Mit vielen **Videos** zu den Projekten

# c't PROJEKTE

Basteln · Programmieren · Selbst bauen

## Basteln mit Elektronik und Holz

Projekte mit Sensor und Mikrocontroller
Selbstgebaut: Luftreiniger und Aerosol-Ampel

## Produzieren mit CNC-Maschinen

CNC-Fräse selbst gedruckt ab 600 Euro
So finden Sie die optimalen Fräsparameter

## Gärtnern mit dem FarmBot

Gartenroboter bauen und programmieren
Vollautomatisch säen, wässern und jäten

## Konstruieren mit CAD-Software

Fusion 360: Werkzeugpfade für CNC-Fräsen planen
Onshape: Dinge erfinden und virtuell zusammenbauen




## Richtig loslegen mit dem Raspi

Eigene Ideen umsetzen mit Python und Elektronik
Strom, Daten, Netzwerk: Raspi aufbohren und tunen

€ 14,90
CH CHF 27.90
AT € 16,40
LUX € 17,10

4 195957 614908 03

www.ctspecial.de

SHAPER

shapertools.com

# BRING FRISCHEN SCHWUNG IN DEINE WERKSTATT!

„Mit Origin fräse ich jetzt einfach alle Teile für meine Projekte selbst.“

Shaper Origin ist die CNC-Handfräse, die Arbeiten in unvergleichlicher Genauigkeit, Flexibilität und Effizienz ermöglicht. Sie ist die ideale Wahl für Macher und Macherinnen, die ihre Werkstatt um CNC-Fähigkeiten und -Präzision erweitern möchten.

Origin im Video:
shapertools.com/3minuten



Mit Origins automatischer Fräskorrektur kannst du Projekte schnell und einfach umsetzen.

# Editorial

## Ein Zeitalter der Maker

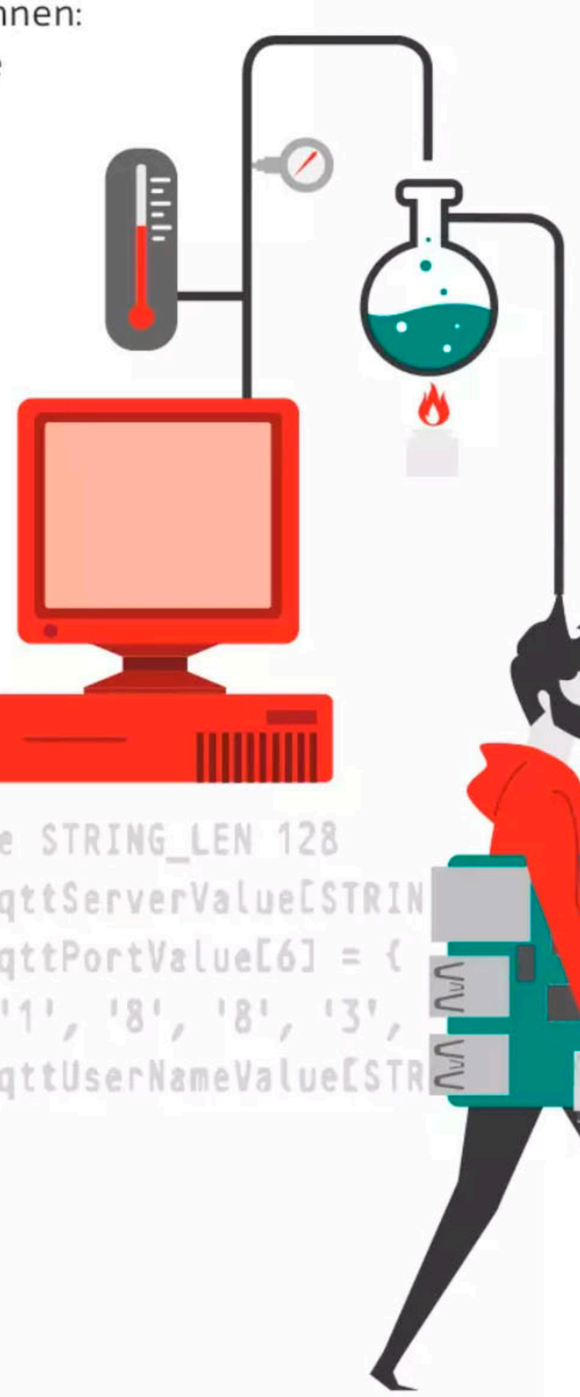
Als ein Affe einen Stock anspitzte, um einen Speer zu bauen, begann das Zeitalter der Menschheit. Seitdem bauen Menschen immer aufwendigere Dinge, um sich das Leben leichter und angenehmer zu machen. Viel hat sich seit dem ersten Speer verändert: Nie war es so einfach, komplexe und tolle Dinge selbst herzustellen. Mit kleinen Computern baut man nützliche und mitdenkende Geräte und programmiert sie exakt nach eigenen Wünschen.

Genau in diesem Punkt sind Selbstbau-Projekte viel besser als gekaufte Gadgets: Wer selbst entwickelt, erschafft Dinge, die perfekt den eigenen Bedürfnissen entsprechen und die zum eigenen Geschmack passen.

Für das perfekte Gadget muss man zum Maschinenbauer, Elektroniker und Produktdesigner werden. Aber auch das war noch nie so einfach wie jetzt. Mit den Projekten, die wir in diesem Heft gesammelt haben, lernen Sie in Windeseile die wichtigsten Komponenten kennen: Raspberry Pi, ESP8266, Arduino Micro und Mega. Sie finden einen schnellen Einstieg in Raspberry Pi OS für Multimedia-Projekte, GRBL für CNC-Fräsen und IotWebConf für selbst gebaute Smart-Home-Sensoren. Und Sie lernen die richtige Software zum Programmieren, Flashen, Konstruieren und Planen kennen.

Alle unsere Projekte können Sie in Dutzenden vergnüglichen Stunden in Ihrer Werkstatt nachbauen. Oder Sie lassen sich von unseren Basteleien ganz bequem auf der Couch lehrreich unterhalten. Vergessen Sie dabei keinesfalls die vielen Videos zu den Artikeln.

In jedem Fall sollten Sie Ihre Ideen verwirklichen und nach der Lektüre eigene Projekte starten!




Pina Merkert

# Inhalt



## LOSLEGEN MIT DEM RASPI

**Der Raspberry Pi ist der beliebteste Einplatinencomputer auf dem Markt. Bevor Sie mit ihm eigene Projekte starten, müssen Sie sich um Stromversorgung, Betriebssystem und Netzwerk kümmern. Mit diesen Tipps gelingt das zügig und sicher.**


**6** Der Raspberry Pi
**10** Grundlagen zur Einrichtung
**14** Stromversorgung
**18** Elektronik anschließen
**22** FAQ: Raspberry Pi
**26** Raspi 400 im Tastaturgehäuse
**30** Compute Module 4


## BASTELN MIT ELEKTRONIK UND HOLZ

**Mit diesen Projekten bauen Sie nützliche Dinge für mehr Bequemlichkeit, mehr Sicherheit und mehr Spaß. Dabei dienen die Projekte auch als Anregungen für Ihre eigenen Ideen, denn das Vorgehen ist immer ähnlich, auch wenn Sie einen anderen Sensor verwenden möchten.**


**38** Lüftersteuerung mit PWM, Poti und WLAN
**46** Luftreiniger mit sehr feinem HEPA-Filter
**56** AeVOC misst Aerosole & Feinstaub
**64** ESP-Projekte mit und ohne WLAN
**68** Mehr GPIO-Pins am Mikrocontroller
**76** Arduino Micro lernt Lautstärkeregler
**80** Arduino-Cockpit für den Flugsimulator


## PRODUZIEREN MIT CNC-MASCHINEN

**CNC-Maschinen bringen Perfektion und Präzision in die Werkstatt. Wenn man sie mit dem richtigen Code versorgt, fräsen sie Teile aus Holz und Metall oder drucken sie aus Kunststoff. Damit das rund läuft, sollten Sie die wichtigsten Parameter kennen.**

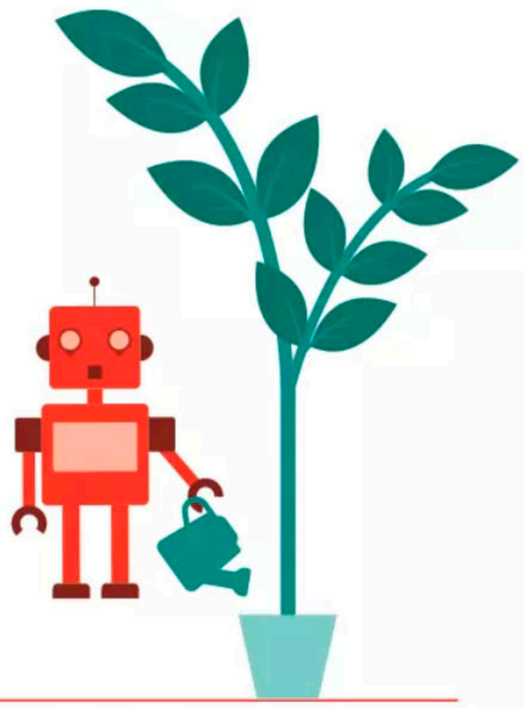

**86** Kaufberatung CNC-Fräse
**90** MPCNC gegen OX-CNC
**96** G-Code programmieren und generieren
**102** Eigene Portalfräse selbst ausdrucken
**110** Stabile Portalfräse OX-CNC selbst bauen
**114** Optimale Parameter für die CNC-Fräse
**122** CNC-Oberfräse mit Shaper Origin


## KONSTRUIEREN MIT CAD-SOFTWARE

**Eigene Ideen in 3D am Computer umsetzen oder Ersatzteile nachbauen – davon träumen viele Bastler. Mit einsteigerfreundlichen CAD-Programmen ist das kein Problem mehr.**


**126** CNC-Programme mit Fusion 360 erzeugen
**136** Onshape: 3D-Designs im Browser
**146** Ersatzteile konstruieren mit Onshape



## GÄRTNERN MIT DEM FARMBOT

**Der Gartenbau-Roboter FarmBot gießt zum Sonnenaufgang, jätet im Regen und überwacht das Beet, wenn man im Urlaub ist. Ein so schlauer Garten nimmt Arbeit ab und pflegt die Pflanzen so regelmäßig wie nie. Bevor das alles klappt, muss man allerdings basteln – vom Wasseranschluss bis zur Energiekette.**


**154** Vollautomatisch gärtnern
**158** Tipps für den Bau eines FarmBot
**166** Roboter programmieren ohne Code
**172** Der FarmBot im Praxiseinsatz
**177** Mit dem Raspi besser bewässern


## ZUM HEFT


**3** Editorial
**178** Impressum


### c't-Projekt-Videos mit Tipps zum Nachbauen

Zu vielen Projekten in diesem Heft gibt es zusätzliche praktische Tipps und Anleitungen per Video. Die jeweiligen Links zu den Videos lassen sich innerhalb der Artikel über den abgedruckten QR-Code mit dem Smartphone oder Tablet direkt aufrufen. Eine Gesamtübersicht aller Videos finden Sie über den hier abgebildeten QR-Code.





# Tipps und Tricks zum Raspberry Pi

**Seit der Vorstellung des Raspberry Pi im Jahr 2012 hat er einen Trend losgetreten, der viele Nachahmer gefunden hat. Durch ihn sind unzählige Projekte entstanden. Ob Sie eines nachbauen oder selbst etwas erfinden wollen: Mit unseren Tipps gelingt der Einstieg.**

Von **Merlin Schumacher**

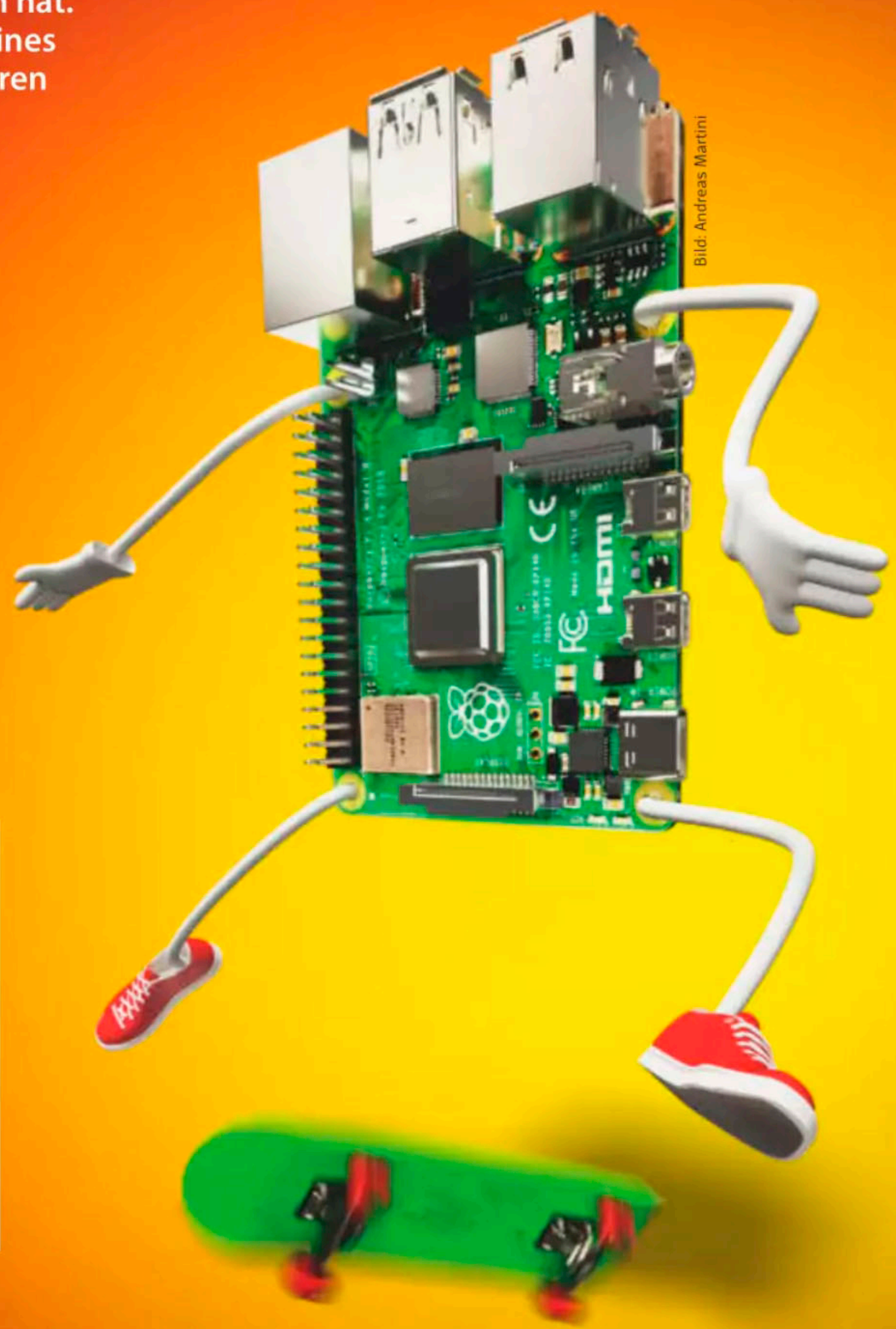


Bild: Andreas Martini


| | |
|---|---|
| Der Raspberry Pi | 6 |
| Grundlagen zur Einrichtung | 10 |
| Stromversorgung | 14 |
| Elektronik anschließen | 18 |
| FAQ: Raspberry Pi | 22 |
| Raspi 400 im Tastaturgehäuse | 26 |
| Compute Module 4 | 30 |

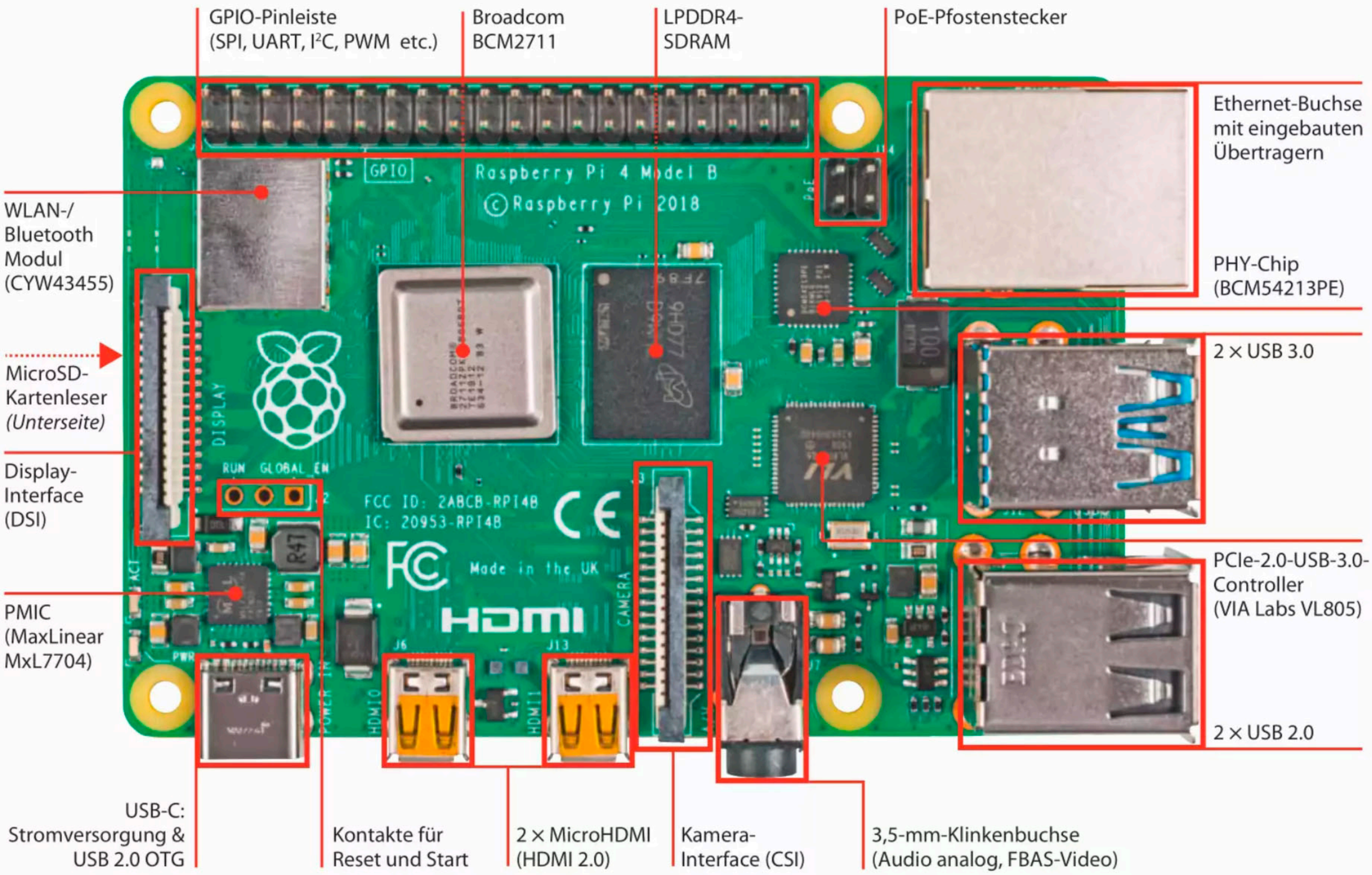
Mit einer 700-MHz-CPU und 256 MByte Arbeitsspeicher lockt man niemanden mehr hinter dem Ofen hervor – selbst für 35 Euro. 2012 war das noch anders. Die Raspberry Pi Foundation hatte quasi eine neue Kategorie von Computern erschaffen und einen Hit gelandet. Der Raspi sollte ein einfacher Lerncomputer werden, ähnlich den Heimcomputern der 80er-Jahre. Programmierbar mit Python, erschwinglich und flexibel. Die herausgeführten GPIO-Pins erlauben Elektronikbasteleien, die mit einem modernen PC unmöglich sind.

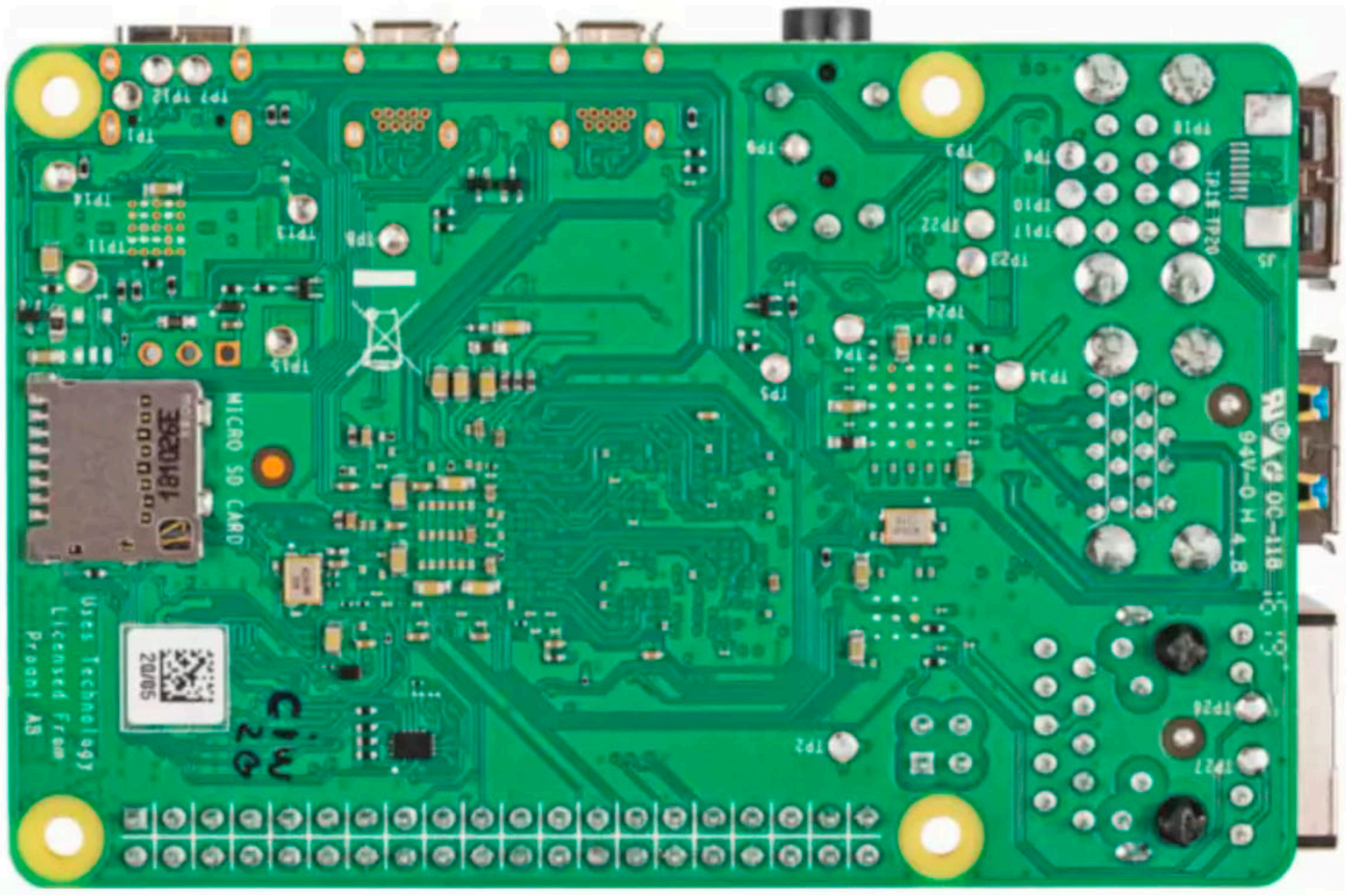
## Note 1, Klassenziel verfehlt

Der Raspi ist eine tolle Erfolgsgeschichte, aber sein Ziel, ein Lerncomputer für Kinder zu sein, hat er völlig verfehlt. Während bei fast jedem Computer- oder Elektronikbastler mittlerweile drei Raspis in der Schublade liegen, findet man ihn in Schulen fast gar

### Die Komponenten des Raspberry Pi 4 Model B

Der Platz auf der Platine des Raspberry Pi 4 ist knapp: Die Entwickler mussten im Vergleich zum Vorgänger etwa Anschlussbuchsen umsetzen und verkleinern, um alle Komponenten unterzubringen.






**Der Steckplatz für die MicroSD-Karte befindet sich auf der Rückseite gegenüber der USB-Anschlüsse.**

nicht. Das ist schade, aber auch ein Hinweis darauf, dass der Raspi mehr ist als die Summe seiner Teile. Er war nie der leistungsfähigste der Mini-PCs, nie der mit den meisten Schnittstellen. Er war immer der vernünftigste, der verlässlichste, flexibelste und der mit der aktivsten Community. Das macht ihn attraktiv und hat dafür gesorgt, dass der kleine Computer 30 Millionen Mal verkauft wurde.

## Mehr von allem

Aus dem einen Kern mit 700 MHz beim Raspi 1 sind inzwischen vier mit 1,5 GHz geworden und das RAM hat sich auf 2 GByte verachtfacht – für noch immer nur 35 Euro. Wer weitere 40 Euro drauflegt, bekommt sogar 8 GByte RAM. Für anspruchsvollere Projekte kann sich das lohnen.

Viel Kritik gab es an den langsamen Schnittstellen des Raspi. USB 2.0 und Fast-Ethernet sind schon lange nicht mehr befriedigend. Diese Kritikpunkte hat die Raspberry Pi Foundation mit der Version 4 mit rasend schnellem USB 3.0 und Gigabit-Ethernet endlich ausgeräumt. Dazu kommen 4K-Unterstützung und ein zweiter Display-Ausgang. Solche Upgrades bekommt der Raspi dann, wenn es wirtschaftlich ist, denn an den 35 Euro hält man fest.

Die Tatsache, dass auf dem Raspi meist ein Linux-System läuft, könnte Ihre Berührungsängste vergrößern. Die Einrichtung und Bedienung eines Raspi ist aber sogar leichter als die eines Windows-PC. Die grundlegende Einrichtung des Raspberry Pi inklusive der Erzeugung einer bootfähigen SD-Karte sowie der Einrichtung von WLAN und Diensten zur Fernsteuerung erklärt der Artikel auf Seite 10.

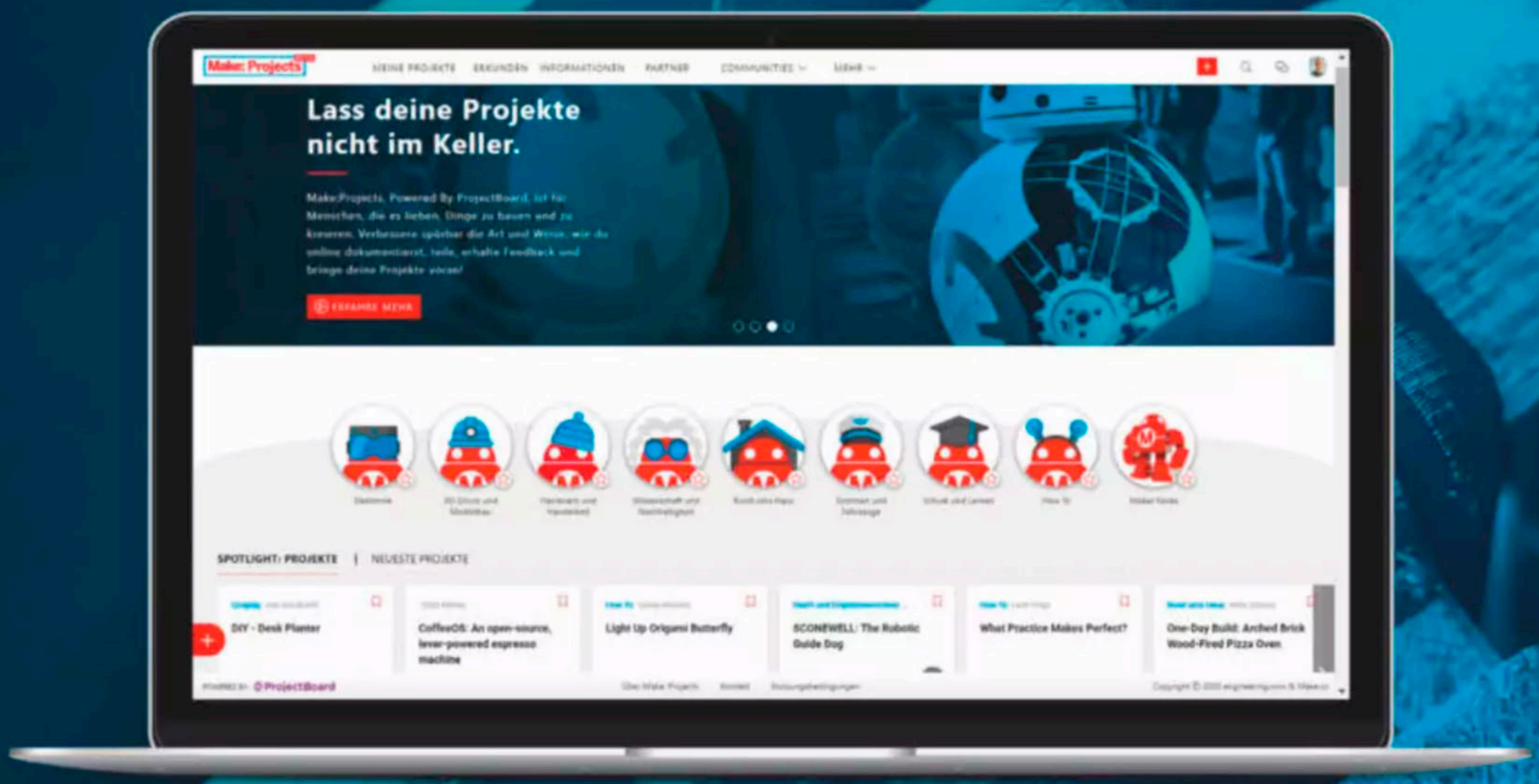
## Angeschlossen

Der erste Weg zur Stromversorgung des Raspi ist üblicherweise der USB-Anschluss. Aber es geht auch anders. Nicht nur über die GPIO-Pinleiste, sondern auch Power-over-Ethernet, also per Netzwerkkabel, kann man den Kleinen versorgen (siehe S. 14). Apropos GPIO-Pinleiste! Im Artikel „Versuchs-Hut“ auf Seite 18 zeigen wir Ihnen, wie Sie Elektronikexperimente mit dem Raspi durchführen und diese mittels Python steuern können. Die häufigsten Fragen haben wir in einer FAQ auf Seite 22 gesammelt.

Wenn Sie Ihren ersten Raspi am Laufen haben, mag die Frage kommen: Was nun? Kein Problem, denn Ideen für Raspi-Projekte gibt es genug. Die Projekte reichen dabei von kinderleicht bis komplex. Einsteiger nutzen den Raspi im Tastaturgehäuse (siehe S. 26) erfahrene Bastler schrauben den Raspi 3 in den FarmBot oder nutzen ihn zur Steuerung ihrer Gartenbewässerung (ab S. 154).

Der Raspi-Zug ist noch lange nicht abgefahren. Springen Sie ruhig auf und basteln Sie mit, denn irgendwie ist der Raspberry Pi doch ein Lerncomputer geworden. Aber für alle und nicht nur für Kinder. (jam) c't

# Make: Projects




## Zusammen mehr machen –
## Teile deine Ideen, hol dir Inspiration, revolutioniere deine Projekte

**... all das gemeinsam auf Make Projects!**

**www.makeprojects.com/de**

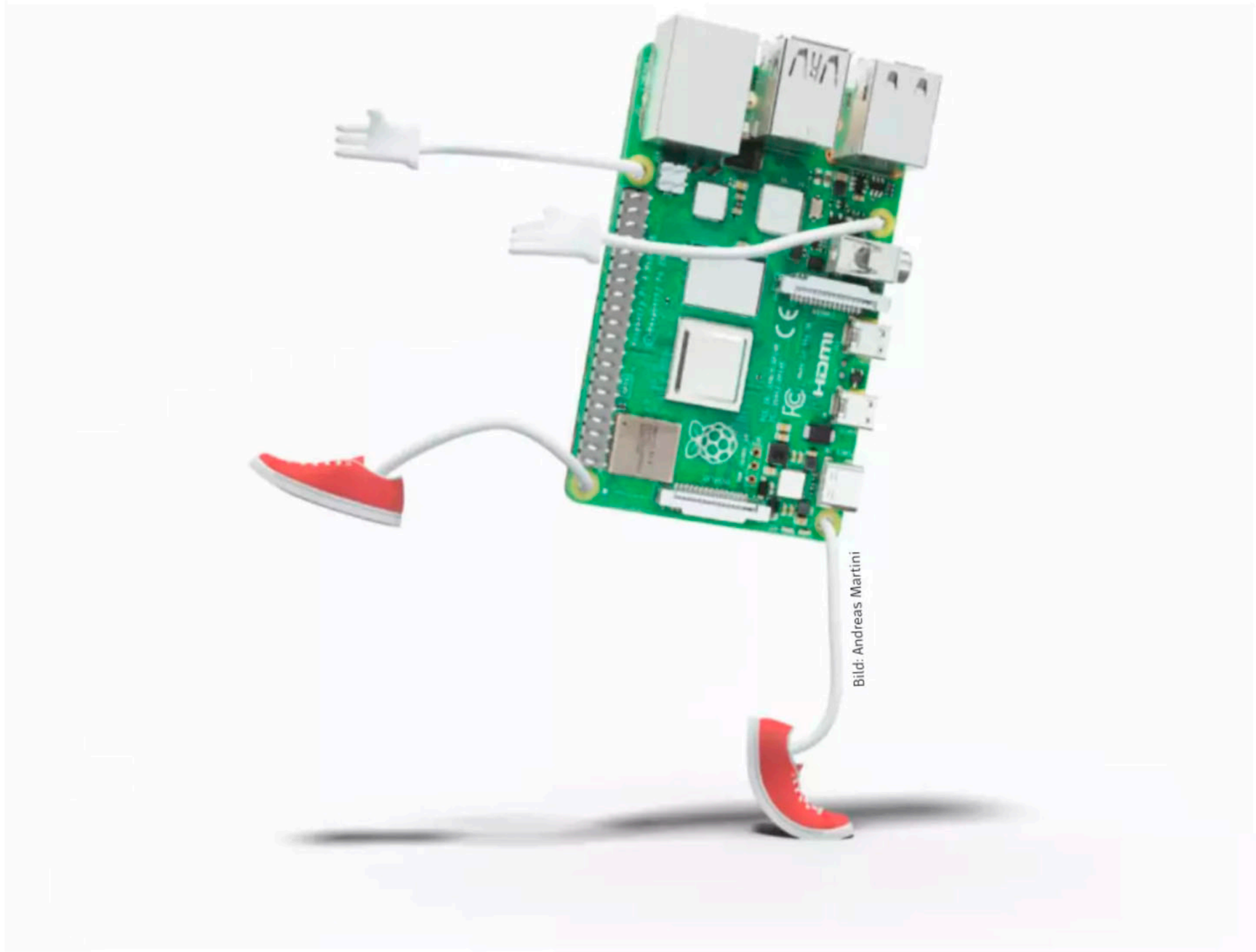


Bild: Andreas Martini

# Die ersten Schritte mit dem Raspberry Pi

**Falls Ihr Erstkontakt zu einem Raspi bislang nur daran scheitert, dass der angeblich so kompliziert einzurichten ist: Probieren Sie es ruhig mal! Die Installation ist einfach und Sie haben den Raspi bestenfalls in weniger als 15 Minuten am Start.**

Von **Lutz Labs und Ronald Eikenberg**

Der Einstieg in die Welt des Raspi geht auch Laien schnell von der Hand. Man muss nicht groß in Konfigurationsdateien herumfuhrwerken oder komplexe Programme beherrschen. Wir zeigen, wie Sie den Raspi mit einem Betriebssystem versorgen, sein WLAN einrichten und ihn fit für die Fernsteuerung vom PC aus machen.

Einsteiger sollten sich nicht nur den Raspberry Pi zulegen, sondern am besten gleich ein Raspi-Paket. Das besteht meistens aus dem Raspi selbst (2 GByte RAM reichen für die meisten Anwendungen völlig aus) und einigem Zubehör: dem Netzteil, einem Gehäuse, einer MicroSD-Karte und einem HDMI-Kabel für den Monitoranschluss. Das HDMI-Kabel ist recht

speziell, weil es auf der Raspi-Seite einen eher seltenen Micro-HDMI-Stecker hat – mit einem Adapter würde es auch klappen, aber nur mit einer der beiden HDMI-Buchsen des Raspi, denn diese liegen sehr nah beieinander. Wenn Sie planen, den Raspi über das Netzwerk zu steuern, noch eine MicroSD-Karte in Ihrem Fundus haben und auf das Gehäuse keinen Wert legen, kommen Sie günstiger weg. Das Netzteil sollte mindestens 10, besser 15 Watt leisten.

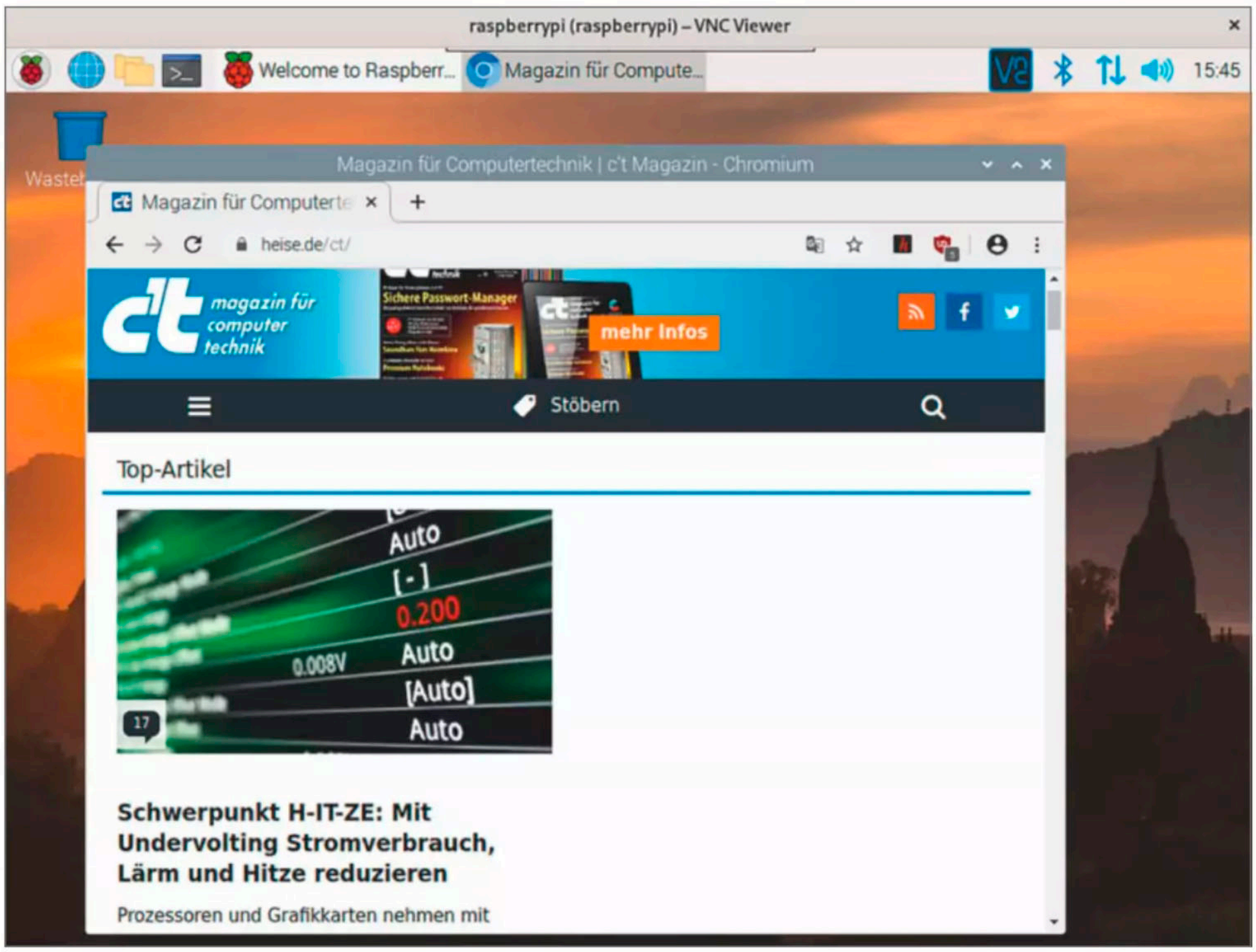
Die Mindestgröße für die MicroSD-Karte beträgt 8 GByte, mehr schadet nicht. Die Installation des Betriebssystems – im einfachsten Fall das von der Raspberry Pi Foundation bereitgestellte – erfolgt über das Schreiben des Image auf die Speicherkarte. Das erledigen Sie mit einem PC, für den benötigen Sie also noch einen Speicherkartenleser.

Sie brauchen den Raspi nicht unbedingt mit einer eigenen Tastatur, Maus und einem Monitor versorgen. Er kann auch ganz ohne Peripherie ferngesteuert werden. Dazu später mehr.

## Raspberry Pi OS statt Raspbian

Lange Zeit hieß das bevorzugte Betriebssystem für den Raspberry Pi  Raspbian, später hat die Raspberry Pi Foundation es umbenannt: Um Verwirrungen zu vermeiden, heißt es ab sofort Raspberry Pi OS. Das Betriebssystem ist eine angepasste Variante der Linux-Distribution Debian.

Gleichzeitig hat die Foundation eine frühe Betaversion einer 64-Bit-Variante von Raspberry Pi OS veröffentlicht. Einziger Vorteil dieser Version  ist, dass sie den gesamten Speicher des Raspi einem einzelnen Prozess zuordnen kann – die 32-Bit-Version beschränkt den Speicher auf 3 GByte pro Prozess. Das war bislang noch kein großes Problem, denn den Raspi gab es nur mit maximal 4 GByte RAM. Seit kurzem liefert die Foundation auch eine Version mit 8 GByte RAM.Die 1-GByte-Version wird hingegen bereits seit Monaten nicht mehr produziert.




**Raspberry Pi OS bringt alles mit, was man für die ersten Schritte mit dem Raspi braucht.**

## Varianten von Raspberry Pi OS

Die Raspberry Pi Foundation bietet das eigene Betriebssystem in drei Varianten an: als Lite-Version ohne grafischen Desktop, mit dem Desktop und einer Software-Grundausstattung und mit einer größeren Auswahl an Programmen. Zwar braucht die letztgenannte Variante rund 5,8 GByte Speicherplatz und damit mehr als das Doppelte der einfachen grafischen Version, doch ist dann in einem Rutsch alles Wesentliche dabei.

Weiterhin listet die Foundation auf ihrer Website (siehe ct.de/wt19) noch Links zu einigen Ubuntu-Versionen sowie Distributionen für spezielle Zwecke, etwa die Multimedia-Distro LibreELEC oder das Open Source Media Center OSMC.

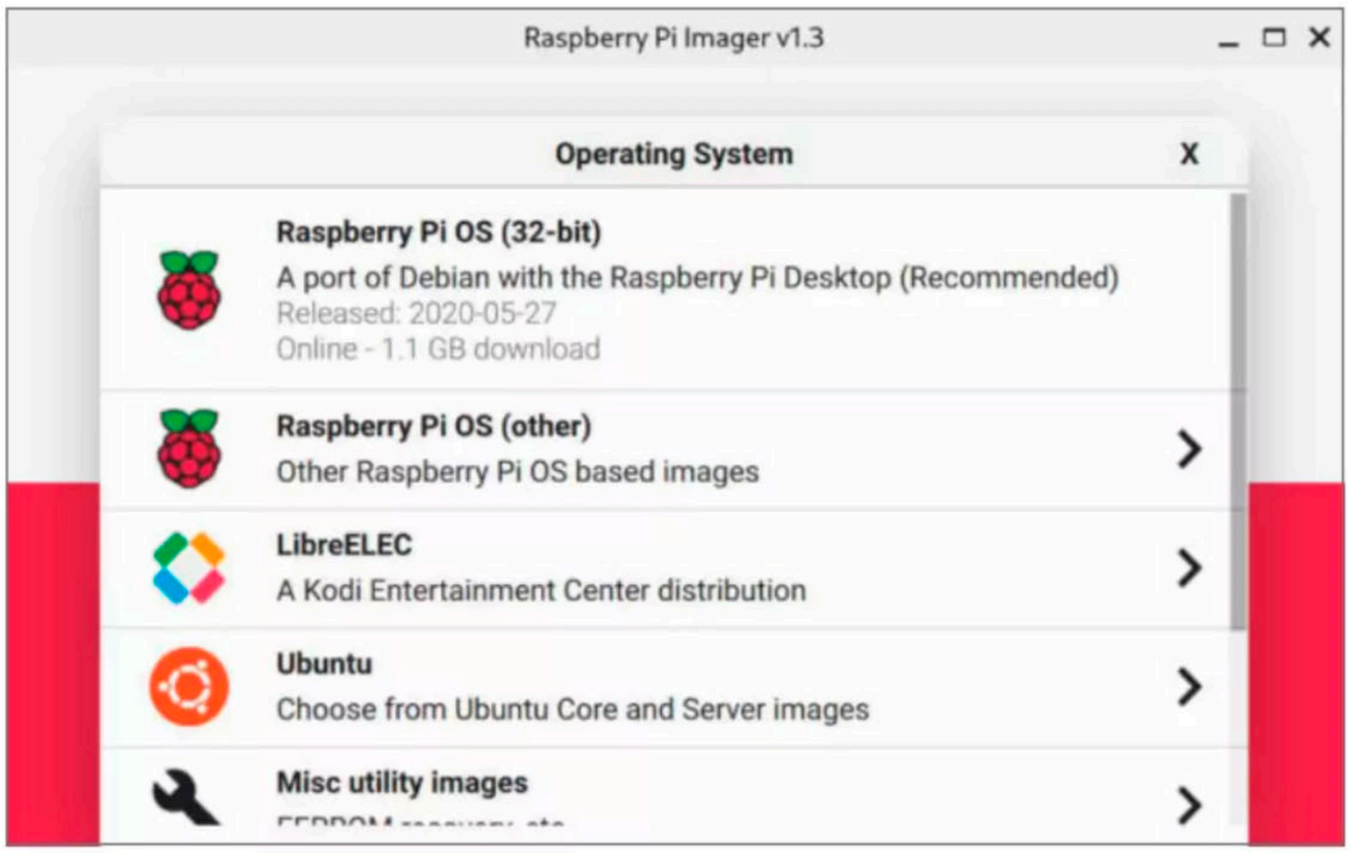
## Installation von Raspberry Pi OS

Am einfachsten gelingt die Installation mit dem Raspberry Pi Imager, den die Foundation für macOS, Ubuntu und Windows zum Download auf ihrer Website anbietet (siehe ct.de/wt19). Natürlich klappt es auch mit anderen Imager-Apps, dann aber muss man das benötigte Image selbst herunterladen.

Legen Sie die MicroSD-Karte in den Kartenleser, schließen Sie ihn an den PC an und starten Sie den Imager. Wählen Sie nun das gewünschte System und das richtige Laufwerk aus.

Ganz neu ist ein bisher verstecktes Menü, mit dem Sie das Betriebssystem direkt vor dem ersten Start einrichten können. Die erweiterten Optionen erreichen Sie derzeit ausschließlich über die „magische Tastenkombination“ (Originalzitat der Entwickler) Strg+Umschalt+X. Es öffnet sich eine lange Liste mit allem, was das Herz begehrt. Zunächst stellen Sie ganz oben unter „OS-Modifizierungen“ ein, ob Sie die Einstellungen nur einmalig („Nur für diese Sitzung“) oder dauerhaft („Immer verwenden“) mit auf die SD-Karte schreiben möchten. Die erste Option „Overscan deaktivieren“ ist dann interessant, wenn Sie einen Monitor an Ihren Raspi anschließen wollen: Sie schalten damit den schwarzen Rahmen ab, der standardmäßig rund um das ausgegebene Bildsignal angezeigt wird. Darunter legen Sie mit „Hostname“ einen individuellen Hostnamen fest.

Standardmäßig horcht der Raspi auf den Namen „raspberrypi“, beim Einsatz mehrerer Raspis im gleichen Netz führt das jedoch schnell zu Verwirrung. Legen Sie am besten einen einsatzbezogenen Namen fest, etwa „raspi-smarthome“, „raspi-server“ oder raspi-kodi“. Wenn Sie Ihren Raspberry übers




**Der Raspberry Pi Imager macht das Aufspielen eines Betriebssystems einfach und bietet auch Alternativen zu Raspberry Pi OS an.**

Netzwerk steuern möchten, dann wollen Sie wahrscheinlich „SSH aktivieren“ über das gleichnamige Häkchen, um den SSH-Server dauerhaft auf dem Standardport 22 zu aktivieren. Standardmäßig ist auf dem System ein Nutzer „pi“ mit dem Passwort „raspberry“ eingerichtet, was natürlich alles andere als sicher ist, weil das jeder kennt. Sie sollten daher unter „Password zur Authentifizierung verwenden“ ein individuelles Passwort vorgeben. Selbst wenn Sie dieses Passwort für alle Ihre Raspis nutzen, ist das allemal sicherer als das vorgegebene Standardpasswort.

## Fernsteuerung

Zur Steuerung eines Raspis ohne Eingabegeräte und Bildschirm können Sie auf der Kommandozeile das Werkzeug SSH (Secure Shell) benutzen oder für eine grafische Steuerung VNC verwenden, das den gesamten Desktopinhalt des Raspi überträgt. Ein SSH-Client wird bei allen modernen Betriebssystemen mitgeliefert, einen VNC-Client müssen Sie üblicherweise nachinstallieren.

Um den VNC-Zugang auf dem Raspi einzurichten, loggen Sie sich zunächst per SSH in den Raspi ein. Dazu rufen Sie auf einer Kommandozeile folgenden Befehl auf:

```
ssh pi@<IP des Raspberry Pi>
```

Den Raspi-Benutzernamen `pi` müssen Sie bereits über die Befehlszeile übergeben, weil Ihr System sonst Ihren PC-Benutzernamen nutzen möchte. Bestätigen Sie, dass Sie dem Gerät vertrauen, und tippen Sie anschließend das Passwort ein. Rufen Sie nun das Raspi-Konfigurationwerkzeug auf:

```
sudo raspi-config
```

und aktivieren Sie über Punkt 5 (Interfacing Options) den VNC-Server. Wenn kein Monitor angeschlossen ist, zeigt der Raspi aber auch über den VNC-Remote-Zugang nichts an, bevor Sie dafür nicht eine Auflösung festgelegt haben. Gehen Sie also im Hauptmenü auf Punkt 7 (Advanced Options) und dort A5 (Resolution). Wählen Sie hier nun eine passende Auflösung aus und beenden Sie das Konfigurationsprogramm. Beim Verlassen werden Sie gefragt, ob Sie den Raspi neu starten möchten: Bestätigen Sie mit „Yes“.

Für die Fernsteuerung installieren Sie auf dem PC nun einen VNC-Viewer für Ihr Betriebssystem (siehe ct.de/wt19) und richten darin eine neue Verbindung ein. Nutzen Sie die bekannte IP-Adresse, den Benutzernamen `pi` und Ihr Passwort, als Name für die Verbindung wählen Sie etwa Raspi. Wenn Sie sich nun das erste Mal per VNC mit dem Raspi verbinden, sollte dessen Desktop mit der Software für die Ersteinrichtung erscheinen. Wählen Sie das passende Land, der Rest geschieht mehr oder weniger automatisch. Die Suche nach einem WLAN können Sie überspringen – Sie sind ja schon verbunden. In jedem Fall sollten Sie das Passwort für den Benutzer `pi` ändern, wie es im Assistenten angeboten wird.

Zur Sicherheit können Sie jetzt dem Raspi noch eine feste IP verpassen, damit er immer unter der gleichen Adresse erreichbar ist. Klicken Sie dazu mit der rechten Maustaste auf das WLAN-Symbol oben rechts und wählen Sie das richtige Interface aus. Nutzen Sie dann eine freie Adresse aus dem Adressbereich Ihres DSL-Routers.

Et voilà, das war’s. Viel Spaß mit dem Raspi!

(jam) c't

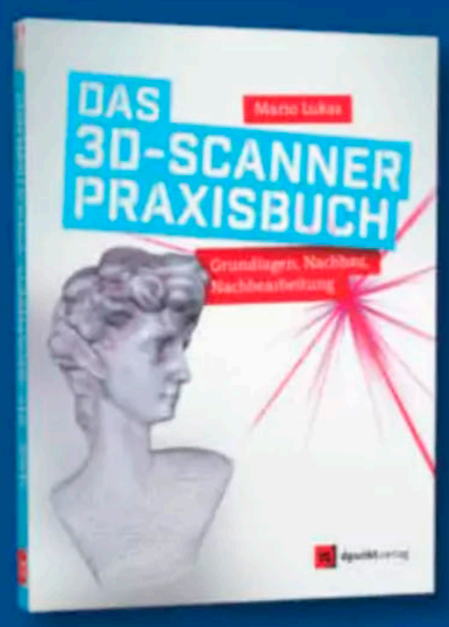
**Hilfreiche Downloads**

ct.de/wt19





ab 25,99 €
Print | E-Book | Bundle
*ISBN 978-3-86490-621-3*




ab 27,99 €
Print | E-Book | Bundle
*ISBN 978-3-86490-640-4*

**Für Heimwerker und Bastler**

**Bundle up!**
Print & E-Book
www.dpunkt.de




ab 25,99 € · Print | E-Book | Bundle
*ISBN 978-3-86490-752-4*




ab 27,99 € · Print | E-Book | Bundle
*ISBN 978-3-86490-800-2*




Print: 14,95 €
*ISBN 978-3-86490-549-0*

dpunkt.verlag

dpunkt.verlag GmbH
Wieblinger Weg 17
D-69123 Heidelberg
fon: 0 62 21 / 14 83 40
fax: 0 62 21 / 14 83 99
e-mail: bestellung@dpunkt.de
**www.dpunkt.de**

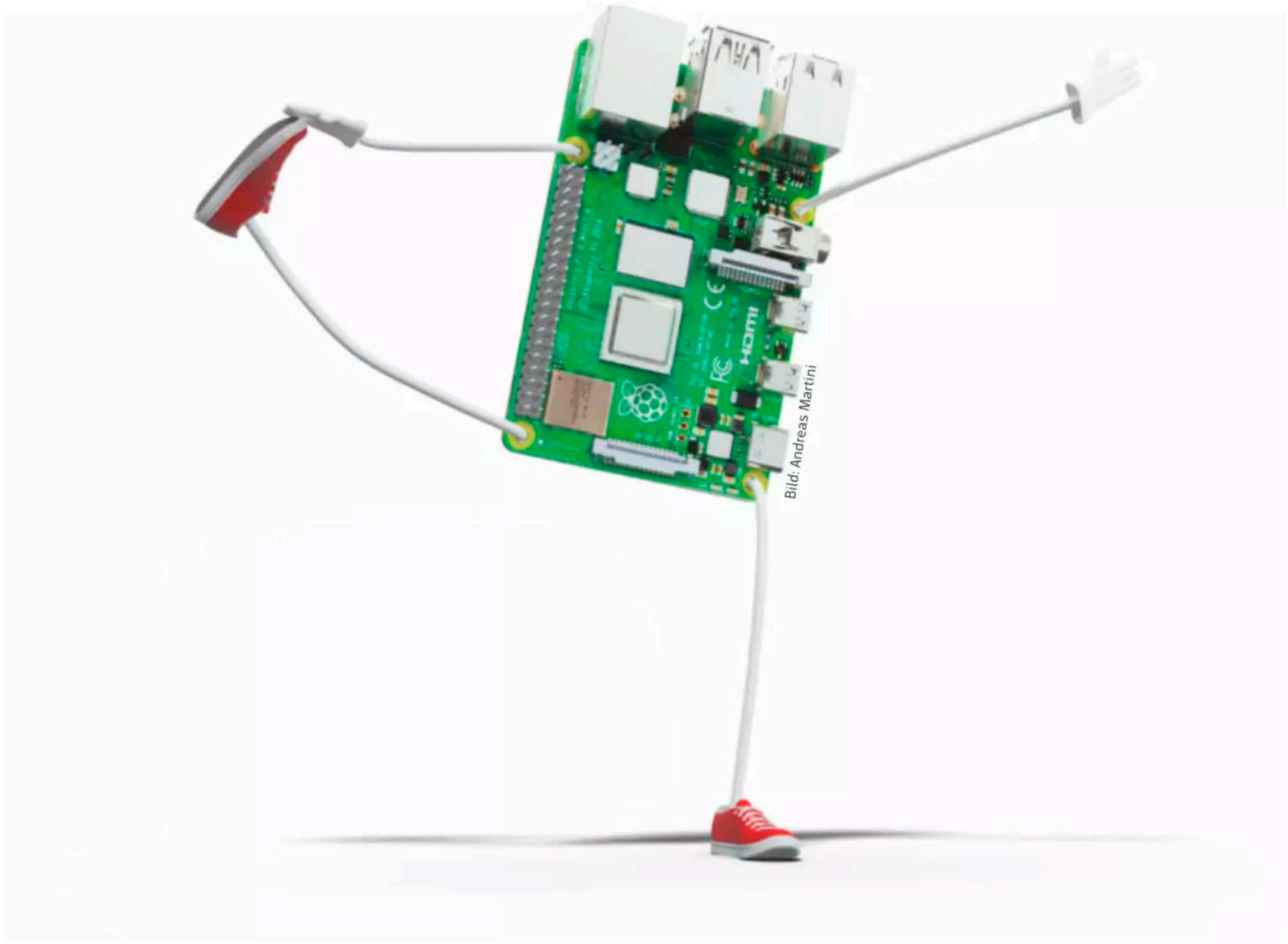


Bild: Andreas Martini

# Raspi 4 alternativ mit Strom versorgen

**Es muss kein Steckernetzteil sein: Der Raspi 4 lässt sich auch via USB-C von einem iPad, aus dem Netzwerk oder über den GPIO-Anschluss mit Strom versorgen.**

Von **Mirko Dölle**

Der Raspberry Pi 4 ist das bislang stromhungrigste Modell der Raspi-Familie, je nach Erweiterungen und angeschlossener Peripherie sind es bis zu 15 Watt Dauerlast, die der Mini-Rechner benötigt. Diese 15 Watt sind aber keine harte Grenze. Schließt man etwa eine externe Festplatte an, kann der Strombedarf noch darüber hinaus gehen, genau wie bei kurzzeitigen Lastspitzen. Die meisten Steckernetzteile reagieren auf solche Lastspitzen mit einem kurzzeitigen Einbrechen der Versorgungsspannung, was der Raspi mit einer blinkenden oder verlöschenden roten LED und einer Drosselung der CPU quittiert – und so meist einen Absturz des Betriebssystems vermeiden kann.

Bei älteren Raspi 4 können ausgerechnet leistungsstarke USB-C-Netzteile und hochwertige USB-C-Kabel dazu führen, dass der Raspi im Zweifel gar keinen Strom bekommt. Schuld daran ist ein Schaltungsfehler in der ersten Fertigungsserie. Wie Sie solche Raspis erkennen, steht in der FAQ auf Seite 22. Einfacher ist, ein billiges USB-C-Kabel für unter 10 Euro zu benutzen, das keine integrierten Chips besitzt, oder auf das Original-Netzteil der Raspberry Pi Foundation zurückzugreifen.

## Zu kurz gekommen

Es ist aber auch möglich, den Raspi über den GPIO-Anschluss zu versorgen. Die vergleichsweise niedrige Versorgungsspannung von nominell 5 Volt sorgt allerdings dafür, dass dünne Stromkabel zum Flaschenhals werden. Bei 3 Ampere und mehr kommt es schnell zu einem erheblichen Spannungsabfall, weshalb wir Litzen mit mindestens 0,75 $mm^2$ Leitungsquerschnitt empfehlen, damit der Raspi nicht einen Gang zurückschaltet. Mit folgendem Befehl im Terminal können Sie jederzeit nachsehen, ob es aktuell ein Versorgungsproblem gibt oder seit dem letzten Neustart gab:

```
/opt/vc/bin/vcgencmd get_throttled
```

Beim Rückgabewert, etwa `0x50000` oder `0x50005`, sind nur die erste und die letzte Stelle des hexadezimalen Werts interessant – also die `5` und die `0` in `0x50000` und die `5` und die `5` in `0x50005`. Die `5` an der ersten Stelle bedeutet, dass seit dem Systemstart mindestens einmal Unterspannung auftrat und die CPU deshalb gedrosselt wurde – der Raspi also nicht genug Strom bekommen hat.

Eine zweite `5` würde bedeuten, dass die CPU aktuell gedrosselt ist und akut eine Unterspannung vorliegt – meist, weil das System gerade ausgelastet ist, weshalb auch die rote LED abgeschaltet ist oder im Takt der Belastung blinkt. Schließt man dann noch eine externe SSD oder einen stromhungrigen USB-Stick an, kann es zum Absturz des Raspi kommen. Manchmal bootet der Raspi auch neu, nur um erneut in eine Unterversorgung zu geraten – so entstehen Endlos-Bootschleifen, die schlimmstenfalls das Dateisystem zerstören können.

## Versorgungslücke schließen

Wenn Sie den Raspi etwa als USB-Netzwerkgerät an einem Notebook oder iPad Pro betreiben, übernehmen Notebook und iPad auch die Stromversorgung. Doch gerade ältere Notebooks sind nicht darauf ausgelegt, so hohe Ströme zu liefern – und beim iPad Pro ist das Problem, dass der Raspi den einzigen USB-Anschluss blockiert, weshalb sich das iPad nicht mehr aufladen lässt.

Abhilfe versprechen in beiden Fällen USB-C-Hubs mit Power Delivery (PD). Allerdings konnten wir keinen finden, der in allen Betriebszuständen genügend Strom für den Raspi 4 lieferte – weshalb die rote Betriebs-LED des Raspi gelegentlich blinkte oder sogar ganz aus ging.

Beim Betrieb am iPad Pro war Apples USB-C Digital AV Multiport Adapter für stolze 80 Euro noch die beste Wahl, der neben dem USB-C-Anschluss für das Netzteil noch eine HDMI- und eine USB-A-Buchse zum Anschluss des Raspi besitzt. Mit einem USB-A-auf-C-Adapterkabel kam es beim Betrieb eines Raspi 4 nebst externer SSD nur selten zu Engpässen, Abstürze gab es keine. Beim Goobay 62111 für 50 bis 60 Euro und dem halb so schnellen, aber auch nur halb so teuren Nachfolger Goobay 62112 hingegen blinkte die Power-LED deutlich öfter, trotzdem genügte es, um absturzfrei auch eine externe SSD am Raspi zu betreiben. Da die beiden Goobay-Hubs gleich zwei USB-C-Buchsen haben, benötigen Sie kein Adapterkabel zum Anschluss des Raspi. Für ältere Laptops, die über einen USB-Port nicht genug Strom bereitstellen, gibt es zwei weitere Alternativen: ein Y-Kabel, über das der Raspi von zwei USB-Anschlüssen parallel versorgt wird, oder ein herkömmlicher USB-Hub mit genügend Leistung.

Völlig ungeeignet sind hingegen die oft als „USB-C-Splitter“ angepriesenen Adapter, mit denen man einen USB-C-Kopfhörer und ein USB-C-Ladekabel anschließen kann: Diese nutzen aus, dass bestimmte Datenleitungen des USB-C-Anschlusses zur Übertragung analoger Audiosignale verwendet werden können. Dementsprechend sind an dem einen vermeintlichen USB-C-Anschluss nur die Pins für die Stromversorgung beschaltet und an dem anderen nur die Pins für die analogen Audiosignale herausgeführt. Schließen Sie dort statt eines Kopfhörers ein anderes USB-C-Gerät an, bekommt es meist nicht einmal Strom, keinesfalls aber eine USB-Datenverbindung.

## Abhängigkeiten

Auf den Batteriebetrieb ist der Raspi von Haus aus nicht ausgelegt, es gibt keinen Anschluss für einen Akku und somit auch keine Ladeschaltung. Diese

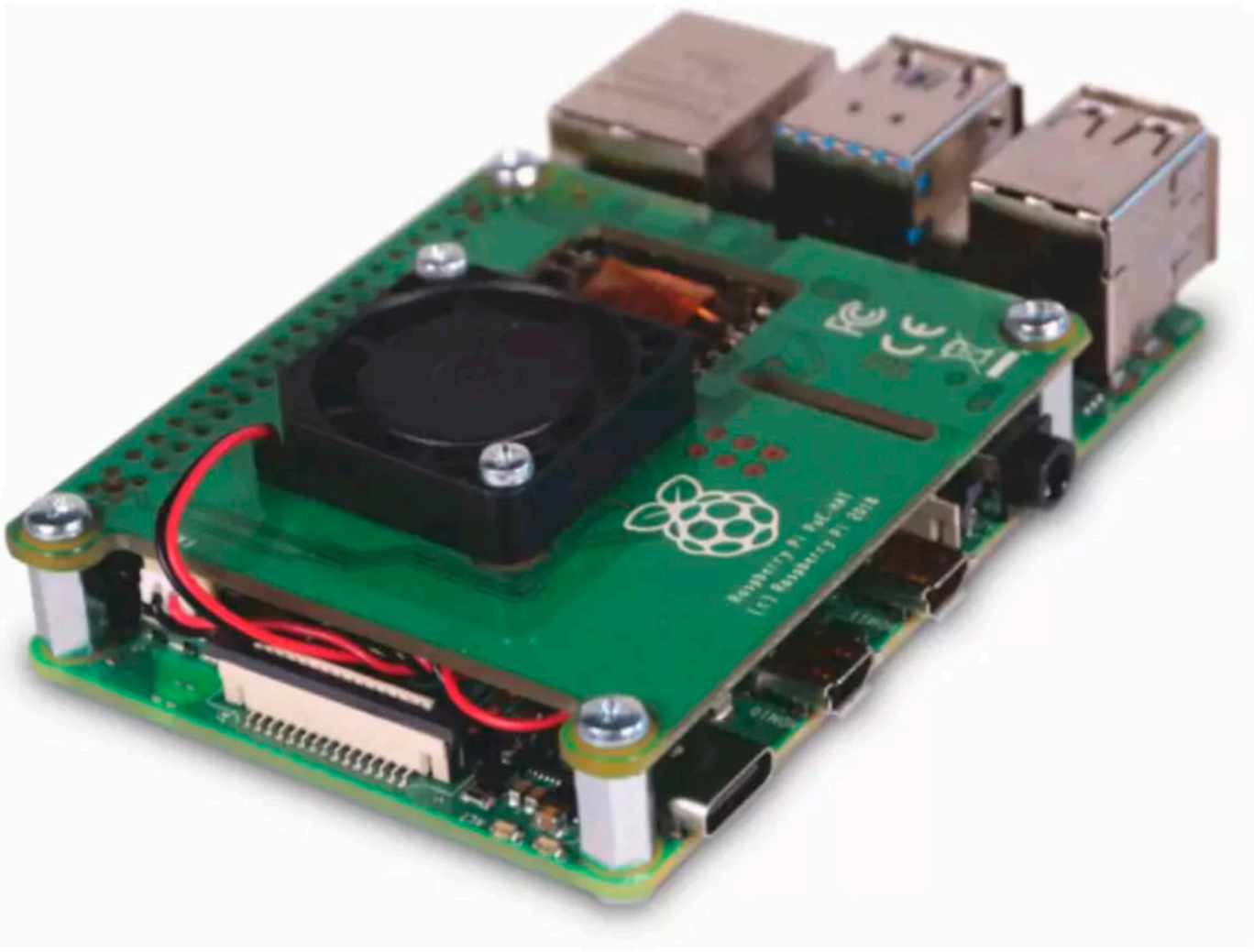
enthält das StromPi 3 Hat von Joy-It für fast 40 Euro, zusammen mit einem Weitbereichs-Eingang für Spannungen von 6 bis 61 Volt. Gegenüber früheren Modellen kann die aktuelle Version 3 des Hat bis zu 3 Ampere Strom liefern, wie es bei voller Auslastung des Raspi 4 erforderlich ist. Ein passender 1000-mAh-Akku kostet weitere knapp 30 Euro, versorgt den Raspi aber nur für 30 bis 60 Minuten – ein doppelt so großer Akku kostet fast 50 Euro. Dementsprechend bewirbt Joy-It das StromPi 3 Hat auch als Stromversorgungseinheit für mehrere Stromquellen mit USV-Funktion. Das StromPi 3 Hat ist trotzdem erwähnenswert, weil der Hersteller statt der im Zweifel brandgefährlichen Lithium-Ionen- und Lithium-Polymer-Akkus auf Lithium-Eisenphosphat-Akkus setzt, bei denen es zu keinem „thermischen Durchgehen" mit Selbstentzündung kommen kann.

Die Hauptprobleme beim mobilen Einsatz sind, dass der Raspi weder einen Standby-Modus kennt, in dem er nur noch wenige Milliwatt aufnimmt, noch nach dem Herunterfahren ausschaltet. Selbst nach einem `poweroff` beträgt die Leistungsaufnahme noch bis zu einem Watt. Die bewährte Methode ist, den Raspi im portablen Einsatz an eine große Powerbank anzuschließen und, nach dem Herunterfahren, das Stromkabel zu trennen. So arbeitet auch das StromPi 3 Hat: Fällt die Stromversorgung via Netzteil aus, fährt es den Raspi herunter und schaltet nach einer programmierbaren Zeit die Stromzufuhr komplett aus, um ein Tiefentladen des Akkus zu verhindern.

Beim Kauf einer neuen Powerbank sollten Sie darauf achten, dass sie gleichzeitig den Akku laden und angeschlossene Geräte versorgen kann. Das ist keineswegs selbstverständlich. Ansonsten müssten Sie den Raspi erst wieder herunterfahren und an die Powerbank anschließen, bevor Sie ihn aus dem stationären Betrieb mitnehmen können.

## Stromnetz

Falls es am Installationsort keine Steckdose gibt oder Sie den USB-C-Anschluss des Raspi 4 für den Datenaustausch verwenden wollen, ist Power over Ethernet (PoE) eine interessante Alternative zum Steckernetzteil. Das setzt allerdings voraus, dass Sie einen PoE-Switch oder einen PoE-Injektor besitzen und ein PoE-Hat oder einen PoE-Adapter beim Raspi nachrüsten. Das Original-PoE-Hat der Raspi-Foundation ist mit knapp 25 Euro vergleichsweise teuer, dafür aber sehr kompakt – es dürfte in die meisten Gehäuse für den Raspi 4 und auch für den



**Das PoE-Hat der Raspberry Pi Foundation ist so kompakt, dass es in die meisten Standard-Gehäuse passt. Mit nur 12,5 Watt Ausgangsleistung ist es für den Raspi 4 aber etwas unterdimensioniert, was beim Betrieb mit einer SSD zu Problemen führen kann.**

Raspi 3B+ passen. Außerdem hat es einen geregelten Lüfter für die CPU des Raspi, unter dem auch ein flacher Kühlkörper gut Platz findet. Allerdings ist es nur auf maximal 2,5 Ampere ausgelegt und es blockiert den kompletten GPIO-Anschluss, der nicht durchgeführt ist.

Im Test kam es zu einem äußerst seltsamen Phänomen im Zusammenhang mit diesem PoE-Hat: Als wir mittels USB-SATA-Adapter eine SSD an den Raspi anschlossen und dann mit `fio` die Schreib- und Lese-Datentransferrate ermitteln wollten, lieferte der Raspi nur bei den Adaptern mit UAS-Protokoll nach Abschluss des zweiten Benchmarks reproduzierbar Ein-/Ausgabe-Fehler – und zwar sowohl auf der SSD als auch auf der SD-Karte, von der wir Pi OS gebootet hatten. Nach einem Neustart funktionierte der Adapter bis zum nächsten Benchmark wieder klaglos. Versorgten wir den Raspi 4 über das Steckernetzteil der Raspi-Foundation mit Strom oder tau-

schten wir das PoE-Hat der Raspi-Foundation gegen das eines anderen Anbieters aus, gab es dieses Problem nicht.

Unser Tipp: Sollten Sie nach der Umrüstung auf PoE über seltsame Effekte stolpern, versuchen Sie es zunächst mit einem herkömmlichen Netzteil, bevor Sie den Raspi oder irgendwelche Peripherie für defekt erklären. Auch wenn uns der Raspi keine Unterspannung anzeigte, vermuten wir, dass das die 2,5 Ampere des PoE-Hats letztlich zu wenig waren. Eine zufriedenstellende Erklärung, warum das Problem nur beim Benchmark der UAS-Adapter auftrat, nicht aber bei den mit USB-Storage-Adaptern, und warum es auch die SD-Karte des Raspi betraf, haben wir aber nicht.

## Alternativen nutzen

Andere Anbieter verkaufen ihre wenn auch deutlich klobigeren PoE-Hats für weniger als die Hälfte dessen, was die Raspi-Foundation verlangt. In herkömmliche Gehäuse passt der Raspi 4 dann nicht mehr, dafür reichen diese Hats aber meist den GPIO-Anschluss durch, sodass man zusätzliche Erweiterungsplatinen aufstecken kann. Manche Modelle haben außerdem einen Klemmenanschluss und liefern bis zu 4 Ampere Strom, sodass man damit auch kleine Pumpen oder Motoren betreiben kann.

Falls Sie den USB-C-Anschluss des Raspi 4 nicht für Datentransfers benötigen, sind externe PoE-Wandler eine weitere Alternative. Ihr Aussehen ähnelt USB-Ethernet-Adaptern, sie haben aber neben dem USB-C-Anschluss zur Stromversorgung zusätzlich einen RJ45-Stecker zum Anschluss an die Netzwerkbuchse des Raspi. Sie sind mit unter 10 Euro auch die billigste Lösung für den Fall, dass es am Installationsort des Raspis keine Steckdose für ein traditionelles Netzteil gibt und man den Raspi über das Netzwerk mit Strom versorgen muss. (mid) c't






Bild: Andreas Martini

# Schalter, Taster und LEDs verdrahten

**An den Raspi können Sie Schalter und LEDs direkt anschließen – toll zum Experimentieren. Wir zeigen an einer aufsteckbaren Lochrasterplatine, was Elektronik-Neulinge beim Verdrahten beachten sollten.**

Von **Pina Merkert**

Auf dem Raspberry Pi ragen hinter der Netzwerkbuchse 40 Drahtstifte aus der Platine, auf die man Erweiterungsplatinen wie einen Hut aufstecken oder Jumperkabel direkt anstecken kann. 27 dieser Pins sind sogenannte General-Purpose-Input/Output-Pins (GPIO), also Ein- und Ausgabe-Pins ohne von vornherein festgelegte Aufgabe. Jeden davon kann man als Ausgang konfigurieren und mit eigenen Programmen eine Spannung von 3,3 Volt anlegen oder mit Masse verbinden (GND-Pegel). Man kann sie aber auch jeweils als Eingang benutzen, sodass das eigene Programm erkennt, ob sie mit der 3,3-Volt-Spannungsversorgung oder mit Masse verbunden sind.

## Eingänge nicht in die Luft hängen!

Eingänge sollten in einer Schaltung entweder mit Masse (GND-Pin) oder mit 3,3 Volt verbunden sein. Keinesfalls dürfen Sie sie direkt mit den 5-Volt-Pins

verbinden, Sie würden sonst den Raspi irreparabel beschädigen. Hängen die Eingänge ohne elektrische Verbindung in der Luft, liest der Raspi zufällig mal einen hohen oder niedrigen Pegel ein, je nachdem, ob sich auf dem Drahtstückchen ein paar Elektronen gesammelt haben oder nicht. Einen so undefinierten Pegel sollte man vermeiden.

Schließt man beispielsweise einen Taster an, stellt dieser im gedrückten Zustand eine Verbindung zu 3,3 Volt her. Ist er allerdings nicht gedrückt, würde eine frei hängende Verbindung entstehen. Deswegen sollte man vom Pin zur Masse eine Verbindung einbauen, über die Ladungen abfließen können. Wäre diese Verbindung aber nur ein Draht, entstünde beim Drücken des Schalters ein Kurzschluss. Man verwendet deswegen sogenannte „Pull-down-Widerstände“, die den Strom begrenzen, wenn der Schalter geschlossen ist. 10 kΩ sind dafür ein gebräuchlicher Wert. Über einen so großen Widerstand können nur ganz kleine Ströme fließen, die geringe Restladung beim Öffnen des Schalters fließt aber verlässlich ab.

## Prellen

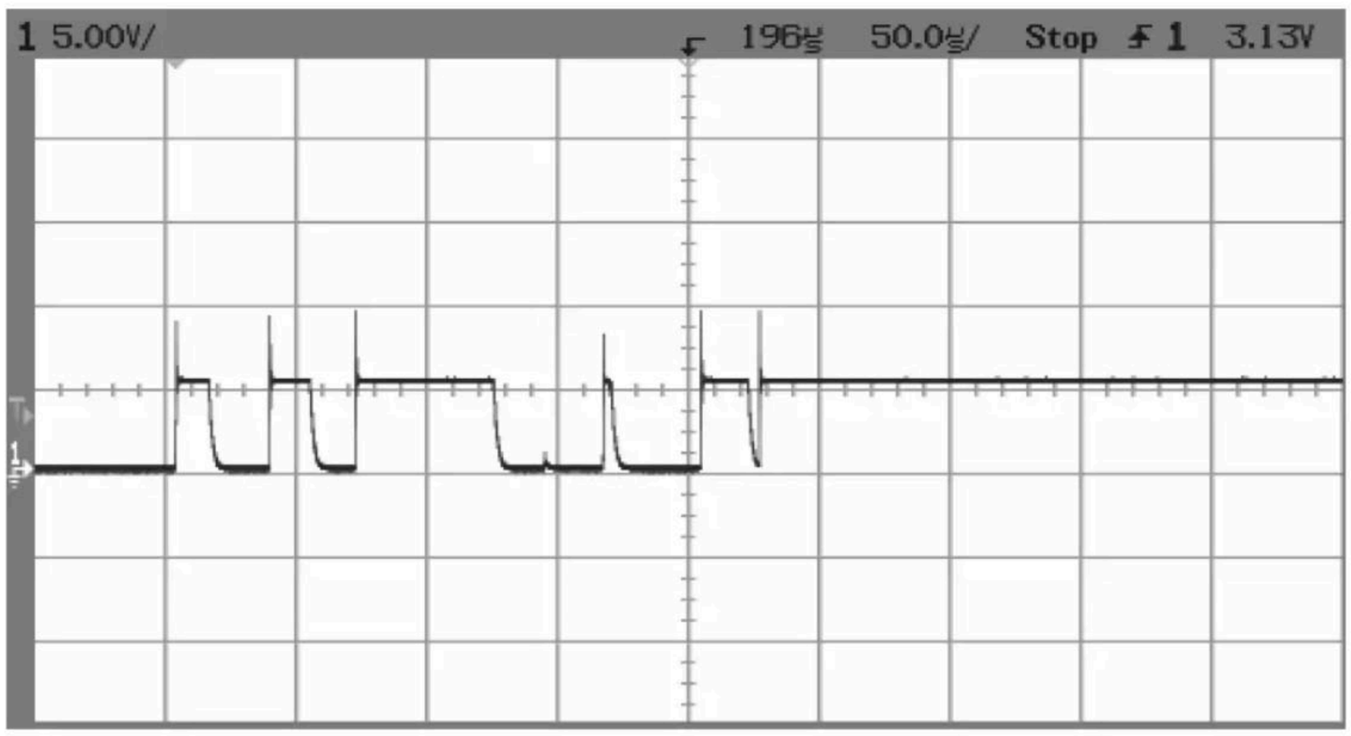
Beim Schließen des mechanischen Kontakts in einem Schalter kommt es innerhalb einer kurzen Phase von weniger als einer Zehntelsekunde zu einer noch nicht vollständig geschlossenen Verbindung. Meist schwingen federnde Bleche und stellen mehrfach kurz eine Verbindung her, die sie gleich darauf wieder verlieren. Elektroniker bezeichnen dieses Hin und Her als „Prellen“.

Da der Raspi per Interrupt auf steigende und fallende Flanken an einem GPIO reagieren kann, registriert er das Prellen als mehrfaches Schalten. Gegen diese Störung kann man sowohl hardwaremäßig als auch softwareseitig vorgehen.

Zum Entprellen per Hardware lötet man beispielsweise einen Kondensator mit 0,1 µF zwischen den Pin und GND. Der muss sich beim Einschalten erst mal aufladen, weshalb er das Potenzial beim Einschalten kurz niederhält. Beim Ausschalten entlädt er sich und hält das Potenzial für einen Moment noch hoch. Eine größere Kapazität entprellt noch sicherer, aber bei sehr großen Kondensatoren reagiert der Schalter mit einer merklichen Verzögerung. Passen Sie den Wert also ruhig an Ihren Schalter und Ihre Bedürfnisse an. Je nach Schalter verhindert das nicht jeden ungewollten Interrupt, reduziert das Hin und Her aber auf wenige Ausnahmen.

Denen kann man auch in der Software begegnen: Beispielsweise kann das Programm einen GPIO einfach nur in Zehntelsekundenabständen abfragen. Oder man fügt ein `time.sleep(0.1)` hinter der Reaktion auf ein Schaltereignis ein, um erst nach einer Zehntelsekunde wieder auf einen weiteren Interrupt zu reagieren. Im Code unter ct.de/wtq4 haben wir ein Beispiel für ein Software-Entprellen eingefügt.

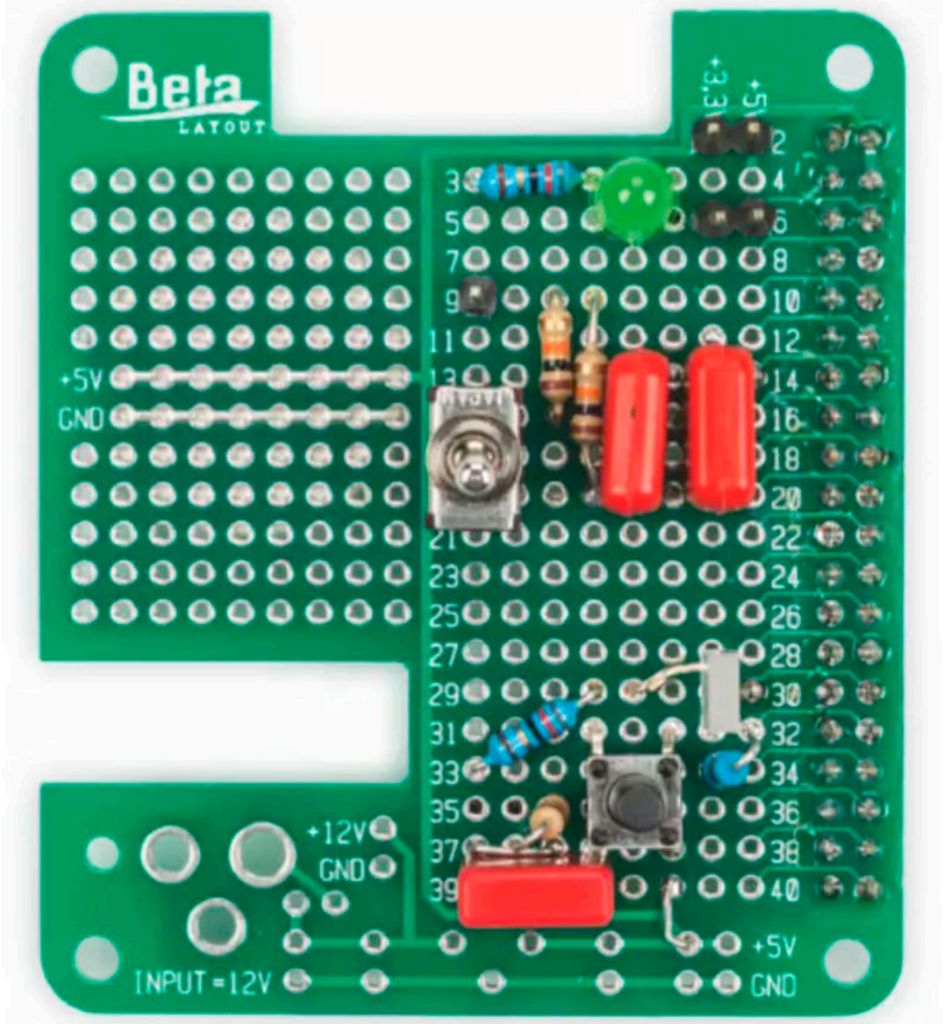
Bauteile wie LEDs haben einen recht kleinen Widerstand, weshalb auch bei 3,3 Volt vergleichsweise große Ströme fließen können. Im schlimmsten Fall geht eine LED kaputt, wenn sie direkt am GPIO hängt. Es empfiehlt sich daher, die Spannung an der LED zu reduzieren, was laut Ohm’schem Gesetz auch den Strom begrenzt. Dafür fügt man einen Vorwiderstand in den Stromkreis ein. Den perfekten Wert für den Vorwiderstand einer LED können Sie mit den Werten aus dem Datenblatt der LED berechnen. Falls Ihnen das zu mühevoll ist: 80Ω ist ein passender Wert für alle üblichen LEDs. Finden sich keine 80Ω in der Bastelkiste, tun es auch 100Ω.




**Das Speicheroszilloskop zeigt, wie oft der Pegel eines prellenden Schalters wechselt. Das Hin und Her dauert bei diesem Schalter etwa 250 µs.**

## Lochrasterplatine

Damit die Bastelei nicht zum Kabelverhau wird, lohnen sich schon ab wenigen Bauteilen Lochrasterplatinen. Sie halten die Bauteile an Ort und Stelle und sorgen für Übersicht. Um möglichst billig wegzukommen, reicht es, von einer größeren Platine für weniger als einen Euro ein Raspi-großes Stück zum Experimentieren abzusägen.




**Unser Raspi-GPIO-Testaufbau entstand auf einem Werbegeschenk eines Platinenfertigers, das dieser auf der Maker Faire in Hannover verteilte. Vergleichbare Platinen verkaufen Pimoroni und andere für circa 5 Euro.**

Wer richtig viel Platz braucht, kann auch ein altes IDE-Flachkabel als Verbindung zu einer großen Platine nutzen. Die 40-poligen Stecker passen genau auf die Pinleiste vom Raspi. Die Litzen jüngerer UDMA-Kabel führen zwischen allen Signalkabeln allerdings eine zusätzliche Masseverbindung, sodass die Litze zu fein wird, um sie mit heimischen Mitteln zu verarbeiten. Aus einem alten Kabel kann man dafür aber mindestens drei Stecker ernten.

Wer sich mehr Bequemlichkeit erkaufen will, erwirbt eine an den Raspi angepasste Platine für 5 bis 10 Euro. Diverse Hersteller bieten leicht unterschiedliche Modelle unter Bezeichnungen wie „Raspberry Pi Prototyping HAT“ an. Sie verbinden die GPIO-Pins zu Löchern, die nicht direkt nebeneinander liegen, sodass man Bauteile dazwischen löten kann. Außerdem führen sie GND und 3,3 Volt als Streifen über die Platine, was kürzere Wege beim Verkabeln erlaubt.

## Ansteuerung mit Python

Der Bequemlichkeit zuliebe nutzen die folgenden Beispiele für die Ansteuerung Python. Dort ist die Schnittstelle zu den Pins die `GPIO`-Bibliothek. Ihr teilt man zunächst mit, welches Nummerierungsschema man verwendet (`GPIO.BCM` sind die Pin-Nummern des SoC) und welche Pins als Ein- oder Ausgänge dienen:

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setup(2, GPIO.OUT,
           initial=GPIO.LOW)
GPIO.setup(22, GPIO.IN)
```

Eine Endlosschleife sorgt danach beispielsweise für eine blinkende LED:

```
import time
while True:
  GPIO.output(2, GPIO.HIGH)
  time.sleep(0.5)
  GPIO.output(2, GPIO.LOW)
  time.sleep(0.5)
```

LEDs einfach nur blinken zu lassen, sieht bei weitem nicht so schön aus wie ein weicher Wechsel der Helligkeit. Das geht erstaunlich leicht mit Pulsweitenmodulation (PWM). Der Raspi beherrscht PWM in Hardware an den Pins 12, 13, 18 und 19. An allen anderen Pins emuliert er den Effekt durch schnelles Umschalten softwareseitig. Das Beispiel moduliert die Pulsweiten mit 50 Hz:

```
GPIO.setup(12, GPIO.OUT)
led = GPIO.PWM(12, 50)  # 50Hz
while True:
  for dc in range(0, 101, 5):
    led.ChangeDutyCycle(dc)
    time.sleep(0.1)
  for dc in range(100, -1, -5):
    led.ChangeDutyCycle(dc)
    time.sleep(0.1)
```

Glättet man ein PWM-Signal mit einem Kondensator, lassen sich damit feingliedrig Spannungen zwischen 0 und 3,3 Volt einstellen. Die optimale Kapazität hängt von der Last ab, da große Ströme große Kondensatoren trotzdem schnell entladen. Generell sorgt ein größerer Kondensator für eine glattere Spannung. Falls Sie das geglättete Signal modulieren wollen, sollten Sie den Kondensator allerdings

nicht zu groß wählen, weil er dann auch die Modulation glättet. Probieren Sie für Ihr Projekt einfach verschiedene Kapazitäten aus.

## Leseoptionen

Auslesen von GPIOs geht mit `GPIO.input(pinNo)`. Falls der Raspi aber noch mit anderen Berechnungen beschäftigt ist, bekommt er mit dieser Funktion nicht sofort mit, wenn sich das Potenzial am Pin ändert. Abhilfe schafft die Funktion `GPIO.wait_for_edge(pinNo, GPIO.RISING)`, die blockiert, bis der Raspi einen Interrupt wegen einer steigenden Flanke verarbeitet. Da das Programm dabei blockiert ist, empfiehlt es sich, eine solche Abfrage in einen eigenen Thread auszulagern (das `time.sleep(0.1)` dient dem Entprellen):

```
from threading import Thread
def ck_btn_forever():
  while True:
    GPIO.wait_for_edge(26,GPIO.RISING)
    print("Button pressed")
    time.sleep(0.1)
Thread(target=ck_btn_forever).start()
```

**Beispielcode**

ct.de/wtq4

Stattdessen kann man die Überwachung der Bibliothek überlassen und bei einem Ereignis ein Callable ausführen, im hier folgenden Code-Ausschnitt eine `lambda`-Funktion:

```
GPIO.add_event_detect(
  22, GPIO.RISING,
  callback=lambda _: print(
            "<- Switch to the left."))
GPIO.add_event_detect(
  10, GPIO.RISING,
  callback=lambda _: print(
           "-> Switch to the right."))
```

## Serielle Schnittstellen

Über die PIN-Leiste des Raspi stellt der Minirechner auch zwei SPI-Schnittstellen, einen I2C-Bus und eine serielle Verbindung bereit. Komplexere Sensoren mit eigener Intelligenz bindet man über diese Schnittstellen an. An einem Bus können dabei meist Dutzende Sensoren hängen. Über I2C unterstützt der Raspi auch Chips, die noch mehr GPIO-Pins bereitstellen (siehe Seite 68).

All diese Erweiterungen belegen nur einen Teil der GPIOs, sodass bei jedem Projekt noch Pins frei bleiben, um Modi bequem per Kippschalter umzuschalten oder mit einer Status-LED anzuzeigen, was der Raspi gerade macht. (pmk) c't


Es gibt 10 Arten von Menschen. iX-Leser und die anderen.




Jetzt Mini-Abo testen:
3 Hefte + Bluetooth-Tastatur nur **16,50 €**

www.iX.de/testen

www.iX.de/testen 49 (0)541 800 09 120 leserservice@heise.de

iX MAGAZIN FÜR PROFESSIONELLE INFORMATIONSTECHNIK

# Antworten zum Raspberry Pi

**Einsteiger wie Raspi-Profis stellen sich immer wieder Fragen zum Umgang mit dem Einplatinencomputer.**

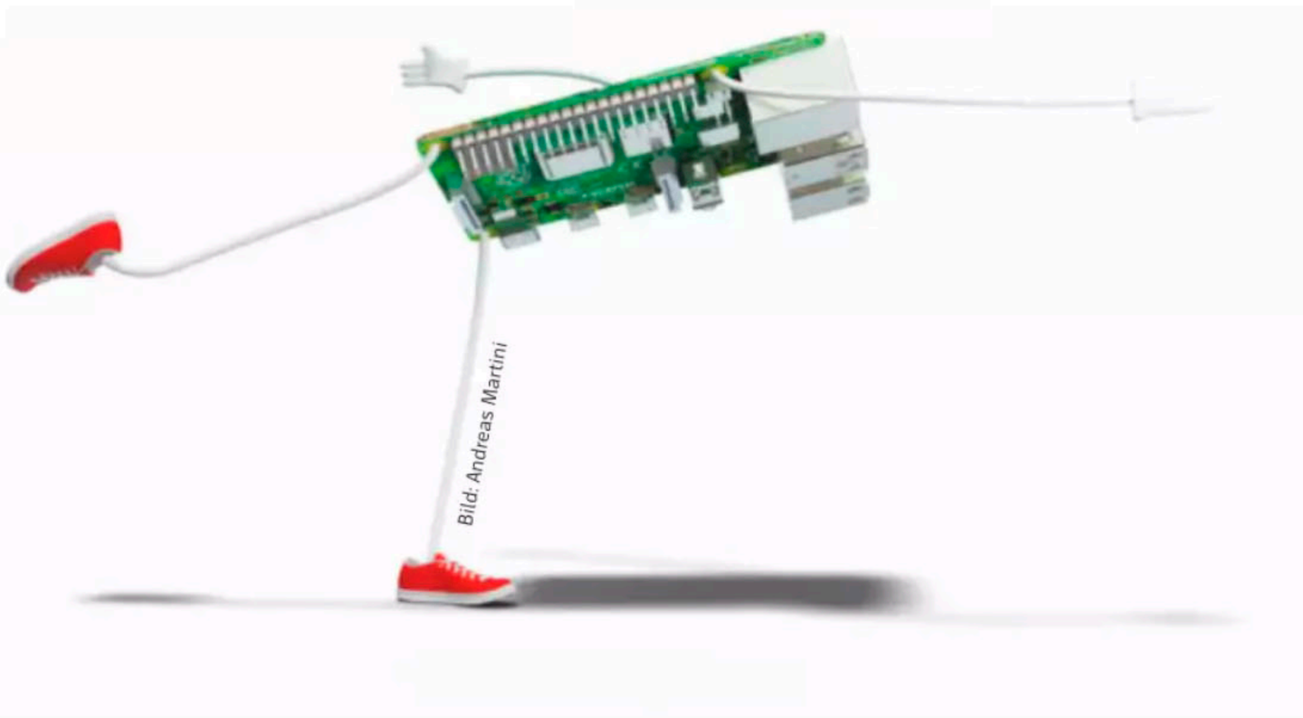
Von **Merlin Schumacher**




## Spannungsversorgung und USB-C

**? Ich habe mir einen Raspi gekauft, aber bin nicht sicher, wie viel Strom das USB-Netzteil liefern muss, damit das System stabil läuft.**

! Je nach Modell sind die Anforderungen unterschiedlich. Kurz gefasst sind sie für einen Raspi 1, Raspi Zero (W) und Raspi 2 mit einem 2-Ampere-Netzteil auf der sicheren Seite. Für den Raspi 3 müssen es 2,5 Ampere sein und für den Raspi 4 schon 3 Ampere. Dazu kommt noch der Strombedarf für Peripherie wie Festplatten. Die Raspberry Pi Foundation pflegt eine detaillierte Liste über die nötige Spannungsversorgung und den Maximalverbrauch von USB-Peripherie. Diese finden Sie über ct.de/w76v.




**? Ich habe gelesen, dass der USB-C-Anschluss des Raspberry Pi 4 nicht korrekt arbeitet, was ist das Problem?**

! Die Entwickler des Raspi 4 haben bei der Elektronik für den USB-C-Anschluss einen Widerstand sparen wollen. Das führte jedoch dazu, dass es Probleme mit sogenannten „intelligenten“ USB-C-Kabeln gibt. Diese Kabel enthalten einen kleinen Controller-Chip, der Details zur Verbindung aushandelt. Bei der Schaltung des Raspi gehen diese Chips jedoch davon aus, dass es sich um ein Audio-Gerät handelt, das keinen Strom über das Kabel bezieht.

Leider kann man „dumme“ Kabel ohne Chip nicht von solchen mit Chip unterscheiden, es hilft also nur ein Versuch. Ein Indiz ist der Preis des Kabels, denn die Kabel mit Chip sind üblicherweise ein paar Euro teurer.

Häufig wird behauptet, dass der Raspi an bestimmten Netzteilen nicht funktioniert. Das ist nicht richtig. Lediglich das Kabel ist entscheidend – außer natürlich, das Kabel ist im Netzteil integriert. Neuere Revisionen des Raspi haben das Problem nicht.

Statt ein spezielles Kabel zu benutzen, können Sie aber auch mit viel Lötgeschick den Schaltungsfehler des Raspi 4 beseitigen: Dazu müssen Sie den Kurzschluss zwischen den beiden CC-Pins an der USB-C-Buchse beseitigen, indem Sie den vierten Pin vom Platinenrand aus gesehen ablöten und etwa einen Millimeter hochbiegen. Das geht aber nur mit einer starken Lupe und einer sehr feinen Lötspitze – denn die Beinchen sind nur Bruchteile eines Millimeters voneinander entfernt. Zum Hochbiegen benutzen Sie am besten eine Stecknadel oder eine Skalpellklinge der Form 11, denn auf Stahl haftet das Lötzinn nicht. Im nächsten Schritt müssen Sie einen 5,1-kΩ-Widerstand an das Bein anlöten, damit das Signal auf Masse gezogen wird. Am besten führen Sie den Widerstand an der Buchse entlang bis zur nahegelegenen Diode und verlöten ihn mit deren Masseanschluss. Damit sind beide CC-Pins korrekt terminiert und Sie können künftig jedes USB-C-Kabel mit Ihrem Raspi verwenden [1].

**? Ich plane einen Raspi 4 zu kaufen, möchte aber ein Modell ohne die bekannten USB-C-Probleme. Wie unterscheide ich die Geräte?**

! Von außen ist das schwierig. Die Verpackung verrät die korrigierte Fassung nicht. Im Allgemeinen sollten alle aktuell verkauften Raspi 4 Versionen mit der korrekten USB-C-Schaltung sein. Falls Sie planen, einen Raspi 4 mit 8 GByte RAM zu erstehen, sind Sie auf der sicheren Seite: Alle großen Raspis enthalten die korrigierte Schaltung.

Auf der Platine selbst können Sie schauen, ob sich direkt neben dem Wort „MICRO“ der Beschriftung des Micro-SD-Kartenschachts ein kleiner dreibeiniger Transistor befindet. Nur die korrigierten Boards der Revision 1.2 haben den Transistor an dieser Position.

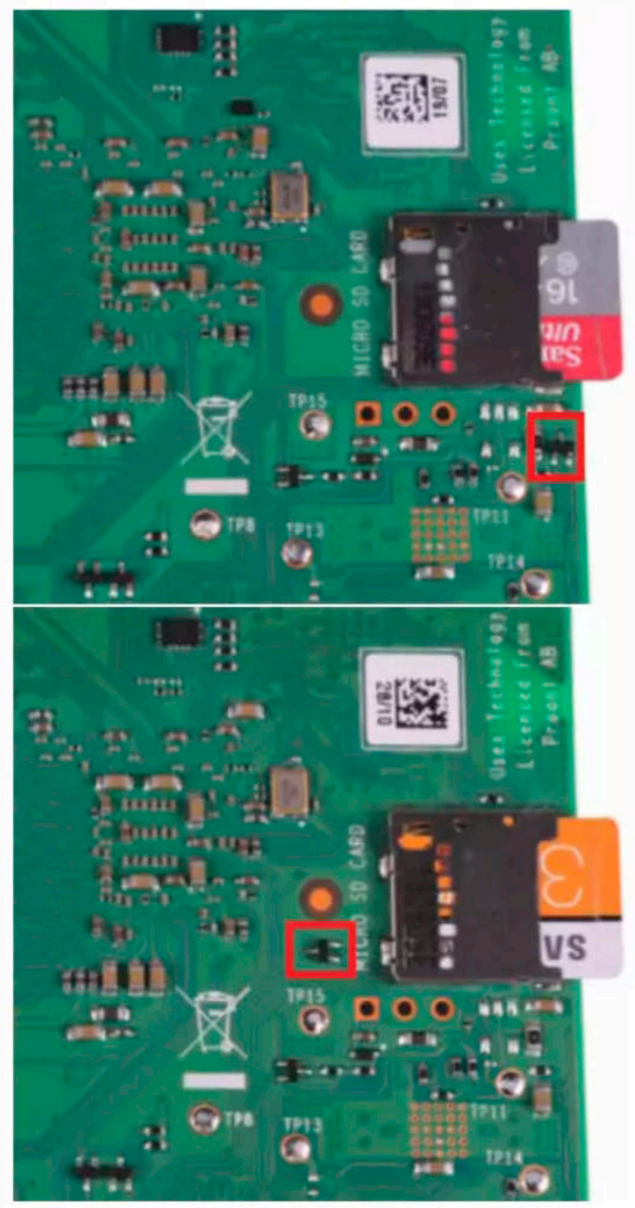
## Gehäuse, Industrie und Kühlung

**? Brauche ich zwingend ein Gehäuse für den Betrieb eines Raspi? Kann ich den nicht einfach in ein Regal legen?**

! Das können Sie tun. Der Raspi funktioniert problemlos auch ohne Gehäuse. Im Raspi sind zwar keine mechanischen Teile verbaut, aber es kann trotzdem sein, dass eines der Bauteile Schaden nimmt, wenn er herunterfällt. Bei Kontakt mit metallischen Gegenständen kann es außerdem zu Kurzschlüssen kommen, die den Raspi und vielleicht auch angeschlossene Peripherie beschädigen. Daher ist es sinnvoll, den Raspi zum Schutz in ein Gehäuse zu packen oder zumindest so abzusichern, dass ungeschickte oder unbefugte Hände ihn nicht kaputt machen.

**? Ich arbeite in einem kleinen Unternehmen und möchte den Raspberry Pi in einem Projekt für die Arbeit einsetzen. Ist das sinnvoll?**

! Das kommt sehr auf die Anwendung an. Die Raspberry Pi Foundation bietet mit dem Compute Module 4 (siehe S.30) eine spezielle Bauform für den Industrieeinsatz an. Zum Entwickeln benötigen Sie dann noch das Entwickler-Kit.




**Ob ein Raspberry Pi 4 bereits mit einer korrekten USB-C-Schaltung daherkommt, ist mit einem Blick auf die Platine in Erfahrung gebracht. Die korrigierten Modelle (unteres Bild) haben einen Transistor (hier rot markiert) links neben dem Micro-SD-Kartenschacht.**

Für kleinere Projekte können Sie auch einen Raspi in normaler Bauform einsetzen. Gehäuse für den Industrieeinsatz gibt es einige.

Beachten Sie jedoch, dass solche Selbstbauprojekte möglicherweise nicht von Ihrer Versicherung oder der Ihres Arbeitgebers abgedeckt sind. Gibt es Störungen und Probleme oder kommt jemand zu Schaden, sind Sie im Zweifel alleine haftbar. Klären Sie also vorher die Umstände. Auch die Raspberry Pi Foundation haftet nicht für Schäden. Haben Sie alles Rechtliche geregelt, können Sie aber mit dem Projekt loslegen.

**? Zum Start des Raspi 4 gab es viel Berichterstattung darüber, dass der Chip sehr heiß wird. Auch beim Raspi 3 entsteht nach meinem Eindruck viel Abwärme. Ist das normal? Sollte ich den Chip kühlen?**

! Die Wärmeprobleme des Raspi 4 haben die Entwickler durch Firmware-Anpassungen beseitigt. Eine gewisse Menge Abwärme ist aber wie bei jedem Prozessor normal. Der Raspi reduziert seinen Takt, wenn er zu heiß wird, geht also auch ohne Kühlkörper nicht kaputt. Außerdem nutzt er nicht nur die Oberfläche seiner CPU, sondern die gesamte Platine als Kühlkörper. Dadurch kommt einem der Raspi oft warm vor. Meist kühlt er aber schnell wieder ab, denn selten wird die CPU über lange Zeit voll ausgelastet. Wissen Sie bereits, dass der Raspi Schwerstarbeit leisten muss, lohnt sich ein Kühler, weil er dann seinen Takt nicht reduzieren muss. Gut funktionieren Gehäuse aus Aluminium, die die Wärme über ihre ganze Oberfläche abführen. Wir haben in [2] einige Kühler und Gehäuse getestet.

## Betriebssystem und Peripherie

**? Ich möchte Raspberry Pi OS gern auf einem anderen Einplatinencomputer nutzen. Geht das?**

! Nein. Alle Einplatinencomputer laufen mit speziell angepassten Kerneln, Firmwares und Bootloadern. Daher können Sie Raspberry Pi OS nur mit dem Raspi nutzen. Alternativ bietet sich aber Armbian oder eine andere für Einplatinencomputer ausgelegte Linux-Distro an. Armbian basiert, wie Raspberry Pi OS, auf Debian und unterstützt zahlreiche Boards. Wie gut die Unterstützung ist, schwankt aber je nach Modell. Die Hintergründe dieser Problematik sind sehr komplex, wir haben sie in [3] erläutert.

**? Sie haben mal von einem Projekt berichtet, dass Windows 10 auf dem Raspi 3 lauffähig macht. Gibt es das inzwischen auch für den Raspi 4?**

! Es gibt einige Bestrebungen, Windows 10 auf dem Raspi 4 laufen zu lassen. Am Grundproblem hat der Raspi 4 aber nichts geändert: Windows 10 läuft unerträglich langsam auf dem Einplatinencomputer und ist nicht dafür ausgelegt. Nutzen Sie Ihre Zeit für sinnvollere Dinge! Nach mehreren Stunden Arbeit und verschwendeter Wartezeit haben Sie ein unbenutzbares Windows 10 auf Ihrem Raspi installiert. **Es lohnt sich nicht!**

**? Ich habe mir einen neuen Raspberry Pi 4 gekauft, weil der alte Raspi 3 den Geist aufgegeben hat. Nach dem Einlegen der SD-Karte will der Raspi 4 jedoch nicht booten. Was mache ich falsch?**

! Vermutlich ist das Raspberry Pi OS auf Ihrer SD-Karte zu alt und es fehlt die Firmware für den Raspi 4. Falls Sie noch einen anderen Raspi haben, starten Sie diesen mit der SD-Karte und aktualisieren Sie das System auf Raspberry Pi OS Buster. Anschließend sollte der Raspi 4 davon starten können. Haben Sie keinen weiteren Raspi zur Hand, müssen Sie in den sauren Apfel beißen und das System neu installieren und die von Ihnen durchgeführten Änderungen neu einrichten.

Theoretisch könnten Sie auch versuchen, die Firmware von einer neuen SD-Karte auf die alte umzuziehen. Das birgt aber allerhand andere Probleme. Wir haben das nicht getestet und die Entwickler unterstützen den Raspi 4 erst mit Raspberry Pi OS Buster.

## Der Raspi und Video

**? Ich möchte meinen Raspi 4 an meinem TV benutzen, um Streamingdienste wie Amazon Prime oder Netflix zu schauen. Was brauche ich dafür?**

! Das Nutzen von Streamingdiensten auf dem Raspi ist nicht ganz einfach. Der beste Weg ist die Nutzung von LibreElec mit einem passenden Kodi-Add-on. Anleitungen dafür finden Sie zahlreich im Internet. Unserer Erfahrung nach ist die Nutzung aber oft hakelig. Manchmal braucht man mehrere Versuche, um ein Video zu starten, manchmal klappt es gar nicht, weil der Dienst seine Schnittstellen geändert hat, und die Suche funktioniert auch nur sehr zäh.

Darüber hinaus kann man die Inhalte über diesen Weg aufgrund von Digital Rights Management (DRM) oft nur in 720p abspielen. Das ist auf einem kleinen Monitor noch okay, aber auf dem 4K-TV unansehnlich.

Falls Streaming von Netflix & Co. die Hauptaufgabe Ihres Raspis werden sollte, lautet unsere Empfehlung daher: Sparen Sie sich das Gefummel und kaufen Sie einen Streaming-Stick. Dort bekommen Sie je nach Modell 4K-Video und HDR und eine schicke, funktionierende Oberfläche. Auf den meisten Sticks lässt sich Kodi dann auch noch zusätzlich installieren.

**? Kann ich mit meinem Raspi 4 Videos in 4K-Auflösung oder gar in HDR abspielen?**

! Jein. Sie können mit dem Raspi 4 Videos mit HEVC/H.265 in 4K-Auflösung abspielen, aber keine, die mit H.264 komprimiert wurden. Bei letzterem Codec unterstützt er nur Full-HD. Das liegt daran, dass für das Dekodieren zwei unterschiedliche Hardwarebausteine zuständig sind. Videos in HDR sind bislang nicht möglich. Das betrifft auch Videos mit Dolby Vision oder HDR10.

## Der Raspi im Netzwerk

**? Ich richte häufiger Raspis per SSH als Server ein. Dafür lege ich die leere Datei „SSH“ auf die SD-Karte und schließe ihn dann per Netzwerkkabel an. Leider bekommt er dann per DHCP eine IP-Adresse zugewiesen und ich muss sie in der Oberfläche des Routers suchen, bevor ich mich per SSH verbinden kann. Meist gebe ich ihm dann per Hand eine andere IP. Gibt es eine Möglichkeit, die Adresse schon auf der SD-Karte festzulegen?**

! Die gibt es. Öffnen Sie auf der SD-Karte die Datei cmdline.txt. Sie hat nur eine Zeile, an deren Ende Sie, mit einem Leerzeichen getrennt, die Konfiguration für das Netzwerk anhängen können. Im einfachsten Fall schreiben Sie nur: `ip=<gewünschte IP>`. Sie können auch noch mehr Angaben machen, die durch Doppelpunkte getrennt werden. Die Syntax lautet:

```
ip=<Raspi-IP>:<Netboot-Server>:
  <Gateway-IP>:<Subnetzmaske>:
  <Hostname>:<Netzwerkkarte>:<Autoconf>
```

Eine typische Konfiguration für Heimnetze sieht zum Beispiel so aus:

```
ip=192.168.178.50::192.168.178.1:
  255.255.255.0:mein-raspi:eth0:off
```

Netboot werden Sie nur in seltenen Fällen nutzen wollen, daher bleibt die Angabe leer. Stecken Sie die SD-Karte in den Raspi und starten Sie ihn. (jam) c't

### Literatur


[1] Mirko Dölle, **Raspberry Pi 4 ausnutzen**, c't Raspi 2020, S. 68

[2] Ingo T. Storm, **Kühllösungen für den Raspberry Pi 4**, c't Raspi 2020, S. 62

[3] Thorsten Leemhuis, **Linux auf ARM: Mit Haken und Ösen**, c't Raspi 2020, S. 118


**Weiterführende Links**

ct.de/w76v


WILLKOMMEN IN DEN GUTEN ALTEN ZEITEN...

Testen Sie Retro Gamer mit 30 % Rabatt!

2 Ausgaben als Heft oder digital + Geschenk nach Wahl







Jetzt bestellen:

www.emedia.de/rg-mini

(0541) 800 09 126 leserservice@emedia.de eMedia Leserservice, Postfach 24 69, 49014 Osnabrück




# Raspberry Pi 400 im Tastaturgehäuse

**Der Raspberry Pi 400 ist ein Mini-PC mit Raspi-4-Technik, der im Gehäuse einer Tastatur steckt. Der 70-Euro-Rechner ist vor allem für Schülerinnen und Schüler gedacht, aber auch für andere attraktiv.**

Von **Christof Windeck**

Nach der überraschenden Vorstellung des Raspberry Pi 4 im Sommer 2019 hat die Raspberry Pi Foundation nun ein weiteres Kaninchen aus dem Hut gezogen: den Schülercomputer Raspberry Pi 400. Sein kompaktes Tastaturgehäuse birgt die leicht abgewandelte Technik des Raspi 4. Wer sofort loslegen will, kauft den Raspi 400 als 100 Euro teures Kit inklusive Netzteil, Maus, Micro-HDMI-Anschlusskabel, gedrucktem Handbuch sowie einer Micro-SD-Karte, auf der Raspberry Pi OS vorinstalliert ist. Im Handel sind Versionen mit unterschiedlichen Tastaturlayouts für mehrere Länder erhältlich, auch für Deutschland. Wir haben die Kit-Version mit der vorinstallierten 32-Bit-Version von Raspberry Pi OS getestet und das 64-bittige Ubuntu 20.10 for Raspberry Pi ausprobiert.

## Mit Einschaltknopf

Die Kit-Version lässt sich kinderleicht in Betrieb nehmen, die MicroSD-Karte mit dem Betriebssystem steckt schon drin. Beim ersten Start fragt das Setup nach den gewünschten Einstellungen für Sprache, Zeitzone, Nutzername und Passwort. Dann laufen

Updates durch und nach ein bis zwei Neustarts ist der Raspi 400 betriebsbereit. Auf der nicht sonderlich schnellen MicroSD-Karte aus der Edge-Serie von SanDisk für Embedded Systems bleiben nach den ersten Updates rund 10 GByte frei.

Wer den Raspi 4 kennt, findet sich sofort zurecht – und freut sich über eine Neuerung: Die Funktionstaste F10 dient in Kombination mit der Fn-Taste als Ausschalter. Im Tastaturcontroller hat die Raspi-Foundation eine Einschaltlogik untergebracht, die anderen Raspis fehlt. Nach dem „Herunterfahren“ nimmt der Raspi 400 mit dem Originalnetzteil lediglich 0,14 Watt Leistung auf und lässt sich per F10-Taste wieder hochfahren. Im Betrieb ohne Rechenlast braucht der Raspi 400 nur etwa 2,6 Watt, unter Volllast auch mal etwas über 5 Watt – deutlich weniger als ein normaler Mini-PC.

Die Grundfläche des Tastaturgehäuses mit 25,5 und 12,5 Zentimetern Kantenlänge belegt weniger als 60 Prozent der Fläche eines DIN-A4-Blatts; die Höhe beträgt 2 Zentimeter. Die Tasten sind ähnlich wie bei vielen kompakten Notebooks mit 13-Zoll-Displays angeordnet und haben einen recht kurzen



**Alle Anschlüsse des Raspberry Pi 400 sitzen an seiner Rückseite.**

Hub. Es lässt sich ganz gut darauf tippen, aber die Qualität ist nicht mit der von teuren Notebooks vergleichbar. Auch die Kit-Maus ist eher schlicht, reicht aber für den Zweck.

## Magere Rechenleistung

Im Raspi 400 sitzt die Version C0 des „Hauptprozessors“ Broadcom BCM2711 [1], der auf bisherigen

| Raspberry Pi 400 | |
|---|---|
| **Mini-PC mit ARM-SoC im Tastaturgehäuse** | |
| Anbieter | Raspberry Pi Foundation (raspberrypi.org) |
| Prozessor | Broadcom BCM2711 C0 (4 × ARM Cortex-A72, max. 1,8 GHz) |
| GPU | Broadcom VideoCore IV in BCM2711 |
| RAM | 4 GByte LPDDR4 (verlötet) |
| Massenspeicher | MicroSD-Kartenleser mit SanDisk Edge 16GB (A1-Spezifikation, 10 GByte frei, SDSQAD-016G) |
| WLAN-Adapter | Broadcom/Cypress CYW43455, SDIO, Wi-Fi 5 (2,4/5 GHz), 1 × 1, Bluetooth 5.0 |
| Anschlüsse (alle hinten) | 2 × Micro-HDMI, 2 × USB-A 3.0, 1 × USB-A 2.0, 1 × Gigabit Ethernet (RJ45), Stromversorgung (USB-C), MicroSD-Reader, 40-polige GPIO-Leiste, Kensington Lock |
| Abmessungen | 25,5 cm × 2 cm × 12,5 cm |
| Gewicht | 395 Gramm |
| Zubehör Kit-Version | USB-C-Netzteil (Kabel 1,5 Meter), USB-Maus, MicroSD-Karte, Micro-HDMI/HDMI-Kabel (1 Meter) |
| Betriebssystem | Raspberry Pi OS (32-Bit, Buster) |
| **Messwerte** | |
| Leistungsaufnahme Soft-off | 0,14 Watt |
| Leistungsaufnahme Leerlauf / CPU-Last | 2,6 / 5,5 Watt |
| WLAN nah / 20 m | 90 / 31 MBit/s (2,4 & 5 GHz ähnlich) |
| Bootdauer | 25 s (Ubuntu 20.10: 54 s) |
| MicroSD-Karte (schreiben/lesen) | 22 / 42 MByte/s, 536 / 2150 IOPS |
| 7-Zip Bench (RPi 4)[1] | 3763 / 9001 MIPS (3502 / 7648 MIPS) |
| JetStream 2 (RPi 4)[1] | 28,3 Punkte (24,7 Punkte) |
| Preis | 70 €, Kit 100 € |
| [1] Werte für Raspberry Pi 4 in Klammern | |



**Unter der Tastatur liegt ein Aluminiumblech, welches auch den BCM2711-Chip kühlt – ohne Lüfter.**

Raspi 4 als Version B0 zum Einsatz kommt. Beim Funktionsumfang gibt es keine Unterschiede, wichtigste Bestandteile sind vier CPU-Kerne vom Typ ARM Cortex-A72, die man auch in billigen und älteren Smartphones findet. Für Grafik ist ein Broadcom VideoCore IV zuständig, für den es ordentliche Linux-Treiber gibt. Im Raspi 400 takten die vier ARM-Kerne ab Werk mit bis zu 1,8 GHz, das sind 20 Prozent mehr als bei einem Raspi 4 (1,5 GHz), wenn man ihn nicht übertaktet. Zudem sitzt im Raspi 400 unter der Tastatur ein Aluminiumblech, welches den BCM2711 dermaßen gut kühlt, dass er seinen Takt nie drosselt – ganz anders als beim Raspi 4 ohne Zusatzkühler. Dadurch lässt sich die im Vergleich zu aktuellen x86-Prozessoren sehr magere Rechenkraft des Raspi 400 immerhin voll nutzen. Ein Intel Celeron N4100 oder N4120, den man in 300-Euro-Notebooks findet, rechnet je nach Anwendung bis zu 95 Prozent schneller.

Der höhere Takt des BCM2711 im Raspi 400 führt in Benchmarks zu 5 bis 18 Prozent besseren Ergebnissen als beim Raspi 4, was man bei der praktischen Nutzung jedoch nicht spürt. Die Rechenleistung genügt mit etwas Geduld zum Webbrowsen, für Büroarbeiten (LibreOffice ist vorinstalliert) und simple Spielchen. Im Browser laufen YouTube-Videos flüssig bis zur Auflösung 720p (1280 × 720 Pixel), darüber ruckelt's. Sofern die Player-Software die Hardware-Decoder im BCM2711 nutzt, laufen auch 4K-Filme im Format H.265 (HEVC). Bei H.264 (MPEG-4) ist maximal Full-HD möglich.

Unter Raspberry Pi OS funktionierten Videokonferenzen etwa mit Big Blue Button, Jitsi Meet, Microsoft Teams und Zoom, wenn man zusätzlich eine USB-Webcam nach UVC-Standard anschließt (USB Video Class). Der Start von Teams im Browser dauert wegen der mageren Rechenleistung aber rund 35 Sekunden, man sollte also rechtzeitig vor der Sitzung loslegen. Ubuntu brauchte zum Booten mehr als doppelt so lange wie Raspberry Pi OS (54 statt 25 Sekunden). Unter Ubuntu gibt es manchmal Bluetooth-Probleme, aber dagegen gibt es einen Patch.

## Wenige Anschlüsse

Alle Anschlüsse des Raspi 400 sitzen hinten, für Peripheriegeräte sind vor allem die drei USB-A-Buchsen wichtig. Zwei davon beherrschen USB 3.0 (alias USB 3.2 Gen 1) mit 5 GBit/s, eine nur USB 2.0 – dort stöpselt man meistens wohl die Maus ein. An die beiden Micro-HDMI-Buchsen des Raspi 400 lassen sich wie beim Raspi 4 zwei Displays gleichzeitig mit jeweils 4K-Auflösung anschließen. Leider fehlt eine Audio-Klinkenbuchse – Kopfhörer oder Lautsprecher mit analogem Eingang kann man aber am HDMI-Display anschließen, sofern es den passenden Ausgang hat. Alternativ lässt sich ein Bluetooth-Headset koppeln oder man verwendet ein USB-Headset oder einen USB-Audio-Adapter.

Die Raspi-typische GPIO-Anschlussleiste mit 40 Kontakten hat zwar auch der Raspi 400, manche Raspi-Erweiterungen (HATs) passen dort aber mechanisch nicht. Als Verlängerung der GPIO-Leiste kann man ein altes IDE-Kabel nehmen, bei dem Pin 20 nicht mit einem Stopfen blockiert ist. Im Vergleich zum Raspi 4 fehlen auch die Anschlüsse für Kamera (CSI) und DSI-Display sowie die Pins für Speisung per Power-over-Ethernet (PoE).

## Fazit

Für 70 Euro – oder als 100-Euro-Kit – bekommt man mit dem Raspi 400 einen Linux-PC mit knapper Rechenleistung für einfache Aufgaben. Der niedrige Preis zwang die Raspberry Pi Foundation jedoch zu vielen Abstrichen, am meisten vermisst man eine analoge Headset-Klinkenbuchse. Wer vor allem basteln und experimentieren möchte, nimmt besser einen Raspberry Pi 4. Der Raspberry Pi 400 hingegen ist ein niedlicher, bezahlbarer, geräuschloser und sparsamer Mini-Komplettcomputer mit freier, gut gepflegter Software, der nicht nur jungen Menschen Spaß macht. (ciw) c't

### Literatur

[1] Dr. Maik Merten, **Der Raspi-Prozessor im Detail**, c't Raspi 2020, S. 40

**Video zum Raspi 400**

ct.de/wstv

heise Security TOUR

# WISSEN SCHÜTZT

Praxiswissen für die Security-Achterbahn 2021

**Online-Konferenz**
**10. Juni / 9. September 2021**

## Auszug aus dem Programm:

- Die IT-Security-Lage – aktuelle Gefahren, neue Trends, darauf sollten Sie sich vorbereiten
  *Jürgen Schmidt*
- Vorbereitung ist alles - der richtige Umgang mit Datenschutzvorfällen
  *Joerg Heidrich, Christoph Wegener*
- Cyber! Versicherung! Will ich das wirklich haben?
  *Linus Neumann*

**Podiumsdiskussion:**
Der Emotet-Takedown – was darf der Staat?




**JETZT ANMELDEN**
Finden Sie alle weiteren Informationen unter:
www.heisec.de/tour

Organisiert von:
heise Events




# Mini-Raspi Compute Module 4 (CM4)

**Die miniaturisierten Raspberry-Pi-Module der Raspberry Pi Foundation sind „Embedded Systems“, die andere Hersteller in ihre Geräte einbauen können. CM4-Funktionen wie der PCIe-Steckplatz sind auch für (NAS-)Bastler spannend, haben in der Praxis aber so ihre Tücken.**

Von **Christof Windeck**

Den Raspberry Pi gibt es nicht nur in vier Generationen sowie mit unterschiedlich viel Arbeitsspeicher, sondern auch in Miniaturausführungen: den „Compute Modules“ (CM). Sie sind für professionelle Hardware-Entwickler und Gerätehersteller gedacht, die sie als kompakte Rechenmodule in ihre eigenen Produkte einbauen, beispielsweise in Steuergeräte für Schaltschränke. Doch die Module sind nicht teuer und locken Bastler mit einigen zusätzlichen Funktionen, die normalen Raspis fehlen: Etwa die Möglichkeit, eine SATA-Adapterkarte für ein NAS anzubinden. Das und noch einiges mehr haben wir im c’t-Labor ausprobiert.

Das aktuelle Compute Module CM4 hat im Wesentlichen die Technik des Mitte 2019 eingeführten Raspberry Pi 4, also insbesondere das System-on-Chip Broadcom BCM2711 mit vier ARM-Cortex-A72-Kernen sowie 1, 2, 4 oder 8 GByte RAM. Anders als ein normaler Raspi hat ein CM aber weder Anschlussbuchsen noch Pfostenstecker, sondern man kann

seine Schnittstellen nur indirekt verwenden, nämlich über eine passend ausgelegte Platine. Auf dieses Basis- oder Breakout-Board steckt man das jeweilige CM. Die Raspberry Pi Foundation sowie einige andere Firmen verkaufen universelle Basisplatinen, die die CM-Anschlüsse herausführen. Für das CM4 gibt es ein rund 38 Euro teures „Compute Module 4 IO Board“, das wir für diesen Testbericht gekauft haben. Am Beispiel von zwei verschiedenen CM4-Typen zeigen wir, wie man es in Betrieb nimmt.

Im Vergleich zu den Vorgängern CM (2014), CM3 (2017) und CM3+ (2019) hat das CM4 eine neue Bauform: Es ist deutlich kompakter als die älteren Versionen im SODIMM-Format von Notebook-Speichermodulen und hat statt eines seitlichen Kontaktkamms zwei winzige Steckwannen auf der Rückseite. Sie sind nur je 2,5 Millimeter schlank und 2,1 Zentimeter lang, stellen aber zusammen 200 Anschlüsse zum IO-Board bereit.

## Funktionsunterschiede

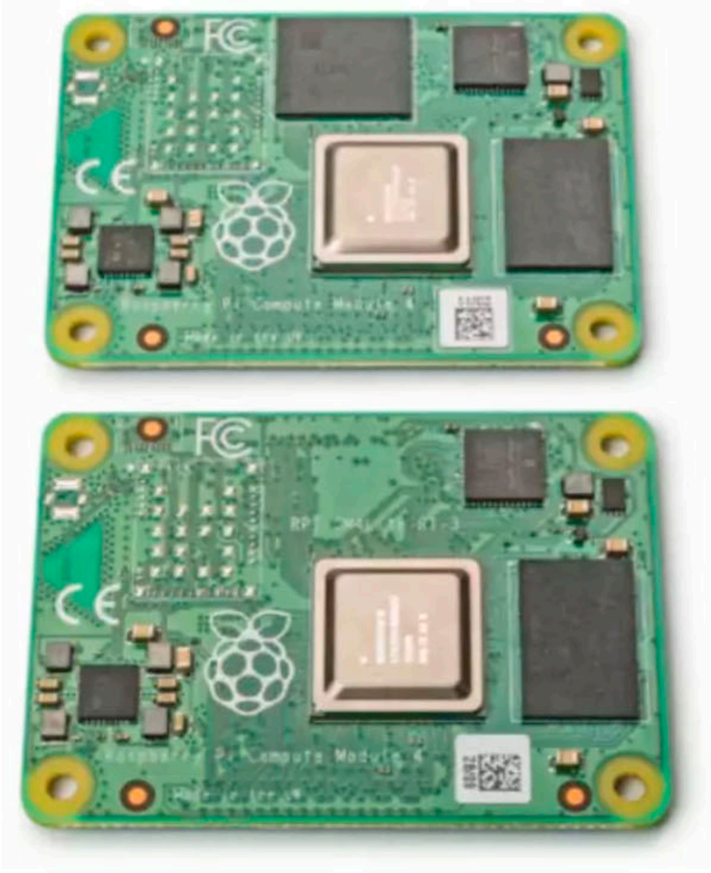
Ein CM4 stellt im Vergleich zu einem Raspi 4 einige zusätzliche Schnittstellen bereit, insbesondere eine PCI-Express-Lane (PCIe 2.0 x1) sowie je zwei statt nur je einem CSI- und DSI-Anschluss für Kameras (CSI) und Displays (DSI). Doch dem CM4 fehlen auch Funktionen, vor allem der USB-3.0-Controllerchip, der beim Raspi 4 die erwähnte PCIe-Lane belegt. Auf CM4-Versionen mit aufgelötetem WLAN-Adapter sitzt eine leicht abgewandelte Ausführung des vom Raspi 4 bekannten Broadcom/Cypress CYW43455 mit Wi-Fi 5 (802.11ac, 2,4/5 GHz) und Bluetooth 5.0/BLE.

Das IO-Board wiederum bietet einige zusätzliche Chips und Anschlüsse im Vergleich zum Raspi, darunter eine PCIe-x1-Steckfassung, einen USB-2.0-Hub für insgesamt bis zu vier USB-Buchsen (zwei davon per Pfostenstecker nutzbar) und eine Echtzeituhr (RTC, NXP PCF85063A) samt Halterung für eine CR2032-Pufferbatterie.

Wer eine PCIe-Karte einsetzen möchte, muss das IO-Board mit 12 Volt speisen; ansonsten genügen 5 Volt, allerdings gibt es dafür keine USB-C-Buchse, sondern Pfostenstecker. Für die Einspeisung von 12 Volt ist eine runde Hohlbuchse (5,5/2,5 mm) vorhanden, Onboard-Wandler erzeugen daraus 5 und 3,3 Volt. An einem „Floppy-Stromstecker“ auf dem IO-Board liegen 12 und 5 Volt an; die 5-Volt-Schiene ist mit bis zu 3 Ampere belastbar und die 12-Volt-Schiene je nach Netzteil. Im Test reichte es für zwei SSDs samt Adapterkarte. Alternativ lässt sich das IO-Board über den vierpoligen Floppystecker mit 12 und 5 Volt speisen; dann soll man laut Datenblatt jedoch die Spule L5 auslöten, damit der 5-Volt-Wandler nicht dazwischenfunkt.

Die Micro-USB-Buchse dient nicht zur Stromversorgung, sondern ist vor allem zur Installation des Betriebssystems bei CM4-Versionen mit eMMC-Flash wichtig. Denn bei diesen kann man ja nicht – wie sonst bei Raspis üblich – eine MicroSD-Karte im Kartenleser am PC beschreiben und dann umstecken. Für die CM4-„Lite“-Typen ohne eMMC-Flash gibt es einen MicroSD-Kartenleser, der bei den CM4-Typen mit eMMC wiederum funktionslos ist, also nicht etwa zusätzlich nutzbar.

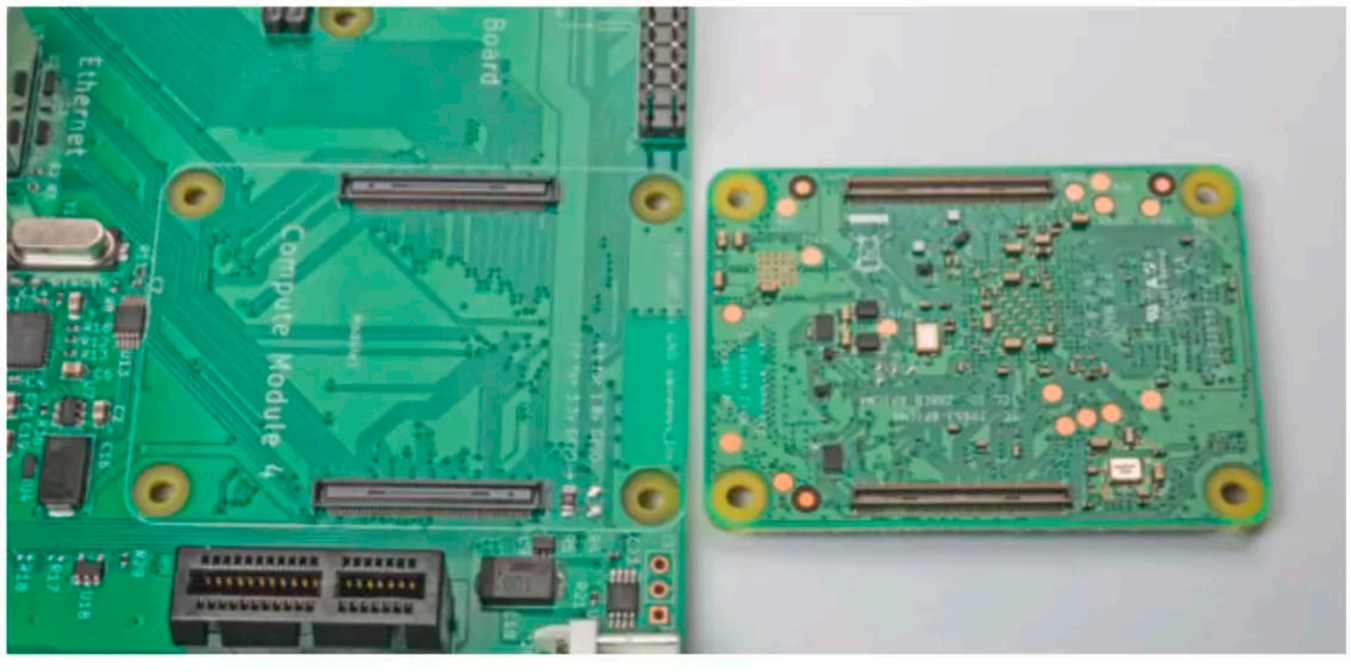
Die Raspi-typische GPIO-Kontaktleiste mit 40 Pins ist ebenso vorhanden wie die vier Kontakte für ein PoE-HAT zur Speisung per Power-over-Ethernet. Das IO-Board hat zwei normal große HDMI-Buchsen anstelle der beiden MicroHDMI-Buchsen des Raspi 4. Schließlich gibt es noch einige Lötaugen sowie Pfostenstecker für nicht mitgelieferte Jumper (2,54-Millimeter-Raster), um bestimmte Konfigurationen einzustellen, sowie einen 12-Volt-Lüfteranschluss. Eine Audio-Klinkenbuchse fehlt dem IO-Board hingegen.



**Das Raspberry Pi Compute Module 4 (CM4) gibt es in 32 verschiedenen Ausführungen. Hier zwei Varianten jeweils ohne WLAN-Adapter. Das obere CM4 im Bild hat einen eMMC-Flash-Chip (Mitte oben), das untere „CM4 Lite“ nicht.**

## Inbetriebnahme

Die Montage des CM4 auf dem IO-Board ist einfach: Man steckt es auf die Kontaktleisten; die richtige Orientierung erkennt man an den je vier Bohrungen im IO-Board und im CM4, die übereinanderliegen müssen. Die beiden 100-poligen Steckverbinder




**Zwei winzige längliche Steckverbinder mit je 100 Kontakten koppeln ein CM4 an das IO-Board.**

rasten durch sanften, aber bestimmten Druck ein. Um das CM4 wieder zu entfernen, schiebt man am Rand des IO-Boards ein flaches Hebelwerkzeug vorsichtig zwischen Platine und CM4, um letzteres hochzudrücken.

Etwas schwieriger als die Montage ist die Inbetriebnahme eines CM4 auf dem IO-Board. Zwar gibt es PDF-Datenblätter und Online-Anleitungen der Raspi-Foundation; sie zielen aber eher auf Profis als auf unerfahrene Bastler. Außerdem braucht man für manches CM4 noch mehr als Tastatur, Maus, Monitor und (12-Volt-)Netzteil. Für ein CM4 Lite sind noch eine MicroSD-Speicherkarte samt Kartenleser nötig, während man für ein CM4 mit eMMC-Flash auch ein Micro-USB-Kabel braucht sowie einen Jumper.

Setzt man ein CM4 Lite ohne eigenen Flash-Speicher ein, schreibt man wie auch bei normal großen Raspis zuerst das Raspberry Pi OS (oder ein anderes kompatibles Betriebssystem wie Ubuntu for Raspi) auf eine MicroSD-Karte. Das klappt an einem PC mit Windows, macOS oder Linux, alle nötigen Tools finden sich via raspberrypi.org (siehe ct.de/wjjj). Das CM4 Lite bootet anschließend von der MicroSD-Karte, die man zuvor in den Kartenleser des IO-Board steckt.

Bei einem CM4 mit eMMC-Flash ist die Installation komplizierter, weil man das Betriebssystem-Image per Micro-USB-Kabel in den Flash-Speicher des CM4 schreiben muss. Man spart sich zwar die MicroSD-Karte, braucht aber einen Jumper für die Pins 1 und 2 des Pfostensteckers J2 auf dem IO-Board. Denn nur mit diesem Jumper meldet der CM4 seinen eigenen Flash-Speicher als USB-Massenspeicher am Host-PC an. Läuft auf diesem Windows, muss man zudem einen speziellen Treiber installieren sowie die im Treiberpaket enthaltene Anwendung RPiboot.exe starten, damit der eMMC-Speicher beschreibbar wird. Mit dem Win32DiskImager (oder unter Linux mit dd) schreibt man schließlich das entpackte Betriebssystem-Image in den Flash-Speicher des CM4, was wegen der mageren Datentransferrate von nur rund 5 MByte/s über 20 Minuten dauert. Anschließend trennt man das IO-Board sowohl vom Netzteil als auch vom Host-PC und entfernt den J2-Jumper wieder. Schließt man das Netzteil wieder an, startet das CM4 wie jeder andere Raspi auch – die Anleitung und die Treiber finden Sie unter ct.de/wjjj.

## USB-Falle

Beim nächsten Schritt tappt man nun in eine Falle: Um Strom zu sparen, ist bei der Standard-Firmware für das CM4 der interne USB-2.0-Controller im BCM2711 abgeschaltet. Er lässt sich erst durch eine kleine Änderung der Datei `/boot/config.txt` einschalten, an die man ohne (USB-)Tastatur oder SSH-Verbindung aber nicht herankommt. Beim CM4 Lite kann man die MicroSD-Karte in einen anderen Raspi stecken, dort davon booten und die Änderung vornehmen. Da die Partition `/boot` aber FAT-formatiert ist, kann man die Datei `config.txt` auf der MicroSD-Karte via Kartenleser aber auch an einem Windows-PC oder Mac bequem bearbeiten. In `/boot/config.txt` fügt man mit einem Editor wie folgende Zeile hinzu: `dtoverlay=dwc2,dr_mode=host`. Außerdem lässt sich SSH-Zugriff aktivieren: nämlich einfach durch das Anlegen einer leeren Datei namens `ssh` unter `/boot/`.

Beim CM4 mit eMMC kann man keine MicroSD-Karte entnehmen. Hier ändert man `config.txt` daher besser gleich, wenn das CM4 noch per USB-Kabel

mit dem PC verbunden ist. Alternativ kann man später einen PCIe-USB-Hostadapter einstecken, an den man Maus und Tastatur für die Änderung anschließt; ob man einen USB-2.0- oder USB-3.0-Adapter nimmt, ist dabei egal – es muss aber einer mit PCIe x1 sein. Nach dem Abspeichern und einem Neustart ist der interne USB-2.0-Controller im BCM2711 aktiv und man kann Tastatur und Maus an die beiden USB-A-Buchsen des IO-Board umstöpseln sowie die PCIe-USB-Karte wieder entfernen – vorsichtshalber nach Abziehen des Netzteils!

Dank des USB-Hub-Chips auf dem IO-Board lassen sich über einen zehnpoligen Pfostenstecker noch zwei weitere USB-2.0-Buchsen anschließen. Dazu eignet sich im Prinzip ein PC-Slotblech mit einem solchen Buchsenpaar, allerdings ist auf dem IO-Board der zehnte Pin vorhanden und nicht wie bei PC-Mainboards üblich zur Kodierung der Steckerorientierung entfernt. Man muss also das verschlossene Loch an den Steckern der USB-2.0-Slotbleche öffnen oder den zehnten Pin auf dem IO-Board abknipsen.

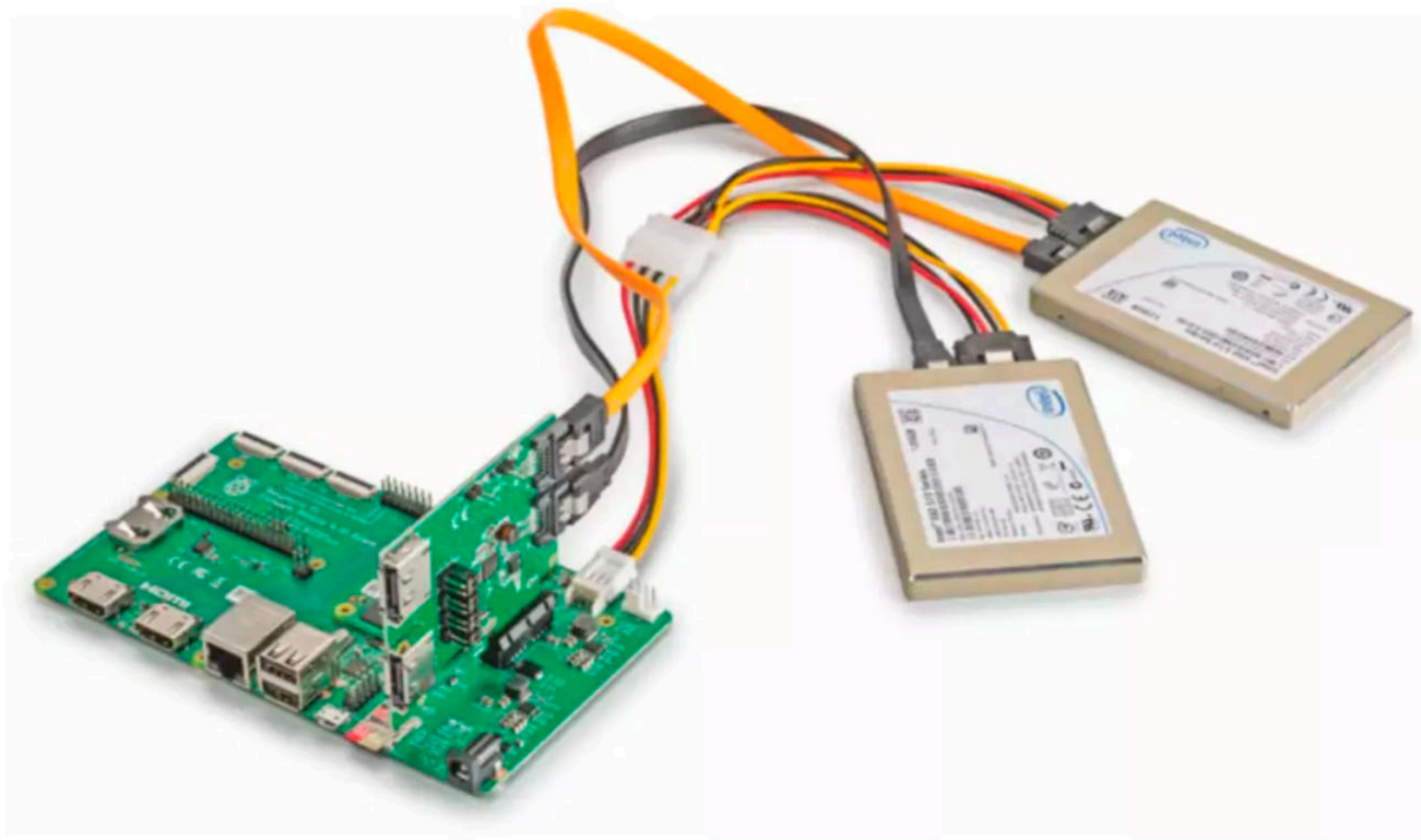
## Performance

Für unsere Experimente haben wir ein CM4 mit 8 GByte RAM und 32 GByte eMMC-Flash gekauft (Typ CM4008032) sowie ein CM4 Lite mit 2 GByte RAM, aber ohne Flash (CM4002000). Beide lieferten praktisch dieselbe Performance wie ein normaler Raspberry Pi 4, Benchmarks mit 7-zip und openssl waren rund 2 Prozent langsamer. Beim Ethernet-Durchsatz konnten wir ebenfalls keine Unterschiede zum Raspi 4 feststellen und auch die Leistungsaufnahme lag auf ähnlichem Niveau, nämlich im Leerlauf bei rund 2,4 Watt ohne Ethernet-Verbindung und bei etwa 2,5 Watt mit. Als 12-Volt-Netzteil haben wir das Leicke NT03122 mit 2 A Belastbarkeit für 13 Euro bei Amazon bestellt. Es arbeitete effizient, aber bei der Berührung der Masseleitung war ein Berührstrom zu spüren. Während sich der eMMC-Speicher per Micro-USB-Kabel nur gemächlich befüllen lässt, ist er intern deutlich schneller an den BCM2711 angebunden, er liefert mehr als 80 MByte/s und 3500 IOPS.

### Raspberry Pi Compute Module 4 (CM4) & IO-Board

| | |
|---|---|
| **Compute Module CM4 (CM4002000, CM4008032)** | |
| Hersteller | Raspberry Pi Foundation |
| Prozessor | Broadcom BCM2711 |
| RAM | 1, 2, 4 oder 8 GByte LPDDR4 |
| Flash-Speicher | ohne oder mit 8, 16 oder 32 GByte eMMC |
| WLAN-Adapter | ohne oder mit Broadcom/Cypress CYW43455 |
| Anschlüsse CM4 | 2 × 100-Pin-Wannenstecker |
| Abmessungen | 5,5 cm × 4 cm, ca. 4 mm hoch |
| **IO-Board für CM4** | |
| Anschlussbuchsen | 2 × HDMI, 1 × RJ45, 2 × USB-A 2.0, 1 × MicroUSB, 1 × MicroSD-Reader |
| interne Anschlüsse | 40-Pin GPIO, 1 × PCIe 2.0 x1, 1 × Batterie (CR2032), 2 × CSI, 2 × DSI |
| Stromanschlüsse | 1 × Hohlbuchse 12 V, 1 × 4-Pin 5 V/12V, 1 × PoE-HAT |
| Abmessungen | 16 cm × 9,3 cm, ca. 19 mm hoch |
| **Messungen mit Leicke-Netzteil NT03122 (12V/2A)** | |
| Leistungsaufnahme Soft-off | 0,2 W |
| Leerlauf ohne LAN | 2,4 W |
| mit LAN / Volllast | 2,5 / 4,8 W |
| Leerlauf / Schreiben[1] mit ASM1061[2] oder Marvell 88SE9120[3] | 4,4 / 11 W |
| Leerlauf / Schreiben[4] mit Marvell 88SE9215 | 8,1 / 15 W |
| Lesen / Schreiben auf RAID 1 | 89 / 105 MByte/s |
| openssl aes-128-cbc | 85 MBit/s |
| 7-Zip (7z b) | 3356 / 7351 MIPS |
| Preis | 25 – 100 € für CM4, 38 € für IO-Board |

[1] mit Adapterkate und 2 SATA-SSDs [2] InLine 76696B [3] Syba SI-PEX40061/IOCrest SY-PEX40061 [4] DeLock 90382



**Raspi-NAS mit Haken: Auf das IO-Board passt eine PCIe-Karte (mit abgeschraubtem Slotblech), hier ein SATA-Adapter für zwei SSDs.**

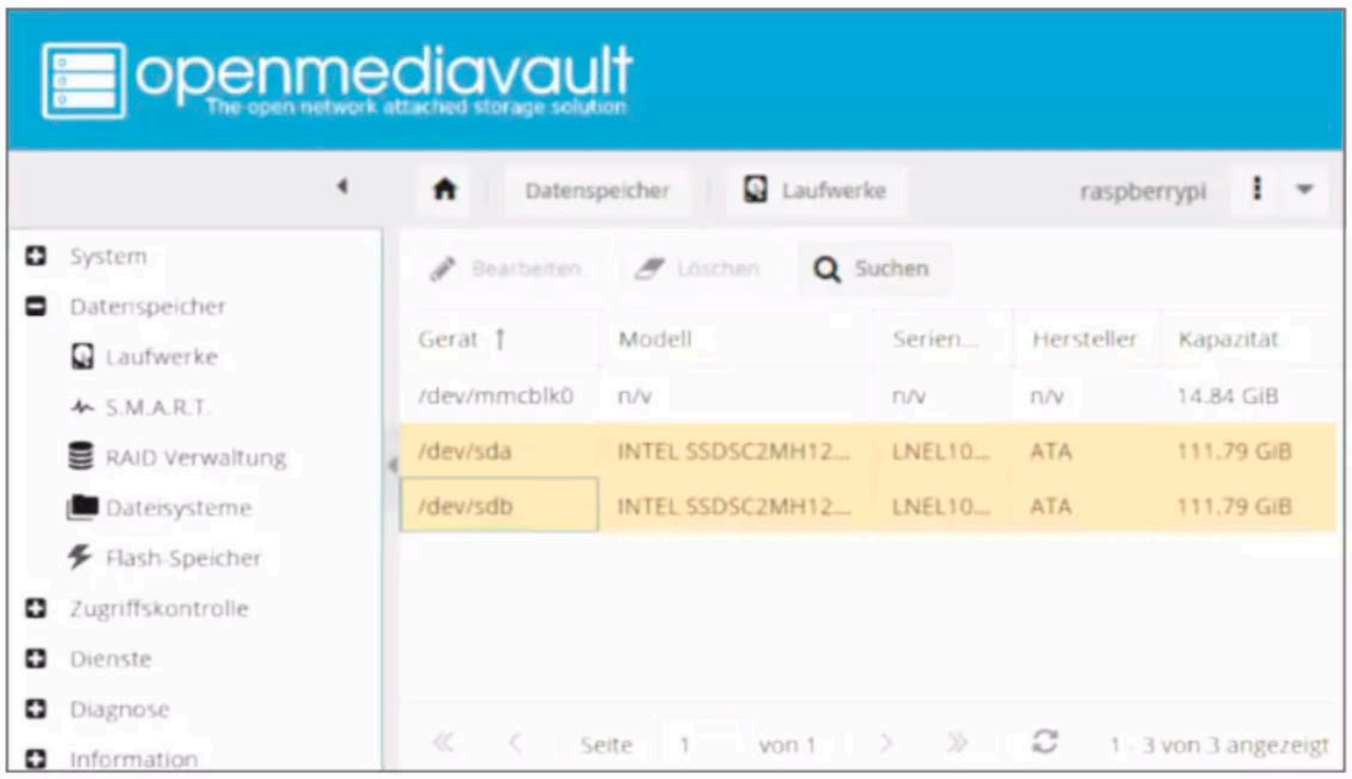
Bei einem CM4102000-Modul mit WLAN lagen die WLAN-Datentransferraten auf demselben Niveau wie beim Raspi 4, also bei maximal rund 80 MBit/s sowohl im 2,4-GHz-Band als auch bei 5 GHz. Anders als ein Raspi 4 hat das CM4 zusätzlich zur Onboard-Antenne eine winzige (U.FL-)Buchse für eine externe Antenne, die man separat kaufen muss. Sie brachte im Test nur moderate Verbesserungen, ist aber unverzichtbar, wenn das CM4 in einem Gerät mit Metallgehäuse zum Einsatz kommt.

## NAS-Herausforderung

Der PCI-Express-Steckplatz des IO-Board erfreut Bastler, die ein Raspi-NAS zusammenstellen wollen. Während das beim normalen Raspi 4 nur mit (externen) USB-Platten oder USB-SSDs möglich ist, kann man auf das CM4-IO-Board auch einen SATA-Adapter stecken. Um es gleich vorweg zu nehmen: Die Sache hat einige Haken.

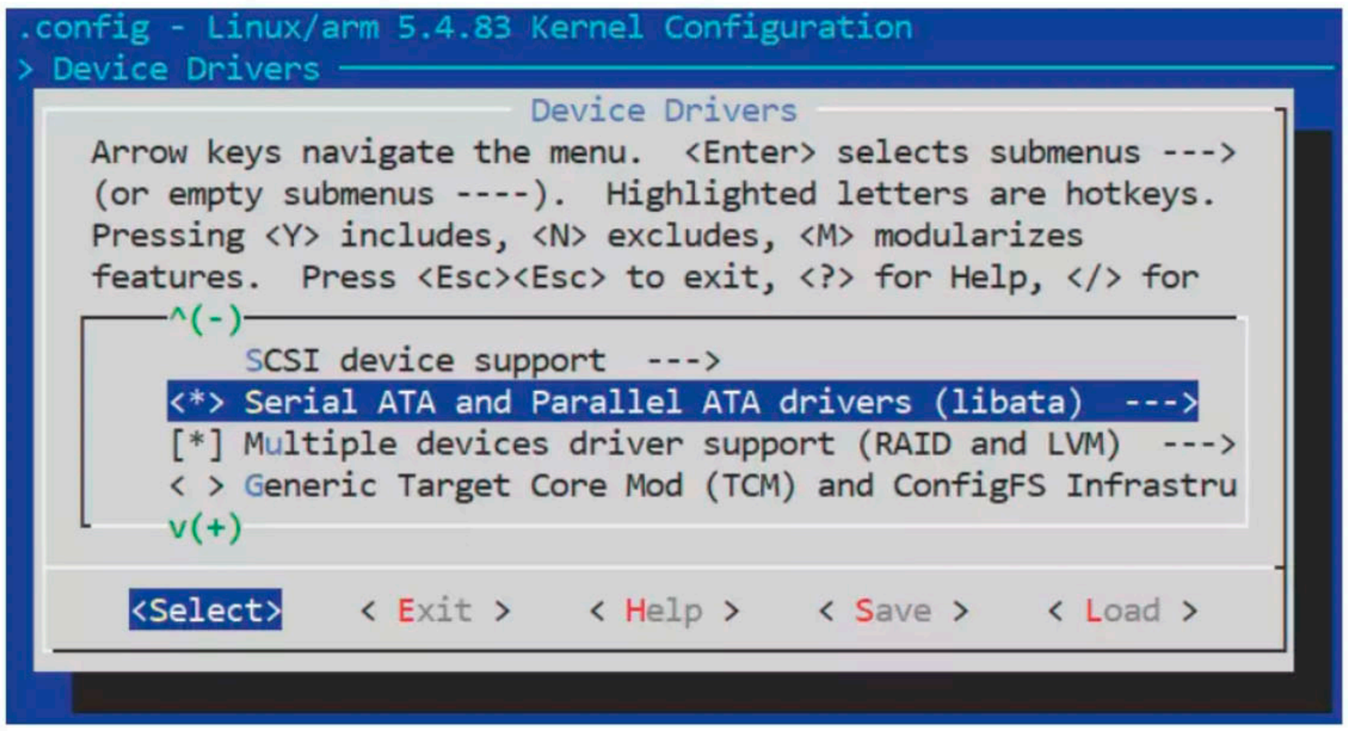
Der Videoblogger Jeff Geerling hat eine Reihe von PCIe-SATA-Adaptern ausprobiert und Tipps dazu veröffentlicht (siehe ct.de/wjjj), von denen wir manche nachvollzogen haben. Zwei Aspekte sind dabei besonders wichtig: Erstens funktionieren nur bestimmte SATA-Adapterkarten gut und zweitens muss man dazu den Linux-Kernel des Raspberry Pi OS und je nach Karte auch die nötigen Treiber neu kompilieren. Letzteres dauert auf dem CM4 nicht nur über eine Stunde, sondern zieht bei späteren Updates des Betriebssystems möglicherweise einigen Mehraufwand nach sich [1]. Denn falls ein Kernel-Update kommt, muss man manchmal wiederum Kernel und Treiber neu übersetzen. Manche PCIe-Karten bringen zudem das System zum Absturz (Kernel Panic), wenn man nicht den PCIe-Adressbereich im RAM (Base Address Register, BAR) durch einen Eingriff in den Device Tree Blob erweitert.

Der angepasste Kernel ist deshalb nötig, weil Raspberry Pi OS standardmäßig keine SATA- beziehungsweise SATAAHCI-Funktionen mitbringt. Außer-




**Im Prinzip lässt sich NAS-Software wie OpenMediaVault auf dem CM4 mit PCIe-SATA-Adapter verwenden, aber dazu muss man den Linux-Kernel neu kompilieren.**

**Um den SATA-Port vom Compute Module nutzen zu können, muss man den Raspi-Kernel leider selbst kompilieren. Das macht bei jedem Kernel-Update Arbeit.**




dem funktionieren bestimmte PCIe-Chips wie die SATA-Adapter Marvell 88SE9120 und 88SE9215 nur mit speziellen Treibern. Genügsamer war in unseren Versuchen eine PCIe-Karte mit dem Asmedia-Chip ASM1061, der mit den SATA-AHCI-Standardfunktionen auskam und auch keine Anpassung des PCIe BAR benötigte.

Die Marvell-Karten verlängerten auch die Bootdauer des CM4 deutlich. Mit mehr als zwei SATA-SSDs wird außerdem die Stromversorgung schwierig: Bei unseren Versuchen mit einer 88SE9215-Karte mit vier SATA-Ports, an die wir insgesamt drei SSDs anschlossen, stieg die Leistungsaufnahme des gesamten Systems beim Schreiben auf ein RAID 5 kurzzeitig auf mehr als 15 Watt. Außerdem lief dieser Versuchsaufbau nicht stabil, schon beim „Syncing" des frisch eingerichteten RAID stürzte das System mehrfach ab. Auf der DeLock-Adapterkarte 90382 trägt der Marvell 88SE9215 einen kleinen Kühlkörper, der sich auf mehr als 70 Grad Celsius erhitzte. Der Chip verheizt mindestens 4,5 Watt – also mehr als der Raspi unter Volllast –, selbst wenn er gar nichts tut. Das IO-Board schaltet zudem die PCIe-Karten im Soft-off-Zustand – also nach dem „Herunterfahren" – nicht ab, sie ziehen dann weiterhin kontinuierlich Strom. Der Asmedia-Chip begnügte sich hingegen mit rund 2 Watt.

Mit zwei SATA-SSDs an einer InLine-Adapterkarte mit ASM1061 haben wir unter OpenMediaVault (OMV5) ein RAID 1 eingerichtet, das erwartungsgemäß funktionierte und über den Gigabit-Ethernet-Anschluss des IO-Boards auf ordentliche Transferraten von 90 bis 105 MByte/s lieferte. Auch eine Syba-Karte mit Marvell-9120-Chip lief ähnlich gut, allerdings wie erwähnt nur mit kompiliertem Treiber und angepasstem PCIe BAR. Als NAS-Ersatz taugt der Aufbau allerdings kaum, weil nach jedem Betriebssystem-Update das Kompilieren des Kernels nötig werden kann und auch ein passendes Gehäuse fehlt.

SATA-RAID-Karten funktionieren nicht, es müssen reine SATA-Adapter sein. Mit Netzwerkkarten von Intel (82874L) und TP-Link (Realtek RTL8168) hatten wir ebenfalls Pech: Die Karten tauchten laut `lspci` zwar im System auf, stellten aber keine Netzwerkverbindung her, selbst mit neu übersetztem Kernel inklusive Treiber. Auch mit dem 64-Bit-Linux Ubuntu for Raspi in Version 20.04 traten ähnliche Schwierigkeiten auf.

## Fazit

Das Raspberry Pi Compute Module 4 macht die Raspi-Technik noch flexibler nutzbar, vor allem dank der PCI-Express-Schnittstelle. Doch um manche PCIe-Karte zum Laufen zu bringen, sind tiefgründige Linuxkenntnisse nötig. Schon um ein CM4 auf dem günstigen IO-Board der Raspberry Pi Foundation in Betrieb zu nehmen, muss man sich ein wenig einarbeiten. Dabei kommt man mit einer „Lite"-Version des CM4 und einer separaten MicroSD-Karte leichter ins Ziel als mit einer der eMMC-Flash-Versionen. Für Bastler wiederum ist ein normaler Raspi fast immer besser geeignet als ein Raspi-Modul wie das CM4. (ciw) c't

### Literatur

[1] Thorsten Leemhis, **FAQ**, Basics zum Linux-Kernel, Teil 3, c't 5/2018, S. 156

**Downloads und Anleitungen**

ct.de/wjjj

WIR MACHEN
KEINE WERBUNG.
WIR MACHEN EUCH
EIN ANGEBOT.

c't

ct.de/angebot

# ICH KAUF MIR DIE c't NICHT. ICH ABONNIER SIE.

Ich möchte c't 3 Monate lang mit 35 % Neukunden-Rabatt testen. Ich lese 6 Ausgaben als Heft oder digital in der App, als PDF oder direkt im Browser.

**Als Willkommensgeschenk erhalte ich eine Prämie nach Wahl, z. B. einen RC-Quadrocopter.**

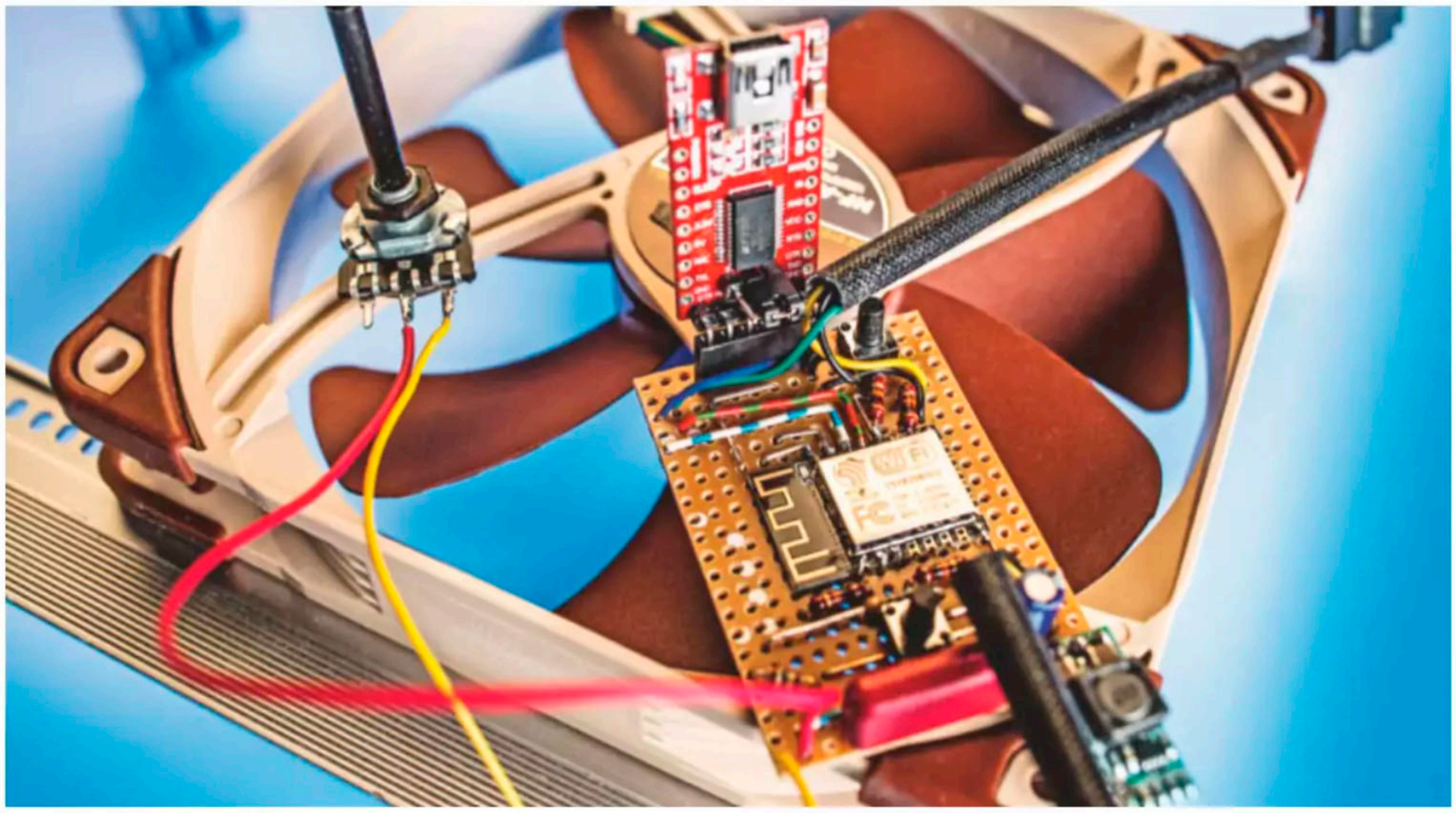


**Jetzt gleich bestellen:**

ct.de/angebot

+49 541/80 009 120

leserservice@heise.de



5V-Lüfter Standfuß

# Lüftersteuerung mit PWM, Poti und WLAN

**Dank eines leisen Luftstroms vom selbst gebauten Lüfter funktionieren zur Überhitzung neigende Geräte auch an heißen Sommertagen. Mit einem ESP8266 zur Lüftersteuerung – auch im Smart-Home – bleiben keine Wünsche offen.**

Von **Pina Merkert**


| | |
|---|---|
| Lüftersteuerung mit PWM, Poti und WLAN | 38 |
| Luftreiniger mit sehr feinem HEPA-Filter | 46 |
| AeVOC misst Aerosole & Feinstaub | 56 |
| ESP-Projekte mit und ohne WLAN | 64 |
| Mehr GPIO-Pins am Microcontroller | 68 |
| Arduino Micro lernt Lautstärkeregler | 76 |
| Arduino-Cockpit für den Flugsimulator | 80 |

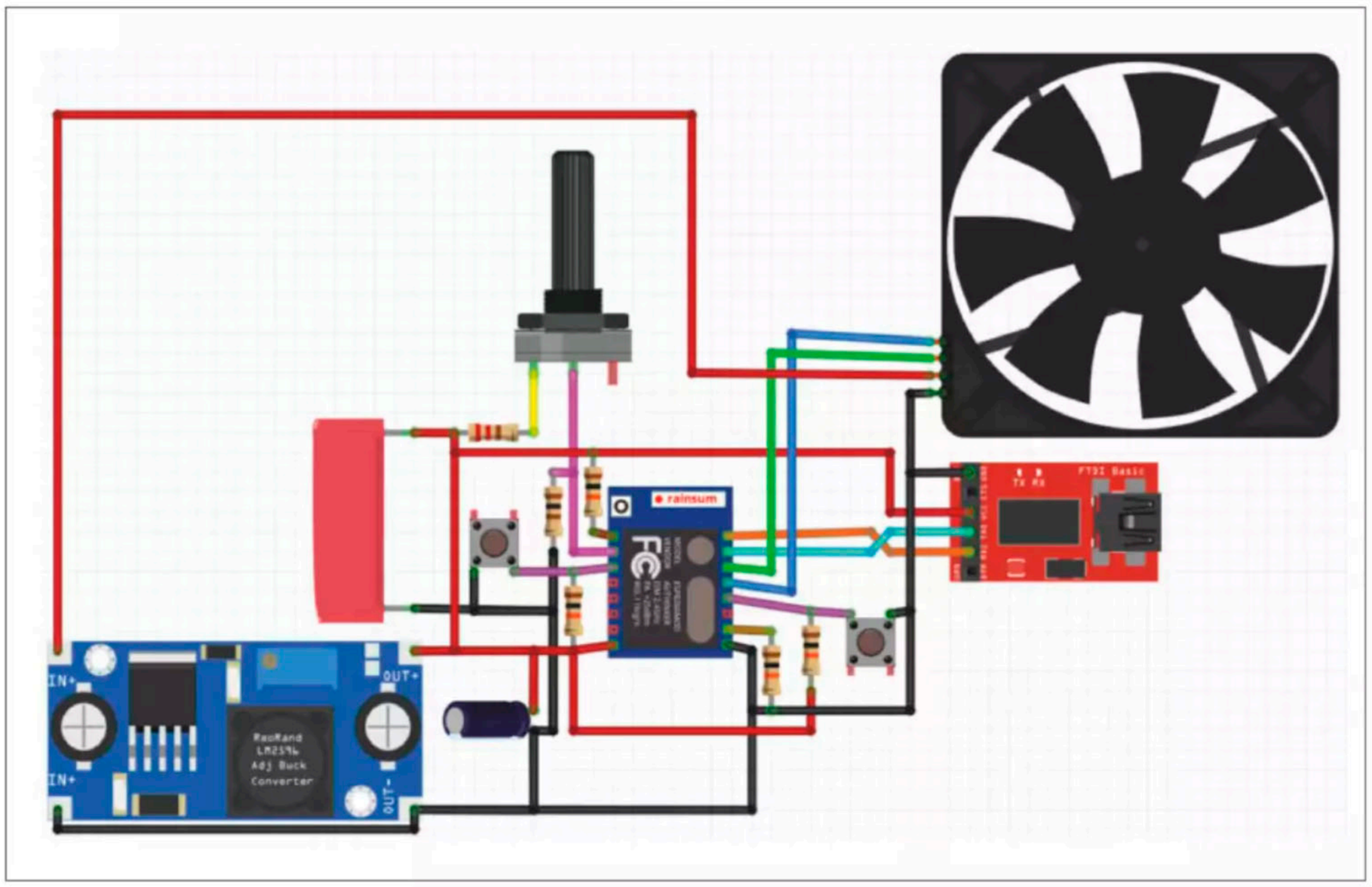
Wer sich für den Videochat im Homeoffice möglichst professionell in Szene setzen will, schließt statt einer Webcam besser per HDMI-Grabber eine Systemkamera wie beispielsweise die Sony α6300 am Rechner an. Blöd nur, wenn die Kamera zum Überhitzen neigt und sich an heißen Sommertagen nach zwanzig Minuten notabschaltet. Schon ein leichter Luftstrom verhindert den Hitzetod, beispielsweise von einem 5-Volt-Lüfter, der praktischerweise vom USB-Port des Rechners Strom bezieht.

Mit dem beim Lüfter beigelegten Adapterkabel dreht der Ventilator aber auf unnötig brausende 2400 Umdrehungen pro Minute auf und stört damit die Tonaufnahme. Eine Steuerung schafft Abhilfe, und moderne Lüfter – egal ob mit 5 oder 12 Volt – bringen die nötige Elektronik gleich mit. Vier Kabel enden im Stecker, der aus dem Gehäuse hängt: ganz links Masse und Stromversorgung (je nach Typ 5 oder 12 Volt), auf Position 3 ein Tacho-Signal, um die Drehzahl zu messen. Den Anschluss zieht der Lüfter bei jeder Umdrehung einmal auf Masse. Der Clou verbirgt sich im vierten Kabel, an dem moderne Lüfter ein pulsweitenmoduliertes Signal (PWM) annehmen (laut Spezifikation mit 25 kHz). Dort stellt man über die Länge der PWM-Impulse die Drehzahl von 0 bis zur höchstmöglichen Drehzahl ein, die der Lüfter schafft, und muss dabei nicht mal den Spannungspegel der Versorgungsspannung nutzen. Das bedeutet, dass sich Lüfter mit 4 Anschlüssen auch ohne Pegelanpassung mit Mikrocontrollern regeln lassen, die mit 3,3 Volt arbeiten.

Was liegt da näher, als die ultimative Lüftersteuerung mit einem ESP8266 zu bauen? Der kleine ESP hat einen einzelnen Analog-Eingang (für Spannungen von 0 bis 1 Volt), an den man mit ein paar Widerständen ein Potenziometer anschließen kann, um die Drehzahl direkt am Gerät zu steuern. Der ESP spricht von Haus aus WLAN und kommuniziert auf Wunsch auch standesgemäß per MQTT [1] mit der Smart-Home-Zentrale. Die kann also automatisch Drehzahlen einstellen, wodurch sich der Lüfter auch skripten lässt.

## Poti am ESP

Potenziometer gibt es in den unterschiedlichsten Bauarten. Besonders wenn man eines aus der Bastelkiste zieht, rät es sich nachzumessen, welche Widerstandswerte es minimal und maximal anneh-




**Unsere Lüftersteuerung erweitert die Standard-Beschaltung eines ESP-12-Moduls (siehe Datenblatt) um einen Spannungsteiler nebst Potenziometer und den Lüfter.**

men kann. Dabei ist es besonders wichtig, das Potenziometer so anzuschließen, dass die Spannung am analogen Eingang des ESP nie über 1 Volt steigt. Der ESP kann sonst Schaden nehmen.

Man berechnet das, indem man zweimal das Ohm'sche Gesetz anwendet (*R=U/I*): Einmal berechnet man den Strom *I*, der durch Poti und alle Widerstände fließt. Danach berechnet man den Spannungsabfall am Widerstand zur Eingangsspannung. Letztlich müssen an dem 2,3 Volt abfallen. Drehen Sie also das Poti ganz nach links, damit Sie am Multimeter den kleinsten Widerstand messen können. Bei unserem Exemplar waren das gerade einmal 10 Ω. Da so ein kleiner Widerstand zu einem hohen Strom führt, empfiehlt es sich, einen Widerstand mit dem Poti in Reihe zu schalten. Wir haben 87 Ω hinters Poti gelötet. Poti plus Widerstand haben Verbindung zu den 3,3 Volt der Stromversorgung. Die Verbindung zur Masse geht

## Potenziometer linearisieren

Bei der Programmierung des ESP genügt ein `analogRead(A0)`, um die Spannung auszulesen. Unser Poti erzeugte aber keineswegs einen linearen Spannungsverlauf, sodass wir die Kurve lieber begradigen wollten. Mit folgender Methode geht das relativ leicht: Schreiben Sie zunächst eine Mini-Firmware, die nur den gemessenen Spannungswert auf die serielle Konsole ausgibt. Befestigen Sie dann eine Klemme als Zeiger am Schaft des Poti. Stellen Sie es nun aufrecht auf ein Blatt Papier und halten den Körper fest. Bewegen Sie nun den Klemme-Zeiger in regelmäßigen Schritten von circa 5 Grad übers Papier und markieren Sie mit einem Bleistiftstrich jeweils die Position des Zeigers. Lesen Sie währenddessen den vom ESP gemessenen Spannungswert auf der seriellen Konsole ab. Nachdem Sie den gesamten Bereich protokolliert haben, über den sich das Poti bewegen kann, nehmen Sie es weg und messen den Winkel der einzelnen Striche mit einem Geodreieck. Tragen Sie die Winkel und die zugehörigen gemessenen Spannungswerte in eine CSV-Datei ein.

Danach geht es mit SciPy und einem Jupyter-Notebook weiter. Mit dem suchen Sie eine näherungsweise Umkehrfunktion zur gemessenen Datenreihe. Wir haben im Repository „fit_fan_poti_curve" (siehe ct.de/w8q9) alle Berechnungen durchgeführt und die Daten geplottet. Unsere Daten finden Sie als CSV im selben Repository. Wenn Sie Ihre Daten im gleichen Format eingeben, müssen Sie im Code nur den Dateinamen ändern.

Zuerst haben wir unsere Daten so geplottet, wie wir sie für die Umkehrfunktion brauchen: Zu jeder Spannung auf der x-Achse gehört ein Winkel auf der y-Achse. An diesem Diagramm erkennt man, welche Form die gesuchte Funktion annehmen muss. Bei uns zeigt der Plot einen fast waagerechten Abschnitt in der Mitte, links einen steilen Abfall und rechts einen noch steileren Anstieg. Ein Polynom mit ungeradem Exponent zeigt eine ähnliche Form. Je höher dieser Exponent, desto größer der flache Bereich und desto steiler die Flanken.

Der Plan ist, ein geeignetes Polynom mit einigen wenigen Parametern zu wählen und deren optimale Werte mit SciPys `curve_fit()` zu ermitteln [2]. Für unser Poti reicht das Polynom $a \cdot ((x-x_{off})^{33}) + b \cdot x + y_{off}$. Der erste Summand sorgt mit dem Exponenten 33 für eine sehr flache Mitte und ziemlich steile Flanken. Der Parameter *a* skaliert die Steilheit der Flanken und $x_{off}$ verschiebt die flache Stelle entlang der x-Achse. Der letzte Summand $y_{off}$ verschiebt die Kurve entlang der y-Achse. Der zweite Summand mit dem Parameter *b* addiert einen Linearanteil, damit das Mittelstück nicht völlig eben sein muss. Die Python-Funktion sieht im Beispiel so aus:

```
def f(x, a, b, x_off, y_off):
  s1 = a*pow(x-x_off, 33)
  return s1+b*x+y_off
```

Um das passende Polynom für Ihr Poti zu finden, sollten Sie sich den Plot Ihrer Daten ansehen und die Steilheit abschätzen, um die maximale Potenz per Hand festzulegen. Achten Sie dabei besonders darauf, wie stark die Kurve abknickt: Größere Exponenten führen zu härteren Knicks. Sie können auch weitere Summanden hinzufügen, falls sich für Ihr Poti eine komplexere Kurve ergibt. Verwenden Sie aber möglichst wenig Parameter und keinesfalls mehr als die Anzahl der Datenpunkte.

Die genauen Werte der Parameter sucht `curve_fit()` exakter, als Sie es per Hand könnten. Für den Aufruf müssen die X- und Y-Werte als gewöhnliche Numpy-Arrays vorliegen:

über einen weiteren Widerstand mit 229 Ω. An dem fallen mindestens 2,3 Volt ab, sodass der in der Mitte angeschlossene Analog-Pin des ESP nie mehr als 1 Volt abbekommt. Beim gegenüberliegenden Extrem unseres Poti hat das 16,6 kΩ, was auf eine Spannung von 0,044 Volt hinaus läuft. Perfekt konnten wir den Wertebereich des analogen Eingangs damit nicht abdecken; wer das optimieren möchte, ersetzt die festen Widerstandswerte durch kleine Potis.

## PWM am Lüfter

Statt mit PWM eine niedrigere Spannung als die Versorgungsspannung zu erzeugen – so regelt man ältere Lüfter mit zwei oder drei Anschlusskabeln – werten die modernen Lüfter lediglich die Länge der PWM-Impulse aus. Wenn die 50 Prozent der Zeit auf high bleiben, geht der Lüfter auf halbe Drehzahl, bei 85 Prozent Impulslänge auf 85 Prozent der Maximal-

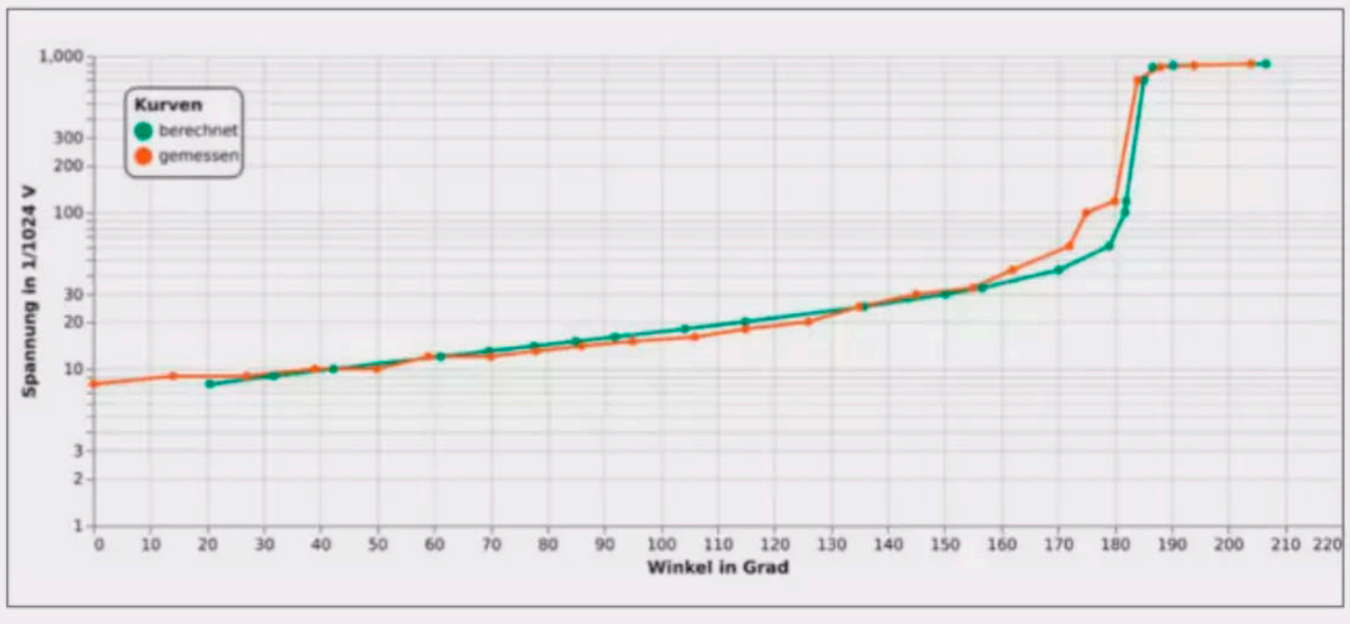
```
angles = np.array(data["Winkel"])
voltages = np.array(data["Spannung"])
params, _ = curve_fit(f,
         xdata=voltages,
         ydata=angles,
         p0=[3e-86, 5.3e-3, 462, 181],
         maxfev=10000)
```

Die optimierten Parameter stehen anschließend in `params`. Für einen grafischen Vergleich können Sie die optimierte Funktion nun für die gleichen X-Werte wie im Datensatz berechnen:

```
simulation = []
for v in data["Spannung"]:
  simulation.append(np.array(
     [f(v, *params), v, "berechnet"]))
simulation = pd.DataFrame(simulation,
                          columns=["Winkel",
                          "Spannung", "Kurve"])
```

Der Vorteil eines angenäherten Polynoms ist, dass die eher begrenzte Mathematik-Bibliothek des Arduino-Universums die nötigen Funktionen besitzt (ein Arcus-Tangens-Hyperbolicus, den man für die Umkehrung einer Sigmoid-Funktion braucht, ist nämlich beispielsweise nicht drin). Die fertige Funktion sieht bei uns dann wie folgt aus:

```
float voltageToAngle(float voltage) {
  return 3.16306348e-86 *
   pow(voltage - 4.62751482e+02, 33) +
   5.33460164e-03 * voltage +
   1.81394601e+02;
}
```




**Der Drehwinkel des Poti und die gemessene Spannung am analogen Eingang stehen in keinem linearen Zusammenhang. Ein Polynom 33. Grades nähert die Umkehrfunktion aber so gut an, dass in der Firmware wieder Winkel ankommen.**

drehzahl und so weiter. Der Vorteil dieser Regelung besteht darin, dass den Lüftern die volle Versorgungsspannung zur Verfügung steht und sie mit deren Energie auch Ungleichheiten im Magnetfeld der Permanentmagnete in den Brushless-Motoren ausgleichen können. In der Praxis heißt das, dass die modernen Lüfter auch bei kleinen Drehzahlen sicher anlaufen.

Der ESP8266 ist so schnell, dass er PWM an allen Pins in Software emulieren kann. Man initialisiert für die Lüftersteuerung in `setup()` daher einfach einen beliebigen Pin als PWM-Ausgang:

```
#define FAN_PWM_PIN 5
pinMode(FAN_PWM_PIN, OUTPUT);
analogWriteRange(PWMRANGE);
analogWriteFreq(25000);
```

In der `loop()`-Funktion genügt später ein `analogWrite(FAN_PWM_PIN, value)`, wobei `value` dabei ein Integer zwischen 0 und `PWMRANGE` (standardmäßig 1024) sein sollte.

Wer messen möchte, wie gut sich die Regelung des Lüfters schlägt, lötet zusätzlich das Tacho-Signal vom dritten Draht des Lüfters an einen Pin an (beispielsweise GPIO 4). Da die Signale oft kommen, bietet sich ein Interrupt an, um sie auszuwerten:

```
#define TACHO_PIN 4
unsigned long lastTrigger;
ICACHE_RAM_ATTR void onFanRotation() {
  unsigned long now = millis();
  reportTimer += now - lastTrigger;
  if (reportTimer > 5000) {
    reportTimer = 0;
    char* fanSpS = (char*) malloc(50);
    sprintf(fanSpS, "%lu",
            60000/(now - lastTrigger));
    Serial.printf(
      "Fan speed: %s rpm.\n", fanSpS);
    free(fanSpS);
  }
  lastTrigger = now;
}
```

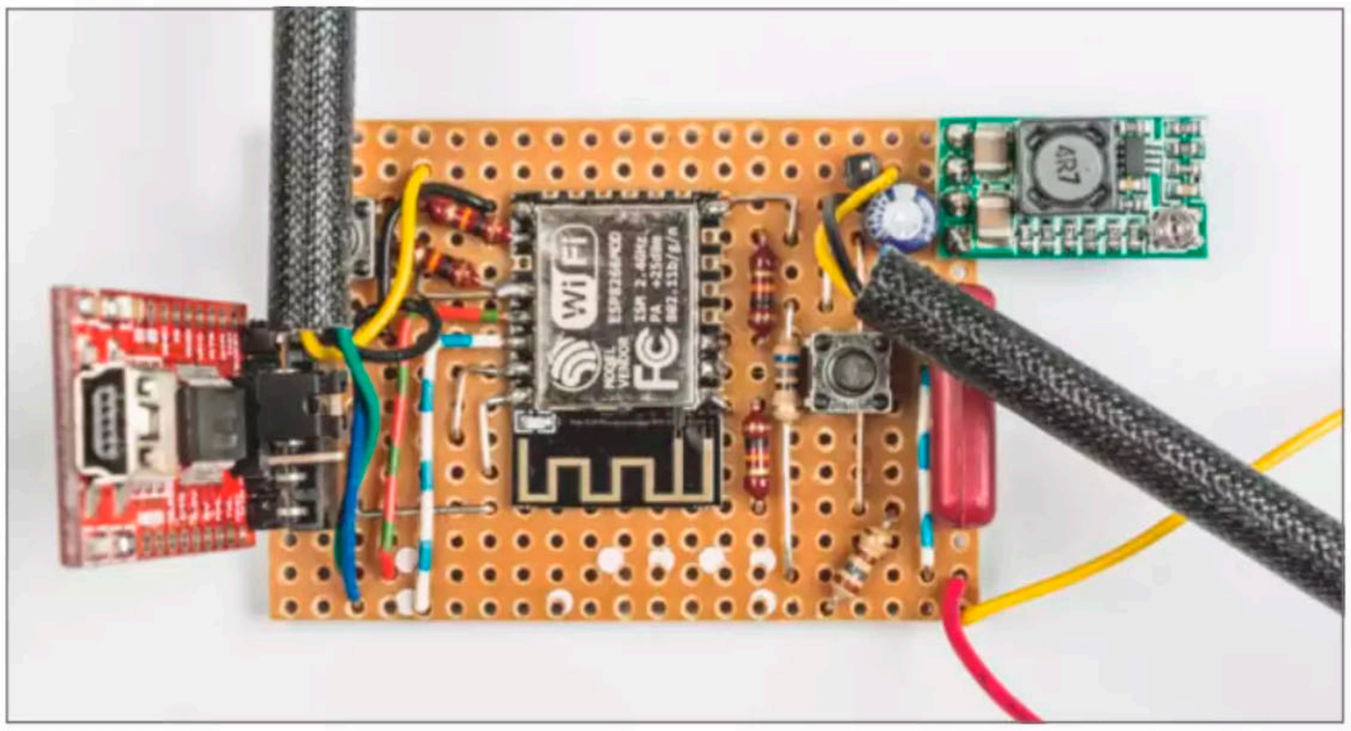
In `setup()` aktiviert man den Interrupt mit:

```
lastTrigger = millis();
attachInterrupt(
     digitalPinToInterrupt(TACHO_PIN),
     onFanRotation, RISING);
```

## WiFiManager

Beim Einbinden des ESP ins WLAN hilft die Bibliothek WiFiManager. Sie kümmert sich darum, ein eigenes WLAN aufzuspannen, wenn der ESP kein WLAN sieht, für das er die Zugangsdaten kennt. Mit dem aufgespannten WLAN (SSID „Wifi-Poti-Fan-<MAC-Adresse-des-ESP>“) kann sich beispielsweise das Smartphone verbinden. Ein Captive-Portal leitet sämtliche Anfragen ins Internet auf die Konfigurations-Webseite des ESP um. Das bedeutet, dass Smartphones automatisch unter dem Titel „im WLAN anmelden“ ein Browser-Widget öffnen, in dem man den ESP konfigurieren darf. Über die Webseite wählt man die SSID des eigenen Heimnetzes, gibt das WLAN-Passwort ein und konfiguriert alle Variablen, die die Firmware zusätzlich abfragt. Die Lüftersteuerung fragt beispielsweise nach der IP-Adresse des MQTT-Brokers, dem Port (normalerweise 1883) und dem Benutzernamen nebst Passwort, falls man MQTT mit Authentifizierung nutzt.

All diese Einstellungen muss der ESP über Neustarts hinweg speichern. WiFiManager speichert zwar automatisch die WLAN-Zugangsdaten, Zusatzparameter wie die MQTT-Konfiguration muss die Firmware aber selbst speichern. Das geht relativ leicht mit der `EEPROM`-Bibliothek. Der ESP besitzt zwar kein EEPROM, kann aber einen Teil seines Flash-Spei-



**Auf einer Lochrasterplatine ist die Schaltung schnell aufgebaut und passt problemlos in ein kleines Gehäuse. Das sollte nicht aus Blech bestehen, da der ESP sonst kein WLAN empfängt.**

## Poti lieber als Spannungsteiler

Man kann ein Potenziometer durchaus wie hier beschrieben als veränderlichen Widerstand betreiben. Dann fällt aber in vielen Fällen eine aufwendige Linearisierung der Widerstandskurve an und die Auflösung des Analog-Digital-Wandlers produziert Aliasing-Effekte. Wie die Linearisierung geht, ist jedoch interessant genug, um sie hier zu erklären.

Wer dagegen lediglich ein Poti optimal auslesen will, schließt es lieber wie ab Seite 46 beschrieben als Spannungsteiler an. Die Schaltung ist dort ansonsten identisch. Wir haben den Code dort außerdem auf IotWebConf umgestellt, da WifiManager blockiert, wenn er keine WLAN-Daten hat. Wie das geht, steht auf Seite 64.

chers nutzen, um ein EEPROM zu emulieren. Gegenüber der Arduino-Bibliothek besteht der Unterschied nur darin, dass man zuerst mit `EEPROM.begin(size)` den Bereich des Flash-Speicher reserviert. Danach funktioniert `EEPROM.get()` und `EEPROM.put()` wie beim Arduino, indem man eine Adresse angibt, die bei 0 beginnt. Nach dem Schreiben in den Speicher muss man die Operation mit `EEPROM.commit()` abschließen, da der ESP standardmäßig eine Kopie im RAM vorhält, was Schreibzugriffe auf den Flash reduziert und so seine Lebensdauer schont.

Die ganze Konfiguration des WiFiManager und die EEPROM-Verwaltung finden Sie in der Datei `src/config.cpp` im Git-Repository zum Artikel (siehe ct.de/w8q9). Hier nur ein Überblick aus der Vogelperspektive: Den WiFiManager initialisieren Sie, indem Sie ein Objekt anlegen:

```
WiFiManager wifiManager;
```

Mit diesem Objekt können Sie anschließend Konfigurationsvariablen definieren, die die Konfigurations-Webseite des Captive-Portal abfragt:

```
WiFiManagerParameter mqtt_srv(
          "mqttServer", "MQTT Server",
          configData.mqttServer,
          MQTT_SERVER_STRLEN);
wifiManager.addParameter(&mqtt_srv);
```

Wir haben zum Speichern der Konfigurationsvariablen ein `struct` definiert:

```
#define MQTT_SERVER_STRLEN 64
#define MQTT_PORT_BYTES_LEN 4
#define MQTT_USERNAME_STRLEN 64
#define MQTT_PASSWORD_STRLEN 64
#define CONFIG_LEN 196
struct configDataStruct {
  char mqttServer[
          MQTT_SERVER_STRLEN] = "";
  unsigned long mqttPort = 1883;
  char mqttUsername[
          MQTT_USERNAME_STRLEN] = "";
  char mqttPassword[
          MQTT_PASSWORD_STRLEN] = "";
};
```

Das macht es besonders einfach, die gesamte Konfiguration im EEPROM zu sichern:

```
EEPROM.put(4, configData);
EEPROM.commit();
```

Damit ab Byte 4 sinnvolle Daten landen, muss der WifiManager aber zuerst sein Interface anzeigen:

```
wifiManager.autoConnect(
        getDeviceName(), "loving_ct");
```

Der String `"loving_ct"` definiert dabei das Passwort für das aufgespannte WLAN des unkonfigurierten ESPs.

Nach `autoConnect()` liest der Code die Parameter folgendermaßen aus:

```
strncpy(configData.mqttServer,
        mqtt_server.getValue(),
        MQTT_SERVER_STRLEN);
```

WifiManager und EEPROM kümmern sich um den komplizierten Teil des Konfigurationsprozesses. Bis alles funktioniert und alle Parameter verlässlich gespeichert sind, debuggt man aber dennoch eine Weile an der Konfiguration herum. Da aber 90 Prozent der selbst gebauten Smart-Home-Geräte mit ESP8266 genau dieselben Parameter abfragen (WLAN-Zugangsdaten und Daten des MQTT-Brokers), können Sie einfach unsere Dateien `config.cpp`, `config.h`, `conversionHelpers.cpp` und `conversionHelpers.h` kopieren und nur den Gerätenamen an Ihr Projekt anpassen.

## MQTT

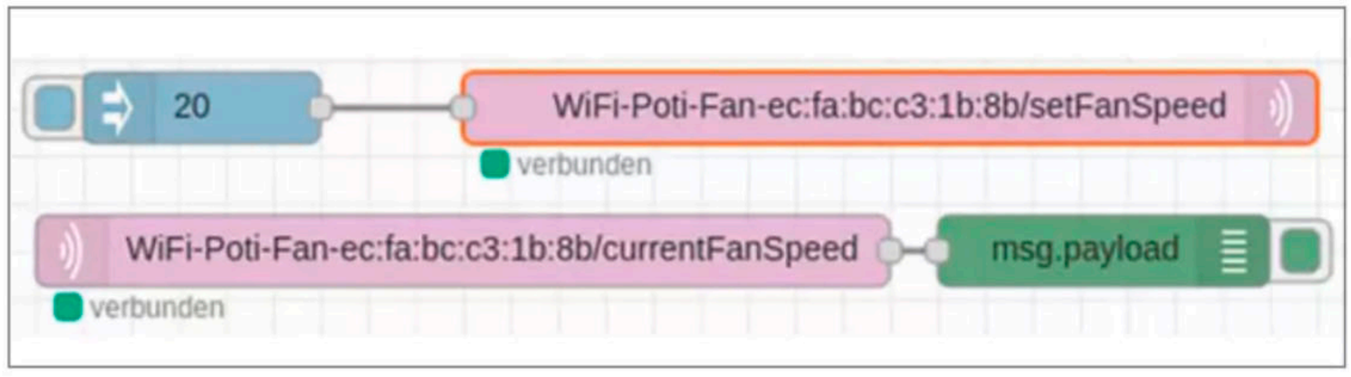
Für die Kommunikation per MQTT macht „PubSubClient“ einen guten Job. Die Bibliothek abstrahiert das Protokoll vollständig. Wenn Sie unsere `mqtt.cpp` und `mqtt.h` übernehmen, wird die Kommunikation aber noch einfacher.

In `setup()` rufen Sie einfach `mqttSetup(mqttSubscribe)` auf. `mqttSubscribe()` ist eine Callback-Funktion, in der Sie definieren, auf welche Topics das MQTT-Gerät reagiert. Bei der Lüftersteuerung sieht sie folgendermaßen aus:

```
void mqttSubscribe() {
  subscribeToTopic("setFanSpeed",
                   setFanSpeed);
}
```

Die Lüftersteuerung meldet beim MQTT-Broker lediglich Interesse am Topic „Wifi-Poti-Fan-<MAC-Adresse-des-ESP>/setFanSpeed“ an. Die Callback-Funktion `void setFanSpeed(char* payload)` ruft der Code in `mqtt.cpp` auf, wenn eine Nachricht zum Topic eingeht. Die erhält einen `payload` als nullterminierten String, der in unserem Fall eine Dezimalzahl enthält. Man könnte dort auch JSON verschicken, der Parser-Aufruf der JSON-Bibliothek gehört dann in die Callback-Funktion.

Möchte der eigene Code Infos an den Broker schicken, geht das mit `publishToMqtt()`. Die Funktion erwartet analog zum Subscribe ein `topic`, das unser Code automatisch um den Gerätenamen ergänzt. Als zweiter Parameter folgt der `payload`. Für die gemessene Drehzahl des Lüfters sieht das beispielsweise so aus:




**Die WLAN-Lüftersteuerung nimmt ihre Befehle per MQTT entgegen. Die Smart-Home-Steuerung Node-Red bringt passende MQTT-Nodes mit. Wer aus dem Urlaub per Web lüften möchte, ergänzt ein Node-Red-Dashboard.**

```
publishToMqtt("currentFanSpeed",
              fanSpS);
```

Damit PubSubClient aufmerksam bleibt, muss `loop()` die `mqttLoop()` regelmäßig aufrufen.

Den kompletten Code unseres MQTT-Helferlein finden Sie im Git-Repository. Für eine schnellere Orientierung im Code hier nur eine kurze Beschreibung: Der Code erweitert bei jedem Subscribe eine Liste mit MQTT-Topics. Jeder Listeneintrag enthält den Topic-Namen und die Callback-Funktion, die auf die Nachricht reagiert. Im PubSubClient treffen alle empfangenen Nachrichten in derselben Funktion `onMqttMessage()` ein. Die geht jeweils die Liste durch und gibt den `payload` an die Callback-Funktion weiter, wenn der Name übereinstimmt. Ansonsten holt sich der Code nur selbsttätig die Konfiguration aus `config.h` und spart so ein paar Zeilen in `main.cpp`.

## Betrieb

Einmal konfiguriert, merkt sich die Lüftersteuerung die Zugangsdaten, verbindet sich automatisch mit dem WLAN und meldet sich beim MQTT-Broker. Ist der Strom mal weg, dauert ein erneuter Verbindungsaufbau circa 5 Sekunden. Zum Einstellen der Drehzahl bewegt man einfach das Poti. Ganz nach links gedreht, stoppt der Lüfter. Unserer dreht auf höchster Stufe auf 2400 Umdrehungen pro Minute auf, sodass man den Wind spürt und leider auch hört.

Zur automatischen Steuerung erzeugt man die nötigen MQTT-Nachrichten leicht mit Node-Red [3]. Die Smart-Home-Steuerung wertet beispielsweise die Messwerte eines Sensors für biologische Verbindungen im Bad aus (ja, es gibt Sensoren für Ge-

### Literatur


[1] Jan Mahn, **Weltsprache**, Das Protokoll MQTT für robusten Datenaustausch in Industrie und Hausautomation, c’t Projekte 2019, S. 96

[2] Pina Merkert, **Formelsuche für Faule**, Parameter optimieren mit SciPy curve_fit(), c’t 12/2019, S. 188

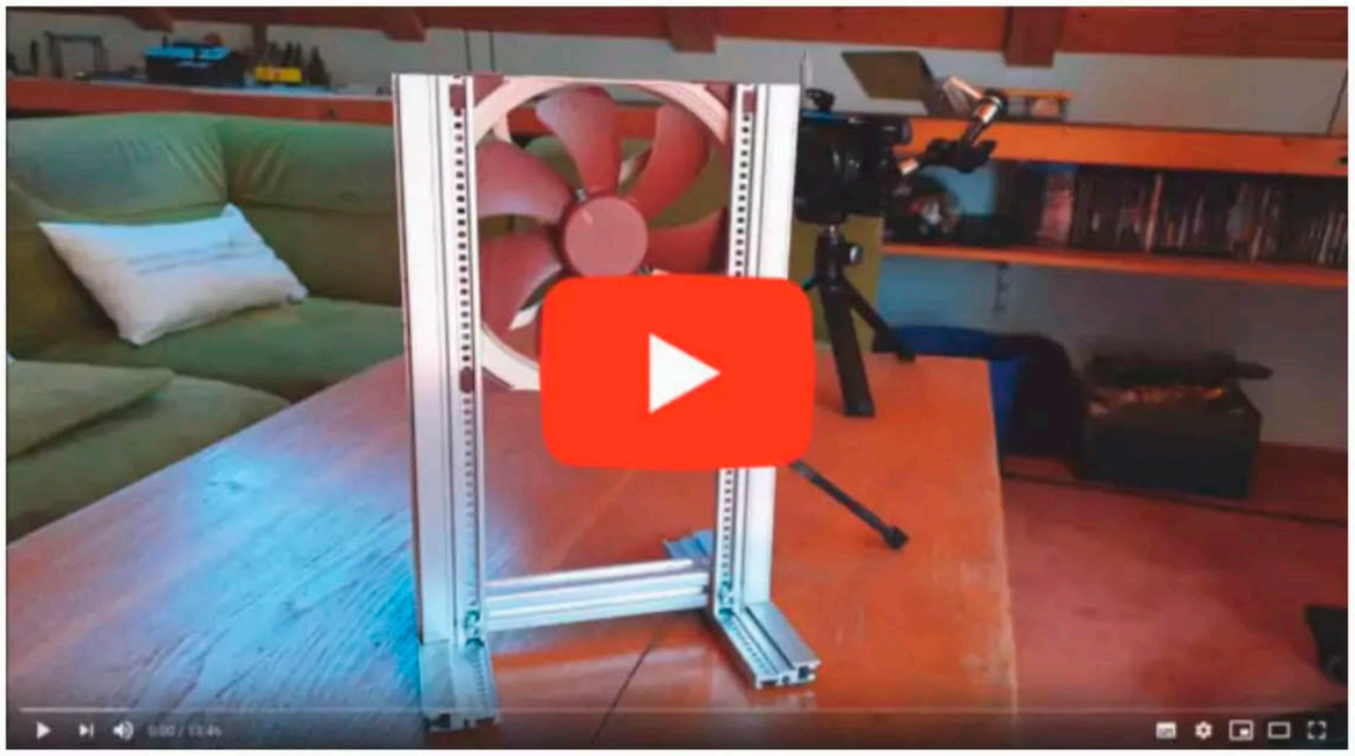
[3] Jan Mahn, **Reaktionsmaschine**, Einstieg in Heimautomation mit Node-Red, c’t Projekte 2019, S. 100

[4] Merlin Schumacher, **Rotfront!**, Grafische Oberflächen für Node-Red entwickeln, c’t Projekte 2019, S. 108


**Repositories bei GitHub**

ct.de/w8q9

**Unter https://www.youtube.com/watch?v=60N5t6khfiA gibt es ein Bastel-Video für den Standfuß.**



stank) und sendet dem Lüfter eine 100 für volle Leistung und fünf Minuten später eine 0, damit er Ruhe gibt. Will man, wie ich, ein Gerät kühlen, bietet es sich auch an, die Luftmenge mit der Temperatur zu regeln.

Wer aus dem Urlaub mal Wind zu Hause machen möchte, integriert einen Stellregler ins Node-Red-Dashboard [4] und gibt das Smart-Home-Interface per Portfreigabe frei (Passwortschutz nicht vergessen!). Ob Sie das brauchen, hängt davon ab, wie kreativ Sie Ihr Smart-Home steuern möchten. Wir wollten mit der relativ einfachen Lüftersteuerung an einem Beispiel zeigen, wie Sie selbst gebaute Aktoren ins Smart Home integrieren.

Wenn Ihnen MQTT-Signale mal zu ätherisch werden, können Sie jederzeit in den manuellen Modus mit Poti zurückschalten. Solange das Poti nicht nennenswert bewegt wird, behält die Steuerung die per MQTT gesetzte Drehzahl bei. Verdrehen Sie das Poti aber weit, wechselt die Steuerung zurück in den manuellen Modus und sie können sofort wieder jede Drehzahl übers Poti einstellen. Im Zweifel einfach einmal ganz hoch und wieder runter drehen.

Wie Sie den Lüfter genau befestigen, hängt stark davon ab, wo Sie den Wind brauchen. Den Bau unseres Ständers zum Kühlen der Kamera haben wir in einem Video dokumentiert, das Sie über ct.de/w8q9 finden. Wir haben Aluprofile zusammen geschraubt. Vielleicht bringt Sie das auf Ideen für Ihre Befestigung. Wenn Sie etwas Raffiniertes gebaut haben, freuen wir uns über Bilder. (pmk) c't





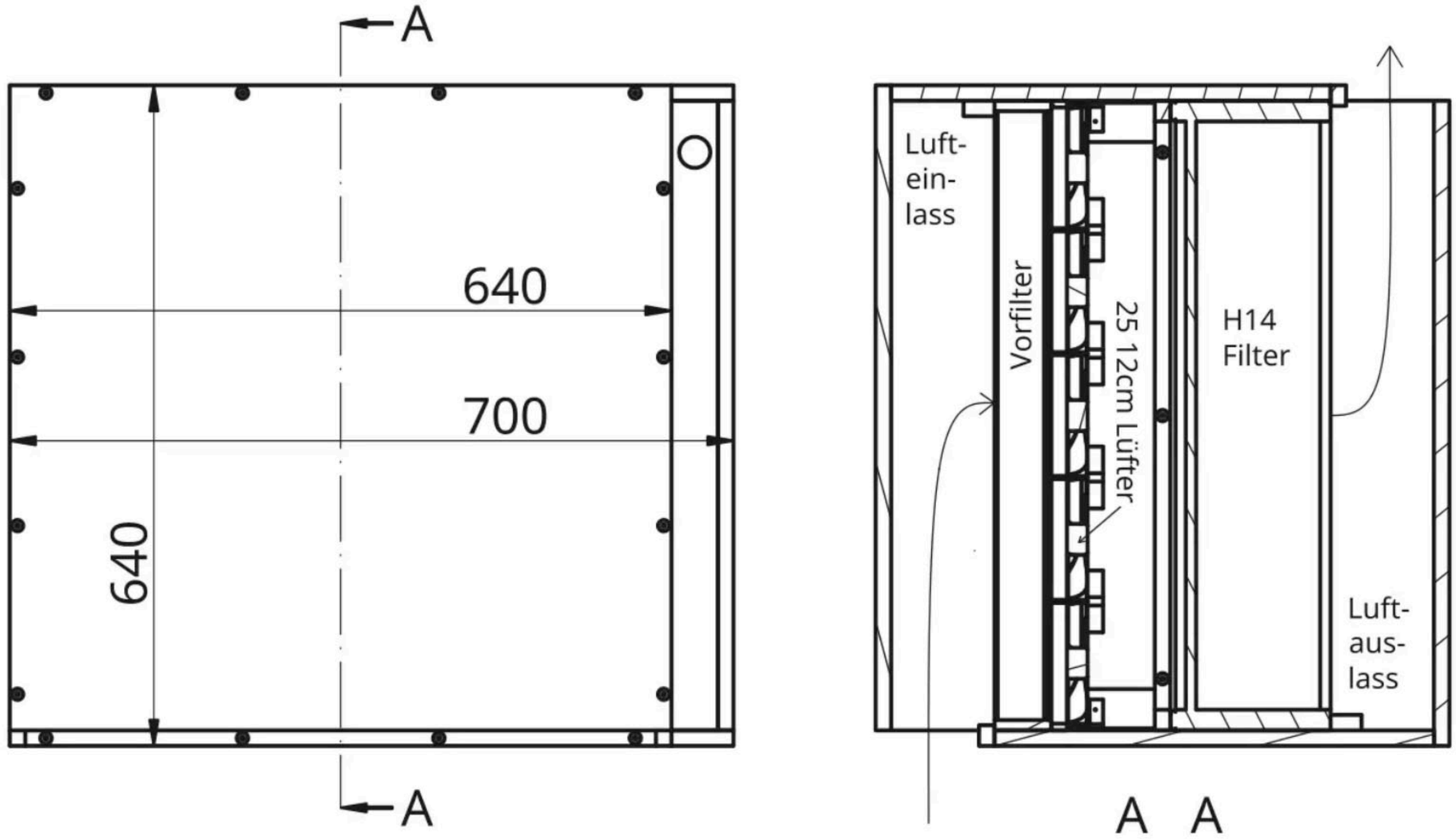

Luftreiniger

# Luftreiniger mit sehr feinem HEPA-Filter

**Coronaviren, Feinstaub, Partikel: Ein Luftreiniger entfernt lungengängige Gefahren. Statt eines tosenden Gebläses bewegen bei diesem Selbstbau-Luftreiniger 25 leise PC-Lüfter 32 Kubikmeter Raumluft pro Stunde – regelbar und mit WLAN.**

Von **Pina Merkert**

Die Coronapandemie hält die Welt im Atem. Denn mit dem Atem stoßen Infizierte Aerosole aus, die sich im Innenraum schnell anreichern. Wer mit im Raum ist, steckt sich über diesen Weg leider ruckzuck an. Vor kritischen Aerosolkonzentrationen schützt ständiges Lüften oder ein Luftreiniger. Denn Luftreiniger pusten die Raumluft durch feine Filter, in der die Aerosol-Tröpfchen hängen bleiben. Im Handel erhältliche Geräte nutzen HEPA-Filter. „HEPA“ (High-Efficiency Particulate Air/




**Schnitt durch den Luftreiniger: Die Luft strömt vorne unten ein, durchquert den Vorfilter, wird von den PC-Lüftern beschleunigt, quetscht sich durch den H14-Filter und gelangt hinten oben wieder in den Raum.**

Arrestance) ist ein Oberbegriff für verschiedene Filterklassen: Luftreiniger für den Heimbereich verkauft die Branche bis zur Filterklasse H13, Experten empfehlen gegen die winzigen Coronaviren aber H14. H14 filtert 99,995 Prozent aller Partikel aus der Luft, einschließlich Aerosolen, Viren und Bakterien. H14-Luftreiniger gibt es zwar auch zu kaufen, allerdings als große, hochpreisige Geräte, beispielsweise für Arztpraxen.

Folgen Sie dieser Anleitung, können Sie für 750 Euro einen angenehm leisen Luftreiniger mit H14-Filter bauen, der bis zu 32 Kubikmeter pro Stunde bewegt. Die Luftmenge entspricht in einem Wohnzimmer mit 25 Quadratmetern etwa dem, was eine niedrig eingestellte Lüftungsanlage bewegt. Gemessen haben wir das mit einem Anemometer, denn die gemessene Strömungsgeschwindigkeit multipliziert mit der Querschnittsfläche des Luftauslasses ergibt die Luftmenge pro Zeit. Auf höchster Stufe drücken die meisten kommerziellen Luftreiniger mehr durch ihre nicht ganz so feinen Filter. Dafür bleibt das Gerät aber stets sehr leise (im Bereich um 45 Dezibel).

Im Inneren sitzt ein quadratischer Feinfilter mit 61 Zentimetern Seitenlänge und einer Dicke von 15 Zentimetern. Dazu kommen ein 2,4 Zentimeter dicker Vorfilter und zwischendrin ein Brett, auf dem 25 PC-Lüfter mit 12 Zentimetern Seitenlänge Platz finden. Um die zu befestigen und genug Platz für die Verkabelung zu haben, sitzen auf dem Brett 10 Zentimeter hohe Abstandshalter. Vorn saugt das Gerät in einem 10 Zentimeter breiten Kanal kurz überm Boden Luft an, hinten pustet es gereinigte Luft über einen ebenso breiten Kanal oben wieder heraus. Es entsteht eine Kiste mit 64 Zentimetern

## Teileliste

| Anzahl | Teil | Preis |
|---|---|---|
| 1 | H14-Filter (SFM14-610X610X150-1S) | 157,66 € |
| 1 | G4-Vorfilter (FZ4-592x592x24-1) | 12,80 € |
| 25 | Noctua NF-F12 PWM | 497,50 € |
| 1 | 12V 2A Netzteil | 8,99 € |
| 1 | ESP-12F (ESP8266) | 1,53 € |
| 4 | 10 k Widerstand | 0,16 € |
| 2 | Kurzhubtaster 6×6 mm, Höhe: 4,3 mm | 0,32 € |
| 1 | Elko radial, 10 µF | 0,14 € |
| 1 | Vielschicht-Kerko, 100 nF | 0,03 € |
| 1 | DC/DC Spannungsregler 12 V zu 3,3 V | 1,92 € |
| 1 | Drehpotentiometer, 10 kΩ, linear | 7,80 € |
| 1 | Trimmpotentiometer, 10 kΩ, linear | 0,31 € |
| 1 | Buchsenleisten 2,54 mm, 1×6, gerade | 0,21 € |
| 1 | Streifenrasterplatine | 1,99 € |
| 1 | DC-Einbaubuchse 5,5 mm/2,1 mm | 2,45 € |
| 3 | Siebdruckplatte Birke 125 cm×83 cm | 59,97 € |
| | **Summe** | **753,78 €** |

Höhe und Breite und 56 Zentimetern Tiefe. Die steht auf 10 Zentimeter hohen Rollen, sodass die Konstruktion insgesamt die Höhe eines klobigen Beistelltischs annimmt. Solange man den Luftauslass frei lässt, darf man ruhig Dinge auf den Luftreiniger stellen.

Auf der rechten Seite ist ein 6 Zentimeter schmaler Kasten für die Kabel und die Elektronik angeschraubt. Die Steuerung übernimmt ein ESP8266, der die 25 Vier-Pin-Lüfter mit einem PWM-Signal versorgt. Die Verschaltung und die Firmware entsprechen der Lüftersteuerung, die wir bereits veröffentlicht haben (siehe S. 38), allerdings wertet nun ein lineares und als Spannungsteiler angeschlossenes Poti beide Projekte auf. Eine aufwendige Linearisierung der Potistellungen entfällt dadurch. Auch die Firmware ist verbessert. Sie nutzt nun IotWebConf (siehe S. 64) und steuert damit die Lüfter auch ohne WLAN oder MQTT-Server. Ist ein WLAN per Webinterface eingerichtet und ein MQTT-Server erreichbar (beispielsweise mit unserem Projekt c't Smart Home [1]), befolgt der Luftreiniger Steuerbefehle vom Smart Home. Auf YouTube finden Sie ein Video zum Bau des Luftreinigers (siehe ct.de/wwkx).

## Teilebeschaffung

Der feine Filter ist wegen erhöhter Nachfrage eventuell schwer oder nur mit Verzögerung zu bekommen. Das verwendete Modell hört auf den Namen „SFM14-610X610X150-1S“ und schlug mit 157,66 Euro zu Buche. Achten Sie bei der Beschaffung neben der Filterklasse H14 auf die richtigen Außenmaße (61 × 61 × 15 cm) und darauf, dass am MDF-Rahmen des Filters eine Dichtung an der Lufteinlassseite mitgeliefert wird.

Der Vorfilter kostet nur 13 Euro, wird aber meist mit einem wenig maßhaltigen Papprahmen und ohne Dichtung geliefert. Er ist aber schnell mit Heißkleber luftdicht in einen selbst gesägten MDF-Rahmen eingeklebt und um eine selbstklebende Fensterdichtung auf der Luftauslassseite ergänzt. Die Güteklasse G4 reicht, um groben Staub von den Lüftern fern zu halten und bremst den Luftfluss nur minimal.

Als Lüfter genügen alle üblichen 12V-PC-Lüfter mit 12 Zentimetern Seitenlänge und einem Anschlusskabel mit vier Pins. 25 Noctua NF-F12 PWM haben im Test mit etwa 45 dB gepustet, was hörbar ist, aber angenehm leise. Die Lüfter produzieren keine hochfrequenten Motorgeräusche, sondern ein niederfrequentes Windgeräusch. Da sie wegen ihrer großen Anzahl auch den größten Kostenfaktor darstellen, ist es durchaus erlaubt, passende Lüfter aus alten Rechnern zu plündern und Hersteller zu mischen. Es sollte nur keiner mit ausgeschlagenem Lager dabei sein, da schadhafte Lüfter zu laut drehen und mit hochfrequentem Lärm nerven.

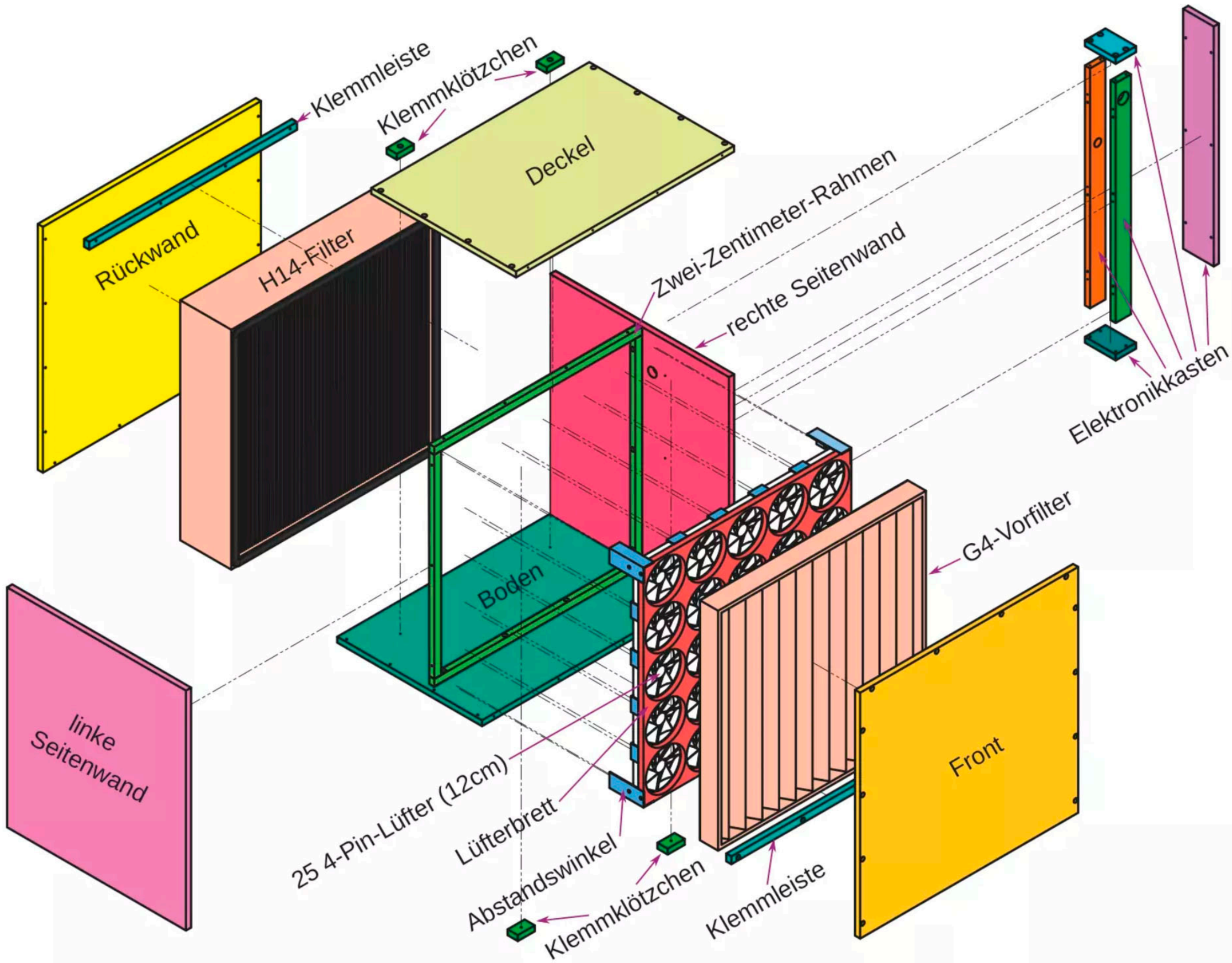
Als Prozessor dient ein ESP-12F-Modul (Schaltung auf S. 39). Dazu kommt ein Step-Down-Regler, den



**Statt eines lauten Radiallüfters drücken 25 leise PC-Ventilatoren die Raumluft durch die Filter.**

Sie auf 12 Volt Eingangsspannung und 3,3 Volt Ausgangsspannung einstellen. Um Brownouts zu vermeiden (Rechenfehler des Mikrocontrollers wegen unzureichender Spannung), empfiehlt das Datenblatt des ESP-12F einen 10µF- und einen 100nF-Kondensator zwischen Eingangsspannung und Masse. Dazu brauchen Sie noch vier Pull-Down- oder Pull-Up-Widerstände mit 10 kΩ und für Reset und Programmiermodus noch zwei Taster. Für die Drehzahleinstellung nutzen Sie ein lineares Poti aus dem Audiobereich und zum Einstellen des Spannungsteilers noch ein Trimmpoti. Zum Flashen stecken Sie einen FTDI-Adapter in eine Sechs-Pin-Buchsenleiste.

Ein 12V-2A-Steckernetzteil reicht als Spannungsversorgung. Um es abziehen zu können, brauchen Sie eine 5,5-Millimeter-Hohlsteckerbuchse zum Einschrauben. Als Fronten für das Poti und die Buchse




**Vorfilter und Lüfter stützen sich von rechts gegen einen Rahmen aus 2 Zentimeter breiten Leisten im Korpus. Der Hauptfilter drückt und dichtet von links. Vorn öffnet sich der Zuluftkanal hinter der blassgelben Platte nach unten, hinten der Abluftkanal vor der kräftig gelben Platte nach oben.**

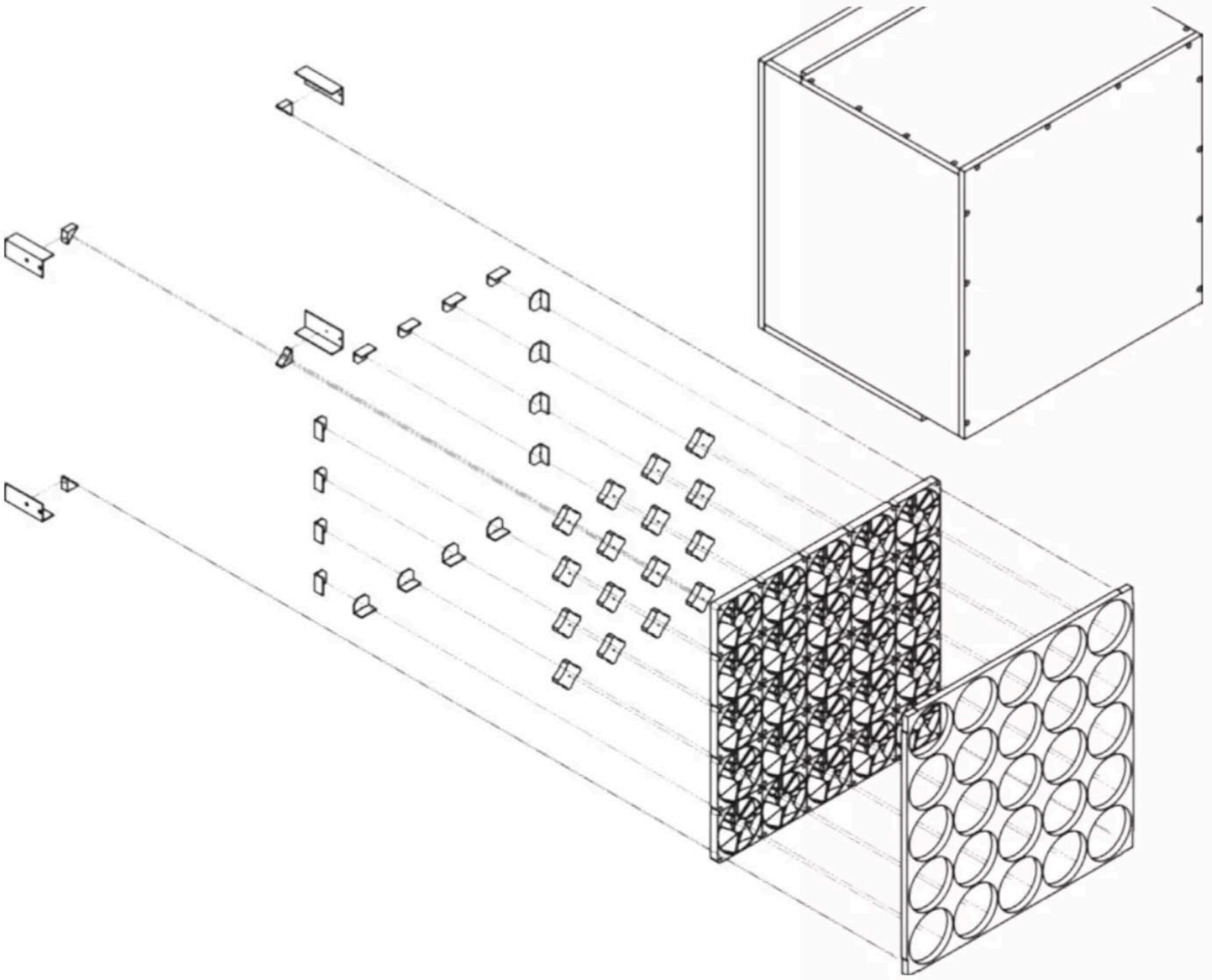
brauchen Sie kleine Bleche, beispielsweise zwei Quadrate mit 5 und 3 Zentimetern Seitenlänge aus 1 Millimeter dickem Messingblech. Für die Verdrahtung sollten Sie mindestens 4 Meter Litze parat haben und für die Spannungsversorgung ein etwas dickeres 2-Meter-Kabel mit zwei Adern und einem Querschnitt von 0,75 Quadratmillimeter. Kleine Abschnitte aus Schrumpfschlauch mit 6 Millimetern Durchmesser halten die Kabel zusammen.

Der Rahmen besteht aus 15 Millimeter dicken Siebdruckplatten, weil die eine glatte Seite haben, die sehr gut gegen eine Dichtung abschließt. Für den Zuschnitt-Service im Baumarkt können Sie die Maße den technischen Zeichnungen entnehmen (siehe ct.de/wwkx). Wer selbst sägt, kommt mit drei handelsüblichen Platten (125 × 83 cm) nur hin, wenn die Rückwand aus zwei Teilen zusammengesetzt wird. Außerdem brauchen Sie noch vier schwenkbare 6 bis 12 Zentimeter hohe Rollen, wovon zumindest zwei eine Bremse haben sollten.

## Lochbrett

Stellen Sie zuerst die Platte her, auf der die Lüfter sitzen. Sie braucht 25 runde Ausschnitte, durch die die Lüfter ungehindert saugen können. Sägen Sie zuerst die Außenmaße von 61 Zentimetern. Falls Sie den H14-Filter schon da haben, können Sie ihn als Schablone zum Anzeichnen verwenden. Danach platzieren Sie die 25 Lüfter regelmäßig auf der Platte. Außen sollte zu jedem Rand ein Abstand von 2 Millimetern bleiben. Zeichnen Sie nun den Innenkreis der Lüfter nach. Das wird nicht perfekt rund, reicht aber für einen Ausschnitt mit der Stichsäge. Verschaffen Sie deren Sägeblatt etwas Platz mit einer Bohrung im Inneren. Achten Sie darauf, dass beim Bohren kein Ausriss entsteht, der die glatte Oberfläche zum Rand zerstört, weil dort später die Dichtung des Vorfilters andocken soll.

Sollten die Lüfter wie die NF-12 an den Ecken gummiert sein, können Sie den Lärmpegel senken,



**Stoßen vier Lüfter aneinander, kann ein quadratisches Brettchen sie mit einer Schraube festklemmen. Am Rand fehlen zwei Lüfter als Gegenpol, weshalb ein Aluwinkel in gleicher Höhe für die nötige Klemmung sorgt. In den Ecken dienen die Abstandswinkel als Halter für klemmende Holzdreiecke.**



**Die Klemmwinkel an den Seiten müssen auf einer Seite exakt der Höhe der Filter entsprechen. Da wir kein Material in der richtigen Größe vorrätig hatten, mussten wir eine Seite abfräsen.**

weil die Dämpfer keine Vibrationen auf das Gehäuse übertragen. Sägen Sie dafür aus den ausgesägten Lochscheiben kleine Quadrate, die genau auf vier aneinanderstoßende Ecken der Lüfter Druck ausüben. Versehen Sie diese Klemmhölzer mit einem abgesenkten Schraubenloch in der Mitte.

An den Rändern funktioniert diese Konstruktion nicht, da sich der Klemmer auf einer Seite nicht abstützen kann. Abhilfe schaffen Aluwinkel, deren eine Seite genau der Dicke der Lüfter entspricht und deren andere Seite ebenfalls auf die gummierten Ecken drückt. Ihre Bohrung ist knapp hinter der Außenkante, damit die Holzschraube ebenfalls im Zwischenraum zwischen den abgerundeten Ecken Platz findet.

An allen vier Ecken der Platte sägen Sie eine Aussparung, damit Sie jeweils einen 10 Zentimeter langen Aluwinkel als Abstandshalter anschrauben können. Die Befestigungsschraube muss so versenkt sein, dass Winkel und Schraubenkopf nicht über das Außenmaß von 61 Zentimetern hinausstehen. An diesen Winkel können Sie zusätzlich innen ein Holzdreieck ankleben und -schrauben, das die Außenecken der vier Lüfter in den Ecken klemmt.

Bohren Sie nun alle Löcher für die Holzschrauben vor und klemmen Sie beim Verschrauben gleich alle 25 Lüfter fest. Die Kabel sollten auf der Seite herausstehen, wo die Eckwinkel Platz schaffen. Die Lüfter müssen in die gleiche Richtung pusten.

## Holzkiste

Im nächsten Schritt bauen Sie die Holzkiste, in die Sie beide Filter und das Lüfterbrett später einschieben. Ihre Innenmaße entsprechen genau den Außenmaßen des Filters, sodass Sie diesen für den Zusammenbau als Lehre verwenden können. Sägen Sie die Seitenwände möglichst exakt, da sie verleimt und verschraubt einen luftdichten Rahmen bilden müssen.

An der Unterseite lassen Sie vorn 10 Zentimeter Platz für den Lufteinlass. Erst dahinter steht der Vorfilter, dann das Lüfterbrett mit Abstandswinkeln. Daran anschließend bietet das Gehäuse eine Kante aus 2 Zentimeter breiten Holzleisten, an die die Winkel auf der groben Seite anstoßen. Auf der gegenüberliegenden glatten Seite dockt der H14-Filter mit seiner Dichtung an, weshalb der Zwei-Zentimeter-Rahmen allseitig eine luftdichte Verbindung zum Gehäuse haben sollte. Hinter dem H14-Filter bleiben wieder 10 Zentimeter Platz für den Luftauslass oben, weshalb der Deckel dort 10 Zentimeter Freiraum lässt.

Damit dieser Aufbau genau winklig wird, setzen Sie die Platten zuerst auf der linken Seitenwand auf und platzieren danach die Filter und das Lüfterbrett im Inneren. An die Innereien setzen Sie nun Boden und Decke mit einem dünnen Karton als Abstandshalter an und setzen die rechte Seitenwand auf. Auf dieser markieren und bohren Sie die Schraubenlöcher und übertragen die Löcher auf die Stirnseiten von Boden und Decke, um dort passende Löcher vorzubohren. Bohren Sie nur einen Millimeter kleiner als der Außendurchmesser der Schraube, um das Holz mit den Schrauben in der Stirnseite nicht zu sprengen. Da nun alle Innenmaße exakt passen müssen, können Sie die rechte Wand verleimen und verschrauben. Lassen Sie den Leim zwölf Stunden trocknen.

Drehen Sie danach alles vorsichtig um und bohren, verleimen und verschrauben Sie die linke Seite genauso. Auch ihr sollten Sie zwölf Stunden Zeit geben, damit der Leim richtig abbindet.

Für die Zwei-Zentimeter-Leisten im Innenraum sägen Sie sich zuerst vier zehn Zentimeter hohe Stempel, auf die Sie den H14-Filter mit der Dichtung nach oben aufsetzen. Am einfachsten geht das, indem Sie den Gehäuserahmen mit der Vorderseite voran über den Filter stülpen. Bohren Sie danach die Schraubenlöcher in den vier Leisten vor und



**Schiebt man den H14-Filter beim Zusammenbau in den Rahmen, wird dieser automatisch winklig. Um nicht zu fest zu klemmen, empfiehlt es sich, dünne Pappe als Abstandshalter zwischen Filter und Bretter zu stecken.**

legen Sie diese mit der glatten Seite auf die Dichtung des Filters. Drücken Sie dann eine erste Leiste einen Millimeter tiefer in die Dichtung, wobei sie aber perfekt horizontal bleiben sollte. Markieren Sie in dieser Position mit einer Ahle, wo Sie die Schrauben vorbohren. Heben Sie den Rahmen etwas an, decken Sie den Filter zum Schutz ab und bohren Sie die Schraubenlöcher dieser einen Leiste vor. Leimen Sie sie nun an und verschrauben Sie sie gleich.

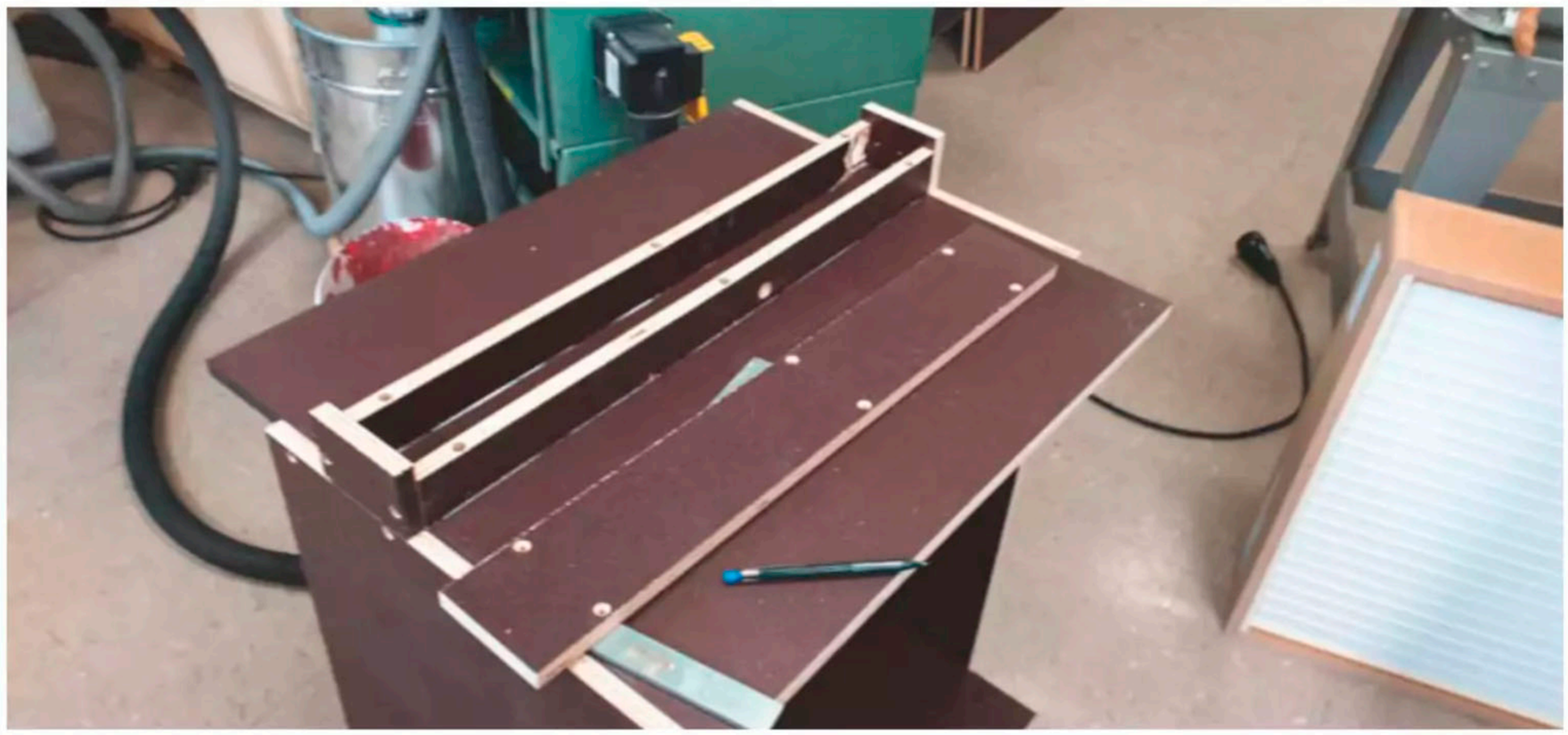
Ohne Trocknungspause können Sie danach an die rechts anschließende Leiste gehen und diese nach dem gleichen Schema positionieren und befestigen. Die nächste Leiste sollte die Gegenüberliegende sein, damit Sie bei der letzten Leiste alle Ungenauigkeiten so ausgleichen können, dass die glatten Seiten der Leisten in allen vier Ecken ohne Höhenversatz aneinanderpassen. Sollte sich beim Verleimen ein Spalt zwischen den Leisten ergeben haben, füllen Sie den mit Heißkleber und schneiden mit einem scharfen Cuttermesser den erkalteten Kleber so ab, dass eine glatte und luftdichte Kante entsteht.

## Elektronikbox

Bohren Sie nun zwei Zentimeter vor der groben Kante der Leisten ein Loch in die rechte Außenwand, durch das die Anschlusskabel samt Stecker passen (falls Sie für eine leichtere Montage Stecker verwenden). Um das Loch bauen Sie anschließend an die rechte Wand einen Kasten in voller Höhe an, in dem Kabel, ESP, Buchse und Poti Platz finden. Dieser Kasten muss luftdicht sein, da hinter den Lüftern im Inneren Überdruck herrscht und sich dieser Druck durch das Kabelloch auch im Elektronikkasten breitmacht.

Schneiden Sie zuerst Deckel und Boden der Box 9 Zentimeter tief und 6 Zentimeter breit. Die Seiten sind 15 Millimeter schmaler, damit die Front noch drauf passt. Diese hat die Maße 61 × 9 Zentimeter. Bohren Sie zuerst die Schraubenlöcher vor und achten Sie bei den Senkungen der Schrauben quer durch die Seiten, dass die Schraubenköpfe nirgends überstehen dürfen. Bohren Sie auch mit einem großen Forstnerbohrer ein Loch in die vordere Seitenwand, in das das Poti hineinpasst. Das Blech fürs Poti muss dieses Loch wieder vollständig verschließen. Gleiches gilt für das etwas kleinere Loch in der hinteren Seitenwand, in das die Buchse für die Stromversorgung gehört. Um alles abzudichten, verschließt Heißkleber jeden noch so kleinen Spalt zwischen Blech und Seitenwand.

Verleimen und verschrauben Sie nun die Seitenwände sowie Boden und Deckel an der Seitenwand



**Die Elektronikbox vor dem Einbau der Platine: Im Loch im vorderen Brett wird das Poti in eine kleine Blechabdeckung geschraubt. Die 5-mm-Buchse fürs Stromkabel findet im Loch in der hinteren Wand Platz.**

des Gehäuses. Durch die Leimverbindung werden die Stoßkanten luftdicht. Der Deckel kommt erst drauf, wenn die Verdrahtung fertig ist, die Schraubenlöcher können Sie aber schon markieren und vorbohren.

## Poti einmessen

Die Elektronik bauen Sie ziemlich flott auf einer Loch- oder Streifenrasterplatine auf. Bauteile und Verbindungen entnehmen Sie unserer Fritzing-Zeichnung oder dem Schaltplan (Download über ct.de/wwkx). Über den Link finden Sie auch die Fritzing-Datei. Falls Sie sich eine Platine fertigen (lassen) möchten, ist sie ein guter Startpunkt.

Gegenüber der Lüftersteuerung aus dem vorhergehenden Artikel Seite 38 hat sich nur die Anbindung des Poti geändert. Das arbeitet jetzt als Spannungsteiler. Wenn Sie ein lineares Poti verwenden, kann dann nämlich eine Linearisierung der Messwerte in der Software wegfallen. Der ESP verträgt am analogen Eingang nicht mehr als 1 Volt, weshalb Sie die Versorgungsspannung am linken Pin des Poti auf 1 Volt begrenzen müssen. Statt wie beim Projekt auf Seite 38 zwei konstante Widerstände zu verbauen, leistet sich die neue Schaltung ein Trimmpoti, das ebenfalls als Spannungsteiler arbeitet. Stellen Sie das zunächst auf etwas unter 1 Volt ein, indem Sie seinen mittleren Pin mit einem Multimeter messen. Drehen Sie dann das große Poti ganz nach rechts und schalten dann erst die Spannung am ESP ein.

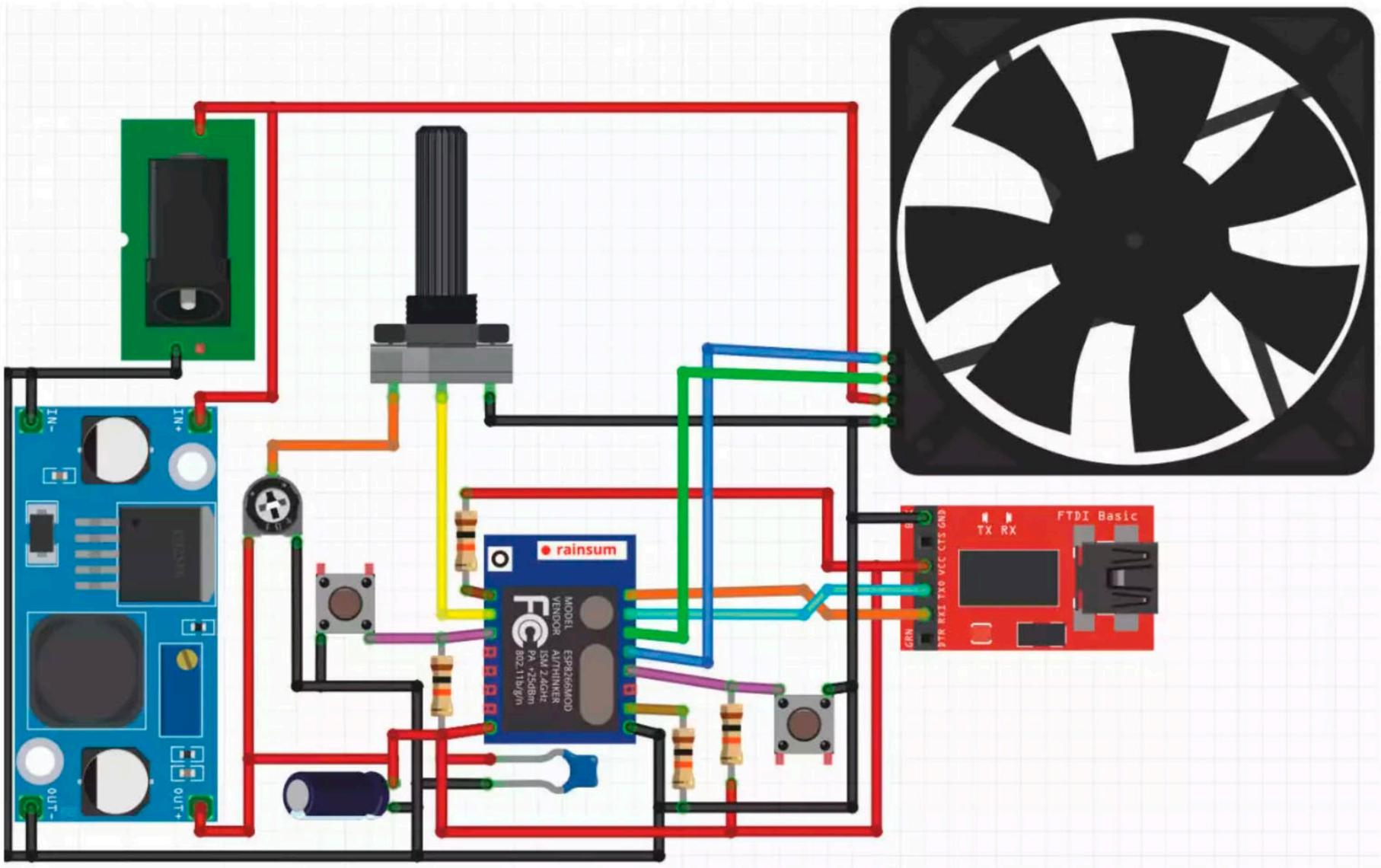
Die Firmware müssen sie nun zweimal flashen, einmal eine leicht modifizierte Version, um das Trimmpoti perfekt einzustellen und dann die normale Version für den Regelbetrieb. Der Code auf GitHub (siehe ct.de/wwkx) geht von Visual Studio Code mit dem Plugin PlatformIO als Entwicklungsumgebung aus. Beides ist kostenlos und für alle gängigen Betriebssysteme verfügbar. Laden Sie zuerst den Code mit Git:

```
git clone https://github.com/pinae/↵
                    ↳WiFi-Poti-Fan.git
```

Öffnen Sie das Projekt dann mit Visual Studio Code und wählen in der linken Leiste `src/main.cpp` aus. Fügen Sie dort hinter `int a = analogRead(A0)` in Zeile 67 Folgendes ein:

```
Serial.printf("Analog in: %d\n", a);
```

Um die Firmware zu flashen, bringen Sie den ESP zunächst in den Programmiermodus, indem Sie den FTDI-Adapter aufstecken, mit einem USB-Port des




**Das ESP12F-Modul verdrahten Sie wie im Datenblatt angegeben und ergänzen lediglich den Anschluss für Poti und Lüfter. Das Poti arbeitet als Spannungsteiler und liefert null bis ein Volt an den analogen Eingang. Statt eines Lüfters können Sie auch 25 anschließen, wenn Sie das Tachosignal einfach nur mit dem ersten Lüfter verbinden.**

Rechners verbinden, den linken Programmiertaster gedrückt halten und den rechten Reset-Taster kurz drücken. Den Programmiertaster können sie danach wieder loslassen und in PlatformIO in der blauen Leiste am unteren Bildschirmrand den Pfeil nach rechts anklicken. Im „Task“-Fenster erscheinen nun Compiler-Meldungen und danach Statusausgaben des Flash-Vorgangs. Sobald der 100 Prozent erreicht hat, klicken Sie “in der blauen Leiste auf das Steckersymbol, was die serielle Konsole öffnet, die die Compiler-Ausgaben verdrängt. Achten Sie nun auf die Ausgaben der eben eingefügten Zeile. Der gelesene Analogwert nimmt maximal Werte bis 1023 an und sollte zunächst noch etwas unter diesem Wert liegen.

Um die Position des Poti möglichst genau auszulesen, drehen Sie nun vorsichtig das Trimmpoti hoch. Da das große Poti währenddessen in Maximalstellung steht, lesen Sie in der seriellen Ausgabe nun ab, wie sich die Spannung an 1 Volt beziehungsweise den Wert 1023 annähert. Drehen Sie keinesfalls zu weit! Sobald in der seriellen Ausgabe 1023 erreicht wird, müssen Sie das Trimmpoti auf dieser Position stehen lassen.

Danach entfernen Sie die eingefügte Zeile wieder und flashen die Firmware erneut. Sie ist nun wieder im Originalzustand und der ESP ist einsatzbereit.

## Kabelbau

Spätestens jetzt müssen Sie die 25 Lüfter parallel verdrahten. Die Kabel sollten den Luftstrom möglichst wenig stören, weshalb Sie sie zwischen den Lüftern verlegen sollten. Kleine auf die Klemmquadrate geschraubte Streifenrasterplatinen helfen, die Kabel sauber zu verlegen, und Verbindungen zwischen den Platinen sind automatisch in den Zwischenräumen.

Löten Sie anschließend ein mindestens ein Meter langes Kabel für die Spannungsversorgung an. Für das PWM-Signal und das Tacho-Signal von einem der Lüfter reicht dünne Litze. Damit die nicht chaotisch rumhängt, schieben Sie alle 10 Zentimeter ein kurzes Stück Schrumpfschlauch über den

Kabelbaum und schrumpfen ihn fest. Das Kabel stecken Sie anschließend durch das Loch in der Seitenwand und verbinden es mit der Platine. Wenn Sie an dieser Stelle Stecker verbauen, machen Sie sich das Leben viel leichter, falls Sie den Luftreiniger aus irgendeinem Grund später noch mal zerlegen müssen.

Die verlötete Platine können Sie nun im Elektronikkasten verschrauben. Schieben Sie anschließend das Lüfterbrett mit den Abstandshaltern voran in den Rahmen. Die Abstandshalter sollten an der rauen Seite des Zwei-Zentimeter-Rahmens anstoßen und vor dem Brett sollte gerade genug Platz für den Vorfilter in seinem Rahmen sein. Ziehen Sie nun das Kabel durch das Loch, bis nur noch das absolute Minimum vor den Lüftern hängt. Die Schlaufe verstauen Sie nun im länglichen Elektronikkasten, der im unteren Bereich bequem Platz dafür bieten sollte. Zuletzt können Sie den Deckel der Elektronikbox aufsetzen. Falls Sie befürchten, dass die Spalten undicht sein könnten, ergänzen Sie vorher innen eine Dichtung. Wenn die Front exakt passt, ist das aber nicht nötig.

## Filterklemmen

Da die Lüfter nun schon am richtigen Platz sitzen, können Sie anschließend die beiden Filter einschieben. Der H14-Filter drückt mit seiner Dichtung gegen die glatte Seite des Zwei-Zentimeter-Rahmens und steht danach etwa einen Millimeter beim Luftauslass über. Die am Rahmen des Vorfilters verklebte Dichtung drückt auf der anderen Seite auf die glatte Seite des Lüfterbretts und steht einen Millimeter beim Lufteinlass über.

Schneiden Sie nun zwei 61 Zentimeter lange und 3 Zentimeter schmale Leisten und vier Klötze mit 5 × 3 Zentimetern. Eine der Leisten schrauben Sie nun (ohne Leim) an der Oberkante des Rahmens im Luftauslass so an, dass sie oben bündig abschließt und unten den Filter einen weiteren Millimeter in den Rahmen schiebt und fixiert. Schrauben Sie nun zwei der Klötzchen so am Boden fest, dass sie die Unterkante des Filters ebenfalls einen Millimeter nach innen schieben und ihn dort fixieren. Beim Vorfilter läuft es genauso, nur umgedreht: Die Leiste klemmt auf der Seite des Lufteinlasses und schließt bündig mit der Bodenplatte ab, die Klötzchen klemmen oben und sind mit der Decke verschraubt.

Da nun alle Innereien festsitzen, können Sie die Rückwand und die Front aufsetzen. Sie werden einfach ohne Leim in die Stirnseiten des Rahmens geschraubt. Falls Sie mal einen Filter oder Lüfter tauschen wollen, lösen Sie diese Platten und die Klemmleisten und -klötzchen wieder.

Schrauben Sie zuletzt die Rollen am Boden des Luftreinigers fest. Sie dienen nicht nur dazu, ihn leichter in der Wohnung bewegen zu können, sie sorgen auch für ausreichend Freiraum unter dem Lufteinlass.

## Nutzung

Dank IotWebConf ist die WLAN-Fähigkeit des Luftreinigers optional. Sobald Sie das Netzteil anschließen, können Sie mit dem Poti die Drehzahl regeln. Zum Ausschalten drehen Sie das Poti ganz nach links, bei Besuch während der Pandemie ganz nach rechts. Im Leerlauf verbraucht der ESP 3,2 Watt, auf voller Leistung zieht das Netzteil sparsame 23,5 Watt aus der Steckdose. Um das Heimkino nicht zu stören, haben Sie außerdem 1022 Drehzahlen dazwischen, von unhörbar bis säuselnd. Sind WLAN und MQTT eingerichtet, können sie ihn außerdem fernsteuern und in die Heimautomation einbinden. Zusammen mit unserem AeVOC-Sensor (siehe S. 56) sind da intelligente Programme möglich.

Der Luftreiniger ist darauf optimiert, im Alltag nicht zu stören. Dieses Versprechen löst er mit maximal 45 dB auch ein. Dass er auf höchster Stufe nur 32 Kubikmeter pro Stunde umwälzt, bedeutet aber auch, dass er in vielen Räumen allein nicht reicht, um Corona-Infektionen über die Luft ganz zu verhindern. Mit regelmäßigem Lüften kann man ihn aber leicht unterstützen. Bewerten Sie also trotz Luftreiniger stets das Gesamtrisiko und ergänzen Sie Ihre Strategie, um sich und andere ausreichend zu schützen.

Optimal positionieren Sie den Luftreiniger so, dass die Atemluft direkt zum Lufteinlass wandern kann und die Aerosole auf dem Weg nicht an einer gefährdeten Person vorbeiziehen. Das bedeutet, dass Sie ihn auch mal in die Mitte des Raums rollen und zum Beispiel als Beistelltisch nutzen sollten.

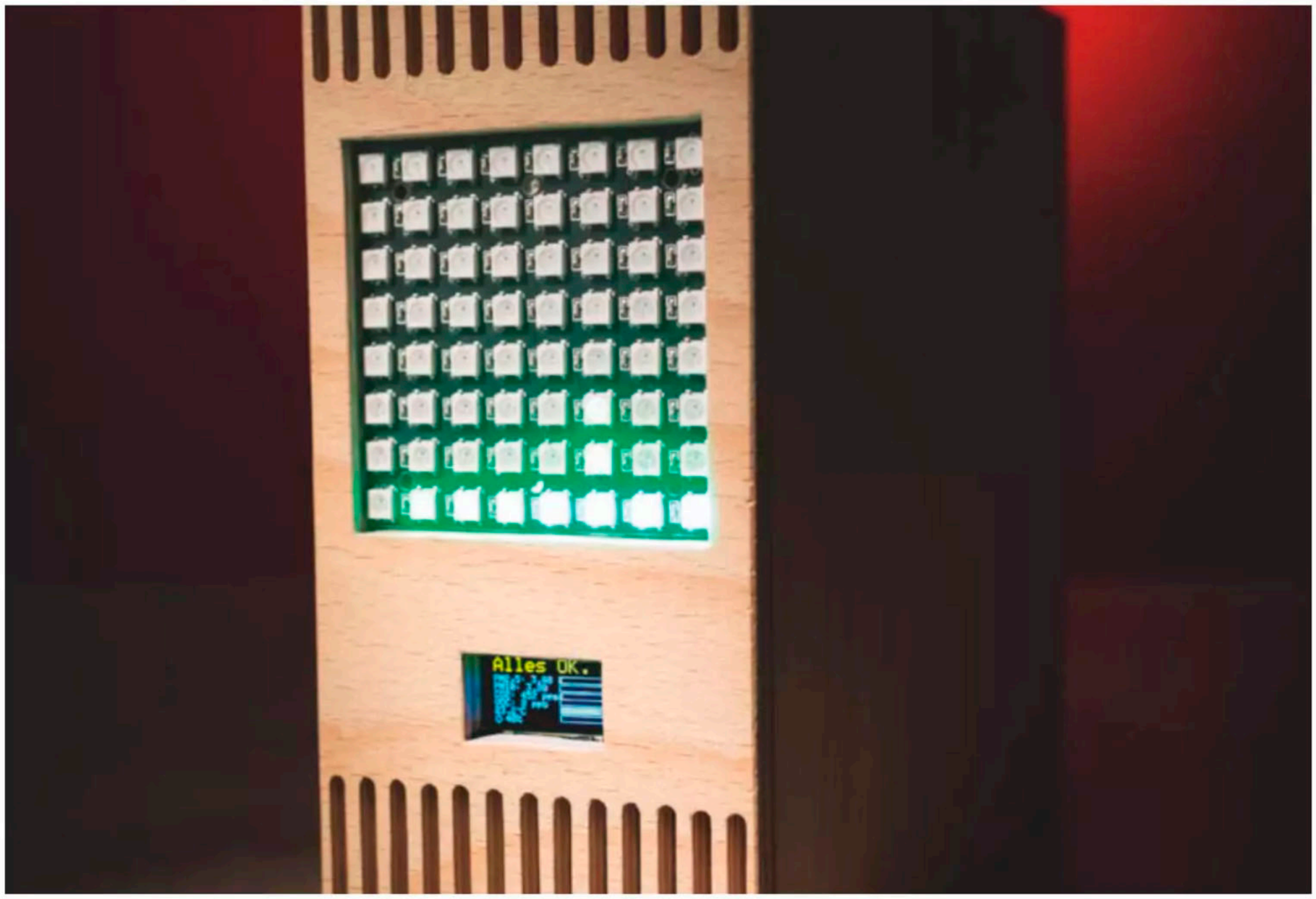
Während der Pandemie reduziert der Luftreiniger das Risiko für Infektionen über Aerosole. Hausstaub-Allergiker können ihn auch darüber hinaus einsetzen, um die Staubmenge im Innenraum zu senken. Nach einigen Stunden Betrieb konnten wir im Test sehr niedrige Feinstaubwerte messen (PM10 und PM2,5 jeweils unter 1,5 ppm). Zu einem Reinraum wird Ihre Wohnung dadurch nicht, aber Sie können die Basiswerte durchaus halbieren, möglicherweise sogar dritteln. (pmk) c't

### Literatur

[1] Merlin Schumacher, Andrijan Möcker, Jan Mahn, c't Smart Home, https://heise.de/-4249476

**CAD-Design, Zeichnungen, Quellcode und Video**

ct.de/wwkx




Aerosolsensor

# AeVOC misst Aerosole & Feinstaub

**Treffen in Corona-Zeiten verpflichten zum häufigen Lüften, da sich das Virus über in der Luft schwebende Tröpfchen, die Aerosole, verbreitet. Im Winter öffnet man die Fenster aber nur ungern. Der WLAN-Sensor AeVOC mit ESP8266 warnt, wenn es höchste Zeit zum Lüften wird.**

Von **Pina Merkert**

Aerosole sind in aller Munde. Leider. Die winzigen Tröpfchen transportieren nämlich auch Coronaviren und sind daher für die meisten Ansteckungen mit Covid-19 in Innenräumen verantwortlich. Häufiges Lüften hilft, die Konzentration der Tröpfchen niedrig zu halten. Leider vergisst man aber schnell, die Fenster aufzureißen und die möglicherweise ansteckenden Tröpfchen reichern sich an. Ein Sensor, der misst und warnt, könnte daran erinnern.

Es gibt aber leider keine Sensoren, die Aerosol-Konzentrationen direkt messen. Deswegen empfehlen Aerosol-Experten oft $CO_2$-Sensoren, um die Raumluft

zu überwachen. In einem Raum mit wenigen Pflanzen sollte nämlich auch die $CO_2$-Konzentration steigen, wenn sich durch Atemluft die Aerosole anreichern. Unsere Schwesterzeitschrift Make hat in Ausgabe 5/20 einen sehr günstigen $CO_2$-Sensor gebaut, der ab 1000 ppm warnt [1]. Das ist die aktuelle Grenzwert-Empfehlung des Umweltbundesamts.

Auch wenn $CO_2$ und Aerosole in den meisten Fällen stark korreliert sind, gibt es dennoch keine Hinweise, dass just 1000 ppm $CO_2$ die magische Grenze darstellen, unter der es keine Corona-Ansteckungen gibt. Die Ergebnisse eines gemeinsamen Forschungspapiers von Aerosol-Experten zeigen, dass sich innerhalb von Minuten eine kritische Menge Tröpfchen anreichern kann [2]. Zu wissen, ob gerade mehr Aerosole als üblich im Raum schweben, würde aber bei aller Unwissenheit um genaue Grenzwerte trotzdem helfen.

## Aerosole im Feinstaubsensor

Kleine Tröpfchen und feiner Staub ähneln sich in einer für Sensoren wichtigen Eigenschaft: Sie brechen und reflektieren Laserlicht. Feinstaubsensoren wie ein SDS011 geben daher traditionell nach einem Regen zu hohe Werte aus. Dieser Effekt lässt sich für Aerosol-Messungen nutzen. Ein SDS011 misst Staub und Tröpfchen in den Größen PM10 (10 µm) und PM2,5 (2,5 µm). Dafür beleuchtet er die Partikel mit einem Laser, von dessen Licht bei klarer Luft nichts auf die eingebaute Fotodiode trifft. Zieht der Lüfter des Sensors aber Staub oder Tröpfchen in die Messkammer, streuen der Staub oder das Wasserkügelchen das Licht so, dass ein Teil die Fotodiode beleuchtet. Das klappt bis zu Schwebeteilchen von 2,5 Mikrometern Größe. Kleinere Staubpartikel und Tröpfchen kann der Sensor nicht in voller Konzentration erkennen.

Leider können auch Tröpfchen, die kleiner als PM2,5 sind, noch Coronaviren enthalten. Vermutlich spielen gerade die Tröpfchengrößen PM1 und PM0,1 eine große Rolle für die Infektionen, da sich so kleine Tröpfchen erst nach vielen Stunden absetzen. Beim Atmen stößt ein Mensch aber gleichzeitig alle möglichen Aerosol-Größen aus. Es ist daher eine sinnvolle Annahme, dass nach dem Betreten eines Raums zunächst die Konzentrationen aller Aerosol-Größen gemeinsam ansteigt. Die größeren Tröpfchen setzen sich dann schneller ab oder verdunsten so, dass sie zu kleineren Tröpfchen werden. Mit der Zeit sinkt also die Konzentration großer Aerosole, während die der ganz kleinen Tröpfchen praktisch gleich bleibt. So lange sollte man momentan aber ohnehin nicht warten. Steigt die mit einem SDS011 messbare Menge großer Aerosole an, sollte man dringend Lüften und mit dem Lüften die kleinen Aerosole gleich auch in den Wind schicken. Ein schneller Abfall der Messwerte zeigt dann, dass das Lüften geklappt hat.

## Olfaktorische Belastung

Aerosole verbreiten sich turbulent in Mustern, die sich nur schwer vorhersagen lassen. Außerdem stoßen Menschen sehr unterschiedliche Aerosol-Mengen aus. Während viele nur 100 Tröpfchen pro Sekunde ausatmen, setzen andere in Extremfällen die 50-fache Menge frei. Im ungünstigsten Fall schlägt der Feinstaubsensor nicht Alarm, obwohl die Menschen im Raum schon so lange zusammensitzen, dass fünf Minuten Stoßlüften stark angeraten wäre.

Abhilfe schafft da ein günstiger VOC-Sensor. „VOC“ steht für „Volatile Organic Compound“, ein Sammelbegriff für verschiedene organische Moleküle, die Menschen ausdünsten oder ausatmen. Ein VOC-Sensor wie der CCS811 misst die Konzentration solcher Moleküle über eine Oberfläche, mit der diese chemisch reagieren. Durch die Reaktion ändert sich der elektrische Widerstand ein kleines bisschen und ein integrierter Mikrocontroller rechnet das in eine Konzentration um. Er berechnet aus dem gleichen Wert auch eine geschätzte $CO_2$-Konzentration, in unserem Test wich diese Projektion aber erheblich von den Werten eines optischen $CO_2$-Sensors ab.

Viele der vom Sensor gemessenen VOCs sind für Gestank verantwortlich. Wer sich vor hohen VOC-Werten warnen lässt, lüftet also nicht nur gegen ein Corona-Risiko, sondern auch für riechbar frische Luft.

## Warnleuchte im Buchformat

Egal ob Aerosole, Feinstaub oder VOC-Gestank: Steigende Werte geben Anlass, das Fenster aufzureißen. Um diese Empfehlung klar darzustellen, haben wir den Feinstaubsensor SDS011 zusammen mit dem VOC-Sensor CCS811, einem Temperatur- und Luftfeuchte-Sensor DHT22 und einer Matrix aus 64 WS2812-RGB-LEDs in ein wörterbuchgroßes Gehäuse gepackt. Die in einem 8×8-Raster angeordneten LEDs leuchten umso heller und röter, je dringender das Lüften wird. Für Zahlenliebhaber zeigt ein winziges OLED-Display darunter auch die Messwerte an. Mit einem ESP8266 als Prozessor versendet das Gerät seine Werte auch per WLAN über das Heimautomatisierungsprotokoll

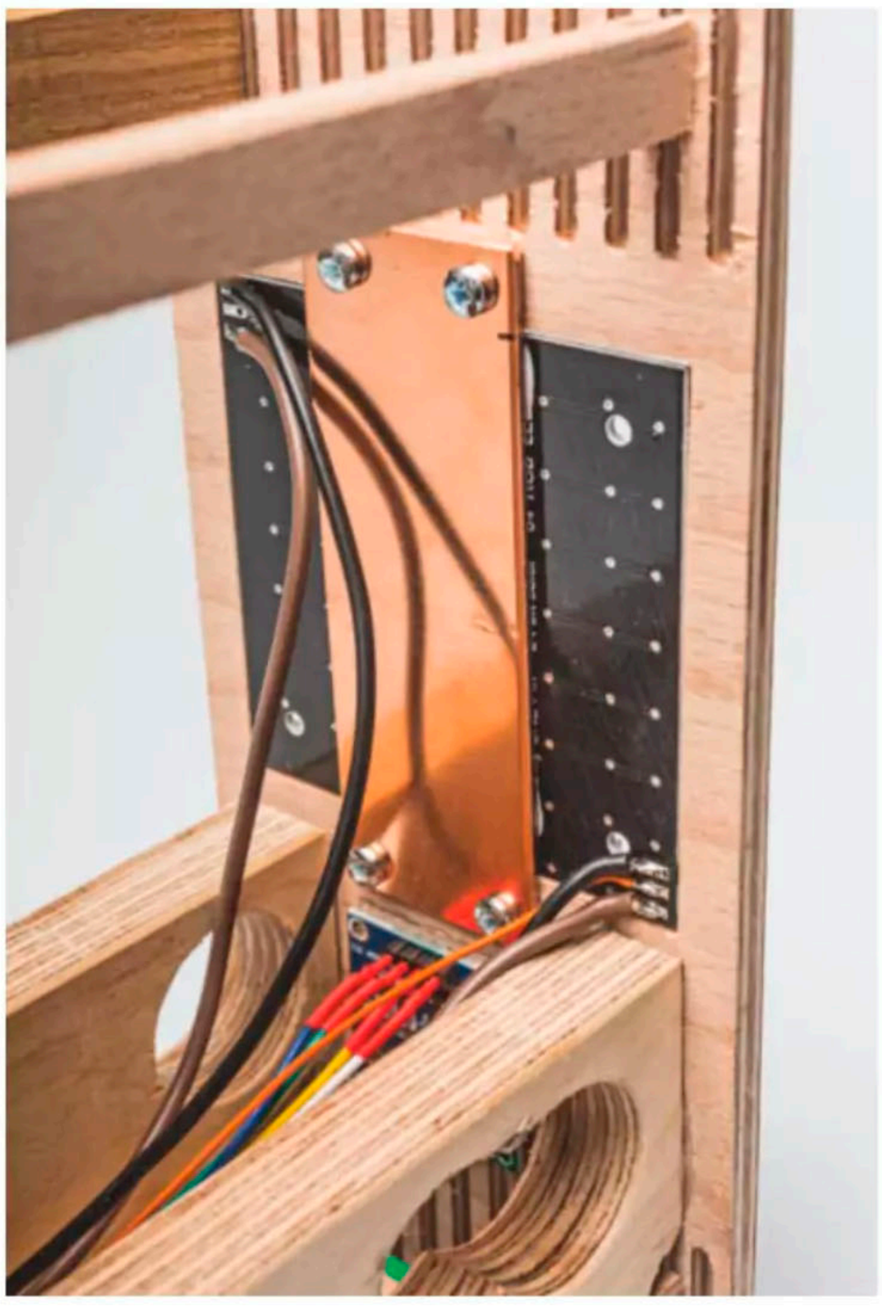
MQTT, damit ein schlaues Zuhause automatisch auf hohe Werte reagieren kann.

Neben der Aufstellung im Bücherregal eignet sich unsere Bastelei auch als Warnleuchte, die Lehrer im Unterricht auf ihr Pult stellen können. Da auch die Schüler die LEDs im Blick haben, vergisst die Klasse das Lüften nicht. Der Nachbau lohnt sich also nicht nur für Heimautomatisierer, sondern auch für Pädagogen und Bastler, die Lehrer im Bekanntenkreis mit Technik gegen die Pandemie ausstatten möchten. Sollte sich eine Elektronik-AG an den Nachbau für die ganze Schule machen, wären wir natürlich begeistert!

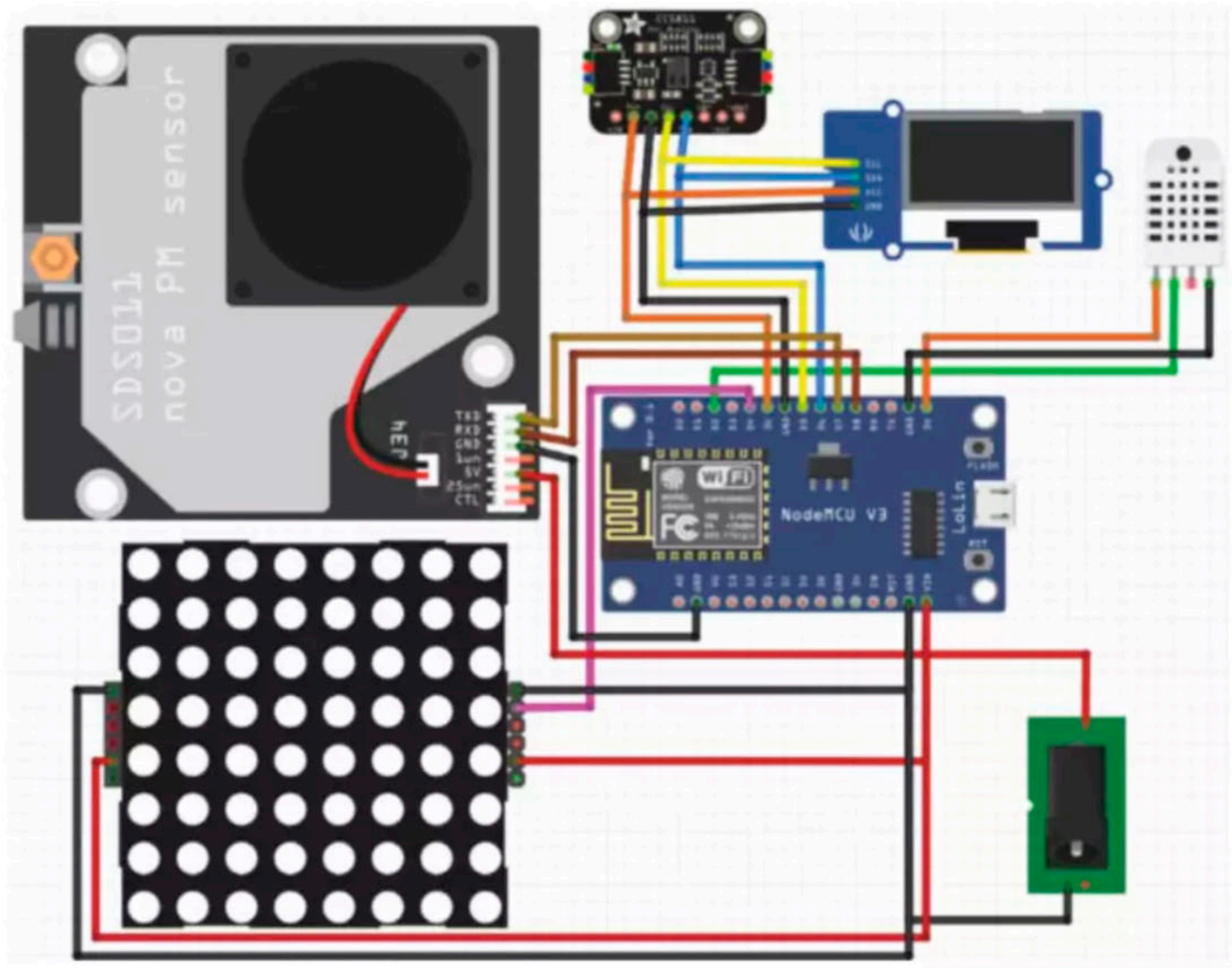
## Holzarbeit

Das Gehäuse besteht aus einem Rahmen, in den eine Inneneinrichtung eingeschoben wird, an der auch Front und Rückwand hängen. Durch diese Konstruktion kommt man auch mit fertigem Gehäuse zur Wartung noch einigermaßen leicht an die Elektronik im Inneren heran. Die Inneneinrichtung teilt das Gehäuse in zwei Bereiche: In einem L-förmigen Abschnitt finden ganz unten der Feinstaubsensor und hinter der Rückwand das kleine OLED-Display und die LEDs Platz. Die LEDs produzieren reichlich Abwärme, die durch Lüftungsschlitze unten und oben in der Front den Weg nach außen findet. Durch die Lüftungsschlitze unten pustet der Feinstaubsensor auch seine Abluft nach draußen. Frischluft bekommt er durch einen knapp 10 Zentimeter langen Gummischlauch durch ein Loch in der Rückwand. Der Sensor hängt quer unter dem Zwischenboden des Innenausbaus, da er laut Datenblatt in dieser Position die verlässlichsten Messwerte liefert. In dem vom L freigelassenen Quader kommen der ESP, der Temperatur- und der VOC-Sensor unter. Die Trennung in zwei Bereiche ist nötig, damit die warmen LEDs nicht die Messwerte des Thermometers verfälschen. Damit die Sensoren im kalten Bereich Raumluft messen, enthält die Rückwand oben und unten je eine Reihe aus fünf Lüftungslöchern. Da die sich nach hinten öffnen, verfälscht die Warmluft vorne am Gehäuse die Messung nicht.

Sie finden ein in Onshape gezeichnetes CAD-Modell des Gehäuses mitsamt millimetergenauer technischer Zeichnungen über ct.de/w867. Das Onshape-Design ist mit einem Browser einsehbar und darf kopiert und verändert werden (mehr zu Onshape siehe Seite 136). Da sich die Lochabstände der Sensor-Platinen von Modell zu Modell leicht unterscheiden, sollten Sie die Holzteile nach unseren Plänen zuerst aussägen, die Befestigungslöcher aber mit Ihren Platinen als Lehre anzeichnen und vorbohren. Nach dem Verleimen der Teile kommen Sie an einige der Befestigungsschrauben nur schwer heran. Daher ergeben sich zwei mögliche Vorgehensweisen: Wenn Sie kleine oder mit Gelenken knickbare Schraubendreher haben, können Sie zuerst leimen und die Elektronik danach verschrauben, ohne das Risiko, sie mit Leim zu versauen. Falls Ihnen die Spezialschraubendreher fehlen, lohnt es sich, die Platinen schon vor dem Verleimen anzuschrauben und die Teile dann besonders vorsichtig zu behandeln. Ob Sie auch gleich die nötigen Kabel anlöten wollen, hängt davon ab, ob Sie Fummelei beim Verlöten oder Kabelsalat beim Verleimen vermeiden wollen. Wir haben zuerst geleimt, entspannt geschliffen, beim Verschrauben geflucht und beim Verlöten nur knapp verbrannte Finger vermieden. Falls Sie mehr Platz im Regal haben, können Sie unsere Onshape-Zeichnung auch kopieren und einige Maße vergrößern, um weniger zu fummeln. Verkleinern sollten Sie das Gehäuse keinesfalls, da wir schon mit dem absoluten Mindestmaß geplant haben.



**Die LED-Matrix produziert ordentlich Abwärme. Um sie abzuführen, haben wir sie mit Wärmeleitkleber auf ein Kupferblech geklebt. Da die Leiterbahnen auf der Platine für die großen Ströme knapp bemessen sind, brauchen die 64 LEDs auch eine 5-V-Einspeisung am Ende der Kette (oben links).**




**Achten Sie bei der Verdrahtung darauf, dass der Feinstaubsensor und die LEDs mit 5 Volt, der VOC- und der Temperatur-Sensor aber mit 3,3 Volt versorgt werden. Der nötige Spannungswandler sitzt auf der Platine des NodeMCU.**

## Verschaltung

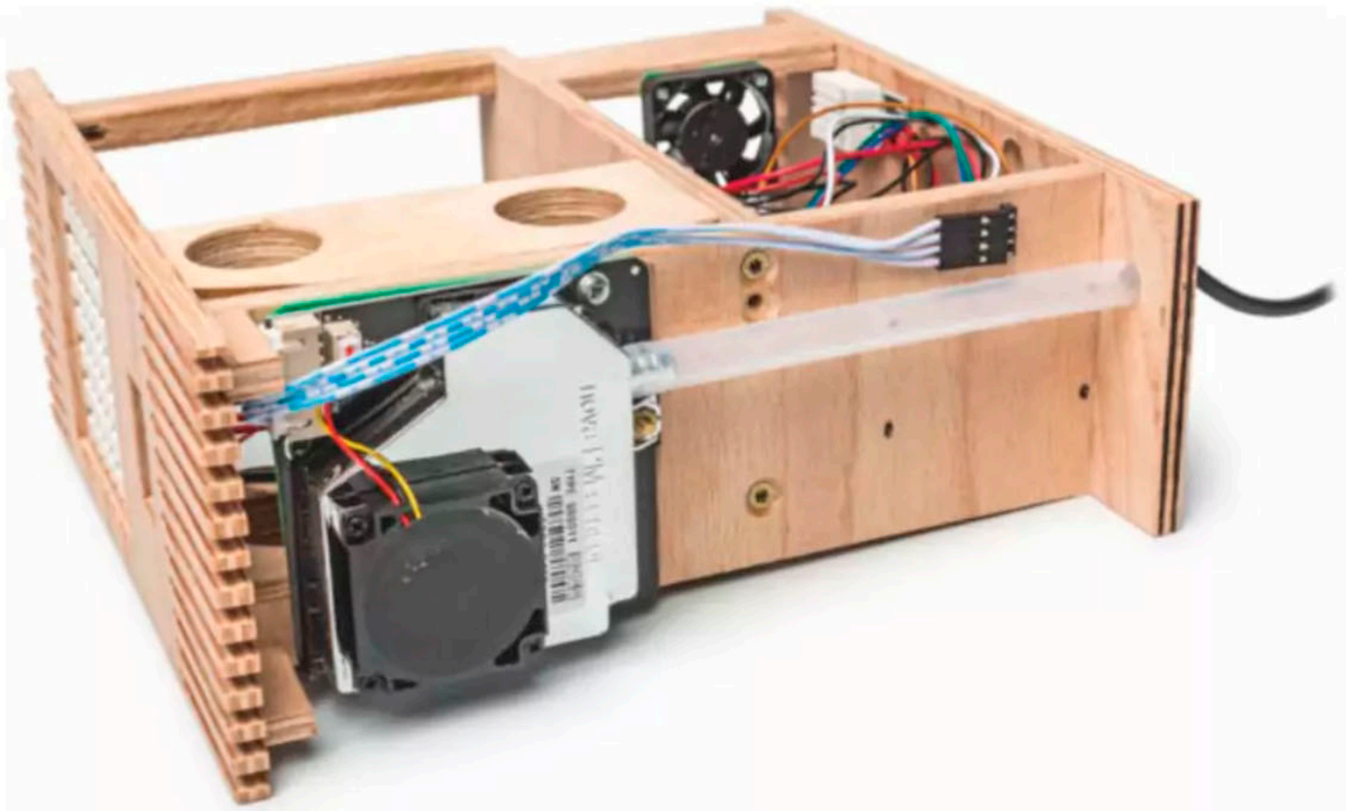
Schließen Sie zuerst die Matrix mit 64 WS2812-LEDs an. Die Platine hat auf der Rückseite zweimal drei Lötpads. Das mit „In“ beschriftete mittlere Lötpad findet in der linken unteren Ecke Platz. Löten Sie dort ein dünnes Datenkabel für die serielle Ansteuerung an. An „+“ und „-“ löten sie sowohl unten links als auch oben rechts dickere Litze. Sie wird später direkt mit den 5V vom 4A-Netzteil verbunden. Bei lediglich 5V Versorgungsspannung ziehen die LEDs bei voller Helligkeit so viel Strom, dass der Spannungsabfall über den Leiterbahnen auf der Platine so groß ist, dass sich die Farben der letzten LEDs sichtbar von den ersten LEDs in der Reihe unterscheiden. Die zusätzliche Einspeisung am Ende der LED-Kette verhindert diesen Effekt.

Als Nächstes führen Sie vier Kabel vom OLED-Display in den kalten Bereich zum VOC-Sensor. Beide Platinen hängen am I2C-Bus und brauchen neben Masse und 3,3 Volt noch eine Datenleitung („sda“) und eine fürs Zeitsignal („scl“). Display und VOC-Sensor bringen meist fertig verlötete Pinleisten mit, auf die Sie Jumperkabel aufstecken können. Beim Sensor brauchen Sie aber einen Abzweig, den Sie sich aus einer winzigen Streifenrasterplatine bauen können. Die Mini-Platine sollte allerdings direkt auf den Pins des Sensors stecken. Mit einem abzweigenden Kabel hatten wir bei einem Testaufbau zufällige Fehler, die wahrscheinlich auf Laufzeitprobleme zurückzuführen sind. Für den fertigen Sensor wollten wir nicht darauf vertrauen, dass die Stecker nicht herausrutschen und verlöteten die Kabel direkt. Das funktionierte sofort und ohne Fehler.

Führen Sie nun vier Kabel vom Feinstaubsensor in den hinteren Raum. Der Sensor wird mit 5 Volt versorgt, weshalb Sie auch eine kleine Streifenrasterplatine an die Unterseite des Zwischenbretts schrauben können, die die Netzteilspannung an LEDs und SDS011 verteilt. Hinter dem Sensor ist noch etwas Platz, wo nur der Luftschlauch verläuft. Dort kommen Sie leicht an die Verteiler-Platine heran.

Befestigen Sie nun den DHT22 Temperatur- und Luftfeuchtesensor. Er bekommt 3,3 Volt vom ESP. Seinen Datenausgang verbinden Sie mit D2 am NodeMCU. Ob Sie Masse am NodeMCU oder direkt am Netzteil verbinden, spielt keine Rolle. Alle GND-Pins am NodeMCU sind miteinander verbunden.

Das Gehäuse hat einen ovalen Ausschnitt, um später den ESP flashen zu können, ohne es zu öffnen. Die Höhe dieses Ausschnitts passt genau, wenn Sie den NodeMCU ohne zusätzliche Platine direkt auf seine Pinleisten stellen. Falls Sie ihn in eine Platine stecken wollen, müssen Sie den Ausschnitt entsprechend etwas höher platzieren. Wir haben ihn



**Im unteren Teil des Gehäuses ist neben dem Schlauch noch Platz für eine Platine, die 5 Volt an den SDS011 und die LED-Matrix verteilt.**

ohne Platine verschraubt, was aber nur funktioniert, wenn man die Kabel direkt an die Pins lötet.

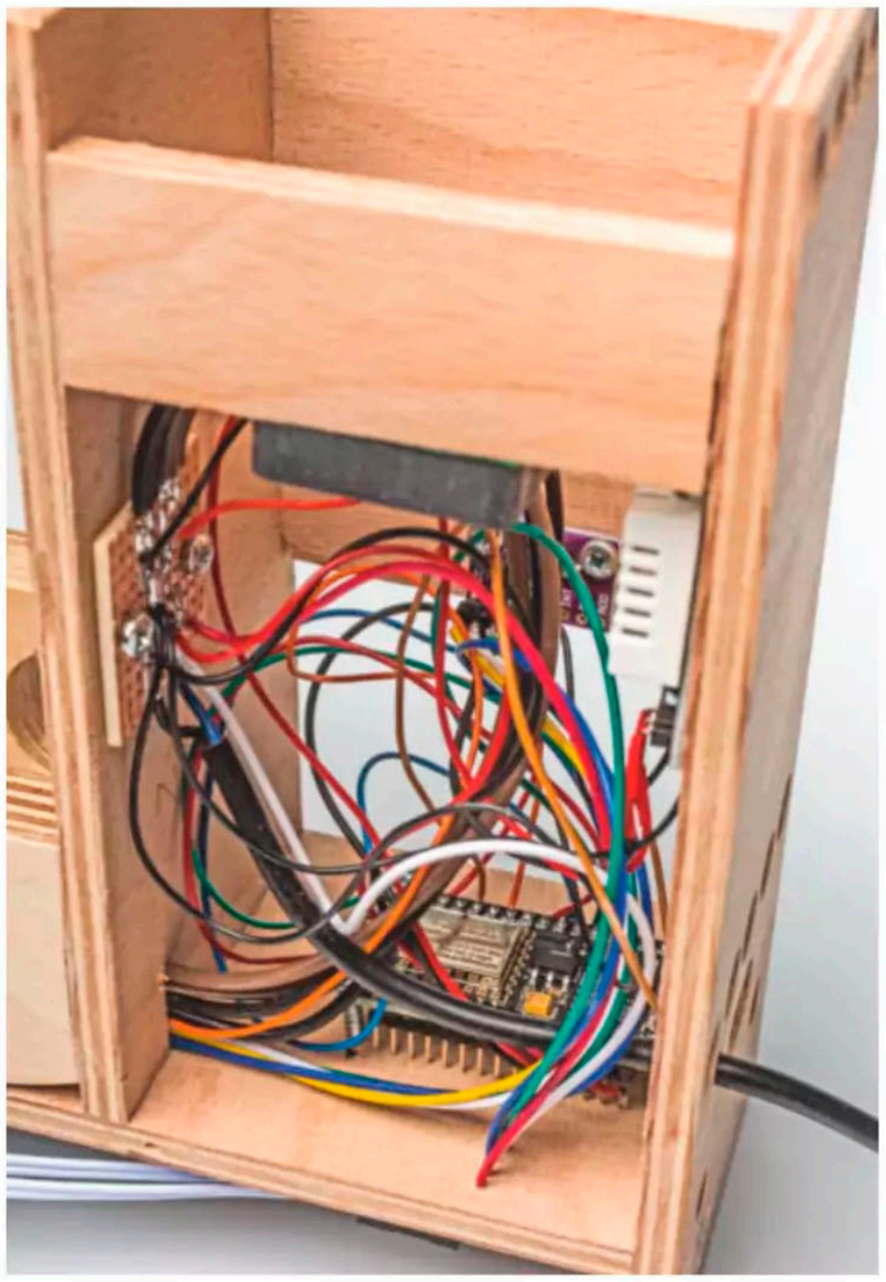
Verbinden Sie D2 mit dem Datenausgang des DHT22 und D4 mit dem seriellen Eingang der LED-Matrix. D5 gibt das Zeitsignal (scl) des I2C-Bus aus, D6 die Daten. Von diesen Pins gehen Kabel zum VOC-Sensor und von dort weiter zum Display. Praktischerweise führt der NodeMCU an den beiden benachbarten Pins GND und 3,3V nach außen, was Sie ebenfalls zum VOC-Sensor führen. An Pin D7 erwartet der ESP ein serielles Eingangssignal (Rx) für den Feinstaubsensor. Der sendet auf diesem Kabel, weshalb die Verbindung beim Sensor an Tx endet. Mit D8 ist es umgekehrt: Rx am Sensor führt hier zum sendenden Pin D8. Die 5V vom Netzteil verbinden Sie mit „Vin“ am NodeMCU. Der Pin dafür liegt auf der anderen Seite in der Ecke. Daneben gibt es einen GND-Pin, den Sie für die Masseverbindung zum Netzteil verwenden können.

Wenn Sie dieser Anleitung in der beschriebenen Reihenfolge folgen, haben Sie einen etwas unübersichtlichen bunten Kabelsalat oberhalb des NodeMCUs. Da das Gehäuse hier recht viel Platz bietet, stört das aber nicht. Falls Sie während des Verdrahtens zweifeln, können Sie die korrekten Verbindungen einfach in unserer Fritzing-Zeichnung nachschauen. Damit die übersichtlich bleibt, haben wir alle Hinweise auf die Gegebenheiten des Gehäuses entfernt. Sie müssen unser Farbschema nicht übernehmen, aber wir raten zu verschiedenfarbigen Litzen, damit Sie leichter die Übersicht behalten, wo ein Kabel herkommt.

## Firmware

Wie üblich haben wir die Firmware mit PlatformIO programmiert. Sie steht als freie Software auf GitHub (siehe ct.de/w867). Um sie zu flashen, klonen Sie das Repository und öffnen den Ordner mit PlatformIO. PlatformIO bietet als Erweiterung für Visual Studio Code einen wesentlich komfortableren Editor als die Arduino-IDE. Board und Bibliotheken müssen Sie nicht händisch über einen Bordverwalter installieren, die von uns vorbereitete `platformio.ini` konfiguriert das Projekt schlüsselfertig. Zum Hochladen reicht daher der kleine Pfeil nach rechts, den Sie unten in der bei der Installation von PlatformIO hinzugefügten blauen Leiste finden.

Ohne Änderung am Code spannt der ESP ein WLAN namens „AEVOC-“ gefolgt von der MAC-Adresse des ESP auf. Diesen Namen verwendet das Gerät auch bei der Kommunikation über MQTT zu Beginn aller Topics. Um sich mit diesem WLAN zu verbinden, geben Sie das Passwort „loving_ct“ an. Falls Sie das ändern möchten, müssen Sie nur in den String in Zeile 43 von `main.cpp` etwas anderes reinschreiben. Falls Sie für den Verbindungsaufbau ein Mobilgerät verwenden, leitet das Sie nach einigen Sekunden auf eine Webseite um. Smartphones erwarten hier die Login-Seite eines öffentlichen WLANs, unser Gerät leitet aber nur zu seiner Konfigurationsseite weiter (siehe Seite 64). Auf einem Notebook erreichen Sie die Seite über `http://192.168.4.1/config`. Wenn Sie den Pfad weglassen, kommen Sie auf eine Landingpage mit einem Link zur Konfigurationsseite.



**Im hinteren Bereich steht der NodeMCU einfach auf seinen Pins. Zu ihm führen alle Datenleitungen, die, wenn man ein paar Zentimeter zum bequemen Anlöten lässt, einen unübersichtlichen Kabelsalat bilden. Farbige Litze erleichtert die Übersicht.**

Die präsentiert sich schmucklos mit einer Liste an Eingabefeldern. Der Cursor landet zuerst im Eingabefeld für ein Passwort, mit dem Sie auch nach der Ersteinrichtung an die Konfiguration kommen (dann über die IP-Adresse des Geräts in Ihrem Heimnetz). Danach geben Sie die SSID Ihres WLANs an. Die sollten Sie auswendig kennen, da die Firmware keine Vorschläge macht. Ins Feld darunter gehört das WLAN-Passwort. Danach folgt der MQTT-Server. Haben Sie beispielsweise ct-Smart-Home [4] auf einem Raspi eingerichtet, geben Sie hier einfach dessen IP-Adresse an. Den Port 1883 ändern Sie nur, wenn Ihr MQTT-Server von der Standardkonfiguration abweicht. Dahinter können Sie noch einen Benutzernamen und das zugehörige Passwort für den MQTT-Server angeben, falls Ihr Server Authentifizierung nutzt. Wenn Sie die nicht konfiguriert haben, lassen Sie die Felder einfach leer.

Nach einem Klick auf „Speichern“ startet der ESP neu, verbindet sich mit dem WLAN im Heimnetz und feuert fortan sekündlich seine Messwerte an den MQTT-Server. Die Topics bestehen aus dem Namen des Geräts, den Sie schon als SSID gesehen hatten, gefolgt von `/temperature`, `/humidity`, `/voc` und `/particles_and_aerosoles`. Über die ersten beiden Topics versendet die Firmware Strings mit einer Zahl, bei den letzten beiden kommt je ein JSON-Objekt mit zwei Werten beim MQTT-Server an.

Über MQTT können Sie auch die Helligkeit der LEDs einstellen. Schicken Sie dafür einfach eine ganze Zahl zwischen 0 und 255 als String an `<AEVOC-Name>/setMatrixBrightness`.

Das Gerät funktioniert auch ganz ohne WLAN und MQTT. Es hält dann zwar ein eigenes WLAN für die Konfiguration offen, die Messungen starten aber wenige Sekunden, nachdem der ESP Strom bekommt. Das Display zeigt dann sofort Messwerte an und auch die LED-Matrix erlaubt einzuschätzen, wie dringend sofortiges Stoßlüften ist. Möglich macht das die Bibliothek „IotWebConf“ (siehe Seite 64). Sie ersetzt den verbreiteteren „WiFiManager“, blockiert im Gegensatz zu dem aber nicht, wenn WLAN und MQTT noch nicht konfiguriert sind.

## Einbrennzeit

Der VOC-Sensor liefert erst nach einer „Einbrennzeit“ von mehreren Tagen verlässliche Werte. Ist er noch fabrikneu, muss seine Sensorfläche erst mal oxidieren, was zunächst zu falschen Werten führt. Außerdem kalibriert er sich selbst mit einer Baseline. Wenn Sie den ESP per USB mit einem Rechner verbinden und eine serielle Konsole starten (beispielsweise mit dem Stecker-Symbol in PlatformIO) sehen Sie in den Debug-Ausgaben den aktuellen Wert der Baseline. Die Baseline wird bei jedem Neustart zurückgesetzt. Beim Stoßlüften verändert sich die Baseline stark, bleibt dann aber bestehen. Wir haben daher unsere Grenzwerte so eingestellt, dass der Sensor erst dann sinnvoll warnt, wenn er einmal richtige Frischluft gemessen hat.

Beim Feinstaubsensor ist eine solche Kalibrierung nicht nötig. Trägt man das Gerät aber von einem Raum in den nächsten, braucht er wegen seines begrenzt starken Lüfters einige Minuten, bis die Messwerte wirklich zum Raum passen. In unseren Tests lieferten die Sensoren die verlässlichsten Werte, nachdem wir das Gerät für ein paar Stunden am gleichen Ort stehen ließen.

## Messwerte interpretieren

In unserem Test gingen die Feinstaub-Messwerte merklich hoch, wenn wir im Raum Sport trieben und

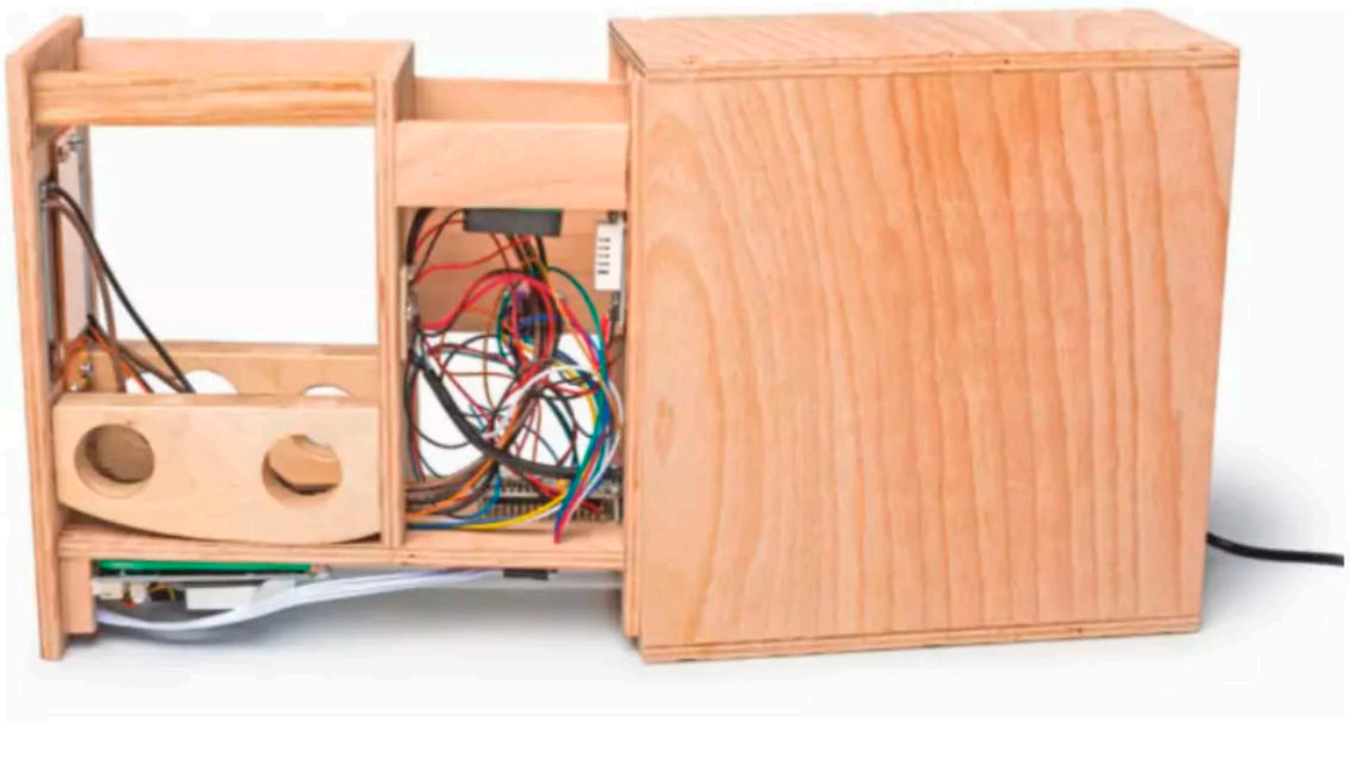
außer Atem kamen. Typisch stieg PM10 schneller als PM2,5. Rissen wir dann die Fenster auf, schossen sie teilweise aber weiter in die Höhe, obwohl die frische Außenluft bereits die möglicherweise virenbelastete Luft verdrängt hatte. Das passiert nicht nur an nebligen Wintertagen, sondern schon bei moderater Luftfeuchtigkeit vor den Fenstern. Die sinkenden VOC-Werte geben in dieser Situation aber einen klaren Hinweis, dass das Lüften sein Ziel erreicht.

Schließt man die Fenster wieder, wärmt sich die Luft schnell wieder auf und kann dann auch mehr Feuchtigkeit aufnehmen. Die Aerosole verdunsten dann im Rekordtempo, weshalb die Feinstaubwerte in 10 bis 15 Minuten wieder auf niedrige Werte sinken. Typisch sinkt PM10 dann schneller als PM2,5.

Andert sich der Aerosol-Trend wieder zu steigenden Werten, ist es dann auch schon wieder Zeit zum Lüften. Da die Menschen im Raum neue Aerosole ausatmen, während noch Reste der Aerosole von draußen verdunsten, sinken die Werte in dieser Situation nie auf ein wirklich niedriges Niveau. Außerdem sollte man beachten, dass die Verdunstung möglicherweise sehr hohe Konzentrationen der großen Aerosole (PM10 und PM2,5) verhindert. Entwarnung kann solch ein mittlerer Messwert also nicht geben, da es dann vermutlich noch viele kleinere Aerosole (PM1 und PM0,1) gibt, die leider auch infektiöse Viren enthalten können. Wenn unser Sensor nur gelb leuchtet, kann es durchaus bereits Zeit zum Stoßlüften sein, während feuchte Außenluft auch mal für einen rot leuchtenden Sensor sorgt, ohne dass ein Infektionsrisiko besteht. Der in der LED-Matrix angezeigte zeitliche Verlauf erlaubt hier eine Einschätzung.

Leider müssen Sie trotz Sensor anhand der Situation einschätzen, wann Sie lüften müssen. Treffen sich mehrere Menschen im Raum, sollten Sie lieber schon bei relativ geringen Werten lüften. Unser Gerät kann den Blick auf die Uhr nicht ersetzen. Ein zusätzlicher $CO_2$-Sensor kann auch helfen, zu einer realistischen Einschätzung zu kommen. $CO_2$ verdunstet nicht und reichert sich im Raum mit der Zeit an. Außerdem senkt Stoßlüften verlässlich die $CO_2$-Konzentration, auch an nebligen Tagen.

Ein Feinstaubsensor misst dagegen tatsächlich Aerosole und verlässt sich nicht auf eine Korrelation zwischen $CO_2$ und Viruslast in der Luft. Die Messwerte für PM2,5 und PM10 können helfen, den Zustand der Raumluft genauer einzuschätzen und schneller auf Aerosol-Anstiege zu reagieren. VOCs ergänzen das Bild, da sie in einem Wohnraum ohne Haustiere vollständig von den Menschen im Raum stammen.



## Weiter optimieren

Mit unserem Sensor wollen wir eine Basis schaffen, damit Sie selbst Ihre Raumluft im Blick behalten können. Unsere 64 LED-Warnleuchten werden momentan einfach rot, wenn einer der beiden Feinstaub-Messwerte erhöhte Werte annimmt. Da beide Messwerte zur Verfügung stehen, bestünde aber durchaus die Möglichkeit, schlauer auf die Entwicklung der Werte zu reagieren. Beispielsweise steigt meist zuerst PM10 und PM2,5 zieht verzögert nach. Beim Lüften passen sich die beiden Werte dagegen gleich schnell an die Aerosolkonzentrationen der Außenluft an. Unsere freie Firmware können Sie leicht forken und um eine intelligente Auswertung ergänzen. Wir freuen uns auch über Pull-Requests, falls Ihre Verbesserung allen anderen Lesern zugutekommen kann.

Relativ sicher müssen Sie unsere voreingestellten Grenzwerte anpassen. Sie finden die Minimal- und Extremwerte in `configHelpers.h` in den Zeilen 20 bis 31. Je nach Region sollten Sie die Feinstaub-Minimalwerte zum bei offenem Fenster gemessenen „Basisdreck“ einstellen. Beim Maximum haben wir für Sie Sport getrieben und die nach 15 Minuten erreichten Werte als Extreme eingetragen. Diese hemdsärmelige Kalibrierung nutzten wir nur, weil es vom Umweltbundesamt anders als beim $CO_2$ keinerlei Hinweise gibt, wie hoch ein sinnvolles Aerosol-Maximum ist. Es können aber jederzeit neue Forschungsergebnisse zu Aerosolen veröffentlicht werden, mit denen Sie zu besseren Grenzwerten kommen. Scheuen Sie also nicht, die Werte an neue Erkenntnisse anzupassen. (pmk) c't

### Literatur


[1] Guido Burger, Richard Fix und Klaus-Uwe Gollmer, **Der CO2-Warner für die Schule**, Make 5/2020, S. 10

[2] Lidia Morawska und Donald K. Milton, It Is Time to Address Airborne Transmission of Coronavirus Disease 2019 (COVID-19): https://heise.de/s/BKWE

[3] Song Tang et al., Aerosol transmission of SARS-CoV-2? Evidence, prevention and control: https://heise.de/s/grej

[4] Merlin Schumacher, Jan Mahn, Andrijan Möcker, c't Smart Home: https://heise.de/-4249476


**Quellcode, CAD-Design, Schaltplan**

ct.de/w867

# Der enterJS Vue Day 2021

## Einstieg und Deep Dive in Vue.js

**15. Juni 2021**

**10 % Mai-Rabatt für c't-Leser – Ihr Code: 2021VUEJS4CT_KA07F**

**www.enterjs.de/vueday.php**

iX Magazin für professionelle Informationstechnik

heise Developer

dpunkt.verlag

in Kooperation mit vuejs.de

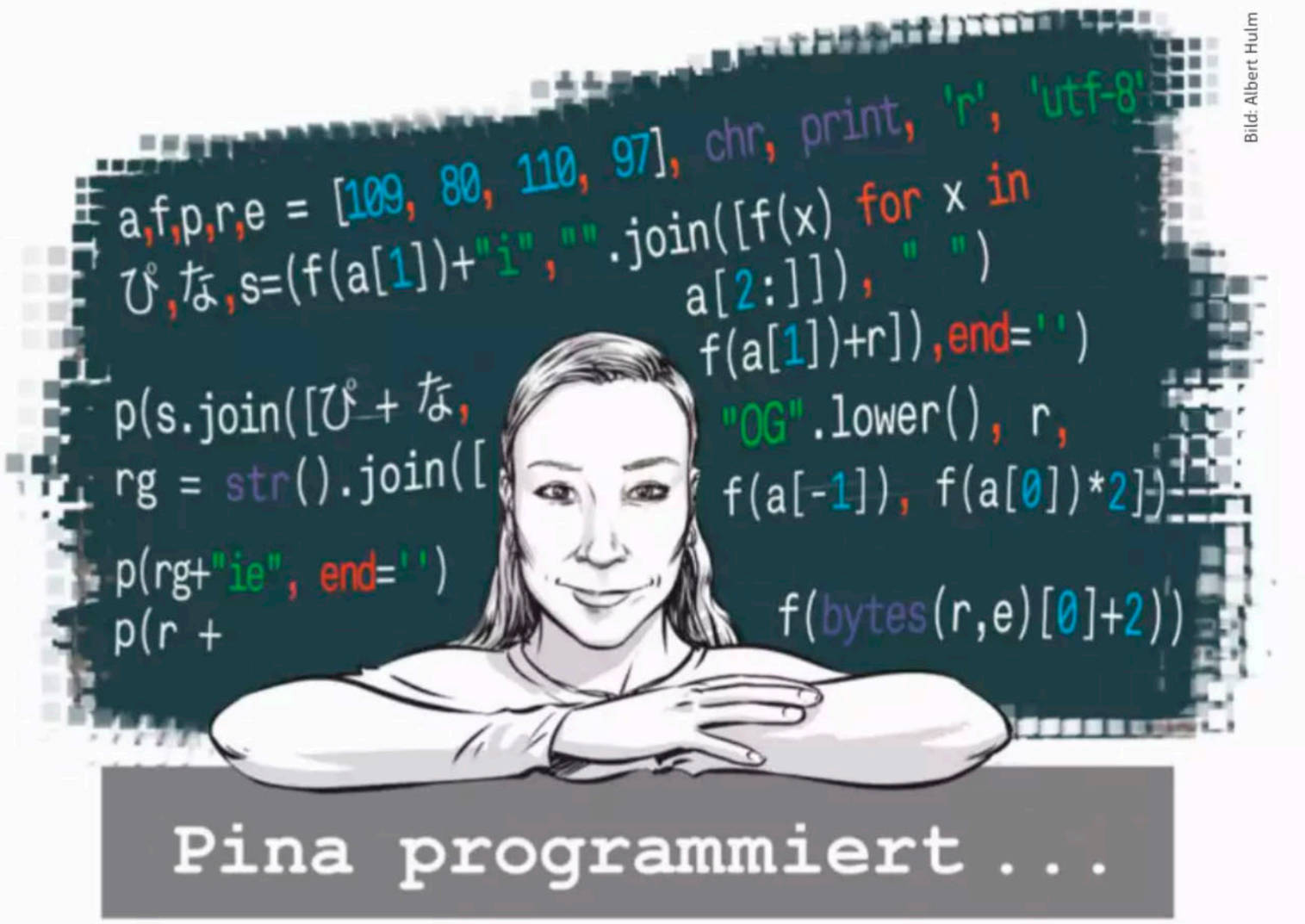



Bild: Albert Hulm

# ESP-Projekte mit und ohne WLAN

**Selbstgebastelte Smart-Home-Geräte mit dem ESP8266 nehmen meist mit dem WifiManager Kontakt zum heimischen WLAN auf. Ist das nicht erreichbar, hängt die Bibliothek im Konfigurationsmodus fest. Verwaltet stattdessen IotWebConf die WLAN-V erbindung des Mikrocontrollers, erledigt der auch dann noch sinnvolle Aufgaben, wenn das WLAN mal nicht erreichbar ist.**

Von **Pina Merkert**

Der Mikrocontroller ESP8266 kann ohne zusätzliche Bauteile Verbindungen zu einem WLAN aufbauen. Der günstige Chip eignet sich damit ideal für selbst entwickelte Smart-Home-Geräte. Die sollen für die Konfiguration der WLAN-Zugangsdaten ein eigenes Netz aufspannen und mit einem Webserver die Daten über ein browserbasiertes Formular abfragen und alle Konfigurationsdaten im Flash des Mikrocontrollers ablegen. Die meisten Projekte nutzen dafür die Bibliothek

WifiManager, die Webserver, Formular und Verbindungsaufbau nach der Konfiguration automatisch organisiert.

Kennt der WifiManager noch keine WLAN-Zugangsdaten, wartet die Bibliothek in `setup()` darauf. `loop()` führt der ESP dann erst aus, wenn das WLAN konfiguriert ist. Was für viele Smart-Home-Basteleien ein sinnvolles Vorgehen ist, blockiert allerdings Geräte, deren WLAN-Kommunikation nur ein Zusatzfeature ist. Der Aerosol- und VOC-Sensor AeVOC (siehe Seite 56) ist so ein Gerät: Er warnt über farbige LEDs vor belasteter Luft und zeigt seine Messwerte auf einem kleinen OLED-Display an. VOCs sind flüchtige organische Moleküle (Volatile Organic Compounds).

LEDs, Display und Sensoren funktionieren auch ohne WLAN-Verbindung. Besteht die, kann die Sensorkiste zusätzlich ihre Daten über das Heimautomationsprotokoll MQTT versenden, wodurch das Smart Home automatisch lüften kann. Trägt man das Gerät aus dem Empfangsbereich des heimischen WLANs heraus, soll es aber nicht auf Daten warten, sondern auch offline messen, anzeigen und warnen.

Der Entwickler Balázs Kelemen baute vermutlich mal ein ähnliches Gerät. Er schrieb nämlich mit „IotWebConf“ eine vom WifiManager inspirierte Bibliothek, die jedoch nicht blockiert, wenn noch keine WLAN-Zugangsdaten da sind. Sie lässt sich ähnlich leicht wie der WifiManager nutzen. Die Firmware für AeVOC nutzt die Bibliothek und eignet sich als Beispiel, wie man IotWebConf in eigene Projekte für den ESP8266 einbaut.

## Flugs initialisiert

Für die Installation mit der Entwicklungsumgebung PlatformIO reicht es, in der Datei `platformio.ini` unter `lib_deps=` die Zeile `prampec/IotWebConf` hinzuzufügen. Wer sich der Bequemlichkeit von PlatformIO verweigern möchte, findet den Code auch im gleichnamigen Repository auf GitHub.

Im C++-Code importiert man die Bibliothek mit

```
#include <IotWebConf.h>
```

Danach empfiehlt es sich, ein globales Objekt vom Typ `IotWebConf` anzulegen. Das will im Konstruktor einen Namen, Referenzen zu einem DNS-Server und einem Webserver-Objekt, ein Passwort für das Konfigurations-WLAN als String und einen Versionsstring für das Datenformat der gespeicherten Parameter mitgegeben bekommen. Die folgenden Zeilen erledigen das:

```
#define CONFIG_VERSION "v0.2"
const char apPass[] = "loving_ct";
DNSServer dnsServer;
WebServer server(80);
IotWebConf iotWebConf("ESP-Name", &dnsServer,
            &server, apPass, CONFIG_VERSION);
```

## Parameter hinzufügen

IotWebConf fragt automatisch zusätzliche Parameter wie die Adresse des MQTT-Servers ab und speichert die eingegebenen Werte auch, anders als der WifiManager, selbst. Dafür reicht es, die Parameter zunächst als Objekte anzulegen und mit üblichen Standarwerten zu belegen:

```
#define STRING_LEN 128
char mqttServerValue[STRING_LEN];
char mqttPortValue[6] = {
      '1', '8', '8', '3', '\0', '\0'};
char mqttUserNameValue[STRING_LEN];
char mqttUserPasswordValue[STRING_LEN];
IotWebConfParameter mqttServerParam=
  IotWebConfParameter("MQTT server",
  "mqttServer", mqttServerValue, STRING_LEN);
IotWebConfParameter mqttPortParam=
  IotWebConfParameter("MQTT port", "mqttPort",
  mqttPortValue, 6, "text", "1883", "1883");
IotWebConfParameter mqttUserNameParam=
  IotWebConfParameter("MQTT user",
  "mqttUser", mqttUserNameValue, STRING_LEN);
IotWebConfParameter
  mqttUserPasswordParam=
  IotWebConfParameter("MQTT password", "mqttPass",
  mqttUserPasswordValue, STRING_LEN, "password");
```

In `setup()` übergibt der Code anschließend die Referenzen an `addParameter()`:

```
iotWebConf.addParameter(&mqttServerParam);
iotWebConf.addParameter(&mqttPortParam);
iotWebConf.addParameter(&mqttUserNameParam);
iotWebConf.addParameter(&mqttUserPasswordParam);
```

## MQTT nicht vergessen!

Da IotWebConf nicht blockiert, kommt die Bibliothek bei jedem Durchlauf der `loop()` kurz zum Zug. Damit sie weiß, was dann alles noch zu tun ist, legt man am einfachsten Boolean-Variablen an, die daran erinnern, ob ein Reset oder ein Verbindungsaufbau zum MQTT-Server nötig ist:

```
boolean needMqttConnect = false;
boolean needReset = false;
```

Ein Reset ist nötig, wenn neue WLAN-Daten abgespeichert wurden. Dann startet IotWebConf, ohne ein eigenes Netzwerk aufzuspannen. Es hat nämlich dann WLAN-Zugangsdaten und kann sich mit dem Heimnetz verbinden. Sobald das passiert ist, wird es Zeit, sich beim MQTT-Server zu melden. In C++ sieht das folgendermaßen aus:

```
void wifiConnected() {
  needMqttConnect = true;
}
void configSaved() {
  needReset = true;
}
```

Diese beiden Funktionen übergibt man `iotWebConf`, um auf die beiden Ereignisse zu reagieren:

```
iotWebConf.setConfigSavedCallback(
  &configSaved);
iotWebConf.setWifiConnectionCallback(
  &wifiConnected);
```

Mit dem Aufruf von `iotWebConf.init()` nimmt die Bibliothek ihre Arbeit auf. Die Funktion gibt einen Boolean zurück, ob für alle zuvor definierten Parameter sinnvolle Werte vorhanden sind. Deswegen ist es wichtig, bei Änderungen an den Parametern eine neue Versionsnummer zu vergeben. Sonst könnte eine unvollständige Konfiguration zu seltsamem Verhalten führen. Ist keine valide Konfiguration vorhanden, sollte man in `setup()` die zu den Parametern gehörenden Variablen löschen:

```
boolean validConfig = iotWebConf.init();
if (!validConfig) {
  mqttServerValue[0] = '\0';
  mqttUserNameValue[0] = '\0';
  mqttUserPasswordValue[0] = '\0';
}
```

## Webserver zum Austoben

IotWebConf nutzt den in `WebServer server(80);` initalisierten Webserver, um seine Konfigurationsseiten auszuspielen. Für die Konfiguration und eine 404-Seite bringt die Bibliothek Funktionen mit. Das Gestalten der Landingpage überlässt sie allerdings der Programmiererin. Eine Minimalkonfiguration prüft auf Anfragen an das „CaptivePortal“, also ob beispielsweise ein Handy über das Konfigurationsnetzwerk ins Internet will und liefert ansonsten eine HTML-Seite mit einem Link zur Konfiguration aus:

```
void handleRoot() {
  if (iotWebConf.handleCaptivePortal()) {return;}
  String s = "<!DOCTYPE html><html lang=\"en\">"
  "<head><meta name=\"viewport\" "
  "content=\"width=device-width, initial-scale=1,"
  "user-scalable=no\"/><title>"
  "IotWebConf 01 Minimal</title></head>"
  "<body>Hello world! Go to <a href='config'>"
  "configure page</a> to change settings."
  "</body></html>\n";
  server.send(200, "text/html", s);
}
```

Diese und die vorbereiteten Funktionen teilen die folgenden Zeilen danach nur noch zusammen mit dem Pfad, dem Webserver mit:

```
server.on("/", handleRoot);
server.on("/config", []{
          iotWebConf.handleConfig(); });
server.onNotFound([](){
          iotWebConf.handleNotFound(); });
```

## Geschleift statt blockiert

Damit IotWebConf nicht in `setup()` blockiert, legt die Bibliothek in `init()` nur ihre Datenstrukturen an. Die eigentliche Arbeit passiert in `loop()`, weshalb man die Funktion `doLoop()` auch bei jedem Schleifendurchlauf aufrufen muss:

```
iotWebConf.doLoop();
```

Dieser Funktionsaufruf übernimmt den Löwenanteil der Arbeit. Um Neustarts kümmert sich die Bibliothek allerdings nicht. Zusätzlich fallen meist noch ergänzende Aufgaben wie der Verbindungsaufbau zum MQTT-Server an (bei uns gesammelt in der Funktion `mqttSetup(mqttSubscribe)`, siehe Seite 38). Ob Neustart oder MQTT-Verbindungsaufbau anstehen, regeln die zuvor gesetzten Boolean-Variablen:

```
if (needMqttConnect) {
  mqttSetup(mqttSubscribe);
  needMqttConnect = false;
}
```

## Klasse Namen

Wer mehrere gleichartige Geräte im Smart Home betreibt, will deren Namen nicht jedes Mal beim Flashen anpassen müssen. Integrieren die Geräte in ihren Namen automatisch die MAC-Adresse, benahmen die Geräte sich damit einzigartig. Damit sich so ein Name leicht in die Initialisierung von Objekten als globale Variablen einbinden lässt, hilft eine kleine Klasse. Ihre Struktur definieren Sie in einer Header-Datei mit dem Namen `configHelpers.h`:

```
#ifndef DEV_NAME_CLASS
#define DEV_NAME_CLASS
class DeviceName {
  public:
    DeviceName();
    char* get();
  private:
    char deviceName[23] = {'A', 'E',
          'V', 'O', 'C', '-',
           0, 0, 0, 0, 0, 0, 0, 0, 0,
           0, 0, 0, 0, 0, 0, 0, 0};
};
#endif
```

Die zugehörige C++-Datei `configHelpers.cpp` definiert den Konstruktor und die `get()`-Funktion:

```
#include <ESP8266WiFi.h>
DeviceName::DeviceName() {
  byte mac[6]; WiFi.macAddress(mac);
  char macStr[18];
  sprintf(&macStr[0], "%x:%x:%x:%x:%x:%x",
          mac[0], mac[1], mac[2],
          mac[3], mac[4], mac[5]);
  strncpy(&deviceName[6], macStr, 18);
}
char* DeviceName::get() {
  return &deviceName[0];
}
```

Initialisiert man so ein `DeviceName`-Objekt mit der Zeile `DeviceName devName;`, kann man danach ganz bequem `devName.get()` überall dort verwenden, wo sonst der statische Name stünde.

```
if (needReset) {
  Serial.println("Rebooting after 1 second.");
  iotWebConf.delay(1000);
  ESP.restart();
}
```

## Blockadeloser Sensor

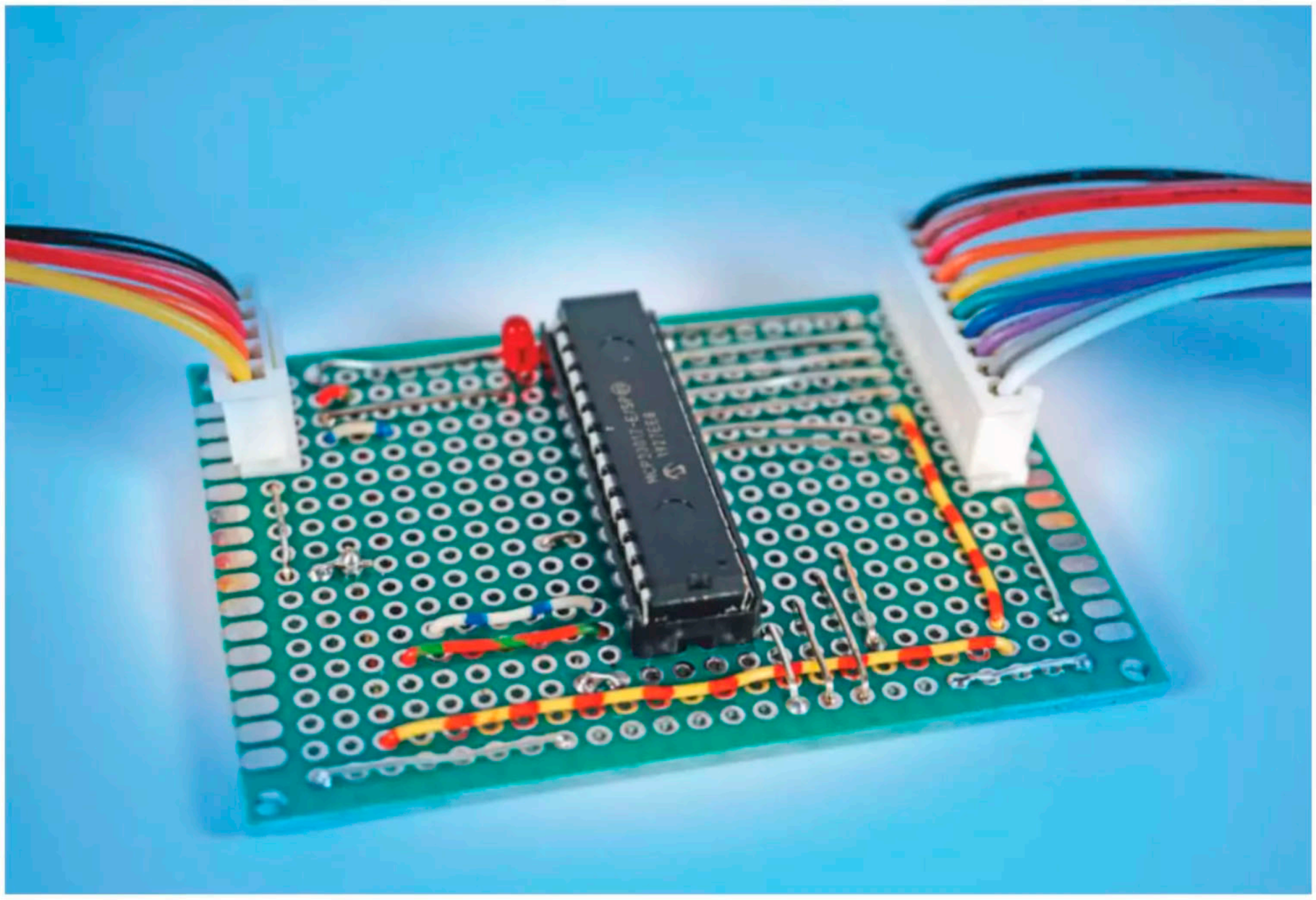
IotWebConf vermeidet, dass ein ESP beim Abfragen der WLAN-Daten blockiert. Für Geräte, die sowohl für sich als auch eingebunden in ein WLAN funktionieren sollen, ist das eine gute Lösung. Der Preis dafür ist, dass die `loop()`-Funktion regelmäßig in die Bibliothek springen muss. `delay()` ist in `loop()` daher fast verboten und der Code muss stattdessen mit `millis()` prüfen, ob genug Zeit für den nächsten Schritt vergangen ist. Ein Beispiel, wie so etwas aussieht:

```
unsigned long now = millis();
iotWebConf.doLoop();
loopWifiChecks();
if ((now - lastMeasurement) > 1000) {
  measure();
  lastMeasurement = now;
}
```

Der Code des Aerosol- und VOC-Sensors eignet sich als realistisches Beispiel und steht unter GPLv3 auf GitHub. Das Repository zur Bibliothek enthält aber auch einige kürzere Beispiele mit wachsender Komplexität. Leider enthält das Repository sonst nur wenig Dokumentation. Wer einem Open-Source-Projekt ohne programmieren unter die Arme greifen möchte, könnte sich hier mit dem Schreiben von Doku einen Orden verdienen. (pmk) c't

**Repositories bei GitHub**

ct.de/wgjk



# Mehr GPIO-Pins am Mikrocontroller

**Bei Mikrocontroller-Elektronik mit vielen Schaltern, LEDs und Relais reichen die integrierten GPIO-Pins am Arduino, ESP8266, ESP32 oder Raspi oft nicht aus. Das Problem lösen per I2C-Bus angeschlossene GPIO-Extender-Chips wie der MCP23017. Aus zwei Pins macht man so 16 zusätzliche digitale Ein- oder Ausgänge.**

Von **Pina Merkert**

Auf dem Basteltisch breiten sich die Teile fürs neueste Projekt aus: fünf Relais, eine LED, zwei Schalter und ein Temperatursensor. Kurz durchgezählt: Die Bauteile belegen 10 GPIO-Pins am Mikrocontroller. Der geplante ESP-12f hat aber nur acht nutzbare Pins. Was tun?

Eine Lösung sind GPIO-Extender wie der MCP23017, den man für weniger als einen Euro bekommt. Den Chip steuert der Mikrocontroller über den seriellen I2C-Bus an und der Baustein kümmert sich seinerseits ums Auslesen der Schalterstellung und das Schalten der Relais. Die seriellen Befehle

kosten ein paar Millisekunden Zeit, dafür liefert der Chip satte 16 konfigurierbare digitale Ein- oder Ausgabe-Pins.

Einen I2C-Bus können alle gängigen Bastelplatinen bereitstellen: Die verschiedenen Arduinos reservieren dafür fest Pin 4 (Daten) und 5 (Zeitsignal). ESP8266 und ESP32 können jeweils zwei beliebige ihrer GPIO-Pins für I2C verwenden und arbeiten bei Bedarf sogar als Master für mehrere I2C-Busse. Beim Raspi muss man I2C über `raspi-config` aktivieren und findet das Zeitsignal anschließend an GPIO 5 und die Daten an GPIO 3.

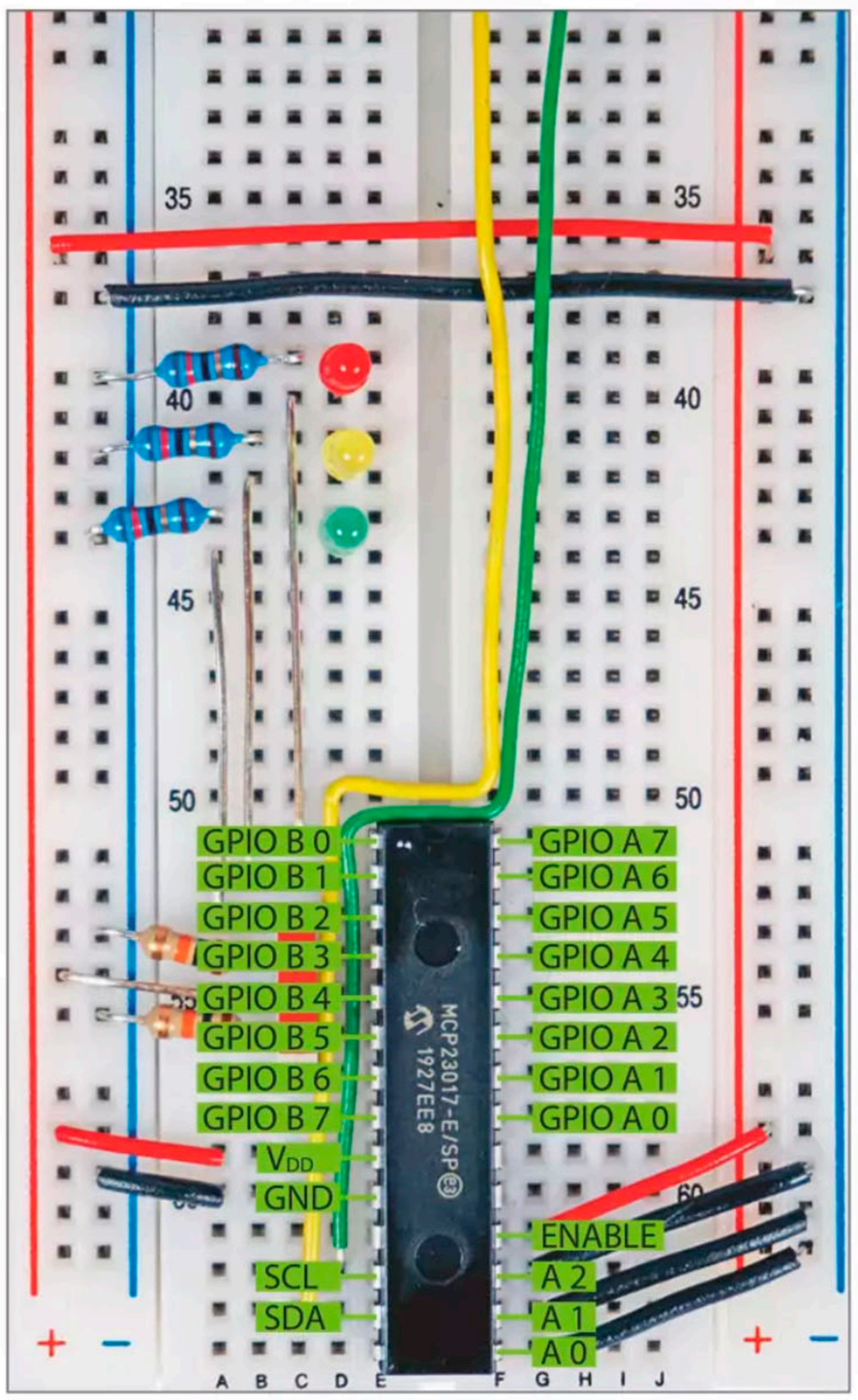
## Hausnummer gelötet

Auf einem I2C-Bus kommuniziert immer ein Master mit einem oder mehreren Slaves. Der Mikrocontroller braucht als Master keine Adresse, aber jeder Slave muss eine eindeutige 1 Byte lange Adresse haben. Das kleinste Bit der Adresse gibt per Konvention an, ob der I2C-Slave vom Bus Daten annimmt (0) oder schreibt (1). In der Praxis nutzen die Chips oft nur gerade Adressen, weil der Slave meist nur Daten annimmt.

Der Master schreibt entweder die Adresse, gefolgt von den Daten für diesen Slave auf den Bus, oder er liest eine bestimmte Zahl von Bytes. An die erwähnte Konvention hält sich der MCP23017 nicht: Er benutzt für Lesebefehle die gleiche gerade Adresse. Damit Slaves wissen, welche Daten der Master lesen will, bestehen Lesebefehle meist aus einer kurzen Schreibsequenz, gefolgt vom Lesen einer festen Zahl an Bytes.

Von der 8 Bit langen Adresse, unter der ein MCP23017 antwortet, sind nur 3 Bit konfigurierbar: Die höchstwertigen vier Bit sind fest auf 0100 konfiguriert. Die drei folgenden Bits legt man beim Verlöten des Chips fest: Pin 15 bis 17 verbindet man einfach mit der Spannungsversorgung für eine 1 und mit GND für eine 0. Das kleinste Bit ist wieder fest auf 0 konfiguriert. Damit lässt sich der MCP23017 auf alle geraden Adressen zwischen 64 und 78 löt-konfigurieren.

Verbindet man beispielsweise die Pins 15, 16 und 17 (die drei Pins unten rechts) mit GND, ergibt das binär die Adresse 0100 000 0, was dezimal 64 entspricht. Verbindet man beispielsweise den zweituntersten Pin 16 stattdessen mit der Spannungsversorgung, hört der Chip auf die binäre Adresse 0100 010 0, also dezimal 68.




**Der Beispielaufbau auf dem Breadboard zeigt, wie man den MCP23017 verdrahtet. Pin 1 ist oben links in der Ecke und gehört zum ersten GPIO von Bank B. Der I2C-Bus wird unten links angeschlossen (gelber Draht: Daten, grüner Draht: Takt).**

## I/O konfigurieren

Damit der Chip weiß, welche Pins als Ein- und welche als Ausgang arbeiten sollen, muss der Master diese Konfiguration vor allen Schreib- oder Lesebefehlen versenden.

Die GPIO-Pins des Bausteins sind in zwei Bänke zu je 8 Pins aufgeteilt. Bei der Bauform für PDIP-Sockel mit 28 Beinen befindet sich Bank A auf der rechten und B auf der linken Seite, sodass GPIO-B-0 auf Pin 1 liegt.

Um die Pins von Bank A zu konfigurieren, genügt eine Nachricht, deren erstes Byte `'\x00'` ist. Danach folgt ein Byte, das alle 8 Pins der Bank in einem Rutsch konfiguriert. Ist Bit 0 eine 0, konfiguriert der Chip GPIO-A-0 zu einem Ausgang. Eine 1 erzeugt einen Eingang. Bit 1 konfiguriert GPIO-A-1 und so weiter. Bank B konfiguriert man genauso, verschickt als erstes Byte aber `'\x01'`.

## Lesen und Schreiben

Ähnlich funktioniert das Lesen: Der Master sendet zuerst das Kommando, beim Lesen von Bank A das einzelne Byte `'\x12'` an den Chip und liest anschließend ein einzelnes Byte als Antwort. Dieses Antwort-Byte enthält eine 1 für die Bits, deren Pins aus Bank A mit Spannung versorgt werden. Ist ein Pin als Ausgang konfiguriert, steht am zugehörigen Bit der am Ausgang geschriebene Wert.

Man kann dem Chip also mit Lesebefehlen auf die Finger schauen, ob er die Schreibbefehle umgesetzt hat. Sind beispielsweise zwei Schalter an GPIO-A-3 und an GPIO-A-6 eingeschaltet, liefert die Leseoperation 01001000 binär beziehungsweise 72 dezimal. Zum Lesen von Bank B sendet der Master als Kommando `'\x13'`.

Zum Schreiben muss der Master das Kommando `'\x14'` für Bank A oder `'\x15'` für Bank B senden. Danach folgt ein Byte, also 8 Bits für die 8 Pins. Die Byte-Folge `'\x15\x28'` setzt also GPIO-B-3 und GPIO-B-5 unter Spannung und zieht alle anderen als Ausgang konfigurierten Pins auf der linken Seite auf Masse (GND). Ein versehentlich gesetztes Bit für einen nicht als Ausgang konfigurierten Pin ignoriert der MCP23017.

## Die Wire-Bibliothek

In praktischem C-Code für den Arduino und den ESP sind die Bytes zum Steuern des MCP23017 dank der I2C-Bibliothek `Wire.h` schnell und einfach versendet:

```
#include <Wire.h>
```

Besonders leicht geht es am Arduino, der einfach fest Pin 4 und 5 für I2C verwendet. Dort genügt in der `setup()`-Funktion ein

```
Wire.begin();
```

Beim ESP geht es nur minimal komplizierter: Er verwendet zwei beliebige Pins, deren Nummern man dem Funktionsaufruf mitgeben muss:

```
#define I2C_SCL_PIN 4
#define I2C_SDA_PIN 5
Wire.begin(I2C_SDA_PIN, I2C_SCL_PIN);
```

So initialisiert, konfigurieren beispielsweise die folgenden Zeilen Pin 0 bis 6 als Ausgabe und Pin 7 als Eingabe:

```
#define GPIO_E_ADDR '\x20'
Wire.beginTransmission(GPIO_E_ADDR);
Wire.write('\x00');
Wire.write((unsigned char)0b10000000);
Wire.endTransmission();
```

## Bequemlichkeit

Statt mit Bitmasken zu hantieren, setzt die folgende Funktion die Ausgabe-Pins ähnlich wie `digitalWrite()` aus der Arduino-Bibliothek:

```
unsigned char i2cOutput[2] = {'\x00', '\x00'};
void i2cDigitalWrite(
                unsigned short pinNo, bool value) {
  Wire.beginTransmission(GPIO_E_ADDR);
  unsigned short bank;
  if (pinNo < 8) {
    Wire.write('\x14');
    bank = 0;
  } else {
    Wire.write('\x15');
    bank = 1;
  }
  unsigned short pinNo8 = (pinNo & 7);
  unsigned char pm = 1 << pinNo8;
  if (value) {
    i2cOutput[bank] |= pm;
  } else {
    i2cOutput[bank] &= ('\xff' ^ pm);
  }
  Wire.write(i2cOutput[bank]);
  Wire.endTransmission();
}
```

Damit ein Aufruf die zuvor gesetzten Pegel für andere Pins auf der gleichen Bank nicht umsetzt, speichert die Funktion in `i2cOutput` zuerst die gesetzten Bits und sendet danach den Schreibbefehl für die ganze Bank.

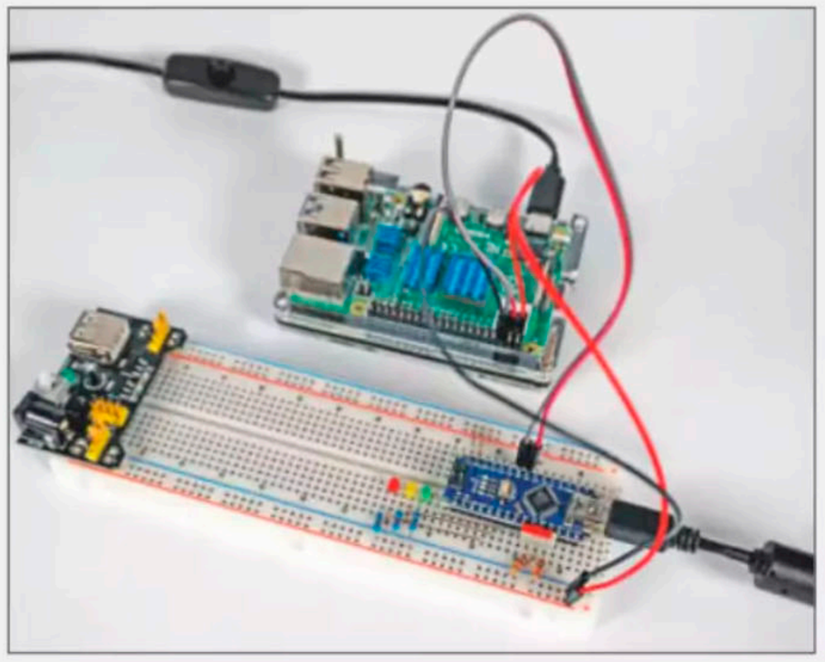
Per bitweisem Und mit der Maske 7 (binär 00000111) erzeugt die Funktion zunächst eine Pin-Nummer innerhalb der Bank (`pinNo8`), also aus dem Bereich 0 bis 7. Die Variable `pm` enthält in der nächsten Zeile per bitweisem Schieben nach links eine Bitmaske mit einer 1 an der durch `pinNo8` definierten Stelle. Um der gespeicherten Bank eine 1 an der richtigen Stelle hinzuzufügen, reicht dann ein bitweises Oder (`|`) mit dieser Maske. Für eine 0 muss diese Maske per XOR (`'\xff' ^ pm`) invertiert und per bitweisem Und (`&`) mit der gespeicherten Bank verrechnet werden.

## Arduino Nano statt MCP23017

Falls Sie keinen MCP23017 zur Hand haben, aber noch Arduino Nanos in der Schublade einer würdigen Verwendung harren, können Sie den Arduino mit unserer Firmware (siehe ct.de/wbg9) auch als GPIO-Extender programmieren. Der kleine Arduino bringt nur 12 statt 16 GPIO-Pins mit, sodass bei Bank B in der Firmware nur die kleineren 4 Bit belegt sind. Abgesehen von dieser Einschränkung verhält sich der Nano mit unserer Firmware aber genauso wie ein MCP23017.

Als kleinen Bonus können Sie die Adresse in der Firmware beliebig festlegen. Außerdem hat der Nano zusätzlich 6 Analog-digital-Wandler (eigentlich 8, aber die Pins A4 und A5 sind durch I2C belegt), die Sie ebenfalls per I2C auslesen können: Der Master schickt dafür ein Byte zwischen `'\x20'` und `'\x27'` (Pin A0 bis A8), um festzulegen, welchen der 8 analogen Werte er lesen möchte. `'\x24'` und `'\x25'` sind tabu, da der I2C-Bus die Pins belegt. Der Arduino antwortet dann mit dem Messwert in Form von 4 Bytes, die zusammen ein `unsigned int` mit 32 Bit ergeben. Das höchstwertige Byte überträgt die Firmware zuerst, die Bytefolge ist also Little-Endian.

In unserem GitHub-Repository liegt ein komplett konfiguriertes Projekt für PlatformIO. Im Prinzip reicht es, die Adresse zu ändern, den Arduino Nano per USB anzuschließen und den Hochladen-Pfeil zu drücken. Sie können unser Projekt aber auch gern auf GitHub forken und die Daten im Arduino vorverarbeiten. Behandeln Sie den kleinen Atmel einfach als zusätzlichen Prozessorkern.



**Verbindet man den Raspi mit dem Arduino Nano über I2C, kann man mit dem die LEDs auf dem Breadboard per Python-Programm blinken lassen. Die Stellung des kleinen roten Schalters erfährt der Raspi beim Auslesen von Bank A.**

Ein Beispiel macht klar, was passiert: Die Funktion bekommt `pinNo=10` und `value=false` übergeben, soll also das dritte Bit von Bank B auf 0 setzen. Zunächst startet sie die I2C-Kommunikation. Da `pinNo` nicht kleiner 8 ist, geht es um Bank B, sodass die Funktion das Byte `'\x15'` versendet und die Variable `bank` auf 1 setzt. Anschließend berechnet sie die Nummer des Pins in Bank B, indem sie alle Bits außer den niederwertigsten 3 mit der Maske `0b00000111` (Dezimal 7) auf 0 setzt und speichert den Wert in der Variablen `pinNo8`. Im Beispiel setzt sie also `pinNo8=2`. In der nächsten Zeile berechnet sie die Maske `pm`, die an der soeben berechneten Stelle eine 1 hat und deren restlichen Bits alle 0 sind. Im Beispiel ergibt `pm=1<<pinNo8` binär `0b00000100`. Da `value=false`, invertiert die Funktion im `else`-Pfad die Maske `pm` durch `'\xff'^pm`. Das XOR (^) mit einem Byte voller Einsen ergibt binär `0b11111011`. Diese invertierte Maske verrechnet die Funktion mit einem bitweisen Und mit dem bisherigen Byte für Bank B. Für das Beispiel könnte es `0b00001101` lauten. Dann gilt: `0b00001101 & 0b11111011 = 0b00001001`. Die Funktion hat damit erfolgreich das dritte Bit für Bank B von 1 auf 0 geändert. Die Funktion überträgt zuletzt das geänderte Byte an den MCP23017 und beendet die I2C-Kommunikation.

Das Lesen besteht aus dem Senden eines Bytes, das Bank A oder B auswählt und dem Lesen eines Bytes bestehend aus den 8 gelesenen Bits:

```
unsigned char i2cDigitalRead(bool bankA) {
  Wire.beginTransmission(GPIO_E_ADDR);
  if (bankA) {
    Wire.write('\x12');
  } else {
    Wire.write('\x13');
  }
  Wire.endTransmission();
  Wire.requestFrom(GPIO_E_ADDR, 1);
  while (!Wire.available()) {};
  return Wire.read();
}
bool i2cDigitalRead(unsigned short pinNo) {
  bool bankA = (pinNo >> 3) == 0;
  unsigned char m = 1 << (pinNo & 7);
  return i2cDigitalRead(bankA) & m;
}
```

Für Bequemlichkeit sorgt hier die überladene Funktion, die für eine einzelne Pin-Nummer verrät, ob das Bit auf hohem (`true`) oder GND-Potenzial (`false`) liegt. Sie verwendet dafür die gleiche Bitmaske wie die Schreibfunktion und das bitweise Und (`&`) zum Anwenden der Maske.

Die Auswahl der Bank erledigt ein Shift um 3 Bit nach rechts, weil dann bei den Pin-Nummern 0 bis 7 nur Nullen stehen bleiben. Bei höheren Pin-Nummern ist nach dem Shift mindestens eines der Bits 1, sodass eine Zahl größer 0 übrig bleibt.

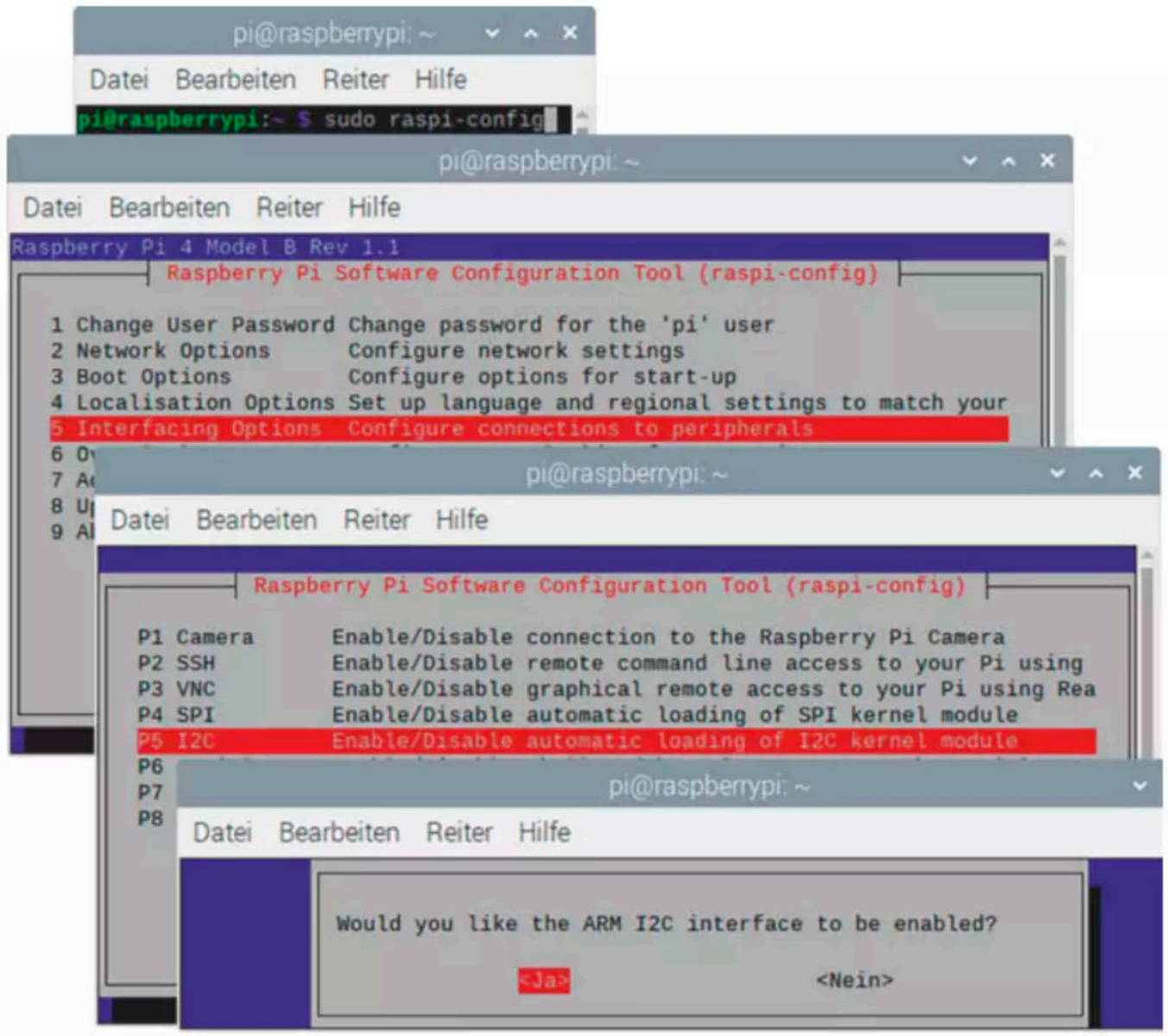
## I2C am Raspi

Ein MCP23017 erweitert im Handumdrehen auch die GPIOs am Raspi. Dafür müssen Sie die i2c-Schnittstelle zunächst aktivieren. Das geht am schnellsten mit der Konsole:

```
sudo raspi-config
```

Wählen Sie dort „5 Interfacing Options/P5 I2C“ und bejahen Sie die Frage, ob Sie I2C aktivieren möchten.

Das Tool trägt im Prinzip nur die beiden Kernelmodule `i2c-bcm2708` und `i2c-dev` in `/etc/modules` ein. Sie können I2C daher mit allen Linux-Distributionen für den Raspi nutzen, die diese Module mitbringen. Überprüfen Sie zusätzlich, ob die Module mit `bcm2708` im Namen in `/etc/modprobe.d/raspi-blacklist.conf` aufgeführt sind. Kommentieren Sie die Zeilen dort aus, falls die Datei die Module aufführt, da der Kernel sonst das Laden verweigert:




**Nach der Installation ist die I2C-Schnittstelle am Raspi zunächst deaktiviert. Aktiviert man sie mit raspi-config, lädt das alle nötigen Kernelmodule automatisch.**

```
#blacklist spi-bcm2708
#blacklist i2c-bcm2708
```

Um über Python und die Konsole auf I2C zugreifen zu können, installieren Sie unter Raspberry Pi OS (früher „Raspbian“) noch die folgenden zwei Pakete:

```
sudo apt install python-smbus
sudo apt install i2c-tools
```

Danach sollten Sie einmal neu starten.

Das I2C-Zeitsignal SCL legt der Raspi an seinem Pin 5 an. Daten (SDA) laufen über Pin 3. Zusätzlich braucht der MCP23017 noch eine Versorgungsspannung von 3,3 Volt (Pin 1) und eine Masseverbindung (Pin 6, 9, 14, 20, 25, 30, 34 oder 39). Insgesamt brauchen Sie also vier Jumperkabel vom Raspi zum Breadboard.




**Der Befehl i2cdetect fragt alle 256 Adressen ab. Antwortet ein Chip, taucht seine Adresse im Raster auf.**

## I2C-Konsolentools

Um sich einen Überblick über die I2C-Geräte zu verschaffen, sollten Sie auf der Konsole zunächst nach Geräten suchen:

```
sudo i2cdetect -y 1
```

Mit der so gefundenen Adresse 64 (`'\x40'`) kann man nun die Pins von Bank A konfigurieren (A-0 bis A-2 sind im Beispiel Ausgabe-, der Rest Eingabe-Pins):

```
sudo i2cset -y 1 0x40 0x00 0xf8
```

Die drei Ausgabe-Pins schaltet der folgende Befehl danach ein:

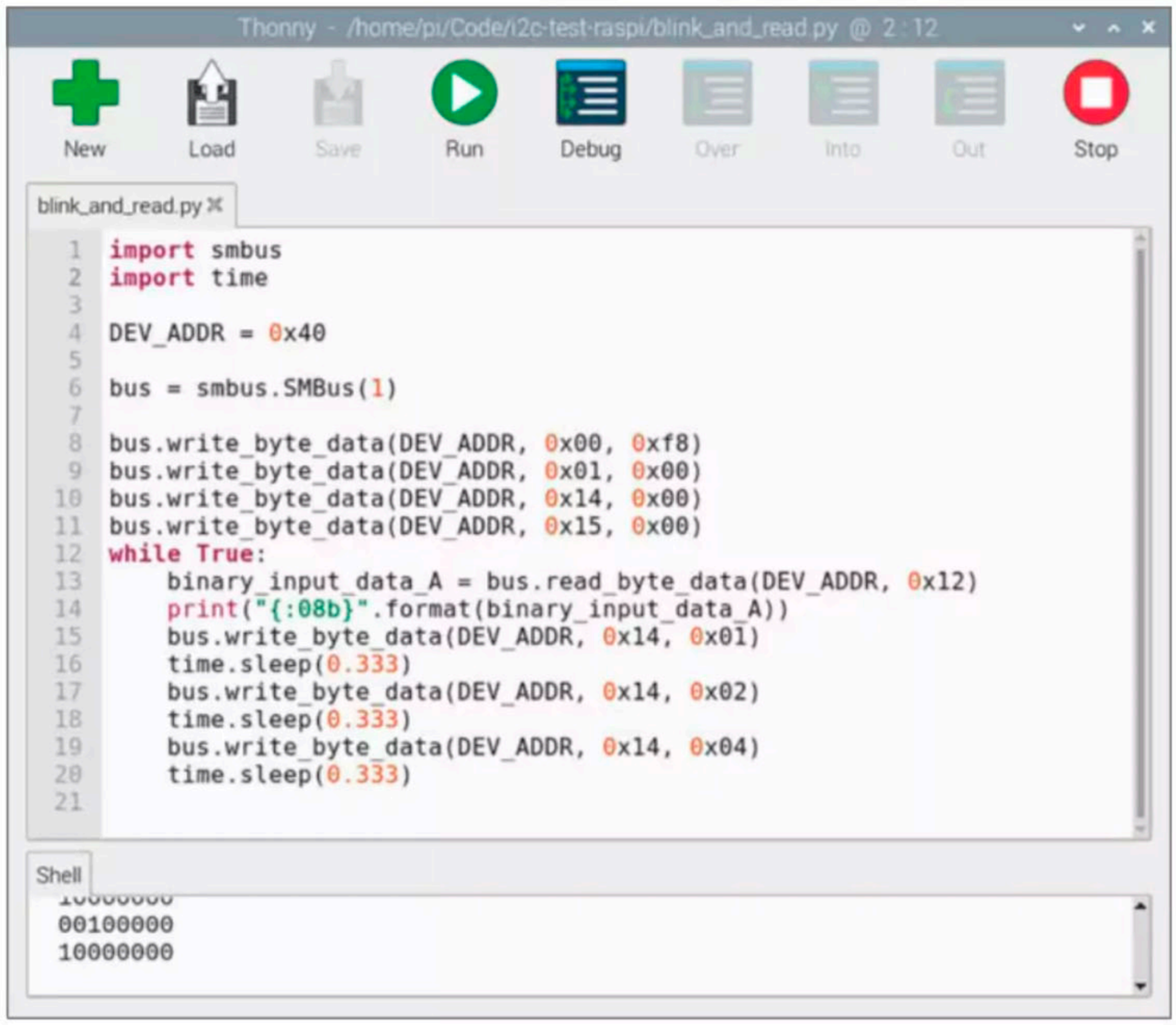
```
sudo i2cset -y 1 0x40 0x14 0x07
```

## I2C über Python

Das Python-Modul `smbus` erlaubt noch flexibleren Zugriff auf I2C. Initialisiert wird es ab Raspi 2 mit der Versionsnummer 1:

```
import smbus
bus = smbus.SMBus(1)
```




**Ein Blinkprogramm mit I2C braucht auf dem Raspi nur wenige Zeilen. Das Beispiel liest zusätzlich noch eine Schalterstellung ein und gibt die Bits aus.**

Das Schreiben übernimmt danach die Methode `write_byte_data()`:

```
ADDR = 0x40
bus.write_byte_data(ADDR, 0x00, 0xf8)
bus.write_byte_data(ADDR, 0x14, 0x00)
```

Zum Lesen steht eine praktische Kombi-Methode zur Verfügung, die in einem Befehl Bank A spezifiziert und die Daten annimmt:

```
data = bus.read_byte_data(ADDR, 0x12)
```

Es gibt auch eine Variante dieser Methode, der man als weiteren Parameter eine Anzahl an Bytes angibt. Sie liest in einem Rutsch eine Liste von Bytes (hier zum Lesen des Analogwerts von Pin A0 vom Arduino Nano; siehe Kasten):

```
data = bus.read_i2c_block_data(ADDR,
                               0x20, 4)
def bytes_to_int(x) -> int:
  return (x[0] << 24 + x[1] << 16 +
          x[2] << 8 + x[3])
analog_value = bytes_to_int(data)
```

## Fazit

Für viele Projekte mit zahlreichen Schaltern, LEDs und Relais ist ein per I2C angebundener GPIO-Extender die ideale Lösung, um an mehr Pins zu kommen. I2C sollte man nicht über meterlange Kabel schicken, aber für eine Modularisierung der Projekte taugt der Bus.

Man sollte allerdings nicht vergessen, dass die serielle Übertragung etwas Zeit kostet. Wer sich also für einen möglichst exakten Zeitpunkt interessiert, an dem ein Taster geschlossen wurde, sollte ihn lieber direkt an den Mikrocontroller hängen. Der Zeitversatz wäre beispielsweise ein Problem für unsere Selbstbau CNC-Fräsen auf Seite 102 und 110. Die Zusatz-GPIOs sind auch ungeeignet, um zeitkritische Signale für einen seriellen Bus zu erzeugen. Software-PWM geht darüber nur, wenn man mit sehr niedrigen Frequenzen schaltet.

Der MCP23017 ist mit einem Preis von meist 99 Cent billiger als ein Arduino Nano und obendrein etwas kleiner. Der Arduino punktet aber mit sechs A/D-Wandlern und der Möglichkeit, Daten schon mal vorzuverarbeiten, bevor sie an den I2C-Master, beispielsweise einen ESP, gehen. (pmk) c't

**Repositories und Datenblätter**

ct.de/wbg9

# IIoT Conference

Industrial IoT CONFERENCE

## Die Heise-Konferenz für Industrial IoT in Kooperation mit der automatica

**16. Juni 2021**

**10 % Mai-Rabatt** für c't-Leser – Ihr Code: 2021IIOT4CT_SDF189

### Die Brücke zwischen Automatisierung und Softwareentwicklung

### 12 Vorträge unter anderem zu

- W3C-Standard Web of Things
- Serverless IIoT
- Deep Dive MQTT 5
- Die Verwaltungsschale am Beispiel FabOS
- Robot Operating System (ROS)

**www.iiot-conference.de**

**Zusätzliche Workshops zu MQTT 5 und Machine Learning mit IIoT-Sensordaten**

Gold-Sponsor

VECTOR

Veranstalter

iX heise Developer dpunkt.verlag



Arduino-Gehäuse

Lautstärkeregler

# Arduino lernt Lautstärkeregler

**Arduino Micro und Leonardo können sich dank internem USB als Eingabegeräte ausgeben und dem PC Tastendrücke und Mausbewegungen senden, die jedes Betriebssystem ohne Treiber versteht. Ein Lautstärke-Drehknopf ist mit einem Arduino Micro schnell gebaut.**

Von **Pina Merkert**

Den ganzen Abend YouTube-Videos zu schauen kann lehrreich und entspannend sein. Manche Videos sind professionell, andere ideenreich, aber schlecht produziert. Besonders der Tonpegel schwankt enorm: Mal dringt die Stimme des YouTubers kaum hervor, mal erschlägt sie einen komprimiert, normalisiert und vor allem laut! Jedes Anpassen des Pegels am Rechner verlangt mit der Maus zwei präzise Klicks und dauert eine Sekunde, in der man entweder nichts versteht oder unter stimmlicher Tongewalt zusammenschrumpft. Und bei jedem Klick-Schieb wächst die Sehnsucht nach einem Drehregler wie an alten Verstärkern.

Statt ewiger Sehnsüchtelei wird man besser tätig und greift zum Lötkolben. Ein Eingabegerät mit Drehregler lässt sich nämlich schnell und günstig selbst bauen. Herzstück ist ein Arduino mit einem Atmel Mega-23U4-Prozessor wie der Leonardo oder der Micro. Die integrieren im Gegensatz zum Mega-328p (beispielsweise im Arduino Nano) die USB-Schnittstelle direkt in den Chip und können sich daher neben der üblichen seriellen Schnittstelle auch als HID-Gerät (Human Interface Device) ausgeben. Das sind Tastaturen, Mäuse und Gamepads, die dank ihres einheitlichen USB-Standards an allen modernen Betriebssystemen ohne Treiber funktionieren.

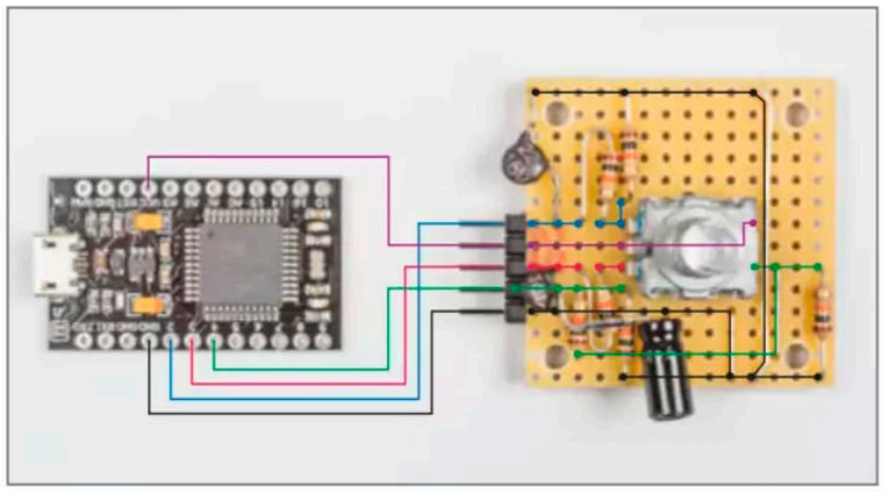
Übliche Tastendrücke sendet der Arduino Micro mit der integrierten Bibliothek `Keyboard.h` [1]. Eine Lautstärkeregelung muss aber die Scancodes von Sondertasten außerhalb des üblichen Layouts versenden und die sind in `Keyboard.h` leider nicht enthalten. Eine andere Bibliothek schließt aber die Lücke: `HID-Project.h`. Mit dieser Bibliothek kann ein Arduino nahezu alles, was mit dem HID-Standard möglich ist: Tastatur, Sondertasten, relative Mausbewegungen, absolute Maus wie bei einem Grafiktablett und die Simulation eines Gamepads mit Joysticks und Dutzenden Tasten. Der ersehnte Lautstärke-Drehknopf braucht von all diesen Möglichkeiten nur die Sondertasten für lauter, leiser und stumm.

## Drehencoder

Das Bauteil hinter dem Drehknopf ist ein Drehencoder. Was aussieht wie ein Potentiometer, ist eigentlich ein raffinierter Schalter. Im Inneren werkeln bei den billigeren Varianten Schleifkontakte, bei den teureren Lichtschranken, die an zwei Pins digitale Impulse abgeben, die erkennen lassen, in welche Richtung der Encoder gedreht wurde. Auf dem Weg rastet der Drehencoder nach jedem Impuls einmal ein. Ein Minimum oder Maximum kennen Drehencoder nicht: Sie senden nur Impulse je nach Drehrichtung, sodass man sie auch um mehr als 360° drehen kann.

Drehencoder nutzen drei Anschlüsse. Den mittleren verbindet man mit der Spannungsversorgung und die beiden äußeren geben beim Drehen jeweils einen Impuls ab. Die Impulse entstehen im Inneren über zwei Schleifkontakte, die über eine Platine mit so vielen Kontaktflächen gleiten, wie der Encoder Rastungen hat. Die Kontaktflächen sind aber so versetzt, dass einer der Kontakte bei einer Rechtsdrehung zuerst schließt, danach erst der zweite Kontakt schließt und die beiden Kontakte bis zur nächsten Rastung in der gleichen Reihenfolge auch den Kontakt verlieren. Beide Pins liefern also Impulse, allerdings zeitlich versetzt. Bei einer Linksdrehung ist die Reihenfolge vertauscht, sodass man aus der zeitlichen Reihenfolge der Impulse auf die Drehrichtung schließen kann.

Als Bonus integrieren viele Drehencoder einen Taster. Dessen Kontakt schließt, wenn man von oben auf den Knopf drückt. Elektrisch ist das im Inneren aber komplett getrennt, weshalb solche Drehencoder insgesamt fünf Pins haben: drei für die Drehung und auf der gegenüberliegenden Seite zwei für den Taster. Der Taster dient beim vorgestellten Lautstärkeregler dem Stummschalten.



**Die sechs Widerstände und drei Kondensatoren auf der Streifenraster-Platine sorgen für verlässliche Pegel. Mehr Verdrahtung braucht das Projekt nicht.**

## Entprellen

Besonders die günstigen Drehencoder mit Schleifkontakten neigen zum Prellen. Berührt ein federnder Schleifer eine Kontaktfläche, schließt das meist nicht sofort den Kontakt, sondern der Schleifer stellt mehrfach in kurzer Folge Stromfluss her und unterbricht ihn wieder. Ein Prellvorgang dauert typischerweise 250 Mikrosekunden (siehe Seite 18) – eine sehr kurze Zeitspanne, die ein Arduino aber durchaus registrieren kann.

Zum Entprellen reicht an jedem Pin ein Kondensator mit mickrigen 22 nF, solange der sich nicht mit zu großem Strom laden und entladen kann, was ein 10-k-Widerstand sicherstellt. Ein weiterer 10-k-Pull-down-Widerstand verhindert, dass der Pin innerhalb der Rastungen in der Luft hängt.

Für die insgesamt sechs Widerstände und drei Kondensatoren reicht eine kleine Streifenrasterplatine. Die hat neben Masse und Stromversorgung nur die drei entprellten Verbindungen zum Mikrocontroller, die man mit den Pins 2, 3, und 4 am Arduino Micro verbindet.

## Endlicher Automat

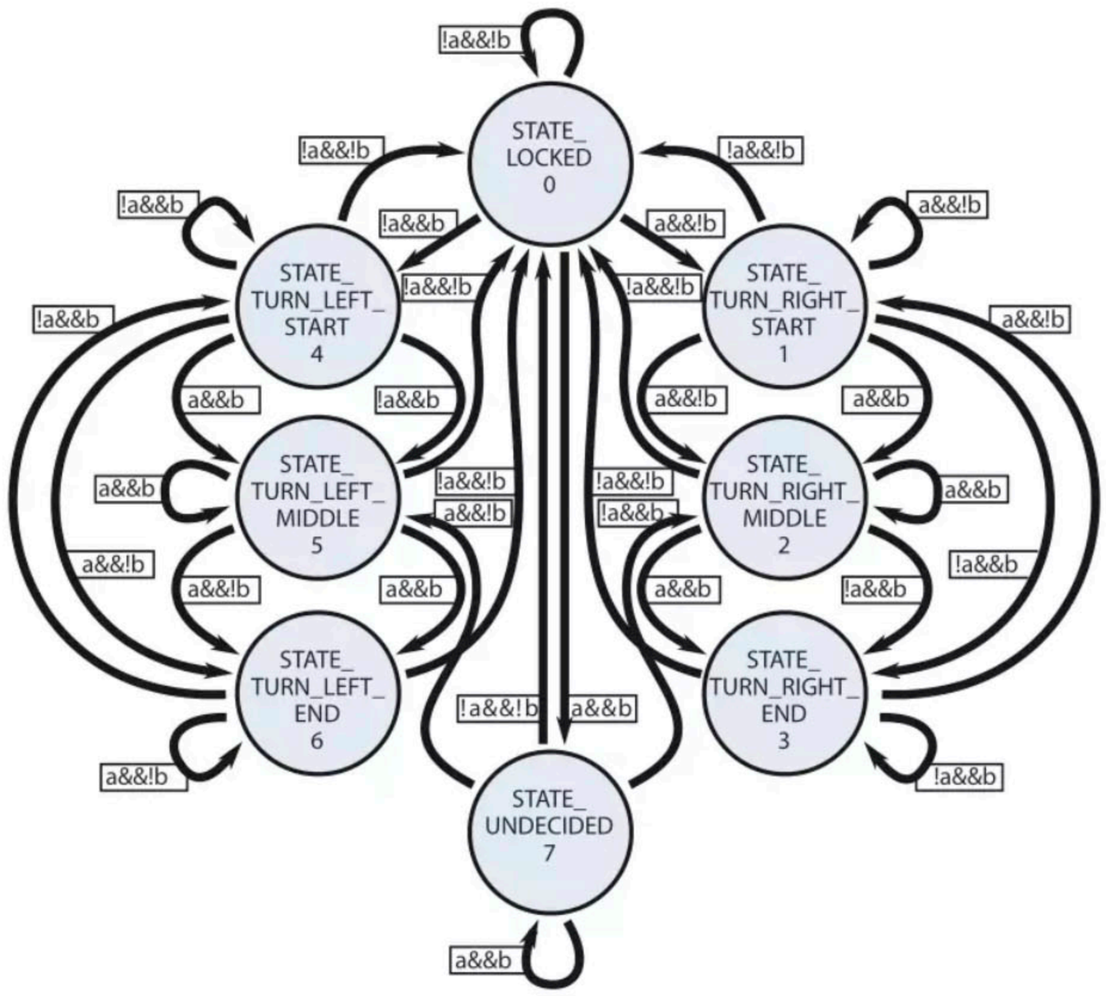
Die Software auf dem Arduino muss nun die Pins 2 und 3 auswerten und aus dem zeitlichen Versatz der Impulse auf die Drehrichtung schließen. Da der Arduino kaum etwas zu tun hat, reicht es dafür, in der `loop()`-Funktion mit `digitalRead()` die Eingänge zu pollen (siehe Listingkasten). Die Firmware speichert die Werte als `a` und `b`. Die Schleifkontakte eines günstigen Drehencoders sind aber notorisch ungenau, sodass beispielsweise beide Pins gleichzeitig von low auf high wechseln können und die Software dann nicht weiß, welche Richtung zu diesem Impuls

gehört. Die Auswertung sollte daher möglichst fehlertolerant sein.

Der Beispiel-Code nutzt für eine möglichst robuste Auswertung einen endlichen Automaten. Ein endlicher Automat befindet sich immer in einem von endlich vielen Zuständen und wechselt bei Ereignissen wie Eingaben in einen anderen Zustand (im Graph rechts als Pfeile dargestellt). Bei den Zustandsübergängen kann ein solcher Zustandsautomat etwas ausgeben (im Beispiel durch einen Funktionsaufruf), muss er aber nicht.

In Ruhe startet der Drehencoder im Zustand `STATE_LOCKED`. Wechselt `a` auf high und `b` bleibt auf low, zeigt das den Beginn einer Drehung nach rechts an und der Automat wechselt in den Zustand `STATE_TURN_RIGHT_START`. Die Lautstärke ändert die Firmware deshalb aber noch nicht, weil es sein könnte, dass der Nutzer den Knopf wieder zurückdreht und der Automat ohne Aktion zurück zu `STATE_LOCKED` wechseln muss (Zustandsübergang `!a && !b`, das `!` markiert eine logische Verneinung in C). Dreht er stattdessen weiter, wechselt auch `b` auf high und der Automat in den Zustand `STATE_TURN_RIGHT_MIDDLE`. Von dort ist es sinnvoll, beim Zustandsübergang `!a && !b` die Lautstärke zu erhöhen, auch wenn der Knopf den Zustand `STATE_TURN_RIGHT_END` überspringt und die Firmware nie `a` auf low und `b` auf high sieht.

Insgesamt ergibt sich ein endlicher Automat mit acht Zuständen und je vier Zustandsübergängen für die vier möglichen Kombinationen `a&&b`, `!a&&b`, `a&&!b` und `!a&&!b` (siehe Graph). Im Code realisiert eine `switch-case`-Anweisung den Automaten (siehe Listing).




**Ein endlicher Automat stellt sicher, dass die Firmware den Drehencoder möglichst sinnvoll auswertet, selbst wenn die Schleifkontakte den Stromfluss nicht immer verlässlich herstellen.**

## HID-Project

Bindet man in den Code die Header-Datei `HID-Project.h` ein, stellt die Interfaces für alle Funktionen zur Verfügung, die man braucht, um standardkonforme Tastendrücke an den PC zu senden. In der Arduino-IDE müssen Sie das HID-Project als externe Bibliothek nachinstallieren, in PlatformIO reicht es, in `platformio.ini` unter `lib_deps` eine Zeile mit `HID-Project` hinzuzufügen. Danach klappt der Import ganz oben im Kopf von main.cpp per `#include <HID-Project.h>`

Die Bibliothek enthält mehrere Klassen für Mäuse, Tastaturen und Gamepads (die Dokumentation finden Sie über ct.de/w18v). Die Sondertasten eines Multimedia-Keyboards (der Lautstärke-Drehregler fällt unter diese Kategorie) sprechen Sie über die Klasse `Consumer` an. Dafür initialisieren Sie diese in der `setup()`-Funktion mit `Consumer.begin();`. Anschließend kann das Programm Tastendrücke versenden, beispielsweise beim Drehen des Encoders nach rechts:

```
void stepRight() {
  Consumer.write(MEDIA_VOL_UP);
}
```

Die Bibliothek bringt alle nötigen Definitionen mit, damit Sie im Code beispielsweise `MEDIA_VOL_UP` schreiben können, statt den Scancode dieses Tastendrucks zum Erhöhen der Lautstärke auswendig wissen zu müssen.

Alle weiteren Tastendrücke funktionieren nach dem gleichen Schema. Den kompletten Code des Projekts samt `platformio.ini` finden Sie über ct.de/w18v.

## Gehäuse und Knopf

Drehencoder und Arduino Micro finden in einem kleinen Gehäuse aus Holz mit Blechfront Platz. Durch eine kleine Öffnung in der Rückseite verbindet

**Schaltplan, Video, Quellcode der Firmware und CAD-Dateien**

ct.de/w18v

```
#define ENC_A_PIN 3
#define ENC_B_PIN 2
#define STATE_LOCKED 0
#define STATE_TURN_RIGHT_START 1
#define STATE_TURN_RIGHT_MIDDLE 2
#define STATE_TURN_RIGHT_END 3
#define STATE_TURN_LEFT_START 4
#define STATE_TURN_LEFT_MIDDLE 5
#define STATE_TURN_LEFT_END 6
#define STATE_UNDECIDED 7
uint8_t a = digitalRead(ENC_A_PIN);
uint8_t b = digitalRead(ENC_B_PIN);
switch (encoderState) {
  case STATE_LOCKED:
    if (a && b)       { encoderState = STATE_UNDECIDED; }
    else if (!a && b) { encoderState = STATE_TURN_LEFT_START; }
    else if (a && !b) { encoderState = STATE_TURN_RIGHT_START; }
    else              { encoderState = STATE_LOCKED; };
    break;
  case STATE_TURN_RIGHT_START:
    if (a && b)       { encoderState = STATE_TURN_RIGHT_MIDDLE; }
    else if (!a && b) { encoderState = STATE_TURN_RIGHT_END; }
    else if (a && !b) { encoderState = STATE_TURN_RIGHT_START; }
    else              { encoderState = STATE_LOCKED; };
    break;
  case STATE_TURN_RIGHT_MIDDLE:
  case STATE_TURN_RIGHT_END:
    if (a && b)       { encoderState = STATE_TURN_RIGHT_MIDDLE; }
    else if (!a && b) { encoderState = STATE_TURN_RIGHT_END; }
    else if (a && !b) { encoderState = STATE_TURN_RIGHT_END; }
    else              { encoderState = STATE_LOCKED; stepRight(); };
    break;
  case STATE_TURN_LEFT_START:
    if (a && b)       { encoderState = STATE_TURN_LEFT_MIDDLE; }
    else if (!a && b) { encoderState = STATE_TURN_LEFT_START; }
    else if (a && !b) { encoderState = STATE_TURN_LEFT_END; }
    else              { encoderState = STATE_LOCKED; };
    break;
  case STATE_TURN_LEFT_MIDDLE:
  case STATE_TURN_LEFT_END:
    if (a && b)       { encoderState = STATE_TURN_LEFT_MIDDLE; }
    else if (!a && b) { encoderState = STATE_TURN_LEFT_START; }
    else if (a && !b) { encoderState = STATE_TURN_LEFT_END; }
    else              { encoderState = STATE_LOCKED; stepLeft(); };
    break;
  case STATE_UNDECIDED:
    if (a && b)       { encoderState = STATE_UNDECIDED; }
    else if (!a && b) { encoderState = STATE_TURN_RIGHT_END; }
    else if (a && !b) { encoderState = STATE_TURN_LEFT_END; }
    else              { encoderState = STATE_LOCKED; };
    break;
}
```

**Für den endlichen Automaten nutzt der Code eine einfache `switch-case`-Anweisung. Von jedem Zustand gibt es anhand vier möglicher Eingaben Übergänge zu anderen Zuständen. Ausgaben mit den Funktionen `stepRight()` und `stepLeft()` macht der Automat nur bei 4 der 32 Übergänge.**

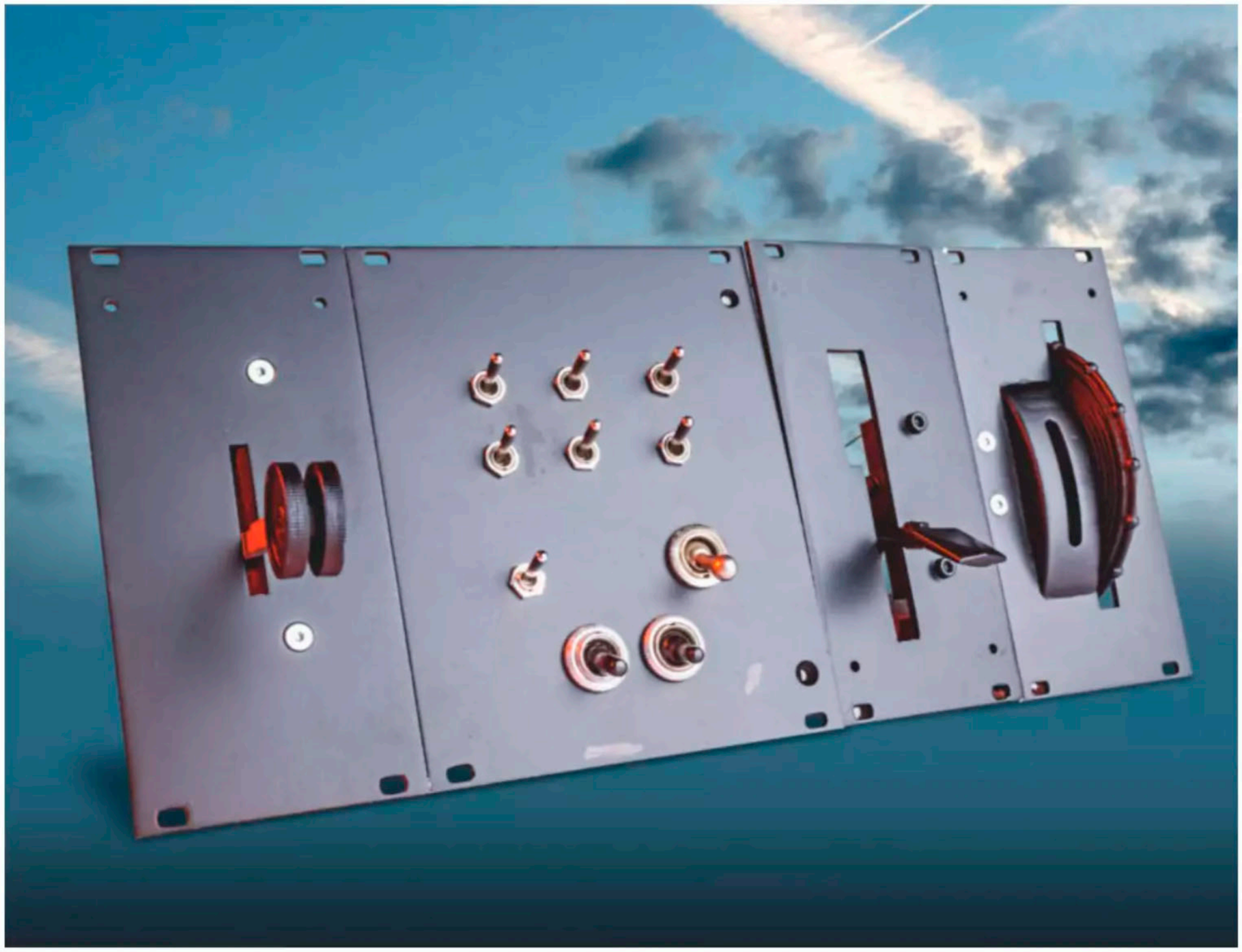
## Literatur


[1] Tim Gerber: **Anschläge**, PC-Programme mit Arduino steuern, c't 8/2016, S. 156


ein kurzes Micro-USB-Kabel den Lautstärkeregler mit dem PC. Der erkennt das Eingabegerät ohne Zeitverzögerung, sodass man unter allen Betriebssystemen damit direkt die Lautstärke regeln kann.

Um den Encoder schön greifen und vor allem seinen Taster gut bedienen zu können, spendiert man ihm noch einen Knopf aus dem 3D-Drucker. Die CAD-Dateien und STL-Vorlagen finden Sie im Git-Repository zusammen mit der Firmware.

Bei YouTube-Abenden kann man nun noch besser entspannen, da die perfekte Lautstärke nur noch eine kleine Drehung erfordert. Und das Basteln dieser Schreibtisch-Ergänzung macht sogar noch mehr Spaß, als die Videos zu schauen. (pmk) c't





HID-Flugsimulator-Interface

# Arduino-Cockpit für den Flugsimulator

**Mit den richtigen Knöpfen und Hebeln fühlt sich Fliegen mit dem Microsoft Flugsimulator 2020 noch authentischer an. Ein passender USB-Controller mit Arduino Micro ist schnell gebaut und bindet Trimmrad, Landeklappen- und Fahrwerkshebel einer Cessna an den Rechner an.**

Von **Pina Merkert**

Baut man Bedienelemente eines Flugzeug-Cockpits mit physischen Hebeln, Rädchen und Schaltern nach, holt man den virtuellen Raum des Flugsimulators ein Stück weit in die Realität. Die Simulation fühlt sich authentischer an und das Fluggefühl wird intensiver. Da jeder virtuell fliegende Pilot ein anderes Flugzeug mag, hat jeder auch andere Vorstellungen für die perfekte Form der Bedienelemente – und die setzt man am besten mit einem selbst gebauten Controller um. Mit einem Arduino Micro als Herzstück geht das einfach und kostengünstig.

Gibt sich der Arduino als Tastatur aus [1], kann man jedes Mal, wenn am selbst gebauten Controller ein Schalter umgelegt wird, einen Tastendruck oder eine Tastenkombination an den Rechner schicken. Moderne Betriebssysteme brauchen für solche „Human Interface Devices“ (HID) keine besonderen Treiber und über die Einstellungen im Flugsimulator und in anderen Spielen kann man bequem definieren, was passieren soll, wenn der Arduino in die Tasten haut.

Ein Mikrocontroller kann sich nur dann als HID ausgeben, wenn er die USB-Kommunikation selbst abwickelt. Steckt wie beim Arduino Uno oder Nano für USB ein Extra-Chip auf dem Board, lässt sich damit kein HID emulieren. Ein Atmel MEGA 32U4, der als HID arbeiten kann, ist auf einem Arduino Micro, Leonardo oder auf dem inoffiziellen 3,3-V-Klon „Pro Micro“ verlötet. Wir haben einen Arduino Micro verwendet, der 16 GPIO-Pins zur Verfügung stellt. Die waren uns aber nicht genug, weshalb wir per I²C einen MCP23017 (siehe S. 68) angebunden haben, der weitere 16 GPIO-Pins zur Verfügung stellt (für I²C gehen dafür am Arduino zwei GPIO verloren).

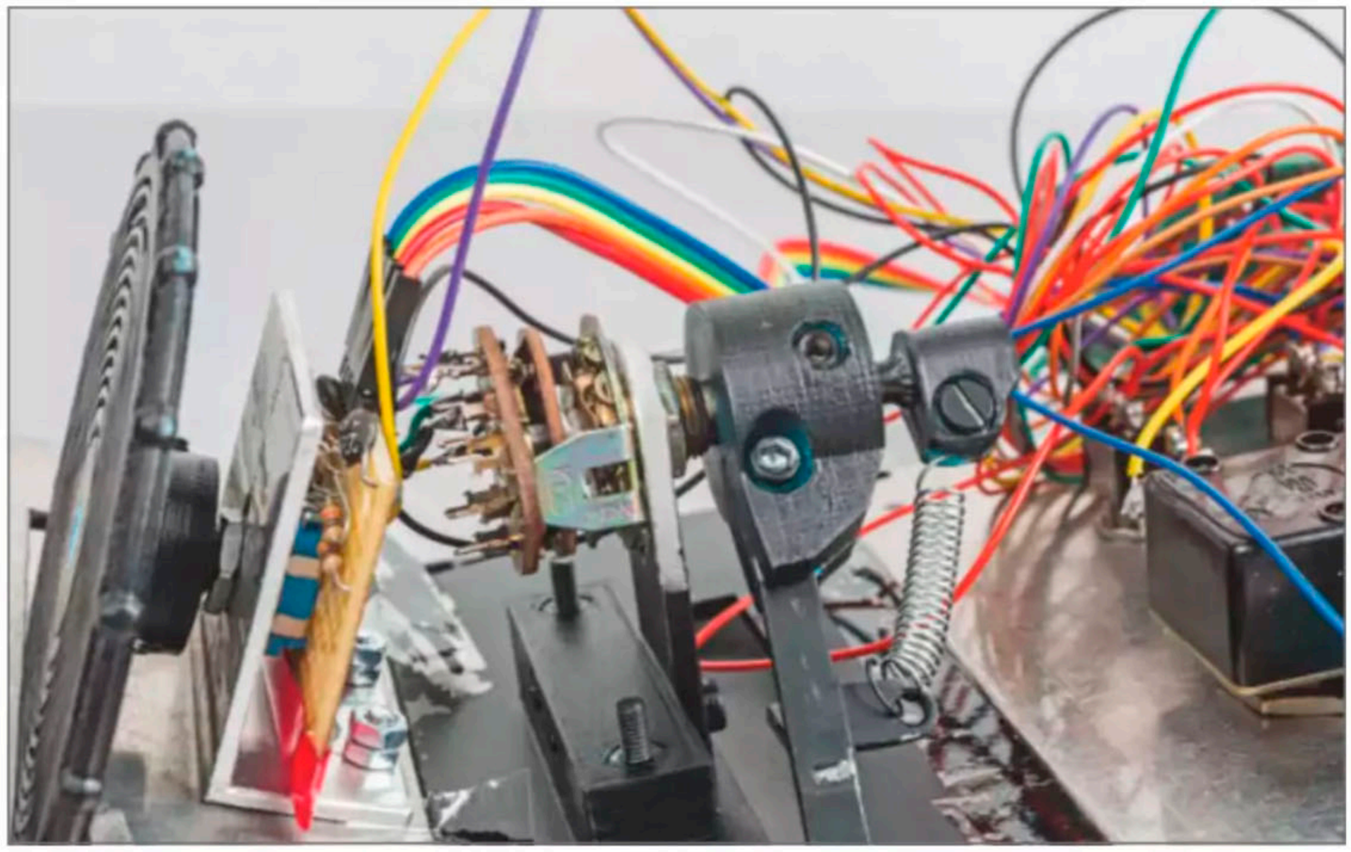


**Den Fahrwerkshebel einer Cessna haben wir originalgetreu nachgebaut. Damit der Hebel genug Hub bekommt, bewegt er keinen Kippschalter, sondern einen Schiebeschalter.**

Wir haben unseren Controller „Fliegender Frederick“ getauft. Wenn Sie wollen, können Sie ihn gern nachbauen – Vorlagen und Beispiele dazu finden Sie auf ct.de/wzts. Alle einzelnen Punkte in einer Schritt-für-Schritt-Anleitung aufzulisten würde aber den Umfang dieses Artikels sprengen. Da Ihre Werkstatt wahrscheinlich anders als unsere ausgestattet ist, müssen Sie für einen Eins-zu-eins-Nachbau eigene Wege finden, die Fronten auszuschneiden oder die Befestigungen herzustellen. Außerdem macht das Projekt ohnehin mehr Spaß, wenn Sie unseren Frederick nur als Vorlage benutzen und Ihren Controller an die Fundstücke in der eigenen Bastelkiste und die eigenen Bedürfnisse anpassen. Vielleicht ist Ihr Lieblingsflugzeug ja gar keine Cessna und die Hebel müssen etwas anders aussehen.

Unser Controller nutzt genau wie der Lautstärkeregler auf Seite 76 einen Drehencoder. Der registriert Drehungen am Trimmrad, mit dem man die Nase des Flugzeugs nach oben oder unten neigt. Die Verschaltung haben wir eins zu eins vom Lautstärkeregler übernommen, da die dort beschriebene kleine Platine unübersichtlichen Kabelverhau an den Anschlüssen verhindert. Die restlichen Bedienelemente sind elektrisch gesehen einfache Schalter, sehen aber teilweise anders aus. Für den Hebel fürs Ausfahren des Fahrwerks haben wir beispielsweise einen Schiebeschalter mit einem langen Hebelarm verheiratet und ans Ende zwei 3D-gedruckte Kunststoffscheiben geschraubt. Das Ergebnis sieht dem Fahrwerkshebel einer Cessna täuschend ähnlich.

Für die Landeklappen hat eine Cessna einen Hebel, der vier Stellungen annehmen kann. Ihn haben wir mit einem um 90° gedreht eingebauten Drehschalter umgesetzt. An seiner Welle setzt ein Hebelarm an, der eine Besonderheit aufweist: Offensichtlich als Schutz vor versehentlich zu extrem gehobenen Landeklappen kann man ihn mit einer geraden Bewegung nach unten nur auf die erste Stufe stellen. Für größere Bremswirkung muss der Pilot ihn etwas zur Seite schieben und dann nach unten drücken. Damit unser Schalter in der geschützten Stellung einrastet, zieht eine Zugfeder den Hebelarm zur Welle. Eine Stufe im Ausschnitt für den Hebel in der Frontplatte sperrt den Hebel für unüber-

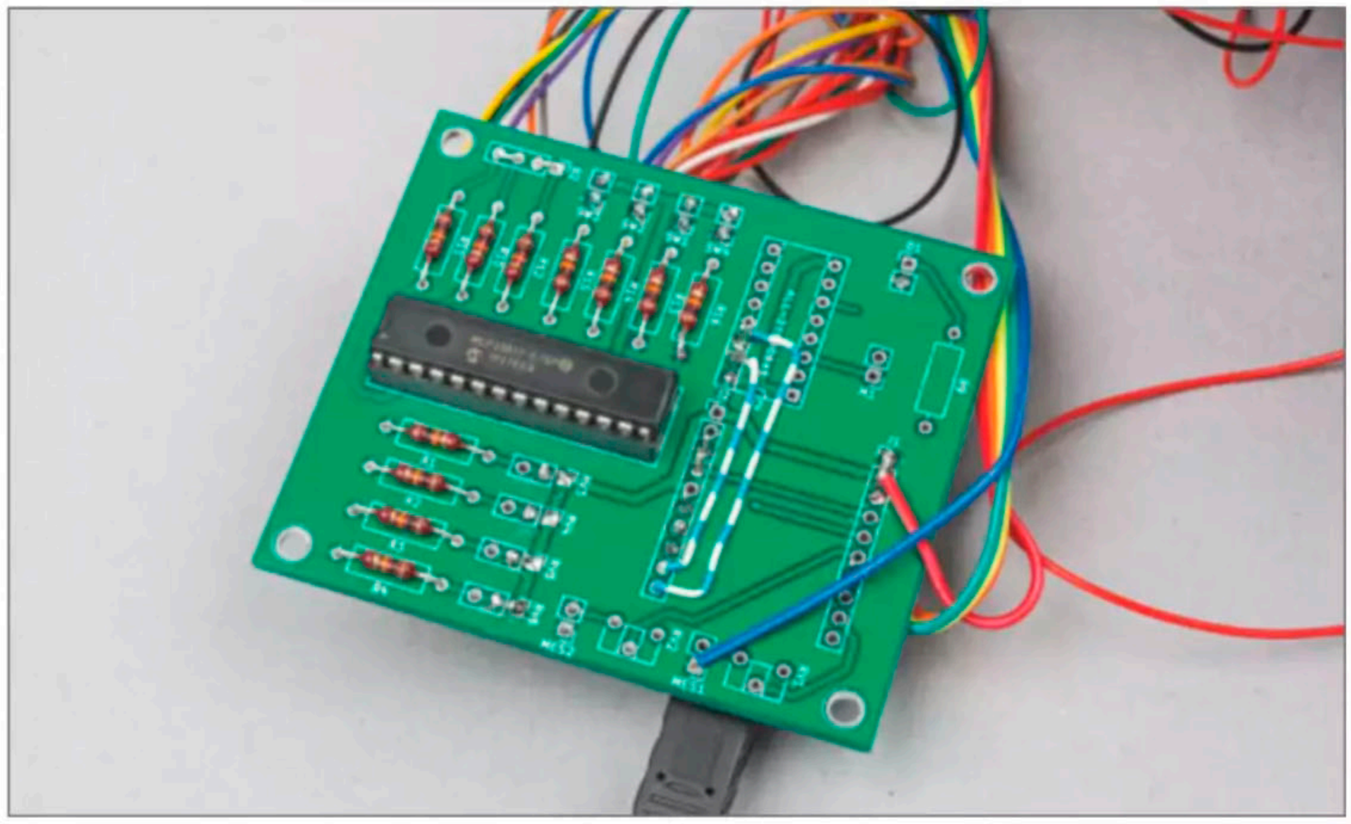


**Der Hebel für die Landeklappen verhindert Fehlbedienungen mit einer mechanischen Sperre: Die Feder zieht den Hebel nach rechts, sodass er bei Stufe 1 auf einer Kante des Ausschnitts in der Frontplatte aufliegt. Wer weiter schalten will, muss ihn erst nach links und dann nach unten bewegen.**

legte Bewegungen. Der Flugsimulator-Pilot muss den Hebel also gegen die Federkraft nach rechts drücken, um Stufe 2 oder 3 zu erreichen.

## Platinen-Recycling

Die beiden Drähte für die I²C-Verbindung zum MCP23017 und die Pulldown-Widerstände an allen GPIO-Pins baut man leicht auf einer Lochrasterplatine auf. Wir hatten jedoch für ein anderes Projekt in der Redaktion vor einiger Zeit Platinen fertigen lassen, die einen MCP23017 und diverse Widerstände aufnehmen können. Da man bei Platinenfertigern immer gleich einen ganzen Stapel bekommt, lagen noch mehrere dieser Platinen ungenutzt im Lager. Wir haben sie, wo es sinnvoll war, bestückt und auf einige ungenutzte Bohrlöcher huckepack den Arduino Micro aufgelötet. Die Gerber-Files für das Platinenlayout finden Sie unter ct.de/wzts – da die Platine aber nur halb passt, lohnt es sich allerdings nicht, sie ohne Änderung fertigen zu lassen. Falls Sie größer in die Controller-Produktion einsteigen



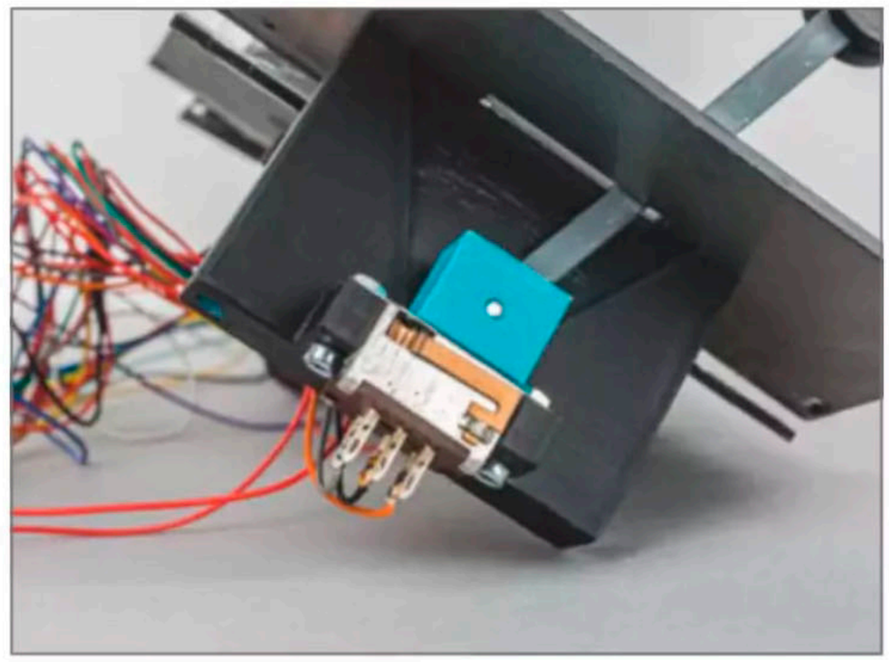
**Unsere zweckentfremdete Platine mit dem MCP23017 trägt die meisten Pulldown-Widerstände. Nötig ist sie aber nicht. Eine etwas breitere Loch- oder Streifenrasterplatine reicht ebenfalls.**

möchten, können Sie aber gern unsere KiCAD-Datei als Basis nehmen.

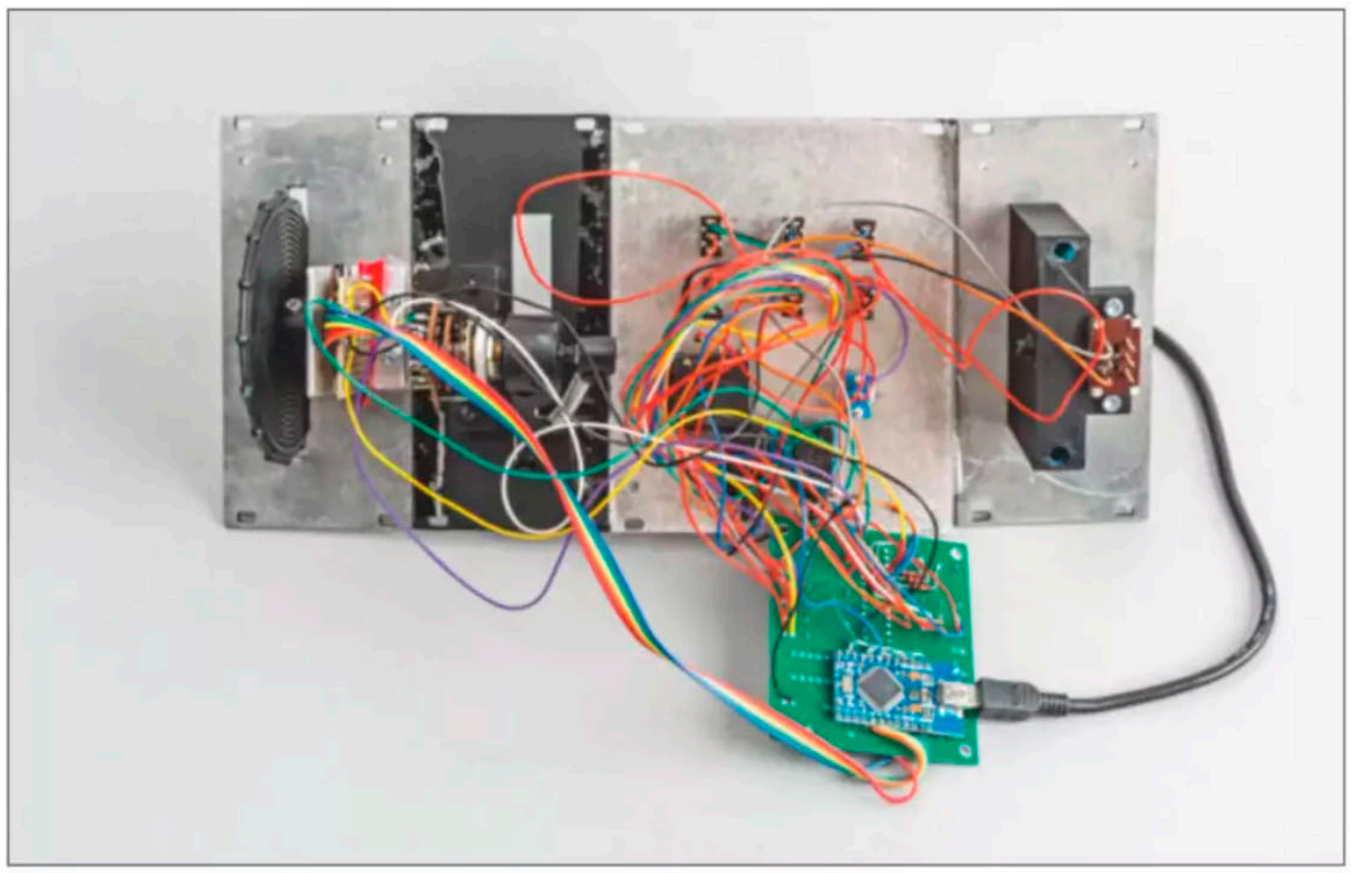
Außer beim Drehencoder (die Schleifkontakte sind besonders anfällig für Prellen) haben wir die Schalter alle in Software entprellt und auf Kondensatoren verzichtet. Falls Ihnen eine Hardware-Lösung lieber ist, können Sie sich an den Kapazitäten und Widerstandswerten unserer Entprellung für den Raspi orientieren (siehe S. 18).

## Mechanik

Trimmrad, Fahrwerkshebel und den Landeklappen-Wahlschalter haben wir jeweils mit einer eigenen Frontplatte aus 2 Millimeter dickem Aluminium verschraubt. So entstehen Module, die man beliebig anordnen und bei Bedarf auch noch erweitern kann. Eine vierte, etwas breitere Aluplatte nimmt noch zehn zusätzliche Kippschalter auf, die weitere Funktionen des Flugzeugs schalten können. Sechs der Kippschalter haben eine kompakte Bauform und passen so in zwei Reihen zu je drei Stück. Darunter finden drei größere Kippschalter Platz, die wir aus alten Geräten ausgebaut haben. Die etwas antiken Schalter bieten ein sehr befriedigendes Schaltgefühl, da sie etwas schwerer gehen und mit Macht in der neuen Stellung einrasten. Das Schalter-Recycling spart nicht nur Geld, die unterschiedlichen Druckpunkte sorgen auch für haptische Abwechslung und somit mehr Spaß. Aus denselben Gründen haben wir als zehnten Schalter keinen Umschalter eingesetzt, sondern einen Taster, allerdings mit einem Hebel zum Herunterdrücken, statt eines Knopfs zum Eindrücken. Dadurch sieht er aus wie die anderen Kippschalter, fühlt sich aber anders an.



**Der türkisene Schuh bewegt den Zapfen des Schiebeschalters für den Fahrwerkshebel. Er sitzt mit etwas Spiel auf dem Zapfen, was den Unterschied zwischen der radialen Bewegung des Hebels und der linearen Bewegung des Schiebeschalters ausgleicht.**



**Drehencoder, Drehschalter und Schiebeschalter sind in ungewöhnlicher Orientierung hinter den Aluminium-Frontplatten verschraubt.**

Die Schalter und Regler sind teilweise anders eingebaut, als ihre Datenblätter spezifizieren. Beispielsweise hängen Drehencoder und Drehschalter jeweils horizontal hinter der Frontplatte. Die Welle ragt nicht wie üblich vorn aus dem Gehäuse, stattdessen interagiert der Pilot mit einem angeschraubten Rad oder Hebel. Beim Fahrwerkshebel schiebt der Hebel einen Schiebeschalter in seine Endstellungen. Da sich der Hebel entlang einer Kreisbahn bewegt und sich der Schiebeschalter linear verschiebt, gleicht ein über den Zapfen des Schiebeschalters gestülpter Schuh die Ungenauigkeit aus. Die Drehachse von Hebel und Schuh bildet jeweils eine Schraube.

Um die Bauelemente so ungewöhnlich zu befestigen, haben wir diverse Bauteile entworfen und 3D-gedruckt. Alle CAD-Dateien der Designs und fertige STLs für den 3D-Drucker finden Sie über ct.de/wzts. Unsere Bauteile stammen allesamt aus der Bastelkiste. Wenn Sie einen eigenen Controller bauen wollen, müssen Sie die Designs also even-

tuell an Ihre Bauteile anpassen. Eine Einführung in OpenSCAD finden sie in [2], für einen Block am Trimmrad haben wir auch FreeCAD [3] verwendet. Wir haben ein Video zum Bau aufgenommen, das Sie ebenfalls über ct.de/wzts finden.

## Programmcode

Den Programmcode der Firmware für den Frederick finden Sie im GitHub-Repository über ct.de/wzts. Die Architektur ist Arduino-Standardkost: In `setup()` konfiguriert der Code die vielen GPIOs für die Schalter als Eingang. Die `loop()`-Funktion pollt die Schalter und speichert deren gemeldete Stellung zwischen, um per Software zu entprellen. Wegen der Prellung lohnt es sich auch nicht, auf Interrupts zu setzen; außerdem hat der Arduino ohnehin wenig zu rechnen. Den endlichen Automaten für den Drehencoder haben wir einfach vom Lautstärkereglerprojekt auf Seite 76 übernommen.

Um die Kommunikation mit dem MCP23017 per I²C kümmert sich die Wire-Bibliothek (siehe S. 68). Die Funktion `i2cDigitalRead(unsigned short pinNo)` in i2cMCP.cpp filtert einzelne Bits aus den Bytes, die der MCP23017 für seine Eingänge liefert. Der Code in der `loop()` in main.cpp nutzt `i2cDigitalRead()` und `digitalRead()` für die direkt am Arduino angeschlossenen Schalter im gleichen Format.

## Entprellen per Software

Wir haben uns die Widerstände und Kondensatoren fürs Entprellen der Schalter gespart und erledigen das per Software. Zur Erinnerung: Beim Prellen schaltet ein Schalter nicht einmal um, sondern wechselt in schneller Folge zwischen hohem und niedrigem Pegel, bis sich nach üblicherweise maximal 250 Millisekunden ein stabiler Pegel eingestellt hat. Da der Code die Schalter alle 25 Millisekunden abfragt, registriert er beim Umschalten ein wildes Auf und Ab, das sich nach einer Weile stabilisiert. Die Software macht bei dieser Hektik nicht mit, sondern wartet, bis sie acht Mal den gleichen Wert gelesen hat. Erst dann geht sie von einem gekippten Schalter aus. Das verzögert die Reaktion um circa 1/4 Sekunde – ausreichend schnell, dass es einem noch so vorkommt, als würde der Controller sofort reagieren.

Für das Warten auf stabile Pegel enthält der Code zusätzlich zur Stellung ein `history`-Byte für jeden Schalter. In C++ sieht das für den ersten Schalter so aus:

```
unsigned char switchA0History = 0;
bool switchA0State = false;
```

Ums Protokoll kümmert sich die Funktion `debounce()`. Damit sie mit jedem Schalter funktioniert, nimmt sie einen Zeiger auf ein `history`-Byte und einen Zeiger auf den Status an, außerdem den gerade gelesenen Wert (die Rückgabe von `i2cDigitalRead()` oder `digitalRead()`). Das `history`-Byte verwendet die Funktion, um 8 Werte als einzelne Bits zu speichern. Daher schiebt sie mit einer Shift-Operation (`<< 1`) alle Bits zunächst um eine Stelle nach links. Das älteste Bit fällt dabei weg und Bit 0 ist dafür eine 0. An dieser Stelle (Bit 0) speichert sie nun den gelesenen Wert. C stellt boolesche Variablen intern als Integer 0 oder 1 dar. Konvertiert man das zu einem Byte, kommen also nur die Werte `0b00000000` und `0b00000001` vor. Bit 1 bis 7 sind bei diesem Byte also immer 0. Verrechnet man die verschobene `history` also über ein bitweises Oder `|` mit dem Boolean, können sich Bit 1 bis 7 also nicht verändern. Da die verschobene `history` an Stelle 0 aber garantiert eine 0 stehen hat, schreibt das Oder dort nur die 0 oder 1 aus dem Boolean hin.

Mit einem so geführten Protokoll ist es einfach, auf eindeutige Schalterveränderungenzu warten. Solange in der History Nullen und Einsen gemischt vorkommen, bleibt `debounce()` beim alten Wert für `state`. Wenn aber nur Nullen oder nur Einsen in der `history` stehen, setzt die Funktion `state` neu und gibt auch den geänderten Wert zurück:

```
bool debounce(unsigned char* history,
 bool* state, bool reading) {
 *history = (((*history) << 1) |
                          (unsigned char) reading);
 switch (*history) {
 case 0b00000000: *state = false;
 break;
 case 0b11111111: *state = true;
 break;
 }
 return *state;
}
```

Der Code überwacht für alle Schalter, ob sich der `state` ändert und ruft bei Bedarf eine Funktion auf. Um die Überwachung kümmert sich `detectSwitchToggle()`. Um duplizierten Code zu vermeiden, nutzt sie ebenfalls Zeiger auf `history` und `state` und für den Funktionsaufruf beim Umschalten einen Zeiger auf eine Callback-Funktion:

```
void detectSwitchToggle(
 unsigned char* history,
 bool* state,
 bool reading,
 void (*callback)(bool state)) {
   bool oldState = *state;
   if (oldState != debounce(history,
                            state, reading)) {
     callback(*state);
   }
}
```

## Arduino tippt

Was beim Umschalten passiert, steht in der Datei keypresses.cpp. Die Reaktion aufs Umschalten des ersten Kippschalters sieht beispielsweise so aus:

```
void switchAO(bool state) {
 if (state) {
   Serial.println("AO ON");
   Keyboard.press(KEY_LEFT_ALT);
   Keyboard.press('A');
   delay(100);
   Keyboard.releaseAll();
 } else {
   Serial.println("AO OFF");
   Keyboard.press(KEY_LEFT_ALT);
   Keyboard.press(KEY_RIGHT_SHIFT);
   Keyboard.press('A');
   delay(100);
   Keyboard.releaseAll();
 }
}
```

Die Funktion nutzt das per `Keyboard.begin()` initialisierte Tastatur-Objekt aus der HID-Project-Bibliothek. Der Arduino „tippt“ hier Tastenkombinationen: Ein `Keyboard.press()` entspricht dem Herunterdrücken einer Taste einer Tastatur. Die Funktion nimmt entweder Integer-Konstanten für Sondertasten wie `KEY_LEFT_ALT` entgegen (die Arduino-Doku zu Keyboard.h listet auf, welche Konstanten zur Verfügung stehen, siehe ct.de/wzts) oder druckbare Zeichen wie `'a'` oder `'O'`. Die Funktion `Keyboard.releaseAll()` lässt alle Tasten wieder los, allerdings erst nach der mit `delay(100)` erzwungen Wartezeit von 0,1 Sekunden.

## Abfrageintervall

Um eine schnelle Reaktion auf Eingaben zu garantieren, verzichtet der Code in `loop()` auf Aufrufe von `delay()`. Stattdessen protokolliert er in Timern wie `unsigned long debounceTimer`, wann die letzte Interaktion stattfand. Es reicht dann ein Vergleich mit zwischen der von `millis()` gelieferten Zeit mit der um ein Offset erhöhten Zeit der letzten Interaktion:

```
if (millis() > debounceTimer + 25) {
 debounceTimer = millis();
 detectSwitchToggle(
    &switchAOHistory,
    &switchAOState,
 digitalRead(SWITCH_A_O_PIN),
    switchAO);
}
```

Ohne Warten braucht der Arduino weniger als 5 Millisekunden für einen Durchlauf der `loop()`. Die Schalter fragt der Code aber frühestens alle 25 Millisekunden ab, sodass fürs Entprellen genug Zeit verstreicht.

Stattdessen hätten wir auch mit je einer 4 Byte langen `history` arbeiten können, 8 gelesene Werte im Abstand von 25 Millisekunden reichten im Test aber bereits für ein fehlerfreies Entprellen.

Das gleiche Vorgehen mit einem anderen Timer nutzt der Code auch, um den Status aller Schalter außer dem Trimmrad alle 20 Sekunden auf die serielle Schnittstelle zu schreiben. Zum Debuggen ist die Ausgabe praktisch und den Flugsimulator stört das zusätzliche serielle Interface nicht.

## Spaßige Bastelei

Egal, ob Sie den fliegenden Frederick nachbauen oder nur als Ideengeber nutzen: Die Bastelei macht Spaß, weil sie viel Modellbau umfasst und die überschaubare Elektronik wenig Stolperfallen auslegt. Das Grundprinzip bleibt nämlich für alle Varianten gleich: Der billige Arduino Micro übernimmt die USB-Kommunikation als HID und tippt als virtuelle Tastatur bei jeder Hebelbewegung. Für eine Handvoll Schalter reichen die Arduino-GPIOs aus, sodass Sie sich für kleine Interfaces den MCP23017 und die Platine sparen können. Wir haben am fliegenden Frederick mit Unterbrechungen circa sechs Wochen lang gebastelt. Planen Sie also ausreichend Zeit für das Projekt ein. Dafür dürfen Sie sich auf doppelte Freude einstellen: zuerst Bastelspaß, bis alle Spezialschalter funktionieren, und danach der Spaß am Fliegen mit jeder Menge haptischen Schaltern. (pmk) c't

### Literatur

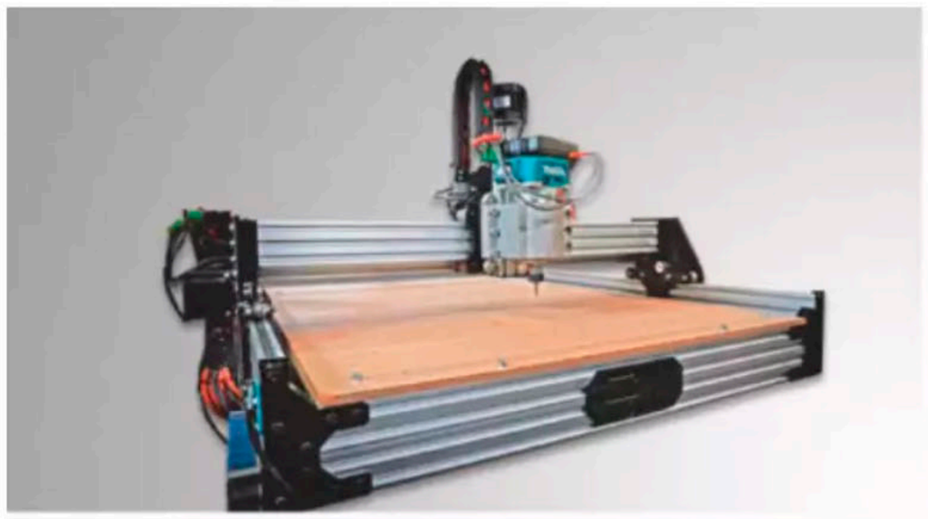

[1] Tim Gerber, **Anschläge**, PC-Programme mit Arduino steuern, c't 8/2016, S. 156

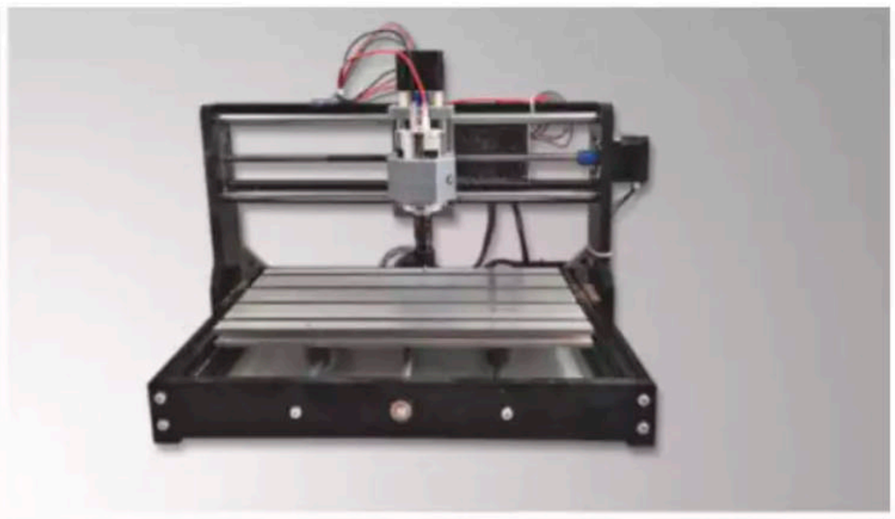
[2] Pina Merkert, **OpenSCAD**, Modelle für den 3D-Druck programmieren, c't 12/2017, S. 172

[3] Christoph Schmith, **CAD-Content**, 3D-Konstruktion mit FreeCAD, c't 13/2018, S. 140

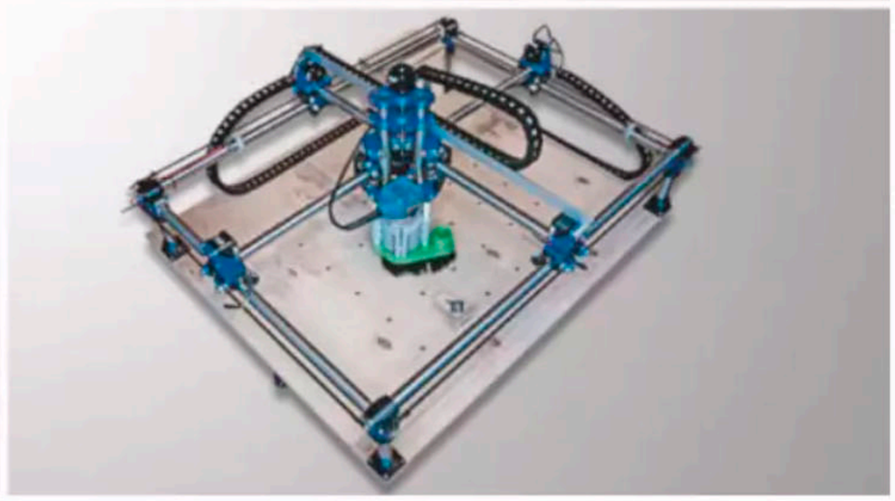

**Repository bei GitHub, 3D-Designs, Link zum Video**

ct.de/wzts











Spindelantriebe

# Die optimale CNC-Fräse

**300 Euro bis 300.000 Euro kann man für eine CNC-Fräse ausgeben. Wir klären, welche Maschine Sie für Ihre Werkstatt wirklich brauchen und wo Sie durch Selbstbau sparen können.**

Von **Jan Mahn und Pina Merkert**


| | |
|---|---|
| Kaufberatung CNC-Fräse | 86 |
| MPCNC gegen OX-CNC | 90 |
| G-Code programmieren und generieren | 96 |
| Eigene Portalfräse selbst ausdrucken | 102 |
| Stabile Portalfräse OX-CNC selbst bauen | 110 |
| Optimale Parameter für die CNC-Fräse | 114 |
| CNC-Oberfräse mit Shaper Origin | 122 |

CNC-Fräsen sind nützliche Werkzeuge für die Bastel- und Elektronikwerkstatt. Wer etwa Gehäuse, Frontplatten, Schilder oder andere Projekte aus Holz, Kunststoff oder Metall plant, bekommt von einer Maschine, die Computer-Numerically-Controlled (CNC) ist, präzise bearbeitete Werkstücke, nachdem er die Fräsbahnen in der Programmiersprache G-Code programmiert hat (siehe Seite 96).

Ein schnell rotierender Fräser entfaltet aber nicht unerhebliche Kräfte auf die Mechanik, wenn diese ihn gegen den Widerstand des Werkstücks auf die festgelegte Position zwingt. Die Mechanik muss daher ausreichend robust sein. Höhere Anforderungen an die Steifigkeit bedeuten aber schwerere Rahmenteile, diese erfordern stabilere und schwerere Lager und stärkere Motoren, die ihrerseits ebenfalls mehr wiegen und deshalb einen steiferen Rahmen erfordern – ein Teufelskreis. Jede CNC-Fräse ist daher ein Kompromiss.

## Riesige Bandbreite

Die Preisspanne für CNC-Maschinen ist riesig: Eine Saintsu 3018 Graviermaschine bekommt man bei Amazon für weniger als 300 Euro, für ein industrielles CNC-Bearbeitungszentrum zahlt man mehr. Beispielsweise kostet eine Tormach 1100mx 21.500 Euro oder eine Knuth PFG 2513 55.700 Euro. Die billige CNC-Graviermaschine passt noch auf den Schreibtisch, rattert aber bereits, wenn millimeterdünne Fräser durch weiches Holz pflügen, während die teure Maximallösung auch Edelstahl mit flottem Vorschub zerspant und automatisch das jeweils perfekte Werkzeug einspannt. Von beiden Extremen sollten Sie als Hobbyanwender die Finger lassen: Die billigen Graviermaschinen sind zu klein, zu schwach und zu ungenau für die meisten Projekte, das Bearbeitungszentrum sprengt das Budget und passt nicht in die Werkstatt.

Aber auch im für Hobbyisten und Holzwerker interessanten Zwischenbereich gibt es ein großes Angebot. Kommerzielle Portalfräsen für kleine Schreinereien fangen bei Modellen wie der Stepcraft M700 für 2400 Euro (Preis ohne Steuerung) [1], CNC-Step High-Z S-750 für 2000 Euro oder Hammer HNC 47.82 für 3700 Euro an. Sie bearbeiten Material von mindestens 50 × 50 Zentimetern. Auf automatische Werkzeugwechsler muss man in dieser Preisklasse verzichten, aber die Stabilität genügt für alle Arten von Holz und Kunststoffen. Leichtmetalle wie Aluminium und stabile Kunststoffe wie Delrin können Sie ebenfalls auf solchen CNC-Fräsen bearbeiten, allerdings mit reduzierter Geschwindigkeit.

Viele Hobby-Handwerker wünschen sich eine noch günstigere Portalfräse, die groß genug für kleine Möbelbauprojekte ist und gelegentlich auch Aluminium bearbeitet. Wer nicht allzu tief in die Tasche greifen möchte, landet schnell bei Anleitungen für Selbstbauprojekte. Zwei sehr unterschiedliche Selbstbaufräsen erfreuen sich dabei besonderer Beliebtheit: Die OX-CNC setzt auf das Prinzip einer klassischen Portalfräse mit Riemenantrieb, unterbietet Fertiggeräte dank vieler Standardteile aber im Preis und kostet nur rund 1200 Euro. Noch günstiger ist die Mostly-Printed-CNC (MPCNC), die auf einen anderen Aufbau setzt und den Fräsmotor über zwei sich kreuzende Portale verfährt. Den besonders günstigen Preis von 600 Euro erreicht sie, weil man die meisten Teile selbst mit einem 3D-Drucker herstellt.

## Antriebsstrang

CNC-Fräsen sollten sich in X- und Y-Richtung möglichst spielfrei bewegen. Damit sich Fräser beim Einschneiden ins Material nicht zum ungewollten Antreiber verwandeln, lassen sich Trapezspindeln hier nur mit einer zweiten Mutter nutzen, die durch eine Feder gespannt wird. Kleine Graviermaschinen nutzen dieses Prinzip, doch es kommt schon bei geringen Belastungen zu Vibrationen durch die Feder.

Riemenantriebe des Typs GT2 oder GT3 haben abgerundete Zähne. Die erlauben zwar weniger Kraftübertragung als keilförmige Zähne, dafür legen sich die Zähne so in die Rillen der Riemenscheiben, dass Vorder- und Hinterkante gleichzeitig anliegen. Das garantiert einen spielfreien Antrieb. Riemen können sich jedoch ein kleines bisschen dehnen, was sich im Fräsbild durch Rattermarken zeigen kann.

Ein spielfreier Spindelantrieb funktioniert nur mit teuren Kugelumlaufspindeln (siehe Kasten S. 88 „Eine Frage der Spindel“). Die bieten nur dann einen Mehrwert, wenn die Steifigkeit des Rahmens der Fräse mithalten kann. Bestehen Rahmenteile aus Alu oder Plastik, lohnen sich Kugelumlaufspindeln nicht.

## Rahmensteifigkeit

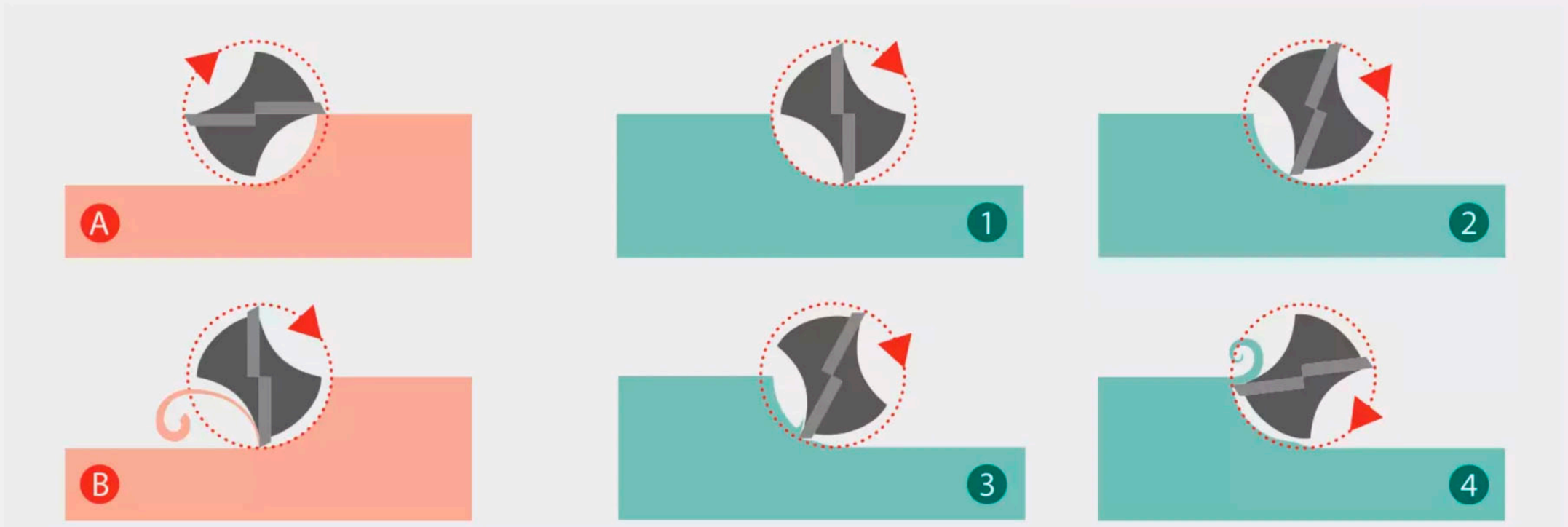
Gegen die Kräfte beim Zerspanen muss eine Fräse mit einem möglichst steifen Rahmen wirken. Gerät er ins Schwingen, entstehen Rattermarken auf dem Werkstück. Die geeignete Gegenmaßnahme ist, den Vorschub zu reduzieren. Bei den meisten Graviermaschinen vom Typ 3018 etwa haben die-

# Eine Frage der Spindel

Handbetriebene Fräsen nutzen oft günstige Trapezspindeln, die prinzipbedingt ein Umkehr-Spiel aufweisen. Je nach Richtung, in der man die Spindel kurbelt, läuft die Mutter links oder rechts an der Flanke an. In zwei Richtungen ist eine Fräse mit Trapezspindeln also stets stabil geführt. Kurbelt man beispielsweise nach links und nach vorne, darf der Fräse in diese Richtungen große Kräfte ausüben. Würde der Fräser allerdings eine Kraft nach rechts oder hinten ausüben, könnte er den Fräskopf um das Spiel verschieben und sich mit dieser plötzlichen Bewegung zu tief ins Material hineinfressen.

Außerdem sollte man auf manuellen Fräsen mit Trapezspindeln darauf achten, im Gegenlauf, also gegen den Uhrzeigersinn zu fräsen. Bei dieser Fräsrichtung ergeben sich nämlich immer Kräfte in Richtung der Bewegungsrichtung der Spindeln, sodass das Spiel keine Rolle spielt. Komplizierte Formen wie eine Kreisbahn werfen aber ein Problem auf: Wie das Spindelspiel an den Scheitelpunkten überwinden? Man könnte das Spiel zwar ausmessen und versuchen, es exakt am Scheitel zu kompensieren. Würde der Fräser sich aber aufgrund einer kleinen Ungenauigkeit doch ins Material fressen, würde er verreißen und im schlimmsten Fall brechen.

Nahezu spielfreie Antriebe mit Riemen oder Kugelumlaufspindeln lösen nicht nur dieses Problem, sie erlauben auch Gleichlauffräsen. Dabei fährt der Fräser im Uhrzeigersinn und trifft mit steilem Winkel, im Idealfall mit 90° aufs Material A. Bei einem steilen Winkel kann der Fräser nicht ausweichen und muss direkt beim Auftreffen aufs Material einen Span abschälen B. Warum das bessere Oberflächen erzeugt, zeigt der Blick auf die Alternative: Beim Gegenlauffräsen trifft die Schneide in einem extrem flachen Winkel auf das Material 1. Zunächst reibt die Schneide auf der Oberfläche entlang, ohne einzudringen 2. Diese Reibung erhitzt den Fräser, wodurch er im schlechtesten Fall überhitzt, weich und danach zwangsläufig stumpf wird. Dreht sich der Fräser weiter, werden die Kräfte an der Schneide aber auch beim Gegenlauffräsen so groß, dass die Schneide ins Material eindringt 3. Ab hier schält der Fräser einen Span ab 4. Je steifer eine Fräse konstruiert ist, desto kleiner die Fläche, wo der Fräser nur reibt und desto besser das Schnittbild.



Führungen nur einen Durchmesser von 8 Millimetern, sind aber gut 30 Zentimeter lang. Sie verbiegen sich deshalb schon bei geringer Belastung um einige Zehntel Millimeter. Nutzt man eine solche Graviermaschine zum Fräsen, führt das unweigerlich zum Rattern. CNC-Maschinen benötigen möglichst steife Führungen, was in der Praxis mehr Durchmesser oder Breite und damit mehr Gewicht bedeutet.

Die Selbstbau-Fräsen OX-CNC und MPCNC kann man in mehreren Größen bauen, was großen Einfluss auf die Rahmensteifigkeit hat. Wer vor allem Aluminium fräsen will, sollte beide Fräsen möglichst klein bauen. Aber auch Holzwerker sollten überlegen, ob sie wirklich eine Riesenfräse brauchen. Die Rohre der MPCNC können in der Mitte ins Schwingen geraten, wenn sie sich über mehr als einen Meter strecken müssen. Die Profile der OX-CNC bieten wegen des größeren Querschnitts mehr Stabilität, schwingen aber auch irgendwann. Da der Hebel zum Rand kleiner wird, empiehlt es sich, Metallteile in einer Ecke aufzuspannen.

Wer von allem mehr will, findet auch Selbstbau-Portalfräsen mit Stahlwangen, Kugelumlaufspindeln und Nema32-Schrittmotoren. Diese sprengen allerdings allein mit den Bauteilen schon ein Budget von 3000 Euro.

## Fräser

Ein vierschneidiger Fräser mag auf einer stabilen Metallfräse millimeterdicke Späne in den Kühlmittelstrom katapultieren, auf den ab Seite 90 getesteten Portalfräsen wird man mit der gleichen Vorschubgeschwindigkeit aber nur Rattermarken und Frust ernten. Um auf den beiden Maschinen Alu zu fräsen, sollten Sie die Schnittkräfte mit einschneidigen Fräsern und einem deutlich geringeren Vorschub minimieren.

Mit Wirbelfräsen (auch als Trochoidialfräsen bekannt), können Sie beim Schruppen Zeit sparen: Dabei fährt der Fräser gleich mit einer kleinen Rampe tief ins Material (bis zum Dreifachen des Durchmessers). Mit hoher Drehzahl arbeitet er sich dann stückweise auf Kreisbahnen vor, sodass der Fräser immer nur kurz im Teilschnitt Material abnimmt.

## Maschinenwahl

Bei der Wahl einer CNC-Fräse sollten Sie zuerst eine Schmerzgrenze fürs Budget festlegen und überlegen, wie viel Platz Sie der Maschine in der Werkstatt zugestehen können. Danach sollten Sie überlegen, ob ein Selbstbau infrage kommt, da ein solches Projekt zwar viel Geld spart, aber viel Zeit kostet. Haben Sie diese drei Fragen geklärt, stehen meist nur noch wenige Maschinen zur Auswahl.

Viele Maschinen lassen sich in kleinen Wohnungen gar nicht aufstellen. Wer Geld, aber keinen Platz hat, sollte einen Blick auf den SnapMaker 2.0 werfen, einer Kombination aus 3D-Drucker, CNC-Fräse und Lasercutter für 1800 Euro. Gibt es einen Werkstattraum, bieten Selbstbaufräsen ein gutes Preis-Leistungs-Verhältnis. Der Test auf Seite 90 vergleicht zwei Selbstbaufräsen aus dem unteren Preissegment.

Stabile Portalfräsen wie eine Killer Bee oder Stepcrafts Q-Serie lohnen sich nur, wenn man großes Metall-Plattenmaterial verarbeiten möchte. Wer nur gelegentlich Teile braucht und sich nicht allzu tief einarbeiten will, sollte überlegen, stattdessen einen der zahlreichen Online-Fertiger wie laserteileonline.de zu beauftragen. Die liefern meist auch an Privatpersonen und häufig auch Einzelstücke. (pmk) c't

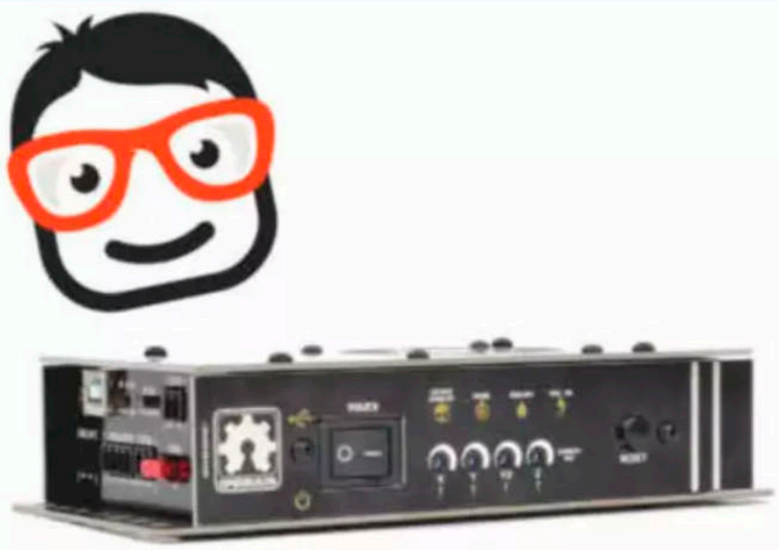
### Literatur

[1] Carsten Meyer, **Stepcraft M.700**, Make Sonderheft 2020, S. 114

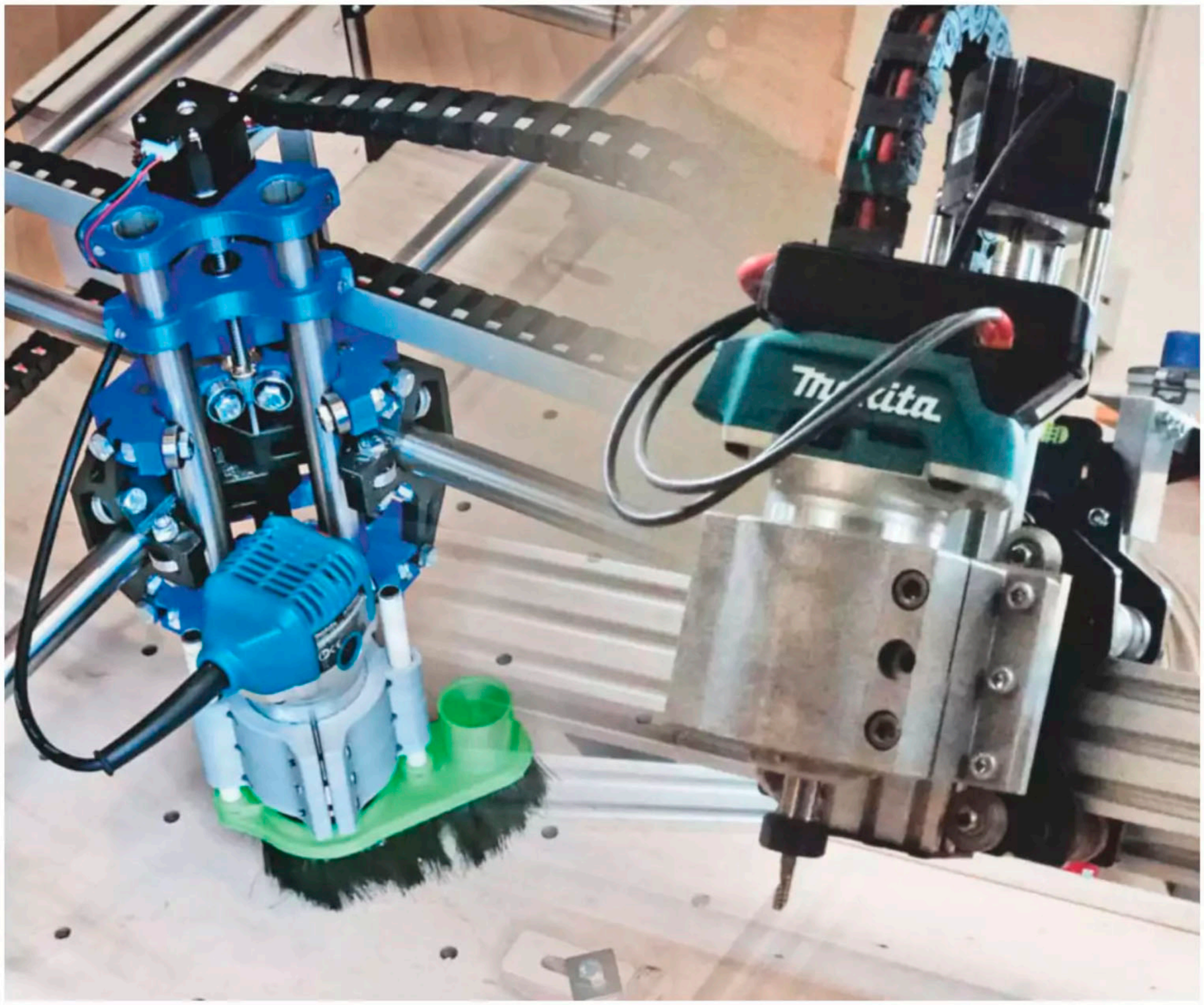
**Bauanleitung, erweiterte CAD-Designs, Druckvorlagen**

ct.de/wm4s

3D CNC-Steuerungen mit USB · OpenBuilds Fräsen und Lasercutter · BlackBox · LEAD CNC · Workbee




CONCUON · Billhorner Brückenstraße 40 · 20539 Hamburg · info@conucon.de · 040 75679167000 · www.conucon.de



# MPCNC gegen OX-CNC

**CNC-Fräsen kann man selbst bauen und damit gegenüber kommerziellen Maschinen viel Geld sparen. Wir haben zwei der beliebtesten Selbstbaufräsen gegeneinander antreten lassen.**

Von **Jan Mahn und Pina Merkert**

Anstatt viel Geld in eine CNC-Portalfräse zu stecken, können Sie daraus ein Selbstbauprojekt machen und kräftig Geld sparen. Der Zeitaufwand ist allerdings nicht unerheblich, dafür sinkt die Einstiegsschwelle von 3000 auf 600 Euro. Im Netz finden sich dutzende Konstruktionen für CNC-Fräsen, allerdings wurden viele davon nur einmal vom Konstrukteur gebaut und die Bauanleitung ist oft lückenhaft. Zwei der Open-Hardware-Designs ragen aber aus der Masse heraus, weil sie schon oft nachgebaut wurden, eine gute Bauanleitung mitbringen und sich um beide Projekte eine rege Community gebildet hat.

Die Mostly Printed CNC (MPCNC) spricht Besitzer eines 3D-Druckers an. Auf dem druckt man komplexe Verbinder und verbaut ansonsten nur günstige Standardteile wie Edelstahlrohre, Schrauben und Skateboard-Kugellager. Der Antriebsstrang mit GT2-Riemen und Nema-17-Schrittmotoren stammt aus dem 3D-Drucker-Bedarf, einem Massenmarkt mit günstigen Preisen. Nach unserer Recherche ist die MPCNC die billigste Fräse, mit der sich Teile in sinnvollen Größen fertigen lassen. Billigere Maschinen gehören eher in die Kategorie „Graviermaschine“ mit starken Einschränkungen bei Spindelleistung, Größe und Genauigkeit. Unsere Erfahrungen beim Bau einer MPCNC haben wir ab Seite 102 ausführlich beschrieben.

Bei der OX-CNC sind alle Komponenten eine Nummer größer: Hier treiben Nema-23-Schrittmotoren mit mehr als 2 Ampere dickere GT3-Riemen an. Den Zusammenbau haben wir ab Seite 110 beschrieben. Statt auf Rohren rollen bei der OX-CNC etliche Kunststoffräder in Aluminiumprofilen mit V-Nuten. Kombiniert mit 5 Millimeter dicken Aluminiumplatten erreicht die OX-CNC eine hohe Rahmensteifigkeit. Sie setzt dabei auf die klassische Konstruktion mit einem in Y-Richtung verfahrbaren Portal, statt wie die MPCNC den Fräskopf über kreuzende Rohre zu positionieren. Die OX ist wegen ihres Konstruktionsprinzips erheblich schwerer und fast doppelt so teuer. Beide Maschinen lassen sich in verschiedenen Größen bauen; bei der MPCNC muss man prinzipbedingt bei einem sehr großen Rahmen mit stark reduzierter Steifigkeit rechnen, während die OX-CNC auch in groß noch relativ stabil bleibt.

Beide Maschinen setzen auf einen spielfreien Riemenantrieb für X- und Y-Achse. Die Z-Achse treibt jeweils eine Trapezspindel an. Als Fräsmotor eignen sich diverse Oberfräsen, die hier getesteten Maschinen nutzen gleichwertige Motoren von Makita: RT0700C oder DRT50Z, die einstellbar mit 10.000 bis 30.000 Umdrehungen pro Minute rotieren. Beide Fräsen passen physisch noch in viele Kellerräume oder Bastelzimmer und finanziell mit 600 Euro (MPCNC) und 1200 Euro (OX-CNC) ins Budget von Hobbyisten.

## Bastler getestet

Bei einem normalen Produkttest fordern wir beim Hersteller Testgeräte an und prüfen deren Qualität in den c’t-Laboren in dem Zustand, wie sie aus dem Karton kommen. Eine Selbstbaufräse funktioniert aber nicht, ohne dass wir viele Stunden Arbeit investieren. Die Qualität dieser Arbeit und die Qualität der Maschinen, die wir dabei einsetzen, haben einen Einfluss auf die Qualität der Maschinen, die wir zusammengebaut haben. In diesem Test tritt die Ingenieursleistung der MPCNC-Entwickler kombiniert mit den handwerklichen Fähigkeiten von Redakteur Jan Mahn gegen die Ingenieurleistung der OX-CNC-Konstrukteure kombiniert mit den Bastelfähigkeiten von Redakteurin Pina Merkert an.

## Exaktheit

Was eine CNC-Fräse produzieren kann, hängt davon ab, wie exakt sie den Fräser positioniert. Da die Maschinen Schrittmotoren verwenden, hängt die Genauigkeit von der Anzahl der Schritte der Motoren und der Übersetzung des Antriebs ab. Die OX-CNC fährt in X- und Y-Richtung mit 14 Schritten pro Millimeter, in Z-Richtung dank des Spindelantriebs mit 50 Schritten pro Millimeter. Die MPCNC fährt X- und Y-Positionen dagegen mit 50 Schritten pro Millimeter und Z-Positionen mit 100 Schritten pro Millimeter an. Die MPCNC verdankt die höhere Schrittzahl ihren kleineren Riemenscheiben und den Schrittmotortreibern, die 32 Mikroschritte ansteuern. Unsere OX-CNC nutzt Treiber mit nur 16 Mikroschritten.

Mehr Schritte pro Millimeter erlauben aber nicht automatisch auch präzisere Teile. Biegt sich der Fräser um 0,1 Millimeter weg, hilft es nicht, wenn die Maschine im Prinzip auf 0,02 Millimeter genau positionieren kann. Wir konnten trotz der relativ groben Positionierung auf 0,07 Millimeter keine Stufen an den Fräsbahnen der OX-CNC erkennen, auch die MPCNC erzeugt keine mit bloßem Auge sichtbaren Stufen.

Ein Indikator für Ungenauigkeiten im Antrieb ist die Wiederholgenauigkeit. Niemandem ist geholfen, wenn die Fräse am Ende einer Bewegung nicht wie-




**Eine Messuhr, befestigt an der Spindel, diente zum Test der Wiederholgenauigkeit. Die Fräsen mussten einen Parcours fahren und wieder exakt zum Ausgangspunkt zurückkehren.**

der exakt am Startpunkt ankommt, sondern unterwegs vom rechten Fräspfad abkommt. Um die Wiederholgenauigkeit präzise zu testen, haben wir einen kleinen Versuchsaufbau vorbereitet und mit einem eigens dafür konstruierten Halter eine Messuhr an der Frässpindel installiert. Die Messuhr haben wir an einen Metallblock gefahren, dort den Nullpunkt in der CAM-Software abgenommen, die Messuhr genullt und die Fräse auf einen Testparcours geschickt. Nach einigem Umherfahren in alle drei Achsen musste die Maschine exakt zum selben Punkt zurückkehren. Die OX-CNC konnte auf der X-Achse mit 0,01 mm und auf der Y-Achse mit 0,015 mm Abweichung zum gleichen Punkt zurückkehren. Auch die MPCNC schlug sich in dieser Aufgabe hervorragend mit 0,01 mm in X- und 0,02 mm in Y-Richtung.

Mit diesen Messungen haben wir auch den mit diesem Aufbau realistisch messbaren Bereich erreicht. Das Fazit: Trotz Plastikteilen bei der MPCNC und Riemenantrieben bei beiden Fräsen landete die Frässpindel wieder exakt am Ausgangspunkt. Die Wiederholgenauigkeit der Z-Achse haben wir nicht gemessen, da die spindelgetriebene Z-Achse der Maschinen ohnehin sehr genau und steif ist. Wenn Sie unsere Tests nachvollziehen wollen, können Sie den G-Code über ct.de/wujb herunterladen.

Der Test für die Wiederholgenauigkeit fand nicht unter Last statt. Wir haben bei den Testfräsungen deswegen auf Stufen zwischen den Tiefenschritten geachtet. Bei Fräser-breiten Nuten im Aluminium hätte sich bei ungenügender Wiederholgenauigkeit am linken oder rechten Rand der Nut eine Kante zeigen müssen. Eine solche war bei beiden Maschinen nicht erkennbar. Dieses gute Ergebnis gilt, solange die X- und Y-Motoren keine Schritte verlieren, dann wäre nämlich plötzlich die ganze Fräsbahn verschoben und das Werkstück vermurkst.

## Rahmensteifigkeit

Der größte Unterschied zwischen OX-CNC und MPCNC ist das mechanische Prinzip. Die MPCNC geht da ganz eigene Wege, weil sie ihre Z-Achse über zwei sich kreuzende Rohre positioniert. Statt eines Portals bewegt sie zwei, wodurch die Y-Motoren die X-Motoren nicht mitbewegen müssen. Durch das geringere Gewicht reichen ihr kleine Nema-17-Schrittmotoren und kleinere Treiber. Der Querschnitt der Rohre in der MPCNC ist aber quer und längs kleiner als der der Nutprofile in der OX. Ob die geringere Stabilität und Masse zu Vibrationen und damit zu Rattermarken führt, hängt in der Praxis von den gespannten Fräsern und den gewählten Einstellungen ab.

Wir haben uns für den Test hochwertige HSS-Fräser mit 6 Millimeter Durchmesser bestellt und beide Maschinen jeweils mit einem fabrikneuen Fräser getestet. Für Holz haben wir einen geraden Fräser mit zwei Schneiden benutzt. Der Verzicht auf eine gewendelte Nut verschlechtert zwar die Spanabfuhr, dafür gibt es aber keine Ausrisse, sondern eine glatte Schnittkante. Wer mit solchen Fräsern arbeitet, muss sich um angemessene Absaugung kümmern, damit der Fräser freie Bahn hat. Beim Holz-Test zerspanten wir maximal 6 Millimeter Holz in einem Tiefenschritt.

Für Aluminium haben wir einen gewendelten Einschneider benutzt. Nur eine Schneide im Eingriff zu haben, begrenzt zwar den horizontalen Vorschub (wir frästen mit gemächlichen 200 Millimetern pro Minute), belastet die Maschine aber weniger. Sowohl die MPCNC als auch die OX sind primär für die Holz- und Kunststoffbearbeitung gedacht, mit den richtigen Werkzeugen lässt sich aber auch mal Leichtmetall fräsen.

Wir frästen je eine L-Förmige Nut in Birke-Multiplex, Eichenholz und AlMg4,5Mn, eine gut zerspanbare Aluminiumlegierung. Die Fräsen mussten dabei beweisen, dass sie die gleiche Form mit dem Fräser in verschiedenen Tiefen abfahren können. An der Wand der Nut sollten sich die Tiefenschritte möglichst gar nicht abzeichnen. Eine Kante des L frästen

wir auch in zwei Schritten, um sehen zu können, ob der Fräser ein schöneres Schnittbild liefert, wenn er nur auf einer Seite schneiden muss (Teilschnitt).

Die OX-CNC schlug sich im Holz absolut hervorragend und zerspante Multiplex und Eichenholz ähnlich mühelos mit 2, 3 und 6 Millimetern Tiefenzustellung. Das eigentlich 10 Millimeter breite L fräste sie in der Birke nur 0,2 Millimeter größer als gewollt aus – für Tischlerarbeit eine hervorragende Präzision. Auch das L aus der MPCNC war für Holzprojekte mehr als ausreichend genau. Die Abweichung lag bei 0,3 mm.

Am Eichenholz konnten wir sogar nur 0,05 Millimeter Übermaß bei der OX messen. Die Oberflächen der Nut waren nach allen Fräsungen angenehm glatt. Bei Eiche konnten wir aber eine geringfügig bessere Oberfläche an jener Kante erfühlen, wo der Fräser nur auf halber Breite Material entfernen musste. Mit einem Schlichtdurchgang produzierten OX und MPCNC in Vollholz perfekte Oberflächen.

In Aluminium frästen wir zunächst Nuten in ein Teststück, um Tiefenzustellungen und Vorschubgeschwindigkeiten auszuprobieren. Die Spindeldrehzahl der Makita-Fräsmotoren lässt sich nur zwischen 10.000 und 30.000 Umdrehungen pro Minute einstellen, was bei einem 6-mm-Fräser selbst auf geringster Geschwindigkeit zu leicht zu hohe Schnittgeschwindigkeiten führt. 0,2 Millimeter Tiefenzustellung und 200 Millimeter pro Minute Vorschub lieferten dennoch sehr zufriedenstellend geschlichtete Oberflächen. Ein Vorschub von 300 mm/min hinterließ wegen der einzelnen Schneide und der geringen Drehzahl schon spiralförmige Kratzer im Boden der Nut, während mehr als 0,2 Millimeter Tiefenzustellung auf der OX bereits leichte Rattermarken erzeugten. Auch mit der MPCNC sollte man es mit der Zustelltiefe nicht übertreiben, wenn man perfekte Oberflächen wünscht. Wenn man bemerkt, dass sich die Fräse bei einer bestimmten Geschwindigkeit aufschaukelt, lohnt es sich, etwas mit dem Parameter zu experimentieren.

Für den Test gönnten wir uns die Zeit und frästen mit den getesteten Werten das gesamte L im Schlicht-Modus. Das Ergebnis war eine zufriedenstellende Oberfläche ohne sichtbare Rillen in der Tiefe.

Nimmt man Rattermarken in Kauf, kann man mit beiden Fräsen mit aggressiveren Einstellungen Zeit sparen. Wir empfehlen bei Einschneidern den Vorschub bei circa 200 mm/min zu belassen und die Tiefenzustellung zu erhöhen. Wir konnten bei beiden Maschinen mit Trochoidialfräsen 6 mm in einem Tiefenschritt wegschruppen. Mit der Schneidenzahl



Es gibt 10 Arten von Menschen.
iX-Leser
und die anderen.

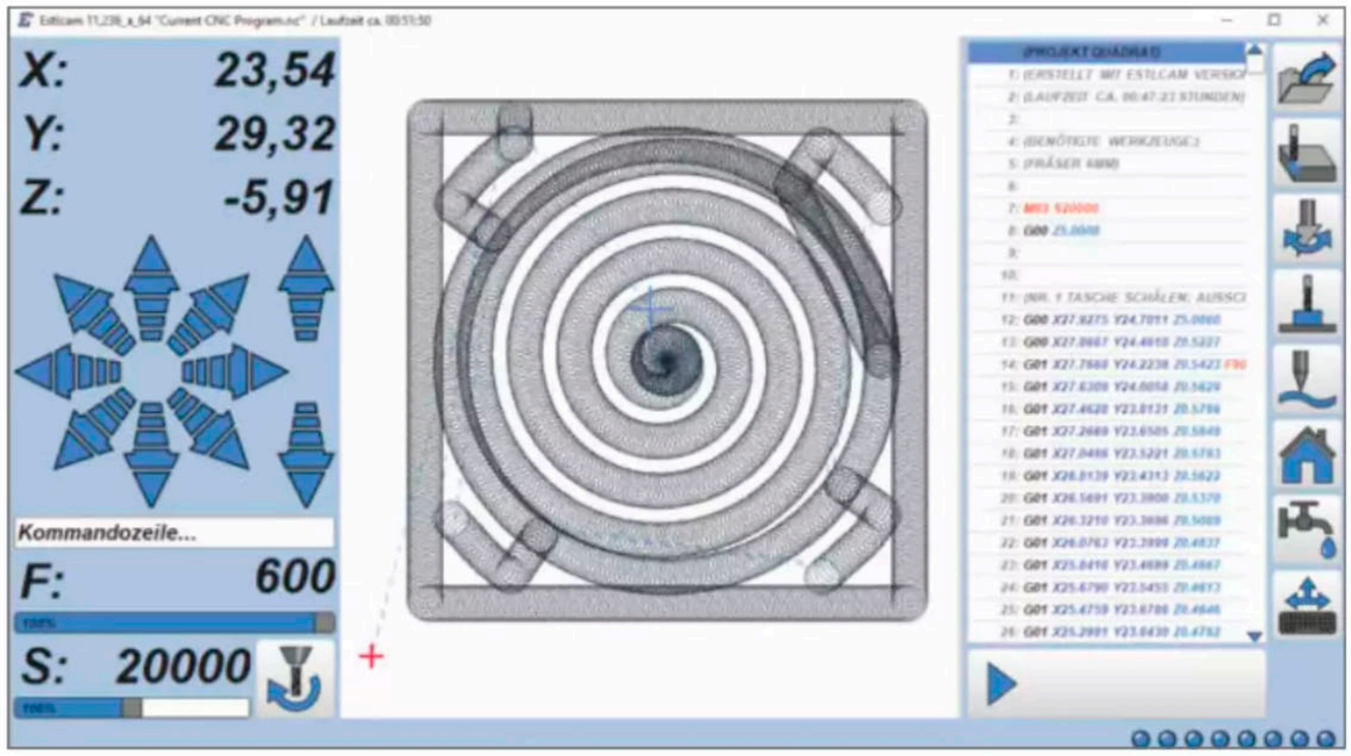



Jetzt Mini-Abo testen:
3 digitale Ausgaben +
Bluetooth-Tastatur
nur **16,50 €**

www.iX.de/digital-testen

iX MAGAZIN FÜR PROFESSIONELLE INFORMATIONSTECHNIK

www.iX.de/digital-testen
leserservice@heise.de
49 (0)541 800 09 120




**Sieht komisch aus, ist aber eine effektive Alternative für die Alu-Bearbeitung. Beim Wirbelfräsen berührt der Fräser immer nur kurz das Material. Dafür arbeitet man mit hohen Zustelltiefen und erzeugt schöne lange Späne.**

des Fräsers können Sie auch den horizontalen Vorschub multiplizieren. Für höchste Oberflächenqualität fahren Sie mit dem Einschneider noch mal langsam hinterher und nehmen sowohl horizontal als auch vertikal die letzten 0,2 Millimeter ab, um aufs Endmaß zu kommen.

Wie gut die Fräsen jeweils mit der Belastung zurechtkommen, kann man am Geräusch fast besser erkennen als am Schnittbild. Wir haben die Testfräsungen deswegen gefilmt und stellen Ihnen das Videomaterial ungeschnitten und ungefiltert zur Verfügung. Sie finden die Links zu den Videos bei YouTube über ct.de/wujb.

## Fazit

Sowohl die OX-CNC als auch die MPCNC eignen sich als kleine Portalfräsen für die Holzwerkstatt. Sie genügen auch für gelegentliche Leichtmetallarbeiten, wenn man sich genug Zeit lässt. Der Selbstbau beider Fräsen ist ein umfangreiches Projekt, das sich eher für erfahrene Bastler eignet. Unsere Bau-Artikel ab Seite 86 und Seite 90 geben entscheidende Tipps. Dank guter Anleitungen gelingt die Fertigstellung aber auch dann, wenn man noch nie eine CNC-Maschine gebaut hat. Da beide Konstruktionen eine Oberfräse als Fräsmotor einsetzen, scheitern Projekte nicht wie bei Mini-Graviermaschinen an der Spindelleistung.

Die Wiederholgenauigkeit beider Fräsen ist im Hundertstel-Bereich und die Positionierung ist selbst fürs Fertigen von Teilen aus Aluminium genau genug. Beide Maschinen liegen in etwa gleichauf, was uns angesichts der sehr unterschiedlichen Konstruktion durchaus überraschte. Bei der Rahmensteifigkeit stoßen sie an ihre Grenzen. Unsere Tests zeigen auch, dass es bei den Eigenbaufräsen darauf ankommt, die individuellen Fräsparameter für die eigene Maschine zu ermitteln.

Wer bereits einen 3D-Drucker besitzt, kann die MPCNC für nur 600 Euro nachbauen, was dieser Maschine zum in unseren Augen besten Preis-Leistungs-Verhältnis verhilft. Die billigeren Graviermaschinen sind schlicht zu klein und zu schwach für ernsthafte Holzprojekte. Die mindestens 1000 Euro teure OX-CNC ist die Alternative für alle, die ein wenig mehr Rahmensteifigkeit und etwas stärkere Motoren wünschen. Die OX hat mehr Potenzial, größere Vorschübe zu nutzen, was sich aber erst ab einer kleinen Serienfertigung lohnt. Wer Einzelstücke fräst, spart Zeit in der CAM-Software, indem er die konservativen Voreinstellungen übernimmt und somit früher loslegt. In diesem Workflow kann die OX den Vorteil ihrer stabileren Konstruktion nicht ausspielen. Trotzdem unterbietet die OX-CNC preislich alle vergleichbaren Fertigmaschinen, das Preis-Leistungs-Verhältnis stimmt hier.

Die Entscheidung zwischen MPCNC und OX-CNC ist nicht nur eine Kaufentscheidung. Beide Maschinen sind für die ersten Wochen bis Monate Bastelprojekte, die Ihnen Spaß machen sollten. CNC-Technik ist faszinierend und Sie bekommen einen Einblick in Maschinenbau – egal, wie Sie sich entscheiden. Wenn die Maschine läuft, ist die Lernkurve noch nicht zu Ende – Sie müssen auch mit der CAD- und CAM-Software umgehen können. Wer die Zeit in den Selbstbau investiert, spart in beiden Fällen mindestens tausend Euro gegenüber einer Fertigmaschine. (pmk) c't

**Videos mit Hörproben**
ct.de/wujb

# Für Maker!

## Zubehör und Lesestoff

shop.heise.de/gadgets



### Waveshare Game HAT für Raspberry Pi

Ein Muss für jeden Retro Gamer! Verwandeln Sie Ihren Raspberry Pi in kürzester Zeit in eine Handheld-Konsole.
Mit Onboard-Speakern, 60 Frames/s, Auflösung von 480x320 und kompatibel mit allen gängigen Raspberrys.

shop.heise.de/game-hat **41,90 € ›**

BEST-SELLER



### ODROID-GO

Mit diesem Bausatz emulieren Sie nicht nur Spiele-Klassiker, sondern programmieren auch in der Arduino-Entwicklungsumgebung.

shop.heise.de/odroid **49,90 € ›**



### NVIDIA Jetson nano

Das Kraftpaket bietet mit 4 A57-Kernen und einem Grafikprozessor mit 128 Kernen ideale Voraussetzungen für die Programmierung neuronaler Netze, die ähnlich wie Gehirnzellen arbeiten.
**Inklusive Netzteil!**

shop.heise.de/jetson **134,90 € ›**



### Raspberry Pi-Kameras

Aufsteckbare Kameras, optimiert für verschiedene Raspberry Pi-Modelle mit 5 Megapixel und verschiedenen Aufsätzen wie z. B. Weitwinkel für scharfe Bilder und Videoaufnahmen.

shop.heise.de/raspi-kameras **ab 18,50 € ›**

NEUER PREIS!



### ArduiTouch-Set

Setzen Sie den ESP8266 oder ESP32 jetzt ganz einfach im Bereich der Hausautomation, Metering, Überwachung, Steuerung und anderen typischen IoT-Applikationen ein! Dieses Gehäuseset beinhaltet ein formschönes Wandgehäuse mit integrierter Leiterplatte.

~~69,90 €~~

shop.heise.de/arduitouch **36,90 € ›**

NEU




### „No Signal“ Smartphone-Hülle

Passend für Smartphones aller Größen bis 23cm Länge blockt diese zusammenrollbare Hülle alle Signale von GPS, WLAN, 3G, LTE, 5G und Bluetooth, sowie jegliche Handy-Strahlung.

shop.heise.de/no-signal-sleeve **29,90 € ›**

NEUER PREIS!




### Komplettset Argon ONE Case mit Raspberry Pi 4

Das Argon One Case ist eines der ergonomischsten und ästhetischsten Gehäuse aus Aluminiumlegierung für den Raspberry Pi. Es lässt den Pi nicht nur cool aussehen, sondern kühlt auch perfekt und ist leicht zu montieren. Praktisch: alle Kabel werden auf der Rückseite gebündelt ausgeführt – kein Kabelsalat!

~~117,60 €~~

shop.heise.de/argon-set **99,90 € ›**



### Stockschirm protec’ted

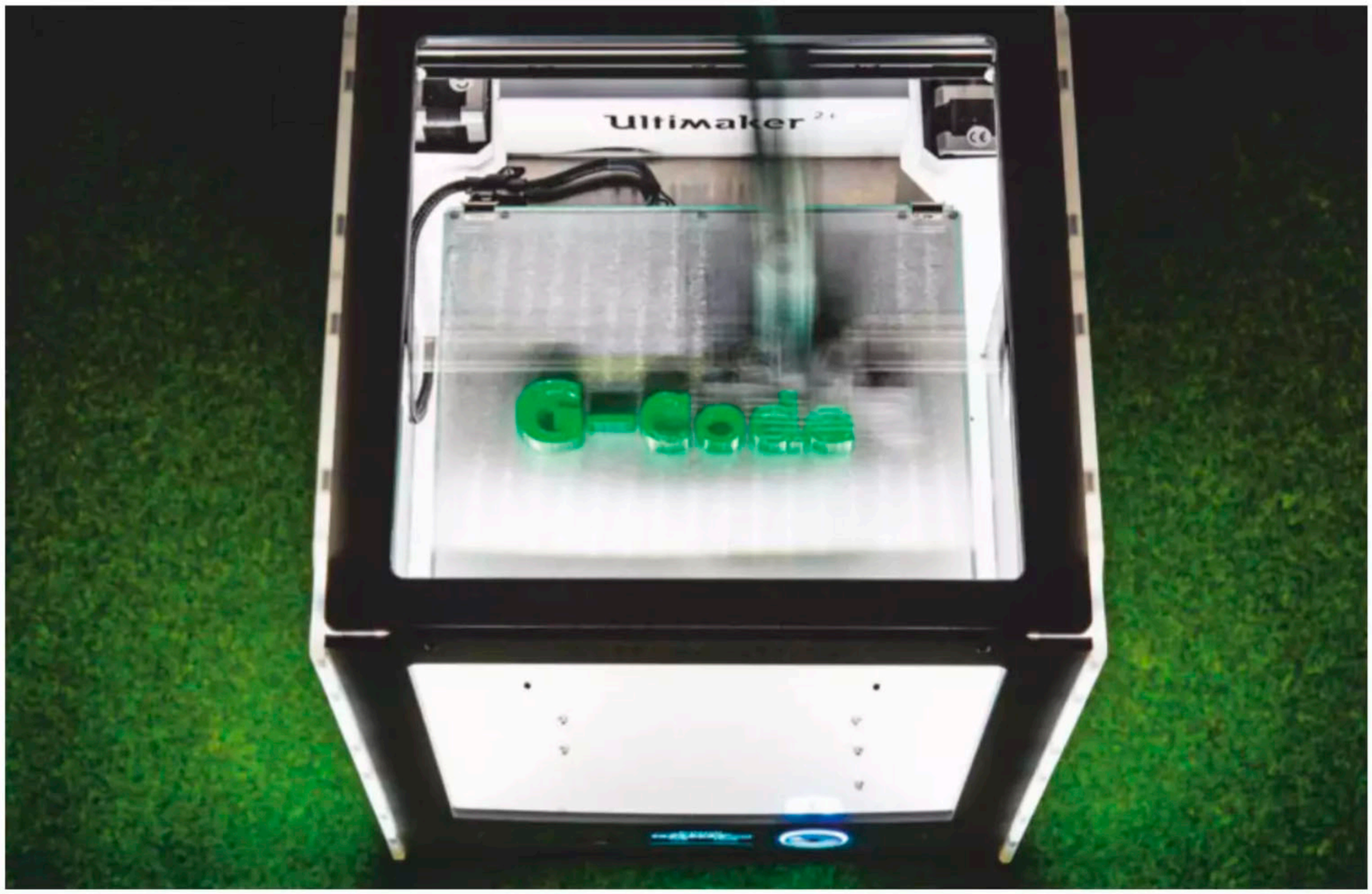
Innen ist Außen und umgekehrt.
Dieser etwas andere Regenschirm sorgt für interessierte Blicke auch bei grauem und nassem Wetter. Als Highlight kommt noch das stilvolle und dezente Design in Schwarz und Blau mit der mehr als passenden Aufschrift “Always protec’ted“ daher.

shop.heise.de/ct-schirm **22,90 € ›**

› Bestellen Sie ganz einfach online unter **shop.heise.de** oder per E-Mail: **service@shop.heise.de**

› Generell portofreie Lieferung für Heise Medien- oder Maker Media Zeitschriften-Abonnenten oder ab einem Einkaufswert von 20 €. Nur solange der Vorrat reicht. Preisänderungen vorbehalten.

heise shop
shop.heise.de ›




# G-Code programmieren und generieren

**G-Code, die Muttersprache von CNC-Maschinen wie Fräsen, Drehbänken und 3D-Druckern, ist eine der einfachsten Programmiersprachen überhaupt. Wir erklären die Basics, damit Sie .gcode-Dateien für schnellere Verarbeitung modifizieren und mit Python eigene erzeugen können.**

Von **Pina Merkert**

Der Maschinencode für CNC-Fräsen, Drehbänke und 3D-Drucker ist eine für Menschen lesbare Programmiersprache, deren Standardbefehle mit dem Buchstaben G anfangen: der G-Code. In der Frühzeit der CNC-Maschinen wurden die Programme von CNC-Programmierern direkt in dieser Sprache geschrieben, ähnlich wie in der Frühzeit der Computer viele Entwickler in Assembler programmiert haben. Inzwischen produzieren CAM-Prozessoren für Fräsen und Slicer für 3D-Drucker den meisten G-Code, vergleichbar mit Compilern, die Maschinencode für CPUs schreiben. Die G-Code-Generatoren erlauben aber nicht alles, was man mit G-Code programmieren kann. Für einige Spezialfälle lohnt

es sich, G-Code per Hand zu programmieren oder mit eigenen Skripten zu erzeugen. Schwer ist das nicht, da G-Code nur aus wenigen Befehlen besteht, die alle ähnlich funktionieren.

Mit dem Wissen aus diesem Artikel verstehen Sie sämtlichen G-Code, den Slicer wie Cura, Skeinforge oder Slic3r erstellen – und können ihn modifizieren, um beispielsweise in einem Druck verschiedene Düsentemperaturen für Ihren 3D-Drucker auszuprobieren und anschließend die beste auszuwählen. Für weitere Teststücke wie ein Kalibrierungskästchen (Einstellen der Schrittzahl für X-, Y- und Z-Motoren und Einstellen der Betthöhe) oder einen Geschwindigkeitsturm (Ermitteln der maximalen Druckgeschwindigkeit Ihres 3D-Druckers) haben wir ein paar Python-Skripte auf GitHub geladen, die Sie über ct.de/wrnh finden. Die Skripte nutzen eine kleine von uns erstellte Bibliothek namens `gcode-helpers.py`, die Sie nutzen können, um mit wenig Aufwand G-Code für eigene Testdrucke zu generieren. Das Erzeugen von G-Code mit einem Programm ist besonders für 3D-Drucker relevant, da der Code eines aussagekräftigen Teststücks schnell mal ein paar tausend Befehle enthält.

## G0 und G1

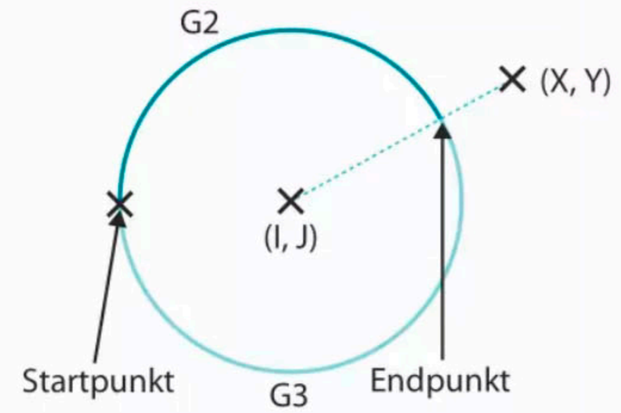
Im Prinzip funktioniert G-Code wie Turtle-Grafik. Sie haben als Kind kein LOGO programmiert? Die Idee ist, dass Sie einer Schildkröte (deswegen Turtle) Befehle geben, wo sie langlaufen soll. Überall wo sie war, hinterlässt sie einen Strich. G-Code ist weniger albern, da sich dort statt einer Schildkröte das Werkzeug, also der Fräser, Drehstahl oder das Hotend samt Düse bewegt. Die aktuelle Position speichert die Maschine in einem karthesischen Koordinatensystem, selbst wenn sie wie Delta-3D-Drucker keine rechtwinkligen Achsen verfährt. Gehen Sie einfach davon aus, mit X-, Y- und Z-Koordinaten in Millimetern zu tun zu haben.

Die Befehle `G0` und `G1` definieren einen neuen Punkt im Raum, zu dem sich das Werkzeug bewegt. `G0` ist für schnelle Maschinenbewegungen gedacht, bei denen eine Fräse kein Material abnimmt und ein 3D-Drucker nichts extrudiert. `G1` verwenden Sie für alle Bewegungen, bei denen die Maschine Material be- oder verarbeitet. Beide Befehle sorgen für eine lineare Bewegung von der aktuellen zur neuen Position.

Die Koordinaten des Zielpunkts schreibt man ohne Einheit mit Leerzeichen getrennt einfach hinter den Befehl. Da man Koordinaten, die sich nicht

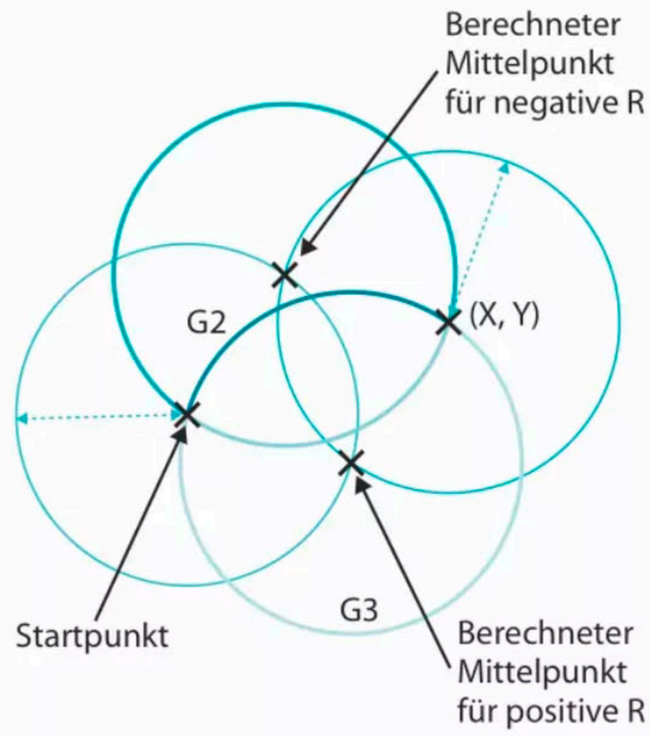
### Kreisbefehle mit Angabe des Mittelpunkts

Gibt man bei den Kreisbefehlen G2 und G3 mit den Koordinaten I und J einen Mittelpunkt an, fährt die Maschine eine Kreisbahn um diesen Mittelpunkt. Liegt der mit X und Y angegebene Zielpunkt nicht auf dem Kreis, stoppt die Maschine am nächstgelegenen Punkt auf dem Kreis. Der Radius ergibt sich aus dem Abstand zwischen Startpunkt und Mittelpunkt.




### Kreisbefehle mit Angabe des Radius

Definiert man bei G2 und G3 keinen Mittelpunkt, sondern gibt stattdessen mit R einen Radius an, konstruiert sich die Maschine selbst den Kreismittelpunkt. Dieser liegt auf dem Schnittpunkt von zwei Kreisen mit Radius R um Start- und Zielpunkt. Von den zwei Punkten, die infragekommen, wählt die Maschine bei positiven R denjenigen, um den sich die kürzere Kreisbahn ergibt, bei negativen R den mit der längeren Kreisbahn.



ändern, weglassen darf, muss man vor jeder Zahl mit den Buchstaben `X`, `Y` und `Z` die Raumachse angeben. 3D-Drucker nach Fused-Deposition-Modeling-Prinzip (FDM) verwenden eine vierte Dimension namens `E` für die Länge des Filamentfadens in Millimeter, die der Extruder durch die Düse drückt. Zusätzlich darf man hinter `F` angeben, wie schnell die Maschine zur neuen Position fährt. Ein Befehl, zur Position (10mm, 20mm, 30mm) mit einer Geschwindigkeit von 1200mm/min zu fahren und dabei 5mm zu extrudieren, sieht so aus:

```
G1 X10 Y20 Z30 E5 F1200
```

Wenn sich die Geschwindigkeit nicht ändert, kann man sie zu Beginn des Programms einmal festlegen und anschließend nur Positionen angeben:

```
G1 F2000
G0 X10 Y10 Z0.3 E0
G1 X20 E0.793
G1 Y20 E1.586
G1 X10 E2.379
G1 Y10 E3.172
```

Ein 3D-Drucker würde mit diesem Programm ein Quadrat mit 10mm Kantenlänge drucken und dabei 3,172mm Filament verbrauchen.

Die Koordinaten gibt man als Gleitkommazahlen mit Punkt statt Komma an. Wie viele Nachkommastellen Sinn ergeben, hängt von der Genauigkeit der Maschine ab. Ordentliche Fräsen weichen von der angegebenen Position weniger als 0,01mm ab, während 3D-Drucker in X- und Y-Richtung üblicherweise nur auf 0,05mm genau positionieren.

## G2 und G3

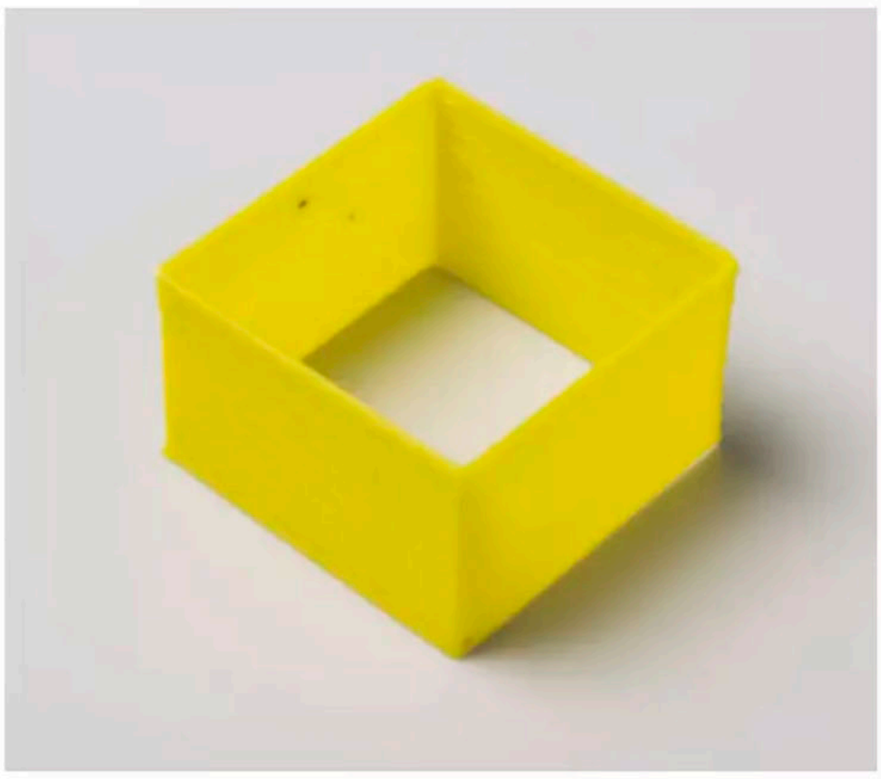
G-Code enthält auch Befehle für Kreisbahnen im (`G2`) und gegen (`G3`) den Uhrzeigersinn. Bei denen interpoliert die Maschine Zwischenpositionen nicht linear sondern zirkulär, sodass das Werkzeug einer Kreisbahn folgt, bis es dem mit `X` und `Y` angegebenen Endpunkt am nächsten ist. Über `I` und `J` gibt man dabei die X- und Y-Koordinate des Kreismittelpunkts an. Eine Z-Koordinate gibt es beim Kreisbefehl nicht, da er standardmäßig in der X-Y-Ebene erfolgt. Wer Kreise in der X-Z- oder Y-Z-Ebene fräsen will, muss die Ebene vorher mit `G18` oder `G19` umschalten. Zurück zur X-Y-Ebene schaltet man mit `G17`.

Moderne CNC-Fräsen verstehen `G2` und `G3` auch mit Angabe eines Radius hinter `R`. Bei dieser Variante darf man mit `I` und `J` keinen Mittelpunkt angeben, da die Maschine den anhand des Radius selbst berechnet. Dann fährt die Maschine auch immer den angegebenen Endpunkt an. Da es üblicherweise zwei verschiedene Kreisbahnen gibt, die am Zielpunkt enden würden, fährt die Maschine bei positiven Radien `R` immer die kürzere der beiden Varianten (weniger als 180°), bei negativen Radien die Längere (mehr als 180°).

G-Code für 3D-Drucker enthält aber üblicherweise keine komplizierten Kreisbefehle, da einige Firmwares diese nicht verarbeiten können. Stattdessen zerlegen die Slicer Rundungen in kleine lineare Bewegungen mit `G1`, was auf jeder Maschine funktioniert.

## Maschinen sind Automaten

G-Code spricht die Maschinen als Zustandsautomaten an. Mit dem Befehl `G91` schalten Sie von absoluten auf relative Koordinaten um, zurück zu absoluten Positionsangaben geht es mit `G90`. Derselbe Befehl `G1 X10 Y10` fährt die Maschine also an unterschiedliche Positionen, je nachdem ob vorher `G90` oder `G91` im Programm stand. Ähnlich verhält es sich mit `G20` und `G21`, die bewirken, dass alle nachfolgenden Koordinaten in Zoll oder in Millimetern interpretiert werden.



**Dieses Kästchen eignet sich gut als erstes Teststück für einen neu zusammengebauten 3D-Drucker. Mit seinen Maßen stellen Sie die Schritte pro Millimeter für X- und Y-Achse ein.**

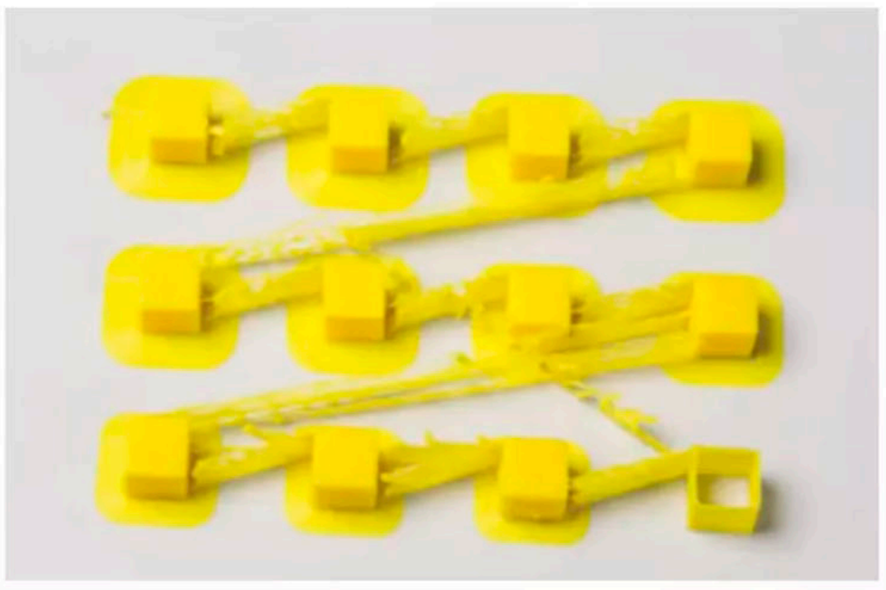
In der ersten Zeile eines G-Code-Programms hat man keine Ahnung, in welchem Zustand sich die Maschine gerade befindet. Daher sollten Programme immer mit einem Satz an Befehlen starten, die den Modus initialisieren, in dem der folgende Code arbeiten soll:

```
G21 ;metric values
G90 ;absolute positioning
G92 ;set current position as zero
```

Der Befehl `G92` definiert, an welchem Punkt sich die Maschine gerade befindet. Er löst keine Bewegung aus, sondern verschiebt das interne Koordinatensystem der Maschine. Wo der Nullpunkt liegt, hängt nämlich von der Maschine und manchmal sogar vom Werkstück ab. Gibt man keine Koordinaten an, verschiebt die Maschine ihren Nullpunkt auf die aktuelle Position. `G92 X10 Y10 Z10` sagt der Maschine hingegen, dass sie jede ihrer Achsen 10mm in negativer Richtung bewegen müsste, um den Nullpunkt zu erreichen.

## M-Codes

Neben den mit `G` beginnenden Befehlen, die auf allen Maschinen funktionieren sollten, gibt es maschinenabhängige Befehle, die mit `M` anfangen. Beispielsweise schaltet `M82` den Extruder eines 3D-Druckers auf absolute Koordinaten (ähnlich wie `G90` das für X, Y und Z tut) und `M107` den Lüfter zum Kühlen des Drucks aus. Auf einer Fräse ergeben diese Befehle keinen Sinn oder sind sogar mit einer ganz anderen Funktion belegt. Bei G-Code müssen Sie daher immer prüfen, ob der Code auch zu Ihrer Maschine passt. Es ist gut möglich, dass für einen RepRap erstellter G-Code auf einem Ultimaker nicht funktioniert und im schlimmsten Fall die Maschine beschädigt.



**An den kleinen Würfeln mit Kragen erkennt man, wie stark sich benachbarte Druckbahnen überlappen dürfen, bevor überschüssiger Kunststoff herausquillt. Der Würfel links oben nutzt gar keine Überlappung, während der rechts unten 100 Prozent des Zwischenraums ausfüllt. Die Fäden zwischen den Würfeln entstanden, weil wir vor Bewegungen zu wenig Filament eingezogen hatten.**

Glücklicherweise geben Slicer und Fräsprogramme üblicherweise in Kommentaren (alles, was hinter einem `;` oder in Klammern steht, ist ein Kommentar) an, für welche Maschine der Code erzeugt wurde und welchen G-Code-Dialekt er verwendet. In der Praxis beschränken sich die Unterschiede meist auf einen Block an Befehlen für die Initialisierung und einen, der ausgeführt wird, nachdem die Arbeit getan ist. Außerdem sollte das Werkstück oder Druckobjekt in den Bauraum passen, was Sie leicht sehen, wenn absolute Koordinaten zum Einsatz kommen (`G90`).

Beim Anpassen von G-Code für Ihren 3D-Drucker werden Sie häufig die Druck- und Betttemperatur ändern wollen. Die Extrudertemperatur setzen die Befehle `M109` und `M104`. Sie unterscheiden sich darin, dass der Drucker beim ersten wartet, bis die hinter `S` angegebene Temperatur wirklich erreicht wurde, während `M104` die Temperatur nur einstellt und der Drucker ohne zu warten den nächsten Befehl ausführt. Genauso verhält es sich mit den Befehlen für die Temperatur des Heizbetts, bei denen `M190` wartet und `M140` nicht.

Mit diesem Wissen können Sie die Aufheizzeit Ihres Druckers verkürzen, indem Sie das Aufheizen des Betts, des Extruders und das Nullen der Achsen parallelisieren. Wir heizen unseren Tevo Tarantula fürs Drucken von ABS beispielsweise mit folgender Sequenz auf:

```
M190 S100 ;set bed to 100°C and wait
M140 S105 ;set bed to 105°C
M109 S220 ;set nozzle to 220°C & wait
M104 S230 ;set nozzle to 230°C
G28 X0 Y0 ;move X/Y to min endstops
G28 Z0 ;move Z to min endstop
```

Während der etwa 10 Minuten, bis das Bett 100°C erreicht, bleibt das Hotend kalt, damit kein Kunststoff heraustropft. Sobald das Bett 100°C erreicht, heizt es weiter bis 105°C, während die Düse auf 220°C

aufheizt. Das Aufheizen des Hotends dauert lang genug, damit das Bett bis dahin etwa 104°C warm wird. Danach beginnt der Drucker schon mal die Achsen zu nullen, während die Düse die letzten 10°C bis zur eigentlichen Drucktemperatur aufheizt. Anschließend hat das Bett die eigentlich geplanten 105°C erreicht, während die Düse nicht weniger als 228°C heiß ist – warm genug, um den Druck zu starten. Cura setzt die Temperaturen mit `M140 S105` und `M109 S230` und braucht damit ungefähr eine Minute länger zum Aufheizen.

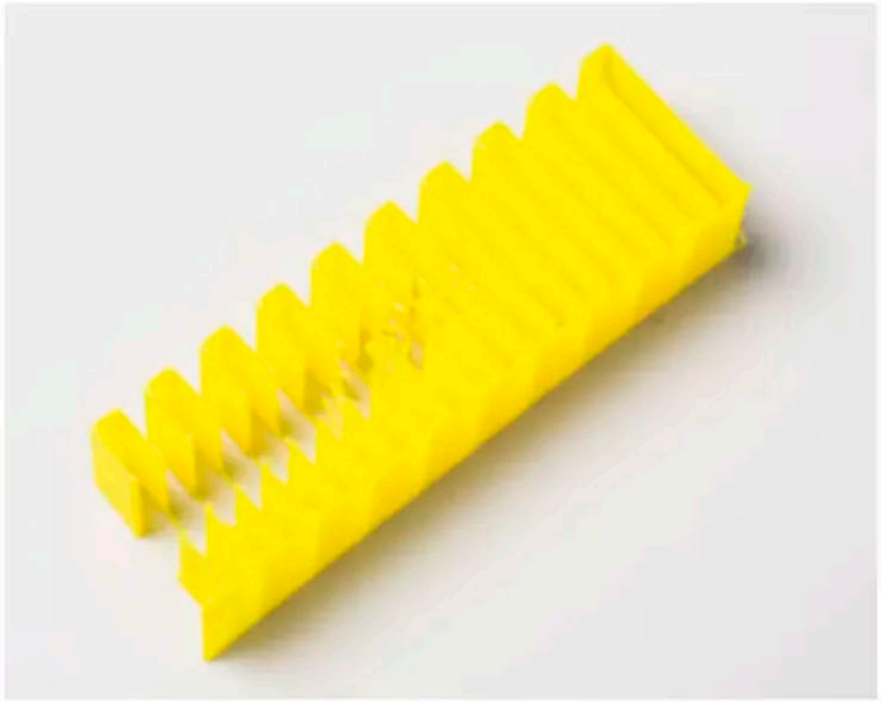
## G-Code mit Python

Schön, dass G-Code so leicht zu verstehen ist. Besonders Programme für 3D-Drucker geraten jedoch schnell sehr lang. Daher bietet es sich an, den Code mit einem Skript zu erzeugen, das in Sekunden Tausende von Codezeilen erzeugt. Ein solches Skript schreibt einfach Zeilen mit `G1`-Befehlen in eine Textdatei; die Frage ist nur, welche Koordinaten es jeweils für `X`, `Y`, `Z` und den Extruder `E` einsetzen soll. Für die Antwort muss man sich eine Detailvergrößerung vorstellen, was beim 3D-Drucken genau passiert.

Der Extruder drückt den Filamentfaden in die heiße Düse, wo der Kunststoff zähflüssig wird. Abhängig vom Druck, den der Extruder aufbaut, fließt mehr oder weniger Plastik durch die Düse und verteilt sich auf der Oberfläche darunter, beispielsweise der zuvor gedruckten Schicht. Um zu berechnen, wie viel Filament der Extruder durch die Düse schieben soll, muss man das Volumen des Plastikfadens kennen, der gedruckt wird. Wenn alles funktioniert, ist der so hoch wie die Schicht und drückt sich seitlich neben der Düsenöffnung leicht heraus. Das recht zähflüssige Plastik nimmt dabei zu den Seiten hin eine nahezu runde Form an. In der Mitte ergibt sich ein rechteckiger Bereich, der leicht breiter ist als die Düse (Breite `w`), also beispielsweise 0,42 Millimeter bei einer Düse mit 0,4 Millimeter großem Loch. Der Flächeninhalt dieser Form (ein Rechteck mit je einem Halbkreis links und rechts) beträgt.

$$a = w \cdot h + \pi \left(\frac{h}{2}\right)^2$$

Das Volumen ergibt sich, indem man diese Fläche mit der Länge der gedruckten Linie multipliziert.



**Bei diesem Teststück hat der Drucker hinten rechts gar kein Filament vor Bewegungen eingezogen und vorne links 9,5 mm³. Optimal ist wenig Einzug, aber auch keine Fäden – hier also 7 mm³.**

$$v = l \cdot a = l \cdot \left(w \cdot h + \pi \left(\frac{h}{2}\right)^2\right)$$

Damit die Düse genau diese Menge Kunststoff extrudiert, muss das Volumen des vom Extrudermotor nachgeschobenen Filaments exakt gleich groß sein.

$$v = e \cdot \pi \left(\frac{1,75mm}{2}\right)^2 \quad \Leftrightarrow$$

$$e = \frac{l \cdot \left(w \cdot h + \pi \left(\frac{h}{2}\right)^2\right)}{\pi \left(\frac{1,75mm}{2}\right)^2}$$

Damit Sie das nicht per Hand rechnen müssen, haben wir das alles in der Funktion `print_move()` unserer Python-Bibliothek gcodehelpers.py zusammengefasst (siehe ct.de/wrnh). Der Funktion geben Sie einfach einen Startpunkt, einen Endpunkt und die aktuelle Position des Extruders, und sie gibt die

neue Position des Extruders und den `G1`-Befehl mit allen Koordinaten zurück. Zusammen mit einer Schleife, die die Schichthöhe jeweils um 0,2 Millimeter erhöht, haben Sie schnell den G-Code für ein hohles rechteckiges Rähmchen generiert (Datei `generate_calibration_box.py` im Repository). Damit dieses Skript Code erzeugt, der problemlos auf Ihrem Drucker läuft, sollten Sie die Start- und Stoppsequenz in `gcodehelpers.py` noch so anpassen, dass sie zu Ihrem Drucker passt.

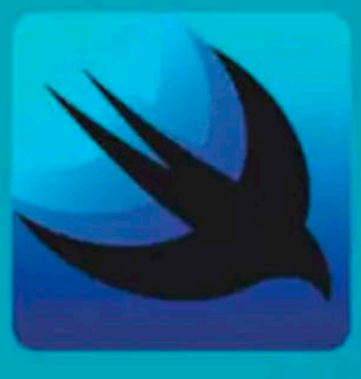
Unsere Bibliothek enthält noch einige weitere Funktionen, die es Ihnen einfach machen, Python-Skripte für eigene Teststücke zu entwickeln. Beispielsweise berechnet `dilate_erode()` mit der Bibliothek `shapely` die Koordinaten für Umrisslinien mit einem Abstand zu einem übergebenen Polygon. Die Funktion `print_brim()` nutzt das, um G-Code für einen Kragen um das Objekt zu erzeugen, der die Betthaftung erhöht. Ganz ähnlich erzeugt `print_wall()` Code für Wände mit mehreren Bahnen Dicke. Die Kombination aus `line_pattern()`, `infill()` und `print_layer()` füllt den Zwischenraum zwischen den Wänden auch aus. Im Skript `generate_line_overlap_cubes.py` haben wir das genutzt, um herauszufinden, wie stark sich nebeneinanderliegende Druckbahnen überlappen müssen, damit möglichst wenig Luft zwischen ihnen bleibt, aber auch kein Plastik überquillt. Mit ABS auf dem Tevo Tarantula konnten wir nicht mehr als 40 Prozent des Raums zwischen den Bahnen füllen – eine Erkenntnis, für die wir mit einem normalen Slicer tagelang einzelne Drucke hätten starten müssen.

Mit selbst programmierten Tests dieser Art finden Sie für die zahlreichen Parameter eines Slicers die bestmöglichen Werte heraus. Dafür verbrauchen Sie nur wenig Filament, können die Qualität Ihrer Drucke aber eventuell deutlich steigern. Das gilt besonders für billige Drucker, da sich die Hersteller hier meist nicht die Mühe machen, optimierte Presets mitzuliefern.

Schicken Sie uns ein Pull-Request via GitHub, wenn Sie `gcodehelpers.py` verbessern oder erweitern – beispielsweise um weitere Infill-Muster. Wir freuen uns auch über Ihre Ideen für Teststücke zum Kalibrieren – oder über Skripte, falls Sie gleich zur Tat schreiten. (pmk) c't

**Repository, Dokumentation**

ct.de/wrnh

SwiftUI für Fortgeschrittene

30.6. bis 1.7. 2021

Mac&i

Online Workshop



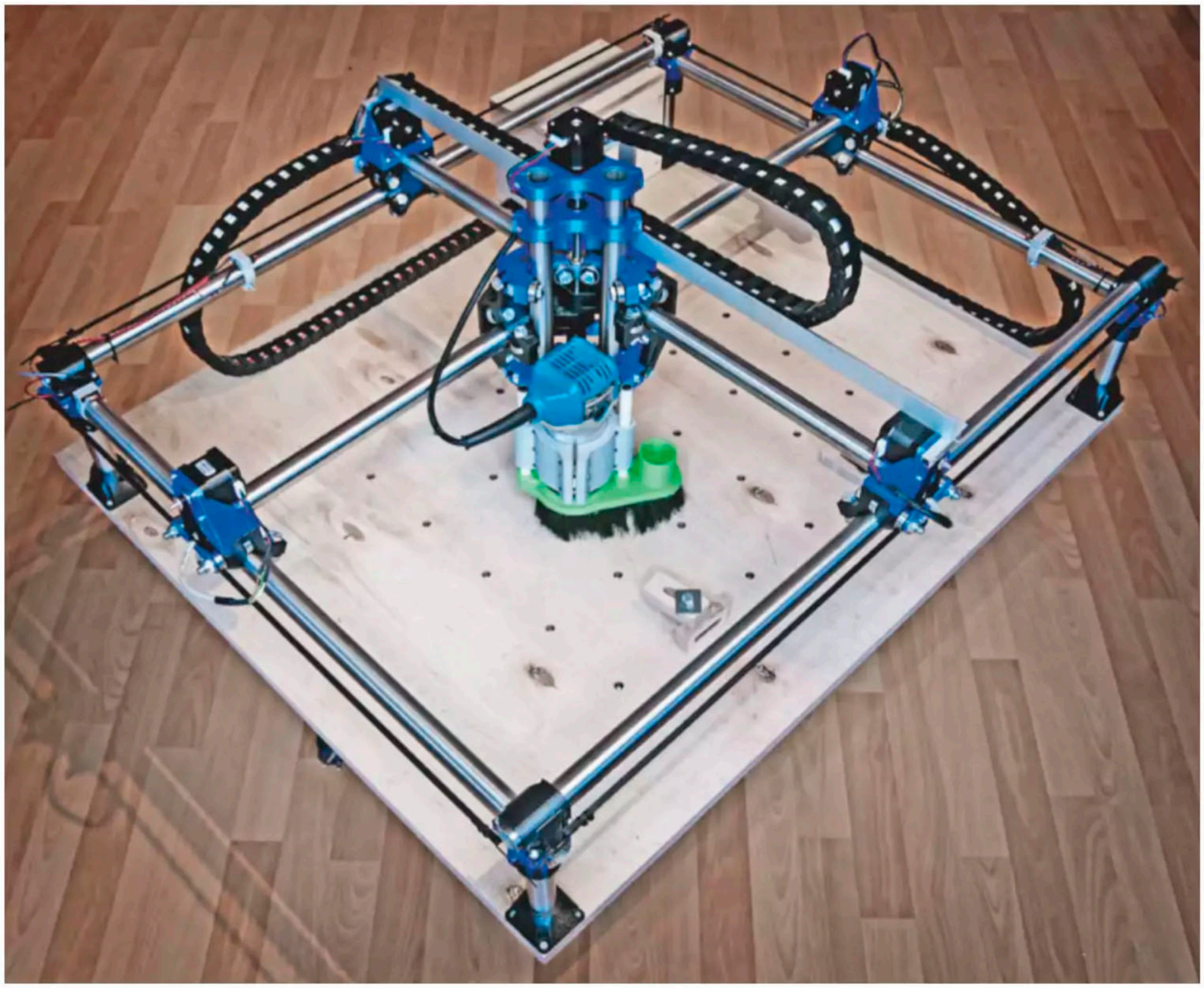
**Thomas Sillmann** ist leidenschaftlicher Autor, Apple Developer und Trainer.

In diesem zweitägigen Mac & i-Workshop dreht sich alles um **SwiftUI im Praxiseinsatz.**

Auf Basis einer iOS-App entwickeln die Teilnehmer:innen gemeinsam mit dem Trainer verschiedene Funktionen, um den Einsatz von SwiftUI in der Praxis zu erproben. Hinzu kommt das Aufstöbern und Beheben typischer Stolpersteine, wie sie immer wieder in SwiftUI-Projekten zu finden sind.

Zu guter Letzt geht es auch um die Entwicklung plattformübergreifender Apps und um Best Practices, welche die Arbeit mit SwiftUI erleichtern.

**JETZT ANMELDEN** heise-events.de/workshops/swiftui_fuer_fortgeschrittene



MPCNC in Aktion

# Eigene Portalfräse selbst ausdrucken

**Eine CNC-Fräse muss robust sein, sonst leidet die Präzision. Daher sind ihre Teile meist aus Aluminium oder Stahl. Die Mostly Printed CNC dagegen soll jeder zu Hause nachbauen können – mit Kunststoffteilen aus dem 3D-Drucker. Kann das fertige Gerät ernsthaft Holz oder gar Aluminium verarbeiten?**

Von **Jan Mahn**

CNC-Maschinen sind perfekte Werkzeuge, um Computerbegeisterung mit Bastelfreude zu verbinden. Füttert man eine CNC-Maschine mit G-Code, einer Sprache, die Bewegungspfade beschreibt (siehe Seite 96), befolgt sie die Anweisungen automatisch und bewegt ein Werkzeug zehntelmillimetergenau oder genauer. Die häufigsten Werkzeuge sind Fräs-Spindeln, Laser oder ein Messer für Folienschnitt. Aber auch ein 3D-Drucker ist eine CNC-Maschine, das Werkzeug ist dort das Hotend, aus dem das heiße Filament quillt. Weil das Drucken von Kunststoffteilen kaum Lärm und Dreck verursacht, ist 3D-Druck für viele der Einstieg ins Arbeiten mit CNC-Maschinen.

Da ist die Idee der „Mostly Printed CNC“ (MPCNC) durchaus naheliegend: eine CNC-Maschine, bei der alle Bauteile entweder Standardteile aus dem Anlagenbau sind oder auf einem 3D-Drucker selbst ausgedruckt werden. Am Ende soll man eine Fräs-Spindel einspannen können, um Platinen zu fräsen, Schilder zu gravieren, aber auch um Plattenwerkstoffe, Massivholz und sogar Aluminium zu verarbeiten. Die Vorschubgeschwindigkeiten und Zustelltiefen von großen Maschinen aus der Industrie sind nicht das Ziel – darauf kommt es in der heimischen Werkstatt und beim Prototypenbau aber oft auch nicht an. Außerhalb der Industrie dürfte es nur selten darum gehen, große Serien zu fertigen.

Etwa 500 Euro sollte man für die Fräse einplanen, so der Erfinder, der alle Pläne und eine ausführliche Anleitung unter dem Namen „V1 Engineering“ veröffentlicht hat (siehe ct.de/wk2b). Dieser Artikel kann und soll die ausführliche Anleitung des MPCNC-Entwicklers nicht ersetzen, ergänzt sie vielmehr um Lösungen zu den Problemen, denen ein Europäer gegenübersteht, zum Beispiel was die Beschaffung der Komponenten und den Zusammenbau angeht. Wir haben die Fräse nach der hervorragenden Anleitung gebaut und dabei hie und da etwas optimiert oder ergänzt. Nach der Fertigstellung haben wir die Fräse um nützliche Komponenten erweitert.

## Entscheidungen

Wenn man sich dazu entscheidet, ein Open-Hardware-Projekt in dieser Größe zu bauen, muss man unweigerlich Entscheidungen treffen. So ging es auch uns beim Bau der CNC-Maschine – dieser Artikel kann also nur den Bau *einer* möglichen MPCNC beschreiben. An vielen Stellen gibt es andere Pfade, die ebenfalls zum Erfolg führen können.

Wenn Sie sich für den Nachbau einer MPCNC entschieden haben, sollten Sie zügig Ihren 3D-Drucker vorheizen und mit dem Drucken der ersten Teile beginnen. Alle Druckdaten finden Sie bei Thingiverse (siehe ct.de/wk2b) in den drei Varianten F, C und J. Das ist die erste Entscheidung, die als Europäer aber leicht zu fällen ist: Die MPCNC besteht größtenteils aus Rohren und man kann die Maschine mit unterschiedlichen Rohrdurchmessern bauen. In Europa am leichtesten zu beschaffen sind Rohre in metrischen Maßen. Sie brauchen Edelstahlrundrohre mit einer Wandstärke von 2 Millimetern, leicht zu bekommen sind sie mit einem Außendurchmesser von 25 Millimetern. Die dazu passenden 3D-Druckvorlagen sind mit dem Buchstaben F gekennzeichnet. Achten Sie beim Drucken immer auf das F im Dateinamen. Schnell hat man sich bei Thingiverse verklickt und druckt falsche Teile, die nur auf den ersten Blick passen. Wenn Sie diesen Artikel in anderen Teilen der Welt lesen, können Sie auch zu Rohren mit dem Außendurchmesser 23,5 Millimeter (zugehöriger Buchstabe C) oder zu 1-Zoll-Rohren (25,4 Millimeter, Kennbuchstabe J) greifen.

Die Entscheidung für die Farbe des Filaments ist rein kosmetisch, wir haben die Teile in Schwarz und Blau gedruckt. Der Entwickler empfiehlt als Filament PETG, wir haben uns für PLA entschieden, von dem wir genug auf Lager hatten. Etwa 2 Kilogramm des Materials sollte man einplanen. Die gute Nachricht



**Möchte man die Fräse in einer staubarmen Umgebung betreiben, empfiehlt sich ein 3D-gedruckter Absaugschuh. Die Borsten stammen aus einem günstigen Haushaltsbesen.**

für 3D-Druck-Einsteiger: Alle Teile sind so konstruiert, dass sie ohne Stützstrukturen gedruckt werden können. Das funktionierte bei unseren Drucken problemlos, auch mit Warping hatten wir keine Probleme.

Bei der Füllung der Drucke wichen wir von der Empfehlung ab, nutzten das von Cura Ende 2018 eingeführte neue Infill-Muster Gyroid und reduzierten die empfohlenen Füllgrade um 10 bis 15 Prozent. Dieses Muster spart Material, Gewicht und vor allem Zeit. Den letzten Faktor sollte man nicht unterschätzen: Etwa 7 bis 10 Tage muss man mit einem einzelnen Drucker einplanen, wenn man es schafft, die Drucke so zu portionieren und an den eigenen Schlafrhythmus anzupassen, dass man immer nach etwa 12 Stunden das Druckbett freiräumen und direkt den nächsten Auftrag starten kann.

Während die Teile nacheinander vom Druckbett purzeln, kann man die weiteren benötigten Teile beschaffen. Zunächst die Edelstahlrohre: Wenn man Glück und einen Metallhändler hat, der sie auch präzise auf Maß zuschneidet, sollte man dort zuschlagen. Im Internet und auf Handelsplattformen gibt es ebenfalls Händler, die das Rohrmaterial auf Maß zuschneiden oder in festgelegten Stücken liefern können. Wer selbst zuschneiden will und nicht gerade eine Metallbandsäge mit Kühlung in der eigenen Metallwerkstatt hat, sollte sich einen Rohrabschneider anschaffen, der explizit auch für Edelstahl geeignet ist. Rohrabschneider für Kupferrohr gibt es in der Sanitärabteilung jedes Baumarkts, solche für Edelstahl aber nur im Fachhandel oder Internet (ab 20 Euro).

Die Maße der Rohre bestimmen Sie – und damit den Arbeitsbereich Ihrer Fräse. Die erste schwierigere Entscheidung. Theoretisch können Sie auch meterlange Rohre verwenden, die Stabilität und Präzision leidet aber massiv darunter. Wir haben uns für Außenmaße von 97 × 77 Zentimeter und eine Arbeitshöhe von 13 Zentimeter entschieden. Das ergibt einen effektiv nutzbaren Arbeitsbereich von 55 × 43 Zentimeter, die Werkstücke können aber problemlos über die Maße hinausragen. Sehr viel größer sollte man die Konstruktion nicht bauen, ein deutlich kleineres Modell, zum Beispiel für überschaubare Gravierarbeiten, ist kein Problem. Wenn Sie planen, per CNC große Möbelwände oder andere extrem große Teile zu fräsen, sollten Sie sich stattdessen die LowRider-CNC desselben Entwicklers ansehen (ct.de/wk2b).

In der Dokumentation des Entwicklers finden Sie einen Kalkulator, der Ihnen alle benötigten Rohrlängen generiert, nachdem Sie den geplanten Arbeitsbereich eingegeben haben. Außerdem erfahren Sie dort, wie groß Ihre Grundplatte mindestens sein muss. Eine solche besorgen Sie sich am einfachsten im Holzzuschnitt des Baumarkts, zum Beispiel 20 Millimeter Multiplex oder (deutlich teurer, aber wasserfest) Siebdruckplatte. Wenn Sie an beiden Seiten mindestens drei bis fünf Zentimeter mehr Platz einplanen, können Sie dort später eine kleine Kiste mit der Elektronik unter- und Energieketten zur Kabelführung anbringen.

## Hart-Ware

Neben den Rohren brauchen Sie diverse weitere Kleinteile. Das gesamte Material gibt es auch in Sets direkt im Shop des Entwicklers V1 Engineering (siehe ct.de/wk2b). Wenn Sie dort kaufen, unterstützen Sie zwar auch die Weiterentwicklung des Projekts, für europäische Kunden ist der Aufwand aber zu groß: Stress mit der Einfuhr und Versandkosten sprechen gegen einen Kauf in den USA. Dankenswerterweise ist die ganze Fräse so konstruiert, dass man die Teile problemlos auch außerhalb dieses Shops bekommt. Weltweite Verfügbarkeit aller Komponenten war eine der Grundideen der MPCNC. Wir konnten alle Teile bei deutschen Händlern beschaffen, meist wurden wir bei den großen Portalen fündig (siehe ct.de/wk2b).

Schrauben in M2,5, M3, M5 und M8 brauchen Sie in verschiedenen Längen (optional finden Sie in der Dokumentation auch entsprechende zöllige Schrauben). Viele bekommt man im Baumarkt, am günstigsten in einem, der Schrauben nach Kilopreis berechnet. Finden Sie eine seltene Länge nicht, kann man zur Not auch eine Schraube kürzen – Mutter aufdrehen, absägen und die Mutter zurückdrehen. So ist die Chance hoch, dass das Gewinde das Absägen überlebt. Die Muttern kaufen Sie größtenteils in der selbstsichernden Variante, einige nichtsichernde Muttern müssen Sie ebenfalls zur Hand haben.

Ein weiterer großer Posten sind Kugellager vom Typ 608RS. Die erforderlichen 45 Stück kosten etwa 25 Euro. Ein paar in Reserve können aber nicht schaden. Die weiteren Komponenten sind Standardware, die jeder kennt, der schon mal an einem 3D-Drucker geschraubt hat: Nema-17-Motoren bewegen das Portal, GT2-Zahnriemen verlaufen auf der X- und Y-Achse. Eine T8-Spindel mit zwei passenden Muttern und einer Aufnahme für den Motor bildet die Z-Achse.

## Elektronik-Entscheidungen

Wenn diese Teile mit der Post unterwegs sind und der 3D-Drucker druckt, müssen Sie wieder Entschei-

dungen treffen, diesmal ist die Elektronik dran: Die Nema-17-Motoren können Sie auf zig Arten ansteuern. Der Entwickler schlägt unter anderem das RAMBo-Board als Steuerung vor. Wir haben uns stattdessen dafür entschieden, einen Arduino UNO mit dem „CNC Shield“ (unter diesem Suchbegriff leicht zu finden) und Treibern vom Typ A4988 als Grundlage zu nutzen. Das Shield könnte maximal vier Achsen ansteuern, für die MPCNC brauchen Sie aber nur drei Treiber. X- und Y-Achse haben zwar je zwei Motoren, diese werden aber jeweils von nur einem Treiber angesteuert. Mit einem anderen Board und weiteren Treibern könnte man noch mehr Kraft auf die Achsen bringen – diese Kraft ist aber beim Fräsen meist nicht der limitierende Faktor. Das Arduino-Shield wird Ihnen für um die 12 Euro zahlreich im Netz angeboten, manchmal sind die Treiber-Platinen schon in einem Set enthalten.

Zur Ansteuerung der Maschine vom PC haben wir die Software Estlcam ausgewählt. Das Programm stammt von einem deutschen Entwickler, die Hilfetexte in der Oberfläche sind also alle von einem Muttersprachler verfasst. Die Software erzeugt nicht nur G-Code, sondern kann den Arduino auch mit seiner Firmware programmieren. Zum Einsatz kommt eine Firmware, die dem Funktionsumfang der populären Open-Source-Firmware GRBL entspricht. Für Einsteiger ins Zerspanen besonders hilfreich ist, dass die im Programm integrierten Hilfetexte auch ganz konkrete Tipps zum Fräsen enthalten. Testen kann man Estlcam mit allen Funktionen kostenlos. Solange man die Vollversion für 49 Euro nicht gekauft hat, wird vor dem Starten des Fräsvorgangs eine Pause eingelegt, die bis zu 60 Sekunden dauert. Estlcam funktioniert nur unter Windows, am besten besorgt man sich ein altes Notebook als Werkstatt-Gerät.

Für die Verkabelung von der Steuerung zu den Motoren haben wir geschirmtes Kabel aus dem Anlagenbau vom Typ LiYCY 4 × 0,5 bestellt. Außerdem einen Satz Dupont-Stecker und ein Sortiment Schrumpfschläuche, um ansehnliche und robuste Kabel zu bauen.

## Fräsen-Zusammenbau

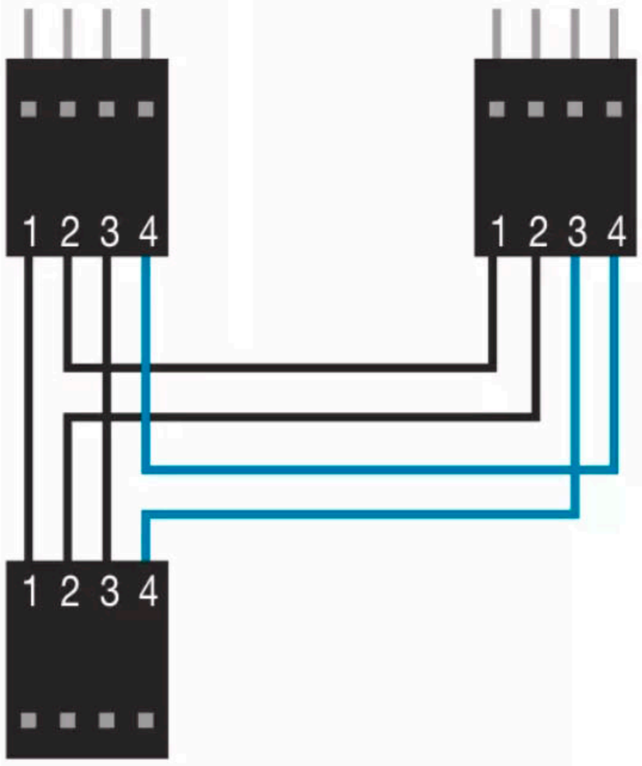
Der gesamte Zusammenbau der Kunststoffteile und der Rohre ist äußerst gut und Schritt für Schritt dokumentiert. Werkzeuge brauchen Sie kaum: Schraubenschlüssel, Schraubendreher, Inbus-Schlüssel und am besten eine Ratsche reichen aus. Für die Z-Achse müssen Sie für insgesamt vier Löcher einen Akkuschrauber zur Hand nehmen. Bei diesem Bauschritt ist eine Standbohrmaschine wirklich von Vorteil.

Wichtig beim Zusammenbau ist, dass das Gestell exakt rechtwinklig ist. Messen Sie also bei allen Rohrabschnitten genau, stellen Sie die Rohre am besten nebeneinander und bringen Sie sie auf gleiche Höhe. Damit Sie ein Rechteck und kein nicht rechtwinkliges Parallelogramm bauen, messen Sie unbedingt die Diagonalen sorgfältig mit einem Maßband. Sind sie gleich lang, ist das Ergebnis rechtwinklig. Am besten schraubt man zwei nebeneinander liegende Füße auf der Platte fest und richtet dann die anderen beiden aus, bis das Rechteck perfekt ist – nur so kann die Fräse später auch perfekte Rechtecke fräsen. Zum Betrieb müssen alle Füße unbedingt fest mit der Grundplatte verbunden sein.

Zu den zahlreichen Kugellagern gibt es wenig zu sagen. Der Zusammenbau der Schlitten ist reine Fleißarbeit. Schraube rein, Kugellager einfädeln und verschrauben. Im letzten Schritt kommen die Zahnriemen ins Spiel. Übertreiben Sie es hier beim Spannen nicht – die Maschine soll leicht über die Rohre gleiten und sich nicht quälen. Beim Zusammenbau

### Y-Kabel für die X- und Y-Achse

Damit zwei Motoren von einem Treiber angesteuert werden, brauchen Sie für beide Achsen je ein Y-Kabel. Wenn man sich die Steckverbindungen sparen will, kann man die oberen Dupont-Stecker auch ersetzen und die Motorkabel direkt anlöten.



der Z-Achse wichen wir von der offiziellen Anleitung ab: Der Entwickler hat sich 3D-gedruckte Halter ausgedacht, in die man Muttern einklebt und in die vorgebohrten Rohre einführt. Nachdem die Muttern bei uns nicht so recht im Kunststoff halten wollten, ließen wir die gedruckten Halter weg und hielten die Muttern beim Zusammenschrauben mit einer ausreichend langen Arterienklemme aus dem Chirurgiebedarf im Rohr fest. Weitere Klippen gab es beim Zusammenbau nicht zu umschiffen. Vorsicht sollte man lediglich beim Einsatz einer Ratsche walten lassen. Mit dem langen Hebel kann man die Schrauben schnell so fest anziehen, dass der gedruckte Kunststoff bricht. Nach etwa einem Arbeitstag ist das Gerät mechanisch fertig zusammengebaut.

## Elektronik-Zusammenbau

Die Arbeiten an der Elektronik kann man problemlos schon beginnen, wenn der 3D-Drucker noch druckt oder man auf Lieferungen warten muss. Beginnen Sie mit dem Zusammenbau des CNC-Shields. Für die drei Achsen müssen Sie die Schrittweite per Jumperstecker einstellen. Leider lagen unserem Shield keine Jumper (Rastermaß 2,54 mm) bei und wir mussten sie separat organisieren (alte Mainboards oder IDE-Festplatten sind gute Teilespender). Oben rechts sehen Sie, wo die insgesamt fünf Jumper gesetzt und wie die drei Treiber-Platinen gesteckt werden müssen.

Für die X- und Y-Achse müssen Sie sich ein Y-förmiges Kabel anfertigen, um mit einem Treiber zwei Motoren anzusteuern. In der Skizze „Y-Kabel für die X- und Y-Achse“ sehen Sie, wie die Kabel beschaltet werden müssen. Bei unseren Motoren lagen kurze Kabel bei, die wir abschnitten, mit der LiYCY-Leitung verlöteten und die Verbindungen mit Schrumpfschlauch sicherten. Zum Installieren auf dem Arduino-Shield muss man am Ende 4-polige Dupont-Stecker aufcrimpen. Die Stecker müssen richtig beschaltet sein, wenn man sie verpolt einsteckt, richtet man aber keinen Schaden an – die Motoren laufen dann nur verkehrt herum. Das Kabelmanagement muss in der ersten Ausbaustufe noch nicht perfekt sein. Verlegen Sie die Verbindungen zwischen den X- und Y-Motoren durch die Rohre. So kann wenig hängenbleiben.

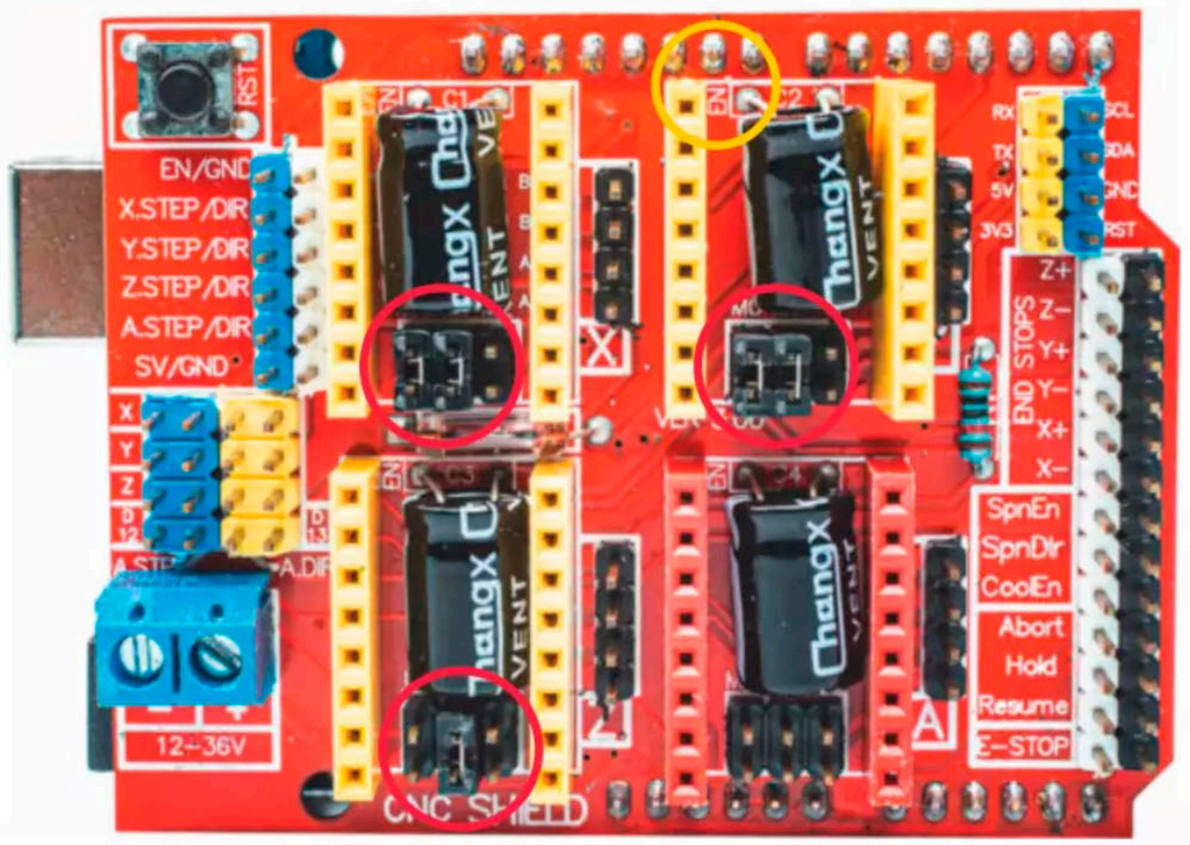
Der Arduino bekommt seine Spannung über das USB-Kabel. Das aufgesteckte CNC-Shield braucht eine separate Spannungsversorgung, ist aber nicht sonderlich wählerisch. Zwischen 12 und 36 Volt soll es laut Datenblatt verkraften. Empfohlen ist ein recht leistungsstarkes Netzteil, das bei 12 Volt mindestens 3,5 Ampere liefert. Empfehlenswert ist zum Beispiel ein ausgedientes Notebook-Netzteil mit 12 oder 19 Volt. Wir entschieden uns für ein altes ATX-PC-Netzteil und zogen die benötigten Drähte schlicht aus dem Mainboard-Stecker. Eingeschaltet wird ein solches Netzteil, indem man den grünen Draht mit einem schwarzen Masse-Draht verbindet (optional baut man sich hier einen Schalter ein). Die 12 Volt bekommt man, indem man einen gelben Draht für den Plus-Pol und einen schwarzen für Minus benutzt. Angeschlossen werden sie an den Schraubklemmen unten links auf dem Shield.

Anschließend müssen Sie die Ausgangsspannung der Treiber an Ihre Spannungsquelle anpassen. Dafür gibt es auf den Treibern ein Poti, das man mit einem sehr kleinen Kreuzschraubendreher bedient. Am einfachsten ist es, einen Pol eines Voltmeters am Minus-Pol der Spannungszufuhr zu fixieren. Den anderen Pol des Messgeräts klemmen Sie mit einer Krokodilklemme an einen nicht isolierten Schraubendreher, den Sie auf das Poti setzen. Drehen Sie dann so lange vorsichtig am Poti, bis die Spannung ganz knapp unter 0,7 Volt liegt.

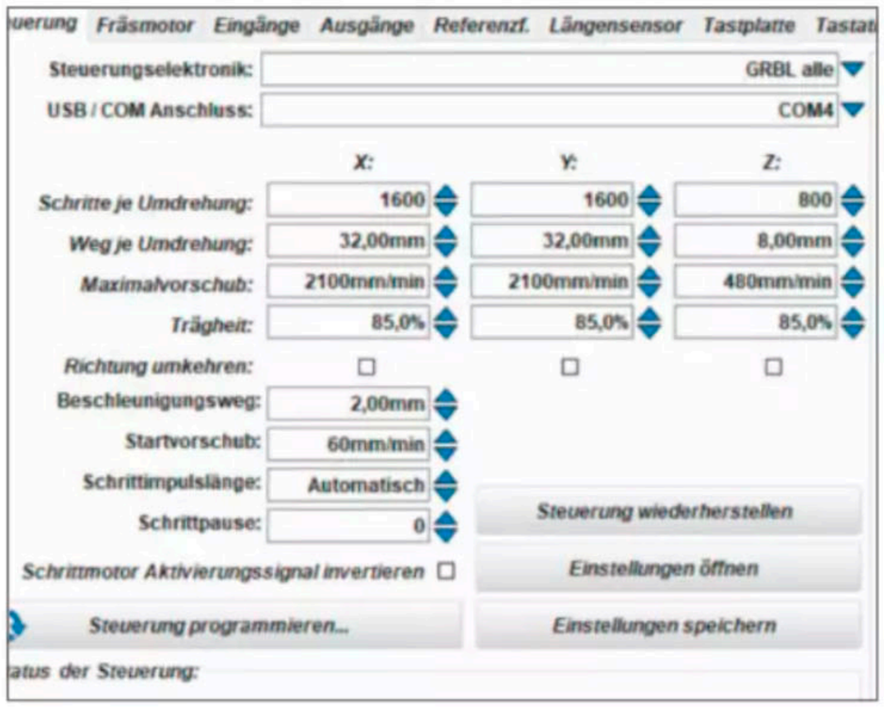
Wenn Sie die Kabel für X- und Y-Achse verlötet, mit Dupont-Steckern versehen und mit den Motoren

### Bestückung des Arduino-CNC-Boards

Für X- und Y-Achse müssen zwei Jumper gesetzt werden, für die Z-Achse einer auf den mittleren Pins (rot markiert). Auf dem Board sowie auf den Treiberplatinen ist der Pin „EN“ gekennzeichnet (gelb markiert). Schauen Sie beim Einbau der Treiber unbedingt genau hin.



**Estlcam überträgt eine GBRL-kompatible Firmware auf den Arduino. Damit die Maße stimmen, muss man die Schrittlängen vorher angeben.**




verbunden haben, können Sie mit dem Flashen der Firmware beginnen. Schließen Sie den Arduino per USB-Kabel am Rechner an. Auf diesem Rechner müssen Sie die Arduino-IDE installieren. Diese finden Sie zusammen mit Estlcam über ct.de/wk2b. Öffnen Sie die IDE und wählen Sie oben im Menü unter „Werkzeuge“ zunächst ein Board vom Typ „Arduino Uno“ und als Port den COM-Port aus, den der Rechner ihm zugewiesen hat. Zum Test können Sie ein leeres Programm mit dem Upload-Button in der oberen Leiste auf den Arduino kopieren. Hat das geklappt, starten Sie Estlcam.

Beginnen Sie dort mit der Einrichtung im Menüpunkt „Einstellungen/Grundeinstellungen“. Wählen Sie im Dialog die Option „GRBL“. Öffnen Sie dann „Einstellungen/CNC Programme“ und wählen Sie wieder „GRBL“. Weiter geht es mit dem Dialog „Einstellungen/CNC Steuerung“. Wählen Sie dort im ersten Feld „GRBL alle“ und darunter den richtigen COM-Port. Im Screenshot oben sehen Sie die weiteren Werte, die zu Ihrer MPCNC passen. Die „Schritte pro Umdrehung“ entsprechen dem Wert, den Sie mit den Jumpern eingestellt haben. Wenn Sie die Daten abgetippt haben, starten Sie den Flashvorgang mit dem Schalter „Steuerung programmieren“. Hat alles geklappt, sollte sich das Steuerungsfenster öffnen. Links finden Sie Pfeiltasten, mit denen Sie Ihre CNC-Maschine fahren lassen können.

Sollte eine Achse in die falsche Richtung fahren, schalten Sie unbedingt zuerst das 12-Volt-Netzteil ab, drehen Sie dann den Dupont-Stecker und aktivieren Sie die Spannung wieder (der Arduino darf am USB-Port angeschlossen bleiben). Reagieren alle Achsen korrekt auf die Steuerbefehle, ist Ihre MPCNC einsatzbereit.

## Auf großer Fahrt

Bei der ersten Fahrt sollten Sie noch keine Fräs-Spindel einsetzen – die Gefahr, bei falscher Verkabelung Schaden anzurichten, ist nämlich recht hoch. Am besten basteln Sie sich aus einem alten Kugelschreiber einen gefederten Stift: Zerlegen Sie den Kugelschreiber, bauen Sie die kleine Feder hinter statt vor der Mine ein und setzen Sie ihn so wieder zusammen. So kann der Stift kleine Schwankungen in der Höhe ausgleichen. Den Stift montieren Sie mit Kabelbindern provisorisch am Werkzeughalter der MPCNC.

## Zeit zur Zerspanung

Die Wahl des Fräsmotors ist die letzte große Entscheidung. Die US-Amerikaner im MPCNC-Forum schwören meist auf Fräsen von DeWalt. Eine weitere Option sind wassergekühlte Motoren ohne Markenname direkt aus China. Zusammen mit einem wenig vertrauenswürdig aussehenden Netzteil gibt es die ab 80 Euro, möglicherweise zuzüglich Einfuhrumsatzsteuer.



**Bei der Probefahrt sollten Sie nicht gleich die Fräse anwerfen. Ein umgebauter Kugelschreiber mit gefederter Spitze reicht als Werkzeug für erste Experimente.**

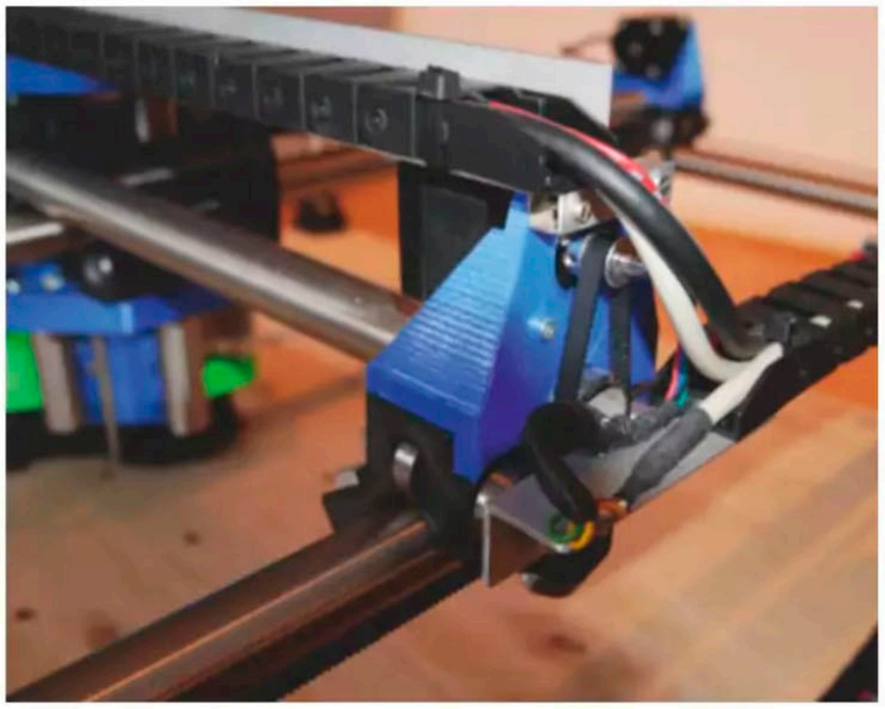
Wir entschieden uns stattdessen für die Oberfräse RT0700 von Makita, die man ohne viel Zubehör für 120 Euro bekommt. Die ist modular aufgebaut: Mit dem aufgesteckten Oberfräsenaufsatz kann man sie mobil als handgeführte Maschine nutzen. Ohne Aufsatz passt das zylindrische Metallgehäuse gut in eine CNC-Fräse. Passende Werkzeughalter zum Ausdrucken gibt es zahlreich bei Thingiverse. Wir entschieden uns für eine Konstruktion mit einer höhenverstellbaren Absaugvorrichtung (siehe ct.de/wk2b). Im gedruckten Absaugschuh ist rundherum eine Nut eingelassen, in die man die Borsten eines billigen Besens oder Handfegers büschelweise mit Sekundenkleber einklebt. Zusammen mit einem leistungsstarken Sauger fliegen dann kaum Späne umher.

Für gute Ergebnisse brauchen Sie auch gute Fräser, am besten aus Hartmetall. Die beste Auswahl gibt es bei Online-Fachhändlern für CNC-Frästechnik. Für den Einstieg empfehlen wir einen 6-Millimeter-Einschneider für Holz und einen weiteren für Alu. Für Gravierarbeiten sollten Sie sich nach einem Gravierstichel mit 1/8-Zoll-, 3- oder 4-Millimeter-Schaft umsehen. Bei der Makita-Fräse liegt je eine Aufnahme für 6 und 8 Millimeter bei. Wenn Sie kleinere Schaftdurchmesser einsetzen wollen, brauchen Sie eine Reduzierhülse zum Beispiel von 3 oder 4 auf 6 Millimeter.

## Optimierungen

In den ersten Wochen sollten Sie noch keine großen Fräsprojekte einplanen und sich genug Zeit für Optimierungen der Fräse nehmen. Uns störte die mit Kabelbindern improvisierte Kabelführung so sehr, dass wir direkt nach dem Zusammenbau 30 Millimeter breite Energieketten für alle drei Achsen bestellten. An den Schlitten der X- und Y-Achse befestigten wir sie mit Alu-Winkeln, die man gut an den bestehenden M8-Schrauben anbringen kann. Für die Versorgung der Z-Achse installierten wir das gleiche Alu-Profil quer über der Y-Achse. Darauf liegt die dritte Energiekette, die am Z-Achsen-Schlitten befestigt ist.

Um Auffahrunfälle am Ende der Achsen zu vermeiden, sollte man schnell in Endschalter für X- und Y-Achse investieren. Insgesamt braucht man vier Mikroschalter (auf Handelsplattformen zu finden über die Suchbegriffe „Mikroschalter Endstop“). An X- und Y-Achse befestigt man sie mit 3D-gedruckten Haltern. Eine Druckvorlage finden dafür Sie über ct.de/wk2b. Zur Verkabelung nutzten wir ungeschirmte Litze und konnten keine Fehlauslösungen beobachten. Die Kabel kann man elegant in den Rohren verschwinden lassen, an den Enden Dupont-Stecker aufcrimpen und auf dem CNC-Board installieren (auf den Steckleisten rechts). Die Endschalter sollte man so positionieren, dass die Fräse nie in einen Bereich fahren kann, in dem sie zum Beispiel mit einem Kabel hängenbleiben kann.



**Gehört nicht zur Standardausstattung: Die Energieketten mit 30 Millimeter Breite haben wir mit Alu-Winkeln installiert.**

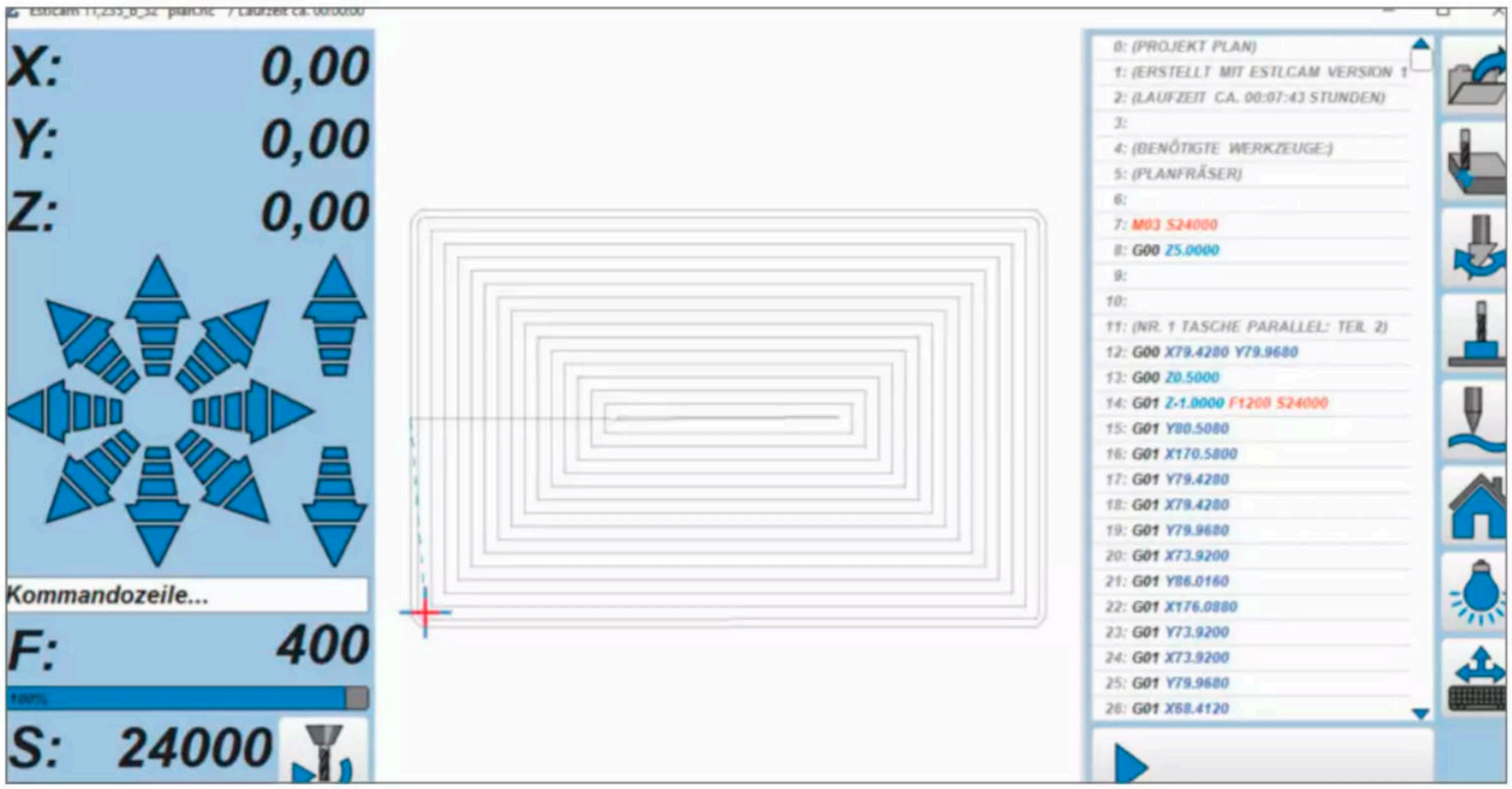
Um Werkstücke zu fixieren, entschieden wir uns dafür, ein Lochraster in die Grundplatte zu bohren, in die wir anschließend M8-Einschlagmuttern von unten einschlugen. Statt zum Akkuschrauber zu greifen, überließen wir die Bohrarbeit der CNC-Fräse. Dafür legten wir in Estlcam ein 10-Zentimeter-Raster über die Arbeitsfläche und platzierten darauf Bohrlöcher mit 10 Millimeter Durchmesser, die ein 6-Millimeter-Fräser ausfräste.

## Erste Projekte

Das Fräsportal besteht zu großen Teilen aus Plastik und die Bestandteile haben inklusive Fräsmotor etwa 600 Euro gekostet. Da man für CNC-Portalfräsen problemlos auch ein Vermögen ausgeben kann, ist durchaus Skepsis angebracht, ob man mit der Konstruktion ernsthaft arbeiten kann. Man kann.

Das erste Projekt entstand aus Holz: Im Online-CAD-Programm Onshape entwickelten wir sogenannte Spannpratzen (die Konstruktion finden Sie über ct.de/wk2b), die die Fräse anstandslos mit einem 6-Millimeter-Fräser aus Multiplex ausschnitt.

**Den Umgang mit Estlcam lernt man schnell. Hat man die Fräsbahnen angelegt, kann man den Fräsvorgang starten.**




In Verbindung mit M8-Schrauben und den Einschlagmuttern eine flexible und robuste Lösung zum Spannen von Werkstücken. Bei Holzprojekten macht die Fräse eine gute Figur und selbst bei harten Hölzern wie Buche kann man sich auf ansehnliche Vorschubgeschwindigkeiten und Zustelltiefen herantasten. Jede MPCNC ist anders und Werte sind nicht auf andere Maschinen übertragbar.

Wenn Sie mit Holz und den Grundlagen von Estlcam vertraut sind, können Sie sich an den Werkstoff Aluminium heranwagen. Auch hier hilft nur Herantasten an geeignete Werte und unser Artikel auf Seite 114. Ein hervorragendes Übungsprojekt für Alu ist eine sogenannte Tastplatte, die nebenbei auch eine sehr sinnvolle Funktion erfüllt. Die Tastplatte aus Alu wird mit einem Draht mit dem Board verbunden. Sie besteht aus einem rechten Winkel, den man auf die Ecke des Werkstücks legt. Estlcam bringt eine sehr mächtige Funktion mit: Mit einer Tastplatte kann es automatisch den Nullpunkt des Werkstücks und dessen Höhe und auch die Verdrehung ertasten – das Positionieren wird zum Kinderspiel.

Ein weiterer sehr dankbarer Werkstoff für Übungszwecke ist Schaumstoff. Wenn Sie nur verschiedene Frästechniken ausprobieren wollen (zum Beispiel das Wirbelfräsen oder eine Gewindefräsung), beschaffen Sie sich Styrodur als Übungswerkstoff. Um den kompletten Ablauf vom Zeichnen eines Designs in einem Vektorgrafik- oder CAD-Programm Ihrer Wahl bis zum Fräsen in Estlcam zu üben, sind Werkzeugeinlagen aus Schaumstoff ein schönes Übungsprojekt. Zweifarbigen Schaumstoff für solche Einlagen gibt es in verschiedenen Stärken. Wenn man zum Beispiel seine Schraubenschlüsselschublade aufräumen möchte, legt man diese auf eine weiße Fläche, fotografiert sie parallel von oben und zeichnet die Konturen mit einem Grafikprogramm nach. In Estlcam werden aus der Zeichnung sogenannte Taschen in passender Tiefe. Für Schaumstoff gibt es spezielle Fräser, ein Holzfräser funktioniert aber auch.

**Vorlagen, Dokumentationen und Videos**

ct.de/wk2b

## Kein Spielzeug

Die Arbeit mit der MPCNC macht Spaß und die Maschine ist ihr Geld absolut wert. Nur weil sie aus Kunststoff ist, ist die MPCNC kein Spielzeug: Die Kunststoffteile sind nicht der limitierende Faktor und das Portal ist stabil genug, um Holz und, etwas langsamer, auch Alu zu verarbeiten. Wie bei 3D-Drucker-Eigenbauten lebt das Projekt von den eigenen Optimierungen. Je nach Anforderung lohnt es, im Thingiverse-Universum unter dem Suchbegriff „MPCNC“ nach Erweiterungen zu suchen. Wer bisher seine Projekte mit dem 3D-Drucker umgesetzt hat, öffnet mit der Eigenbau-Fräse ein ganz neues Kapitel mit neuen Möglichkeiten. (jam) c't




OX-CNC (Tag 1)

# Stabile Portalfräse OX-CNC selbst bauen

**Die OX-CNC aus der Openbuilds-Community ist eine stabile Portalfräse zum Selberbauen. Die Konstruktion mit Rollen in Aluminiumprofilen, Nema-23-Motoren und GT3-Riemen ist stark und stabil genug für flottes Fräsen in Hartholz, Alu und Verbundmaterial. Wir haben sie gebaut.**

Von **Pina Merkert**

Portalfräsen für Profis kosten meist über 10.000 Euro. Kein Wunder, dass Hobby-Ingenieure solche Preise mit selbst gebauten Maschinen unterbieten möchten. Die ab Seite 102 vorgestellte Mostly Printed CNC lässt sich bereits für 650 Euro aufbauen, allerdings mit Motoren und Riemen in 3D-Drucker-Stabilität. Die von Mark Carew und der Openbuilds-Community entwickelte OX-CNC kommt Profigeräten dagegen erstaunlich nah, weil sie wie die meisten Profi-Fräsen ein maximal versteiftes Portal verfährt und alle Rahmenteile aus Metall bestehen.

Sie nutzt kräftige Nema-23-Schrittmotoren, fünf Millimeter dicke Aluplatten und sechs Zentimeter hohe Aluprofile für den Rahmen sowie spielfreie GT-3 Riemen für den Antrieb. Riemen und Schrittmotoren sind eine Nummer größer als bei 3D-Druckern. X- und Y-Achse laufen auf Rollen in den Einkerbungen der Profile, bei der Anzahl der Rollen übertrifft die OX den 3D-Drucker „Ender 3“ aber um Faktor 3. Viel stabiler als die OX lässt sich eine Portalfräse mit solchen Standardteilen gar nicht aufbauen. Wer für Hartmetall eine noch stabilere Maschine braucht, müsste sie wohl mit Stahlrahmen und Kugelumlaufspindeln bauen, was erheblich teurer wäre.

Wie groß die OX wird, darf man vor dem Bau selbst entscheiden. Wir haben uns für eine Fräse mit 75 Zentimeter Breite und 100 Zentimeter Tiefe entschieden. Alle mechanischen Teile kauften wir für 750 Euro als Bausatz bei der Firma „Ooznest“. Zum Redaktionsschluss war der Bausatz leider vergriffen, unserer Erfahrung nach liefert eine Suche bei eBay oder Aliexpress vergleichbare Angebote zutage. Die OX besteht aber ohnehin weitgehend aus Standardteilen für den Anlagenbau, einzig die im Bausatz mitgelieferten 5-mm-Aluplatten müssen nach den frei verfügbaren Konstruktionszeichnungen hergestellt werden. Aber auch die Aluplatten gibt es als Teilesatz bei Aliexpress. Die 750 Euro unseres Bausatzes kann man durchaus unterbieten, wenn man die Teile von den günstigsten Händlern bestellt.

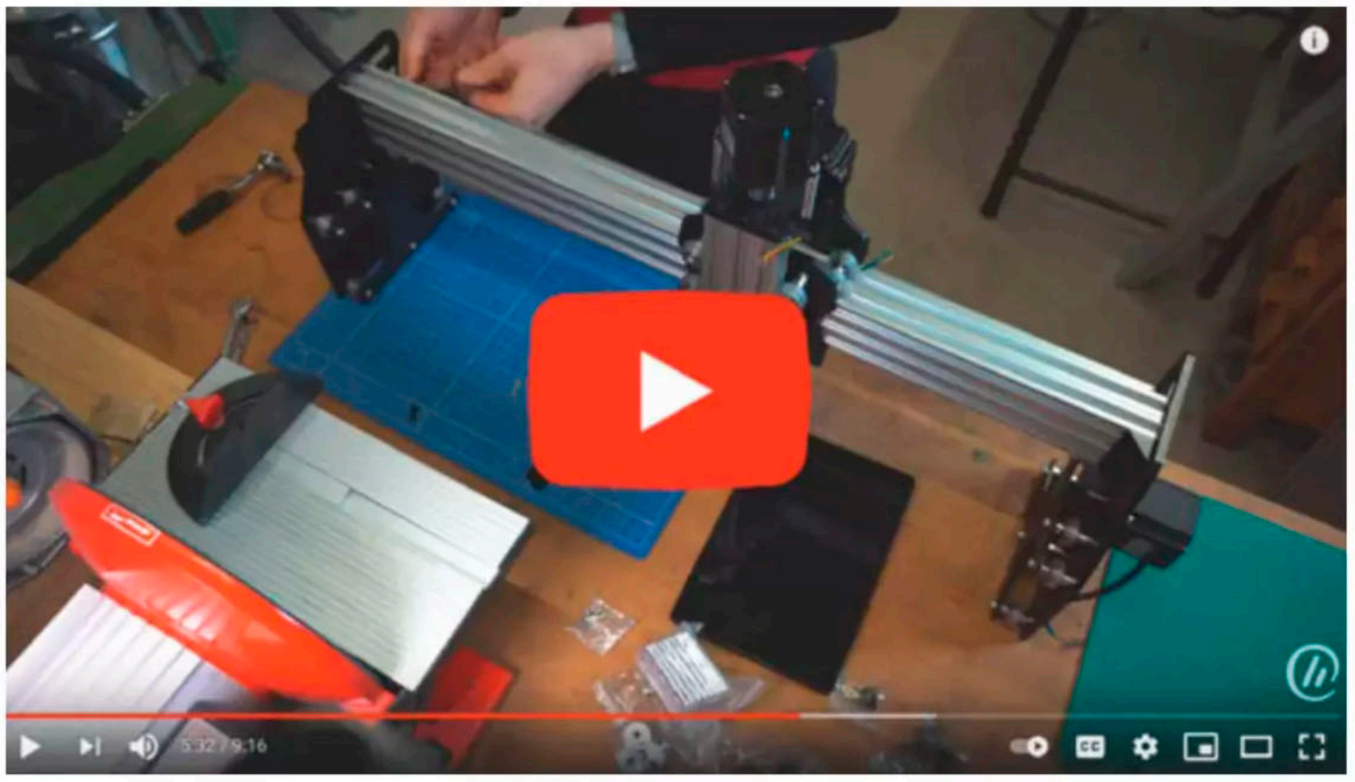
750 Euro reichen aber leider nicht für eine ganze Maschine, da der Bausatz nur Profile, Verbinder, Schrauben, Rollen, Riemen und Motoren enthält. Dazu kommen vier Schrittmotortreiber, ein Arduino Mega, Energieketten, Kabel, Endschalter und nicht zuletzt auch ein Fräsmotor. Wir haben letztlich 1200 Euro ausgegeben, weil wir die Fräse auch auf eine stabile Siebdruckplatte auf einem rollenden Ikea-Schrank namens „Enhet“ geschraubt haben. Den Bau unserer OX-CNC haben wir in elf Videos auf YouTube dokumentiert.

## Bauvergnügen mit Bauanleitung

Ooznest stellt die Bauanleitung samt vollständiger Teileliste zum Download bereit (siehe ct.de/ws31). Die Anleitung ist hervorragend! Wer Ikea-Schränke aufbauen kann, wird den mechanischen Teil mit dieser Anleitung hinbekommen. Wie beim Möbelschweden auch, kann man manche Teile seitenverkehrt zusammensetzen; da aber alles nur verschraubt ist, lassen sich Fehler leicht korrigieren. Wir genossen den Bau wie bei einem großen LEGO-Modell für Erwachsene. Auf der letzten Seite der Anleitung kamen wir am dritten Samstag an.

Nach dieser Zeit steht der Rahmen, und die Riemen samt Schrittmotoren sind installiert. Wir empfehlen, danach direkt an die Aufstellung der Fräse in der Werkstatt zu denken. Für maximale Stabilität greifen die Aufhängungen des Portalschlittens nämlich um die Profile herum, sodass man seitlich an die Rahmen nichts anschrauben kann. Setzt man die Fräse sofort auf eine Platte, kann man die Elektronik und die Befestigungen für die Energieketten

**Wie wir unsere OX-CNC gebaut haben, erzählen elf Videos, die Sie über diesen Link finden: ct.de/ws31.**



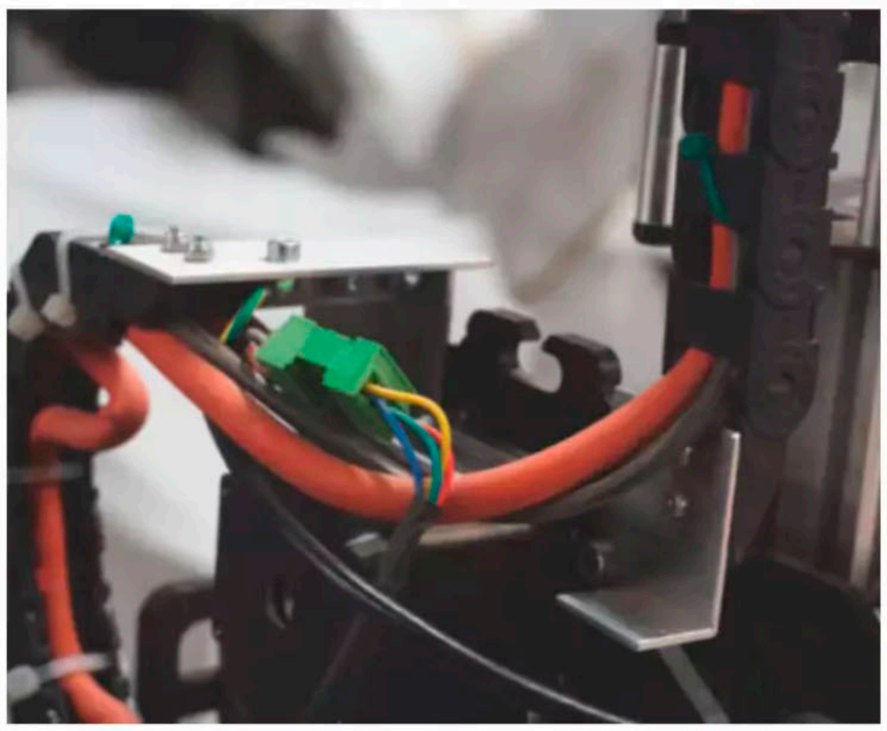
an diese Platte anschrauben. Für unseren Schrank unter der Platte konnten wir also direkt mit der Ikea-Anleitung weiterbauen.

## Verdrahtung

Die Verdrahtung dauert länger als der Zusammenbau der gesamten Mechanik. Die starken Schrittmotoren könnten herumhängende Kabel leicht abscheren, weshalb man die Kabel von Anfang an in Energieketten unterbringen sollte. Wie man die Ketten am Rahmen anbringt, erklärt die Anleitung leider nicht. Wir haben Winkelprofile aus Alu in zwei Größen bestellt und uns für die Befestigungen passende Stücke abgesägt. Wer Geduld hat, kann dafür aber auch den 3D-Drucker anwerfen. Die Energieketten müssen nämlich keine großen Kräfte aufnehmen.

Die Anschlusskabel der Schrittmotoren müssen bis zu 2,8 Ampere über mehr als zwei Meter transportieren. Vieradrige Kabel mit 0,75 mm² pro Ader sollten reichen. Ideal ist spezielles Energiekettenkabel, wir haben, um Geld zu sparen, flexibles Saunakabel mit Silikonmantel verwendet. Kabel dieser Dicke verlangen nach mindestens vier Zentimeter breiten Energieketten.

Die Anschlusskabel der Motoren enden bei uns an Schrittmotortreibern vom Typ TB6600 mit maximal 4A. Die Treiber kommen im Alugehäuse mit Anschlussklemmen, was die Verdrahtung vereinfacht. Mit den wesentlich kleineren Treibern für 3D-Drucker laufen die Motoren auch, allerdings liefern die meisten der steckbaren Platinchen nicht genug Strom, um die Motoren auszureizen. Wer wegen des ruhigeren Laufs auf die Mini-Treiber schielt, sollte auch einen Blick auf die MKS SERVO57A werfen, die den Schrittmotor als Servomotor betreiben. Laufruhige Schrittmotoren nutzen bei einer Fräse allerdings wenig, da der Spindelmotor alles übertönt, sobald er läuft.



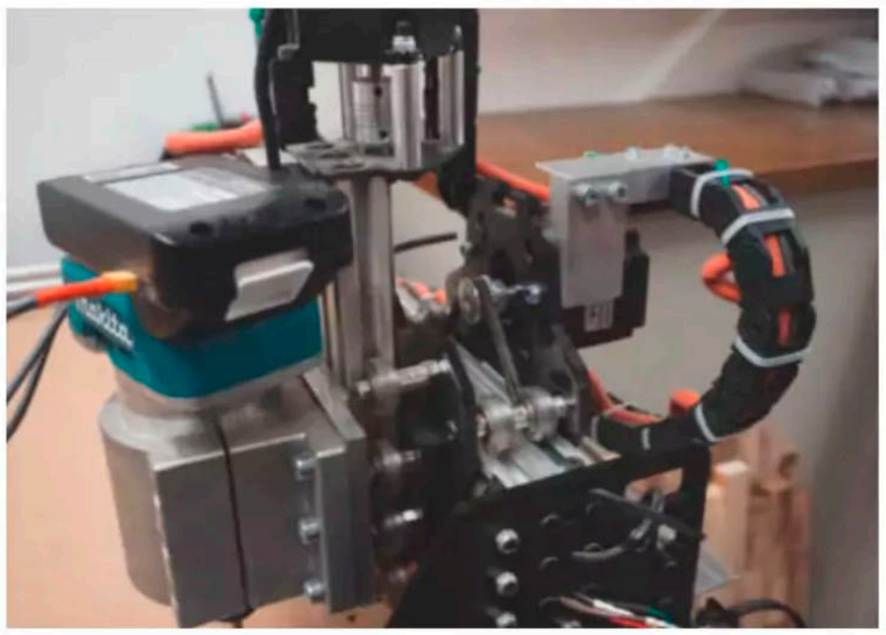
**Für die Befestigung der Energieketten gibt es ausreichend viele ungenutzte Löcher in den Aluplatten. Wie die Halter aussehen sollen, muss man sich aber selbst überlegen.**

Auch wenn eine Portalfräse ohne Endschalter funktioniert, empfehlen wir, sofort einen für jede Achse zu installieren. Denn nur mit Endschaltern kann man beispielsweise nach einem Nothalt die gleiche Position nochmal exakt anfahren. Wir haben Mikroschalter an 3D-gedruckten Haltern (STL-Dateien finden Sie über ct.de/ws31) verwendet, die wir mit Koaxialkabeln verdrahtet haben. Leider kann es elektromagnetische Einstreuungen in die Kabel geben, was dazu führt, dass die Fräse abrupt stoppt. Von der alten BNC-Verkabelung war noch etwas übrig und die geschirmten Kabel sollten induktive Einstreuungen reduzieren. Wichtiger als geschirmte Kabel ist es aber, offen leitende Endschalter zu verbauen. Wenn die Maschine fährt, fließen Ladungen dann nämlich ab. 100nF-Kondensatoren sollen als weitere Maßnahme Spannungsspitzen abflachen. Trotz aller Vorsicht beobachteten wir hin und wieder Abbrüche durch fälschlich ausgelöste Endschaltersignale. Wir mussten den Nothalt beim Anfahren des Endstops letztlich deaktivieren.

## Doppelcomputer

Die Ansteuerung der Motoren übernimmt ein Arduino Mega mit der verbreiteten Fräsen-Firmware GRBL. GRBL läuft auch auf einem Arduino Uno, der Fork für den Mega (siehe ct.de/ws31) nutzt den zusätzlichen Speicher aber für eine bessere Planung der Bewegungen. Für die Verdrahtung ziehen Sie das Pinout im Git-Repository der Firmware zurate. Wir haben uns das Leben mit einer Lochrasterplatine in passender Größe – verkauft als „Experimentier-Hat“ – leichter gemacht, die jede Menge Platz für Stecker bietet.

Den Arduino haben wir an den USB-Anschluss eines Raspberry Pi 4 gehängt, auf dem wir Cnc.js installiert haben. Was Octoprint für 3D-Drucker, ist Cnc.js für Fräsen: ein Webinterface, um die Fräse mit jedem Gerät im Heimnetz steuern zu können.



**Die gegossene Spindelaufnahme konnten wir von der Stange kaufen – ein 3D-gedruckter Halter hätte auch gereicht.**

## Fräsmotor

Wenn die Achsen schön verfahren, fehlt nur noch der Fräsmotor. Wir haben eine Oberfräse von Makita verwendet. Es gibt ein gleichwertiges Modell von DeWalt mit leicht unterschiedlichem Durchmesser. Beliebt sind auch die etwas stärkeren Motoren von AMB Elektrik (ehemals Kress) oder wassergekühlte Spindeln mit bis zu 2000 Watt von diversen chinesischen Herstellern. Auch wenn die Rahmensteifigkeit der OX möglicherweise Vorschübe zulässt, die auch eine 2kW-Spindel ausreizen, sollten Sie überlegen, ob Sie das wirklich brauchen. Während die Industrie stets auf Zeit optimiert, sind Ihnen im Hobbybereich saubere Oberflächen vermutlich wichtiger als eine gesparte Minute.

Uns reichte die Leistung der günstigen Oberfräse. Auf eine Drehzahlstellung durch die Firmware haben wir einfach verzichtet: Bevor es losgeht, schalten wir den Fräsmotor per Hand ein und drehen das Drehrad an der Oberfräse auf die gewünschte Drehzahl. Bei Holz sind die maximal möglichen 30.000 Umdrehungen pro Minute meist angemessen, bei Metall wäre es für Fräser mit sechs Millimetern oder mehr wünschenswert, langsamer als die minimalen 10.000 Umdrehungen zu können. Der Motor ist ein Kompromiss.

Als Aufnahme haben wir für 35 Euro eine Klemme aus gegossenem Aluminium verbaut. Rückblickend hätte eine 3D-gedruckte Klemme wohl für weniger Geld ebenso gut gehalten. Bei der hätten wir auch nicht zwei Aluplatten mit mehreren Bohrungen als Adapter und Abstandshalter dazwischensetzen müssen. Für das hypothetische Druckteil haben wir aber keine Erfahrungswerte, das Aluteil überzeugt mit perfekter Steifigkeit und leitet zudem Wärme ab.

**Videos, Anleitung, Verdrahtung, Firmware**

ct.de/ws31

## Erweiterungen

Nach elf Samstagen war unsere OX-CNC betriebsbereit und hatte gerade ihre Opferplatte plan gefräst. Sobald der Regelbetrieb losgeht, entstehen aber auch Begehrlichkeiten: Nervöse Gemüter dürften auf einen Notschalter Wert legen – GRBL sieht zum Pausieren einen Pin am Arduino vor. Wer Werkzeugwechsel im G-Code einbaut, wünscht sich bald eine Tastplatte, für die ebenfalls ein Pin am Arduino bereitsteht. Außerdem ist nicht zu unterschätzen, mit wie viel Schwung die Fräser Späne aus dem Werkstück schleudern. Wir fanden einige Späne mehr als zwei Meter von der Fräse entfernt. Wer nicht die ganze Werkstatt mitsamt der Riemen und Rollwege der Fräse mit Spänen bedecken möchte, wünscht sich eine Absaugung. Üblich sind da Absaugschuhe, die mit Borsten einen Raum um den Fräser schaffen, aus dem der Staubsauger die Späne saugt. Eine gute Spanabfuhr erlaubt auch größere Vorschübe und glattere Oberflächen.

## Fazit

Die OX-CNC ist ein vergnügliches Bauprojekt und eine solide Fräse für alle Arten von Holz. Metallverarbeitung ist möglich, allerdings nicht mit den beim Fräserkauf empfohlenen Vorschüben. Wer behutsam vorgeht, kann aber gute Ergebnisse erzielen. Die Aluplatten für eine weitere OX-CNC kann die Maschine auf jeden Fall herstellen. Wenn Sie jeden Samstag ein paar Stunden an der Fräse bauen, ist sie nach drei Monaten betriebsbereit.

Bedenken Sie bei der Planung von Projekten, dass, nachdem Sie die Opferplatte aufgeschraubt haben, nur noch etwa sechs Zentimeter Verfahrhöhe unterm Portal frei ist. Die Hälfte davon ist die Fräserlänge, weshalb Sie in der Praxis auf maximal drei Zentimeter dickes Material festgelegt sind. Mit einer tiefer positionierten Opferplatte fräst die OX auch dickere Bretter, allerdings geht das auf Kosten der Stabilität der Z-Achse, die dann weit ausfahren muss.

Wer die Maschine hobbymäßig einsetzt, verbringt vermutlich bis zu zehnmal so viel Zeit vor CAD- und CAM-Software wie vor der Fräse. Unterschätzen Sie nicht, wie viel Einarbeitung eine CNC-Fräse braucht. Erfahrungen aus dem 3D-Druck sind nur bedingt übertragbar, da Sie Schnittgeschwindigkeiten, Schnitttiefe, Fräsmuster und Fräserauswahl per Hand festlegen müssen. Die OX hat aber mechanisch die nötigen Reserven für anspruchsvolle CNC-Projekte. (pmk) c't




Bild: Albert Hulm

Warum CNC-Fräsen einen spielfreien Antrieb brauchen

# Optimale Parameter für die CNC-Fräse

**Die CNC rattert, die Späne verklumpen und die Fräser brechen. Die computergesteuerte Portalfräse kann viel falsch machen. Doch mit den richtigen Formeln fliegen die Späne auf jeder Maschine, sogar die wackelige Graviermaschine 3018 kann Flaschenöffner aus Alu fräsen.**

Von **Christoph Schmith**

CNC-Portalfräsen werden immer erschwinglicher. Selbstbau-Fräsen und leichte Fertiggeräte finden sich in immer mehr privaten Werkstätten. Die kleinen Geräte beschränken sich meist auf weiche Materialien wie Holz und Kunststoffe oder gravieren härtere Oberflächen mit wenig Kraft. Die Produktion echter Metallteile scheuen viele Besitzer solcher Hobby-Fräsen, weil sie befürchten, dass ihre Maschine die Belastung nicht aushält. Hinsichtlich der Bearbeitung von Stahl sind die Bedenken berechtigt, bei Leichtmetallen wie Aluminium sieht das hingegen schon anders aus.

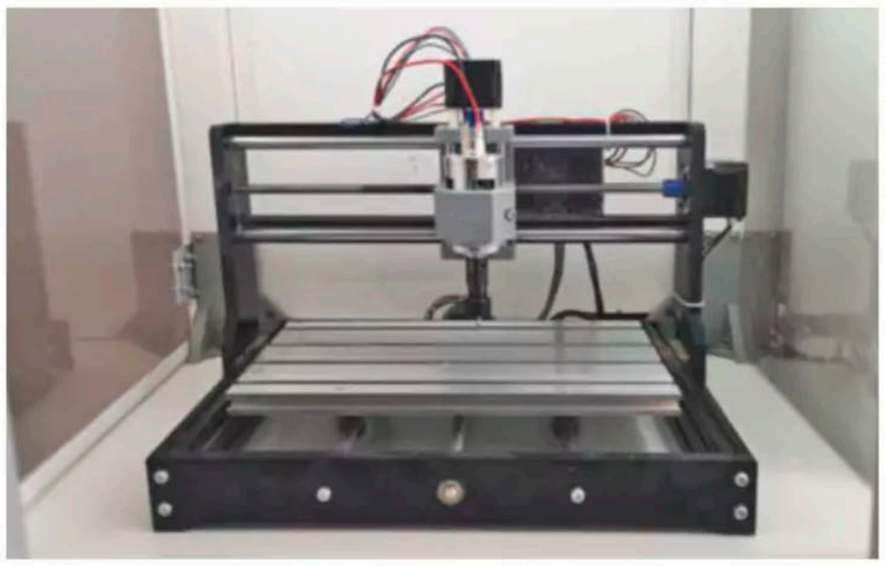


Bild: Christoph Schmith

**Zu schwach für Alu? Selbst der Hersteller Saintsu bewirbt seine Mini-Graviermaschine 3018 nicht fürs Fräsen von Leichtmetall. Mit den richtigen Einstellungen klappt es aber sogar auf dieser Mini-CNC.**

Es gibt so manche Fallstricke beim Zerspanen von Metall. Um sie zu umschiffen, benötigen Sie etwas Fachwissen zu Schnittgeschwindigkeit, Spanvorschub und -abfuhr. Die Tipps aus diesem Artikel werden es Ihnen ermöglichen, selbst auf einer Saintsu 3018, einer der leichtesten, wackeligsten und schwächsten CNC-Fräsen auf dem Markt, einen Flaschenöffner aus Aluminium zu fertigen. Sollte Ihre Fräse einem CNC-Bearbeitungszentrum näherkommen, werden Sie es einfacher haben, schneller zerspanen und dickere Fräser einsetzen können. Mit den in c't vorgestellten Selbstbau-CNCs (siehe S. 102 und S. 110) geht es beispielsweise schon sehr gut.

Dieser Artikel leitet Sie an, Schritt für Schritt die optimalen Einstellungen für Ihre Maschine zu finden. Was dieses Beispiel für Metall erklärt, ist auf Holz und Kunststoff übertragbar, wobei die Probleme bei weicheren Materialien kleiner sind.

## Strategie

Je beschränkter die Hardware, desto ausgefeilter muss die Frässtrategie sein. Ein Azubi mit einer industriellen Fräse könnte einfach einen Block Aluminium mit den Außenmaßen des Werkstücks aufspannen und mit einem möglichst großen Fräser alles zerspanen, was nicht zum Flaschenöffner gehört. Der Fräser ist dabei immer nur an einer Seite im Material (Teilschnitt), was günstig für die Belastungen am Fräser und den Abtransport der Späne ist.

Leider hat nicht jeder Fräs-Anfänger eine so große Maschine wie der Azubi. Eine kleine Fräse wie die 3018 erlaubt nur so winzige Späne, dass die Außen-nach-Innen-Strategie mehrere Stunden dauern würde. Die Alternative ist, das Objekt nur anhand seiner Kontur aus einem Blech der passenden Dicke zu fräsen. Das hat den Nachteil, dass der Fräser auf seiner gesamten Breite Material zerspanen, also im Vollschnitt arbeiten muss. Er muss dann einen langen Span von links bis ganz nach rechts abschälen. Dabei besteht eine erhöhte Wahrscheinlichkeit, dass sich der Fräser verklemmt, es zu Vibrationen kommt oder die Metallspäne nicht aus der Nut abgeführt werden können. Das führt im schlimmsten Fall dazu, dass der Fräser abbricht. Doch auch wenn er nicht bricht, wird die Oberfläche nicht so glatt wie gewünscht.

Der Kompromiss ist sogenanntes „Trochoidalfräsen“: Es funktioniert nach dem gleichen Prinzip wie das Ausfräsen der Kontur, allerdings ist hier die Nut etwas breiter als der Fräser. Der bewegt sich in kreisenden Bewegungen durch das Material und arbeitet daher im Teilschnitt. Durch die breitere Nut fliegen die Späne leichter aus der Nut und die Gefahr eines Fräserbruchs ist deutlich geringer. Einige Programme haben die Funktion direkt integriert, bei Fusion 360 mussten wir dafür ein wenig in die Trick-

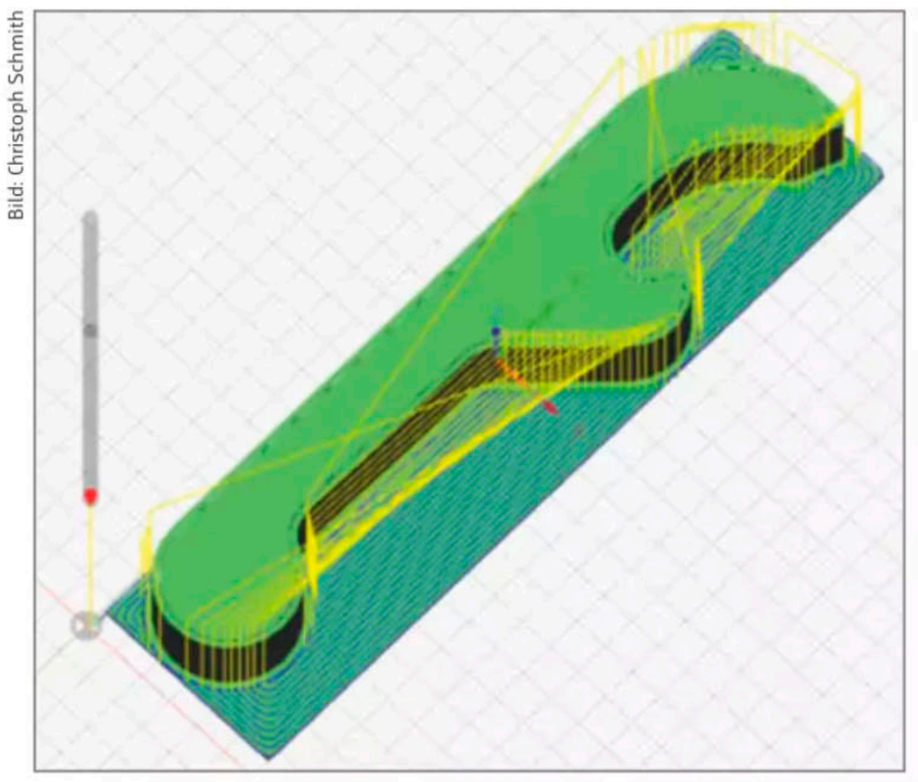


Bild: Christoph Schmith

**Mit der Strategie „alles zerspanen, was nicht Flaschenöffner ist“, wird der Fräser geschont (Teilschnitt) und die Späne fliegen ungehindert weg. Erlaubt die Maschine nur einen geringen Vorschub, dauert die Bearbeitung jedoch sehr lang.**

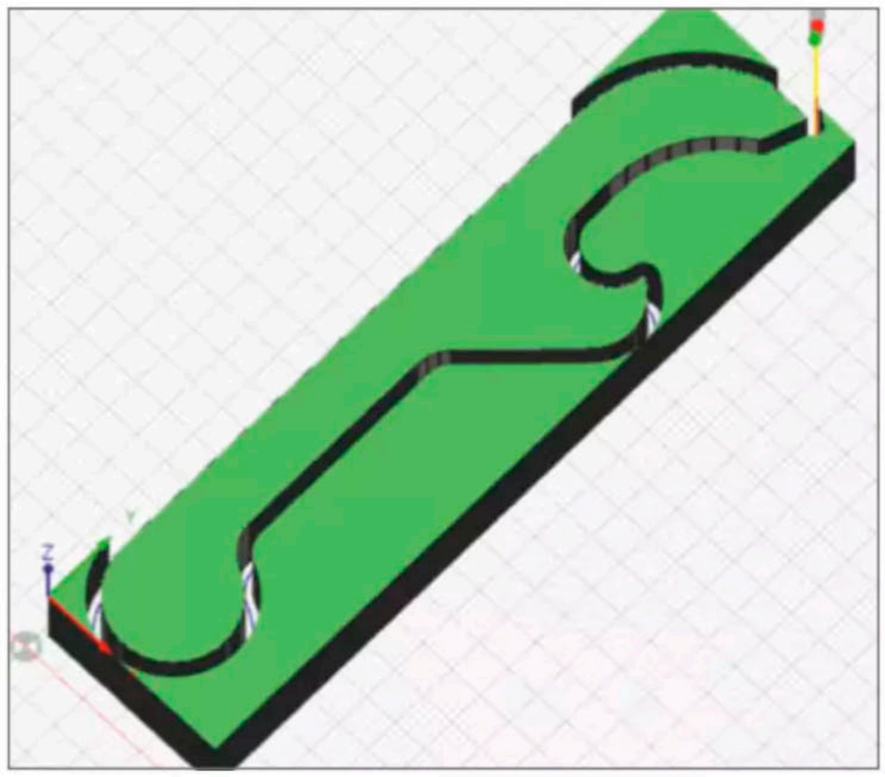


Bild: Christoph Schmith

**Schneller geht Konturfräsen. Hier bleibt überschüssiges Material einfach stehen. Der Fräser muss aber im Vollschnitt arbeiten, was Maschine und Werkzeug stärker belastet und keine so glatten Oberflächen produziert.**

kiste greifen. Zum Vergleich haben wir beide Methoden ausprobiert, genau hingehört und die Oberflächen verglichen – dazu später mehr.

## Weiche Alu-Legierungen neigen zum Schmieren

Hat man sich für eine Strategie entschieden, ist der nächste Schritt die Auswahl des Materials für das zu bearbeitende Werkstück, denn nicht alle Aluminiumlegierungen sind gleichermaßen zum Zerspanen geeignet. In der Regel gilt, je härter die Aluminiumlegierung, desto einfacher lässt sie sich zerspanen. Aluminiumprofile aus dem Baumarkt bestehen meist aus weichen Legierungen und lassen sich nur aufwendig bearbeiten, da sie dazu neigen zu schmieren. Empfehlenswerte Legierungen hingegen sind beispielsweise AlCuMgPb oder AlMg4,5Mn. Tipp: Viele Betriebe verkaufen ihre Reststücke im Internet, sodass man hier günstig an gutes Material für Experimente kommt.

## Fräser

Danach entscheidet man sich für den richtigen Fräser. Es kommt auf die Anzahl an Schneiden an, die Dicke, die Länge, Platz für die Spanabfuhr und ob die Schneiden gewendelt sind oder nicht. Die Hersteller bieten Fräser speziell für Kunststoffe, Holz, Stahl oder Nichteisenmetalle wie Aluminium an. Bei der Auswahl des Durchmessers gilt die Regel: So breit und kurz wie möglich, so schmal und lang wie nötig.

Die Anzahl der Schneiden legt fest, wie viele Späne die Maschine pro Umdrehung des Fräsers abschält. Mehr Schneiden bedeuten daher bei gleicher Drehzahl eine höhere Zerspanungsleistung und eine glattere Oberfläche, gleichzeitig steigen aber die Belastungen (Schnittkräfte), die der Fräser und die Maschine aufnehmen müssen. Gerade bei kleinen Fräsern und Fräsen mit geringer Steifigkeit sind weniger Schneiden daher oft die bessere Wahl. Je weniger Schneiden, desto leichter können die Späne abgeführt werden, was gerade bei Vollnuten von Vorteil ist. Steht hingegen eine gute Oberflächenqualität in möglichst kurzer Zeit im Vordergrund, sind Fräser mit mehr Schneiden im Vorteil. Fürs Konturfräsen auf der 3018 empfehlen wir, einen einschneidigen Fräser mit mindestens 2 Millimetern Durchmesser einzuspannen. Auf der MPCNC und der OX-CNC machte ein Einschneider mit 6 Millimetern eine sehr gute Figur.



**Die langen in der Nut aufgestauten Späne zeigen, dass die Alu-Legierung hier zu weich war.**



**Weiche Alu-Legierungen neigen dazu, zu schmieren, was zu Rattermarken auf der Oberfläche führt. Härtere Legierungen lassen sich erheblich besser zerspanen, da die Späne brechen.**

## Schnittgeschwindigkeit

Hat man sein Material und den Fräser ausgewählt, geht es daran, für diesen die individuellen Schnittparameter zu bestimmen. Im Internet finden sich leider zu viele Pauschalaussagen zu angeblich idealen Einstellungen ohne weitere Infos zu dem Material, dem Fräser, der Aufspannung und der Maschine. Tatsächlich hängen die Parameter aber von so vielen Faktoren ab, dass sie nur im seltensten Fall eins zu eins übertragbar sind.

Die Parameter müssen Sie nur einmalig für die Kombination aus Fräser und Legierung bestimmen. Prinzipiell gibt es nämlich nur drei wesentliche Größen zu ermitteln: Die Drehzahl der Spindel, den Vorschub des Fräsers und die maximale Schnitttiefe im Werkstück. Die Drehzahl *n* der Spindel errechnet sich aus der zulässigen Schnittgeschwindigkeit $v_c$ an der Schneide des Fräsers und dessen Durchmesser *d* anhand der Formel:

$n = (v_c\ [\text{m/min}] \cdot 1000) / (d\ [\text{mm}])$

Welche Schnittgeschwindigkeit dabei richtig ist, hängt sowohl vom zu bearbeitenden Material als auch vom Fräser ab und wird in der Regel vom Werkzeughersteller angegeben. Wenn Sie dort keine Angaben finden, lassen sich Richtwerte aus Tabellenbüchern wie dem Tabellenbuch Metall (siehe ct.de/waae) entnehmen. Gerade die Angaben der Hersteller sind meist am oberen Ende des Möglichen angesiedelt, zum einen, weil die Standzeit der Fräser in der Industrie eine untergeordnete Rolle spielt und zum anderen, weil die Fräser dort durch die Kühlschmiermittelzufuhr unter Idealbedingungen laufen können. Im Zweifel sollten Sie die Schnittgeschwindigkeit im Hobbybereich daher etwas niedriger ansetzen. In Ermangelung genauerer Angaben unseres Fräserherstellers haben wir unsere Werte nach dem Tabellenbuch Metall bestimmt, einer Art Bibel für Maschinenbauer.

Entscheidend für die Schnittgeschwindigkeit ist die Zugfestigkeit $R_m$ des Aluminiums. Bei unserem AlMg4,5Mn handelt es sich beispielsweise um eine Knetlegierung mit einer Zugfestigkeit von circa 270 N/mm². Je nach Behandlung des Materials kann dieser Wert auch bei der gleichen Legierung sehr unterschiedlich ausfallen, daher erfragen Sie den Wert am besten beim Verkäufer.

In den Materialdaten finden Sie zwei verschiedene Kategorien, einmal für Schruppen und einmal für Schlichten. Beim Schruppen geht es darum, in möglichst kurzer Zeit möglichst viel Material abzutragen, meist zulasten der Oberflächenqualität. Um eine glattere Oberfläche zu erhalten, können Sie an den Kanten eine dünne Schicht Material von etwa zehn Prozent des Werkzeugdurchmessers stehen lassen. Die tragen Sie in einem zweiten Durchgang mit den Parametern für Schlichten ab, was schönstmögliche Oberflächen garantiert. Für unser Experiment ist die Oberfläche zunächst jedoch unerheblich und wir beschränken uns auf das Schruppen. Bei einer empfohlenen Schnittgeschwindigkeit von 180 m/min ergibt die Formel eine ideale Drehzahl von 28.647 U/min. Bei kleinen Fräsern und langsamen Spindeln kann es vorkommen, dass der berechnete Wert über der maximalen Drehzahl der Spindel liegt. In dem Fall nimmt man behelfsweise die maximale Drehzahl der Spindel, bei der 3018 sind das 7000 U/min.

## Vorschub

Um den Vorschub *f* zu berechnen, benötigen Sie den sogenannten „Zahnvorschub“ $f_z$. Er beschreibt die Strecke, die sich ein Zahn des Fräsers pro Umdrehung in das Material arbeitet. Die Anzahl der Schneiden *z* mal den Zahnvorschub multipliziert mit der Drehzahl ergibt den Vorschubwert:

$f = n\ [\text{U/min}] \cdot f_z\ [\text{mm/Zahn}] \cdot z$

Den Wert für $f_z$ erhält man wiederum vom Fräserhersteller oder aus Tabellenbüchern. Es ist wichtig, von diesem Wert nicht allzu stark abzuweichen. Fährt die Fräse zu langsam, schält nicht jede Schneide einen Span ab und die Reibung erhitzt den Fräser, sodass er ausglühen kann. Ist die Geschwindigkeit zu hoch, wird der Fräser überlastet, biegt sich, die Maschine gerät in Schwingungen (hörbares Rattern) und der Fräser kann abbrechen. Ist der Durchmesser Ihres Fräsers nicht in der Tabelle aufgeführt, können Sie den Wert einfach linear interpolieren. Für einen 2-Millimeter-Fräser erhielten wir so einen Zahnvorschub von 0,005 Millimetern. In die Formel eingesetzt ergibt das für unseren Vorschub auf der 3018:

7000 U/min · 0,005 mm · 1 = 35 mm/min.

Schaftfräser sind mit ihrer Geometrie nicht für Bohrarbeiten ausgelegt. In Z-Richtung sollten Sie den Vorschub daher deutlich geringer wählen. Rechnen Sie als Faustregel für das Eintauchen ins Material mit etwa einem Drittel des Vorschubs in der Arbeitsebene (X und Y).

## Schnitttiefe

Die maximale Schnitttiefe ist hingegen hauptsächlich von der Stabilität der Maschine abhängig, weshalb es dafür keine Formel gibt. In der Industrie sind Schnitttiefen bis etwa der Hälfte des Werkzeugdurchmessers üblich. Im Hobbybereich sollte man eher mit 20 bis 25 Prozent des Durchmessers beginnen und sich bei Bedarf langsam steigern. Wir haben nach einigen Tests an Aluresten mit 0,5 Millimetern gute Ergebnisse erzielt. Wenn Sie eine insgesamt höhere Leistung in Form des Zeitspanvolumens [$cm^3$/min] anstreben und die Stabilität der Maschine es erlaubt, ist es sinnvoller, einen größeren Fräser mit entsprechend höherem Vorschub zu verwenden, als die Schnitttiefe zu maximieren.

## Spanbeschau

Ob diese Werte nun tatsächlich passen, bestimmen Sie am besten anhand der Form der Späne. Im Idealfall sind sie etwa halbkreisförmig und kurz. Man bezeichnet solche Späne als Bruchspäne. Erinnern die Späne eher an lange Bänder, ist es ratsam, die Schnitttiefe oder den Vorschub zu erhöhen. Bei langen Spänen besteht die Gefahr, dass sich aus ihnen ein Knäuel am Fräser bildet und der nur Späne statt Metall schneidet. Lassen sich hingegen keine Späne erkennen und gleicht der Auswurf eher Metallpulver, ist der Vorschub pro Zahn zu gering. Da der Fräser so mehr reibt als schneidet und die Wärme nur unzureichend abgeführt wird, kann im schlechtesten Fall der Fräser ausglühen, brechen oder das Aluminium schmelzen und sich eine sogenannte Aufbauschneide bilden. Dabei backt das Aluminium an den Fräser an und muss anschließend chemisch mit Natronlauge wieder aufgelöst werden. Natronlauge ist Natriumhydroxid in wässriger Lösung.

Generell sollten Sie die Maschine nie unbeaufsichtigt arbeiten lassen. Spätestens, wenn die Fräse akustisch durch Rattern auf sich aufmerksam macht oder die Tonhöhe der Frässpindel merklich sinkt, wenn der Fräser ins Material eintaucht, ist ihre Belastungsgrenze erreicht. Die Schnittkräfte lassen sich verringern, indem Sie die Schnittgeschwindigkeit und Schnitttiefe reduzieren oder alternativ einen kleineren Fräser mit weniger Schneiden nutzen. In GRBL können Sie sogar, während ein G-Code-Programm läuft, die Vorschübe im Bereich von 50 bis 200 Prozent verändern.

Profis nutzen beim Fräsen stets Kühlschmiermittel. Das kühlt Fräser und Werkstück, reduziert die Reibung und schwemmt in den üblicherweise großen Mengen auch Späne weg. Der Fräser schleudert die Späne-Kühlmittel-Suppe aber in alle Richtungen, weshalb die Maschine dann unbedingt ein dichtes Gehäuse braucht. Die im Hobbybereich beliebten Opferplatten aus MDF würden aber sofort aufquellen. In den Leistungsbereichen von Hobbyfräsen ist es in der Regel nicht notwendig zu kühlen, außer man möchte sehr weiche Legierungen bearbeiten. Die Vorteile liegen eher in einer höheren Lebensdauer des Fräsers und einer besseren Oberfläche.




**Fräser mit Aufbauschneide aus Aluminium und ausgebrochener Schneide.**

## 3018 fräst Alu

Nach Berechnung aller Formeln und ein paar Tests zur Bestimmung der Schnitttiefe haben wir das Blech mit doppelseitigem Klebeband auf einer Opferplatte fixiert und den Fräsvorgang mit den berechneten Parametern gestartet.

Tatsächlich arbeitete sich die kleine Graviermaschine mit unseren Einstellungen in etwa einer halben Stunde durch das 6 Millimeter dicke Blech. Die

betterCode()

# Qt 2021

## Die Heise-Konferenz zu Qt 6 und seinen Neuerungen

**23. Juni 2021 ONLINE**

**10 % Mai-Rabatt** für c't-Leser – Ihr Code: 2021QT4CT_L0756A

### Evolution des Cross-Plattform-Frameworks

- Wie und wann sollte ich die Migration meiner Qt-Projekte angehen?
- Was ändert sich bei den Container-Klassen?
- Wie kann ich den neuen Renderer nutzen?
- Wie schaut es mit den Lizenzen aus?
- Wie kann ich die neuen Möglichkeiten der String-Verarbeitung nutzen?

heise Developer  dpunkt.verlag

**Workshop am 7. Juli: Migration von Qt 5 auf Qt 6**

**qt.bettercode.eu**

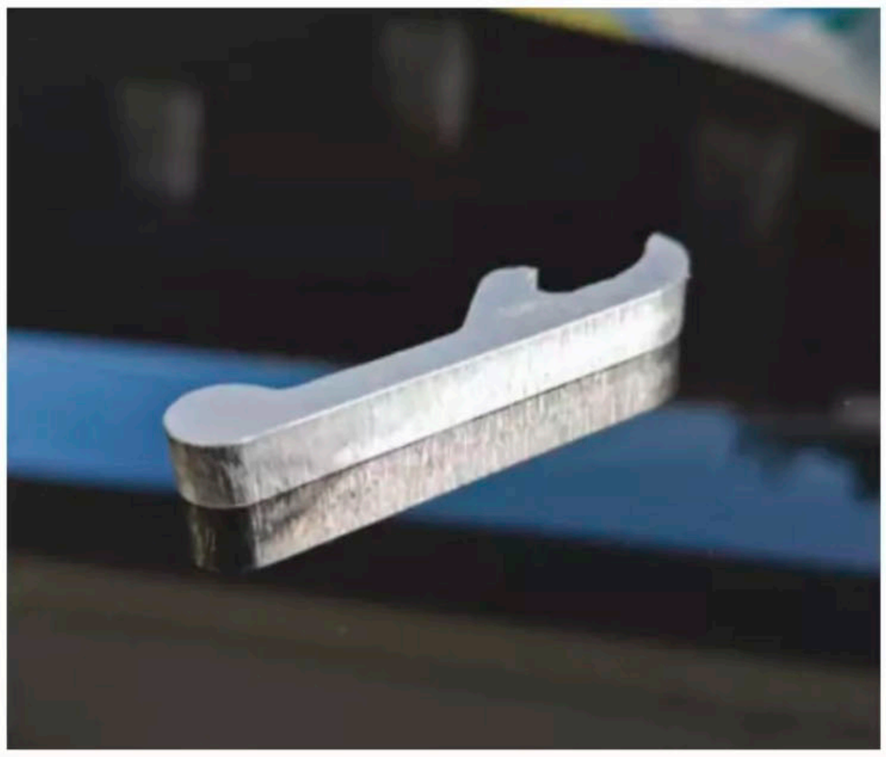


Bild: Christoph Schmith

**Nicht schön, aber selten: Das Experiment ist geglückt! Selbst die Graviermaschine 3018 konnte einen Flaschenöffner aus Aluminium produzieren.**




Bild: Christoph Schmith

**0,1 Millimeter Abweichung vom Soll sind schon ein recht grobes Maß. Die 3018 ist hier klar an ihrer Leistungsgrenze angelangt.**

Oberfläche ist definitiv nicht die schönste und 0,1 Millimeter Abweichung vom Sollmaß sind ebenfalls alles andere als optimal. Das Experiment zeigt aber, dass selbst mit der 3018 Metallbearbeitung grundsätzlich möglich ist.

## Trochoidalfräsen

Mit einer immer noch kleinen, aber schon deutlich praxistauglicheren OX-CNC erreichten wir in unseren Tests dann ein ganz anderes Niveau hinsichtlich Leistung und Stabilität. Auf der OX haben wir den Unterschied zwischen reinem Konturfräsen und Trochoidalfräsen getestet. Die Idee beim Trochoidalfräsen ist es, für das Fräsen einer Nut anstatt eines Fräsers mit identischem Durchmesser einen schmaleren Fräser zu nutzen, der eine kreisende Bewegung ausführt. Dadurch arbeitet der Fräser stets im Teilschnitt, was einige entscheidende Vorteile bringt. So sind die Kräfte auf Fräser und Maschine geringer, wodurch die Oberflächen glatter werden, was zu weniger Verschleiß an der Schneide führt. Zudem erreicht man höhere Schnitttiefen, was zum einen die Bearbeitungszeit verringert und zum anderen den Verschleiß auf einen größeren Bereich der Schneide verteilt. Und nicht zuletzt sorgt die breitere Nut dafür, dass der Fräser nicht mehr so leicht abbricht, weil er sich durch Vibratonen oder Späne verklemmt.

Um nicht unnötig viel Material abzutragen, ist es wichtig, ein gutes Verhältnis zwischen Nutbreite und Breite des Fräsers anzusetzen. Das 1,3-Fache des Fräserdurchmessers hat sich als guter Richtwert erwiesen. Da wir beim Trochoidalfräsen weniger auf Span-

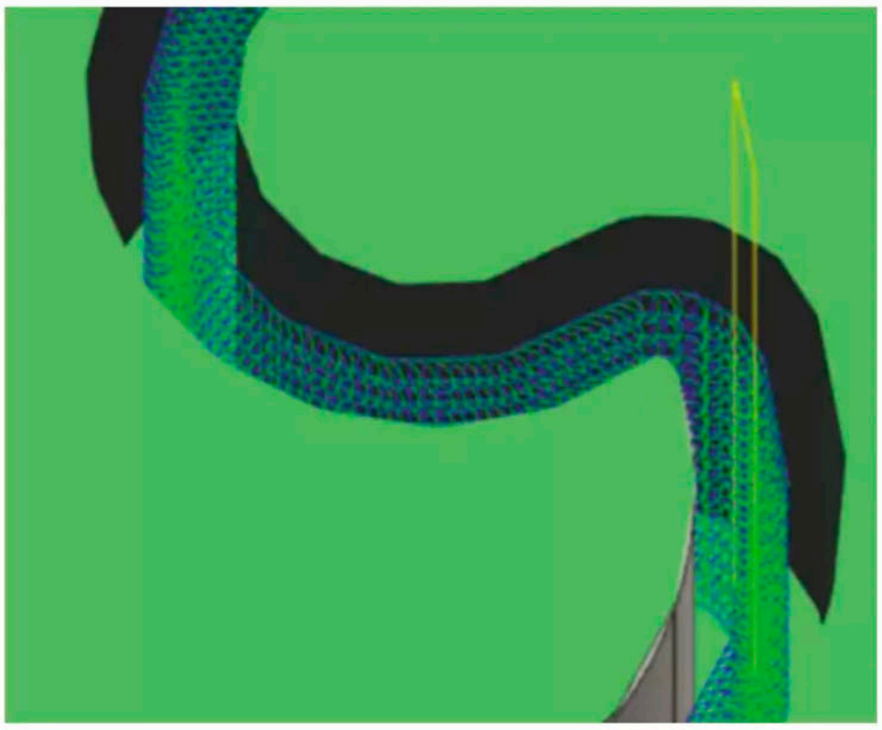
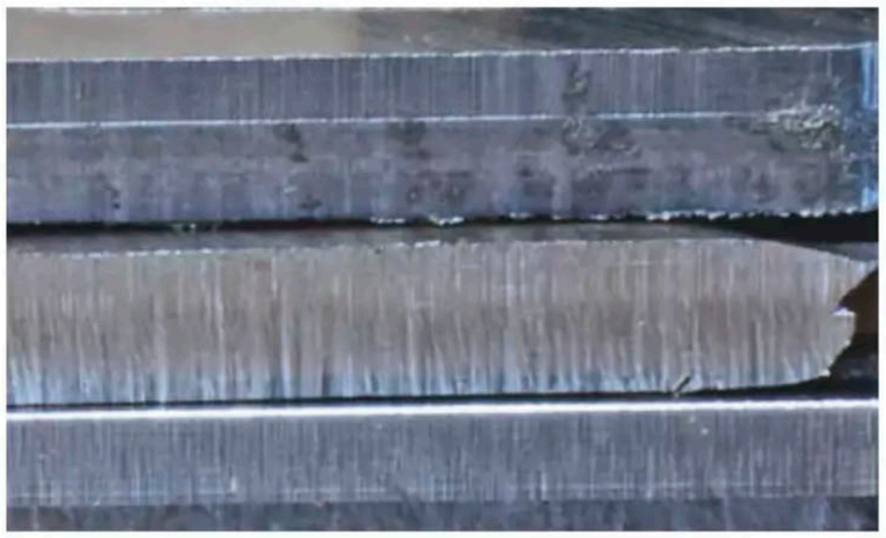


Bild: Christoph Schmith

**Weil die Fräse sich beim Trochoidalfräsen in Kreisbewegungen vorarbeitet, wird die Strategie auch „Wirbelfräsen“ genannt. Ziel ist, dass der Fräser immer im Teilschnitt arbeitet und dennoch eine Nut entsteht.**



**Oben hat der Fräser in einer zu weichen Alu-Legierung geschmiert; in der Mitte gab es leichte Rattermarken; die geschlichtete Oberfläche unten zeigt, wie es aussehen sollte.**

abfuhr achten mussten, haben wir hier direkt einen zweischneidigen Fräser mit 3,175 Millimetern Durchmesser (1/8 Zoll) und eine Nutbreite von 4 Millimetern verwendet. Das Ergebnis kann sich sehen lassen: minimale Rattermarken und eine insgesamt deutlich bessere Oberfläche als beim reinen Konturfräsen. Auch die Maßhaltigkeit fällt generell besser aus. Das beste Oberflächenergebnis erhält man wie erwartet mit einem abschließenden Schlichtdurchgang mit halbem Vorschub – entsprechend länger dauerte der Prozess dann natürlich auch.

## Fazit

Mit Hobbyfräsen können Sie auch durchaus Leichtmetall wie Aluminium bearbeiten, die Voraussetzungen sind jedoch die richtigen Einstellungen und eine geeignete Legierung. Den eigentlichen Kompromiss geht man bezüglich der Geschwindigkeit ein.

Die Gravierfräse 3018 können wir jedoch nicht empfehlen, weil sie einem einfach zu viele Kompromisse abverlangt: Die Stabilität reicht nicht für normal dicke Fräser, für kleine Fräser dreht die Spindel zu langsam und beides sogt für geringe Maßhaltigkeit. Eine Maschine mit einer Oberfräse als Fräsmotor wie bei der MPCNC (S. 102) oder der OX-CNC (S. 110) sollte es mindestens sein.

Wer bereit ist, auf einer solchen Fräse ein wenig Zeit sowohl zum Austüfteln der Einstellungen als auch für den Fräsprozess an sich zu investieren, produziert auch auf für Holz konstruierten Portalfräsen maßhaltige Metallteile. (pmk) c't

**Tabellenbuch, Erklärvideo**

ct.de/waae

## Holz

Das Vorgehen zum Berechnen und Testen der optimalen Parameter funktioniert bei Holz genau wie bei Metall. Die Schnittgeschwindigkeit ist höher, die Spindeldrehzahl deswegen auch und die Vorschübe sind erheblich größer. Die Schnitttiefe darf hier durchaus die Hälfte des Fräserdurchmessers erreichen, die meisten Hobby-Fräsen halten das aus. Rattern sollte die Maschine auch im Holz nicht und auch wenn Holz weniger gleichförmige Späne erzeugt, sollten diese wie abgeschälte Holzkringel und nicht wie Schleifstaub aussehen.

Schlichten ist bei Holz meist nicht nötig. Sollte Sie Ausriss an der Unterseite der Fräsbahn plagen, ist vermutlich der Fräser stumpf. Einen stumpfen Fräser erkennen Sie auch an Brandmarken im Holz. Die lassen sich bei V-förmigen Gravierfräsern im Bereich der Spitze nicht völlig vermeiden.

Ein wie ein Spiralbohrer rechts herum gewendelter Fräser produziert besonders bei Plattenmaterial auf der Oberkante Ausriss. Dafür wird die Unterkante perfekt, sofern Sie komplett durchfräsen, und der Fräser führt die Späne gut ab. Es gibt Downcut-Fräser, die links herum wendeln und eine perfekte Oberkante produzieren. Dafür finden Sie dann an der Unterkante Ausriss und die Spanabfuhr ist besonders schlecht. Ein häufiger Kompromiss sind für Holz daher gerade Fräser, die oben und unten kaum Ausriss produzieren und die Späne zumindest seitlich wegschleudern. Ein Spezialkompromiss sind kombinierte Up- und Downcut-Fräser (Compression Bits). Sie schneiden die Unterkante schräg nach oben und die Oberkante schräg nach unten. Der Trick funktioniert aber nur, wenn die Nut tief genug ist, damit die Stelle, an der sich die Wendelrichtung umkehrt, im Material liegt. Beim Eintauchen produzieren solche Fräser immer etwas Ausriss.





Test „Shaper Origin“

# CNC-Oberfräse Shaper Origin

**Die Shaper Origin bringt Augmented Reality in die Holzwerkstatt. Geführt wird sie wie eine Oberfräse mit der Hand, fräst dabei aber präzise an den Konturen digitaler Vorlagen entlang. Ausgestattet ist sie mit Kamerasystem, WLAN und Cloud-Anbindung. All das ist aber keine Gadget-Spielerei, sondern ein ganz neuer Maschinentyp für Holzprofis und -enthusiasten.**

Von **Jan Mahn**

CNC-Fräsen gibt es schon seit Jahrzehnten, zur Ausstattung von Tischlerwerkstätten gehören sie aber meist nur dann, wenn dort Möbel in Serien gefertigt werden. Für Einzelstücke und Reparaturen greift der Profi zu konventionellen Handwerkzeugen. Das liegt auch daran, dass sich die Arbeitsweise mit CNC sehr vom sonstigen Holzhandwerken unterscheidet: Die meiste Zeit verbringt man erst im CAD-Programm beim Entwerfen der Teile, danach im CAM-Programm beim Erstellen des Fräsprogramms. Das Fräsen der Teile geht dann vergleichsweise schnell, und vor allem reproduzierbar. Demgegenüber steht die handwerkliche Herangehensweise mit Zeichnung auf Papier, mit Messen, Anzeichnen, Sägen und Fräsen per Hand. Weniger Zeit am Computer, mehr Zeit am Werkstück.

Das Unternehmen Shapertools will diese Welten verbinden und hat sich mit der Shaper Origin eine ganz neue Geräteklasse ausgedacht: Die handgeführte CNC-Oberfräse soll das Handwerk digitalisieren, ohne den Workflow auf den Kopf zu stellen. Um beinahe traditionell arbeiten zu können, ist die Origin mit allerlei High-Tech aufgerüstet. Eine Kamera im Rücken scannt die Umgebung und berechnet die aktuelle Position der Maschine auf dem Werkstück zu jedem Zeitpunkt auf den hundertstel Millimeter genau. Schrittmotoren bewegen die Frässpindel über eine Mechanik im Boden in einem Bereich von etwa vier Quadratzentimetern in X- und Y-Richtung, ein dritter Schrittmotor steuert die Höhe.

Der Nutzer positioniert über das Touchdisplay eine Vorlagendatei auf dem Scan des Werkstücks oder zeichnet die Fräsbahnen ganz ohne Computer auf der Fräse selbst. Dann startet er den Fräsvorgang. Weil die Software genau weiß, wo die Maschine steht, führt sie den Fräser immer direkt an der Kontur entlang. Der Nutzer erledigt den groben Vorschub, die Mechanik macht die Feinarbeit. Das Fräsen selbst gleicht einem sehr einfachen Geschicklichkeitsspiel: Statt aufs Werkstück schaut man auf das Display – rund um die Kontur sieht man einen großzügigen Kreis, in dem man die Maschine ungefähr bewegen soll. Dabei kommt es nicht auf Geschicklichkeit an, auch absolute Grobmotoriker erhalten sehr gute Ergebnisse. Entfernt man sich zu weit von der Kontur, fährt der Fräser schnell aus dem Material. Bei den ersten Übungsfräsungen mit der Maschine kann es schon mal vorkommen, dass man versehentlich Macken ins Werkstück haut – wenn man ruckartig abrutscht, weil sich der Widerstand im Material geändert hat oder man mit zu hohen Zustelltiefen arbeitet. Nach wenigen Stunden Einarbeitung verschwinden solche Fehler in der neu gewonnenen Routine.



**Die Shaper Workstation ist eine Auflage für die Werkbank mit aufgedrucktem Domino-Muster und Spannvorrichtungen. Damit ist die Bearbeitung von Stirnseiten und kleinen Teilen sehr komfortabel.**

## Start-up und Familienunternehmen

Das Team, das sich die Fräse ausgedacht hat, erfüllt alle Klischees eines US-Tech-Start-ups: Die Gründer Alec Rivers und Ilan Moyer kommen vom MIT, als CEO haben sie Joe Hebenstreit verpflichtet, der vorher bei Amazon am Kindle und bei Google X an der Google Glass geschraubt hatte und sich daher mit Augmented Reality auskennt. Weitere ehemalige Mitarbeiter von Apple, Tesla und 3D-Drucker-Hersteller Makerbot haben sich Mechanik und Software ausgedacht. Schon sehr früh konnten Kunden ein Gerät vorbestellen und so die Entwicklung finanzieren. Auf der Suche nach einem Fräsmotor hat das Unternehmen den deutschen High-End-Werkzeughersteller Festool zunächst als Kooperationspartner gewonnen. Unter Holzhandwerkern sind deren Maschinen bekannt für Langlebigkeit und Präzision.

Mittlerweile hat TTS, das Mutterunternehmen von Festool, das Start-up übernommen, erhält Start-up-Kultur und -Struktur aber weiter und integriert das Team von Shapertools bewusst nicht in die Hierarchie des sehr bodenständigen Familienunternehmens aus Schwaben. Entwickelt wird in San Francisco, in Schwaben sitzen der deutsche Vertrieb und der deutschsprachige Support.

Trotz ihrer Start-up-Abstammung macht die Shaper Origin auf uns nicht den Eindruck eines typischen Start-up-Crowdfunding-Produkts. Solche neigen dazu, dass sie, getrieben von Deadlines, als „Minimum Viable Product“ halbgar auf den Markt geworfen werden – also gerade mal mit dem Allernötigsten ausgestattet, sodass man sie irgendwie verkaufen kann. Fehler löst man entweder per Software oder gar nicht. Der Shaper Origin dagegen merkt man an, dass sich das Unternehmen Zeit gelassen hat zwischen dem ersten Prototypen 2011 und den ersten verschickten Geräten Anfang 2018. Das Gerät fühlt sich für ein Präzisionswerkzeug solide und werkstatttauglich an.

Nicht nur die Hardware, auch das Betriebssystem ist eine Eigenentwicklung von Shapertools. Die Oberfläche basiert auf Qt und das System ist nicht, wie häufig in Foren behauptet, eine Android-Anpassung. Die Software hat eine kurze Bootzeit. Wenn man den Stecker einsteckt, ist die Maschine nach 35 Sekunden startklar. Herunterfahren kann man das System nicht. Den Stecker kann man einfach jeder-

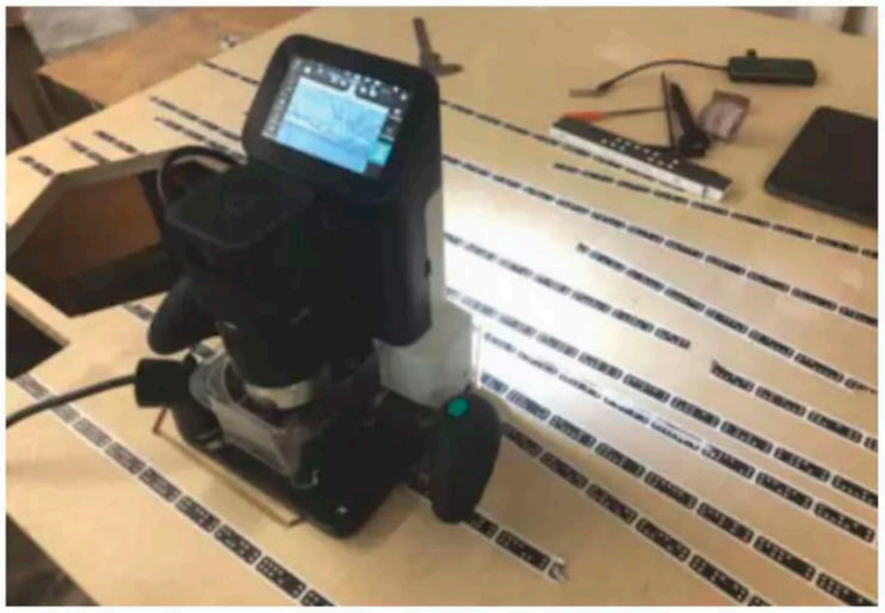
zeit ziehen, denn jede Änderung wird sofort gespeichert und es gehen keine Daten verloren.

## In der Werkstatt

Vor dem Fräsen muss die Shaper Origin ihre Umgebung kartografieren. Dafür gibt es das sogenannte ShaperTape, ein Klebeband mit aufgedruckten Dominosteinen. Das klebt man irgendwie in mehreren Streifen auf das Werkstück, genaues Ausrichten ist nicht nötig. Zwei 45-Meter-Rollen liegen bei, Ersatzrollen kosten 16 Euro. Wichtig ist, dass der Bereich hinter der Fräse mit dem Klebeband versorgt ist. Das Scannen geht schnell: Man schiebt die Fräse einmal über das Werkstück, danach kennt sie anhand des Domino-Musters ihre Umgebung. In der aktuellen Software kann man auch verschiedene Umgebungen verwalten. Weil das Werkstück nahezu beliebig groß werden kann, sind für CNC-Fräsen unübliche Szenarien möglich, zum Beispiel eine bereits eingebaute Küchenarbeitsplatte oder ein verlegter Fußboden.

Sobald man sich einmal mit der Magie der Bilderkennung vertraut gemacht hat, kann man mit der Arbeit beginnen: Nach dem Scan der Dominosteine sieht man auf dem Display das Werkstück zum Zeitpunkt des Scans und kein Live-Bild – der Mittelpunkt des Bildes ist genau dort, wo sich der Fräser befindet, also unter der Fräse. Verschiebt man das Gerät, kann man live beobachten, wie präzise das Tracking-System arbeitet.

Zum Fräsen braucht man jetzt eine Vorlage in Form einer SVG-Datei. Diese kann man entweder auf dem PC auf einen USB-Stick kopieren und diesen an die Maschine anstöpseln, oder über eine WLAN-Verbindung aus der Cloud des Anbieters, dem ShaperHub, herunterladen. Die Internetanbindung ist aber optional, und die Fräse arbeitet auch ohne. Im ShaperHub ist eine Community entstanden, die ihre Projekte dort mit anderen teilt und neben den SVG-Dateien weitere Erklärungen und Bilder veröffentlicht – vergleichbar mit Thingiverse für 3D-Druck-Projekte. Von professionellen Möbeldesignern gibt es anspruchsvolle Premium-Projekte zu kaufen, die sehr gut Schritt für Schritt dokumentiert sind. Die Community-Idee ist keine Totgeburt, zum Zeitpunkt unseres Tests fanden wir 913 Projekte online, darunter auch viele Vorlagen für gängige Beschläge, Scharniere und Griffe aus dem Möbelbau. Zusammen mit der Möglichkeit, Vorlagen mit einem Raster an den Kanten des Werkstücks auszurichten, hat man Beschläge sehr leicht genau passend platziert. Leider fehlt im ShaperHub aktuell eine Möglichkeit, Projekte anderer Nutzer zu kommentieren und Fragen zu stellen.



**Mit Notebook und High-Tech-Fräse in der Werkstatt: Das Arbeiten ganz ohne Anzeichnen und Ausmessen ist ungewohnt, doch schon nach wenig Einarbeitung hat man die Arbeitsweise verinnerlicht.**

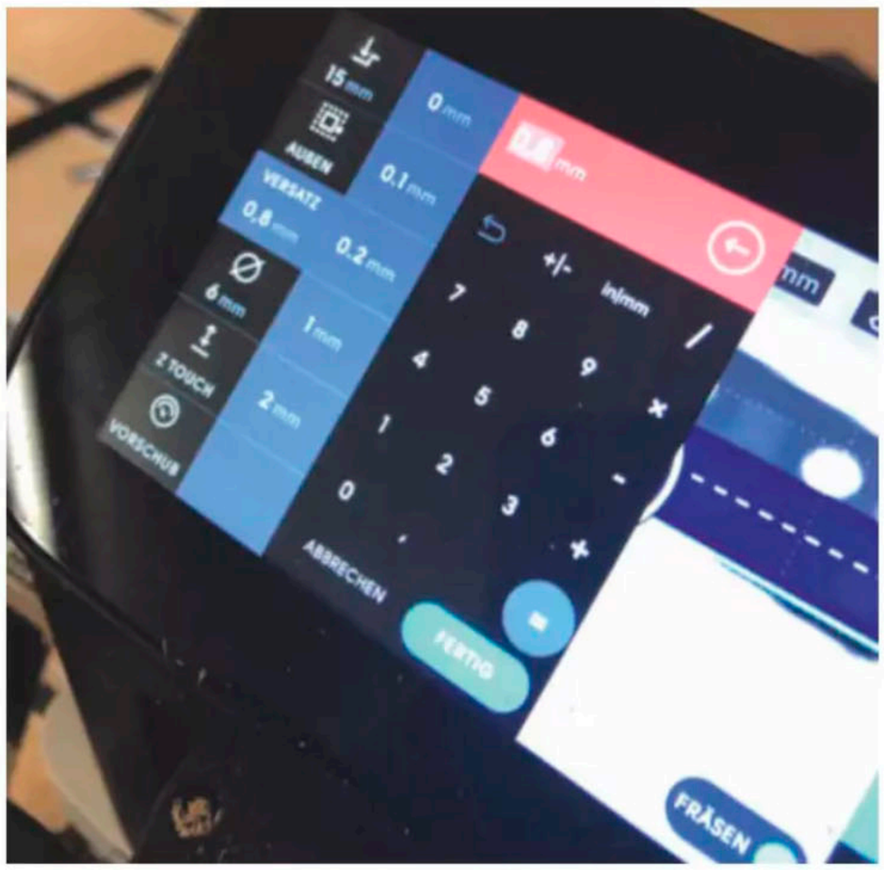
Zum Positionieren der SVG-Dateien haben sich die Entwickler für eine intuitive Alternative zu Wisch- und Zoomgesten auf dem Touchscreen entschieden. Die ganze Fräse funktioniert wie eine Maus, man schiebt und dreht sie auf dem Werkstück und platziert so die Vorlage. Nach demselben Prinzip kann man einfache Formen wie Linien, Kreise, Rechtecke, Schriftzüge und Polygone auch direkt auf der Fräse zeichnen.

In den Handgriffen links und rechts befinden sich ein orangefarbener und ein grüner Knopf. Diese stehen in der Software oft für „Abbrechen“ und „OK“. Leider gibt es ein paar Dialoge, in denen diese Knopfbedienung fehlt – dann muss man den Griff loslassen und das Touchdisplay benutzen. Unsere Anregung, das anzugleichen, notierte man sich bei Shapertools für die Weiterentwicklung der Software.

Von 2017 bis 2020 gab es insgesamt acht große Software-Updates für die Maschine, die neue Funk-



**Eine Zapfenverbindung mit einer selbstgeplanten sechseckigen Form ist keine triviale Aufgabe. Mit der Shaper Origin gelang sie maßhaltig.**




**Die Fräsparameter verwaltet man links auf dem Touchscreen. Auf einem Blick kann man zum Beispiel Tiefe und Fräserdurchmesser ablesen.**

tionen nachrüsteten und Bedienkonzepte optimierten. Ende 2019 kam zum Beispiel eine Erweiterung hinzu, um Fingerzinken zu fräsen, eine typische Anforderung im Möbelbau.

Zum Test unter Realbedingungen haben wir auf dem PC in einem Vektorgrafikprogramm einen dreibeinigen Hocker gestaltet, in den ShaperHub hochgeladen, aus 24 Millimeter dicken Birken-Multiplex-Platten gefräst und zusammengebaut. Eine Besonderheit im Vergleich zu typischen CNC-Möbeln sind die Zapfenverbindungen – wenn man das Werkstück hochkant einspannt, kann man mit der Shaper auch Stirnseiten bearbeiten. Eine ausführliche Dokumentation, in der wir dieses Projekt und die Arbeit mit der Shaper Origin vorstellen, finden Sie über ct.de/wqmg. Im Video bekommen Sie einen Eindruck von der Arbeit mit der Maschine.

## Fazit

Geeignet und gedacht ist die Fräse für alle Materialien, bei denen man nicht kühlen muss: Vorwiegend Hölzer aller Art, aber auch Acrylglas und Schaumstoffe. Für Alu oder Stahl sollte man zur stationären CNC greifen. Bisher noch nicht vorgesehen ist das Fräsen von 3D-Reliefs.

Alle Schritte, nicht nur Fräsen und Umgebungserkennung selbst, auch der Arbeitsablauf vom ShaperHub bis zum Platzieren der Vorlagen auf dem Werkstück, machten auf uns im Test einen durchdachten und praxistauglichen Eindruck. Obwohl die Shaper Origin die erste und bisher einzige Maschine einer neuen Klasse ist, handelt es sich nicht um einen halbfertigen Prototypen. An Verarbeitung und Software haben wir im Test keine entscheidenden Schwachstellen finden können. Bisher wurden die Kunden mehrmals im Jahr mit neuen Funktionen per Update versorgt.

Mit einem Preis von 3440 Euro ist die Shaper Origin ganz klar für professionelle Holzhandwerker konzipiert. Für Profis kann sich die Maschine schnell lohnen, weil sie zeitintensive Arbeiten wie Schablonenbau ersetzt und Routinearbeiten wie das Einlassen von Beschlägen stark vereinfacht. Außerdem eröffnet sie ganz neue Einsatzmöglichkeiten für eine CNC-Fräse: Das Werkstück kann nahezu beliebig groß sein – auch eine Fräsung in einem hunderte Quadratmeter großen Holzfußboden ist möglich. Aus unserer Sicht muss man kein typischer PC- und CNC-Nutzer sein, um mit dem Bedienkonzept warm zu werden. Hilfreicher ist Erfahrung mit einer konventionellen Oberfräse.

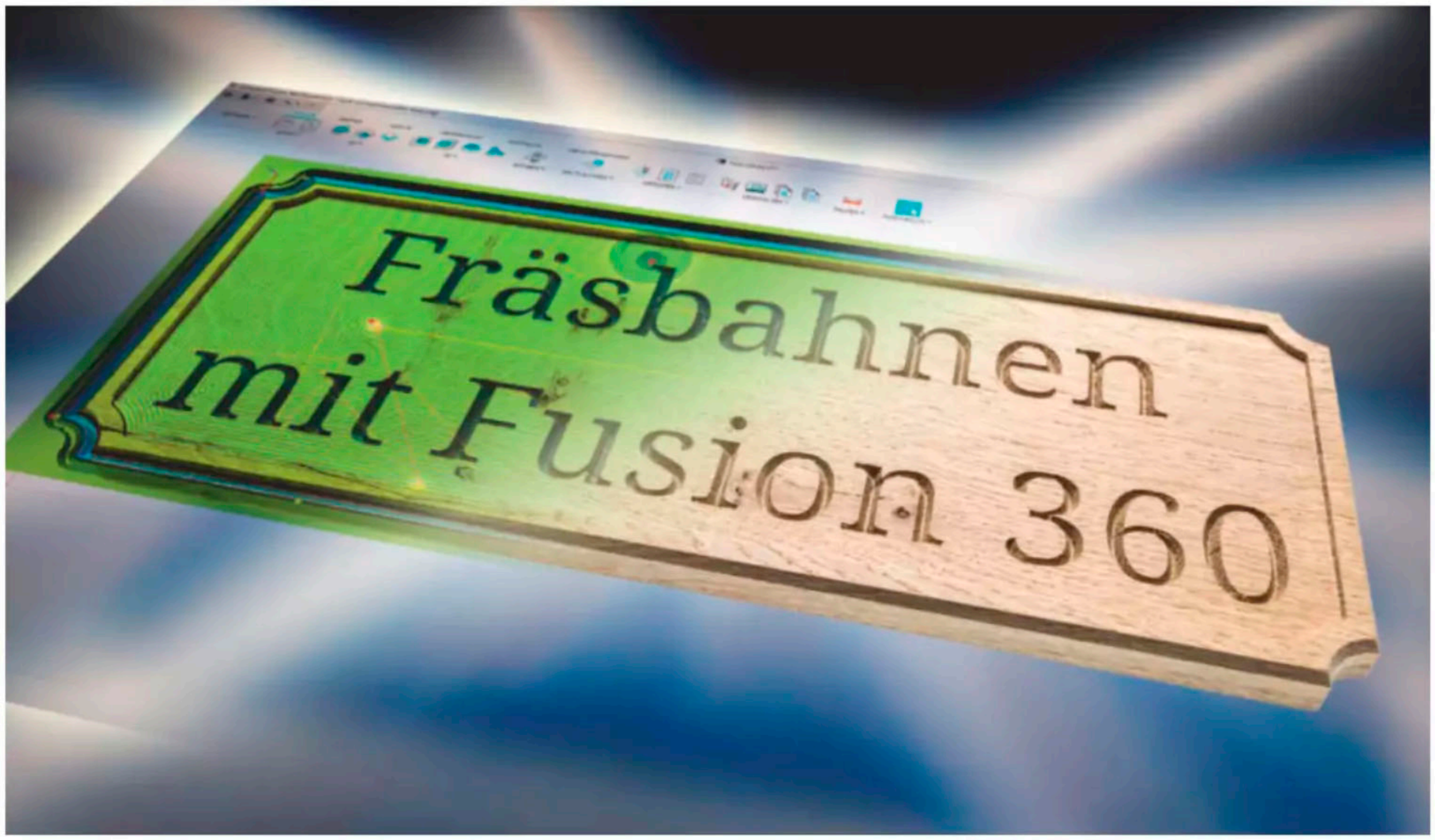
Von unserem deutschen Ansprechpartner bei Shapertools erfuhren wir, dass viele der Geräte aktuell an Privatkunden verkauft werden. Wer als privater Bastler bereit ist, das Geld auszugeben, bekommt neue gestalterische Möglichkeiten wie mit kaum einer anderen Maschine. Dabei ersetzt die Origin aber keinesfalls den Holzwerker und dessen Kreativität, sondern hilft bei der Umsetzung guter Ideen. Erfahrungen mit Vektorgrafik- oder CAD-Programmen sind hilfreich für eigene Projekte, die Premium-Projekte im ShaperHub machen Lust auf anspruchsvolle Arbeiten mit Massivholz. Wer das Geld aufbringen will, sollte für weitere 476 Euro auch die Shaper Workstation kaufen. Mit diesem Werkbankaufsatz mit aufgedrucktem Domino-Muster kann man auch Stirnseiten und Kleinteile gut einspannen und fräsen. Die Ergebnisse unserer Testprojekte waren beeindruckend, wir kamen dank CNC-Unterstützung vollständig ohne Reißnadel und Bleistift aus. (jam) c't

**Video und Zeichnungen unseres Testprojekts**

ct.de/wqmg

**Shaper Origin**

| Handgeführte CNC-Oberfräse | |
|---|---|
| Hersteller, URL | Shapertools, https://shapertools.com |
| Laden von Vorlagen | WLAN (über den ShaperHub), USB-Stick |
| Format für Vorlagen | SVG-Dateien |
| Fräserdurchmesser | 8 mm (Spannzangen für 6 mm und 1/8 Zoll optional) |
| Leistung Fräsmotor | 720 W |
| Preis | Origin 3440 €, Workstation 476 € |




# CNC-Programme mit Fusion 360 erzeugen

**CNC-Fräsen arbeiten Programme in der einfachen Sprache G-Code ab. G-Code-Programme entstehen üblicherweise in einer CAM-Software, die ausgehend vom 3D-Modell die richtigen Bewegungen für die Maschine plant. Mit Fusion 360 gelingen auch Einsteigern professionelle Fräsbahnen.**

Von **Florian Lapport**


| | |
|---|---|
| CNC-Programme mit Fusion 360 erzeugen | 126 |
| Onshape: 3D-Designs im Browser | 136 |
| Ersatzteile konstruieren mit Onshape | 146 |

Eine CNC-Fräse kann beliebig komplexe Werkstücke fertigen. Zwei Selbstbau-Fräsen für Einsteiger haben wir bereits vorgestellt und verglichen (siehe ab S. 90 und S. 102 -113). Die Wundermaschinen fertigen aber nur etwas, wenn der Computer die für die Bewegung nötigen Befehle liefert. Für die Planung der Fräsbahnen gibt es Programme fürs „Computer Aided Manufacturing“ (CAM), beispielsweise Fusion 360. Sie stellen dem Anwender mächtige Werkzeuge bereit, um Wege für die Maschine zu planen. Was die Fräse machen soll, muss ein Maschinist aber letztlich noch per Hand festlegen. Dieser Artikel zeigt einen solchen Workflow mit der kostenlosen Version von Fusion 360 fürs Herstellen eines Türschildes – vom Modell bis zum Surren der Motoren.

## Fusion 360 kostenlos

Autodesks CAD-Software Fusion 360 gibt es für Privatanwender in einer kostenlosen Version mit eingeschränktem Funktionsumfang. Das Computer Aided Design (CAD), also das Konstruieren von 3D-Modellen, ist die Hauptfunktion der Software. Aber selbst die beschränkte Variante bringt zusätzlich alle Funktionen mit, damit Fusion 360 als CAM-Tool professionelle Fräsbahnen für die heimische CNC-Maschine erzeugt. Autodesk setzt bei seinen Programmen auf Cloud-Modelle, weshalb man zunächst ein Konto beim Hersteller anlegen und sich als Privatanwender registrieren muss (siehe ct.de/wzss). Im neu erstellten Konto finden Sie dann unter „Produkte und Services“ einen Download-Link für Fusion 360.

Nach dem Herunterladen, Installieren und Anmelden finden Sie sich zunächst in der Konstruktionsumgebung wieder. Den aktuellen Arbeitsbereich erkennen Sie an dem großen Button „Konstruktion“ oben links. In diesem Bereich arbeitet Fusion 360 als CAD-Programm. Daneben enthält Fusion weitere Arbeitsbereiche mit anderen Werkzeugen – dazu später mehr. Zunächst nehmen Sie ein paar Grundeinstellungen vor. Die Einstellungen finden Sie ganz oben rechts mit einem Klick auf Ihre Initialen, dann im Menü unter „Voreinstellungen“. Für die Arbeit mit der CNC-Fräse ist es sinnvoll, unter „Allgemein“ die „Vorgabeausrichtung beim Modellieren“ auf „Z nach oben“ einzustellen. So entwirft man gleich in dem Koordinatensystem, in dem auch die Maschine fährt. Außerdem sollten Sie prüfen, dass metrische Einheiten als Standard eingestellt sind. Alles Weitere belassen Sie zunächst bei den Voreinstellungen.

## CAD mit Fusion

Fräsbahnen definiert Fusion immer passend zu einem 3D-Modell, weshalb Sie das Namensschild zuerst konstruieren müssen – eine ideale Gelegenheit, ins C-A-Design mit Fusion einzusteigen.

Starten Sie das 3D-Modell zuerst mit einer zweidimensionalen Zeichnung. Die dient als eine Art Grundriss, dem Sie später eine Dicke geben. Den Knopf mit dem Tooltipp „Skizze erstellen“ finden Sie in der Werkzeugleiste oben ganz links. In der 3D-Ansicht sehen Sie nun drei gelbe Quadrate, die orthogonal zueinander stehen. Wählen Sie das von X- und Y-Achse (die rote und die grüne Achse) aufgespannte aus. Die Ansicht dreht sich nun automatisch in die Draufsicht und die Werkzeugleiste wechselt zu Zeichenwerkzeugen. Die bleiben stehen, bis Sie den grünen Haken ganz rechts in der Leiste mit der Aufschrift „Skizze fertig stellen“ drücken.

Bleiben Sie vorerst im Zeichenmodus und wählen Sie das Werkzeug „Rechteck mit zwei Punkten“, das zweite in der Leiste. Sobald das Werkzeug ausgewählt ist, erscheint ein kleines Fenster „Skizzenpalette“, in dem Sie unter Optionen auf das Icon rechts neben „Konstruktion“ klicken und damit festlegen, dass das Rechteck zunächst nur als Konstruktionslinien gezeichnet wird. Mit einem Klick auf die Zeichenfläche legen Sie nun den linken oberen Eckpunkt des Rechtecks fest, das später das Türschild werden soll. Danach können Sie die Länge und Breite des Rechtecks in Millimeter eintippen. Um zwischen den Eingaben für Länge und Breite zu wechseln, nutzen Sie die Tab-Taste. Das Beispiel verwendet 70 Millimeter für die Höhe und 180 Millimeter für die Breite. Nachdem Sie die Maße mit der Enter-Taste bestätigt haben, erscheinen das Rechteck aus braun gestrichelten Linien und in Schwarz die Maße. Die Größe dieses Rechtecks können Sie auch später noch beliebig anpassen.

Als Nächstes konstruieren Sie als Dekoelement runde Ecken mit dem Werkzeug „Bogen durch Mittelpunkt“. In der Werkzeugleiste sammelt Fusion nur die am häufigsten genutzten Werkzeuge, den Bogen finden sie im Menü unter „Erstellen“ unterhalb der Zeichenwerkzeuge in der Kategorie „Bogen“. Im Fenster „Skizzenpalette“ deselektieren Sie nun zunächst die Konstruktionslinien, um normale Linien zu zeichnen. Setzen Sie mit einem ersten Klick den Kreismittelpunkt in eine Ecke des Rechtecks und tippen Sie danach den Radius von 12,5 Millimetern ein. Ohne den Radius mit Enter zu bestätigen, können Sie mit dem zweiten Klick den Startpunkt

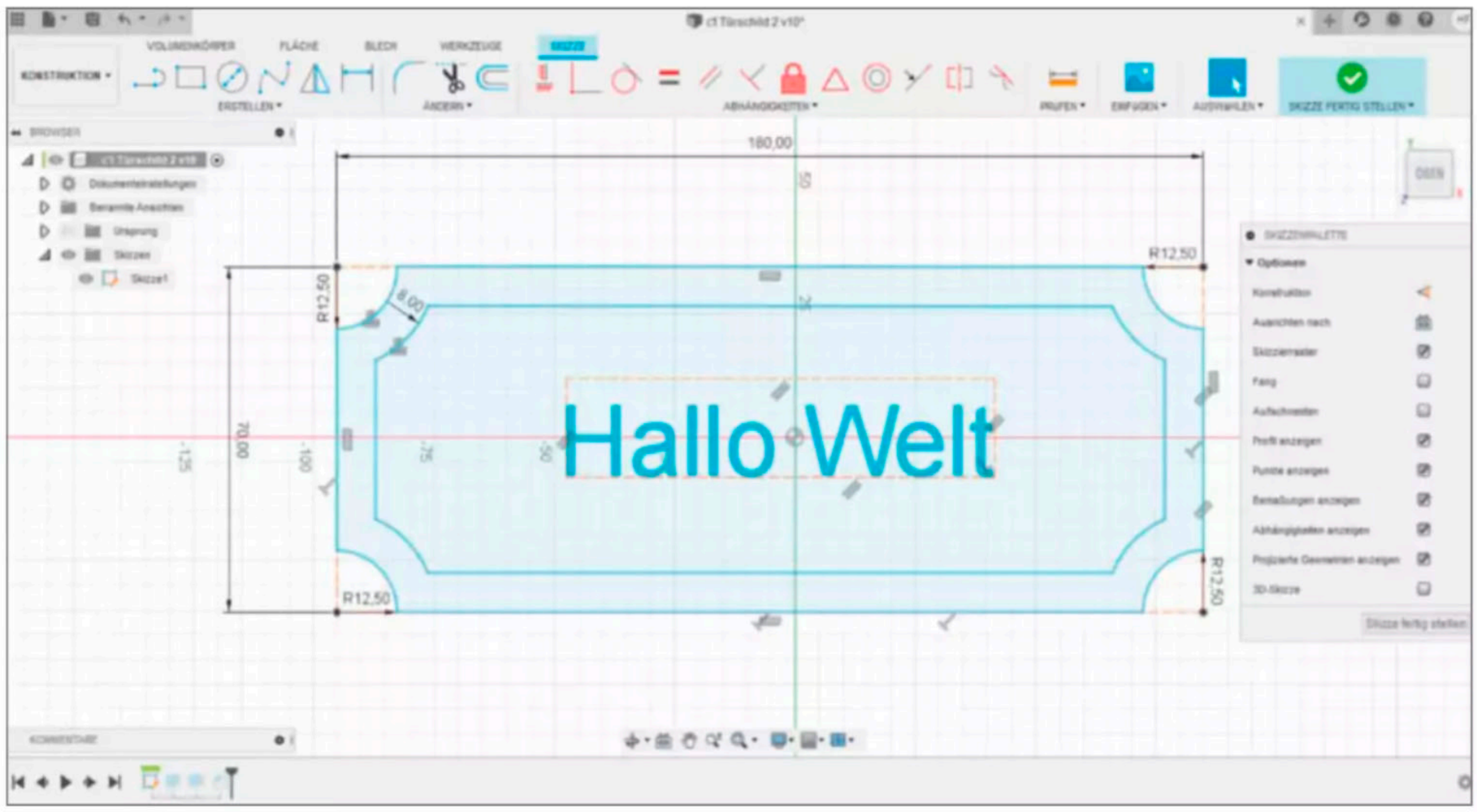
des Bogens setzen und mit dem dritten den Endpunkt. Start- und Endpunkt sollten in Fusion automatisch auf den Konstruktionslinien des Rechtecks einrasten. Sollte das bei Ihnen nicht passieren, fügen Sie die Abhängigkeit nachträglich hinzu. In der Werkzeugleiste gibt es dafür den Abschnitt „Abhängigkeiten“. Um die Punkte nachträglich an die Linien zu binden, wählen Sie die Abhängigkeit „Koinzident“ aus und klicken dann jeweils nacheinander auf den Punkt und die Kante, die aufeinanderliegen sollen. Abhängigkeiten schränken Freiheitsgrade ein und werden Ihnen beim Konstruieren mit CAD-Programmen immer wieder begegnen. Zeichnen Sie nun einen kleidsamen Viertelkreis aus durchgezogenen Linien in jeder der vier Ecken des Konstruktionslinien-Rechtecks.

Vervollständigen Sie die Form nun mit dem Linienwerkzeug, das erste in der Werkzeugleiste, indem Sie die Endpunkte der Bögen verbinden. Die Kontur des Türschilds ist damit fertig.

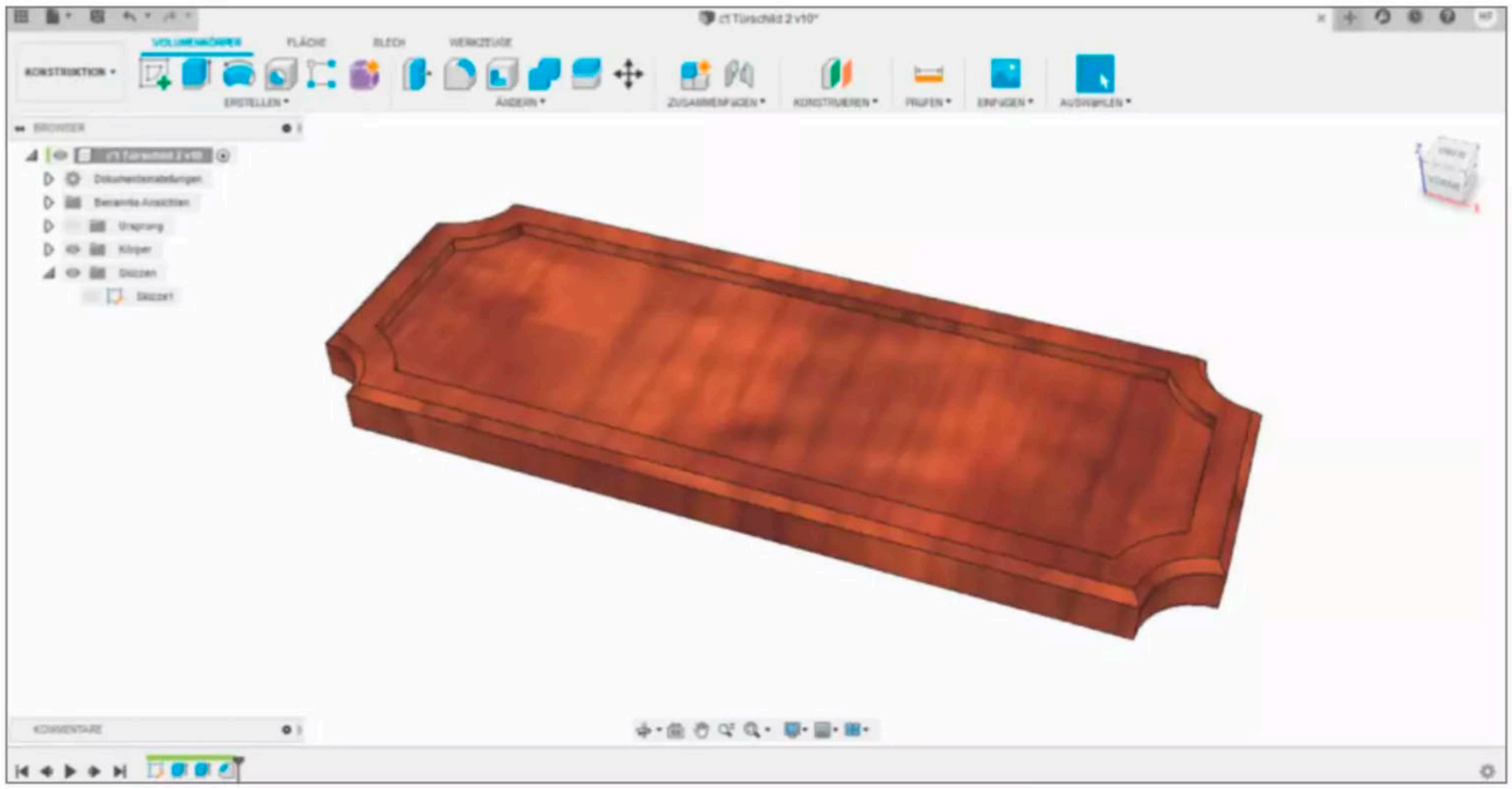
Mit dem „Versatz“-Werkzeug, das Sie entweder in der Werkzeugleiste oder im „Ändern“-Menü finden, können Sie nun die Kontur nach innen mit einem Abstand von acht Millimetern versetzen, um eine zweite Kontur zu erhalten. Falls nach der Eingabe des Abstands der Versatz nach außen statt nach innen erfolgt, setzen Sie einfach ein Minus vor die Zahl, um die Richtung umzukehren.

Als letzten Schritt in der Skizze platzieren Sie den Namen in der Mitte der Konturen. Im Menü „Erstellen“ finden Sie das Werkzeug „Text“, mit dem Sie ein Textfeld in die Skizze einfügen. Falls der Text gedreht erscheint, können Sie ihn mithilfe des Punktes mit einem Kreisbogen dahinter drehen. Die Position können Sie erst verschieben, wenn Sie das Textwerkzeug beendet haben. In der Skizze müssen Sie dann nur die Maus auf einem Buchstaben gedrückt halten, um den Text zu verschieben. Mit einem Doppelklick auf die Schrift kommen Sie dann jederzeit wieder zurück in die Texteinstellungen. Probieren Sie verschiedene Schriftgrößen, Positionen und Schriftarten aus. Sobald es Ihnen gefällt, drücken Sie auf „Skizze fertig stellen“.

Falls Ihnen das Türschild insgesamt zu klein vorkommt, können Sie jederzeit die Skizze wieder aktivieren, indem Sie am unteren Bildschirmrand in




**Beim CAD ist es üblich, mit einer flachen Zeichnung zu beginnen und ihr in einem zweiten Schritt eine Dicke zu geben. Für das Türschild verdicken Sie zweimal: einmal die äußere Kontur bis zur Dicke des Bretts und einmal die innere Kontur für eine dekorative Vertiefung. Der Text bleibt vorerst 2D.**




**Eine Fase, also eine abgeschrägte Kante um die Kontur, komplettiert die 3D-Konstruktion. Das fertige CAD-Modell dient als Grundlage zum Erstellen der Fräspfade für die CNC-Maschine.**

der „Timeline“ doppelt auf das Skizzensymbol klicken. Dann springt Fusion gewissermaßen in der Zeit zurück zu dem Zeitpunkt, bevor Sie die Skizze fertig gestellt hatten und Sie können in der Skizze zum Beispiel durch Doppelklick auf die Bemaßung die Werte ändern. Alle Schritte, die Sie nach der Skizze konstruiert hatten, ändert Fusion beim erneuten „Fertigstellen“ der Skizze automatisch auf die neuen Maße und Einstellungen.

## Erhaben konstruiert

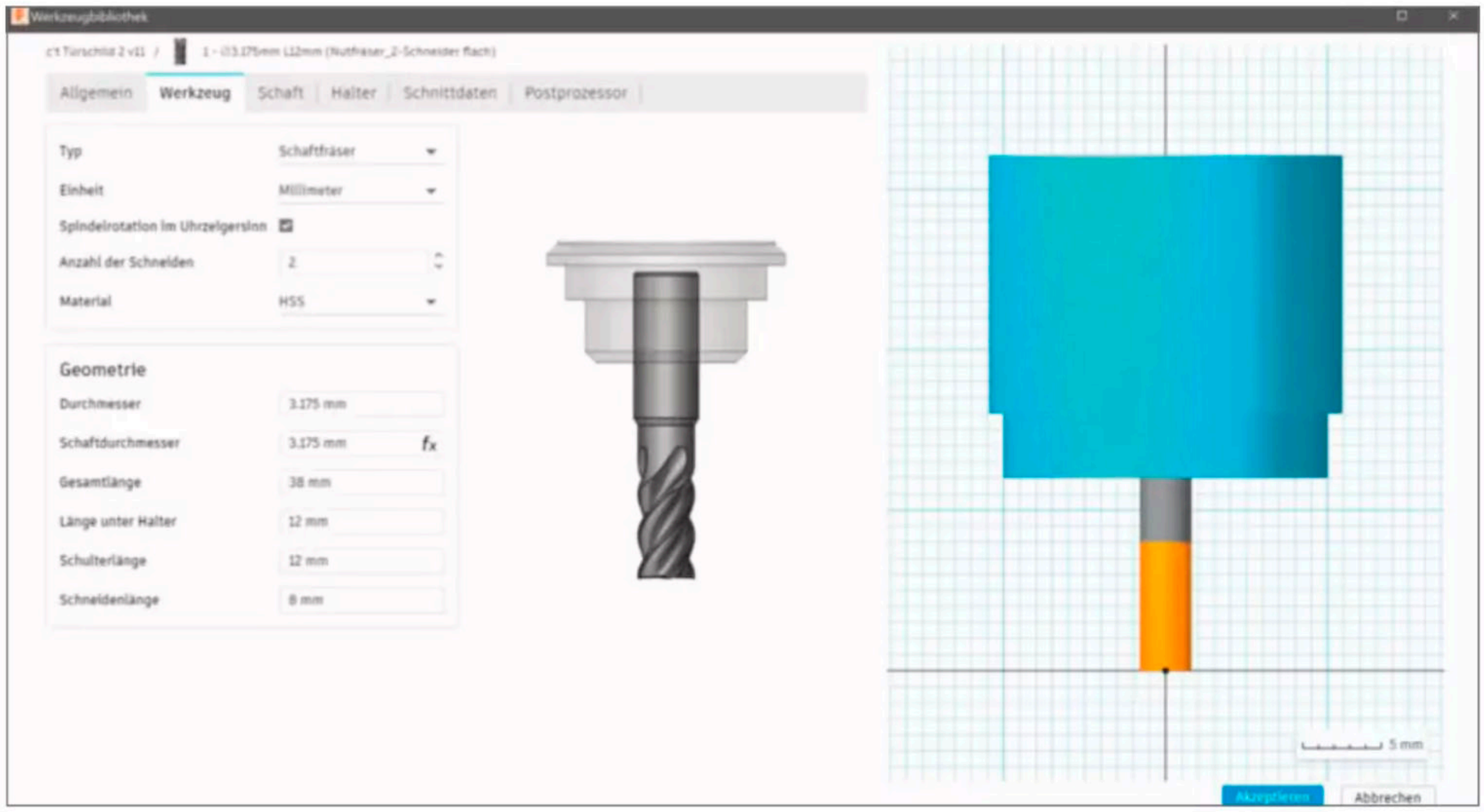
Im nächsten Schritt erzeugen Sie mit dem Werkzeug „Extrusion“ aus der zweidimensionalen Zeichnung ein dreidimensionales Bauteil. Damit sie sehen, wie das Teil eine Dicke bekommt, drehen Sie die Ansicht mit der mittleren Maustaste, während Sie die Umschalt-Taste gedrückt halten. Wählen Sie fürs Extrudieren entweder das Werkzeug in der Werkzeugleiste aus oder verwenden gleich den Tastatur-Shortcut E. Nun selektieren Sie die beiden Flächen der Skizze, die innere Fläche und die zwischen den beiden Konturen. Daraufhin erscheint ein Eingabefeld, in das Sie die Dicke des Türschilds eintippen. Wenn das fürs Fräsen vorgesehene Brettchen zehn Millimeter dick ist, tippen Sie „-10mm“. Achten Sie darauf, dass die Extrusion nach unten ausgeführt wird, also die Skizze danach auf und nicht unter dem Werkstück liegt. Hat Fusion bei Ihnen in die falsche Richtung verdickt, lassen Sie das Minus vor der Extrusionsdicke weg.

Nach dem Extrudieren blendet Fusion die Skizze automatisch aus. Um sie wieder anzuzeigen, öffnen Sie im „Browser“ (das ist die Baumstruktur am linken Rand) den Ast „Skizzen“ und klicken auf das ausgegraute und durchgestrichene Auge neben Ihrer „Skizze1“.

Sobald die Skizze wieder sichtbar ist, nutzen Sie das Werkzeug „Extrusion“ ein zweites Mal und erzeugen die Vertiefung in der Mitte. Wählen Sie dafür nur die innere Fläche aus und ziehen Sie sie drei Millimeter nach unten in das zuvor erstellte Teil hinein. Das schneidet die Vertiefung aus.

Als Nächstes verwenden Sie unter „Ändern“ das Werkzeug „Fase“ und wählen alle Außenkanten aus, um ihnen eine zwei Millimeter breite, im 45°-Winkel abgeflachte Kante hinzuzufügen. Damit ist die Konstruktion des Modells schon abgeschlossen.

Falls auch Ihre digitale Variante des Türschildes eine Holzoptik bekommen soll, finden Sie unter




**Um die richtigen Fräspfade zu erstellen, muss die Software das Werkzeug – den Fräser – genau kennen. In der „Werkzeugbibliothek“ tragen Sie dafür einen Schaftfräser und einen Gravierfräser ein. Die blauen Zylinder markieren die Spannmutter der Spindel, was in der späteren Simulation hilft, Kollisionen mit der Aufspannung zu vermeiden.**

Ändern/Material eine Materialbibliothek, aus der Sie per Drag & Drop jedes Material auf das Werkstück anwenden können.

## Maschinist ans Werk!

Mit dem CAD-Modell ist die Arbeit des Konstrukteurs getan und der Maschinenbediener übernimmt. Für die Arbeit mit Fusion heißt das, dass Sie jetzt mit dem Knopf in der linken oberen Ecke zum Arbeitsbereich „Fertigung“ wechseln. Mental vollziehen Sie den Wechsel, indem Sie sich fortan vorstellen, welche Werkzeuge auf welchen Bahnen das Türschild optimal bearbeiten.

Ein traditioneller Handwerker würde sich nun die Werkzeuge bereitlegen, als Nutzer von CAM-Software geben Sie nun die Fräser in die „Werkzeugbibliothek“ ein. Sie finden diese in der Werkzeugleiste unter „Verwalten“, was ein neues Fenster öffnet. Die Ordnerstruktur links können Sie unterhalb von „Lokal“ um Unterordner erweitern, was Ordnung schafft, falls Ihre Fräserschublade prall gefüllt ist. Was Sie in die Werkzeugbibliothek eingeben, können Sie übrigens in jedem Ihrer Projekte verwenden – es lohnt sich also, hier übersichtlich zu strukturieren.

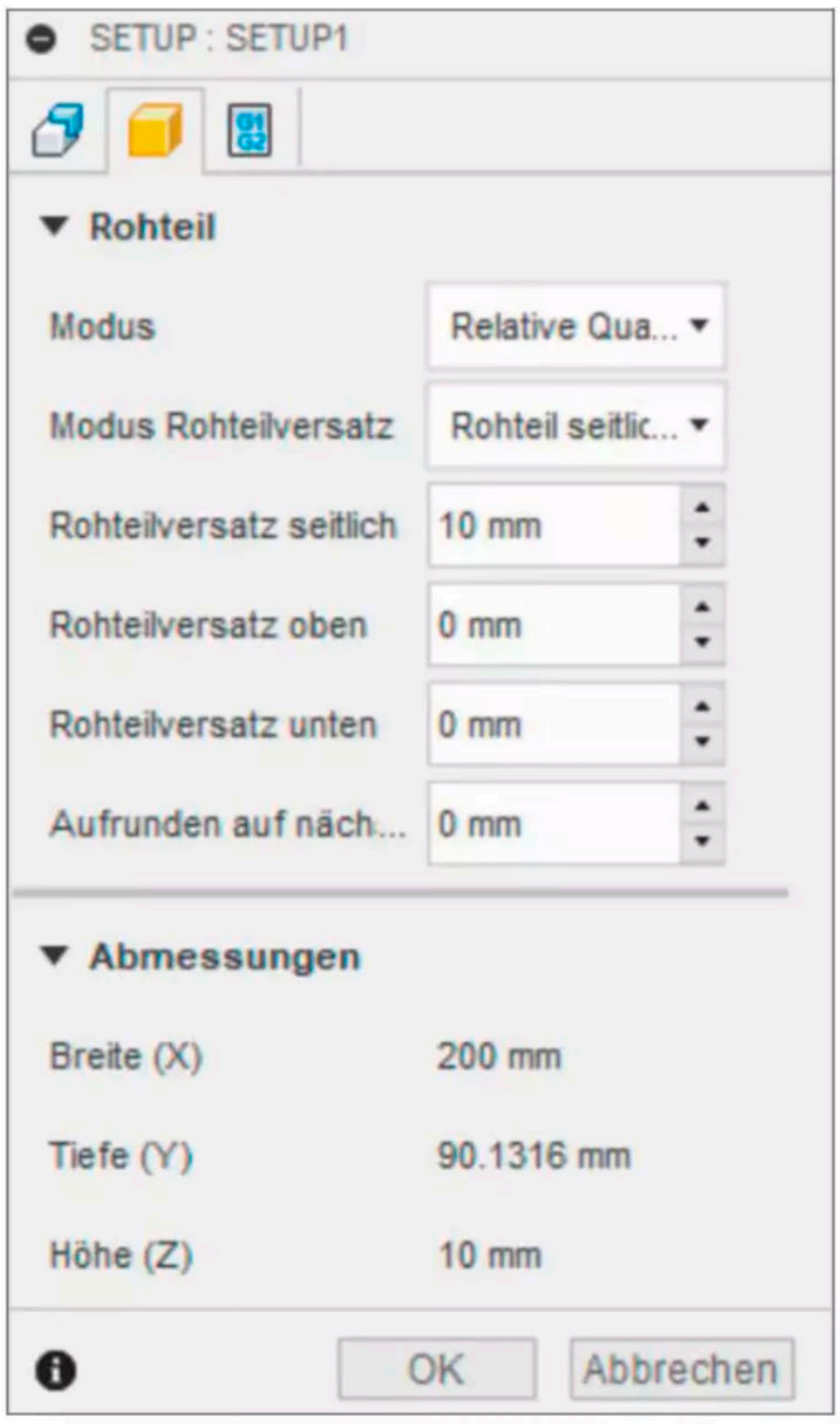
Für die in Fusion integrierte Kollisionserkennung legen Sie zunächst einen Werkzeughalter an. Der Werkzeughalter entspricht bei einfachen Frässpindeln ohne automatischem Werkzeugwechsler der Spannzange und Überwurfmutter. Drücken Sie dafür auf das + in der linken oberen Ecke und wählen dann „Halter“ ganz unten in der Liste. Das Eingabefeld hat zwei Reiter, einen für Namen und Bezugsquellen und einen für die Form. Die legen Sie als Liste von zylindrischen Abschnitten fest, das + erzeugt je einen neuen Abschnitt. Ein Makita-Fräsmotor hat beispielsweise einen zwölf Millimeter hohen Schaft mit 17 Millimetern Durchmesser (Eintrag 1), an den die Überwurfmutter mit 12,1 Millimetern Höhe und 25,3 Millimetern Durchmesser anschließt (Eintrag 2). Um Unfälle zu vermeiden, reicht so ein grobes Modell locker aus.

Als Nächstes fügen Sie den ersten Fräser in der Bibliothek ein. Klicken Sie erneut auf das Plus-Zeichen und wählen Sie unter den Fräswerkzeugen in der ersten Zeile einen Schaftfräser. Es erscheint ein

neues Fenster, in dem Sie das Werkzeug in mehreren Reitern beschreiben. Im ersten Reiter geht es wie beim Halter auch um Informationen für den Nutzer. Im zweiten Reiter beschreibt man die geometrischen Eigenschaften des Fräsers. Diese Daten finden Sie beim Hersteller des Fräsers, falls sie in Ihrem Fall fehlen, messen Sie selbst mit einem Messschieber.

Im fünften Reiter stellen Sie die „Schnittdaten“ ein. Welche Spindeldrehzahl und welcher Vorschub zum Fräser passt, können Sie mit den Formeln aus dem Artikel auf Seite 114 berechnen. Die Hersteller veröffentlichen auch oft Richtwerte als Tabellen zum Download von ihrer Webseite. Für einen Schaftfräser mit 1/8", also 3,175 Millimeter Durchmesser und zwei Schneiden eignet sich beispielsweise eine Spindeldrehzahl von 25.000 U/min, ein Schnittvorschub von 800 mm/min, Einfahrvorschub von 600 mm/min und Eintauchvorschub von 200 mm/min. Diese Einstellungen sind Voreinstellungen, die Sie später beim Festlegen der Fräsbahnen noch überschreiben können. Nach einem Klick auf „Akzeptieren“ ist der Fräser angelegt.

Für die Gravur des Namens und die Fasen legen Sie zusätzlich einen „Fasenfräser“ mit einem Winkel von 45° und einem Spitzendurchmesser von 0 an. Für das Türschild brauchen Sie nur diese beiden Werkzeuge.




**Da Sie definiert haben, wo und wie groß das Rohmaterial ist, kann Fusion die Fräse oberhalb davon mit hoher Geschwindigkeit bewegen.**

## Aufspannungsplanung

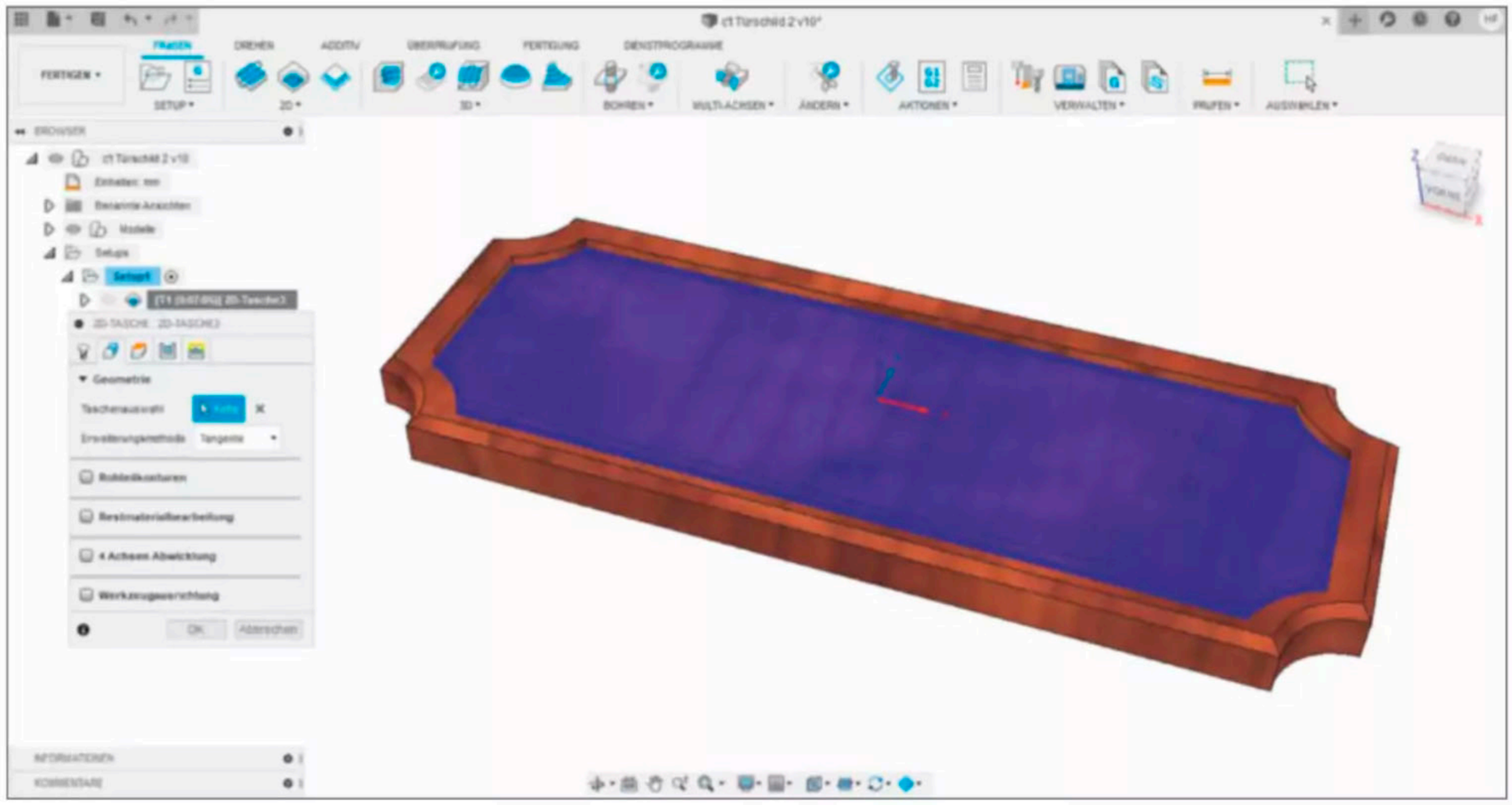
Fusion teilt die Bearbeitung in „Setups“ auf. Ein Setup entspricht all den Arbeitsschritten, die die Fräse ausführen kann, ohne dass Sie das Werkstück umspannen müssen. Mehr als ein Setup brauchen Sie also nur, wenn Sie beispielsweise ein Werkstück von zwei Seiten bearbeiten müssen. Für das Türschild reicht wie für die meisten Projekte ein Setup. Erstellen Sie das Setup mit dem Ordnersymbol links unter „Fräsen“ in der Werkzeugleiste. Fusion öffnet nun ein kleines Fenster mit drei Reitern.

Unter „Bearbeiten“ können Sie nun eine Maschine anlegen. Mit den dort eingetragenen Infos prüft Fusion, ob die Bearbeitung mit Ihrer Fräse überhaupt möglich ist. Sie können den Schritt auch überspringen, indem Sie gar keine Maschine auswählen.

Stellen Sie als Operationstyp „Fräsen“ ein und konfigurieren Sie die Modellausrichtung und den Ursprung. Bei der Modellausrichtung müssen Sie jetzt entscheiden, wie Sie das Holz auf die Fräse aufspannen möchten. Beim „Ursprung“ nutzen Sie den Ursprung, den Sie auf Ihrer Maschine anfahren möchten, beispielsweise die linke untere Ecke auf der Oberfläche des Werkstücks oder der Mittelpunkt der oberen Fläche. Diesen Punkt nutzt Fusion als Referenzpunkt für alle Koordinaten in der G-Code-Datei.

Der zweite Reiter definiert das Rohteil. Bei komplex geformtem Rohmaterial können Sie das im Vorhinein im Konstruktionsbereich als eigenen Körper konstruieren. Im Beispiel fertigt die Fräse das Türschild aber aus Plattenmaterial, weshalb es reicht, einen einfachen Quader anzunehmen. Stellen Sie den seitlichen Rohteilversatz mit 10 mm ein. Es stört nicht, wenn Ihr Material tatsächlich größer ist. Alle anderen Versätze sollten 0 sein.

Im letzten Reiter kann man noch den Namen angeben, unter dem später der G-Code des Setups gespeichert wird. Klicken Sie auf „OK“, um das Setup fertigzustellen.




**Die Geometrie des Bauteils entsteht schrittweise aus den Konturen der Zeichnung. Eine flache Vertiefung nennt sich „2D-Tasche“. Die Operation verlangt nach einer Geometrie, die die Form festlegt. Wählen Sie dafür mit der Maus den Boden der Vertiefung aus.**

## Fräspfade festlegen

Nun geht es endlich ans Festlegen der Fräspfade. Die erste Operation stellt die Vertiefung her, in der später der Name steht. Wählen Sie dafür in der Werkzeugleiste im Menü „2D“ die „2D-Tasche“ aus. Es öffnet sich mal wieder ein Fenster mit mehreren Reitern. Wählen Sie zuerst den Schaftfräser aus Ihrer Materialbibliothek aus. Die Operation übernimmt die in der Bibliothek definierten Schnittdaten, die Sie hier noch mal anpassen können. Im zweiten Reiter definieren Sie die zu bearbeitende Geometrie. Hinter „Taschenauswahl“ sollte „Nichts“ blau hinterlegt sein. Wenn Sie nun die Fläche der Vertiefung auswählen, ändert sich die Angabe im blauen Auswahlfeld und die Fläche ist im Modell blau hervorgehoben. Im dritten Reiter stellen Sie ein, wie viel Sicherheitsabstand die Fräse einhält, bevor sie schnelle Bewegungen macht. Unterhalb der „Vorschubhöhe“ bremst Fusion sicherheitshalber ab. Die „obere Höhe“ zeigt an, wo der Fräser wirklich zu arbeiten beginnt, und die „Endhöhe“ ist die Tiefe der Tasche. Für das Türschild reicht eine Vorschubhöhe von drei Millimetern über der Rohteiloberfläche, Rückzugshöhe zwei Millimeter darüber und Sicherheitshöhe noch mal zwei Millimeter höher.

Im Reiter „Durchgänge“ stellen Sie die „seitliche Kompensation“ auf „Links (Gleichlauf)“, um im Uhrzeigersinn zu fräsen, was auf CNC-Maschinen schönere Oberflächen erlaubt, weil der Fräser steiler ins Material eingreift. „Schlichtdurchgänge“ lassen Sie deaktiviert; sie sind für Metall sinnvoll, kosten aber viel Zeit. Stellen Sie die „Maximale Querzustellung“ auf den halben Fräserdurchmesser, damit die Schneiden des Fräsers mit 90° ins Holz eintauchen. Sollte Ihre Fräse nicht stabil genug dafür sein, können Sie den Wert auch reduzieren. Ähnliches gilt für die „Tiefenschnitte“: Die „Maximale Schrupp-Tiefenzustellung“ entscheidet, wie viel Holz der Fräser in einem Durchgang wegfräst. Wenn Sie hier den halben Fräserdurchmesser einstellen, sollten die Späne noch gut abgeführt werden. Reduzieren Sie die Tiefenzustellung für weniger steife Maschinen, allerdings nicht unter 0,2 Millimeter, weil der Fräser dann mehr reibt als schneidet. Zu viel Reibung führt nicht nur zu Brandmarken im Holz, sondern glüht auch die Schneiden des Fräsers aus, wodurch er weich und danach schnell stumpf wird, was zu noch mehr Reibung führt – ein Teufelskreis. Die Einstellung „Gleichmäßige Tiefenzustellungen verwenden“ hält die Belastung für die Fräse konstant. Das „Rohteil Aufmaß“ deaktivieren Sie, es dient dazu, Material stehen zu lassen, das ein anderer Fräser beim Schlichten in einem zweiten Durchgang abtragen könnte. Bei der „Vorschuboptimierung“ sollten Sie mit den Voreinstellungen starten und die Werte mit wachsender

Erfahrung an Ihre Fräse anpassen. Die Funktion dient dazu, Schrittverluste und Schwingungen durch mangelnde Rahmensteifigkeit zu minimieren. „Nur Innenecken“ aktiviert das Feature nur für Ecken, an denen der Fräser zu viel entfernen könnte, wer seine Maschine schonen möchte, lässt das Häkchen weg.

Der letzte Reiter beschreibt die Bewegungen außerhalb des Materials und die Ein- und Ausfahrten in und aus dem Material. Der erste und zweite Abschnitt „Anfahr- und Wegfahrbewegungen“ und „Ein-/Ausfahrten und Übergänge“ kann für das Beispiel genau so bleiben, wie es Fusion vorschlägt. Bei der dritten Kategorie „Materialeinfahrt“ können Sie den „Rampen/Helixwinkel“ etwas erhöhen, zum Beispiel auf 15°, die voreingestellten 2° sind für Holz unnötig flach. Hier können Sie viel Zeit sparen, da die Maschine beim Eintauchen besonders langsam fährt. Damit haben Sie Einstellungen für den ersten Fräspfad fertig und können mit „OK“ die Pfaderzeugung abschließen.

Fusion zeigt nun mit blauen Linien den Fräspfad für die Vertiefung an. In einem Bild ist der Verlauf der Bewegungen aber nicht erkennbar. Wo entlang der Fräser fahren wird, zeigt „Simulieren“ aus dem „Aktionen“-Menü. Gerade für Anfänger ist es sinnvoll, mit der Simulation mehrfach zu prüfen, ob alle Bewegungen unfallfrei und wie erwartet ablaufen.

## Kontur

Nun geht es zur zweiten Bearbeitung, dem Fräsen der Kontur („2D-Kontur“ im Menü „2D“). Das Werkzeug öffnet das bekannte Fenster mit den fünf Reitern. Nur wenige Dinge sind anders als bei den Einstellungen der Tasche: Im zweiten Reiter gibt es den Punkt „Haltestege“, den Sie unbedingt aktivieren müssen. Die Haltestege sind kleine Verbindungen zwischen dem umliegenden Rohteil und dem Werkstück. Ohne sie wäre das Werkstück am Ende der Kontur lose und könnte vom Fräser durch die Werkstatt geschossen werden. Das kann Fräser abbrechen, das Werkstück ruinieren und Sie im ungünstigsten Fall verletzen. Haltestege lassen etwas Material stehen, das Sie später per Hand beispielsweise mit einem Stemmeisen oder Cuttermesser durchtrennen.

Sie müssen neben der Form und der Größe auch die Position der Haltestege definieren. Die von Fusion vorgeschlagene Größe für die Haltestege ist für Holz meist zu klein und sollte in beiden Richtungen einige Millimeter betragen. Für die Position lassen Sie Fusion entweder Vorschläge errechnen und stellen nur den Abstand zwischen zwei Haltestegen ein oder Sie wählen bei „Haltesteg-Positionierung“ „An Punkten“. Dann können Sie mit der Maus mehrere Punkte an beliebigen Stellen entlang der Kontur auswählen. Platzieren Sie immer mindestens drei Haltestege.

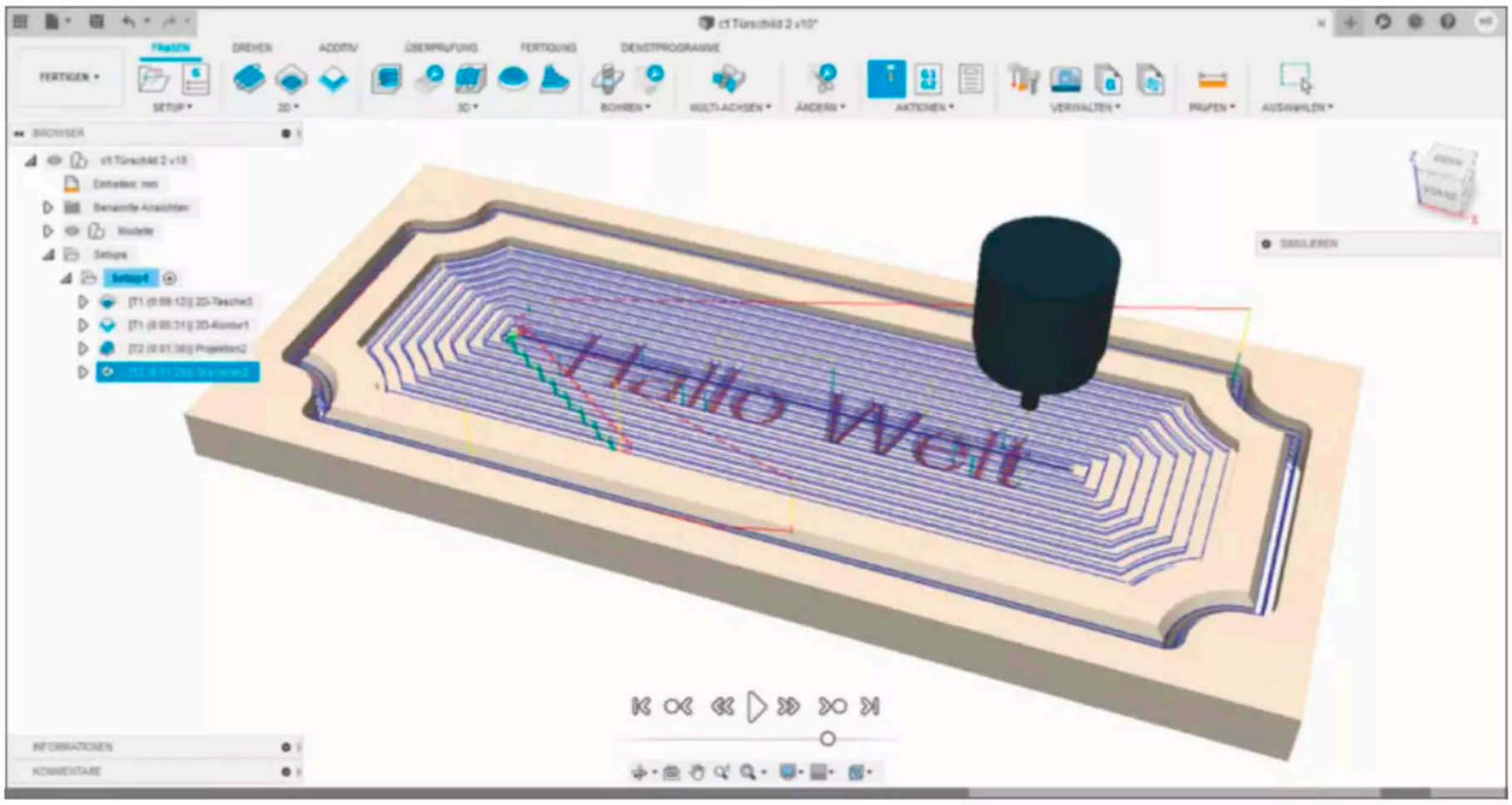
Alle anderen Einstellungen sind grundsätzlich die gleichen wie bei der Tasche. Lediglich bei der „Ausfahrt“ im letzten Reiter sollten Sie den „horizontalen Ausfahrradius“ und den „linearen Ausfahrradius“ auf 0 setzen. Am Ende der Kontur würde die Fräse ansonsten zunächst horizontal vom Werkstück um den Ausfahrradius wegfahren und dann aus dem Material austauchen. Dabei würde der Fräser über die volle Höhe ins Material fahren und höchstwahrscheinlich brechen.

## Werkzeugwechsel

Damit sind alle Bearbeitungsschritte mit dem Nutfräser geschafft, jetzt fehlen noch die Fase und die Gravur des Namens. Beides fräst der 45°-Gravurfräser. Die Kräfte, die der Gravurfräser ausübt, sind gering genug, sodass die Stege die Belastung halten können. Wäre das nicht der Fall, müssten Sie zuerst die Tasche, dann nach einem Werkzeugwechsel die Gravur und nach einem erneuten Werkzeugwechsel die Kontur fräsen.

Um eine Fase zu fräsen, haben Sie mehrere Möglichkeiten. Es gibt eine Frässtrategie „2D Fase“, falls Sie die Fase im Modell nicht vorher konstruiert haben. Wenn die Fase aber wie im Beispiel schon konstruiert ist, eignet sich besser die Strategie „Projektion“, die Sie ebenfalls bei den 2D-Strategien finden. Nach der Auswahl der Strategie erscheint wieder das bekannte Einstellungsfenster. Als Werkzeug wählen Sie nun aber den erstellten Fase- beziehungsweise Gravierfräser aus. Bei der Geometrie wählen Sie die untere Kante der Fase aus. Die Kette von Konturlinien wird von Fusion umlaufend automatisch erkannt. Im vierten Reiter finden Sie einen Unterpunkt „Fase“, der genau für diesen Fall gedacht ist: Hier kann man die genauen Maße der Fase noch mal nachjustieren. Beispielsweise können Sie durch eine „Fasenbreite“ einen zusätzlichen Versatz einstellen. Durch einen „Fasenspitzenversatz“ führt Fusion den Fräser etwas neben der eigentlichen Kante, sodass er nicht mit der Spitze schneidet, was ein etwas besseres Schnittbild erzeugt. An der Spitze ergeben sich nämlich zwangsläufig zu kleine Schnittgeschwindigkeiten.

Zuletzt fräsen Sie nun die Gravur für den Namen. Deren Form bestimmt der Text in der Skizze, die Sie




**Die Simulation zeigt nicht nur alle Wege auf einen Blick, sondern animiert den vereinfacht dargestellten Fräser auch auf seinem Pfad.**

dafür mit dem Auge-Symbol im „Browser“ (Baumstruktur auf der linken Seite) sichtbar schalten. Wählen Sie dann die Frässtrategie „Gravieren“ (seltsamerweise ebenfalls im Menü „2D“). Als Fräser wählen Sie die Fasen-/Gravierfräser vom letzten Schritt und bei der Kontur klicken Sie auf den Text aus der Zeichnung. Bei den Höhen müssen Sie nun die „Obere Höhe“ auf die Grundfläche der Tasche einstellen. Mit „Auswahl“ können Sie die zunächst nicht anklicken, weil die Skizze im Weg ist. Sobald Sie die Skizze mit dem Auge im „Browser“ ausgeblendet haben, können Sie die Fläche auswählen.

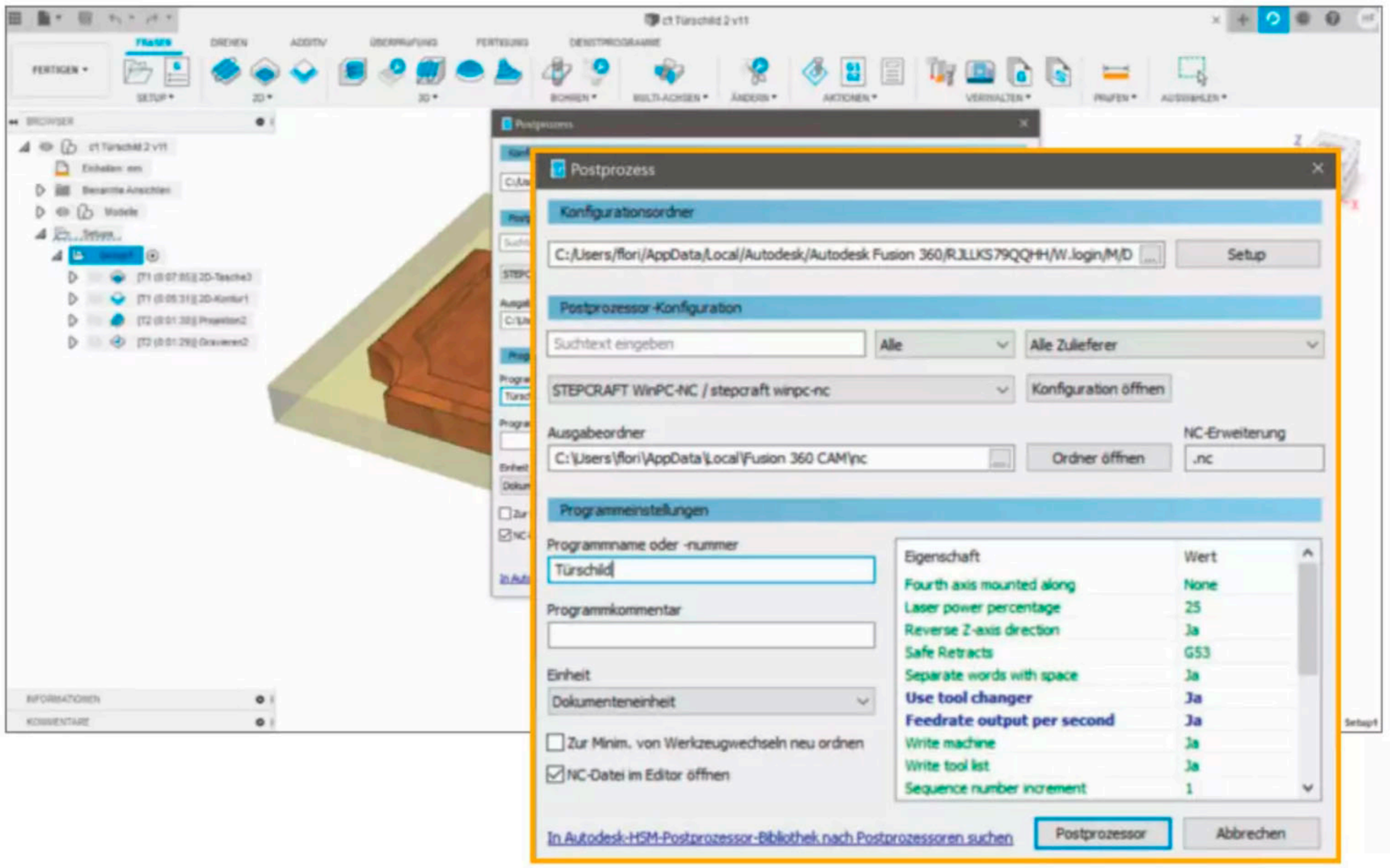
Im vierten Reiter gibt es noch eine neue Einstellung: „Scharfer Eckwinkel“. Beim Abfahren der Kontur mit dem Gravierfräser erhält man zunächst runde Ecken. Durch einen geschickten zusätzlichen Fräsweg, bei dem der Fräser weiter in die Ecke fährt und gleichzeitig nach oben, entsteht eine scharfe Ecke. Der scharfe Eckwinkel gibt den Grenzwert an, ab dem dieser zusätzliche Weg angewendet wird, um eine scharfe Ecke zu erzeugen. Der voreingestellte Wert ist in den meisten Fällen gut geeignet. Nach dem Bestätigen der Einstellungen auf „OK“ wird der Fräspfad berechnet und schließlich auch angezeigt. Falls Sie das Fräsergebnis und den Pfad nicht sehen können, ist der „Körper“ im Weg, den Sie mit dem Auge im Browser verbergen können.

## G-Code exportieren

Vor dem Export sollten Sie noch mal einige Einstellungen prüfen: Im Browser vor den einzelnen Frässchritten wird die Werkzeugnummer angezeigt, zum Beispiel „T1“ oder „T2“. Sie sollten darauf achten, dass die beiden Werkzeuge unterschiedliche Nummern haben. Diese Nummern werden später von der Steuerungssoftware übernommen. Wenn sie gleich sind, wird dazwischen kein Werkzeugwechsel ausgeführt. In der Werkzeugbibliothek können Sie den Werkzeugen entweder globale Nummern zuweisen oder lokale, projektspezifische Nummern. Dazu müssen Sie nur rechts auf den Fräser klicken und dann auf „Werkzeug neu nummerieren“. In die erste Zeile können Sie dann die neue Nummer eintragen. Das gesamte Setup oder auch einzelne Schritte lassen sich jetzt mit der Funktion „Simulieren“ auf Fehler überprüfen. Wenn alles stimmt, sind nun alle Fräspfade erfolgreich erstellt.

Bevor Sie nun den G-Code zum Steuern der Fräse exportieren können, brauchen Sie noch den passen-

**Der Postprozessor passt den G-Code an den „Dialekt“ der eigenen Fräse an. Ein häufiger Fehler sind Vorschubgeschwindigkeiten in mm/s statt mm/min. Ob die richtig ankommen, überprüfen Sie am besten in der exportierten Datei anhand der Zahlen hinter F. GRBL arbeitet beispielsweise in mm/min, weshalb dort dreistellige Zahlen stehen sollten.**




den Postprozessor. Ein Postprozessor passt Koordinaten und die Auswahl der G-Codes daran an, was die Fräsensteuerung verarbeiten kann. Autodesk bietet für die meisten CNC-Fräsmaschinen einen passenden Postprozessor kostenlos zum Download (siehe ct.de/wzss). Falls Sie dort nicht fündig werden, können Sie den Postprozessor auch manuell erstellen. Ihren Postprozessor können Sie dann in die Postprozessorbibliothek importieren. Diese finden Sie in der Werkzeugleiste unter „Verwalten“. Im Fenster der Postprozessorbibliothek gibt es oben einen Button zum Importieren.

Fusion 360 exportiert in der kostenlosen Version keine Fräspfade mit Werkzeugwechseln. Sie können für das Türschild aber einfach zwei getrennte G-Code-Dateien exportieren – die Koordinaten passen perfekt zueinander. Selektieren Sie dafür im Browserbaum die ersten beiden Fräsbahnen mit dem Schaftfräser als Werkzeug. Klicken Sie dann in der Werkzeugleiste bei „Aktionen“ neben „Simulieren“ auf „Postprozessor“. Es öffnet sich ein Fenster mit den Einstellungen für den Postprozess. Dort müssen Sie jetzt den eben installieren Postprozessor auswählen, den Ausgabeordner und rechts unten ein paar Einstellungen überprüfen. Abhängig von Ihrer Maschine und Steuerungssoftware müssen Sie außerdem ein paar Werte ändern. Wenn Ihre Steuerungssoftware in mm/s arbeitet, können Sie beispielsweise „Feedrate output per second“ auf „Ja“ stellen. Zum Bestätigen der Einstellungen klicken Sie auf „Postprozessor“ rechts unten. Sobald die Datei gespeichert ist, selektieren Sie im Browserbaum die restlichen Fräsbahnen mit dem Gravierfräser als Werkzeug und exportieren diese genauso.

Der G-Code ist nun fertig erstellt und kann von der Steuerungssoftware Ihrer Fräse eingelesen werden. Zum Fräsen setzen Sie den X-Y-Nullpunkt wie geplant in die Ecke oder die Mitte und den Z-Nullpunkt auf die Oberfläche des Rohteils. Fixieren Sie das Werkstück lieber zu fest als zu locker und achten Sie darauf, dass der Fräser nicht mit Schrauben oder Spannpratzen kollidieren kann. Der Bahn für den Schaftfräser folgt Ihre Fräse dann alleine. Sobald sie damit fertig ist, spannen Sie den Gravierfräser ein, stellen den Z-Nullpunkt passend ein und lassen den Nullpunkt für X und Y unbedingt unangetastet. Danach arbeitet Ihre Fräse die zweite Datei ab. Nach dem Abtrennen der Stege bleibt Ihnen ein Türschild, das Sie ohne CNC-Fräse nicht so schön und exakt hinbekommen hätten. (pmk) c't

**Download-Link und Beispieldatei**

ct.de/wzss



Bild: Albert Hulm



Einstieg in Onshape



Techn. Zeichnung in Onshape

# Onshape: 3D-Designs im Browser

**Das CAD-Programm Onshape läuft ohne Installation in jedem modernen Browser. Tausende Konstruktionen anderer Designer stehen zum Erkunden bereit und erleichtern den Einstieg. Mit wenigen Klicks exportieren Sie sie zum 3D-Druck oder als Zeichnungen für die Bastelwerkstatt.**

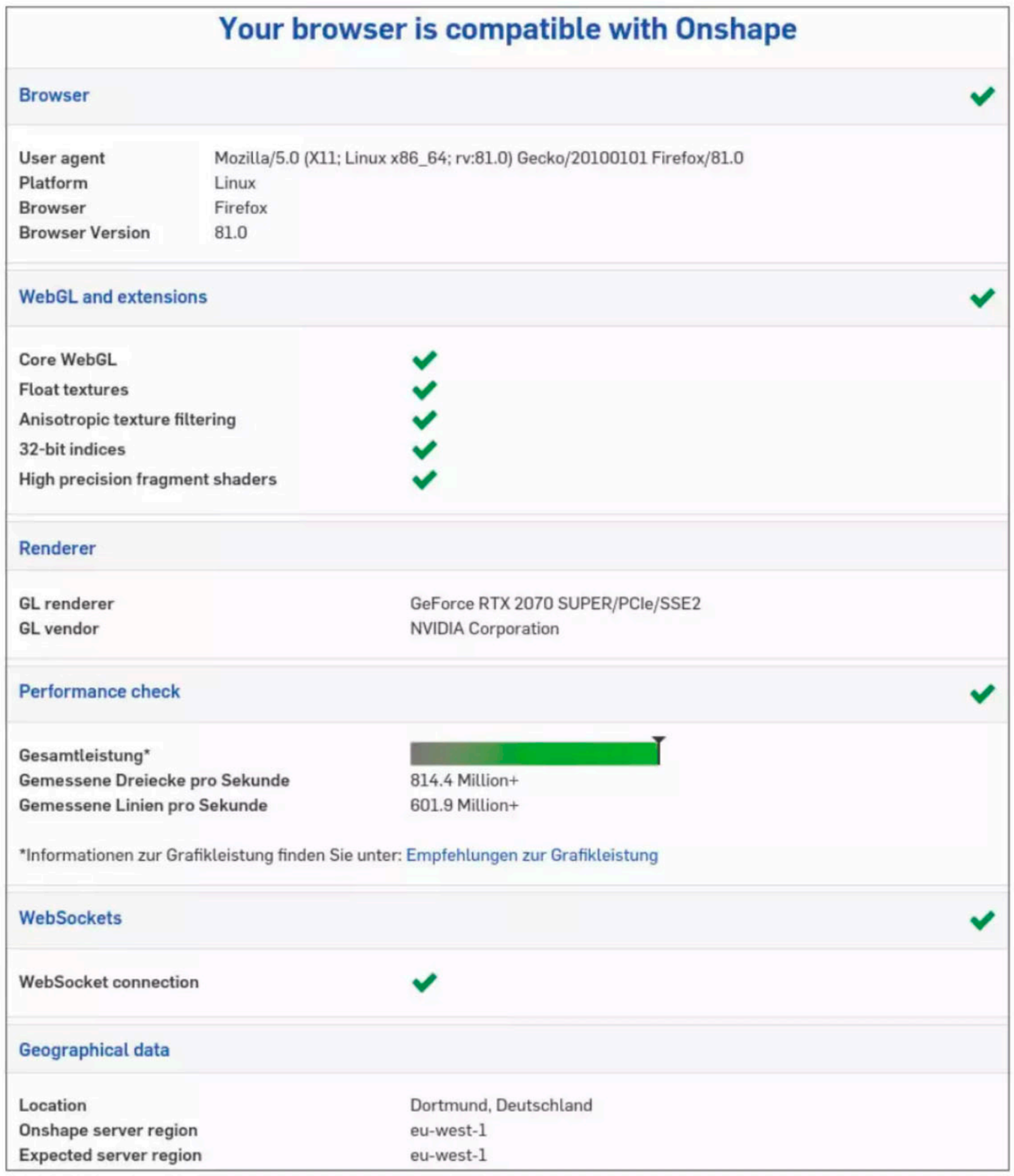
Von **Pina Merkert**

Kennen Sie den FarmBot? Wir auch noch nicht, aber wir lernen ihn gerade kennen (siehe S. 154 bis 176). Der FarmBot ist ein Gartenbau-Roboter, der wie eine große CNC-Fräse aussieht und vollautomatisch ein Hochbeet bewirtschaften soll. Ein amerikanisches Start-up hat die Maschine entwickelt und sämtliche CAD-Daten als Open Hardware im Internet veröffentlicht. Um sie zu sehen, muss man kein teures CAD-Programm installieren. Der Aufruf einer etwas kryptischen URL im Browser reicht (siehe ct.de/wsab).

Es öffnet sich „Onshape“ [1], ein umfangreiches CAD-Programm, das komplett im Browser läuft. Das Geschäftsmodell von Onshape erinnert an das von GitHub: Wer alles veröffentlicht, darf die Software kostenfrei nutzen. Wer seine Designs lieber nicht jedem zeigen möchte, zahlt für den Cloud-Service. So wie die privaten Repositorys bei GitHub viele

öffentliche Repositorys von Open-Source-Projekten mitfinanzieren, finanzieren auch die Firmenkunden von Onshape, die ihre Designs ungern der Konkurrenz offenbaren wollen, die öffentlichen Designs von Open-Hardware-Enthusiasten mit. Auch die Struktur ähnelt sich: „Versionen“ entsprechen Commits und erlauben, frühere Bearbeitungsstände zurückzuholen. Es gibt auch Forks, da man die Designs von anderen Nutzern in den eigenen Workspace kopieren kann, um sie zu verändern oder zu ergänzen. Beispielsweise um eine technische Zeichnung zu erstellen, was wir im zweiten Teil des Artikels tun werden. Onshape erlaubt sogar, mit mehreren Anwendern parallel an der gleichen Konstruktion zu arbeiten. Der Cloud-Zwang der Software offenbart dort seine Vorzüge.

Onshapes Geschäftsmodell ist ideal für Maker: Sie wollen ihre Designs ohnehin mit der Welt teilen und planen nicht, mit ihnen Geld zu verdienen. Dass ein Konkurrent das Design übernimmt und es produziert, ohne Entwicklungskosten zu haben – die Angst der meisten Firmen –, würde ein Maker wohl als Auszeichnung für sein Design empfinden und sich freuen. Die Cloud-Software ist außerdem von vornherein für die Arbeit im Team ausgelegt, was Maker-Spaces und kleinen Projekten zugutekommt. Voraussetzung ist aber, dass die Maker sich in der umfangreichen CAD-Software zurechtfinden und beispielsweise STL-Dateien für den 3D-Drucker exportieren oder technische Zeichnungen anfertigen können. Genau darum geht es in diesem Artikel. Wir zeigen die ersten Schritte mit Onshape und lassen



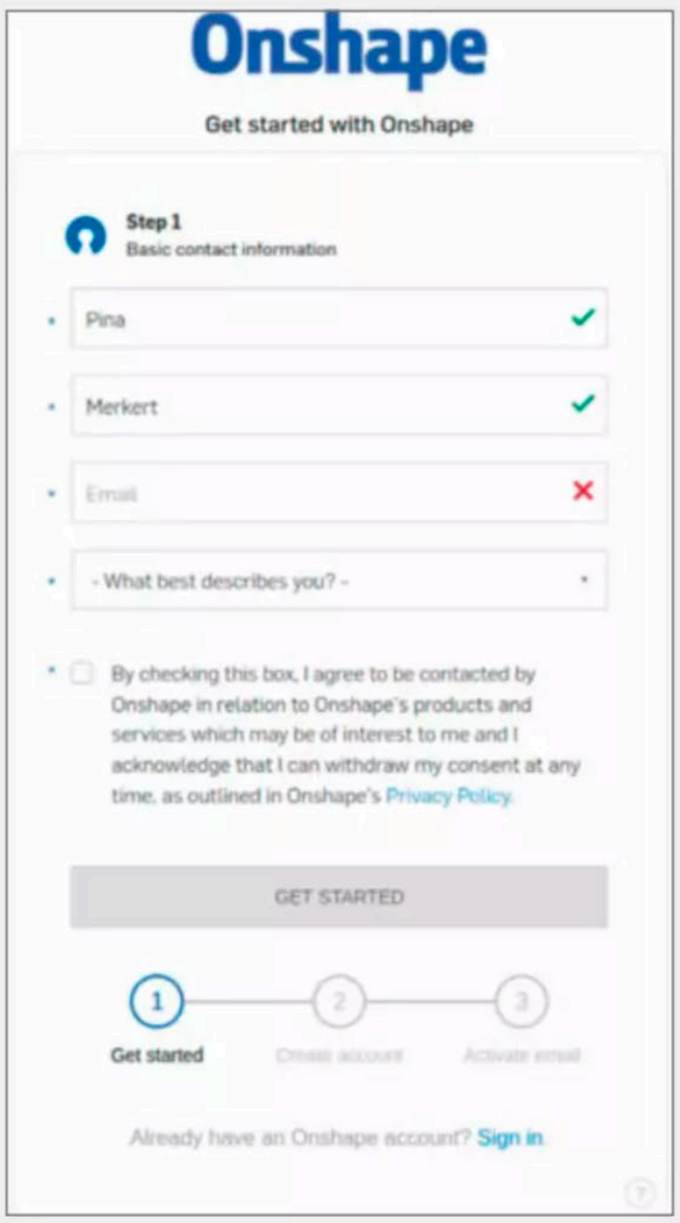

**Onshape testet die Leistung des Browsers nach der Anmeldung. Den Test bestehen Firefox und Chrome problemlos.**

# Bei Onshape registrieren

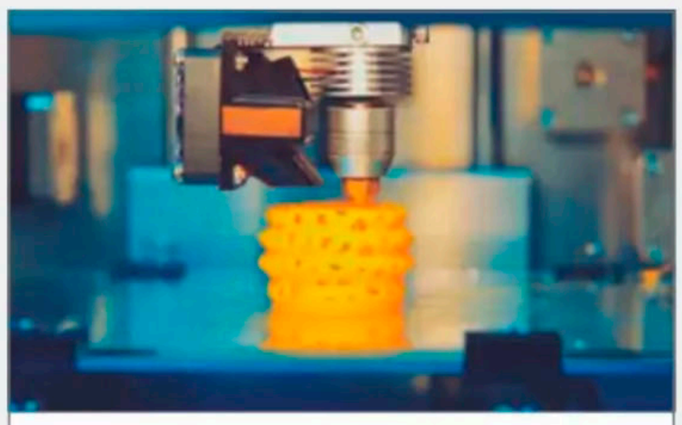
Besucht man die Website onshape.com, sucht man zunächst vergeblich nach der Registrierung für die kostenlose Version. Der passende Button versteckt sich auf der Seite „Produkt“, die Sie über das Hauptmenü am oberen Bildschirmrand ganz links finden. Auf der Produkt-Seite müssen Sie dann erst mal ganz nach unten scrollen, bis Sie in einem blauen Balken einen orangefarbigen Button mit der Aufschrift „Onshape kostenlos testen“ finden. Die direkte URL für diese Seite ist https://cad.onshape.com/signup2.

Im Zuge der Registrierung fragt Onshape neben dem Namen und der Mailadresse noch nach Profession und Firmendaten. Was Sie dort eintragen, spielt keine Rolle und Onshape hat bei unserem Test auch keinen Bezug auf die Angaben genommen. Leider klappt die Registrierung aber nur, wenn Sie zustimmen, sich auch für Werbung kontaktieren zu lassen. Ein Abbestellen der Werbemails ist aber im Nachhinein möglich.

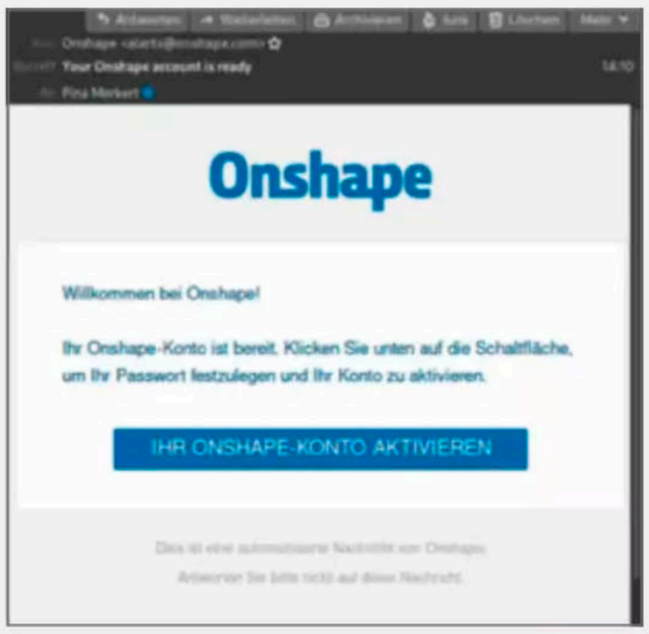
Ihre Mailadresse müssen Sie über eine Bestätigungsmail verifizieren, bevor der Account wirklich freigeschaltet wird. Danach können Sie mit dem kostenlosen Account Kopien öffentlicher Designs anlegen, um Zeichnungen hinzuzufügen. Zum reinen Anschauen der Designs in 3D und zum Exportieren von STL-Dateien brauchen Sie gar keinen Account.




**Die Registrierung fragt außer Name und Mailadresse noch diverse andere Daten ab. In den zusätzlichen Feldern führen aber auch Alibi-Einträge zum Erfolg.**




**Wer Onshape kostenlos ausprobieren möchte, wählt bei der Registrierung „Onshape for Makers“ aus. Onshape versteckt den „Free Plan“ ein wenig auf seiner Website.**




**Wie üblich muss man die Mail bestätigen, bevor es losgeht. Die obligatorische Zustimmung zu Werbemails nervt, man kann die Mails aber im Nachhinein abbestellen.**

die Design-Werkzeuge erst mal bewusst weg. Auf sie gehen wir ab Seite 146 ein. Wer nämlich öffentliche Onshape-Designs wie den FarmBot nachbauen will, muss CAD-Konstruktionen nicht entwerfen können. Nur in ihnen navigieren, Teile exportieren und drucken.

## Navigieren

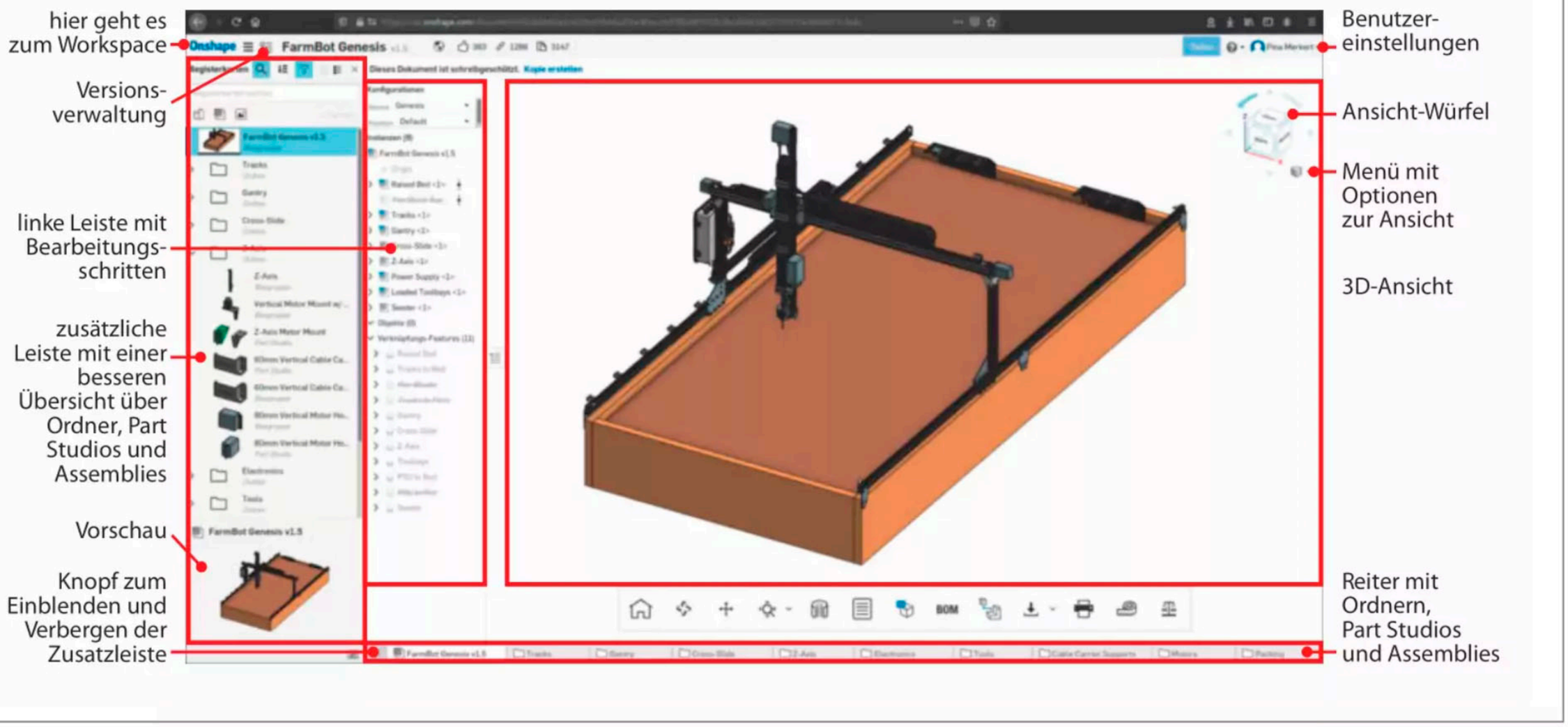
Wenn Sie das CAD-Design des FarmBot Genesis öffnen (den Link finden Sie über ct.de/wsab), begrüßt Sie zunächst eine Ladeanimation. Nach einigen Sekunden baut sich in sichtbaren Schritten die 3D-Ansicht der gesamten Maschine in der Mitte des Bildschirms auf. Ohne Anmeldung sind alle Beschriftungen in englischer Sprache. Wenn Sie sich anmelden und die Sprache im Account auf Deutsch umgestellt haben, zeigt Onshape den FarmBot mit deutschen Beschriftungen.

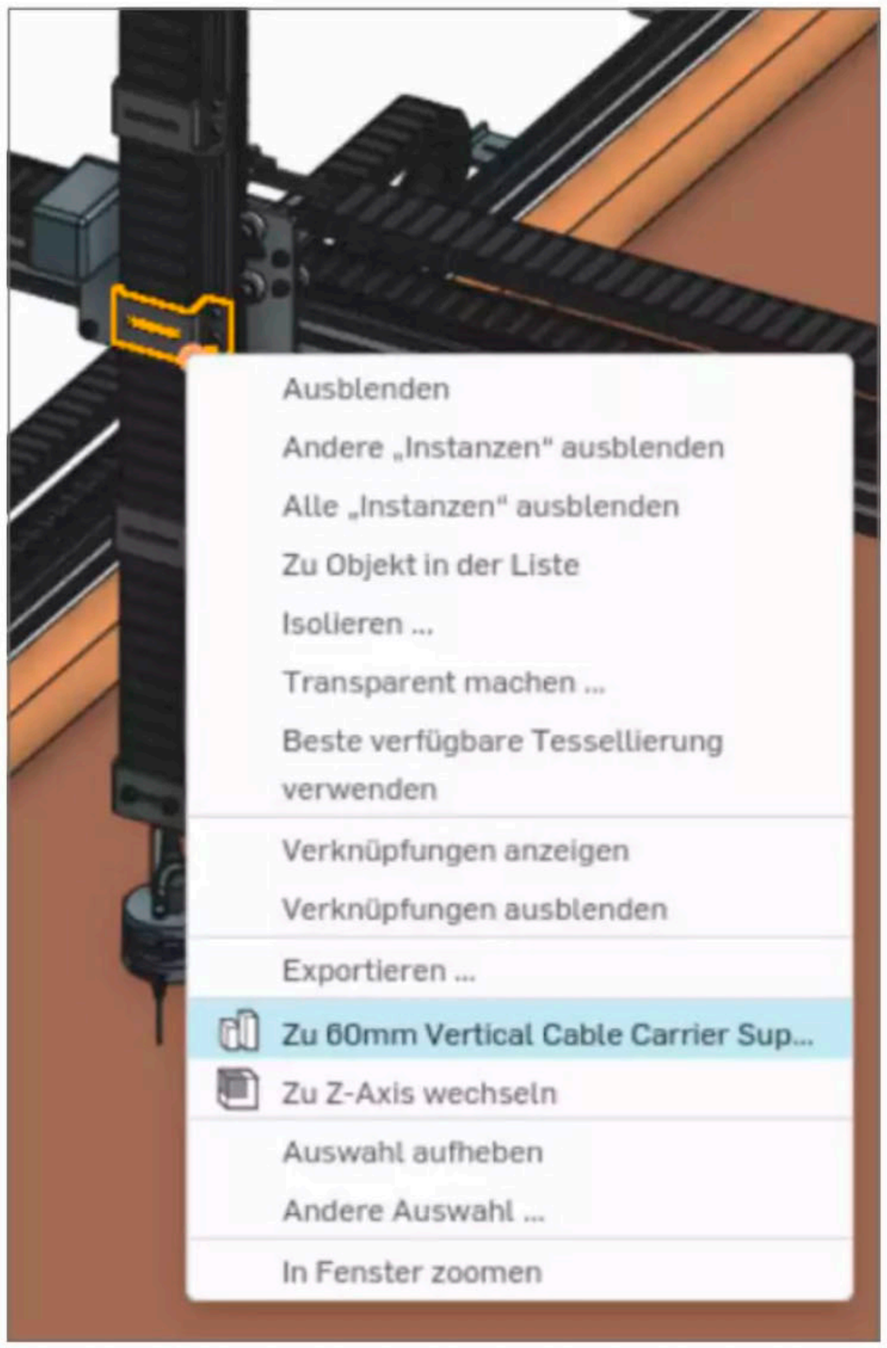
In der rechten oberen Ecke dieser Ansicht informiert Sie ein Würfel mit beschrifteten Seiten („Oben“, „Vorn“ und „Rechts“), dass Sie von schräg oben auf das Modell schauen. Mit dem Würfel kann man schnell die Ansichten per Mausklick umschalten, im Menü rechts darunter (Würfelchen mit Pfeil) findet man Standardansichten, die sehr helfen, wenn man sich mit der Kamera vernavigiert hat. Außerdem kann man dort die Perspektive an- und ausschalten. Klickt man mit der rechten Maustaste in die 3D-Ansicht und hält sie gedrückt, kann man die Ansicht mit Mausbewegungen drehen. Das Mausrad zoomt und wenn man es gedrückt hält, verschiebt eine Bewegung die Ansicht ohne Drehung. Probieren Sie ruhig mit der Maussteuerung herum. Falls es unübersichtlich oder schief wird, klicken Sie beim Würfel einfach auf die Ecke zwischen „Oben“, „Vorn“ und „Rechts“ und gelangen so zurück zur bekannten Ansicht von schräg oben. Falls Sie so weit gedreht haben, dass diese Ecke nicht mehr sichtbar ist, kommen Sie mit dem Menüpunkt „Isometrisch“ aus dem Menü unter dem Würfel zurück zur bekannten Schrägansicht.

Um die 3D-Ansicht ordnet Onshape mehrere Leisten und Listen an. In der Leiste oben führt Sie das

### Die 3D-Ansicht in Onshape

Beim Betrachten fertiger Konstruktionen landet man zunächst in der 3D-Ansicht. Betrachten, Zeichnen und Zusammenbauen findet hier statt. Für technische Zeichnungen schaltet Onshape um, die Seitenleisten bleiben aber bestehen.






**Anders als auf vielen Webseiten öffnet ein Rechtsklick auf ein Bauteil in Onshape ein Kontextmenü, das zum CAD-Programm und nicht zum Browser gehört. Über dieses Menü kann man Teile schnell exportieren oder das zugehörige Part Studio öffnen.**

Onshape-Logo, wenn Sie angemeldet sind, zurück zu Ihrem Workspace. Dort finden sich Ihre eigenen Designs. Ist man nicht angemeldet, führt das Logo nur zur Log-in-Seite. Im Burger-Menü daneben findet sich die Versionsverwaltung. Bei fremden Designs wie dem FarmBot finden Sie dort den Knopf „Arbeitsbereich kopieren“, der einen Fork des Projekts in Ihrem eigenen Workspace anlegt, den Sie dann auch verändern dürfen. Das Menü rechts daneben gehört ebenfalls zur Versionsverwaltung. Dort finden Sie die Versionshistorie als Diagramm. Rechts davon steht der Projektname und daneben einige Statistiken zu Likes und Aufrufen. Am rechten Rand finden Sie Login oder Benutzereinstellungen. Dort tauchen auch Avatar-Symbole anderer Benutzer auf, wenn Sie mit mehreren Personen parallel das gleiche Design editieren.

## Öffentliche Designs finden

Wer sich bei Onshape registriert und angemeldet hat, kann mit dem Suchschlitz im Workspace nicht nur in den eigenen Konstruktionen suchen. Schaltet man nämlich in der linken Seitenleiste auf „Öffentlich“ um, findet die Suchfunktion Tausende frei einsehbare Konstruktionen. Die Suche durchforstet standardmäßig die Namen der Projekte; über den kleinen Pfeil rechts im Suchschlitz lässt sich die Ergebnisliste aber auch mit Filtern eingrenzen.

Die Leiste links wird wichtig, wenn Sie eigene Designs konstruieren. Dann sammeln sich dort die Bearbeitungsschritte, deren Parameter Sie auch im Nachhinein verändern können. Beim FarmBot sehen Sie dort „Instanzen“ und „Verknüpfungs-Features“, da das komplexe Design des CNC-Gärtners in viele Baugruppen aufgeteilt ist. Um das Modell nur zu inspizieren und nicht zu verändern, brauchen sie die linke Leiste nicht. Mit dem Fähnchen mit Pfeil-nach-unten-Symbol können Sie die Leiste einklappen und mehr Platz für die 3D-Ansicht gewinnen.

Am unteren Bildschirmrand sammelt Onshape Reiter für „Part Studios“, „Baugruppen“ und „Ordner“. Ordner funktionieren genauso, wie man das von Dateisystemen kennt. Part Studios sind Dateien mit Designs und Baugruppen sind Dateien mit zusammengebauten Designs, die aus einem Part Studio stammen. Wer also ein Einzelteil sucht, um es als STL-Datei für den 3D-Druck zu exportieren oder um eine technische Zeichnung anzufertigen, sollte nach dem passenden Part Studio suchen. Wer wissen will, wie die Teile zusammengehören, findet diese Information in einer Baugruppe.

Baugruppen müssen nicht nur aus Einzelteilen bestehen, Designer können auch Baugruppen in Baugruppen einfügen. Die Baugruppe „FarmBot Genesis v1.5“ auf der Root-Ebene des FarmBot-Designs enthält beispielsweise die Baugruppen der Schienen, des Portals und der Z-Achse des Roboters. Im FarmBot-Design sind alle Baugruppen und Part Studios auf-

schlussreich benannt. Wer selbst entwirft, sollte auf eine erklärende Benennung achten.

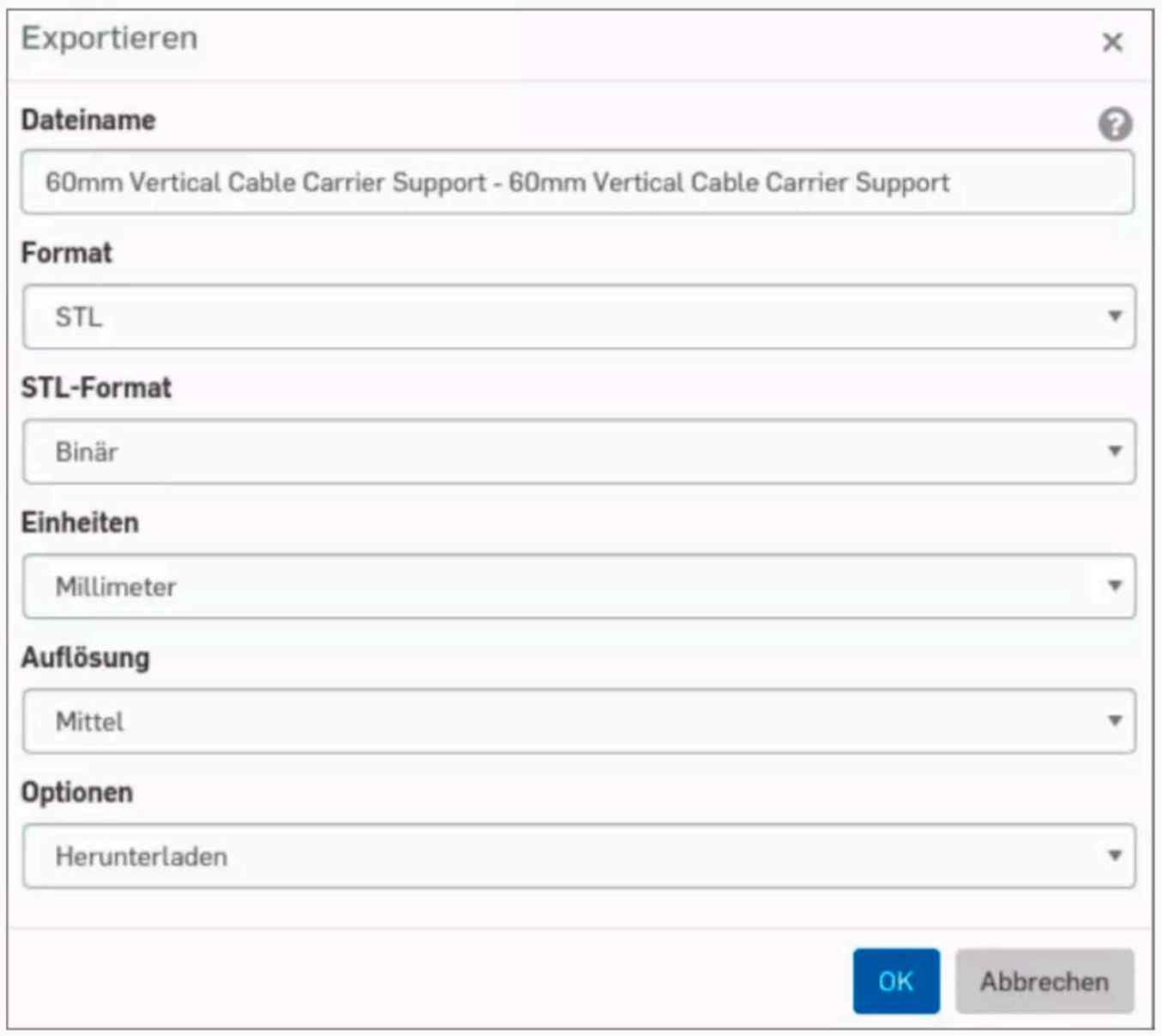
Die untere Leiste reicht im Prinzip, um alle Einzelteile zu finden. Man kann aber mit der Lupe ganz links eine weitere linke Leiste ausklappen, die die Ordnerstruktur noch übersichtlicher darstellt. Ordner klappen dort als eingerückte Bereiche aus und es gibt hilfreiche Vorschaubilder. Ein erneuter Klick auf die Lupe klappt die Leiste wieder ein.

## Teile isolieren per Kontextmenü

Statt über die Reiter unten oder die Zusatzleiste links händisch nach in Unterordnern versteckten Part Studios zu fahnden, kann man die Einzelteile auch in der 3D-Ansicht suchen. Dafür dreht und zoomt man die Ansicht so lange, bis man mit der Maus über dem gesuchten Teil schweben kann. Das hebt Onshape dann mit einem gelben Rahmen hervor. Onshape überschreibt die Kontextmenüs der Browser, sodass ein Rechtsklick ein Menü zum Bauteil öffnet. Dort kann man es ausblenden oder in einem wählbaren Dateiformat exportieren. Zusätzlich kann man aber auch direkt zu dem Part Studio wechseln, in dem das Teil entworfen wurde. So gelangt man von einer Baugruppe (Assembly) mit nur zwei Klicks zu einem Einzelteil.

## Exportieren

Egal ob per Kontextmenü vom Assembly oder vom Part Studio aus, „Exportieren“ öffnet ein Dialogfenster. Dort kann man den Dateinamen anpassen und das Dateiformat wählen. Wer Teile in Blender rendern möchte, exportiert im Collada-Format, für 3D-Drucke bietet sich STL an. Für STL akzeptiert Onshape noch weitere Parameter: Im Binärformat wird die exportierte Datei kleiner. Die Koordinaten erwarten Slicer von 3D-Druckern üblicherweise in Millimetern statt Metern. Da STL nur Polygone speichert, muss Onshape Rundungen zu Vielecken umrechnen. Die Auflösung bestimmt, wie fein die Unterteilung wird. „Mittel“ übersteigt meist schon die vom Drucker umsetzbare Genauigkeit. Nach einem Druck auf „OK“ startet der Browser den Download der exportierten Datei.



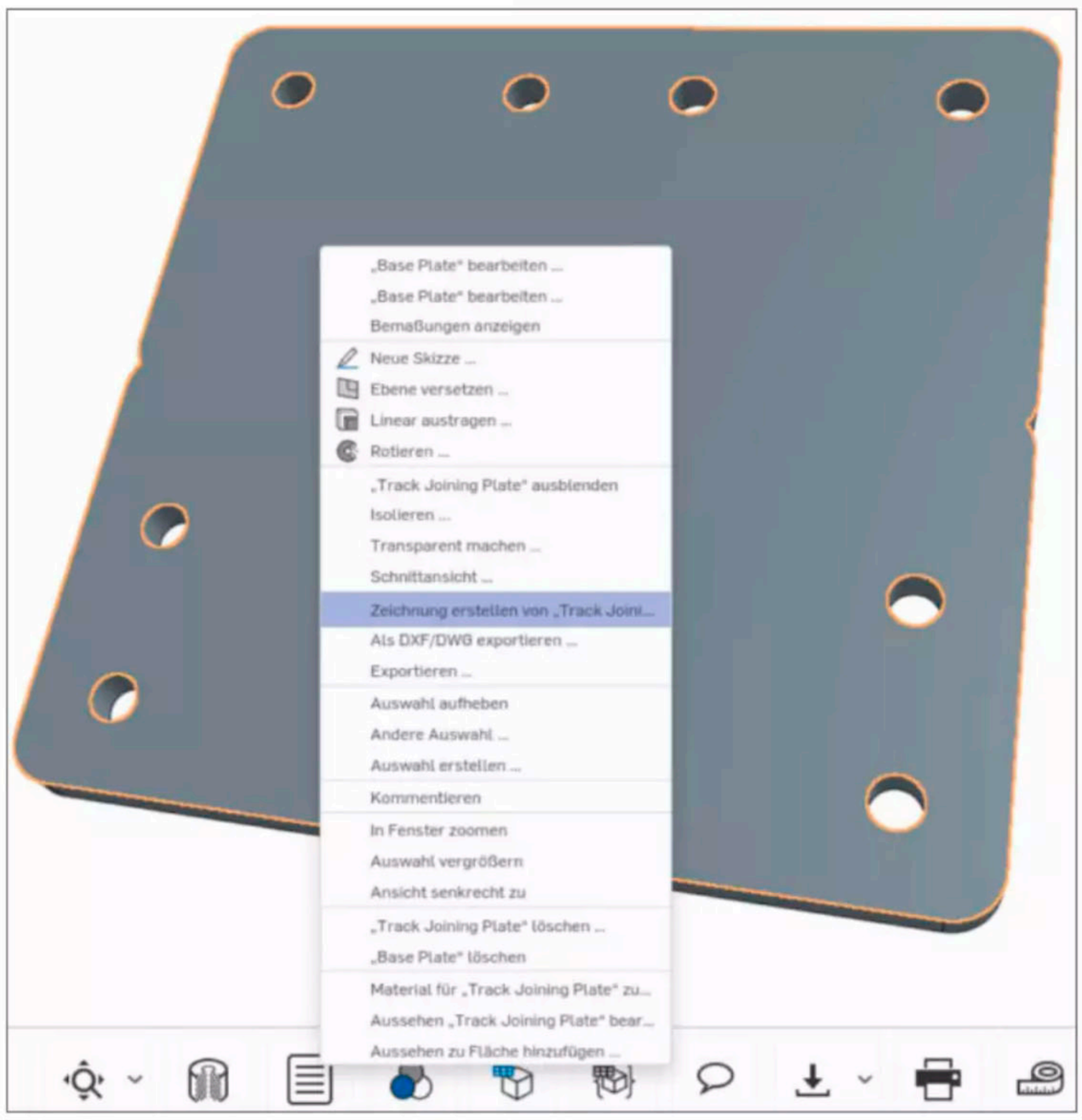

**Exportiert man ein Teil für den 3D-Druck (STL), muss man an den Exporteinstellungen nur die Einheiten auf Millimeter ändern. OK startet den Download.**

## Technische Zeichnungen

Der FarmBot nutzt Aluplatten als Knotenbleche, um die Aluprofile an ein Hochbeet zu schrauben. Die von so einem Teil erwartete Stabilität kann ein 3D-Druck nicht bieten. Also muss man per Hand anreißen, bohren und feilen. Mit einer guten technischen Zeichnung produziert man so eine Platte in weniger als einer halben Stunde. Vorher müssen sich aber alle Maße für den Höhenreißer auf der Zeichnung materialisieren. Eine Automatik gibt es für technische Zeichnungen nicht, Onshape stellt aber effiziente Werkzeuge bereit, mit denen eine Zeichnung in weniger als fünf Minuten aus dem 2D-Drucker fällt.

Ein Klick mit der rechten Maustaste öffnet ein Menü, in dem man mit „Zeichnung erstellen von“ zu einem Auswahldialog fürs Papierformat gelangt. Die Platte von unserem Beispiel passt auf eine A4-Seite nach ISO-Norm, wer einen größeren Drucker hat, kann hier auch A3 auswählen. Für einen schnellen Start bringt Onshape zwei Templates mit: eine leere Seite und eine mit Standardansichten von drei Seiten. Letzteres spart Zeit, man kann die Ansichten aber auch später noch hinzufügen. Da fürs Beispiel die Seitenansichten der Platte wenig nutzen, haben wir mit einer leeren Seite gestartet.

Nach einem Klick auf „OK“ legt Onshape die Zeichnung im gleichen Ordner wie das Part Studio des Bauteils an. Der Browser zeigt nun eine leere Normseite für technische Zeichnungen mit Feldern für Datum, Name, Maßstab et cetera in der rechten unteren Ecke. Da eine Zeichnung ganz ohne Ansicht des Bauteils sinnlos wäre, aktiviert Onshape automatisch das Ansicht-Einfügen-Werkzeug. Das Symbol dafür ist in der oberen Werkzeugleiste ganz links. Dort öffnet Onshape auch einen kleinen Dialog, in dem man per Dropdown-Menü auswählt, welche




**Eine technische Zeichnung erstellt man leicht vom Kontextmenü eines Bauteils aus. Das Dokument muss dafür editierbar sein, weil die Zeichnung das Projekt ergänzt. Von öffentlichen Designs kann man dafür einfach eine Kopie erzeugen.**



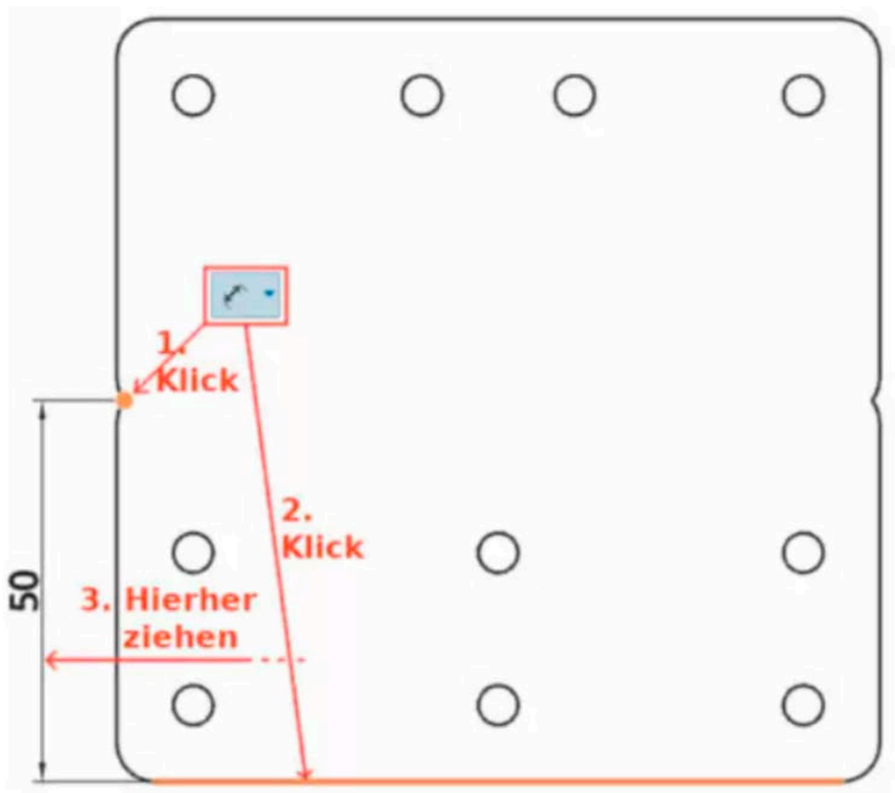

**Im Dialog für neue Zeichnungen wählt man standardkonforme Templates für das bevorzugte Papierformat und – falls gewünscht – direkt Ansichten von drei Seiten.**

Ansicht man einfügen möchte. Für unsere Platte stellen wir dort von „Vorne“ auf „Oben“ um, sodass statt der Schmalseite nun die Seite mit den Bohrungen unter dem Mauszeiger erscheint. Im gleichen Dialog legt man auch den Maßstab der Ansicht fest. 1:2 fanden wir unnötig klein und stellten deswegen auf 1:1 um. Die so vergrößerte Ansicht der Platte schwebt nun unter dem Mauszeiger und man kann sie an einer beliebigen Stelle im Blatt positionieren. Mit einem Klick bestätigt man die Position und die Linien der Ansicht werden schwarz. Unter dem Mauszeiger erscheint nun eine weitere Ansicht, je nachdem, an welche Stelle man ihn bewegt. Diese Funktion dient dazu, weitere Ansichten einzufügen, die ihre Position relativ zur ersten Ansicht behalten. Mit einer Bewegung nach rechts erscheint beispielsweise eine Seitenansicht, deren Höhe und vertikale Position exakt zur ersten Ansicht passt. Ein weiterer Klick fügt diese Ansicht ebenfalls ein, ein Druck auf ESC verwirft die zweite Ansicht.

Die Ansicht im Dokument ist nun zwar maßstäblich, zum Nachbau fehlen aber die Maße. Die fügt das Bemaßungswerkzeug ein. Üblicherweise nutzt man es mit drei Mausklicks: Der erste Klick wählt das Objekt (Punkt oder Kante), von dem gemessen wird, der zweite Klick wählt eine weitere Kante oder einen weiteren Punkt, zu dem gemessen wird. Unter dem Mauszeiger erscheinen nun das Maß als Zahl und zusätzlich Pfeile, die zeigen, welchen Abstand die Zahl bezeichnet. Je nachdem, wie man die Maus bewegt, wechselt das angezeigte Maß zwischen vertikalen, horizontalen oder schrägen Abständen. Der dritte Klick bestätigt die Position und zeichnet das Maß ein. Das Werkzeug steht dann sofort wieder für die nächste Bemaßung bereit. Wer das gerade platzierte Maß noch mal verschieben möchte, deaktiviert zuerst das Bemaßungswerkzeug mit Esc und klick-verschiebt das Maß danach.

Das Bemaßungswerkzeug kennt auch einen Zwei-Klick-Modus: Wählt man beispielsweise eine Kante, erscheint unter dem Mauszeiger sofort ihre Länge. Platziert man dieses Maß, statt eine weitere Kante oder einen Punkt auszuwählen, hat man einen Klick gespart. Wählt man statt einer Kante einen Kreis, entsteht auf die gleiche Art ein Maß für den Durchmesser oder den Radius. Wer lieber einen Radius statt des Durchmessers (oder umgekehrt) einzeichnen möchte, platziert zunächst den Durchmesser, schließt dann mit ESC das Bemaßungswerkzeug und öffnet zuletzt mit einem Doppelklick auf die Zahl einen Dialog für das Maß. In der unteren rechten Ecke findet sich dort ein Knopf, um zwischen Durchmesser und Radius hin- und herzuschalten.

Für Winkel braucht man wie beim Abstand zwischen zwei parallelen Kanten drei Klicks. Man wählt

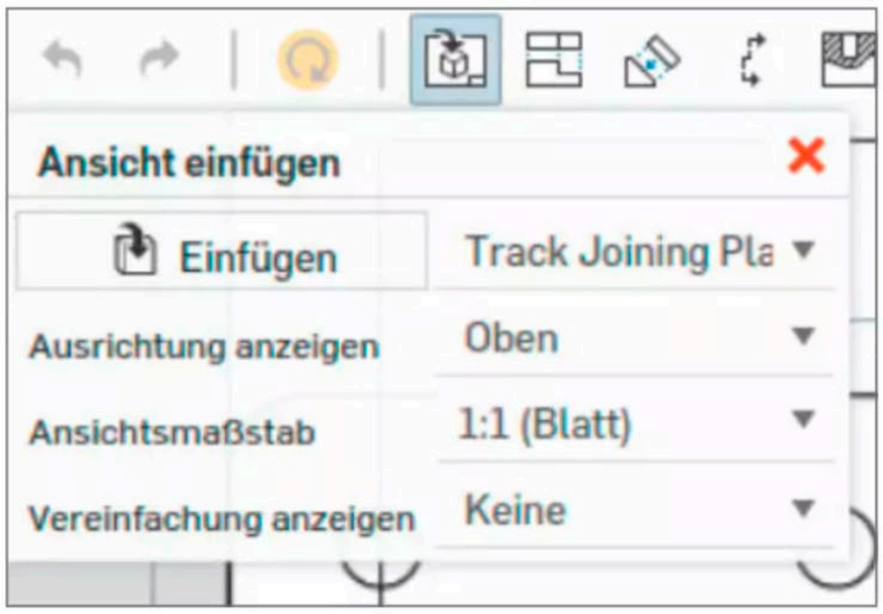



**Das Bemaßungswerkzeug funktioniert mit drei Klicks: Zuerst wählt man einen Punkt oder eine Kante, danach den zweiten Punkt oder die Kante, zu der das Werkzeug den Abstand einzeichnen soll. Zuletzt positioniert man den Pfeil und die Beschriftung der Bemaßung.**

zunächst die beiden sich kreuzenden Kanten und platziert dann das Winkelmaß. Onshape verwendet in Zeichnungen ausschließlich Winkel in Grad.

Für einen professionellen Touch kann man noch die Felder in der Normtabelle unten rechts ausfüllen. Ein Doppelklick öffnet jeweils einen Dialog und ein Eingabefeld. Datum und Skalierung füllt Onshape automatisch aus. Den Titel muss man aber beispielsweise selbst eintragen. Die drei grauen Striche sind ein Zeichen, das man einfach löschen kann.

Der Knopf zum Drucken befindet sich im Burger-Menü rechts neben dem Onshape-Logo in der obersten Leiste. Er erzeugt ein PDF auf dem Onshape-Server, das moderne Browser direkt anzeigen. Über die üblichen Knöpfe des Browsers kann man die Datei lokal speichern oder direkt an den Drucker schicken.




**Im Dialog fürs Einfügen einer Zeichnung wählt man die Ansichtsseite und den Maßstab.**

## Probieren erlaubt

Mit den hier beschriebenen Schritten finden Sie sich in bestehenden Onshape-Konstruktionen zurecht und können Teile für den 3D-Druck exportieren. Von den Werkzeugen zum Erstellen von Zeichnungen haben wir nur die wichtigsten vorgestellt. Mit denen können Sie in Kopien von Designs wie dem FarmBot aber im Handumdrehen Zeichnungen erstellen, mit denen die Produktion von Teilen per Hand leicht fällt. Eine gut bemaßte Zeichnung vermeidet auch Fehler am Bohrständer oder mit der Bandsäge. Zeichnungen im Maßstab 1:1 können Sie sogar aufs Rohmaterial kleben und als Schablone verwenden.

Die meisten Funktionen von Onshape haben wir hier mit keinem Wort erwähnt. Sie kennenzulernen erfordert aber nicht unbedingt eine umfangreiche Anleitung. Das Programm setzt sein intuitives Bedienkonzept konsequent um und funktioniert komplett mit Mausbedienung. Shortcuts müssen Sie nur lernen, um mit der Zeit schneller zu werden. Man kann mit Onshape sogar von einem Tablet aus konstruieren.

Wenn Sie noch mehr sehen wollen als die Screenshots in diesem Artikel, können Sie zwei YouTube-Videos aufrufen, die wir ergänzend zum Artikel produziert haben. Eines erklärt die Oberfläche und die Navigation, das zweite zeigt das Erstellen einer Zeichnung. Sie finden die Videos über ct.de/wsab. Probieren Sie Onshape danach am besten selbst ein bisschen aus. Um beispielsweise den FarmBot zu erkunden, brauchen Sie ja nicht mal einen Account. (pmk) c't




**Die Leerstellen in der Normtabelle füllt man mit einem Doppelklick auf die Kästchen.**

### Literatur


[1] Ralf Steck, **Solider Kern**, Neues Cloud-CAD-System Onshape, c't 09/2015, S. 150


**Erklärvideos bei YouTube, Direktlinks zu den Designs**

ct.de/wsab

secIT by Heise
SPECIAL

Die Online-Konferenzen zur secIT by Heise

# secIT Special
## IT-Security im Fokus

Vertiefen Sie in einer spannenden Keynote sowie in 3 interaktiven Vorträgen Ihr Wissen rund um IT-Security. Jedes secIT Special präsentiert Ihnen ein exklusives Thema.

### DIE TERMINE

**IR, SOC und SIEM**

Verteidigungschinesisch (nicht nur) für Anfänger!

*Als Aufzeichnung erhältlich*

16. September 2021

**Cybersicherheit für KMU**

Kaufen, mieten oder selber machen?

9. Dezember 2021

**Hacking for Security**

Was Sicherheitstests für Unternehmen bringen!

**JETZT ANMELDEN**
https://konferenzen.heise.de/secit-special

Jetzt Kombi-Rabatt sichern
und **über 30% sparen!**



Bild: Albert Hulm

Onshape für Konstrukteure

# Ersatzteile konstruieren mit Onshape

**Mit dem Browser-CAD-Konstruktionsprogramm Onshape gelingen maßhaltige 3D-Designs ohne Installation. Unser Einstieg ins Computer Aided Design erklärt alle Schritte zum Konstruieren eines Ersatzteils für eine Rennbahn aus den 90ern.**

Von **Pina Merkert**

Wenn ein Fünfjähriger auf Schatzsuche gehen will, kann man schwer nein sagen. Nun wühlt der kleine Mann auf dem Speicher durch Kisten mit Spielzeug, das die alte Tante dorthin verbannt hat. Plötzlich zieht er wie eine Plastik-Klapperschlange die Fahrbahn einer Strax-Bahn aus einem Karton. Eine Strax-Bahn ist eine Art Murmelbahn für Autos. Die flexible Fahrbahn steckt man auf Stützpfeiler und baut damit eine Mini-Achterbahn, der batteriebetriebene Autos folgen, weil

eine Bordsteinkante sie auf Spur hält. Der Neffe findet Fahrbahn und Pfeiler; von den Verbindungsstücken, die die Bahn auf dem Pfeiler halten, ist aber nur noch eines da. Kaum vom Speicher gerettet, baut der kleine Achterbahn-Ingenieur sofort einen aufgeständerten Rundkurs ins Wohnzimmer. Doch leider rutscht die Fahrbahn ohne die Verbinder viel zu schnell von den Pfeilern.

Die fehlenden Verbinder gibt es leider nicht als Ersatzteil, da der Hersteller die Bahn nicht mehr verkauft. Aber die Tante hat einen 3D-Drucker und einer der Verbinder ist als Vorbild noch übrig. Wenn der Neffe das nächste Mal zu Besuch kommt, könnten die Verbinder konstruiert sein und fertig gedruckt auf ihn warten. Er akzeptiert den Vorschlag nach kurzem Zögern gönnerhaft und beschäftigt sich stattdessen mit einem Malbuch.

## Konstruieren in Onshape

Wer wie die Tante aus unserem Beispiel eher zu den Gelegenheitsnutzern eines CAD-Programms zählt, freut sich über kostenlose Nutzungsmöglichkeiten ohne langwierige Installation. Hier bietet sich Onshape an: Es läuft direkt im Browserfenster und ist auch als Gratisvariante für nicht kommerzielle Nutzung verfügbar. Voraussetzung ist lediglich ein Account bei dem Cloud-Dienst.

Wir haben ab Seite 136 bereits das Interface des Programms beschrieben und den Export von Teilen für den 3D-Druck erklärt. In zwei ergänzenden Videos erklärten wir die Programmoberfläche und wie man aus fertigen Formen technische Zeichnungen erstellt (frei verfügbar, siehe ct.de/wkwk). Fürs Konstruieren sind wir bisher aber eine Erklärung schuldig geblieben und genau dafür gibt es diesen Artikel. Was wir hier erklären, ist übrigens zu großen Teilen auf andere CAD-Programme wie Fusion360 (siehe S. 126) oder FreeCAD [1] übertragbar.

Zum Konstruieren brauchen Sie einen Onshape-Account. Besuchen Sie dafür einfach die Webseite onshape.com und klicken Sie im oberen Menü auf „Produkt“. Auf der Produktseite scrollen Sie ganz nach unten, bis Sie in einem blauen Balken einen orangen Button mit der Aufschrift „Onshape kostenlos testen“ finden. Die URL dieser Seite lautet https://cad.onshape.com/signup2.

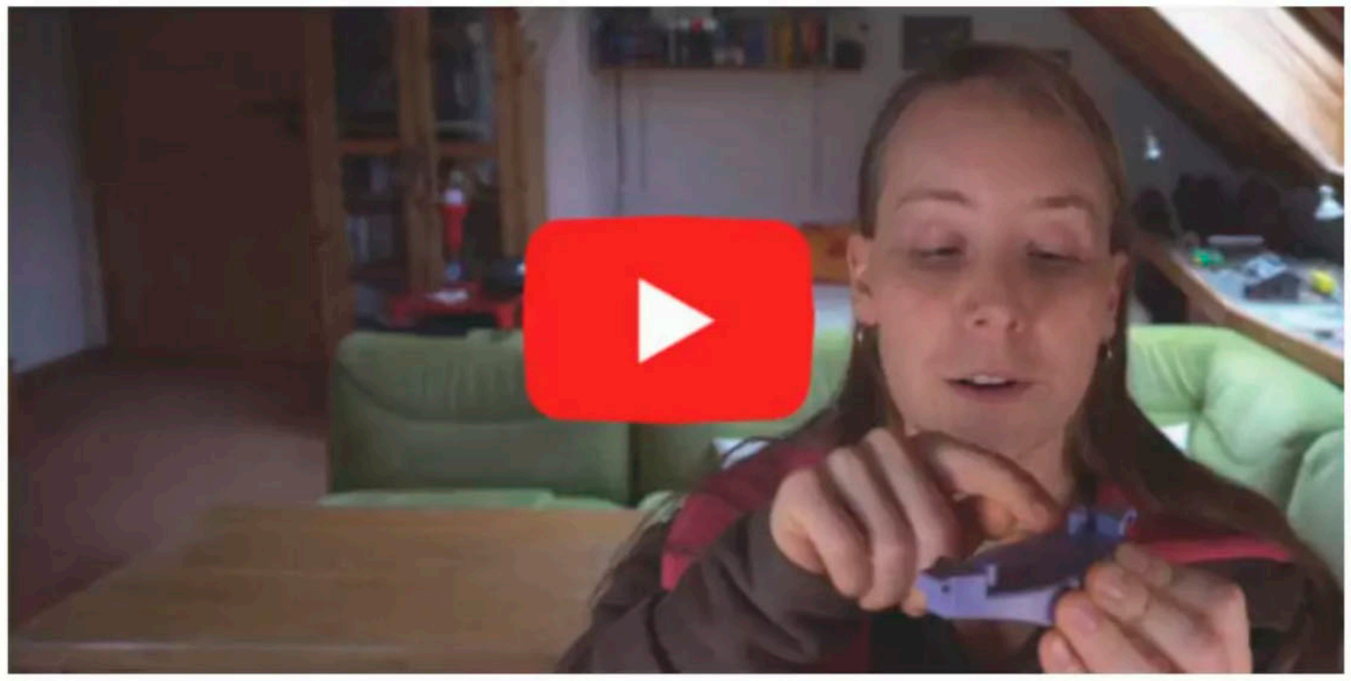
Nach der Anmeldung landen Sie in Ihrem „Workspace“, in dem Sie mit dem blauen Button oben links ein neues Dokument anlegen. Nachdem Sie einen Namen wie „Straxteil“ für die neue Konstruktion vergeben haben, wechselt Onshape in eine leere 3D-Ansicht, in der sich drei Zeichenebenen im 90°-Winkel schneiden. Selektieren Sie die „Top“-Ebene. Selektierte Objekte hebt Onshape mit einer senfgelben Umrandung hervor.

## Eine erste Skizze

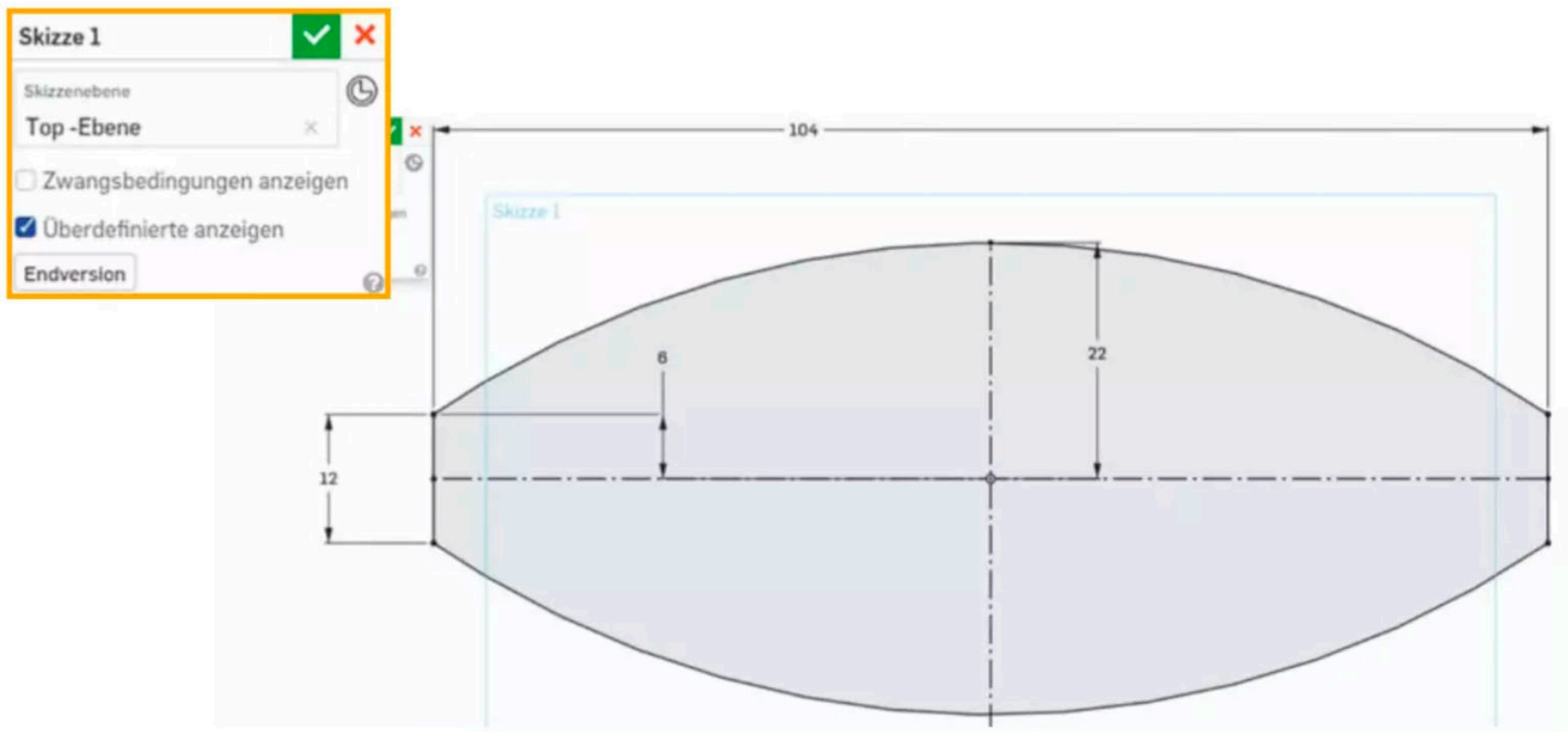
Ein Klick auf den Button „Skizze“ (der dritte von links in der Werkzeugleiste oben) erstellt eine neue Zeichnung auf dieser Ebene. In dieser von Onshape als „Skizze“ bezeichneten Zeichnung legen Sie nun zunächst ganz zweidimensional den Umriss des Bauteils fest. Dieses Vorgehen ist typisch für CAD: Aus 2D-Zeichnungen entstehen Körper mit einem Volumen, die Sie verbinden, schneiden und nachbearbeiten können. Eine Zeichnung auf einer der Grundebenen ist stets der erste Schritt.

Die Ansicht bleibt beim Erstellen einer Skizze einfach stehen. Zum Zeichnen in 2D ergibt es aber Sinn, gerade von oben auf die Skizze zu schauen. Die perfekte Ansicht erreichen Sie mit einem Rechtsklick und dem Menüpunkt „Ansicht senkrecht zur Skizzenebene“. Für eine bessere Übersicht können Sie nun noch die Standardebenen mit dem Auge-Symbol in der linken Seitenleiste ausblenden. Danach sehen Sie nur noch den Nullpunkt und eine blasse Umrandung für die Zeichenebene.

Das Vorbild für das Ersatzteil hat runde Seitenflächen und kurze gerade Schmalseiten. So eine Form entsteht aus zwei Kreisteilen und zwei Linien. Starten Sie dafür mit einem „3-Punkt-Bogen“ (in der Werkzeugleiste rechts vom Kreis). Klicken Sie einmal



**Zu diesem Artikel gibt es ein umfangreiches Video mit allen Schritten und detailreichen Erklärungen.**




**Zwei Kreisbögen und zwei kurze Linien begrenzen die Grundfläche des Ersatzteils. Die Maße sind vom Originalteil abgenommen.**

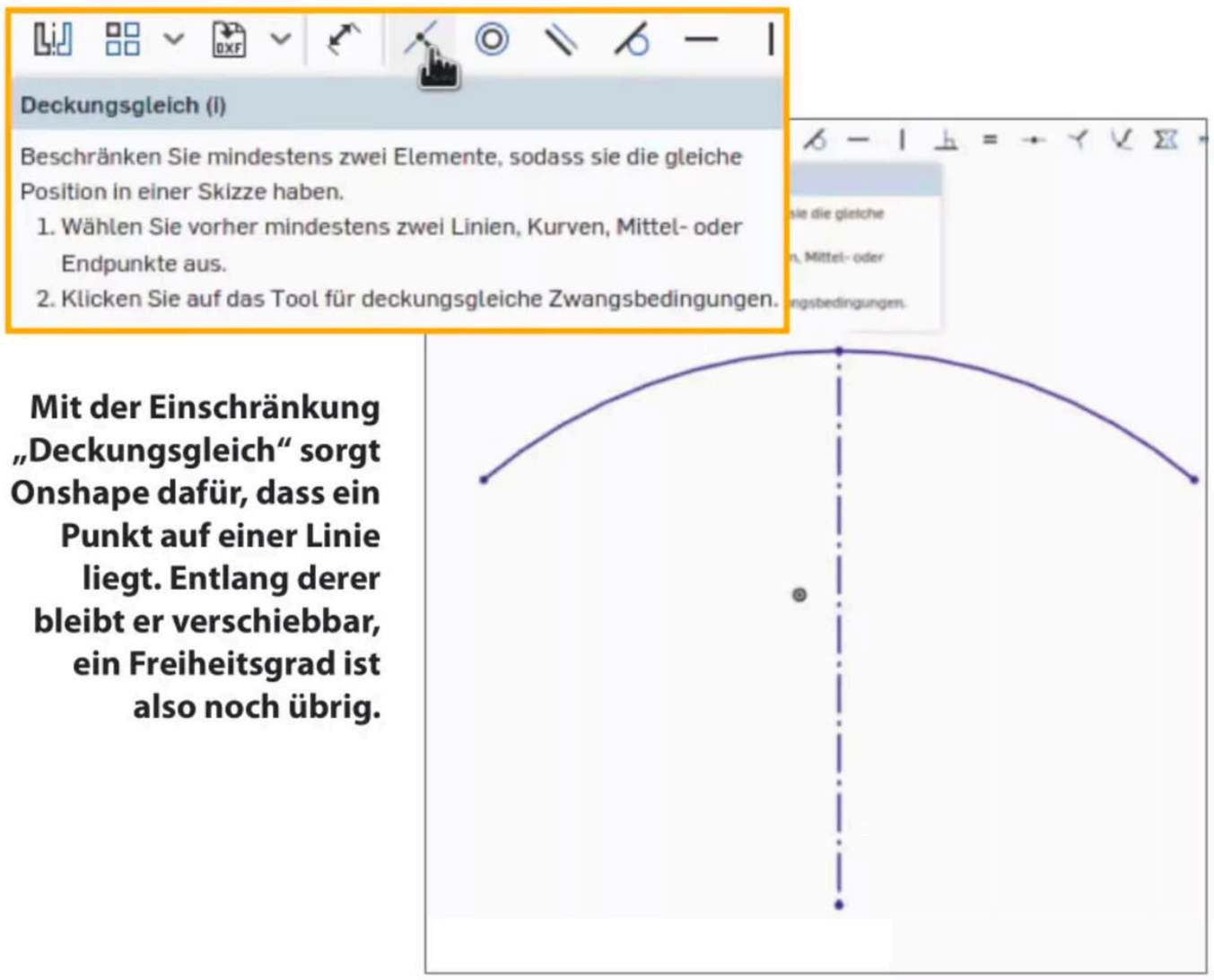
mit der linken Maustaste links in die Zeichnung, dann einmal auf die rechte Seite und legen Sie mit einem dritten Klick den Radius des Kreisteils fest. Wo die Punkte landen und wie gekrümmt der Kreisabschnitt ist, spielt erst mal keine Rolle.

## Freiheitsgrade einschränken

Onshape zeichnet den Bogen in Blau, was anzeigt, dass er noch „Freiheitsgrade" hat. Freiheitsgrade sind eine Grundidee vom CAD. Legt man Abstände, Radien, Parallelitäten und andere „Constraints" fest, bleiben Punkten und Linien immer weniger Richtungen, in die sie sich bewegen könnten, ohne die festgelegten Einschränkungen zu missachten. Ein unbemaßter Punkt hat beispielsweise zwei Freiheitsgrade, weil er sich in zwei Richtungen verschieben lässt. Legt man dagegen seinen Abstand zu einer Linie fest, hat er nur noch einen Freiheitsgrad, weil er sich nur noch entlang der Linie verschieben lässt. Beim Konstruieren versucht man üblicherweise, alle Freiheitsgrade wegzunehmen, wodurch das Programm exakt berechnen kann, wo sich Punkte und Linien befinden müssen. Hat man ein Zeichenelement vollständig eingeschränkt, zeichnet es Onshape in Schwarz statt Blau.

Beim Kreisbogen sind noch viele Freiheitsgrade übrig, die Sie nun Schritt für Schritt einschränken. Sorgen Sie zunächst dafür, dass die beiden Endpunkte auf der gleichen Höhe bleiben. Das geht mit dem „Horizontal"-Constraint (Symbol mit einer horizontalen Linie in der Werkzeugleiste an zehnter Stelle von rechts). Aktivieren Sie dafür das Werkzeug mit einem Klick und selektieren Sie anschließend nacheinander beide Endpunkte. Onshape passt die Zeichnung nun mit einem Sprung an die neue Einschränkung an.

Als nächste Einschränkung bemaßen Sie den Abstand zum Ursprung des internen Koordinatensystems (Nullpunkt). Das Bemaßungswerkzeug befindet sich links neben den anderen Constraints in einem eigenen Bereich. Das Symbol zeigt einen schrägen Pfeil mit zwei Spitzen, die auf zwei dünne kurze Linien zeigen. Aktivieren Sie das Werkzeug und wählen Sie zuerst den Nullpunkt und danach den linken Endpunkt des Kreissegments aus. Unter dem Mauscursor erscheint nun ein Maß (Pfeil zwischen schmalen Hilfslinien). Bewegen Sie den Mauscursor etwas herum, wechselt Onshape zwischen direktem, horizontalem oder vertikalem Abstand. Führen Sie die Maus links neben den Kreisteil-Endpunkt und bestätigen Sie das Maß für den




**Mit der Einschränkung „Deckungsgleich“ sorgt Onshape dafür, dass ein Punkt auf einer Linie liegt. Entlang derer bleibt er verschiebbar, ein Freiheitsgrad ist also noch übrig.**

vertikalen Abstand mit einem weiteren Klick. Nun erscheint ein Eingabefenster, in das Sie den gewünschten Abstand eintippen: „6 mm“. Onshape versteht die üblichen Einheiten und rechnet automatisch in die eingestellte Standardeinheit um. Die sollte bei Ihrem Account auf Millimeter konfiguriert sein.

Das Bemaßungswerkzeug bleibt standardmäßig aktiviert, was Sie nutzen können, um gleich noch den Abstand zwischen Kreisbogen und Nullpunkt festzulegen. Klicken Sie dafür beides nacheinander an und platzieren die Abstandsangabe danach dort, wo sie gut erkennbar bleibt. Geben Sie hier „22 mm“ an. Ergänzen Sie außerdem den horizontalen Abstand zwischen den beiden Endpunkten von 104 Millimetern.

## Hilfslinien

Um den Kreisbogen nun symmetrisch zu platzieren, wäre eine senkrechte Mittellinie als Hilfslinie praktisch. Die zeichnen Sie einfach mit dem „Linie“-Werkzeug (bei den Zeichenwerkzeugen ganz links). Bei den Zeichenwerkzeugen aktiviert der Button ganz rechts (das Icon zeigt eine durchgezogene und gestrichelte Linie mit horizontalen Pfeilen dazwischen) den Hilfslinien-Modus. Den können Sie entweder vor dem Zeichnen der Linie zusätzlich zum Linienwerkzeug aktivieren, wodurch sofort gestrichelte Hilfslinien erscheinen. Oder Sie zeichnen erst die Linie, selektieren Sie und klicken dann auf das Werkzeug, was die durchgezogene Linie zur gestrichelten Hilfslinie umwandelt. Hilfslinien-Kreise et cetera funktionieren analog.

Sorgsam platzierte Mausklicks aktivieren auch bei der Hilfslinie bereits beim Zeichnen Constraints, was viel Zeit sparen kann. Fahren Sie dafür zuerst mit der Maus auf dem Bogen entlang, bis dort ein oranges Quadrat um den Mittelpunkt auftaucht. Klicken Sie dort, um eine Mittellinie anzufangen, die Sie nun nach unten ziehen. Solange die Maus dabei nur wenige Pixel von der Vertikalen abweicht, zeigt Onshape neben dem Mauszeiger das Symbol für das Vertikal-Constraint (Icon: eine vertikale Linie) an. Der Kreismittelpunkt muss auf der Vertikalen liegen, weshalb Sie die Linie bis zum Mittelpunkt verlängern können. Wenn Sie dort klicken, setzt Onshape statt des Vertikal-Constraints ein Deckungsgleich-Constraint für den Punkt. Welche der beiden Optionen Sie wählen, spielt keine Rolle, da die Linie in beiden Fällen perfekt vertikal bleibt.

Die Hilfslinie führt nun knapp am Nullpunkt vorbei. Um den Kreisbogen exakt mittig zu verschieben, setzen Sie einfach ein Deckungsgleich-Constraint (bei den Constraints der Knopf ganz links, der einen Punkt, eine dicke und eine dünne diagonale Linie zeigt). Aktivieren Sie dafür das Werkzeug und klicken Sie dann zuerst auf die Hilfslinie und dann auf den Nullpunkt. Bogen und Hilfslinie springen nun in eine symmetrische Position.

## Spiegeln

Zeichnen Sie als Nächstes eine vertikale Linie, die am linken Endpunkt des Kreisbogens beginnt. Das Vertikal-Constraint können Sie wie bei der Hilfslinie schon beim Festlegen des Endpunkts setzen. Die Länge setzen Sie danach mit dem Bemaßungswerkzeug auf 12 Millimeter.

Für das symmetrische Pendant auf der anderen Seite können Sie mit dem „Spiegeln“-Werkzeug etwas Zeit sparen. Das befindet sich in der Mitte des Bereichs neben den Zeichenwerkzeugen etwa in der Mitte der Werkzeugleiste. Beim Spiegeln kommt es auf die richtige Reihenfolge der Klicks an. Aktivieren Sie zuerst das Werkzeug. Selektieren Sie dann die Linie, an der gespiegelt werden soll; im Beispiel ist das die gestrichelte vertikale Hilfslinie. Selektieren

Sie dann das zu spiegelnde Element, die kurze Linie links. Die neue Linie taucht dann sofort am anderen Ende des Kreisbogens auf. Länge und Position übernimmt die neue Linie automatisch in Form eines Gespiegelt-Constraint, sodass die Zeichnung nach der Operation sofort wieder vollständig festgelegt ist (keine der Linien ist blau).

Nun fehlt noch ein symmetrischer Kreisbogen auf der Unterseite. Den zeichnen Sie entweder wie den ersten Kreisbogen mit Horizontal-, Vertikal- und Deckungsgleich-Constraints, oder Sie zeichnen eine horizontale Hilfslinie durch den Nullpunkt, an der Sie den oberen Bogen spiegeln. Der letzte Bogen schließt die Form, was Onshape mit einer blassgrauen Hinterlegung anzeigt. Diese Markierung zeigt, dass Sie die gezeichnete Form nun extrudieren, ihr also eine Dicke geben können. Schließen Sie dafür zunächst den Zeichenmodus, indem Sie die Skizze mit dem grün hinterlegten Haken im Fensterchen links oben quittieren.

## 2D zu 3D

Eine Skizze ist immer flach. Um eine dreidimensionale Form daraus zu machen, bietet Onshape diverse Werkzeuge in der oberen Leiste ganz links an. Das wichtigste heißt „Linear austragen“ und sein Icon zeigt einen weißen Quader, aus dessen Breitseite ein kleiner grauer Quader herauswächst. Das Werkzeug extrudiert eine flache Form orthogonal zu der Skizze. Das heißt, aus dem zweiseitig abgerundeten Rechteck der bisherigen Skizze wird ein Quader mit ausgewölbten Seiten. Anders ausgedrückt verpasst das Werkzeug einer Form eine Dicke.

Sobald Sie es aktiviert haben, erscheint in der oberen linken Ecke ein Fensterchen mit Parametern der Operation. In der obersten Reihe lassen Sie den Reiter „Volumenkörper“ stehen. Darunter gibt es die Reiter „Neu“, „Hinzufügen“, „Entfernen“ und „Schneiden“. Auch hier lassen Sie zunächst „Neu“ stehen, auf alle anderen Modi gehen wir später ein. Darunter befindet sich bereits vorausgewählt (hellblau hervorgehoben) eine leere Liste mit Formen.

Klicken Sie nun in den grau hervorgehobenen Bereich der Zeichnung, um diese Form für das Werkzeug auszuwählen. Es erscheint dann „Fläche von Skizze 1“ in der hellblau hervorgehobenen Liste oben im Fensterchen. Drehen Sie anschließend die Ansicht mit der rechten Maustaste, um zu sehen, dass die Form in der 3D-Ansicht bereits eine Dicke bekommen hat. Die ist allerdings noch etwas hoch.

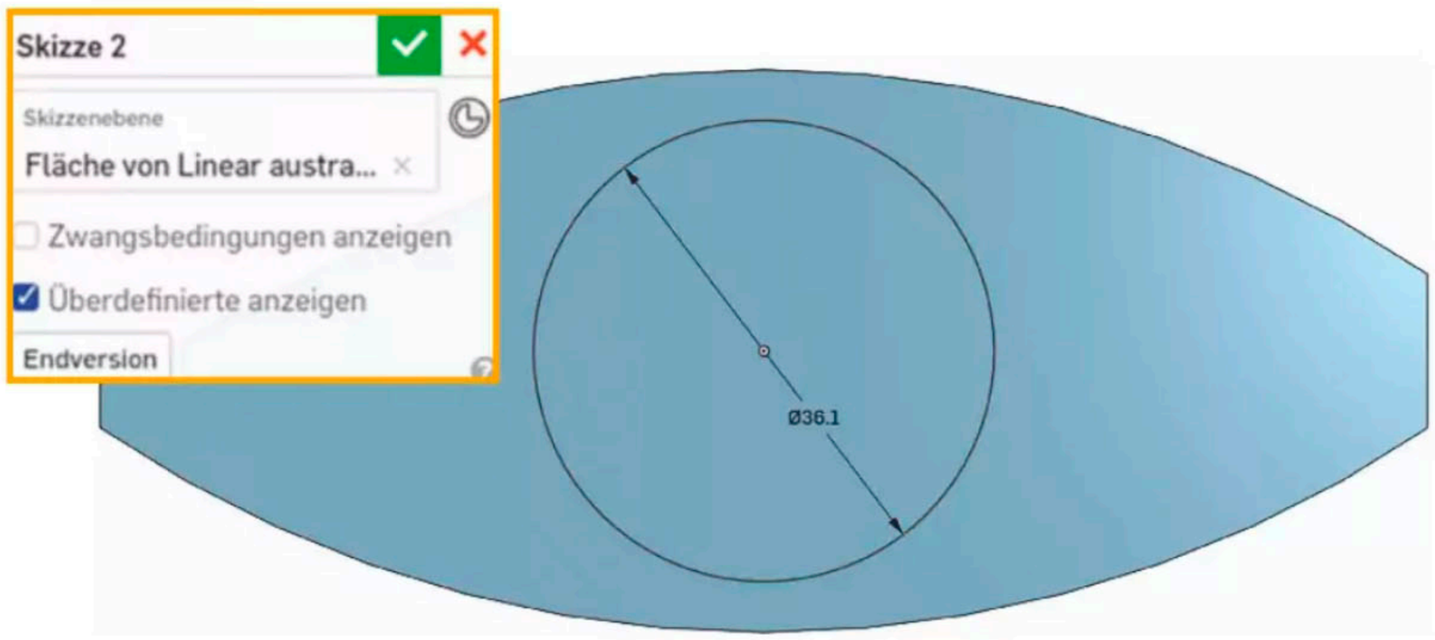
Da die Combo-Box unter der Liste von vornherein auf „Blind“ steht, können Sie darunter direkt eine „Tiefe“ angeben. Onshape hat hier mit einem Wert von 25 Millimetern gestartet, damit Sie schon mal etwas sehen können. Ändern Sie das Maß auf 18,5 Millimeter. In welche Richtung sich die Form verdickt, entscheidet der Pfeil neben der Combobox. In der Standardeinstellung wächst die Form nach oben heraus, was für das Strax-Ersatzteil genau richtig ist. Sie können das Fensterchen daher nun mit dem grün hinterlegten Haken schließen. Jetzt erscheint in der 3D-Ansicht ein blaues Teil mit der richtigen Dicke und in der Liste der Arbeitsschritte in der linken Seitenleiste steht „Skizze 1“ in Grau und darunter in Schwarz „Linear austragen 1“.

## Entfernen

Im nächsten Schritt erhält das Teil eine Bohrung auf der Unterseite, die aber nicht ganz bis zur Oberkante reicht. Die erzeugen Sie mit einer Vertiefung, die ebenfalls vom Linear-Austragen-Werkzeug ausgebohrt wird. Dafür brauchen Sie aber zuerst eine neue Skizze. Als Skizzenebene wählen sie diesmal nicht eine der Grundebenen aus, sondern selektieren stattdessen den flachen Boden des schon gezeichneten Teils. Drehen Sie dafür das Bauteil mit der rechten Maustaste so, dass Sie den Boden sehen können und selektieren ihn dann mit einem Linksklick. Wählen Sie danach mit dem Rechtsklick-Menü „Ansicht senkrecht zur Skizzenebene“.

Die Skizze besteht nur aus einem einzelnen Kreis. Wählen Sie das Kreiswerkzeug, klicken Sie dann auf den Nullpunkt und ziehen Sie den zweiten Kreispunkt dann knapp vor den Kreisbogen der Seitenwand. Bestätigen Sie den Kreis mit einem Klick. Durch den festgelegten Mittelpunkt kann sich der Kreis nun schon nicht bewegen, da der Durchmesser aber noch nicht festgelegt ist, zeichnet Onshape ihn noch blau. Setzen Sie den Durchmesser mit dem Bemaßungswerkzeug auf 36,1 Millimeter, um die Skizze vollständig festzulegen. Schließen Sie die Skizze mit einem Klick auf das grün hinterlegte Häkchen im Fensterchen.

Mit „Skizze 2“ sind Sie nun vorbereitet für „Linear austragen 2“. Wählen Sie nun den Kreis statt der äußeren Form („Fläche von Skizze 2“) und ändern Sie den Reiter in der zweiten Zeile auf „Entfernen“. „Blind“ bleibt bestehen, aber bei Tiefe stellen Sie 14,5 Millimeter ein, um das Teil nicht ganz durchzubohren. Sollte in Ihrer 3D-Ansicht ein roter Zylinder herauswachsen, statt ein Loch zu bohren, korrigieren




**Jede ebene Fläche eines Objekts taugt auch als Skizzenebene, beispielsweise um mit einem Kreis eine Bohrung zu skizzieren.**

Sie die Arbeitsrichtung des Werkzeugs mit dem Pfeil neben „Blind“. Bestätigen Sie die Bohrung mit dem grün hinterlegten Häkchen im Fensterchen.

## Mengenlehre

Im nächsten Arbeitsschritt bekommt die Form eine abgerundete Oberkante. Das geht über den „Schneiden“-Modus von „Linear austragen“. Um zu verstehen, wie der arbeitet, stellt man sich 3D-Objekte als Menge aller von ihrer Form eingeschlossenen Punkte vor. Beim „Schneiden“ bleiben alle Punkte stehen, die zu zwei Formen gehören. Was mathematisch klingt, ist in der Praxis gar nicht so kompliziert: Um nur die obere Fläche abzurunden, konstruiert man ein Objekt, in das die gesamte bisher erstellte Form hineinpasst. Nur die Ränder der oberen Fläche stehen aus dem Objekt heraus und werden deswegen abgeschnitten.

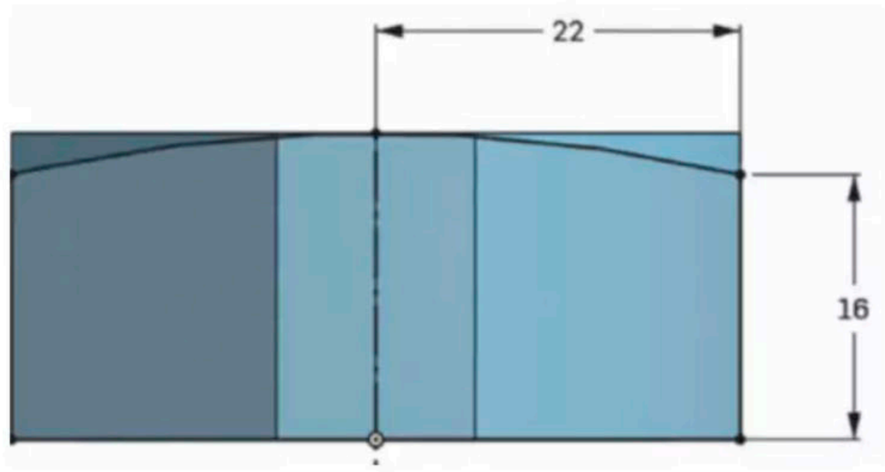
Erstellen Sie dafür zunächst eine neue Skizze und wählen Sie die rechte Schmalseite als Skizzenebene. Die Kanten des existierenden Objekts wirken in dieser Zeichnung wie Hilfslinien, sodass Sie mit fünf Klicks und dem Linien-Werkzeug ruckzuck die bestehende Form seitlich und unten in einem eckigen U eingefasst haben. Setzen Sie dabei die oberen Eckpunkte etwas unterhalb der bestehenden Form auf die Seitenkante, da hier ein Kreisbogen definieren soll, was weggeschnitten wird.

Zeichnen Sie dann einen Kreisbogen von einem Ende des U zum anderen. Richten Sie den Mittelpunkt des Bogens mit einer Hilfslinie orthogonal und mittig zur Oberkante aus. Die beiden Punkte an den Seitenkanten zwingt das auf die gleiche Höhe, der Radius des Kreises bleibt aber noch veränderlich. Setzen Sie dann mit dem Deckungsgleich-Constraint den Mittelpunkt des Kreisbogens auf die Oberkante der bisherigen Zeichnung. Wenn Sie nun mit dem Bemaßungswerkzeug die Höhe einer Seite auf 16 Millimeter festlegen, ist die Zeichnung vollständig definiert.

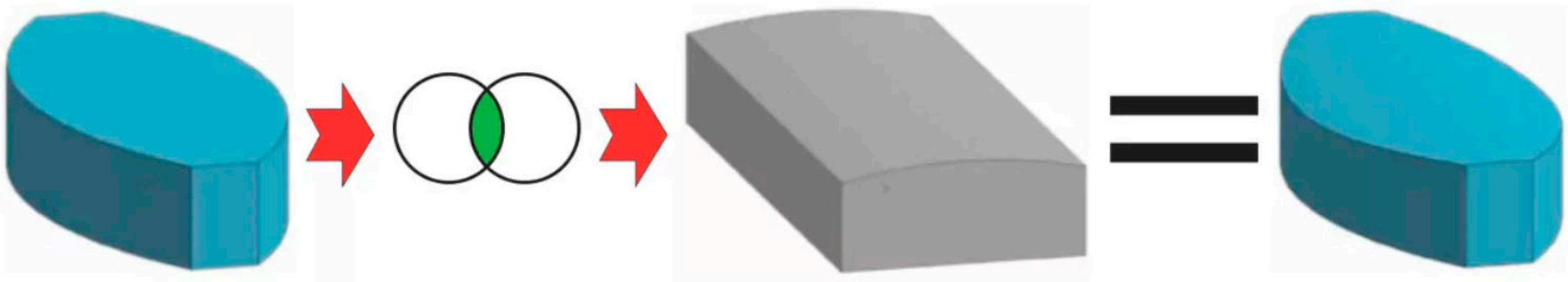
Quittieren Sie die Zeichnung mit dem grün hinterlegten Haken und wählen Sie schon wieder „Linear austragen“. Als Form dient die gerade erstellte Zeichnung. Onshape teilt die gezeichnete Form in drei Abschnitte, die Sie mit drei Klicks alle auswählen. In der Liste im Fensterchen steht dann dreimal „Fläche von Skizze 3“. Als Modus (zweite Zeile) wählen Sie nun „Schneiden“. Für die Tiefe vertrauen Sie nun nicht mehr auf „Blind“ und stellen die Combo-Box stattdessen auf „Bis Fläche“ um. Dadurch verschwindet die Tiefenangabe darunter und stattdessen zeigt ein Rahmen mit der Beschriftung „Bis Fläche“ an, dass Sie hier eine Fläche auswählen dürfen. Klicken Sie in diesen Rahmen, um ihn zu selektieren, wodurch er blau hinterlegt wird. Nun können Sie die Schmalseite auf der anderen Seite auswählen.

Die Schneiden-Operation entfernt gar nicht so viel, weshalb Ihnen in der Vorschau eventuell nicht sofort auffällt, was wirklich abgeschnitten wird. Drehen Sie deswegen die Ansicht in verschiedene Richtungen, bis Sie einen visuellen Eindruck bekommen haben, wo die Operation etwas wegschnitzt.

Falls Sie von der Mengenlehre-Erklärung verwirrt sind, können Sie testweise auf „Blind“ zurückstellen




**Beim Schneiden-Modus vom Linear-Austragen-Werkzeug zeichnen Sie eine Form, die das bisherige Objekt umschließt. Was sie nicht umschließt, wird abgeschnitten.**



**Beim „Schneiden“ bleibt die Schnittmenge stehen. Mathematisch: die Menge aller Punkte, die zu beiden Objekten gehören. Das Teil hat danach eine oben gewölbte Fläche.**

und mit der Tiefe herumspielen. Ein paar Experimente mit verschiedenen Tiefen und ein paar Drehungen des geschnittenen Objekts zeigen jeweils, was passiert und geben so einen intuitiveren Eindruck, wie das Werkzeug funktioniert. Mathematik klingt manchmal kompliziert, weil sie in Formeln oder Worte fasst, was man besser mal gesehen hätte. Onshape ist aber durchaus geeignet, um zu zeigen, wie sich Mengen schneiden.

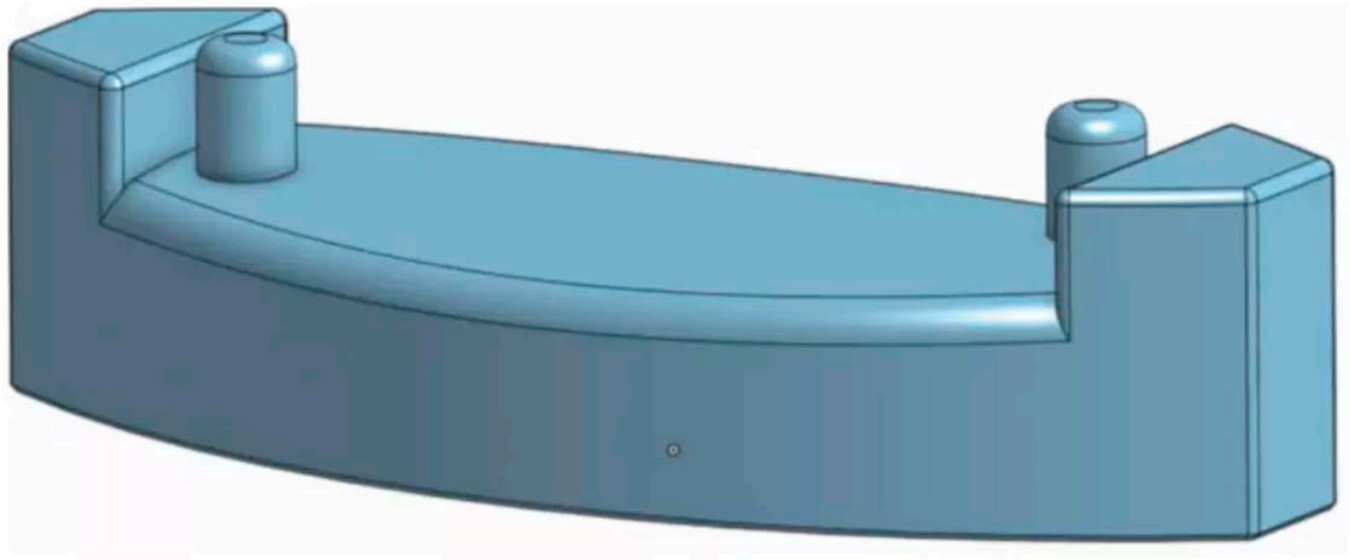
## Hinzufügen

Das Ersatzteil bekommt nun an beiden Enden einen Höcker, damit die Fahrbahn seitlich nicht wegrutschen kann. Erstellen Sie dafür eine neue Skizze auf der Unterseite des Teils und drehen Sie die Ansicht senkrecht zur Skizzenebene. Zeichnen Sie zuerst die Grundfläche des linken Höckers mit zwei vertikalen Linien und zwei Bögen, deren Endpunkte auf dem bestehenden Kreisbogen liegen. Den Abstand zwischen den geraden Linien stellen Sie auf 13 Millimeter ein. Für den Bogenradius nutzen Sie das „Gleich“-Constraint (=-Symbol), für das Sie zuerst auf den großen Bogen und dann auf den kleinen Bogen klicken. Die beiden Bögen und die Linie außen liegen danach exakt auf der bestehenden Kontur.

Zeichnen Sie nun eine Hilfslinie quer durch die bestehende Form und durch den Nullpunkt. An dieser Hilfslinie können Sie nun alle bereits gezeichneten Elemente spiegeln. Selektieren Sie dafür zunächst ohne aktiviertes Werkzeug beide Linien und Bögen, indem Sie mit gedrückter Maustaste einen Rahmen darum ziehen. Aktivieren Sie dann das Spiegeln-Werkzeug und selektieren Sie zuletzt die Hilfslinie. Nun enthält die Skizze Grundflächen für beide Höcker und Sie können sie mit dem Haken quittieren.

Anschließend wählen Sie schon wieder „Linear austragen“, stellen den Modus-Reiter in der zweiten Zeile aber auf „Hinzufügen“. Selektieren Sie dann beide Grundflächen der Höcker. Die Combo-Box bleibt „Blind“, aber die Tiefe erhöhen Sie auf 26 Millimeter. Anders als „Neu“ erzeugt „Hinzufügen“ kein separates 3D-Objekt, sondern ergänzt die bestehende Form um das vom Werkzeug definierte Volumen. Welche Form es ergänzt, steht ganz unten im Fensterchen unter „Zusammenführungsbereich“, wobei Sie für das Beispiel an „Part 1“ nichts ändern müssen.

Die letzte Ergänzung zur Form sind zwei Zylinder, die diagonal aus der Fläche ragen und in Aussparungen im Fahrbahnrand passen. Zeichnen Sie dafür in einer neuen Zeichnung auf der Unterseite zwei Kreise mit einem Durchmesser von 7,2 Millimetern und einem direkten Abstand der Mittelpunkte von 66 Millimetern. Der vertikale Abstand zum Nullpunkt beträgt jeweils 5 Millimeter, allerdings einmal nach



**Abgerundete Kanten und Fasen sind schnell erzeugt und geben dem Objekt ein professionelles Aussehen. 3D-Drucker bauen die vielen Rundungen klaglos nach.**

oben und einmal nach unten. Beide Kreise extrudieren Sie mit „Linear Austragen“ blind auf 28 Millimeter.

## Rundungen und Fasen

Das Objekt hat nun prinzipiell die richtige Form, es ist allerdings mit scharfen Kanten gespickt. Die sollten des Neffen feine Kinderhände besser nicht in Gefahr bringen! Das Werkzeug „Verrundung“ macht daraus schön abgerundete Kanten, als ob man das Bauteil mit der Feile bearbeitet hätte. Selektieren Sie dafür zuerst mit Shift und Mausklicks alle Kanten, die mit dem gleichen Radius rund werden sollen; im Beispiel fangen Sie dafür mit den gebogenen Kanten seitlich der oberen gekrümmten Fläche an. Drücken Sie dann auf das Verrundungswerkzeug in der Werkzeugleiste links. Das Icon zeigt einen Quader mit einer runden Kante oben links. Onshape zeichnet in der Vorschau nun sofort eine runde Kante, allerdings mit einem Standard-Radius von 5 Millimetern. Im Fensterchen oben links ändern Sie den auf 3 Millimeter. Runden Sie danach mit 2 Millimetern Radius die Oberkanten der Zylinder und nach Belieben weitere Kanten oben und seitlich mit einem Millimeter ab.

Für die Kante unten verwenden Sie besser eine „Fase“, also eine mit 45° abgeschrägte Kante. Das Werkzeug finden Sie in der Werkzeugleiste direkt rechts von der Verrundung. Bei einer Fase entsteht kein Überhang steiler als 45°, mit dem sich 3D-Drucker schwertun würden. Eine kleine Fase von mindestens 0,3 Millimetern an der Unterkante verhindert bei 3D-Druck-Vorlagen, dass eine zu knapp eingestellte Betthöhe einen sichtbaren „Elefantenfuß“ erzeugt. Druckt man allerdings mit Brim (ein vom Slicer, dem Druckbahn-Planer ergänzter Rand ums Druckobjekt), ist von der Fase nach dem Druck nichts zu sehen. Wir haben das Teil aus ABS gedruckt, weshalb wir aus Furcht vor Warping (verbogener Druck wegen thermischer Spannungen im Inneren) lieber einen Brim eingestellt haben und die Fase unten eher unnötig war. Wer mit PLA druckt, dürfte auch ohne Brim drucken können und dank der Fase eine ganz saubere Unterkante bekommen, die keine Nacharbeit verlangt.

## Strax repariert

Mit dem Strax-Ersatzteil haben Sie die wichtigsten Werkzeuge kennengelernt, die Sie für einen schnellen Einstieg in Onshape brauchen. Schauen Sie auf jeden Fall auch in unser Video auf YouTube hinein, in dem Sie alle Schritte mit Bild und Ton nachvollziehen können (siehe ct.de/wkwk). Das Programm kann aber auch Drehteile, Zugelemente und diverse andere Formen erzeugen. Die Doku des Herstellers hilft hier weiter und im offiziellen YouTube-Kanal finden Sie auch lehrreiche englische Videos, die erklären, wie Sie die fortgeschrittenen Werkzeuge einsetzen. Wie Sie technische Zeichnungen für Ihr frisch konstruiertes Teil erzeugen, haben wir ab Seite 136 erklärt. Bei uns gilt es nun erst mal mit einem 5-jährigen Achterbahn-Ingenieur wilde Fahrten durchs Wohnzimmer aufzubauen. Er war übrigens sehr zufrieden mit der Stabilität und Funktion des Ersatzteils. Altes Spielzeug kann ganz schön viel Spaß machen, wenn es denn funktioniert.

(pmk) c't

### Literatur

[1] Christoph Schmith, **CAD-Content**, 3D-Konstruktion mit FreeCAD, c't 13/2018, S. 140

**Screencast-Video bei YouTube und Link zur Original-Datei in Onshape**

ct.de/wkwk

Make:

DAS KANNST DU AUCH!




GRATIS!



2 × Make testen und 6 € sparen!

Ihre Vorteile:
✓ **GRATIS dazu:** Arduino Nano
✓ **Zugriff auf Online-Artikel-Archiv***
✓ **NEU:** Jetzt auch im Browser lesen!
✓ Zusätzlich digital über iOS oder Android lesen

**Für nur 15,60 Euro statt 21,80 Euro.**

* Für die Laufzeit des Angebotes.

**Jetzt bestellen:**
**make-magazin.de/miniabo**



FarmBot auf Instagram

Einführung FarmBot-Projekt

# Vollautomatisch gärtnern mit dem FarmBot

**Der Gartenbau-Roboter FarmBot verspricht, automatisch ein Hochbeet voller Gemüse zu umsorgen: Er sät, jätet, gießt und dokumentiert das Wachstum der Pflanzen. Ein Experiment für die Landwirtschaft der Zukunft.**

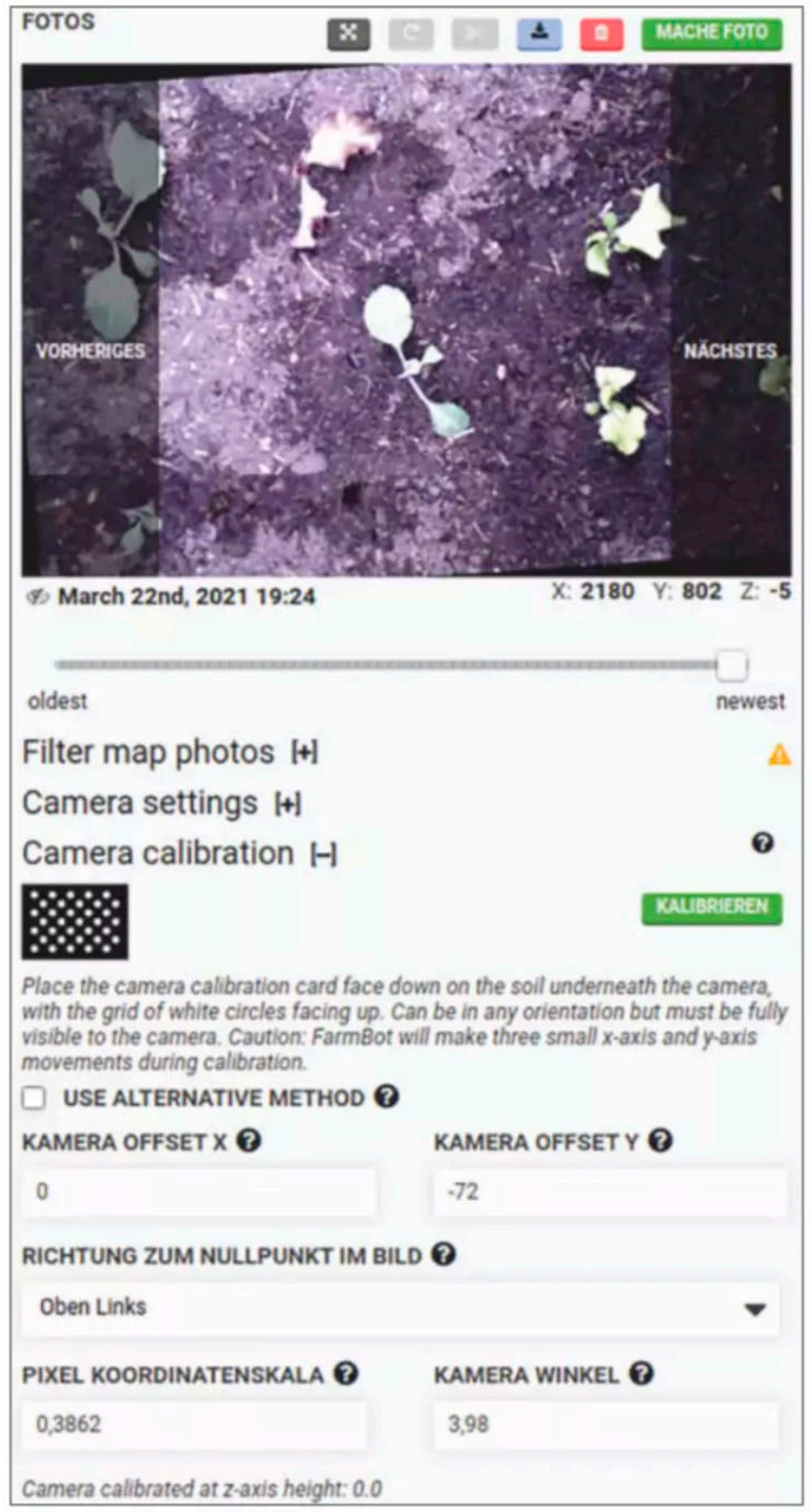
Von **Pina Merkert**


| | |
|---|---|
| Vollautomatisch gärtnern | 154 |
| Tipps für den Bau eines FarmBot | 158 |
| Roboter programmieren ohne Code | 166 |
| Der FarmBot im Praxiseinsatz | 172 |
| Mit dem Raspi besser bewässern | 177 |

Gärtnern macht Arbeit: Gießen, jäten, säen, beobachten – wie schön wäre es, wenn ein Roboter das Malochen übernähme, Tag und Nacht automatisch zur idealen Zeit. Ein Helfer, der nie etwas vergisst und nie Urlaub macht. Was wie Science-Fiction klingt, lässt sich für ein Hochbeet durchaus umsetzen. Der Roboter-Gärtner heißt „FarmBot“ und sieht nicht nach Science-Fiction aus: Die vom gleichnamigen Start-up entwickelte Maschine ähnelt eher einer Portalfräse als C3PO. Mit Riemen und Schrittmotoren rollt der FarmBot auf Aluminiumprofilen und positioniert so sein Werkzeug an jeder beliebigen Stelle in den 50 Zentimetern oberhalb eines 1,5 × 3 Meter großen Hochbeets. Ob Gießkanne, Jätwerkzeug, Bodenfeuchtesensor oder Einsähnadel – der FarmBot kann sein Werkzeug selbst wählen und wechseln. Wie er all diese Möglichkeiten sinnvoll nutzt, muss man allerdings selbst programmieren.

FarmBot Inc. verspricht ein vollautomatisches Hochbeet, das sich selbst hegt und pflegt, wenn Mensch keine Zeit dafür hat. Da die komplette Konstruktion als OpenHardware veröffentlicht ist und die Maschine einen Bereich des Lebens automatisiert, den wir bisher eher mit Tradition und Handwerk assoziierten, haben wir sie gebaut.

Smart Gardening beschränkt sich nämlich bisher auf ferngesteuerte Bewässerung und Rasenmäh-Roboter. Beetpflege gab es höchstens als Forschungsprojekt und dann auch nur für Bauern, die mit Robotern jäten, statt dem Unkraut mit Gift zu Leibe zu rücken. Der FarmBot verkleinert Roboter-Trends aus der landwirtschaftlichen Forschung auf ein einzelnes Beet. Diese neue Art von Gerät taugt nicht nur als Gartenbau-Experiment, sondern soll dem Gärtner auch viel Handarbeit ersparen.




**Die eingebaute Kamera kann der FarmBot mit einer gepunkteten Karte selbst kalibrieren. Mit den Daten dreht er die Bilder automatisch gerade.**

## Aufwand und Kosten

Unser FarmBot ist ein „kleiner“ Genesis in Version 1.5. Das amerikanische Start-up betreibt einen Shop für Einzelteile und komplette Bausätze. Als wir im Spätsommer 2020 mit dem Bau begannen, war der Bausatz aber nicht lieferbar. Momentan kann man die Teile für Version 1.6 für 2805 Euro vorbestellen. Wir orderten deswegen nur Einzelteile in Amerika (für den Einkauf muss man zusätzlich 19 Prozent Einfuhrumsatzsteuer einrechnen), bestellten Standardteile bei europäischen Händlern und fertigten viele weitere Bauteile mit 3D-Drucker und Fräsmaschine selbst. Im November 2020 zeigte der FarmBot-Shop den Bausatz als lieferbar an, zum Redaktionsschluss war wieder nur der Genesis XL lieferbar; der kleine Genesis kann vorbestellt werden. Je nach Lieferdatum kann es sein, dass Strafzölle der EU gegen US-Einfuhren den Bausatz um zusätzliche 25 Prozent verteuern.

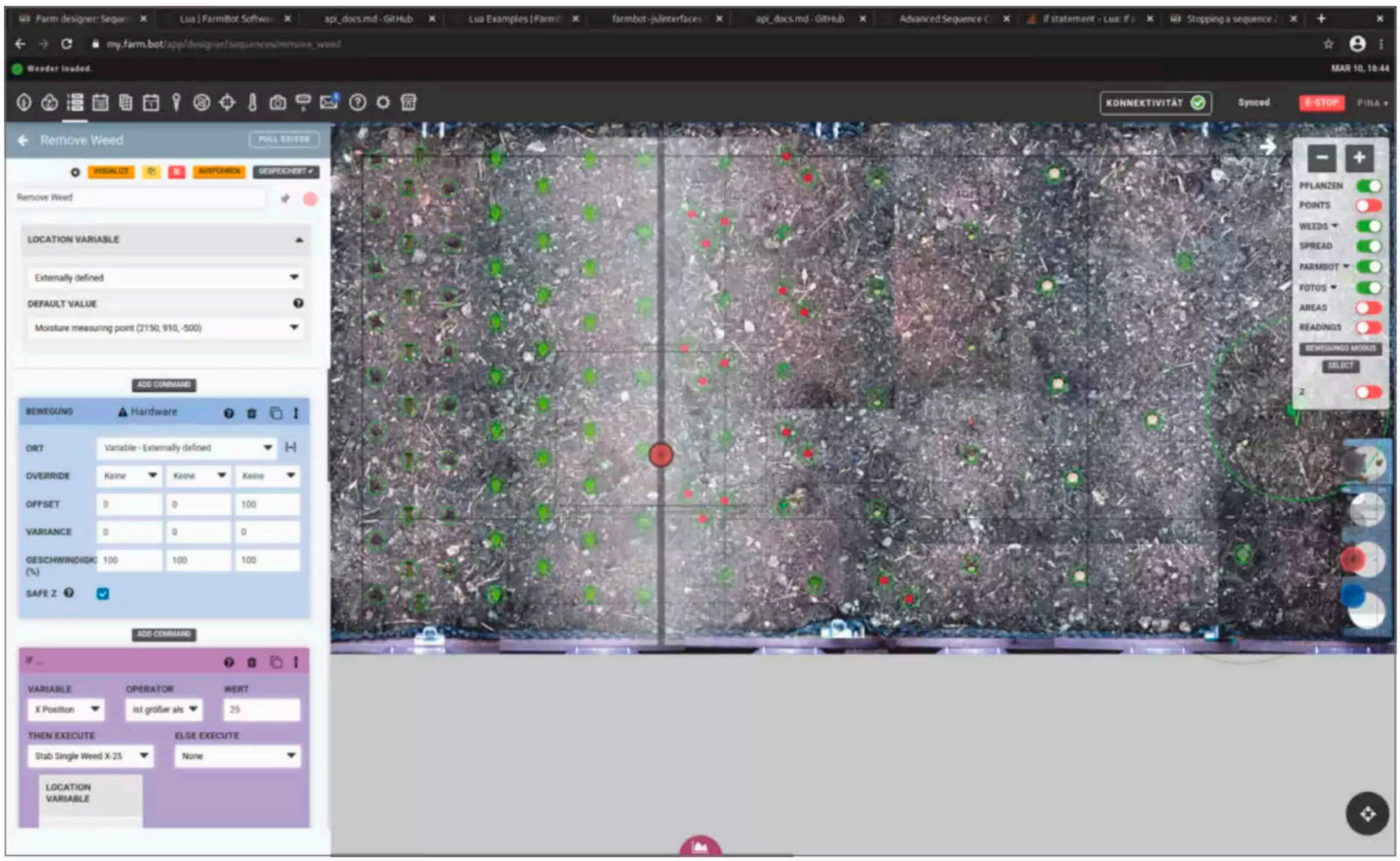
Mit vielen Stunden zusätzlicher Handarbeit konnten wir den Preis der Einzelteile auf circa 1600 Euro drücken (ohne Hochbeet, dessen Teile beim Bausatz

auch nicht dabei sind). Der Bausatz mit seinen vorkonfektionierten Kabeln, fertigen Aluplatten und sofort verbaubaren Spritzgussteilen hilft, den Zeitaufwand auf einige Wochenenden einzugrenzen. Wir halten ihn daher nicht für überteuert, sondern für eine gute Option, um gegen Bezahlung schneller fertig zu werden.

Neben dem FarmBot Genesis gibt es noch die abgespeckte Version „Express“ für 1273 Euro. Die Nachteile dieser Variante beschreiben wir ab Seite 154. Beide Varianten gibt es als „XL“ auch in vierfacher Größe, also mit doppelter Länge und doppelter Breite. Für das nötige Riesen-Hochbeet, bei dem man von der Seite mit der Hand nicht mehr bis zur Mitte kommt, fehlte uns allerdings der Platz im Garten. Die größeren Bausätze sind nur moderat teurer, sodass sich ein erheblich günstigerer Preis pro Fläche ergibt.

## Exzellente Anleitung

Die englische Dokumentation zum FarmBot ist exzellent. Sie beschreibt nicht nur alle Schritte des Aufbaus mit Text und Bild, sondern ergänzt auch nützliche Hinweise an fehlerträchtigen Stellen. Sie geht allerdings davon aus, dass man einen Bausatz vor sich hat. Wer wie wir Teile selbst fertigt, muss Zeichnungen und Maße aus dem öffentlichen Onshape-Design des FarmBot exportieren (siehe S. 136). Alle Tipps zum Bau der Maschine lesen Sie ab Seite 158.




**Die Weboberfläche läuft zentral bei my.farm.bot und ist mit Passwort von überall im Internet erreichbar.**

## Programmieren ohne Code

Von sich aus tut der FarmBot gar nichts. Die Bauanleitung endet damit, das FarmBot-OS auf einem Raspberry Pi 3B im Elektronikkasten zu installieren und mit diesem die Firmware auf den „Farmduino“, das eigentliche Mainboard der Maschine, zu flashen. Bei der Einrichtung teilt man dem FarmBot die eigenen Zugangsdaten für my.farm.bot mit, einer vom Start-up gehosteten Weboberfläche, mit dem sich die Maschine auch aus der Ferne steuern und programmieren lässt.

In der Weboberfläche kann man dem FarmBot in Sequenzen Anweisungen geben und im Kalender und in „Regimens“ festlegen, wann er die ausführt. Mit „Regimen“ bezeichnen die FarmBot-Entwickler Kalender, mit denen sich Anweisungen ab der Einsaat einer Pflanze planen lassen. Die Entwickler haben für die Programmierung zwar eine blockbasierte Programmiersprache entwickelt, das Anordnen der Anweisungsblöcke fällt aber leichter, wenn man die Konzepte einer Programmiersprache bereits kennt. Uns hat die Oberfläche an die Block-Programmiersprache der Lego-Mindstorms-Roboter erinnert. Es gibt nur wenige vordefinierte Sequenzen. Ab Seite 166 erklären wir daher im Detail, wie man dem FarmBot das Gärtnern beibringt.

In der Oberfläche kann man sich auch computerspielartig Gärten zusammenklicken. Viele übliche Gemüse sind in der angebundenen Datenbank von OpenFarm bereits angelegt, neue Sorten kann man hinzufügen. Die Software taugt nicht nur als Planungswerkzeug für die Hochbeetbestückung, die Koordinaten der Pflanzen stehen auch für Sequenzen zur Verfügung. Eine Pflege-Anleitung in Form einer „Sequenz“ kann man ganz leicht für alle Pflanzen oder auch nur für die Radieschen abarbeiten lassen.

## Unkraut

Eine Automatik haben die Entwickler praktischerweise vordefiniert: Neben dem Werkzeug klemmt am Kopf des FarmBot eine Endoskopkamera. Definiert man eine Sequenz, die mit dieser Kamera das ganze Hochbeet abfotografiert, kann die Software automatisch Unkraut erkennen. Der FarmBot weiß nämlich, wo er Gemüse eingesät hat, und wenn die Kamera an anderer Stelle grüne Blättchen entdeckt, registriert er automatisch Unkraut-Pflanzen. Deren Koordinaten kann dann eine Jät-Sequenz nutzen, die die ungewollten Blättchen mit Kunststoffmessern in den Boden drückt. Sprießt ein Unkraut zu nah an einer Nutzpflanze, hält der FarmBot Abstand, man muss das ungewollte Gewächs dann per Hand entfernen.

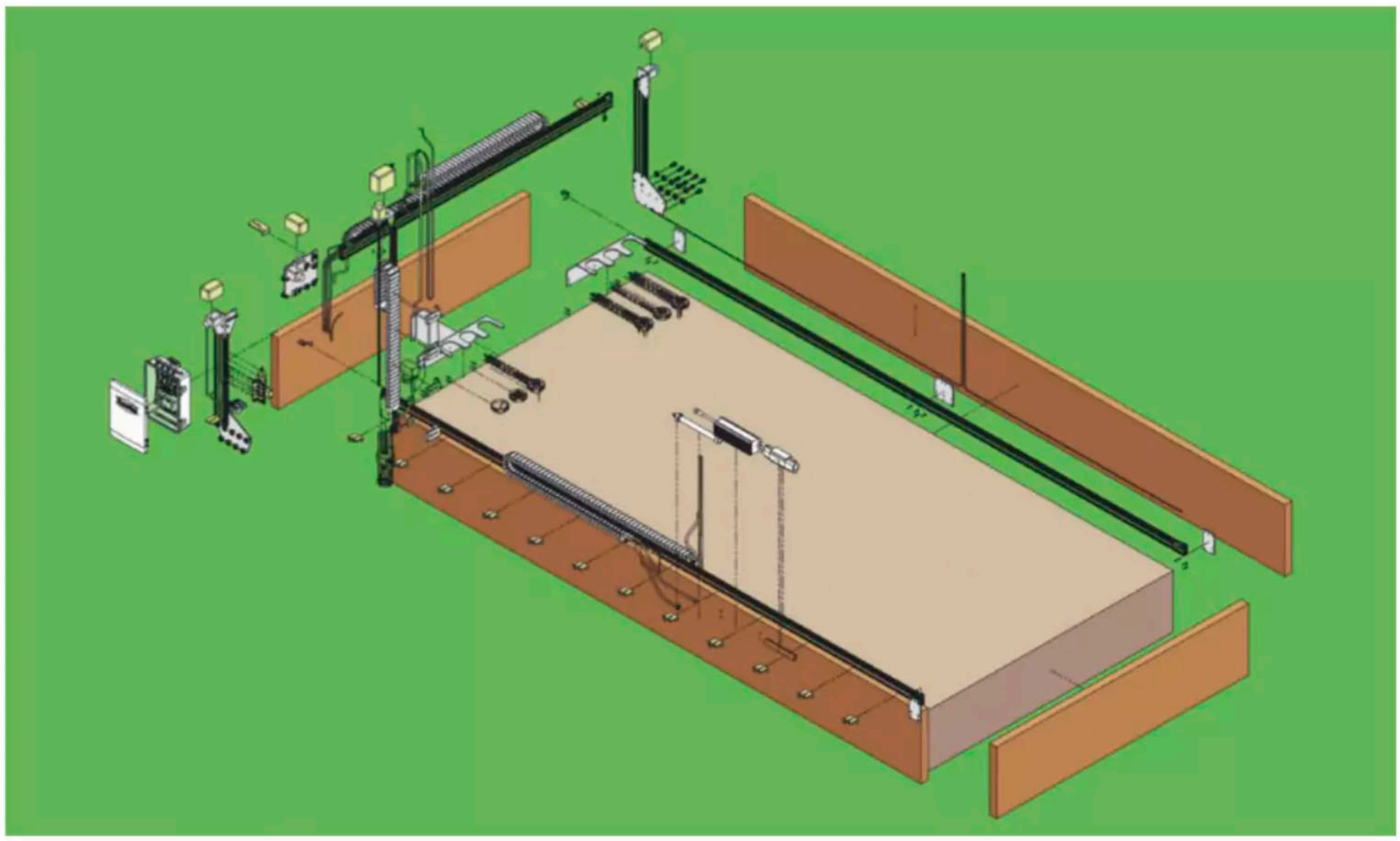
## Schlauer Garten

So viel „Smart Garden“ wie mit dem FarmBot haben wir noch nie gebaut. Die Maschine verspricht, viel mehr Gärtnerarbeit abzunehmen als Bewässerungssysteme oder Mähroboter. Wie intelligent der Roboter-Gärtner dabei vorgeht, hängt aber vom Besitzer ab, da man ihn ganz klassisch programmiert. Ob der FarmBot eine ideale Kombination aus Maschinenbau und Gärtnern darstellt oder eher eine teure Maschine ist, die kaum Handgriffe abnimmt, klären wir ab Seite 172. (pmk) c't

**Dokumentation, CAD-Design, Video**

ct.de/wy7z



# Tipps für den Bau eines FarmBot

**Schon allein wegen der Abmessungen von 1,5 × 3 Metern ist der FarmBot ein großes Projekt. Mit unseren Tipps und der guten offiziellen Anleitung bauen Sie den automatischen Gärtner zusammen, auch ohne Maschinenbaustudium.**

Von **Pina Merkert**

Der Bau eines FarmBot besteht aus drei Schritten: Zuerst bereiten Sie den Bauplatz im Garten vor und zimmern ein Hochbeet zusammen. Da dieses einen Strom- und einen Wasseranschluss haben muss, ist dieser Schritt erstaunlich komplex. Danach schrauben Sie in der Werkstatt Baugruppen des FarmBots zusammen und verdrahten den größten Teil der elektrischen Komponenten. Im dritten Schritt installieren Sie Schienen und Portal im Garten, schließen Strom und Wasser an, richten das Betriebssystem ein und stellen den einwandfreien Lauf aller drei Achsen sicher.

Bei all diesen Schritten dient Ihnen die offizielle Dokumentation als Anleitung. Sie leitet Sie schrittweise durch den Bau und weist auf typische Fehlerquellen hin. Bei Problemen hilft Ihnen die offizielle FAQ weiter. Klärt auch diese das Problem nicht, sollten Sie nicht davor zurückschrecken, ins CAD-Design hineinzuschauen. Das wurde im browserbasierten CAD-Programm Onshape erstellt, sodass Sie es an-

schauen können, ohne ein Programm installieren zu müssen. Wie Sie in Onshape navigieren, haben wir am Bespiel vom FarmBot ab Seite 136 erklärt. Einen Link zu beiden Dokumenten und zum öffentlichen CAD-Design finden Sie über ct.de/w95f.

Wir haben einen knapp 1,5 × 3 Meter großen FarmBot Genesis in Version 1.5 gebaut und dabei genug über die Maschine gelernt, um Ihnen einige wertvolle Tipps für den Bau Ihres eigenen FarmBot auf den Weg geben zu können. Wenn Sie sich noch nicht sicher sind, ob Sie einen FarmBot bauen und besitzen möchten, erlaubt Ihnen unser Baubericht, einzuschätzen, ob Sie sich das Projekt zutrauen.

## Roboterwahl

Bevor es losgeht, müssen Sie sich zuerst für ein Modell entscheiden. Es gibt den FarmBot Genesis mit allen Funktionen, der selbst Werkzeuge wechseln kann, und den erheblich billigeren FarmBot Express mit einem nicht erweiterbaren Universalwerkzeug. Jede der beiden Maschinen können Sie auch als XL-Variante in doppelter Breite und mit doppelter Länge bauen. Da Sie die Größe Ihres FarmBot einfach in der Software einstellen, funktionieren auch kleinere Abmessungen, falls Sie ein bestehendes Hochbeet nachträglich automatisieren möchten.



**Viel Technik, wenig Garten: Mit dem FarmBot kann man das Gemüsebeet perfekt planen. Das muss man aber auch, da die Anbaufläche begrenzt ist.**

Beim FarmBot Express sollten Sie bedenken, dass sein Portal einfach auf den Seitenbrettern des Hochbeets rollt. Diese müssen daher sehr gerade sein und eine glatte Oberseite aufweisen. Passendes Konstruktionsholz ist möglicherweise nicht leicht im Baumarkt zu bekommen. Drei Meter lange Bretter verbiegen oder verwinden sich schnell, besonders bei wechselnder Feuchtigkeit. Für den Genesis sind die Anforderungen ans Hochbeet geringer, da er auf Aluprofilen rollt, die an die Seiten des Hochbeets geschraubt werden. Wir dachten, wir hätten ein perfekt maßhaltiges Hochbeet gebaut, mussten nach dem Befüllen allerdings auf einer Seite Klötzchen zwischen Brett und Profilhalterung schrauben, um die verzogene Seitenwand auszugleichen.

Der zweite Faktor für den großen Preisunterschied ist der Werkzeugkopf. Beim Express ist dort ein Universalwerkzeug mit einem ausgehöhlten Kunststoffkegel angeschraubt. Durch das Loch kann er Samen mit einer Vakuumpumpe ansaugen. Da die Spitze aber auch als Jätwerkzeug dient, hinterlässt sie ein breites Loch im Boden. Beim Jäten zupft der FarmBot nämlich nicht nach Unkraut, sondern er ersticht es von oben.

Zum Gießen läuft Wasser beim Express einfach seitlich am Kegel entlang. Der Genesis hat an gleicher Stelle einen Werkzeughalter, an dem magnetisch Werkzeuge andocken. Zum Säen nutzt er Kanülen mit austauschbarer Dicke, sodass die Vakuumpumpe auch sehr kleine Samen ansaugen kann. Das Jätwerkzeug besitzt vier Kunststoffklingen statt einer breiten Spitze und das Bewässerungswerkzeug ähnelt mit vielen kleinen Löchern einer kleinen Gießkanne. Zum Standardrepertoire gehört außerdem ein Werkzeug mit einem Bodenfeuchtesensor, den der Express gar nicht besitzt. Der Werkzeughalter kontaktiert sogar noch zusätzliche elektrische Kontakte durch und neben Anschlüssen für Wasser und Vakuum gibt es einen freien Schlauchanschluss für Erweiterungen wie eine Düngerflüssigkeit. Beim Genesis kann man leicht eigene Werkzeuge konstruieren, die die Fertigkeiten des Roboters erweitern. In einer „Tool Bay“ parkt der Genesis die Werkzeuge und kann sie automatisch tauschen.

## Hochbeet

Ein FarmBot braucht zwingend ein Hochbeet. Sonderlich hoch muss das allerdings nicht sein. Die offizielle Doku empfiehlt einen Rahmen aus 30 Zentimeter hohen Brettern – wahrscheinlich die billigste Konstruktion. Die Entwickler haben auch mal einen




**Am „Universal Tool Mount“ (UTM) docken beim FarmBot Genesis magnetisch verschiedene Werkzeuge an, die der Roboter selbst wechseln kann. Rechts vom UTM hängt die Kamera.**

gigantischen 10 × 3 Meter großen FarmBot gebaut, dessen Aluprofile auf betonierten Pfosten ruhen – ebenfalls eine funktionierende Konstruktion.

Wir wollten uns nicht bücken müssen und haben unser Beet gut 80 Zentimeter hoch gebaut. Bei dieser Höhe konnten wir die Unterseite mit einem Gitter gegen Wühlmäuse absichern und hatten an der Außenseite genug Platz für einen Schneckenzaun, der den salatvertilgenden Nacktschnecken den Zugang verwehrt. Damit der Schneckenzaun funktioniert, mussten wir den Wasser- und Stromanschluss innen im Beet nach oben führen, da Kabel und Rohr den Schnecken sonst als Brücke gedient hätten.

Als Holz nutzten wir Terrassendielen aus Douglasie, da der hohe Harzgehalt dieses Holz auf natürliche Weise vor Verwitterung schützt. Allein auf das Harz wollten wir uns jedoch nicht verlassen und haben die Planken zusätzlich mit Bootsöl gestrichen. Zusammengehalten wird unser Beet mit Winkelprofilen aus Aluminium und Schlossschrauben aus Edelstahl. Die Konstruktion macht mehr Arbeit als der Vorschlag aus der FarmBot-Doku. Falls sie Ihnen gefällt, finden Sie eine CAD-Zeichnung unseres Beets mit allen Maßen über ct.de/w95f.

Ein tiefes Hochbeet erlaubt den Luxus, in seinen tieferen Schichten Biomasse unterzubringen, die über mehrere Jahre zerfällt und dabei kontinuierlich Nährstoffe an den Boden im Beet abgibt. Wir haben die unteren 40 Zentimeter mit grobem Baum- und Heckenschnitt befüllt und danach eine Schicht frischen Kompost ausgebreitet. Darüber kam dann eine Schicht des am FarmBot-Bauplatz vorherrschenden Lehmbodens, danach eine Schicht Herbstlaub und im Frühjahr zuoberst fünf Zentimeter Komposterde. Wenn Sie Ihr Hochbeet so oder so ähnlich befüllen, starten Sie im ersten Jahr mit sehr nährstoffreicher Erde, in der alle Gemüse gut gedeihen. Der Nährstoffgehalt nimmt in den Folgejahren zwar ab, wegen der eingebrachten Biomasse aber langsamer als in üblichen Gemüsebeeten. Falls Sie für Ihren Standort unsicher sind, ob die Erde zu mager sein könnte, nehmen Sie eine Bodenprobe und schicken sie zur Laboranalyse. Für unseren Standort ergab die Bodenanalyse keinen Düngebedarf.

Wer im Frühjahr besonders früh anfangen möchte, kann das Beet nachts mit einer Folie abdecken. Damit die Folie die kleinen Pflänzchen nicht niederdrückt, haben wir sie über bogenförmige Abschnitte eines Plastik-Wasserrohrs von 1,6 Metern Länge gespannt. Die Befestigungen dieser Bögen an der Beetwand kommen aus dem 3D-Drucker, die Vorlagen finden Sie über ct.de/w95f. Wenn Sie den letzten Frost im April abwarten, können Sie sich diesen Schritt sparen.

## Teileselbstbau

Wer wie wir viele Teile selbst baut, kann durchaus einiges an Geld sparen. Der FarmBot besteht größtenteils aus Standardteilen, die Sie bei Händlern für CNC-Bedarf wie RatRig bekommen. Dort bekommen Sie millimetergenau zugeschnittene Aluminium-V-Profile und auch die kugelgelagerten Kunststoffrollen, die in den V-Nuten laufen. Der FarmBot braucht außerdem so viele M5-Linsenkopfschrauben, dass wir den Mindestbestellwert vom Schraubengroßhändler erreichten. Die Liste aller Einzelteile finden Sie in der Dokumentation unter „Bill of Materials“.

Flache 5-mm-Aluplatten haben wir selbst geschnitten und gebohrt. Die Außenmaße müssen hier nicht sonderlich exakt sein, wir waren aber froh, für die Achsenlöcher der Rollen einen Kreuztisch genutzt zu haben, da die obere Lochreihe exakt fluchten sollte.

In der Konstruktion kommen auch abgewinkelte Aluteile vor, die wir als Einzelteile im FarmBot-Shop bestellten, da uns die Maschine zum Abwinkeln fehlte. Im Shop bestellten wir außerdem das Mainboard „Farmduino“ des Genesis 1.5, das Universal Tool Mount (UTM) samt Kabel und die Motoren mit angeflanschten Dreh-Encodern. Die Encoder registrieren optisch, wie weit sich die Motoren drehen, womit die Elektronik erkennen kann, wenn ein

Motor blockiert. Die Kombination aus diesen Nema-17-Motoren und Encodern ist ziemlich unüblich und erscheint auf den ersten Blick teuer. Wir haben den chinesischen Hersteller identifiziert und einen Preis für vier Motoren erfragt: Der FarmBot-Shop bot sie mehr als 100 Dollar billiger an. Vermutlich würden auch Schrittmotoren mit angeflanschten Encodern mit 1000 statt 360 Schritten funktionieren. Die Firmware erlaubt, den Faktor zwischen Schrittzahl des Motors und des Encoders einzustellen. Wir wollten aber nah am Original bleiben und haben die 1000-Schritt-Encoder nicht getestet. Einige Teile wie die Energieketten bot der Shop günstig an, während andere Komponenten sehr teuer waren. Wer wie wir dort Einzelteile einkauft, sollte unbedingt Preise vergleichen.

Wenn Sie einen 3D-Drucker besitzen, können Sie bei den Plastikteilen einiges sparen. Die Teile im Originaldesign sind für die Herstellung in einer Spritzgussmaschine optimiert. Um es beim 3D-Druck einfacher zu haben, haben wir einige Teile äußerlich umkonstruiert. Falls Sie auch drucken wollen, finden Sie unser abgeleitetes Design mit den geänderten Teilen über ct.de/w95f.

Die Riemenbefestigung ist eigentlich als gestanztes Aluteil vorgesehen. Per Hand war uns das zu kompliziert zu fertigen, sodass wir die Teile 3D-gedruckt haben. Bisher scheint die Stabilität völlig auszureichen. Aus Neugier haben wir die Teile später mit der OX-CNC (siehe S. 110) aus Alu gefräst, was hervorragend funktioniert hat. Wer eine CNC-Fräse hat, kann die Aluplatten damit sehr maßhaltig fertigen.

Als wir im Sommer 2020 mit dem Bau begannen, waren im FarmBot-Shop nur das UTM-Kabel und das Motor- und Encoderkabel der Y-Achse lieferbar. Nach den richtigen Steckern, um die Kabel selbst bauen zu können, mussten wir ganz schön suchen: An den Encodern sind Molex-Stecker der Serie 502380 mit neun Pins (Pinabstand 1,25 mm) richtig, Stromanschluss und Peripherie am Farmduino laufen über Molex WM16859-ND. Für den Anschluss der Schrittmotoren haben wir vergeblich nach Dupont-Steckern mit Verschluss gesucht, letztlich haben die üblichen unverschlossenen Stecker mit 2,54 mm Pinabstand funktioniert. Die Kabel selbst zu crimpen war eine ziemliche Fummelei und hat sich wegen horrender Versandkosten für die speziellen Stecker auch finanziell nicht gelohnt. Falls Sie die Kabel fertig kaufen können, empfehlen wir Ihnen, das Geld zu investieren.




**Wer die Aluplatten selbst fertigt, sollte darauf achten, dass die Schraubenlöcher für die Rollen exakt fluchten. Ein Kreuztisch hilft, hier exakt zu arbeiten.**

## Zusammenbau

Mit den Teilen aus dem Bausatz, egal ob selbst gefertigt oder fertig bestellt, ist der Zusammenbau ein Genuss! Die Anleitung führt Sie Schritt für Schritt durch den gesamten Bau und beantwortet nebenbei die meisten Fragen. Die Freude dauert etwa zwei Wochenenden. Sie können das Portal weitgehend drinnen vormontieren. Die Anleitung schickt Sie früher raus als nötig. Drinnen sollten Sie aber genug Platz vorsehen: Selbst unser „kleiner“ Genesis 1.5 hat ein anderthalb Meter langes Portal und die Z-Achse steht über einen Meter hoch nach oben weg!

Beim Verlegen des Wasserschlauchs können Sie sich das Einfädeln in die Energiekette erleichtern, wenn Sie ein 90°-Knie am Übergang zwischen Y- und Z-Achse einbauen. Dadurch muss der Schlauch nicht so ewig lang sein und es besteht an der Ecke keine Gefahr eines Knicks, der den Wasserfluss verschließen könnte. Beim Ende des Schlauchs haben Sie auch Wahlmöglichkeiten: Die Anleitung sieht das Magnetventil am Seitenprofil des Portals vor, wir haben es dagegen an der Seitenwand vom Hochbeet angeschraubt. Uns erschien es sinnvoll, es direkt hinter dem von uns zusätzlich verbauten Absperrhahn zu befestigen. Die 3D-gedruckten Halter dafür finden Sie im Onshape-Design über ct.de/w95f.

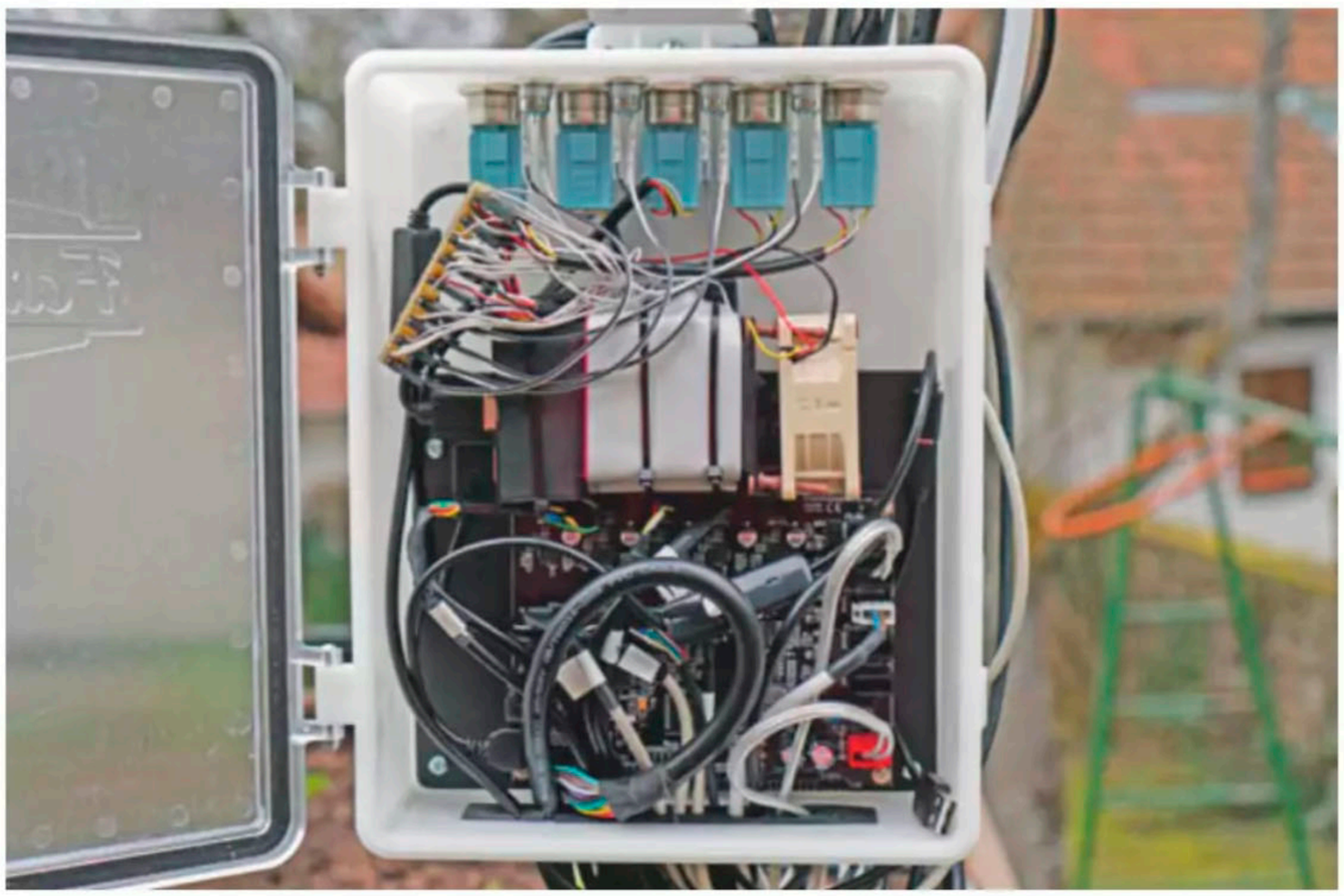
Der UTM-Anschluss auf dem Farmduino-Board funktioniert nur, wenn man rechts davon fünf Jumper für die fünf bei den Standardwerkzeugen verwen-

deten Pins setzt. Ohne diese bleibt das UTM stromlos und erkennt nicht, ob der Werkzeugwechsel geklappt hat. Wer weitere Pins für Erweiterungen nutzen will, findet Anschlüsse für Jumperkabel unterhalb des UTM-Steckers auf dem Board.

In der Elektronikbox ist nicht genug Platz, um darin Schlaufen für zu lange Kabel zu legen. Wir wollten trotzdem Ordnung halten und haben an die Rückseite einen Halter für Kabelbinder geschraubt. So bleiben kleine und große Kabelschlaufen, wo sie sein sollen und verhängen sich nicht in der Mechanik der Achsen.

Die Schienen müssen am Hochbeet völlig parallel zueinander verlaufen. Auch wenn das Portal im Prinzip Scherbewegungen von wenigen Millimetern mitmachen würde, entstünde dadurch zu viel Reibung. Nehmen Sie sich lieber etwas mehr Zeit, um die X-Schienen parallel und eben auszurichten. Der Boden unter unserem Hochbeet hatte sich beim Befüllen auf einer Seite abgesenkt, weshalb wir nicht nach Wasserwaage ausrichten konnten. Stattdessen maßen wir die Diagonalen und die Breite an den Enden und in der Mitte. Unser FarmBot neigt sich jetzt um knapp 2° nach links, zu dieser etwas schrägen Bezugsebene stimmt aber alles.

Beim Aufsetzen des Portals sollten Sie zunächst die Schrauben an der Querverbindung offen lassen, damit die Breite des Portals einstellbar bleibt. Stellen Sie die Rollen dann ungefähr ein und verschieben Sie die Portaloberkante, bis die Wangen orthogonal zum Querprofil ausgerichtet sind. Sobald das fest ist, sollten Sie die Rollen noch mal nachjustieren: Stellt man mit den Exzentermuttern einen Anpressdruck wie bei einer Portafräse ein, sind die beiden X-Motoren zu schwach, um das Portal zu bewegen. Ihr Ziel sollte stattdessen ein sehr leichtgängiger Lauf ohne Klappern oder Wackeln sein. Gleiches gilt für die Riemen, die weniger Spannung haben dürfen, als Sie von 3D-Druckern gewöhnt sind.



**In der Elektronikbox finden oben die Taster und die LEDs Platz, die über die schräge Streifenrasterplatine mit dem Raspberry Pi in der Mitte verbunden sind (hier in einem Gehäuse). Die große Platine unten ist der „Farmduino“ mit den Schrittmotortreibern.**

## Inbetriebnahme

Für die Software beschreiben Sie eine Micro-SD-Karte genau wie beim Raspberry Pi OS. Das FarmBot OS laden Sie dafür von der Webseite herunter (siehe ct.de/w95f) und bespielen die Karte mit Balena Etcher oder einem vergleichbaren Programm. Legen Sie als Nächstes einen Account bei my.farm.bot an, da Sie die Zugangsdaten beim ersten Boot angeben müssen. Danach stecken Sie die Karte in den Raspi in der Elektronik-Box und verbinden den Farmduino mit dem 24V-Anschluss. Der Farmduino versorgt den Raspi 3 über einen USB-Stecker und ein winziges Kabel mit Strom.

Das FarmBot OS spannt bei der Ersteinrichtung ein eigenes WLAN auf, mit dem Sie sich verbinden und über ein einfaches Webinterface die WLAN-Daten angeben. Bei diesem Schritt teilen Sie dem FarmBot auch die Zugangsdaten zu Ihrem Account bei my.farm.bot mit. Das Farmbot OS auf dem Raspi flasht auch automatisch die neueste Firmware auf den Farmduino. Sie müssen nur einstellen, welche Version des Mainboards Sie besitzen (aktuell ist 1.5, ab Juli 2021 soll es 1.6 geben). Danach startet der Raspi neu, verbindet sich mit dem WLAN und taucht gleich darauf auch bei my.farm.bot auf.

Die Konfiguration und Programmierung laufen nun über die offizielle Weboberfläche, die stets mit dem FarmBot synchronisiert. Navigieren Sie zuerst zu den Einstellungen (das Zahnrad-Symbol in der oberen Leiste). Dort können Sie dem Roboter einen Namen geben, die Firmware neu flashen und das System neu starten.

In den Abschnitten „Axes“, „Motoren“ und „Encoders“ können Sie die Motoreinstellungen festlegen. Für den Bausatz sollten die Standardwerte ei-

gentlich stimmen. Testen Sie zuerst für jede Achse mit dem gelben Button „FIND HOME“, ob sie den Nullpunkt automatisch anfahren kann. Wenn das klappt, können Sie den FarmBot danach mit „FIND LENGTH“ automatisch ausprobieren lassen, wie lang die Achse ist. Der FarmBot bewegt die Achse für beide Funktionen, bis sie anstößt und er mit den Encodern registriert, dass der Schrittmotor Schritte verliert. Ein Rattern ist hier also gewollt.

## Motorprobleme

Bei uns klappte die Inbetriebnahme auf Anhieb leider nur bei der Y-Achse. Für die Fehlersuche sollten Sie zuerst die Encoder deaktivieren, da der FarmBot dann Bewegungsversuche nicht frühzeitig abbricht. Testen Sie nun mit dem Bewegungs-Menü (Fadenkreuz-Symbol), ob Sie die Achse 100 Millimeter bewegen können. Mechanische Blockaden sollten Sie mit abgeschalteten Encodern deutlich erkennen können.

Wir brachten zuerst die Z-Achse mit etwas Kettensägenöl in Gang. Kettensägenöl ist im Gegensatz zu den meisten anderen Schmierölen kein Mineralöl, sondern pflanzlich. Sie sollten zwar ohnehin so sparsam ölen, dass nichts ins Beet tropft. Sollte der Regen aber dennoch etwas in den Boden spülen, landen dort keine Giftstoffe.

Die Schrittmotoren sind standardmäßig auf einen Motorstrom von 600 Milliampere eingestellt. Auf dem Farmduino 1.5 sind TMC2209-Treiber verbaut, die unter Dauerlast bis zu 1200 Milliampere liefern können. Beim Vortrieb für Y- und Z-Achse haben wir 900 Milliampere gewählt.

Bei der X-Achse hatten wir aber sogar mit dem Maximalstrom von 1700 Milliampere Probleme. Eine Seite des Portals setzte sich in Bewegung, der Motor der Gegenseite vibrierte nur. Das zog das Portal schräg, was die Reibung erhöhte und die Maschine völlig blockierte. Interessanterweise konnten wir die Seite, auf der das Problem auftrat, umkehren, indem wir die Kabel für X1 und X2 auf dem Board tauschten. Der Support gab uns diverse Tipps, die X-Achse leichtgängiger zu bekommen, ging aber auf wiederholte Nachfrage nicht auf das offensichtliche elektrische Problem ein. Letztlich stellte sich heraus, dass der X1-Treiber bei unserem Farmduino kaputt war. Die Gewährleistung von FarmBot Inc. beschränkt sich auf 30 Tage nach dem Kauf und da wir beim Direktimport aus den USA nicht auf die gesetzlichen zwei Jahre Gewährleistung der EU zurückgreifen konnten, mussten wir mit dem Schaden leben.

Letztlich bekamen wir unseren FarmBot aber doch noch zum Fahren. Die Lösung war ein Y-Kabel wie bei der MPCNC (siehe S. 102), mit dem wir beide X-Motoren am X2-Treiber anschließen konnten. Da wir mit den Tipps vom Support die X-Achse sehr leichtgängig eingestellt hatten, reichen 1200 Milliampere für beide Motoren.



**Beim FarmBot Genesis fahren alle Achsen mit kugelgelagerten Rollen in v-förmigen Nuten der Aluprofile. Den Rollenabstand sollten sie mit den Exzentermuttern so einstellen, dass alle Achsen sich sehr leicht bewegen lassen.**

## Werkzeugwechsel

Sobald der FarmBot fährt, können Sie die „Toolbays“ ans Beet anschrauben. Achten Sie darauf, sie horizontal exakt auszurichten, den Abstand zur Seite und die Höhe können Sie danach einstellen. Wir haben den Abstand zwischen den beiden Haltern so gewählt, dass der Zwischenraum auch ein Werkzeug aufnehmen kann, beispielsweise einen Behälter für die Samen.

Setzen Sie sich anschließend mit einem Notebook neben das Beet und fahren Sie den Kopf des FarmBot exakt zur Position der ersten Toolbay. Die Werkzeuge haben seitlich Nuten, die etwa 1,5 Millimeter Platz zu den Seiten des Halters frei lassen. Mitteln Sie diese Position exakt aus. In X-Richtung sollten Sie mit der Hinterkante der Basis eines Werkzeugs etwa einen Millimeter Platz zum Rand lassen. Das Tool sollte an den Magneten in der Halterung hängen, ohne die Ränder zu berühren. Nur so reicht die Anziehungskraft der Magneten, um das Werkzeug perfekt auszurichten. Bei der Z-Position empfehlen wir, nicht auszumitteln. Wählen Sie hier lieber einen knappen Abstand (etwa einen halben Millimeter) zur oberen Kante der Nut. Das stellt sicher, dass die Magnete das Tool immer bis zum Anschlag anziehen können. Nur in perfekt eingerasteter Position schließt sich der elektrische Kontakt, mit dem der FarmBot prüft, ob das Werkzeug geladen ist (UTM-Pin C ist verbunden mit Pin B, der Masse).

Die so ermittelte Position übernehmen Sie mit dem blauen Button in der Toolbay-Konfiguration als Position. Wiederholen Sie den Prozess mit allen weiteren Toolbays. Wenn Sie sich für diesen Schritt Zeit nehmen, sparen Sie die später mehrfach, weil nur mit sehr exakten Positionswerten auch Werkzeugwechsel klappen.

## Kamera

Die Kamera war eines der Bauteile, die wir nicht im FarmBot-Shop bestellen konnten. Wir haben stattdessen zuerst eine sehr billige USB-Endoskop-Kamera mit gleichen Abmessungen verbaut. Die Bildqualität dieser Kamera konnte uns aber nicht zufriedenstellen. Sie produzierte unscharfe Bilder in niedriger Auflösung, die zum Rand deutlich dunkler wurden. Um systematisch aus mehreren Positionen das ganze Beet abzulichten, ist das ungünstig, da die Foto-Kacheln dann hinsichtlich ihrer Helligkeit nicht aneinanderpassen.



**Die wechselbaren Werkzeuge in den „Toolbays“: Bodenfeuchtesensor, Pflanz-Spitze, Jätwerkzeug, Gießkopf. An den Bodenfeuchtesensor dockt gerade das „Universal Tool Mount“ (UTM) an.**

Die Kamera aus dem Bausatz scheint nicht ganz so unterirdisch in der Bildqualität; im Forum beklagten sich aber dennoch einige Nutzer über unschärfere Bilder als in der Dokumentation.

Wir stiegen wegen der schlechten Bilder auf eine etwa 60 Euro teure Endoskopkamera mit HD-Auflösung und deutlich größerem Kopf um. Den 3D-gedruckten Halter dafür und den Link zum Angebot finden Sie über ct.de/w95f.

Die teurere Kamera ist ein klares Upgrade, da sie schärfere Bilder ohne dunklen Rand liefern kann. Leider fokussiert sie nicht immer richtig und auch beim Weißabgleich kommen im Automatikmodus nicht unbedingt konsistente Bilder heraus. Leider kann man mit dem FarmBot OS nur die Auflösung umschalten, für manuellen Fokus und Weißabgleich gibt es keine Einstellung. Unser momentaner Workaround schießt vor dem Scannen des Beets einige sinnlose Bilder, in der Hoffnung, dass sich die Automatik auf sinnvolle Werte festlegt.

Im Reiter „Fotos“ muss man die Kamera kalibrieren. Beim Bausatz ist dafür eine Kalibrierungskarte dabei, die Doku bietet auch ein PDF zum Selbstdrucken an. Auf der Karte ist ein Punktmuster, das man unter der Kamera positioniert. Der FarmBot fährt bei der Kalibrierung auf drei Positionen, fotografiert und errechnet den Öffnungswinkel und die Drehung der Kamera automatisch. Das klappt allerdings nur, wenn die weißen Punkte auf dem Kamerabild nicht überstrahlen. Selbst unsere HD-Kamera konnte dafür am Tag kein geeignetes Bild liefern. Die Kalibrierung gelang uns erst gegen Ende des Sonnenuntergangs.

Im Kamera-Reiter kann man auch die Bodenhöhe bestimmen lassen. Das klappt nur mit scharfen Fotos – dann allerdings recht exakt. Man kann sie aber in der halben Zeit auch per Hand ausmessen.

## Energiekette und Frost

Die Energiekette für die X-Achse ist sehr lang. Fährt man das Portal auf die am weitesten entfernte Position, hängt sie zu weit durch. Als Unterstützung sind Halter aus Kunststoff vorgesehen, an denen sich bei uns allerdings manchmal Glieder der Energiekette verhakten. Wir haben auf die Halter deswegen eine Kufe aus einer glatten Gardinenstange geschraubt. Die lässt die Kette nun fast reibungsfrei gleiten.

An der Y-Achse hatten wir ein Klemm-Problem Anfang März, als die Beet-Foto-Session zum Sonnenaufgang scheiterte, weil Energiekette und Rollen am Aluprofil festgefroren waren. Schaltet man den Wasseranschluss ab, ist der FarmBot durchaus frostsicher, einen Betrieb im deutschen Winter können wir für die in Kalifornien entwickelte Maschine aber nicht empfehlen.

**Bauanleitung, erweiterte CAD-Designs, Druckvorlagen**

ct.de/w95f

## Zwischenfazit

Für ein umfassendes Bild des Projekts FarmBot sollten Sie unseren Artikel über die Programmierung auf Seite 166 lesen, da die Programmierung ein entscheidender Faktor für die Beurteilung ist. Aufbau und Mechanik sind ein großes Projekt für Bastler, die zumindest schon mal einen 3D-Drucker oder eine CNC-Fräse zusammengebaut haben sollten. Für uns war es das spannendste Projekt des vergangenen Jahres! Die schiere Größe der Maschine, die vielen Schritte vom Beet bis zur Riemenspannung und die durchdachte Konstruktion begeistern für einen langen Zeitraum. Bei Problemen hatten wir aber auch schlaflose Nächte.

Wenn Sie einen FarmBot bauen, werden Sie auf mechanische und elektrische Probleme stoßen und Sie sollten Lust haben, die zu lösen. Die gute Nachricht ist nämlich, dass keine der Komponenten so exotisch oder komplex ist, dass einer mit c't-Projekten erfahrenen Bastlerin nicht Lösungen einfallen würden. Der FarmBot ist keine Raketentechnik, sondern ein Einblick in Maschinenbau, den man auch ohne Studium wagen kann.

Wer sich nur fürs Gärtnern und nicht für Technik interessiert, sollte nur unser Hochbeet bauen und das per Hand bewirtschaften. Im Idealfall fasziniert Sie beides. Dann nämlich ist der FarmBot ein tolles Bastelprojekt mit hervorragender Anleitung, nach dem Sie Gartenbau mit anderen Augen sehen. Ob der FarmBot nach dem Basteln auch Chancen hat, erfolgreich Ihr Gemüse zu umhegen, während Sie zwei Wochen in den Urlaub düsen, verraten wir auf Seite 172. (pmk) c't



**Die Energiekette für die X-Achse kann sich an den roten Haltern verklemmen. Eine Kufe aus einer Gardinenstange sorgt bei unserem FarmBot für reibungsfreies Gleiten.**

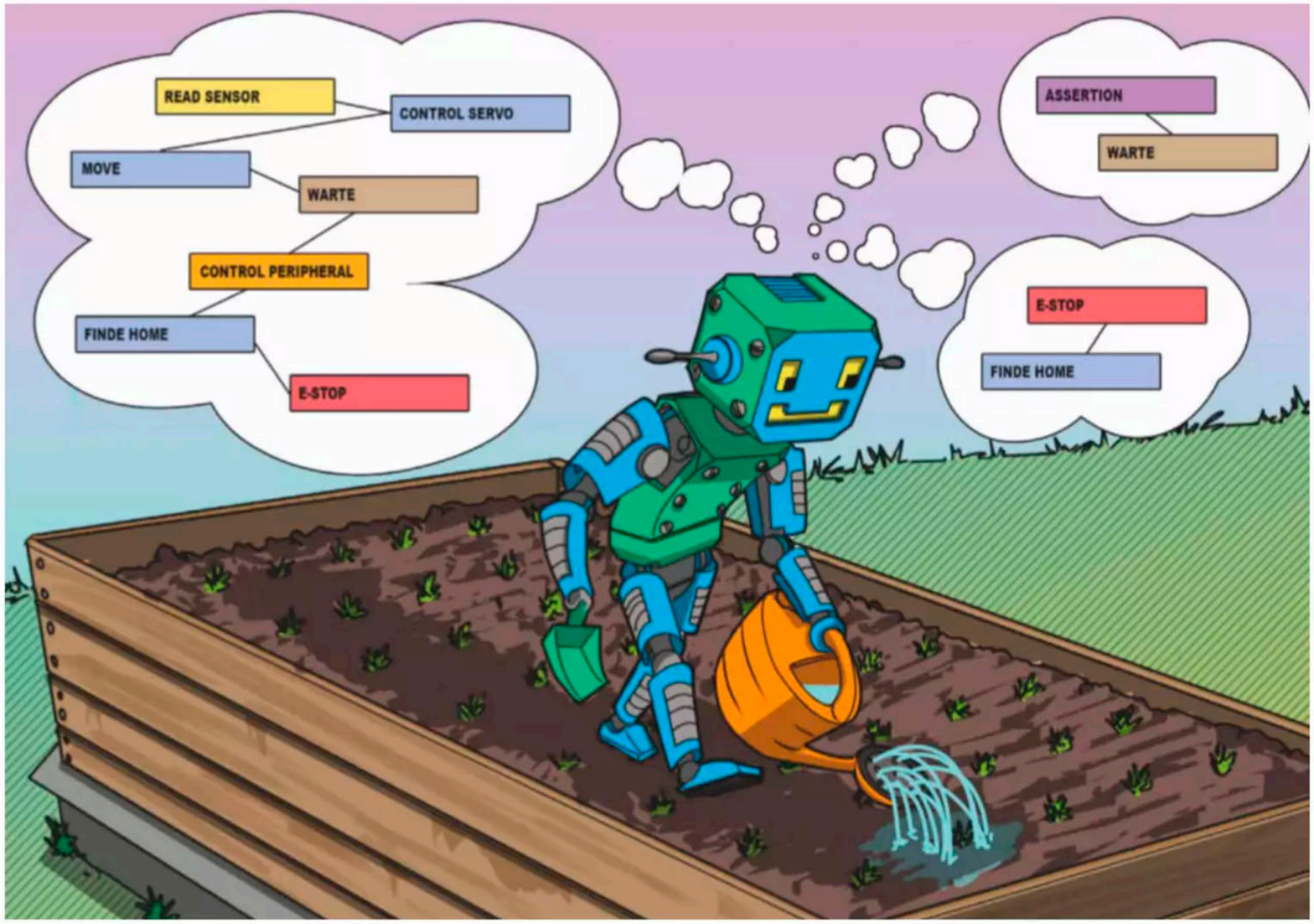



Bild: Thorsten Hübner

# FarmBot programmieren ohne Code

**Bewegen, Messen, Ansaugen und Wasser marsch! Der Roboter-Gärtner FarmBot kann nur wenige Dinge. In der richtigen Reihenfolge wird daraus aber akkurate Gartenarbeit wie Einsäen samt Angießen. Die Schritt-für-Schritt-Anleitungen für den Roboter klickt man wie eine To-do-Liste zusammen.**

Von **Pina Merkert**

Der FarmBot ist der dümmste Gärtner der Welt. Er weiß nämlich nichts über Pflanzen – und von den Intuitionen, die einen „grünen Daumen“ ausmachen, hat er gar keine. Maschinelles Lernen, neuronale Netze: Fehlanzeige! Stattdessen programmiert man ihn entsprechend des eigenen Gärtnerwissens und lässt ihn regelmäßig, konsequent und geduldig das Hochbeet umsorgen (siehe S. 154). „Programmieren“ heißt dabei aber nicht, dass man Code tippen müsste. Im Webinterface des Ro-

boters klickt man stattdessen Kästchen zusammen und lehrt damit die dumme Maschine das Gärtnern – ganz ohne Quelltext.

Der FarmBot folgt damit einem Trend namens „No Code“ [1]. Das Interface zeigt übersichtlich an, aus welchen Blöcken sich das Verhalten des Roboters zusammensetzen kann und niemand muss Befehle auswendig lernen, um sie eintippen zu können. Wer noch nicht programmieren kann, lernt dabei ganz beiläufig die Grundkonzepte von Programmiersprachen. Deswegen ist die FarmBot-Programmierung auch abseits der Gärtnerei und der Robotik interessant. Algorithmisches Denken wird in einer zunehmend digitalisierteren Welt immer nützlicher und wichtiger. Mit einem Account bei my.farm.bot können Sie Sequenzen auch dann zusammenklicken, wenn Sie keinen FarmBot haben. Ohne Roboter können Sie sie nur nicht ausprobieren.

Wer schon Erfahrung im Programmieren hat, findet im FarmBot-Interface zusätzlich einen Block, in den man Lua-Code tippen kann [2]. Die handliche Programmiersprache setzt im Zweifelsfall alle Ideen um, die mit den Blöcken nicht gehen.

## Bausteine des Verhaltens

Programme für den FarmBot nennt die Software „Sequenzen“. Sie bestehen aus einer Liste von pastellfarbig hinterlegten Blöcken. So ein Block definiert ein Verhalten des Roboters. Beispielsweise sorgt ein blauer Bewegungsblock dafür, dass der Roboter zu einer Position fährt. Ein gelber Block fragt einen Sensorwert ab, beispielsweise, ob ein Werkzeug am Werkzeughalter sitzt.

Die Sequenz bringt diese Blöcke in eine Reihenfolge, sodass eine Anleitung für den Roboter entsteht. Den obersten Block geht er zuerst an, den darunter, nachdem die erste Aktion fertig ist. Der Unterschied zu einem Kochrezept liegt in der Exaktheit der Beschreibung: Der FarmBot gießt nicht einen „Schwall Wasser“, sondern er öffnet das Magnetventil des Wasseranschlusses, wartet exakt zwei Sekunden und schließt das Ventil wieder.

## Dort so wie hier

Das Beispiel mit den zwei Sekunden Bewässerung ist eine der Sequenzen, die im FarmBot-Interface schon angelegt sind. Sie beginnt mit einer Bewegung zu der Stelle, die der Roboter gießen soll. Typischerweise nimmt man als Ziel der Bewegung die Position einer Pflanze. Damit man aber nicht eine




**Gießen besteht aus der Bewegung zur Position der Pflanze, Öffnen des Wasserventils, zwei Sekunden Warten und Schließen des Ventils. Dank der externen Variable genügt eine Sequenz für alle Pflanzen.**

eigene Gieß-Sequenz für jede Pflanze zusammenklicken muss, arbeitet die Sequenz mit einer Variablen, einem Platzhalter, der die Position jeder Pflanze enthalten kann. In einer klassischen Programmier-

sprache entspräche die Sequenz einer Funktion und die Variable einem Parameter, der an die Funktion übergeben wird.

Im blauen Block „Bewegung“ gibt es eine Zeile „Ort“ mit einem Auswahlfeld. Sobald Sie dort „Variable – Add new“ auswählen, taucht ganz oben in der Sequenz ein graues Kästchen für die Variable auf. In dessen Überschrift steht „Location Variable“, da der Bewegungsblock den Ort wissen möchte. Diese Überschrift entspricht in einer klassischen Programmiersprache einem Datentyp. Im grauen Kästchen gibt es ein Auswahlfeld für die Position. Sie können dort alle Koordinaten auswählen, die der FarmBot kennt. Damit die Sequenz eine erst später festgelegte Position verarbeitet, müssen Sie aber „Externally defined“ auswählen. Genauso stellt es auch die mitgelieferte Bewässerungs-Sequenz ein.

Wenn Sie die Sequenz auswählen und oben auf den Button „Ausführen“ klicken, fragt das Webinterface mit einem kleinen Fenster, für welche Position Sie die Sequenz ausführen möchten. Damit der FarmBot loslegen kann, darf es nämlich keine Unklarheiten geben, was genau er tun soll. Jede Variable braucht dann einen konkreten Wert. Mit den Parametern einer Funktion in einer klassischen Programmiersprache ist es genauso.

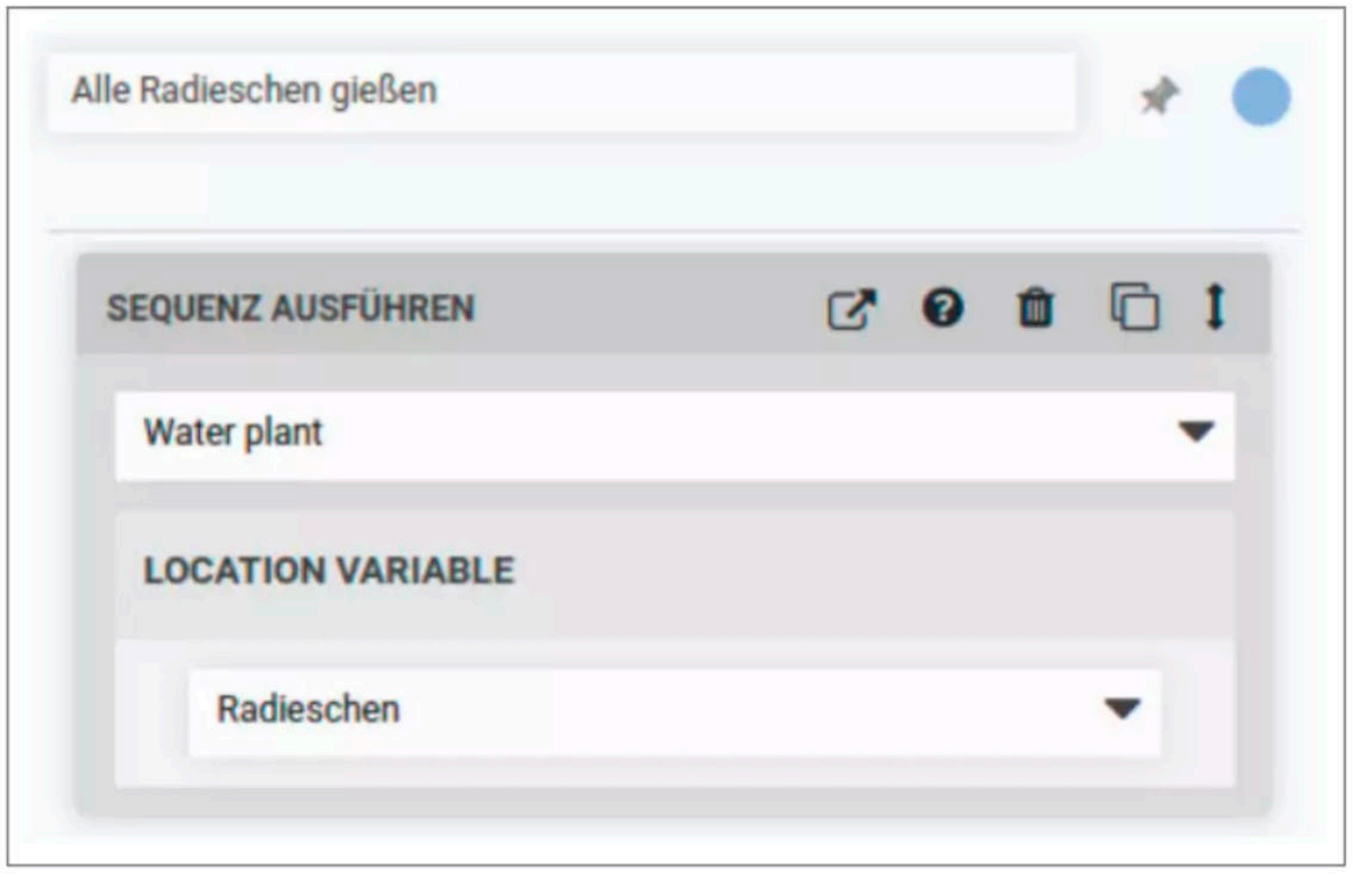
## Alle Radieschen

Wenn Sie im FarmBot-Interface unter „Pflanzen“ (erster Reiter) einige Radieschen gepflanzt haben, können Sie diese mit dem zweiten Reiter in eine Gruppe packen. Die Gruppe dient nicht nur der Übersicht, mit ihr können Sie auch Sequenzen vervielfachen.

Legen Sie dafür eine neue Sequenz „alle Radieschen gießen“ an. Ziehen Sie in diese den grauen Block „Sequenz ausführen“. Im Drop-down-Menü wählen Sie nun zwischen allen bisher erstellten Sequenzen. Wenn Sie nun „Water Plant“ auswählen, erscheint darunter ein grauer Kasten mit der „Location Variable“ der Sequenz. In dem Kasten können Sie alle einzelnen Pflanzen auswählen, am Ende der Liste aber auch die ganze Gruppe.

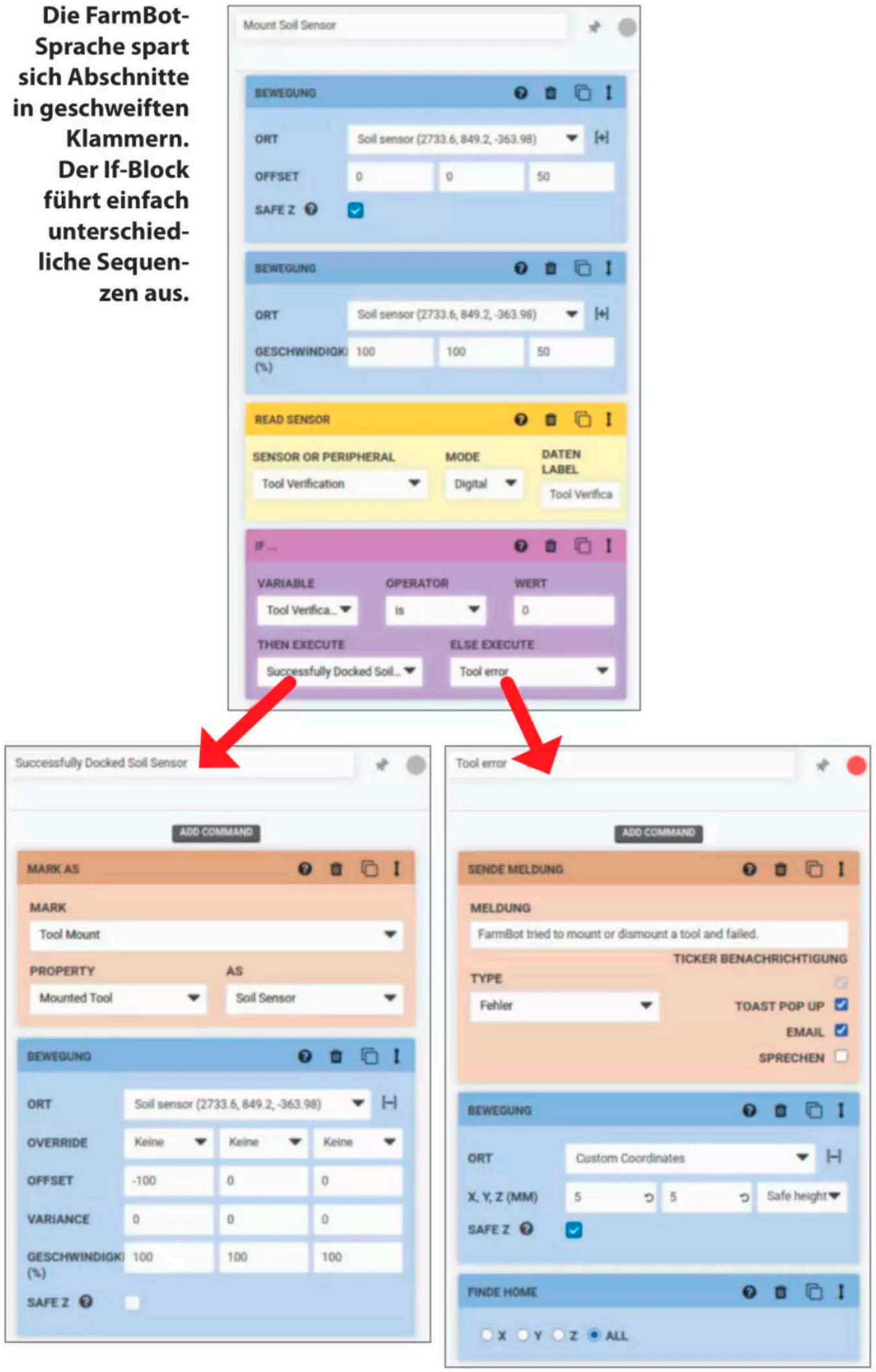
In einer klassischen Programmiersprache wäre das eine Schleife, die eine Funktion für jedes Element einer Liste ausführt. Der FarmBot führt nun nämlich für alle Radieschen aus der Gruppe die Gieß-Sequenz aus und arbeitet sich somit systematisch durchs Beet. Jede Gieß-Sequenz bekommt von der „alle Radieschen gießen“-Sequenz eine Variable, mit der sie arbeiten kann.

Der graue Block „Sequenz ausführen“ akzeptiert auch Sequenzen, in denen so ein grauer Block enthalten ist, sodass Sie mit solchen Sequenz-Sequenzen eine Menge Aktionen in einem Rutsch in Auftrag geben können. Wer hier algorithmisch denkt und die Arbeitsschritte in gleichartige Häppchen aufteilt, braucht wenige Blöcke, um dem FarmBot viel Arbeit zu geben. Dieser braucht ganz schön lang für die vielen einzelnen Schritte, da er aber nicht müde oder nachlässig wird, stört das nicht.




**Befüllt eine Sequenz die Variable einer anderen Sequenz mit einer Gruppe, führt der FarmBot die Sequenz für jede Pflanze in der Gruppe aus.**

**Die FarmBot-Sprache spart sich Abschnitte in geschweiften Klammern. Der If-Block führt einfach unterschiedliche Sequenzen aus.**




## Wenn, dann

Der FarmBot kann sein Verhalten auch an seine „Wahrnehmung“ anpassen. In welchen Schritten das passiert, müssen Sie ihm natürlich auch haargenau mitteilen. Ein gutes Beispiel dafür ist das Abholen eines neuen Werkzeugs. Die Werkzeuge lagern in einer „Toolbay“, wo sie mit ein paar Millimetern Spiel mit seitlichen Nuten in einer Gabel hängen. Zum Andocken muss das Universal Tool Mount (UTM) von oben auf das Werkzeug fahren. Dieses wird magnetisch angehoben und rastet ein. Fährt der FarmBot dann horizontal in X-Richtung von der Toolbay weg, ist das Werkzeug frei, hängt fest an den Magneten und der FarmBot kann es beliebig bewegen.

Die Sequenz zum Andocken fährt also mit einer „Bewegung“ über die Toolbay. Wo die genau ist, definiert eine im System festgelegte Variable. Die letzten fünf Zentimeter fährt das UTM mit halber Geschwindigkeit zur Position, um ganz weich anzudocken.

Im nächsten Schritt soll der FarmBot überprüfen, ob das Tool wie gewünscht an den Magneten hängt. Als Sensor dient dafür Pin C am UTM: Besteht eine Verbindung zu Pin B (Masse), liest der Farmduino, die Steuerplatine des Roboters, eine 0. Die Tools verbinden diese beiden Kontakte stets mit einer Drahtbrücke. Ist das Werkzeug nicht da oder irgendwie verdreht, besteht die elektrische Verbindung nicht und der Farmduino misst eine 1.

Zunächst liest die Sequenz also mit einem gelben Sensorblock, welchen Wert der Farmduino gerade misst. Damit der Wert korrekt zugeordnet wird, muss im Datenlabel „Tool Verification“ stehen. Nun ist die Information zwar da, aber noch nicht ausgewertet.

Darunter sitzt dafür nämlich ein violetter If-Block. Bei dem definiert die Variable, welchen Wert der Block unter die Lupe nimmt (`Tool Verification`). Daneben wählen Sie den Vergleich, beispielsweise „ist größer als“ oder einfach „ist“ für die Prüfung auf einen bestimmten Wert. Die Prüfung soll Erfolg erkennen, wenn sich der Satz ergibt „Tool Verification ist 0“, weshalb bei Wert 0 stehen muss.

In der letzten Zeile im Block unter „THEN EXECUTE“ steht nun im Eingabefeld eine Sequenz, die im Erfolgsfall ausgeführt wird und bei „ELSE EXECUTE“ eine, die ausgeführt wird, wenn die Prüfung fehlschlägt. Die meisten Programmiersprachen würden hier eingerückte Abschnitte mit Anweisungen erlauben, die im jeweiligen Fall abgearbeitet werden. Die FarmBot-Sprache lässt das der Einfachheit halber weg: Statt einer Einrückung legen Sie zwei zusätzliche Sequenzen an. Unter dem If-Block steht deswegen meist nichts mehr. Was dort stünde, würde unabhängig von Ergebnis der Prüfung ganz am Ende ausgeführt.

Beim Aufnehmen des Werkzeugs gibt es eine Erfolgs-Sequenz, die zehn Zentimeter in X-Richtung

wegfährt, um das Tool zu lösen und es mit dem braunen „Mark As“-Block als geladen markiert. Die Fehler-Sequenz fährt den FarmBot zur Nullposition und benachrichtigt den Nutzer mit einem „Sende Meldung“-Block.

## Lua-Block für Coder

Mit Lua-Code-Blöcken kann man Platz im Interface sparen, muss dafür aber Quelltext tippen. Die Dokumentation listet auf, welche Befehle man dort als Lua-Funktionsaufrufe nutzen kann. Bewegen, Sensoren lesen, Fotos machen und Meldungen senden, das geht jeweils mit einer Zeile Quellcode, was gegenüber den gleichwertigen Blöcken Platz auf dem Bildschirm spart. Der Lua-Interpreter kennt auch lokale Variablen, If-Anweisungen mit eingerückten Code-Blöcken und mehrere Arten von Schleifen. Wer Code tippt, kann mit Lua-Blöcken zusätzliche Sequenzen einsparen.

Seltsamerweise kann man aus Lua-Blöcken jedoch keine Sequenzen aufrufen. Die Beschränkung kann man jedoch ein Stück weit mit dem Assertion-Block umgehen. In den tippt man ebenfalls Lua-Code ein, allerdings mit einem `return`, das einen Wahrheitswert zurückliefert. Eigentlich dient das zum Debuggen: Der FarmBot prüft einen Wert und springt in eine Fehler-Sequenz, falls der Wert nicht den Erwartungen entspricht.

Wir haben das zweckentfremdet, um beliebige Werkzeuge korrekt zurückzubringen. An Device-Variablen kommt man nämlich ohne Lua gar nicht ran. Der Befehl `get_device("mounted_tool_id")` gibt aber konsistent die gleiche Kommazahl zurück, wenn ein bestimmtes Tool angedockt und als geladen markiert ist. Wir haben keine Möglichkeit gefunden, mit normalen Blöcken an diese Information zu kommen. Gibt man die Zahl mit `send_message()` einmal für jedes Tool zurück, kennt man die Zuordnung zwischen Zahlen und Tools. Vier Assertion-Blöcke nach dem folgenden Schema reichen dann, um das aktuell geladene Tool in die jeweils richtige Toolbay zurückzubringen:

```
id = get_device("mounted_tool_id")
-- Run sequence if the
-- Watering Nozzle is mounted
tool_id = 14892.0
return not ((id > tool_id - 0.5) and
            (id < tool_id + 0.5))
```

Bei „IF TEST FAILS“ sollte „Recover and continue“ stehen, damit die Sequenz alle vier Prüfungen ausführt; bei „RECOVERY SEQUENCE“ steht die Sequenz zum Abkoppeln des getesteten Tools. Wer If-Anweisungen gewöhnt ist, muss die Logik umdrehen, weil der Block die Sequenz genau dann ausführt, wenn der Test in `return` fehlschlägt.

## Intelligenz des FarmBot

Wie schlau der FarmBot handelt, hängt allein davon ab, wie intelligent Sie die Sequenzen programmieren. Die Entwickler haben leider bisher keine Exportfunktion vorgesehen, mit der die Community Programme tauschen kann. Wir wollen Ihnen unsere Programme aber trotzdem zur Verfügung stellen. Aufgrund der Block-Darstellung nehmen unsere Sequenzen als Beispiel viel Platz weg, weshalb wir sie nicht abdrucken, sondern in Form vieler Bilder zum Download anbieten; die Dateien finden Sie in unserem Repository auf GitHub über ct.de/w4k9. Hier nur eine Übersicht, was unser FarmBot über den Tag so macht:

Mitten in der Nacht startet er seine Bewässerungs-Sequenz. Dafür lädt er zunächst den Bodenfeuchtesensor und misst an einem Punkt zwischen den Pflanzen, wie nass der Boden ist. Hat es geregnet, misst er einen niedrigen Widerstand (Wert über 600), was bedeutet, dass er nicht gießen muss. Zeigt der Messwert einen zu trockenen Boden (Messwert unter 600), fährt er alle Pflanzen ab und gießt sie für zwei Sekunden. Zusätzlich bewässert er auch den Feuchte-Messpunkt für zwei Sekunden. Den testet er erneut, sobald er das Bewässerungswerkzeug zurückgebracht und den Bodenfeuchtesensor wieder geladen hat. Ist es weiter zu trocken, beginnt das Spiel von vorne.

Zum Sonnenaufgang fotografiert der FarmBot das ganze Beet ab und erkennt Unkraut. Zu dieser Tageszeit ist das Beet immer im Schatten und die Kamera bekommt meist relativ gute Bilder hin. Um dem Autofokus Zeit zu verschaffen, schießt sie vorab fünf unnütze Bilder in der Mitte des Beets. Danach geht es systematisch entlang der Punkte eines Gitters, die alle in einer Gruppe stecken.

Im Verlauf des Vormittags jätet der FarmBot dann alle erkannten Unkräuter, die nicht zu nah an einer Nutzpflanze stehen. Dafür holt er das Jät-Werkzeug und stochert fünfmal im Boden: Einmal direkt dort, wo das Unkraut sitzt und zur Sicherheit noch mal um je 15 Millimeter versetzt in alle vier Himmelsrichtungen. Das so zerstocherte Unkraut markiert er als „removed“.

Unsere HD-Endoskopkamera macht im Halbdunkeln bessere Bilder als im Tageslicht, weshalb es

### Literatur


[1] Jo Bager, **Klick, klick – Programm**, Low Code und No Code: Möglichkeiten und Grenzen, c't 8/2021, S. 54

[2] Tim Schürmann, **Auf der Spur der Zeit**, Einstieg in die Lua-Programmierung, c't 5/2021, S. 154


**Alle Sequenzen bei GitHub, Dokumentation, Instagram-Account unseres FarmBot**

ct.de/w4k9

erst eine Viertelstunde vor Sonnenuntergang Zeit für die nächste Fotosession wird. Die ist aber eher zum Spaß: Der FarmBot macht sein Licht an und knipst ein Foto von jeder Pflanze, um später eine Zeitrafferaufnahme von deren Wachstum zusammenschneiden zu können.

Am dunklen Abend steht dann noch mal eine Runde Gießen nach dem bereits beschriebenen Schema an: Messen und Bewässern, wenn der Sensor das verlangt. Damit die Gießerei bei Messfehlern nicht zur Endlosschleife wird, bricht unsere Sequenz übrigens nach fünf Durchgängen ab. Mehr als zweimal zehn Sekunden gezielte Bewässerung braucht keine Pflanze im deutschen Sommer.

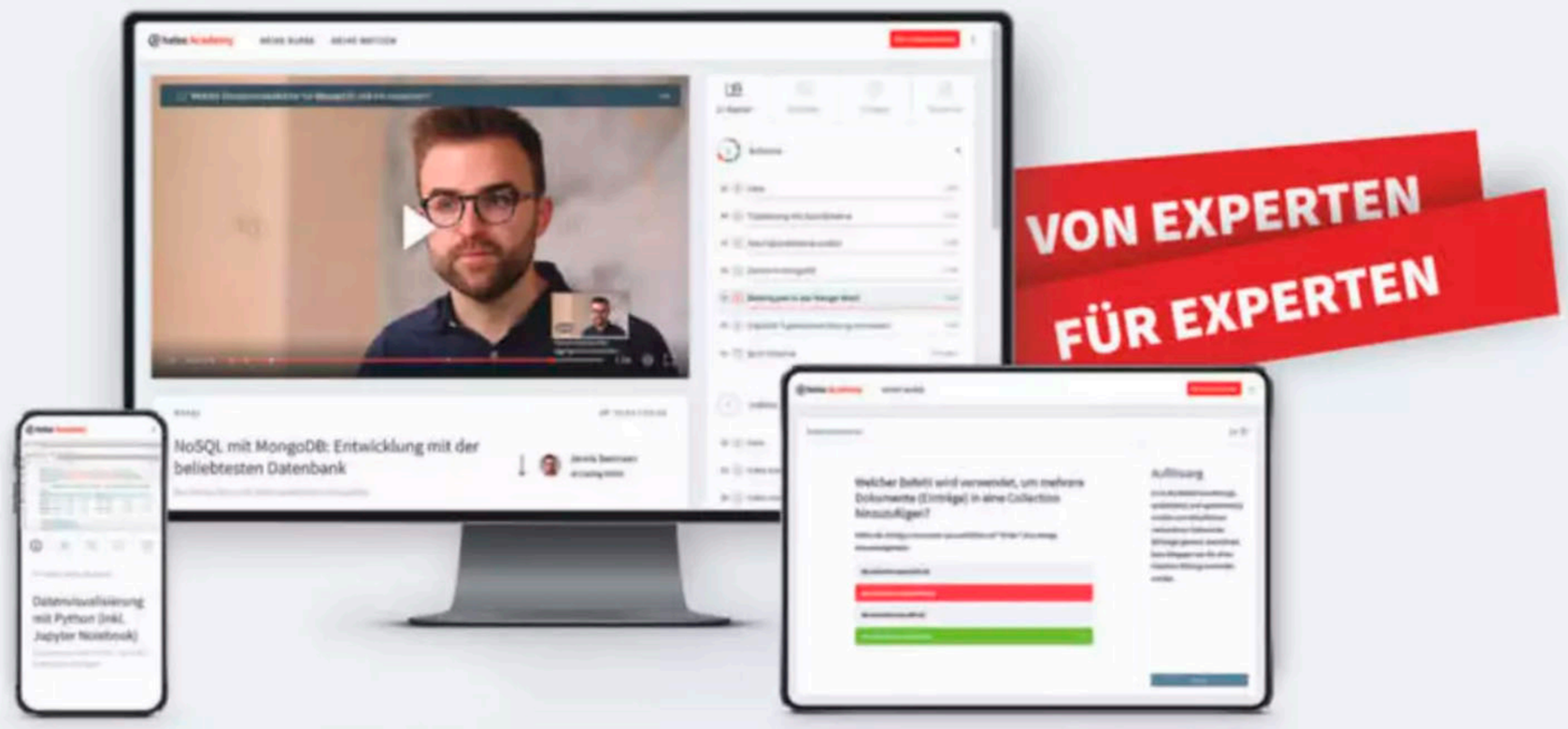
## KI in die Cloud

In früheren Versionen unterstützte der FarmBot auch Plug-ins unter der Bezeichnung „Farmware“. Die Entwickler hatten damit beispielsweise die Unkrauterkennung umgesetzt. Anscheinend hat sich das System aber nicht bewährt. Farmwares sind abgekündigt und die Unkrauterkennung ist nun ein integriertes Modul, das das FarmBot OS beim Booten lädt.

Wer dem Robo-Gärtner doch mehr Intelligenz beibringen möchte als mit den Sequenzen möglich, muss wie in der Künstlichen Intelligenz üblich in die Cloud auslagern. Sequenzen können nämlich HTTP-Requests an beliebige URLs absetzen. Wer also 3D-Bilder vom Beet zusammensetzen oder die Blattfarbe mit einem neuronalen Netz einschätzen möchte, exportiert die Daten, rechnet die Ergebnisse auf einem Rechner im Heimnetz oder Internet und ruft am Ende die Resultate wieder per HTTP ab. Die Dokumentation nennt zwar nur wenige konkrete Beispiele, die API des FarmBot ist aber umfassend dokumentiert. (pmk) c't


heise Academy

Die digitale Lernplattform **für IT-Professionals**




Praxisnahe Online-Kurse zu den wichtigsten IT-Themen – Kubernetes, Data Science, JavaScript, C++, NoSQL u.v.m.

**Jetzt GRATIS-LEKTIONEN ansehen unter www.heise-academy.de**



# Der FarmBot im Praxiseinsatz

**Geduldiger Helfer oder ewiger technischer Patient? Der FarmBot will den Gartenbau revolutionieren, kann aber klemmen, verdrecken, festfrieren und abstürzen. Nach ein paar Wochen im Regelbetrieb beantworten wir die Frage, ob die Maschine mehr nützt, als sie Arbeit macht.**

Von **Pina Merkert**

Auf dem Tisch neben dem Hochbeet stehen Pflänzchen und ein mit lehmigen Fingern eingesautes Notebook. Die Baumärkte haben gerade aufgemacht und Mutter und Tochter nutzen die Gelegenheit, so früh wie möglich das Gemüsebeet zu bepflanzen. Der FarmBot stochert fleißig, aber langsam Löcher in den Boden, in die wir die vorgezogenen Salate einpflanzen.

Doch dann verdunkelt sich der Himmel und ein Schauer ergießt sich über den Garten. Wir retten uns und das Notebook ins Trockene – der Roboter arbeitet geduldig weiter. Als der Schauer nach einer halben

Stunde abebbt, hat er trotzdem nur bis zur Mitte des Beets Löcher gedrückt. Er ist dafür konstruiert, am Tag 24 Stunden Zeit zu haben – für flottes Arbeiten ist er nicht gedacht. Also holen wir schnell auf und schludern: Ein Klumpen nasse Erde fällt auf die Aluschiene. Prompt überrollt ihn die X-Achse und blockiert. Der Encoder am rechten X-Motor meldet den Schrittverlust und die Sequenz bricht ab. Wir pulen den Schlamm aus der Schiene, putzen Riemen und Rollen, starten die Sequenz neu und fragen uns, ob ohne Roboter nicht alles schneller gegangen wäre.

Nachdem wir einen FarmBot gebaut haben (siehe S. 154 und S. 158) wollten wir wissen, wie viel Arbeit uns die Maschine beim Gemüseanbau abnehmen kann. Können wir beispielsweise ein paar Wochen in Urlaub fahren, ohne dass das Unkraut wuchert oder der Kohlrabi vertrocknet? Wie gut überlebt die Maschine Hagel, Frost, Starkregen und Wind? Wann klemmt die Mechanik und wie oft versagt die Software?

## Was er kann

Zum Einsäen besitzt der FarmBot ein Werkzeug, das Samen mit Unterdruck am Ende einer Nadel ansaugen und ein paar Zentimeter tief im Boden absetzen kann. Die Samen lagert er in offenen Behältern, was bedeutet, dass der Prozess nur teilautomatisch abläuft. Man füllt die Samen per Hand in die Behälter, kann dann ein paar Minuten abhauen, während der FarmBot die Samen platziert, muss aber am gleichen Tag wiederkommen und die restlichen Samen einsammeln. Ein Nachsäen der nächsten Reihe Radieschen während des Urlaubs ist nicht möglich, weil es in die Behälter regnen würde.

Für einen frühen Start im März mit im Gewächshaus vorgezogenen Pflänzchen haben wir ein Pikierwerkzeug als Alternative zur Sänadel entwickelt. Es drückt mit einer breiten Spitze ein Loch in die Erde, in das man per Hand die vorgezogenen Pflänzchen setzt. Damit weiß der FarmBot, wo genau die Pflanzen sitzen und der Mensch kann die FarmBot-Software als Gartenplaner nutzen. Automatisch geht hier also nichts, das Einpflanzen ist eher Teamarbeit zwischen Mensch und Roboter.

Gießen kann der FarmBot dank Wasseranschluss vollautomatisch. Da er exakt weiß, wo die Pflanzen stehen, geht das mit einem Minimum an Wasser. Mit dem Bodenfeuchte-Sensor des Genesis kann er vorher sogar prüfen, ob es dem Boden an Wasser mangelt, und nach einem Regenguss die Bewässerung aussetzen. Was im deutschen Sommer für moderate Ersparnisse an der Wasseruhr sorgt, ist in trockeneren Teilen der Welt ein bitter nötiges Sparprogramm für die kostbare Ressource Wasser.

Zum Jäten macht der FarmBot Fotos vom ganzen Beet. In denen sucht er nach grünen Pflanzen und versucht mit Messern aus Kunststoff alles Grün unter die Erde zu drücken, was nicht dort wächst, wo die geplanten Pflanzen sitzen. Vom Blumenkohl und den Kopfsalaten hält er Sicherheitsabstand, weshalb bei einem eng bepflanzten Beet der größte Teil des automatisch erkannten Unkrauts von Hand gejätet werden muss. Fast immer markiert die Automatik jedoch fälschlicherweise Stöckchen oder Nussschalen als Unkraut.

Eine nette Spielerei sind Zeitrafferaufnahmen der Pflanzen beim Wachsen. Dafür macht der FarmBot jeden Tag ein Foto. Ein Skript am Rechner macht aus den Einzelbildern nach der Ernte Videos.

Erkennt Mensch während des mehrwöchigen Urlaubs in den Fotos ein Problem im Beet, kann er die Freunde und Verwandte vorbeischicken, die er in den Jahren zuvor zum Gießen engagiert hat. Automatisch ist daran nichts, aber zumindest müssen die Gartenhelfer nicht nur zum Nachschauen vorbei kommen.



**Den FarmBot besiedeln innerhalb weniger Tage diverse Lebewesen. Meist ist das ein friedliches Nebeneinander; nur einmal blockierte ein überfahrener Käfer eine der Achsen des Robo-Gärtners.**

## Verklemmt und vermessen

Da wir mit unserem Hochbeet früh loslegen wollten, haben wir die Sä-Funktion gar nicht getestet und sämtliche Pflänzchen mit dem Pikierwerkzeug



**Dank einer nachts übers Beet gespannten Folie erfrieren im März und April keine Pflänzchen. Damit sich der FamBot bewegen kann, muss man die Folie aber zuerst entfernen.**

gesetzt. Das hat länger gedauert, als die Reihen im Hochbeet ohne Roboter zu bestücken, der Gartenplan aus dem Webinterface war aber beim Einkauf im Gartencenter eine nützliche Hilfe. Ohne FarmBot hätten wir vermutlich mit einer App geplant.

Damit die kleinen Pflänzchen nicht erfrieren, haben wir das Beet nachts mit einer Folie abgedeckt, die jegliche Bewegungen des FarmBot blockiert. Im Nothalt-Modus konnte er dann zum Sonnenaufgang das Beet nicht abfotografieren, weshalb wir den Unkraut-Scan in den Sonnenuntergang verlegt haben. Danach mussten wir im Halbdunkeln raus und das Beet wieder abdecken.

Leider unterstützt der Farmduino 1.5 bisher nur den Vollschritt-Betrieb der Schrittmotortreiber TMC2130. Die könnten eigentlich auch den leisen Modus StealthChop, stattdessen nervt der Genesis 1.5 mit Schrittmotor-Singsang in Zimmerlautstärke – laut genug, um es im Haus hinter eine Scheibe noch zu hören.

Beim Abfotografieren fokussierte unsere HD-Kamera nicht immer richtig und auch die Belichtung war trotz idealer Tageszeit nicht konsistent. Unkräuter erkannte die Automatik dennoch in den meisten Fällen, die Anzahl an Falsch-Positiven war aber extrem hoch. Die kosten den Roboter nur Zeit, wovon er über den Tag reichlich hat. Die Markierung der nicht automatisch jätbaren Unkräuter im Beet empfanden wir nicht als große Hilfe. Meist drückten wir einfach den Knopf, um alle ungejäteten Unkräuter zu löschen und zupften am Beet einfach, was uns ins Auge fiel.

Unser FarmBot hat lehmigen Boden im Beet, der ruckzuck zwischen den Messern der Jätspitze kleben bleibt. Passiert das, drückt die nur noch Vertiefungen ins Beet, die die Unkräuter nicht stören. Dringen die Spitzen dennoch in nassen Lehmboden ein, reicht die Anziehungskraft der Magnete nicht mehr zum Herausziehen und das Jätwerkzeug bleibt stecken, bis ein Mensch es herauszieht. Wir haben zum



**Die Rillen der Aluprofile müssen unbedingt sauber bleiben. Schon kleine Hindernisse können die X-Achse des Roboters blockieren.**

Reduzieren der Reibung alle Messer bis auf eines abgeschraubt. Danach reichte die Magnetkraft zwar, das Werkzeug jätete aber sehr langsam. Bei bereits tief eingedrückten Unkräutern funktionierte das Jäten gar nicht mehr.

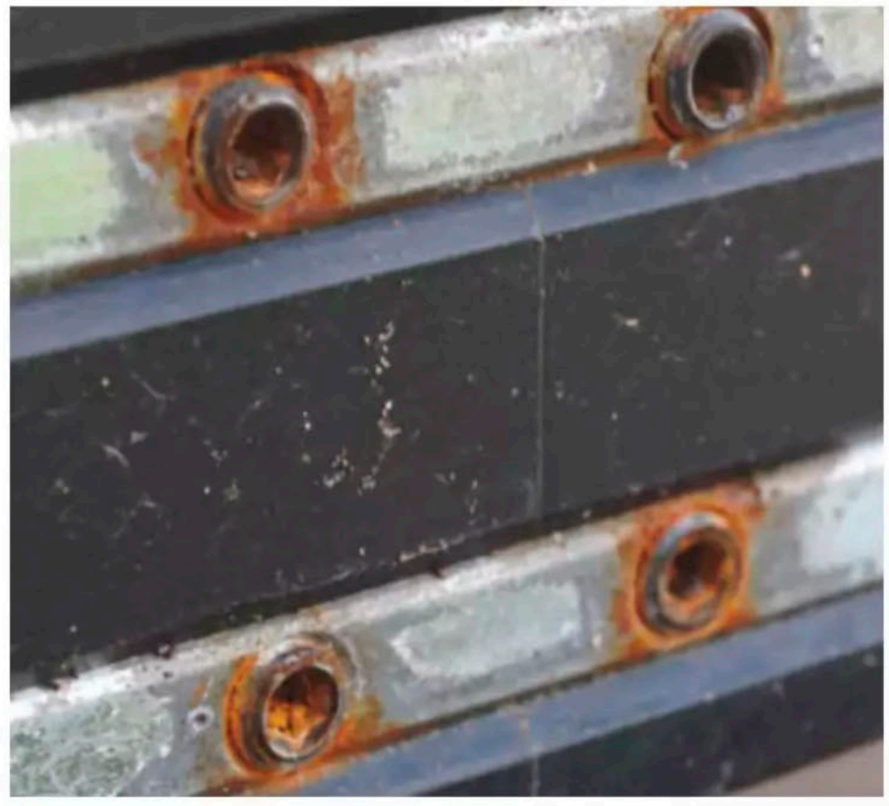
Die Messungen der Bodenfeuchte vor dem Gießen schienen großen Schwankungen unterworfen zu sein. Neben der mangelhaften Genauigkeit des Sensors machte uns wiederum der Lehmboden zu schaffen, da hin und wieder ein Erdklumpen zwischen den Gabeln des Sensors hing. Statt der Feuchtigkeit des Bodens misst der Sensor dann nämlich die des Klumpens. Wir waren daher froh, die Gieß-Schleife (Messen – Wässern) auf fünf Durchläufe pro Tag begrenzt zu haben. Trotz aller Probleme funktionierte die Bewässerung gut genug. Wir hätten für mehrere Wochen wegfahren können, ohne dass die Gemüsepflanzen verdorrt wären.

Mechanische Fehler traten dennoch häufiger auf: Einmal scheiterte die Unkrauterkennung, weil die Y-Achse festgefroren war. Ein anderes Mal brach die Jät-Sequenz ab, weil der Jätkopf auf dem Beetboden aufsetzte, was der FarmBot als verklemmte Z-Achse interpretierte. Die Werkzeugwechsel schlugen mehrfach fehl, weil die Werkzeuge nach Wind und Regen zu fest in den Toolbays saßen. Nach einem Wetterumschwung mussten wir mehrfach die Positionen der Toolbays neu kalibieren. All diese Probleme ließen sich aber aus der Ferne lösen. Wir fuhren den FarmBot in die Nähe des Problems, ließen ihn Fotos machen, passten Parameter und Positionen an und probierten die Sequenzen immer wieder aus. Gießt und jätet man mehrfach am Tag, ist es nicht schlimm, wenn eines der Programme auch mal ausfällt. Auch wenn wir meist nicht vor Ort sein mussten, funktionierte der FarmBot bei uns höchstens ein paar Tage ohne Zwischenfall. Man muss im Webinterface regelmäßig prüfen, ob er noch richtig läuft.

Der einzige Fehler im Test, der sich nur mit Hilfe eines Menschen vor Ort lösen ließ, war ein Werkzeug, das aus der Toolbay gekippt war. Und das, obwohl es während unseres Testzeitraums mehrere Frühlingsstürme, Starkregen und Hagel gab.

## Softwaremängel

Mehrfach traten Probleme in der Software auf: Die WLAN-Verbindung hakte manchmal für einige Sekunden; zweimal zeigte die Weboberfläche keine neuen Fotos mehr an; einmal ging die Verbindung zwischen Raspi und Farmduino verloren. Ein simpler Neustart ließ das Problem jeweils verschwinden.



**An einigen Schrauben zeigt sich nach einem Winter mit Wind, Schnee und Regen etwas Rost. Das war aber an keiner Stelle ein Problem.**

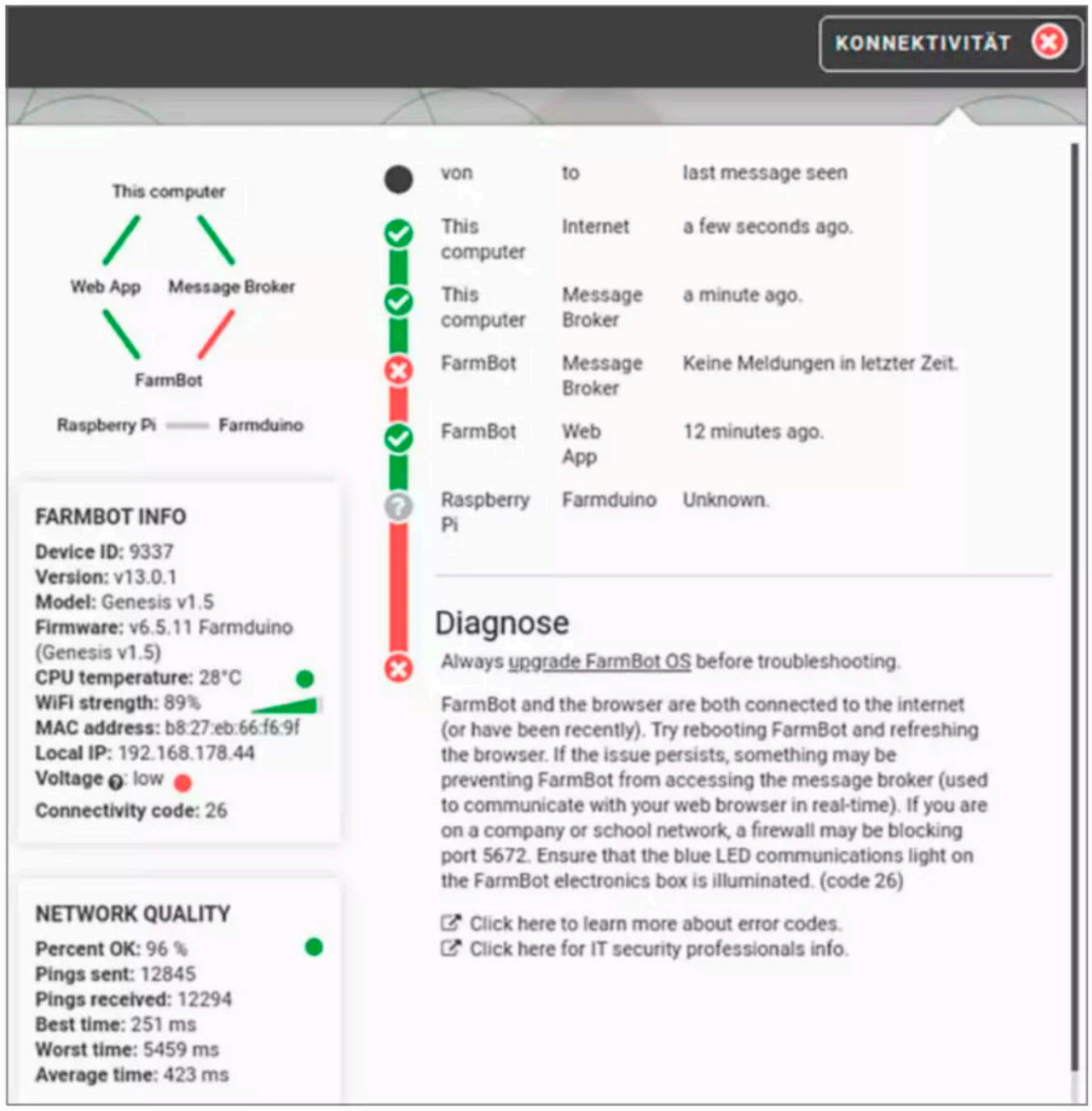
Wir überlegten, deswegen eine Sequenz anzulegen, die den FarmBot einmal jede Nacht neu bootet.

Unser FarmBot stürzte einige Male ohne erkennbaren Grund ab, woraufhin er automatisch neu startete. Meist erkennt man den Absturz nur am Log-Eintrag zum erfolgreichen Hochfahren. Bei der Stabilität der Software müssen die Entwickler dringend nachbessern.

Der FarmBot ist in erster Linie ein umfangreiches Bastelprojekt für gärtnernde Technikfans. Sowohl der Aufbau als auch der Betrieb machen nur dann Freude, wenn man die Technik versteht und vor der Programmierung keine Angst hat.

Wenn die Maschine mal steht, funktioniert sie nicht zu 100 Prozent fehlerfrei, aber gut genug, um Arbeit abzunehmen. Ähnlich wie ein Staubsauger-Roboter nicht in jede Ecke kommt, muss man beim FarmBot immer mal wieder per Hand ans Hochbeet. Der FarmBot nimmt einem nicht ab, sich damit zu beschäftigen, wie man Gemüse richtig anpflanzt und pflegt. Die Maschine arbeitet aber geduldig jede Sequenz ab, die man programmiert – auch zu unmenschlichen Tages- und Nachtzeiten. Sie kennt aber weder Augenmaß noch grünen Daumen. Umgraben, Düngen und Ernten muss man in jedem Fall per Hand.

Gut eingestellt und getestet arbeitet der FarmBot verlässlich genug, um ihm während des Urlaubs das Beet zu überlassen. Bei der Rückkehr muss man




**Verliert der FarmBot seine Netzwerkverbindung, geht nichts mehr. Wir mussten für verlässliches WLAN im Garten einen Outdoor-Repeater installieren.**

zwar ein paar Unkräuter jäten, aber weniger als ohne Roboter. Die Pflanzen verdorren auch dann nicht, wenn es längere Zeit nicht regnet. Mit den Fotos hat man vom Strand aus im Blick, was daheim im Hochbeet passiert. Ein bisschen fühlt man sich dabei aber wie das Nasa-Team, das den Mars-Rover steuert: Alles geht langsam und zwischen den Aktionen stehen Wartezeiten für die Datenübertragung.

Ein automatischer Gemüsebauer ist der FarmBot nicht; es fällt weiter einiges an Handarbeit an. Er ist ein etwas dümmlicher Hilfsarbeiter, der nach exakter Anleitung geduldig einfache Aufgaben übernimmt. Außerdem reicht ein Hochbeet von 1,5 × 3 Meter nicht mal für den Gemüsebedarf einer Person. Die Behauptung der Entwickler, man könnte mit dem XL mit vierfacher Fläche eine Familie versorgen, stimmt nicht. Wir empfehlen, nur die Gemüse und Kräuter mit dem FarmBot anzubauen, die frisch geerntet die meiste Freude bereiten. Bei uns wachsen unter anderem Pflücksalat, Radieschen, Kohlrabi, Blumenkohl, Rosmarin und Petersilie. So früh im Jahr sind wir bisher sehr zufrieden, wie kräftig unsere vorgezogenen Pflänzchen gewachsen sind. Kartoffeln oder Karotten sind dagegen so hochwertig und billig von Bauern zu bekommen, dass uns die Beetfläche dafür zu schade war.

## Fazit

Alles in allem nützt der FarmBot im Garten mehr, als er zusätzliche mechanische Probleme bringt – grob geschätzt drei zu eins bei der Zeit. Der Aufbau mit Wasser- und Stromanschluss, Hochbeet und Installation frisst aber vorher immens Zeit. Zudem ist der FarmBot mit mindestens 3000 Euro ein ziemlich teures Projekt. Wenn Sie die Zeit zum Basteln genießen, kann sich das lohnen, aber für die Kosten von Hochbeet und Bausatz können Sie auch einen menschlichen Gärtner für viele Stunden bezahlen – und der schneidet nebenbei auch die Obstbäume. (pmk) c't

**Links und Videos**

ct.de/ws8r

# Mit dem Raspi besser bewässern

**Der OpenSprinkler-Aufsatz macht den Raspberry Pi zur Bewässerungssteuerung für Garten und Balkon. Dank Open-Source-Software und großer Community wird der Garten zum geeigneten Sommer-Projekt für Hausautomatisierer.**





Von **Jan Mahn**

Der FarmBot, den wir ab Seite 154 ausführlich vorstellen, ist ein Projekt für alle, die im Garten nichts mehr dem Zufall und der eigenen Handarbeit überlassen wollen. Wer mit etwas weniger Aufwand ins computergesteuerte Gärtnern einsteigen will, kann zunächst die Gartenbewässerung mit OpenSprinkler und einem Raspi automatisieren.

Bis zu acht Bewässerungskreise, also Magnetventile, steuert die Platine, die man als Hat auf den Raspi aufsteckt. Die Ersteinrichtung ist klassisches Raspi-Handwerk: MicroSD-Karte mit Raspberry Pi OS bespielen und OpenSprinkler auf der Kommandozeile herunterladen und starten. Als Garten-Admin muss man sich dann darum kümmern, dass der Raspi auch draußen eine Verbindung zum Netzwerk hat – per WLAN oder Kabel. Anschließend steuert man die Bewässerung über eine lokale Weboberfläche. Sie ist recht intuitiv gestaltet, man kann Zonen manuell starten, Wochenpläne für die Bewässerung zusammenstellen und sehr viele Details im Menü konfigurieren.

Damit nicht mehr Wasser als nötig im Garten verteilt wird, kann man gleich mehrere Datenquellen zur Optimierung heranziehen: OpenSprinkler bezieht auf Wunsch Wetterdaten für den eingetragenen Standort von einem API und fragt einen Bodenfeuchtesensor, der am Sensor-Port Anschluss findet, nach dem Zustand des Beets. Aus den Informationen errechnet die Software die benötigte Wassermenge. Das Forum zur Software ist eine Fundgrube voll mit Optimierungsideen für das eigene Bewässerungskonzept.

Mit dem Aufstecken der Platine auf den Raspi ist die Bewässerungslösung natürlich nicht fertig, die echte Arbeit beginnt im Garten: Für jeden Bewässerungskreis braucht man ein 24-Volt-Magnetventil, für das man etwa 25 Euro einplanen muss. Dann geht es an die Verkabelung und Verrohrung der Ventile.

OpenSprinkler Pi wird mit einem Acryl-Gehäuse geliefert, das leider nicht ganz durchdacht ist. Um an die SD-Karte zu kommen, muss man den Deckel abbauen. Wasserdicht ist es bewusst nicht konzipiert, schließlich müssen Sie für eine Außeninstallation noch mehr als den Raspi vor Regen schützen. Passende Kleinverteilerkästen, in denen das Netzteil, der Raspi und gegebenenfalls ein WLAN-Access-Point Platz finden, gibt es in der Elektro-Abteilung in jedem Baumarkt.

OpenSprinkler ist nicht der schnellste Weg zur automatischen Bewässerung, aber flexibler als Lösungen der großen Garten-Hersteller. Smart-Home-Bastler können ihr Hobby im Sommer damit gut in den Garten verlagern. In der Beta-Version der Software wird gerade eine MQTT-Anbindung erprobt, mit der man die Gartensteuerung mit einer Vielzahl offener Hausautomationen wie ioBroker oder Node-Red verknüpfen kann. (jam) c't

## OpenSprinkler Pi (OSPi)

| Bewässerungssteuerung für den Raspberry Pi | |
|---|---|
| Bewässerungskreise | max. 8 |
| Spannungsversorgung | 24 V Wechselspannung (und Raspi-Netzteil) |
| benötigter Raspberry Pi | Raspi 2 oder 3 empfohlen |
| Anbieter | https://opensprinkler.com |
| Preis | 119 € (Platine, Gehäuse, ohne Raspi) |

# IMPRESSUM


## Redaktion

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.ct-special.de

**Leserbriefe und Fragen zum Heft:** sonderhefte@ct.de

Die E-Mail-Adressen der Redakteure haben die Form xx@ct.de oder xxx@ct.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:** Jobst-H. Kehrhahn (keh)
(verantwortlich für den Textteil)

**Konzeption:** Pina Merkert (pmk)

**Koordination:** Ann-Sophie Eikermann (ase),
Angela Meyer (anm)

**Redaktion:** Mirko Dölle (mid), Lutz Labs (ll), Jan Mahn (jam), Pina Merkert (pmk), Angela Meyer (anm), Christof Windeck (ciw)

**Mitarbeiter:** Florian Lapport, Merlin Schumacher, Christoph Schmith

**Assistenz:** Susanne Cölle (suc), Tim Rittmeier (tir), Christopher Tränkmann (cht), Martin Triadan (mat)

**DTP-Produktion:** Lara Bögner, Beatrix Dedek, Madlen Grunert, Lisa Hemmerling, Sarah Hiller, Natalie Hollstein, Kirsten Last, Steffi Martens, Marei Stade, Matthias Timm, Ricardo Ulbricht, Ninett Wagner

**Fotografie:** Andreas Wodrich, Melissa Ramson

**Produktion:** Christine Kreye (Ltg.),
Melanie Becker, Anna Hager, Kevin Harte, Martin Kreft, Pascal Wissner

**Illustration:** Thorsten Hübner, Berlin; Albert Hulm, Berlin; Andreas Martini, Wettin

**Titel:** Steffi Martens, www.flaticon.com, www.freepik.com, www.stock.adobe.com

## Verlag

Heise Medien GmbH & Co. KG
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.heise.de

**Herausgeber:** Christian Heise, Ansgar Heise, Christian Persson

**Geschäftsführer:** Ansgar Heise, Dr. Alfons Schräder

**Mitglieder der Geschäftsleitung:** Beate Gerold, Jörg Mühle

**Verlagsleiter:** Dr. Alfons Schräder

**Anzeigenleitung:** Michael Hanke (-167)
(verantwortlich für den Anzeigenteil),
www.heise.de/mediadaten/ct

**Anzeigenverkauf:** Verlagsbüro ID GmbH & Co. KG,
Tel.: 05 11/61 65 95-0, www.verlagsbuero-id.de

**Leiter Vertrieb und Marketing:** André Lux (-299)

**Service Sonderdrucke:** Julia Conrades (-156)

**Druck:** Firmengruppe APPL Druck GmbH & Co. KG,
Senefelder Str. 3–11, 86650 Wemding

**Vertrieb Einzelverkauf:**
VU Verlagsunion KG
Meßberg 1
20086 Hamburg
Tel.: 040/3019 1800, Fax: 040/3019 145 1800
E-Mail: info@verlagsunion.de

**Einzelpreis:** € 14,90; Schweiz CHF 27.90;
Österreich € 16,40; Luxemburg € 17,10

**Erstverkaufstag:** 25.05.2021



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, vervielfaltigt oder verbreitet werden. Die Nutzung der Programme, Schaltpläne und gedruckten Schaltungen ist nur zum Zweck der Fortbildung und zum persönlichen Gebrauch des Lesers gestattet.

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 c't erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes.

Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.



Hergestellt und produziert mit Xpublisher:
www.xpublisher.com

Printed in Germany.



Alle Rechte vorbehalten.

© Copyright 2021 by
Heise Medien GmbH & Co. KG

# Aufrüsten im Homeoffice

NEU




Heft + PDF mit 29 % Rabatt




## c't Hardware fürs Homeoffice 2021

Die Arbeit im Homeoffice gelingt am besten, wenn nicht nur die richtige Software zur Verfügung steht, sondern auch die passende Hardware dazu. Mit einer schlechten Tastatur vertippt man sich ständig, auf pixeligen Monitoren will man keine langen Texte lesen und wenn das WLAN zickt, friert ständig das Bild in der Videokonferenz ein. Die Tests und Kaufberatungen in diesem Heft helfen Ihnen bei der Technik-Auswahl.

**Auch als einzelnes Heft erhältlich.**

shop.heise.de/ct-hardware-ho21

Heft + PDF für nur 19,90 € ›

› Generell portofreie Lieferung für Heise Medien- oder Maker Media Zeitschriften-Abonnenten oder ab einem Einkaufswert von 20 €. Nur solange der Vorrat reicht. Preisänderungen vorbehalten.

heise shop

shop.heise.de/ct-hardware-ho21 ›

BIST DU DABEI?

# Hannover Maker Faire®

## Digital Edition

**18.06.2021** 11 – 16 Uhr

**clevere Maker**

**inspirierendes Live-Programm**

**jede Menge Platz für Austausch**

**kostenlose Teilnahme**

**www.maker-faire.de/hannover**