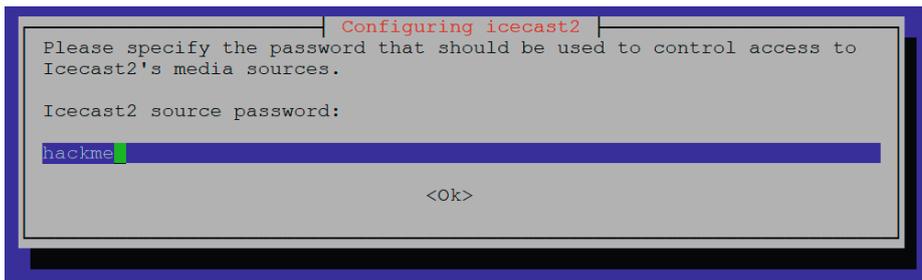
Alles zum Artikel im Web unter [make-magazin.de/x5jb](https://make-magazin.de/x5jb)

wie eine externe Soundkarte, mit der man aus Sicht des Pi aber nur aufnehmen kann. Unter Linux dockt die Hardware nahtlos an die *Advanced Linux Sound Architecture* (ALSA) an, die auf dem Raspberry Pi das Standard-Soundsystem ist.

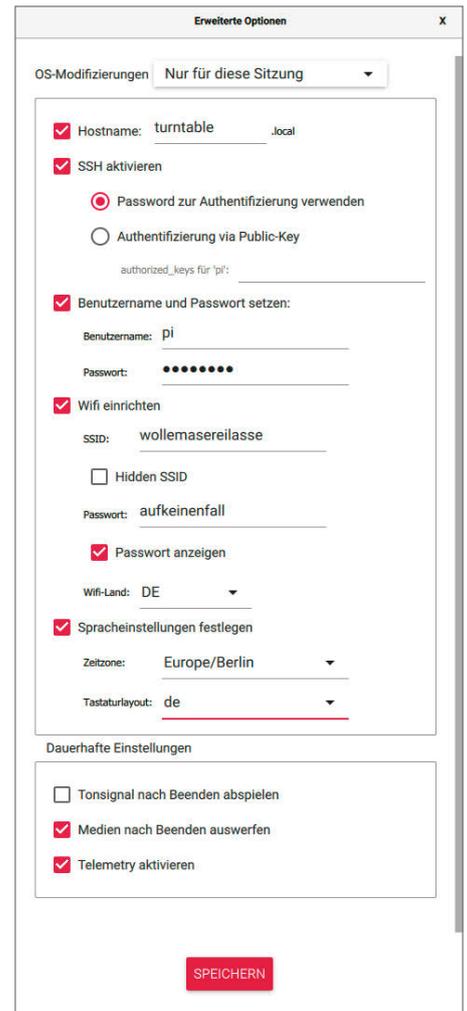
Um schließlich die Musik über die Boxen anzuhören, muss man den *Icecast*-Server als Radiostation in seiner *Sonos*-App anlegen und den Stream auf die Boxen lenken. Prinzipiell funktioniert *Icecast* nicht nur im heimischen (W)LAN, sondern man könnte auch ins Internet streamen. Man kann weitere Streaming-Clients mit *Icecast* verbinden und parallel verschiedene Streams veröffentlichen.

## Installation

Wir haben das Projekt mit einem Raspberry Pi 3B und 4 mit Pi OS 32 Bit sowie Pi OS 64 Bit Lite ohne Desktop getestet. Prinzipiell müsste es auch mit einem Pi Zero 2W funktionieren, dort am besten mit dem Pi OS Lite ohne grafische Oberfläche. Man kann ein bestehendes System leicht modifizieren und *Darkice* und *Icecast*



2 Die Installation von *Icecast* startet einen Konfigurationsdialog zur Änderung der Passwörter.



1 Mit dem Imager kann man die Konfiguration des Pi bereits vor dem ersten Booten festlegen.

3 Die Web-Oberfläche des Icecast-Servers liefert Infos.

4 Wenn Darkice einen Stream an Icecast liefert, zeigt der Server zusätzliche Informationen an.

nachträglich hinzufügen. Beide verbrauchen relativ wenig Ressourcen, sodass ein paralleler Betrieb etwa mit Pi Hole und anderen Diensten möglich ist.

Wir zeigen hier die Installation von Anfang an und zwar mit dem Raspberry Pi Imager, weil man dort Host-Namen, SSH-Zugriffe, WLAN-Einstellungen und Passwörter vorkonfigurieren kann. Den Imager müssen Sie von den Webseiten der Raspberry Pi Foundation herunterladen (siehe Link). Nach dem Start wählt man das OS und die richtige SD-Karte aus. Bevor man auf *Schreiben* klickt, klickt man auf das Zahnrad-Icon und füllt die Punkte in 1 analog zu den eigenen Gegebenheiten aus. Am besten benennt man den Pi nach seiner Funktion, also *turntable* oder *vinyl*.

Nach dem Flashen der SD-Karte steckt man diese in den Pi und schließt ein Netzteil an. Tastatur, Maus und Monitor benötigen wir nicht. Mit einem SSH-Client (*Putty* unter Windows oder dem Kommando-Zeilen-Befehl *ssh* unter Linux) baut man die Verbindung zum Pi auf. Als Adresse gibt man nur den Pi-Namen OHNE *.local* an, also *turntable* oder was auch immer Sie gewählt haben. Loggen Sie sich mit den vorkonfigurierten Daten ein, unter Linux etwa mit `ssh pi@vinyl1`.

Stecken Sie jetzt den USB-Stecker des Plattenspielers in eine der USB-Ports des Pi und rufen im SSH-Client den Befehl `dmesg` auf. Gibt der Pi etwas in der Art

```
input: BurrBrown from Texas Instruments
USB AUDIO CODEC as /devices/platform/
soc/3f980000.usb/usb1/1-1/1-1.3/1-
1.3:1.3/0003:08BB:29C0.0001/input/
input4
hid-generic 0003:08BB:29C0.0001:
input,hidraw0: USB HID v1.00 Device
[BurrBrown from Texas Instruments USB
AUDIO CODEC] on usb-3f980000.usb-1.3/
input3
usbcore: registered new interface
driver snd-usb-audio
```

aus: Glückwunsch, das Sound-Interface wurde automatisch erkannt. Mit dem Befehl `arecord -l` zeigt uns der Pi an, als welches Device er die Karte registriert hat, das müssen wir gleich bei der Configuration von Darkice angeben:

```
**** List of CAPTURE Hardware Devices
****
card 1: CODEC [USB AUDIO CODEC],
device 0: USB Audio [USB Audio]
Subdevices: 1/1 Subdevice #0:
subdevice #0
```

Wir notieren uns `card 1` für die Configuration von Darkice. Manchmal erkennt der Pi den Plattenspieler auch als `card 2`, weil zuerst das `vc4-HDMI-Audio`interface erkannt wurde. Das ALSA-System spielt in seltenen Fällen ein wenig Lotto. In schwerwiegenden Fällen,

bei denen sich die Reihenfolge bei jedem Boot ändert, kann man über die Datei `/etc/modprobe.d/alsa-base.conf` die Reihenfolge statisch festlegen. Weitere Infos dazu finden Sie unter dem Link in der Kurzinfo.

Mit `sudo apt install darkice` installiert man den Darkice-Streamer aus den Pi-Repositories, in dessen Zuge noch ein paar Bibliotheken mitinstalliert werden. Leider bringt Darkice keine Standardkonfiguration mit, weshalb wir sie mit `sudo nano /etc/darkice.cfg` selbst anlegen und mit den im Listing `darkice.cfg` angegebenen Werten befüllen müssen. Die Datei steht auch zum Download in unserem Github-Repository zum Artikel bereit. Details zur Bedeutung der einzelnen Parameter finden Sie unter dem Link. Speichern Sie die Datei in `nano` mit `STRG-O` und verlassen Sie den Editor mit `STRG-X`.

### Eisverkäufer

Nun installieren wir noch Icecast mit `sudo apt install icecast2`. Die Installationsprozedur fragt, ob wir den Server konfigurieren wollen. Wollen wir. Als ersten tragen wir wieder den anfangs gewählten Host-Namen ein. Anschließend fragt der Dialog nach drei Passwörtern: eines zur Authentifizierung von Darkice 2, eines für den Fall den Betriebs als Relais-Station und eines für den Webzugriff auf die Web-Oberfläche von Icecast. Sie können das vorgeschlagene `hackme` in allen drei Fällen erstmal so lassen, sofern nicht mit bedrohlichen Angriffsszenarien in Ihrem eigenen WLAN zu rechnen ist. Wohnen Sie in einer WG und haben lustige Mitbewohner, sollten Sie die Passwörter ändern.

Icecast2 startet automatisch nach der Installation und trägt auch alles Notwendige in die Init-Skripte des Pi ein, um auch künftig beim Booten des Systems zu starten. Lassen Sie den Plattenspieler am Pi angeschlossen, booten Sie mit `sudo reboot` und verbinden sich per SSH erneut mit dem Pi.

## Icecast2 Admin

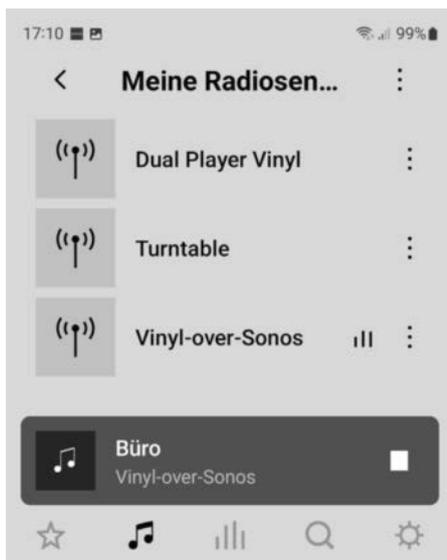
5 Sobald sich Clients wie die Sonos-Boxen verbinden, tauchen sie in der Client-Liste auf. Interessanterweise gaukeln sie im User-String vor, dass sie ein Winamp-Client seien.

### darkice.cfg

```
[general]
duration                = 0
bufferSecs              = 1
reconnect               = yes
realtime                = yes
rtprio                  = 3

[input]
device                  = hw:1,0 #card 1
sampleRate              = 48000
bitsPerSample           = 16
channel                 = 2

[icecast2-0]
bitrateMode             = cbr
format                  = mp3
bitrate                 = 320
server                  = local
hostport                = 8000
password                = hackme # pw für icecast2
mountPoint              = records
name                    = Turntable
description              = DarkIce on Raspberry Pi
url                     = http://vinyl
genre                   = vinyl
public                  = no
localDumpFile           = recording.m4a
```



**6** So erstellen Sie eine neue Radiostation in der Sonos-App und starten sie.

Der Webserver von Icecast horcht standardmäßig auf Port 8000. Mit `http://vinyl:8000` rufen wir in einem beliebigen Browser die Startseite auf. Klicken wir auf *Admin Home*, fragt der Server nach den Zugangsdaten, die wir eingeben: *admin* (klein) und *hackme* (sofern Sie das vorgeschlagene übernommen haben). Soweit sollten das Bild **3** dem Ihrer Installation ähneln. Geben Sie nun im SSH-Client *darkice* ein, um den Source-Client zu starten. Meckert *Darkice* mit der Meldung

```
DarkIce: DarkIce.cpp:1273: can't open connector [0]
```

ist die Soundkarte falsch eingetragen. Passen Sie in *darkice.cfg* die Zeile

```
device = hw:1,0
```

entsprechend der Ausgabe von `arecord -l` an. Ist der Eintrag richtig, startet *Darkice* und verbindet sich mit Icecast. Rufen Sie den Webserver erneut auf, sehen Sie zusätzliche Informationen wie in **4** und beim Klick auf die *Mount Points* Angaben wie in **5**.

### Sonographie

Für einen ersten Test müssen wir nun noch in der Sonos-App den Icecast-Server als Radiostation einrichten, mit der sich die Boxen verbinden sollen. Dazu geht man in der App unter *Tuneln/Meine Radiosender* und fügt unter den drei Punkten rechts oben eine Station hinzu. Die Stream-URL

lautet `http://vinyl:8000/records`, als Sendernamen geben Sie etwas aussagekräftiges ein, ähnlich wie in **6**. Speichern Sie die Konfiguration und wählen Sie nun den eben angelegten Sender aus. Sofern alles geklappt hat, sollte die App den Sender auswählen und sofort starten. Sie zeigt dann statt des Play-Symbols beim Sender das Stop-Symbol an.

Spielen Sie nun eine Platte ab. Der Stream hat mitunter eine Verzögerung von einigen Sekunden. Bei mir sind es teilweise 4 Sekunden. Also Geduld. Als dann sollte irgendwas aus den Boxen zu hören sein. Wenn nicht: Hat *Darkice* auf der Konsole einen Fehler wegen einer falschen Konfiguration gemeldet? Dann beseitigen Sie die Fehler.

Sollten im späteren Betrieb die Boxen keinen Ton von sich geben, hat eventuell ein vorheriger Spotify-Stream (oder was sonst noch so über die Boxen läuft) die Streaming-Quelle umgebogen. Sie müssen dann in der Sonos-App erneut den Vinyl-over-Sonos-Sender auswählen und starten.

Im Prinzip läuft nun alles. Leider konfiguriert der *Darkice*-Client seine Init-Skripte

nicht automatisch. Aber selbst die manuelle Einbindung in *systemd* funktioniert nicht nachhaltig. *Darkice* startet aufgrund von Fehlern beim Booten nicht. In Streaming-Foren berichten andere User ebenfalls von diesem Fehlverhalten. Wir vermuten, dass *Darkice* sich zu früh mit den Audioschnittstellen verbindet und bei nicht vollständig initialisiertem System dann scheitert.

Wir umgehen das Problem, in dem wir *Darkice* einfach über die Reboot-Option von *crontab* mit einer Verzögerung starten. Dazu gibt man im SSH-Client `sudo crontab -e` ein, wählt als Editor *nano* aus und fügt der Datei unten die Zeile

```
@reboot sleep 20 && /usr/bin/darkice -c /etc/darkice.cfg
```

hinzu. Speichern und schließen Sie die Datei (STRG+O, STRG+X). Nach einem Reboot wird *Darkice* 20 Sekunden später gestartet. In unseren Tests hat das zuverlässig funktioniert. Wie gehabt startet man die Radiostation und kann seine Plattensammlung genießen. Alternativ zu den Sonos-Boxen kann man den Stream auch über einen Player auf dem PC oder das Smartphone anhören. Im Prinzip kann jeder moderne Browser durch Aufruf der Streaming-URL `http://vinyl:8000/records` den Stream abspielen.

### Ohne USB

Um ein altes Plattenspieler-Schätzchen ohne integrierte USB-Schnittstelle Sonos-fähig zu machen, greift man auf den Konverter *Behringer UFO 202 U-Phone* zurück **7**. Er enthält einen RIAA-Entzerrervorverstärker, der die Signale über zwei Plattenspielerkompatible Cinch-Buchsen aufnimmt. Ein A/D-Wandler gibt die Daten per USB aus. Für knapp 25 Euro bei 48kHz Samplerate und 16 Bit Auflösung ist das eine günstige Aufrüstoption.

Mit der vorgestellten Streaming Lösung kann man nun seine Plattensammlung ohne Hifi-Verstärker genießen. Während dieser Artikel entstand, lief im übrigen viel Kraftwerk ... —*dab*

**7** Der Konverter Behringer UFO 202 macht alte Plattenspieler ohne USB-Port streamingfähig.



# Experimentieren für Maker

**ELV** Kompetent in Elektronik

## ELV Experimentier-/Steckboard EXSB1

Das Experimentierboard lässt mit seiner großen Steckbrett-Fläche und der Peripherie fast keine Wünsche beim Experimentieren offen.

- Steckboard-Feld mit 2x 830 Kontakten
- 3 Spannungseingänge
- Rechteckgenerator mit ICM7555 (0,01 Hz - 100 kHz)
- Umfangreiche Peripherie mit LEDs, Potentiometern, Messpunkten, Umschaltern und diversen Buchsen
- Anschlussbelegung auf der Platine aufgedruckt

Abm. (B x H x T): 310 x 110 x 173 mm, Gewicht: 780 g



BAUSATZ

**99,95 €**

Lieferung ohne Verbindungskabel

Artikel-Nr. 153753

### Exklusives Spar-Bundle

## ELV Mini-Voltmeter/Strommessadapter

Das Duo aus Mini-Voltmeter und Strommessadapter ist ein unverzichtbares Werkzeug für Experimentierschaltungen.

- Messbereiche: bis max. 3 A und 24 V, Auflösung: 1 mA und 1,25 mV
- Breadboard-kompatibel
- Über die USB-A-Buchse können auch USB-Verbraucher analysiert werden – die USB-Typ-C-Buchse dient als Spannungsquelle
- Anzeige der Messwerte digital, analog, Bargraph und Plotter



Mini-Voltmeter

BAUSATZ

Angebot bis 31.10.2022

**39,90 €**

statt 44,90 €

Artikel-Nr. 253003

## ELV Mini-DDS-Signalgenerator MDS1

Der Mini-DDS-Signalgenerator MDS1 kann Sinus- und Rechtecksignale mit einer hohen Genauigkeit von 0,1 Hz bis 1 MHz ausgeben.

- Arbeitet auf Basis des AD9837
- Ausgabe von Sinus- und Rechtecksignalen
- Signalfrequenz: 0,1 Hz - 1 MHz, Auflösung: 0,1 Hz
- Hohe Genauigkeit: 15 ppm, kalibrierbar
- Sonderfunktionen: Frequenz-Kalibrierung, Werksreset, Bootloader für Firmware-Update
- Spannungsversorgung über USB-C-Buchse oder Stiftheisten

Abm. (B x H x T): 69 x 29 x 50 mm, Gewicht: 47 g



BAUSATZ

**39,95 €**

Lieferung ohne Breadboard

Artikel-Nr. 157710

ELV bietet eine reichhaltige Auswahl an Bausätzen zum Experimentieren an, mit denen man Maker-Projekte schnell und einfach verwirklichen kann.

Mehr zu den Produkten finden Sie unter:

**de.elv.com/maker**



# ESP32-CAM zeigt Filmnegative

Ist der Berghang auf dem Foto tief verschneit oder grob überbelichtet? Wer steht da neben der Tante, lacht der freundlich oder schaut er grimmig? Auf den Kleinbildnegativen vergangener Zeiten ist das nur schwer zu erkennen. Doch verbannen Sie dieses Problem dahin, wohin es gehört – ins vorige Jahrhundert. Denn heute zeigt nach kurzer Bastelarbeit unser ESP32-Gucki die Schnappschüsse von damals mit korrekten statt umgekehrten Tonwerten und Farben an – und so groß, wie es die Displays hergeben.

von Peter König



Hamburg in den frühen 90ern, live aufs Tablet gebeamt vom ESP32-Gucki.

Wer bereits in der vor-digitalen Ära einigermaßen ambitioniert fotografierte, hat ganz sicher noch ein sorgfältig gelagertes Archiv voller Negative im Haus – schließlich sind das bei analogen Fotos die eigentlichen Originale und die positiven Abzüge, die man ins Album klebt, bloß Kopien davon. Dummerweise kann man auf den Negativen meist nur schemenhaft was erkennen – die wenigsten Menschen sind in der Lage, im Kopf hell in dunkel zu verwandeln und umgekehrt. Die Zuordnung zwischen einem vorhandenen Abzug und seinem Negativ bekommt man zwar in der Regel noch hin, aber wenn nur das Negativ zur Hand ist, ist es schwierig, Personen darauf zu identifizieren oder gar Gesichtsausdrücke zu deuten – oder hätten Sie das bei Bild 1 nur aufgrund des Negativs hinbekommen?

In analogen Zeiten gab es das Problem natürlich auch schon. Die notdürftige Lösung waren Kontaktabzüge, die die Fotos zwar positiv zeigten, aber bei Kleinbild gerade mal auf einem Format von 36mm × 24mm. Da war die Lupe gefordert.

Wie gut, dass es mittlerweile die *ESP32-CAM* gibt: Dieses Mikrocontroller-Board samt Kamera bekommt man vielfach für deutlich unter 10 Euro zu kaufen. Und es bringt schon alles mit, um mit wenig zusätzlichem Material in einer Stunde einen einfachen Negativbetrachter zu bauen. Der zeigt die Bilder nicht nur gleich positiv, sondern auch vergrößert auf dem Display jedes WLAN-fähigen Geräts an, sei es Tablet, Smartphone oder Rechner. Unser Projekt *ESP32-Gucki* (zur Namensgebung siehe Kasten) eignet sich damit auch für den Ad-hoc-Aufbau vor dem Besuch bei der betagten Verwandtschaft, bei der noch die Fotoarchive aus dem letzten Jahrhundert schlummern, die mal gesichtet und von den Zeitzeugen erklärt werden wollen, ehe Filme und Erinnerungen verblassen. Braucht man hinterher seinen Gucki nicht mehr, lässt er sich weitgehend rückbauen und die Teile anderweitig verwenden. Denn neben der vielseitig nutzbaren *ESP32-CAM* besteht unser Projekt nur aus einem Breadboard, ein paar Kabelbrücken, einem Smartphone-Netzteil samt Adapterplatine und einer Bühne aus Pappe für die Negative.

## Das Board

Die *ESP32-CAM* haben wir in der *Make* schon in diversen Projekten eingesetzt (siehe *Mehr zum Thema* in der Kurzinformatio). Im Artikel *Intelligente Webcam für 5 Euro* (*Make* 1/20, S.28) haben wir gezeigt, wie man seinen Code mit einem universellen externen USB-zu-seriell-Konverter draufbekommt. Diese Methode funktioniert natürlich nach wie vor und kann auch für den Gucki zum Einsatz kommen, falls Sie schon eine *ESP32-CAM* und einen solchen

## Kurzinformatio

- » Film-Negative positiv anzeigen lassen
- » Bilder auf dem Smartphone, Tablet oder Notebook betrachten
- » Funktioniert dank ESP-Accesspoint ohne externes WLAN
- » ESP32-CAM auf kurze Entfernungen scharf stellen

---

### Checkliste

-  **Zeitaufwand:**  
zwei Stunden
-  **Kosten:**  
maximal 25 Euro – Stand September 2022, ohne Smartphone-Netzteil und Hintergrundbeleuchtung
-  **Programmieren:**  
Code aufs Board laden mit der Arduino IDE
-  **Basteln:**  
einfaches Basteln mit Pappe
-  **Löten:**  
eventuell Pins an die Adapterplatine fürs Smartphone-Netzteil löten

---

### Mehr zum Thema

- » Daniel Bachfeld, Intelligente Webcam für 5 Euro, *Make* 1/20, S. 28
- » Peter König, Alternative Optik für die *ESP32-CAM*, *Make* 2/20, S. 8
- » Peter König, Raspi-Kamera nachgeschärft, *Make* 3/16, S. 58
- » Peter König, Lieblingswerkzeug: Scheckkarte, *Make* 3/16, S. 10
- » Carsten Wartmann, Flexible Stative im Eigenbau, *Make* 1/22, S. 68

---

### Material

- » *ESP32-CAM* mit USB-zu-Seriell-Adapter für die Programmierung
- » **Micro-USB-Kabel** zur Datenübertragung zwischen Rechner und *ESP32*
- » **Breadboard** lang
- » **4 Drahtbrücken** oder Jumperkabel *male-male*
- » **Smartphone-Netzteil** mit Micro-USB-Stecker
- » **Adapterplatine** mit Micro-USB-Buchse fürs Breadboard
- » **Pappe 3mm**, 27cm × 13cm reicht
- » **Pappe 1mm**, 12cm × 5cm reicht
- » **Alleskleber** ggf. noch Klebepads (*Patafix* o.ä.)
- » **Streifen Transparentpapier** (technisches Zeichenpapier) oder Butterbrotpapier als Lichtdiffusor, optional
- » **Hintergrundbeleuchtung** etwa durch Tageslicht, Deckenfluter und Zimmerdecke, Schreibtischlampe oder Fotoleuchte mit Tischstativ

---

### Werkzeug

- » **Drucker** zum Drucken der Schneidevorlagen
- » **Cutter, Bastlerskalpell und Stahllineal** sowie Schneideunterlage, alternativ Lasercutter
- » **Zwei Spitzzangen** oder ähnliche Haltewerkzeuge fürs Fokussieren der Kamera
- » **3D-Drucker** optional für Fokuserwerkzeug
- » **LötKolben** falls an die Adapterplatine fürs Netzteil noch Pins gelötet werden müssen

---

Alles zum Artikel im Web unter [make-magazin.de/xj9h](https://make-magazin.de/xj9h)



Konverter haben. Wer sich hingegen eigens für dieses Projekt das Kameraboard kauft, kann zu einem Angebot greifen, bei dem für wenige Euro mehr ein maßgeschneiderter Konverter zum Drunterstecken gleich mitgeliefert wird 2 (Bezugsquellen siehe Link in der Kurzinformatio). Über diesen Adapter findet nicht nur der Code bequem seinen Weg von der Arduino IDE aufs Board, für die ersten Experimente lässt sich die *ESP32-CAM* darüber auch mit Spannung versorgen. Spätere *ESP32-CAMs* für weitere Projekte kann man dann ohne neuen Konverter kaufen.

Achtung: Für unseren Gucki sollte es auf jeden Fall eine *ESP32-CAM* mit dem Kameramodul *OV2640* mit knapp 2 Megapixeln (UXGA, 1600 × 1200) sein, nicht eines mit dem ebenfalls

unterstützte *OV7670* mit nur 0,3 Megapixeln. Außerdem gibt es gelegentlich irreführende Angebote bei *AliExpress & Co.*, bei denen etwa nur das *Kameramodul*, aber nicht das Board enthalten ist ...



Roman Samborskiy / Shutterstock.com

# Das Original: Der Gucki

Vor allem die Dia-Fans unter den Fotoamateuren vergangener Jahre werden sich noch an den originalen *Gucki* erinnern: Das war ein einfaches Plastikgehäuse mit einer gewölbten Mattscheibe am einen und einer Sammellinse als Okular am anderen Ende. Schob man ein Dia vor die Mattscheibe, richtete diese gegen das Licht und blickte durchs Okular, sah man das Dia vergrößert (wenn auch nicht wirklich in 3D, wie auf der Packung unseres Guckis aus altem Familienbesitz behauptet wird). Theoretisch konnte man auch Negativstreifen in den Gucki schieben, aber das Grundproblem blieb: Man sah seine Bilder mit umgekehrten Tonwerten, was wenig nützte.



## Die Software

Wir benutzen für den Gucki den universell einsetzbaren *esp32-cam-webserver* des Git-Hub-Users *easytarget*. Den Link zum Original finden Sie in der Kurz-URL in der Kurzinfor. Wenn Sie stattdessen den Server aus unserem GitHub-Repository herunterladen, ist gleich eine fertige Konfigurationsdatei *myconfig.h* enthalten, mit der Sie direkt loslegen können, ohne auch nur ein einziges Zeichen am Code zu ändern.

Wenn das Ihr erstes Projekt mit der Arduino IDE, mit einem ESP und auch mit GitHub ist, finden Sie online über die URL in der Kurzinfor eine ausführliche Schritt-für-Schritt-Anleitung, wie Sie den Code herunterladen, die IDE (Entwicklungsumgebung) und das Board miteinander bekannt machen und schließlich das Programm kompilieren und erfolgreich

hochladen. Erfahrenen Makern müssen wir das hier nicht mehr erklären, es gibt keine Überraschungen dabei. Als Board stellt man beim hier verwendeten und in den Bezugsquellen verlinkten Modul *AI Thinker ESP32-CAM* ein – das kennt die Arduino IDE, falls schon ESP32 bei den Boardverwaltern eingetragen ist. Falls nicht, bekommen Sie Hilfe über den Link in der Kurzinfor. Der verwendete *esp32-cam-webserver* kommt übrigens auch mit ähnlichen Kameramodulen klar, das erfordert allerdings dann in der *myconfig.h* ab Zeile 182 eine entsprechende Änderung.

Sobald Sie das Programm auf die ESP32-CAM überspielt haben, resettet sich das Board automatisch und startet dann neu. Der ESP ist nach unserer vorbereiteten Konfiguration standardmäßig im *Accesspoint*-Modus aktiv, startet ein WLAN mit der Kennung *GuckiWLAN*, mit dem man sich vom Smartphone, vom Tablet oder Rechner aus mit dem Passwort *MakeMake136472* verbinden kann. Dann öffnet man einen Browser, gibt in der Adresszeile <http://ESP-Gucki> oder <http://192.168.4.1> ein und sieht anschließend die (arg reduzierte) Startseite des Servers <sup>3</sup>. Kommt die Verbindung nicht zustande, kann es sein, dass sich Ihr Mobilgerät automatisch sofort wieder mit einem anderen erreichbaren WLAN verbunden hat, was einen Internetzugang verspricht (das halten etwa manche Android-Smartphones für existenziell). Man kann aber in der Regel bei den WLAN-Einstellungen zum *GuckiWLAN* festlegen, dass man mit diesem Netz verbunden bleiben will, auch wenn das keinen Internetzugang bietet.

Falls Sie den Gucki lieber im heimischen WLAN verwenden wollen, können Sie dessen SSID und Passwort in der Datei *myconfig.h* als zusätzlichen Eintrag im Array `stationList[]`

(Zeile 25) einfügen. In diesem Fall startet der Gucki nur dann sein eigenes WLAN mit den Zugangsdaten in der ersten Zeile, falls keines der weiteren WLANs im Array erreichbar ist. So ist der Negativbetrachter stets benutzbar, ob zu Hause oder unterwegs, ohne dass man im Code was ändern müsste. Sicherheitsfanatiker sollten natürlich auch die Kennung und das Passwort für das ESP-eigene WLAN ändern, man weiß ja nie ...

## Der Bildcheck

Klicken Sie jetzt im Browser auf den roten Button *Full Viewer*, dann öffnet sich das Menü mit allen Einstellmöglichkeiten. Für den ersten Test lassen Sie alles wie voreingestellt, klappen die *Settings* über Klick oder Tipp darauf zu <sup>4</sup> und drücken auf die Schaltfläche *Start Stream* <sup>5</sup>.

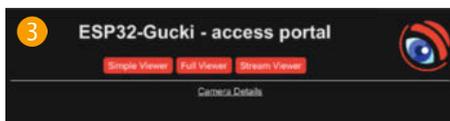
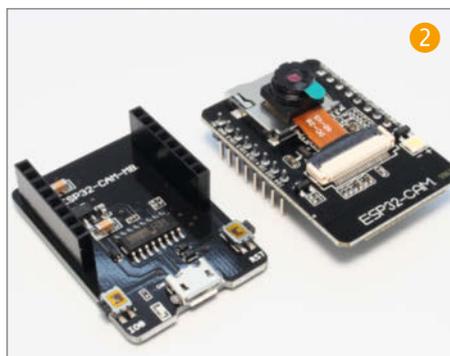
Jetzt sollte ein Live-Bild von der Kamera zu sehen sein, dass Sie für einen wichtigen Test brauchen: Schauen Sie so in die Kamera auf dem Board, dass Sie die Beschriftung *ESP32-CAM* auf der Platine lesen können. Sehen Sie dann im Browser sich selbst in einer Querformat-Aufnahme, mit dem Kopf nach oben? Dann brauchen Sie nachher die *Querformat*-Schneidevorlage. Denn wie sich bei der Arbeit für diesen Artikel herausstellte, gibt es auch (vermutliche ältere) ESP32-CAMs, bei denen der Sensor im Kameramodul um 90 Grad gedreht eingebaut ist. Ragt Ihr Konterfei im Browser also von der Schmalseite rechts oder links ins Bild hinein, ist die *Hochformat*-Schneidevorlage später richtig für Sie.

Die *Settings* können Sie jederzeit auch im laufenden Betrieb wieder verändern. Noch ist das Bild positiv, aber wenn Sie unter *Special Effect* <sup>6</sup> den Eintrag *Negative* wählen, ist der große Teil der Magie schon vollbracht. Unter *Resolution* <sup>7</sup> wählt man die gewünschte Auflösung (idealerweise eine im Seitenverhältnis 3:2 wie bei den Negativen). Zwischen Bildern im Hochformat und solchen im Querformat wechselt man später beim Sichten der Negative über das Menü *Rotate in Browser* <sup>8</sup> und seitenverkehrte oder auf dem Kopf stehende Bilder korrigiert man mit *H-Mirror Stream* und *V-Flip Stream* <sup>9</sup> – oder man fädelt ganz wie früher den Negativstreifen einfach seitenrichtig ein.

Achtung: Die manuellen Einstellungen für Kontrast und Helligkeit vertragen sich nicht mit dem Negativ-Filter, also lässt man es besser bei der Automatik.

## Das Breadboard

Ist die Software an Bord und funktioniert die Kamera, zieht die ESP32-CAM aufs Breadboard um. Lassen Sie sich nicht täuschen: Ein kurzes Exemplar mit 30 Reihen kann knapp zu kurz sein, je nach verwendeter Pappe und Breite der Adapterplatine für den USB-Anschluss.



Elektronisch passiert nicht viel – das Steckbrett sorgt lediglich für die mechanische Basis der Konstruktion und die Spannungsvorsorgung der ESP32-CAM **10**. Als Adapter für das Smartphone-Netzteil mit Micro-USB-Anschluss haben wir hier das PAD aus dem Elektronik-Experimentierset genommen (siehe auch Seite 98), alternative Breakout-Boards für diesen Anschluss gibt es von diversen Anbietern (Links in der Kurzinfor). Die Drahtbrücken führen einerseits +5 Volt und Masse von der Adapterplatine an die beiden Spannungsschienen des Breadboards und andererseits von dort die +5 Volt an den ersten Pin und Masse an den zweiten Pin der ESP32-CAM links oben, wenn man wieder in die Kamera schaut und die Beschriftung auf dem Board aufrecht lesen kann (zur Sicherheit sind die Pins auch winzig klein beschriftet). Die Anzahl der Buchsenreihen und damit der Abstand zwischen ESP32-CAM und Adapterplatine ist nicht entscheidend, er sollte nur so groß sein, dass später die Bühne ausreichend Platz hat – und die bauen wir jetzt.

## Die Bühne

Die Bühne für die Negative besteht zum einen aus einem Würfel, der über das Kameramodul gestülpt wird, das größte Streulicht aussperrt und für den nötigen Abstand zwischen Linse und Negativ sorgt. Zum anderen gibt es noch zwei Schienen, durch die der Negativstreifen geschoben wird und die so konstruiert sind, dass der Bildbereich der empfindlichen Filmstreifen an keiner Stelle über irgendeine Oberfläche oder Kante schleift, was böse Kratzer zur Folge haben könnte. Den Aufbau zeigt das Schema **11**.

Experimentell haben wir vorab ermittelt, dass das Negativ idealerweise etwa 50mm über der Oberfläche des Breadboards liegen sollte, damit die Kamera das Negativ möglichst formatfüllend aufnimmt, aber auf keinen Fall am Rand etwas abgeschnitten wird. Das ist aber nur eine Hintergrundinfo. Selber messen und konstruieren müssen Sie nichts, wir haben Schneidevorlagen für alle Pappteile in unsere Downloads gepackt. Sie können die Vorlage als PDF auf normales A4-Papier drucken, das auf die Pappe kleben und dann mit einem Cutter und Stahllineal sowie einem Bastelkalpell für saubere Ecken und Kerben ausschneiden. Hier ist etwas Geduld gefragt, der Zuschnitt der Pappteile von Hand ist der zeitaufwendigste Teil des Gucki-Baus. Falls Sie einen Lasercutter zur Verfügung haben, geht es damit natürlich bequemer und schneller; dann können Sie nach Belieben auch andere Materialien wie Sperrholz oder Kraftplex benutzen.

Achtung: Es gibt von den Vorlagen zum Download jeweils eine Version für *Hoch-*

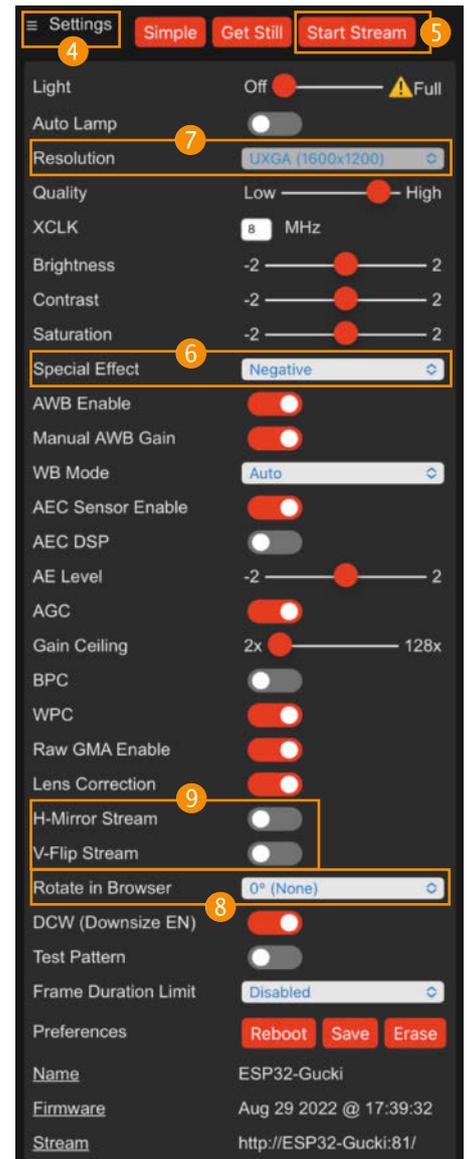
*format*-Keramodule (bei denen läuft später der Filmstreifen parallel zum Breadboard) und eine für *Querformat*-Keramodule (da läuft der Film quer zum Breadboard, diese Version ist in **11** zu sehen). Welche Variante Sie brauchen, haben Sie ja bereits ermittelt (und wissen das hoffentlich noch).

Wenn Sie für Ihren Gucki extra Pappe besorgen müssen, schauen Sie mal, ob Sie im Künstlerbedarf sogenannte *Finnpappe* mit 3mm Stärke bekommen. Die ist günstig, lässt sich trotz der Dicke gut schneiden und man bekommt durch die Stärke satt Klebefläche zwischen den Teilen. Der Nachteil: Sie neigt etwas zum Fusseln, ein Staubsauger in Reichweite ist beim Schneiden nützlich. Es geht aber im Prinzip jede Pappe, die mindestens 1,5mm dick ist – die minimalen nötigen Modifikationen an der Schneidevorlage durch dünneres Material beschreiben wir online. Wir haben speziell für noch dünnere Pappe auch eine alternative Schneidevorlage und Bauanleitung ins Web gestellt, solider wird es aber mit der dicken Pappe.

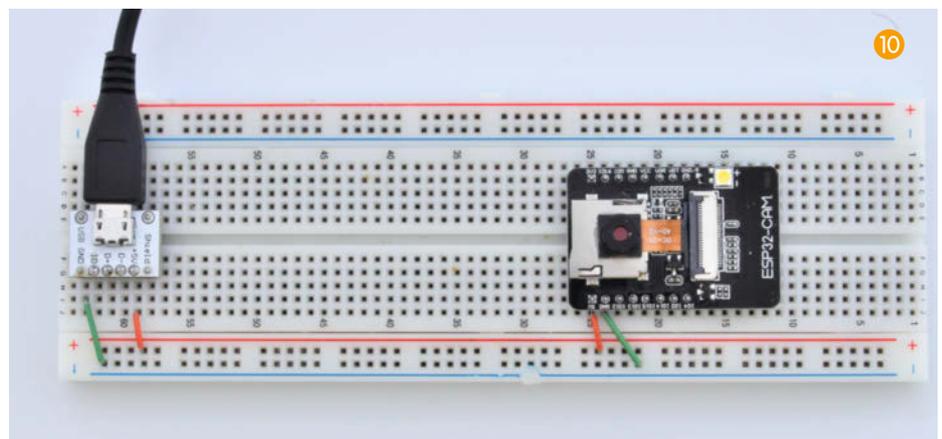
## Der Zusammenbau

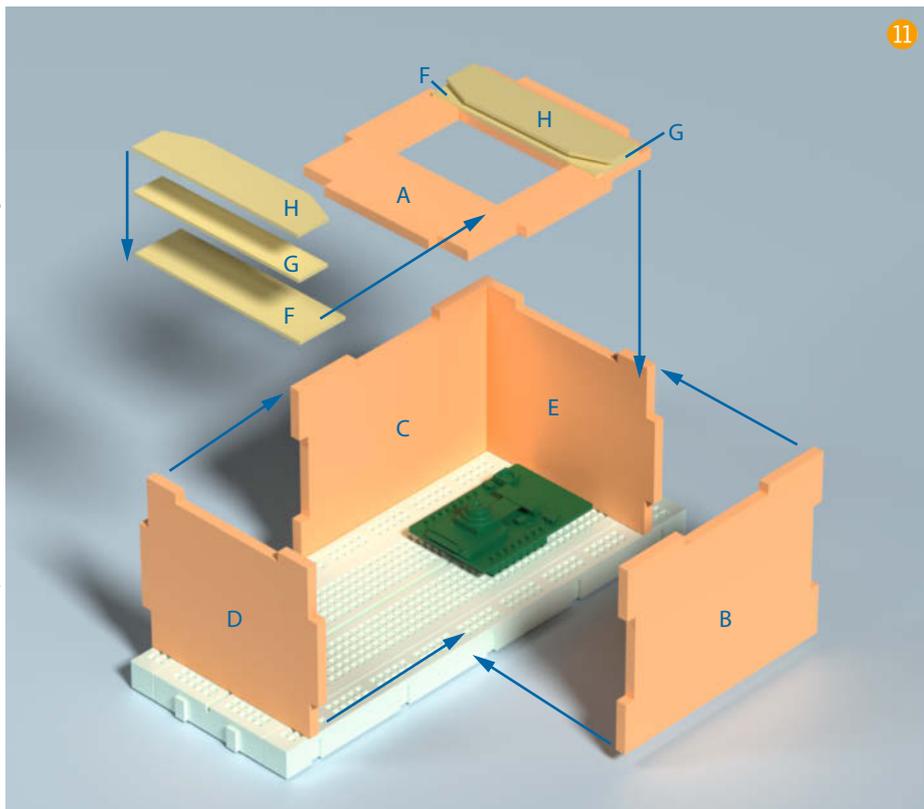
Schneiden Sie nach der Vorlage alle Teile zu recht und vergessen Sie beim Deckel **A** nicht den Ausschnitt in der Mitte für das Negativ (ja, wir sprechen da aus Erfahrung ...)! Geben Sie dann Alleskleber in die Aussparungen der Rückwand **C** und der Seitenteile **D** und **E** und kleben Sie diese untereinander und am Deckel **A** aneinander fest. Die Front **B** können Sie lose einsetzen, damit alles im Winkel ist, diese aber bitte noch nicht festkleben – sonst wird das Fokussieren später mühsam!

Dann kleben Sie im nächsten Schritt die beiden Schienen zusammen. Die Schienenteile sollten aus ungefähr 1mm dicker Pappe bestehen. Ist die Pappe dicker, wird die Führung eher lose, ist sie dünner, kann sich der Negativstreifen verklemmen. Haben Sie nur dünnere Pappe, kleben Sie lieber je zwei oder mehr Streifen zu doppelt so dicken Pappstreifen zusammen. Achtung: Die beiden



Unterteile **F** sind etwas breiter als die Ober- teile **H**, damit der obere Teil der Schiene das Negativ am Rand später nicht abschattet. Abschrägungen an den Enden von **H** erleichtern zudem das Einfädeln der Negativstreifen.





Kleben Sie die beiden Sandwiches jeweils aus **F**, dem Mittelstreifen **G** und dem Oberteilen **H** so zusammen, dass die später außen liegenden Kanten aller drei Streifen miteinander fluchten. Hat der Kleber abgetrocknet, die beiden Schienen mit den Unterteilen **F** so auf den Deckel **A** kleben, dass die Innenkanten von **F** genau auf den Kanten des Bildausschnitts in **A** zu liegen kommen (die gelben Flächen auf den Vorlagen zeigen an, wo genau die beiden Unterteile **F** sitzen müssen). Und bevor der Kleber anzieht, schieben Sie probeweise einen echten Negativstreifen (alternativ einen Papierstreifen von 35mm Breite) durch die Schienen, um sicherzustellen, dass der Streifen gut geführt wird.

### Das Fokussieren

Jetzt geht es dem Kameramodul der ESP32-CAM ans Objektiv. Das ist ins Gehäuse des Kameramoduls eingeschraubt und mit Lack fixiert, den es zu knacken gilt. Dazu trennt man das Board von der Stromversorgung, falls es gerade noch läuft, hält mit einer ausreichend langen Spitzzange oder einem ähnlichen Werkzeug den viereckigen Kamerakörper fest und dreht gefühlvoll mit einer zweiten Zange am Objektiv gegen den Uhrzeigersinn. Eventuell ist auch ein selbstgeschnittenes Werkzeug aus einer alten Scheckkarte hilfreich, wie seinerzeit bei den defokussierten Raspikameras (siehe Make 3/16).



Wenn Sie einen 3D-Drucker haben, ist das *ESP32-CAM Focus tool* des *Thingiverse*-Nutzers *AskePetter* **12** sehr nützlich: Stülpen Sie dieses einfach so auf das runde Objektivgehäuse, dass die kleinen Stege im Werkzeug in die Kerben rings um das Objektiv greifen und drehen dann gegen den Uhrzeigersinn, während Sie das Kameragehäuse festhalten.

Einmal gelöst, lässt sich das Objektiv leicht mit der Hand verdrehen. Durch sein Gewinde wird bei Drehung gegen den Uhrzeigersinn der Abstand zwischen Linse und Sensor größer, die Kamera wird sozusagen kurzsichtig. Das ist beabsichtigt, denn im Werkzustand bildet sie alles ab etwa 20cm Entfernung scharf ab, wir wollen aber bis auf rund 5cm heran, damit das Negativ möglichst formatfüllend abgebildet wird.

Zur Feineinstellung kommen wir aber gleich erst, vorher sollte das Kameramodul noch am Board befestigt werden, denn im Lieferzustand lömmelt es am Ende eines kurzen Folienbandkabels herum. Am einfachsten klebt man die Kamera mit einem Tropfen Alleskleber auf die darunterliegende Oberseite des SD-Kartenslots. Achten Sie dabei darauf, dass das Modul unbedingt kantenparallel zum Board fixiert wird, sonst hat der Gucki später Schlagseite. Natürlich kann man auch mit dem 3D-Drucker einen aufklipsbaren Halter drucken, aber falls sich das Modul unter dem verankert, ist seine Schiefelage schwierig zu korrigieren.

Setzen Sie jetzt die Bühne auf das Breadboard, sodass die Schmalseite der ESP-Platine mit der Beschriftung innen an die Wand des Gehäuses stößt. Möglicherweise verhindern noch die Nupsis an der Seite des Breadboards, die zum Zusammenstecken mehrerer davon gedacht sind, dass die Unterkanten der Bühnen-Seitenteile **D** und **E** auf dem Breadboard zu liegen kommen. In diesem Fall müssen Sie nochmal mit dem Cutter ran und eine passende Kerbe in die Front **B** oder die Rückwand **C** schneiden **13**. Da sich die Breadboards verschiedener Hersteller in manchen Details unterscheiden, sitzen die Nupsis nicht an einheitlichen Stellen und auch ihre Breite variiert, deshalb haben wir die Kerbe dafür in der Schneidevorlage noch nicht vorgesehen.

Schieben Sie dann ein möglichst kontrastreiches Negativ mit feinen Strukturen in die Schiene – alternativ tut es auch der eben erwähnte 35mm-Papierstreifen mit ein paar dünnen schwarzen Linien drauf. Versorgen Sie das Board mit dem Smartphone-Netzteil mit Strom und rufen Sie die Webseite mit dem Live-Kamerabild auf. Dann stellen sie die Kamera durch gefühlvolles Drehen am Objektiv auf das Negativ oder den Papierstreifen scharf. Falls Sie das 3D-gedruckte *Focus tool* benutzen, können Sie praktischerweise durch das Loch in dessen Mitte live beim Drehen auf das Negativ schauen.

Erscheint das Testbild scharf, können Sie die gefundene Position etwa mit einem Lackstift auf dem Objektivrand und dem Kamerahalter markieren, falls Sie befürchten, dass es sich beim Transport verstellt. Achtung: Vermeiden Sie möglichst, das Objektiv ganz heraus zu schrauben, da sonst sehr leicht Dreck auf den Sensor gerät, den man nicht mehr los wird ...

Jetzt darf man endlich auch die Front **B** am Rest der Bühne festkleben. Leider gibt es bei den Steckbrettern keine ganz einheitliche Breite, sodass es sein kann, dass die Bühne auf dem Breadboard nur lose sitzt und zu verrutschen droht. In diesem Fall kann man sie etwa mit einem knetbaren Klebepad wie *Patafix* sichern. Ein bisschen Schiebereserve ist aber auch ganz wünschenswert, denn die Bühne sollte so platziert werden, dass auf dem Kamerabild auf allen Seiten höchstens ein minimaler Rand der Bühne zu sehen ist und das Negativ später zentriert angezeigt wird.

### Die Beleuchtung

Wenn Sie jetzt den *Special Effect: Negative* aktivieren und die gewünschte Auflösung wählen, ist der ESP32-Gucki einsatzbereit und Sie können Ihre Negative am laufenden Filmstreifen sichten! Die visuelle Qualität der Anzeige ist allerdings davon abhängig, wie viel Licht durch das Negativ auf die Kamera fällt und wie gleichmäßig das passiert.

Praktisch und sehr hell sind natürlich spezielle Fotolampen **14**, die man in ein Tischstativ spannt. Eigens anschaffen wird man die aber nicht, sie kosten ein Vielfaches des ganzen Projekts. Kein Problem, es geht auch ohne. Liegt der Gucki mit dem Breadboard auf dem Tisch, reicht schon eine helle Zimmerdecke als Lichtquelle aus; abends kann man sie zusätzlich mit einem Deckenfluter aufhellen. Die auto-



matische Belichtungsreglung des Kameraservers liefert auch bei schwacher Beleuchtung noch erstaunlich gute Darstellungen ab.

Auch eine Schreibtischlampe, etwa 20 bis 30cm über dem Gucki platziert, kann gute Ergebnisse bringen, besonders, wenn man noch einen Streifen Butterbrotpapier, transparentes Zeichenpapier oder notfalls auch das dünnste weiße Papier in Reichweite als leicht gebogener Diffusor darüber anbringt. Heften Sie den Diffusor bevorzugt als Bogen an, dann hat das Papier genügend Abstand von der Fokusebene der Kamera, dass sich eventuell sichtbare Papierfasern nicht störend auf dem Live-Bild abzeichnen **15**. Auch Tageslicht ist nicht zu unterschätzen, besonders, wenn man es vom Fenster durch eine schräggestellte, möglichst helle Pappe, Styroporplatte oder ähnliches auf das Negativ reflektiert. Eine *Dritte Hand*, unsere *Locline*-Stative aus der *Make 1/22* (S. 68) oder Ähnliches kann hierbei als Halter gute Dienste leisten – hier ist Findigkeit gefragt, damit

Sie aus den vor Ort verfügbaren Mitteln die beste Beleuchtung herausholen.

Möchte man ein Bild nicht nur betrachten, sondern gleich auch festhalten, macht man einfach einen Screenshot beziehungsweise klickt auf *Get Still* und dann auf das eingefrorene Bild und speichert es als Grafik auf seinem Smartphone, Tablet oder Notebook. Natürlich geht das nur in der maximalen Auflösung der Kamera, also knapp 2 Megapixel – und für die Farben übernehmen wir keine Haftung. Unser Gucki ist eben nur ein *Gucki*, kein *Scanni* – im Gegenzug bleibt der Aufwand für dieses Projekt im Rahmen. Und noch ein Tipp: Widerstehen Sie der Versuchung, durch Wischen auf dem Smartphone zum nächsten Bild zu wechseln, das klappt nicht (sorgt aber sicher für allgemeine Erheiterung). Sie müssen schon zum Gucki greifen und den Negativstreifen dort von Hand weiterschieben. Aber jetzt: Viel Spaß beim Nachbau und beim Schwelgen in fotografischen Erinnerungen! —pek

# FLUX MAL WAS LASERN



LASER, DIE SICH ANPASSEN

ab 2.195,- € zzgl. MwSt.

flux-cameolaser.de

# Per Anhalter durch die Amiga-Disk-Galaxie

Wie man Amiga-Disketten vor dem Untergang bewahrt, diese weiter verarbeitet und anschließend seinen Amiga wieder damit füttert!

von Thomas Ronzon und Michael Schröder



**N**eulich fand ich (Michael) in einem meiner Schränke alte Amiga-Disketten. Die sind inzwischen mehrere Male mit umgezogen und sicherlich schon 35 Jahre alt. Ob die wohl noch funktionieren? Also holte ich meinen *Amiga 1200* aus dem Bettkasten, in dem er lange geschlafen hatte. Er war schnell am Fernseher angeschlossen, einen SCART-Anschluss gibt es ja zum Glück weiterhin. Hey, der Rechner läuft noch. Auch die Disketten scheinen – bis auf ein paar Ausnahmen – noch lesbar zu sein. Das erzählte ich meinem Freund Thomas (der andere Autor dieses Artikels), mit dem ich mich damals oft getroffen habe, um auf dem Amiga gegeneinander *Stunt Car Racer* oder *Ports of Call* zu spielen. Er berichtete, dass er auch noch einige alte Datensätze zu Hause hätte, die er gerne mal wieder sichten würde. Also überlegten wir, wie wir das am besten anstellen könnten.

Am schönsten wäre es ja, wenn wir die Daten auf den PC transferieren könnten, um sie in einem Amiga-Emulator (z.B. *UAE*, *Universal Amiga Emulator*) zu benutzen oder die Daten am PC weiterzuverarbeiten. Leichter gesagt als getan, da der Amiga ein proprietäres Diskettenformat verwendet und somit nicht mit einem Standard-PC-Diskettenlaufwerk und dem normalen (IBM)-PC-Controller gelesen werden kann.

### Der Amiga-Weg

Also untersuchten wir verschiedene Möglichkeiten: Eine serielle Verbindung vom Amiga zum PC wäre möglich, aber nicht nur, dass moderne PCs keine serielle Schnittstelle mehr besitzen und deshalb ein Seriell-zu-USB-Konverter verwendet werden müsste. Nein, dieser Weg ist zusätzlich noch sehr langsam.

Wenn der Amiga eine Ethernet-Karte hätte, könnte man das beispielsweise per FTP erledigen. Dazu werden die Disketten auf dem Amiga eingelesen, auf dem PC ein FTP-Server installiert, auf dem Amiga ein *TCP/IP Stack* (etwa *AmiTCP*) eingerichtet und dann per *FTP-Client* zum PC geschickt. Netzwerkkarten für den Amiga sind allerdings schwer zu finden und das Einlesen mit anschließendem Transfer zum PC wäre sehr mühselig.

Der Amiga 1200 hat eine *PCMCIA*-Schnittstelle, für die es *CompactFlash*-Karten-Adapter gibt. Damit kann man die Diskettenabbilder am Amiga auf eine *CompactFlash*-Karte kopieren. Danach kann die *CompactFlash*-Karte mit passendem Lesegerät wieder ausgelesen werden. Das geht recht zügig. Am Amiga müssen dafür aber erst mal ein PC kompatibles Filesystem und einige Patches installiert werden. Amiga-Besitzer ohne *PCMCIA*-Schnittstelle (wie 500er, 1000er oder 2000er) schauen aber in die Röhre.

### PC-Laufwerke modifizieren

Alles aus dem letzten Abschnitt erfordert aber einen Amiga mit funktionierendem Disklauf-

## Kurzinfo

- » PC-Laufwerke hacken, um Amiga-Disketten zu lesen
- » Arduino und Open-Source-Software geschickt kombinieren
- » Daten auf PC, Emulator und original Amiga nutzen

### Checkliste

-  **Zeitaufwand:**  
2 Stunden
-  **Kosten:**  
ca. 50 Euro
-  **Elektronik:**  
Lötfreie Verdrahtung
-  **Software:**  
Arduino-IDE, Drawbridge

### Material

- » **PC-Diskettenlaufwerk** mit 34-poligem Anschluss
- » **5V-Netzteil für das Laufwerk** einige Laufwerke benötigen auch 12V, dann ggfs. ATX-Netzteil
- » **Stromkabel für Diskettenlaufwerk** ATX-Netzteile bringen die gleich mit
- » **Arduino UNO** oder auch Nano, ggfs. USB-zu-Seriell Adapter
- » **kleines Breadboard**
- » **1kΩ Widerstand**
- » **Jumper Wires** Dupont-Verbinder, Buchse/Stecker

### Mehr zum Thema

- » Carsten Wartmann, Old Stuff: Amiga-Audio-Digitizer, *Make* 3/22, S. 100
- » Maik Schmidt, So geht Heimcomputer im Jahr 2020, *Make* 5/20, S. 24
- » Helga Hansen, Stiftespeicher, *Make* 6/15, S. 74
- » Philip Steffan, Floppy-Symphonie, *Make* 1/14, S. 148
- » Video: Floppy Action

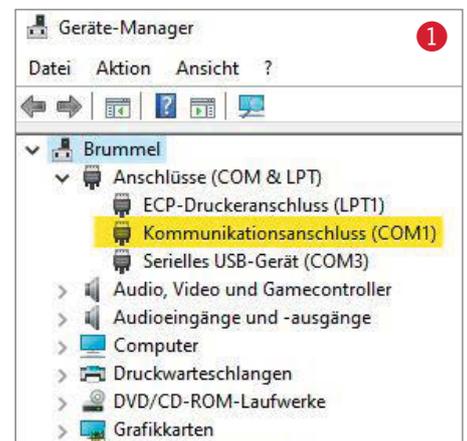


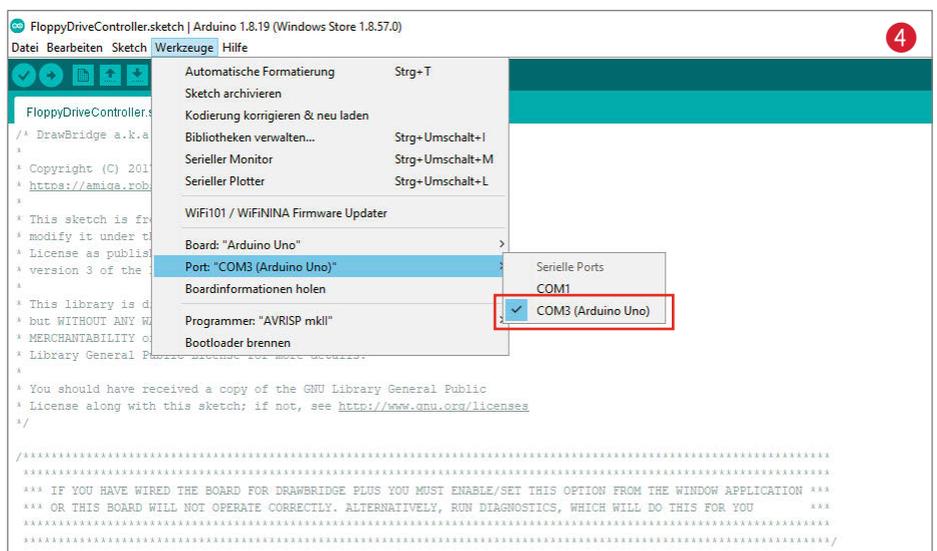
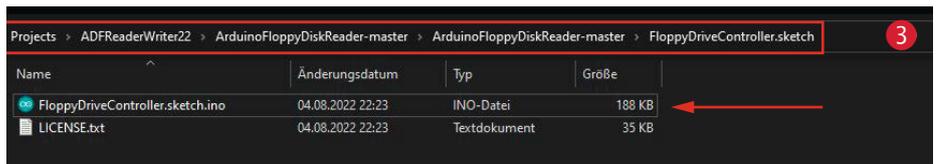
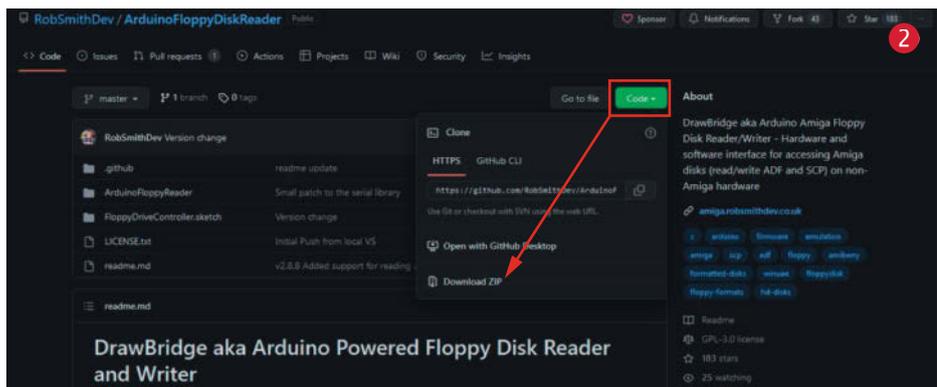
Alles zum Artikel im Web unter [make-magazin.de/xakh](https://make-magazin.de/xakh)

werk. Amiga-Laufwerke kommen mittlerweile in die Jahre und sind inzwischen sehr teuer, deshalb gibt es den *Arduino Floppy Disk Reader*. Mit einem Arduino-Board und einem normalen PC-Diskettenlaufwerk erlaubt dieser die Amiga-Disketten direkt am PC zu lesen und als Diskettenimages zu speichern. Der Arduino steuert das Diskettenlaufwerk und dekodiert die gelesenen Daten. PC-Laufwerke sind noch leichter zu bekommen und viel billiger als Amiga-Laufwerke. Aber auch PC-, *Atari-ST*-Formate und auch viele Diskettenformate von elektronischen Musikinstrumenten oder alten Messgeräten können so gelesen werden.

Schauen wir uns das Ganze einmal genauer an: Viel Hardware ist nicht nötig, ein PC-Diskettenlaufwerk, ein ATX-Netzteil, ein paar Kabel, ein Arduino, ein Breadboard oder etwas Löterfahrung. Genaueres finden Sie in der Kurzinfo und in den Links zum Artikel. Möchte man als Maker nicht alltägliche Aufgaben lösen, muss man nicht alles selber machen, es kommt nur auf die richtige Kombination der unterschiedlichsten Projekte und Werkzeuge an!

Eventuell wird ein USB-zu-RS232-Seriell-Konverter (am besten mit originalem FTDI-Chip) benötigt, wenn der verwendete Arduino keinen USB-Port hat. Beim *Arduino UNO* ist das nicht der Fall und somit wird keine weitere Hardware benötigt (zu anderen Arduino-Varianten siehe in den Links). In manchen Fällen gibt es laut Foren Probleme



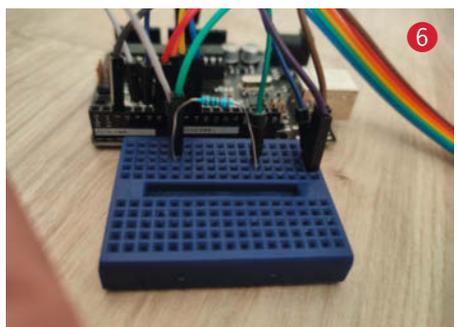


auf verschiedenen Arduino-Boards, die etwa CH340-basierende USB-Brücken verwenden. Mit unserem Arduino UNO sind bisher allerdings noch keine Probleme aufgetreten. Wenn man auch Amiga-Disketten beschreiben möchte, findet man die passenden Informationen auf der Projektseite des *Arduino Amiga Floppy Disk Readers*.



## Software

Unser Arduino UNO besitzt bereits einen USB-Anschluss und Amiga-Disketten beschreiben möchten wir auch nicht. Daher kann es gleich ans Übertragen der Firmware gehen, also des Programms, das die Wandlung der Daten vornimmt. Für unseren Elegoo-Arduino UNO kam



der passende Treiber praktischerweise mit der Arduino-IDE und musste nicht separat installiert werden. Danach wird das Arduino-Board mittels USB-Kabel an den PC angeschlossen. Dieses sollte nun als USB-Serial-Port im Windows-Gerätetmanager **1** erscheinen. Hier notiert man sich am besten gleich die entsprechende Adresse des COM-Ports, in unserem Fall COM3.

Als Nächstes lädt man die Firmware von der GitHub-Projektseite *Arduino Amiga Floppy Disk Reader*. Nachdem das Archiv **2** geladen ist, wird es entpackt und die Datei *FloppyDriveController.sketch.ino* **3** aus dem gleichnamigen Ordner in der Arduino-IDE geöffnet. Alternativ bringt auch *DrawBridge* (s. u.) den Sketch für den Arduino mit.

In der Arduino IDE wird das passende Board **4** und die zugehörige, aus dem Geräte-manager notierte serielle Schnittstelle ausgewählt.

Jetzt übersetzt man durch Drücken des *Hochladen*-Knopfes (Icon-Pfeil nach Rechts in **4**) das Programm und lädt es auf das Arduino Board.

## Hardware

Nun geht es daran, die Hardware zusammenzubauen. Das haben wir erst mal experimentell realisiert. Später, wenn alles funktioniert, kann dann alles zusammen in ein Gehäuse eingebaut werden, eventuell komplett mit einem kleineren Arduino-Board und einem richtigen Laufwerkskabel. Der Autor des Projektes stellt auf *Thingiverse* sogar eine Vorlage für ein passendes Gehäuse zur Verfügung.

Zunächst werden die Verbindungen mit Drahtbrücken gemäß der Tabelle hergestellt. Bei jedem Stecken und Abziehen der Verbindungen sollten der Arduino und das Laufwerk vom Netzteil getrennt sein. Die Verbindung kann mit Jumperkabeln **5** direkt an die Stiftleiste des Laufwerks erfolgen.

Die Verbindungen, die für das Schreiben von Disketten notwendig sind, stehen in Klammern. Mehr zu den Änderungen, um Amiga-Disketten auch schreiben zu können, finden Sie beim originalen Projekt. Es ist zu beachten, dass die Verbindung von */RDATA* an Arduino-Pin 4 von einem Pull-Up-Widerstand (1kΩ) auf 5V gezogen werden muss. Dazu bietet es sich an, diesen Teil der Verdrahtung auf ein kleines Breadboard **6** auszulagern. Die Schaltung sieht man zur Übersicht in **7** nochmals als Schaltplan.

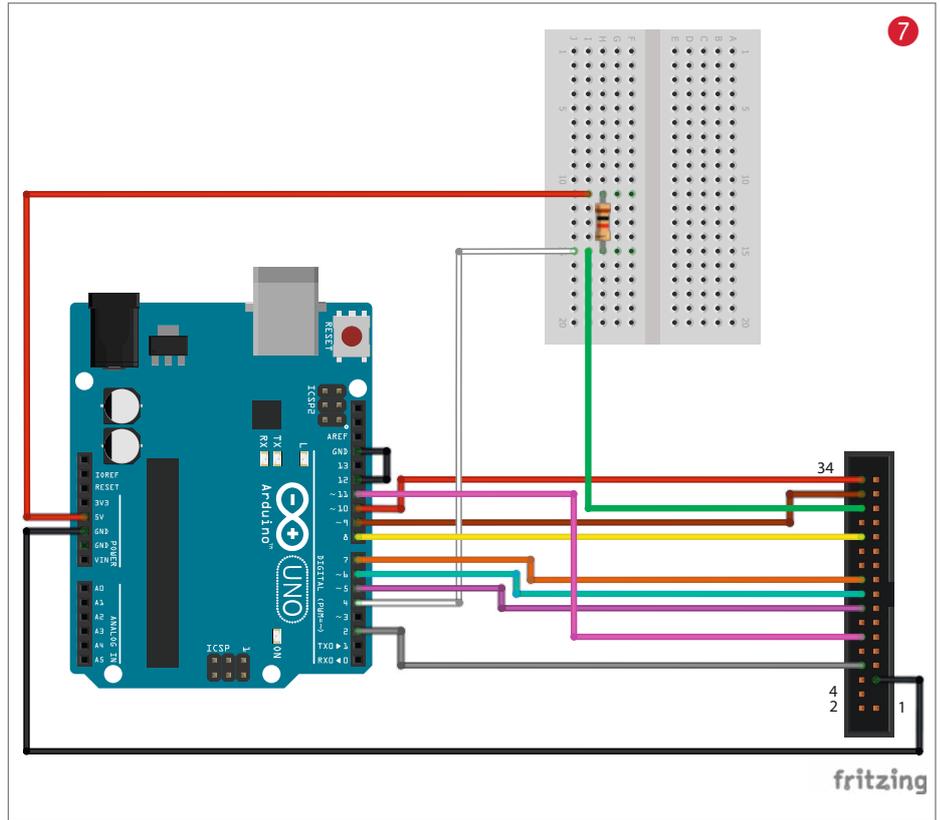
Wenn nun alles fertig verdrahtet und doppelt und dreifach überprüft ist, kann es ans Ausprobieren gehen. Zur Stromversorgung wird das Diskettenlaufwerk an das Netzteil angeschlossen und der Arduino per USB-Kabel mit dem PC verbunden. Um ein ATX-Netzteil manuell zu starten, verbindet man am ATX-Stecker *PS\_ON* mit Masse (GND). Vorsicht! Nicht die falschen Pins verbinden!

Als Nächstes lädt man sich die Software *DrawBridge* (Links unter der Kurz-URL) herunter und entpackt das ZIP-Archiv. Nun *DrawBridgeWin.exe* **8** starten. Im Programm stellt man wieder den passenden COM-Port des Arduino ein, die restlichen Konfigurationen passen für normale Amiga-Disks. Danach wählt man mit dem Button mit den drei Punkten aus, wohin das Diskettenabbild gespeichert werden soll, legt eine Amiga-Diskette ein und wählt *Start...*

Wenn alles geklappt hat, beginnt der Kopiervorgang und am Ende wird eine Erfolgsmeldung angezeigt **9**. Danach befindet sich das *ADF-Abbild* (Amiga Disk File, ein Disketten-Abbild im AmigaOS-Format) der Diskette auf der Festplatte des PC und kann weiter verarbeitet werden.

### Die weitere Reise

Nachdem wir diese einfache Möglichkeit gefunden hatten, ADF-Images zu erzeugen, ging es weiter. Schließlich wollten wir unsere selbst geschriebenen Programme und andere Dateien wieder auf dem PC lesbar machen. Möchte man die Disketten als Ganzes verwenden, um zum Beispiel ein Spiel zu starten, kann man diese direkt in einem Emulator wie UAE



# Keine Angst vor Python!



**Heft + PDF mit 29% Rabatt**

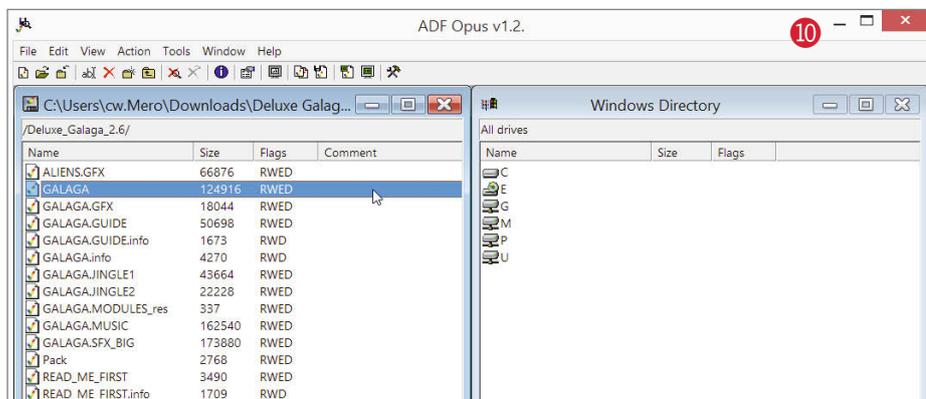
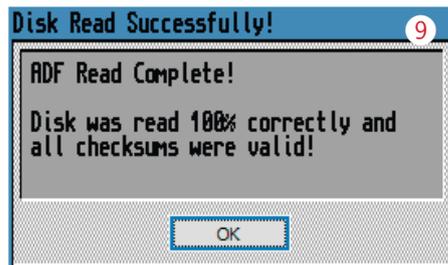
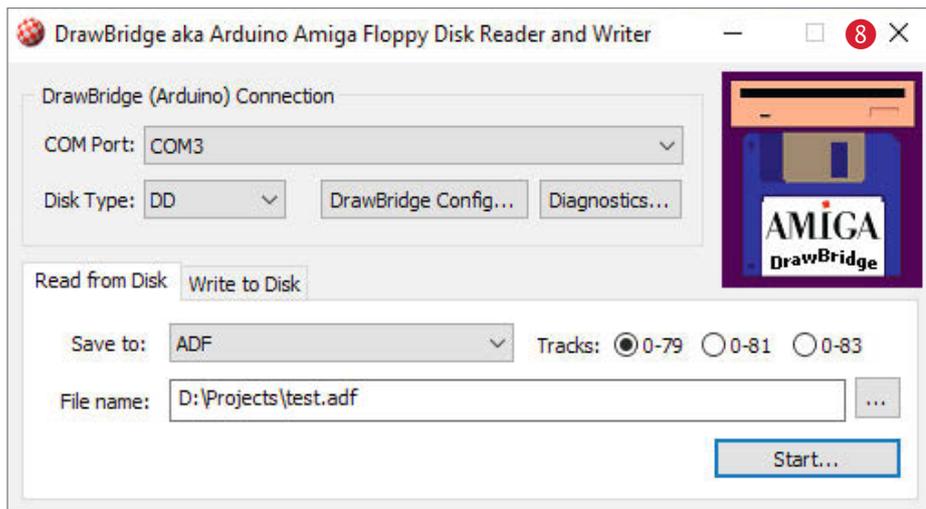
Mit diesem c't-Sonderheft überspringen Sie mühelos Einstiegshürden und erlernen in kurzer Zeit die Grundlagen des Programmierens mit Python. Außerdem zeigt Ihnen dieses Heft die erstaunliche Vielseitigkeit von Python anhand vieler praktischer Projekte:

- ▶ Python-Programme auf den Raspi Pico portieren
- ▶ Programmieren mit KI-Unterstützung
- ▶ Web-Programmierung mit Django
- ▶ inkl. GRATIS Python-Onlinekurs im Wert von 119,- €

**Heft für 14,90 € • PDF für 12,99 €  
Bundle Heft + PDF 19,90 €**

[shop.heise.de/ct-python22](https://shop.heise.de/ct-python22)

**GRATIS**  
Online-Kurs  
„Das Python-Bootcamp“



mounten. Natürlich kann UAE auch einzelne Dateien einer simulierten Disk auf den PC exportieren, nur ist dies etwas umständlich.

Unser Ansatz war daher ein anderer. Wir wollten die einzelnen Amiga-Dateien von Diskette lesen und dann auf dem PC weiter ver-

arbeiten. Dabei sind Linux-User besonders im Vorteil, da es Amiga-Filesystem-Kernel-Module gibt, die es erlauben, solche Images direkt zu mounten. Mehr dazu im Online-Teil.

Für Windows Benutzer bietet es sich das Tool *ADFopus* <sup>10</sup> an, den Link finden Sie

über den Kurzlink in der Kurzinfor. Auch wenn die letzte Änderung an ADFopus aus dem Jahr 2003 ist, so kann es trotzdem unter Windows 10 verwendet werden. Selbst mit *Wine* (Laufzeitumgebung, um Windows-Programme unter Linux laufen zu lassen) versteht es seinen Dienst ohne Probleme.

In der Online-Ergänzung zu diesem Artikel haben wir ein paar Tipps zusammengestellt, wie man die Dateien dann auf der PC-Seite weiterverarbeiten kann. Da das Image ja gemountet wurde, können die veränderten Dateien sogar wieder in dieses hineingeschrieben werden.

### Die nächste Fahrt geht rückwärts

Wo wir einmal wieder dabei waren, kam die Idee auf, ob man nicht die geänderten Images wieder mit einem Original-Amiga ablaufen lassen könnte. Das Ganze dann aber bitte mit etwas mehr Komfort als mit alten langsamen Disketten.

Die Firma *GoTek* baut Floppy-Disk-Emulatoren <sup>11</sup>, die man wie ein normales Diskettenlaufwerk in den PC oder andere Geräte einbauen kann. Diese Emulatoren lesen Diskettenabbilder von einem USB-Stick und stellen diese über die (IBM)-PC-Diskettenschnittstelle zur Verfügung. Man kann sogar mittels zweier Knöpfe am Emulator zwischen mehreren Diskettenabbildern wählen.

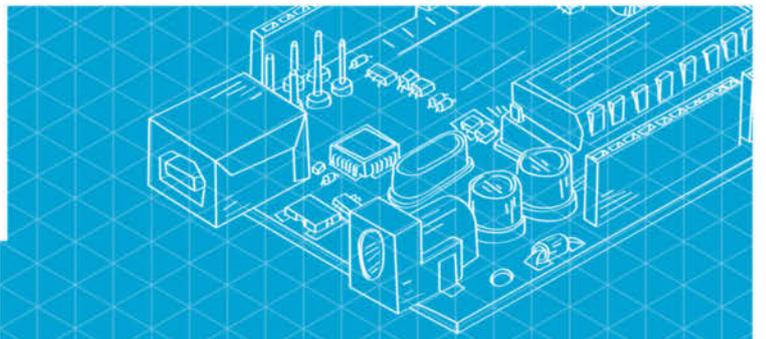
Leider funktioniert der unmodifizierte GoTek Emulator nur am PC. Und hier kommt jetzt das *FlashFloppy*-Projekt ins Spiel. FlashFloppy ist eine alternative Firmware für den GoTek-Floppy-Emulator. Es ersetzt die originale Firmware und unterstützt eine große Anzahl von verschiedenen Computersystemen. Unter anderem auch den Amiga. Weiteres finden Sie im Online-Ergänzungsartikel. —caw

## Pins und Verbindungen vom Laufwerk zum Arduino

| Laufwerk Pin <sup>1</sup> | Signal           | Beschreibung        | Arduino Pin                           |
|---------------------------|------------------|---------------------|---------------------------------------|
| 1                         | GND <sup>1</sup> | Ground              | GND                                   |
| (2)                       | /REDWC           | Density (DD/HD)     | (A3)                                  |
| 8                         | /INDEX           | Index               | 2                                     |
| 10                        | /MOTEA           | Motor enable A      |                                       |
| 12                        | /DRVSB           | Drive select B      | 11                                    |
| 14                        | /DRVSA           | Drive Select A      |                                       |
| 16                        | /MOTEB           | Motor enable B      | 5                                     |
| 18                        | /DIR             | Direction           | 6                                     |
| 20                        | /STEP            | Step                | 7                                     |
| (22)                      | /WDATA           | Write data          | (3)                                   |
| (24)                      | /WGATE           | Write enable        | (A0)                                  |
| 26                        | /TRK00           | Track 0             | 8                                     |
| (28)                      | /WPT             | Write protect       | (A1)                                  |
| 30                        | /RDATA           | Read data           | 4 (über 1KΩ Pull-Up Widerstand an 5V) |
| 32                        | /SIDE1           | Head select         | 9                                     |
| 34                        | /DSKCHG          | Disk change / ready | 10                                    |

<sup>1</sup> Alle ungeraden Pinnummern am Laufwerk sind GND. Die Pins 4 und 6 sind nicht benutzt (N/C)

# Make:



## DAS KANNST DU AUCH!



**GRATIS!**



## 2x Make testen und über 9 € sparen!

### Ihre Vorteile:

- ✓ **GRATIS dazu:** Make: Tasse
- ✓ Jetzt auch im Browser lesen!
- ✓ Zugriff auf Online-Artikel-Archiv\*
- ✓ Zusätzlich digital über iOS oder Android lesen

Für nur **16,10 € statt 25,80 €**

\* Für die Laufzeit des Angebotes.

Jetzt bestellen: [make-magazin.de/miniabo](https://make-magazin.de/miniabo)

# Raspi-Super8, Teil 2

Jetzt geht es mit der Raspberry-Super8-Filmkassette in die freie Wildbahn: Wir richten sie als eigenständigen WLAN-Accesspoint ein und können so auch ohne verfügbares Netzwerk per Smartphone oder Tablet darauf zugreifen.

von Heinz Behling



**E**insatzbereit ist die Raspi-Super8-Kamera aus der vorigen Make-Ausgabe ja schon. Allerdings ist ihr Arbeitsbereich noch auf das bei der Installation der Software eingestellte WLAN begrenzt. Das werden wir in diesem Artikel ändern: *MotionEyeOS*, das von der Kassette benutzte Betriebssystem, kann nämlich auch ein eigenes WLAN aufspannen, also als Accesspoint arbeiten. Darin kann man sich dann beispielsweise mit dem Smartphone einloggen und auf die Web-Oberfläche der Filmkassette und die aufgenommenen Filme zugreifen.

## Eigene Speicherkarte

MotionEyeOS kann entweder als WLAN-Client oder als Accesspoint arbeiten, aber nicht als beides gleichzeitig. Falls man nur eine Speicherkarte verwenden würde, müsste beim Wechsel von der einen zur anderen Betriebsart das Betriebssystem jeweils neu auf der Karte installiert werden. Das ist nicht nur unnötig kompliziert, sondern bringt auch den Verlust der auf der Karte gespeicherten Videos mit sich. Daher sollte man für den Accesspoint-Betrieb eine eigene Karte verwenden und dann bei Bedarf die SD-Karte im Raspberry wechseln. Das Gehäuse der Filmkassette ist so gestaltet, dass das ohne Öffnen möglich ist **1**.

Das Image des Betriebssystems MotionEyeOS ist dasselbe wie im ersten Teil. Falls Sie es noch nicht auf Ihren Computer übertragen haben, erhalten Sie es per Download über den Kurzinfo-Link. Achtung: Verwenden Sie nur das im Link angebotene Image. Da sich der Raspberry Pi Zero 2 in wichtigen Details von der Vorversion unterscheidet, ist die Verwendung von Images für den alten Zero nicht möglich!

Installieren Sie das Image auf der SD-Karte wie im ersten Teil beschrieben mit dem Programm *Pi Imager* (oder ähnlichen Tools wie *Balena Etcher*). Die Datei *wpa\_supplicant.conf* darf diesmal aber nicht auf der Karte angelegt werden, denn die Kassette soll ja keine Verbindung mit einem bereits vorhandenem Netzwerk aufnehmen.

Nach dem Schreiben des Betriebssystems nehmen Sie die Karte aber noch nicht aus dem PC. Sie müssen noch einiges hinzufügen.

## Konfigurationsdateien

Statt nur einer brauchen wir für den Accesspoint-Betrieb vier Konfigurationsdateien, mit denen alles Notwendige eingestellt wird. Im Einzelnen sind dies diese Dateien, die auch per Download zur Verfügung stehen:

- hostapd.conf
- dnsmasq.conf
- os.conf
- static\_ip.conf

In der Datei *hostapd.conf* **2** wird unter anderem der Name und das Passwort des neuen

## Kurzinfo

- » MotionEyeOS als Accesspoint einrichten
- » Videodownload aufs Smartphone
- » Einstellmöglichkeiten für die Aufnahme

### Checkliste



**Zeitaufwand:**  
1 Stunde



**Kosten:**  
10 Euro

### Material

- » Micro-SD-Speicherkarte mit mind. 32 GB
- » Raspi-Super8-Filmkassette und Kamera aus Make 4/22

### Mehr zum Thema

- » Heinz Behling, Raspi-Super8, Make 4/22, S. 42
- » Heinz Behling, Reingeschaut: Super8-Tonfilm-Kamera, Make 4/20, S. 88

Alles zum Artikel im Web unter [make-magazin.de/x2du](https://make-magazin.de/x2du)



WLANs festgelegt. Dies geschieht in den ersten beiden Zeilen. Ersetzen Sie dort WLAN-Name und WLAN-Passwort durch neue Einträge, zum Beispiel *raspi8* für den Netzwerknamen und *12345678* als Passwort (bitte wählen Sie ein besseres!). Beide Angaben brauchen Sie später, um sich per Smartphone oder anderen Geräten ins WLAN der Kassette einzuloggen.

Als Accesspoint muss die Filmkassette allen Geräte, die sich einloggen, IP-Adressen zuweisen. Der Inhalt der zweiten Datei *dnsmasq.conf* **3** legt den Adressbereich fest, der dazu benutzt wird. Hier sind es Adressen von *192.168.42.50* bis *192.168.42.150*. Aber Vorsicht: Auch wenn sich damit theoretisch 101 Geräte in dieses WLAN einloggen könnten, sollten Sie das nie ausnutzen. Dazu reicht die Leistung



**1** Die Speicherkarte ist von außen zugänglich und kann ohne Öffnen der Kassette gewechselt werden.

#### **4** static\_ip.conf

```
STATIC_IP="192.168.42.1/24"
STATIC_GW="192.168.42.1"
STATIC_DNS="192.168.42.1"
```

#### **5** os.conf

```
OS_NETWORKLESS="true"
```

#### **3** dnsmasq.conf

```
interface=ap0
dhcp-range=192.168.42.50,192.168.42.150,24h
dhcp-leasefile=/data/dnsmasq.leases
```

#### **2** hostapd.conf

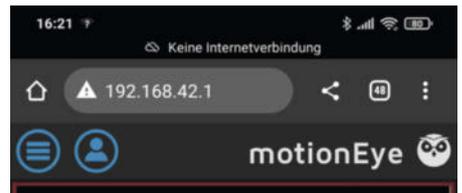
```
ssid=WLAN-Name
wpa_passphrase=WLAN-Passwort
wpa=2
wpa_key_mgmt=WPA-PSK
interface=ap0
channel=6
#driver=nl80211
hw_mode=g
ieee80211n=1
ieee80211ac=1
```



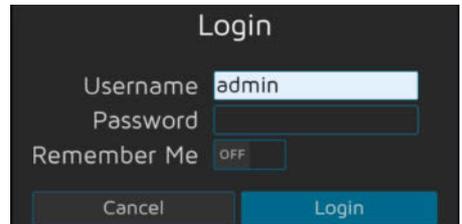
6 Wenn alles geklappt hat, erscheint hier das Filmkassetten-WLAN.



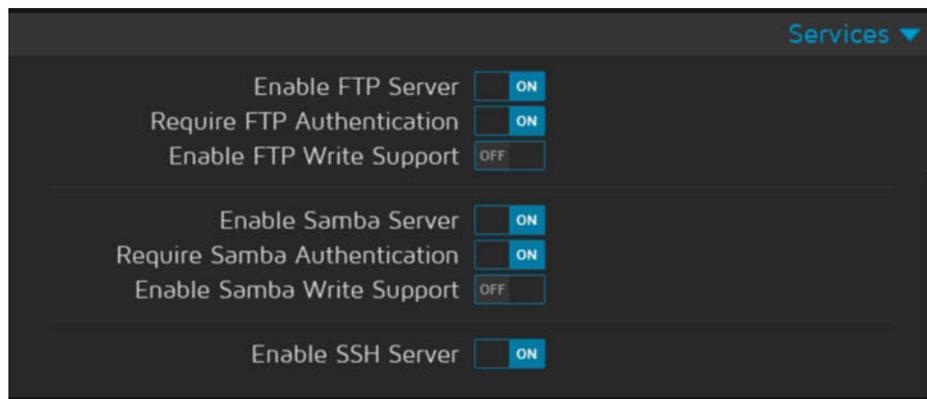
7 Einige Geräte wollen die Internet-lose WLAN-Verbindung noch bestätigt haben.



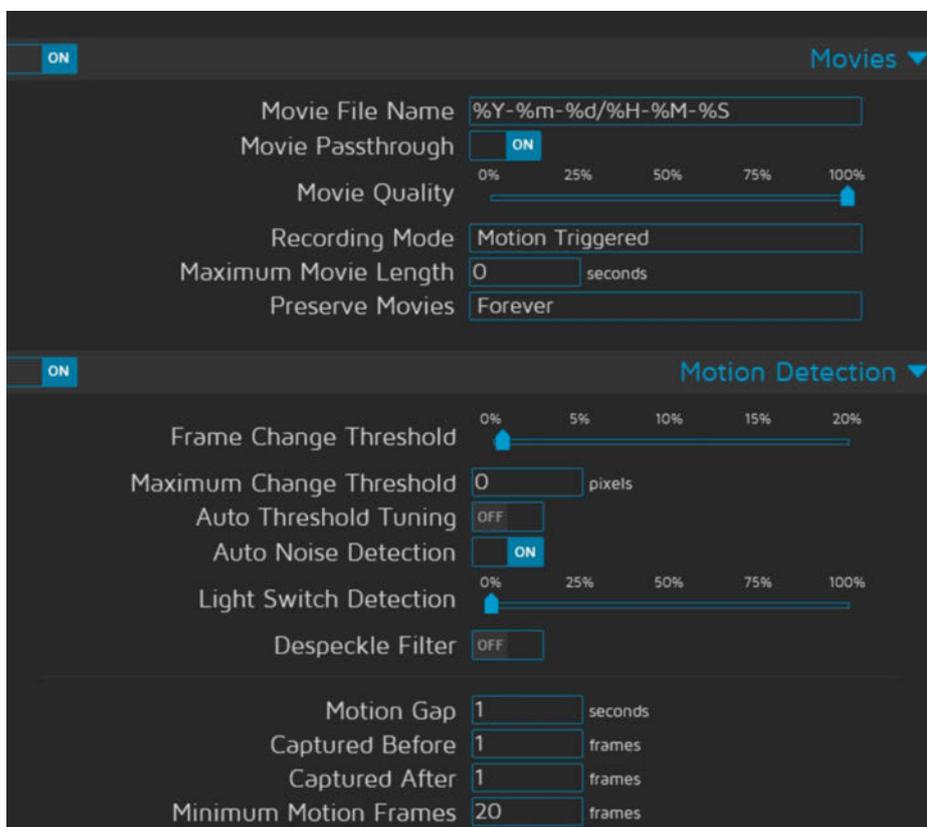
8 Geschafft: Die Web-Oberfläche auf dem Smartphone zeigt die funktionierende Verbindung.



9 Der voreingestellte Benutzername lautet admin.



10 Der Samba-Server ist wichtig für den Datei-Transfer aufs Smartphone.



11 Stimmt die Bewegungserkennung nicht, reagiert die Kassette nicht auf den Kamera-Auslöser.

des Rasperrys nicht aus. In der Praxis wird man aber kaum mehr als ein Gerät einloggen.

Der Accesspoint braucht aber auch eine eigene IP-Adresse, und die sollte immer gleich sein. Denn mithilfe dieser Adresse gelangen Sie im Browser der eingeloggten Geräte zur Web-Oberfläche der Filmkassette. In der Datei *static\_ip.conf* (4) wird diese Adresse eingestellt. Hier verwenden wir 192.168.42.1. Im Browser ist die Web-Oberfläche später dann mit *http://192.168.42.1* erreichbar.

Schließlich muss MotionEyeOS noch daran gehindert werden, beim Booten den Kontakt zu einem anderen Netzwerk zu suchen. Andernfalls würde es in einer Endlosschleife beim Bootvorgang ewig hängen bleiben. Diese Netzwerk-Blockade geschieht mit der Datei *os.conf* (5).

Falls Sie die Konfigurationsdateien nicht downloaden, sondern selbst eintippen, dann beachten Sie bitte, dass vor und nach Gleichheitszeichen keine Leerzeichen stehen dürfen. Andernfalls funktioniert das ganze nicht.

Die vier Dateien müssen vor dem ersten Booten ins Boot-Verzeichnis der SD-Karte (bei Linux-Computern/Macs) beziehungsweise in die FAT32-Partition (bei Windows-PCs) eingefügt werden. Erst danach melden Sie die Karte beim System ab und nehmen Sie aus dem Reader heraus.

Legen Sie die Karte in die Filmkassette ein, stecken Sie die Kassette in die Kamera und schließen Sie deren Deckel. Der Raspi im Inneren bootet nun. Der erste Start dauert ein wenig länger. Nach etwa ein bis zwei Minuten können Sie Ihr Smartphone nach dem neuen WLAN suchen lassen (6).

Loggen Sie sich in das Kassetten-WLAN ein. Dazu brauchen Sie das Passwort. Eventuell fragt das Smartphone noch einmal nach, ob Sie dieses WLAN trotz fehlender Internet-Verbindung wirklich verwenden möchten. Bestätigen Sie das einfach per *OK* oder *Ja* (7).

Sollte ein Einloggen nicht möglich sein, schauen Sie im Handbuch Ihres Smartphones nach, ob weitere Maßnahmen, etwa das Deaktivieren des mobilen Internets, notwendig sind.

### Web-Oberfläche einstellen

Starten Sie den Browser auf dem Smartphone und geben Sie die Adresse <http://192.168.42.1> ein. Die Web-Oberfläche erscheint **8**. Als Erstes sollten Sie einen Benutzernamen und ein Passwort eingeben. Das dient nicht nur dem Datenschutz, sondern wird auch vom noch zu aktivierenden Samba-Server benutzt. Und den brauchen wir, um die Videoaufnahmen aufs Smartphone überspielen zu können. Klicken Sie dazu auf das blaue Personensymbol neben motionEye.

Geben Sie dann als Benutzernamen *admin* ein **9**.

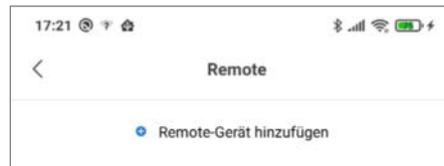
Sie gelangen zurück in die Web-Oberfläche. Per Klick auf das blaue Symbol mit den drei waagerechten Strichen gelangen Sie ins Konfigurationsmenü. Hier stellen Sie alles genauso ein wie im ersten Teil beschrieben. Wichtig sind hier insbesondere die Eingabe eines Passworts für den *admin* und die Aktivierung des Samba-Servers **10**.

Vergessen Sie auch nicht die Bild-Auflösung und die Bewegungserkennung **11**.

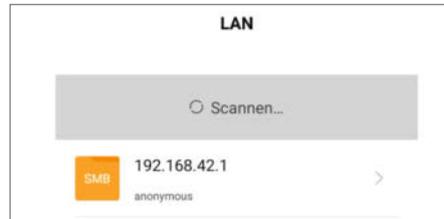
Nun haben wir alles zusammen. Sie können das Konfigurationsmenü wieder schließen (nochmal auf das blaue Symbol tippen). Zum Test drücken Sie einmal den Auslöser der Kamera: Auf dem Smartphone erscheint dann das aufgenommene Bild.

### Videodownload aufs Smartphone

Um Videos von der Filmkassette aufs Smartphone kopieren zu können, brauchen Sie dort



**12** Erst muss der Samba-Server hinzugefügt werden, um an die Videos zu kommen.

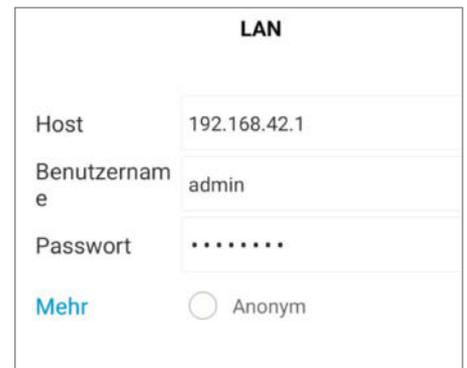


**13** Falls die IP-Adresse des Samba-Servers nicht gefunden wird, müssen Sie sie von Hand eintippen.

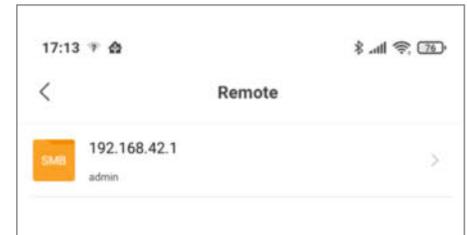
einen Samba-Client. Auf einigen Android-Smartphones ist bereits ein Samba-fähiger Dateimanager installiert. Ansonsten erhält man im App-Store zahlreiche entsprechende Programme. Für iPhones dürfte das ähnlich sein.

Dem Dateimanager müssen Sie zunächst die IP-Adresse und die Zugangsdaten des Samba-Servers mitteilen. Das geschieht meist unter dem Menüpunkt *remote*. Je nach verwendetem Programm müssen Sie eventuell erst das Konfigurationsmenü der Software öffnen. Dann tippen Sie auf *Remote-Geräte hinzufügen* **12**.

Anschließend wählen Sie eine Samba (smb)- beziehungsweise LAN-Verbindung. Nach einem kurzen Scannen wird Ihnen die IP-Adresse der Filmkassette meist bereits angezeigt. Tippen Sie einfach darauf **13**.



**14** Hier bitte Groß- und Kleinschreibung beachten!



**15** Hinter diesem Symbol stecken die Ordner der Filmkassette.

Dann sind noch der Benutzername und das Passwort einzugeben. Beides ist mit den Eingaben in der Web-Oberfläche identisch, also *admin* und das dazugehörige Passwort **14**.

Wenn Sie danach auf den Eintrag des Servers tippen **15**, gelangen Sie in die Ordnerstruktur der Filmkassette. Die Videos finden Sie dort unter *SD-Card* und *Camera1*.

Damit ist die Raspberry-Super8-Kamera nun auch für den Betrieb in freier, WLAN-loser Wildbahn gewappnet. Viel Spaß beim Aufnehmen Ihrer Kurzfilme. —hgb

Lebe deinen Spaß!

modell hobby Spiel

30.9. - 2.10.2022

www.modell-hobby-spiel.de

LEIPZIGER MESSE

SAFE EXPO Leipzig Messe

f /modellhobbyspiel

# So war die Maker Faire Hannover 2022

Drei Jahre lang konnte die Maker Faire Hannover nicht in der gewohnten Form stattfinden, jetzt endlich war es wieder möglich – und fühlte sich an, als wäre nichts gewesen: Rund 12.600 Besucherinnen und Besucher, Roboter und Fabelwesen, Steampunks und Cosplayer (und Batman!) bevölkerten am 10. und 11. September das HCC in Hannover. Wer da war, bestaunte die ausgestellten DIY-Projekte, legte in Workshops selbst Hand an, hörte spannende Vorträge und verfolgte bei bestem Spätsommerwetter das bunte Programm auf der Showbühne auf dem Außengelände. Zwei Highlights im Wortsinn spuckten – klar – Feuer, das muss sein: Die Tiergottheit *Garuda* aus der hinduistischen Mythologie breitete dabei ihre Flügel bis zu einer Spannweite von 15 Metern aus, sobald ihr Erbauer Matthias Vijvermann das Kommando per Fernbedienung gab. Und aus dem überdimensionalen Doppel-Laufrad *MansterWheel* des tschechischen *PropanePunk Circus* schlugen ebenfalls Flammen. Etwas weniger krachend, aber desto stimmungsvoller diente zum zweiten Mal eine der drei Ausstellungshallen im HCC als *Dark Gallery*, in der interaktive Lichtinstallationen und Leuchtkunstwerke besonders intensiv zur Geltung kamen.

Wie es bei Maker Faires guter Brauch ist, gab es Inspiration und Anleitung bei einer Menge Mitmach-Aktionen – so konnte man Roboter bauen, Programmieren und Lötten lernen, 3D-

Scanner oder das Upcycling von Kunststoffen zu 3D-Druck-Filamenten testen. Alternativ wurden Schlüsselbänder genäht, Wasserraketen gestartet, Linoldruck ausprobiert und Gold geschmiedet.

Wer beim *Make*-Stand vorbeikam, traf dort nicht nur auf Projekt-Klassiker wie den Tee-Hasen, die Sanduhr 2.0 und das Oszi-Röhren-Asteroids, die Redaktion zeigte dort auch ganz aktuelle Projekte, etwa die Posteule mit Cyborg-Hülle und ESP32-Kern (Seite 46) sowie die Kombination aus Solar-Wetterstation mit



Ob Wall-E (der zum Aufräumen nach der Messe sicher auch praktisch ist) ...



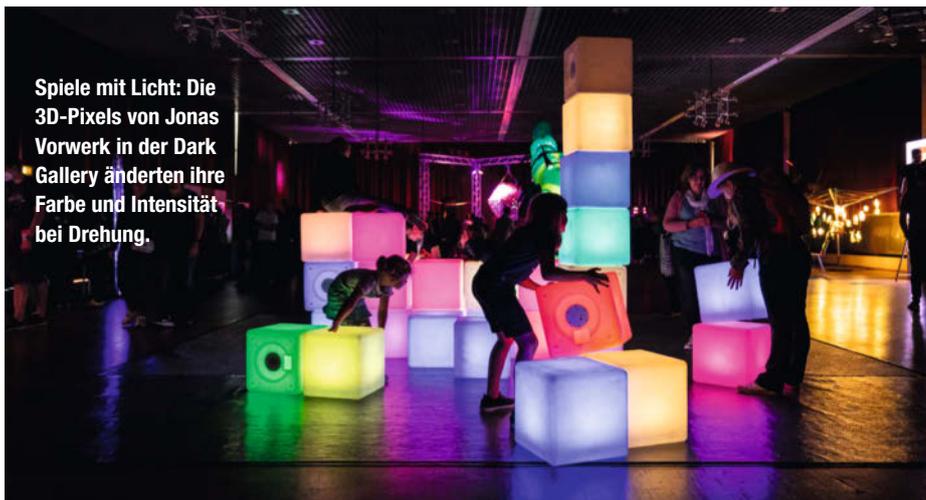
... oder Pit Droid: Roboter gehören fest zur Maker Faire. R2-D2 und BB-8 waren gleich mehrfach da.



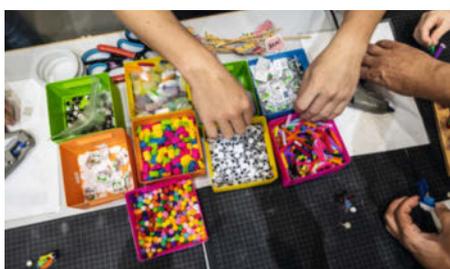
Dale Dougherty, Gründer aller Maker Faires, auf der Showbühne zusammen mit Guido Burger.



Zwischen Posteule und Oszi-Röhren-Asteroids: Projekte zum Anfassen auf dem Make-Stand.



Spiele mit Licht: Die 3D-Pixels von Jonas Vorwerk in der Dark Gallery änderten ihre Farbe und Intensität bei Drehung.



Maker Faire heißt: Selber machen! Hier werden zum Beispiel Zahnbürstenroboter geboren.

Wettermühle (Seite 58 und 64). Auch anderswo waren Artikel und Projekte aus den vergangenen *Make*-Ausgaben präsent – wir freuen uns sehr, dass so viele unserer Autorinnen und Autoren mit eigenen Ständen an der Maker Faire teilnahmen und dort ihre Werke zeigten.

So ist die Maker Faire Hannover seit dem Start im Jahr 2013 nicht nur ein Ort des Staunens für Besucher, sondern auch zu einem wichtigen Treffen für den Austausch und die gegenseitige Inspiration innerhalb der Maker-Szene geworden. Das gilt auch für die DIY-YouTuber, die sich seit Jahren auf der Maker Faire treffen und auch diesmal vor Publikum diskutierten und Impulse aus der Handwerkszene zum Nachahmen ans Publikum weitergaben. Besonders gefreut haben wir uns *last but not least* auch über hohen Besuch aus den USA: Dale Dougherty, der Gründer des *Make: Magazine* aus den USA und auch der Vater aller Maker Faires, war von der Eröffnung bis Sonntag vor Ort und im Gespräch mit unserem Autoren Guido Burger auch live auf der Showbühne zu sehen. Und wir konnten ihn persönlich mit Daniel Springwald bekannt machen, der mit seiner in *Make* 4/20 beschriebenen animatronischen Tesla-Büste gerade einen der von den US-Kollegen ausgelobten *Amazing Maker Awards* gewonnen hat. Vielen Dank an alle, die mitgemacht haben und dabei waren – wir sehen uns spätestens nächstes Jahr in Hannover auf der Maker Faire 2023 wieder!  
—pek



Drinnen laufen, draußen Feuer: MansterWheel vom PropanePunk Circus aus Tschechien.



Privileg der Mitmacher: Garuda bei Nacht sah nur, wer aktiv als Aussteller oder Helfer bei der Maker Faire dabei war – beim Get-together am Samstagabend.



Keine Maker Faire ohne Steampunk – ein Style, der sogar Furby steht.

## Kommende Maker Faires

**In diesem Jahr öffnet noch eine weitere Maker Faire in Deutschland ihre Pforten für Besucher:** In Kooperation mit der Gottlieb-Daimler-Schule 2 veranstaltet das *Jugendforschungszentrum Energie und Umwelt Landkreis Böblingen* am **19. November 2022** bereits die fünfte **Maker Faire Sindelfingen**. Der Call for Makers läuft noch bis zum 30. September. Alle weiteren Informationen gibt es unter:

► [www.maker-faire-sindelfingen.de](http://www.maker-faire-sindelfingen.de)

**Und fürs nächste Jahr gibt es auch schon einen Termin vorzumerken:** Die **Maker Faire Hannover 2023** findet am **19. und 20. August 2023** wie gewohnt im *Hannover Congress Centrum* (HCC) statt. Wir freuen uns jetzt schon! Weitere Informationen dazu wird es beizeiten wie üblich auf der Webseite zu den Maker Faires im deutschsprachigen Raum geben:

► [maker-faire.de/hannover](http://maker-faire.de/hannover)



Python ist auf dem PC und *Raspberry Pi* sehr beliebt und man findet für fast jedes Problem Lösungen und Hilfs-Bibliotheken (Libraries, Module). Oftmals ermöglichen diese dann auch rechenintensive Aufgaben in Python zu lösen, die sonst nicht mit einer Interpreter-Sprache zu bewältigen sind. Die Varianten *MicroPython* und *CircuitPython* bringen Python auf Mikrocontroller, die als Arbeitspferde für uns Maker dienen und bisher eher in C, Arduino-C oder gar Assembler programmiert wurden.

Auch wenn *CircuitPython* und *MicroPython* die hardwarenahe Programmierung der Mikrocontroller leicht machen, so kann dies mit der *Picozero*-Bibliothek noch weiter vereinfacht werden: Genau dies ist die Intention der *Raspberry Pi Foundation*, um besonders Einsteigern die ersten Schritte zu erleichtern. Aktuell ist diese Bibliothek mit der Version 0.3.0 noch in der Betaphase, kleine Änderungen können mit dem Release entstehen. Die Skripte sollten aber leicht anzupassen sein. Wir halten Sie auf dem Laufenden.

## Thonny und Firmware

*Thonny* ist ein Python-Editor, der besonders für Einsteiger gut geeignet und recht übersichtlich ist. Damit werden die Python-Skripte eingegeben, auf den Controller geschickt und im Fehlerfall untersucht.

Die Installation von *Thonny* kann per Installer auf Windows oder den Paketmanager von Linux oder macOS geschehen. Wir verwenden in diesem Artikel Version 4.0.0. In einigen Bereichen von *Thonny* erscheinen in dieser Version nicht übersetzte englische Texte. Auf Windows-Systemen benutze ich gerne die portable Version, die einfach durch Entpacken des ZIP-Archivs und Starten der Anwendung aus dem Ordner benutzt werden kann.

Haben Sie einen fabrikneuen *Pico*, so stecken Sie ihn per USB an den Rechner. Es wird sich ein Laufwerkfenster öffnen bzw. das *Pico*-Laufwerk im Dateimanager auftauchen. Passiert dies nicht, so befindet sich bereits eine Firmware auf dem Board. In diesem Fall trennen Sie die USB-Verbindung wieder und stecken sie das Kabel wieder ein, während Sie den *BOOTSEL*-Knopf auf dem Board gedrückt halten.

In *Thonny* können Sie nun über das Menü rechts unten im *Thonny*-Fenster **1** den Board-Manager anklicken. Dies kann ein paar Sekunden dauern, während *Thonny* die Schnittstellen absucht.

Hier wählen Sie dann *MicroPython installieren...* aus. In der Auswahl *MicroPython variant* **2**, dann passend zu Ihrem Board *Raspberry Pi Pico* oder *Raspberry Pi Pico W* auswählen. Das *H* bei den Varianten meint übrigens einfach Boards mit Pinheader und ist nicht weiter in-

## Kurzinfo

- » Einstieg in Python mit Pico und Thonny
- » Einfach programmieren mit Picozero, einer Bibliothek der Raspberry Pi Foundation
- » LEDs, Taster, Motorsteuerung

## Checkliste



### Zeitaufwand:

1 Stunde



### Kosten:

ab 5 Euro

## Material

- » Raspberry Pi Pico oder Pico W
- » Micro-USB-Kabel
- » Breadboard und Jumperkabel
- » LEDs, Taster, Motoren, Motor-Treiber, was die Bastelkiste so hergibt
- » Widerstände für die LEDs, 220Ω

## Werkzeug

- » Lötwerkzeug optional für permanente Verbindungen

## Mehr zum Thema

- » Daniel Bachfeld, Pi Pico mit WLAN, Make 4/22, S. 9
- » Hans-Martin Hilbig, Stromausfall-Monitor mit dem Pico, Make 4/22, S. 62
- » Detlef Heinze, Pfadplanung für Roboter, Make 1/21, S. 78
- » Thomas Euler, MicroPython beschleunigen, Make 1/21, S. 72
- » Helen Leigh Steer, Python auf Hardware, Make 6/20, S. 90



Alles zum Artikel im Web unter [make-magazin.de/xgkv](https://make-magazin.de/xgkv)

teressant. Anschließend die neueste Version wählen (wir benutzen hier die stabile 1.19.1). Die Dateien werden jetzt aus dem Internet geladen und auf das Board gespielt.

Jetzt kann das Board im Manager-Menü **1** ausgewählt werden. Sobald dies geschehen ist, meldet sich der *Pico* in der Kommandozeile. Sollte diese nicht sichtbar sein, so können Sie dieses Fenster im Menü *Ansicht/Kommandozeile* einblenden.

Klicken Sie einmal mit der Maus in das Kommandozeilen-Fenster **3** und geben Sie `print("Hello World")` gefolgt von der Eingabetaste ein. Sie haben gerade den ersten Python-Code auf dem *Pico* ausgeführt.

## Picozero aufspielen

Jetzt besitzen Sie einen *Raspberry Pi Pico* mit *MicroPython*. Nun geht es daran, die *Picozero*-Bibliothek einzubinden. Wählen Sie *Extras/Verwalte Pakete...* aus *Thonnys* Menüleiste. In der ersten Zeile des Dialogs **4** geben Sie nun als Suchbegriff *picozero* ein und klicken *Suche auf PyPi* an. *PyPi* ist ein Softwareverzeichnis für Python-Erweiterungen. Nach ein paar Sekunden wird ein Ergebnis angezeigt. Klicken Sie auf den blauen Link *picozero* und es werden noch weitere Informationen

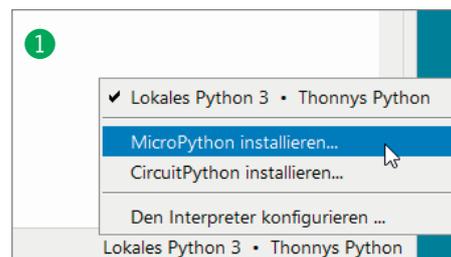
angezeigt. Im Moment ist die Version 0.3.0 aktuell.

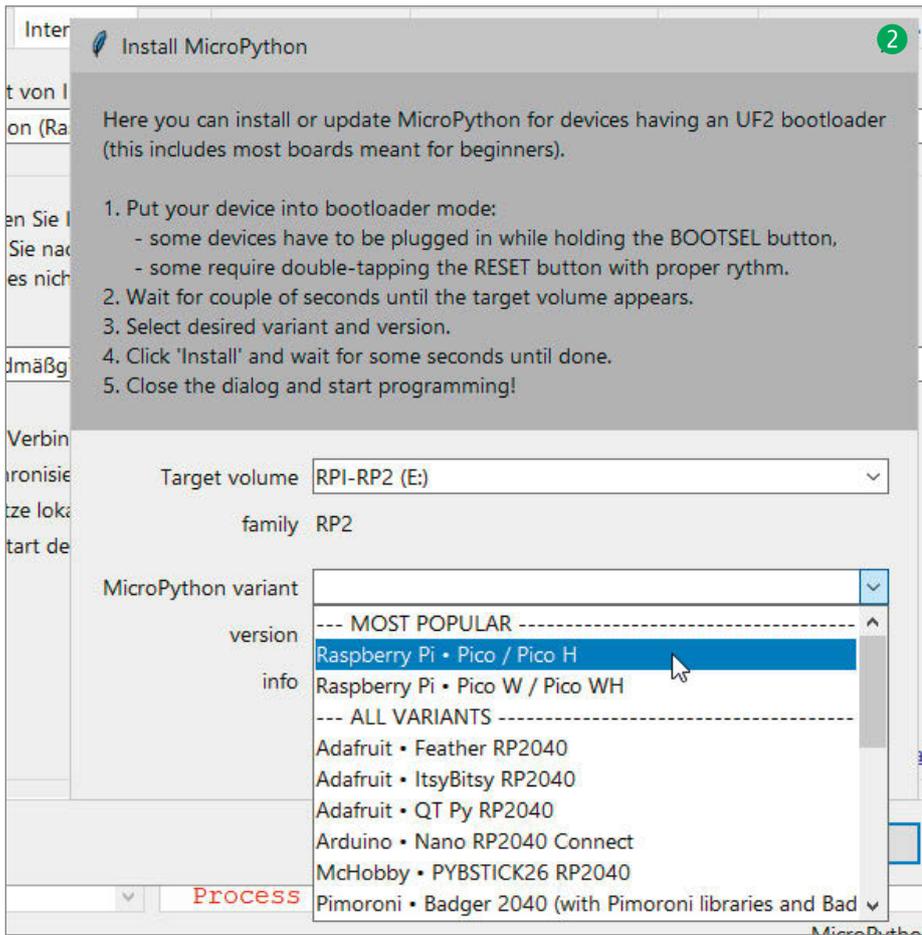
Durch Klick auf *Installieren* läuft der Prozess durch, dabei werden die nötigen Dateien auf dem *Pico* in dem Ordner */lib* installiert und sind jetzt auf diesem *Pico* benutzbar. Schließen Sie den Paketverwalter.

## Picozero erste Schritte

Für den Anfang benutzen wir die Kommandozeile, um ein wenig herumzuprobieren. Dies führt auch gleich eine tolle Eigenschaft von Interpretersprachen vor: Kein langwieriges Übersetzen und Aufspielen von Programmen ist nötig.

In einem *Datei*-Fenster (Menü *Ansicht/Dateien*) werden die Dateien auf dem *Pico*





und auf dem PC angezeigt: Erweitern Sie doch einmal mit den +-Knöpfen den Ordner *lib* und dann den *picozero*-Ordner. Hier wurde *picozero.py* installiert und wartet auf die Benutzung.

Klicken Sie in die Kommandozeile und tippen ein:

```
from picozero import pico_led
```

Nach Abschluss mit der Eingabetaste erscheint wieder der Prompt `>>>`, nur eine kleine Verzögerung gibt einen Hinweis darauf, dass etwas geschehen ist: Picozero ist nun im Kon-

text der Kommandozeile benutzbar. Tippen Sie nun, wieder gefolgt von der Eingabetaste:

```
pico_led.on()
```

Die LED auf dem Board leuchtet auf. Sie vermuten es vielleicht schon: `pico_led.off()` schaltet diese wieder aus. Versuchen Sie doch einmal folgenden Befehl:

```
print(pico_led)
```

Die Ausgabe bringt einen weiteren Aspekt von Python zutage: Python ist eine selbstdokumentierende Sprache und gibt uns Hinweise,

was es mit Python-Variablen und Objekten auf sich hat. Die Ausgabe *DigitalLED (pin 25)* sagt uns das die LED an Pin 25 des Picos sitzt, was bei weiteren Boards mit MicroPython anders sein kann. Auf dem Pico W benutzt man mit LED einen symbolischen Namen, weil auf diesem Board die LED nicht direkt per GPIO am RP2040-Prozessor angebunden ist, sondern über das WLAN-Modul. Nach außen hin funktioniert aber ein Skript, das `pico_led` benutzt, auf beiden Boards gleich, die Hardware wurde abstrahiert.

Noch weiter geht folgender Befehl (Ausgabe gekürzt):

```
>>> print(dir(pico_led))
['_class__', ... 'close', 'value', 'off', 'on', 'toggle', 'pin', 'blink' ...]
```

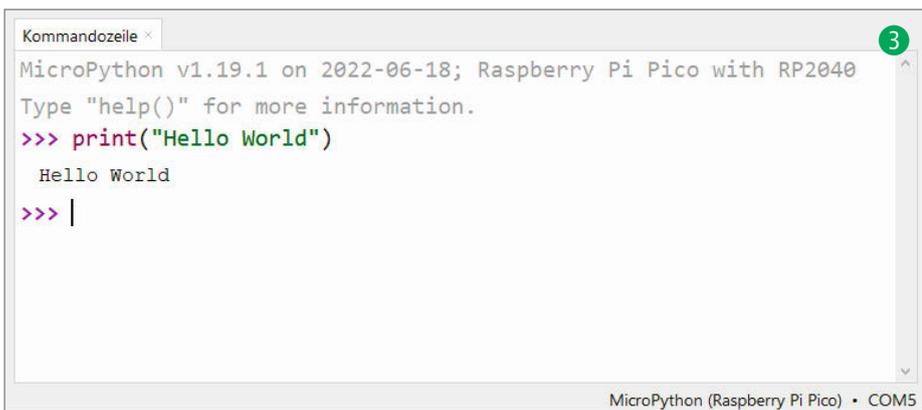
Dies gibt alle Eigenschaften und Methoden (Funktionen) des `pico_led`-Objekts aus. Hier finden Sie auch die Methoden `on` und `off` wieder, die wir ja oben schon benutzt haben. Die Klammern `()` für eventuelle Übergabeparameter müssen mit eingegeben werden. Versuchen Sie doch einmal die Methode `toggle()`. Die LED wird bei jedem Aufruf umgeschaltet. Mit *Pfeil-Hoch* und *Pfeil-Runter* können Sie in der Kommandozeile von Thonny übrigens in der Liste der bereits verwendeten Befehle navigieren und sich so eine Menge Tipperei sparen.

Eine weitere spannende Methode ist `blink()`. Probieren Sie einmal `pico_led.blink(0.5)`, die LED wird mit 1Hz blinken. Der Parameter `0.5` gibt die Ein- und Ausschalt-dauer der LED im Blinkzyklus in Sekunden an. Warum man nicht die Frequenz gleich in Hertz angeben kann? Versuchen Sie einmal `pico_led.blink(0.1,0.5)`: durch die per Komma getrennten Parameter definieren Sie die Ein- und Ausschalt-dauer der LED getrennt.

Hier hört dann das Spicken per `dir()` für MicroPython (nicht so in Python auf PCs) leider auf, weiteres ist aber im Quellcode und in der *API-Dokumentation* (Application Programming Interface, Links über die Kurzinfo) von *Picozero* beschrieben. Diese API-Dokumentationen sollte man immer unter dem Kopfkissen oder wenigstens in seiner Linkliste parat haben, wenn man mit Bibliotheken arbeitet. Der Aufruf der `blink`-Methode sieht dort dann so aus:

```
blink(on_time=1, off_time=None,
      n=None, wait=False)
```

In der weiteren Beschreibung sind die einzelnen Parameter genauer beschrieben, `on_time` ist z. B. die Einschalt-dauer in Sekunden, `n` gibt die Anzahl der Blinkzyklen an, `wait` definiert, ob der Befehl die Kontrolle an das folgende Programm sofort übergibt (die LED blinkt unabhängig vom weiteren Programm) oder erst komplett abgearbeitet wird (`n > 0`), bis die gewünschte



## Strom an den GPIOs

Der Spannungsregler auf dem Pico-Board liefert laut Datenblatt 100mA und ist gegen Kurzschluss und Überlast abgesichert. Alle Stromabgaben an den GPIOs des RP2040-Chips sollen 50mA nicht überschreiten (Datenblatt RP2040). Ein GPIO liefert bei Kurzschluss gegen Masse ca. 30mA, kaputt geht dabei nichts, wir sind aber schon nahe am Limit des ganzen Systems. Zu bedenken ist auch, dass mit steigender Strom-Belastung des GPIO die Spannung sinkt. Dies stört eine LED weniger, wenn aber zugleich noch ein anderer Chip die Signale auswerten soll, wird bei zu geringer Spannung der Logiklevel nicht mehr eindeutig sein und es treten Fehler auf.

Blinkzahl erreicht ist. Achtung: bei `wait=True` wird mit `n=0` dauerhaft die weitere Ausführung des Programms blockiert, welches dann per `Strg-C` abgebrochen werden muss.

## Eine Schaltung

Zum warm werden schalten wir nochmals eine Leuchtdiode, diesmal eine externe LED auf einem Breadboard wie im Schaltbild 5 von Fritzing dargestellt.

Wegen des 220-Ohm-Widerstands fließen nur etwa 5mA (siehe Kasten *Strom an den GPIOs*) durch die Leuchtdiode, genug für moderne LEDs und nicht blendend mit älteren LEDs. Die Anode (langes Beinchen, im Bild oben) der LED ist über den Widerstand mit GPIO15 verbunden.

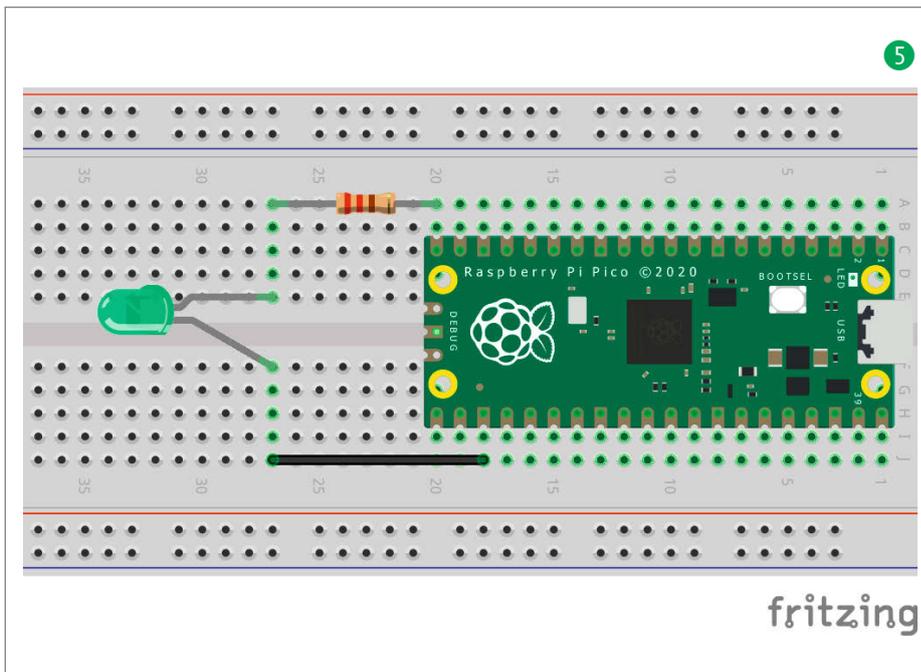
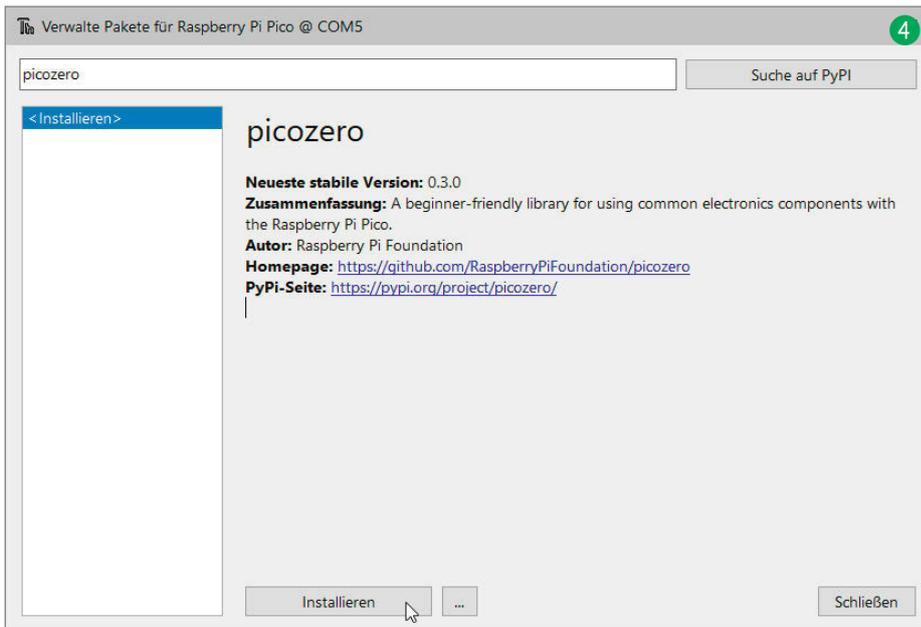
```
from picozero import LED
led = LED(15)
led.blink()
```

Wenn Sie diese Befehle in die Kommandozeile eingeben, sollte die LED wiederum blinken. Beachten Sie, dass wir nun das allgemeinere LED-Objekt importieren und benutzen. Der Parameter in der zweiten Zeile ist 15, dies bezeichnet den GPIO des Picos, nicht die Nummerierung der physikalischen Pins auf dem Board. Ein weiteres Dokument für unters Kopfkissen ist die Pinbelegung des Pico, vor allem weil die Belegung auf dem Board leider unten angebracht ist.

Soweit so gut, allerdings hat das LED-Objekt gegenüber `pico_led` noch ein paar Tricks mehr zu bieten. Geben Sie einmal Folgendes ein:

```
led.brightness = 0.4
```

Die LED hört auf zu blinken (leider, sonst könnte man so einfach eine LED weniger hell

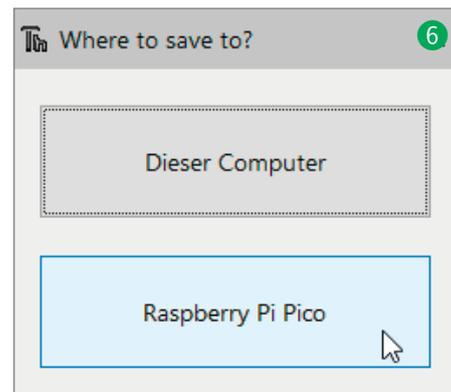


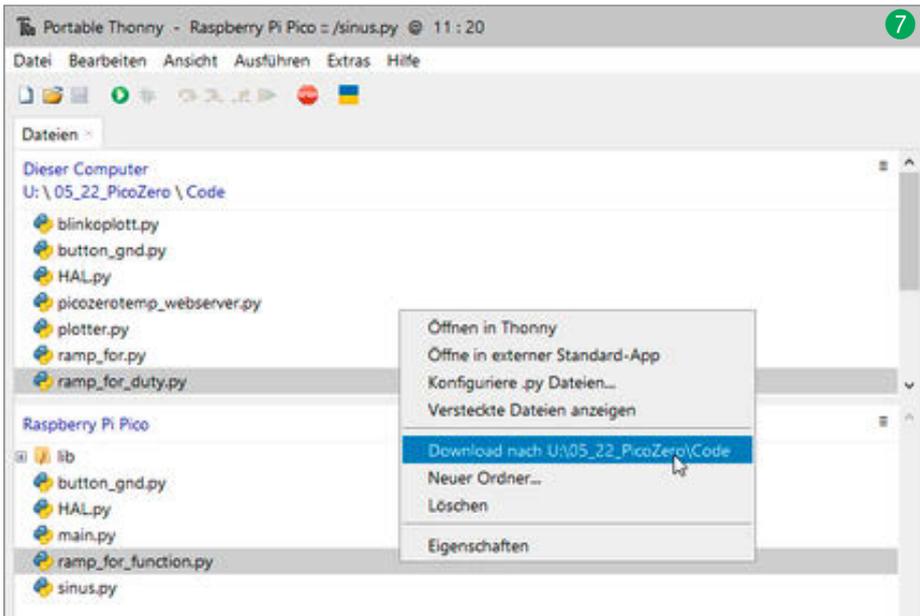
```
LED_puls.py

from picozero import LED
from time import sleep

led = LED(15)

while 1:
    for i in range(0,100):
        led.brightness = i/100.0
        sleep(0.01)
    for i in range(100,0,-1):
        led.brightness = i/100.0
        sleep(0.01)
```





blinken lassen) und leuchtet nun viel schwächer. Die Helligkeit wird digital über sogenannte *Pulsweitenmodulation* (PWM) gesteuert. Dabei schaltet man die LED schnell (100Hz standardmäßig) an und aus: Der Anteil der Anschaltzeit (*Duty Cycle*) bestimmt dann die Helligkeit. Das Auge kann dieser hohen Frequenz nicht folgen und die LED erscheint entsprechend heller oder dunkler, ohne dass ein Flackern wahrnehmbar wäre.

### Skripte und Autostart

Bisher haben wir nur die Kommandozeile benutzt. Das funktioniert natürlich nicht, wenn

der Pico allein ohne Computer arbeiten und etwas steuern soll. Für das folgende Beispiel benötigen Sie wiederum die Schaltung aus dem vorherigen Abschnitt.

Geben Sie jetzt das Programm aus dem Listing *LED\_puls.py* in das Editorfenster von Thonny ein. Nach dem Start sollte dieses Fenster noch als *unbenannt* betitelt sein.

Mit dem grünen Startknopf in der Werkzeugleiste oder der Funktionstaste *F5* können Sie das Skript direkt auf dem angeschlossenen Pico ausführen. Die LED sollte nun weich ein- und ausgeblendet werden. Ändern Sie doch einmal die Geschwindigkeit des Abblendens.

Gestoppt wird das Programm mittels des *STOP*-Knopfes in der Werkzeugleiste oder Drücken von *Strg-C* in der Kommandozeile. Wenn Sie etwas am Programm ändern und erneut das Programm ausführen, stoppt Thonny automatisch. Möchten Sie das Skript speichern, muss allerdings erst manuell gestoppt werden.

Speichern geschieht mit den üblichen Menüs und Icons. Eine Besonderheit ist aber, dass man durch eine kleine Abfrage **6** gefragt wird, wohin man speichern möchte: Auf die (Netz-)Laufwerke des Computers oder direkt auf den Pico. Auch beim Speichern auf den Pico sollte man einen sinnvollen Namen angeben. Dieser muss aber für Python-Skripte um die *.py*-Endung ergänzt werden. Zur Datensicherung sollte man immer eine Kopie des Skriptes auf den Computer speichern. Zu leicht überschreibt man aus Versehen mal einen Pico mit einer anderen Firmware. Die Sicherung der Skripte auf einem Pico kann man bequem in den Dateifestern (Menü *Ansicht/Dateien*) erledigen, hier selektiert man einzelne oder mehrere Skripte auf dem Pico **7** und kann sie dann über das Kontextmenü auf den Computer exportieren.

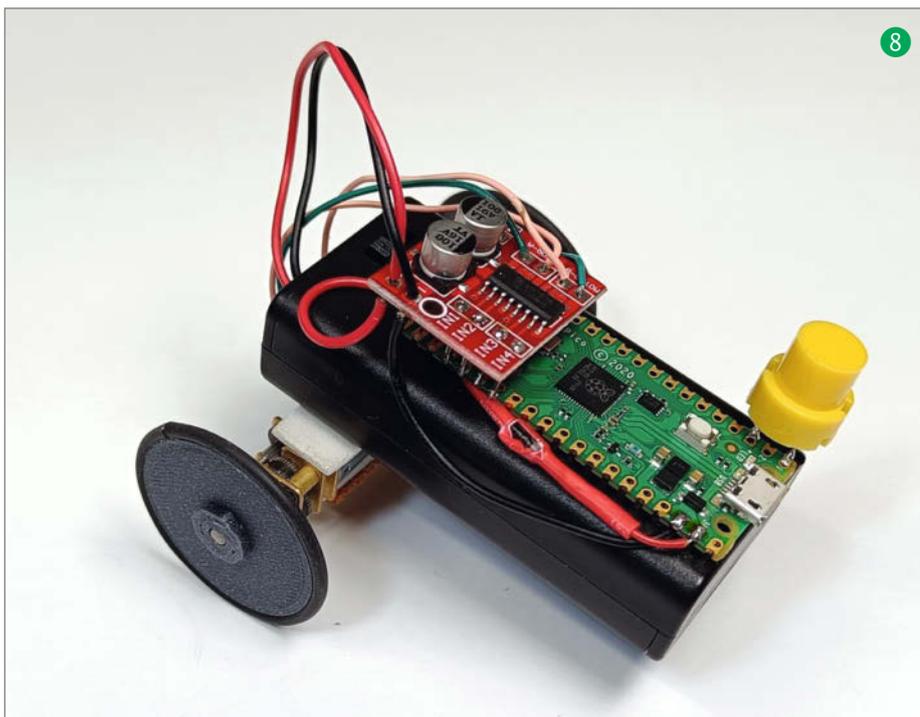
Wie wird nun ein Skript beim Anschalten des Raspberry Pi Pico ausgeführt? Die Lösung ist ganz einfach, speichern Sie das Skript als *main.py*: Fortan wird es ausgeführt, sobald der Pico mit Strom versorgt wird. Schließen Sie so einen Pico an den PC an und wollen das Programm in Thonny ändern, müssen Sie es zuerst mit dem *STOP*-Knopf unterbrechen.

### Rover mit Picohirn

Picozero in Version 0.3.0 unterstützt im Moment folgende Hardware: LEDs, RGB-LEDs, Summer, Lautsprecher, Motortreiber, generell digitale und PWM-Ausgabe, digitale Eingabe (Taster, Schalter), analoge Eingabe (Potis, Thermosensoren, Lichtsensoren etc.) und Abstandssensoren.

Also habe ich meine Bastelkiste ausgeschüttet und zwei Getriebemotoren, einen passenden Zweifach-Motortreiber und einen Taster herausgesucht. Eine Akkubox übernimmt die Stromversorgung, dabei sorgt eine Schottky-Diode (wie beim *Stromausfall-Monitor*, siehe *Mehr zum Thema*) dafür, dass der Pico per Akku oder über USB betrieben werden kann, die Akkus nicht versehentlich geladen werden und man ohne das Programm zu unterbrechen zwischen Batterie und Kabel wechseln kann: Sehr praktisch für die Programmierung eines Gefährts, das autonom fahren soll.

Zusammengehalten wird der Rover **8** mit etwas Doppelklebeband, die Räder sind mit Silikon-schlauch beklebte 3D-Drucke, Bastelzeit etwa eine Stunde. Das Programm ist sehr primitiv, ohne Sensoren ist das Gefährt auch



noch nicht wirklich spannend. Ein Stift zwischen den Antriebsrädern macht aber schnell eine *Logo-Turtle* aus dem Gefährt. Eine tolle Sache besonders für Kinder, um Mathematik, Geometrie und Programmierung spielerisch zu erfahren.

Im Listing *Rover.py* können die GPIOs abgelesen werden, einen Schaltplan gibt es aber natürlich in unserem GitHub. Eine sehr angenehme Sache ist, dass man die Abfrage von digitalen Eingaben (hier der Taster) Picozero im Hintergrund überlassen kann. Dem Knopf wird in Zeile 10 die Funktion `button_toggle` (ja ohne Klammern!) zugewiesen. Bei Knopfdruck wird die Funktion (Zeilen 7-8) aufgerufen, egal was das eigentliche Programm gerade macht. In der `while`-Schleife wird dann auch der Zustand (`value`) der LED benutzt, um zu entscheiden, ob sich der Rover bewegen soll oder nicht. So spart man eine Variable. Auch die Verkabelung, wenn man dies noch so nennen darf, des Tasters ist denkbar einfach: Er verbindet GND mit dem GPIO.

Also plündern Sie Ihre Teilekiste und legen Sie los mit dem Pico und Python: Mir fielen noch zwei lichtempfindliche Widerstände (LDR) in die Hände, vielleicht baue

## Rover.py

```
01 from picozero import Robot, Button, pico_led
02 from time import sleep
03
04 rover = Robot(left=(18,19), right=(20,21))
05 button = Button(0)
06
07 def button_toggle():
08     pico_led.toggle()
09
10 button.when_pressed = button_toggle
11
12 while 1:
13     if pico_led.value:
14         rover.forward(t=3, wait=True)
15     if pico_led.value:
16         rover.left(t=0.65, wait=True)
17         sleep(1)
18     else:
19         sleep(.5)
```

ich den Rover in eine fahrende Motte um. Aber auch Taster als Bumper oder Abstandssensoren sind möglich. Und auf dem Pico W kann der Rover sogar per WLAN gesteuert werden. Wenn dann die Ambitionen größer werden und die Programme komplizierter,

dann schauen Sie sich einmal den Quellcode von Picozero an. Diese Bibliothek ist nämlich auch komplett in Python geschrieben und für den fortgeschrittenen Maker eine gute Quelle, um die Feinheiten von Python zu lernen. —caw

## Make:markt

### MIKROELEKTRONIK



#### AZ-Delivery

Ihr Experte für Mikroelektronik  
**20% sparen – Code: AZD-Make20**  
 Die passenden MCU, Sensoren, Displays,  
 etc. für Ihr nächstes Projekt!  
 Projektideen im Blog  
 Spezialangebote im Newsletter

[www.az-delivery.de](http://www.az-delivery.de)

### BÜCHER/ZEITSCHRIFTEN



#### Der Verlag für kreative Köpfe!

Informatik und Elektronik können komplex,  
 theoretisch und anstrengend sein. Es geht  
 aber auch einfach, anschaulich und leicht  
 nachvollziehbar – wenn man die Dinge in die  
 eigenen Hände nimmt und zum »Maker« wird:  
 Mit Büchern vom dpunkt.verlag.

[www.dpunkt.de](http://www.dpunkt.de)

### METALLBAU



#### MakerBeam: Mini T-Nut Alu-Profil

Unbegrenzte Möglichkeiten in Modell- und  
 Prototypenbau  
 Das MakerBeam Sortiment:  
 - 10mm & 15mm Profile  
 - Linearlager, Scharniere, Eckwürfel  
 - Halterungen für Servos & NEMA17  
 - M3 Schrauben, Nutensteine, Abstandshalter

[www.makerbeam.de](http://www.makerbeam.de) | [www.chartup.com](http://www.chartup.com)



#### Was Maker schon alles geschaffen haben!

Die Antwort und viele Beispiele finden Leser  
 in unseren Zeitschriften  
 „Space – das Weltraum Magazin“,  
 „Wissen 2022“ und dem „Urknall“ vieler  
 Computer- und Make-Enthusiasten – dem  
 „Retro Gamer“.

[www.emedia.de](http://www.emedia.de)

# „I'll be back!“

Ob mit Plüsch oder ohne: Die animatronische Posteule aus der Make 1/22 und 2/22 läuft jetzt auch mit einem ESP32 und dem YX5300-MP3-Player. Kein Problem also, wenn ihr keinen Raspberry Pi zur Hand habt.

von Ákos Fodor



Nachdem wir den ersten Posteulen-Artikel veröffentlicht hatten und aufgrund der Halbleiterknappheit der Raspberry Pi aus vielen Online-Shops verschwand oder die Preise hochschossen, erreichten mich Leseranfragen, ob die Eule **1** auch mit alternativer Hardware wie dem Raspberry Pi Zero betrieben werden könne. Technisch konnte ich mir das durchaus vorstellen, aber auch diese Mikrocontroller fielen kurze Zeit später der akuten Chipknappheit zum Opfer. Wieso also nicht auf ESP32 umsatteln? Schließlich ist er günstig, nach wie vor verfügbar und verbraucht mit durchschnittlichen 80mA weniger Strom als die ursprüngliche Raspberry-Variante. Wie sich der Aufbau und die Programmierung durch den Boardwechsel geändert haben, zeige ich in dieser ergänzenden Anleitung. An einigen Stellen verweise ich, wenn bereits erklärt, auf die ursprünglichen Artikel. Ihr findet alle notwendigen Links zu Artikeln, Programmcodes, Treibern und 3D-Daten in der Kurzinfor.

## Geänderter Aufbau

Konstruktiv unterscheidet sich die ESP-Eule kaum von ihrer Vorgängerversion. Der Raspberry Pi hat den Rumpf verlassen und Platz für das vielfach kleinere *ESP32 Core Board* (KS0413) von *keystudio* gemacht. Hinzugekommen ist außerdem ein YX5300-MP3-Player, der seriell angesteuert wird und Sounds von einer microSD-Karte liest. Falls ihr die Eule ohne Plüsch zusammenbauen möchtet, könnt ihr jetzt auch das coole Cyborg-Gehäuse (siehe Titelbild) verwenden, das auf der Maker Faire ausgestellt war.

Wie auch beim ersten Entwurf habe ich zunächst die Platinen ausgemessen und in CAD nachgebaut, damit ich sie präzise in der bestehenden Eulenkonstruktion platzieren kann. So entstand die Halterung **H1E**, die genau dieselben Lochmaße verwendet wie der Raspberry Pi und nahtlos zwischen die Front- und Rückenteile geschraubt werden kann. Das YX5300-Board wird an **H1E** mit M2x4mm-Schrauben befestigt. Der ESP32 kann mit seinen Pins in die Halterung gedrückt werden und wird später mithilfe des modifizierten Teils **R1E** von der Rückseite aus festgehalten. Ich musste größtenteils auf Dupont-Stecker für die Kabelverbindungen verzichten und habe den LötKolben in die Hand genommen. Denn der ESP32 mag zwar schmal sein, aber mit dem Pin-Header ist er stellenweise ungefähr so dick wie der Raspberry Pi und die Kabel samt Stecker wären sonst mit den anderen Bauteilen kollidiert. Solltet ihr ein anderes ESP32-Board verwenden wollen, müsst ihr die Halterung im Zweifelsfall selbst anpassen. Dafür stelle ich euch die Halterung in CAD-Austauschformaten im GitHub-Repository des Projekts bereit.

## Kurzinfor

- » Animatronisches Projekt mit ESP32
- » Sounds über den YX5300-MP3-Player abspielen
- » Servos mit ESP32 und dem PCA9685 Servocontroller steuern

### Checkliste



**Zeitaufwand:**  
6 Stunden zzgl. 3D-Druck



**Kosten:**  
10 bis 150 Euro

### Werkzeug

- » 3D-Drucker
- » LötKolben
- » Seitenschneider
- » Skalpell oder Cutter

### Material

- » ESP32 KS0413 von *keystudio*
- » YX5300-MP3-Player
- » Alle Materialien aus den vorherigen Anleitungen, bis auf den Raspberry Pi 3B

### Mehr zum Thema

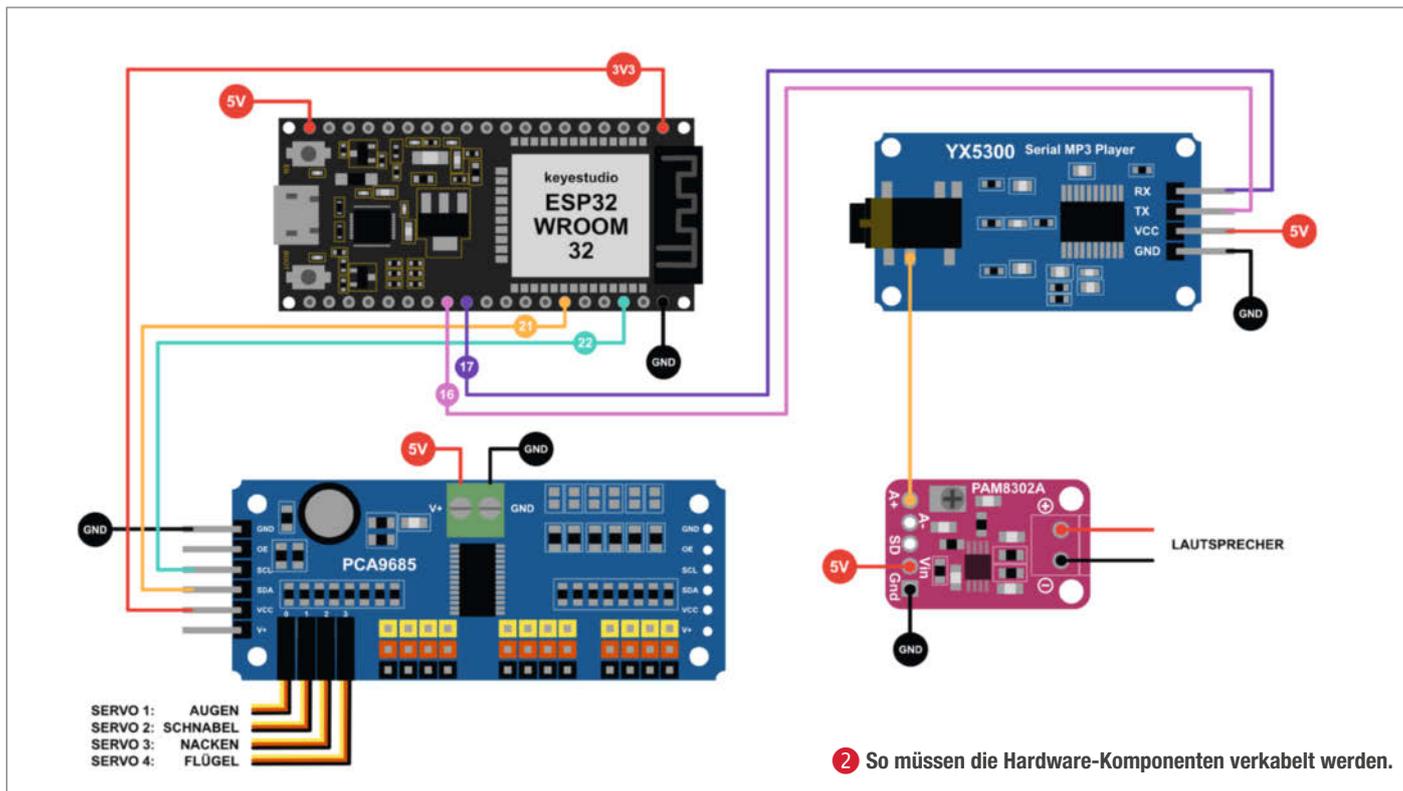
- » Ákos Fodor, Die animatronische Posteule: Teil 1, Make 1/22, S. 8
- » Ákos Fodor, Die animatronische Posteule: Teil 2, Make 2/22, S. 76

Alles zum Artikel  
im Web unter  
[make-magazin.de/x6du](https://make-magazin.de/x6du)



**1** Die animatronische Posteule mit Plüsch aus der Make 1/22 und 2/22





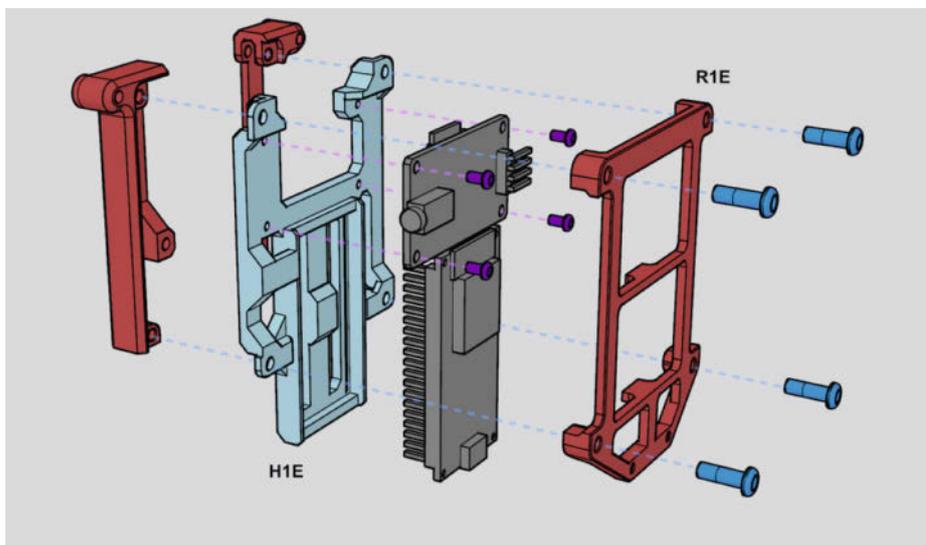
Beginnt am besten damit, die notwendigen Kabel an den ESP32 zu löten 2. Einerseits sind die Pins nach dem Zusammenschrauben kaum erreichbar und andererseits überträgt sich sonst die Hitze beim Löten auf die gedruckte Halterung. Achtet darauf, vor allem die etwas dickeren Stromkabel länger zu lassen. Diese können wir ganz zum Schluss kürzen. Steckt und verschraubt anschließend den ESP32 und den YX5300 mit der Halterung H1E 3. Kürzt danach auf jeden Fall die Pins

auf der Unterseite des PCA9685-Servo-Controllers 4, da sie dem ESP32 sonst zu nahe kommen werden. Folgt anschließend dem ersten Aufbau-Artikel (Link in der Kurzinfo), bis nur noch die beweglichen Elemente mit den Servos verbunden werden müssen – die Servo-Wellen sollten also frei liegen. Nachdem ihr alles verlötet, verbunden und verschraubt habt 5, sollten nur noch vier 5V- und fünf GND-Kabel übrigbleiben, da der Chip des Servo-Controllers 3,3V über den ESP32 erhält.

Achtet unbedingt darauf, die grüne Buchse des PCA9685 mit Strom zu versorgen, weil sich die Servos sonst nicht bewegen werden 6.

### Arduino-IDE vorbereiten

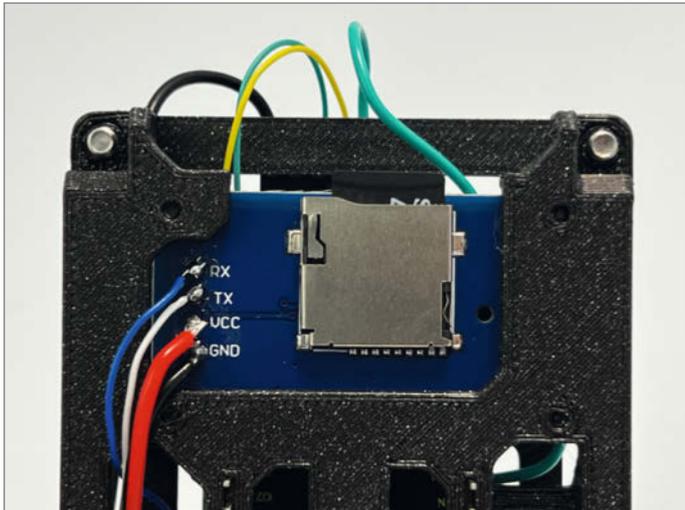
Den ESP32 habe ich mithilfe der Arduino IDE programmiert. Damit der Computer mit dem Board kommunizieren kann, müssen wir den Mikrocontroller zunächst über die Boardverwaltung zur Arduino IDE hinzufügen. Unsere



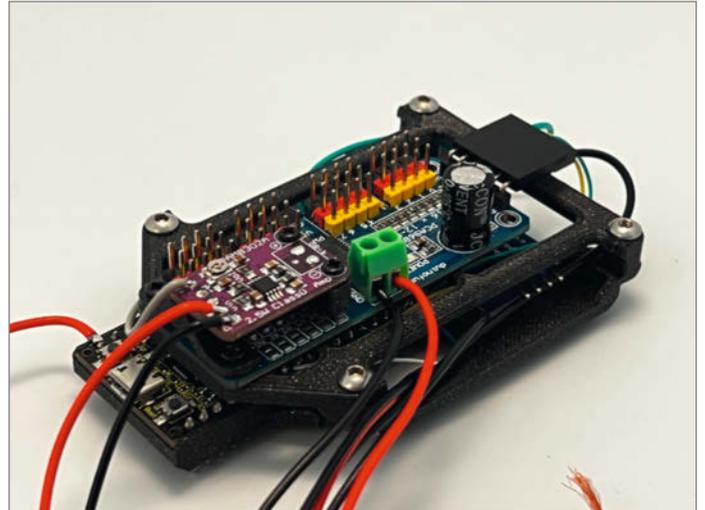
3 Die neue Halterung H1E: Schraubt den YX5300 mit 4 M2 x 4mm-Schrauben fest und steckt den ESP32 nach dem Löten mit den Pins in die Aussparung. Das Bauteil R1E hält ihn später in Position.



4 Mit einem Seitenschneider könnt ihr leicht die herausragenden Lötspitzen kürzen.



5 Dank der Aussparung lassen sich die Kabel auch nach dem Festschrauben des YX5300 auf der Rückseite anlöten.



6 Die grüne Buchse am Servo-Controller muss mit Strom versorgt werden. Ansonsten bewegen sich die Servos nicht.

Anleitung und ergänzende Treiber findet ihr als Link in der Kurzinfo. Als Nächstes benötigen wir noch zwei Bibliotheken, um den Servo-Controller ansteuern zu können und E-Mails abzurufen. Geht über den Menüpunkt *Werkzeuge* in die *Bibliotheksverwaltung* und installiert folgende Libraries:

- Adafruit PWM Servo Driver Library
- ESP Mail Client

Startet anschließend die Arduino IDE neu. Danach solltet ihr im Menü *Werkzeuge* unter dem Menüpunkt *Boards* den Eintrag *ESP32 Dev Module* finden. Wenn ihr später Sketches auf dieses ESP32-Board überträgt, müsst ihr die Boot-Taste (rechts vom Micro-USB-Anschluss)

gedrückt halten, sobald die Konsole der Arduino-IDE in Rot *Connecting...* anzeigt. Achtet also darauf, dass die Taste während der Konfiguration erreichbar ist. Ich habe die Rückschale **G2** angepasst, sodass ihr die Boot-Taste durch ein kleines Loch drücken könnt, selbst wenn die Eule komplett zusammengebaut ist.

### Der Sound

Mit dem YX5300-MP3-Player können wir das Abspielen von MP3- oder WAV-Sounddateien auslagern. Das kleine Modul unterstützt Samplingraten von 8–48kHz und liest diese von einer microSD-Karte, die auf FAT oder FAT32 formatiert sein muss. Die serielle Ansteuerung

```
esp2mp3(sndPlayFolderFile, 1, 2);
```

↓            ↓ ↓

7E FF 06 0F 00 01 02 EF

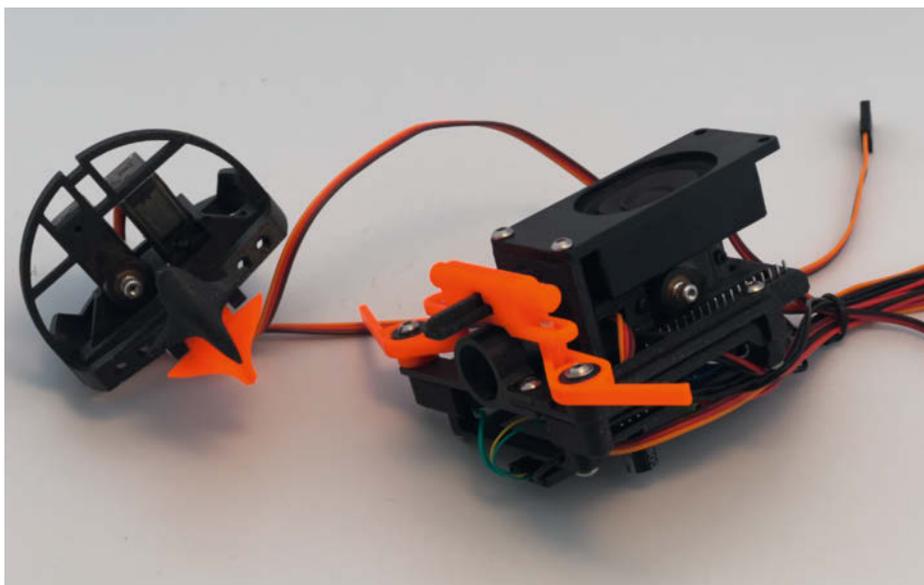
7 **Serielle Kommunikation:** Die Arduino-IDE wandelt den Befehl in ein Format um, das der YX5300 versteht.

des YX5300-MP3-Players ist ziemlich einfach, auch wenn sie zunächst kryptisch wirkt. So spielen wir beispielsweise den zweiten Sound in dem Ordner 01 mit der Hex-Befehlsfolge **7E FF 06 0F 00 01 02 EF** ab. Das Zeichenpaar **0F** steht in diesem Fall für die Abspielfunktion,



# WAZER

Erster DESKTOP-WASSERSTRAHLSCHNEIDER der Welt.  
ab 9.990,- € zzgl. MwSt.



8 Die Servos sollten zunächst nicht verbunden werden, wenn wir sie auf 90 Grad einstellen.

01 gibt den Ordner an und 02 die zweite Datei in diesem Ordner. Alle anderen Zeichenpaare sind zwar für die Kommunikation relevant, bleiben jedoch über alle Funktionen hinweg gleich – also müssen wir für die Steuerung des YX5300 lediglich drei Parameter ändern.

Wenn ihr den Sketch *audioTest.ino* öffnet, seht ihr zunächst, dass die benötigten Befehle als Konstanten definiert sind. Das erleichtert die Programmierung, da ihr euch nicht den Hex-Code der Befehle merken müsst. Außerdem wird ein Array erstellt, das mit den 8 Werten befüllt wird, die wir später an den YX5300 senden.

Die Funktion `esp2mp3()` regelt, welche drei Parameter in das Array eingesetzt werden 7. Auch wenn der YX5300 noch mehr Steuerungsmöglichkeiten anbietet (siehe Link zum Handbuch in der Kurzinfor), verwenden wir für die Eule den Befehl *Play with folder and file name* (hier definiert als `sndPlayFolderFile`) und

*Set Volume* zu Beginn bzw. bei Bedarf, falls einzelne Sounds zu laut oder zu leise sind. Damit der YX5300 die Sounds findet, müsst ihr auf der microSD-Karte einen Ordner mit dem Namen 01 erstellen und die Sounddateien z.B. *001\_sound.mp3*, *002\_sound.mp3* usw. benennen. Ladet ein paar Sounds nach diesem Schema auf die Speicherkarte, passt die Befehle im `void setup()` des Sketches an und ladet ihn anschließend auf den ESP. Der Sound wird nun über den YX5300 an das PAM8302-Verstärkermodul übertragen und auf dem Lautsprecher ausgegeben. Mithilfe des Poti am PAM könnt ihr mit einem Schraubendreher die Gesamtlautstärke vorsichtig einstellen.

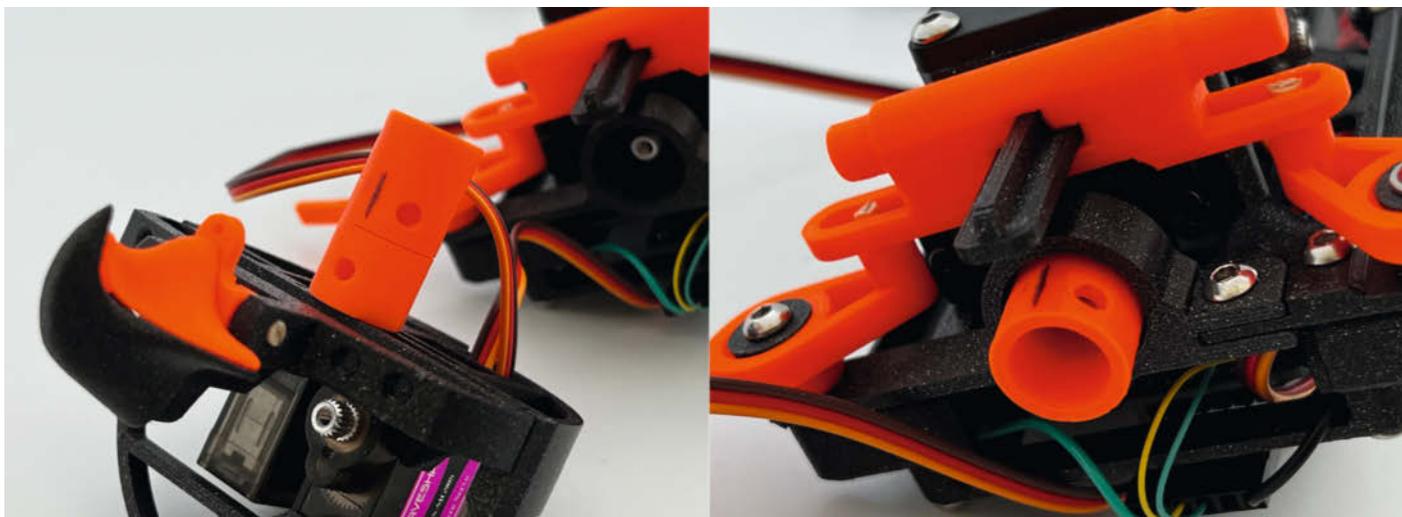
### Die Eule vorbereiten

Da wir die Sketches per Kabel an den ESP32 übertragen, findet der Kalibriervorgang auch

etwas abgewandelt statt. Verbindet zunächst alle Servos mit dem Servo-Controller, wie im Schaltplan angegeben. Achtet auf die Reihenfolge, da die Definition der zugehörigen Servos in allen noch folgenden Sketches darauf abgestimmt ist. Öffnet die Datei *servosTo90.ino* in der Arduino-IDE (siehe GitHub-Repository des Projekts) und übertragt ihn auf den ESP32. Dieser Sketch stellt alle Servos auf 90 Grad ein, damit wir sie anschließend in beide Richtungen justieren können 8.

Trennt die USB-Verbindung und montiert die beweglichen Teile der Eule an den Servos, wie in der zweiten Anleitung aus der Make 2/22 (siehe Kurzinfor) beschrieben. Ebenso findet ihr auf unserem YouTube-Kanal ergänzende Videos, die euch dabei unterstützen. Die Nackenteile **N1** und **N2** habe ich für die ESP-Eule überarbeitet. Beide lassen sich komplett ohne Stützkonstruktion drucken und **N1** hat ein zusätzliches Loch erhalten, das vorher manuell gebohrt werden musste. Um **N2** mittig auf dem zugehörigen Servo zu befestigen, empfehle ich, ihn vorher auf **N1** zu stecken und mit einem Stift eine Markierung wie auf dem Foto 9 zu machen, wenn der Kopf gerade ausgerichtet ist.

Es kann passieren, dass die Servos sich beim Festschrauben leicht verdrehen, z.B. wenn ihr den Schraubendreher zu schwach auf die Welle drückt. In diesem Fall könnt ihr die Eule einfach mit dem Netzteil verbinden und der Sketch *servosTo90.ino* stellt noch einmal alle Servos auf 90 Grad ein. Vermeidet nach der Montage, den Kopf manuell zu drehen, weil ihr sonst Gefahr lauft, ihn direkt wieder abzuschrauben. Die bei Servos standardmäßig mitgelieferten Aufsätze haben im Vergleich zu den gedruckten Eulen-Teilen Zähne, die exakt auf der Servo-Welle sitzen und das Durchdrehen der Welle verhindern. Da jedoch jeder Hersteller auch hier minimal abweichende Maße verwendet, habe ich in



9 Markiert die Mitte von N2, damit es beim Verschrauben richtig ausgerichtet werden kann.

meinem Entwurf keine Zähne verwendet, damit die Teile zu möglichst vielen MG90s-Servos passen. Wer mehr Grip benötigt und mit den Servo-Verbindern experimentieren will, findet die CAD-Modelle im GitHub-Repository und kann sie gern nach Belieben erweitern und anpassen.

## Servos kalibrieren

Nachdem die Eule zusammengesraubt ist und im schützenden Gehäuse sitzt, können wir beginnen, die Servos nacheinander zu kalibrieren. Verwendet dafür den Sketch *servoCalib.ino*. Zu Beginn habe ich die Min- und Max-Werte definiert und auf 90 Grad gestellt. Im `void start()` findet ihr die Befehle, die ihr jetzt benötigt. Mit `pwm.setPWM(servoEyes, 0, servoAngle(eyesMin))`; wird als Erstes der Augen-Servo auf die Position `eyesMin` bewegt. Ändert den Wert in der Definition auf `#define eyesMin = 85`; und ladet den Sketch per Kabel auf den ESP32. Die Augen sollten sich leicht öffnen. Ändert den Wert erneut auf 75, 65 usw. Bei 30 solltet ihr langsam eine Untergrenze erreicht haben, bei der die Servo-Arme hinter den Augen mit **K1** kollidieren. Lasst den Servos immer ein wenig Spiel. Sobald `eyesMin` kalibriert ist, ersetzt im `void start()` die Konstante `eyesMin` durch `eyesMax` und wiederholt den Vorgang, nur in die andere Richtung. Vermutlich werden die Augen bei 90 Grad bereits geschlossen sein. Falls nicht, stellt `eyesMax` auf 95 usw. Wiederholt diesen Vorgang für alle anderen definierten Werte. Habt Geduld beim Kalibrieren und macht nicht zu große Sprünge mit den Werten. Wenn ihr den Servo verkeilt, läuft er heiß, weil er kontinuierlich versucht den angestrebten Winkel anzusteuern. Für die Kopfbewegung findet ihr nur die Konstante `headMid`, denn die äußeren Winkel sind 0 und 180. Im finalen Sketch *bubo.ino* gibt es für die äußeren Winkel ebenso die Konstanten `headMin` und `headMax`. Es kann aber, wie oben beschrieben, passieren, dass der Servo bei der Montage durchgedreht ist und der Kopf bei 90 Grad nicht exakt nach vorn schaut. Meine Eule hat bei `headMid` z.B. den Wert 80. Notiert euch die ermittelten Werte und fahrt mit dem nächsten Abschnitt fort.

## Bewegungen testen und erstellen

Mit dem Sketch *servoTest.ino* könnt ihr den Bewegungsapparat eurer Eule testen. Übertragt die Min- und Max-Werte der Servos in den Sketch und ladet ihn auf den ESP32. Das Programm bewegt anschließend alle Motoren nacheinander an ihre äußeren Positionen. Wer Lust hat, kann den Sketch mit ein paar eigenen Animationen erweitern und sie später in den finalen Sketch übertragen.

Es gibt im Wesentlichen zwei Varianten, die Servos zu bewegen: Entweder gebt ihr einen

## Bewegungen mit einstellbarer Geschwindigkeit (Beispiel)

```
wingsMin = 35; //gehobener Flügel
wingsPos = 90; //aktuelle Flügelposition

void wingsUp(int aniSpeed){
  if (wingsPos>=wingsMin){

    for (int i=wingsPos; i>=wingsMin; i--){
      pwm.setPWM(servoWings, 0, servoAngle(i));
      delay(aniSpeed);
    }
    wingsPos = wingsMin;
  }
}

void start(){
  wingsUp(50);
}
```

direkten Befehl wie beim Kalibrieren oder ihr steuert die Geschwindigkeit der Servos mithilfe einer `for`-Schleife. Die Funktion `delay(aniSpeed)`; fügt zwischen den einzelnen Winkelschritten Pausen ein. Wenn im `void start()` oder `void loop()` beispielsweise der Befehl `openEyes(50)`; ausgeführt wird, übergibt diese Zeile die 50 als `aniSpeed` an die Funktion `openEyes(int aniSpeed)` und spielt die Animation mit 50 Millisekunden-Pausen ab. So kann jedesmal direkt beim Aufrufen der Animation bestimmt werden, wie schnell die Bewegung ablaufen soll. Im Sketch habe ich eine Reihe von Grundbewegungen vordefiniert, wie Blinzeln, Flügelschlag oder den Kopf nach links und rechts drehen. Vielleicht wollt ihr aber auch mit zufälligen Bewegungen spielen, wie in `headRand()` oder aus vielen Einzelbewegungen ganze Animationsabläufe choreografieren. Dazu könnt ihr sie in weitere Funktionen verschachteln. Für den finalen Sketch ist es nur wichtig, dass ihr alle Funktionen mitnehmt, die ihr zusätzlich erstellt habt.

## Der finale Sketch

Jetzt, da sich die Eule bewegt, benötigt sie noch einen Internetzugang und eine eigene E-Mail-Adresse. Ich verwende dafür G-Mail mit einem App-Passwort, wie es in der zweiten Anleitung in der Make 2/22 beschrieben ist. Dennoch möchte ich hier noch einmal darauf hinweisen, dass die Eule nach dem Abrufen eines Posteingangs sämtliche E-Mails darin löscht. Verwendet daher unbedingt eine separate E-Mail-Adresse und leitet E-Mails an eure Eule weiter.

Wenn ihr den Sketch *bubo.ino* öffnet, befinden sich im Header des Programms die Einstellungen und Konstanten. Dort findet ihr die Eingabefelder für euer WLAN sowie die zu verwendende E-Mail-Adresse und alle

Komponenten, die wir bisher besprochen haben. Mit der Konstante `checkFreq` könnt ihr einstellen, wie oft die Eule ihre Nachrichten abrufen soll. Der Wert muss auch hier mit Millisekunden angegeben werden: also z.B. 60000 für eine Minute. Ich empfehle euch außerdem, den Spam-Filter eures E-Mail-Providers auszuschalten oder zumindest die Absender-Adressen, von denen die Eule E-Mails erhalten wird, als *Nicht-Spam* zu markieren. Nachrichten, die im Spam-Ordner landen, werden von der Funktion `checkIncoming()` nicht berücksichtigt.

Übertragt nun eure Min- und Max-Werte für die Servos und fügt eure Animationen ein, falls ihr welche erstellt habt. Achtet darauf, dass sich keine Funktionen doppeln und ersetzt im Zweifelsfall bestehende durch die neuen. Passt danach in der Funktion `checkIncoming()` an, welche Animation abgespielt werden soll. Ihr könnt mit `randAnimation()` auch per Zufall jedes Mal eine andere Animation abspielen lassen. Dazu müsst ihr innerhalb der Funktion vorher die Platzhalter mit den Namen eurer Animationen ersetzen. Das ist eine gute Möglichkeit, sie etwas lebendiger und weniger roboterhaft wirken zu lassen. Außerdem kann die ESP-Variante der Eule diesmal nicht zwischen verschiedenen E-Mail-Adressen unterscheiden. Dadurch wird selbst bei unterschiedlichen E-Mails im Posteingang nur eine Animation abgespielt und nicht mehrere hintereinander. Aber es gibt wie immer auch hier noch Anpassungspotenzial. Interessant fände ich noch, wenn die Eule auf Befehle reagieren könnte, die als Begriffe im Betreff stehen.

Wenn ihr mit den Anpassungen fertig seid, sollten alle Funktionen so verschachtelt sein, dass im `void loop()` nur noch `checkIncoming()` steht. Übertragt als Letztes *bubo.ino* auf die ESP-Eule und freut euch auf die eintreffenden E-Mails. —akf

# Mit dem Game Boy ins Internet

Ursprünglich war er nur zum Spielen gedacht. Doch dank eines ESP8266 kann er jetzt auch über WLAN kommunizieren. Mit seiner WiFi-Cartridge zeigt Sebastian Stacks, wie man den Game Boy mit dem Internet verbindet.

von Ákos Fodor



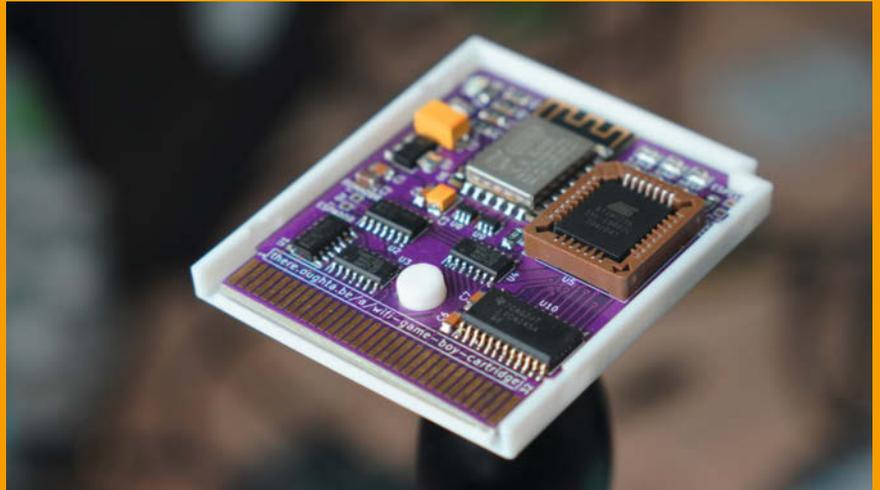
Sebastian Stacks

Der *Nintendo Game Boy* aus dem Jahr 1989 ist in etwa so alt wie das kommerzielle *World Wide Web*. Als tragbare Spielekonsole ausgelegt, hatte damals jedoch niemand geplant, den kultigen Handheld mit dem Internet zu verbinden. Einzig das Link-Kabel ermöglicht bis heute lokale Multiplayer-Partien in *Tetris* und *Street Fighter II* oder den Tausch gesammelter *Pokémon*. Dass der Game Boy zu weit mehr imstande ist, demonstriert Sebastian Staacks auf seiner Projekt-Website und dem Youtube-Kanal *there oughta be*: In seinem bekanntesten Video präsentiert er das Computerspiel *GTA V* aus dem Jahr 2013 – auf einem Game Boy, monochrom, mit 160×144 Pixeln. Wieso er so etwas tue, beantwortet Staacks scherzhaft mit: „Weil ich es kann.“ Denn die Cartridge, die ihm diesen Trick ermöglicht, hat er selbst entwickelt. Mit ihrer Hilfe kann der Game Boy über WLAN kommunizieren und Inhalte aus dem lokalen Netzwerk oder dem Internet laden und abbilden, z.B. von Wikipedia, der Make-Website oder als Videostream – mit einem Game Boy Color sogar in Farbe.

Als Recheneinheit verwendet Staacks für seine Cartridge einen *ESP8266* und einen *32kB-EEPROM*-Speicher. Die „elegante Lösung“, wie er sie bezeichnet, darf jedoch nicht mit einer einfachen verwechselt werden. Spätestens, wenn man den detaillierten Ausführungen zur Speicherverwaltung, Verschaltungen von Logik-Komponenten oder Kommunikationsmustern folgt, wird deutlich, wie viel Arbeit in die Recherche und Entwicklung geflossen sind. Wer sich für diese Art von Retro-Technik interessiert, erfährt in seinen englischsprachigen Texten und Videos, wie der Game Boy Grafiken als Sprites verwaltet und darstellt, was man beachten muss, wenn man Videostreams mit 20FPS Bildwiederholrate abspielen möchte und vieles mehr.

Mit Kenntnissen der Elektronik ist es vermutlich leichter, den Entstehungsprozess der Cartridge nachzuvollziehen. Aber Staacks bemüht sich durchweg, anschaulich und möglichst niedrigschwellig zu erklären. Einsteigern, die daran interessiert sind, die Funktionsweise des Game Boy zu verstehen oder planen, für diesen zu programmieren, empfiehlt er zusätzlich den *Ultimate Game Boy Talk* von Michael Steil sowie die Referenzquelle *Pan Docs*. Dort gibt es viele Informationen zu der grundlegenden Technik des Gerätes. Ansonsten lohnt es sich, einen Blick in das knapp 3,5h lange Q&A-Video zur WiFi-Cartridge zu werfen, in dem eine Menge Fragen beantwortet werden. Alle, die Sebastian Staacks Modul nachbauen wollen, können sich sämtliche Projekt-Daten als Open Source auf GitHub herunterladen. —akf

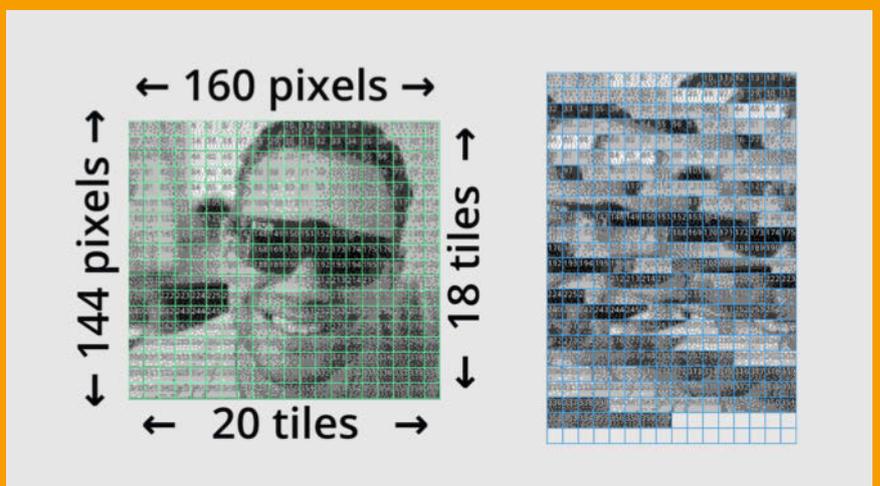
► <https://there.oughta.be>



Die fertige Cartridge von innen: Der ESP8266 und ein 32kB-EEPROM-Speicher bilden die zwei wichtigsten Komponenten.



Programmiert wurde die Cartridge mit der Arduino IDE und zum Teil mit Assembler.



Grafiken werden auf dem Game Boy in Sprites zerteilt, die aus 8×8 oder 8×16 Pixeln bestehen. Später werden sie wieder zusammengesetzt.

# RC-Schlepper im Benchy-Design

Der Sommer ist Badezeit, Schlauchboot- und SUP-Board-Zeit. Der Maker und YouTuber Luis druckt einen motorisierten, ferngesteuerten Schlepper im ikonischen Benchy-Design und lässt sich dann entspannt über den See ziehen.

von Luis Marx



**D**er Schlepper ist so konstruiert, dass er auf fast jedem 3D-Drucker gedruckt werden kann. Aus diesem Grunde habe ich mich beim Design am bekannten Druckertestobjekt *Benchy* orientiert. Da der Rumpf in zwei Hälften gedruckt wird, müssen diese anschließend mit Epoxydharz zusammengeklebt werden. Es empfiehlt sich außerdem, den ganzen Rumpf mit einer Schicht Harz zu überziehen. Wie beim kleinen *Benchy* ist auch bei meiner Variante ein kleines Fach im Heck des Bootes zu finden. Dieses hat bei meinem Design praktischerweise genau die richtige Größe, um eine Bierflasche zu beherbergen.

Als Antrieb für das Boot kommen zwei Brushless-Motoren zum Einsatz. Die Motoren werden von außen an den Rumpf angeschraubt und die Kabeldurchführungen gut abgedichtet. Da in den Motoren nur komplett isolierte Kupferdrähte verbaut sind, ist ein Betrieb unter Wasser problemlos möglich. Die Bedenken, dass die Kugellager der Motoren bald den Geist aufgeben, blieb bis jetzt unbegründet. Die Motoren haben zusammen eine Leistung von fast 800 Watt. Als Akku kann entweder ein handelsüblicher dreizelliger Li-Po oder auch ein Li-Ion mit ähnlicher Spannung genutzt werden.

Weiterhin habe ich eine eigene wasserdichte Fernbedienung gebaut. Die Funkübertragung wird per *ESP-Now* zwischen den beiden *D1-Mini* (ESP 8266) in Fernsteuerung und Boot realisiert. Das Hauptkriterium an den Sender war, dass er mit einer Hand bedient werden kann und dass langes Steuern des Bootes nicht anstrengend wird. Die Fernsteuerung hat drei Knöpfe: Mit dem Knopf auf der Vorderseite gibt man Schub, wenn man nun gleichzeitig einen der Lenkknöpfe betätigt, fährt das Boot einen leichten Bogen. Werden ausschließlich die Lenkknöpfe gedrückt, wird eine stärkere Kurve gefahren. Alle Knöpfe, gleichzeitig kurz gedrückt, aktivieren einen Tempomat, der so lange aktiv bleibt, bis ein beliebiger Knopf auf der Fernsteuerung gedrückt wird.

Die erste kleine Testfahrt am See endete mit einem Salto. Die Motoren waren einfach zu stark. Bis auf 20% der Leistung gedrosselt reichen sie locker für den Schlauchboot-schlepp. Nachdem alle Tests abgeschlossen waren, konnte es auch schon zur Jungfernfahrt mit dem Stand-Up-Paddle-Board an den See gehen. Die Steuerung über die drei Tasten funktioniert erstaunlich gut. Auch die Power reicht aus, um mich mit ca. 3-5 km/h über den See zu ziehen. Der Akku hält 30-40 Minuten bei durchgehendem Betrieb, auch wenn die Motorregler ab und zu mal eine Abkühlpause brauchen. Meine *Thingiverse*-Seite enthält die Druckdateien, Links zum Video und meinem *GitHub* mit dem Code der Steuerungsanlage. Volle Fahrt voraus! —caw

► [thingiverse.com/thing:5408501](https://www.thingiverse.com/thing:5408501)



Der Rumpf wird mit Kunstharz versiegelt.



Zwei Drohnenmotoren sorgen für massig Power.



Die wasserdichte Fernsteuerung. Beim ersten Test gab es einen Überschlag. Aber dann: Entspannung pur.

# Sortieranlage als Projektarbeit

Eine komplette Eigenkonstruktion vermittelt und lehrt Aspekte der Mülltrennung, Automatisierung, Sensorik und Steuerung per Mikrocontroller.

von Felix Watzlawek



Während ich meine Fachhochschulreife an der Christian-Schmidt-Schule in Neckarsulm im technischen Bereich absolvierte, wurden bereits einige Arduino-Projekte durchgeführt. Bei einem internen Schulprojekt sollte ich mich wiederum mit diesem Thema beschäftigen, wollte jedoch kein bereits behandeltes Projekt aufgreifen. Ich habe mir daraufhin ein aktuelles und großes Problem herausgesucht: die Mülltrennung und deren technische Ausführung.

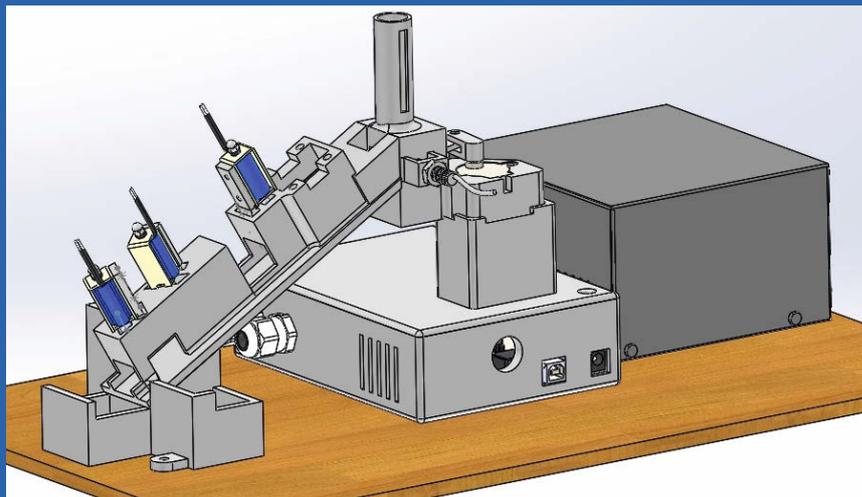
Mein Ziel war eine handliche, auf einem *Arduino* basierte Anlage mit einfach herstellbaren Teilen. Die Anlage soll kleine zylindrische Teile mit einem Durchmesser von 15mm und einer Höhe von 10mm aus Kunststoff zwischen den Farben blau und rot, sowie zwischen Metall und Kunststoff unterscheiden und anschließend sortieren. Es sollten so wenig wie möglich bewegliche Teile verwendet werden, daher entschied ich mich für eine Art Rutsche, auf welcher sich die Klötze durch ihr Eigengewicht herunterschieben und dabei sortiert werden können.

Auf der Rutsche befinden sich Abzweigungen, welche mithilfe von Hubmagneten elektrisch auf und ab bewegt werden. Am oberen Ende der Anlage befindet sich ein Speicher mit den Testobjekten, welche dann nacheinander auf die Rutsche geschoben werden. Mittels eines induktiven Sensors wird zwischen Metall und Kunststoff unterschieden. Bei einer Metalldetektion wird der Zylinder gleich weitergeleitet und durch die Rutsche in den passenden Zielcontainer befördert. Ist der Zylinder nicht aus Metall, wird mithilfe des Farbsensors zwischen Blau und Rot differenziert. Entsprechend werden auf der Rutsche die passenden Abzweigungen durch die Hubmagneten nach oben oder unten gefahren, sodass die Zylinder in ihre Zielcontainer rutschen.

Der gesamte Aufbau wurde in *Solidworks* konstruiert und dann mit dem 3D-Drucker aus PLA-Kunststoff gefertigt. Es waren mehrere Iterationen in der Konstruktion und Programmierung nötig, bis alles reibungslos vonstatten ging. Anfangs gab es Probleme mit der Reibung zwischen den Klötzen und der Rutsche, sodass einzelne Klötze besonders an den Abzweigungen stecken blieben. Auch die Form der Testobjekte wurde von Quadern zu Zylindern geändert, als sich herausstellte, dass die Quader an den Abzweigungen hängen bleiben, was mit zylinderförmigen Teststücken so gut wie unmöglich ist. Die Stromversorgung wird mit einem alten PC-Netzteil gewährleistet, welches die benötigten Spannungen bereitstellt. Diese sind die 12V für die Hubmagnete und 5V für die Sensoren, das Relaismodul und den *Arduino Mega*. Die gesamte Anlage wurde dann ordentlich verkabelt und für eine gute Stabilität und Transportmöglichkeit auf einer Multiplexplatte verschraubt.

—caw

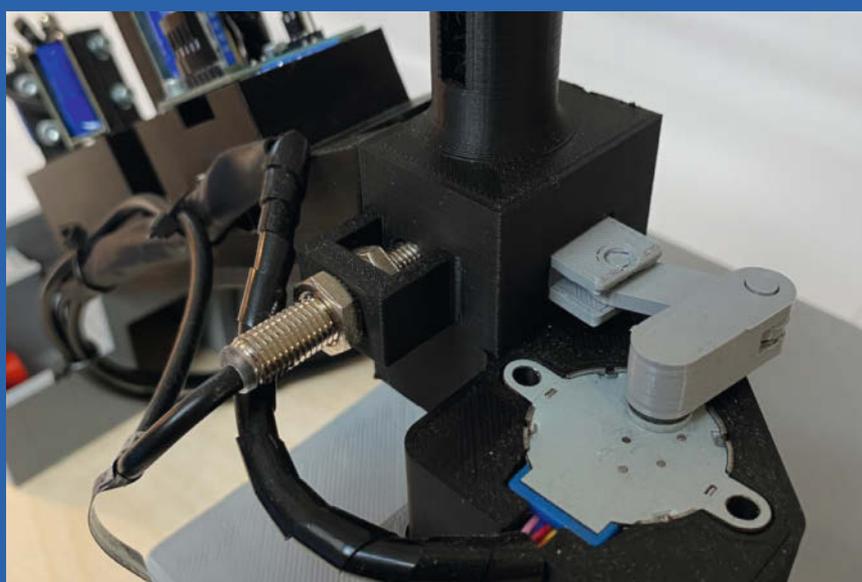
► <https://heise.de/s/AAXq>



Die Anlage wurde in Solidworks entworfen.



Der 3D-gedruckte Elektronikasten dient auch als Basis für die Anlage.



Auswurfmechanismus mit Schrittmotor und Induktiv-Sensor, im Hintergrund befindet sich der Farbsensor.

# Solar-Wetterstation, Teil 2

Temperatur und Co. auf dem Smartphone anzeigen kann unsere Wetterstation ja schon. Nun geht es darum, die Daten inklusive Messzeitpunkt dauerhaft zu speichern und kompatibel zur Wettermühle aus dieser Ausgabe zu machen.

von Heinz Behling



**R**echtzeitig bevor im Herbst nun die interessanten Wettererscheinungen wie Sturm und Kälte auftreten, rüsten wir unsere solarbetriebene Wetterstation aus der vorigen Make-Ausgabe noch um einige wichtige Details auf. In Zeiten des Klimawandels wäre es doch interessant, später auf die Wetterdaten von 2022 zuzugreifen. Vielleicht stellt sich ja heraus, dass dieser Sommer dann eher noch als kühl bezeichnet werden muss.

Also braucht unsere Station ein Speichermedium. Eine SD-Karte bietet sich dafür an. Ob Normal- oder Micro-Ausführung bleibt dabei Ihrer Entscheidung überlassen. Die entsprechenden Module mit den Kartenslots werden auf dieselbe Weise angeschlossen. In dieser Anleitung wird eine Micro-SD-Karte verwendet. Die dort gespeicherten Daten lassen sich aber nur dann sinnvoll verwenden, wenn auch der jeweilige Messzeitpunkt bekannt ist. Den stellt uns die Uhr im ESP32-Chip der Station zur Verfügung, die zuvor mit der genauen Zeit eines Internet-Servers gestellt wird. Und damit die Daten auch möglichst universell weiterverarbeitet werden können, beispielsweise mit einem Tabellenkalkulationsprogramm wie Excel, werden die Daten im **CSV-Format** (**c**omma **s**eparated **v**alues, durch Komma getrennte Werte) auf die Speicherkarte geschrieben. Für jedes Jahr wird darauf ein eigener Ordner angelegt.

Und dann ist da noch etwas zu erledigen: Auf Seite 64 finden Sie die Bauanleitung der Wettermühle, die Windrichtung, -geschwindigkeit, Temperatur und mehr originell anzeigt, ein echter Hingucker also. Sie bezieht die Wetterdaten normalerweise vom Internet-Dienst *OpenweatherMap*, kann aber auch auf unsere eigene Wetterstation hören, wenn die ihre Daten im geeigneten Format, nämlich als JSON-Objekt, verschickt. Genau das bringen wir ihr in diesem Artikel bei.

Viel zusätzliche Hardware ist nicht erforderlich: Ein geeignetes Speicherkarten-Modul gibt es für weniger als 2 Euro. Speicherkarten sind in 32GB-Größe für etwa 7 Euro zu haben. Alles in allem reichen also etwa 10 Euro für den Ausbau.

### Kartenslot einbauen

Die Hardware-Erweiterung ist schnell erledigt: Mit nur sechs Adern wird das Speicherkarten-Modul mit dem ESP32-Board verbunden. Der Schaltplan **1** zeigt die genauen Verbindungen. Das Modul selbst kann an die 3D-Druck-Rückseite der Wetterstation mit etwas dickerem doppelseitigem Klebeband sicher befestigt werden. Jetzt noch eine mit FAT32 formatierte Speicherkarte einlegen und schon ist die Hardware fertig.

Die Speicherkarte muss übrigens nicht sonderlich schnell sein, denn es werden nur sehr geringe Datenmengen (etwa 50Byte pro

## Kurzinfo

- » JSON-Objekte aus Wetterdaten generieren
- » ESP32-Uhr mit Internet-Zeitserver synchronisieren
- » Daten auf SD-Karte schreiben

### Checkliste



**Zeitaufwand:**  
1 Stunde



**Kosten:**  
10 Euro (zzgl. Kosten für Wetterstation aus Make 4/22, Stand August 22)

### Material

- » Wetterstation aus Make 4/22
- » (Mikro-) SD-Kartenslot-Modul für ESP-Boards
- » Speicherkarte 32GB passend zum Slot
- » Schalllitze

### Werkzeug

- » Lötkolben
- » Schraubendreher

### Mehr zum Thema

- » Heinz Behling, Solarstrom für Mikrocontroller, Teil 1, Make 4/22, S. 32
- » Daniel Bachfeld: Die IoT-Alleskönner: ESP32 und ESP8266, Make 6/19, S. 8

Alles zum Artikel im Web unter [make-magazin.de/xw6h](https://make-magazin.de/xw6h)

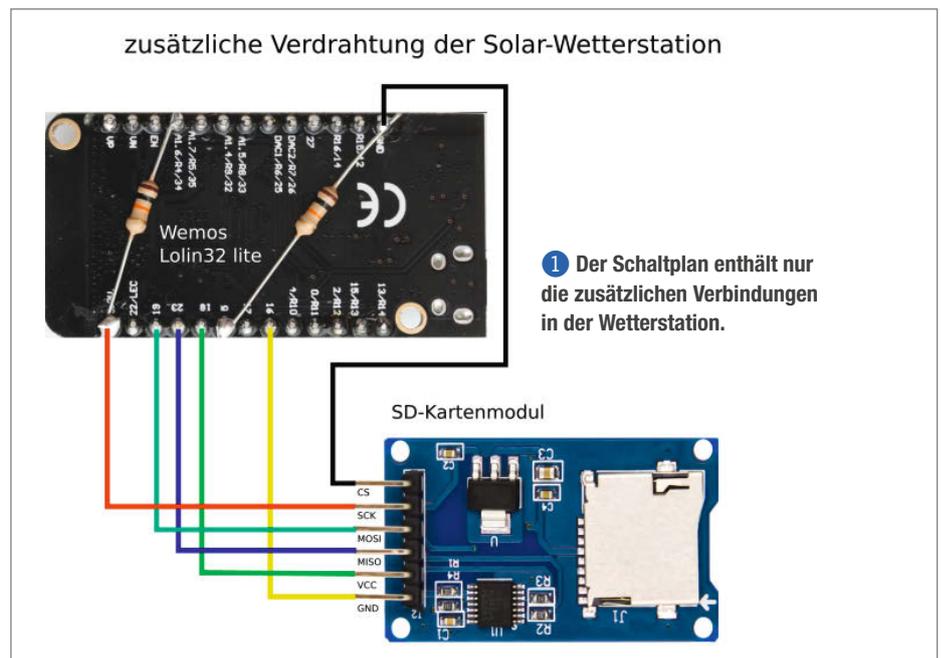
Messung) darauf geschrieben und das auch nur im Minutenabstand. Im Prototyp werkelt eine 32GB-Karte, weil das im Moment die gängige Größe ist. Aber auch kleinere Karten sind geeignet.

Die Daten werden jahresweise in eigenen Ordnern unter dem Dateinamen *Wetterdaten.csv* abgelegt. In einem Jahr sollte diese Datei auf knapp 27 Megabyte (50 Byte × 60 Minuten × 24 Stunden × 365 Tage) anwachsen. Selbst

eine 4GB-Karte würde so rein rechnerisch für 150 Jahre ausreichen, eine gute Gelegenheit also, Uralt-Karten aus der Bastelschublade wiederzubeleben.

### Software aufspielen

Gespeichert wird aber erst, nachdem die neue Software mit der Arduino-IDE auf den ESP32 der Station übertragen wurde. Sie heißt



```

12 const char* ssid = " ";
13 const char* password = " ";
14 String openWeatherMapApiKey = " ";
15 String city = "Hannover";
16 String countryCode = "DE";
17 String IPEigenstation = "192.168.10.74";
18

```

2 Die IP-Adresse Ihrer Station wird natürlich anders lauten als die hier genannte.

### 3 Die Funktion getWebpage()

```

void getWebpage() {
  Serial.println("Get Webpage");
  page += "<head><title>Make-Solarwetter</title></head>";
  page += "<img src='http://www.heise.de/make/icons/make_logo.png'
alt='Make:' height=10%><h1>Solar-Wetter </h1><br>";
  page += "<table>";
  page += "<tr><td><b>Temperatur:</b> </td><td>";
  page += String(temperatur);
  page += " Grad Celsius<br></td></tr>";
  page += "<tr><td><b>Luftfeuchtigkeit:</b> </td><td>";
  page += String(feuchtigkeit);
  page += " %<br></td></tr>";
  page += "<tr><td><b>Luftdruck:</b> </td><td>";
  page += String(druck);
  page += " hPa<br></td></tr>";
  page += "<tr><td><b>Windrichtung:</b> </td><td>";
  page += String(directions[richtung]);
  page += " <br></td></tr>";
  page += "<tr><td><b>Windgeschwindigkeit:</b> </td><td>";
  page += String(geschwindigkeit);
  page += " km/h<br></td></tr>";
  page += "</table>";
  server.send(200, "text/html", page);
}

```

SolarwetterV2.ino und ist über den Kurzinfolink bzw. den QR-Code erreichbar. Genau wie die Wettermühle ist auch diese Software nun OTA-fähig. Sie müssen sie also nur einmal per USB-Kabel übertragen. Danach geht das drahtlos übers WLAN. Als Port finden Sie dann eine Bezeichnung, die mit esp32 beginnt und mit einer IP-Adresse endet. Da eine Wetterstation ja in der Regel draußen befestigt ist, schien mir das eine sehr sinnvolle Arbeitser-

leichterung. Nähere Infos zu OTA finden Sie im Windmühlen-Artikel auf Seite 64.

### Mühle anpassen

Damit die Windmühle ihre Wetterdaten künftig nicht mehr von Openweathermap, sondern von der Solar-Wetterstation holt, ist auch in deren Software eine Anpassung nötig. Aber keine Angst, wenn Sie die Mühle bereits fertig aufgebaut und verklebt haben: Deren Software ist ja ebenfalls OTA-fähig, sodass Sie nicht alles auseinanderreißen müssen, nur um an die USB-Buchse zu kommen.

Die Anpassung findet in der Zeile 17 der Mühlen-Software statt 2. Dort müssen Sie zwischen die Anführungsstriche hinter IP-Eigenstation die IP-Adresse der Wetterstation eintragen. Die hat Ihnen ja nach dem Software-Update die Arduino-IDE in der neuen OTA-Portadresse verraten. Falls Sie später dann mal wieder Openweathermap benutzen möchten, müssen Sie lediglich die IP-Adresse der Wetterstation wieder entfernen, sodass nur die beiden Anführungsstriche übrig bleiben, die Software erneut via OTA auf die Mühle übertragen und ab dann benutzt sie erneut den Internet-Wetterdienst.



4 Die Web-Oberfläche der Station entspricht der Vorversion.

Wenn nun die Mühle erneut startet, sucht sie zunächst ihre Null-Position. Das entspricht der Stellung Flügelachse in Richtung Nord, wie üblich. Dann aber nimmt sie Kontakt zur Wetterstation auf und zeigt deren Daten durch die Flügeldrehzahl und -stellung sowie auf dem Display an.

### Software erklärt

Was geschieht aber in der Wetterstation? Schauen Sie sich dazu am besten einmal den Programmtext in der Arduino IDE an. Zunächst startet das Programm mit der üblichen Einbeziehung diverser Bibliotheken. Neu sind hier gegenüber der Vorversion unter anderem die Bibliotheken SPI.h für die Kontaktaufnahme und SD.h für die Funktionen der Speicherkarte. Außerdem gibt es die time.h, die für die Uhrzeit notwendig ist. Wir wollen ja die Messwerte zusammen mit dem Messzeitpunkt auf der Karte speichern.

Dann folgen die Variablen und Konstanten: Hier wird in Zeile 13 die Adresse des Zeitervers festgelegt:

```
const char* ntpServer = "pool.ntp.org";
```

Die beiden folgenden Zeilen

```
const long gmtOffset_sec = 3600;
const int daylightOffset_sec = 3600;
```

legen die am Ort geltende Abweichung gegenüber der vom Server gelieferten Greenwich-Zeit (3600 Sekunden, also plus eine Stunde) und die Abweichung während der Sommerzeit (ebenfalls eine Stunde) fest. Möchten Sie die Daten mit der international üblichen Greenwich-Zeit speichern, setzen Sie beide Werte auf 0.

In Zeile 41 wird der Webserver inklusive seiner Port-Adresse definiert: `WebServer server(80);`. Wie üblich bei Internetseiten hört er auf Port 80. Es folgen die Inbetriebnahme des SD-Moduls und die Routinen fürs OTA-Update (wie im Mühlenartikel beschrieben).

Das Synchronisieren der ESP32-Uhr mit der Zeit des zuvor definierten Internet-Zeitervers geschieht in Zeile 85:

```
configTime(gmtOffset_sec,
daylightOffset_sec, ntpServer);
```

Mehr ist dafür nicht notwendig. Übrigens: Falls die Wetterstation Probleme hat, den Zeitserver zu erreichen, merkt man dies zuerst an den Reaktionen der Mühle: Die startet dann nämlich in Minutenabständen immer wieder mit dem Suchen der Nullposition, fährt also auf Nord, gleichgültig, woher der Wind weht. Beobachten Sie dies Verhalten, dann checken Sie Ihren Internet-Anschluss!

Interessant wird es in den Zeilen 88 bis 90:

```
server.on("/", getWebpage);
server.on("/weather", getWeather);
server.begin();
```

Hier wird festgelegt, wie der Server auf eine Anfrage aus dem Internet reagiert. Wird der Server beispielsweise von einem Browser aus mit dieser Adresse aufgerufen:

http://192.168.10.74/

dann ist der Schrägstrich hinter der IP-Adresse das Signal, die Funktion `getWebpage()` aufzurufen ③. Die setzt einen String aus HTML-Befehlen zusammen, die wie in der Vorversion an den Browser geschickt wird und dort die Web-Oberfläche der Station mit den Wetterdaten anzeigt ④.

Ruft man die IP-Adresse hingegen so auf:

http://192.168.10.74/Weather

erhält man etwas ganz anderes zurück ⑤, nämlich das JSON-Objekt, das die Mühle braucht.

Wenn Sie sich die Rohdaten anzeigen lassen, erscheint das so, wie die Station es sendet.

```
{ "main": { "temp": 26.91, "pressure": 1011.09, "humidity": 44.95 }, "wind": { "speed": 0.00, "deg": 225 } }
```

### JSON handgemacht

Die Mühle erwartet die Wetterdaten in genau diesem JSON-Format (siehe Mühlenartikel), gleichgültig, ob sie vom Internet-Wetterdienst oder Ihrer Wetterstation stammen. Dementsprechend muss die neue Software der Station solch ein JSON-Objekt aus den Daten für Temperatur, Luftdruck und -feuchtigkeit sowie Windgeschwindigkeit und Richtung zusammenbauen und dann als JSON-gekennzeichnete Daten über den Webserver der Station versenden. Dafür enthält die Software eine eigene Funktion namens `getweather()` ⑥. Sie speichert alles schrittweise inklusive der benötigten Klammern und sonstiger Sonderzeichen in der String-Variablen `buffer1`.

Das beginnt bei JSON immer mit einer öffnenden geschweiften Klammer `{` (Zeile 156 in der Arduino IDE). Darauf folgt der Name des ersten Objekts `main`. Dieser Name muss übrigens in doppelten Anführungsstrichen stehen. Damit der Compiler der Arduino-IDE diese Anführungsstriche aber nicht fälschlicherweise als Anfang beziehungsweise Ende des Variableninhalts versteht und entfernt, muss ein Schrägstrich `\` davor stehen. `main` enthält die Größen für die Temperatur (`temp`), den Luftdruck (`pressure`) sowie die Luftfeuchtigkeit (`humidity`) (Zeilen 157 bis 161).

Dann folgt als nächstes `wind`, das wiederum die Größen `speed` (für die Windgeschwindigkeit) und `deg` (für die Windrichtung) enthält. Alle Bezeichnung halten sich dabei an die von



⑤ So zeigt der Browser Firefox das empfangene JSON-Objekt an.

### ⑥ Funktion `getweather()`

```
void getWeather() {
  Serial.println("Get weather");
  buffer1 = "{ \"main\": { \"temp\": ";
  buffer1 += String(temperatur);
  buffer1 += ", \"pressure\": ";
  buffer1 += String(druck);
  buffer1 += ", \"humidity\": ";
  buffer1 += String(feuchtigkeit);
  buffer1 += ", \"wind\": { \"speed\": ";
  buffer1 += String(geschwindigkeit);
  buffer1 += ", \"deg\": ";
  buffer1 += String(45 * richtung);
  buffer1 += " } }";
  server.send(200, "application/json", buffer1);
  Serial.println("Gesendetes JSON-Objekt: ");
  Serial.println(buffer1);
}
```

Openweathermap verwendeten Namen, damit für die eigene Wetterstation keine Änderungen in der Mühlen-Software notwendig wurden.

Der Befehl `server.send(200, "application/json", buffer1)`; schließlich sendet den Inhalt von `buffer1` als `json` markiert über den Webserver an die Station, die eine entsprechende Anfrage gesendet hat.

Danach startet die Hauptschleife des Programms `loop()`. Dort wird zunächst abgefragt, ob eine Anfrage für ein OTA-Update vorliegt: `ArduinoOTA.handle()`; oder ob es eine Anfrage an den Webserver gibt: `server.handleClient()`; Anschließend werden in den Zeilen 98 bis 102 die aktuellen Messwerte ermittelt.

```
temperatur = bme.readTemperature();
feuchtigkeit = bme.readHumidity();
druck = bme.readPressure() / 100;
richtung = winddirect();
wind();
```

Danach kümmert sich das Programm um die Speicherung der Daten auf die SD-Karte ⑦. Immer, wenn die Sekundenzeit der aktuellen Uhrzeit 0, 1 oder 2 ist, wird die Funktion `sdwrite()` aufgerufen ⑧. Warum diese drei

Werte? Würde eine Überprüfung auf exakt 0 nicht reichen? Leider nein, denn das Programm der Wetterstation braucht länger als eine Sekunde, wenn der ESP32 zum Beispiel gerade die Mühle bedient. Dann würde die Schleife die Sekunde 0 gar nicht erwischen und es würden keine Daten auf die SD-Karte geschrieben. Aus diesem Grund wird der Zeitbereich von Sekunde 0 bis 2 abgefragt. Innerhalb von drei Sekunden wird die Zeitabfrage mit Sicherheit einmal ausgeführt und die Daten für die entsprechende Minuten auch gespeichert. Damit das aber nicht mehrmals geschieht, folgt eine Verzögerung um drei Sekunden mittels `delay(3000)`;

Das Schreiben erfolgt dann in `sdwrite()`. Die Funktion setzt in den Zeilen 240 bis 241 zunächst den Dateinamen bestehend aus dem Jahr als Ordnernamen und dem Dateinamen Wetterdaten.csv zusammen. Danach wird der Datensatz bestehend aus Datum, Uhrzeit und den Wetterdaten gebildet.

Anschließend folgt die Prüfung, ob die Datei bereits auf der Speicherkarte (also in diesem Jahr `/2022/Wetterdaten.csv`) existiert. Falls ein Jahreswechsel stattfand, gibt es diese Datei natürlich noch nicht. Deshalb wird in

## 7 Datenspeichern auf SD-Karte

```
getTimevars();
if (seconds == "00" || seconds == "01" || seconds == "02") {
  if (!sdwrite()) {
    Serial.println("Daten nicht in Datei geschrieben!");
  }
  else {
    Serial.println("Daten in Datei geschrieben!");
  }
  delay(3000);
}
```

|    | A        | B        | C          | D                | E         | F                   | G            |
|----|----------|----------|------------|------------------|-----------|---------------------|--------------|
|    | Datum    | Zeit     | Temperatur | Luftfeuchtigkeit | Luftdruck | Windgeschwindigkeit | Windrichtung |
| 1  |          |          |            |                  |           |                     |              |
| 2  | 01.09.22 | 00:00:00 | 23.46      | 48               | 1018      | 3.50                | 45           |
| 3  | 01.09.22 | 00:00:01 | 23.46      | 48               | 1018      | 3.30                | 45           |
| 4  | 01.09.22 | 00:00:02 | 23.42      | 48               | 1018      | 2.90                | 45           |
| 5  | 01.09.22 | 00:00:03 | 23.42      | 48               | 1018      | 2.10                | 45           |
| 6  | 01.09.22 | 00:00:04 | 23.41      | 48               | 1018      | 3.30                | 45           |
| 7  | 01.09.22 | 00:00:05 | 23.41      | 49               | 1018      | 2.50                | 45           |
| 8  | 01.09.22 | 00:00:06 | 23.41      | 49               | 1018      | 2.00                | 45           |
| 9  | 01.09.22 | 00:00:07 | 23.39      | 49               | 1018      | 1.60                | 90           |
| 10 | 01.09.22 | 00:00:08 | 23.39      | 49               | 1018      | 2.40                | 45           |
| 11 | 01.09.22 | 00:00:09 | 23.39      | 49               | 1018      | 3.60                | 90           |
| 12 | 01.09.22 | 00:00:10 | 23.38      | 49               | 1018      | 3.20                | 90           |
| 13 | 01.09.22 | 00:00:11 | 23.38      | 49               | 1018      | 1.80                | 90           |
| 14 | 01.09.22 | 00:00:12 | 23.39      | 49               | 1018      | 2.10                | 45           |
| 15 | 01.09.22 | 00:00:13 | 23.37      | 49               | 1018      | 2.20                | 90           |
| 16 | 01.09.22 | 00:00:14 | 23.35      | 49               | 1019      | 1.80                | 90           |
| 17 | 01.09.22 | 00:00:15 | 23.36      | 49               | 1019      | 2.50                | 90           |
| 18 | 01.09.22 | 00:00:16 | 23.36      | 49               | 1019      | 2.70                | 90           |
| 19 | 01.09.22 | 00:00:17 | 23.34      | 49               | 1019      | 2.50                | 90           |
| 20 | 01.09.22 | 00:00:18 | 23.35      | 49               | 1019      | 2.10                | 90           |
| 21 | 01.09.22 | 00:00:19 | 23.34      | 49               | 1019      | 2.70                | 90           |

### 9 So sehen die importierten Daten aus.

diesem Fall mit dem Befehl `SD.mkdir(Dirname)` ein neuer Ordner mit der aktuellen Jahreszahl angelegt. Das geschieht natürlich auch, wenn die Station mitten im Jahr mit einer leeren Speicherkarte startet. Außerdem wird

dann die Datei mit `File Datas = SDopen(Datafile, FILE_WRITE)`; zum Schreiben geöffnet und die Kopfzeile *Datum, Zeit, Temperatur, Luftfeuchtigkeit, Luftdruck, Windgeschwindigkeit, Windrichtung* geschrieben.

Existiert die Datei hingegen bereits, wird kein neuer Ordner angelegt und die Datei mit `File Datas = SD.open(Datafile, FILE_APPEND)`; geöffnet. Die Verwendung von `FILE_APPEND` statt `FILE_WRITE` bewirkt, dass der folgende Befehl `Datas.println(Datastring)`; den neuen Datensatz an die bereits vorhandenen anfügt.

Schließlich wird die Datei wieder mit `Datas.close()`; geschlossen.

## Daten in Tabellenkalkulation einlesen

Um die Wetterdaten auf der Speicherkarte auszuwerten, sind alle gängigen Tabellenkalkulationsprogramme geeignet, wie Excel oder Open Office Calc [9](#). Sie müssen darin die Daten entweder importieren oder direkt öffnen. Als Dateityp ist eine csv-Datei erforderlich.

## Noch Ideen?

Falls Sie noch weitere Ideen für das Team Solar-Wetterstation und Windmühle haben oder andere originelle Anzeigegeräte entwickeln, würden wir uns in der Redaktion sehr freuen, wenn Sie uns Ihre Einfälle mitteilen. Vielleicht wird daraus ja eine weitere Folge der Wetterstation-Serie oder zumindest eine Erwähnung bei den Leserbriefen oder auf der Make-Internetseite. Viel Spaß beim Knobeln und Basteln! —hgb

## 8 Die Funktion sdwrite()

```
boolean sdwrite() {
  String Datafile = "/" + year + "/Wetterdaten.csv";
  String Dirname = "/" + year;
  String Datastring = datestring + "," + timestring + "," + temperatur + "," + feuchtigkeit + "," + druck + "," + geschwindigkeit + "," + 45*richtung;

  Serial.println("Data-Zeile");
  Serial.println(Datastring);
  if (!SD.exists(Dirname)) {
    SD.mkdir(Dirname);
  }
  if (!SD.exists(Datafile)) {
    File Datas = SD.open(Datafile, FILE_WRITE);
    Datas.println("Datum,Zeit,Temperatur,Luftfeuchtigkeit,Luftdruck,Windgeschwindigkeit,Windrichtung");
    Serial.print("Kopfzeile geschrieben");
    Serial.println("Datum,Zeit,Temperatur,Luftfeuchtigkeit,Luftdruck,Windgeschwindigkeit,Windrichtung");
    Datas.close();
  }
  File Datas = SD.open(Datafile, FILE_APPEND);
  if (Datas) {
    Datas.println(Datastring);
    Datas.close();
    return true;
    Serial.println("Daten in Wetterdaten.csv geschrieben");
  }
  else {
    Serial.println("Error, konnte Daten nicht in Wetterdaten.csv schreiben");
    return false;
  }
}
```

# Neuer Input für Maker

## Make Elektronik Special

Make Elektronik Special bietet einen einfachen und praxisorientierten Einstieg in Transistorschaltungen, die Maker in eigenen Projekten einsetzen können. Das mitgelieferte Experimentierset inkl. Breadboard, Kabeln und 45 Elektronikbauteilen enthält alles, um die gezeigten Schaltungen sofort nachzubauen und testen zu können.

Heft + Experimentierset für 44,95 €

[shop.heise.de/make-elektronik21](https://shop.heise.de/make-elektronik21)



Inklusive Experimentierset und Breadboard

## Make Micropython Special

Diese Make-Sonderausgabe zeigt Ein- und Umsteigern, wie man mit MicroPython leicht und schnell eigene Projekte mit dem ESP32 umsetzt.

Wie immer in Make Specials geht's sofort in die Praxis mit Audio-Projekten, einer CO2-Ampel und mehr.

Heft + PDF für 19,90 €

[shop.heise.de/make-micropython](https://shop.heise.de/make-micropython)



## Make Picaxe Special

Noch einfacher als Arduino: Im neuen PICAXE Special der Make dreht sich alles um den Einstieg ins Programmieren mit BASIC. Dazu gibt es ein neuentwickeltes Programmierboard für den Einsatz von PICAXE-Chips, das Nano-Axe-Board mit USB-Anschluss. Damit können Sie sofort starten!

Heft + Nano-Axe-Board für 24,95 €

[shop.heise.de/make-picaxe](https://shop.heise.de/make-picaxe)



Inklusive Nano-Axe-Board mit PICAXE-08M2

# Wetter- und Wind-Mühle

Erinnern Sie sich noch an die kleinen bunten Wetterhäuschen, aus deren Türchen bei sonnigem Wetter eine gut gelaunte Frau, bei Regenlage ein eher missmutig dreinschauender Herr mit Regenschirm hervortraten? Das führte zur Idee einer Wetter-Windmühle, die ähnlich originell Wind- und Wärmedaten buchstäblich anschaulich macht.

von Heinz Behling



Ob Sie nun Ihre eigene Wetterstation gebaut haben, wie in der vorigen Make beschrieben, oder die Wetterdaten von einem frei zugänglichen Dienst wie *OpenweatherMap* beziehen: Irgendwie möchte man sich die wichtigsten Daten doch anzeigen lassen. Übers Smartphone kann das jeder, ist ja auch praktisch für unterwegs. Zu Hause jedoch sollte es schon etwas Originelles sein. Diese Windmühle macht ihrem Name alle Ehre: Die Flügeldrehzahl zeigt die Windstärke an. Die Richtung, in die die Flügel weisen, entsprechen der Windrichtung. Und schließlich zeigt sie im oberen Teil der Eingangstür noch die Temperatur und weitere Wetterdaten an.

Übrigens kann man die hier benutzen Methoden nicht nur zur Wetteranzeige, sondern auch in einer Modellbahn-Anlage einsetzen, um Standard-Bausätze aufzupeppen und ihnen mittels diverser Bewegungseffekte mehr Leben einzuhauchen. Mit Mikrocontroller, kleinen Motoren und etwas Software entsteht so schnell ein ganz persönliches Miniatur-Wunderland.

## Prinzip

In der Windmühle arbeitet ein ESP8266-Mikrocontroller. Da es in der Mühle eng zugeht, habe ich ein platzsparendes *Wemos-D1-mini*-Board verwendet. Der Controller hat über WLAN Kontakt zu *OpenweatherMap* im Internet beziehungsweise zur eigenen Wetterstation. Von dort bekommt er die Daten im JSON-Format (siehe Kasten 1 *Was ist JSON?*).

Die Windgeschwindigkeit und -richtung sowie die Temperatur, Luftdruck- und feuchtigkeit werden aus dem Datensatz extrahiert. Entsprechend der Windrichtung dreht dann ein kleiner Schrittmotor die Kappe der Windmühle (deshalb *Kappenwindmühle*, auch *Holländerwindmühle* genannt) so, dass die Achse der Flügel in Windrichtung zeigen.

Die Windgeschwindigkeit wird in die Windstärken von 0 bis 12 umgerechnet. Die Spannung des Gleichstrommotors, der die Flügel antreibt, wird entsprechend der Windstärke in 13 Stufen von Stillstand bis zur höchsten Drehzahl über einen analogen Ausgang des Controllers und einen MOSFET (**metal-oxide-semiconductor field-effect transistor**, Metall-Oxid-Halbleiter-Feldeffekttransistor) entsprechend gesteuert, sodass sich bei geringem Wind die Flügel nicht oder nur langsam drehen, bei Sturm jedoch auf vollen Touren laufen.

Übrigens hat der hierfür verwendete Gleichstrommotor ein Getriebe mit einer Übersetzung von 1:298 angebaut. Falls der Ihnen bei geringer Geschwindigkeit zu schnell drehen sollte (siehe Video zum Artikel), es gibt auch Motorversionen mit einer Übersetzung von 1:1000, die dann nur ein Drittel der Drehzahl erreichen, aber größer sind.

## Kurzinfo

- » Windgeschwindigkeit und -richtung mit Windmühlenflügeln anzeigen
- » Modellbahn-Bausatz mit ESP8266 steuern
- » Ansteuerung von DC- und Schrittmotor sowie I<sup>2</sup>C-Display
- » Firmware Over The Air (OTA) auf ESP-Modul übertragen

## Checkliste



### Zeitaufwand:

3 bis 4 Stunden



### Kosten:

ca. 80 bis 90 Euro (Stand 25.8.22)

## Material

- » Windmühlen-Bausatz Faller 130115
- » Controllerboard Wemos D1 mini oder kompatibles Board
- » ProtoBoard-Shield passend zu Wemos D1 mini
- » Schrittmotorset Daypower S-SPSM-5V
- » OLED-Display 0,49 inch 62 × 32 I2C
- » Getriebemotor G298 Pollin Best.-Nr. 820221
- » Kugellager 608zz 8mm Innendurchmesser × 22mm Außendurchmesser × 7mm Breite
- » MOSFET BS108 oder BS170
- » Kohleschicht-Widerstand 220Ω
- » Kohleschicht-Widerstand 10KΩ
- » Diode 1N4148
- » Netzteil 5V 2000mA oder Qi-Set Youmile 5V 2A und Netzteil 12V 2A
- » Schaltlitze 0,14mm<sup>2</sup> am besten in verschiedenen Farben
- » Modellbau- und 2-Komponenten-Kleber
- » M3-Schrauben 3 Stück 8mm und 2 Stück 30mm Länge
- » 2 M3-Muttern
- » Dupont-Stecker

## Werkzeug

- » 3D-Drucker
- » Pinzette
- » Lötkolben
- » Crimp-Zange für Dupont-Stecker

## Mehr zum Thema

- » Heinz Behling, Solarstrom für Mikrocontroller, Teil 1, Make 4/22, S. 32
- » Carsten Meyer, Motoren und Antriebe, Make 4/16, S. 102
- » Video mit Vorführung der Windmühle



Alles zum Artikel im Web unter [make-magazin.de/xu1h](https://make-magazin.de/xu1h)



Der Temperaturwert sowie alle anderen Wetterdaten sind auf einem kleinen OLED-Display sichtbar, das sich hinter der oberen Hälfte der geteilten Eingangstür verbirgt.

Bleibt noch die Stromversorgung: Zwei Varianten stehen hier zur Auswahl, ein klassisches Steckernetzteil (5V, 2A), das über ein Kabel die Mühle direkt versorgt. Oder ein drahtloses Qi-Ladegerät, dessen Empfangsspule und -elektronik im Sockel der Mühle untergebracht ist. Die Mühle wird dann auf das Qi-Ladegerät (Sender) gestellt, ist dort aber frei beweglich und kann so leichter entsprechend der Himmelsrichtung ausgerichtet werden. Übrigens sollte nur das in den Bezugsquellen genannte Qi-Set benutzt werden, da dessen Sender mit 12V versorgt werden kann.

Nur so gibt der Empfänger ausreichend Strom ab. Herkömmliche Qi-Ladegeräte von Smartphones reichen in der Regel nicht. Leider sind diese Qi-Ladegeräte auch noch nicht sehr ausgereift. Meines streikte nach gut 24 Stunden, weswegen ich davon abrate.

## Rohbau

Grundlage ist ein Bausatz der Firma *Faller* mit der Nummer 130115. Der Bausatz enthält übrigens einen eigenen Wechselstrommotor 2, der für unsere Mühle aber nicht geeignet ist. Tipp: Bieten Sie ihn auf eBay an – dort werden um die 20 Euro für solche Motoren bezahlt. Das refinanziert schon einen Teil der Baukosten.

## 1 Was ist JSON?

JSON (Java Script Object Notation) ist ein unabhängiges Datenformat mit einer sehr einfachen Struktur, um Daten zwischen Systemen auszutauschen. Ein JSON-Dokument wird zwischen Anwendungen immer als Ganzes ausgetauscht. Der Inhalt eines JSON-Dokuments ist grundsätzlich objektorientiert aufgebaut.

Der Inhalt eines JSON-Dokuments folgt immer einer strikten Struktur. Ein JSON-Dokument beginnt und endet immer mit { und }. Zwischen den geschweiften Klammern werden die Inhalte platziert. Die geschweiften Klammern umfassen ein Objekt. Inhalt solch eines Objekts können ein oder mehrere Datenfelder oder auch weitere Objekte sein, die dann ebenfalls in geschweiften Klammern eingefasst werden. Auch Arrays sind erlaubt, die dann in eckigen Klammern stehen müssen.

Ein Datenfeld wird mit einem Namen eingeleitet und der Wert folgt nach einem Doppelpunkt. Der Wert kann dann wieder ein Objekt sein. Dabei gelten einige Regeln, unter anderem:

- » Alle Eigenschaftsnamen in einem Objekt müssen in doppelten Anführungszeichen stehen.
- » Führende Kommas in Objekten und Arrays sind verboten.
- » Bei Zahlen sind führende Nullen verboten und einem Dezimalpunkt muss mindestens eine Ziffer folgen.

» Strings müssen durch doppelte Anführungszeichen begrenzt sein.

Hierzu ein Beispiel mit den Daten eines Mannes, einem Array namens *kinder*, das zwei Objekte, bestehend aus den Daten seiner beiden Kinder, enthält:

```
{
  "vorname": "Georg",
  "nachname": "Meier",
  "alter": 52,
  "verheiratet": false,
  "beruf": "Ingenieur",
  "kinder": [
    {
      "name": "Lukas",
      "alter": 19,
      "schulabschluss": "Gymnasium"
    },
    {
      "name": "Lisa",
      "alter": 14,
      "schulabschluss": null
    }
  ]
}
```

JSON wird oft benutzt, um Daten völlig unabhängig von Anwendungsprogrammen oder Programmiersprachen auszutauschen. Für die meisten Programmiersprachen stehen JSON-Generatoren bereit, um Daten zu JSON-Objekten zusammenzufassen, sowie Parser, um die einzelnen Daten wieder daraus zu extrahieren.



2 Laut diversen Foren im Netz ist dieser Motor von Faller nicht der haltbarste, weswegen er als Ersatzteil bei eBay gefragt ist.

Aus dem Bausatz sollten Sie zunächst den Sockel, den schmal zulaufenden Dachteil, die Flügel und die Kappe zusammenbauen 3.

### 3D-Druck

In der Mühle geht es eng zu, trotzdem müssen zahlreiche Teile dort untergebracht werden 4.

Für die ordentlich und insbesondere kurzschlussfreie Unterbringung sorgen diverse 3D-gedruckte Teile (siehe Kasten 5 Die 3D-Druckteile), für die Sie die Druckdateien über den Kurzinfo-Links downloaden können. Für den Prototypen wurden sie auf einem Prusa MK3S+ gedruckt, der besonders maßgenau arbeitet, sodass zusammenzusteckende Teile ohne Nachbearbeitung ineinander passen. Bei anderen Druckern muss das nicht so sein, sodass eventuell mit Feile oder Schleifpapier etwas nachgearbeitet werden muss.

Nach dem Druck entfernen Sie sorgfältig alle Stützstrukturen. Insbesondere an den Stellen, an denen die Teile später an den Bauteilen des Bausatzes anliegen, dürfen keine Stützenreste vorhanden sein. Andernfalls würden die Teile schräg stehen und die Mühle sich nicht richtig bewegen können.

Als Erstes nehmen Sie den Stepperhalter und befestigen mit kurzen M3-Schrauben den Schrittmotor und die Lichtschranke 6. Die Lichtschranke muss genau senkrecht stehen. Sie dient später in Verbindung mit der Kappenachse dazu, die Nullposition der Windrichtung nach Einschalten der Mühle zu ermitteln.

Legen Sie dann das große graue Achteck in den Mühlensockel (Achtung, Aussparung an der Rückseite muss passen) und stecken Sie den Stepperhalter hinein 7.

Die Lichtschranke muss der Aussparung direkt gegenüber und über der Eingangstür der Mühle liegen. In dieser Stellung kleben Sie den Stepperhalter mit 2-Komponenten-Kleber am Achteck fest. Das Achteck darf aber noch nicht am Mühlensockel festgeklebt werden!



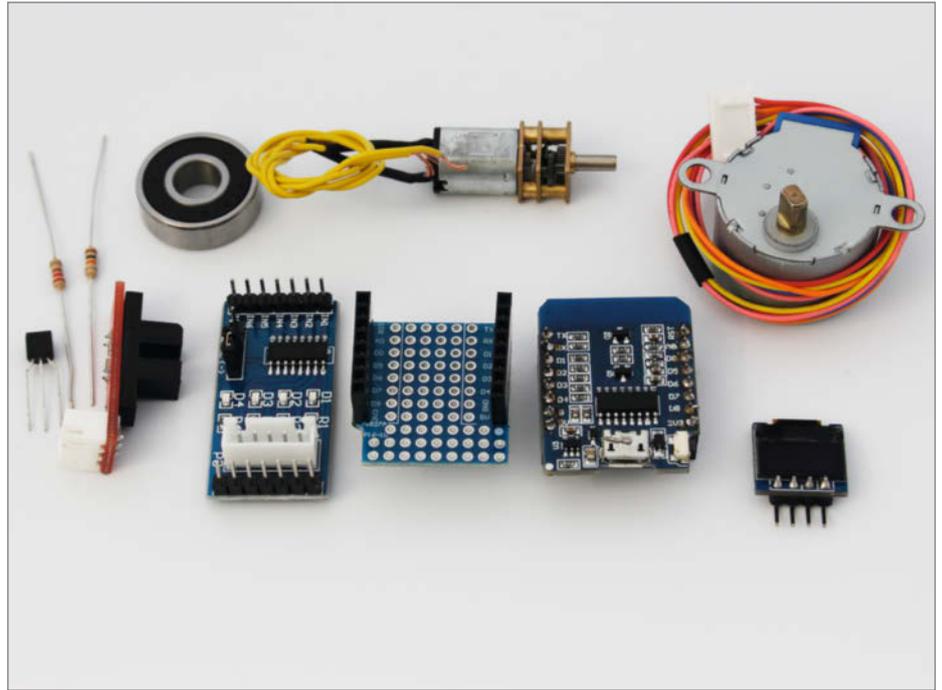
3 Soweit sollten Sie den Bausatz zusammenbauen. Die Anleitung von Faller ist dabei sehr hilfreich.

Ins Lagergehäuse für das Kappenlager setzen Sie das Kugellager ein **8**. Es muss nicht weiter befestigt werden. Achten Sie aber darauf, dass es parallel zur Oberfläche des Halters sitzt. Ist das nicht der Fall, befinden sich noch Reste der Stützstruktur in der Öffnung des Halters.

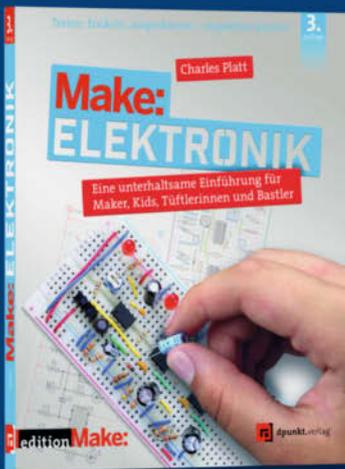
Um die Passgenauigkeit zu testen, stecken Sie den Lagerhalter einmal in den Dachteil der Mühle: Der Lagerhalter hat an seinem achteckigen Unterteil eine breitere Seite. Die muss dort liegen, wo im Dachteil innen ein Pfeil ist **9**.

### Elektronik vorbereiten

An einige Elektronik-Teile müssen Anschlussdrähte gelötet werden. Grund: In der Mühle ist nicht genug Platz, um alle Verbindungen mit Steckern zu versehen. Im Folgenden werden hier zwar auch gecrimpte Dupont-Stecker benutzt, aber nur an einer Seite einer Leitung. Das erleichtert den Zusammenbau und ein eventuelles Auseinandernehmen später bei einer Reparatur oder ähnlichem. Es geht aber notfalls auch ganz ohne Stecker. Dann müssen halt beide Seiten gelötet werden. Die Anschlussleitungen sollten dann jeweils ein



**4** All diese Teile müssen in der Mühle eingebaut werden. Obere Reihe: Kugellager für Kappe, Flügelmotor, Schrittmotor; untere Reihe: MOSFET, Widerstände, Lichtschranke, Stepper-Treiberplatine, Lochraster-Shield, Controllerboard, Display



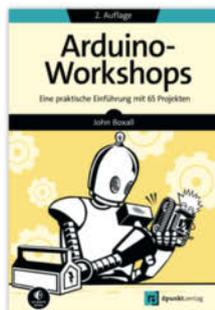
## Die beliebte Elektronik-Schule von Charles Platt – neu in dritter Auflage!

Baue innerhalb weniger Stunden einen Einbruchsalarm, ein Quizspiel oder Zahlenschloss, lass' dabei auch mal eine Sicherung schmelzen oder eine LED durchbrennen – und: lerne ganz nebenbei alle Grundlagen von Spannung, Strom, Widerstand, Kapazität, Induktivität, Logikchips, Funk, Mikrocontrollern und Elektromagnetismus (und vielem mehr) kennen. Für alle ab 10 Jahren, die Freude an Elektronik und Experimenten haben.

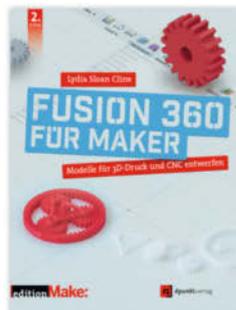
3. Auflage 2022 · 366 Seiten · 36,90 €  
ISBN 978-3-86490-867-5



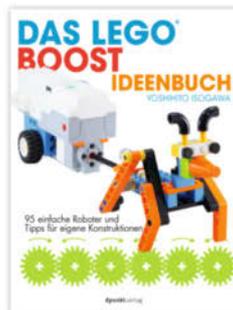
248 Seiten · 27,90 €  
ISBN 978-3-86490-905-4



508 Seiten · 34,90 €  
ISBN 978-3-86490-918-4



382 Seiten · 34,90 €  
ISBN 978-3-86490-866-8

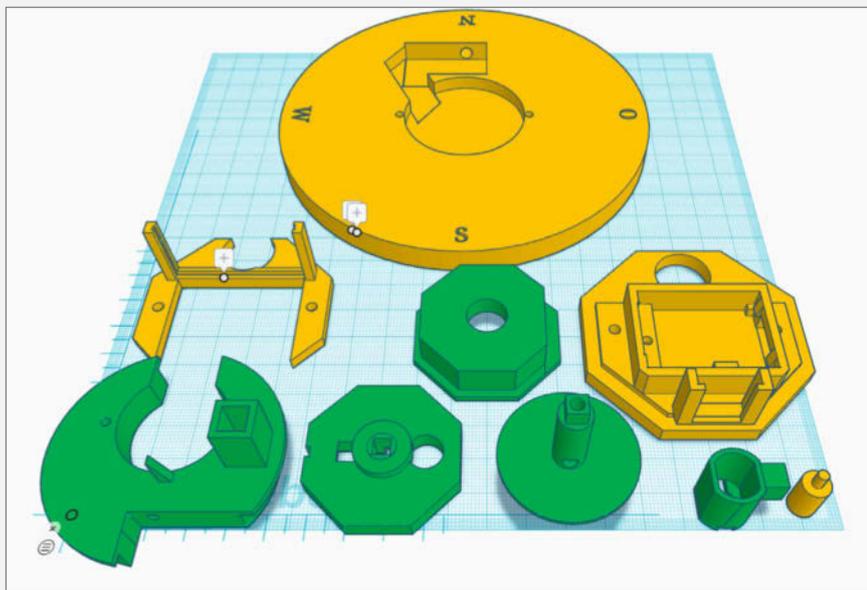


274 Seiten · 27,90 €  
ISBN 978-3-86490-897-2



## 5 Die 3D-Druckteile

Insgesamt neun Teile sind für die Mühle zu drucken. Als Material wurde PLA verwendet und mit einer Fülldichte von 15 Prozent gedruckt. Im Bild sind alle Teile enthalten: Die gelb eingefärbten müssen ohne Stützstrukturen, die grünen mit Stützen gedruckt werden.



paar Zentimeter länger sein, damit man beim Zusammenbau etwas mehr Spielraum hat.

Der Schaltplan 10 zeigt, wie am Ende alles miteinander verbunden sein muss.

Für die MOSFET-Schaltung zur Steuerung des Flügelantriebs verwende ich hier eine kleine Lochrasterplatine, die in der Größe dem Wemos-D1-mini-Board entspricht und direkt darauf aufgesteckt werden kann (siehe Bild 4 unten in der Mitte). Diese Platine ist außerdem der zentrale Verteilerknoten der Mühle, denn für jede andere Komponente enthält sie Kontaktstifte, auf die die Dupont-Stecker jeweils passen. Wie genau diese kleine Platine verdrahtet werden muss, sehen Sie im Kasten 11 *Die Lochraster-Platine*.

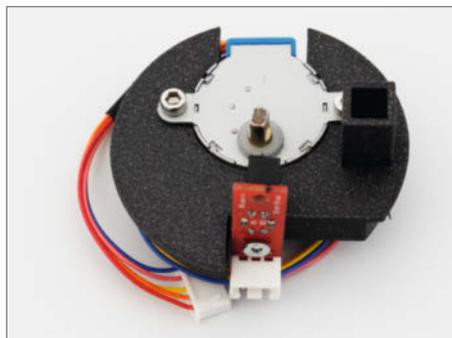
Als Nächstes setzen wir das Display in die Grundplatte der Mühle. Die Anschlussstifte des Displays müssen dazu vorsichtig um 90 Grad gebogen und jeweils mit einem etwa 5 bis 6cm langem Stück Schalllitze versehen werden 12. Ich empfehle hier, Litzen mit verschiedenen Farben zu verwenden. Das erleichtert beim Zusammenstecken in der Enge die Arbeit enorm. Die Farben in den folgenden Fotos entsprechen jeweils den Farben im Schaltplan.

Dann folgt die Treiberplatine für den Schrittmotor: Sie sitzt in einem eigenen Halterahmen. Löten Sie insgesamt sechs Litzenstücke an die Pins IN1 bis IN4 sowie an GND und VCC 13. In meiner Mühle habe ich für die GND-Leitungen stets schwarze Litzen, für VCC (5V) rote benutzt.

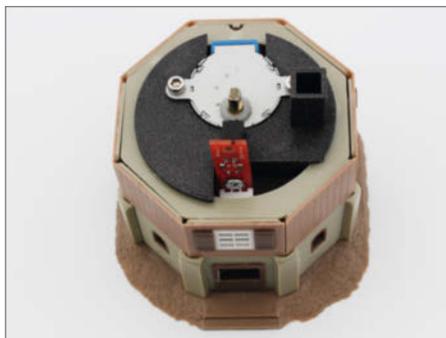
Falls Sie die Mühle über Qi mit Strom versorgen möchten, löten Sie jeweils etwa 12cm Litze (mit Steckern) an den Plus- und Minus-Ausgang der Qi-Platine. Stecken Sie dann Spule und Platine in die Sockelplatte 14. Falls Sie ein herkömmliches 5V-Steckernetzteile benutzen möchten: Schneiden Sie den Stecker am Ende des Kabels ab und führen Sie das Kabel durch das seitliche Loch in die Sockelplatte ein. Auch an dieses Kabel sollten Sie dann einen 2poligen Dupont-Stecker anbringen. Mit einem Multimeter messen Sie nach, welches der Pluspol ist und markieren ihn bitte.

Hier nochmal einen kleinen Kommentar zur Qi-Version: Dieser Aufbau lief bei meinem Prototyp zunächst problemlos. Dann aber setzte die Stromversorgung aus und schließlich drangen Rauchwölkchen aus der Ladeplatte unter der Mühle. Ich rate daher davon ab. Sollte sich die Qualität einmal verbessern, kann man das ja nachrüsten.

Falls Sie die Qi-Variante bauen, brauchen Sie auch das entsprechende Ladegerät. Auch dafür gibt es eine Druckdatei zum Download. Legen Sie die Spule in die Mitte der Ladeplatte, stecken Sie das Kabel des in dieser Variante erforderlichen 12V-Netzteils durch das seitliche Loch der Platine und löten Sie es an die Qi-Elektronik. Beachten Sie die Polung: Messen Sie nach, falls die Adern nicht markiert sind. Beim Netzteil aus den Bezugsquellen ist die Minus-Ader gekennzeichnet 15.



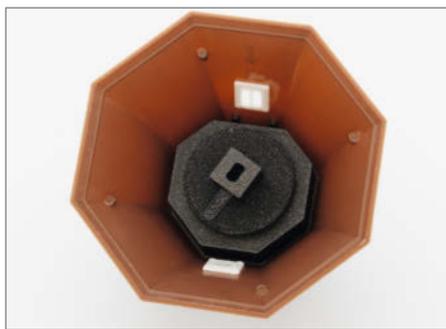
6 Der Schrittmotor und die Lichtschranke werden einfach angeschraubt. Die Lichtschranke muss senkrecht ausgerichtet sein.



7 Der Zapfen am Mühlensockel muss in der Aussparung des Achtecks liegen. Der kleine Schacht rechts dient zur Durchführung des Kabels für den Flügelmotor.



8 Dieses Kugellager hält später die Achse und die Kappe der Mühle.



9 Die Achse sollte sich leicht ins Lager einschieben lassen, aber möglichst spielfrei sitzen.

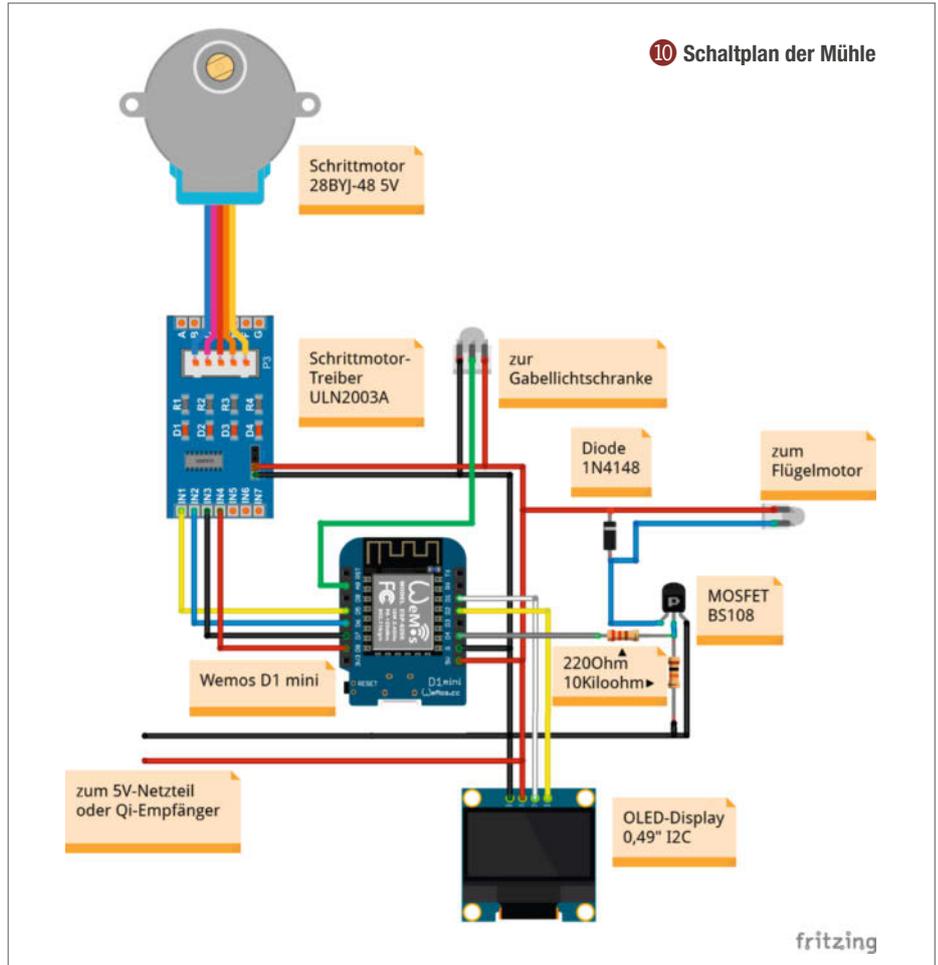
## Software aufspielen

Da es nach dem Zusammenbau eng wird in der Mühle und man kein USB-Kabel mehr an das ESP-Board anschließen kann, muss jetzt bereits die Firmware auf den ESP übertragen werden. Sie ist selbstverständlich auch per Download über den Kurzinfo-Link zu bekommen. Die Firmware ist OTA-Update-fähig (siehe Kasten 16 *Over The Air*), kann also später ohne USB-Kabel übers WLAN geändert werden. Nur beim ersten Mal ist eine Kabelverbindung notwendig.

Laden Sie sie in die Arduino-IDE. Die Zeilen 12 bis 16 müssen Sie entsprechend Ihren örtlichen Verhältnissen anpassen 17. Den API-Key erhalten Sie von OpenweatherMap. Den Link dorthin erfahren Sie in der Kurzinfo-Linkliste. Hier können Sie auch die Zugangsdaten zu Ihrer eigenen Selbstbau-Wetterstation eintragen. Wie das geht, lesen Sie im zweiten Teil des Wetterstationsartikels auf Seite 58.

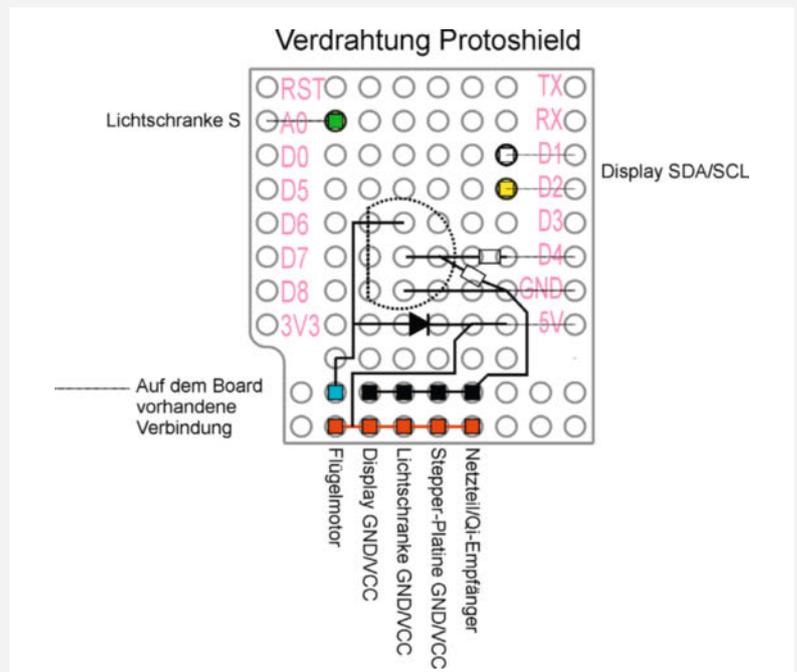
Außerdem müssen einige Bibliotheken in der IDE installiert sein. Falls nicht, holen Sie das nach. Die Liste der erforderlichen Bibliotheken sowie die Links zum jeweiligen Download der ZIP-Dateien finden Sie über den Kurzinfo-Link. Installiert werden Sie über *Sketch/Bibliothek einbinden/.ZIP-Bibliothek hinzufügen*.

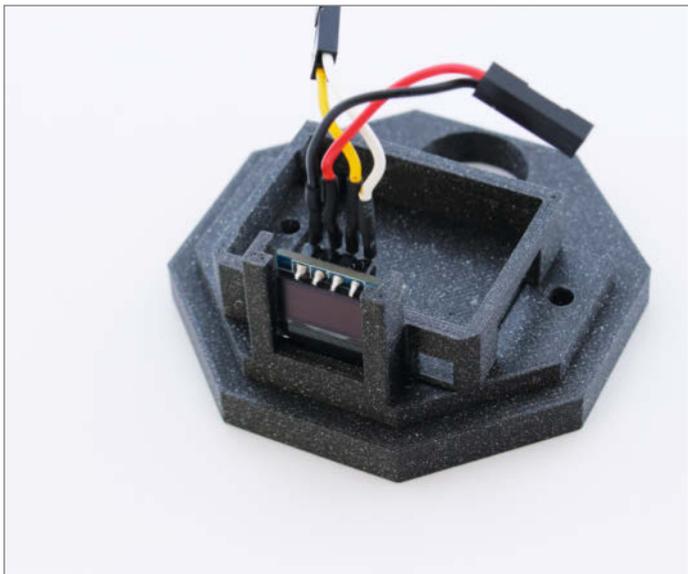
Die Software kompilieren Sie und übertragen sie dann auf das Board. Stellen Sie unter *Werkzeuge* den richtigen Board-Typ ein, nämlich *LOLIN(WEMOS) D1 mini (clone)*, und den



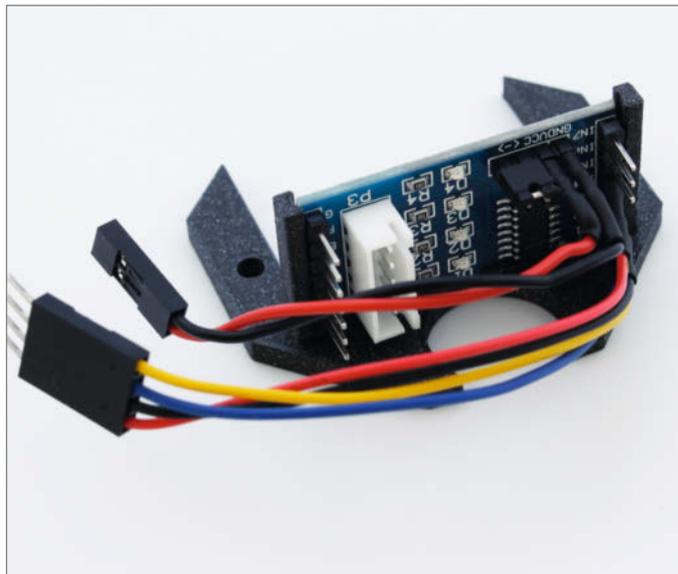
## 11 Die Lochraster-Platine

Die kleine Lochrasterplatine erleichtert die Verkabelung ungemein. Sie ist gleichzeitig Träger für den MOSFET und seine begleitenden Bauteile, enthält aber auch Steckanschlüsse für alle weiteren Teile. Im Bild sind die Anschlüsse jeweils mit den gleichen Farben versehen wie im Schaltplan. Außerdem sehen Sie dort, wie Sie den MOSFET, die beiden Widerstände und die Diode einlöten sowie die benutzten Lötunkte miteinander verbinden müssen. Gepunktete Verbindungen sind bereits auf der Platine von Haus aus vorhanden. Die Platine gibt es inklusive einlötbare Kontaktleisten für etwa 2 Euro zu kaufen.





**12** Das Display in der Grundplatte sollte mit ein wenig 2-Komponenten-Kleber an den beiden oberen Ecken gegen Herausrutschen gesichert werden.



**13** Auch die Treiberplatine sollte mit etwas Kleber in ihrem Halterahmen gesichert werden.

richtigen Port (in der Regel der mit der höchsten Nummer).

Nachdem alle Teile mit Kabeln vorbereitet, die Lochrasterplatine gelötet und das ESP-Board programmiert ist, beginnt der Zusammenbau. Als Erstes stecken Sie von unten zwei M3 x 30-Schrauben in die beiden Löcher der Sockelplatte. In das Unterteil der Mühle stecken Sie ebenfalls von unten die Grundplatte mit dem bereits eingeklebten Display und dem eingesetzten Controller-Board. Stecken Sie beides auf die Schrauben des Sockels.

Als nächstes wird der Halter für die Stepper-Platine ebenfalls auf die beiden Schrauben gesteckt. Mit zwei Muttern wird der ganze Sandwichpack zusammengesraubt. Dabei ist eine Pinzette zum Halten der Muttern sehr hilfreich **18**.

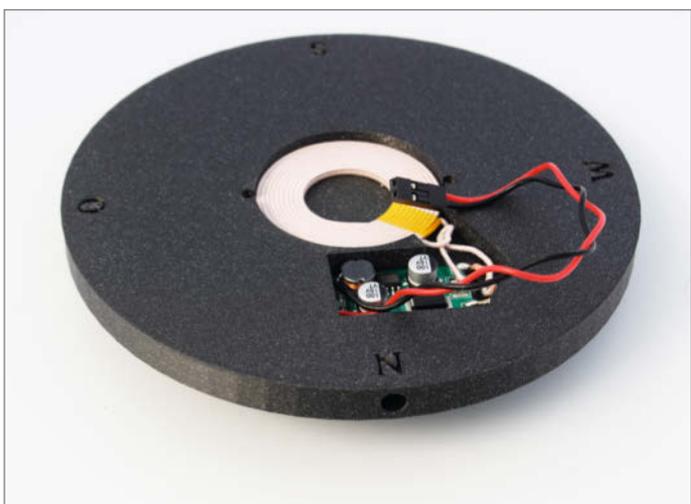
Schließen Sie nun die Stecker des Displays, der Stepper-Platine und des Lichtschrankenkabels auf die Steckverbinder der Lochraster-Platine (siehe auch Kasten **11**.) Die Stecker für das Display und den Stepper müssen etwas heruntergebogen werden, damit noch genügend Platz darüber für den Schrittmotor und die Lichtschranke bleibt. Stecken Sie auch den Stecker des Motors in die Stepper-Platine und das Lichtschrankenkabel in die Lichtschranke **19**.

Die achteckige Platte legen Sie jetzt in das Mühlenoberteil ein. Das geht nur in einer Position, da das Achteck hinten eine Aussparung hat, in die ein Zapfen des Unterteils greift **7**. Mit drei bis vier kleinen Tropfen Modellbaukleber heften Sie die Platte am Unterteil an. Die Verbindung sollte halten, aber noch

lösbar sein, falls später beim Test etwas nicht funktioniert.

Jetzt geht es um die oberste Etage der Mühle: die Kappe mit dem Flügelmotor. Um den mit Strom zu versorgen, fertigen Sie ein etwa 10cm langes Verbindungskabel **20** vom Erdgeschoss zur ersten Etage. Das stecken Sie dann auf den Motoranschluss der Lochraster-Platine **11** und führen es durch den kleinen Schacht der Schrittmotorplatte **7** nach oben.

Dann folgt das nächste Stück, das sich zunächst um die Achse der Kappe wickelt und dann durch die Achse zum Motor geht. Das Einfädeln gelingt am besten vom oberen Ende der Achse aus. Die Bohrung im Inneren ist am unteren Ende schräg nach außen geführt, sodass die Kabel sich praktisch von selbst hindurchfädelt **21**.



**14** Die Sockelplatte hat Aussparungen für den Qi-Empfänger. Vorn das Loch für ein Kabel zum alternativ verwendbaren Steckernetzteil.



**15** Die Ladeplatte mit Netzteil: ein Kabelbinder dient hier als Zugentlastung.

## 16 Over The Air

Mittels OTA kann die Firmware von der Arduino-IDE aus ohne Kabelverbindung per WLAN zum Controller-Board (Arduino oder ESP) übertragen werden. Nützlich ist das, wenn das Gerät mit dem Board unzugänglich zum Beispiel im Freien montiert ist oder es erst aufwändig zerlegt werden müsste, um an die USB-Buchse zu gelangen.

Um ein Board OTA-fähig zu machen, sind nur die zusätzliche Software-Bibliothek *ArduinoOTA* (in der IDE enthalten) sowie einige Zeilen Programmcode erforderlich.

Im Setup des Programms müssen diese Zeilen hinter dem Starten der WLAN-Verbindung stehen:

```
ArduinoOTA.onStart([]() {
  String type;
  if (ArduinoOTA.getCommand() == U_FLASH) {
    type = "sketch";
  } else { // U_FS
    type = "filesystem";
  }
  Serial.println("Start updating " + type);
});
ArduinoOTA.onEnd([]() {
  Serial.println("\nEnd");
});
ArduinoOTA.onProgress([](unsigned int progress,
unsigned int total) {
  Serial.printf("Progress: %u%%\r", (progress /
(total / 100)));
});
ArduinoOTA.onError([](ota_error_t error) {
```

```
  Serial.printf("Error[%u]: ", error);
  if (error == OTA_AUTH_ERROR) Serial.println("Auth
Failed");
  else if (error == OTA_BEGIN_ERROR)
Serial.println("Begin Failed");
  else if (error == OTA_CONNECT_ERROR)
Serial.println("Connect Failed");
  else if (error == OTA_RECEIVE_ERROR)
Serial.println("Receive Failed");
  else if (error == OTA_END_ERROR)
Serial.println("End Failed");
  });
  ArduinoOTA.begin();
```

Zu Beginn der Loop-Schleife ist dann noch

```
ArduinoOTA.handle();
```

notwendig. Das Programm mit diesen Zeilen muss einmal per USB-Kabel aufs Board überspielt werden. Danach ist das OTA-Update möglich. Statt eines COM-Ports wählt man dann aber die OTA-Schnittstelle.

MMU: "32KB cache + 32KB IRAM (balanced)"  
 Non-32-Bit Access: "Use pgm\_read macros for  
 Port: "esp8266-1673c4 at 192.168.10.69"

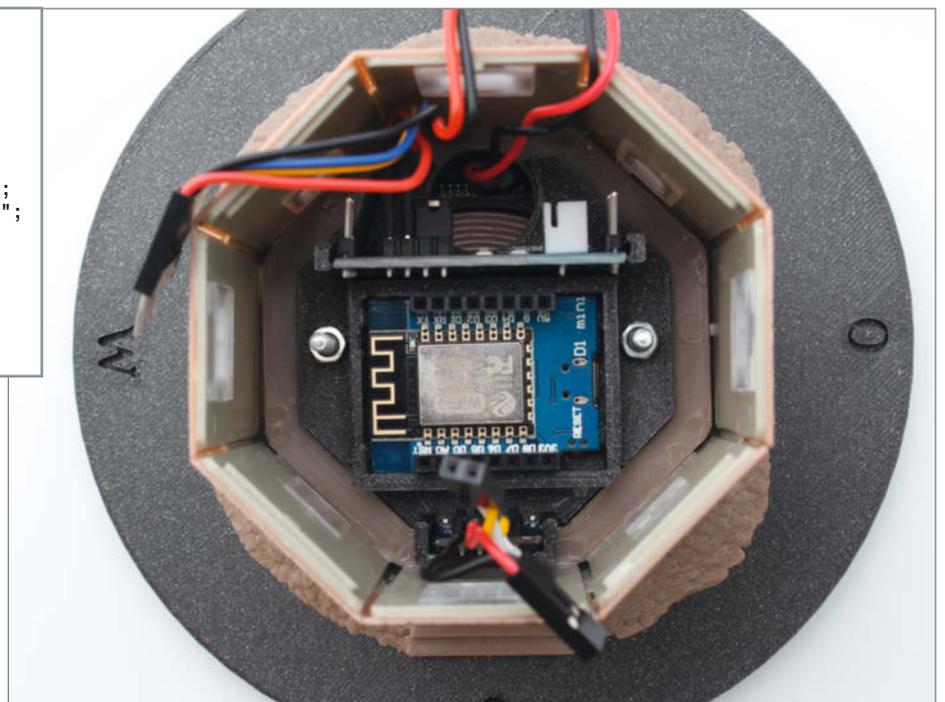
Bei einigen Windows-Versionen ist zusätzlich noch die Installation von Python notwendig.

## 17 Anzupassende Zeilen im Mühlenprogramm

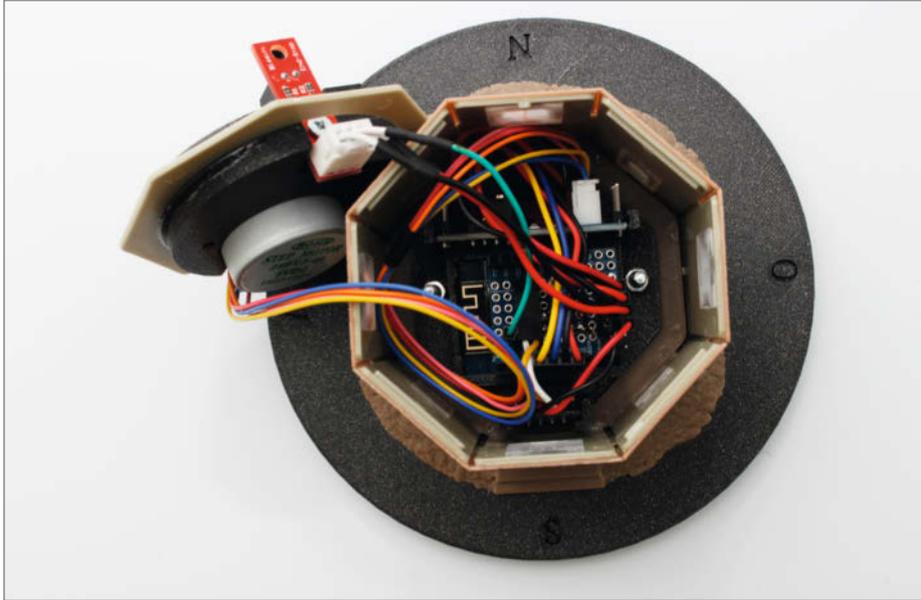
```
const char* ssid = "WLAN-Name";
const char* password = "WLAN-Passwort";
String openWeatherMapApiKey = "API-Key";
String city = "Hannover";
String countryCode = "DE";
```

Geben Sie nun einen kleinen Tropfen 2-Komponenten-Kleber auf die Oberseite der Schrittmotor-Achse. Stecken Sie dann die Achse auf den Schrittmotor und drehen Sie sie vorsichtig, bis der kleine Anschlag an der Achse im Schlitz der Lichtschranke liegt **22**. Den Stecker des Kabels stecken Sie in das Gegenstück vom Verbindungskabel aus dem Erdgeschoss und verstauen die Steckverbindung im Kabelschacht.

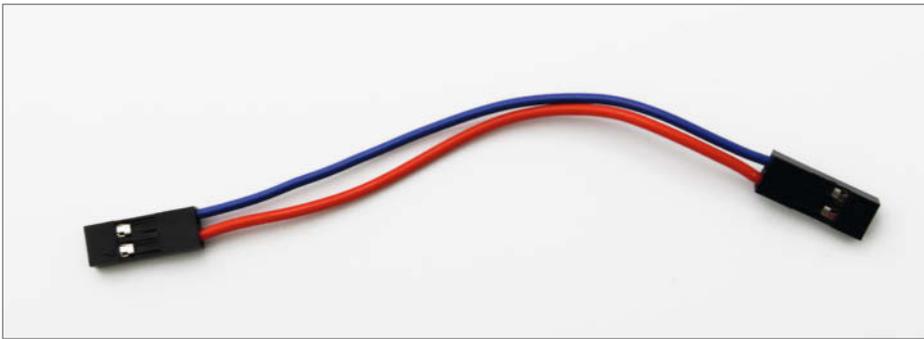
Noch bevor der Kleber aushärtet, setzen Sie nun das Mühlendach mit dem Achslager



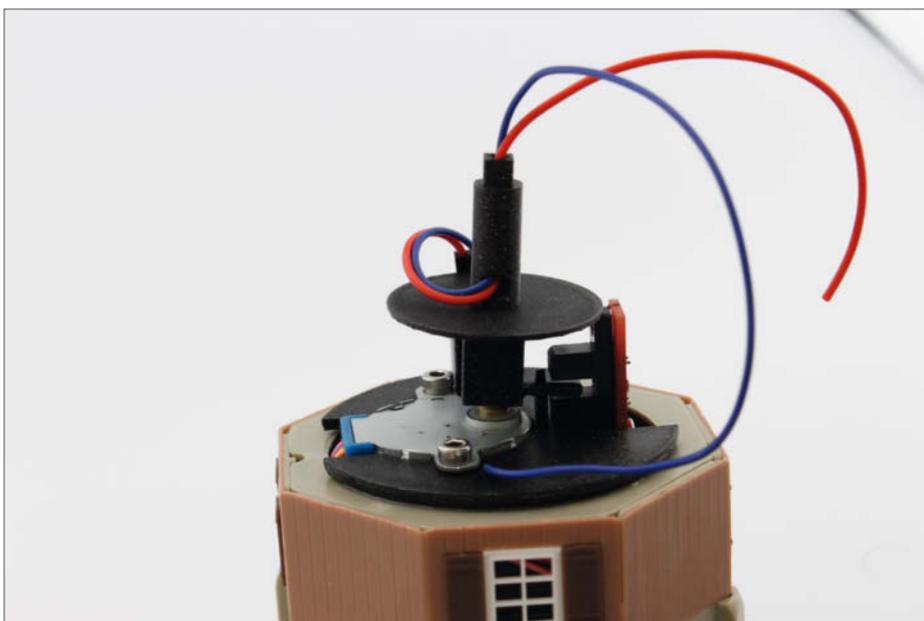
**18** Beim Anschrauben geht es sehr eng zu.



19 Das Aufstecken des Motorkabels erfordert etwas Geschick.



20 Mit diesem Kabel gelangt der Strom für den Flügelmotor in die erste Mühlenetage.



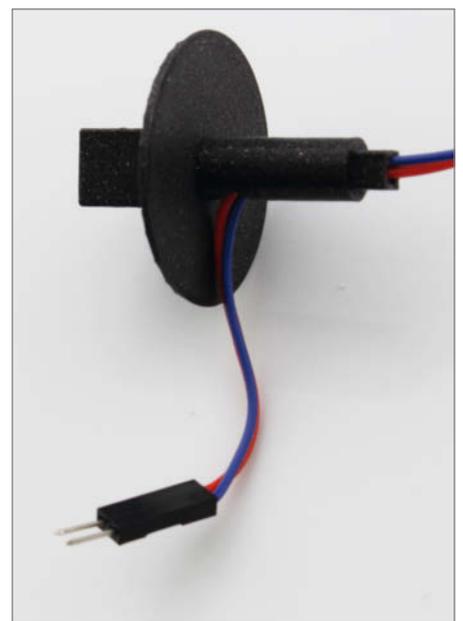
22 Das Kabel für den Flügelmotor sollte sich halb um die Achse wickeln und noch etwas Luft haben, damit die Achse sich später frei drehen kann.

auf. Dabei müssen Sie die Achse vorsichtig durch das Kugellager führen. Sie sollte eng darin sitzen. Falls sie infolge von Drucktoleranzen nicht hineinpasst, wenden Sie keine Gewalt an, sondern bearbeiten Sie die Achse etwas mit Schmirgelpapier. Der dicke Teil der Achse sollte bündig mit der oberen Kante des Lagers abschließen. Nur der Vierkant mit dem Kabel darf hinausragen. In der Zeit, die der Kleber zum Aushärten braucht, kleben Sie den Motorhalter in den Kappenboden und den Boden in den Rahmen für die Kappe, der dem Bausatz beiliegt. Außerdem spendieren Sie dem Motor kurze Litzen mit Crimp-Steckern 23.

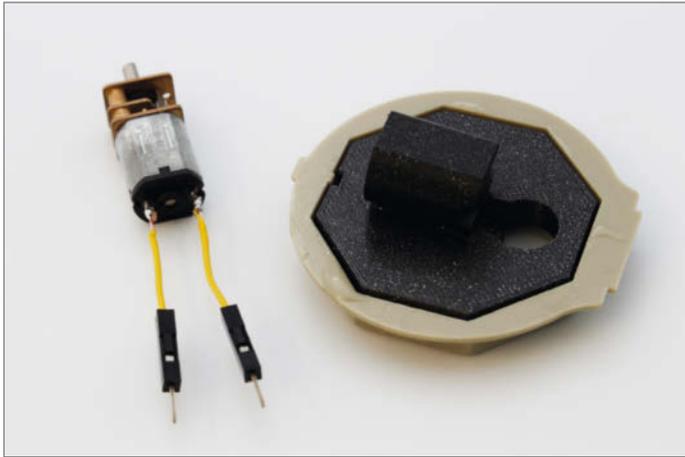
Fädeln Sie nun die beiden oben aus der Achse herausragenden Litzen durch den Kappenboden und stecken Sie den Kappenboden auf die Achse. Verpassen Sie den beiden Litzenenden die entsprechenden Gegenstücke zu den Motorsteckern.

Jetzt kommt der erste Test: Versorgen Sie die Mühle mit Strom (entweder auf die Qi-Plattform stellen oder das 5V-Steckernetzteil einstecken). Die Mühle sucht als Erstes ihren Nullpunkt, das heißt, sie stellt die Flügelachse in Richtung Norden. Falls das bei Ihrer Mühle nicht stimmt, unterbrechen Sie den Strom, nehmen Sie den Kappenboden noch einmal ab und setzen ihn entsprechend versetzt wieder auf 24.

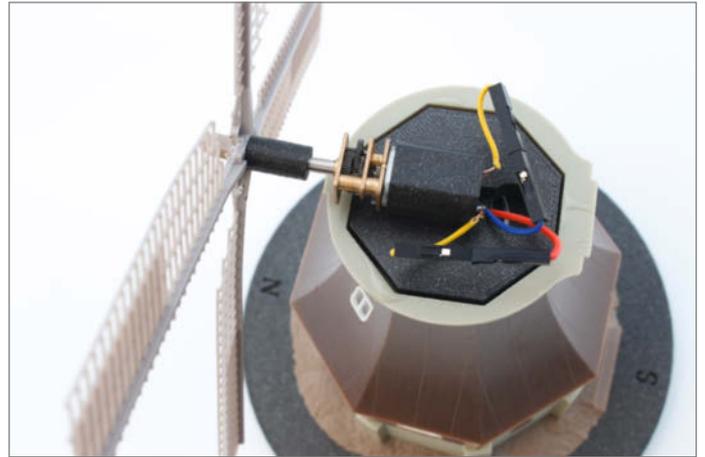
Sitzt der Kappenboden richtig, geben Sie nochmals Strom auf die Mühle. Jetzt läuft einmal das volle Programm ab: Die Mühle richtet sich zunächst Richtung Nord aus. Sobald sie Wetterdaten empfangen hat, dreht sie die Flügelachse in die Windrich-



21 So sollte das Kabel durch die Achse der Mühlenkappe verlaufen.



23 Motorhalterung und der verkabelte Motor: Hier passen nur Motoren mit den Übersetzungen 1:50, 1:150 und 1:298.



24 Beim Test muss die Flügelachse nach Norden zeigen, erkennbar an der Marke auf der Sockelplatte.

Die Flügel beginnen sich ab Windstärke 1 zu drehen. Je höher die Windstärke, umso schneller rotieren die Flügel. Kontrollieren Sie dann noch, ob die Wetterdaten (Temperatur, Luftfeuchte und -druck sowie Windstärke und -richtung) auf dem Display angezeigt werden (siehe Titelbild des Artikels). Falls

nicht, ist wahrscheinlich im Inneren der Mühle ein Kabel abgerutscht oder falsch aufgesteckt.

Wenn alles richtig arbeitet, können Sie die Teile endgültig verkleben, die graue Kappe aufsetzen und sich künftig einer echten Windmühle erfreuen! Im Video zum Artikel

sehen Sie, wie sie arbeitet und auf die eigene Wetterstation reagiert. Übrigens spricht auch nichts dagegen, sie in eine Modelleisenbahn-Anlage zu integrieren. Vielleicht schicken Sie uns dann ja ein Bild davon, das wir gerne in einem der nächsten Hefte oder online zeigen. —hgb

*Mac & i Wissen erfahren*

# Apple School Manager

Der Apple School Manager ist essenziell beim Einsatz von iPads im Bildungsbereich. Unser **Webinar** zeigt Ihnen, wie Sie den Apple School Manager gratis beantragen und registrieren. Im weiterführenden **Workshop** lernen Sie dessen Einrichtung und Bedienung.

Jetzt  
Tickets  
sichern!

**WEBINAR**

Apple School Manager gratis  
beantragen und registrieren

05.10. und 29.11.2022

[mac-and-i.de/webinar-1](https://mac-and-i.de/webinar-1)

**WORKSHOP**

Apple School Manager beherrschen

12.-13.10. und 07.-08.12.2022

[mac-and-i.de/workshop-1](https://mac-and-i.de/workshop-1)

# Symmetrische Signalübertragung

„Symmetrie ist die Ästhetik des kleinen Mannes“ – ein spöttischer Ausspruch in Künstler- und Architektenkreisen. Allerdings: Gerade im hochwertigen Tonstudio ist die symmetrische Technik zu Hause. Was ist also das Geheimnis dahinter, und wie kann man solche Audio-Verbindungen möglichst einfach nutzen?

von Thomas Hirschberg



**E**s könnte alles so schön sein, wenn nicht immer jemand stören würde. Der Signalfrieden wird beeinträchtigt durch Störquellen wie Schaltnetzteile, PCs, Laptops und Tablets, Smartphones, Licht-Dimmer sowie durch Signale von anderen Kabeln in der Nähe (Übersprechen). Somit sind alle Geräte, die getaktete und/oder hochfrequente Elemente enthalten, die potenziellen Feinde. Diese senden hochfrequente elektromagnetische (kurz EM-) Wellen aus, die per kapazitiver (E) oder induktiver (M) Einstrahlung das Nutzsignal im Kabel überlagern können. Was tun? Abstände vergrößern (geht nicht immer), Kabel anders verlegen (hilft manchmal), Abschirmung verbessern mit ferromagnetischem Material, d.h. Eisen oder speziellem Mu-Metall (Achtung: Traglast des Fußbodens beachten).

### Bitte nicht stören

Im Studiobetrieb (und natürlich auch bei Live-Konzerten) ist vor allem eines wichtig: Eine sichere Verbindung zwischen Mikrofonen, Mischpult und weiteren Geräten. Sicher heißt: mechanisch belastbar, rastende Steckverbindung, niederohmiger Kontakt (am besten durch Vergoldung) und elektrisch störungsfreie Übertragung. Das letztgenannte Feature ist bei längeren Kabeln nur mit symmetrischer Übertragungstechnik erreichbar.

Eine typische unsymmetrische Verbindung ist das Cinch-Kabel zwischen CD-Player und Verstärker. Die äußere Abschirmung (meist ein dünnes Cu-Drahtgeflecht) ist nicht perfekt und wirkt gegen äußere M-Felder überhaupt nicht. Ein Teil des äußeren Störfeldes dringt bis zum Innenleiter durch und addiert sich als Störspannung  $U_{stör}$  zum Nutzsignal  $U_{sig}$ . Der Verstärkereingang  $U_{in}$  kann zwischen Störsignal und Nutzsignal nicht unterscheiden. Die Störung wird wie das Nutzsignal verstärkt und wiedergegeben: Einfach geht hier einfach nicht gut.

### Doppelt hält besser

Das Kennzeichen einer symmetrischen Verbindung ist, dass neben dem Originalsignal  $U_{sig}$  auch das invertierte Signal  $-U_{sig}$  übertragen wird. Dieses muss im sendenden Gerät erzeugt werden, entweder durch einen zusätzlichen invertierenden Ausgangsverstärker oder durch einen Ausgangsübertrager. Dieser Trafo ist bei hohem Qualitätsanspruch sehr teuer, bietet aber konkurrenzlos das Feature einer erdfreien (galvanisch getrennten) Verbindung zwischen Quelle und Empfänger. Brummschleifen durch Masseverkopplungen werden somit sicher vermieden.

Das symmetrische Kabel hat demnach zwei Innenleiter, die sehr dicht aneinander geführt werden, häufig sogar miteinander verdreht sind. Kommt nun eine EM-Störung von außen, durchdringen die Felder die Abschirmung und

## Kurzinfo

- » Symmetrische vs. asymmetrische Signalübertragung
- » Aktive Symmetrierung von unsymmetrischen Analogausgängen
- » Passive Lösung direkt im Kabel für 10 Cent

### Checkliste

-  **Zeitaufwand:**  
30 Minuten bis 3 Stunden
-  **Kosten:**  
10 Cent bis 10 Euro (ohne Kabel)
-  **Messen:**  
Multimeter, besser AC-Millivoltmeter
-  **Löten:**  
Einfache Verdrahtungsarbeiten

### Material

- » XLR- oder Stereo-Klinkenkabel
- » Bauteile nach Schaltplan (aktive Lösung) oder
- » Mono-Klinkenstecker 6,3mm
- » Widerstand 1/10 Watt Wert siehe Text

↓ Alles zum Artikel im Web unter [make-magazin.de/xeq5](http://make-magazin.de/xeq5)



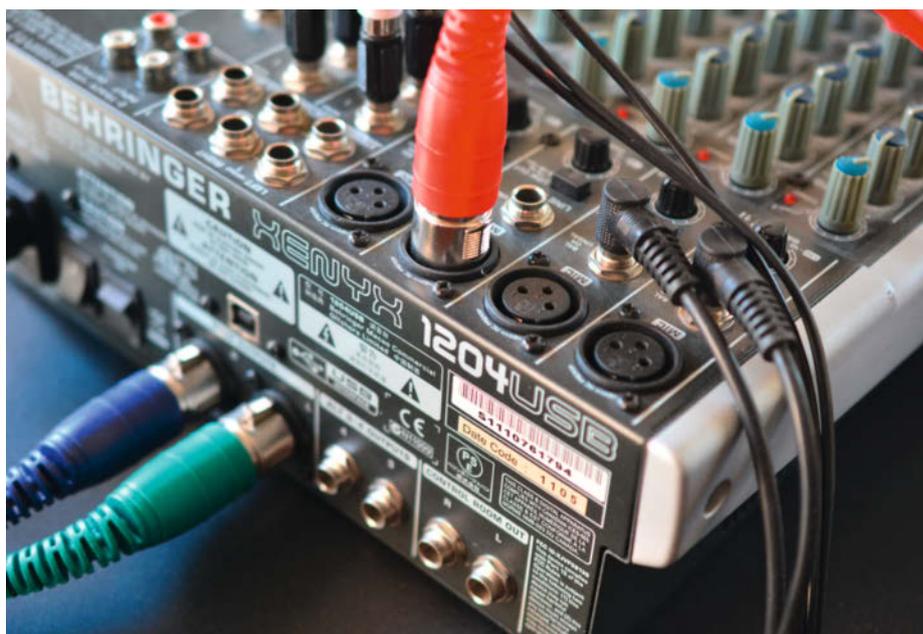
erzeugen wie oben die Störspannung  $U_{stör}$ , aber es besteht die berechtigte Hoffnung, dass diese sich auf beide Innenleiter identisch auswirkt (Bedingung dafür siehe unten).

Im Empfänger (also das Mischpult oder ein Verstärker) muss zwingend ein Differenzverstärker am (symmetrischen) Eingang vorhanden sein, der das invertierte Signal wiederum invertiert und dann beide Signale addiert. Es ergibt sich:

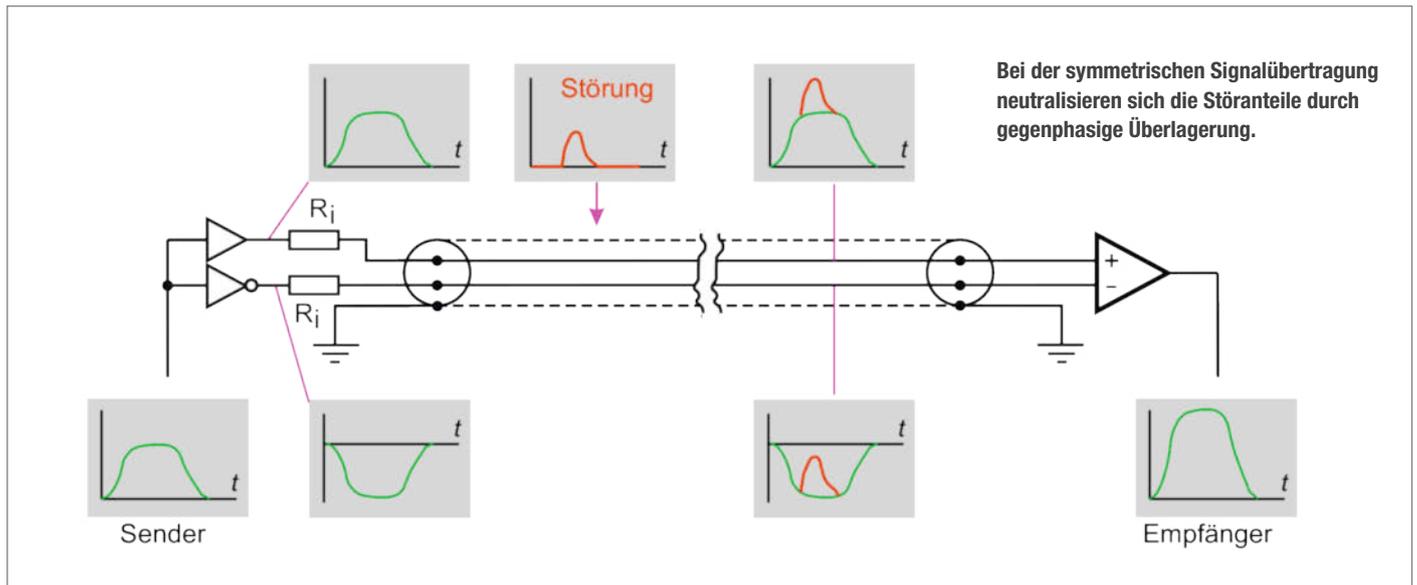
- Auf der Leitung des Originalsignals  $U_1 = U_{sig} + U_{stör}$
- Auf der Leitung des inversen Signals  $U_2 = -U_{sig} + U_{stör}$

- Nach der Differenzbildung:  
 $U_{in} = U_1 - U_2 = U_{sig} + U_{stör} - (-U_{sig} + U_{stör}) = 2 U_{sig}$   
Die Störspannung verschwindet und die Nutzspannung verdoppelt sich, ein willkommener Effekt, gerade wenn Mikrofonsignale im mV-Bereich übertragen werden.

Das typische professionelle Verbindungskabel für symmetrische Technik besitzt beidseitig dreipolige XLR-Steckverbinder. Mit Hilfe dieser Technik ist es möglich, selbst schwache Mikrofonsignale über weite Strecken (z.B. 30m) störungsfrei zu übertragen und anschließend in Studioqualität weiterzuverarbeiten.



Symmetrische Ein- und Ausgänge findet man auch an preiswerten Mischpulten. Passende XLR-Ausgänge an Instrumenten sind hingegen, wenn überhaupt, nur in der Oberklasse zu finden.



Bedingung: Die Kompensation des Störsignals funktioniert nur dann optimal, wenn die Leitung mit dem invertierten Signal an Start- und Endpunkt jeweils dieselbe Impedanz „sieht“ wie die Leitung mit dem Originalsignal. Konkret: Der Eingangswiderstand des Differenzverstärkers (im Mixer) muss für beide Eingänge (+ und -) identisch sein. Aber auch die Innenwiderstände der Signalquelle müssen für beide Signalleitungen denselben Wert haben, sonst wird auf den Leitungen nicht der gleiche (zu kompensierende) Störanteil eingekoppelt.

Da die Leiterführung im Studio-Gerät oder Instrument meist asymmetrisch ausgeführt

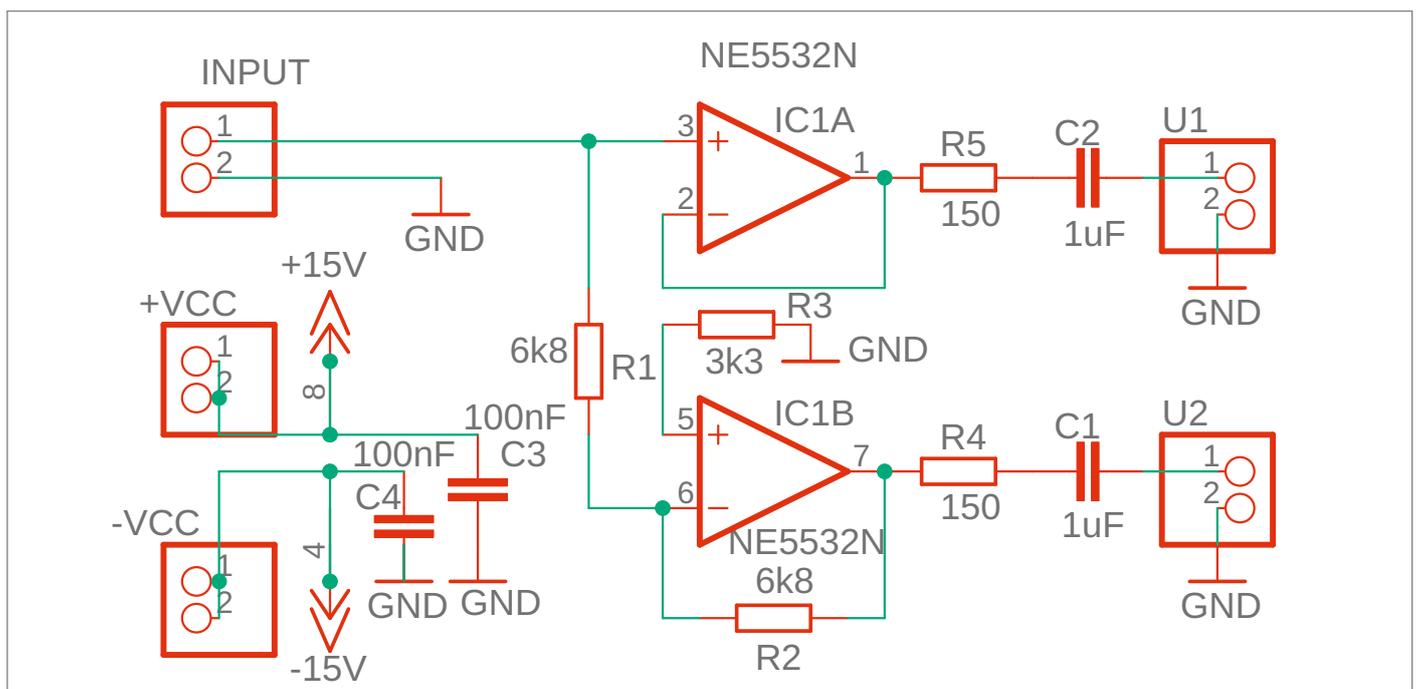
ist, wechselt man erst am Ausgang auf ein symmetrisches Signal – dort, wo es darauf ankommt. Dafür reicht eine einfache Inverterstufe. Symmetrische Ausgänge findet man selbst an preiswerten Mischpulten und Mikrofonen, aber leider nicht an den meisten Keyboards, Synthesizern und E-Pianos – schade eigentlich.

### Tuning gegen Störungen

Der gut betuchte Profi-Keyboarder beschafft sich am besten etliche sogenannte DI-Boxen (direct injection) bester Qualität, eine für je-

den Ausgang eines jeden Instruments. Diese symmetrieren das unsymmetrische Tonsignal (Mono-Klinke auf Stereo Klinke oder XLR), entweder elektronisch mit einem Doppel-OpAmp (der natürlich eine Stromversorgung benötigt), oder per Übertrager (ein kleiner Trafo). Bei allzu billigen Übertragern ist die Qualität allerdings fragwürdig.

Der DI-Eingang wird mit einem kurzen Patchkabel mit dem unsymmetrischen Ausgang des Keyboards verbunden. Die Stromversorgung der DI-Box kann mit externem Netzteil oder Batterie erfolgen, besonders elegant ist aber die Versorgung durch das angeschlossene



Vorschlag für eine symmetrierende Ausgangsstufe: Am Steckverbinder U2 liegt das invertierte Eingangssignal an, U1 führt das gepufferte, aber nicht invertierte Signal.

Mischpult per Phantomspeisung (48 Volt bei XLR-Verbindung), so vorhanden.

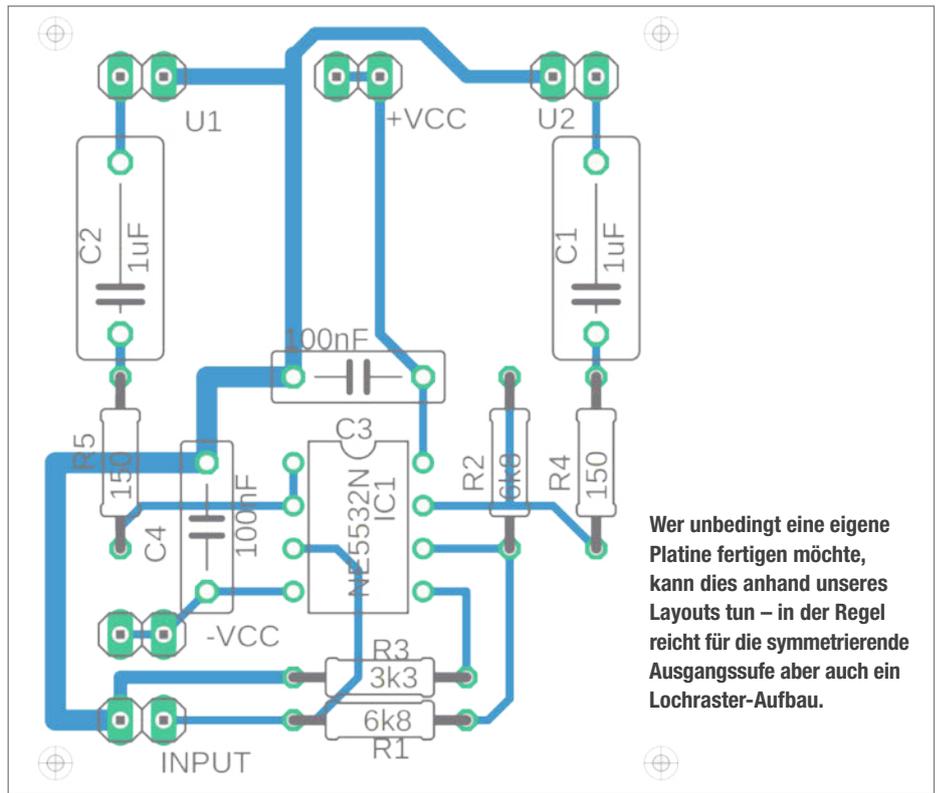
Die beste Lösung ist aber, gleich eine Symmetrierstufe und professionelle XLR-Steckverbinder zusätzlich in das Instrument einzubauen, wenn genügend Platz ist. Ersatzweise kann man auch Stereo-Klinkenbuchsen für eine symmetrische Übertragung missbrauchen: An den Ringkontakt legt man das invertierte Signal, an den Spitzenkontakt das nicht-invertierte, der Masseanschluss bleibt. Am anderen Ende des (zweipolig abgeschirmten) Kabels wird dann ein XLR-Stecker (male) für das Mischpult montiert:

- Pin 1 = Masse, Gehäuse = Schirm (Masse)
- Pin 2 = Originalsignal (plus, hot)
- Pin 3 = invertiertes Signal (minus, cold)

Das hat den Vorteil, dass der Umbau auch notfalls mit asymmetrischen Verbindungen (Mono-Klinkenkabel) kompatibel bleibt, das invertierte Signal wird dann schlicht kurzgeschlossen.

### Schaltungsentwurf

Egal ob XLR oder „Stereo“-Klinke: Man benötigt eine zusätzliche Ausgangsstufe, die das invertierte Signal bereitstellt. Die notwendige



Wer unbedingt eine eigene Platine fertigen möchte, kann dies anhand unseres Layouts tun – in der Regel reicht für die symmetrierende Ausgangsstufe aber auch ein Lochraster-Aufbau.



# DIE NEUE LERNPLATTFORM FÜR IT-PROFESSIONALS

Wir machen IT-Weiterbildung digital

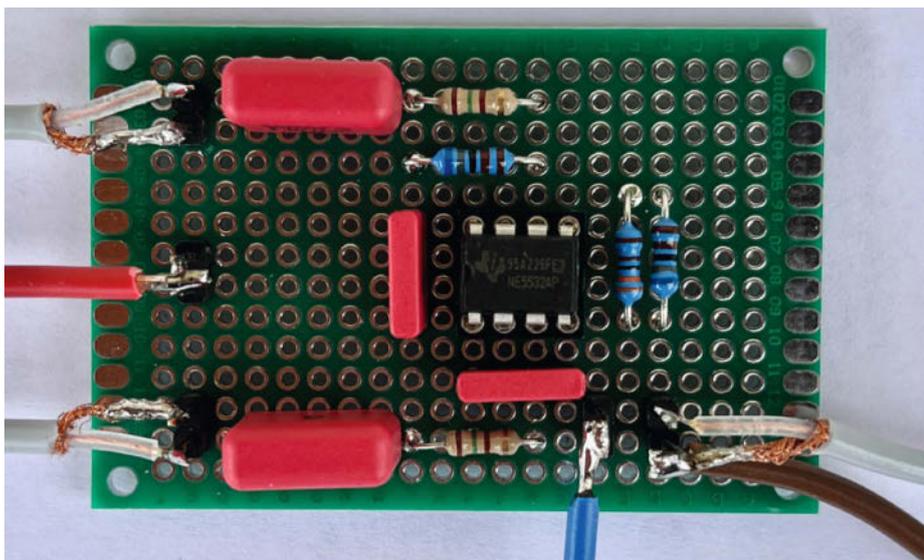
JETZT  
KOSTENLOS  
TESTEN

**Das erwartet dich:**

- Über 100 Online-Trainings und 80 Online-Kurse
- Die wichtigsten IT-Themen für heute und morgen
- Erfahrene IT-Experten
- Individuelle Lernumgebung
- Übungsaufgaben und Wissenstests

Hier geht's zu deiner Weiterbildung: [heise-academy.de](https://www.heise-academy.de)





Lochraster-Platine mit unserer Symmetrier-Schaltung: Die beiden grauen Kabel links führen das symmetrische Audio-Signal.

Schaltung mit einem Doppel-OP ist recht einfach und eine Platine muss nicht unbedingt sein, das Ganze kann man mit etwas Geschick auch auf einer kleinen Lochrasterplatine aufbauen. Einzig die geräteinterne Spannungsversorgung ( $\pm 12V$  oder  $\pm 15V$  und Masse) muss man identifizieren und anzapfen. Der ursprüngliche Ausgang des Keyboards, den man intern an der alten Klinkenbuchse abgreift, wird an den Eingang der Symmetrier-schaltung gelegt. Die zwei neuen Ausgänge kommen an die neue (ggf. zusätzliche) Buchse.

Verwendet wird der bekannte Doppel-OpAmp NE5532 in Bipolartechnik. Er ist als Ausgangsstufe robuster als CMOS-Versionen, kann recht viel Strom liefern, ist rauscharm und angeblich kurzschlussfest. IC1A arbeitet als Spannungsfolger mit der Verstärkung 1.

IC1B ist ein invertierender Verstärker mit der Verstärkung  $V = -R_2/R_1 = -1$ . Der Wert für  $R_3$  ergibt sich aus  $R_2$  parallel zu  $R_1$  – er sorgt lediglich für eine geringe Offset-Spannung am Ausgang. Die OpAmp-Profis wissen Bescheid, für alle anderen eine kurze Erklärung: Durch den unvermeidlichen Eingangsstrom bei bipolaren OpAmps würden unterschiedliche Quellenwiderstände an seinen Eingängen sonst zu einer leichten Verschiebung der Ruhe-Gleichspannung am Ausgang führen.

Der Eingangswiderstand beträgt 6,8 kOhm ( $R_i$ ), niederohmiger ist nicht zu empfehlen, da ansonsten eventuell tiefe Frequenzen durch den Ausgangskondensator des Instruments gedämpft werden. Der Ausgangswiderstand ist auf beiden Ausgängen  $U_1$  und  $U_2$  identisch mit 150 Ohm bemessen, bedingt durch die

Schutzwiderstände  $R_4$  und  $R_5$ ,  $C_1$  und  $C_2$  sind recht groß, sorgen aber dafür, dass bei einem angenommenen Eingangswiderstand im Mischpult von mindestens 10 kOhm auch die tiefsten Bässe durchkommen (-3dB bei 16Hz). Eine XLR-Buchse (male) wird dann wie folgt mit der Schaltung verbunden:

- Pin 1 an Gnd (Masse)
- Pin 2 an Ausgang  $U_1$
- Pin 3 an Ausgang  $U_2$

### Minimalisten-Lösung

Für alle, denen die vorgestellten Maßnahmen zu teuer oder zu aufwändig sind oder die einen Garantieverlust fürchten, kommt jetzt die supersmarteste Lösung: eine pseudo-symmetrische Leitung. Die Kosten liegen unter 20 Cent (ohne Kabel) und die Störunterdrückung ist fast so gut wie bei der „echten“ symmetrischen Übertragung.

Angenommen, in der beschriebenen Symmetrierschaltung wäre die Ausgangsspannung  $U_2 = 0$ : Man entfernt dazu in Gedanken IC1B und legt seinen Ausgang (hier Pin 7) auf Masse. Die obigen Berechnungsformeln sehen nun so aus:

- Auf der Leitung des Originalsignals

$$U_1 = U_{sig} + U_{stör}$$

- Auf der Leitung des inversen Signals

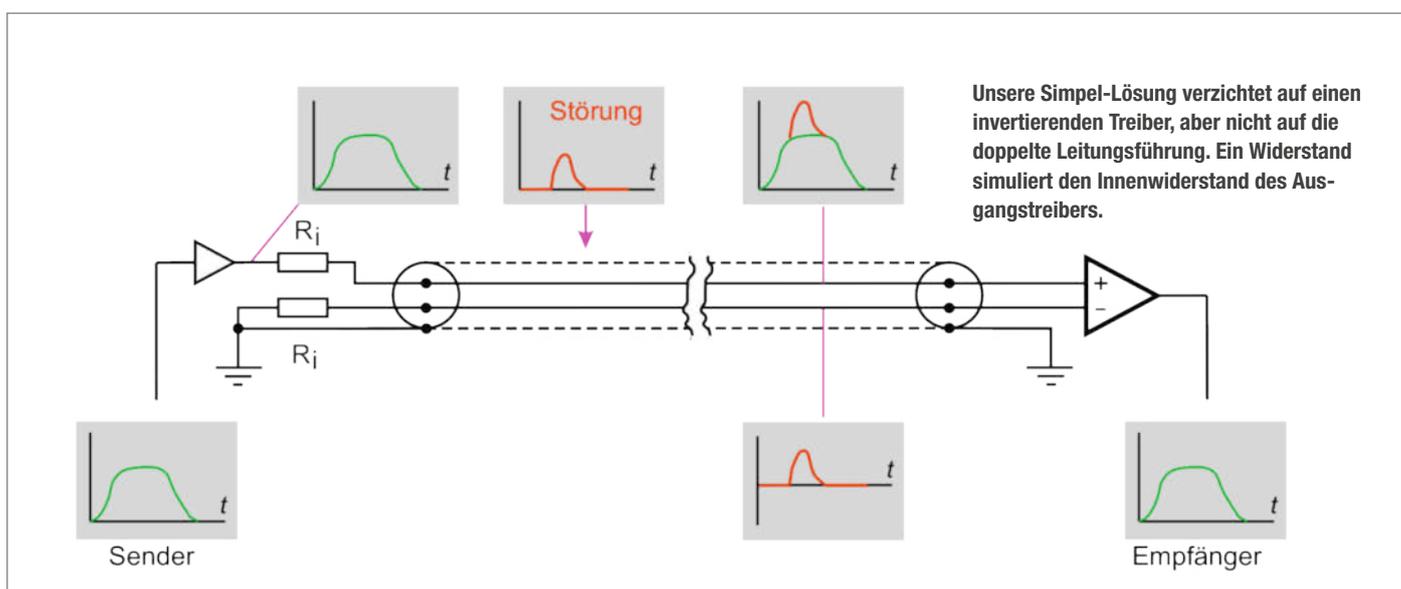
$$U_2 = 0 + U_{stör}$$

- Nach der Differenzbildung:

$$U_{in} = U_1 - U_2 = U_{sig} + U_{stör} - (0 + U_{stör}) = U_{sig}$$

Wir sehen: Lediglich der Faktor 2 vor  $U_{sig}$  fehlt, die Störauslöschung funktioniert aber weiterhin perfekt. Bedingung: Die Leitung für das invertierte Signal muss denselben Quellenwiderstand ( $R_4$ ) „sehen“ wie die andere Leitung ( $R_5$ ). Und diesen  $R_4$  lötet man direkt in den Klinkenstecker:

- Basis ist ein hochwertiges XLR-Kabel (für Stereo natürlich zwei).



- An einer Seite wird der XLR-Stecker entfernt und stattdessen ein Mono-Klinkenstecker montiert.
- Masse (Abschirmung) und roter Innenleiter (plus/hot) werden wie üblich angelötet.
- Der andere Innenleiter wird im Stecker über einen Widerstand (1/10Watt, Wert folgt gleich) mit Masse verbunden.
- Das Ganze mit einem Schnipsel Schrumpfschlauch isolieren.

Um den einzulötenden Widerstand bemessen zu können, muss der Innenwiderstand der geräteeigenen Ausgangsstufe bekannt sein. Dieser befindet sich aber irgendwo in unserem Gerät oder Keyboard. Mit einem Ohmmeter quasi rektal in den Ausgang hineinzumessen geht schief – eine rein ohmsche Messung berücksichtigt nicht einen eventuell vorhandenen Ausgangskondensator und den Innenwiderstand der Halbleiter, die ja nur bei eingeschaltetem Gerät aktiv sind.

Lieber einmal nachdenken: Die Spannung einer Quelle mit dem (hier noch unbekanntem) Innenwiderstand  $R_i$  bricht dann auf die Hälfte ein, wenn der äußere Lastwiderstand  $R_L$  gleich  $R_i$  ist. Das bedeutet ganz praktisch:

- Keyboard einschalten und möglichst gleichmäßigen Ton ohne Modulation einstellen, etwa einen nicht-perkussiven Synthi-Klang. Eine mittlere Taste drücken und blockieren.
- Mono-Klinkenstecker einstecken und die Leerlauf-Ausgangsspannung messen (vorzugsweise Audio-Millivoltmeter oder Multimeter im AC-Bereich).
- Jetzt einen veränderbaren Widerstand (Poti 1... 10kOhm) oder eine Widerstandsdekade an den Ausgang und an Masse anschließen.
- Den Widerstand so lange verändern, bis sich die Ausgangsspannung exakt halbiert hat, bezogen auf die Leerlaufspannung.
- Stecker herausziehen und diesen Widerstandswert mit einem Ohmmeter bestimmen.

Der ermittelte Wert dürfte zwischen 100 und 1000 Ohm liegen, man wählt dann einen Widerstand aus der E24-Reihe, der diesem Wert am nächsten kommt. Die Praxis hat gezeigt, dass auch bei kleinen Abweichungen die Störunterdrückung noch gut funktioniert. Hauptsache ist, dass die zweite Leitung am Ring des Klinkensteckers nicht offen bleibt ( $R = \infty$ ) oder direkt an Masse liegt ( $R = 0$ ). Wer also gar nicht messen will oder kann, lötet 220 Ohm oder 330 Ohm in den Stecker und hangelt sich dann schrittweise an die beste Störunterdrückung heran.

Ein kleiner Nachteil unserer Sempel-Störunterdrückung sei hier nicht verschwiegen: Das so angefertigte Kabel passt nur zu dem Instrument, für das der Widerstand selektiert wurde. Praktischerweise versieht man es mit einem Label, das auf die besondere Verwendung hinweist. —cm



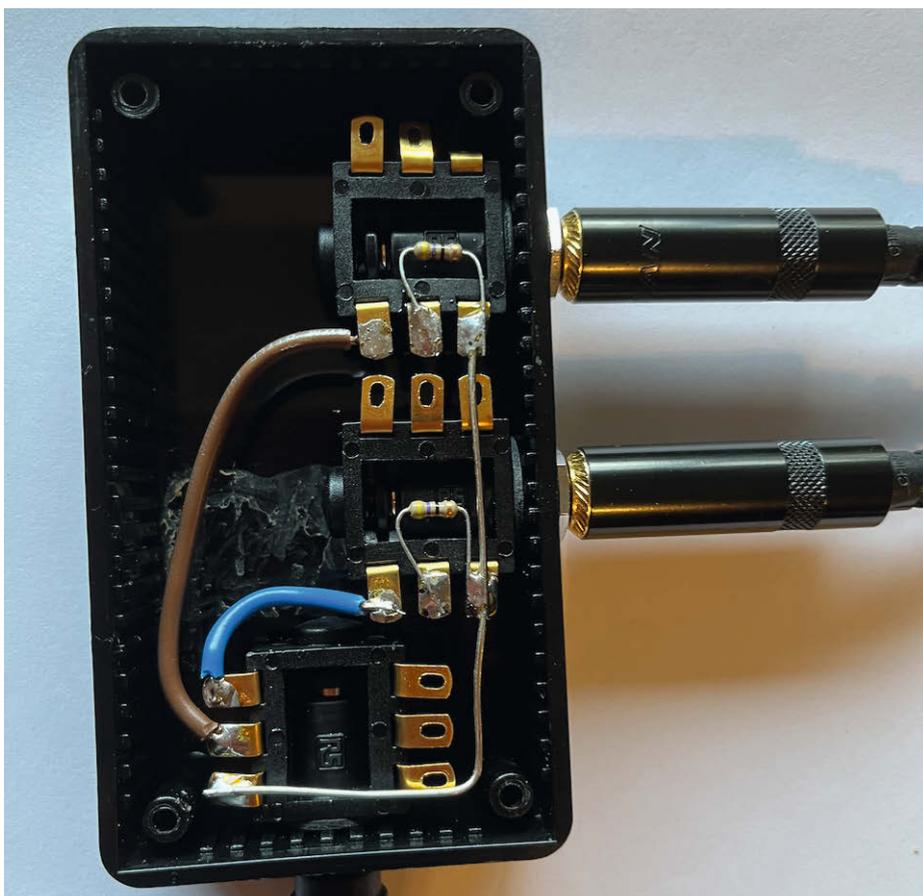
Stecker montieren: Die „heiße“ Leitung des XLR-Kabels kommt an die Klinkenstecker-Spitze.



Die „kalte“ Leitung (hier blau) wird über den vorher bestimmten Widerstand mit Masse verbunden.



Dann das Ganze mit Schrumpfschlauch isolieren – fertig!



Bei Stereo-Ausgängen, die am Keyboard auf einer Stereo-Klinkenbuchse liegen, verlagert man die Pseudo-Symmetrierschaltung besser in ein kleines Kästchen. Statt der zwei Ausgangs-Klinkenbuchsen rechts können hier auch XLR-Buchsen montiert werden. Die Buchse unten erhält das Stereo-Signal vom Instrument.

# Spiele ohne Logik(-Gatter)

Josef Hesse restauriert alte Spielautomaten. Die interessantesten davon kommen ohne digitale ICs aus, denn sie sind rein elektromechanisch implementiert. Viele dieser alten Ideen verblüffen bis heute und können Anregungen für eigene Projekte geben. Ein Werkstatt-Besuch der ganz eigenen Art.

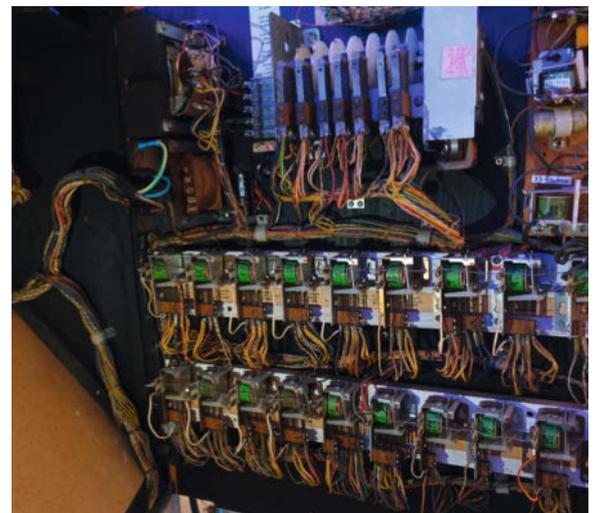
von Clemens Gleich



Eigentlich ist viel Platz in Josef Hesses Halle – eigentlich. Denn viel Platz für weitere restaurierungsbedürftige elektromechanische und elektronische Arcade-Automaten ist auch dort nicht mehr.



Die Technik drinnen muss originalgetreu funktionieren, draußen dürfen die Gebrauchsspuren sichtbar bleiben, das ist Josef Hesses Credo bei der Restauration.



In vor-digitalen Zeiten hatten die Automatenbauer weder Mikrocontroller noch Farbdisplays zur Verfügung, da war elektromechanische Kreativität gefragt.

Die Stimme kommt aus einem Sperrholzschränkchen, auf dem *SEGA Monaco GP* steht: „Ich hab's dir ja gesagt: 90 Prozent der Wiederherstellung besteht meist darin, einfach einmal richtig sauberzumachen.“ Wir haben den Automaten eben mit dem Gabelstapler vom Hänger abgeladen. „Und der zweite Punkt ist: Immer vor dem Anschalten reinschauen, weil sich beim Herumstehen ja Kurzschlussmöglichkeiten gebildet haben können.“

Josef Hesse zieht den Kopf heraus, schließt die Stromversorgung an und siehe da: Es tut sich bereits etwas auf dem Bildschirm ①. Segas Rennspielautomat von 1979 steht als Zeitzeuge für jene Arcade-Ära, in der Spiele noch ohne integrierte Rechenwerke wie CPUs gebaut wurden, damals rein aus Logikgatter-ICs ②. Die Platine nimmt deshalb viel Platz ein, sieht entsprechend uniform aus ③. Die Autos fahren, die Steuerung funktioniert, Geräusche erklingen, doch die Dioden-Segment-Zählwerke funktionieren nicht. „Das muss ich mir noch in Ruhe mal genau anschauen“, sagt Josef.

Josef sammelt Arcade-Automaten, und obwohl er überproportional viel Platz für sie hat, steht schon jede verfügbare Hallen-Ecke voll mit laufenden oder abgeschlossenen Projekten. Er muss sich hier eher zügeln, damit das Hobby eheverträglich bleibt. Denn was anfang mit einigen Adoptionen alter Automaten und Flipper aus Partykellern, wuchs sich mit der Zeit zu einer kleinen Obsession aus. „Ich will die Technik erhalten“, erzählt er. Deshalb interessieren ihn auch solche Restaurationen weniger, bei denen das alte Innenleben durch eine moderne Platine ersetzt wird, die alte Hardware emuliert. Er will die Originaltechnik reparieren. Damit unterscheidet er sich auch von den Retro-Arcade-Fans, bei denen oft die Spiele im Mittelpunkt stehen und für die die verwendete Technik als Zweck dazu zweiträn-

gig ist. Dabei kommen oft Raspis zum Einsatz wie bei der *Bartop Arcade* in Heft 4/20. In diesem Artikel gehen wir aber zeitlich noch weiter auf die andere Seite der Logikgatter-Automaten, zurück zu den ganz alten Geräten, die rein elektromechanisch konstruiert wurden. Manche dieser Automaten waren erstaunlich aufwendig.

### Zurück vor die Fünzfziger

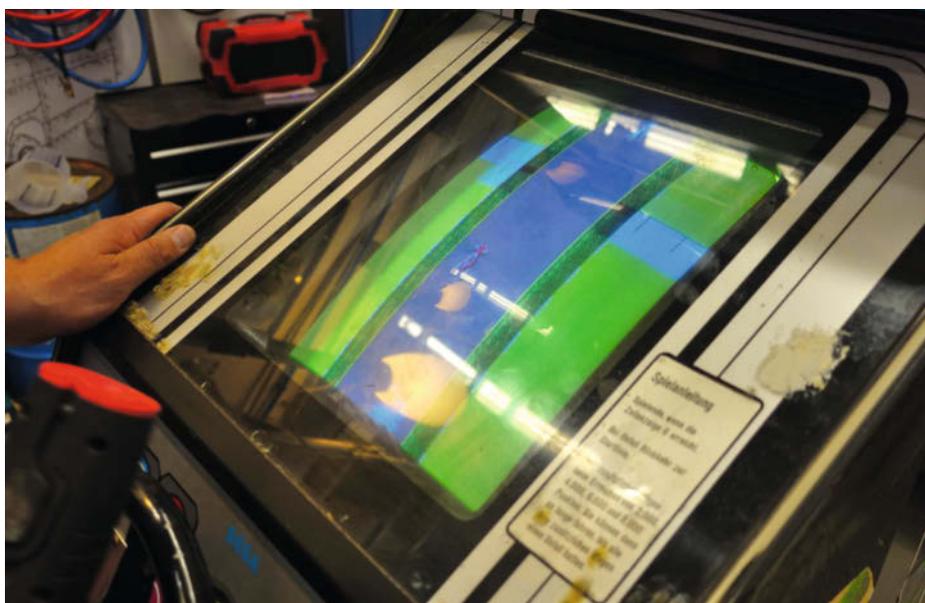
Doch am Anfang meiner Zeitreise durch die Arcade-Technik mit Josef Hesse stehen ganz einfache Spiele. Wir beginnen den Tag mit einem Spielchen auf einem rein mechani-



### Video zum Artikel online

Weil sich die Spielmechanik der Automaten oft in bewegten Bildern besser zeigen lässt, war natürlich die Videokamera beim Besuch bei Josef Hesse mit dabei. Das Ergebnis erreichen Sie über den QR-Code oder indem Sie den folgenden Kurzl ink in die Adresszeile Ihres Browsers tippen:

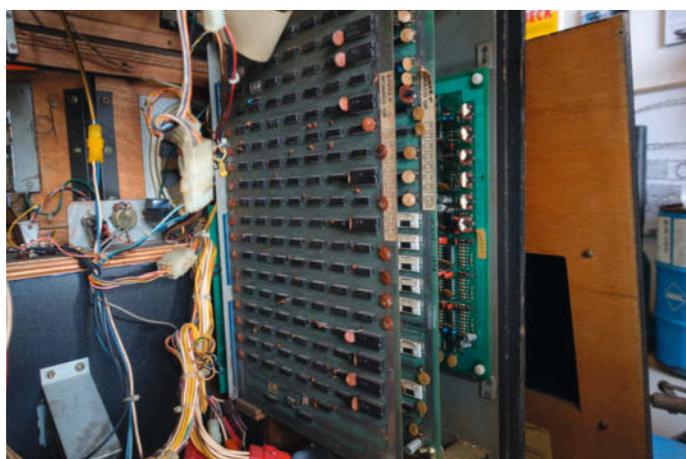
Alles zum Artikel im Web unter [make-magazin.de/xvbb](https://make-magazin.de/xvbb)



① Das erste Anschalten bleibt ein spannender Moment: Beim Sega Monaco GP flitzen immerhin noch Rennautos über den Bildschirm.



② Frisch vom Stapler, rein in die Werkstatt: Der Rennspielautomat stammt aus dem Jahr 1979.



③ Im Inneren des Monaco GP stecken noch Dutzende TTL-Chips, kurz darauf bestimmten Mikroprozessoren die Szene.



4 Ein Kettenantrieb koppelt im Hockeyautomat den Controller mit der Spielfigur ...



5 ... die bei der Ausgabe des Automaten aus den Dreißigerjahren noch aus Gusseisen besteht.



6 Schwerkraft reicht aus: Der Schieber für die Münzen am Hockeyautomat.



7 Die Hockey-Version aus den 50er Jahren hat immerhin schon ein Kabel – für die elektrische Beleuchtung und die Zählwerke für die Punkte.



8 Die Schaltungstechnik im Inneren des Hockeyautomaten funktioniert noch rein analog und elektromechanisch.

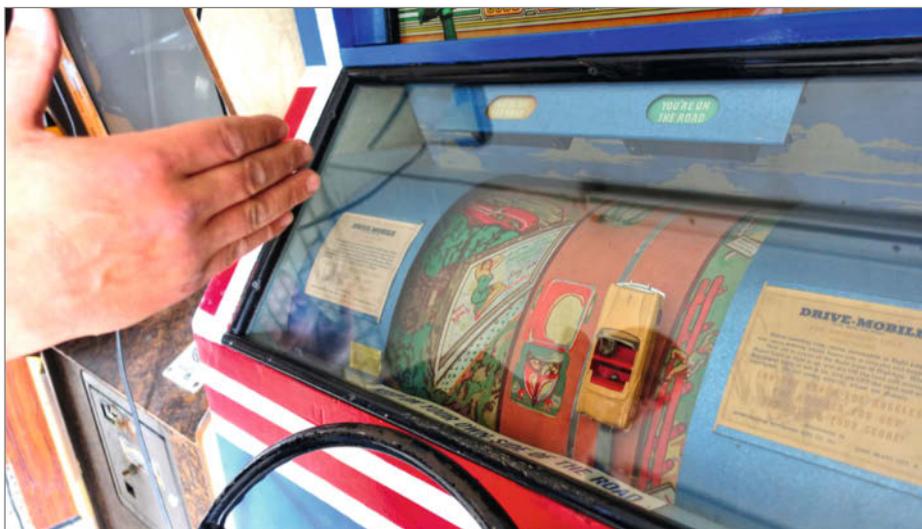
daneben steht jedoch eine schon bereits deutlich komplexere Variante aus den Fünfzigern mit Stromkabel 7. Der Strom versorgt Lampen und Zählwerke mit Energie 8. Das ist noch überschaubar in der Komplexität und eignet sich folglich gut zum Einstieg ins Verständnis.

„Das ist das Schöne an den elektromechanischen Automaten: Du brauchst Geduld, aber wenn du langsam, logisch, Schritt für Schritt herangehst, kannst du das Meiste selbst reparieren“, sagt Josef. Eine weitere schöne Besonderheit: Nach so einem Hineinfuchen in die Funktion, in das „wo könnte der Fehler liegen?“, versteht man oft schon den Großteil aller Funktionen des Geräts. Wir schauen hinüber zu *Drive-Mobile*, das die Firma *International Mutoscope* 1941 erstmals herausbrachte 9. Das Spiel simuliert eine Autoreise durch die USA auf einer großen Rolle. Der Automat gibt eine Zeit vor, in der das Ziel der Reise erreicht werden muss. Bei jeder erreichten Stadt leuchtet eine Lampe auf. Der Punktezüher zählt weiter, solange das kleine Auto auf der Straße fährt und nicht mit Verkehr kollidiert 10. Bei Fahrten neben der Straße brummt es und es gibt keine Punkte.

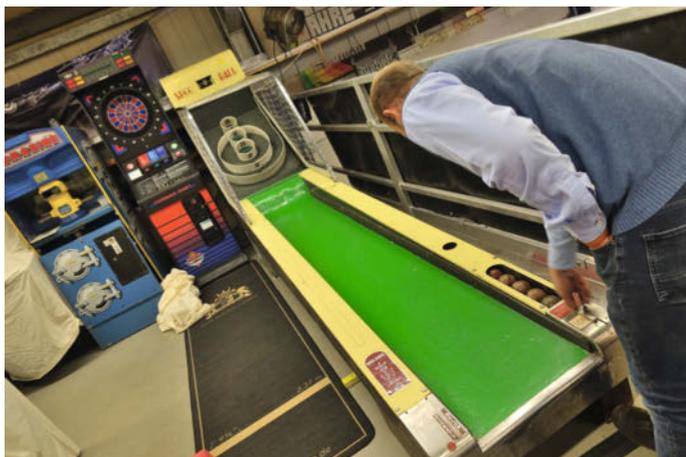


9 Trotz liebevoll bemalter Rolle, auf der die Straße abgespült wird, ist der Fahrsimulator *Drive-Mobile* aus den 40er Jahren spielerisch eher schnell ausgereizt.

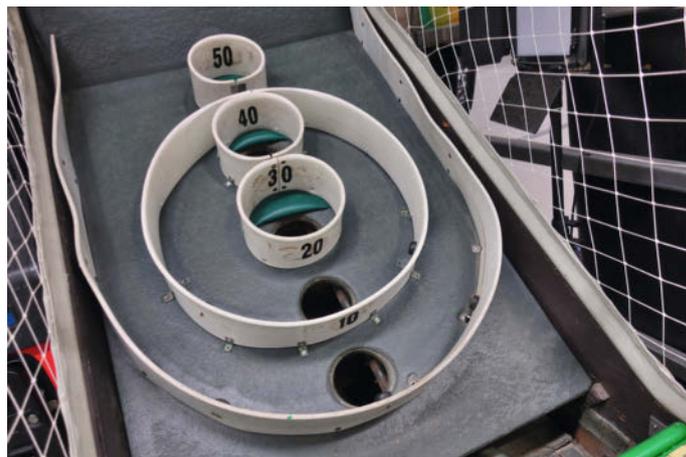
schen Hockey-Spieltisch aus den späten Dreißigern, bei dem das Bedienelement über einen Kettenantrieb 4 einen gusseisernen Hockey-Spieler bewegt 5. Mit dem schlagen die zwei Spieler eine Kugel über das in der Mitte hochgewölbte Feld hin und her, bis sie in eins der Tore hinter den Spielfiguren fällt. Der Münzeinwurf gibt die Kugeln per Schwerkraft frei 6. Strom braucht das Spiel nicht. Bei uns kennen wir das von Tischkickern. Gleich



10 Wer sich von den Reklametafeln am Straßenrand ablenken lässt und von der Fahrbahn abkommt bekommt keine Punkte und muss das Brummen des Automaten ertragen.



11 Der Skee-Ball-Automat erinnert ein wenig an eine Kegelbahn, doch gilt es hier, die Kugel über die Schanze am Ende der Rollbahn ...



12 ... in ein möglichst hochdotiertes Loch zu katapultieren – in Aktion in unserem Video zu sehen.

Mutoscope verwendete die Idee später wieder in den Automaten *Cross Country Race* (1948) und *Drive Yourself* (1954). Ziemlich komplex für die 40er-Jahre! „Das ist eigentlich total langweilig, wenn man es einmal gemacht hat“, lacht Josef. Doch die liebevoll handgemalten Details erfreuen bis heute Sammlerherzen. Und auch hier ist es nicht schwer, alle Funktionen zu verstehen. Die alten Geräte bilden krasse Kontraste zu heute, wo selbst Profis als Einzelpersonen nicht mehr die gesamte Funktion der hochkomplexen Maschinen kennen, an denen sie arbeiten.

### Projekte mit Azubis

Hinter seiner Liebe für die Oldies steckt jedoch mehr als Nostalgie. „Diese elektromechanischen Spiele eignen sich hervorragend, um Technik verstehen zu lernen“, findet Josef. Wir wechseln die Halle zu Josefs „Man Cave“, wie er sie nennt. Ein Gremlin turnt am Geländer. Durch einen Wanddurchbruch lugt die *DeLoorean-Zeitmaschine*. Die Wände sind mit bunter Reklame tapeziert, und natürlich steht alles voller Spielautomaten. „Das hier habe ich eben auch mit unseren Azubis restauriert – die lernen dabei wirklich viele Grundlagen“, erzählt Josef. „Das hier“ ist eine *Skee-Ball-Maschine* 11. Dabei rollt der Spieler eine Kugel eine Bahn mit Sprungschanze am Ende entlang, um sie möglichst mit hoher Punktzahl einzulochen 12. Ein Zählwerk 13 addiert die Punkte auf und bei jedem Volltreffer klingelt eine Glocke. Das alles funktioniert ohne Strom, die Energie stammt aus der potenziellen Energie der Kugeln, also letztendlich aus den Spielern. Beim Volltreffer läuft die Kugel zum Beispiel schlicht an der Glocke vorbei und schlägt sie dabei an. Außer mit den Tücken alten, mikrobiologisch gesättigten Holzes kämpf-



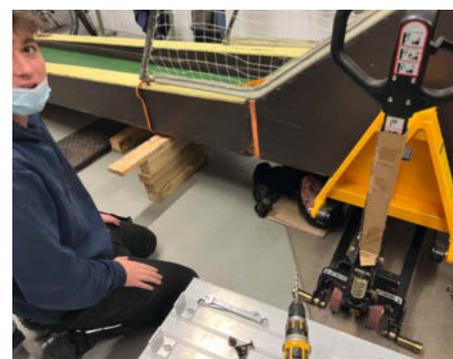
ten Josef und seine Azubis auch mit dem schlichten Verfall der Tragstrukturen, die sie daher aus Alu nachschweißten 14.

Im Foyer der Firma *Sportgeräte Schäper*, die Josef leitet, stehen weitere alte Spiele, darunter ein Basketballautomat. „Da spielen auch Kollegen und Kunden“, freut sich Josef. „Ich will, dass die Sachen dann auch benutzt werden. Dazu sind sie da.“ Ein simples, aber erstaunlich schwieriges Spiel im Gang der Firma ist das rein mechanische *Kicker-Catcher* 15: Man schießt eine Kugel nach oben, die durch ein Nagelbrett abgelenkt herunterklackert. Mit einem Drehknopf steuert man den Fänger. Richtig positioniert fällt die Kugel in den Catcher, der sie am Rand deponiert, wo sie über einen anderen Weg zurücklaufend das Zählwerk passiert 16. Viel einfacher geht es kaum in der technischen Umsetzung.

Viele seiner Spiele hat Josef nicht aus Spielhöhlen, sondern auf der Kirmes kennengelernt: Ein Armdrück-Automat 17. Ein Hau-den-Lukas-Automat 18. Spielende arbeiten hierbei gegen eine simple Feder. Die Beleuchtung ist nur Aufmerksamkeitserzeuger. Ein *Schützenstand-Automat* 19 überträgt die Position des Gewehrs auf einen Stanzer, der beim Abzug ziehen das Zielscheiben-Kärtchen locht, das die Spielenden später mitnehmen können 20 21. Da die Druckplatten für die Kärtchen längst verschollen sind, hat Josef sie digital neu aufgebaut und Kärtchen nachdrucken lassen. „Automaten wie diese hier standen früher bei uns auf jedem Rummel“, erinnert sich Josef. „Den *Schützenstand* will ich auf dem nächsten Schützenfest aufstellen und die Einnahmen dann spenden.“ Etwas komplexer wird es beim Dosenschießen-



13 Das Zählwerk im Inneren ist denkbar simpel konstruiert.



14 Die Skee-Ball-Bahn benötigte nach vielen Jahrzehnten Benutzung einen soliden neuen Unterbau.



15 Simple Spielprinzip, erstaunlich schwer: Beim Kicker-Catcher muss man den Ball fangen, der durch die Nagelreihen klackert.



16 Und so sieht der Kicker-Catcher auf der anderen Seite aus. Von Elektronik keine Spur, dafür Walzen, Nägel, Hebel und Federn.

Automat der Firma *Ermatronic* 22: Ein Lichtimpuls aus dem Spielgewehr löst (bei korrekter Ausrichtung) über eine Fotodiode 23 den Feder-belasteten Haken mit der Dose darauf 24: Die Dose fällt, wie von einem Luftgewehr-Pellet getroffen 25 „So simpel kann es sein.“

### Spieglein, Spieglein

Überhaupt waren Schießspiele schon in der Frühzeit der Spielmaschinen ähnlich verbreitet wie heute. Das liegt zum einen an der großen Faszination, dass das Schießen auf viele ausübt, zum anderen jedoch sicher auch

daran, dass sich die vergleichsweise überschaubaren geometrischen Gegebenheiten zwischen Lauf, Geschossbahn und Ziel elektromechanisch gut bauen ließen, wenn man sie dem Spiel entsprechend vereinfachte: So fliegt in den Spielen eine Kugel meistens eine lineare Flugbahn, in der Realität jedoch eine parabelförmige. Dabei stellte sich ein Platzproblem: Damit die Spiele nicht zu einfach wurden, brauchte man einen gewissen Abstand. Die Spielautomaten sollten jedoch mit möglichst wenig Stellplatz auskommen, denn der war in allen Kneipen und in den meisten Spielhallen knapp. Die Konstrukteure fanden den Platz schließlich in der Vertikalen.



17 Bananenbrecher, eher Klavierträger oder doch Weltmeister? Darüber entscheidet im Armdrückautomat eine simple Feder.



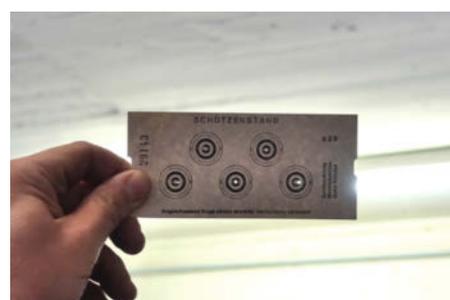
18 Beim Hau-den-Lukas-Automat drischt man per Faust auf das gelb-rote Kissen ein.



19 Der Schützenstand im Kasten gewährt nur durch das Zielfernrohr Einblick ins Innere und auf die Scheiben ...



20 ... die „Treffere“ werden einfach vom Automat eingestanz. Die gelochte Karte kann man anschließend als Trophäe mitnehmen.



21 Die Karten hat Josef Hesse eigens nachdrucken lassen, samt dem originalen Hinweis, dass genau das verboten sei.

## Wer ist dieser Verrückte?

Wer schon länger Make liest, erinnert sich vielleicht noch an Josef Hesse. Hat die erwähnte *Zeitmaschine* es klingeln lassen? Mit seinen DeLorean-Filmprojekten war er in Make 2/17 schon einmal ausführlich im Heft. Die Popkultur als solche bleibt seine Spielwiese. Aktuell steht ein von Josef gebautes *Jurassic-Park-Auto* zur Aufführung des neuesten Streifens der Reihe in einem lokalen Kino. Die elektromechanischen Automaten haben ihn aber über ihre kulturelle Relevanz hinaus gepackt. „Ich überlege mir ernsthaft, noch einmal Elektrotechnik zu studieren“,



sinniert er lachend. „Wie das aussähe! Ich alter Dinosaurier da zwischen den ganzen jungen Hüpfen!“

Zuvorderst sucht Josef jedoch Gleichgesinnte. Denn jedes Mal, wenn er nicht weiterkam, halfen ihm Experten alter Technik weiter – nachdem er ihr Vertrauen erworben hatte. „Ohne Didi, Wini, Wolle, Frank und viele andere würde ich oftmals schlicht nicht weiterkommen“, lacht Josef. Da es viele kommerzielle Protagonisten in dieser Szene gibt, braucht das etwas Geduld. Die erfahrene Hilfe will er seinerseits auch weitergeben: „Viele Leute sind sich der Werte gar nicht bewusst, die da in ihren Kellern stehen. Das sind dann natürlich tolle Schnäppchen für einige Händler, die eben auch gut daran verdienen können. Daran habe ich aber kein Interesse. Ich will Leute finden, die dieses Zeug genauso inspirierend finden wie ich. Deshalb habe ich dich angerufen.“ Und ich – als der so



Angesprochene – denke, eine eventuelle Ansteckung (oder auch Nichtansteckung) findet schon in dem Moment statt, in dem Sie das Video angucken, das ich von meinem Besuch bei Josef und seinen Automaten mitgebracht habe und das Sie über den Link im Kasten auf Seite 81 erreichen. Wenn es Sie inspiriert – die Make-Redaktion stellt gerne den Kontakt zu Josef Hesse her.

In den meisten elektromechanischen Schießspielen hängt daher ein im 45°-Winkel angebrachter Spiegel, der den Blick erweitert – in die Senkrechte nach unten in den Automatenkasten. Dieser simple Trick sorgt für einen Effekt, den Leute als „holographisch“ beschrieben haben. Es sieht schlicht sehr faszinierend aus, wenn das Spielgeschehen weiter entfernt ist, als der Kasten tief [26](#). Bei *Machine Gun Defender* aus dem Jahr 1971 von *Chicago Coin* fliegen beleuchtete kleine Flugzeuge vor einem schwarzen Hintergrund. Man beschießt sie durch einen Controller, der wie moderne Controller ein Force-Feedback durch einen Unwuchtmotor erzeugt (nur mehr davon). Bei einem Treffer erleuchten Explosionslampen eine Landschaft, die an manche Märklin-Hobby-Bauten erinnert.

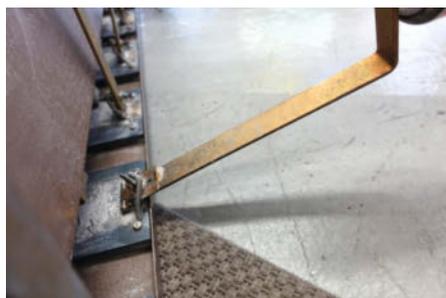
Chicago Coin baute das Prinzip in *Foreign Legion* noch etwas um (oder aus), mit einem



[22](#) Selbst Dosenschießen wurde schon als Automat nachgebildet, hier von der Firma Ermatronic.



[23](#) Unter den Dosen sind Fotodioden angebracht, die Treffer mit dem Lichtgewehr registrieren ...



[24](#) ... dann den Haken lösen, sodass die Feder wirken kann, ...



[25](#) ... woraufhin die „getroffene“ Dose nach hinten wegklappt.



27 Der Automat Foreign Legion aus den 70er Jahren erforderte einiges an Restaurierung ...



28 ... ehe die mechanischen Heckenschützen und Flugzeuge wieder aus den Tiefen des Wüstendioramas auftauchten.



26 Wie bei vielen elektromechanischen Arcade-Automaten erweitert auch bei diesem Exemplar ein um 45 Grad gekippter Spiegel das Revier für die beiden Jäger-Spieler optisch über die Rückwand des Geräts hinaus.



29 Vor allem die beiden Controller in Form von Maschinenpistolen hatten unter dem Elan der vielen Freizeit-Fremdenlegionäre vergangener Jahrzehnte gelitten. Die neuen Attrappen stammen aus der Fräse und dem 3D-Drucker.

Wüsten-Diorama, anderen Flugzeugen und Aggressoren, die hinter den Dünen hervor-  
klappten, um abgeschossen zu werden – und vor allem von zwei Spielern gleichzeitig 27  
28. Bei diesem Spiel waren die Controller – also die Waffenattrappen – kaputt, sodass  
Josef Teile davon aus Metall nachfräsen musste. Das Plastikgehäuse druckte er im  
3D-Drucker nach 29.

### Kino-Qualität

Gegen Ende ihrer Zeit, kurz bevor sie von elektronisch-digitalen Automaten abgelöst wurden, erreichten elektromechanische Spiele eine erhebliche Komplexität. Ein Beispiel für den getriebenen Aufwand: Nintendos *Battle Shark* von 1977. In diesem Spiel hält man einen Maschinengewehr-Controller aus Stahl und schießt auf fotorealistische Torpedoboote. Sie sind deshalb fotorealistisch, weil sie tatsächlich von 16-mm-Filmstreifen kommen. Bei Treffern wechselt der Automat auf die Explosions-Animation. „Das ist natürlich sehr empfindlich“, sagt Josef. „Und auch begrenzt erhaltbar. Wenn die Filmstreifen kaputtgehen, kann ich das kaum noch reparieren.“ Deshalb überlegt er sich, den Film zu einer Filmrettung zu geben, die das Material digitalisiert und aufbereitet. Dann wäre vorgesorgt. Dieser Automat hat für ihn eine besondere Bedeutung, weil es der erste elektromechanische

seiner Sammlung war – das Gerät, das ihn feststellen ließ, dass er diese Geräte noch spannender findet als die späteren Arcade-Automaten mit elektronisch-digitalen Rechnern.

Ein anderes komplexes Beispiel steht neben der Skee-Ball-Bahn: *Submarine* von Namco war 1978 einer der letzten elektromechanischen Arcade-Automaten <sup>30</sup>. Er wurde gelobt für seine „realistische“ Spielumgebung (im *Cash-box-Magazin*, 1978) und lief ab 1979 auch als Lizenz bei *Midway*. Doch bis die Spielumgebung dieses speziellen Gerätes wieder sichtbar wurde, stand einiges an Restaurierungsarbeit an. Als Josef den Automaten fand, war er von Efeu umschlungen.

„Wahrscheinlich hat das Efeu die Mechanik konserviert“, vermutet er <sup>31</sup>. Nach viel Säubern (vor allem der Kontakte und Elektromotoren) und ein paar neuen Lampen lief das Gerät wieder. Eine beleuchtete Trommel projiziert ein bewegtes Wolkenband auf den Hintergrund <sup>32</sup>. Darunter liegt das Meer, das sich zu heben und senken scheint, weil ein bewegter Spiegel die Meeresoberflächengrafik verzerrt. Den Torpedo simuliert eine Lichtbahn, und bewegt wird das Ganze durch Eingabe von oben entlang einer Schiene per Servo. Das Ganze ist so laut, dass der Automat keine Hintergrundmusik braucht (oder haben könnte). Einziger Soundeffekt: der Treffer. Das Spiel ist eine liebevolle, interaktive Theaterkulisse in miniature. Namco konstruierte noch einen Nachfolger mit dem Arbeitstitel *Submarines*, der jedoch nicht mehr produziert wurde. Die Zeit der elektromechanischen Spiele war vorbei.



## Der Aufstieg der Videospiele

Elektromechanische Spiele waren schwer, groß, teuer und wartungsintensiv. Manche Maschinen des Pioniers *Kasco* in Japan kosteten mehr als ein Toyota Corolla. Für ihre Rennspiele verwendete *Kasco* (wie auch *Mutoscope*) echte Auto-Lenkräder von Sportwagen. Solche und andere teure Teile zogen mit der Zeit auch Diebe an. Für Betreiber von Arcade-Hallen oder Barbesitzer waren die schweren Schränke zudem sehr unhandlich. Kannten die Gäste ein Spiel, sanken die Einnahmen rapide. Erst etwas Neues würde die Gäste wieder dazu animieren, Geld einzuwerfen. Das jedoch war immer eine große Investition und die empfindliche Elektromechanik verlangte viel Wartung.

Obwohl die ersten Videospiele ähnlich aufwendige und individuelle Gehäuse hatten wie die elektromechanischen Automaten vorher,



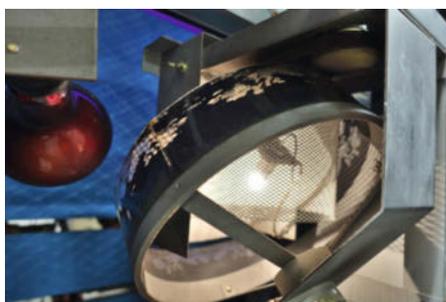
<sup>30</sup> *Submarine* aus dem Jahr 1978 simuliert den Kommandostand eines U-Boots, der Blick durchs Persikop zeigt ein mechanisch animiertes Bild mit Wolken, Wellen, Schiffen und leuchtenden Torpedospuren.

setzten sich mit der Zeit mehr und mehr Standardgehäuse durch, in denen die Betreiber einfach durch einen Platinentausch ein neues Spiel anbieten konnten (siehe *JAMMA-Standard*, ab 1986).

Doch später verschwanden die Spielautomaten praktisch ganz aus der deutschen Öffentlichkeit. Da spielten verschiedene Faktoren hinein: PC- und Konsolenspiele wurden besser und erschwinglicher. 1985 verbannte eine Jugendschutz-Novelle Arcade-Automaten aus der Öffentlichkeit. Wer denkt, die *Killerspiel-Debatte* sei noch jung: Sie wurde damals bereits geführt. Nach dem Verbot standen Videospieleautomaten nur noch in Erwachsenenbereichen, was ihnen einen etwas halbseitigen Ruf einbrachte. Heute kümmern sich Vereine um das kulturelle Erbe und arbeiten daran, diesem auch in Deutschland einen

Ruf wie in den USA zu verleihen: Als Teil der Entwicklung, der Geschichte unserer Spiele, der lebendig erlebt werden kann. Bis eine „Spielhalle“ in Deutschland auch für Laien, für Nicht-Kenner ähnlich positiv aufgenommen wird wie eine Retro-Arcade-Halle in Kalifornien, muss aber wahrscheinlich noch etwas Gras über unsere Einstellung zu diesen Maschinen wachsen. Die oft technisch faszinierenden und manchmal auch wunderschönen elektromechanischen Spiele könnten hier Brücken schlagen helfen.

—pek



<sup>32</sup> Diese von innen beleuchtete Trommel projiziert das bewegte Wolkenband in das Miniaturtheater der Spielwelt von *Submarine*. Im Video zum Artikel sieht man es ebenso in Aktion wie den geschwenkten Strahler zur Anzeige von Torpedospuren.



<sup>31</sup> Einfach mal richtig saubermachen sind 90 Prozent der Restaurierungsarbeit, sagt Josef Hesse. Der *Submarine*-Automat war sogar von Efeu berankt, als Josef ihn fand.

# Retro-Corner: Analoges Speicher-Oszilloskop

Heute ist es selbstverständlich, auf einem Digital-Oszilloskop einmalige Signalereignisse aufzuzeichnen. Vor 50 Jahren musste man zur Darstellung schon recht tief in die Trickkiste greifen, wie unser Blick in ein altes Speicher-Oszilloskop zeigt.

von Carsten Meyer



**D**igitale Speicheroszilloskope haben längst ihre analogen, mit einer Kathodenstrahlröhre ausgestatteten Vorgänger ersetzt – nicht zuletzt, weil ein paar Chips und ein TFT-Display deutlich billiger herzustellen sind als das feinmechanische Meisterwerk, das im Innern einer Bildröhre (CRT, *cathode ray tube*) den Elektronenstrahl erzeugt und ablenkt.

Über viele Jahrzehnte war die Firma *Tektronix* der Weltmarktführer bei Oszilloskopen, und viele Geräte aus den 1960er- und 1970er-Jahren tun heute noch treu ihren Dienst. Aus den frühen 1970ern stammt auch unser Gerät, das mit einer beachtenswerten Besonderheit ausgestattet ist: einer analogen Speicher-Bildröhre.

### Frühe Ansätze

Schon früh machten sich die Ingenieure bei Tektronix darüber Gedanken, wie man einmalige oder seltene Signalverläufe dauerhaft sichtbar machen kann. Eine erste Idee war es, den Bildschirminhalt einfach als Momentaufnahme abzufotografieren; die Auslösung der angeflanschten Sofortbildkamera erfolgte über das Trigger-Signal. Doch bereits 1964 hatte man dort eine analoge Speicherröhre zur Serienreife entwickelt, die im Modell 564 debütierte und quasi eine von Millisekunden bis Minuten einstellbare Nachleuchtdauer aufwies. Damit ersparte man sich das umständliche Hantieren mit Film und Kamera, außerdem ließ die neue Technik eine deutlich höhere Schreibgeschwindigkeit zu; für sehr schnelle Signale war ein fotografischer Film einfach zu unempfindlich.

Erst sehr viel später, Anfang der 1980er-Jahre, hielt die Digitaltechnik Einzug in die Oszilloskope. Da aber noch keine ultraschnellen A/D-Wandler zur Verfügung standen, behalf

## Kurzinfo

- » Analoges Speichern in der Kathodenstrahlröhre
- » Entwicklungsschritte bei Speicher-Oszilloskopen
- » Technik des Tektronix 466

### Mehr zum Thema

- » Olaf Göllner, Signalen auf der Spur, Make 4/2013, S. 62
- » Jan Mahn, Wellengucker, Einstieg in die Arbeit mit dem Oszilloskop, c't 16/2021, S. 162

Alles zum Artikel  
im Web unter  
[make-magazin.de/x2ch](https://www.make-magazin.de/x2ch)

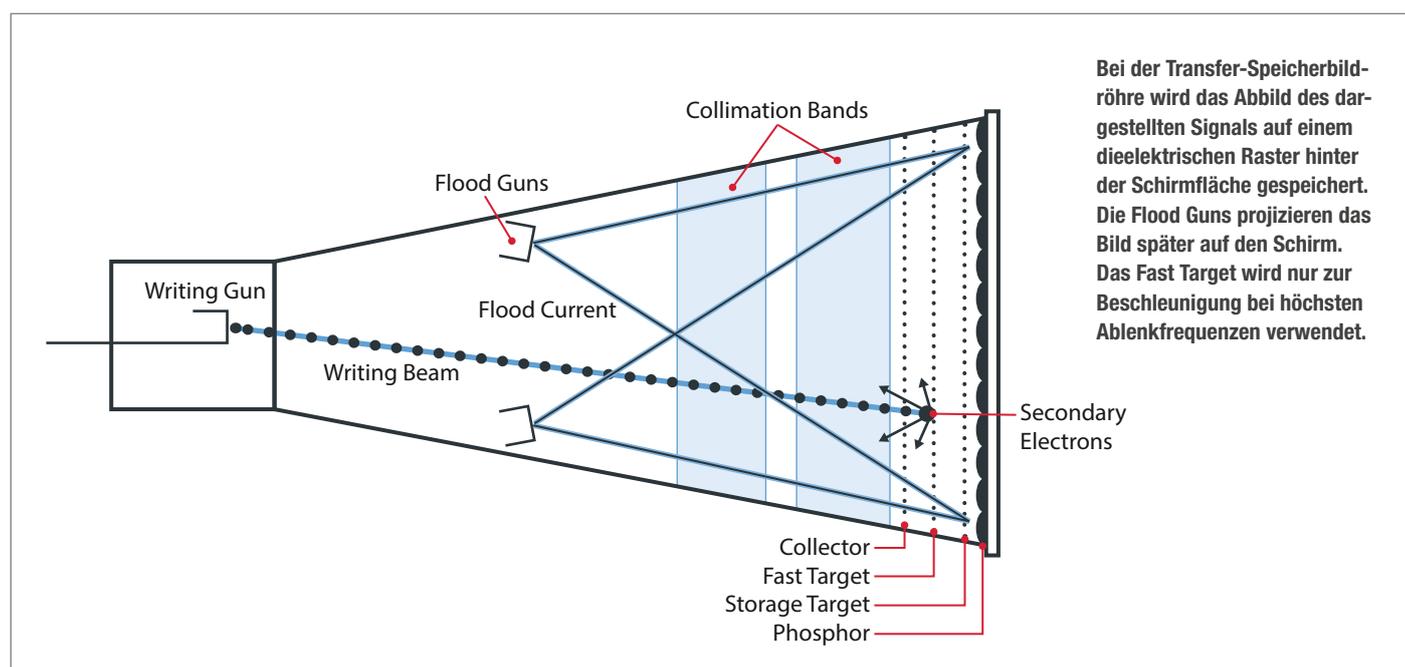
man sich mit einem Trick: Schnelle Eingangssignale wurden zunächst per (analogem) Sampling portionsweise mit hoher Geschwindigkeit in ein analoges Schieberegister nach dem CCD-Prinzip geschoben, um die „Eimerkette“ dann mit einer dem A/D-Wandler (ADC) genehmen Geschwindigkeit wieder zu verlassen. Das CCD konnte rund 1000 analoge Messwerte aufnehmen, schnelle Exemplare wie im Modell 2440 brachten es in den späten 1980ern schon auf 500 MHz Sampling-Takt (entsprechend 250 MHz Bandbreite).

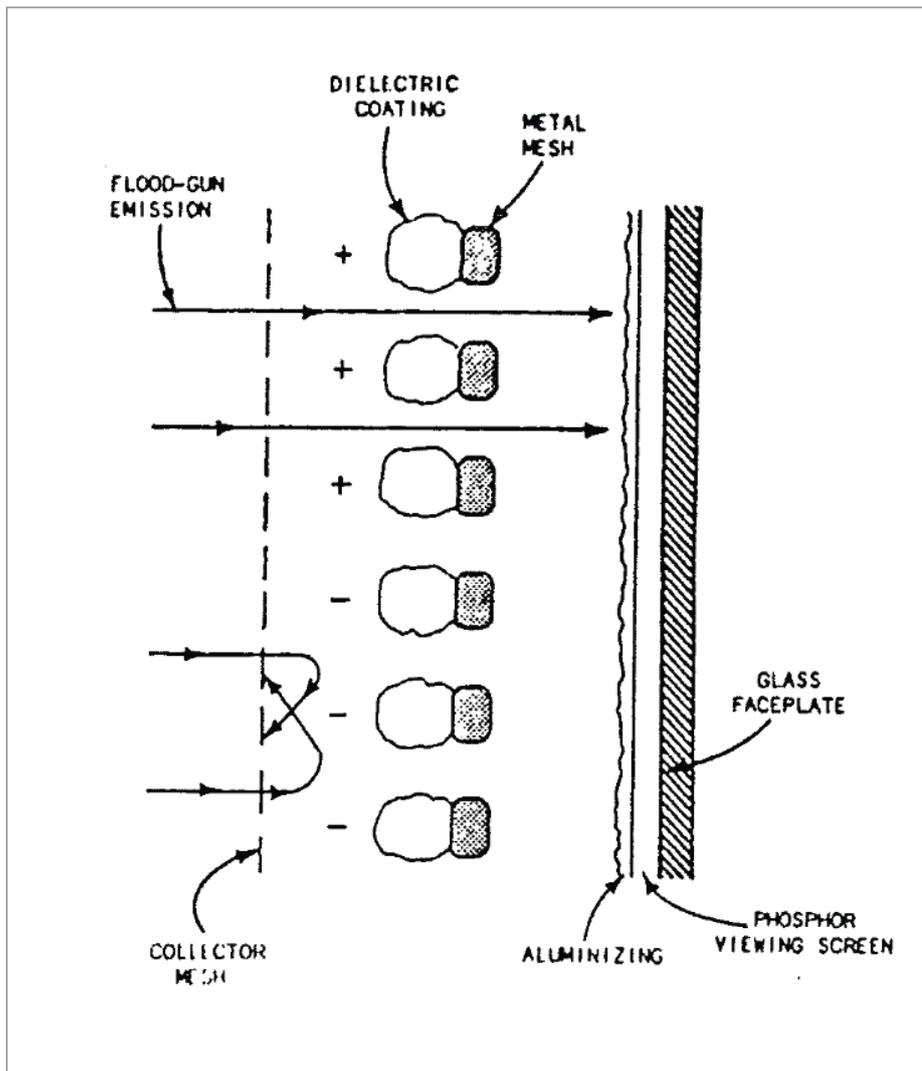
Tatsächlich entsprach das Funktionsprinzip der Analogspeicher-CCDs der von Bildaufnehmern, nur das hier halt kein Bild aufgenommen wurde, sondern die nach dem Eimerketten-Prinzip weitergereichten Ladungen über eine Leitung schrittweise in das IC gelangten. Kein Wunder, dass Tektronix auch in Sachen Bildaufnehmer-CCDs auf dem Markt mitmischte – die Technik war ja fast identisch. Dem CCD-Prinzip blieb der Hersteller bis zur

Jahrtausendwende treu; erst dann standen ausreichend schnelle ADCs zur Verfügung, die schließlich dem digitalen Speicheroszilloskop den endgültigen Durchbruch brachten.

### Analoges Abbild

Zurück zu unserem Oszilloskop: In den 1960er Jahren lagen schnelle CCDs noch in weiter Ferne, weshalb man sich zu einer Speicherung des Bildinhaltes direkt in der Röhre entscheiden musste. Dabei erinnerte man sich an die alten *Williamsröhren*, die in den allerersten Computern als Hauptspeicher Verwendung fanden und einer Bildröhre sehr ähnlich waren: Bei einer Williamsröhre entspricht ein Speicher-Bit einem Leuchtfleck an einer adressierbaren Position auf dem Bildschirm; zur (kurzfristigen) Speicherung der Information dient die Nachleuchtdauer des Phosphors. Zum Auslesen beobachtet man das Potential an einer Gegelektrode: Solange der Phosphor nach-





Arbeitsweise des dielektrischen Rasters und des Kollektor-Gitters: Nur an Stellen ohne negative Ladung gelangt der Elektronenfluss aus den Flood Guns auf den Schirm.

leuchtet, entsteht ein leicht veränderter Impuls, wenn der Elektronenstrahl eine „Speicherstelle“ trifft.

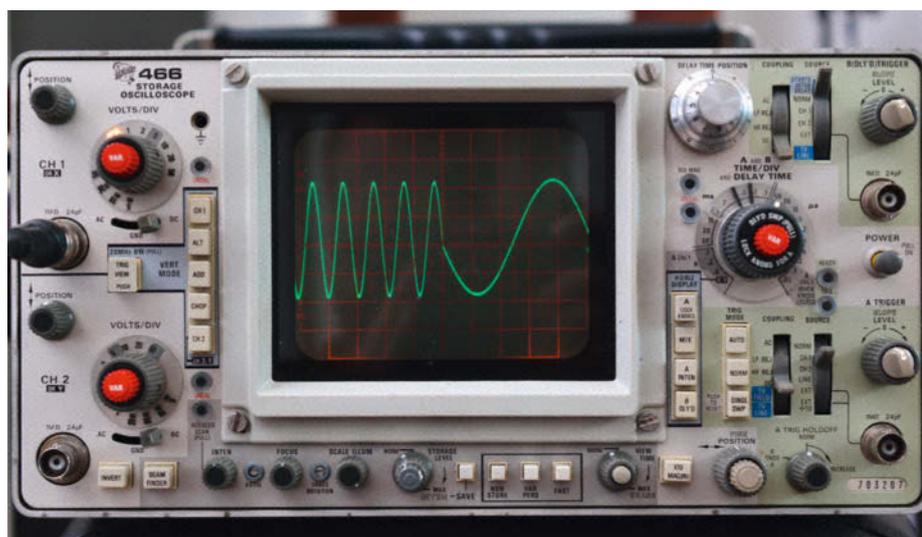
Die bistabile Tektronix-Speicherröhre nutzt Sekundärelektronen, die eine dielektrische Folie einsammelt. Dazu hat man knapp hinter dem Leuchtschirm zwei extrem feinmaschige Metallgitter angebracht, von denen das zweite (aus Sicht der Elektronenkanone) mit der dielektrischen Schicht überzogen ist. Eine Steuerschaltung kann nun an die Gitter verschiedene Pegel anlegen, die den Betriebsmodus vorgeben. Im Normalbetrieb erreicht der Elektronenstrahl die Leuchtschicht durch die Gitter hindurch, wie bei einer normalen Kathodenstrahlröhre auch.

### Ladungen sichtbar gemacht

Spannend wird es im Speicherbetrieb: Hier beschreibt der Elektronenstrahl neben dem Leuchtschirm auch das dielektrisch beschichtete, mit einer hohen Spannung vorbelegte Gitter. Anhaftende Ladungen werden dort, wo der Elektronenstrahl das Dielektrikum trifft, durch den Strahlstrom entfernt – etwa so wie bei der Fototrommel eines Laserdruckers. Anschließend liefern zwei unfokussierte, breit streuende Flood Guns Elektronen, die durch das erste Kollektor-Gitter das mit dem Signalverlauf beschriebene dielektrische Raster erreichen. Ist dort noch eine negative Ladung vorhanden (also an den „dunklen“ Stellen), kehren die Elektronen durch Abstoßung zum ersten Gitter (das Collector Mesh im Bild) zurück und werden dort eingesammelt.

Das dauerhafte Schirmbild entsteht also nur durch die von den Flood Guns ausgestoßene Elektronenwolke; es ist daher dunkler als der direkte Strahl. Über hohe Spannungsimpulse an den Gittern lässt sich der Bildschirminhalt wieder löschen: Der Bildschirm leuchtet dann kurz komplett hell auf und wird nach dem Laden des Dielektrikums wieder dunkel. Dieser Vorgang dauert einige Zehntelsekunden, er kann automatisch oder manuell ausgelöst werden. Interessant ist, dass man mit einem erhöhten Strahlstrom der Flood Guns und entsprechenden Pegeln an den Gittern sogar eine variable Nachleuchtdauer erreichen kann.

Klingt kompliziert? Ist es auch: Nicht nur die Röhre mit ihrem trickreichen und reichlich komplizierten Innenaufbau ist ein Kunstwerk, auch die Ansteuerschaltung, die eine Abfolge definierter Pegel von einigen hundert Volt liefern muss. Höhepunkt der Entwicklung bei den analogen Speicherröhren war 1987 sicherlich das (inflationbereinigt 55.000 Euro teure) Tektronix-Oszilloskop 7934 mit einer Bandbreite von 500 MHz – wohlgermerkt zu einer Zeit, als die schnellsten Mikroprozessoren noch mit 40 MHz liefen.



Vorbildliche Ergonomie – nicht trotz, sondern wegen der vielen Bedienelemente: Das Tektronix 466 ist fast ebenso leicht zu handhaben wie sein beliebter Bruder ohne Speicher-Röhre, das 465. Mit Hilfe der zweiten Zeitbasis wurde das Signal hier in der rechten Bildschirmhälfte gedehnt.

## Schnäppchen

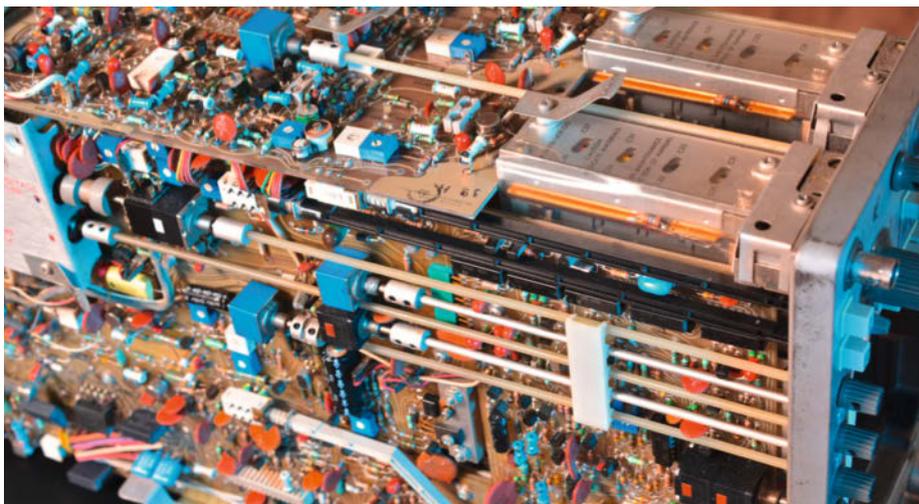
Unser Modell 466 gibt sich da bescheidener, obwohl 100 MHz Bandbreite auf zwei Kanälen Mitte der 1970er schon eine ordentliche Hausnummer waren. Wir ergatterten es auf dem Flohmarkt für 30 Euro, etwa einem Hundertstel des Neupreises. Allerdings funktionierte die Triggerung nicht richtig, weil auf einer der zahlreichen Baugruppen ein paar Tantal-Kondensatoren abgeraucht waren – ein typischer Fehler bei älteren Geräten.

Die waren schnell ersetzt, und das 466 lief wieder wie vor knapp 50 Jahren. Wie sagte schon der großartige Jim Williams, Applikations-Ingenieur bei Linear Technologies, angesichts der freudigen Momente bei einer Messgeräte-Reparatur: „Fixing is simply a lot of fun. I may be the only person at an electronics flea market who will pay more for the busted stuff!“

Doch auch ein Schnäppchen wirft die Frage auf, ob der Einsatz eines solchen Gerätes heute noch sinnvoll ist. Sein größter Vorteil liegt zweifelsfrei in der ergonomischen, geradlinigen Bedienung: Statt mehrfach belegter Buttons gibt es richtige Drehknöpfe und Schalter, was Fehlbedienungen minimiert. Zudem funktioniert es zu 95 Prozent wie ein normales analoges Oszilloskop; die bei Digital-Scopes auftretenden Aliasing-Effekte (Phantom-Signalformen durch zu niedrige Abtastraten) sind ihm völlig fremd. Und mit der Doppel-Zeitbasis lassen sich sogar Ausschnitte aus Wellenformen vergrößert darstellen, etwa um eine Signalfanke genauer beurteilen zu können.

Was es nicht kann: Jedes anständige Digital-Speicheroszilloskop zeigt durch „Zurückspulen“ des Speicherinhalts auch Ereignisse vor dem eigentlichen Triggerpunkt an, während das 466 natürlich immer erst auf das Triggerereignis warten muss, bevor es mit der Aufzeichnung beginnt. Für die Analyse eines kurzen Datenpakets beispielsweise einer seriellen Übertragung reicht es trotzdem. Die richtige Einstellung für die Speicher-Darstellung zu finden, ist allerdings etwas fummelig: Zwei sich gegenseitig beeinflussende Potis wollen dafür justiert sein.

Die späteren CCD-Speicheroszilloskope boten demgegenüber nur wenig Vorteile, auch wenn sie bereits mit tollen Cursor-Funktionen zur Signalanalyse glänzten. Ein großer Nachteil dieser Geräteklasse ist die relativ lange „Totzeit“ zwischen den Aufzeichnungen: Ist ein Signalpaket im CCD, benötigt der auslesende Mikrocontroller relativ lange (bis zu 50 Millisekunden), bevor wieder neu aufgezeichnet werden kann; in diese Zeit fallende Ereignisse werden ignoriert. Neuzeitliche Speicheroszilloskope der mittleren Preisklasse schaffen dagegen schon eine Million komplette Aufzeichnungen pro Sekunde (*waveforms per second*). —cm



Innenansicht: Dank der diskreten Bauteile lassen sich viele Defekte auch heute noch reparieren. Einige der roten Tantal-Kondensatoren auf der Platine haben wir ersetzen müssen.



Das komplizierte Innenleben der Bildröhre, dem mit Abstand teuersten Bauteil des Oszilloskops, wird von außen kaum ersichtlich. Man erkennt hier nur die Elektronenkanone mit den XY-Ablenksplatten, hinten rechts im Röhrenhals noch eine der Flood Guns (helle Scheibe).



Zwischenschritt: Das Digital-Oszilloskop Tektronix 2440 aus dem Jahre 1988 speicherte aufgezeichnete Signale vor der A/D-Wandlung in analogen Eimerketten-ICs (CCDs).

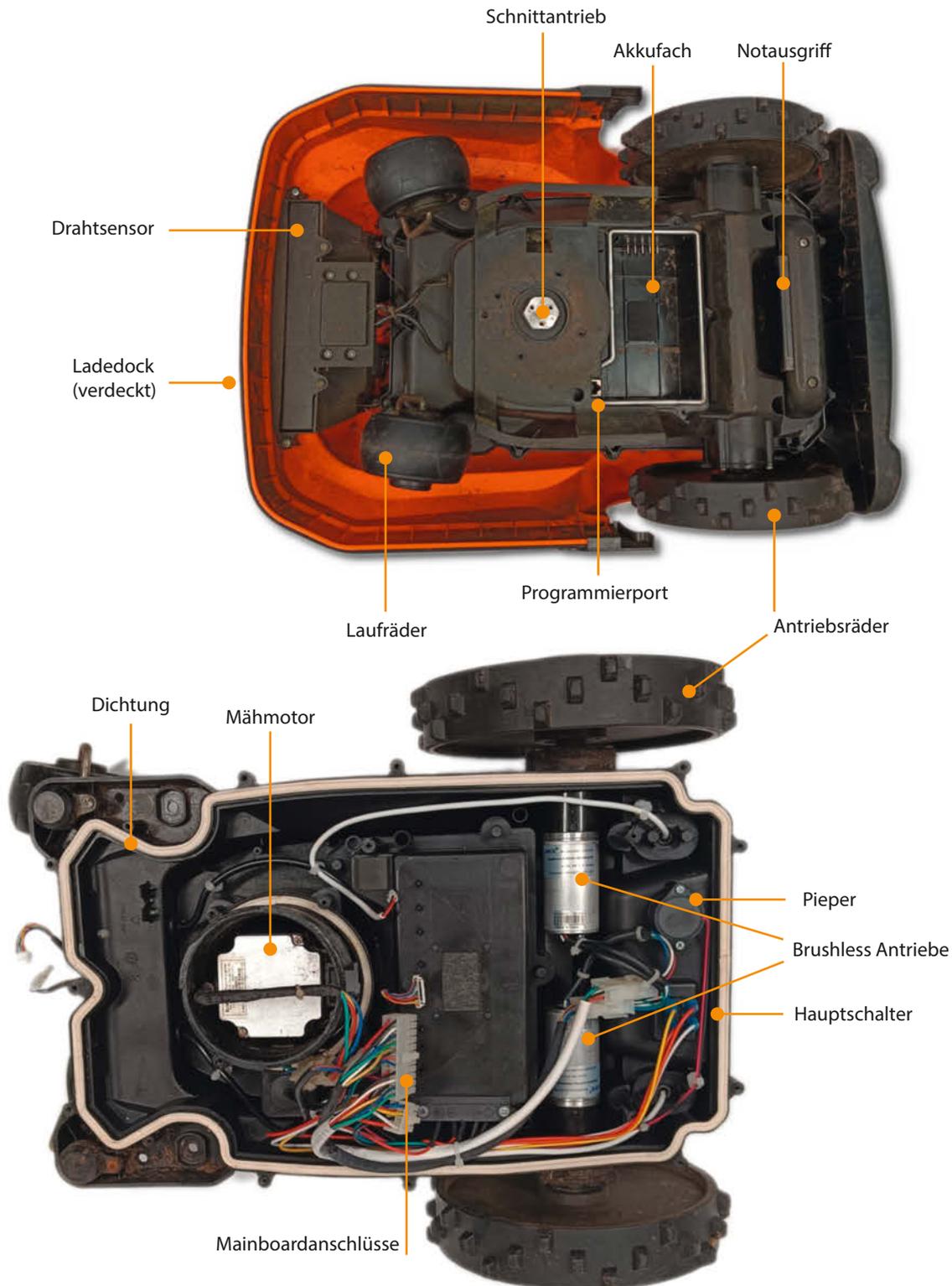
# Rasenmähroboter

Ich habe einen nicht zu alten, aber defekten Rasenmähroboter vor der Verschrottung gerettet und aufgeschraubt. Wie ist er konstruiert und welcher Fehler war Grund, dass er durch einen neuen ersetzt wurde? Taugt der Mäher als Basis für einen Ardumower oder nur als Teilespender?

Alles zum Artikel  
im Web unter  
[make-magazin.de/x659](https://make-magazin.de/x659)



von Carsten Wartmann



**D**er Rasenmäroboter eines Nachbarn tat keinen Mucks mehr, ein neuer wurde gekauft. Zum Glück war der alte noch nicht beim Recyclinghof und ich trug ihn erwartungsfroh in meinen Garten. Die Fehlerbeschreibung lautete sinngemäß: *Geht nicht!*

Äußerlich sah der *Yardforce SC600ECO* (Hersteller *Sumec*) noch ganz gut aus, bloß Abnutzungsspuren und ein Riss der Folientastatur auf dem *Unlock*-Knopf fielen auf. Dieser scheint seinen früheren Besitzer in Rage gebracht zu haben, was ich nach der Lektüre der Anleitung nachvollziehen kann. Auf den Unterboden geschaut war der Mäher sauberer als gedacht. Hier und da wurde Edelstahl oder gut galvanisiertes Metall verwendet: kein Rost zu sehen an vitalen Teilen. An der Ladestation leuchtete nur die Anzeige für den Signaldraht, am Mäher auch nach 30 Minuten nichts: kein gutes Zeichen.

Der von mir vermutete Defekt lag am Akku. Das Ausbauen gestaltete sich etwas kompliziert, alles gut verschraubt und abgedichtet. Als ich die Zellen ( $7 \times 18650$ ,  $2000\text{mAh}$ , Li-Ion) gemessen habe, waren viele auf 0 Volt oder deutlich unter der Nennspannung und ließen sich auch einzeln nicht mehr reaktivieren. Wie in so vielen neuen Geräten steckt im Akku viel (Lade-)Elektronik, die man bei jedem Neukauf mitbezahlt.

Beim weiteren Demontieren fand ich sehr solide Lösungen, aber auch Details, die mich an den Designern zweifeln ließen. Oder waren es Entscheidungen der Verkaufsabteilung? Je tiefer man in das Gerät eindringt, desto billiger werden die Schrauben, und wenn dann wie am Mähmotor keine Dichtung gegen eindringende Feuchtigkeit existiert, rostet es. Auf der anderen Seite wurden hochwertige Steckverbinder zwischen den Komponenten verwendet, was eine Reparatur beim Hersteller



**Der Mäher ist eine sehr brauchbare Grundlage für Experimente und mit Wackelaugen sieht sowieso alles besser aus.**

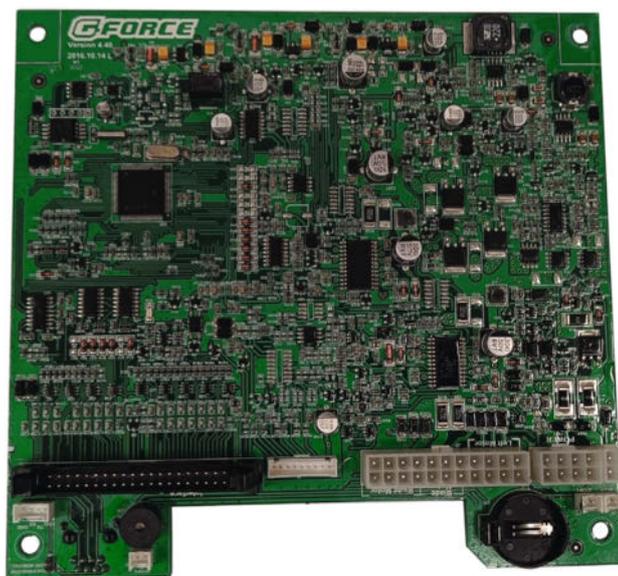


**Der Akku – selbst mit viel Heißluft habe ich die Verklebung der Zellen nicht lösen können.**

sehr erleichtert. Die Platinen sind mit Schutzlack gegen Feuchtigkeit versehen. Die Displayplatine aber komischerweise nur auf einer Seite, sodass sie durch über die Lock-Taste eindringendes Wasser korrodiert war. Die Stützbatterie auf dem Mainboard hat auch einen Tropfen Wasser abbekommen, jedenfalls war sie rostiger als jede ausgeladene Batterie, die ich bisher gesehen habe.

Schon beim Entfernen des Akkus war mir ein Anschluss für *RJ45*-Stecker aufgefallen und tatsächlich ist es ein serieller Port, der auf der Hauptplatine auch zu einer Buchse mit *TX*- und *RX*-Pins führt. Ferner ist noch ein Jumper vorhanden, beschriftet mit *PROGRAMM/NORMAL*. Hier wäre also Hacking möglich: Der Prozessor ist ein *Silicon Labs C8051F120*. Die Platine ist eng mit vielen interessanten Teilen wie Motortreibern und Leistungselektronik bestückt.

Die Antriebsmotoren sind Brushlessmotoren mit Getriebe (Links über den Kurzlink/*QR*) und kosten in der aktuellen Situation ab etwa 70 Euro pro Stück. Der Mähmotor ist ebenfalls ein Brushless mit  $30\text{W}$  und  $3500\text{U/min}$  laut Typenschild. Auch der Mähmotor ist gebraucht kaum unter 40 Euro zu bekommen, meiner sitzt leider durch die Korrosion an der Achse fest. Aber auch ohne Mähwerk wäre die ganze Konstruktion eine brauchbare Basis



**Die Hauptplatine, viele interessante Bauteile**

für einen fahrenden Roboter: ein Mikrocontroller, Akku und passende Motortreiber würden wohl reichen.

Nach ein paar Experimenten werde ich mich aber daran wagen, einen Mäherroboter auf Basis dieses defekten Mähers nach dem *ArduMower*-Projekt zu bauen. Der *ArduMower* fährt dank *Differential-GPS* komplett ohne Begrenzungsdraht durch den Garten. Ein Bausatz kostet nicht wenig, ich denke aber, ich kann durch die vorhandenen Teile schon mal eine Menge Geld sparen. Wenn die Chipkrise anhält, kann ich vielleicht noch mit den Bauteilen der Hauptplatine reich werden. —caw

# Tipps & Tricks

Diesmal ein Tipp zur soliden Befestigung von Dingen mittels Lochblech und ein Tipp zum Thema Multimaterialdruck auf herkömmlichen 3D-Druckern

von Hans-Jürgen Pretzel und Manfred Maroszek

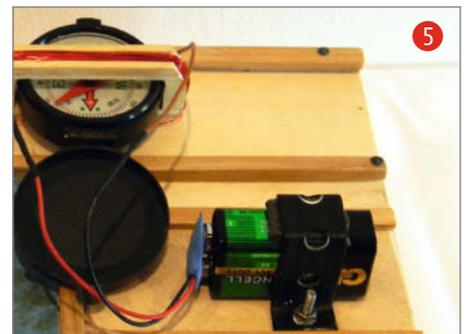
## Lochband als Befestigung und Halbzeug für Maker

Jeder Maker kennt und schätzt die Vorteile, kennt aber auch die Anwendungsgrenzen von Klebebändern. Ein Hilfsmittel mit ähnlichen Anwendungsmöglichkeiten, aber für dauerhaftere Befestigungen sind handelsübliche *Lochbänder*. Vom einfachen, gelochten Blechstreifen mit und ohne Beschichtung bis hin zu *Windrispenband* für Dachstühle gibt es unterschiedlichste Anwendungen und Lösungsmöglichkeiten für Befestigungen, schnelle Fixiermöglichkeiten oder sogar für die flexible Anfertigung von Blechteilen für mechanische Aufbauten, was ich im Folgenden zeige.

Die klassischen Anwendungsfälle für Lochbänder sind schellenartig geformte Blechstreifen zur Befestigung von Stangen **1**, Kabeln, Rohren oder Leitungsbündeln im Baugewerbe und zum Abhängen von Zwischendecken. Die benötigte Blechstreifenlänge wird von der Rolle durch einfachen Schnitt nach Bedarf mit einer Handblechschere abgelängt und mittels der regelmäßig gestanzten Löcher wird das Band mit Schlagstiften oder Schrauben bzw. Nägeln mit größerem Kopf befestigt.

Lochbänder werden in unterschiedlichsten Materialausführungen hergestellt. Die mit Kunststoffummantelung **2** (rechts im Bild) finde ich als Maker besonders interessant, da die Kunststoffhülle seitlich scharfe Blechkanten verhindert, eine gewisse elektrische Isolation bietet und optisch ansehnlicher ist als ein nackter Blechstreifen.

Die Bänder werden in unterschiedlichen Breiten, Blechdicken und Lochungen angeboten. *Rispenband* hat aufgrund der Belastungen im Dachstuhl die größten Blechstärken. Die regelmäßig gestanzten Löcher sind bei einfachem Lochband hintereinander in einer Reihe angebracht. Es gibt aber auch Lochband mit unterschiedlich großen Löchern z. B. in drei Reihen, was den Vorteil bietet, dass dieses Lochband längst der Mitte abgewinkelt werden kann und dann seitlich an jedem Schenkel Lochungen für die Befestigung hat.



Weiterhin kann man Lochband als Halbzeug verwenden, um daraus schnell kleinere Bauteile **3** wie Winkel, Halter, Befestigungsstreifen oder Aussteifungen zu fertigen. Durch die vorgestanzten Löcher besitzen diese Teile dann bereits Befestigungsbohrungen in einem festen Raster. Für die wiederkehrende Anfertigung von Bauteilen ist es sinnvoll, sich Hilfswerkzeuge **4** zum Abwinkeln, Biegelehren für Schellen und Vorrichtungen etwa für flachgekloppte 45° Winkel anzufertigen. Zur Aussteifung von Streben kann man die Lochstreifen längs in der Mitte knicken. Dafür eignet sich ein breiter L-Winkel aus Stahl, in welchen das Lochband über die Innenkante eingelegt und mit einem seitlich zugespitzten Flachstahl eingedrückt oder eingeschlagen wird. Für dünneres Lochband kann man diese Hilfswerkzeuge auch aus Hartholzstücken anfertigen.

Die praktische Handhabung ist relativ einfach und simpel, sodass auch Kinder in der Lage sind, das nicht so dicke, kunststoffbeschichtete Rollenmaterial zu schneiden, zu biegen und zu formen. So wird vor allem das gefährliche Bohren von Löchern in Blech vermieden, da sich Bleche schwer fixieren lassen und leicht zu schnell rotierenden Messern werden. Sehr vorteilhaft sind Lochbänder, die aufgerollt in Spendern (Bild **2** links) zum Rausziehen vertrieben werden. So wird verhindert, dass sich die unter Spannung stehende Lochbandrolle selbst abwickelt.

Bei meinen bastelnden Enkelkindern erfolgte der Umstieg von Klebeband auf Lochband sehr schnell und es kam zu kindlich kreativen Lösungen, die Opa nie in den Sinn gekommen wären. Ohren und Schwänzchen an Holzstücken waren da noch die einfachsten Kreationen. Achslager für das Holzauto oder den Vollernte-Anhänger hinter dem Plastiktreyer waren da schon anspruchsvoller. Mein Enkelsohn brachte es auf die Spitze, indem er sich nach den ersten positiven Erfahrungen gleich einen Vorrat an kleinen Winkeln und unterschiedlich langem Stangenmaterial anlegte, um mit Schrauben, Scheiben und Muttern komplexere Gebilde zusammenschrauben zu können. Das erinnerte mich sehr stark an meine Kindheit und den Bau mit dem Metall-Stabilbaukasten. Auch der Batteriehalter **5** für einen selbst gebauten Experimentierkasten wurde aus Lochblech gefertigt. Bald kam auch die Forderung der Kleinen nach stabileren Blechstreifen, denn die dünnen ummantelten Lochblechstreifen waren für größere Projekte zu labil. Allerdings ist bei stabilerem Band, beispielsweise Rispenband mit Blechstärken über einem Millimeter, das Ablängen für Kinder und die Verletzungsgefahr an Bleckecken und Graten besonders zu bedenken! Da muss Opa unterstützen oder Arbeitsschutz schulen.

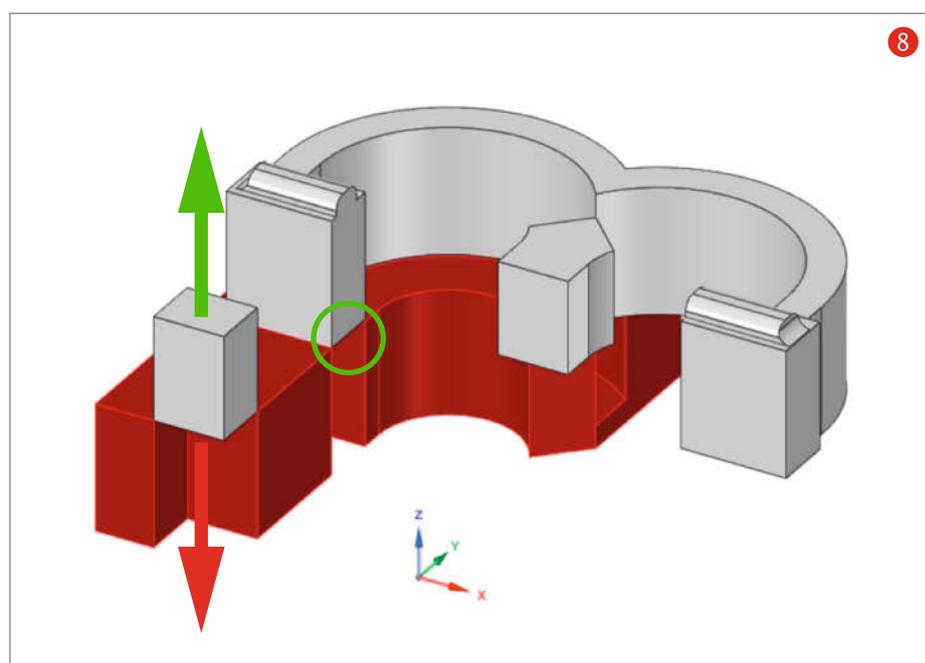
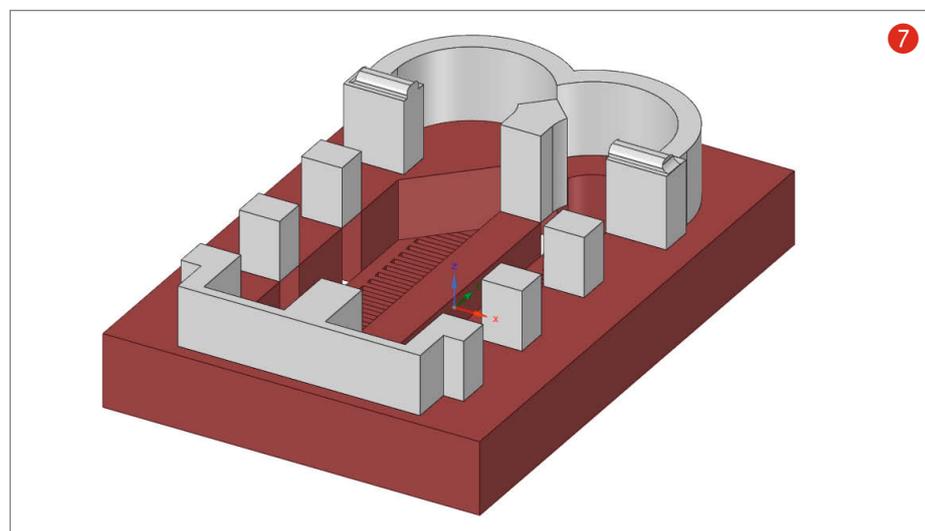
Hans-Juergen Pretzel

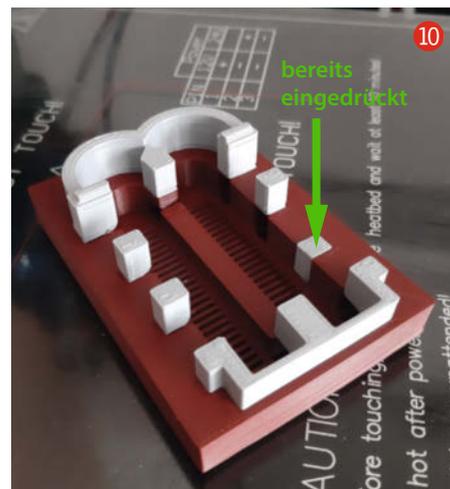
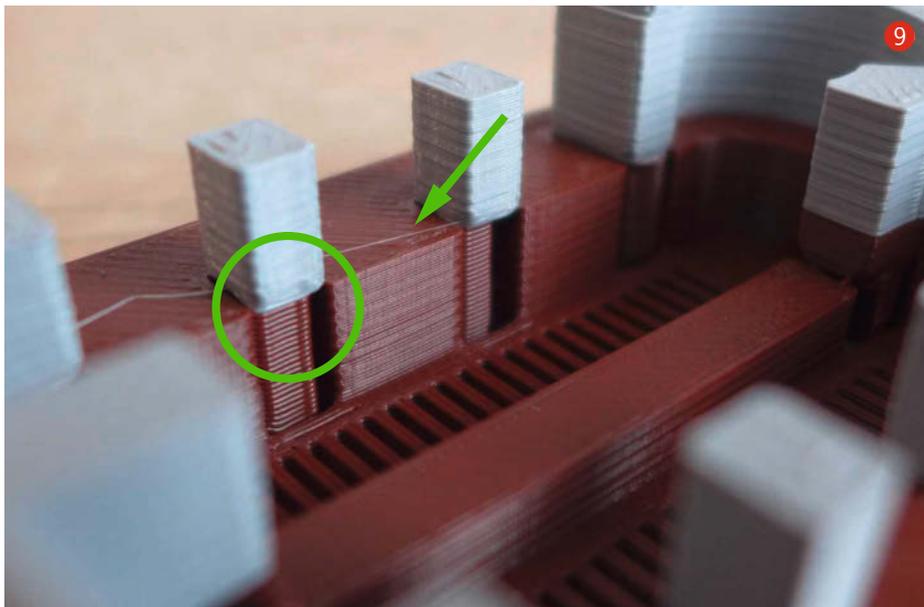
## Print-In-Place: Drucken und Drücken

Es sollte unsere örtliche Kirche in CAD konstruiert und dann 3D-gedruckt werden. Hierbei ergaben sich mehrere Herausforderungen, deren Lösungen ich Ihnen hier gerne vorstellen möchte.

Zur Verfügung stand ein Portaldruker mit Direkt-Extruder, beides ein Eigenbau. Als Konstruktionssoftware kam *DesignSpark Mechanical* zum Einsatz und als Slicer *Simplify3D*. Andere CAD- und Slicer-Software sollte aber genauso gut funktionieren.

Im Rahmen meines Projekts *Kirche* stellte sich mir bei dem abgebildeten Schallfenster **6** die Frage, ob mir eine Umsetzung mittels 3D-Druck im Rahmen meiner Möglichkeiten gelingen könnte. Dabei ging es konkret um





G1 X0 Y0 ;Extruder Wechselpos.  
M300 ;Druck stoppen  
M0 Continue? ;Druck fortsetzen



die Farbe der unterschiedlichen Gesteinsarten der Fassade, der Fensterlaibung sowie der vorstehenden Fassadenelemente.

Ein Filamentwechsel nach einer Druckunterbrechung würde für die äußere Fassade genügen, die Laibung mit ihren Materialwechseln aber nicht berücksichtigen. Hier könnten zwar separate Teile gedruckt und an den entsprechenden Positionen eingefügt werden. Die große Anzahl der Einzelteile und der damit verbundene Aufwand machte diese Lösung für mich indes wenig attraktiv, zumal das Schallfenster im Glockenturm insgesamt viermal benötigt wird.

Mein Ziel war es, das ganze Fenster mit seiner Geometrie und Farbgebung in einer CAD-Datei zu beschreiben und bis auf die Säule quasi in einem Stück zu drucken.

Anhand eines Referenzbildes im CAD wurde eine zweidimensionale Abbildung mit allen Details erzeugt und durch Extrudieren in ein 3D-Modell (7) umgewandelt. Die andersfarbigen Bereiche wurden dann aus dem Grundkörper herausextrudiert (8) und über die nun offenen Stellen der Fassade als Volumenkörper in die Höhe gezogen (grüner Pfeil). Die extrudierte Höhe entspricht dabei der Tiefe der Laibung.

Da die einzudrückenden Teile maßgleich (8) Kreis) über die darunterliegende Öffnung gedruckt werden, ist eine ausreichende Haftung gegeben. Je nach der Größe der offenen Stellen können jedoch Stützstrukturen erforderlich sein (9) Kreis).

Der Druckvorgang wurde mit Erreichen der Fassadenoberfläche (9) Pfeil) angehalten, das Filament und damit die Farbe wurde gewechselt. Bis zum Austritt der neuen Farbe wurde das Filament händisch in den Extruder geschoben. Danach wurde der Druck mit der zweiten Farbe fortgesetzt. Dabei wird folgender G-code benutzt:

Nach dem Entfernen eventuell nötiger Stützstrukturen können die extrudierten Teile in die Fassade gedrückt werden (8) Pfeil). Eine glatte Zange oder ein kleiner Schraubstock können hier helfen. Da es sich um eine Maß-auf-Maß-Presspassung handelt, ist die Verbindung sehr fest und es bedurfte keinerlei Verklebung.

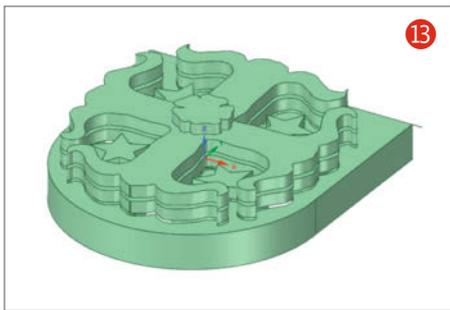
Die Bilder (10) und (11) zeigen das Fenster nach dem Drucken und im fertigen Zustand. Das Bild (12) zeigt das bisherige Ergebnis meines Kirchen-Projekts nach dem Einbau der vier Schallfenster.

Aufgrund der positiven Erfahrung habe ich diese Technik auch für ein anderes Projekt zum Einsatz gebracht: der Druck eines vierfarbigen Wappens in ebenfalls nur einer Druckdatei. Auch hierfür wird wie oben beschrieben ein 3D-Modell mit allen Elementen (13) in einer Druckdatei erstellt. Jedem Element wird dabei ein eigener farbiger Layer (14) zugeordnet.

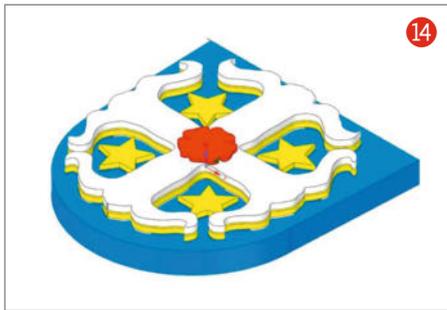
Da die Schichthöhe, im Gegensatz zum obigen Kirchenmodell mit seinen vorstehenden Elementen, kein besonderes Merkmal darstellt, kann sie relativ gering gehalten werden. Um eine sichere Überdeckung der jeweils unteren Farbschicht zu erreichen, sollte sie jedoch mindestens drei Layer betragen.

Falls ein plastischer Effekt gewünscht ist, können auch mehrere Schichten für jede Farbe erforderlich sein. Die für den Farbwechsel vorgesehenen Layer werden, wie bei dem vorbeschriebenen Schallfenster, im Slicer ermittelt und im G-Code eingestellt.

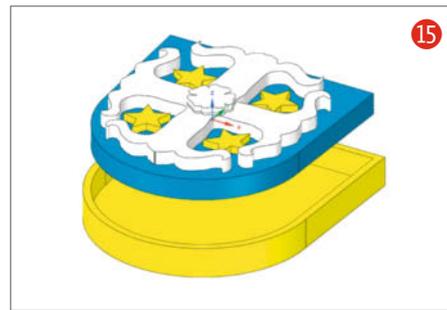
Falls das Wappen irgendwo eingesetzt oder aufgeklebt werden soll, bedarf die Rückseite keiner weiteren Behandlung. Ist indes eine glatte, ebene Unterseite gewünscht, empfiehlt sich das Drucken einer gesonderten Kasette (15), wobei deren Umrandung gleichzeitig eine Begrenzung der Eindringtiefe (16) darstellt.



13



14



15

Nach dem Drucken sind die unterschiedlichen Layer 17 noch erkennbar. Nach dem Eindrücken sind dann nur noch die gewünschten originalen Wappenfarben 18 sichtbar.

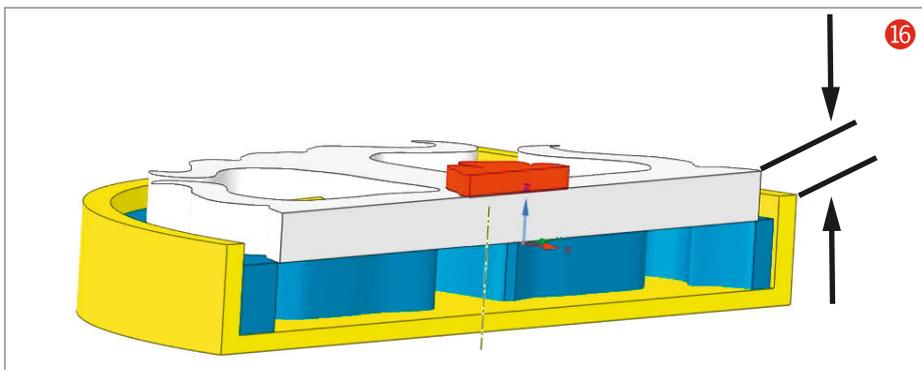
Zum Schluss noch zwei Tipps, die sich aus diesem Projekt ergaben:

Da für ein Wappen oder ein vergleichbar kleines Objekt nur wenig Filament benötigt wird, genügt es, wenn man kurze Filamentreste oder -proben in den gewünschten Farben verwendet. Um die Handhabung der lose gewickelten Filamente zu erleichtern, halte ich diese mit Klemmen 19 zusammen.

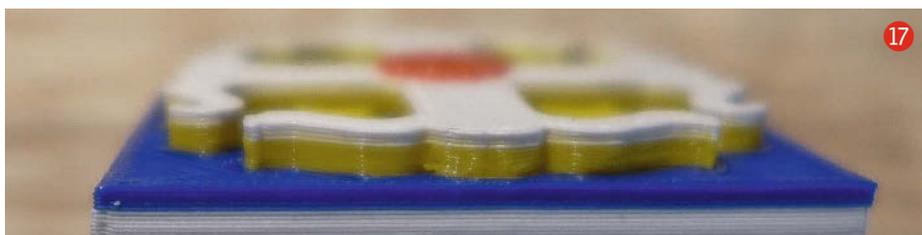
Da das beschriebene Verfahren viele Filamentwechsel erfordern kann, habe ich, um das Einfädeln in den Extruder zu erleichtern, die Enden mit einem Minenanspitzer 20 angespitzt 21.

Manfred Maroszek

—caw



16



17



18



20



21

Alles zum Artikel im Web unter [make-magazin.de/xsaj](https://make-magazin.de/xsaj)



19

### Machen Sie mit!

Kennen Sie auch einen raffinierten Trick? Wissen Sie, wie man etwas besonders einfach macht? Wie man ein bekanntes Werkzeug oder Material auf verblüffende Weise noch nutzen kann? Dann schicken Sie uns Ihren Tipp – gleichgültig aus welchem Bereich (zum Beispiel Raspberry, Arduino, 3D-Druck, Elektronik, Platinenherstellung, Lasercutting, Upcycling ...).

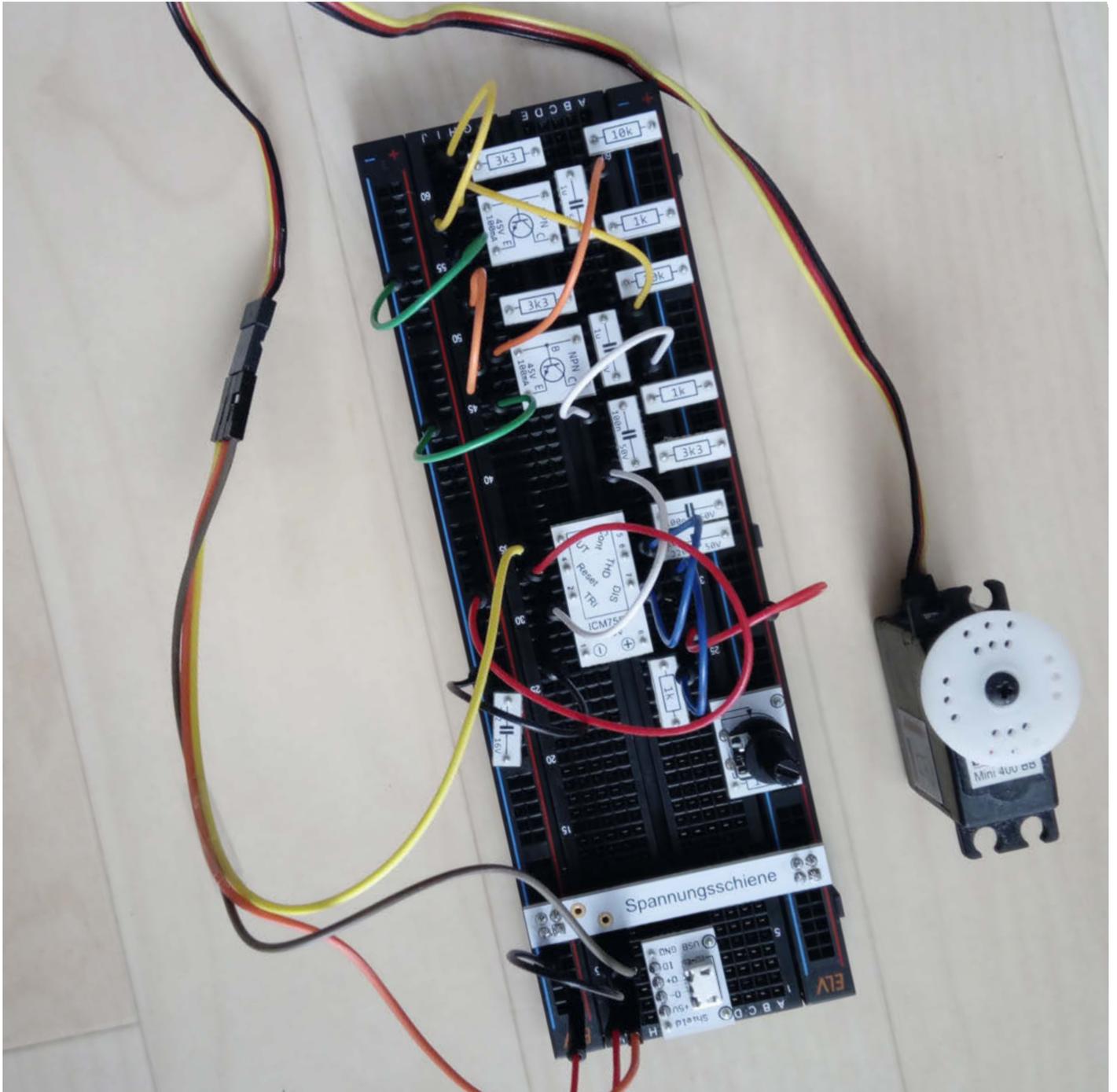
Wenn wir Ihren Tipp veröffentlichen, bekommen Sie das bei Make übliche Autorenhonorar. Schreiben Sie uns dazu einen Text, der ungefähr eine Heftseite füllt und legen Sie selbst angefertigte Bilder bei. Senden Sie Ihren Tipp mit der Betreffzeile *Lesertipp* an:

► [mail@make-magazin.de](mailto:mail@make-magazin.de)

# Experimentierset: Servo-Tester selber bauen

Modellbau-Servos werden mit einer Pulsweitenmodulation gesteuert. Wie das funktioniert und wie man einen Servo auch ohne Mikrocontroller auf Funktionalität überprüft, zeigen wir mithilfe des Experimentiersets aus dem Make-Elektronik-Special 2021.

von Michael Gaus und Miguel Köhnlein



**B**evor man einen Servo in ein Projekt einbaut, ist es ratsam, diesen auf seine Funktion hin zu überprüfen. Das erspart Zeit bei einer späteren Fehlersuche und ist mit einem handelsüblichen Servo-Tester **1** schnell erledigt. Mit ihm lässt sich herausfinden, ob die Winkel ordnungsgemäß angesteuert werden können oder ob es Probleme mit der Reaktionszeit gibt. Hat man ein solches Gerät gerade nicht zur Hand, kann man es sich auch aus ein paar Einzelkomponenten zusammenbauen, die man zum Beispiel im Make Experimentierset findet. Aus welchen Komponenten der Aufbau besteht und was einen Servo veranlasst, einen bestimmten Winkel anzusteuern, zeigen wir in diesem Artikel.

### Servos und PWM

Servo-Motoren werden mithilfe sich wiederholender digitaler Pulse gesteuert. Diese bestimmen die Ausrichtung bzw. Position der Servo-Welle. Üblicherweise beträgt die Wiederholrate (Periode) eines solchen Pulses 20ms. Innerhalb dieser Periode wird eine High-Zeit zwischen ca. 1ms (Minimalposition) und ca. 2ms (Maximalposition) erzeugt. Auf einen 180-Grad-Servo übertragen entspricht dessen Minimalposition 0 Grad und die Maximalposition 180 Grad. Bei einer High-Zeit von ca. 1,5ms ist die Mittelstellung mit 90 Grad erreicht. Über die Veränderung dieser sogenannten Pulsbreite oder -weite wird also der Winkel gesteuert, auf den der Servo-Arm gestellt werden soll. Wir sprechen deshalb von einer Pulsweitenmodulation (PWM) **2**.

Mikrocontroller können solche einstellbaren PWM-Signale erzeugen. Es geht jedoch auch ohne: Mit einem astabilen Multivibrator und einem nachgeschalteten Mono-Flop lässt sich schnell ein Servo-Tester bauen, bei dem die Pulsbreite über ein Potentiometer einstellbar ist.

### Funktionsweise und Aufbau

Die Wiederholrate des Servo-Steuersignals erzeugt ein Multivibrator **3**, bestehend aus den Transistoren T1 und T2, den Kondensatoren C1 und C2 sowie ein paar Widerständen. Sollten Sie nicht das genannte Experimentier-



**1** Im Handel erhältlich: ein einfacher Servotester

## Kurzinfo

- » Servo-Tester aus Elektronikkomponenten bauen
- » PWM-Signal selbst erzeugen
- » Servos ohne Mikrocontroller steuern

### Checkliste



**Zeitaufwand:**  
1 Stunde



**Kosten:**  
10 bis 45 Euro

### Dieses Vorwissen ist hilfreich

- » Wie ein Transistor funktioniert
- » Wie ein astabiler Multivibrator funktioniert
- » Wie ein ICM7555-Timer funktioniert

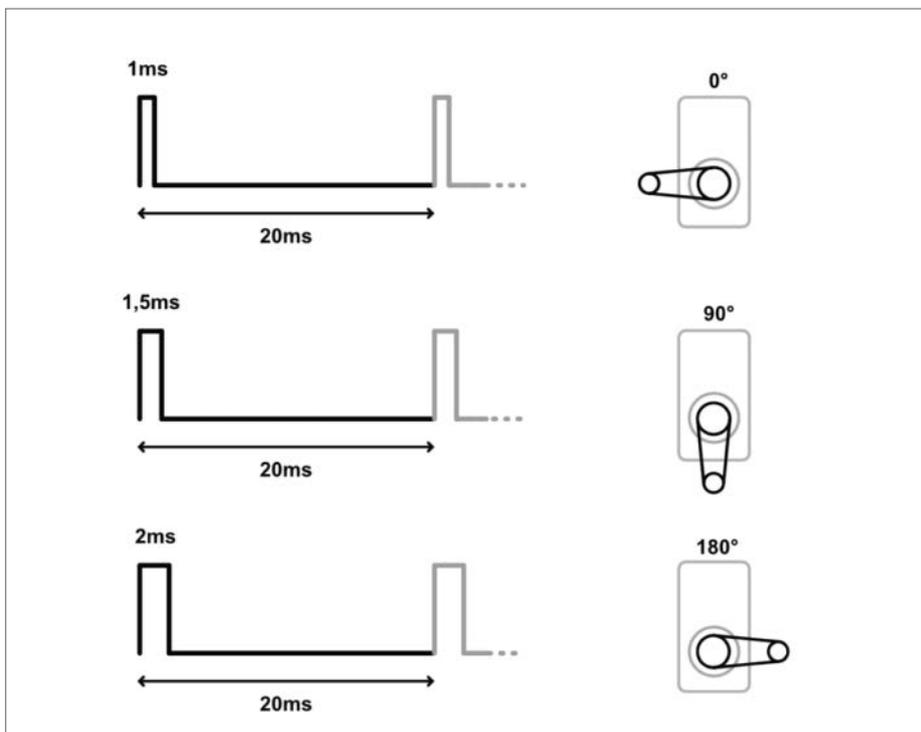
### Material

- » Steckkabel
- » Breadboard
- » 1 x Micro-USB-Anschluss
- » 1 x Spannungsschiene
- » 1 x ICM7555
- » 2 x Widerstand 10kΩ
- » 3 x Widerstand 3,3kΩ
- » 3 x Widerstand 1kΩ
- » 1 x Potentiometer 10kΩ
- » 1 x Kondensator 2,2μF
- » 2 x Kondensator 1μF
- » 1 x Kondensator 220nF
- » 2 x Kondensator 100nF
- » 2 x NPN-Transistor
- » 1 x Modellbau-Servo

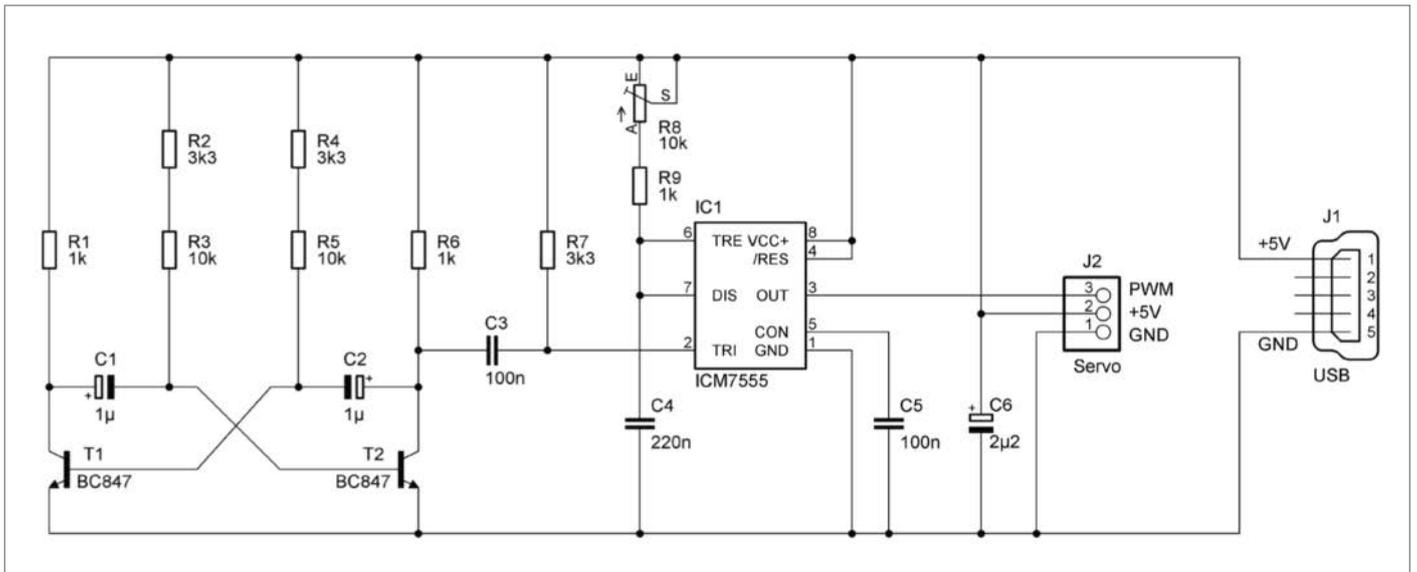
### Mehr zum Thema

- » Florian Schäffer, Servos für jeden Zweck, Make 3/18, S. 46
- » Thomas Wiemken, Astabile Kippstufe, Make-Special 2021, S. 52
- » Thomas Wiemken, Timer-IC ICM7555, Make-Special 2021, S. 76

Alles zum Artikel im Web unter [make-magazin.de/xe68](https://make-magazin.de/xe68)



**2** Pulsweitenmodulation (PWM) steuert die Ausrichtung eines Servos. Die Drehrichtung kann je nach Modell variieren.



3 Im Schaltplan links zu erkennen: der astabile Multivibrator

set verwenden und die 1µF-Elektrolyt-Kondensatoren aus Ihrer Bastelkiste einsetzen, achten Sie auf deren Polung (beim Set ist die nicht von Belang). Durch die symmetrische Dimensionierung der Bauteile hat das Ausgangssignal am Kollektor von T2 – je nach Bauteiltoleranz – ein ungefähr symmetrisches Tastverhältnis.

Wie groß die Widerstände innerhalb des Multivibrators sein müssen, um eine Periodendauer von 20ms zu erreichen, lässt sich aus der Halbwertszeit der Entladekurve des Kondensators ( $T = 2 \times \ln(2) \times R \times C$ ) wie folgt errechnen:

$$R = T / (2 \times \ln(2) \times C)$$

$$R = 20\text{ms} / (2 \times \ln(2) \times 1\mu\text{F}) \approx 14,427 \text{ k}\Omega$$

Um mit dem Experimentierset möglichst nah an diesen Wert zu gelangen, lassen sich

jeweils ein 10kΩ- und ein 3,3kΩ-Widerstand seriell schalten. Daraus ergibt sich immerhin eine Periodendauer von 18,4ms, wie die folgende Rechnung zeigt:

$$T = 2 \times \ln(2) \times R \times C$$

$$T = 2 \times \ln(2) \times (10\text{k}\Omega + 3,3\text{k}\Omega) \times 1\mu\text{F} \approx 18,4\text{ms}$$

Über den Kondensator C3 wird bei der negativen Flanke – wenn der Transistor T2 geschaltet ist – der Timer ICM7555 getriggert, der dann einen einzelnen positiven Impuls am Ausgang-Pin 3 erzeugt. Dessen Pulsbreite ist abhängig vom Kondensator C4 sowie dem gemeinsamen Widerstandswert von R8 und R9. Die High-Zeit des Pulses beträgt ca.  $1,1 \times (R8 + R9) \times C4$ . Je nach eingestelltem Wert des Potentiometers (R9) bedeutet das:

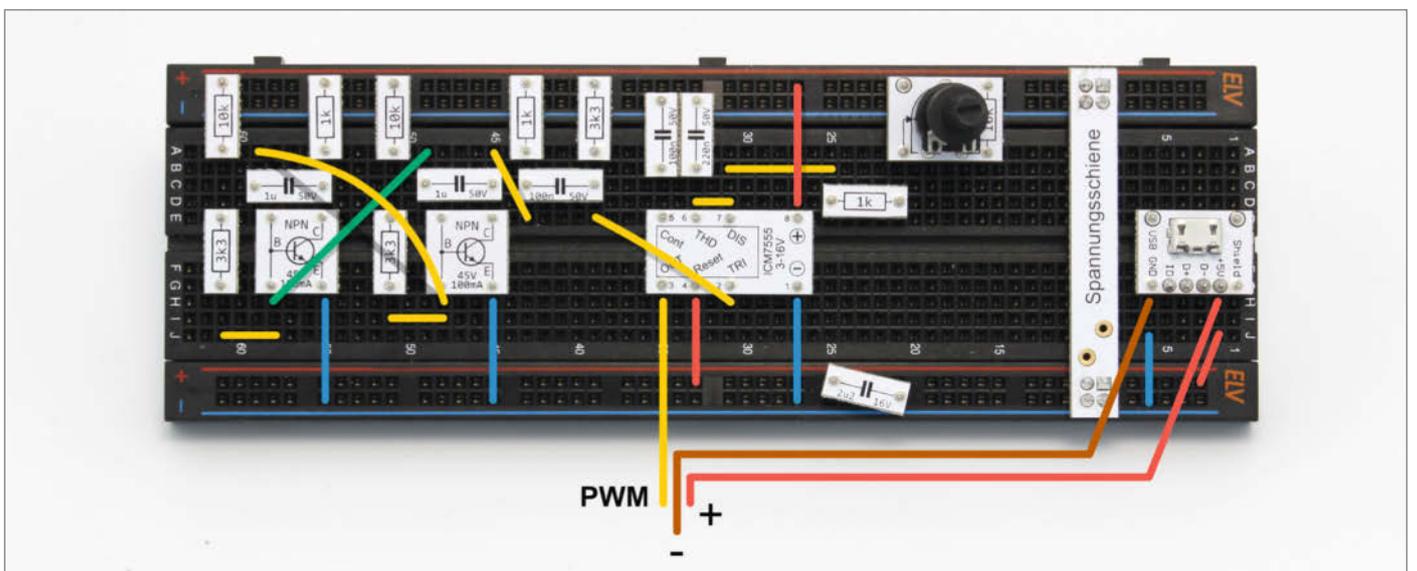
Bei einem Potentiometer-Widerstand von 0Ω

$$T = 1,1 \times 1\text{k}\Omega \times 220\text{nF} \approx 0,24\text{ms}$$

Bei einem Potentiometer-Widerstand von 10kΩ

$$T = 1,1 \times 11\text{k}\Omega \times 220\text{nF} \approx 2,66\text{ms}$$

Der übliche Wertebereich für das Servo-PWM-Signal von 1,0 – 2,0ms kann somit abgedeckt werden. Wenn man einen Servo testet, sollte jedoch das Poti nicht bis zum äußersten Anschlag gedreht werden, da der Servo ansonsten mechanisch überbelastet und beschädigt werden kann. Vor dem Einschalten der Versorgungsspannung ist es daher ratsam, das Poti etwa auf Mittelstellung zu drehen.



# Einstieg in die Elektronik

Wer schon immer wissen wollte, wie Transistoren arbeiten und wie Schaltungen damit funktionieren, findet im *Make Elektronik Special* Grundlagen, Aufbauanleitungen sowie Tipps und Tricks. Zusammen mit Redakteuren und Entwicklern von ELV haben wir das Heft konzipiert und produziert. Das Heft ist im Bundle mit dem praktischen Experimentierset für 44,95 Euro inkl. Versandkosten im heise shop zu kaufen (siehe Link).

Es führt Schritt für Schritt in die Grundlagen ein und erklärt beispielsweise, warum eine LED einen Vorwiderstand braucht und wie man ihn berechnet. Ein Großteil der Artikel im Heft ist aber dem Transistor gewidmet, wie man ihn als Schalter einsetzt oder in Kombination mit Mikrocontrollern größere Lasten wie Motoren oder Power-LEDs schaltet. Wir zeigen zudem, wie man mit Transistoren Signale verstärkt und Blink-, Tongenerator- und Intervallschaltungen baut.

Alle Schaltungen lassen sich mit dem Experimentierset nachbauen. Jede Schaltung ist als Schaltplan abgedruckt und als Breadboard-Aufbau abgebildet. Die Prototypen-Adapter (PAD) für das



Breadboard machen den Aufbau von Schaltungen zum Kinderspiel, denn man muss weder in Datenblättern nach der Belegung von Pins und Beinchen der elektronischen Bauteile suchen, noch muss man den Aufdruck mit einer Lupe entziffern, um den Wert des Elements herauszubekommen. Die PADs vereinen praktische Verkabelung und Übersichtlichkeit in einem.

Insgesamt 44 PADs für Widerstände, Kondensatoren, NPN-Transistoren, Doppelklemmen, Micro-USB-Buche, LEDs, Potentiometer, einem N-Channel-MOSFET, einem Relais, einem Piezosummer sowie einem ICM7555 liegen dem Set nebst Kabelbrücken und Breadboard bei. Zur Stromversorgung dient ein 5V-Ladenetzteil, wie es für Smartphones in jedem Haushalt verfügbar ist.

## Servos testen

Modellbau-Servos haben üblicherweise drei Anschlusspins: Einen für die Versorgungsspannung (rot), für die Masse/GND (braun) und die PWM-Signalleitung (gelb). Standardmäßig werden sie an einer 3-poligen Dupont-Buchse herausgeführt. Mit drei Steckkabeln des Typs male-male aus dem Experimentierset kann man sie mit dem Breadboard ver-

binden. Hat man jedoch einen Servo mit offenen Kabelenden, so kann man diese über die PAD-Adapter mit den Schraubklemmen anschließen. Als Stromversorgung genügt den meisten Modellbau-Servos ein Spannungsbereich zwischen ca. 4,8V–6V. In diesem Fall betreiben ein USB-Netzteil oder eine USB-Powerbank mit 5V die Schaltung. Die Stromversorgungs-Anschlüsse des Servos sollten am besten direkt am PAD mit der USB-Buchse

verbunden werden, um Signalstörungen zu minimieren.

Sobald der Tester und der Servo mit Strom versorgt sind, stellt sich die Servo-Welle auf den Winkel ein, den das PWM-Signal vorgibt. Verstellt man anschließend das Poti, sollte sich der Servo ebenfalls drehen und man kann überprüfen, ob das Ansteuern der Winkel ordnungsgemäß funktioniert. —akf



# Kleiner Pico – großer Spaß!

Exklusiv im heise Shop

Mit dem **Make Special Pi Pico** steigen Sie ein in die Welt der Programmierung von ARM-Mikrocontrollern. Make zeigt in dem 64-seitigen Special, welche Entwicklungsumgebungen es für den Raspberry Pi Pico gibt, wie man sie installiert und wie man sie nutzt:

- ▶ Der Pi Pico programmieren mit Basic, MicroPython oder C++
- ▶ Sensoren und Mini-Displays nutzen
- ▶ NeoPixel per PIO-Funktion kontrollieren
- ▶ inkl. Raspberry Pi Pico, Modell RP2040 Mikrocontroller-Board

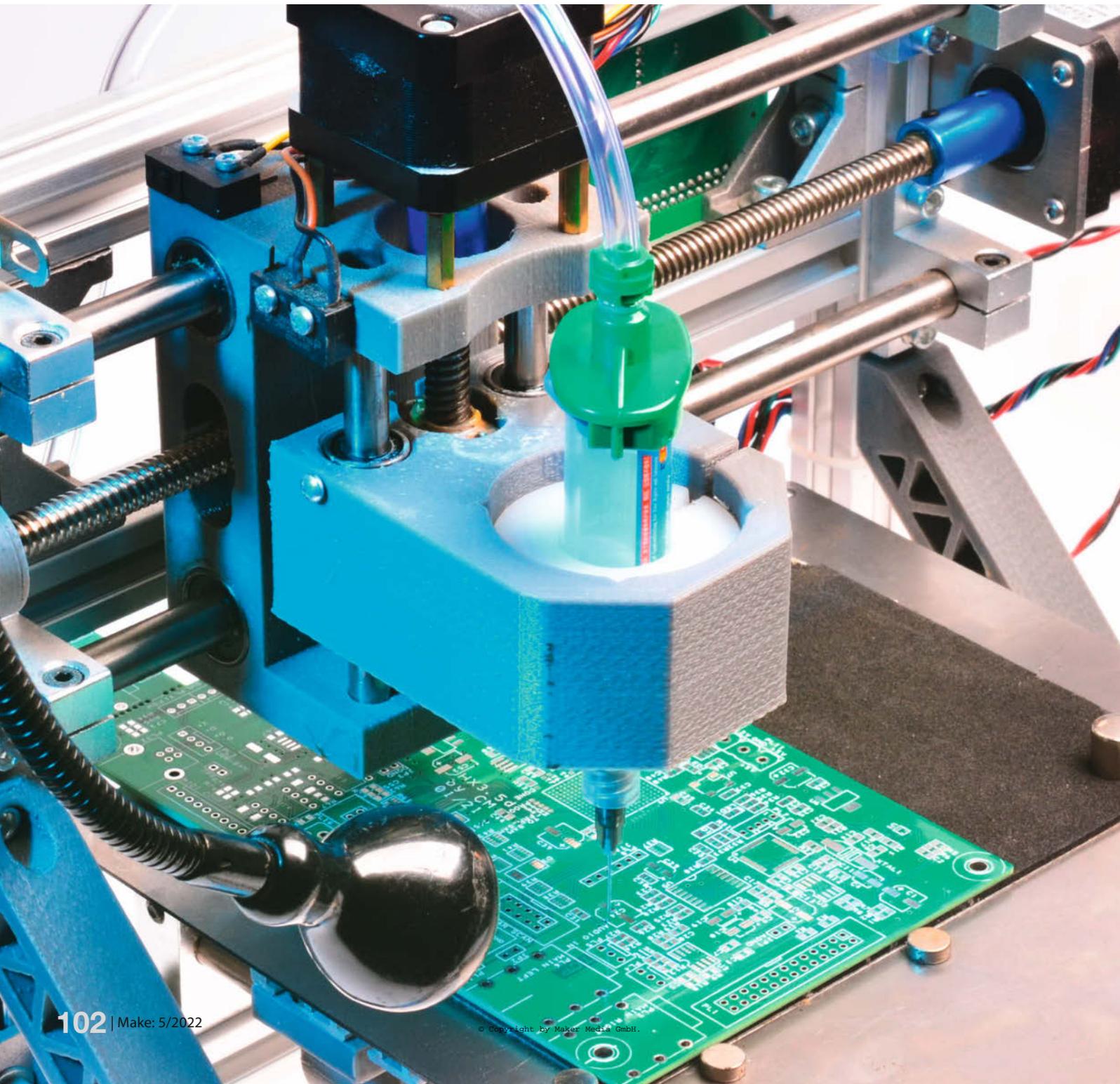
Heft + Raspberry Pi Pico für 24,95 €

[shop.heise.de/make-pico](https://shop.heise.de/make-pico)

# Statt Schablone: Lötpasten-Dispenser

Teure Edelstahl-Pastenschablonen und die unvermeidliche Sauerei beim Auftragen erspart man sich mit unserem DIY-Dispenser, den wir aus einer kleinen CNC-Fräse mit minderem Gebrauchswert erbastelt haben: So macht SMD-Bestücken wieder Spaß!

von Carsten Meyer



**W**er als Maker nicht nur Einzelstücke, sondern auch mal eine Platinen-Kleinserie auflegt, kommt um einen unbeliebten Arbeitsschritt nicht herum: Das Auftragen der SMD-Lötpaste mit Edelstahl-Schablone und Spachtel schmutzt und ist nicht jedermanns Sache – von den Kosten für die (möglicherweise nur wenige Male benutzte) Schablone ganz abgesehen. Verschmierte Schablonen müssen nach kurzer Verwendung sorgfältig gereinigt werden, sonst gibt es unscharfe Löt pads und einen ungleichmäßigen Pastenauftrag.

In unserer Make-Werkstatt stand seit vielen Monaten eine kleine Billig-CNC-Fräse namens *CNC1610* herum, die wegen eines eiernden Spindelmotors und der eher labilen Konstruktion niemand ernsthaft benutzen wollte. Das brachte uns auf die Idee, statt des Motors eine Lötpasten-Dosierspritze einzuspannen, sodass der Pastenauftrag programmgesteuert erfolgen kann.

## Dosieren mit Druckluft

Bei AliExpress und einigen eBay-Händlern findet man das Dosiergerät *AD-982*, das mit Druckluft arbeitet; solche Geräte verwendet man sonst gern zum Klebstoff- und Lackauftrag. Über Taster oder Fußschalter löst man einen definierten Druckluft-Impuls aus, der ein Tröpfchen aus der über einen Adapter-schlauch angeschlossenen Spritze presst.

Für verschiedene Viskositäten lässt sich der Luftdruck und die Impulszeit in weiten Grenzen einstellen; lediglich eine bescheidene Druckluftquelle (ein Airbrush-Kompressor mit min. 4 bar reicht völlig) ist zusätzlich erforderlich. Der *AD-982* lässt sich zudem auf manuellen Betrieb umstellen; die Dosierzeit wird dann nicht mehr vom internen Timer vorgegeben, sondern von der Länge des Tastendrucks.

Der *AD-982* ist mit rund 80 Euro preiswerter als die Beschaffung seiner Einzelteile (Druckregler, Manometer, 2/3-Wege-Ventil, Timer, Stromversorgung) kosten würde, zudem sind im Lieferumfang gleich Schläuche und Spritzenadapter enthalten. Für eine programmgesteuerte Dosierung benötigt man nur noch ein kleines 12V-Relais, das anstelle des Motors an die Steuerplatine der Mini-Fräse angeschlossen wird. Die Relais-Kontakte werden dann mit dem Taster-Eingang (3,5mm-Klinkenbuchse) verbunden, und schon kann man den Dosiervorgang über die G-Code-Befehle *M3/M5* (Spindelmotor ein/aus) und gegebenenfalls einem Pausen-Befehl (*G4 PX.XXX* mit *X.XXX* = Zeit in Sekunden) steuern.

## Fräsen-Umbau

Die Fräse muss, sofern noch nicht geschehen, zunächst mit Endschaltern (geeignet sind Miniatur-Mikroschalter) für die Achsen nach-

## Kurzinfo

- » Automatische Tröpfchen: Dosieren mit Druckluft
- » Umbau einer CNC-Billigfräse zum Lötpasten-Dispenser
- » Ansteuerungs-Software mit Gerber-Import

## Checkliste



**Zeitaufwand:**  
4 Stunden



**Kosten:**  
90 Euro (ohne CNC-Fräse)

## Material

- » *AD-982* Dispenser
- » Bausatz-Fräse *CNC1610* oder *MaXYposi*
- » Kleiner Kompressor min. 4 bar
- » Lötpaste in Spritze
- » Adapterring 44,5mm für Spritze (aus POM gedreht oder 3D-gedruckt)

## Mehr zum Thema

- » Carsten Meyer, Pneumatik: Grundlagen für Maker, *Make 2/2022*, S. 44
- » Carsten Meyer, *MaXYposi*, *Make 1/2017*, S. 12
- » Carsten Meyer, *MaXYposi* bestückt Platinen, *5/2018*, S. 120
- » Video: Lötpasten-Dispenser aus Billig-CNC-Fräse



Alles zum Artikel im Web unter [make-magazin.de/xwmg](https://make-magazin.de/xwmg)

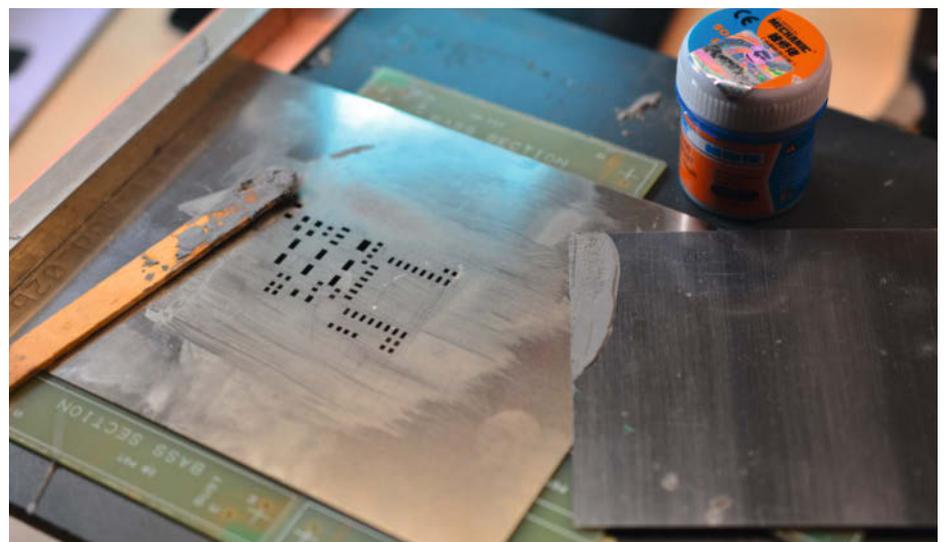


gerüstet werden, damit ein *Homing* (Anfahren des Maschinen-Nullpunkts) ausgelöst werden kann; Anschlüsse dafür sind bei unserer *CNC1610*-Steuerplatine auf Stiftleisten geführt (siehe Bild).

Die Steuerung unserer Fräse arbeitet übrigens mit der beliebten GRBL-Firmware, sodass eine Einrichtung über ein Terminal-Programm

problemlos möglich ist. Für unser mit nur 12V betriebenes Gerät fanden wir folgende optimalen GRBL-Einstellungen (erreichbar über ein Terminal-Programm mit dem GRBL-Befehl \$\$):

Hierfür müssen die Endschalter links, hinten und an der Z-Achse oben montiert werden, die Home-Position ist dann (durch den beweg-



Zum Auftragen der SMD-Lötpaste auf die zu bestückende Platine benötigt man normalerweise eine Schablone und eine einfache Halterung.



Der Universal-Dispenser AD-982 benötigt zur Funktion einen kleinen Airbrush-Kompressor mit mindestens 4 bar.

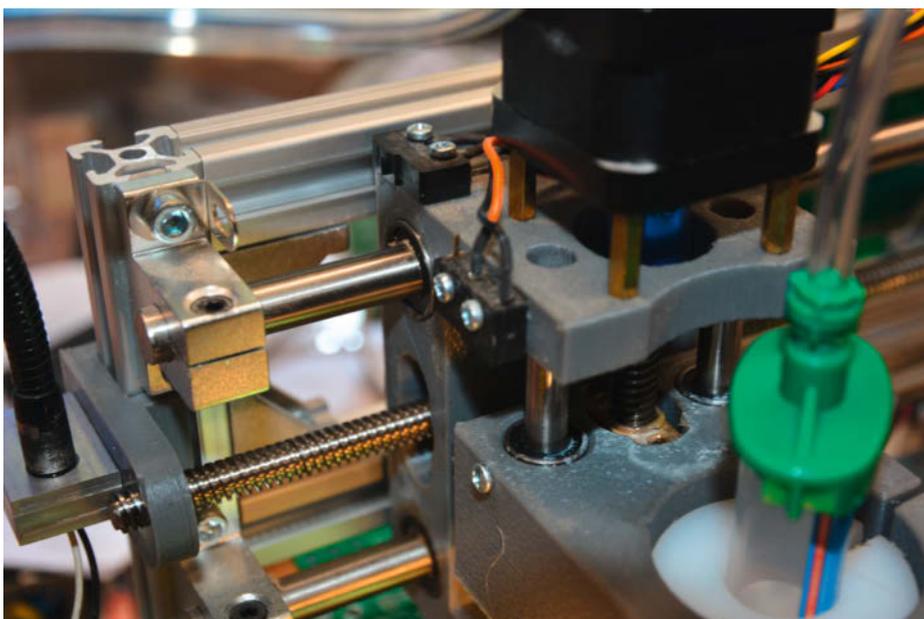
ten Tisch) links vorn. Auf den Maschinentisch schraubten wir eine 1mm-Stahlblechplatte; die zu bearbeitende Platine kann dann sehr leicht mit kleinen Magneten fixiert werden.

Für die Einspannung der Spritze haben wir auf der Drehbank einen 20mm hohen Adapterring aus POM-Rundmaterial gedreht, der den Spritzendurchmesser von 18,5mm auf den Motorträger-Durchmesser von 44,5mm adaptiert; selbstverständlich kann man den Adapterring auch auf einem 3D-Drucker anfertigen. Ein Schlitz im Adapterring erlaubt das Einspannen

der Spritze mit der Klemmschraube für den Motor.

## Ansteuerung

Für die Ansteuerung unseres CNC-Dosierers benötigt man ein Programm, das die Pad-Positionen aus den vom Layout-Programm ausgegebenen Gerber-Daten in G-Code-Befehle für die GRBL-Steuerung umsetzt und mit 115200 Bd/8n1 an den virtuellen COM-Port ausgibt – die Fräsensteuerung arbeitet wie



Die CNC1610 ist ab Werk nicht mit Endschaltern ausgerüstet, diese müssen nachträglich installiert werden.

viele Arduino-Clones aus China mit dem USB-Seriell-Umsetzer CH340.

Die Gerber-Schablonendatei enthält ganz zu Anfang eine Format-Spezifikation und eine Liste mit den verwendeten „Blenden“, also den Pad-Abmessungen – ein Relikt aus Zeiten der CNC-Fotobelichter. Der Blendenliste folgen simple XY-Positionen; bei Pads sind sie mit dem Suffix D03\* (Flash-Belichtung) abgeschlossen.

Wir verwenden die Pad-Abmessungen, um bei größeren Pads durch Verlängern des Druckluft-Impulses mehr Paste aufzutragen und gegebenenfalls auf mehrere Punkte zu verteilen, sodass auch größere Bauteile (z.B. Spannungsregler) eine ausreichende Menge Lötpaste erhalten.

## Dosier-Software

Wegen ihrer äußerst mächtigen Listen- und Tabellenfunktionen wählten wir für die Programmierung die *Delphi*-Entwicklungsumgebung von *Embarcadero*, die als Community Edition für Maker kostenlos heruntergeladen werden kann (siehe Link). Unser Dosier-Programm importiert nicht nur Gerber-Pastendaten, sondern kann auch einfache SMD-Patterns (etwa für SO- und BGA-Gehäuse) selbst anlegen. Das ist äußerst praktisch, wenn zum Beispiel ein BGA-Chip nach dem Auslöten nachbestückt werden muss; ein Pastenauftrag mit Schablone ist dann ja nicht mehr möglich, und niemand möchte 196 oder 256 Pastenkleckse von Hand aufbringen.

Nach einigen Experimenten fanden wir einen optimalen Dosier-Algorithmus: Zunächst fährt die Nadel mit etwa 2mm Z-Abstand an das Pad heran und es wird ein kurzer Druckluft-Impuls (50 bis 200ms, abhängig von der Pad-Größe) ausgelöst, sodass sich an der Kanülenspitze ein kleines Pasten-Tröpfchen bildet. Dann wird die Spitze mit Z-Höhe 0 aufgesetzt und rasch wieder nach oben gezogen; das Tröpfchen bleibt dann am Pad kleben. Bei größeren Pads (> 2,5mm) werden zwei oder vier Tröpfchen auf dem Pad verteilt.

Stellen Sie nach dem Start zunächst die Schnittstelle und deren Parameter ein (Button *Setup*). Für GRBL 0.9 verwenden Sie die *Baudrate* 115200, 8 *Data Bits*, 1 *Stop Bit*, kein *Parity* und kein *Flow Control*. Das Programm ist auch mit der neueren GRBL-Version 1.1 kompatibel, nutzt aber dessen luxuriöse Jog-Funktionen nicht. Nach dem anschließenden *Connect* führt die Software ein automatisches *Homing* der Maschine durch, der Maschinen-Nullpunkt ist damit bekannt.

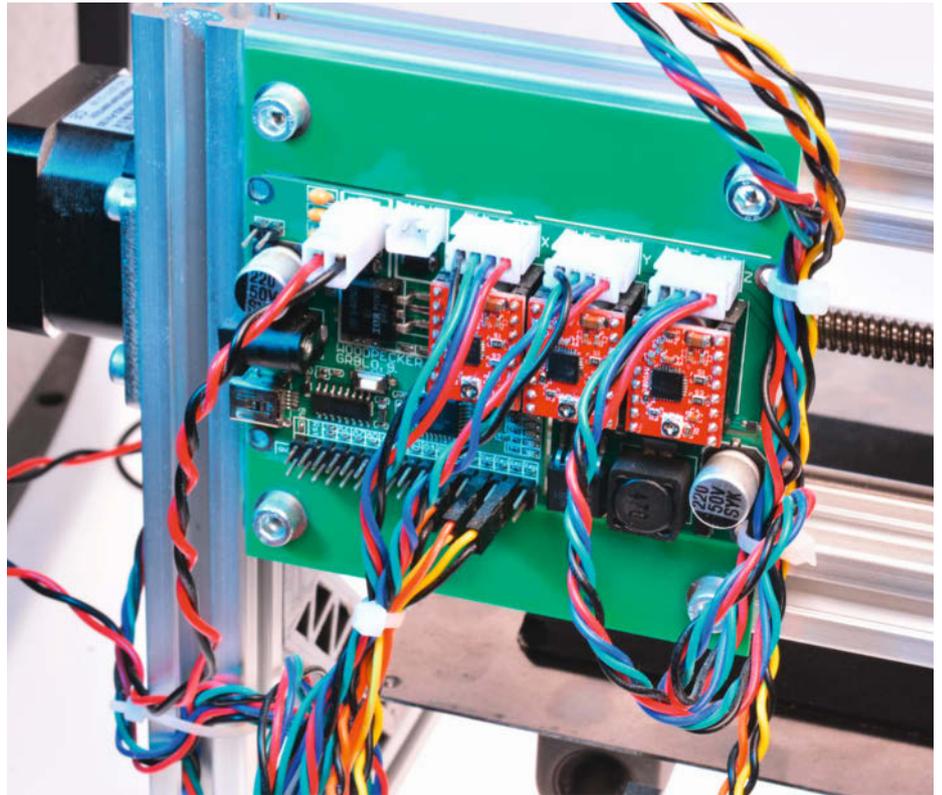
## Platinen-Nullpunkt setzen

Wichtig ist, dass man vor dem Start den Nullpunkt der Platine einstellt. Da die Gerber-Pad-Daten möglicherweise einen (unbekannten) Offset aufweisen, ermittelt unser Programm

## GRBL-Einstellungen

```

$0=10 (step pulse, usec)
$1=25 (step idle delay, msec)
$2=0 (step port invert
mask:00000000)
$3=5 (dir port invert mask:00000101)
$4=0 (step enable invert, bool)
$5=0 (limit pins invert, bool)
$6=0 (probe pin invert, bool)
$10=3 (status report mask:00000011)
$11=0.010 (junction deviation, mm)
$12=0.002 (arc tolerance, mm)
$13=0 (report inches, bool)
$20=0 (soft limits, bool)
$21=0 (hard limits, bool)
$22=1 (homing cycle, bool)
$23=3 (homing dir inv mask:00000011)
$24=100.000 (homing feed, mm/min)
$25=750.000 (homing seek, mm/min)
$26=100 (homing debounce, msec)
$27=1.000 (homing pull-off, mm)
$100=800.000 (x, step/mm)
$101=800.000 (y, step/mm)
$102=800.000 (z, step/mm)
$110=1500.000 (x max rate, mm/min)
$111=1500.000 (y max rate, mm/min)
$112=1500.000 (z max rate, mm/min)
$120=300.000 (x accel, mm/sec^2)
$121=300.000 (y accel, mm/sec^2)
$122=300.000 (z accel, mm/sec^2)
$130=151.000 (x max travel, mm)
$131=101.000 (y max travel, mm)
$132=36.000 (z max travel, mm)
    
```



Für den Endschalter-Anschluss ist bei der CNC1610-Steuerelektronik eine Stiftleiste vorgesehen.

nach dem Laden der Pad-Datei zunächst das am weitesten links unten liegende Pad (in der Grafik-Darstellung mit *Ref Zero* gekennzeichnet) und verwendet dessen Koordinaten als Ausgangspunkt. Dieses Pad fährt man nun mit den Jog-Tasten exakt mittig an, sodass die Kanülenspitze gerade an der Oberfläche kratzt, und setzt dann mit *All Zero* den Nullpunkt.

### Papier-Trick

Der Z-Nullpunkt lässt sich mit einem Blatt Schreibpapier zuverlässig ermitteln: Das Papier auf die Platine legen und die Dosier-nadel mit den Z-Jog-Buttons vorsichtig herunterfahren, bis das Papier ganz leicht eingeklemmt wird. Die Z-Höhe beträgt dann 0,1mm. Dann das Papier wegziehen, weitere 0,1mm nach unten gehen und den Nullpunkt setzen.

Sie können in der Pad-Positionsliste aber auch ein anderes Pad als Referenz auswählen, wenn etwa wie im Bild das Pad links unten zu groß ist, um exakt seine Mitte bestimmen zu können. Das in der Liste aktivierte Pad wird in der Grafik rot hervorgehoben. Dann genau dieses Pad auf der Platine anfahren, mit der rechten Maustaste auf den Listeneintrag klicken und im Pop-up-Menü den Eintrag *Set Zero Reference (Nozzle is at this Pad)* auswählen. Der Platinen-Nullpunkt wird dann automatisch berechnet. *All Zero* darf dann nicht mehr betätigt werden!

Damit Sie nicht viele hundert Pads auswählen müssen, um das passende zu finden, kann man in der *Apertures*-Liste eine Vorauswahl treffen. Klicken Sie hier auf eine Zeile mit einer

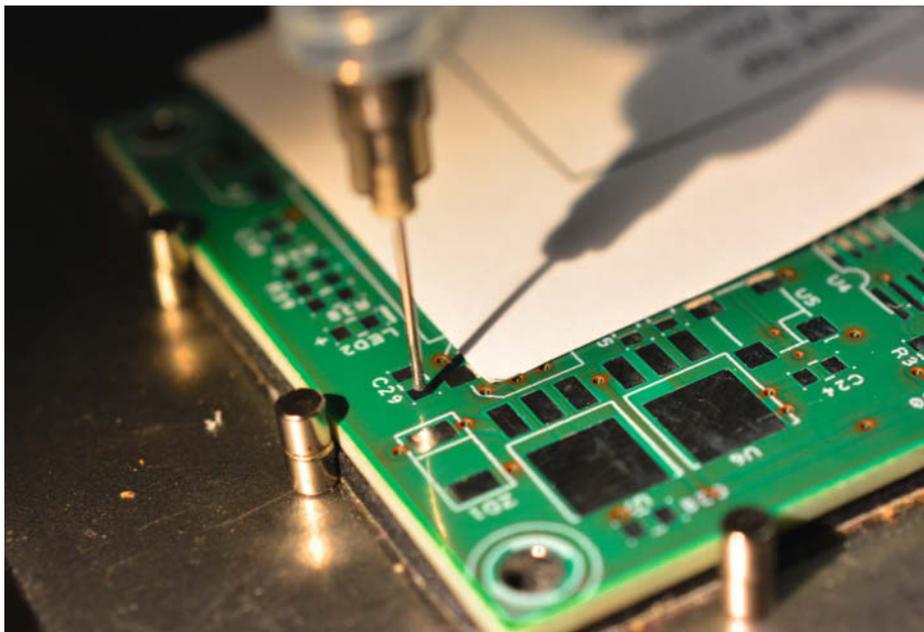
kleine Blende, und die aktive Zeile der Pad-Positions-Liste springt dann automatisch auf das erste gefundene Pad mit ebendieser Blende (bzw. diesen Abmessungen). Für bestimmte

The screenshot shows the software interface for a CNC machine. It includes a status bar at the top with coordinates and machine status. Below that are jog controls for X, Y, and Z axes, along with buttons for 'All Zero', 'X Zero', 'Y Zero', and 'Z Zero'. There are also buttons for 'Run', 'Abort', 'Home', and 'Goto Zero'. A 'Gerber Apertures' table is visible on the right, and a 'Gerber Pad Positions' table is at the bottom right. The 'Gerber Pad Positions' table has a 'Zero Ref' marker on the leftmost pad.

| #   | DX   | DY   | Type   |
|-----|------|------|--------|
| D11 | 1.05 | 1.30 | Rect   |
| D13 | 1.50 | 2.49 | Rect   |
| D14 | 0.20 | 0.20 | Circle |
| D10 | 1.93 | 0.53 | Rect   |
| D12 | 1.30 | 1.05 | Rect   |

| #  | Pos X | Pos Y | Aperture | Size X | Size Y | Type |
|----|-------|-------|----------|--------|--------|------|
| 6  | 22.54 | 6.89  | D10      | 1.93   | 0.53   | Rect |
| 7  | 22.54 | 8.16  | D10      | 1.93   | 0.53   | Rect |
| 8  | 22.54 | 9.43  | D10      | 1.93   | 0.53   | Rect |
| 9  | 22.54 | 10.70 | D10      | 1.93   | 0.53   | Rect |
| 10 | 22.54 | 11.97 | D10      | 1.93   | 0.53   | Rect |
| 11 | 22.54 | 13.24 | D10      | 1.93   | 0.53   | Rect |
| 12 | 22.54 | 14.51 | D10      | 1.93   | 0.53   | Rect |
| 13 | 32.20 | 0.54  | D10      | 1.93   | 0.53   | Rect |
| 14 | 32.20 | 1.81  | D10      | 1.93   | 0.53   | Rect |
| 15 | 32.20 | 3.08  | D10      | 1.93   | 0.53   | Rect |
| 16 | 32.20 | 4.35  | D10      | 1.93   | 0.53   | Rect |
| 17 | 32.20 | 5.62  | D10      | 1.93   | 0.53   | Rect |
| 18 | 32.20 | 6.89  | D10      | 1.93   | 0.53   | Rect |
| 19 | 32.20 | 8.16  | D10      | 1.93   | 0.53   | Rect |
| 20 | 32.20 | 9.43  | D10      | 1.93   | 0.53   | Rect |

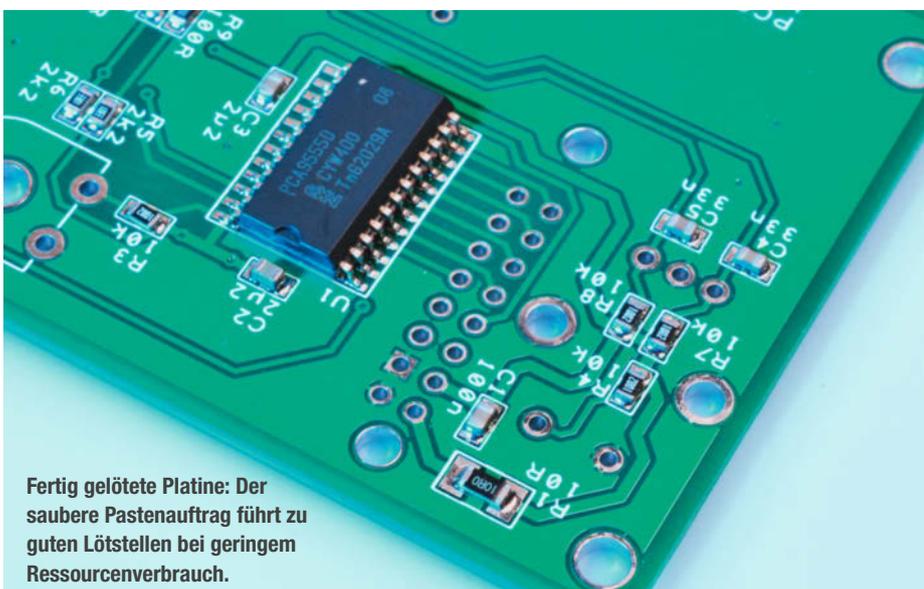
Nach dem Import einer Gerber-Schablonendatei wählt man in der Liste rechts ein kleines Pad als Nullpunkt aus, falls das links unten zu groß ausfällt.



Nach dem Import der Gerber-Daten für die Pastenschablone wählt man in der Pad-Liste ein kleines Pad aus und führt die Dosierspitze mit den Jog-Buttons an dieses Pad heran.



Ein solches Dosiernadel-Sortiment gibt es für wenig Geld bei Amazon oder eBay. Brauchbar sind hier allerdings nur die mittleren Durchmesser von 0,7 bis 1mm.



Fertig gelötete Platine: Der saubere Pastenauftrag führt zu guten Lötstellen bei geringem Ressourcenverbrauch.

Aufgaben (etwa Nachbestückung) ist es möglich, Blenden oder Pads zu deaktivieren oder neu zuzuweisen (siehe Einträge der Pop-up-Menüs). Es ist sogar möglich, nur ein einzelnes Pad neu zu „dispensieren“ (Pop-up-Menü in der Tabelle).

Unsere Vorgaben für Dosierzeit (*Dwell ms*) und Padgrößen-Multiplikator (*Pad Mult*), mit dem man die Dosierzeit je nach Pad-Abmessungen verlängert, sind empirisch ermittelt. Gegebenenfalls müssen Sie diese Werte nach Bedarf und Dosiernadel-Durchmesser anpassen. Das Programm merkt sich seine Einstellungen in einer INI-Datei im gleichen Verzeichnis, das gilt auch für die Schnittstellen-Parameter (über den Button *Setup*).

### Tipps zur Anwendung

Den *Vacuum*-Regler am AD-982 benötigen wir nicht, er wird ganz nach rechts gedreht; die eingebaute Vakuumdüse verbraucht sonst ständig Druckluft. Bei dünnflüssigen Medien sorgt er ansonsten dafür, dass ein Unterdruck die Flüssigkeit wieder etwas in die Dosierspritze zurücksaugt. Gute Ergebnisse erzielen wir mit einem eingestellten Druck von 3 bar, auf dem auch unsere Dosierzeit-Vorgaben basieren. Je nach Umgebungstemperatur (die beeinflusst die Pastenviskosität) und Dosiernadel-Durchmesser sind die Werte von Fall zu Fall anzupassen.

Damit der vom Programm errechnete Dosierimpuls ausgegeben wird, muss der grüne Schalter am AD-982 auf *MAN* stehen. Als Dosiernadel verwenden wir extrakurze, stumpfe Kanülen mit 0,8mm Außendurchmesser oder für sehr feine BGA-Pads solche mit 0,7mm. Die bei Lötpasten-Spritzen oft mitgelieferten 1,25-mm-Kanülen sind schon für IC-Pads zu grob. Die Farbcodes der Dosiernadeln sind leider bei chinesischen Fabrikaten abweichend von üblichen Konventionen – am besten mit der Schiebellehre ausmessen.

Bedenken Sie, dass die Lötpaste nach einigen Tagen eintrocknet und dann die Kanüle verstopft; diese ist dann unbrauchbar. Bei längeren Arbeitspausen verschließt man die Düsenspitze mit einem Klümpchen Knetmasse oder dergleichen.

Für den Fall, das eine neue oder längere Zeit unbenutzte Kanüle verwendet wird, gibt es in unserer Software einen *Feed-Air*-Button. Man drückt ihn so lange, bis (bei hochgefahrterer Z-Achse) ein feines Würstchen an der Kanüle erscheint, das man vor dem Dosiervorgang natürlich abwischen muss. Beim Wechsel der Dosiernadel wird oft eine kleine Luftblase eingeschlossen, auch diese muss vor dem Dispensieren mit dem *Feed-Air*-Button herausgedrückt werden.

Die Materialliste, ein ausführbares Programm und die Delphi-Sourcen finden Sie in unserem Github-Repository zum Projekt (siehe Link im Info-Kasten). —cm



» Continuous Lifecycle »

[Container Conf]

Die Konferenzen für Continuous Delivery, DevOps, Containerisierung und Developer Experience

15. – 17. November 2022  
in Mannheim

## Endlich wieder vor Ort

**Continuous Delivery, DevOps** und **Containertechniken** sind fest etabliert in der zeitgemäßen professionellen Softwareentwicklung – unterliegen aber ständigem Wandel. **Developer Experience** und eine optimal integrierte Developer-Plattform werden zu entscheidenden Faktoren für die erfolgreiche Umsetzung in der Praxis. Continuous Lifecycle und ContainerConf halten Sie mit regelmäßigen Updates erfahrener Expert:innen auf dem Laufenden.

### Das Programm bietet Vorträge zu folgenden Themen:

- ✓ Continuous Integration & Continuous Delivery
- ✓ DevOps, Dev(Sec)Ops, GitOps, AIOps, FinOps
- ✓ Containerisierung, Kubernetes
- ✓ Infrastructure as Code
- ✓ Softwarearchitektur
- ✓ Teamwork & Weiterentwicklung



Jetzt anmelden: [www.continuouslifecycle.de](http://www.continuouslifecycle.de)

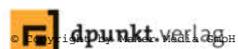
#### Gold-Sponsoren



#### Silber-Sponsoren



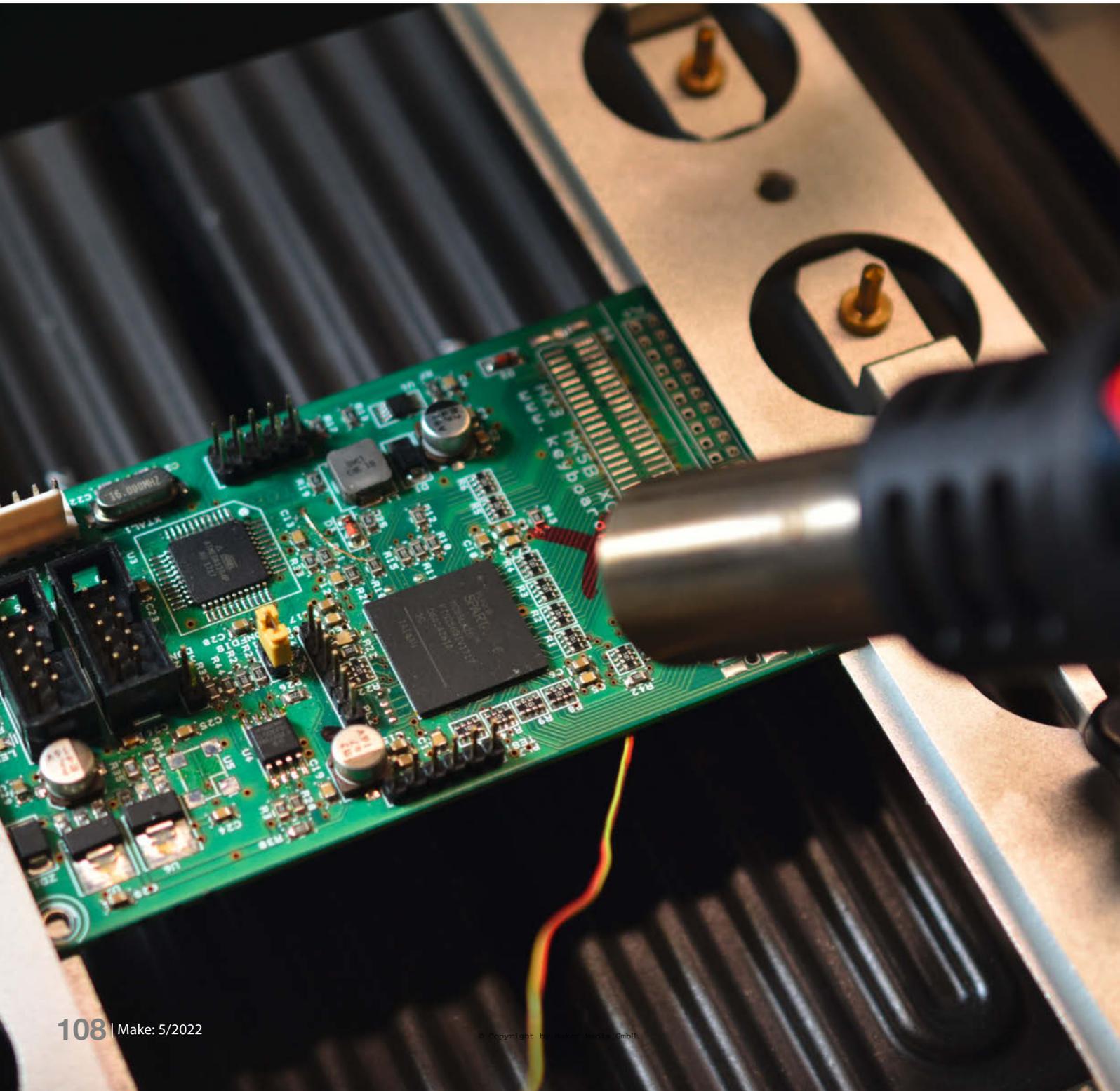
#### Veranstalter



# Reballing und Löten von BGA-Chips

Ein Angstgegner vieler Maker sind ICs in BGA-Bauform – und erst recht, wenn sie ausgelötet und wiederverwendet werden sollen: Die Ball Grid Arrays sind zwar modern und platzsparend, aber halt nicht von Hand mit dem Kolben zu löten. Wir zeigen praktikable Lösungen.

von Carsten Meyer



**A**ngesichts der Chip-Knappheit muss man sich als Maker an gewisse Prozeduren gewöhnen, ob man will oder nicht: Das Wiederverwenden ausgelöteter, aber ansonsten funktionsfähiger Bausteine ist inzwischen auch in der Industrie üblich, während es früher billiger war, Überbestände an Baugruppen einfach wegzuwerfen und die Chips neu zu beschaffen.

Dank SMD-Technik ist das Bauteile-Recycling heute in den allermeisten Fällen zerstörungsfrei und ohne Qualitätseinbußen möglich – wer vor 25 oder 30 Jahren noch Prozessoren im DIL-Gehäuse auslöten musste, weiß von ganz anderen Schwierigkeiten zu berichten.

## SMD-Recycling

Für das Recycling von SMD-Teilen benötigt man nicht mehr als eine Vorwärmplatte und eine Heißluftpistole oder besser eine Heißluft-Lötstation. Mit der Vorwärmplatte (dafür eignet sich zum Beispiel auch unsere Bügeleisen-Lötstation) erhitzt man die zu „entstückende“ Platine auf rund 150°C Oberflächentemperatur. Dann das auszulötende Bauteil kurz und gezielt mit der Heißluftpistole auf die Schmelztemperatur des verwendeten Lötzinns bringen (bei Bleifrei-Löten etwa 215°C) und das Bauteil mit einer Pinzette abheben (siehe Aufmacher-Bild).

Ohne Vorwärmplatte erwärmt man die Platine zunächst großflächig mit der Heißluftpistole, möglichst von beiden Seiten. Das Auslöten ohne Vorwärmen führt wegen des hohen Schmelzpunktes bleifreier Lote zu großem thermischen Stress für das Bauteil und ist zu vermeiden.

Wer mit den ausgelöteten Bauteilen nicht gerade kritische Baugruppen zu bestücken hat, muss ansonsten nicht übertrieben zimperlich sein: Die thermische Leidfähigkeit von Halbleitern ist bei nicht allzu abrupten Temperatursprüngen überraschend groß.

Mit diesem Verfahren lassen sich auch vielpolige Bauteile sicher auslöten. Vor der Wiederverwendung entfernt man Lötzinnreste am Bauteil mit reichlich Flussmittelpaste und einer sauberen Lötspitze; die LötKolbentemperatur sollte dabei nicht über 330°C liegen.

## BGAs recyclen

Das Schreckgespenst vieler Maker sind Chips im Ball Grid Array-Gehäuse: Hier sind die Anschlüsse nicht seitlich, sondern unter dem Chip angebracht; ein Verlöten ist deshalb nur im Reflow-Ofen oder wiederum mit Vorwärmplatte und Heißluft möglich. Wir empfehlen den Ofen, der mit seinem konsistenten Temperaturprofil eine zuverlässige Lötung ermöglicht.

## Kurzinfo

- » SMD-Chips auslöten und recyceln
- » BGA-Reballing mit Schablone und Lotkugeln
- » Lötpasten-Dispenser statt Lotkugeln

### Checkliste

-  **Zeitaufwand:**  
0,5 bis 1 Stunde pro Chip
-  **Kosten:**  
15 Euro für Lötpaste, Flussmittel und Lotkugeln
-  **Feinwerkzeug:**  
Feine Pinzette, weicher Aquarellpinsel
-  **Maschinen:**  
Vorwärmplatte und Heißluft-Lötstation oder Rework-Station, ggf. Reflow-Löten
-  **Löten:**  
Fortgeschrittenes Können vorteilhaft, ruhige Hand essentiell

### Material

- » BGA-Lotkugeln Durchmesser nach IC-Datenblatt
- » Rework-Flussmittelpaste klebrig, „tacky“
- » Lötpaste in Spritze für Dispenser

### Werkzeug

- » BGA-Reballing-Schablonen Durchmesser und Pitch nach IC-Datenblatt
- » Reballing-Schablonenhalter
- » USB-Mikroskop empfehlenswert

### Mehr zum Thema

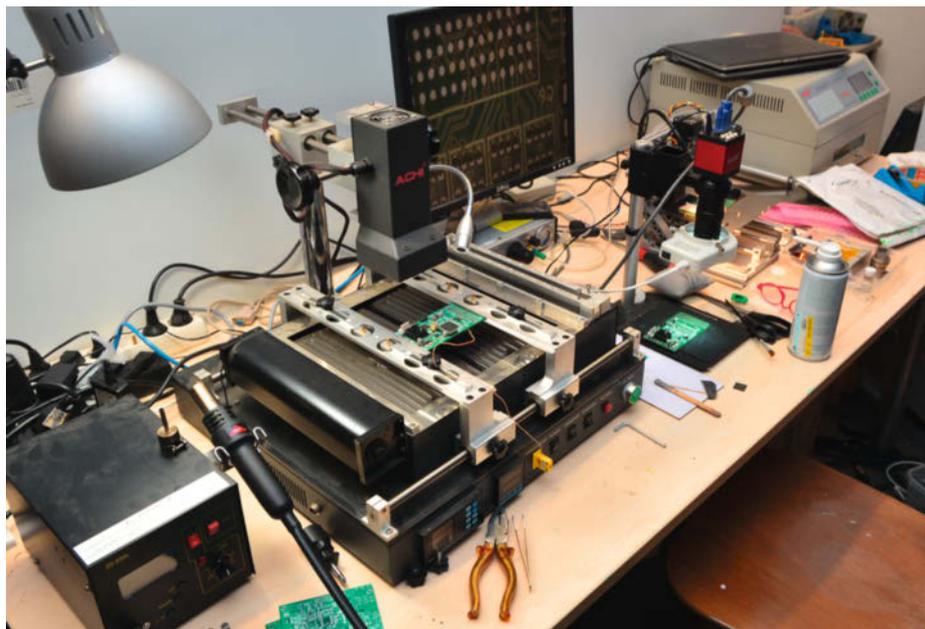
- » Moritz König, Reflow-Löten mit dem Bügeleisen, Make 5/20, S. 116

Alles zum Artikel  
im Web unter  
[make-magazin.de/x3mn](https://make-magazin.de/x3mn)

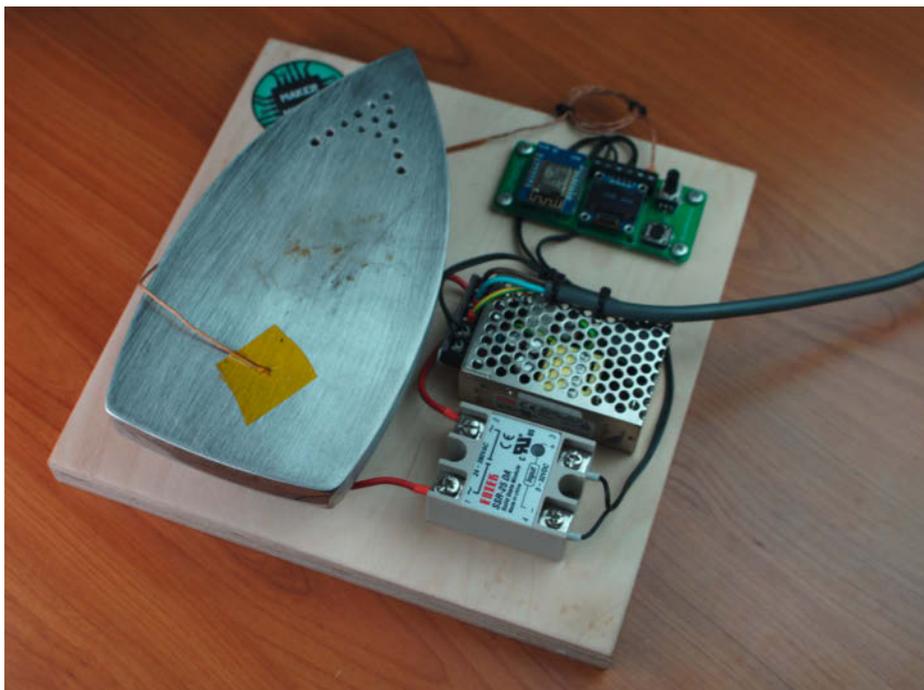


Mit dem oben beschriebenen Verfahren lassen sich natürlich auch BGAs entlöten, allerdings steht man dann vor einem Problem: Man hat das Bauteil so seiner Anschlüsse be-

raubt, die im Lieferzustand aus lauter kleinen Lötzinnperlen bestanden. Die sind aber beim Lötvorgang aufgeschmolzen und hängen jetzt deformiert teils an der Platine, teils am Bauteil.



Idealerweise ist der Rework-Arbeitsplatz mit Heißluft-Lötstation, Vorwärmplatte, USB-Mikroskop und Reflow-Ofen ausgestattet.



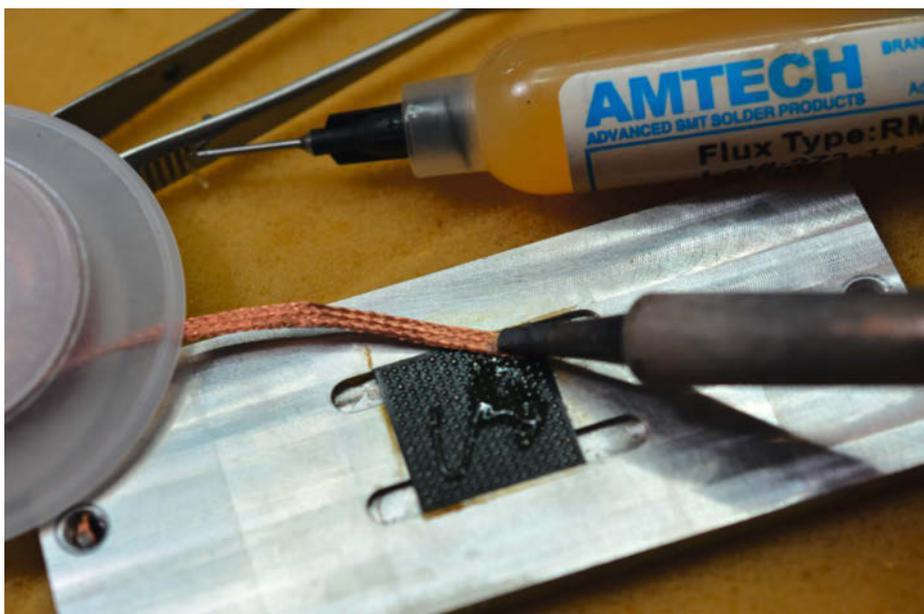
Unsere Bügeleisen-Lötstation aus Make 5/20 eignet sich auch als Vorwärmplatte.

## Sauber!

Zuerst ist also das restliche Lötzinn am Bauteil zu entfernen: Dazu das Bauteil zunächst mit den Lötperlen nach oben fixieren (etwa mit ringsherum angeordneten Platinenresten). Entlötlitze mit SMD-Flussmittelpaste tränken (aushärtender Lötack oder Opas Löthönig aus der Dose eignen sich nicht!) und mit einer breiten LötKolbenspitze die Litze über die Lötzinnreste wischen.

Auch hier sollte die Spitzentemperatur nicht sonderlich über 330°C liegen, ansonsten verbrutzelt das Flussmittel und hinterlässt stumpfe, verzünderte Pads auf dem Bauteil. Die werden an der Oberfläche dunkel und nehmen später die Lötperlen nicht mehr an. Nicht ruckartig an der Entlötlitze ziehen, die Pads können sonst abreißen.

Damit ein BGA-Bauteil weiterverwendet werden kann, müssen neue Lötperlen aufgeschmolzen werden – dazu gibt es verschie-



Säubern des BGA-Chips von Lötzinnresten: Entlötlitze nimmt das anhaftende Lötzinn auf. Hierfür reichlich Flussmittelpaste und eine eher breite Lötspitze verwenden.

dene Verfahren. Eine besonders schonende Technik wird in der Industrie angewandt: Die im Raster aufgebrachtene Löt-Kügelchen werden dabei einzeln von einem Laser angeschmolzen. Das minimiert den Stress für das Bauteil. Solche kostspieligen Maschinen kann sich der Maker natürlich nicht erlauben. Für den gelegentlichen Bedarf haben sich folgende Verfahren etabliert, die wir der Reihe nach beschreiben.

## Reballing mit Lotkugeln

Benötigt wird eine Reballing-Schablone (*Stencil*, Rastermaß und Pad-Durchmesser beachten), ein Vorrat an Löt-Kügelchen (*BGA Balls*), pastöses SMD-Reballing-Flussmittel (in Spritze) und eine Fixiervorrichtung mit Schablonenhalter. Sehr vorteilhaft bei allen Arbeiten ist ein USB-Mikroskop. Den nötigen Kugel-Durchmesser erfahren Sie aus dem Datenblatt des Bauteils; Abweichungen von 0,05mm sind durchaus zulässig (wir haben hier 0,55-mm-Kugeln verwendet, obwohl im Datenblatt 0,6mm angegeben waren).

Der gesäuberte BGA-Chip wird fixiert und mit einer dünnen(!), aber möglichst gleichmäßigen Schicht Flussmittelpaste versehen. Anschließend platziert man eine Edelstahl-Schablone, eine Art Sieb in der Rasterweite der Lötperlen, über dem Bauteil, sodass Löcher und Bauteil-Pads deckungsgleich sind, die Schablone den Chip und das aufgebrachte Flussmittel aber gerade noch nicht berührt. Verwenden Sie hier auf jeden Fall ein klebriges Flussmittel (*tacky flux*) wie das *Ersa NC5070*.

Die aus dem Vorratsfläschchen aufgestreuten Kügelchen können nun durch die Schablone auf den Chip fallen und bleiben am Flussmittel kleben. Restliche Kügelchen schüttelt man durch Schräghalten der Vorrichtung und leichtes Klopfen ab. Anfangs versuchten wir, überschüssige Kügelchen mit einem feinen Pinsel herauszuwischen; das führte aber dazu, dass sich die korrekt platzierten mitamsam des anhaftenden Flussmittels drehten, wodurch sie dann aber an der Schablone und nicht am Chip kleben blieben.

## Hoffen und Bangen

Dann lässt man das Flussmittel einige Minuten antrocknen und hebt die Schablone vorsichtig ab, hoffend, dass keine Lotperlen an der Schablone haften geblieben sind; die kann man notfalls mit einer Pinzette unter dem Mikroskop nachbestücken – wer das ein paar Mal gemacht hat, kann danach auch seinen Namen auf ein Reiskorn schreiben. Der Chip wird schließlich entnommen, auf einen Platinenrest gelegt und wie eine normale SMD-Platine im Reflow-Infrarotofen gebacken. Heißluft eignet sich nicht, man würde die sorgsam platzierten Kügelchen wieder wegblasen.

Keine Sorge, wenn das Kugel-Raster vor dem Löteten etwas unruhig erscheint: Die Löt-kugeln flutschen beim Löteten „automagisch“ in die richtige Position. Wir verwenden für unsere (privaten) Rework-Arbeiten Kugeln aus einer Zinn-Blei-Legierung, die dank einer Schmelztemperatur von rund 190°C unkritischer zu verarbeiten sind als die erst bei 215°C aufschmelzenden Kugeln aus bleifreiem Lötzin.

Beim Reballing mit Löt-kugeln gibt es noch ein weiteres *direct-reballing*-Verfahren, bei dem man die Schablone auf dem Chip belässt und kein definierter Abstand zwischen Stencil und Chip nötig ist. Die nötige Vorrichtung ist deutlich einfacher aufgebaut: Der Chip wird hierbei mit einer kleinen Feder gegen die Schablone gepresst, das Verlöten erfolgt dann direkt in der Vorrichtung, notfalls sogar mit Heißluft.

Nachteil: Da das aufgebrauchte Flussmittel dann an der Schablone klebt, lässt sich der Chip möglicherweise nur mühsam entfernen (ggf. wieder etwas anwärmen), und die Schablone muss vor der nächsten Verwendung sorgfältig gereinigt werden. Bei unseren Versuchen blieb manchmal eine Lotperle in der Schablone hängen; wenn dann das Pad mit abgerissen ist, hat man Pech gehabt.

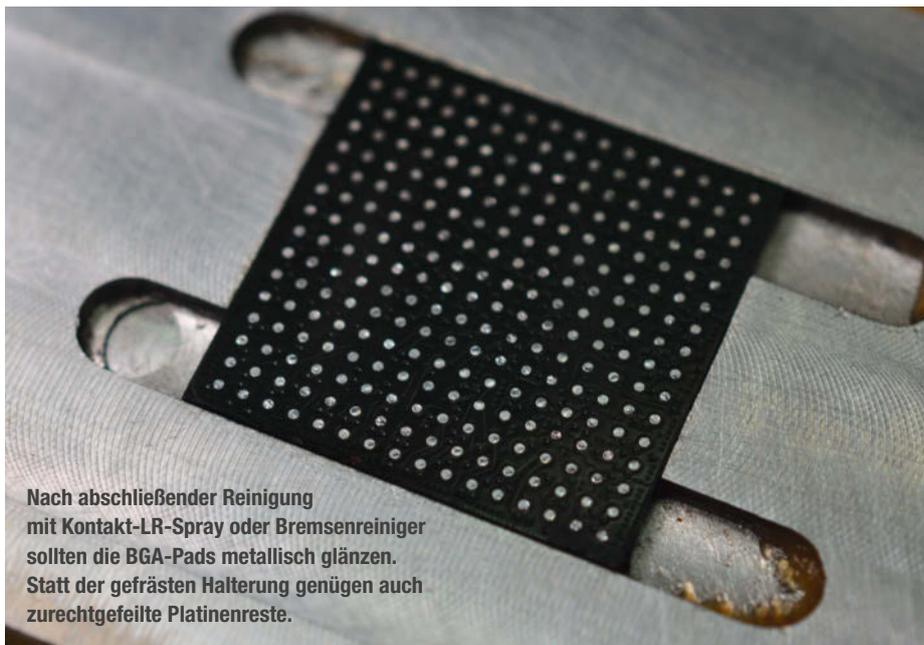
## Reballing mit Löt-paste

Etwas unorthodox, aber trotzdem praktikabel ist das Reballing mit Löt-paste statt mit Löt-kügelchen. Es funktioniert analog zum Pastenauftrag mit Schablone oder Dispenser auf einer Platine.

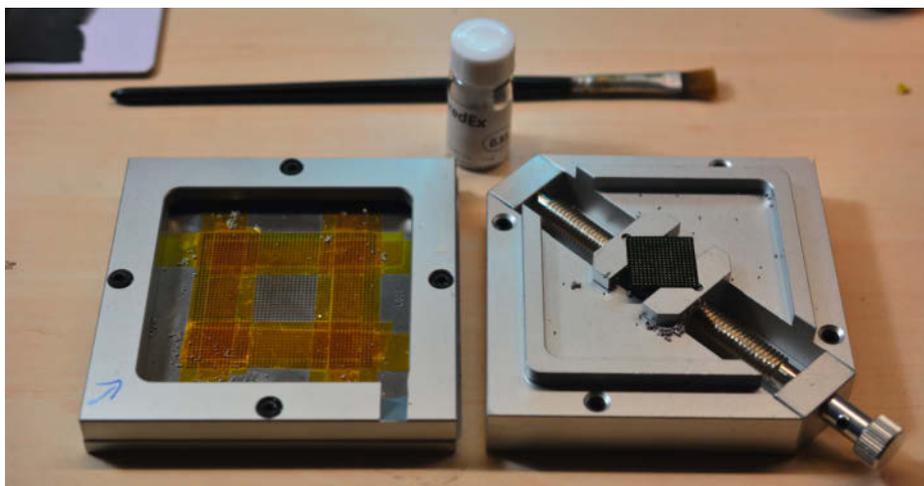
Man braucht zunächst eine Fixierung für den Chip, etwa aus Platinenresten (oft leider zu dick) oder einem exakt ausgefrästen Alu-Blechstreifen (wie in den Bildern oben, so eine Fräsmaschine vorhanden ist). Zum Auftragen der Löt-paste ist wiederum eine Schablone nötig, wie man sie auch zum Pastenauftrag auf der Platine selbst verwenden würde. Ein zusätzliches Flussmittel wird nicht gebraucht.

Das BGA-Stencil haben wir einfach mit einer kleinen Blechschere aus einer Pastenschablone für eine alte Platine ausgeschnitten, überflüssige Löcher klebt man mit Kapton-Kleband ab. Die Schablone wird exakt über den Pads des Chips ausgerichtet und fixiert, ganz so, wie Sie es vom Beschichten einer Platine kennen. Dann Löt-paste aus der Dose (die ist etwas fester als die eher dünnflüssige aus der Spritze) aufbringen und mit einem Mini-Spachtel in das Lochraster drücken. Damit der Chip beim Abheben der Schablone nicht daran kleben bleibt, kann man den Chip vorher mit etwas Flussmittel-Paste auf dem Rücken „anheften“.

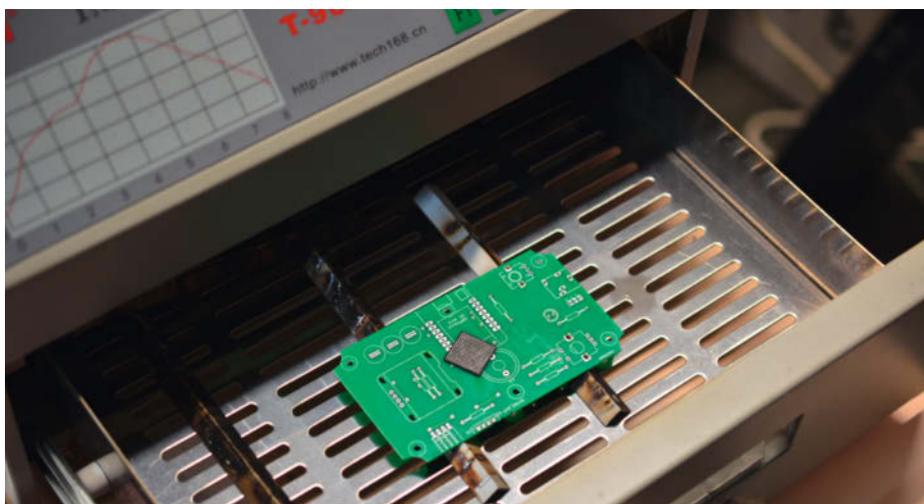
Hier sind oft mehrere Versuche nötig, da die Löt-paste manchmal nicht am Pad haftet, sondern im Loch der Schablone stecken-



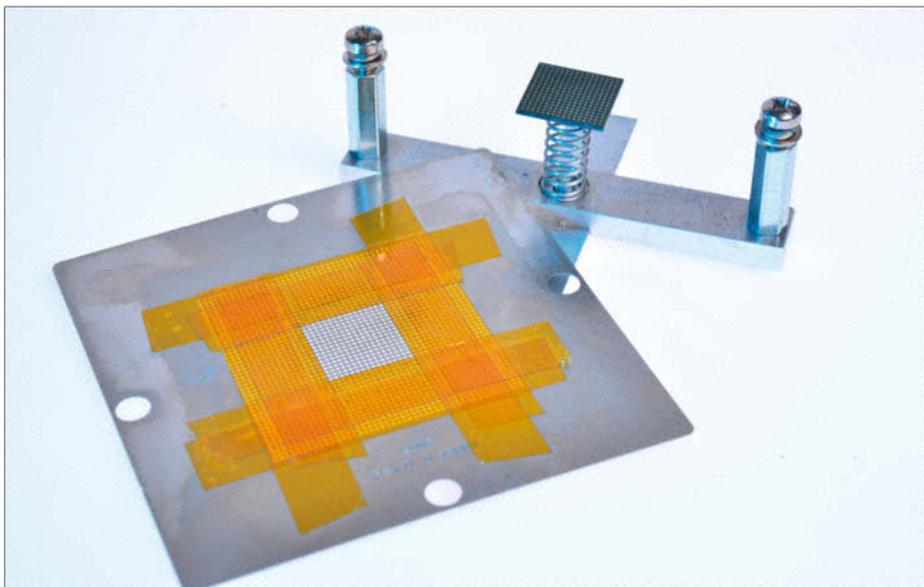
Nach abschließender Reinigung mit Kontakt-LR-Spray oder Bremsenreiniger sollten die BGA-Pads metallisch glänzen. Statt der gefrästen Halterung genügen auch zurechtgefeilte Platinenreste.



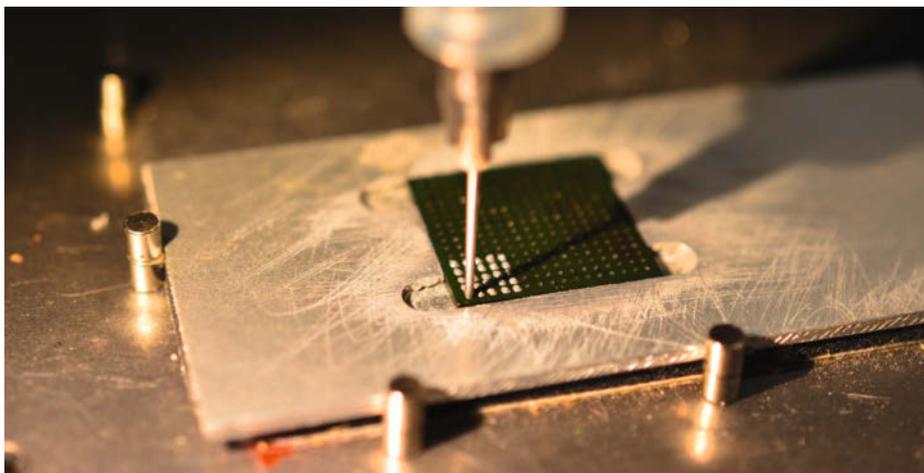
Nicht das Gelbe vom Ei: Billige Reballing-Vorrichtungen wie diese weisen ein großes Spiel auf und eignen sich nur für gelegentliche Arbeiten. Wir würden nach diesen Erfahrungen lieber zum Schablonenhalter aus dem Ersa-Shop greifen.



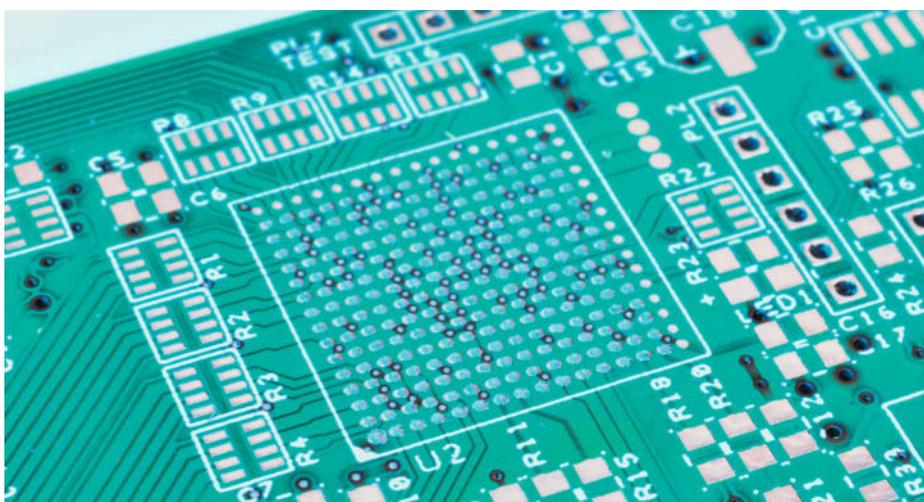
Der BGA-Chip wird mit den Anschlüssen nach oben auf einen Platinenrest gelegt und die Kugeln im Reflow-Verfahren aufgeschmolzen. Dabei nicht ruckeln!



Eine Direct-Reballing-Vorrichtung ist sehr einfach aufgebaut und kann aus Alu-Resten auch leicht selbst hergestellt werden.



Auftragen von Lötpaste auf die gereinigten BGA-Pads mit unserem Lötpasten-Dispenser. Achten Sie auf eine gleichmäßige Häufchenbildung. Sollten Pads verschmieren: Einfach abwischen und nochmal von vorn.



Vorteil unseres Dispensers: Selbst wenn die Platine bereits andere Bauteile enthält oder der Rest von Hand bestückt werden soll, kann man damit die Lötpaste für das Reflow-Löten auftragen.

bleibt. Die Lötpaste wird, wenn der Auftrag sauber erfolgte, nun im Ofen (so Sie ihrer Vorrichtung das zutrauen) oder mit Heißluft aufgeschmolzen.

### Reballing mit Dispenser

Auch damit gelingt der Pastenauftrag. Die Wahl des optimalen Dosiernadeldurchmessers nebst Druck und Impulsdauer ist allerdings recht heikel; schon eine kleine Änderung der Pasten- und Umgebungstemperatur macht leichte Anpassungen der Parameter nötig.

Vorteil des Dispensers ist, dass man den Chip direkt zum Aufschmelzen der Lötpaste entnehmen kann. Das geschieht auch hier besser im Reflow-Ofen, der durch Abfahren seines Temperaturprofils sicherstellt, dass das in der Paste enthaltene Lösungsmittel gut verdunstet, der Flussmittelanteil aktiviert wird und das Zinn gut aufschmilzt.

Wichtig: Egal, ob sie per Schablone oder Dispenser aufgetragen wurden, sind die entstandenen Lötzinn-Kügelchen eigentlich nur Kalotten mit leicht differierender Höhe. Um eine über den gesamten Chip gleichbleibende Höhe zu gewährleisten, schubbert man den Chip bei leichtem Druck über ein Blatt Papier mit glatter, harter Unterlage. Das nimmt etwas Material von den zu hohen Halbkugeln ab (erkennbar am Abrieb auf dem Papier) und sorgt zudem noch für leichte Abplattungen, was für den späteren Lötvorgang durchaus vorteilhaft ist.

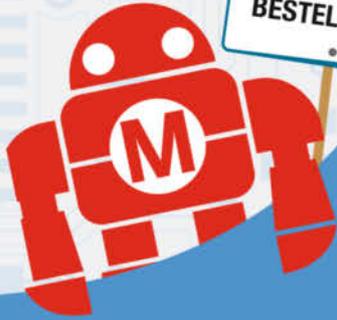
Mit den härteren bleifreien Kügelchen kann man auch 1000er Schleifpapier zum Egalisieren versuchen. Ist bis hierhin alles gutgegangen, entfernt man noch die anhaftenden Flussmittelreste, und schon haben Sie einen (heutzutage möglicherweise unbezahlbaren) Chip vor dem Mülleimer gerettet.

### BGAs bestücken

Übrigens lassen sich BGA-Chips mit 1mm oder 0,8mm *ball pitch* durchaus von Hand bestücken, wie unsere vielen Musterplatten beweisen – Voraussetzung ist eingeschränkter Kaffeegenuss, ein exakter Positionsdruck und ein sauberer Pastenauftrag auf der Platine. Das Bauteil muss möglichst parallel aufgesetzt werden (dazu eignet sich am besten eine Saugpipette oder eine abgewinkelte Pinzette) und wird dann leicht angedrückt.

Wenn das Bauteil ein, zwei Zehntelmillimeter fehlplatziert wurde, schwimmt es beim Löten in die richtige Position. 0,5mm-BGAs trauen wir uns dann aber doch nicht, die sind auch schon beim Layout einer Multilayer-Platine mit den notwendigen 0,2-mm-Vias eine heftige Herausforderung. —cm

# Noch mehr Stoff für Maker



**PORTOFREI**  
AB 20 €  
BESTELLWERT

2012 – 2014

2015 Heft 1-6



2016 Heft 1-7

2017 Heft 1-7



2018 Heft 1-7

2019 Heft 1-7



2020 Heft 1-7

2021 Heft 1-7



Lies sie alle!



[shop.heise.de/make-magazin](https://shop.heise.de/make-magazin)

# Fräswerkzeuge selbst schärfen und bauen

Fräser sind teures Verbrauchsmaterial: Einmal stumpf können die gängigen mehrschneidigen Typen nur mit Fachwissen und Übung wieder geschärft werden, was sich nur bei hochwertigen Werkzeugen lohnt. Aber es existieren auch einfachere Varianten, die der Maker selbst schärfen und sogar herstellen kann.

von Achim Bertram

Jeder Maker mit einer Fräsmaschine oder gar CNC-Fräse kennt das: Um gute Ergebnisse beim Fräsen zu erzielen, muss man mit möglichst scharfen Werkzeugen arbeiten.

Das Nachschleifen eines mehrschneidigen Fräasers ist aber, anders als bei einem Bohrer, nur mit einer speziellen Schleifeinrichtung oder über einen Werkzeugservice machbar. Beides ist mit hohen Kosten verbunden, sodass oft nur eine Neuanschaffung übrig bleibt.

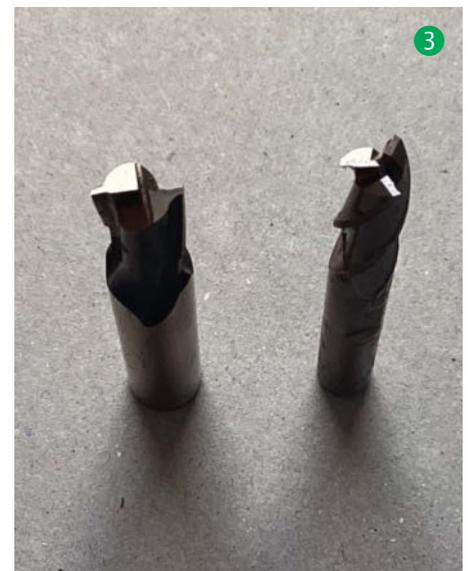
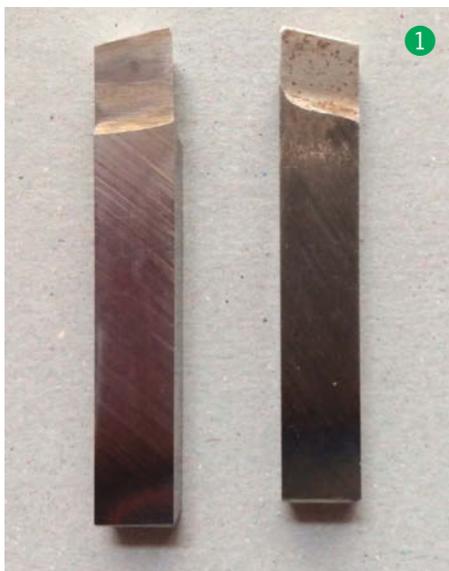
Um Werkzeugkosten zu sparen, kann man einfache Fräswerkzeuge aber auch selbst anfertigen oder andere Werkzeuge zu solchen umgestalten. Das wohl einfachste Fräswerkzeug ist der *Schlagzahn*, oder auch *Einzahn* genannt. Je nach Ausführung können mit ihm sowohl ebene Oberflächen als auch nutförmige Vertiefungen gefertigt werden. Sonderfälle für den Einsatz von Einzahnfräsern sind Gravuren und passgenaue Ausdrehungen.

Solche *Einschneider* können einfach von Hand geschliffen, somit immer wieder geschärft und durch geeignete Schliff-Winkel dem zu schneidenden Werkstoff angepasst werden.

### Schlagzahn als Planfräser

Um die Oberflächen von rohen Werkstücken eben und parallel zueinander zu bearbeiten und dabei eine hohe Abtragsleistung zu erzielen, werden in Industrie und Handwerk Walzenstirnfräser oder Messerköpfe verwendet. Der Hobbyzerspaner kann auf diese teuren Werkzeuge verzichten und einen *Schlagzahn* benutzen.

Im einfachsten Fall ist das ein HSS-Drehling, der in eine spezielle Aufnahme geklemmt wird. Ein Drehling ist einfach ein Rohling für Drehmeißel, normalerweise zum Gebrauch an der Drehmaschine. HSS steht hier für Hochleistungs-Schnellarbeitsstähle. Mehr zu den verschiedenen Schneidstoffen finden Sie



## Kurzinfo

- » Einzahnfräser als einfach zu fertigende Alternative
- » Fräserarten unter der Lupe
- » Tipps fürs Schleifen und Abziehen

### Checkliste



**Zeitaufwand:**  
1 Stunde



**Kosten:**  
ab 0 Euro



**Maschinen:**  
(CNC)-Fräsmaschine,  
ggfs. Drehmaschine

### Material

- » Stumpfe Fräser
- » Drehlinge
- » abgebrochene Gewindebohrer

### Werkzeug

- » Schleifscheibe mit feiner Körnung  
Edelkorund für HSS, Siliciumkarbid für HM
- » Handläpper 200–600er Körnung

### Mehr zum Thema

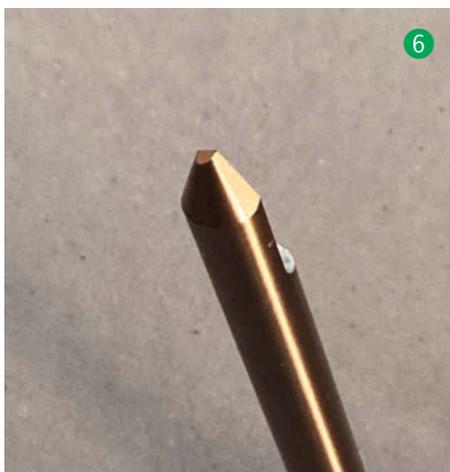
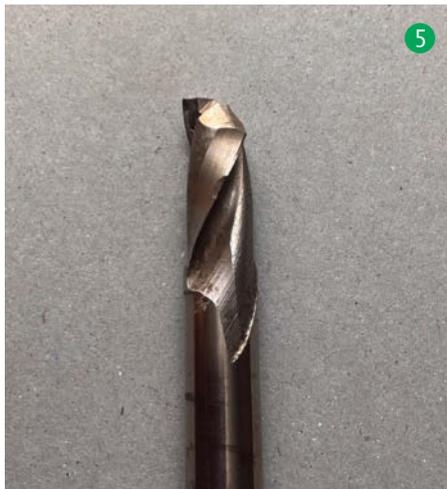
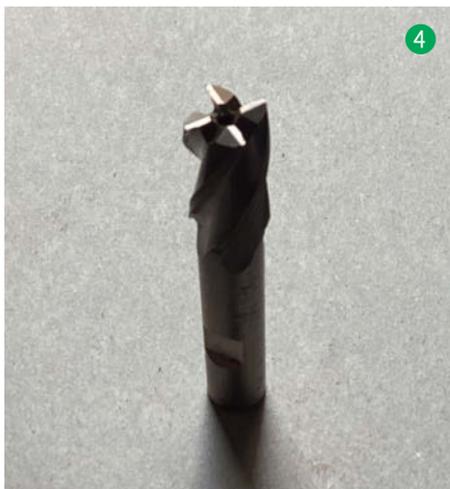
- » Hermann Dengler, Metalle fräsen, Make Sonderheft 2021 „Loslegen mit Metall“, S. 148
- » Birgit Hellendahl, Einstieg ins CNC-Fräsen, Make Sonderheft 2020 „Loslegen mit Holz“, S. 102
- » Carsten Meyer, Stepcraft M.700, Make Sonderheft 2020 „Loslegen mit Holz“, S. 114

Alles zum Artikel  
im Web unter  
[make-magazin.de/x9de](http://make-magazin.de/x9de)



in den Links zum Artikel. Im Bild 1 sind ein linker (links in Bild) und ein rechter Seitendrehmeißel abgebildet. Der Name kennzeichnet die Hauptbearbeitungsrichtung. Ein rechter Meißel arbeitet von rechts nach links. Der linke Seitendrehmeißel ist geschärft und mit 0,5 mm Eckenradius versehen, der rechte ist bereits deutlich abgenutzt.

Als Aufnahme dient bei mir ein selbstgefertigter Halter 2 für Drehlinge mit quadratischem (10mm x 10mm) Querschnitt. Ähnliche Konstruktionen gibt es auch fertig zu kaufen, suchen Sie nach Schlagzahn oder *Flycutter*. Der Körper besteht aus zwei Hälften. Die Aufnahmenuten sind spiegelbildlich unter 45° bzw. 90° zur Achse eingefräst. In eine optionale



waagerechte Nut kann alternativ ein Kreisschneider eingesetzt werden. Die Hälften werden verschraubt oder verlötet. Dann wird der Halter rund gedreht und mit einem Spannanzapfen versehen.

Bei rechtsdrehender Spindel ist der Anschliff analog zu dem eines linken Seitendrehmeißels. Die Winkel an der Schneide (siehe auch im Online-Teil, Link oder QR-Code in Kurzinfo) Freiwinkel, Keilwinkel und Spanwinkel



## Drehzahlen und Sicherheit

Werkzeuge wie der Halter aus Bild 2 dürfen nur mit niedriger Drehzahl gefahren werden, die sich aus der Schnittgeschwindigkeit und dem Durchmesser des Schnittes errechnet. Durch die extreme Unwucht kommt es sonst zu Schwingungen, die im Resonanzfall zu schweren Schäden an Werkstück und Spindellagern führen können. Durch die große Ausladung der Schneide wird aber dennoch eine ausreichende Schnittgeschwindigkeit erreicht.

### Beispielrechnung:

Ausladung der Schneidkante aus der Mitte: 50mm (entspricht  $d = 100\text{mm}$ ),

Schnittgeschwindigkeit Aluminiumlegierung: 40m/min (lt. *Garant Zerspanungshandbuch*)

$$\text{Drehzahl: } n = (v \cdot 1000) / (d \cdot \pi)$$

$$n = (40\text{m/min} \cdot 1000\text{mm/m}) / (100\text{mm} \cdot \pi)$$

$$n = 127 \text{ 1/min}$$

Es empfiehlt sich der Einsatz von Schutzeinrichtungen, um die seitlich wegfliegenden Späne aufzuhalten. Das Tragen einer Schutzbrille ist bei allen spanenden Bearbeitungen obligatorisch.

sind werkstoffabhängig. Hier sollte in entsprechenden Tabellen nachgeschlagen werden. Der Einstellwinkel ist durch die 45° Schräglage gegeben und für Schrumparbeiten geeignet.

Um eine wirklich ebene Oberfläche zu erzielen, muss die Spindelachse Z so genau wie möglich rechtwinklig zur XY-Ebene ausgerichtet sein. Anderenfalls ergibt sich je nach Winkelfehler eine mehr oder weniger stark ausgeprägte konkave Frässpur.

## Einzahn als Nut- oder Schaftfräser

Fräswerkzeuge haben in der Regel zwei oder mehr Schneiden, die jeweils aus Hauptschneide an der Stirn und Nebenschneide am Umfang bestehen. In Bild 3 sind Bohrnutenfräser zu sehen: ein Zweischneider und ein Dreischneider. Jeweils eine der Hauptschneiden reicht bis zur Mitte. Bohrnutenfräser können zum Tauchfräsen verwendet werden und haben als Schrufffräser bei kleiner Schneidanzahl relativ große Spanräume.

Schaftfräser 4 mit vier oder mehr Zähnen und entsprechend kleineren Spanräumen werden zum Schlichtfräsen von Aussen- und Innenkonturen verwendet. Die Zentrierung im Zentrum erlaubt keine Zustellung in volles Material.

Verschleiß tritt an allen Schneiden auf: an der Stirn mehr, am Umfang weniger und besonders stark an der Außenkante. Meist genügt es daher, einen stumpfen Fräser an der Stirnseite zu schärfen. Das gelingt aber von Hand nur mit der ersten Schneide. Bereits die Zweite wird unweigerlich länger, kürzer oder schief.

Abhilfe schafft folgender Trick: Bei Nutfräsern ist für das Tauchfräsen immer eine der Schneiden bis zur Mitte ausgebildet. Diese eine Schneide wird geschärft, alle anderen werden so weit wie möglich verschliffen 5.

Auf diese Weise erhält man einen Einzschneiden-Fräser, der wieder schneidet und der an seiner Hauptschneide noch oftmals mit Heimwerkermitteln geschärft werden kann.

## Stichel als Gravier- oder Kopierfräser

Eine Sonderform des Fräasers ist der Gravierstichel. Er wird hauptsächlich benutzt, um Teilstriche, Symbole oder Schriftzeichen und ganze Texte in eine Oberfläche zu fräsen. Der Gravierstichel hat eine Kegelform, die Frästiefe beträgt nur wenige Zehntelmillimeter und die Schneide ist oft sehr schmal. In Bild 6 sieht man einen Gravierstichel für 2mm breite Linien. Gut zu erkennen ist die Schneide am Umfang mit ca. 15° Hinterschliff (Freiwinkel).

Neben den im Handel erhältlichen Rohlingen eignen sich z. B. abgebrochene Gewindebohrer oder ähnliche Werkzeuge aus HSS, um daraus einen Stichel oder Formfräser zu schleifen.

Solche Stichel werden im Werkzeug- und Formenbau aus vorgefertigten (rundgeschliffen, Halbierung vorgeschliffen) Rohlingen aus HSS oder VHM (siehe online, Link in Kurzinfo), meist mit speziellen Maschinen fertig geschliffen. Diese Stichelschleifmaschinen besitzen eine dreh- und schwenkbare Werkzeugaufnahme mit einstellbaren Anschlägen um die Schneide zu formen.

Ein Stichel ist nicht einfach nur spitz, er würde in diesem Fall ja nicht schneiden, sondern nur ritzen. Er muss an Haupt- und Nebenschneide, so klein diese auch sind, ebenso wie ein Bohrwerkzeug mit Keil-, Span- und Freiwinkel versehen sein.

Als Erstes wird der Rohling ein kurzes Stück weit auf  $(D/2)+0,02\text{mm}$  halbiert. Diese Fläche bildet später die Zahnbrust. Das geringe Übermaß ist zwingend notwendig. Ein Untermaß führte bei rotierendem Werkzeug zu einem Hohlraum in dessen Zentrum und hier bliebe ein kleiner Zapfen Material stehen, der dann beim Fräsen nicht geschnitten, sondern gequetscht wird: Der Stichel schneidet nicht.

Als Nächstes wird die Form des Werkzeuges am Umfang geschliffen. Zum Gravieren eignet sich ein Kegel mit  $45^\circ$  bis  $60^\circ$  Spitzenwinkel.

Zum Schluss bekommt der Stichel **7** seine eigentliche Schneide. Mit einem feinen Abziehstein wird die Spitze nach hinten und zur Seite hin etwa  $5^\circ$  bis  $10^\circ$  freigeschliffen. Die neue Spitze rotiert nun außermittig und kann einen Span abheben. Da der kleine Wirkdurchmesser dieses Werkzeugs höchste Drehzahlen erfordert, werden zum Gravieren gern Hochfrequenzspindeln verwendet.

### Formfräser für Radien oder Zahnflanken

Die Schneide eines Einschneide-Fräasers kann natürlich beliebig geformt werden, wie oben beim Gravierstichel zum Beispiel als Kegel. Um Werkstückkanten mit Fasen, Rundungen oder Hohlkehlen zu versehen, benutzt man entsprechende Formfräser, eine Auswahl meiner Werkzeuge für Radien und Fasen aus HSS-E mit 6 mm Schaft ist in Bild **8** zu sehen. Auch hier ist beim Halbieren das Maß  $(D/2)+0,02\text{mm}$  unbedingt einzuhalten. Jeder Zehntel mehr führt zu einem deutlich negativerem Spanwinkel, der aber bei duktilen, also verformbaren Werkstoffen ungeeignet ist. Die Schneidfähigkeit wird vermindert.

Anschließend wird die Schneide geformt. Radien oder sogar Zahnflanken werden dabei mit einer selbsterstellten Lehre geprüft. Das kann zum Beispiel ein Teil des defekten



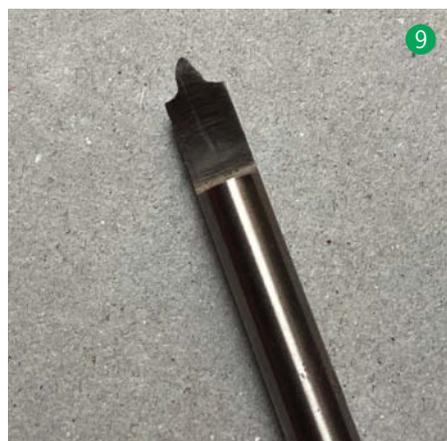
Zahnrad sein, welches mit dem Zahnformfräser **9** neu angefertigt werden soll. Beim Schleifen der Schneide muss diese wieder nach hinten freigeschliffen (Freiwinkel) werden.

### Das Schleifen

Zum Schleifen von Werkzeugen aus Schnellarbeitsstahl verwendet man Schleifscheiben feiner Körnung aus Edelmetall (Al<sub>2</sub>O<sub>3</sub>). Wer auch Hartmetalle schleifen will, benötigt dazu eine Scheibe aus Siliciumkarbid (SiC). Die Feinbearbeitung, das *Läppen* der Zahnbrust, kann mit einem *Handläpper* **10**

(Borcarbid oder Diamant) mit 200er bis 600er Körnung erfolgen.

Diese Schleifarbeiten sind nicht gerade einfach, vor allem wenn sie ohne Vorrichtungen frei Hand ausgeführt werden sollen. Wie immer macht auch hier die Übung den Meister. Facharbeiter wie Zerspanungsmechaniker oder Werkzeugmacher sollten es beherrschen oder die Fähigkeit schnell wieder reaktivieren können. Der Laie kann es wenigstens einmal versuchen. Es lohnt sich auf jeden Fall. Selbstgemachte Werkzeuge senken nicht nur die Kosten für DIY-Projekte, sie geben dem Hobby auch eine besondere Note. —caw



## AT&T Tech Channel

### Videos aus der Blütezeit des Tech-Giganten



Für Fans von Vintage-Technik sind die Videos im *AT&T Archive* aus den *Bell Labs* feinstes Popcorn-Kino. Die Bell Laboratories sind die ehemalige Forschungsabteilung der Telefongesellschaft AT&T, in der im letzten Jahrhundert bahnbrechende technische Erfindungen gemacht wurden. Nicht wenige Nobelpreisträger arbeiteten dort. Der einstige Monopolist AT&T hat nebenbei zahlreiche, eher kurze Dokumentationen produziert, die im *Tech Channel* gesammelt sind. Unter den Videos sind Erklärungen zu digitalen Computern, Glasblasen für Röhren, Einführung in Unix und Interviews etwa mit Arthur C. Clarke zu finden. Aber auch nachrichtentechnische Themen wie die Einführung der Single-Side-Band-Funkübertragung auf Richtfunkstrecken werden in den für die jeweilige Zeit typischen Video- und Audiostilen erklärt.

Legenden wie Dennis Ritchie, Ken Thompson und Brian Kernighan sind in Dokumentationen über Unix zu sehen und zu hören (meine Güte, was für Bärte). Ein aus heutiger Sicht überraschend großer Anteil an Frauen erklärt etwa in dem Video *The UNIX System: Making Computers Easier to Use*, wie das Dateisystem aufgebaut ist, Piping funktioniert und wie man Texte analysiert. Wer Serien wie *Halt and Catch Fire* mag, wird solche Videos lieben. —dab

Anbieter AT&T  
 URL <https://techchannel.att.com/showpage/ATT-Archives>

## Open Circuits

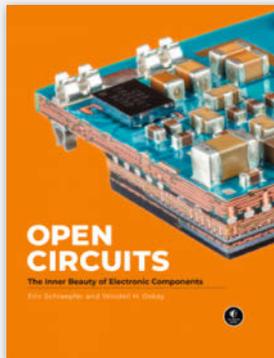
### The inner Beauty of Electronic Components

Manche Erlebnisse lassen sich nur schwer durch etwas anderes ersetzen. So auch die Einblicke, die uns die Autoren Eric Schlaepfer und Windell H. Oksay mit ihrem Buch *Open Circuits* gewähren. Sie haben gesägt, geschliffen, gebohrt und geätzt, um sichtbar zu machen, was wir sonst nur von schematischen Darstellungen kennen: das Innenleben elektronischer Komponenten.

Auf knapp 300 Seiten werden wir auf eine Reise voller funktionaler Design-Ästhetik mitgenommen, die mit beeindruckenden Nahaufnahmen zum Staunen einlädt: Aufgesägte Halbleiter wie aus einer anderen Welt, Schalter und Klinkenstecker, die wie miniatur-

rierte *Rube-Goldberg*-Maschinen wirken, oder allerlei zerschnittene Kabel, die verblüffend anatomisch aussehen. Kurze Begleittexte ergänzen die Fotos und geben Hinweise auf die Herstellung der jeweiligen Komponenten und wo sie anzutreffen sind. Die Stärke des Buches steckt jedoch vorrangig in seinem visuellen Detailreichtum.

*Open Circuits* erscheint voraussichtlich am 1. November in gedruckter Form bei *no starch press* in englischer Sprache, ist aber schon jetzt als Early-Access digital erhältlich. Wer vorab einen Blick hineinwerfen möchte, kann sich auf der Verlagsseite eine kostenfreie Probe herunterladen. —akf



Autoren Eric Schlaepfer und Windell H. Oksay  
 Verlag no starch press  
 Umfang 304 Seiten  
 ISBN 9781718502345  
 Preis E-Book: 31,99 US-\$, gedruckt: 39,99 US-\$ bzw. 23,99 €  
 Website <https://nostarch.com/open-circuits>

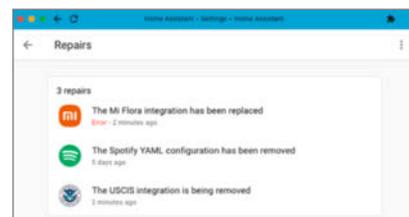
## Home Assistant Release 2022.8

### Smarthome-Server (wieder) mit Bluetooth

Mit dem Update auf die Core-Version 6.7 entfiel die Bluetooth-Unterstützung im Smarthome-Server *Home Assistant*. Das Release 2022.8 bringt sie zurück. Die bisherige, seit 2018 nicht mehr aktualisierte Bluetooth-Unterstützung fiel dem Wechsel auf Python 3.1 zum Opfer. Folge: Bluetooth-gesteuerte Geräte wie Heizkörper-Thermostate von *Equiva* ließen sich nicht mehr steuern.

Das Update-Release 2022.8 mit dem Core 8.7 bringt auch die aktuelle Linux-Bluetooth-Version als eigenständige Integration mit. Infos zur Installation gibt es online (siehe Link in der Tabelle, unter *Integrations/Bluetooth*). Dort findet sich auch ein Button für die Installation direkt auf dem eigenen Home-Assistant-Server.

Jetzt haben die Entwickler von Bluetooth-abhängigen Geräte-Integrationen die Chance, ihre Software an die neuen Bluetooth-Funktionen und -Tools anzupassen. Ein erster Test in der Redaktion zeigte aber, dass auch die bisherigen Integrationen für *eq3*-Heizkörper-



Thermostate und Bluetooth-Thermometer von *Xiaomi* ohne Änderungen funktionieren.

Das Release bringt außerdem eine *Repair*-Funktion mit, die bei fehlerhaft arbeitenden Integrationen darauf aufmerksam macht, wenn es online eine Lösung für das jeweilige Problem gibt. Die Funktion verbirgt sich im Einstellungs-menü, zeigt übersichtlich die vorhandenen Lösungen an (siehe Bild) und kann so manches Problem ohne aufwändige Internet-Recherche lösen. —hgb

URL [home-assistant.io](https://home-assistant.io)  
 Preis kostenlos

# Plate und Studio

## Universelle Schablone und CAD-Webanwendung von Shaper Tools

Für die CNC-Oberfräse Shaper Origin (siehe Make 3/21) hat der Hersteller zwei Neuheiten angekündigt: Bei *Plate* handelt es sich um eine universelle Schablone. Dank derer diversen Anschlagmöglichkeiten sollen sich damit im Handumdrehen präzise Aussparungen für Beschläge aus dem von *Shaper Tools* selbst gepflegten umfangreichen Online-Katalog fräsen lassen – oder auch mal schlicht eine Lochreihe im 32mm-Raster. Wenn nötig, lassen sich aber auch eigene Fräsdaten in Kombination mit der Schablone nutzen.

Unter dem Namen *Studio* soll ab Mitte September eine einfach bedienbare 2D-CAD-Software zur Verfügung stehen, die sich von jedem Gerät aus im Netz nutzen lässt und intern mit SVG als Datenformat arbeitet. Damit sollen sich Vorlagen nicht nur für die *Shaper Origin* zeichnen und anpassen lassen, sondern auch für alle anderen CNC-Maschinen, die SVG als Datenformat akzeptieren,

etwa Schneideplotter und Lasercutter. Das kostenlose *Studio Lite* ist dabei auf einfache Formen, einige Schriften und vorgefertigte Motive sowie den Download fertiger SVG-Dateien beschränkt, die Vollversion kostet Geld, soll dafür aber auch SVG- und DXF-Dateien importieren, viel mehr Schriften und vorgefertigte Formen bieten. Der Bezahlversion ist ebenfalls das *Shape-Shifter*-Werkzeug vorbehalten, mit dem sich die Teile überschneidender Grundformen zu komplexen Formen zusammenfügen lassen. —pek

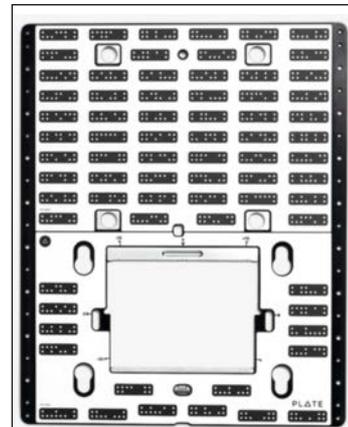


Bild: Shaper Tools

|            |   |
|------------|---|
| Hersteller | Shaper Tools  |
| URL        | shapertools.com/de-de                                   |
| Preis      | Plate: 416 €, Studio: 119 €/Jahr, Studio Lite kostenlos |

# Kiprim DC605S

## Kleines Labornetzteil mit großer Leistung

Der chinesische Hersteller *Kiprim* bietet seine Produkte bei Amazon feil und hat zwei Ausführungen einstellbarer Labornetzteile mit Schaltreglertechnik im Angebot: Das *DC310S* mit 30V und 10A für etwa 130 Euro und das mit rund 170 Euro etwas teurere *DC605S* mit 60V und 5A. Letzteres haben wir uns einmal genauer angesehen.

Zunächst überraschen das für ein 300W-Netzteil geringe Gewicht von gut 1,5kg, die schmale, kompakte Bauform und das TFT-Grafikdisplay anstelle der sonst in dieser Preisklasse noch üblichen Siebensegment-Anzeigen. Der begrenzte Platz auf der Frontplatte bedingt leider eine eher umständliche Bedienung über Tasten und Dreh-Encoder. Ältere Maker werden die winzigen Voreinstellungs-Werte kaum ohne Lesebrille zuverlässig einstellen können. Die zur Änderung ausgewählten Ziffern sind sehr klein und werden farblich nur schwach hervorgehoben. Letzteres gilt auch für die Liste der Einstellungs-Presets.

Das Display lässt sich auf eine Verlaufsanzeige der Spannungen und Ströme in Form eines (langsam aufzeichnenden) Oszillogramms umschalten – eine tolle Sache zur Beobachtung von Trends bei der Stromaufnahme, was auf einer Ziffernanzeige eher

mühsam ist. Auch die aktuell abgegebene Leistung wird angezeigt. Eine USB-Ladebuchse liefert zusätzliche 5V bei maximal 1A.

Wenig Überraschendes bei der Regelung: Bei einem Lastwechsel von Null auf 1A und 10V Ausgangsspannung ändert sich diese um etwa 20mV (direkt an den Klemmen gemessen), ein durchaus brauchbarer Wert. Die Lastausregelung erfolgt in etwa 4ms, das Überspringen bei Lastabwurf beträgt etwa 50mV. Die Stromregelung benötigt bei anfangs 10V Ausgangsspannung etwa 30ms, um auf Überlast zu reagieren; für den Test von Bauteilen (z.B. LEDs) ist das deutlich zu lang. Dies ist wie die gemächliche Lastausregelung dem Schaltnetzteil-Prinzip geschuldet.

Störender sind hier eher hochfrequente Reste auf der Ausgangsspannung im Bereich von 200kHz, die bei höheren Strömen durchaus eine Amplitude von einigen hundert Millivolt erreichen können. Das wird zum Problem, wenn man Radios, Funkgeräte, hochwertige Audio-Elektronik oder empfindliche Messgeräte versorgen will. Alle anderen Anwendungen dürften damit klarkommen, das Geld ist dann gut investiert. Immerhin lässt sich das Gerät auch



Bild: Kiprim

**Ausprobiert  
— von Make: —**

über USB fernsteuern, mit der beiliegenden Software lässt sich zum Beispiel die Ladekennlinie eines Akkus aufnehmen, oder man kann einen zeitgesteuerten Ablauf von Ausgangsspannung und -strom programmieren. —cm

|            |                   |
|------------|-------------------|
| Hersteller | Kiprim            |
| URL        | make-magazin/xm99 |
| Preis      | 170 €             |

**Ausprobiert**  
— von Make: —

## Revopoint Mini

### 3D-Scanner für kleine Objekte



Der *Revopoint Mini* ist ein *Structured Light*-3D-Scanner, der ein blaues Lichtmuster projiziert und über eine Stereokamera registriert, wie es durch die Oberfläche des zu scannenden Objekts verzerrt wird. Daraus wird dann die 3D-Form berechnet. Zusätzlich nimmt eine RGB-Kamera die Farben und Muster der Oberfläche auf. Der Mini ist gerade mal 12 Zentimeter breit und eignet sich für Objekte zwischen einem Zentimeter und einem halben Meter Größe. Die optimale Scan-Entfernung zur Oberfläche liegt zwischen 10 und 20 Zentimetern, die maximal erreichbare Genauigkeit gibt der Hersteller mit 0,02mm an.

Um größere Gegenstände führt man den Scanner mit der Hand herum, kleinere platziert man auf einem Drehteller und klickt den Scanner auf dem mitgelieferten Mini-Stativ fest. Das funktioniert alles direkt nach dem Auspacken und der Installation der Software – ohne aufwändige Kalibrierung. Die automatische Kombination verschiedener Scan-Durchgänge desselben Objekts erledigt die Software vollautomatisch. Details fängt der Mini hervorragend ein, wenn man bei den Qualitätseinstellungen in der Software in die Vollen geht. Dann braucht man allerdings einen kräftigen (Gaming-)Rechner (Windows oder macOS). Paradedisziplinen für den Mini sind etwa die Reproduktion von Tabletop-Miniaturen und kleinen Ersatzteilen, auch Schmuckdesigner und Zahntechnikerinnen könnten von diesem Gerät profitieren. —pek

Einen ausführlicheren Testbericht mit Video gibt es online:

► [make-magazin/xm99](https://www.make-magazin.com/magazin/xm99)

|            |                 |
|------------|-----------------|
| Hersteller | Revopoint       |
| URL        | revopoint3d.com |
| Preis      | ab 930 €        |

## Creality Ender 3 S1 Pro

### FDM-3D-Drucker

Der *S1 Pro* ist der große Bruder des beliebten *Ender 3*, macht laut Test der Kollegen von *TechStage* alles so gut wie das Original und druckt dank Hochtemperatur-Extruder auch exotische und besonders stabile Kunststoff-Filamente wie TPU, Nylon und Polycarbonat. Die Ausstattung ist hochwertig: 24-Volt-Netzteil mit 350 Watt, ein Direct-Drive-Dual-Gear-Extruder, Auto-Bed-Leveling, doppelt synchronisierte Z-Achsenführung, PEI-beschichtetes Druckbett aus Federstahl, von dem sich Drucke durch einfaches Biegen nach dem Abkühlen leicht lösen, Riemenspanner, 32-Bit-Mainboard mit leisen TMC-Motortreibern und ein schickes User-Interface auf dem farbigen 4,3-Zoll-Touchscreen. Der Bauraum misst 220mm x 270mm x 220mm.

Der Rahmen ist stabil und insgesamt sehr präzise gefertigt. Das untere Chassis ist komplett in einen Kunststoffkasten eingearbeitet. In diesem befindet sich ein großes Ausziehfach für Kleinteile. Der Druckkopf ist nur mit fünf Schrauben befestigt und soll sich gegen ein Lasermodul (24V/1,6W) tauschen lassen, was für 80 Euro zugekauft werden kann.

Der neu entwickelte Extruder überträgt seinen Anpressdruck auf zwei ineinandergreifende Zahnräder, die das Filament mit bis

zu 80N ins Hotend schieben. Durch die 1:3,5-fache Übersetzung und den kurzen Weg zu dem Hotend sollte er so jedes Filament fest im Griff haben. Gerade dabei zeigt sich der *S1 Pro* laut *TechStage* flexibel, denn er bekommt eine breite Palette von Materialien in den Griff, vom flexiblen TPU bis zum warp-anfälligen ABS – letzteres allerdings nur, wenn man nochmal 80 Euro für eine Einhausung drauflegt. —pek



Bild: Creality

Einen ausführlichen Testbericht gibt es online:

► [make-magazin/xm99](https://www.make-magazin.com/magazin/xm99)

|            |                   |
|------------|-------------------|
| Hersteller | Creality          |
| URL        | make-magazin/xm99 |
| Preis      | 530 €             |

## Anycubic Photon D2

### DLP-3D-Drucker

Der Drucker arbeitet mit flüssigem Kunstharz (Resin) als Material, das bei UV-Belichtung aushärtet. Im *Photon D2* arbeitet dazu eine von *Texas Instruments* gefertigte Kombination aus Mikrospeiegeln und Linseneinheit, wobei das UV-Licht nochmal mit Hilfe einer Linse vergrößert und über einen Spiegelreflektor direkt auf die durchsichtige Unterseite des Harztanks projiziert wird. Anders als beim Vorgängermodell *Anycubic Ultra* funktioniert das beim *Photon D2* gut, wie die Kollegen von *TechStage* bei ihrem Test feststellten. Gegenüber SLA-Druckern mit Displays zur Belichtung sparen DLP-Drucker erheblich an Energie und kommen auch ohne Kühlsystem aus, arbeiten deshalb deutlich leiser.

In den Bauraum passen Werkstücke bis zu einer Größe von 131mm x 73mm x 165mm. Die Bedienung des Druckers erfolgt über ein blau beleuchtetes 2,8-Zoll-LCD. Die Druckdaten sind ausschließlich über einen USB-Stick auf



Bild: Anycubic

**Ausprobiert**  
— von Tech-  
stage

den Drucker übertragbar. Alles in allem wirkt das Gerät einfach und doch solide. Die Führung der Bauplattform läuft zuverlässig über eine einarmige Führung des Z-Arms. Die lasergravierte Bauplattform bietet idealen Halt für die Druckstücke.

Laut Hersteller liegt die xy-Auflösung bei 51µm, sollte damit etwa mit 4K-Auflösung bei SLA-Druckern vergleichbar sein. Im *TechStage*-Test wirkten die Drucke aus dem *Photon D2* eher wie 6K; auch Kanten, Rundungen und feinste Details erschienen präziser und glatter gedruckt. Der Drucker arbeitet nahezu lautlos und verbreitet ohne internen Lüfter auch fast keinen Harzgeruch. —pek

Einen ausführlichen Testbericht gibt es online:

► [make-magazin/xm99](https://www.make-magazin.com/magazin/xm99)

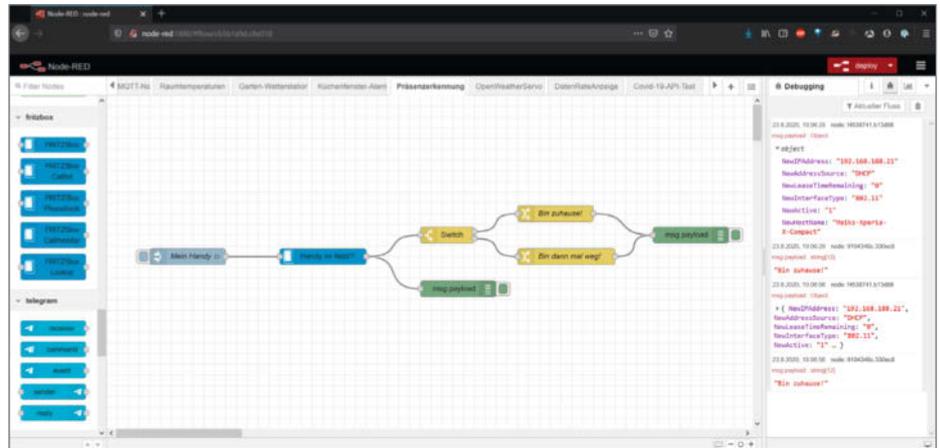
|            |                   |
|------------|-------------------|
| Hersteller | Anycubic          |
| URL        | make-magazin/xm99 |
| Preis      | 729 €             |

# Node-RED 3.0

## Grafische Programmierumgebung für IoT und Smart Home

Auch ohne tiefere Programmierkenntnisse kann man mit *Node-RED* komplexe Anwendungen für IoT und Smart Home entwerfen. Statt mit einem Editor Programmcode zu schreiben, verknüpft man fertig verfügbare grafische Bausteine zu einem *Flow*. Version 3.0 von *Node-RED* hat auf verschiedenen Ebenen Neuerungen zu bieten: So öffnet sich im Editor bei Rechtsklick auf einen Node ein Kontextmenü, das passende Optionen schneller erreichbar macht. Ein neuer Node-Typ für Kreuzungen (*Junctions*) soll bei der besseren Organisation der eigenen *Flows* helfen. Neu angelegte Nodes bekommen automatisch individuelle Namen und der Cursor merkt sich seine Position innerhalb einer Funktion. So kann man beim Editieren direkt dort weitermachen, wo man aufgehört hat, wenn man zwischendrin die Funktion verlassen hat.

Auch gestoppte *Flows* lassen sich editieren und aus dem Editor starten und anhalten – gestoppt behalten sie zudem ihren aktuellen Zustand bei. Diese Funktion ist allerdings



standardmäßig deaktiviert und muss erst angeschaltet werden. Schließlich gibt es auch noch Updates für einige Node-Typen. Die wichtigsten Neuerungen stellt der *Node-RED*-Entwickler Nick O'Leary in einem Video kurz vor (siehe Link).

► [make-magazin/xm99](#)

|            |   |
|------------|---|
| Hersteller | OpenJS Foundation                                     |
| URL        | <a href="https://nodered.org">https://nodered.org</a> |
| Preis      | kostenlos   |

—pek

**Ausprobiert**  
— von Make! —

# EcoFlow River Mini

## Portabler Powerbrick für Haushalt, Making und Campen

Die Powerbank/Powerbrick, *River Mini* von *Ecoflow* besitzt nur einen 210Wh-Akku, bleibt dadurch aber preislich und gewichtsmäßig (2,8kg) im Rahmen. Nicht gespart wurde bei den Features: Die Ladung per 230V mit fast 300W erfolgt schnell. Alternativ kann auch mit Solarstrom oder sonstiger Gleichspannung von 11 - 39V bei 8A über eine XT-Buchse geladen werden, allerdings maximal mit 100W, was die Standard-Solarpaneele mit 300Wp leider ausschließt. Immerhin: (Gebrauchte) Paneele mit 100W sind recht günstig und wesentlich portabler.

Als Ausgänge gibt es drei USB-A-Ports mit 2,4A (insgesamt 24W max.), eine KFZ-Buchse (Zigarettenanzünder) mit 12V und 10A und eine 230V-50Hz-Schuko-Dose mit 300W Dauerleistung und 600W kurzzeitiger Anlaufleistung. In der App kann ein sogenanntes *X-Boost* eingeschaltet werden, dann ist die Frequenz der Ausgangsspannung nicht mehr richtig sinusförmig, aber es kann mehr Leistung für Geräte mit Motor und Heizung abgegeben werden.

Alle Ausgänge sind einzeln per Taster oder App schaltbar, so werden Verluste durch nicht benötigte Ausgänge verhindert.

Als weitere Bedienelemente gibt es nur noch einen Taster zum An- und Ausschalten und einen IoT-Reset-Taster, mit dem man das Gerät per App neu in ein WLAN einbinden kann. Das Display gibt Informationen über die vermutliche Restzeit bei der aktuellen Strombelastung, den Ladezustand des Akkus, die Ein- und Ausgangsströme und ob das Gerät mit einem WLAN verbunden ist.

Die App für Android und iOS verwaltet beliebig viele Geräte von *Ecoflow*. Natürlich muss man einen Account einrichten, mehr als eine E-Mail-Adresse braucht man aber nicht. Der Support ist gut organisiert und zielführend. Neben der App kann man auch einen API-Zugang bekommen, mit dem das Gerät in das eigene Smarthome eingebunden werden kann. Aus der App kann man die Eingänge steuern und bekommt eine detailliertere Übersicht der Ströme auch als Diagramm. Weiterhin sind hier im Detail Konfigurationen zum Verhalten des Geräts, Einschalt Dauern, Auto-Off, Piepen etc. möglich.

Alles in allem ein gutes, tragbares Paket, das in allen Belangen die zugesicherten



Eigenschaften erfüllt. Für mich als Maker und Outdoor-Mensch lädt es diverse Geräte und Werkzeuge, betreibt Lötkolben und gibt Strom für Experimente mit Mikrocontrollern. Die Variante *River Mini Wireless* bietet zusätzlich drahtloses Laden von Mobilgeräten und USB-C.

—caw

► [make-magazin/xm99](#)

|            |         |
|------------|---------|
| Hersteller | EcoFlow |
| Preis      | 330 €   |

# IMPRESSUM

**Make:** Nächste Ausgabe erscheint am 21. Oktober 2022

## Redaktion

**Make: Magazin**  
 Postfach 61 04 07, 30604 Hannover  
 Karl-Wiechert-Allee 10, 30625 Hannover  
 Telefon: 05 11/53 52-300  
 Telefax: 05 11/53 52-417  
 Internet: www.make-magazin.de

**Leserbriefe und Fragen zum Heft:** info@make-magazin.de

Die E-Mail-Adressen der Redakteure haben die Form xx@make-magazin.de oder xxx@make-magazin.de. Setzen Sie statt „xx“ oder „xxx“ bitte das Redakteurs-Kürzel ein. Die Kürzel finden Sie am Ende der Artikel und hier im Impressum.

**Chefredakteur:** Daniel Bachfeld (dab)  
 (verantwortlich für den Textteil)

**Stellv. Chefredakteur:** Peter König (pek)

**Redaktion:** Heinz Behling (hgb), Ákos Fodor (akf), Carsten Meyer (cm), Carsten Wartmann (caw)

**Mitarbeiter dieser Ausgabe:** Beetlebum, Achim Bertram, Michael Gaus, Clemens Gleich, Thomas Hirschberg, Maximilian Kern, Miguel Köhnlein, Manfred Maroszek, Luis Marx, Hans-Jürgen Pretzel, Thomas Ronzon, Michael Schroeder, Felix Watzlawek

**Assistenz:** Susanne Cölle (suc), Christopher Tränkmann (cht), Martin Triadan (mat)

**Leiterin Produktion:** Tine Kreye

**DTP-Produktion:** Martina Bruns, Martin Kreft (Korrektorat)

**Art Direction:** Martina Bruns (Junior Art Director)

**Layout-Konzept:** Martina Bruns

**Layout:** Nicole Wesche

**Fotografie und Titelbild:** Andreas Wodrich

**Digitale Produktion:** Kevin Harte, Thomas Kaltschmidt, Pascal Wissner

**Hergestellt und produziert mit Xpublisher:** www.xpublisher.com

## Verlag

**Maker Media GmbH**  
 Postfach 61 04 07, 30604 Hannover  
 Karl-Wiechert-Allee 10, 30625 Hannover  
 Telefon: 05 11/53 52-0  
 Telefax: 05 11/53 52-129  
 Internet: www.make-magazin.de

**Herausgeber:** Christian Heise, Ansgar Heise

**Geschäftsführung:** Ansgar Heise, Beate Gerold

**Anzeigenleitung:** Michael Hanke (-167)  
 (verantwortlich für den Anzeigenteil),  
 mediadaten.heise.de/produkte/print/  
 das-magazin-fuer-innovation

**Leiter Vertrieb und Marketing:** André Lux (-299)

**Service Sonderdrucke:** Julia Conrades (-156)

**Druck:** Dierichs Druck + Media GmbH & Co.KG,  
 Frankfurter Str. 168, 34121 Kassel

**Vertrieb Einzelverkauf:**  
 DMV DER MEDIENVERTRIEB GmbH & Co. KG  
 Meßberg 1

20086 Hamburg  
 Telefon: +49 (0)40 3019 1800  
 Telefax: +49 (0)40 3019 1815

E-Mail: info@dermedienvertrieb.de  
 Internet: dermedienvertrieb.de

**Einzelpreis:** 12,90 €; Österreich 14,20 €; Schweiz 25,80 CHF;  
 Benelux 15,20 €

**Abonnement-Preise:** Das Jahresabo (7 Ausgaben) kostet inkl. Versandkosten: Inland 77,00 €; Österreich 84,70 €; Schweiz/Europa: 90,65 €; restl. Ausland 95,20 €

Das Make-Plus-Abonnement (inkl. Zugriff auf die App, Heise Magazine sowie das Make-Artikel-Archiv) kostet pro Jahr 6,30 € Aufpreis.

## Abo-Service:

Bestellungen, Adressänderungen, Lieferprobleme usw.:

**Maker Media GmbH**  
**Leserservice**  
 Postfach 24 69  
 49014 Osnabrück  
 E-Mail: leserservice@make-magazin.de  
 Telefon: 0541/80009-125  
 Telefax: 0541/80009-122

Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden. Kein Teil dieser Publikation darf ohne ausdrückliche schriftliche Genehmigung des Verlags in irgendeiner Form reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden. Alle beschriebenen Projekte sind ausschließlich für den privaten, nicht kommerziellen Gebrauch. Maker Media GmbH behält sich alle Nutzungsrechte vor, sofern keine andere Lizenz für Software und Hardware explizit genannt ist.

Für unverlangt eingesandte Manuskripte kann keine Haftung übernommen werden. Mit Übergabe der Manuskripte und Bilder an die Redaktion erteilt der Verfasser dem Verlag das Exklusivrecht zur Veröffentlichung. Honorierte Arbeiten gehen in das Verfügungsrecht des Verlages über. Sämtliche Veröffentlichungen in Make erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes.

Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.

Published and distributed by Maker Media GmbH under license from Make Community LLC, United States of America. The 'Make:' trademark is owned by Make Community LLC Content originally partly published in Make: Magazine and/or on www.makezine.com, ©Make Community LLC 2020 and published under license from Make Community LLC. All rights reserved.

Printed in Germany. Alle Rechte vorbehalten. Gedruckt auf Recyclingpapier.

© Copyright 2022 by Maker Media GmbH

ISSN 2364-2548

## Nachgefragt

Wir haben eine **Vintage-Strecke** im Heft. Von welchen **alten Geräten** konntet ihr euch bislang **nicht trennen?**



**Maximilian Kern**  
 Alzenau, zeigt auf Seite 10, wie man einen Retro-MP3-Player im Design alter Kassettenplayer baut.

Auf meinem Dachboden schlummert noch ein Grundig Röhrenradio aus den 1960ern. Irgendwann möchte ich das Radio wieder zum Laufen bringen und einen Bluetooth-Empfänger einbauen.



**Felix Watzlawek**  
 Schefflenz, lässt auf Seite 56 automatisch Müll sortieren. Ich nutze gerne die alte Ständerbohrmaschine, welche mein Großvater meiner Familie und mir überlassen hat. Diese ist aus massiven Stahl und hat noch fast keine Kunststoffteile verbaut. Von ihr bin ich ein großer Fan.



**Thomas Hirschberg**  
 Adelebsen, eliminiert auf Seite 74 Leitungstörungen mit einem Trick für 10 Cent. Meine 41 Jahre alte Bandmaschine ASC 6002, schwarz und im zeitgenössischen Braun-Design. Ich habe damals meine Revox A77 verkauft. Schade, die hätte ich auch noch behalten sollen.



**Achim Bertram**  
 Lammspringe, schärft ab Seite 114 die Schneiden von Fräswerkzeugen. Als Messerliebhaber habe ich für jeden Zweck die richtige Klinge. Doch ich verwende meist nur das Eine. Mein uraltes, abgewetztes, erstes Ausbeinmesser mit klassischem Holzgriff. Es ist durch nichts zu ersetzen.

## Inserentenverzeichnis

|  |        |
|--|--------|
| AFAG Messen und Ausstellungen GmbH, Nürnberg ..... | 19     |
| Cameo Laser Franz Hagemann GmbH, Stuhr .....       | 27, 49 |
| dpunkt.verlag GmbH, Heidelberg .....               | 67     |
| ELV Elektronik AG, Leer .....                      | 21     |
| Leipziger Messe GmbH, Leipzig .....                | 37     |

|  |     |
|--|-----|
| Reichtel Elektronik GmbH & Co., Sande .....      | 2   |
| Shaper Tools GmbH, Leinfelden-Echterdingen ..... | 9   |
| TUXEDO Computers GmbH, Augsburg .....            | 124 |
| <b>Make:markt</b> .....                          | 45  |

**VIELEN DANK  
an alle diesjährigen  
Teilnehmer!**

**Wir sehen uns im  
nächsten Jahr.**

**Jetzt schon vormerken:  
19.-20. August 2023**

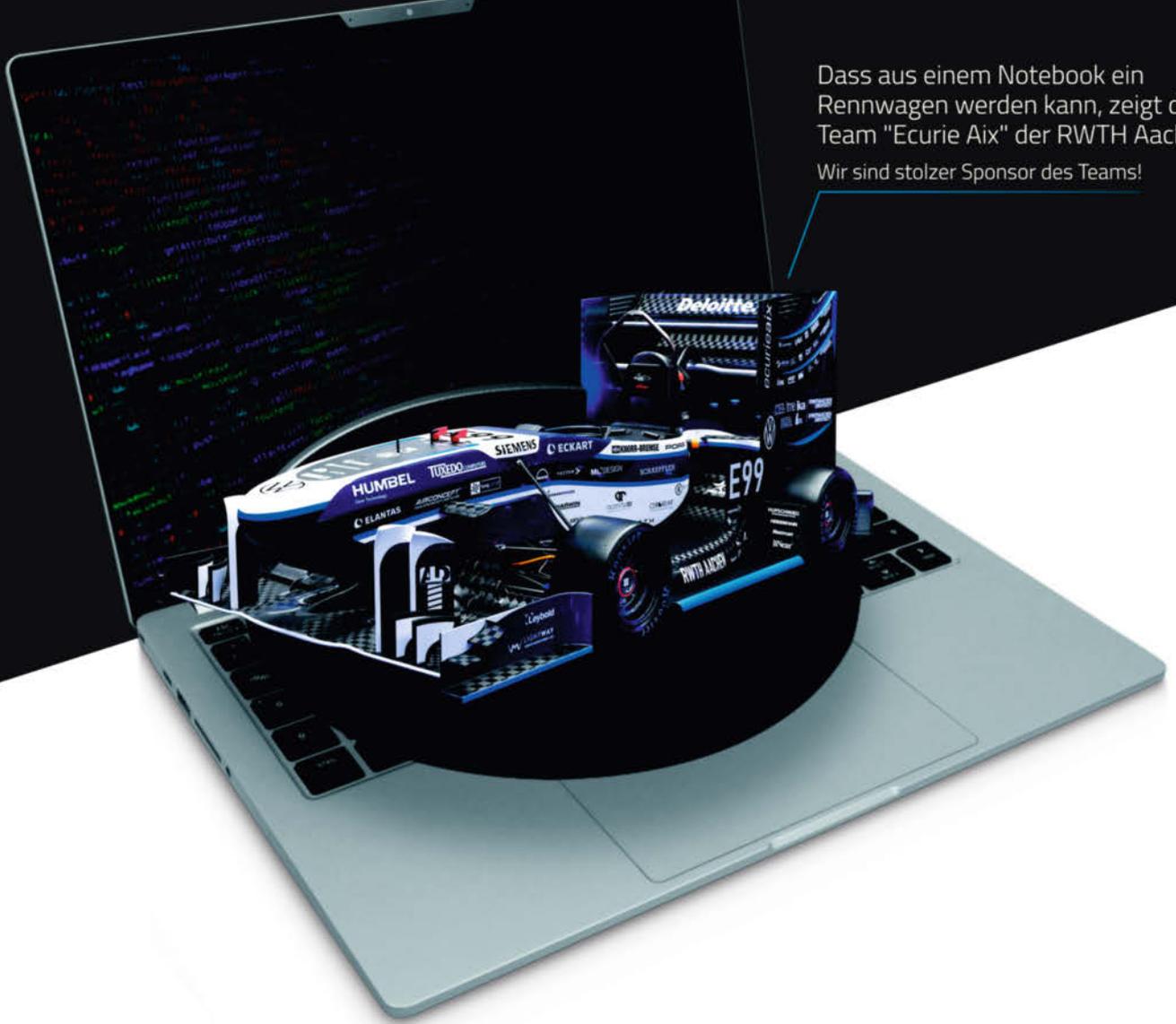
**Hannover**

**Maker Faire®**

Das Format für  
Innovation und  
Macherkultur

**maker-faire.de**

Dass aus einem Notebook ein Rennwagen werden kann, zeigt das Team "Ecurie Aix" der RWTH Aachen. Wir sind stolzer Sponsor des Teams!



# Mach's mit TUXEDO

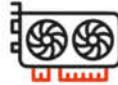
## TUXEDO InfinityBook Pro 14 - Gen7



**Intel Core i7-12700H**  
14 Kerne | 20 Threads



**3K Omnia Display**  
16:10 | 2880 x 1800 Pixel



**GeForce RTX 3050 Ti**  
optionale NVIDIA Grafik



**Magnesiumgehäuse**  
1,5 cm dünn | ab 1 kg leicht



100%  
Linux

5

Jahre  
Garantie



Lifetime  
Support



Gefertigt in  
Deutschland



Deutscher  
Datenschutz



Support  
vor Ort

**TUXEDOCOMPUTERS** 18 JAHRE JUBILÄUM

[tuxedocomputers.com](https://www.tuxedocomputers.com)