



Extra-Tastatur für Videokonferenzen

- ▶ Für Zoom, Teams & Co
- ▶ Hotkeys zur Steuerung von Mikro und Kamera
- ▶ Mit Raspberry Pico
- ▶ Mit beleuchteten Tasten



Projekte

- ▶ ESP32CAM mit Alexa koppeln
- ▶ 3D-Drucker als Schneideplotter
- ▶ Smarthome: Heimkino steuern

Know-how

- ▶ Übersicht: IoT-Dienste für Maker
- ▶ Strom sparen bei ESP-Boards
- ▶ 3D-Druck-Resin im Test

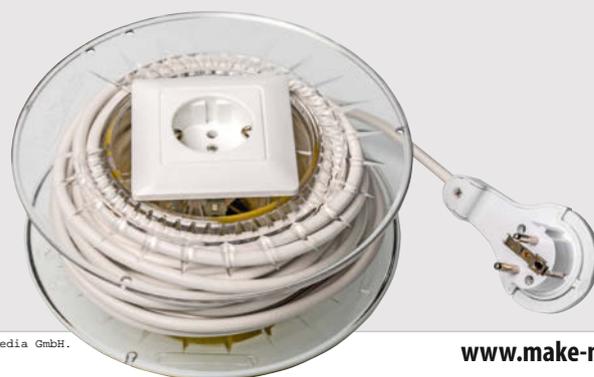
Werkzeugkiste auf Maß bauen

- ▶ Mit integrierten Griffen und raffiniertem Schiebedeckel
- ▶ Online-Rechner für Varianten



Kabeltrommel misst Strom

- ▶ Mit ESP8266, WLAN und Stromsensor
- ▶ Upcycling: Aus alter Filamentspule



3/21
10.6.2021
CH CHF 25.80
AT 14,20
Benelux 15,20
€ 12,90





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.

shapertools.com



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



Bild: Freepic.com

Sprachvielfalt

Die Maker-Community ist vielseitig und bunt und alle sind willkommen, Beiträge und Ideen zu teilen und zu bauen. Auch die Artikel unserer Autor:innen und Redakteure und Redakteurinnen sind bunt, ebenso in der Verwendung der Sprache. Deshalb finden Sie nun auch im Make-Magazin Artikel, die gegendert sind, etwa der Artikel „Maker Education macht Informatikunterricht kreativ“ auf Seite 78.

Wir stellen es allen frei, sich dieser Möglichkeit beim Schreiben der Texte zu bedienen und Artikel zu gendern; am besten mit Gender Gap, denn das schließt alle Geschlechter ein. Der Vereinheitlichung halber haben wir uns beim Gender Gap auf den Doppelpunkt geeinigt – und zwar verlagsweit. Denn der : liest sich fluffiger als * oder _ und wird sogar von manchen Screenreadern für Sehbehinderte unterstützt.

Bevor nun jemand mit Sprachdiktatur, Genderwahn und dem Untergang des Abendlands um die Ecke kommt: Es geht hier nicht um Zwang, Vorschriften und Gleichschaltung. Im Gegenteil: Wir schaffen neue Frei- und Spielräume, um Sprache in unseren Artikeln weiterzuentwickeln. Ebenso wie wir auch gerne Anglizismen in unseren Texten einsetzen – die ja schon seit Jahrzehnten als Sprachgefährder gelten.

Wer will, kann aber wie bisher das generische Maskulin verwenden, Abschnitte mit Gender Gap und generischem Maskulin mischen oder komplett auf Personenbezeichnungen verzichten.

Wir gehen an unsere Texte und die unserer Autoren ähnlich beherzt, spielerisch und offen heran, wie an unsere Projekte.

Ich bin mir sicher, dass Sie, liebes Lesy (siehe Link), in Ihrem Wissens- und Tatendrang nicht gebremst oder de-inspiriert werden. Sagen Sie uns gerne Ihre Meinung dazu, entweder per Mail oder persönlich auf der digitalen Maker Faire am 18. Juni.

► make-magazin.de/xzme

Daniel Bachfeld

Sagen Sie uns Ihre Meinung!

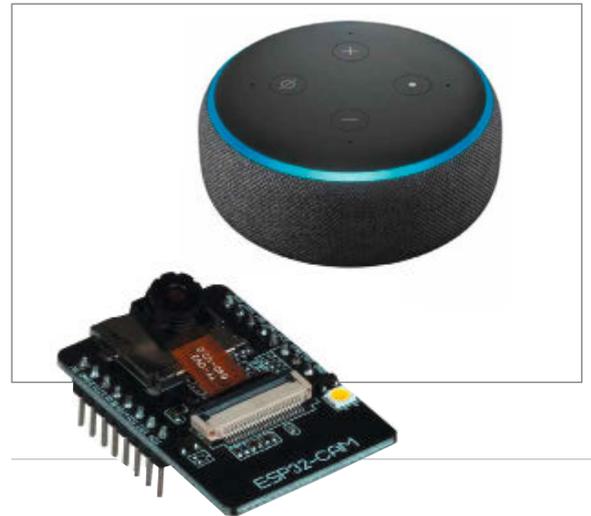
mail@make-magazin.de

Inhalt

IoT-Projekte

Mit der Prämisse „geschickt kombiniert“ gelingen mit erstaunlich wenig Aufwand tolle Internet-of-Things-Projekte: Etwa, wenn die gute Alexa Familienmitglieder am Gesicht erkennt und je nachdem den Tag mit Jazz oder Rap startet. Wir haben uns IoT-Dienste angesehen, die mit Maker-gerechten Schnittstellen und Funktionen aufwarten.

- 32 IoT-Dienste für Maker
- 62 Gesichtserkennung steuert Alexa
- 90 TV und Heimkino mit Home Assistant



- 3 Editorial
- 6 Leserforum
- 8 **Extra-Tastatur für Videokonferenzen**
- 18 **Neue Aufgaben für 3D-Drucker**
- 26 Was uns inspiriert: Sphärische Skulpturmusik, Gitarren aus Skateboards, Bluetooth-Box mit Ferrofluid-Animation
- 30 Wettbewerb: Smart-Home-Projekte
- 32 **IoT-Dienste für Maker**
- 44 Werkstattberichte: Neues aus der Szene, Comic
- 46 **Filament-Kabeltrommel mit Leistungsmessung**
- 52 **Arduino als Lokführer**
- 58 Make: Block – neue Hard- und Software
- 62 **Gesichtserkennung steuert Alexa**
- 68 Test: CNC-Fräse Yeti Tool Smartbench
- 72 Test: CNC-Oberfräse Shaper Origin
- 74 Community-Projekte: Stand-up-Arcade, Unendlichkeitsspiegel mit ESP8266
- 78 Maker Education im Informatik-Unterricht

Extra-Tastatur für Videokonferenzen

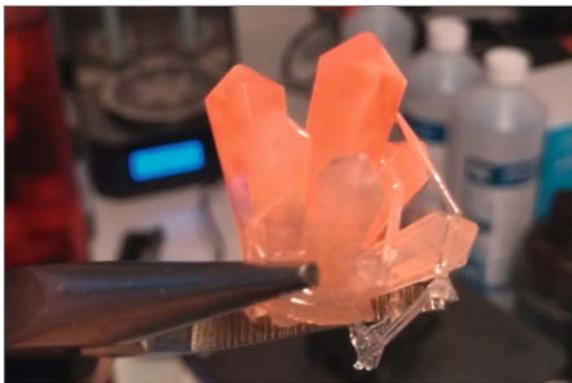
Bei der Konferenz im Büro bleibt das Risiko, dass quengelnde Kleinkinder oder halbnackte Partner durchs Blickfeld huschen, noch überschaubar. Dagegen wünscht man sich bei Videomeetings aus dem Home-Office oft genug einen prominent platzierten Not-Aus-Taster, damit nicht die ganze Abteilung das Malheur live miterlebt. Unsere Pico-gesteuerte Flitzer- und Räuspertaste sorgt hier für Diskretion.

- 8 Extra-Tastatur für Videokonferenzen

Know-how

Die beliebten ESP-Mikrocontroller bieten eine Fülle von Stromspar-Modi für batteriebetriebene Geräte, doch herrscht hier vielerorts Unsicherheit über die richtige Anwendung. Es wird Zeit, die Optionen einmal penibel unter die Lupe zu nehmen, ebenso wie das sich zunehmend etablierende Resin-3D-Druckverfahren: Wir haben verschiedene Harz-Mixturen ausprobiert.

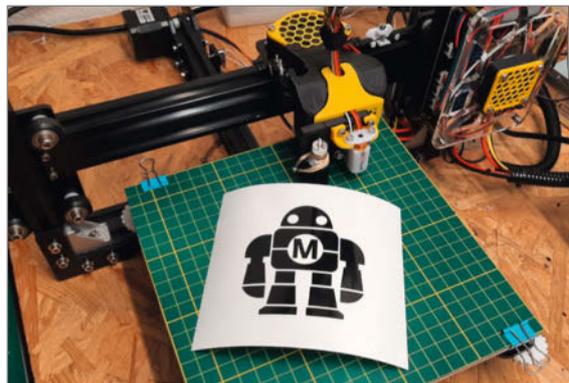
- 84 Strom sparen bei ESP-Mikrocontrollern
- 106 3D-Drucken mit Harz



Upcycling

Seit der Neuanschaffung fängt Ihr allererster 3D-Drucker im Regal Staub? Gönnen Sie ihm ein zweites Leben als Schneideplotter! Die alte Filament-Wickelspule bekommt derweil eine neue Aufgabe als Kabeltrommel mit Mehrwert, und ein paar Holzbrettchen aus der Restekiste werden mit unserem Plan zu einem praktischen Werkzeugkasten.

- 18 Neue Aufgaben für 3D-Drucker
- 46 Filament-Kabeltrommel mit Leistungsmessung
- 116 Werkzeugkiste nach Maß



- 84 **Strom sparen bei ESP-Mikrocontrollern**
- 90 **TV und Heimkino mit Home Assistant, Teil 1**
- 98 Vektorgrafik-Tricks für Maker, Teil 2
- 104 Reingeschaut: Blubbermouth
- 106 **3D-Drucken mit Harz**
- 114 Tipps & Tricks: Interdentalbürsten
- 116 **Werkzeugkiste im japanischen Stil**
- 124 Kurzvorstellungen: Mikrocontroller- und andere Boards, Smarthome-Helfer, FDM-3D-Drucker und Filament, Apps für 3D und Schaltungssimulation
- 128 Medien: 3D-Druck-Podcast, Python-3-Schnelleinstieg, Vintage-Maker auf YouTube, Heimautomation, Eigenbau-Werkstatt
- 130 Impressum/Nachgefragt

Arduino steuert Märklin

Modellbahnen sind ein höchst entspannendes und noch dazu lehrreiches Hobby – aber leider kein billiges. An den Preisen für hochwertige Modelle können wir bedauerlicherweise nichts ändern, wohl aber an denen für Steuergeräte und Zubehör – wenn, wie in unserem Bauvorschlag, ein Arduino die Kontrolle übernimmt.



- 52 Arduino als Lokführer

Themen von der Titelseite sind rot gesetzt.

Leserforum

Toller Artikel

ESP32CAM liest Wasseruhr,
Make 2/21, S. 14

Das hatte ich schon lange gesucht: die Möglichkeit den Wasserzähler auszulesen und das Ergebnis weiter zu verarbeiten. Ich habe den Zähler nachgebaut und dazu zuerst alle Gehäuseteile mit dem 3D-Drucker ausgedruckt, was am meisten Zeit erforderte. Gut wäre ein Hinweis gewesen, ob mit oder ohne Support gedruckt werden muss. Ich habe mit Support gedruckt und diesen angepasst. Wahrscheinlich wäre teilweise aber gar keiner nötig gewesen. Jedenfalls passte der Tubus auf Antrieb auf den Wasserzähler.

In Summe habe ich 26 Euro für das ESP-CAM-Modul, die Micro-USB-Stecker-Platine für das CAM-Gehäuse (gab's nur im Zehnerpack – aber ok, wer weiß, was man noch mal baut) und die 16GB Micro SD Card ausgegeben. Den FTDI-Adapter hatte ich schon aus anderen Projekten.

Die Anleitung im Artikel und im Netz (GitHub) ist komplett und verständlich beschrieben. Hier ist man gut beraten, erst mal alles zu sichten und gut durchzulesen. Dann klappt's auch prima mit dem Nachbau.

Als Variante habe ich beim Flashen bei mir keinen Raspi genommen (war gerade keiner frei), sondern das *esptool* unter Windows im cmd-Fenster genutzt. Dazu muss man aller-

dings vorher Python und natürlich das *esptool* auf dem Rechner installiert haben.

Ich hatte mir viel Mühe gemacht und beim CAM-Modul den Sicherungslack entfernt, wie in der Anleitung beschrieben, um die Optik scharf stellen zu können. Wie sich herausstellte unnötigerweise, weil mein ESPCAM-Modul schon in der Auslieferungseinstellung das Bild scharf abgebildet hat. Alles hat letztendlich prima nach Anleitung funktioniert und der Zähler ist auf Antrieb gelaufen.

Zuletzt habe ich in der Software noch die MQTT-Ausgabe aktiviert. Auch das hat sofort funktioniert, so dass ich die Messwerte in meiner Smarthome-Lösung mit FHEM direkt sehen und verarbeiten kann.

Fazit: ein toller Artikel, gut beschrieben und auch durchzuführen. Die 2h Zeitaufwand sind allerdings etwas zu optimistisch angesetzt :-).

Oliver Weitz (*Ollibaba*)

Wenig Nutzwert

Sprachausgabe für Raspberry-, Arduino- und ESP-Projekte, Make 2/21, S. 8, ESP-32CAM liest Wasseruhr, Make 2/21, S. 14, Plasmaschneiden für Padawane, Make 2/21, S. 104

Aufgrund der Anzeige zu Make 2/21 habe ich mir am Bahnhof das Heft gekauft. Besonders

die Hinweise auf Sprachausgabe und Plasmaschneider trafen mein Interesse. Nach dem Lesen der beiden Artikel muss ich mir leider etwas Naivität bescheinigen. Eigentlich sollte man doch knallige Überschriften zumindest überprüfen.

Zum Thema *Sprachausgabe*: Eigentlich hatte ich mit einem Artikel gerechnet, der mir die Erstellung einer Sprachausgabe erleichtert. Bekommen habe ich eine Auflistung verschiedener Optionen, die meist nur ganz kurz angerissen werden. Was blieb mir übrig, ich habe mir ein MP3-Audio-Modul besorgt, einige Audiodateien gebaut und die kann ich jetzt in meinen kleinen Mikrocomputern zur Sprachausgabe nutzen. Kostet zusammen incl. SD-Karte und Lautsprecher ca. 4 Euro.

Zum *Plasmaschneider*: In der Reklame wird ein 150-Euro-Gerät erwähnt. Im Artikel auch einmal ganz kurz, aber dann wird direkt eins für 550 Euro genommen. Für den Anfänger, der noch nicht weiß, ob das etwas für ihn ist, eine Menge. Besonders, wenn man die zusätzlichen Kosten noch sieht. Ach ja, da wird von einem Gebläse gesprochen, aber was für eins könnte man denn nehmen? Ich würde ja auch eher mit Figuren für den Garten anfangen, und nicht mit einer Axt. Soll ich jetzt diesen Bereich mal angehen oder nicht, der Artikel ist dabei keine Hilfe.

Und zum *Auslesen der Wasseruhr*: eine nette Idee für den Theoretiker. Der praktische Nutzen dürfte eher überschaubar sein. Interessanter wäre etwas, um die aufgezwungenen Smartmeter auslesen und somit ernsthaft nutzen zu können. Die Idee, mit der Taschenlampe zu morsen ist ja wohl eher als Gag zu betrachten. Auch hier wieder, eigentlich eine gute Idee, aber der praktische Nutzen?

Tja, wie Sie sehen, begeistert bin ich von der Make nicht gerade. Bin ich vielleicht die falsche Zielgruppe? Diese Frage ist übrigens ernst gemeint. Ich stelle sie mir.

Jürgen Marsch

Zur Sprachausgabe: Zwei konkrete Beschreibungen für Projekte sind im Artikel zu finden. Was hätten Sie an weiteren Infos zum Nachbau benötigt? Zur Wasseruhr: Mehrere unserer Leser schrieben uns, dass sie die Uhr erfolgreich in Betrieb genommen haben und die Daten auslesen. Das Auslesen von Smartmetern ist als Thema für eine der kommenden Folgen unserer fortlaufenden Home-Assistent-Story geplant.

Ob Sie unsere Zielgruppe sind? Wir kennen Ihre Ansprüche, Ihre weiteren Hobbys und Ihre

Kontakt zur Redaktion

Leserbriefe bitte an:

heise.de/make/kontakt/

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

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

www.make-magazin/forum



www.facebook.com/MakeMagazinDE



www.twitter.com/MakeMagazinDE



instagram.com/MakeMagazinDE



pinterest.com/MakeMagazinDE



youtube.com/MakeMagazinDE

Korrekturen

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

Vorkenntnisse nicht. Allgemein würden wir unsere Zielgruppe als „eklektische Selbermacher“ (aka Maker) bezeichnen, die in verschiedenen Disziplinen unterwegs sind und in der Make Inspiration und Anleitung suchen/finden.

Falscher Arduino

Was uns inspiriert: Der größte funktionierende Arduino der Welt, Make 2/21, S. 64

Die Idee ist nett und die handwerkliche Umsetzung ist gut. Leider gibt es einen gravierenden Fehler, der das ganze Projekt disqualifiziert: Der Maker hat eine Produktfälschung nachgebaut – einen China-Klon mit Arduino-Logo.

Uwe Federer

Lieber Punktschweißen statt löten

Lithium-Ionen-Akkus testen und wiederverwenden, Make 2/21, S. 70

Der Autor ahnte ja schon im Artikel, dass Temperaturen über 40°C bei Akkus nicht so gesund sind, aber das hielt ihn nicht davon ab, die Zellen zusammenzulöten. Wenn man sich mal ein paar Datenblätter von Herstellern der Standard-Zelle 18650 anschaut, liest man da meist was von Lager- und Entladungstemperaturen um die max. 60°C. Um nun die großen Metallflächen zu löten, muss man sie also mit einem Lötkolben auf gut 200°C erhitzen – das geht im besten Fall auf die Lebensdauer des Akkus, im schlechtesten Fall wird der meist integrierte

Tiefentladungsschutz zerstört. Tiefentladene Akkus bilden aber mit hoher Wahrscheinlichkeit kleine Kurzschlüsse im Inneren, aus denen sich Metallbrände entwickeln können – da hilft dann auch keine Munitionskiste mehr.

Also, Anfänger sollten auf jeden Fall ein Punktschweißgerät benutzen – und nicht nur die Profis!

Jörg Zschimmer

Mehr Details zum Schreddern

3D-Druck-Abfall wiederverwerten, Make 2/21, S. 126

Die Wiederverwendung von Druckabfällen ist grundsätzlich eine gute Sache. Eventuell könnte man hier vor allem beim Thema „Schreddern“ noch weiter ins Detail gehen. Die eine oder andere Möglichkeit wurde zwar angesprochen, aber auch hier wäre eine Selbstbauanleitung mit genauer Erklärung, Stücklisten und Lieferanten hilfreich. Bei den Selbstbauschreddern aus dem Internet gibt es zwar zahlreiche Anleitungen, leider benötigt man meist eine voll ausgerüstete Metallwerkstätte für die Umsetzung.

Georg Steiner

In der Tat stellt der Eigenbau von Schreddern hohe Anforderungen an Material und Werkzeug, das liegt leider in der Natur der Sache beziehungsweise der aufzubringenden Kräfte. Aber ja: Wir bereiten derzeit einen weiteren Artikel zum Thema „Kunststoffrecycling“ vor.

Lieber in der EU kaufen

Ausprobiert: Pico Display, Make 2/21, S. 134

Danke für den Beitrag zum Thema *Pi Pico* und den Hinweis auf *Pimoroni*! Pimoroni ist eine coole Adresse für alles rund um den Pi Pico. Beeindruckend, wieviel Hardware und Software das Pimoroni-Team seit Erscheinen des Pi Pico im Februar schon auf die Beine gestellt hat. Ich konnte mich nur mit Mühe zurückhalten, nicht gleich das ganze Sortiment zu bestellen. Nach Abschicken meiner Bestellung im Shop am 6. April wurde es dann spannend: Der Wert meiner Bestellung belief sich auf 52,09 Euro, plus 11,51 Versand per *Royal Mail Tracking*. Nach knapp zwei Tagen ging sie auf die Reise und traf am 13.4. beim Zoll am Frankfurter Flughafen ein. Dort schlummerte es dann zehn Tage, bevor es in Bayern eintraf. Meine *lessons learned*: Die Lieferung dauerte gut drei Wochen und der Versand hat inklusive Zollgebühren 29,58 Euro (57 Prozent des Warenwerts) gekostet. Es gibt auf der Pimoroni-Website einen Link zu ihren Distributoren: <https://shop.pimoroni.com/pages/worldwide-distributors>. Mein Tipp: lieber dort und in der EU kaufen. Allerdings: Die deutsche Distributorenlandschaft gleicht momentan noch eher einer Pi-Pico-Wüste ...

H.M.Hilbig

Die Wege aus Great Britain sind tatsächlich noch/wieder/dauerhaft(?) unerforschlich, wenn man über die Zollgrenze kommt. Ein deutscher Händler ist etwa berrybase.de, hier gibt es viel aus dem Pimoroni-Sortiment, aktuell auch Picos in Tausender-Stückzahlen.

MIT
Technology
Review
Das Magazin für Innovation von Heise

Alle reden heute über
die Zukunft der Arbeit –
wir seit 2013.*



*Ausgabe 11/2013: Computer machen die Arbeit.

Testen Sie 2 Ausgaben
Technology Review
mit 40 % Rabatt.

Jetzt bestellen:
trvorteil.de/testen

✉ leserservice@heise.de

☎ +49 541/80 009 120

+ Ihr
Geschenk:



Extra-Tastatur für Videokonferenzen

Ein Hustenanfall oder der Partner läuft halb bekleidet während einer wichtigen Videokonferenz durch das Zimmer. Sie teilen versehentlich den Bildschirm, statt die *Mute-* und *Kamera-*Icons zu treffen. Dies ist alles vermeidbar mit einer speziellen Zusatztastatur für das Homeoffice. Wir zeigen, wie Sie solch eine Tastatur günstig und einfach mit dem Raspberry Pico bauen.

von Carsten Wartmann



Hatten Sie in einer Videokonferenz schon mal einen Hustenanfall und haben vor lauter Husten das kleine Icon für die Mikrofon-Stummschaltung mit der Maus nicht getroffen? Der Tastaturbefehl dafür ist kryptisch und muss mit beiden Händen eingetippt werden. Oder mussten Sie die Kamera schnell deaktivieren, weil ein weinendes Kind ins Zimmer stürmte? Hier wäre eine zusätzliche Tastatur ein Segen, mit großen und gut zu treffenden Tasten, die selbst die kryptischsten Tastenkürzel in Sekundenbruchteilen an eine Anwendung sendet.

Am Beispiel einer Erweiterungstastatur für *Microsoft Teams* zeigen wir Ihnen, wie man mit dem Raspberry Pico solch einen nützlichen Schnellschalter ganz leicht selbst baut und programmiert. In unserem Beispiel-Projekt verwenden wir die Tastenkürzel für die Funktionen wie *Hand heben*, *Video umschalten* und *Mikrofon umschalten*. Der Raspberry Pico wird in der leicht erlernbaren Sprache *Python* programmiert, damit ist eine Anpassung an andere Anwendungen leicht und wird später im Artikel erklärt.

Alles, was wir brauchen, ist ein Raspberry Pico mit *Circuit Python*, ein USB-Kabel, Tasten zum Anschluss an den Pico, ein Gehäuse und etwas Bastelzeit. Wer nicht löten mag, kann das Ganze auch auf einem Steckbrett aufbauen, hier muss man dann nur sehen, wie die Taster angeschlossen werden. Entweder man benutzt bereits mit Kabeln versehene Taster oder solche, die sich auf das Breadboard stecken lassen. Gerade für einen Test, ob man solch eine Tastatur braucht, ist diese Methode ideal, denn so sind die Bauteile und der Pico in Minuten wieder zu anderen Zwecken bereit.

Pico und Computer vorbereiten

Wir werden für die Konferenzastatur den Pico mit *Circuit Python* programmieren. Dies ist ein Python-Dialekt, der direkt auf dem Pico läuft. Für *Circuit Python* ist eine neue Firmware für den Raspberry Pico nötig. Keine Angst, wenn Sie jetzt eventuell an kryptische Befehle und kompliziert anzuschließende Programmier-Gadgets wie bei anderen Mikrocontrollern denken. Eine Firmware auf dem Pico wird per Drag & Drop installiert und dies ist so einfach wie eine Datei von einem Laufwerk auf ein anderes zu schieben. Laden Sie die Firmware von CircuitPython.org (wir verwenden die Stabile Version *adafruit-circuitpython-raspberry_pi_pico_de_DE-6.2.0.uf2*), Links in der Kurzinfo) herunter und speichern sie auf Ihrem Computer.

Stecken Sie den Pico bei gehaltener BOOT-Taste an den Rechner, warten Sie ein paar Sekunden und lassen den BOOT-Taster los. Der Pico erscheint im Boot-Mode immer

Kurzinfo

- » Spezial- und Zusatztastaturen mit Pico bauen
- » Python-Modul USB-HID für Circuit Python auf Raspberry Pico nutzen
- » Tasten anschließen an den Pico

Checkliste



Zeitaufwand:
etwa eine Stunde



Kosten:
ab 5 Euro

Werkzeug

- » Lötkolben, Lötzinn
- » oder Breadboard
- » Bohrer, Säge, 3D-Drucker
je nach Gehäusevariante

Material

- » Raspberry Pico
- » mehrere Tasten aus der Bastelkiste, oder Marquardt 1005.0507
- » USB-Micro-Kabel
- » Gehäuse Blech, Holz, Plastik, 3D-Druck, Lasercut ...

Mehr zum Thema

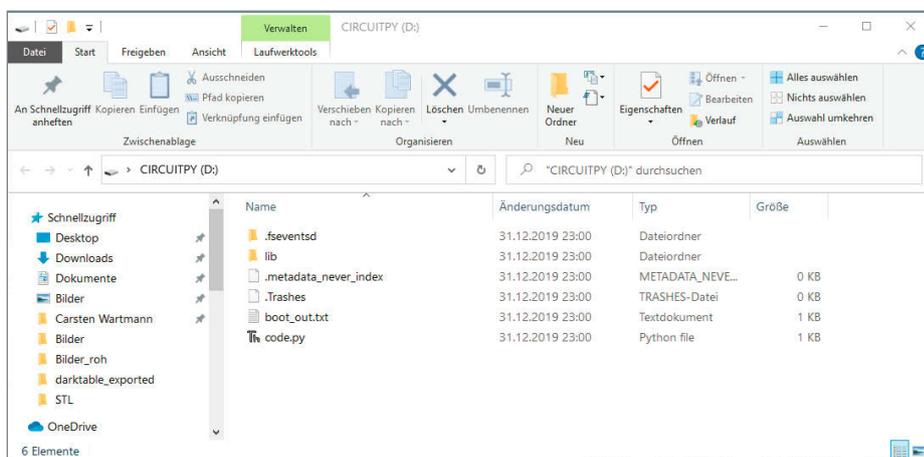
- » Peter König, Ende einer Ära, Make 1/21, S. 3
- » Andreas Perband, Raspberry Pi Pico, Make 1/21, S. 8

Alles zum Artikel im Web unter make-magazin.de/x44x

Raspberry Pico

Der RP2040-Prozessor im Raspberry Pico unterstützt USB-HID, also *Human Interface Devices*, das sind unter anderem Tastaturen und Mäuse. Das bedeutet in diesem Fall allerdings nicht, dass eine Tastatur oder Maus an den Raspberry Pico angeschlossen wird, sondern, dass sich der Pico an Windows-, Linux-, Mac- oder Android-Systemen als Maus oder Tastatur ausgibt.

Der Raspberry Pico ist natürlich fast schon Overkill für solch eine einfache Anwendung und auch andere Mikrocontroller-Boards können sich per USB-HID als Tastaturen ausgeben. Die Kombination aus einfacher Programmierung mit Python, echtem USB, günstigem Preis und weiter Verbreitung wie beim Pico ist aber nicht so leicht zu finden.



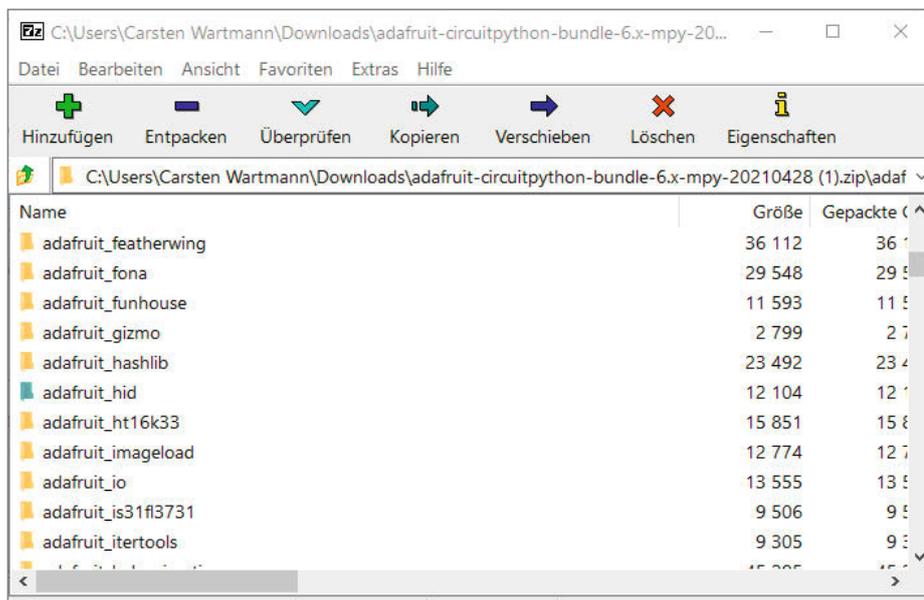
1 Das Picodrive im Dateibrowser

Problembhebung

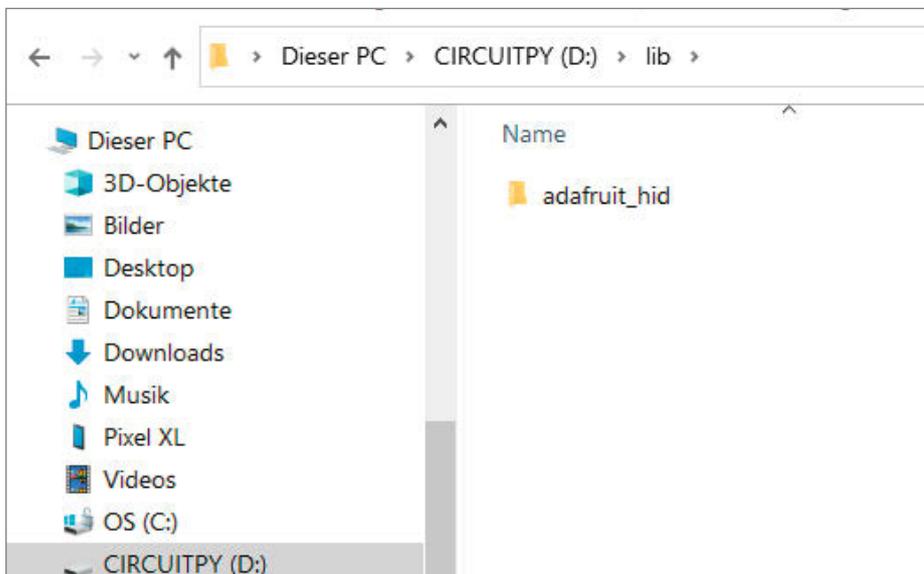
Sollte sich der Pico nicht als Laufwerk mounten lassen, versuchen Sie ein anderes USB-Kabel, am besten eines, mit dem Sie schon erfolgreich Daten auf ein Mobiltelefon oder ähnliches übertragen haben. Weiterhin blocken einige Virens Scanner das Picodrive.

Das Picodrive sollte immer per System ausgeworfen werden. Zieht man einfach das USB-Kabel vom Rechner ab, kann es zu einem Datenverlust kommen.

Wenn Ihr Pico sich nachhaltig seltsam verhält, können Sie als letztes Mittel probieren, die Firmware *flash_nuke.uf2* (Link in Kurzinfo) zu installieren. Dabei gehen alle Dateien und Programme verloren, die auf dem ursprünglichen Laufwerk gespeichert waren! Nach dem „Nuking“ können Sie, wie im Artikel beschrieben, eine neue Firmware aufspielen. Ihre Programme sollten Sie immer auch auf einem anderen Datenträger am Computer speichern.



2 Zip-Archivmanager



3 lib-Ordner auf Picodrive

als Laufwerk mit dem Namen *RPI-RP2* am Computer. Sollten Sie Schwierigkeiten haben, finden Sie im Kasten *Problembhebung* Lösungsvorschläge.

Kopieren Sie nun die heruntergeladene Firmware (*.uf2*) auf das Laufwerk *RPI-RP2*, am einfachsten per Drag & Drop oder Copy & Paste im Dateimanager Ihres Computers. Danach bootet der Pico und das Circuit-Python-Laufwerk mit dem Namen *CIRCUITPY* (im Artikel ab hier als *Picodrive* bezeichnet) erscheint als Laufwerk am Rechner 1. Auf diesem Laufwerk befinden sich schon einige Dateien, hierbei ist *code.py* die Python-Datei, die beim Starten des Pico ausgeführt wird. Mehr dazu später.

Die Circuit-Python-Firmware, die wir gerade installiert haben, enthält nur die wichtigsten Komponenten, um den Pico mit Python zu programmieren. Für spezielle Geräte und Anwendungen braucht man weitere Module. Ein ZIP-Archiv (*adafruit-circuitpython-bundle-6.x-mpy-20210428.zip*) mit allen zur Zeit unterstützten Modulen für Sensoren und Geräte kann man bei Adafruit herunterladen (alle Links in der Kurzinfo). Öffnen Sie dieses Archiv nach dem Download mit einem Archivprogramm 2 oder dem Dateimanager Ihres Betriebssystems.

In dem Archiv sind mehrere Unterordner enthalten. Klicken Sie sich bis in den *lib*-Ordner durch und scrollen darin zu dem Ordner *adafruit_hid*. Kopieren den gesamten Ordner in den *lib*-Ordner auf dem Picodrive 3. Adafruit hat auch immer eine gute Dokumentation und alternative Downloadmöglichkeiten zu allen Modulen, zum Beispiel über Adafruits GitHub.

Als Nächstes werden wir *Thonny* als Entwicklungsumgebung (IDE) installieren. Laden Sie die für Ihren Computer passende Thonny-Version von *Thonny.org*. Für diesen Artikel benötigen Sie mindestens Version 3.3.7! Für Windows gibt es einen Installer (*.exe*), die Mac-Version kommt als *.dpk*. Im Falle von Windows klicken Sie das *.exe* doppelt an und folgen den Anweisungen des Installers. Auf dem Mac benutzen sie den üblichen Weg zur Installation von *.dpk*-Paketen. Beim Installationsvorgang kann, je nach Aktualität der Thonny-Version, eventuell der Viruswarner oder *Windows Defender* ansprechen, dies darf ignoriert werden. Für erfahrende Anwender stehen auch portable Versionen (nur entpacken) und der Quellcode zur Verfügung.

Wenn Sie nun Thonny starten, dann sollte es sich wie in der Abbildung 4 darstellen.

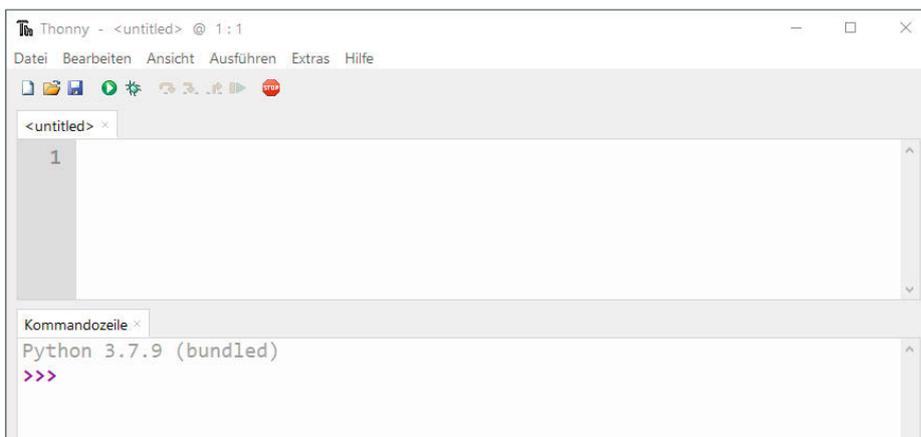
Falls der Pico nicht am Rechner steckt, dann stecken Sie ihn bitte wieder an, aber ohne den BOOT-Taster zu drücken! Damit Thonny mit dem Circuit-Python-Pico zusammenarbeitet, ist der Interpreter *Circuit Python (generic)* im Menü *Extras/Optionen...* 5 im *Interpreter*-Tab einzustellen 6. Damit weiß dann Thonny, wie er mit dem Board sprechen muss.

Das Board sollte sich nun mit *Adafruit CircuitPython 6.2.0 on 2021-04-05; Raspberry Pi Pico with rp2040* im unteren Teil von Thonny melden. Testen Sie nun einmal Thonny und Circuit-Python auf dem Pico: Im unteren Teil des Thonny-Fensters, in der *Kommandozeile*, können Sie Python Befehle eingeben, die direkt auf dem Pico ausgeführt werden. Versuchen Sie doch einmal `1+1` oder auch `print('Hallo Pico!')` **7**. Jetzt sollte dem Erfolg nichts mehr im Wege stehen und Sie können nun die Hardware aufbauen.

Tasten anschließen

Es gibt tausende mechanische Bauformen für Taster **8**. Welche Sie letztlich verwenden, ist Ihnen überlassen. Die meisten Taster sind Schließer, das heißt, der Stromkreis wird bei Druck auf den Taster geschlossen. Ohne Modifikation am Programm können wir auch nur diese in unserem Projekt gebrauchen. Für die ersten schnellen Experimente empfiehlt sich ein Aufbau auf einem Breadboard, einem Steckbrett, auf dem ein mit Pinleisten versehener Pico und ein paar Taster Platz haben.

Die Schaltung ist in **9** dargestellt. Jeder Taster ist an einem Anschluss mit der positiven Versorgungsspannung (3,3V) des Mikroprozessors verbunden, der andere Anschluss führt jeweils zu einem Port (GPIO) am Pico. Diese GPIOs können als Aus- und Eingänge per Programmierung definiert werden. Die Pins sind leider nur auf der Rückseite des Pico komplett beschriftet. Bauen Sie die Schaltung am besten zuerst auf einem Breadboard auf. Aber auch Klemmen oder Löten ist natürlich möglich. Achten Sie darauf, keine Pins am Pico versehentlich zu verbinden und arbeiten Sie nur an einem nicht am USB-Port angeschlossenen Pico.



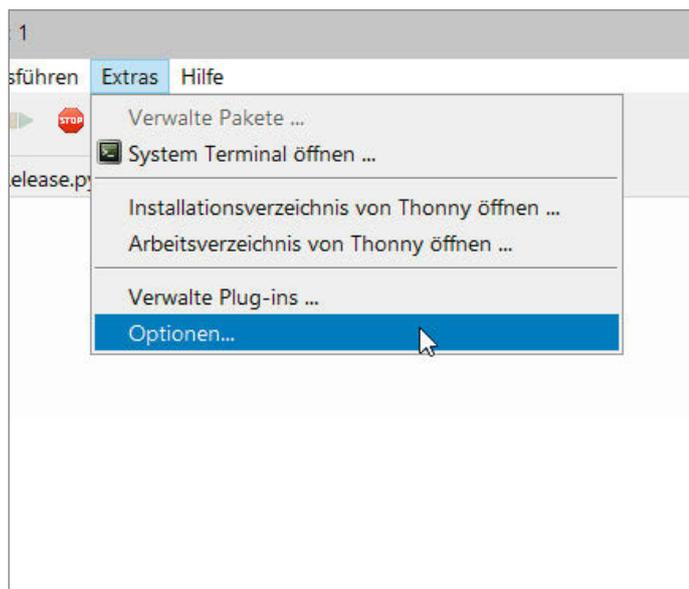
4 Die Thonny-IDE

Benutzen Sie das *Lade-Icon* oder *Datei/Öffnen...* und wählen dann *Dieser Computer*, navigieren dann im Dateifenster zum Pico-drive und laden *code.py*. Eventuell in *code.py* bereits vorhandenen Code ersetzen Sie mit dem Code aus Listing **10** *Drei_Tasten.py*. Natürlich gibt es den Code auch über die Links in unserer Kurzinfor. Sie können diesen direkt herunterladen und dann in den Editor hineinkopieren. Ich empfehle Einsteigern, die das Programmieren erlernen möchten, kurze Programme auch tatsächlich einzutippen, so lernt man die Befehle und Strukturen einer neuen Programmiersprache schneller kennen und verinnerlicht sie. Die Einrückungen sind wichtig für Python, benutzen Sie bitte die Tab-Taste, um sie gleichmäßig zu erstellen.

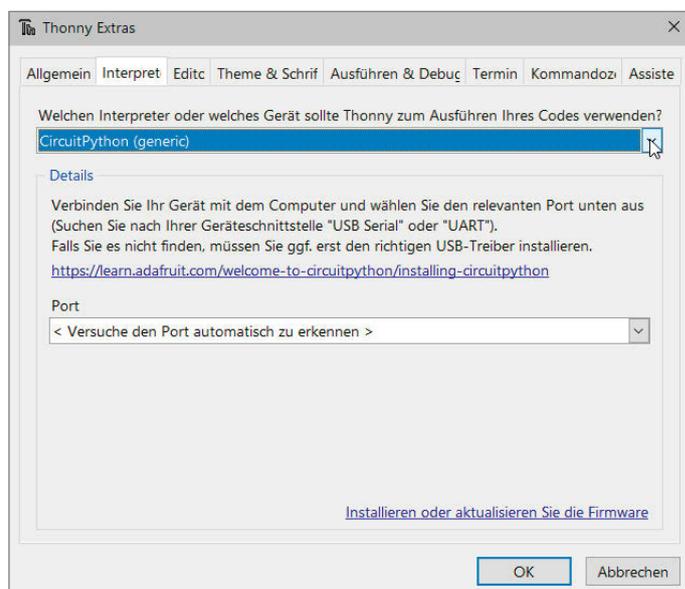
Speichern Sie jetzt *code.py* auf dem Pico-drive und starten es dann mit dem *Run-Icon* oder *F5*. Der Pico sendet jetzt beim Drücken der jeweiligen Taste Shortcuts an den

Computer. Im Skript sind dies die Tastaturkommandos für *Microsoft Teams: Hand heben* (*Strg+Umschalt+K*), *Video umschalten* (*Strg+Umschalt+O*) und *Mikrofon umschalten* (*Strg+Umschalt+M*).

Möchten Sie eine andere Anwendung steuern, muss das Programm geändert und auf dem Pico gespeichert werden. Welche Tastaturkürzel eine Anwendung versteht, bekommt man über deren Anleitung heraus. Oft sind diese Kürzel auch in den Menüs angezeigt. Besonders gut geeignet sind Programme, in denen sich neue Tastenkommandos für alle Befehle und Menüs vergeben lassen. Hier kann man dann lange Kombinationen verwenden, die garantiert nicht mit anderen Funktionen kollidieren. Gut geeignet sind auch die Key-codes der üblicherweise nicht auf Tastaturen vorhandenen Funktionstasten *F13* bis *F24*. Hier ist die Gefahr minimal, mit anderen Programmen in Konflikt zu kommen.



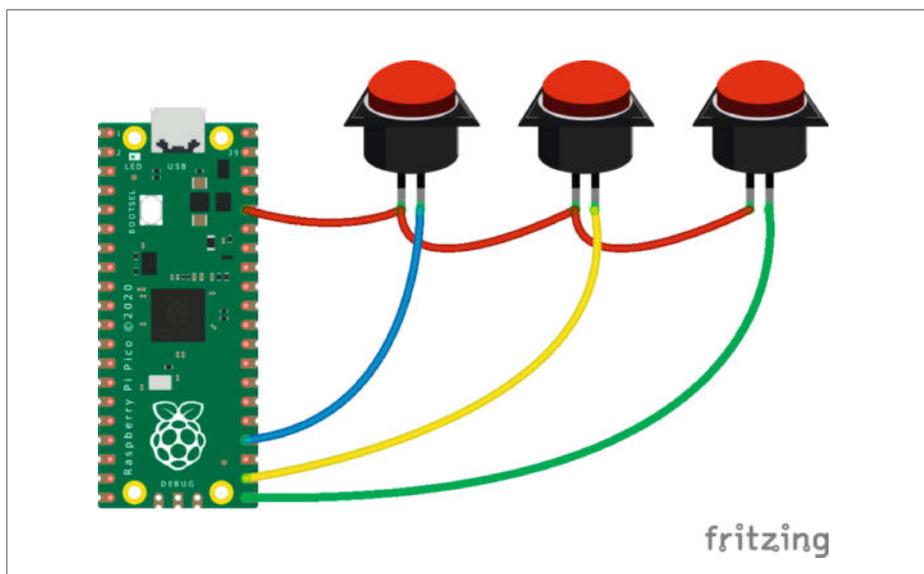
5 Die wichtigste Option ist ...



6 ... welcher Python-Interpreter benutzt werden soll.



7 Erste Lebenszeichen des Pico



9 Prinzipschaltbild für drei Taster

Haben Sie die Informationen gesammelt, suchen Sie die Keycode-Namen aus unserer Tabelle 11 heraus oder schauen online in der Dokumentation des Adafruit-HID-Moduls nach (Link in Kurzinfo). Die relevanten Zeilen im Listing 10 sind 19, 22 und 25 für die drei Tasten. Im Funktionsaufruf `kbd.send()` listen Sie, per Komma getrennt, die gleichzeitig zu drückenden Tasten auf. Möchten Sie zum Beispiel das Tastenkürzel für die erste Taste auf *Videoanruf annehmen (Strg+Umschalt+A)* ändern, dann müssen Sie Zeile 19 in:

```

kbd.send(Keycode.CONTROL,Keycode.
SHIFT,Keycode.A)
ändern.
    
```

Das Skript `code.py` wird nun immer ausgeführt, wenn der Pico angesteckt wird oder neu startet. Der Pico agiert jetzt auf Windows, Mac, Linux und sogar auf Android-Geräten mit OTG-Adapter als Tastatur und kann so Ihre Videokonferenz angenehmer und sicherer machen.

Gehäuse

Als Gehäuse eignet sich eigentlich alles, was der Maker-Schrank so hergibt: Gekaufte Fertiggehäuse 12, Holz- oder Blechkisten, Lego oder Fischertechnik, ausgeschlachtete

Python-Syntax

Codeblöcke, wie der Inhalt von Schleifen oder Funktionsdeklarationen, werden in Python durch eine Einrückung, bestehend aus Leerzeichen, gekennzeichnet und nicht wie in anderen Sprachen durch Klammern oder ähnliches. Dies dient der Lesbarkeit, hat aber auch seine Tücken. Wenn einem einmal ein Leerzeichen zu viel in die Einrückung rutscht, beschwert sich Python mit einem Syntax-Error. Benutzen Sie am besten die Tab-Taste, sie lässt den Cursor zur nächsten Einrücktiefe springen und erspart umständliches Zählen von Leerzeichen. Mit Tab bzw. Shift-Tab können auch ganze selektierte Bereiche in der Einrücktiefe geändert werden.

Geräte mit Tasten, alte Telefone mit Tasten, Gehäuse aus dem Laser-Cutter oder 3D-Drucker – es muss nur der Pico hineinpassen und das USB-Kabel muss herausführen.

Das Gehäuse aus dem Aufmacherbild und 13 finden Sie auf unserem GitHub, es ist für besonders robuste Taster aus Arcade-Joysticks konstruiert. Die transparenten Tastenkappen wurden mit *Prusament-PVB* gedruckt (siehe Kurzvorstellung auf Seite 125). Dann wurden die Tastenkappen mittels Isopropanol geglättet und durchsichtig gemacht. Das Gehäuse ist modular aufgebaut, so dass Sie die Tasten vielfältig kombinieren können, was Farben, Kappen und Einleger angeht. Die Tasten bieten unten Bohrungen für 3mm LEDs, falls Sie die Tasten beleuchten möchten. Weiterhin gibt es auf unserem GitHub eine Blender-Datei, mit der recht einfach das Gehäuse für andere Tastenzahlen abgewandelt werden kann.

8 Einige Taster aus der Bastelkiste



Aber auch gängige 3D-Druck-Plattformen bieten unter dem Suchbegriff „Streamdeck“ viele Gehäuse zum Ausdruck an.

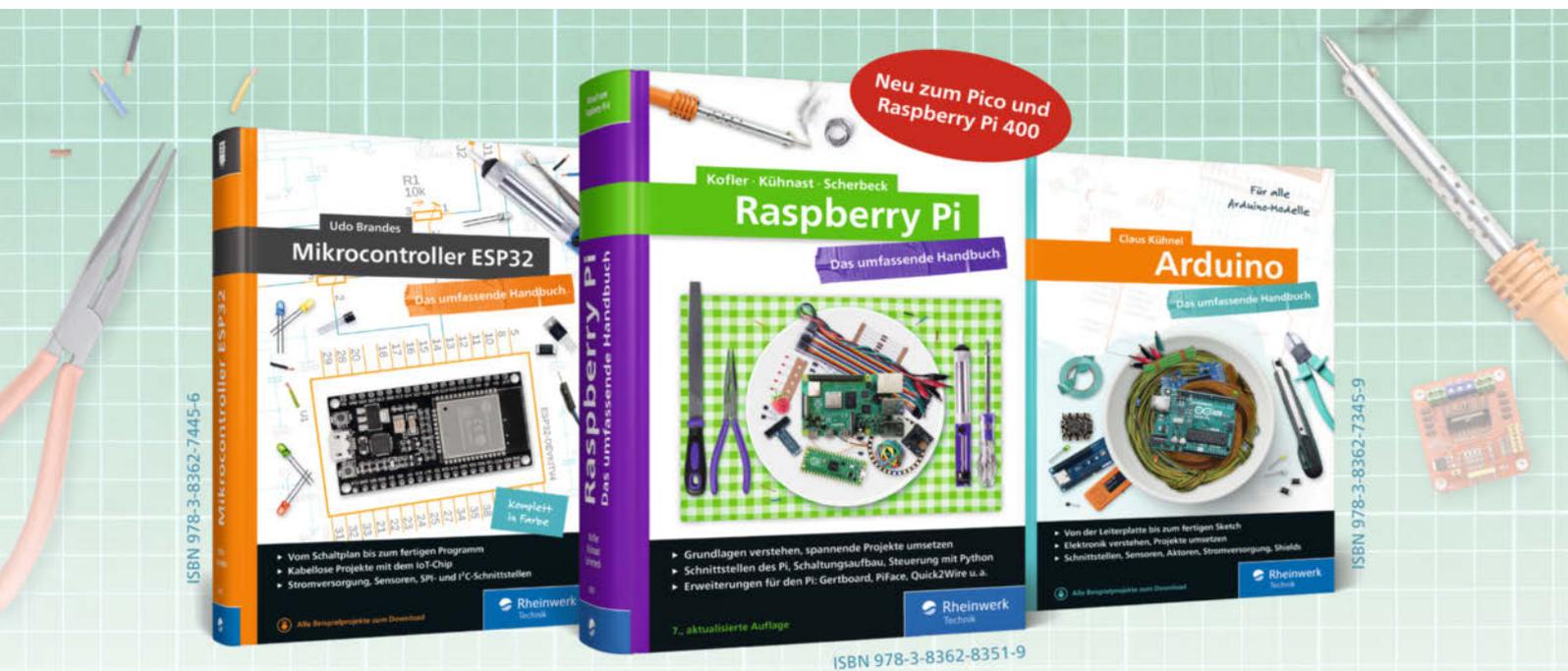
Sie wollen es genauer wissen?

In diesem Abschnitt gehe ich etwas tiefer in die Materie und erkläre einige Details dieses Projekts genauer. Mit diesem Wissen können Sie dann dieses Projekt erweitern oder abwandeln. Ich werde anhand kleiner Beispiele die Funktion der einzelnen Komponenten genauer erläutern. Die GPIOs bleiben gleich, sodass Sie die oben gebaute Hardware weiter verwenden können. Für die Beispiele in diesem Abschnitt verwenden wir nicht `code.py` (Kasten *Programmstruktur auf dem Picodrive*), sondern speichern die Programme unter einem anderen Namen ab, damit man sie später als Referenz und Grundlage für neue Programme benutzen kann. Damit das `code.py` für die Teams-Tasten nicht dazwischen funkt, sollten Sie es für die weiteren Experimente auf dem Picodrive umbenennen oder den Inhalt löschen.

Die Struktur des Programms für unsere Pico-Tastatur ist simpel. Schauen Sie noch einmal ins Listing 10. In den Zeilen 1-6 werden

10 Drei_Tasten.py

```
01 import board
02 import digitalio
03 import time
04 import usb_hid
05 from adafruit_hid.keyboard import Keyboard
06 from adafruit_hid.keycode import Keycode
07
08 kbd = Keyboard(usb_hid.devices)
09
10 taste1=digitalio.DigitalInOut(board.GP16)
11 taste1.switch_to_input(pull=digitalio.Pull.DOWN)
12 taste2=digitalio.DigitalInOut(board.GP17)
13 taste2.switch_to_input(pull=digitalio.Pull.DOWN)
14 taste3=digitalio.DigitalInOut(board.GP18)
15 taste3.switch_to_input(pull=digitalio.Pull.DOWN)
16
17 while True:
18     if taste1.value:
19         kbd.send(Keycode.CONTROL,Keycode.SHIFT,Keycode.K)
20         time.sleep(0.2)
21     if taste2.value:
22         kbd.send(Keycode.CONTROL,Keycode.SHIFT,Keycode.O)
23         time.sleep(0.2)
24     if taste3.value:
25         kbd.send(Keycode.CONTROL,Keycode.SHIFT,Keycode.M)
26         time.sleep(0.2)
```



Bücher für Alles-Erfinder

IoT-Projekte mit dem ESP32, Roboter-Autos, Sensoren und Aktoren mit dem Arduino – unsere Autoren erleichtern Ihnen den Einstieg in die Maker-Welt. Mit allen Grundlagen zu Linux, Programmierung und Elektrotechnik. Hochexplosiver Zündstoff für Ihre Ideen!

Alle Handbücher auch als E-Book und im Bundle.

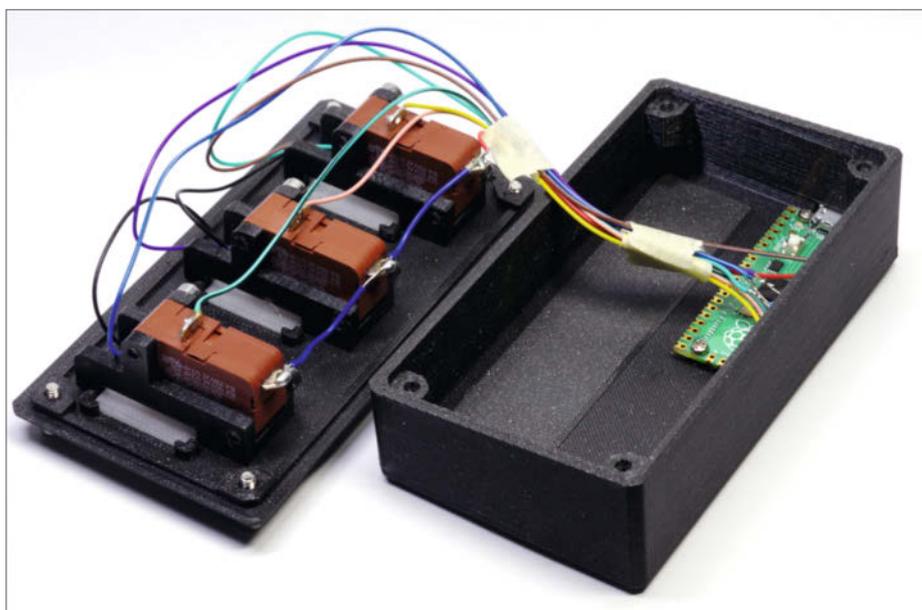
www.rheinwerk-verlag.de Copyright by Maker Media GmbH.

11 Beispiele für Tastencodes

Keycode	Taste
A,B,C...	Entsprechende Buchstabentaste, X und Y vertauscht!
ONE, TWO, THREE...	Ziffern
END	Ende-Taste
CAPS_LOCK	Die ungeliebte Shift-Lock-Taste
COMMAND	Mac-Command-Taste
LEFT_CONTROL, CONTROL	Strg-Taste
ESCAPE	Esc-Taste
F1-F24	Funktionstasten
SHIFT	Shift-Taste, Umschalt-Taste
GUI	Windows-Taste, Command (Mac), Meta
PRINT_SCREEN	Druck-Taste
SPACEBAR	Leertaste



12 Zusatztastatur in Fertiggehäuse aus dem Elektronikbedarf



13 Gehäuse in 3D-Druck

die benötigten Module (ähnlich wie Bibliotheken in anderen Sprachen) in das Programm geladen (*import*). Die Module *board* und *digitalio* stellen die Funktionen zur Behandlung der Tasten an den GPIO-Anschlüssen des Pico. Das *time*-Modul liefert eine Schlaffunktion, in der der Prozessor wartet oder andere Dinge machen kann. Die Module *usb_hid* und die *adafruit_hid* Module stellen die USB-Tastatur-Funktionen bereit.

In Zeile 8 wird das Keyboard-Objekt (Objekte in Python sind mehr als einfache Variablen) erstellt und in der Variablen *kbd* für den späteren Zugriff gespeichert.

In den Zeilen 10-11 wird eines der drei Tasten-Objekte mithilfe der *digitalio.DigitalInOut()* Funktion auf den GPIO 16 eingestellt und in dem Objekt *taste1* gespeichert. Dieses Objekt stellt nun eine Funktion *switch_to_input()* bereit, mit der der Port auf Eingang geschaltet wird. Der Parameter *pull=digitalio.Pull.DOWN* sorgt für eine ordnungsgemäße Funktion in unserer Schaltung, bei der 3.3V über den gedrückten Schalter mit dem GPIO verbunden werden. Dazu wird ein interner Pulldown-Widerstand (siehe Kasten *Widerstände, Pulldown, Pullup?*) aktiviert. In den Zeilen 12-15 geschieht das gleiche für die anderen zwei Tasten.

In Zeile 17 wird eine *while*-Schleife begonnen, die durch die Bedingung *True* eine Endlosschleife ist. Solch eine Schleife oder jedes andere Programm können Sie übrigens beenden, indem Sie das *Stop*-Icon in Thonny benutzen oder *Strg-C* in der Kommandozeile von Thonny drücken. Eine Besonderheit im Zusammenhang mit dem *Stop*-Icon ist, dass dann auch wieder *code.py* ausgeführt wird, was bei der Entwicklung zu unerwünschten Phänomenen führen kann (s.o.).

Die Einrückung im Schleifenkörper sagt dem Python-Programm, dass dies als ein Block von Befehlen anzusehen ist und zusammen gehört. Die Einrückung nach einer *if*-Anweisung ist nochmals eine Ebene tiefer.

Ab Zeile 18 folgen dann drei *if*-Abfragen, die die jeweilige Taste auf einen Tastendruck prüfen und in diesem Falle den eingerückten Block ausführen. Dazu wird das Tasten-Objekt, *taste1.value* abgefragt, das bei gedrückter Taste *True* ist und so den Block in der *if*-Anweisung ausführt. *kbd.send()* sendet dann, wie vorhin im Projekt gezeigt, die entsprechenden Tastencodes. Danach wartet das Programm 0,2 Sekunden.

Die restlichen Zeilen erledigen die Abfrage für die übrigen Tasten und dann wird die Schleife ab Zeile 17 erneut ausgeführt.

Der Ansatz aus Listing 10 wird spätestens bei mehr als drei Tasten unübersichtlich. Aber einmal erstellt und dann in die Pico-Tastatur eingebaut, ist das eventuell die am schnellsten zum Ziel führende, wenn auch nicht die eleganteste Methode. Im GitHub finden Sie eine elegantere Version für beliebige viele Tasten.

Mediensteuerung und Maus

Neben dem Senden von Tastendrücken an den Host-Computer bietet das HID-Modul auch eine *Consumer-Control*- und eine Maussteuerung an. *Consumer Control* ist eine HID-Methode, um Multimedia-Programme per USB zu steuern. Diese beinhaltet Befehle, um die Lautstärke zu ändern, den Ton auszu-schalten (Mute), zwischen Titeln zu springen und verschiedenes mehr. Dabei muss die abspielende Anwendung auf dem Host-Com-puter übrigens nicht im Vordergrund laufen.

Die Befehle und Codes sind nicht in *adafruit_hid.keyboard* gekapselt, sondern in einer eigenen Klasse (siehe Listing 14), aber der Aufbau ist analog wie bei den Keycodes. Wird dieses Programm ausgeführt, wird die Lautstärke auf null gesetzt. Bei erneutem Aufruf des Programms wird die alte Lautstärke wieder hergestellt 15.

Bisher haben wir nur mit kurzen Tastendrücken gearbeitet. Möchten Sie eine oder mehrere Tasten halten, so gibt es dafür die *press()*- und *release()*-Funktionen. Werden *kbd.press(Keycode.A)* und *kbd.release(Keycode.A)* direkt hintereinander

Programmstruktur auf dem Picodrive

Das im Projekt oben verwendete *code.py* wird bei jeder Inbetriebnahme und jedem Neustart des Pico ausgeführt, ideal für Anwendungen die nicht mehr geändert werden. Für Experimente sollten Sie *code.py* nicht direkt ändern. Eine ungewollte, endlose Schleife beim Ausführen des Programms kann es schwierig machen, das Programm nach Einstecken des Pico zu beenden. Auch möchte man vielleicht unterschiedliche Aufgaben und Tests erledigen und nicht jedes Mal dazu *code.py* ändern. Daher ruft man, nachdem man

sicher ist, dass der Code funktioniert, aus *code.py* nur das eigentliche Programm auf. Für ein Programm mit dem Namen *anwendung.py* geschieht das mit *import anwendung* (auch ohne *.py*). In *code.py* kann man aus Sicherheitsgründen auch noch ein paar Sekunden warten, bis es weiter geht, dann hat man im Falle eines Falles noch Zeit, *code.py* zu unterbrechen. Auch wenn wir hier mit Python arbeiten, sind wir doch auf einem Mikroprozessor aktiv, den ein wirres Programm durchaus crashen kann.

ausgeführt, sind sie das Äquivalent zu *kbd.send(Keycode.A)*. Interessant wird es, wenn Sie zwischen den Befehlen eine Zeitverzögerung einbauen. Der Code in Listing 16 drückt die A-Taste zwei Sekunden lang und dies führt zu einer Tastenwiederholung und vielen „aaaaaa“ in der gerade aktiven Anwendung. Hier lauert auch eine Gefahr, denn wenn

durch einen Programmfehler (zum Beispiel ein Tippfehler in der *time.sleep()*-Zeile) die Taste nicht mit *release()* freigegeben wird, sendet das HID-Modul den Tastencode unendlich. In so einem Fall kann man nur noch schnell den Pico abstöpseln.

Entsprechende Funktionen gibt es auch für die Mediensteuerung mit *Consumer-Con-*

9 JAHRE BASTLER-KNOW-HOW KOMPAKT



Das komplette Make-Archiv auf 32 GByte USB-Stick

Auf dem USB-Stick finden Sie 54 Ausgaben c't Hacks (2011 bis 2014) und Make (2015 bis 2020) mit Evergreens wie den Teehasen, die Fräse MaXYposi, den DIY-LötKolben, die Plotclock, Lixie-Anzeigen, die Reißzweckenorgel und viele weitere zeitlose Projekte zum Nachbauen.

shop.heise.de/make-archiv20

99,90 € >

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



shop.heise.de/make-archiv20 >

Nachwuchswettbewerb für junge Elektronikfans

- Zeig, was Du kannst und gewinne bis zu 3.000 €
- Reiche Dein Projekt zum Thema **Smart Energy oder Energy Harvesting ein**
- Einsendeschluss: 1. September 2021



Der PAUL Award ist eine Aktion des Fachverbandes für Design, Leiterplatten- und Elektronikfertigung (FED). Er will junge Menschen fördern und auszeichnen, die eine innovative technische Idee kreativ umgesetzt haben. Benannt ist der Preis nach Paul Eisler, dem Erfinder der Leiterplatte. Im Sommer 2022 werden alle Nominierten zur großen Siegerehrung nach Berlin eingeladen.







Fachverband für Design, Leiterplatten- & Elektronikfertigung

Alle Infos findest Du unter www.paul-award.de

instagram.com/paulawardtalents







Weitere Ideen

Eine Eigenbau-Tastatur kann auch gut eingesetzt werden bei Anwendungen wie öffentlichen Terminals oder Zugangssystemen, bei denen eine komplette Tastatur nicht nötig, erwünscht oder zu empfindlich ist. Dann können spezielle Tasten eingesetzt werden, die wasserfest oder vandalensicher sind. Gehen einem bei einem Raspberry-Pi-Projekt die GPIOs aus, so kann solch eine Pico-Tastatur noch viele weitere Funktionen hinzufügen, auch parallel zu der normalen Tastatur.

Geht man einen Elektronikatalog durch und sieht Neigungsschalter, Reedkontakte

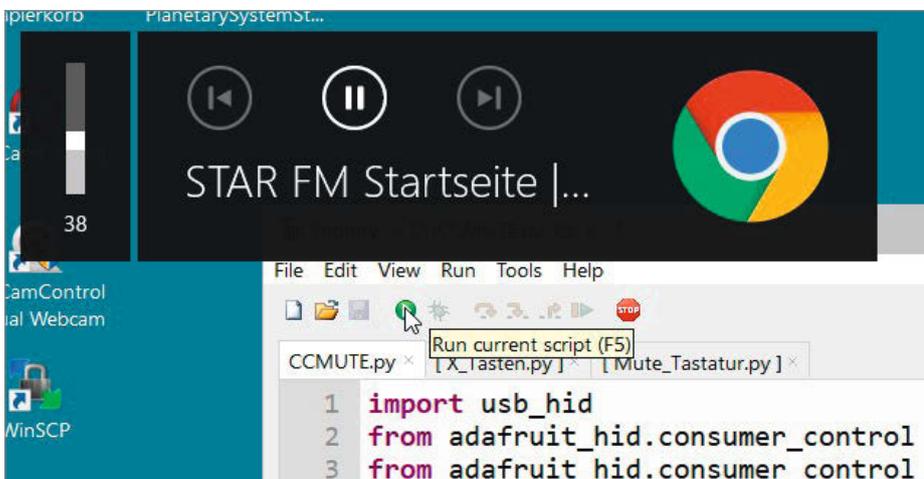
(durch Magnetfelder ausgelöst), Schlüssel-schalter, Näherungsschalter oder Lichtschranken mit Relaisausgang, fallen einem die Ideen geradezu in den Schoß.

Wer mit dem PC Musik macht, wünscht sich vielleicht ein Sustain- bzw. Haltepedal für Software-Synthesizer. Näherungsschalter können kontaktlose Eingaben an einem zu steuernden Gerät ermöglichen. Es sind auch Anwendungen denkbar, um Geräte auf Herz und Nieren zu testen. Entwickeln Sie neuartige, praktische oder behindertengerechte Eingabegeräte, bis hin zu funktionalen Simulator-Cockpits. Seien Sie kreativ!

14 CCMUTE.py

```
# Consumer Control Mediensteuerung
import time
import usb_hid
from adafruit_hid.consumer_control import ConsumerControl
from adafruit_hid.consumer_control_code import ConsumerControlCode

cc = ConsumerControl(usb_hid.devices)
cc.send(ConsumerControlCode.MUTE)
```



15 Der Pico hat gerade meinen Lieblossender wieder laut geschaltet

ontrol-Befehlen. So können Sie dann die Lautstärke Ihres Computers vom Pico aus steuern. Im Listing 17 wird auf diese Weise die Lautstärke eine Sekunde erhöht, dann wird zwei Sekunden gewartet und die Lautstärke wieder gesenkt. So können Sie eine Tastatur mit Tasten für Ihren Mediacenter-PC bauen. Die CC-Codes sind bei Adafruit dokumentiert.

Auch die Maus lässt sich steuern, was durchaus bei Tests auf Hardware, auf denen man keine Anwendungen installieren darf oder kann, verwendbar ist. Bei der Recherche

zu diesem Artikel habe ich auch gelernt, dass es sogenannte „Mouse Wiggler“ gibt, die auf einem von der Firma überwachten Computer eine Aktivität des Benutzers vorgaukeln. Ich warte noch mit dem Kauf, bis diese Geräte auch sonst meine Arbeit abnehmen können. Ein ganz simples Programm, das die Maus bewegt, sehen Sie in Listing 18.

Wird der Taster betätigt, verschiebt sich der Mauscursor etwas nach links oben. Wahrscheinlich kann man mit einem ähnlichen Programm, dass zufalls gesteuert die Maus bewegt,

Widerstände, Pulldown, Pullup?

Pullup- oder *Pulldown-*Widerstände sorgen an Mikrocontrollern für definierte elektrische Zustände an den Eingängen. Ohne diese Maßnahme könnten selbst kleine elektrische Störungen den Eingang ungewollt schalten. Offene Eingänge zeigen dann oft völlig zufällige Werte an. Man kann als Abhilfe externe Widerstände benutzen oder auch die im Chip vorhandenen internen Widerstände. Bei einigen Mikrocontrollern gibt es intern nur *Pullups*, dort hat man keine Wahl, wenn man keine externen *Pulldowns* verwenden möchte.

Pullup-Widerstände ziehen einen Eingang auf die Arbeitsspannung des Prozessors. Damit ist dieser Eingang dann *HIGH*, was im Code ein *True* (wahr) bei der Abfrage ergibt. Bei gedrücktem Taster wird der Eingang mit *GND* (Masse) verbunden und ist dann *LOW*, also *False* (unwahr) im Code. Das liest sich im Code etwas verwirrend, wird aber häufig so verwendet. Bei einem Platinenlayout kann dies Vorteile haben, da *GND* praktisch überall auf der Platine vorhanden ist und so das Layout um die Taster herum einfacher bleibt.

Pulldown-Widerstände legen den Eingang auf *GND*-Pegel. Bei gedrücktem Taster wird der Eingang mit *VCC* (Prozessor-Versorgungsspannung) verbunden und ist dann *HIGH*. Dies liest sich im Code einfacher, *True* bedeutet so „Taster gedrückt“. *Pulldown-*Widerstände wurden traditionell bei der veralteten *TTL*-Technik verwendet, damit konnte der Stromverbrauch gesenkt werden. Bei modernen Chips ist dies aber nicht mehr relevant.

jemanden zum Wahnsinn treiben. Wenn es konstruktiver sein soll, so können Sie hiermit auch einen barrierefreien Mausersatz bauen.

Egal welches Maker-Problem Sie mit einer oder mehreren Zusatztastaturen auf Basis des Raspberry Pico oder auch anderer Mikrocontroller Boards angehen wollen, fangen Sie an zu basteln! Gerne würden wir von Ihren Lösungen und Erkenntnissen hören. Eventuell möchten Sie es auch auf *Make: Projects* veröffentlichen und so der Community bekannt machen. —caw

Neue Online-Kurse
für IT-Professionals

16 Keyb_PressRelease.py

```
# Keyboard send() und send()/release()
import time
import usb_hid
from adafruit_hid.keyboard import Keyboard
from adafruit_hid.keycode import Keycode

kbd = Keyboard(usb_hid.devices)

kbd.send(Keycode.A) # send

# press/release
kbd.press(Keycode.A)
time.sleep(2)
kbd.release(Keycode.A)
```

17 CC_PressRelease.py

```
# Consumer Control Mediensteuerung
import time
import usb_hid
from adafruit_hid.consumer_control import ConsumerControl
from adafruit_hid.consumer_control_code import ConsumerControlCode

cc = ConsumerControl(usb_hid.devices)

# 2 Sekunden Lautstärke erhöhen
cc.press(ConsumerControlCode.VOLUME_INCREMENT)
time.sleep(1)
cc.release()

time.sleep(2)

# 2 Sekunden Lautstärke verringern
cc.press(ConsumerControlCode.VOLUME_DECREMENT)
time.sleep(1)
cc.release()
```

18 Maus.py

```
# Mausbewegung bei Tastendruck
import usb_hid
from adafruit_hid.mouse import Mouse

import board
import digitalio
import time

m = Mouse(usb_hid.devices)

button = digitalio.DigitalInOut(board.GP16)
button.switch_to_input(pull=digitalio.Pull.DOWN)

while True:
    if button.value:
        # 3. Parameter ist Scrollrad
        m.move(-10, -10, 0)
        time.sleep(0.2)
```



Datenvisualisierung mit Python

Der Online-Kurs zu Data Science –
mit vielen praktischen Übungen

► 60 Videos, 8 Std. Spielzeit, inkl.
Jupyter Notebook



Container-Orchestrie- rung mit Kubernetes

Das umfassende Online-Training
für Kubernetes im Praxiseinsatz

► 63 Videos, 6 Std. Spielzeit



NoSQL mit MongoDB

Die Praxiseinführung in
nicht-sequenzielle Datenbanken

► 88 Videos, 7:30 Std. Spielzeit

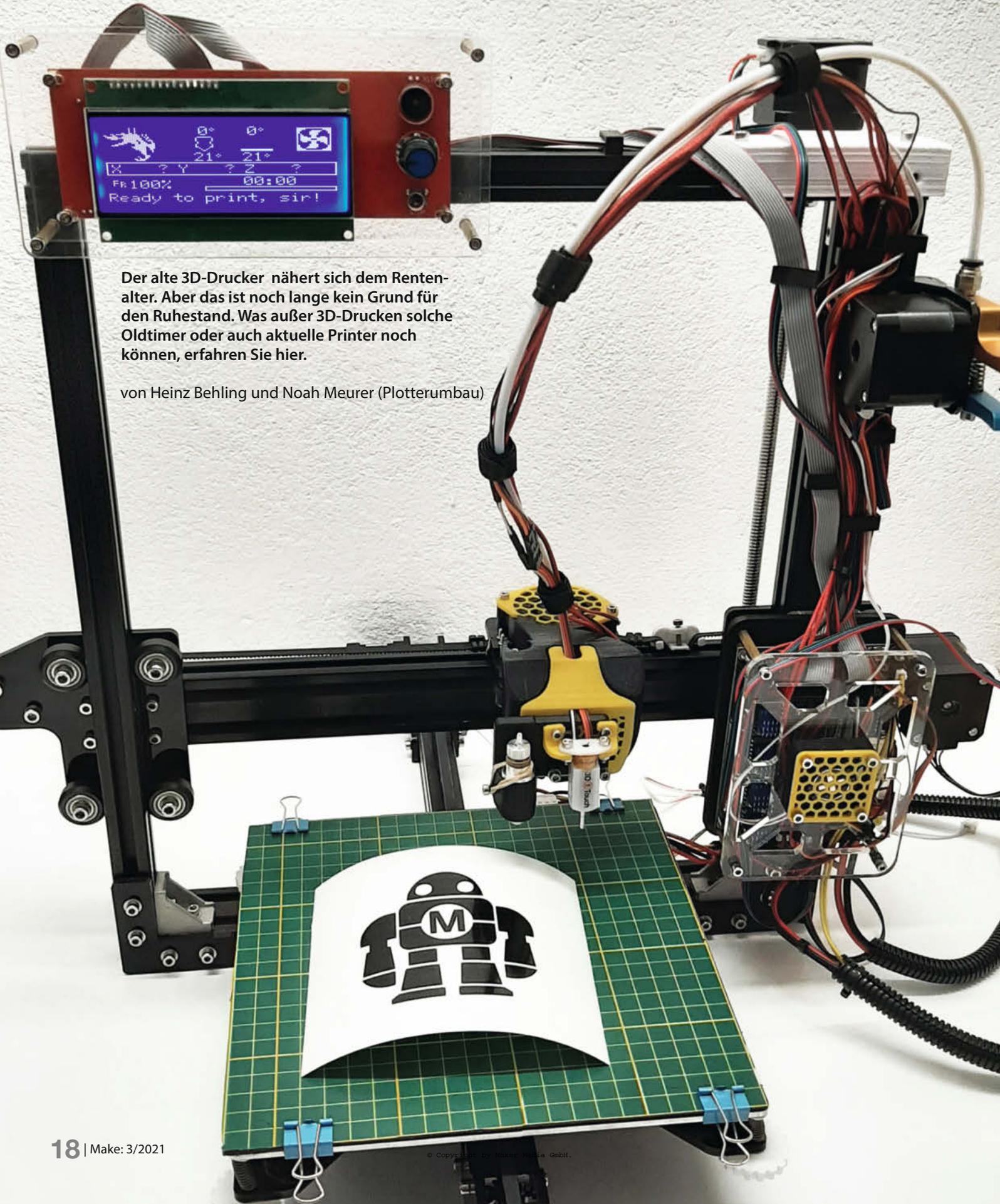
Gratis-Lektionen aus allen Kursen
anschauen unter:

heise-academy.de

Neue Aufgaben für alte 3D-Drucker

Der alte 3D-Drucker nähert sich dem Rentenalter. Aber das ist noch lange kein Grund für den Ruhestand. Was außer 3D-Drucken solche Oldtimer oder auch aktuelle Printer noch können, erfahren Sie hier.

von Heinz Behling und Noah Meurer (Plotterumbau)



Da steht er nun in der Vitrine, der gute alte *Anet A8*. Inzwischen hat er einen Bruder, einen *Creativity Ender 3*, der zum *Silent*-Modell umgebaut wurde und inzwischen wegen seiner leisen Arbeitsweise nahezu alle Druckaufträge übernimmt. Was aber gibt man dem Anet noch zu drucken? Oder finden sich vielleicht ganz neue Aufgaben?

Sinnloses ...

Eine Suche im Internet ergab da so einige Ideen und auch konkret verwirklichte Arbeitsbeschaffungsmaßnahmen, nicht nur für ausgediente Altdrucker. Da gab es zunächst die Klassiker, sprich: Umbauten zur Fräse oder zum Lasercutter. Doch bei näherer Betrachtung zeigte sich: Die Mechanik der meisten 3D-Drucker ist für eine Weiterbeschäftigung als Fräse schlicht zu schwach. Selbst dem inzwischen mit einem Metallrahmen ausgestatteten Anet wollten wir das nicht zumuten. Insbesondere die Riemenantriebe sowie die Kunststoffteile der X-Achse hätten den Belastungen eines Fräsbetriebs nicht lange standgehalten.

Das muss aber keineswegs bei allen 3D-Druckern so sein. Es gibt auch einige Modelle, bei denen der Hersteller den Umbau zur Fräse bereits von Anfang an vorgesehen und den Drucker entsprechend stabil konstruiert hat. Für diese Geräte, wie etwa den *RF 2000* von *Renkforce* (über Conrad erhältlich) gibt es Umbau- beziehungsweise Erweiterungssätze mit allem, was für eine Fräse notwendig ist. Gleich umkehrt verhält es sich bei den Geräten von *Stepcraft*: Das sind in erster Linie solide CNC-Fräsen, für die man auch einen optionalen 3D-Druckkopf bekommt.

Zwar gibt es entsprechende Umrüst-Kits auch für den Lasercutter-Umbau von 3D-Druckern. Doch hierbei ist es mit dem simplen Austausch des Druckkopfes gegen einen Laser (mit meist nicht mehr als 15W Leistung) nicht getan. Es gibt ein Abgas-Problem, das bei der in der Regel völlig offenen Bauweise von 3D-Druckern nicht so einfach zu lösen ist. Wenn man damit lasert, verräuchert man sich den Raum, selbst wenn nur Papier geschnitten wird. Beim Lasern von Kunststoffen können dann noch giftige Gase hinzukommen. Also braucht man mindestens einen Schrank mit Absaugung und Filtern. Sehr viel Aufwand für einen Lasercutter mit einer Schnittfläche, die kleiner als ein DIN-A4-Blatt ist.

... und Sinnvolles

Doch keine Angst, es gibt auch sinnvolle Aufgaben: Interessant sind da zum Beispiel die Möglichkeiten, die das beheizte Druckbett bietet. Schon ganz ohne Umbau des (Alt-)Druckers lässt es sich hervorragend für Klebungen verwenden. Hintergrund: Viele 2-Komponenten-Kleber härten je nach Temperatur unterschiedlich schnell und vor allem unterschied-

Kurzinfo

- » 3D-Drucker als Schneide- und Zeichenplotter nutzen – ohne Firmware-Tausch
- » Ausprobiert: Smartphone-Display reparieren auf dem Heizbett
- » Musik spielen mit dem 3D-Drucker

Checkliste

(Schneideplotter-Umbau)



Zeitaufwand:
ein Wochenende



Kosten:
etwa 20 Euro (ohne Level-Sensor)



3D-Druck:
Halterung selber drucken

Alles zum Artikel
im Web unter
make-magazin.de/xar8

Material

(Schneideplotter-Umbau)

- » Schleppmesser Ersatzteile oder Klone für kommerzielle Schneidplotter, mit Wechselmessern, Bezugsquelle siehe Link
- » Schneideunterlage mindestens in der Größe des Druckbetts
- » Filament zum Druck der Halterung für das Schleppmesser
- » Gummiring und Schraube (optional) für die federnde Lagerung des Schleppmessers

Mehr zum Thema

- » Heinz Behling, Druckbettsensor nachrüsten, Make 6/19, S. 110
- » Online-Meldung zum RF2000
- » Online-Meldung Saucebot

lich fest aus. Ein Beispiel ist *UHU+ 300 endfest*. Dieser Kleber braucht bei 20°C etwa 12 Stunden zum Aushärten und erreicht dann eine Festigkeit von 12N/mm². Bei 100°C ist er bereits nach 10 Minuten ausgehärtet und die Festigkeit ist mit 25 N/mm² mehr als doppelt so hoch ①.

Für den Temperaturbereich dazwischen gelten entsprechende Werte. Wenn es die zu verklebenden Teile zulassen, kann der Drucker da so manche Klebung verbessern und/oder beschleunigen.

Das Heizbett ist aber auch recht nützlich, wenn es ums Gegenteil, also das Lösen von Klebungen geht. Besonders beim Austausch von verklebten Smartphone-Displays soll es hilfreich sein, so eine Idee im Internet (Adressen dazu siehe Kurzinfo-Link). Das haben wir ausprobiert. Den Bericht dazu lesen Sie ab Seite 23.

Keine richtig neue Aufgabe, sondern nur ein neues Druckmaterial wird mit dem *Chocolate Kit* der ukrainischen Firma *chokol3d* mög-

Anwendung von UHU plus endfest 300 ①

In den meisten Fällen genügt ein einseitiger Klebstoffauftrag. Die Teile können nach dem Klebstoffauftrag sofort zusammengefügt und ohne Druck fixiert werden. Passgenauigkeit: Bei ausreichendem Klebstoffauftrag überbrücken UHU plus Klebstoffe kleinere Unebenheiten in den Fügeflächen. Sie können – bei einer optimalen Fuge von 0,1 mm – auch fugenfüllend eingesetzt werden.

Klebefestigkeit UHU plus endfest 300 in Abhängigkeit der Temperatur

Die Aushärtung von UHU plus endfest 300 ist entscheidend von der Temperatur und Zeit abhängig. Die Aushärtung kann durch zugeführte Wärme beschleunigt werden.

Temperatur	Aushärtung	Festigkeit (Zugscherbeanspruchung für ALU)
20°C	12 Std.	ca. 12 N/mm2
40°C	3 Std.	ca. 18 N/mm2
70°C	45 Min.	ca. 20 N/mm2
100°C	10 Min.	ca. 25 N/mm2
180°C	5 Min.	ca. 30 N/mm2

Der Wärmeeinsatz wird durch die Temperaturbeständigkeit von Material und Klebstoff, sowie den Ausdehnungskoeffizienten der zu fügenden Materialien bestimmt. Abhängig vom Material können für den gleichen Klebstoff unterschiedliche Temperatur-/Zeitkombinationen gewählt werden. Im Allgemeinen gilt: Je höher die Temperatur, desto dünnflüssiger wird der Klebstoff und um so besser wird die Vernetzung und damit die Festigkeit des Klebstoffes.

2

Im Video spielt der 3D-Drucker Schach:



lich. Das Kit ist ein Druckkopf mit Extruder und einem beheizten Tank. In dem lässt sich Schokolade schmelzen und dann zu leckeren 3D-Objekten verdrucken (und später verdrücken). Allerdings ist der Spaß mit 350 Dollar nicht gerade billig. Außerdem muss man sich eine passende Halterung für den eigenen Printer noch selbst konstruieren.

Auch einige Spaß-Anwendungen finden sich im Netz: Da ist zum Beispiel die Möglichkeit, Musik mit dem Printer abzuspielen (näheres dazu auf Seite 24). Oder man baut den 3D-Printer zu einem Schachroboter um. Per Elektromagnet im ehemaligen Druckkopf und leicht veränderten Schachfiguren zusammen mit einer Software, die Schachbefehle in G-Code umwandelt, erspart Ihnen der Drucker die körperliche schwere Arbeit des Figurenbewegens 2.

Als sinnvollste Beschäftigungsmaßnahme erscheint uns jedoch die Erweiterung beziehungsweise der Umbau zu einem Schneidplotter. Daher haben wir diese Möglichkeit im Folgenden ausführlich beschrieben.

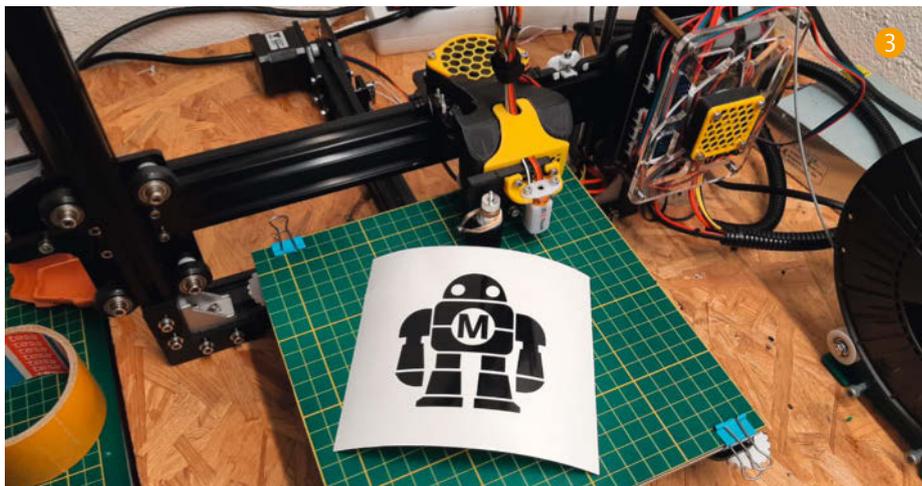
3D-Drucker als Schneidplotter

Ein praktischer, schnell durchgeführter und dazu noch vollkommen umkehrbarer Umbau macht aus einem 3D-Drucker einen Schneidplotter, mit dem man Folie für Aufkleber oder T-Shirts, Papier oder Karton präzise zuschneiden kann 3. Da man dafür keine neue Firmware braucht, lässt sich die Maschine problemlos auch weiterhin als 3D-Drucker nutzen – mit einem Tast-Sensor fürs Leveln des Drucktisches ausgestattet, braucht es nur wenige Handgriffe, um das Gerät vom Druck- auf den Schneidebetrieb umzurüsten.

Das Schleppmesser

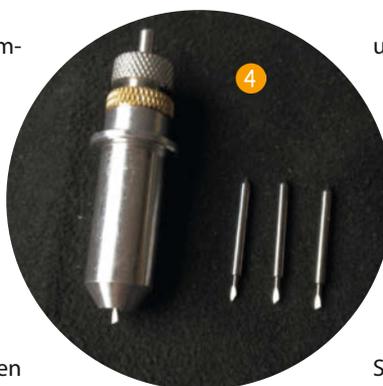
Um Folie, Papier, Karton und alles, was sich sonst noch plotten lässt, auszuschneiden, benötigt man selbstverständlich ein Messer. Da Messer aber normalerweise nur in eine Richtung schneiden, kommt hier ein spezielles *Schleppmesser* zum Einsatz: Durch eine leichtgängige Lagerung dreht sich die Schneide des Messers wie eine Windfahne immer in die Richtung, in die gerade geschnitten wird. Durch diese Konstruktion wird beim Schneidplotter-Upgrade vieles vereinfacht, da keine komplizierte Mechanik verwendet werden muss – es reicht, das Schleppmesser in X- und Y-Richtung über den Drucktisch zu bewegen wie sonst den Extruder auch.

Der Markt für Schleppmesser gestaltet sich sehr übersichtlich. Die meisten sind Klone, die als Ersatzteile für Marken-Schneidplotter gedacht sind. Für diesen Umbau wurde ein Schleppmesser verwendet, dass mit den



3

Plottern von *Roland* kompatibel ist 4, Bezugsquelle siehe Link in der Kurzinfo). Besonders schön daran ist, dass hier die Messeraufnahme aus solidem Aluminium besteht und sich das austauschbare Messer einfach durch Drücken eines Pins herausnehmen lässt. Neben dieser Aufnahme mit dem Lager darin benötigt man noch die Messer an sich, die den meisten Sets schon direkt beigelegt sind. Die Messer



4

unterscheiden sich lediglich in ihrer Spitzheit; in der Regel kann man zwischen 30°, 45° und 60° wählen, wobei eine Faustregel besagt, dass ein dickeres Material einen *größeren* Winkel voraussetzt, also das Messer *spitzer* aufs Werkstück auftritt. Dadurch kann es tiefer in das Material schneiden. Ein Set mit Aufnahme und Messern kostet etwa 10 bis 15 Euro und seine

Anschaffung spart viel Frustration bei Umbau und Betrieb des Druckers als Schneidplotter.

Die Halterung

Bei der Wahl der richtigen Halterung, mit der man das Schleppmesser am Drucker befestigt, wird man leider doch vor eine kleine Herausforderung gestellt: Durch die vielen Bauarten und Designs von 3D-Druckern gibt es im Netz entsprechend viel ausdrückbares Zubehör. Mit etwas Glück findet man eine Halterung für das gewählte Schleppmesser, die tatsächlich für den eigenen 3D-Drucker entworfen wurde. Für gängige Modelle wie die 3D-Drucker von *Creality* finden sich passende Designs auf *Thingiverse* und Co. In der Praxis kommt man aber oft nicht um das Entwerfen einer eigenen Halterung herum.

Der Konstruktionsaufwand dafür hält sich freilich in Grenzen, denn die Halterung ist eigentlich simpel aufgebaut **5**: Es braucht einen am Drucker festzuschraubenden Körper und darin eine Bohrung mit einem Durchmesser von (in meinem Fall) ca. 11,5mm, welche die Messeraufnahme hält und ein Herausrutschen der Halterung nach unten verhindert (meine Messeraufnahme hat praktischerweise eine Art Kragen im oberen Drittel als Anschlag).

Bei Vorlagen aus dem Netz für die Halterung findet sich an der Seite manchmal ein kleines Loch für eine Schraube. Durch Spannen eines Gummibands über das Schleppmesser und die Schraube lässt sich ein konstanter Druck auf das zu schneidende Material gewährleisten. Dies erhöht die Schnittqualität erheblich. Sollte das Schraubenloch oder ein Haken für so ein Gummi nicht vorhanden sein, lohnt sich eventuell eine Überarbeitung der Vorlage. Bild **6** zeigt das Schleppmesser an einem *Tevo Tarantula*.

Wer selbst entwirft, hat in noch einer Beziehung freie Wahl: Entweder kann man das Schleppmesser als einziges Werkzeug am 3D-Drucker befestigen und ihn dauerhaft zu einem Schneideplotter umbauen, oder man konstruiert die Halterung so, dass sie neben das Hotend passt, ohne den 3D-Druck-Betrieb zu behindern. Im Programm für die G-Code-Generierung lässt sich später ein Offset eintragen und man muss den Drucker nicht erneut umrüsten, wenn er wieder für seinen eigentlichen Zweck benötigt wird. Allerdings ist durch diesen Versatz die maximale Größe des zu schneidenden Materials etwas beschränkt. Generell zeigt sich, dass der Umbau des 3D-Druckers besonders gut bei offenen Druckern funktioniert, also solchen ohne Gehäuse. Nicht nur die Montage ist bei diesen Modellen einfacher, das Schleppmesser hat auch mehr Bewegungsfreiraum.

Ist die Halterung ausgedruckt, schraubt man gegebenenfalls das Hotend ab, dann bringt man die Halterung an. Anschließend kann das Schleppmesser eingesetzt werden. Sofern man die Gummiband-Strategie verwendet, sollte das Schleppmesser in der

Halterung frei nach oben und wieder zurück rutschen können. Andernfalls ist ein fester Sitz Pflicht.

Die Unterlage

Man darf nicht vergessen, dass wir es hier immer noch mit einem 3D-Drucker zu tun haben und jeder 3D-Drucker über ein, oft speziell beschichtetes, Druckbett verfügt. Um das Bett vor Kratzern und unser Messer vor dem Stumpfwerden zu schützen, bietet sich eine Extra-Unterlage an. Die besten Ergebnisse erzielte ich, nachdem ich mir eine Schneidmatte mit Kästchenmuster aus dem Ein-Euro-Shop auf die Größe meines Druckbetts zugeschnitten und mit Clips an den Seiten fixiert hatte. Für den Anfang funktioniert aber auch ein Stück dicke Pappe. Die Schneidmatte hat aber den unangefochtenen Vorteil, dass sie durch ihr Muster eine genaue Anordnung des zu schneidenden Materials ermöglicht, wie auf Bild **3** zu sehen ist, was sich später als sehr hilfreich erweisen wird.

Die Software

Nun muss noch eine Grafik zu geeignetem G-Code verarbeitet werden, damit sie geschnitten werden kann. Dazu nehme ich das kostenlose Vektorzeichenprogramm *Inkscape*, zusammen mit dem bekannten Plug-in *JTP Laser Tool* (Download siehe Link in der Kurzinfor). Dieses Plug-in ist zwar eigentlich für Lasercutter entwickelt worden, jedoch sind die Grundfunktionen und die Erweiterbarkeit genau das, was wir für den Plotter brauchen. So lassen sich beispielsweise auch ausgewählte Achsen vor dem Start des Plottens homen, indem man seinen eigenen G-Code-Header in das Plug-in mit einbindet.

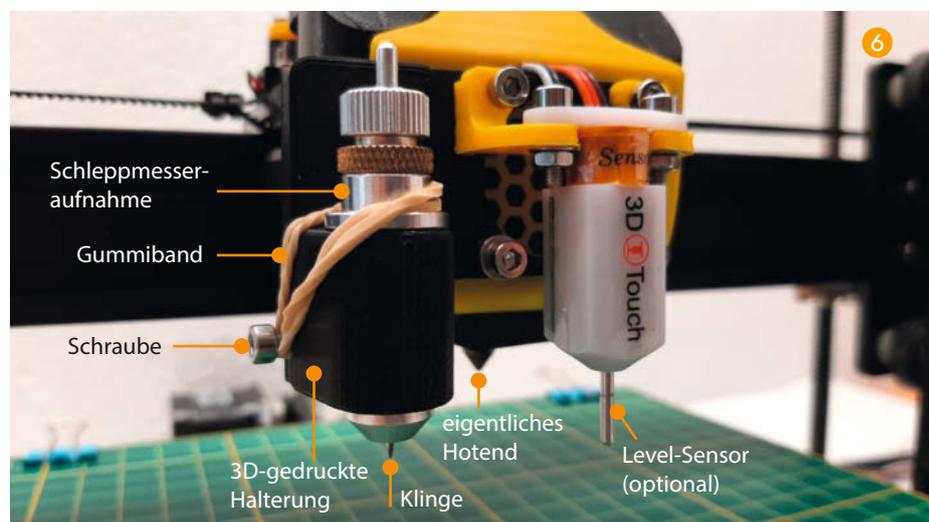
Zum Installieren des Plugins lädt man sich zuerst *Inkscape* und das Plug-in herunter. Man muss allerdings darauf achten, dass man sich

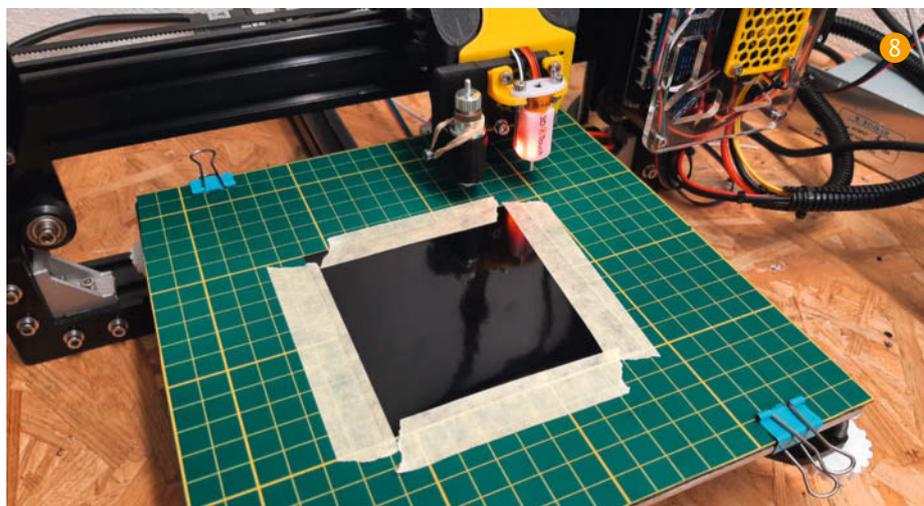
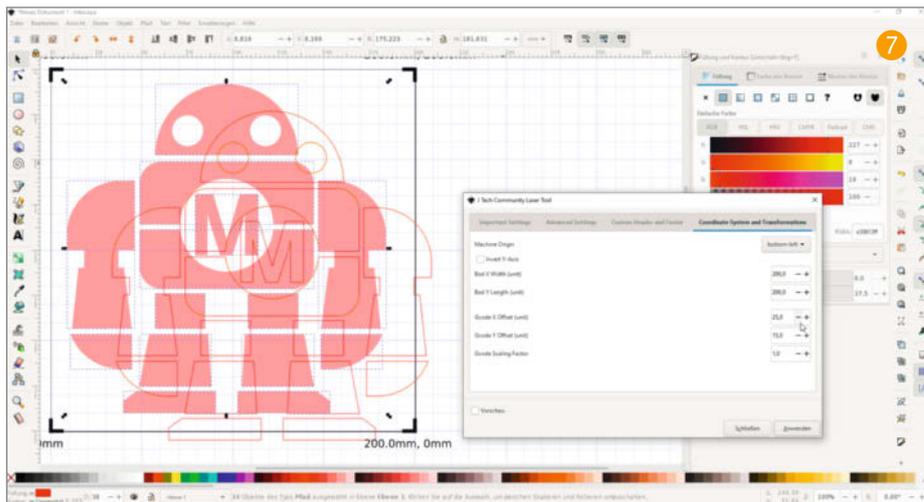


einen Release herunterlädt, in der die Library *svg_to_gcode* integriert ist, und das Repo nicht einfach als ZIP klonet (mehr dazu siehe Link in der Kurzinfor). Sobald *Inkscape* installiert wurde, kann der Inhalt des Ordners *laser*, der sich im ZIP-Archiv *JTP-Laser-Tool_vX.X* befindet, in den *extensions*-Ordner von *Inkscape* abgelegt werden. Dieser befindet sich unter *Inkscape\share\inkscape\extensions*. Nach einem Neustart von *Inkscape* ist das Plug-in betriebsbereit.

Ein erster, sinnvoller Schritt ist, über *Datei/Dokumenteinstellungen* die Seite einzurichten. Die Seitengröße sollte der Größe des Druckbetts in Millimetern entsprechen. Lieber doppelt kontrollieren, ob die Einheit auf Millimeter gesetzt wurde, da ansonsten das Tool mit der Skalierung durcheinander kommt! Im selben Dialog kann man unter *Gitter* ein Raster auf die Seite zeichnen lassen, welche das Ausrichten der Motive vereinfacht: Hat das Raster die gleichen Maße wie das der Schneideunterlage, erhält man automatisch eine Art Projektion der Unterlage in *Inkscape*.

Dann kann eine SVG-Datei oder ein Bild auf die Arbeitsfläche per *Strg+V* oder *Drag & Drop* eingefügt und platziert werden. Bitte beach-





ten: Der G-Code wird absolut zur Arbeitsfläche erzeugt. Liegt das Motiv mittig, so wird der Plotter später auch in der Mitte des Druckbetts schneiden.

Das Plug-in

Anschließend startet man das Plug-in über *Erweiterungen/Generate Laser Gcode/JTech Community Laser Tool*. Im neuen Fenster lassen sich die Einstellungen anpassen:

Unter dem Reiter *Coordinate System and Transformations* können die Größe des Arbeitsbereichs und der Offset des Messers festgelegt werden, bei *Custom Header and Footer* legt man eigenen G-Code am Start und am Ende des Schneidvorgangs fest, bei *Advanced Settings* geht es vor allem um die Eigenschaften des eigentlichen Werkzeugs und unter dem Reiter *Important Settings* werden die projektspezifischen Einstellungen getroffen. Das zur Übersicht, jetzt zur Konfiguration im Detail.

Die *Laser Power/Laser Off Command* im Menü *Advanced settings* verweisen auf die Laser-Herkunft des Plugins. Die dort standardmäßig eingetragenen G-Codes M3 und M5

schalten das Werkzeug ein oder aus. In unserem Fall brauchen wir hier G1 Z0 und G1 Z3. Dadurch wird das Schleppmesser abgesenkt und angehoben. Die anderen Einstellungen dort sind ebenfalls für einen Laser gedacht und können so bleiben. Anschließend kann man noch das Anzeigen der Schnittkante (*Draw Debug*) in Inkscape ein- oder ausschalten.

Hat man seinen 3D-Drucker um die Plotterfunktion erweitert, dann sitzt das Schleppmesser nicht an der Stelle des Hotends. Den Versatz zum eigentlichen Nullpunkt des Werkzeugschlittens kann man auf dem Reiter *Coordinate System and Transformations* bei *Gcode X/Y Offset* eintragen. Der Versatz wird bei Halterungsmodellen, die man aus dem Netz lädt, oft vom Ersteller mit angegeben. Sollte das nicht der Fall sein reicht es aber, sich an den tatsächlichen Versatz anzunähern. Oftmals genügt einfaches Messen mit einem Lineal, es kommt nicht auf den halben Millimeter an. Vorsicht: Der Versatz wird auch in der Vorschau angezeigt, da die Schnittkante dann nicht mehr auf dem Modell liegt. Davon sollte man sich nicht verwirren lassen 7.

Jetzt kann der Reiter *Important Settings* mit persönlichen Einstellungen versehen werden, etwa die *Travel-* und die *Cutting-*Geschwindigkeit, also das Tempo für Schneide- und Fahrvorgänge. Die Werte werden in mm/min angegeben, was den ein oder anderen sicher verwirrt, da beim 3D-Druck mm/s üblich ist. Jedoch kann man seine mm/s-Werte durch Multiplizieren mit 60 kinderleicht auf mm/min umrechnen. Während für die *Travel-*Geschwindigkeit der Wert verwendet werden kann, den man schon beim 3D-Druck nutzt, muss man sich bei der Arbeitsgeschwindigkeit nach dem Material richten: Vinylfolie lässt sich meiner Erfahrung nach mit 1800 mm/min schneiden, Karton mit 1200 mm/min und festeres Papier mit 1500 mm/min. Die Geschwindigkeiten sind jedoch nur Erfahrungswerte und sind sicher von Drucker zu Drucker unterschiedlich.

Darunter finden sich die Einstellungen *Passes* und *Pass Depth*, welche angeben, wie oft die Schnittkante entlang gefahren werden soll und wie tief pro Runde geschnitten wird. Diese Einstellungen können für unseren Zweck nicht benutzt werden, da wir das Schneidmesser immer aktivieren, in dem wir es auf eine konkrete Z-Höhe setzen, was leider die *Pass Depth* überschreibt. Einen Workaround gibt es aber: Man kann seinen G-Code für zwei Schnittvorgänge einfach zwei mal hintereinander ausführen.

Falls gewünscht, kann man eine G-Code-Datei mit Befehlen anlegen, die beim Start ausgeführt werden sollen. Hierfür bietet sich beispielsweise ein *Homing* der X- und Y-Achsen an (G28 X Y). Diese Datei kann dann einfach bei *Custom Header and Footer* eingebunden werden, ebenso eine für den Abschluss.

Zum Schluss kann man bei *Important Settings* einen Namen und Speicherort für die erzeugte G-Code-Datei angeben. Durch einen Klick auf *Anwenden* wird der G-Code dort gespeichert und das Motiv auf der Arbeitsfläche mit einer Vorschau der Schnittkante überdeckt. Hier lohnt es sich, doppelt hinzuschauen: Wer hier einen Fehler findet, hat eventuell eine SVG-Datei mit „verknöteten“ Punkten geladen und muss in der Vorlage noch mal aufräumen. Wer keine Schnittkante sieht, muss sein Motiv noch in einen Pfad umwandeln (*Pfad/Objekt in Pfad umwandeln*).

Ist der Schritt gemeistert, wird der G-Code wie sonst auch auf den Drucker übertragen, in der Regel also auf die SD-Karte kopiert und die in den Drucker gesteckt.

Das Material

Das Material, egal ob Vinylfolie, Pappe oder Karton, legt man nun an die Stelle, die man vorher in Inkscape festgelegt hat. Wenn das Material groß genug ist, kann es mit den Clips an den Ecken des Druckbetts fixiert werden.

Oft ist das Material aber zu klein. Dann empfiehlt es sich, es mit etwas Malerkrepp an den Seiten auf die Schneideunterlage zu kleben **8**. Auch Neodyn-Magnete, die man ober- und unterhalb des Druckbetts anordnet, können die Folie sicher halten.

Mit Material, das direkt von einer Rolle kommt, muss man allerdings aufpassen: Dadurch, dass es aufgerollt wurde, ist das Material leicht nach oben gewölbt. Aber je planer es auf dem Bett aufliegt, desto besser ist das Ergebnis. Gegebenenfalls kann man das Material vor dem Plotten ein paar Stunden mit einem schweren Buch beschweren oder, wenn gar nichts mehr hilft, auf eine selbstklebende Schneidematte aus dem Plotterbedarf zurückgreifen (zur DIY-Alternative siehe Link).

Endlich schneiden!

Jetzt setzt man das Messer der Wahl ein. Die Aufnahme ist zumindest bei meiner Schleppmesseraufnahme magnetisch und das Messer hält problemlos, nachdem es eingesetzt wurde. Vorsicht! Die Messer sind **sehr scharf**.

Bevor es jetzt endlich losgehen kann, muss jedoch noch das Messer in Z-Richtung ausgerichtet werden. Dazu lockert man den gelblichen Befestigungsring oben an der Messeraufnahme und schraubt den kompletten Schneideapparat etwas weiter nach unten oder oben. Danach lässt sich der Ring wieder fixieren. Das Messer sollte so ausgerichtet sein, dass es im abgesenkten Zustand etwa bei Vinylfolie die eigentliche Folie schneidet, das Trägermaterial aber nicht. Bei allen Materialien, die ganz durchgeschnitten werden sollen, lässt es sich nicht vermeiden, dass die Schneideunterlage ein paar Kratzer abbekommt. Das ist aber völlig normal. Zusätzlich kann das Ergebnis verbessert werden, wenn das Druckbett vorab präzise gelevelt wurde, beispielsweise mit einem Kassenbon als Maß für den Abstand.

Für die ersten Schnittversuche sollte die Klinge 1–2mm aus der Aufnahme heraus schauen und ein Blatt Papier geschnitten werden. An die genaue Ausrichtung des Schneidmessers in Z-Richtung tastet man sich dann langsam von oben heran und macht

ein paar Probeschnitte, um einem Unfall vorzubeugen. Sind alle diese Vorkehrungen getroffen, kann der echte G-Code auf dem Drucker mit ausgewählt werden und der Schneideplotter Spaß losgehen!

Wie man sieht, lässt sich ein 3D-Drucker für kleines Geld dauerhaft zu einem Plotter umbauen oder sogar zu einem Multifunktionsdrucker mit Plot-Funktion upgraden – in diesem Fall nimmt man zum 3D-Drucken einfach die Schneidematte und das Gummiband ab und das Schleppmesser heraus und levelt anschließend die Z-Achse (sofern das nicht dank Sensor automatisch passiert). Zwar muss man sich ein kleines bisschen in die Steuerung reinfummeln und seine optimalen Einstellungen experimentell finden, aber nach ein bisschen Ausprobieren läuft das Plotten sehr routiniert ab. Wem das Schneiden zu langweilig wird, der kann auch einen Stift einsetzen und auf gleiche Weise Motive zeichnen lassen, also nochmal eine Funktion mehr erhalten. Kurzum: Ein schönes, praktisches Projekt, das sich an einem Wochenende umsetzen lässt!

Ausprobiert: Smartphone-Reparatur

Typische Nach-Einkaufssituation: Fast alles liegt auf dem schon gut gefüllten Küchentisch. Aber auch die letzten Lebensmittel müssen da noch drauf. Also schiebt man alles etwas zur Seite. Es macht Ssst, dann Peng und Klirr. Ohne Gegenwehr folgte das Smartphone (Modell Sharp SH-A01) der Schwerkraft und fand zielsicher mit einer Ecke den Kachelboden. Und schon war die Spider-App installiert, also das Display gebrochen **9**.

Was tun? Läden, die eine Display-Reparatur anbieten, gibt es viele. Aber mein Exemplar war nicht gerade ein Luxusmodell. Und da überlegt man sich, ob man 60, 70 oder noch mehr Euro in die Wiederherstellung steckt. Wegwerfen? Da rebelliert mein Maker-Herz. Also suche ich eine Möglichkeit, das Display in Eigenregie preiswert zu tauschen. Ein passendes Ersatzteil war sogar bei Ebay für gerade mal 25 Euro inklusive Werkzeugsatz zu haben **10**.

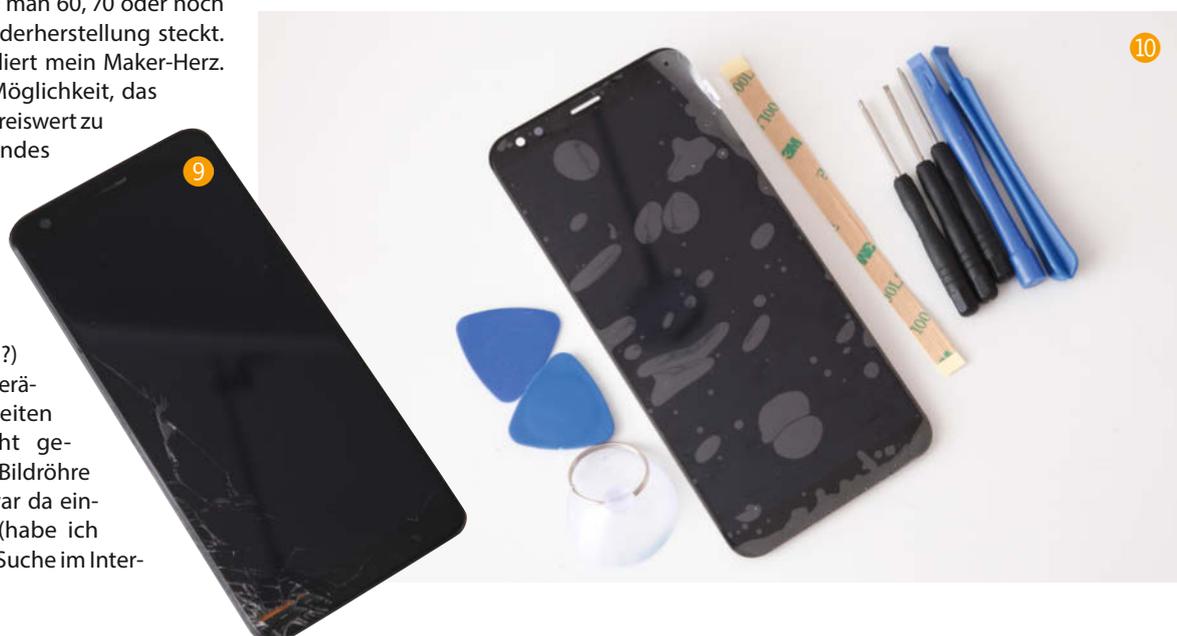
Doch leider sind in modernen (wirklich?) Telekommunikationsgeräten die Anzeigeeinheiten eingeklebt und nicht geschraubt. Selbst eine Bildröhre in alten TV-Geräten war da einfacher zu wechseln (habe ich schon gemacht). Eine Suche im Inter-

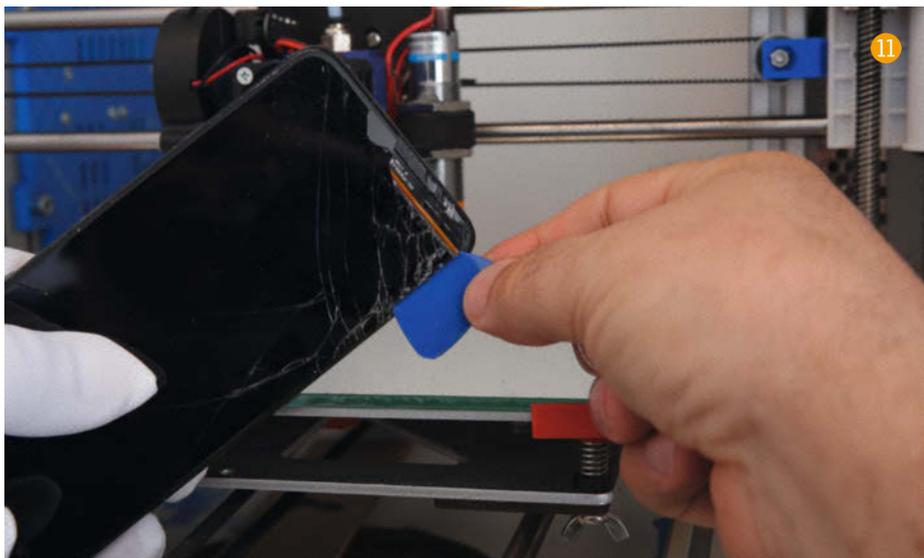
net ergab dann den Tipp: Mit dem Heizbett eines 3D-Druckers kann man das Display schonend erwärmen und so den zu seiner Befestigung benutzten Thermo-Klebstoff lösen. Das habe ich hier ausprobiert.

Das Smartphone-Display wurde zunächst von herausstehenden Glassplittern befreit, die eine plane Auflage auf dem Heizbett und somit den Wärmeübergang behindert hätten. Dann kam es mit dem Display nach unten auf das Druckbett. Die im Internet vorgeschlagene Temperatur von 60° erwies sich als wir-

kungslos. Erst bei 80° gab der Klebstoff des Displays nach. Mit den Kunststoff-Werkzeugen aus dem Kit konnte ich das Display in mehr oder weniger großen Teilen ablösen **11**.

Probleme machte nun das Display-Kabel, das unter einem Alublech verschwand und offenbar verklebt war. Folglich musste ich auch die Rückseite des Smartphone-Gehäuses lösen, die sich zum Glück als lediglich eingeklippt herausstellte. Eingeklebt war aber wiederum der Lipo-Akku, und zwar genau auf dem Display-Kabel **12**.





Und der Akku saß bombenfest. Also kam nochmals das beheizte Druckbett zum Einsatz. Mit dem Akku nach oben legte ich das Smartphone-Skelett auf die Platte und ließ es etwa 5 Minuten durchheizen. Trotzdem ließ sich danach der Akku nur mit etwas Gewalt und deutlichem Verbiegen vom Untergrund soweit lösen, dass ich das alte Displaykabel heraus- und das neue hineinbekam. Beim Betrachten des krummen Akkus wurde mir allerdings etwas mulmig, schließlich sind Lithium-Akkus bekannt dafür, dass sie mechanische Überbeanspruchung übel nehmen können **13**.

Das alte Kabel konnte nun mit dem Display herausgezogen und das neue eingesetzt werden. Blieb noch die Befestigung des neuen Displays. Dazu gab es im Kit mehrere, etwa 1,5mm-breite doppelseitige Thermo-Klebebänder. Die setzte ich auf den schmalen Gehäuserahmen auf und zog mit einer Pinzette vorsichtig die Schutzstreifen ab. Dann legte ich das neue Display auf die Streifen, und das Smartphone kam für etwa 2 Minuten erneut auf die Heizplatte (Display nach unten). Dadurch sollte der Kleber weich werden und sich

mit Display und Gehäuse verbinden. Anschließend nahm ich das Smartphone von der Platte und legte es auf eine kalte Fensterbank zum Auskühlen. Nach etwa einer Minute kam der Test: Das Display saß fest und nach Einschalten erschienen die üblichen Meldungen. Mein Smartphone war repariert **14**. Ich musste nur noch die Rückwand wieder aufstecken.

Insgesamt war die Prozedur aber nicht gerade einfach: Insbesondere hatte ich ständig die Befürchtung, den Akku bleibend zu beschädigen. Und wer weiß, was andere Smartphone-Modelle da noch an Überraschungen auf Lager haben. Daher kann ich die Eigenreparatur eines Displays für niemanden empfehlen. Falls Sie es aber selbst machen möchten: Ein 3D-Drucker ist dabei sehr hilfreich. Sie handeln jedoch auf eigene Gefahr. Insbesondere weiß ich nicht, ob der Akku nicht eventuell nach einiger Zeit doch noch Reaktionen zeigt.

Musik-Player

Es gibt noch weitere Zweckentfremdungen für 3D-Printer, die nicht unbedingt sinnvoll sein müssen, aber Spaß machen können. Mit

dem kostenlosen *Midi-G-Code-Konverter* von *UltimateSolver* zum Beispiel lassen sich Musikdateien im Midi-Format in Steuerdateien für 3D-Drucker umwandeln. Die Schrittmotoren des Druckers werden dann durch die darin enthaltenen G-Code-Befehle so angesteuert, dass sie hörbare Töne von sich geben.

Dazu müssen Sie außer der Musikdatei lediglich angeben, wie viele Motoren der Drucker hat, wie groß die maximalen Verfahrswege sind und wie viele Schritte pro mm Weg die Motoren machen. Danach folgt die Umwandlung und Sie können die Steuerdatei auf Ihren Computer überspielen. Sie lässt sich dann wie ein Druckauftrag an den Drucker schicken beziehungsweise mittels Speicherkarte oder USB-Stick im Drucker einlesen **15**.

Nach dem Start-Befehl wird jedoch kein Kunststoffteil produziert, sondern Musik. Allerdings muss man schon recht genau hinhören, denn sehr laut ist sie nicht. Für Drucker mit Silent-Antrieb ist die Sache ungeeignet, denn deren Motoren arbeiten bei der Musikwiedergabe nahezu geräuschlos.

Saucebot

Liebhaber von Hotdogs werden mit dem *Saucebot* noch eine Verwendungsmöglichkeit für einen 3D-Drucker zu schätzen wissen. Mittels zweier Dosierpumpen, einem modifizierten Druckkopf sowie einer Halterung für Hotdogs kann er Ihr Fastfood mit Senf und/oder Ketchup verfeinern **16**.

Das wäre sicher der Hit in jedem Partykeller oder beim Grillfest. Ob man den danach mit Ketchup verschmierten Drucker dann aber noch für seine ursprüngliche Aufgabe benutzen kann? Finden Sie es raus! Viel Spaß dabei! Und wenn Sie noch einen Job für 3D-Drucker-Rentner kennen, mailen Sie uns. —hgb/pek



Midi → G-Code Konverter 15

MIDI → G-CODE

Version	2.7.3
Beschreibung	Mit diesem Konverter können Musik-Dateien (Midi-Dateien) in G-Code (z.B. für CNC-Fräsen) umgewandelt werden. Mit jeder CNC-Fräse, die mit Schrittmotoren angesteuert wird, ist es somit möglich, Musik zu erzeugen.
Statistik	Der Konverter wurde bisher mehr als 123.200 mal verwendet.
Auswahl der Midi-Datei (*.mid)	<input type="button" value="Durchsuchen..."/> Keine Datei ausgewählt. (max. 800 kB)
Verarbeitung	<input type="button" value="Datei analysieren"/>

ODER

Beispiel laden	<input type="button" value="Ludwig van Beethoven: Für Elise"/> Youtube
	<input type="button" value="Scott Joplin: The Entertainer"/> Youtube
	<input type="button" value="Stille Nacht, Heilige Nacht"/> Youtube



Neue Version!

Facebook @elegoo_official
Twitter @elegoo_official
Instagram @elegoo
ELEGOO

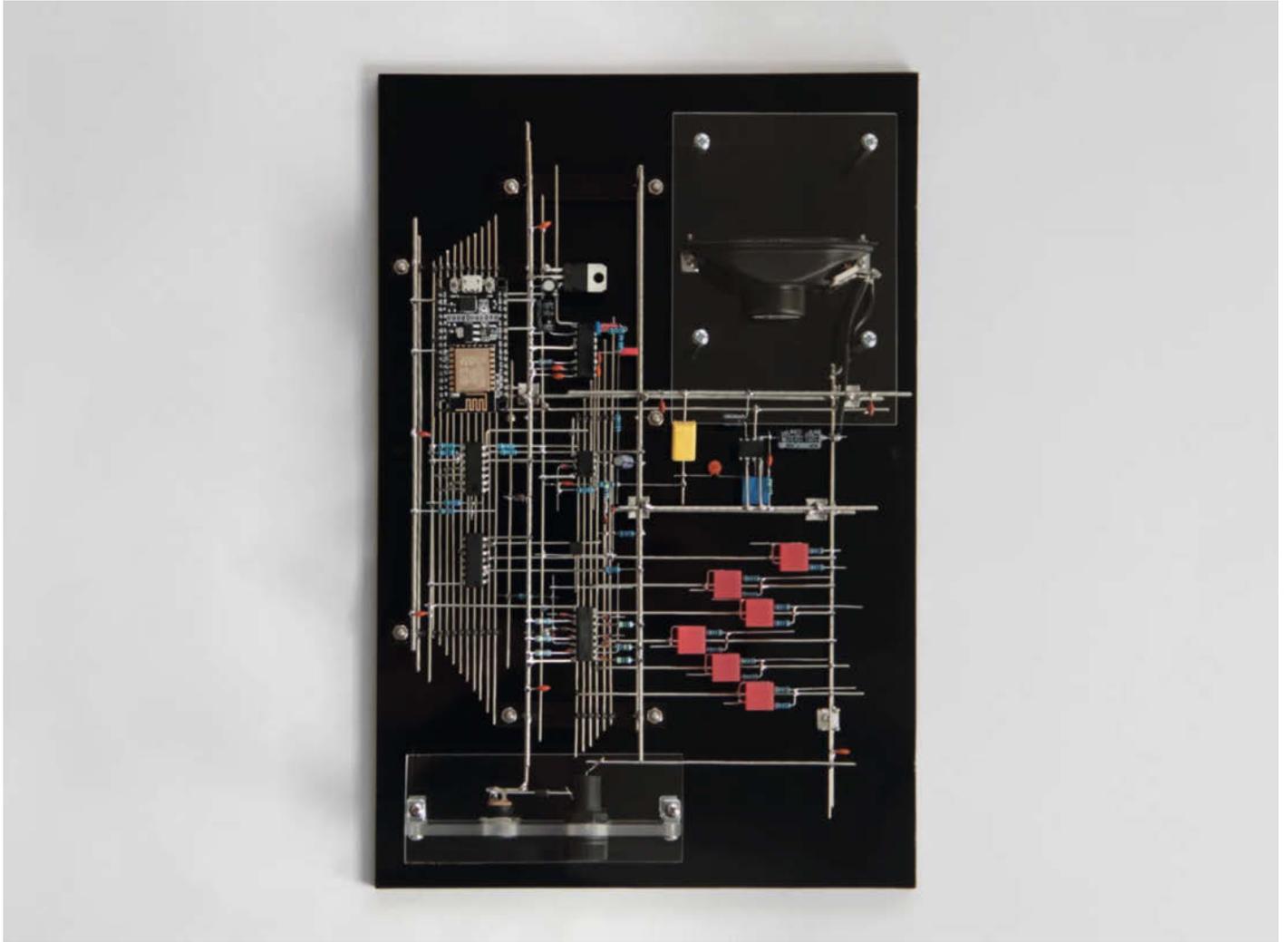
Elegoo Mars3 **Ultra 4K**

MONO LCD Desktop 3D-Drucker

Präferkt für Anfänger und Experten

- ◆ XY Auflösung: 4098 x 2560
- ◆ Druckgeschwindigkeit (Belichtungszeit): 1,5 - 3 s pro Schicht
- ◆ Bauraum: 89,6 x 143,36 x 165 mm
- ◆ Produktgröße: 227 x 227 x 438,5 mm
- ◆ 6,6" Ultra 4K LCD
- ◆ Neue Benutzeroberfläche auf 3,5" Touchscreen
- ◆ Einzigartiges Kühlungskonzept

Model by Archvillain Games

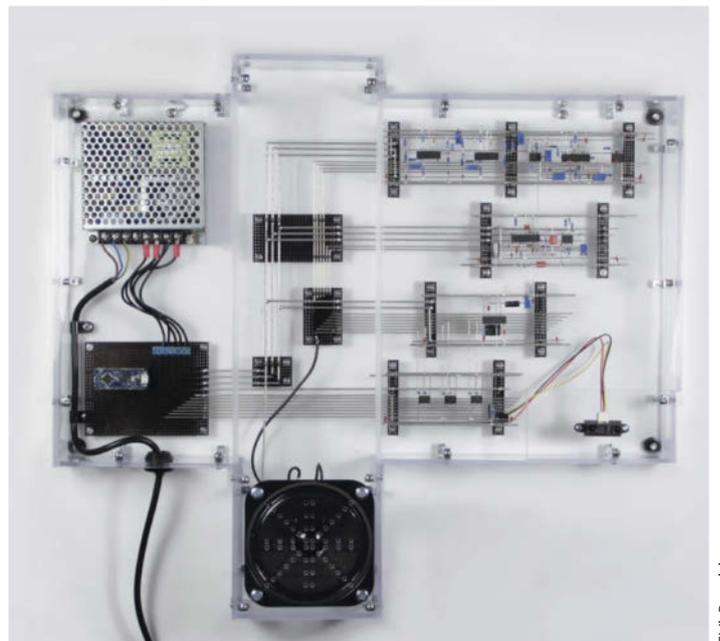


Was uns inspiriert

Was inspiriert Dich?

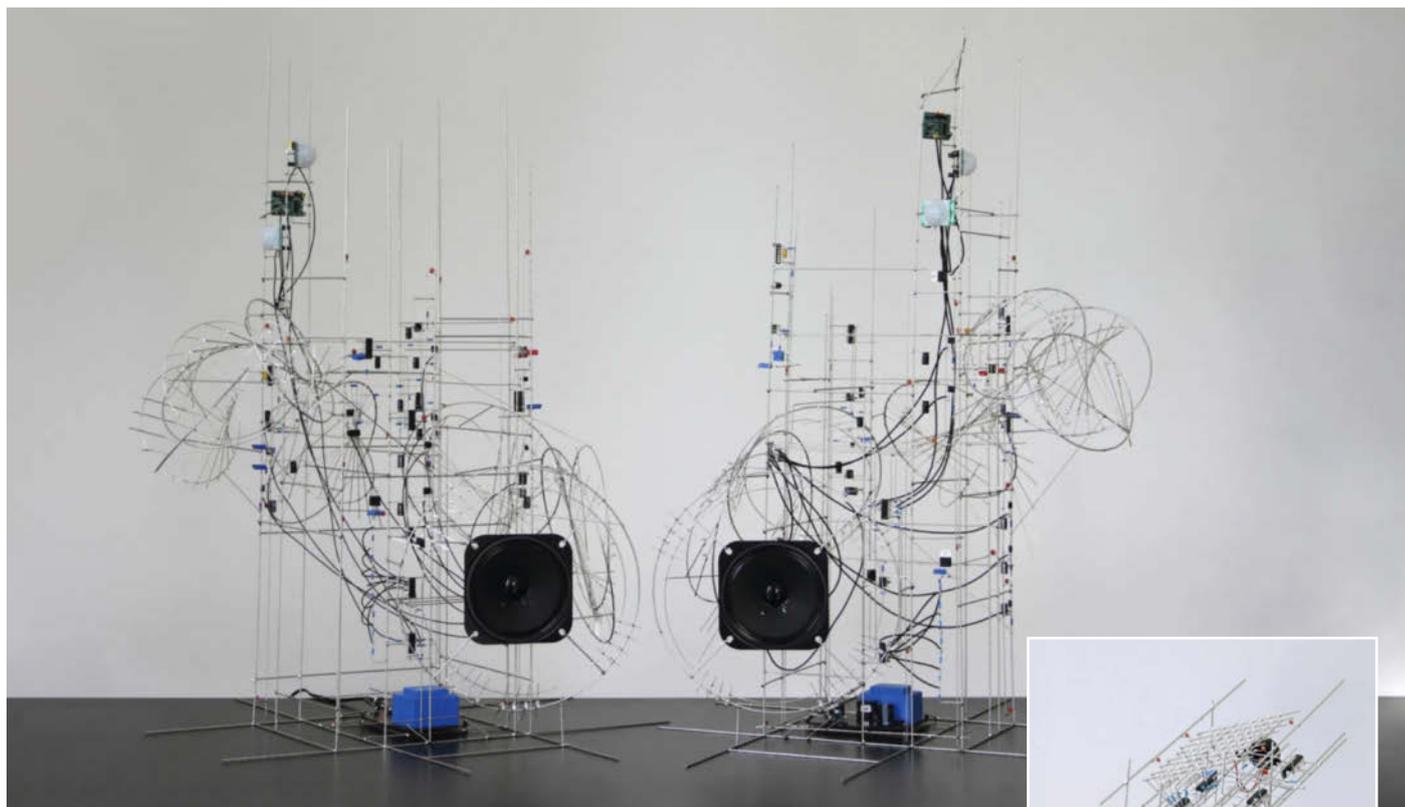
Wir freuen uns über Vorschläge an:

mail@make-magazin.de



plubee

Eirik Brandal



gulfs

Sphärische Skulpturmusik

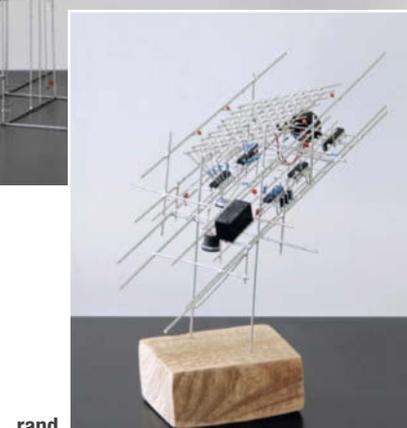
Schade, dass das Make-Magazin noch nicht mit integrierten Lautsprechern ausgeliefert wird, denn die betörenden Klänge der Kunstwerke von Eirik Brandal enthalten wir Ihnen nur ungerne vor. Die Skulpturen bestehen aus fragil wirkenden Schaltkreisen, deren integrierte Synthesizer sanfte Tiefseeesänge surren und LEDs melodisch aufleuchten lassen wie einen Fischschwarm.

Viele von Brandals Konstrukten sind interaktiv: *Gulfs* beispielsweise ist ein Duo von elektronischen Klang- und Lichtskulpturen, die auf Bewegung reagieren. Sie sind mit jeweils drei Infrarotsensoren ausgestattet, die es ihnen ermöglichen, den Raum um sie herum in unterschiedlichen „Zonen“ zu erfassen. Wenn sich durch diese Zonen ein Mensch bewegt, ändern sich Ton und Licht dynamisch.

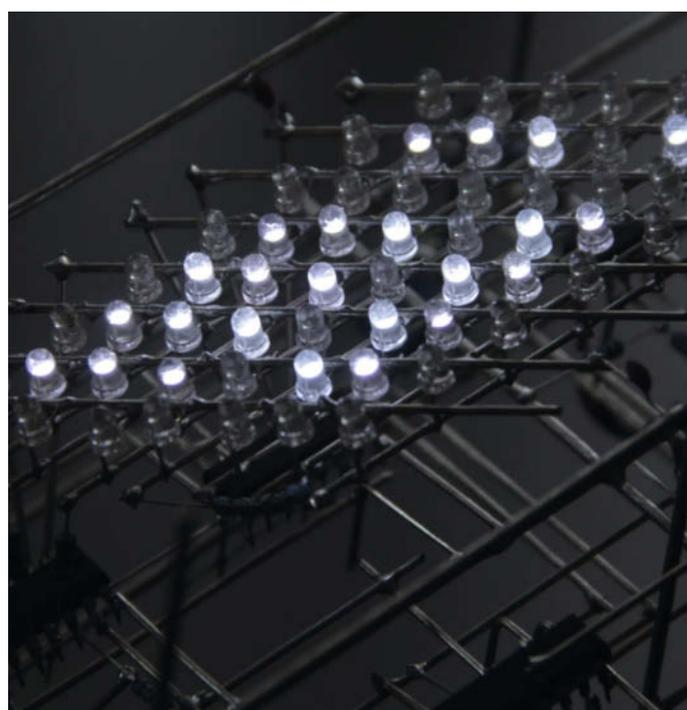
Auch *plubee* ist semi-interaktiv: Die Klangskulptur erzeugt kontinuierlichen Klang, kann aber dank ihres integrierten Näherungssensors durch menschliche Interaktion unterbrochen werden. Der Klang wird mit *subtraktiver Synthese* erzeugt (eine auf oberwellenreichen Oszillatoren und nachfolgende Filter beruhende Technik, einst von Bob Moog entwickelt), die von einem *Arduino Nano Every* gesteuert wird.

Rand dagegen klackert und leuchtet hypnotisch vor sich hin. Diese Skulptur verwendet pseudozufällige Binärsequenzen, um Ton- und Lichtmuster zu erzeugen. Es ist überraschend schwer, die Augen von diesem Schauspiel abzuwenden und nicht nochmal auf *play* zu klicken. —*rehu*

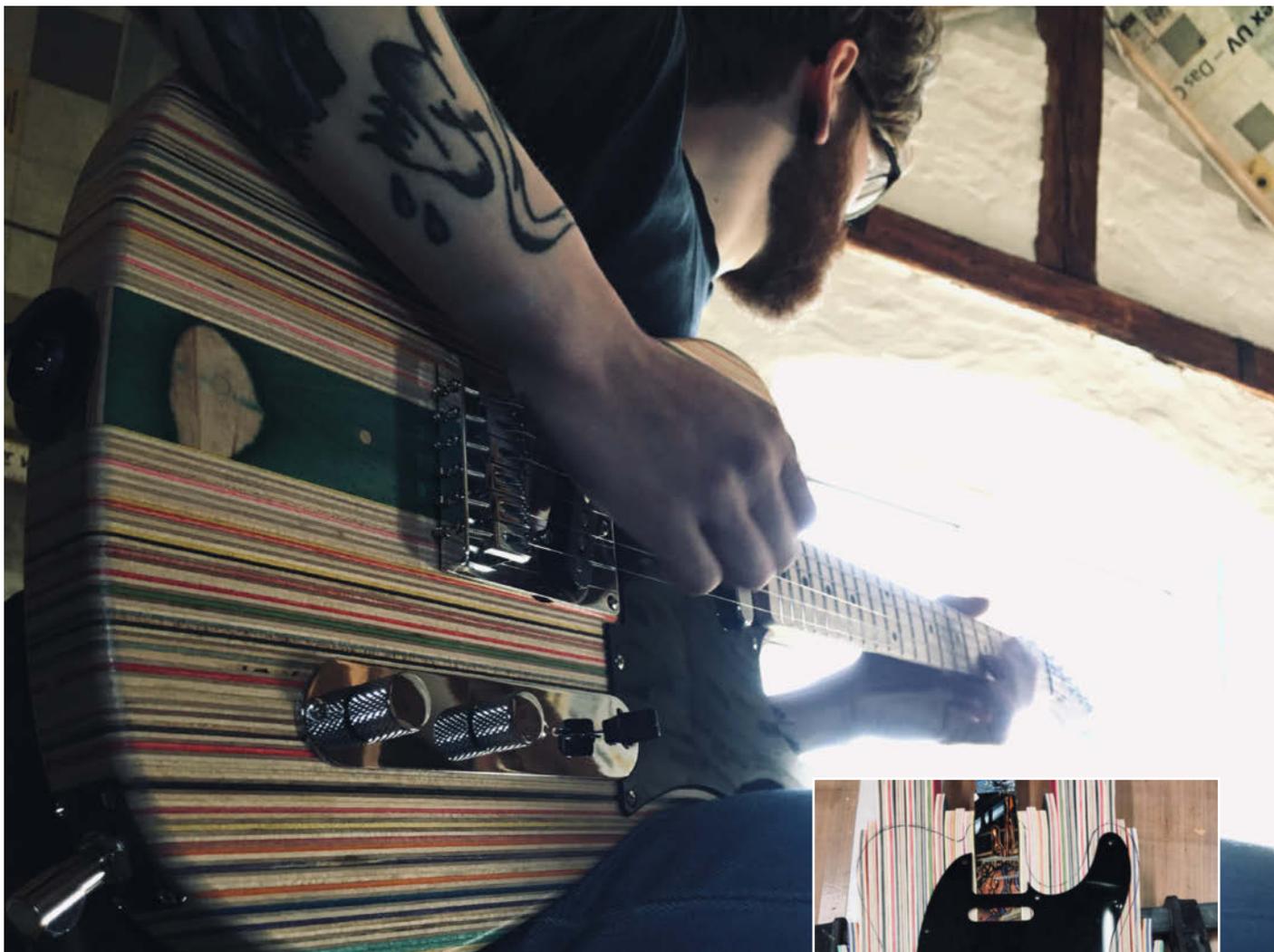
► eirikbrandal.com



rand



rand



Live-Making im Musikvideo

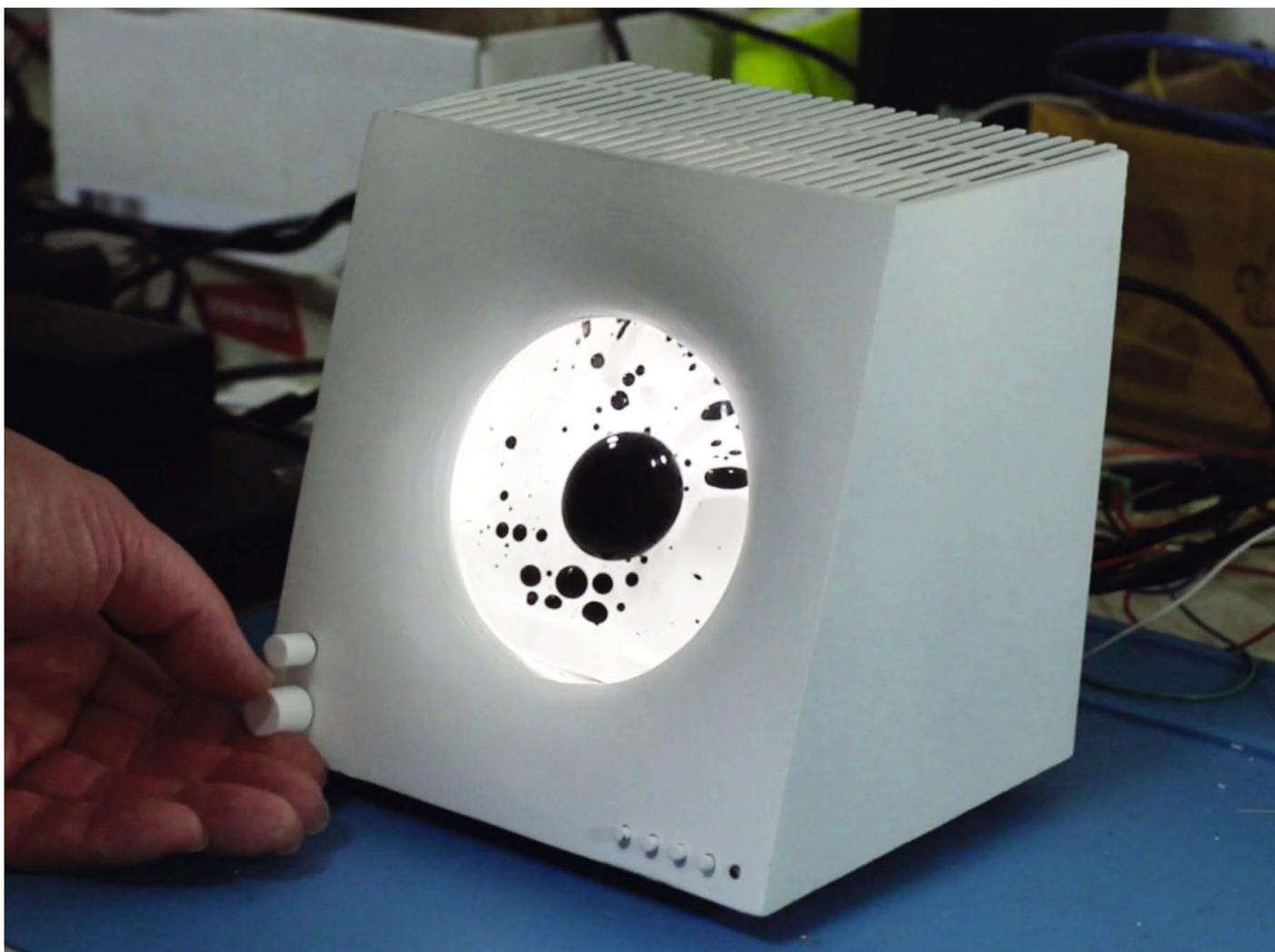
Die Oldenburger Band CATAPULTS macht nicht nur Musik, sondern auch coole Makerprojekte: Nachdem Bassist Lars während einer Probe bei einem Kick-Flip sein Skateboard zerbrochen hat, kam der Band die Idee, aus den alten Skate-Decks eine Gitarre zu bauen. Gesagt, getan: Entstanden ist eine funktionierende E-Gitarre, mit der gleich ein ganzes Album der Band aufgenommen wurde. Dem Bauprozess kann man auf YouTube im Musikvideo zu „If You Don't Matter, Nothing Does“ beiwohnen.

Um das Holz vorzubereiten, hat Sänger Joost das Griptape von den Boards entfernt und den Kleber abgeschliffen. Dann wurden alle Bretter auf die gleiche Größe gekürzt und so zusammengeleimt, dass ein Farbverlauf entsteht. Anschließend hat die Band den Korpus gründlich geschliffen und mit einer Schablone die berühmte *Telecaster*-Form ausgesägt. Für den Gitarrenhals haben sie den Hals einer alten *Telecaster*-E-Gitarre wiederverwertet. (Selber bauen hätte ihre Gitarrenbau-Kompetenz dann doch überstiegen.) Am Ende wurde der Korpus nochmal geschliffen und die Elektronik verlegt. So sieht das Ergebnis nicht nur stylish aus, sondern klingt auch richtig gut. —rehu

► youtube.com/watch?v=aaCAavFTpEk



CATAPULTS



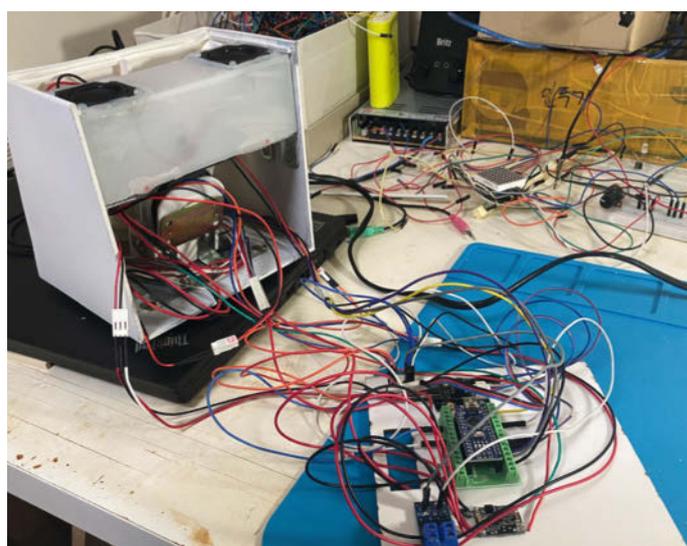
Lautsprecher schunkelt mit

Experimente und Projekte mit Ferrofluid, einer öligen, magnetischen Flüssigkeit, sind immer wieder faszinierend. Bei diesem Hingucker von Dakd Jung tanzt ein Ferrofluid-Klecks zur Musik in einem selbstgebauten Bluetooth-Lautsprecher. Daran werden sich die Gäste bestimmt erinnern! Das Ferrofluid-Behältnis ist eine einfache, flache Glasflasche, hinter der ein Arduino-gesteuerter Elektromagnet sitzt.

Laut Jung hängt der Erfolg maßgeblich vom richtigen Auftrieb im Medium (Wasser) und von der Sauberkeit des Glasbehälters ab. Schon kleinste Verunreinigungen lassen das Ferrofluid an den Wänden haften. Kritisch sei auch die Stärke des Magnetfelds, damit die Ferrofluid-Blase nicht in zu kleine Tropfen zerfällt. Das Gehäuse mit dem Stereo-Lautsprecherpaar wurde mit der CAD-Software *Fusion 360* designt. Vollständige Pläne will der Autor herausrücken, sobald seine Tests abgeschlossen sind. Das Projekt dokumentiert Jung derweil auf hackaday.io.

—cm

► hackaday.io/project/179136



Dakd Jung



The next Level: Mach Dein Smart Home wirklich smart

Du hast das schlaueste Smart Home im ganzen Land? Zeig es uns – beim Wettbewerb von Make und AVM auf Make Projects!



von Rebecca Husemann

Temperatursensoren steuern Deine Heizung und Rollläden? Dein Smart Home schaltet automatisch das Licht und die Stereoanlage an, wenn Du heim kommst? Dein Türschloss öffnet per Fingerprint? Die Katzenklappe erkennt das Gesicht Deiner Katze? Deine Katze mag Dein Smart Home lieber als Dich, weil Du die Katzenfütterung automatisiert hast?



Dann bist Du hier richtig! Präsentiere Dein Projekt beim Wettbewerb *Smart Home – The next Level* auf Make Projects! **Make**

sucht in Kooperation mit AVM Smart Home-Lösungen, die

wirklich nützlich sind. Zeigt uns Projekte, die übers WLAN kommunizieren und verschiedene Sensoren und Aktoren nutzen und miteinander verknüpfen.

Es ist nicht notwendig, Fritz!-Produkte einzusetzen; Ihr habt die freie Wahl, welche Hard- und Software Ihr benutzen wollt.

Uns fallen da gleich viele gute Beispiele ein: In der Make 5/20 hat Autor Stefan Wollner zum Beispiel eine Hausgeräte-Überwachung mit Fritzbox und Telegram für Waschmaschine und Trockner gebaut. Hans Borngräber hat in der gleichen Ausgabe sogar per DECT-Telefon und Fritzbox seine Haustür geöffnet. Und in Ausgabe 3/20 hat Stefan Henschke mit OpenHAB Funkthermostate über die Fritzbox ausgelesen. Lasst Euch davon inspirieren, zeigt uns Eure bestehenden Smart-Home-Lösungen oder startet ein ganz neues Smart-Home-Projekt!

Die Deadline ist der 10. September 2021 um 23:59 Uhr. Das Projekt wird erst nach Ende der

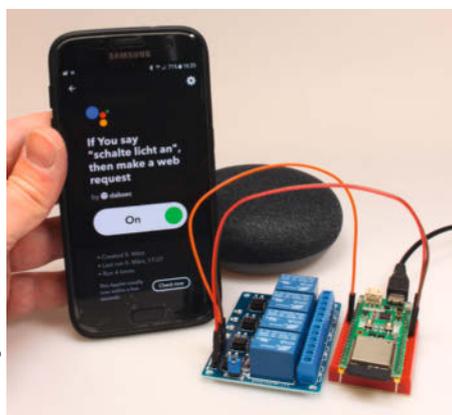
Kurzinfo

- » Smart Home-Projekt auf Make Projects zeigen
- » Sensoren und Aktoren clever einsetzen
- » Gutscheine im Wert von 500 Euro absahnen

Abgabefrist bewertet. Eine Jury, bestehend aus AVM, dem Make-Magazin und Makern aus der Community, **prämiert die zehn coolsten Projekte mit Elektronik-Einkaufsgutscheinen im Wert von 500 Euro**. Und mit *cool* meinen wir zum Beispiel nützlich, nachbaubar oder umwerfend komplex. Los geht's! Schaut auf Make Projects vorbei. —rehu

► makeprojects.com/de/smart-home-challenge

Alles zum Artikel im Web unter make-magazin.de/xdd7



Aus der Make 3/18: Chefredakteur Daniel Bachfeld steuert sein Smart Home mit dem Cloud-Dienst IFTTT.

Make: Magazin



UNSER SORTIMENT VON MAKERN FÜR MAKER

The best part of your project:
www.reichelt.de

Nur das Beste für Sie – von über 900 Markenherstellern.

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

USB-Lötstation mit regelbarer Temperatur

Der USB-Lötkolben ATTEN GT-2010 eignet sich ideal für Präzisionslötarbeiten im Elektronikbereich, bei denen kleine Komponenten flexibel demontiert werden sollen.

- 10 Watt Heizelement, 150 - 450 °C
- 5 V DC Versorgungsspannung (funktioniert auch mit Powerbanks)
- integrierter Kontakttemperatursensor für eine höhere Empfindlichkeit
- drei programmierbare Temperaturspeicher



Bestell-Nr.:
STATION GT-2010

59,95



ATTEN[®]
INSTRUMENTS

ATTEN-Lötstationen: Leistungsstarke Helfer für jeden Zweck

Die Firma Atten ist bekannt für ihre preiswerten, aber dennoch kraftvollen Lötkolben und -stationen.

Mehr Infos ▶



AUF UNSEREM YOUTUBE-KANAL FINDEN SIE SPANNENDE ELEKTRONIK HOW-TO'S & UNBOXINGS



Direkt reinschauen:
youtube.com/reichelt



■ Top Preis-Leistungs-Verhältnis

■ über 120.000 ausgesuchte Produkte

■ zuverlässige Lieferung – aus Deutschland in alle Welt

www.reichelt.de

Bestellhotline: +49 (0)4422 955-333

reichelt
elektronik – The best part of your project

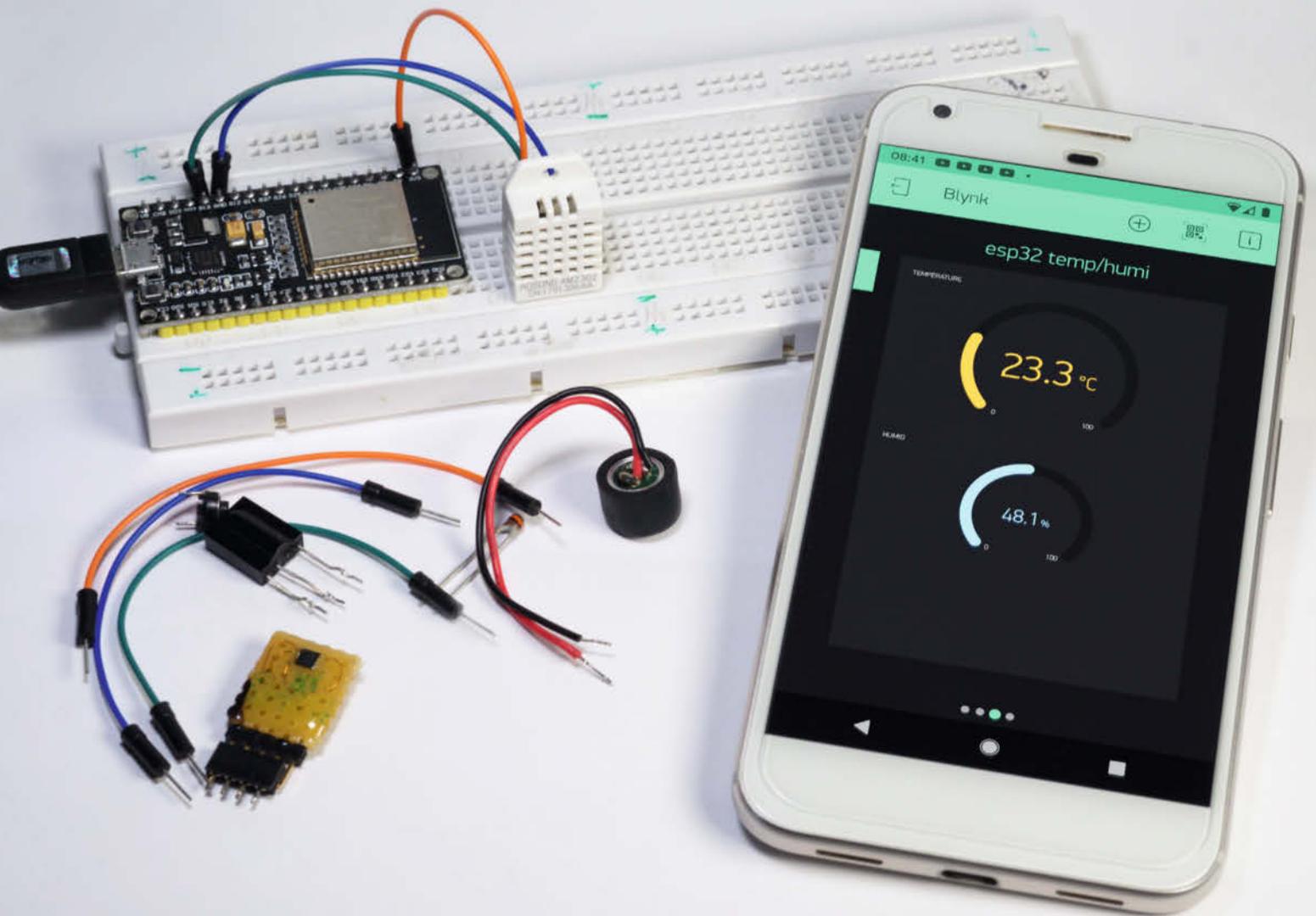
Es gelten die gesetzlichen Widerrufsregelungen. Alle angegebenen Preise in € inklusive der gesetzlichen MwSt., zzgl. Versandkosten für den gesamten Warenkorb. Es gelten ausschließlich unsere AGB (unter www.reichelt.de/agb, im Katalog oder auf Anforderung). Abbildungen ähnlich. Druckfehler, Irrtümer und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

TAGESPREISE! Preisstand: 17. 5. 2021

Internet-of-Things-Dienste für Maker

Wir haben für Sie die gängigsten IoT-Services für Maker kurz ausprobiert und verglichen. Die Anzahl solcher Dienste ist inzwischen sehr unübersichtlich geworden – wir präsentieren eine kleine Auswahl der populärsten und interessantesten Vertreter.

von Carsten Wartmann



Unter dem Schlagwort *Internet of Things*, kurz *IoT*, verbirgt sich ein ganzer Zoo von Technologien und Internetprotokollen, die das gemeinsame Ziel haben, alle möglichen Arten von Geräten an das Internet anzukoppeln. Im Gegensatz zu vernetzten Computern und Servern sind diese Geräte meist mit wenig Rechenleistung und Speicher ausgestattet und für geringe Datenraten ausgelegt. Weiterhin sollen sie wenig Energie verbrauchen (siehe auch Seite 84) und günstig sein. Diese IoT-Geräte stellen dann ihre Dienste und Daten im Netz zur Verfügung, allerdings nicht direkt, sondern über einen Server (*Gateway/Broker*) im Internet, der die Daten sammelt, auswertet und zur Abfrage bereitstellt. Auch der Rückweg ist möglich, sodass IoT-Geräte über das Internet gesteuert werden können.

Wenn hier die Rede vom *Internet* ist, meinen wir natürlich auch ein *Intranet*, das auf Internet-Technologie aufbaut. Viele der hier vorgestellten Dienste bieten ihre Server auch für die Installation im eigenen Netz oder WLAN Zuhause an. Dies ist aber nicht der Fokus dieses Artikels, ebenso wenig wie der Aufbau einer eigenen IoT-Infrastruktur, sei es kommerziell gehostet oder auf einem Raspberry Pi. Der Charme von IoT-Diensten besteht ja gerade darin, dass man sich dies sparen kann.

IoT-Geräte senden ihre Daten, wann immer sie selbst bereit sind, an den IoT-Service bzw. dessen Gateway. Idealerweise ist jedes Gerät nur für eine Aufgabe oder einen Messwert zuständig, daher bleiben die andere Geräte funktional, falls eines zeitweise oder komplett ausfallen sollte. Auch eine Redundanz von Geräten mit gleichen Sensoren ist so möglich und auf dem Server des IoT-Dienstes können bei einem Ausfall Maßnahmen ergriffen werden, von der Abfrage alternativer Quellen bis zu einer Reparaturbenachrichtigung an den Besitzer der IoT-Geräte.

IoT-Dienste, extern bei Anbietern gehostet, sind bequem, hat man sich doch weder mit der Bereitstellung der Dienste, noch Administration, noch Backup zu beschäftigen. Allerdings gibt man damit auch seine Daten aus den Händen. Die Serverstandorte sind oft nicht in der Europäischen Union. Der Gedanke an die Verknüpfung der eigenen Positionsdaten per Google zu einem IoT-Dienst, der über Facebook die Anwesenheit der Familienmitglieder ankündigt, lässt jeden Datenschützer grün anlaufen. Was wir im privaten Bereich dulden können, muss natürlich jeder für sich selbst entscheiden.

Viele Dienste kosten etwas, wenn man sie nicht nur ausprobieren, sondern produktiv einsetzen will. Hier werden schnell zweistellige Beträge pro Monat fällig, dafür bekommt man aber auch einiges geboten. Nicht immer ist garantiert, dass man in Falle einer Pleite des Anbieters noch an seine Daten kommt, ganz davon abgesehen, dass man üblicherweise

Kurzinfo

- » Übersicht IoT-Webdienste
- » Anbindung von Maker-Boards wie Raspi, Arduino, ESP
- » Vernetzung von IoT-Geräten über Webdienste

Checkliste

- Zeitaufwand:** ab zwei Stunden
- Kosten:** kostenlos bis über 400 Euro pro Monat für Profi-Service

Mehr zum Thema

- » Daniel Bachfeld, Projekte steuern mit IFTTT, Make 3/18, S. 100
- » Daniel Bachfeld, Funkfernsteuerung, Make 2/18, S. 56
- » Dr. Veikko Krypacz, Schalter zur Welt, Make 4/20, S. 50
- » Daniel Bachfeld, Docker für Raspberry Pi, Make 1/21, S. 92

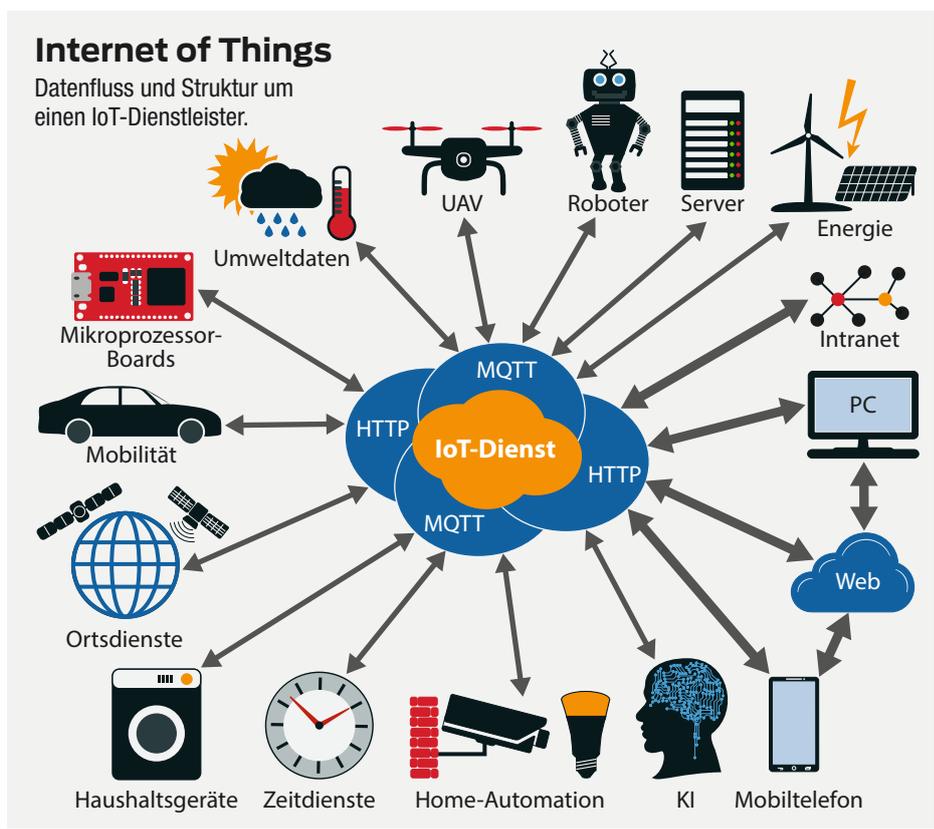
Material

- » IoT-fähiges Maker-Board (optional), ESP32 oder ähnliches
- » Breadboard (optional)
- » Sensoren für Experimente (optional), Licht, Temperatur, Feuchte, ...
- » IoT-Gerät (optional), Smarte Steckdose, Wetterstation, ...

Alles zum Artikel im Web unter make-magazin.de/x4s2

auch eine Menge Arbeit in die Konfiguration der Dienste bei diesem Anbieter gesteckt hat. Kostenlose Angebote, von eher kommerziell orientierten Firmen, werden auch gerne einmal abgeschaltet und dann hat man nur

die Wahl, zu einem Bezahlmodell beim gleichen Anbieter oder zu einem anderen Anbieter zu wechseln. Wir haben daher auch die IoT-Dienste zweier sehr in der Maker-Szene verankerten Anbieter aufgenommen, hier



IoT-Datenquellen und Geräte (Auswahl)

Geräte

- Lampen (Helligkeit, Farbe)
- Haushaltsgeräte
- IT-Technik (Server, Drucker, ...)
- Staubsaugerroboter (Ladestände, Steuerung)
- Heimautomation (Heizung, Sicherheit, Mediacenter, Lüftung)
- Autos/Logistikflotten

Datenquellen

- Umgebungsbedingungen und Wetter
- Temperatur
- Luftfeuchte
- Wind
- Regen

- Helligkeit
- UV-Licht
- Gase/Stäube (CO₂, Feinstaub)
- Verbrauchsmessungen
- Spannungen und Ströme an Geräten
- Gas- und Wasserzähler
- Betriebszeiten
- Füllstände
- Zustandsmeldungen
- Störungsmeldungen
- Türklingel
- Alarmer (Feuer, Einbruch, Wasser)
- WLAN-Stärke-Messungen
- Server-Auslastungen, Temperaturen
- Geofencing, Geolocating, Beacons

oft sehr steil. Dienste, die eine kommerzielle Benutzung nicht explizit in den kostenlosen Plänen ausschließen, sind oft so weit eingeschränkt, dass man nicht weit kommt und doch über ein bezahltes Abo nachdenken muss. Auch diese Überlegungen flossen in unsere Auswahl ein.

Auswahlkriterien für IoT-Dienste

IoT-Dienste sind in ihrer Funktionalität divers aufgestellt und jeder Service hat seine eigenen Stärken und Schwerpunkte. Natürlich müssen alle Dienste irgendwie mit den Geräten in Kontakt treten und die Daten von diesen empfangen. Aber hier fangen dann die Unterschiede an: Manche Dienste speichern oder aggregieren keine Daten, sondern senden sie sofort an andere Dienste weiter. Dienste, die Daten grafisch aufbereitet im Web darstellen, müssen diese Daten auch kurzfristig speichern. Manche leisten aber nur genau dies und bieten die Daten dann nicht zum Zugriff für andere Dienste und Benutzer an. Andere Dienste bieten ausgefeilte Datenbearbeitung, Überwachung und Statistik. Damit können sie

kann man sich einer fairen Behandlung sicher sein.

Um in unsere Auswahl zu kommen, musste mindestens ein *Free Plan* vorhanden sein, also eine kostenlose Möglichkeit, den

Dienst auf Herz und Nieren zu prüfen. Möchte man kommerziell mit diesen Diensten arbeiten, so ist dies nicht selten mit hohen Gebühren verbunden. Der Übergang im Preis vom Maker zum kommerziellen Anwender ist

IoT-Dienste im Vergleich

Dienst	IFTTT	Zapier	Blynk	IO Adafruit	Arduino IoT Cloud	ThingSpeak	Cayenne
URL	ifttt.com	zapier.com	blynk.io	io.adafruit.com	create.arduino.cc/iot/things	thingspeak.com	cayenne.mydevices.com
Einschränkungen bei kostenloser Nutzung	3 Applets, ohne Verzweigung	5 Zaps ohne Verzweigungen, 100 Tasks/m, 15min Update-time	5 Devices / 2000 Energy (siehe Text)	30 Daten/min, 30d Datenspeicher, 10 Feeds, 5 Dashboards.	2 Things/ 100MB Sketches/1d Datenspeicher	8200msg/d, alle 15s max., 20s Matlab, 3 MQTT subscriptions, 4 Kanäle	10 Geräte
Kosten	4\$/Monat, 40\$/Jahr	Starter 30\$/M, 240\$/Jahr	Startup ab 415\$/Monat	IO+ 10\$/Monat oder 99\$/Jahr	Entry 3\$/Monat, Maker 7\$/Monat	Student ab 55€/J, Home ab 75€/J	Auf Anfrage
Diensttypen	Webhooks (Maker Channel), Dienstverbinder (Trigger, QueriesP, FilterP, Actions)	Dienstverbinder (Trigger, LogicP, FilterP, WebhooksP	Webhooks, Library	Webhooks, Library, Triggers, Services	Webhooks, Library, IoT-Applications	Webhook, Library, Apps, Matlab	LoRa, Library, Extensions, Webhooks, Actions, Alerts
Mobile App	ja	nein	ja	nein	ja	nein	ja
Datenspeicher	nein	nein	ja (max. drei Monate)	ja (30/60 Tage)	ja (1/15/90 Tage)	ja (unbegrenzt?)	ja (ein Monat)
HTTP/MQTT/Import/Export	✓/✓/✓/✓	✓/✓/✓/✓	✓/✓/✓/✓	✓/✓/✓/✓	✓/✓/✓/✓	✓/✓/✓/✓	✓/✓/✓/✓
IoT-Geräte und Anbindung	HTTP-fähige oder Trigger/Action vom Hersteller des Geräts	HTTP-fähige oder Trigger/Action vom Hersteller des Geräts	HTTP, Blynklib	HTTP/MQTT/Library	Arduino Library/HTTP ¹ /MQTT ¹	HTTP/MQTT/Library	HTTP/MQTT/Library
Endgerät	Browser/iOS/Android	Browser	iOS/Android	Browser	Browser, Mobile App	Browser	Browser, Mobilgeräte „bald“
Besonderheiten	große Unterstützung der Hersteller von IoT-Geräten	gute Debugging Möglichkeiten	komplette Entwicklung der Apps auf Mobilgerät	Enge Integration in Adafruit Ökosystem	Entwicklung komplett online möglich	Matlab Integration	Lora-Network

¹Nicht in kostenloser Version

Angaben ohne Gewähr und vorbehaltlich Änderungen durch die Betreiber

IoT-Definition? Internet mit Dingsbums!

Internet der Dinge, also *Internet of Things (IoT)* ist keine feste Definition, sondern noch etwas schwammig und im Fluss. Ziel ist es, Geräte und Messwerte aus der physischen Welt miteinander zu verknüpfen und sie Daten austauschen zu lassen. Dabei stand die Idee im Vordergrund, den Erfolg des Internets zu wiederholen, aber diesmal nicht für Menschen, sondern für alle Geräte, seien es Wetterstationen oder Autos. Das Internet wurde zum Erfolg, weil Hardware und Software austauschbar und die Protokolle, mit denen die Server und Clients miteinander reden, standardisiert

und offen sind. Sicher gab es auch schon vor der Entwicklung des IoT-Geräte, die heute als IoT-Devices gelten würden. Allerdings konnte man die Wetterstation von Hersteller A nicht gegen eine von Hersteller B austauschen und damit rechnen, dass diese auch ihre Daten an die Server von Hersteller A übertragen würde. Dazu kam, dass die Geräte immer kleiner, batteriebetrieben und energiesparender wurden. Damit mussten einfache, redundante und sichere Möglichkeiten zum Datenaustausch gefunden werden. Heute übertragen die Geräte Daten primär per

HTTP oder MQTT. Auf welchem Wege diese Daten übertragen werden, sei es per Kabel, Klopfezeichen oder einen beliebigen Funkstandard, ist dabei nicht wichtig, die Geräte haben eine eigene Adresse und sind so eindeutig zu identifizieren. Gegenüber der Fernüberwachung und Fernsteuerung von Geräten, was am Ende immer einen Menschen erfordert, sollen Datensammlung und maschinelle Auswertung im IoT ermöglichen, dass Maschinen schneller auf Ereignisse in den Daten reagieren können, sei es bei Umweltkatastrophen oder an der Börse.

auf Veränderungen in den Daten mit Nachrichten aller Art reagieren. Die meisten Dienste bieten den Zugriff per Webbrowser an, mache auch per spezieller App für Mobilgeräte. Einige Dienste sind sogar ausschließlich per App bedien- und abfragbar. Findet man keinen Dienst, der allen eigenen Anforderungen entspricht, so ist es dank der Standardisierung der Übertragungswege auch möglich, IoT-Dienste zu kombinieren.

Die Kommunikation der Dienste funktioniert im Prinzip immer gleich: Es werden kleine Nachrichten per HTTP oder MQTT verschickt oder empfangen. Das ist so simpel, das selbst sehr langsame Prozessoren dies heutzutage erledigen können. Natürlich muss der Mikrocontroller irgendwie an das Internet angebunden sein. Da die Mikrocontroller in den letzten Jahren immer schneller wurden, ist heute die Übertragung per HTTP sehr beliebt – im Gegensatz zu MQTT ist es einfacher zu verstehen und zu benutzen, kommt aber bei schlechten Verbindungen und komplizierteren IoT-Device-Strukturen auch an seine Grenzen.

Mithilfe der Dokumentation, die der Dienstbetreiber zur Verfügung stellt, baut man sich schnell eine Abfrage oder Sendung zusammen. Komfortabler ist es aber, die Software-Bibliotheken (*Libraries*) der Dienstanbieter zu nutzen. Diese *Libraries* können aber auf Boards und Geräten mit wenig Speicher zu einer Speicherknappheit führen. Bei Verwendung der *Libraries* wird dann für die Übertragung HTTP oder MQTT benutzt, je nachdem was der Dienstleister bevorzugt oder anbietet, ohne das man es als Anwender merkt oder wissen muss.

In *Listing 1* ist beispielhaft die Anbindung eines ESP32 mit Temperatursensor an den Dienst *Blynk* per MicroPython gezeigt. An sich sind nur drei typische Python-Skripte kombiniert, WLAN-Anbindung, Abfrage des Sensors

und die Benutzung der Blynk-Library. In *Listing 3* werden von einem ESP32 Temperaturdaten direkt per MQTT an Cayenne-IoT-Dienst (siehe entsprechender Abschnitt) gesendet. Alle der getesteten Dienste bieten einen großen Vorrat an Beispielen für die typischen Maker-Boards.

IFTTT

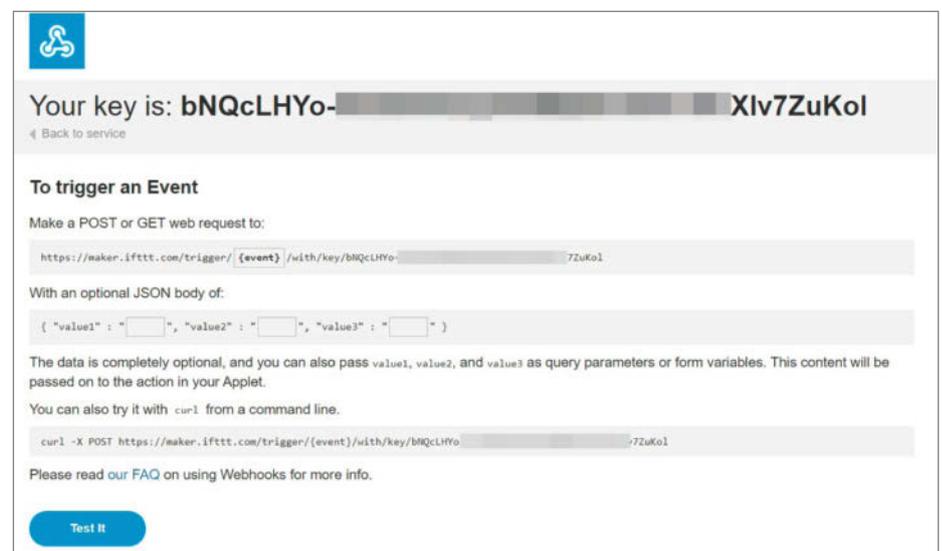
IFTTT (If-This-Then-That) ist ein Internetdienst, der es ermöglicht, verschiedenste Dienste im Sinne von „Wenn dies ist, dann mache das“ miteinander zu verknüpfen.

Das Prinzip ist einfach und sehr grafisch in der Weboberfläche (auch in der Android- bzw. iOS-App) gelöst: Man definiert den Auslöser (*Trigger*) eines Dienstes und bestimmt dann, welche Aktion (*Action*) auf einem anderen Dienst ausgelöst wird. *Trigger* und *Action* bil-

den zusammen das sogenannte *Applet*. Es gibt eine schier unüberschaubare Menge von Diensten, die als Trigger oder für eine Action genutzt werden können. Hier können nicht nur IoT-Devices, sondern auch andere Inter-



Die Oberfläche von IFTTT im Browser



Informationen zu IFTTT-Webhooks

MQTT

Message Queuing Telemetry Transport (MQTT) ist ein offenes Netzwerkprotokoll für Kommunikation zwischen Maschinen (*M2M*). Die Übertragung geht über einen Server (*Broker*), der die Kommunikation sozusagen puffert, sodass die Daten trotz möglicher Verzögerungen und Verbindungsabbrüche stets in einem konsistenten Zustand bleiben und von leistungsfähigen Rechnern

in einem Rutsch abgefragt werden können, ohne auf die IoT-Devices warten zu müssen. Dabei ist das Kommunikationsprotokoll einfach aufgebaut, aber hierarchisch strukturiert, sodass auch einfache Mikrocontroller es verwenden können. Dabei bleibt es erweiterbar, damit kompliziertere Daten in aufwändigen IoT-Netzen sicher kommuniziert werden können.

net-Dienste untereinander verbunden werden. Zum Beispiel, wenn Sie eine Nachricht per Telegram bekommen möchten, wenn es einen neuen Tweet mit einer Erwähnung Ihres Twitter-Namens gibt.

Als Trigger gibt es aber auch einige andere IoT-Dienste, wie *Adafruit IO*. Hier kann dann etwa eine Messwertüberschreitung einen Trigger auslösen. Als Ziel der Action kann aber auch ein Google Spreadsheet dienen, um Daten oder Ereignisse aufzuzeichnen.

Für die Verbindung zu den typischen Maker-Platinen wie den ESP-Boards oder den Raspberry Pi, gibt es den *Webhook*: Hier muss das zu verbindende Gerät nur einen Web-Request an IFTTT senden, der ein gewähltes Schlüsselwort, den Authentifizierungscode und bis zu drei Datenwerte enthält. Die URLs und die Daten für die Webhooks erhält man auf einer nicht so gut zu findenden Seite (Link in Kurzinfo). Hier erfährt man auch seinen *Auth-Key* und den Aufbau der Abfrage unter dem Knopf *Docu-*

mentation. Als Action gibt es auch einen *Make Web Request*, der an eine URL im Netz einen Request sendet. Damit hat man einen Rückkanal zum IoT-Device, bei Geräten hinter einem Router muss das allerdings die Firewall zulassen.

Die Benutzung ohne monatliche Zahlung erlaubt nur drei Applets. Möchte man mehr, so kostet das 4 US-Dollar im Monat oder 40 Dollar im Jahr. Dafür sind dann unbegrenzt Applets erlaubt und es gibt noch *Queries* (Suchabfragen) und *Filter* zwischen *Trigger* und *Action*. Diese können Ergebnisse verfeinern, bearbeiten und filtern. Auch ist man nicht mehr auf einen Strang *If-this-then* begrenzt, sondern kann beliebig viele *Actions* pro Applet benutzen. Weiterhin werden Applets schneller und öfters ausgeführt und die Datenrate ist für populäre *Actions* nicht begrenzt.

Alles in allem ist IFTTT ein sehr interessanter Dienst, der eine Menge Services im Internet koppeln kann. Hier kann man sich

austoben. Für die klassische Datensammelei, bei der man Sensordaten in schönen Grafiken darstellt, ist es eher nur am Rande geeignet und stark von den verfügbaren Diensten abhängig. IFTTT speichert nichts und produziert selbst auch keine Grafiken. Da aber fast jedes IoT- oder Maker-Board eine URL per Webhook ansprechen kann, ist es kein Problem, IFTTT so zu benutzen. Dies ist aber eben nicht die Hauptkompetenz des Dienstes. Das merkt man auch an der Entwicklerdokumentation, die weniger für die „Things“ im Maker-Sinne gedacht ist, sondern sich viel mehr an Web- und App-Entwickler richtet.

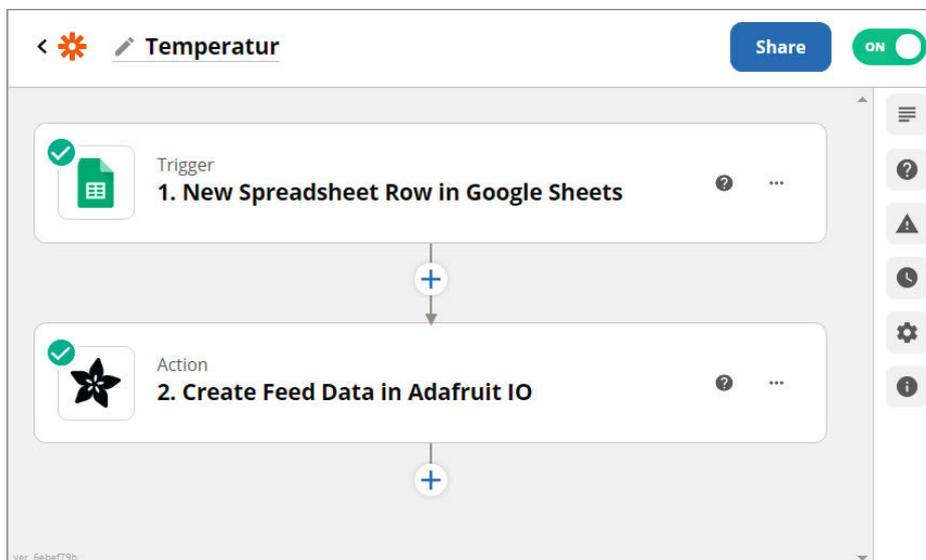
Wenn Sie sich für IFTTT interessieren, sollten Sie prüfen, ob die Dienste, die Sie gerne benutzen, auch im Angebot sind und welche ihrer Funktionen unterstützt werden. Kostenlos ist nicht allzu viel möglich, aber man wird auch nicht arm, wenn man sich mal einen bezahlten Monat testweise austobt.

Zaphier

Zaphier ist ähnlich konzipiert wie IFTTT, auch die Einschränkungen des *Free Plans* sind ähnlich: Nur zwei Stufen in den *Zaps* (entspricht den *Applets* in IFTTT), keine *Premium Apps* (entspricht *Services* in IFTTT, z.B. Paypal oder Facebook *Lead Ads*), keine Verzweigungen. Es gibt, wenn man sich für den *Free Plan* registriert, 14 Tage Zugriff auf die *Professional Features* und das macht durchaus Lust auf den Dienst.

Die Oberfläche ist nicht so schreiend bunt und groß wie bei IFTTT. Man kann auch komplexere Sachen in den Actions machen, etwa Spalten aus Google Tabellen lesen, die gerade aktualisiert wurden. Die Oberfläche zum Erstellen und Verwalten hat uns besser als die bei IFTTT gefallen, wir fühlten uns nicht so eingezwängt. Das Testen und Debuggen von Zaps ist besser realisiert, man sieht sofort, wann und mit welchem Ergebnis das Zap lief. Auch bei der Erstellung von Zaps leitet der Dienst den Anwender klar durch die nötigen Schritte, die man später einzeln ändern und anpassen kann, ohne den Assistenten erneut abarbeiten zu müssen.

Insgesamt hat man hier das Gefühl, dass man auch mit dem *Free Plan* weiter kommt wenn, ja wenn denn der gewünschte Dienst verfügbar ist. Hier klaffen aber Lücken, so gibt es beispielsweise keine Anbindung an Telegram. Dies wird ja dank der Möglichkeit, Telegram-Bots zu benutzen, schon immer gerne als Messenger-Dienst für IoT benutzt, z. B. für Nachrichten vom 3D-Drucker. So bleibt nur ausprobieren, die Liste der Services zu durchsuchen und eventuell Alternativen zum eigentlich gewünschten Dienst zu finden, die unterstützt werden.



Einen Zap in Zaphier erstellen

Blynk

Blynk ist anders. So unser erster Eindruck. Blynk ist ein IoT-Dienst, der sich als Entwicklungsplattform sehr stark an kommerzielle Hersteller wendet, die Applikationen für ihre eigenen IoT-Geräte auf Mobiltelefonen entwickeln wollen. Der *Free Plan* bietet fünf Geräte, ist aber durch die *Energy Balance* (s.u.) limitiert. Der *Free Plan* darf keinesfalls für kommerzielle Belange eingesetzt werden. Blynk arbeitet mit einer *Energy Balance* (Energiebilanz, sozusagen eine interne Währung) und schränkt so die Anzahl der GUI-Elemente im *Free Plan* ein. Möchte man mehr, muss man *Energy* kaufen (In-App-Kauf, ab 3 Euro pro 1000 *Energy*) oder alte Projekte/Elemente „recyclen“.

Der offensichtlichste Unterschied zu anderen Diensten ist, dass man ausschließlich in der App (iOS und Android) am Mobiltelefon entwickelt. Wir konnten nicht herausbekommen, ob dies auch in den bezahlten Tarifen so bleibt oder ob man dort noch andere Entwicklungssysteme bekommt. Diese Entwicklung auf dem Telefon funktioniert erstaunlich gut und der Wechsel von Editor zum Testen der App ist praktisch nahtlos. So klickt, zieht und tippt man sich schnell attraktive Apps zusammen. Damit lassen sich die IoT-Geräte steuern, Daten anzeigen und auswerten. Für fast alle Maker-IoT-Boards und Entwicklungsumgebungen gibt es eine Version der *Blynk-Library*. Die Dokumentation ist ausführlich, gut strukturiert und geht in die Tiefe, wo dies nötig ist.

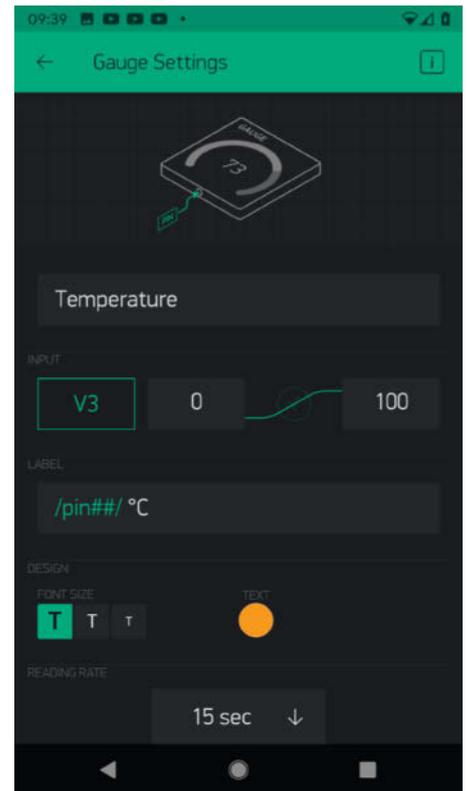
Weiterhin gibt es einen Assistenten, der Beispiel-Code für verschiedenste Boards generiert (siehe *Listing 1*). Die Blynk-App kommuniziert mit dem Server und dieser dann mit dem Board per *Blynk-Library*. Das funktioniert in beiden Richtungen. So lassen sich GPIOs steuern,



Blynk führt unsere Test-App aus.

Messwerte abfragen, sogar ein Video-Gadget gibt es, mit dem man dann von einer Kamera Videostreams in der Blynk-App sehen kann.

All dies sollte aber nicht über die kommerzielle Ausrichtung hinwegtäuschen: der billigste Tarif nach dem *Free Plan* kostet mindestens 415 US-Dollar pro Monat, was eine private Nutzung schon sehr erschwert. Tatsächlich ist der Server Open Source (GPL) und basiert auf



Blynk: Ändern eines Elements im Editor

Java. Durch die App hat Blynk hier natürlich trotzdem den Fuß in der Tür und auch ein eigener Server kann und darf die Einschränkungen des *Free Plans* nicht umschiffen.

IO Adafruit

Adafruit sind ja als Elektronik-Shop, Hardware- und Circuit-Python-Entwickler einer der Vor-

Studiengang

Elektrotechnik und Informationstechnik

Fakultät für Elektrotechnik und Informationstechnik

TECHNISCHE UNIVERSITÄT
ILMENAU

Bachelor

Master

Diplom

Elektrotechnik und Informationstechnik setzt Innovationen in den unterschiedlichsten Bereichen unserer stark digitalisierten Technologiegesellschaft. Erneuerbare Energien, Mobilität oder auch elektronische Kommunikation sind unter anderen Themenschwerpunkte des Ingenieurstudiengangs. - Ingenieure*innen der Elektrotechnik und Informationstechnik tragen somit nicht nur aktiv zu einer Verbesserung der Lebensqualität bei, sondern auch zur Schonung natürlicher Ressourcen.

Studienfachberatung: Univ.-Prof. Dr. rer. nat. Jochen Seitz
jochen.seitz@tu-ilmenau.de
www.tu-ilmenau.de

Listing 1: ESP32 zu Blynk, blynklib_mp

```
# Blynk Library Demo
import blynklib_mp as blynklib
import network
import utime as time
from machine import Pin

# DHT22/AM2302
import dht

WIFI_SSID = 'SSID'      # WLAN SSID
WIFI_PASS = '1234'     # Passwort
BLYNK_AUTH = 'CWHcUKXsrhesIdHvMMQxB7qB1JJPELLE'
GPIO_DHT22_PIN = 13 # Pin der Datenleitung DHT-Sensor

print("Verbinde mit WLAN '{}'.format(WIFI_SSID))
wifi = network.WLAN(network.STA_IF)
wifi.active(True)
wifi.connect(WIFI_SSID, WIFI_PASS)
while not wifi.isconnected():
    time.sleep(1)
    print('WLAN...')
print('WLAN IP:', wifi.ifconfig()[0])

print("Verbinde mit Blynk Server...")
blynk = blynklib.Blynk(BLYNK_AUTH)

T_COLOR = '#f5b041'
H_COLOR = '#85c1e9'
ERR_COLOR = '#444444'

T_VPIN = 3 # virtueller Pin Temp.
H_VPIN = 4 # virtueller Pin Humi.

dht22 = dht.DHT22(Pin(GPIO_DHT22_PIN, Pin.IN, Pin.PULL_UP))

@blynk.handle_event('read V{}'.format(T_VPIN))
def read_handler(vpin):
    temperature = 0.0
    humidity = 0.0

    # Sensor Daten
    try:
        dht22.measure()
        temperature = dht22.temperature()
        humidity = dht22.humidity()
        print("%d°C  %d%%rel. Feuchte"%(temperature, humidity))
    except OSError as o_err:
        print("Keine DHT22-Sensor Daten: '{}'.format(o_err)")

    # Farbe in App je nach Werten ändern
    if temperature != 0.0 and humidity != 0.0:
        blynk.set_property(T_VPIN, 'color', T_COLOR)
        blynk.set_property(H_VPIN, 'color', H_COLOR)
        blynk.virtual_write(T_VPIN, temperature)
        blynk.virtual_write(H_VPIN, humidity)
    else:
        # Rot = fehlerhafte Daten
        blynk.set_property(T_VPIN, 'color', ERR_COLOR)
        blynk.set_property(H_VPIN, 'color', ERR_COLOR)

while True:
    blynk.run()
```

reiter und eine Vorzeigefirma der Maker-Bewegung. Da lag es wohl nahe, einen IoT-Service mit in das Programm aufzunehmen. Erklärtes Ziel ist, ein *Internet of Things* für jeden zu gestalten. Professionelle Anwendungen sind möglich, waren aber nicht Entwicklungsziel.

Im Gegensatz zu IFTTT oder Zapier ist *IO Adafruit* eher ein klassischer Datensammler, der auch schöne Grafiken erstellt. Den Ansatz,

eher für das Lernen und für Maker entwickelt zu sein, sieht man auch daran, dass die eingehenden Daten nur 30 Tage (60 Tage im bezahlten Pro-Account) vorgehalten werden.

Mit *Triggern* können Aktionen ausgelöst werden wenn definierte Bedingungen zutreffen, etwa ein Alarm bei Leckagen oder Überschreitung von Temperaturen. Dies kann dann per Webhook oder Message in den

Stream kommuniziert werden. E-Mails sind nur im *IO+ Abo-Plan* möglich. Mit den *Services* können Sie dann *IO Adafruit* auch mit IFTTT oder Zapier verbinden und so einiges erledigen, was allein mit *IO Adafruit* nicht möglich ist. Weitere *Services* sind *Weather* (Wetter), *Random* (zufällige Daten) und auch ein Zeit-Service per HTTP oder MQTT.

Ansonsten ist alles schick und einfach designed, wie von Adafruit gewohnt, sehr gut dokumentiert, sodass wir als Maker schnell zum Ziel kommen. Natürlich werden die eigenen Boards und Circuit Python besonders hervorgehoben, aber die Umsetzung auf nicht-Adafruit-Hardware ist einfach und Circuit Python ist auch für so gut wie alle moderneren Maker-Boards verfügbar. Wenn nicht, ist für Arduino-kompatible Boards eine IO-Library verfügbar und dokumentiert.

Arduino IoT Cloud

Wie bei Adafruit liegt es auch für Arduino nahe, einen IoT-Dienst für die Anwender des Arduino-Ökosystems bereitzustellen. Hier wird tatsächlich ein Rundumschlag aus Cloud-Speicher, IoT-Dienst, Web- und Mobilanwendungen geführt.

In der Weboberfläche von *Arduino IoT Cloud* erstellt man Geräte (*Things*), vergibt Variablen und erhält einen Sketch, der dann auf das Gerät gespielt wird – wenn man ein Arduino-Plug-in für den Browser installiert, auch direkt aus dem Browser heraus. Man kann auch zu einem kompletten Online-Editor auf *Arduino Create* umschalten, der sehr an die Arduino IDE erinnert. Lädt man den erzeugten Sketch herunter, so muss man dort die WLAN-Zugangsdaten und den geheimen Schlüssel für die Arduino Cloud eintragen.

Nach erfolgreicher Übersetzung des Quellcodes und Upload auf das Board kontaktiert der Sketch dann den Arduino-Dienst per MQTT. Im Sketch sind Variablen angelegt, die man dann mit den gewünschten Werten füttert und periodisch `ArduinoCloud.update()` aufruft (*Listing 2*). So macht man problemlos aus einem Sketch, der seine Ausgaben per serieller Schnittstelle macht, einen, der die Daten zur Arduino-Cloud sendet.

Arduino IoT Cloud trennt zwischen *Devices* und *Things*. *Devices* sind die Boards, *Things* können Daten von verschiedenen *Devices* kombinieren. Diese Trennung ist praktisch, man kann hier gut kombinieren und testen. Als nächstes erstellt man ein Dashboard, das die Werte anzeigt und auch Steuerungsmöglichkeiten für die *Things* bietet, sozusagen einen Rückkanal. Dies geschieht alles per Drag & Drop und es gibt einen ausreichenden Satz von Widgets zur Anzeige von Daten und Steuerung der Geräte (Schalter, Farbwähler, Slider, ...). In einem Dashboard können auch die Daten mehrerer Geräte (*Devices*) kombiniert werden.

Möglichkeiten, die Daten zu überwachen oder auf Ereignisse zu reagieren, gibt es bei *Arduino IoT Cloud* nicht. Aber man kann Webhooks erstellen, die dann den Weg ebnen, um mit anderen IoT-Diensten in beiden Richtungen zu kommunizieren und so zum Beispiel Warnmeldungen per Messenger senden.

Arduino IoT Cloud ist recht flexibel und wer sowieso die Arduino IDE und Arduino-kompatible Boards benutzt, findet hier einen prima Einstieg in die Welt des IoT. Insgesamt erscheint die Anwendung allerdings etwas unstrukturierter als z. B. IO Adafruit, auch die Dokumentation geht an manchen Stellen nicht weit genug. So findet man denn Stellen, an denen es heißt „Your code here!“, statt den Einsteiger abzuholen und an einem konkreten Beispiel zu zeigen, wie man etwa einen Temperaturwert überträgt.

ThingSpeak

ThingSpeak ist ein Teil von *MathWorks*, dem Hersteller der Mathematik-, Statistik- und Simulationssoftware *Matlab* und *Simulink*, die in der Industrie und Wissenschaft oft eingesetzt werden. Im *Free Plan* kann man maximal alle 15 Sekunden neue Daten senden (das entspricht etwa 8200 Nachrichten/Tag) und vier Kanäle benutzen. Berechnungen mit *Matlab* zur Analyse und Vorausschau dürfen maximal zwanzig Sekunden dauern.

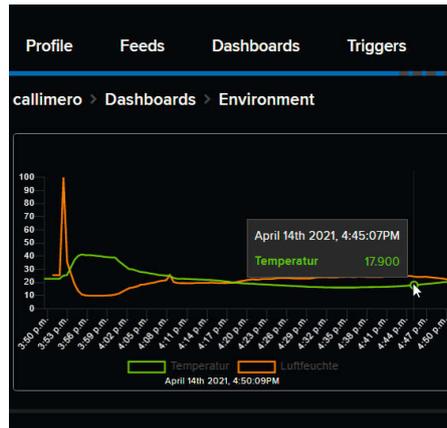
Wie bei Blynk ist der Kosten-Sprung von der freien zur kommerziellen Nutzung steil und startet bei 600 Euro pro Jahr. Dieser Preis lässt sich im Preis-Rechner fein nach oben justieren und ist stark von der Anzahl der Kanäle (üblicherweise ein Kanal pro Gerät) und wie oft man seine Daten sendet abhängig. Für Studenten beginnen die Preise bei 55 Euro pro Jahr, nicht kommerzielle Heimnutzung im Home-Tarif liegt bei ab 75 Euro pro Jahr. Auch bei diesen Tarifen lässt sich per Preis-Rechner der Bedarf optimieren.

ThingSpeak sammelt Daten von Geräten und anderen IoT-Diensten (per HTTP) in den *Channels*. Diese Daten kann man visualisieren und auch wiederum per HTTP-Request abfragen. Darüber hinaus stellt *ThingSpeak* sogenannte *Apps* zur Verfügung, die weitere Funktionalitäten bieten:

React kann einfache Auswertungen von Werten und Daten nach dem Motto „größer, kleiner, gleich als?“ erstellen und dann darauf reagieren, unter anderem mit einer Twitter-App oder per HTTP (s. u.).

ThingHTTP bietet den Zugriff auf andere Dienste, die eine HTTP Schnittstelle anbieten. Dabei wird auch Authentifizierung und Verschlüsselung unterstützt. Das System ist so offen gehalten, das man vom Maker-Board bis zu großen Internet-Plattformen alles erreichen kann.

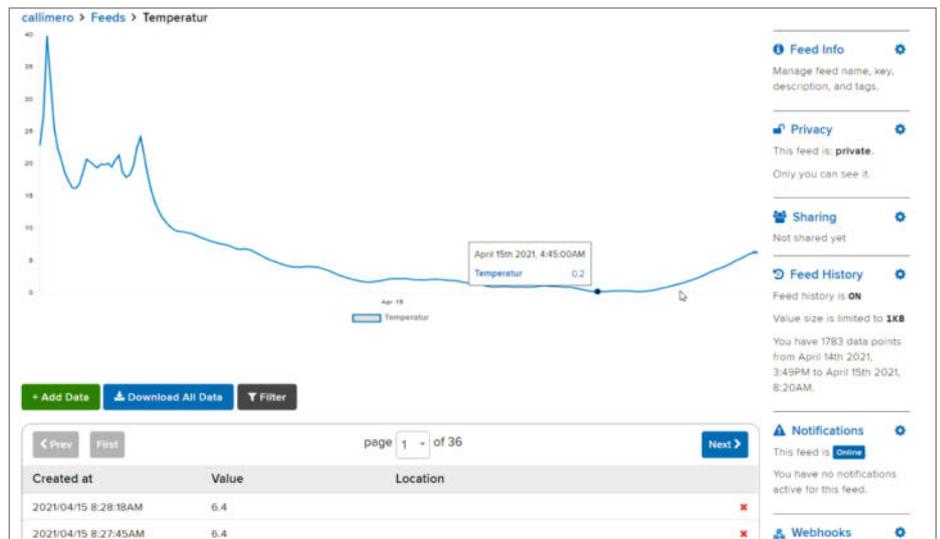
ThingTweet kann Statusupdates auf Twitter posten. So kann die ThingTweet-Instanz



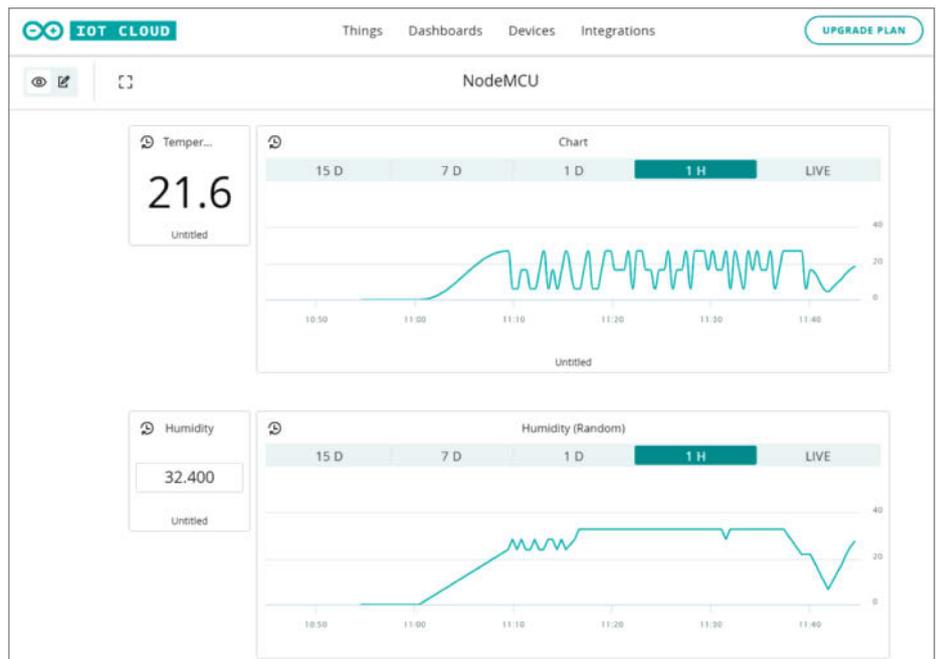
Das Dashboard von IO Adafruit

The screenshot shows the 'Create a new Trigger' form. The 'Reactive Trigger' option is selected. The configuration includes: 'If' set to 'Temperatur', 'Is' set to 'less than', and a 'Comparison Value or F...' field with the value '1'. The time zone is set to 'Europe/Berlin'.

IO Adafruit: Erstellen eines Triggers



Die Feed-Ansicht von IO Adafruit



Dashboard von Arduino IoT

Listing 2: Code von create.arduino.cc

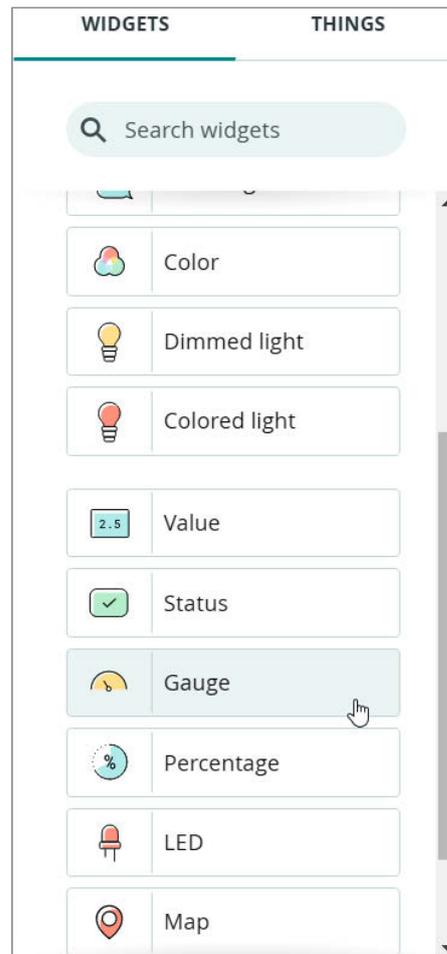
```
//NodeMCU1.0 und DHT22 auf Arduino Cloud
#include "arduino_secrets.h" // von create.arduino.cc laden/anpassen
#include "thingProperties.h" // von create.arduino.cc laden
#include "DHT.h"

#define DHTPIN 12 // D6
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE);

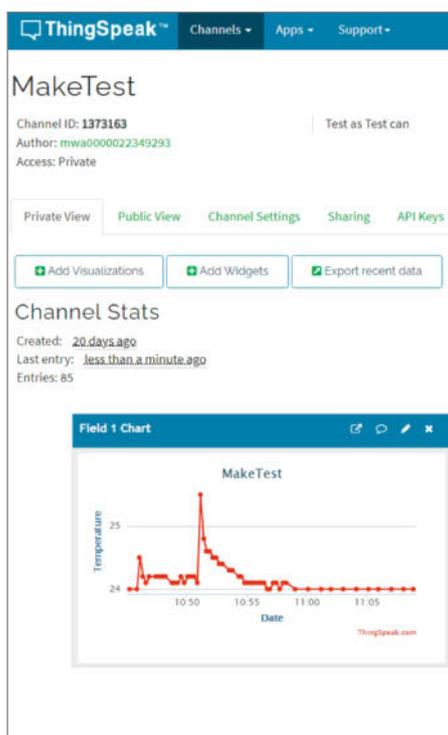
void setup() {
  // Initialize serial and wait for port to open:
  Serial.begin(9600);
  // This delay gives the chance to wait for a Serial Monitor without
  // blocking if none is found
  delay(1500);
  dht.begin();
  // Defined in thingProperties.h
  initProperties();

  // Connect to Arduino IoT Cloud
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
  delay(1500);
}

void loop() {
  ArduinoCloud.update();
  temp = dht.readTemperature();
  humi = dht.readHumidity();
  Serial.print(temp);
  Serial.print(" ");
  Serial.println(humi);
  delay(15000);
}
```



Widgets erstellen mit der Arduino-IoT-Cloud



ThingSpeak: privates Dashboard

als Aktion in einem *React* (s. o.) verwendet werden.

TalkBack kann dazu verwendet werden, IoT-Geräten Anweisungen zu geben, indem Kommandos an eine sogenannte TalkBack-Queue in ThingSpeak geschickt werden.

TweetControl reagiert auf Twitter-Updates. Dazu lauscht die Anwendung unter Verwendung der *Twitter Streaming API* auf neue Statusupdates eines Benutzers oder die Verwendung von definierten Hashtags.

Channels kann man öffentlich verbreiten oder nur bestimmte Benutzer per E-Mail dazu einladen. Damit kann dann jeder Berechtigte die Daten benutzen und per API abrufen. Von Wetterstationen über Radioaktivitätsmessungen bis hin zu kompletten Datensätzen aus Gewächshäusern findet man einiges. Es ist allerdings immer nur der Ausschnitt der Daten verfügbar, den der Datenersteller auch herausgibt, daher auch die Unterscheidung von privaten und öffentlichen Ansichten in der Oberfläche von ThingSpeak. Die Oberfläche ist klar in Tabs strukturiert und man findet schnell die wichtigen Informationen und Funktionen.

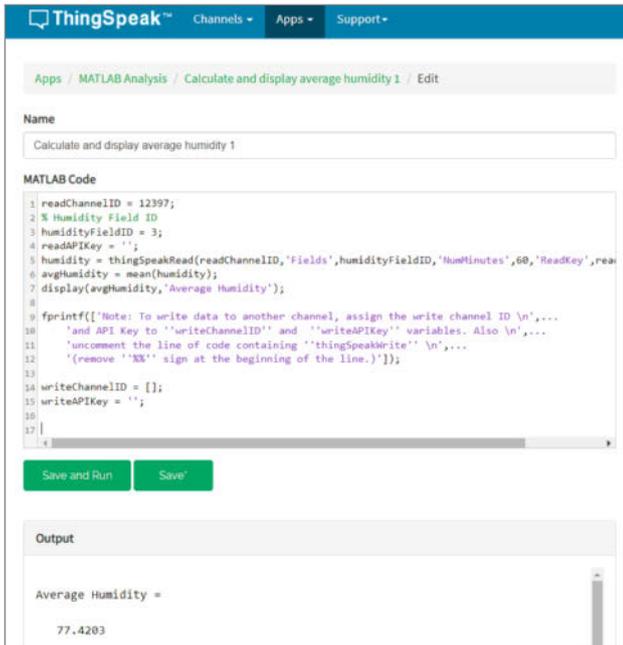
Auch der ThingSpeak Server ist Open Source (GNU GPL), sodass dem Aufbau eines

eigenen Dienstes nichts im Wege steht. Allerdings sind die oben genannten Apps nicht Teil des Quellcodes und natürlich auch nicht die Analyse mit Matlab, hier muss man dann eigene Funktionen programmieren.

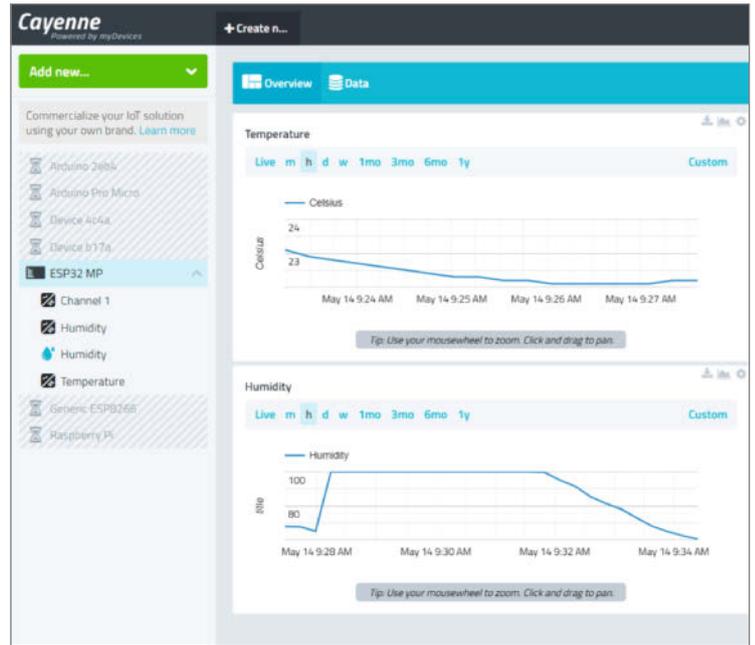
Cayenne

Cayenne führt den Claim „The world's first drag-and-drop IoT project builder“ (Der Welt erster Drag & Drop IoT-Projekt-Baukasten) prominent auf der Website. Cayenne ist ein Service von MyDevices.com, der private Gebrauch ist für bis zu zehn Geräte kostenlos, die Kosten für professionelle Nutzung erfährt man auf Anfrage.

Angemeldet ist man schnell und klickt sich nun durch die Auswahl an Boards, für die es eine Anleitung und Tutorial gibt. Hier bleibt die Auswahl zwischen Raspberry Pi und Arduino (mit diversen Netzwerk-Erweiterungen) aber recht mager. Bei Makern so beliebte Boards wie ESP32 und generell Boards mit Micro Python sind nicht verfügbar. Natürlich findet man im Internet genug Informationen, denn mit dem Cayenne-Server wird primär über MQTT kommuniziert und das ist auf fast allen Geräten möglich, siehe *Listing 3*.



ThingSpeak-Feed mittels Matlab auswerten



Dashboard von Cayenne

Im Dashboard kann man dann tatsächlich per Drag & Drop organisieren und konfigurieren, Geräte, Anzeigen, *Trigger* und *Events* hinzufügen. Es werden immer gute Vorschläge gemacht, wenn es um die Auswahl der Optionen geht, insgesamt erscheint die Auswahl der grafischen Widgets und ihre Konfigurierbarkeit aber etwas eingeschränkt. Trigger können Benachrichtigungen per SMS, E-Mail oder Webhook auslösen, wenn einzelne Daten einen gewissen Wert über- oder unterschreiten. *Events* sind faktisch Kalendereinträge, mit denen man Nachrichten zu definierten und

wiederkehrenden Zeiten an Geräte senden kann.

Auch wenn Cayenne wohl eher den professionellen Anwender im Auge hat, so gibt es doch eine, vom Hersteller MyDevices gehostete und aktive Community, mit Foren und Dokumentationen für Maker und Studierende.

Alternativen und weiterführende Systeme

Reichen die Möglichkeiten kostenloser Dienste nicht, sind die Bezahlmodelle nicht im

Budget oder möchte man die Techniken des IoT von der Pike auf erlernen, so kann man tiefer in die Materie einsteigen. Alle IoT-Dienste im Netz funktionieren ja grob ähnlich: Es gibt einen Server, der per HTTP oder MQTT mit den Geräten kommuniziert, es gibt Datenbanken, die Daten speichern, und Webserver mit Skriptsprachen, die die Daten im Netz anzeigen und auswerten. Wie gut die Schritte zwischen der Datenanlieferung und der Ausgabe gekapselt und damit vor dem Anwender verborgen werden, ist sicher ein Indiz, wie gut ein Einsteiger diese Dienste nutzen kann.

DIGITALES PRODUKTDESIGN BACHELOR OF ARTS (B.A.)

7 Semester Regelstudienzeit | Studienstart zum Wintersemester

Du möchtest mit deinen Ideen innovative Produkte entwickeln, die Welt verändern und in Bewegung bringen? Dann bewirb dich an der HBK Essen!

Bewerbung jederzeit möglich!

**HBK
ES
SEN**

Listing 3: MQTT an Cayenne senden

```

from umqtt.simple import MQTTClient
from machine import Pin
import network
import time
import dht

GPIO_DHT22_PIN = 13
temperature = 0.0

#WLAN
SSID="IhreSSID" #SSID des WLANs
PASSWORD="123456" #Passwort
SERVER = "mqtt.mydevices.com" #Server hier Cayenne
CLIENT_ID = "12312312-b3d8-11eb-883c-123456789012" #client ID
username='dead0a0-b3d3-beef-a2e4-b33ea634e422' #MQTT username
password='1234567890badas1234567890badas123456' #MQTT password
TOPIC = ("v1/%s/things/%s/data/1" % (username, CLIENT_ID))

def connectWifi(ssid,passwd):
    global wlan
    wlan=network.WLAN(network.STA_IF)
    wlan.active(True)
    wlan.disconnect()
    wlan.connect(ssid,passwd)
    while(wlan.ifconfig()[0]!='0.0.0.0'):
        time.sleep(1)

def senddata():
    # Temperatur auslesen
    try:
        dht22.measure()
        temperature = dht22.temperature()
    except OSError as o_err:
        print("Keine DHT22-Sensor Daten: '{}".format(o_err))

    time.sleep_ms(500)
    c.publish(TOPIC, str(temperature)) # An MQTT Broker veröffentlichen
    print("%d °C"%temperature)
    time.sleep(15)

# Verbinden mit WLAN/Server
connectWifi(SSID,PASSWORD)
server=SERVER
c = MQTTClient(CLIENT_ID, server,0,username,password)
c.connect()

# Setup Sensor
dht22 = dht.DHT22(Pin(GPIO_DHT22_PIN), Pin.IN, Pin.PULL_UP))

while True:
    try:
        senddata()
    except OSError:
        pass

```



Aufbereitung von Messwerten mit Grafana

Als eine Zwischenstufe zum selbst machen gibt es Anbieter, die die einzelnen Anwendungen in einer Art Baukastensystem anbieten. Der Anwender muss dann aber diese Anwendungen, etwa Datenbanken und Webserver, gut genug kennen, um eine Anwendung aufzubauen. Solche Dienste bieten z. B. *Stackhero.io* oder auch die *Grafana-Cloud* an. Für Maker bietet sich eher an, einen eigenen Server aufzusetzen, sei es auf Raspberry-Pi-Basis oder auch bei einem Server-Hoster. Dann die einzelnen Komponenten wie Webserver, MQTT-Server, Datenbank und *Grafana* zu installieren, selbst zu konfigurieren und zu administrieren. Unserer Meinung nach ist ein Raspberry Pi, den man zuerst im internen Netz verwendet, eine gute Methode, hier direkt einzusteigen, ohne sich gleich in gefährliche Gewässer im Internet zu begeben. Benutzt man dann noch *Docker*, um die Services als Container zu installieren, hat man eine sehr gute Experimentierplattform.

Meine Erdwärme-Heizung ist seit Jahren per optischem Selbstbau-Interface an einen Raspberry Pi angebunden. Ein Python-Skript fragt die Daten ab und speichert sie in einer *MySQL*-Datenbank. Aus dieser bedient sich *Grafana* und berechnet schöne Grafiken der Temperaturen und Arbeitszustände der Heizung. Alles läuft auf einem einzelnen Raspi 1 und das problemlos, jahrelang ohne Reboot. Wenn ich den Webserver-Port auf der *FritzBox* durchreiche, kann ich die Heizung auch aus dem Urlaub überwachen. Ob das jetzt schon IoT oder sinnvoll ist, weiß ich nicht, aber es ist selbst gemacht, ich habe eine Menge gelernt und es erfreut mich.

Schlussgedanken

Es gibt eine große und schwer zu überschaubare Fülle von IoT-Diensten. Mit unserer Auswahl haben wir einen kleinen Abriss über die Typen von Services gegeben vom kostenlosen Dienst zum Lernen und Testen bis zu Services, die in den bezahlten Modellen ganze Infrastrukturen von vernetzten Geräten zusammenfassen können.

Für einen Einstieg in Maker-Manier empfehlen wir *IO Adafruit* oder die *Arduino IoT Cloud*, je nachdem, ob Sie Python oder Arduino-Code bevorzugen. Fehlen dann Funktionen, kombiniert man mit *IFTTT* oder *Zapier*. *IFTTT* und *Zapier* können auch einen schnellen Einstieg bieten, wenn Sie weniger mit IoT-Boards und mehr mit fertigen Diensten von vorhandenen (auch kommerziellen) IoT-Geräten, Verbindungen und Reaktionsketten bauen möchten. Tiefer gehen dann *Cayenne* und *ThingSpeak* mit ihrem professionellen Unterbau. Wenn dies am Ende alles nicht reicht und kostenlos und frei (im Sinne von Open Source) bleiben soll, dann muss man, wie im vorherigen Abschnitt skizziert, tiefer einsteigen und eine selbst gebaute und gehostete Lösung in Betracht ziehen. —caw



AUSBILDUNG UND STUDIUM

Nicht klausurrelevant, sondern zukunftsrelevant.

Bewirb dich jetzt für unsere dualen Studiengänge
z.B. in Elektro- und Informationstechnik.

ausbildung.siemens.com



SIEMENS

Happy Lab Wien in neuer Adresse

Das *Happy Lab Wien* ist in neue Räume in der Schönngasse 15-17, 1020 Wien umgezogen. Es wurden viele neue Maschinen und Arbeitsplätze eingerichtet. Einen ausführlichen Artikel gibt es online.

heise.de/-6018546

Jugend hackt mit neuen Standorten

Die Initiative *Jugend hackt* hat Labs an bisher sechs neuen Standorten eröffnet: Erfurt, Jena, Isenbüttel, Rösrath, Ravensburg, Traunstein. Es sollen noch mindestens vier weitere Standorte folgen.

jugendhackt.org

Maker-Termine

FabLabKids@home: Schnupperkurs für den Programmierwuchs
12. oder 13.06.2021

online

fablab-muenchen.de

FabLabKids: LabGirls – Stempel-Werkstatt
19.06.2021

FabLab München e.V. – 80339 München

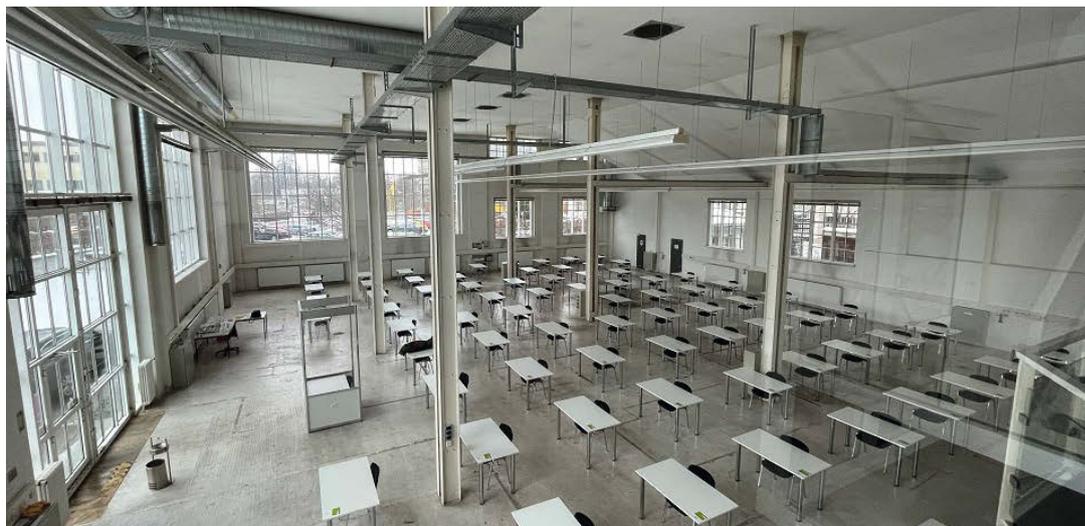
fablab-muenchen.de

Diese und weitere Termine stehen auch laufend aktualisiert in unserem Kalender auf der Webseite unter: www.heise.de/make/kalender/

Veranstalten Sie selbst?

Tragen Sie Ihren Termin in unsere Kalender ein oder schicken Sie uns eine E-Mail an:

mail@make-magazin.de



Habitat Augsburg

Finanzierung für das Habitat Augsburg

Kampagne für Umzug und Wiederaufbau erfolgreich

Als offene Werkstatt bietet das *Habitat* sowohl Privatpersonen als auch Handwerkern und Unternehmen eine Infrastruktur aus Büros, Ateliers und Werkstätten. Aktuell steht das *Habitat* jedoch vor einer großen Herausforderung: Das Gebäude, in dem derzeit die offene Werkstatt untergebracht ist, muss im Juni 2021 einer Wohnbebauung weichen. Ein neuer Standort wurde gefunden und eine Kampagne zur Finanzierung gestartet, um den Umzug und den Erhalt des *Habitats* zu ermöglichen. Der neue Standort bietet mit 760m² Maschinenhalle und 750m² Büro- und Atelierflächen Möglichkeiten für Kunst und Handwerk. Hier werden neben zusätzlichen Büro- und Projekträumen neue Werkstätten entstehen. Das

Angebot wird um den Leichtbau, eine Goldschmiede, ein Tonstudio und vieles mehr erweitert. In der Finanzierungskampagne konnte man auch ohne Mitgliedschaft Geldgeber werden und sogar mit einer Verzinsung rechnen. Die Kampagne wurde erfolgreich mit 187.000 Euro beendet. Zusammen mit Eigenkapital und einem Kredit wurden knapp 360.000 Euro aufgebracht. Dies ist bisher die größte Kampagne dieser Art, soweit uns bekannt, und eine(!) Antwort auf die immerwährende Frage, wie sich Makerspaces und Co. finanzieren lassen.

—caw

► xavin.eu/projects/habitat

Es geht was im Werkstattthaus

Werkstätten, Stadtteiltreff, Veranstaltungsort

Auf der Website heisst es: „Das Werkstattthaus ist ein wunderbarer Ort, Ideen zu entwickeln und sie umzusetzen – im Vertrauen, dass sich Lösungen entlang des Weges entwickeln.“ 2002 aus der Idee einer Nachbarschaftshilfe in Tübingen gegründet, ist das Werkstattthaus nun eine feste Institution. Heute unterstützen 190 Menschen den Verein finanziell mit ihrer Mitgliedschaft, etwa 100 ehrenamtlich Aktive kümmern sich um das Programm, planen Veranstaltungen, sind zu den Öffnungszeiten da, bieten Fachberatung und Kurse an, bringen ihr Wissen im Reparatur-Café mit ein und vieles mehr. Das monatlich stattfindende Reparatur-Café wirbt für einen nachhaltigen Umgang mit Ressourcen und verhilft all den nützlichen oder unnützen Dingen zu einem zweiten Leben. Zum Werkstattthaus gehören außerdem zwei Gemeinschaftsgärten: der *Grüne Tilsiter* und die *Wilde Linde*. Mit der Koordination für das Projekt *FAIRstrickt* existiert seit 2018 ein besonderer Schwerpunkt auf dem Thema *Faire Textilien*. Im Werkstatt-



Werkstattthaus Tübingen

haus gibt es zur Zeit folgende Werkbereiche: Fahrrad-, Holz-, Metall-, Ton- und Nähwerkstatt. Im Frühjahr 2021 sind vielfältige Aktionen und Projekte geplant, etwa Gartenpflanzen-Tauschbörsen, Holzwerkstatt-Treffs oder Workshops für Kinder.

—caw

► www.werkstattthaus.de

Reparatur und das Recht darauf

Reparatur und Recycling im Fokus

Repair Cafés und Reparaturinitiativen, zusammen mit den Themen Recycling und Müllvermeidung, wurden von der Pandemie natürlich stark beeinträchtigt. Mit großem Einsatz werden aber die Aktivitäten weiter geführt. Auch Online-Reparatur kann funktionieren, wie das *Netzwerk Reparatur-Initiativen* zeigt. Hier wird pragmatisch die Hilfe zur Selbsthilfe gegeben. Termine gibt es genug und auch wenn derzeit alles online stattfindet, kann man ja schon mal Networking betreiben, für die Zeit nach der Pandemie.

Aber auch die Aufklärung, Lobbyarbeit und politischen Initiativen laufen weiter: Das *Reparaturnetzwerk Wien* wird nicht müde aufzuklären: „Die Reparatur hängt alle anderen abfallwirtschaftlichen Maßnahmen beim Potenzial für neue Arbeitsplätze locker ab! Das belegt eine internationale Studie und beweist aufs Neue, dass der Weg aus der Wegwerfgesellschaft nur über mehr Reparatur führen kann.“ Daneben werden beim *Reparaturnetzwerk Wien* natürlich auch die praktischen Belange nicht vernachlässigt.

Einen Abriss der Entwicklung 2019 bis 2021 des Rechts auf Reparatur vom *Runden Tisch Reparatur* fasst die Entwicklung auf einer Zeitachse zusammen (alles zu finden über den Link unten). Leider mussten hier oft die von



Netzwerk Reparatur-Initiativen

Verein Runder Tisch Reparatur e.V.

Reparaturnetzwerk Wien

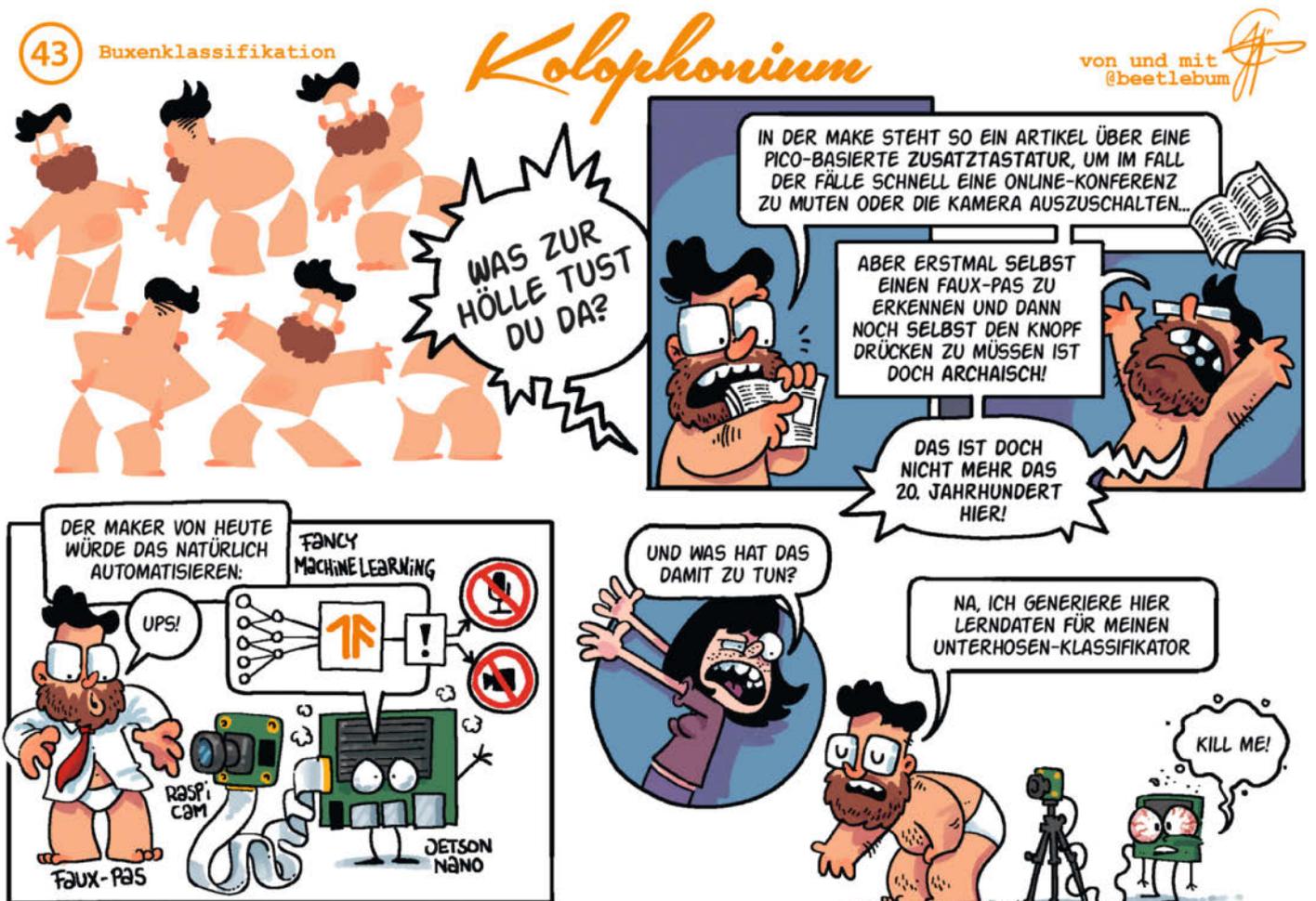
Industrie und Politik gern genutzten Worte „verspricht“, „fordern“, „wollen“, „geplant“ verwendet werden. Die positiven Aussagen sind eher in den Nachbarländern zu finden und erhielten hierzulande eine Absage. Als kleines Fazit bleibt aber, das bald Wahlen sind und wir alle am Thema dran bleiben müssen. —caw

► heise.de/-6017294

Makerspace in Reinach im Aufbau

In Reinach BL (Schweiz) werden Maker:innen zum Aufbau eines Makerspace oder Fablab gesucht. Im Moment finden die Planungstreffen online statt. Der Makerspace soll ein Ort des Austauschs, des Lernens und des generationenübergreifenden Wissenstransfers werden.

makerspace-reinach.ch



Filament-Kabeltrommel mit Leistungsmesser

Was tun mit leeren Filamentspulen? Eine Antwort auf diese typische 3D-Druck-Frage ist unsere Kabeltrommel mit Leistungsmessung. Sie ist mehr als ein einfaches Verlängerungskabel, sondern sie sagt Ihnen auch, wie viel Strom ein Gerät aufnimmt und nennt den Verbrauch in Kilowattstunden.

von Heinz Behling



Der Zufall pflanzt manchmal Ideen ins Hirn. So ging es mir auch, als ich einen Stapel leerer Filamentspulen (Marke *DevilDesign*) neben meinem 3D-Druckerschrank liegen hatte und nach einem Baumarkt-Einkauf zufällig eine Steckdose daneben ablegte. Moment, passt die Steckdose nicht in den Hohlraum der Spulen? Ja, zufällig waren das annähernd die richtigen Maße: Mit etwas Druck konnte ich die Steckdose in die Spule schieben. Da ich ohnehin eine Kabeltrommel brauchte, war das nächste Projekt geboren ❶.

In den nächsten Tagen kam noch die Idee dazu, dann auch gleich die Möglichkeit einer Spannungs-, Strom- und Leistungsmessung mit einzubauen, selbstverständlich mit Fernübertragung der Messwerte via WLAN. Die entsprechende Hardware war schnell gefunden: ein Stromsensor vom Typ PZEM004 sowie ein Wemos-D1-mini-Modul, dessen ESP8266 die Datenübertragung übernehmen sollte. Für die Stromversorgung des ESP kam dann noch ein Netzteilmodul dazu.

Stromsensor in zwei Versionen

Den PZEM004 gibt es in zwei Versionen, wie ich durch ein unfreiwilliges Experiment herausfand: Der PZEM00410A ❷ ist für Wechselströme bis 10A ausgelegt, sein Bruder PZEM004100A ❸, der den Strom mit einem zusätzlichen induktiven Ringkern-Sensor misst, entsprechend für bis zu 100A. Bis auf einen kleinen Aufdruck und eine andere Leiterbahnführung an der Unterseite sehen sich beide zum Verwechseln ähnlich. Ich bestellte



Unbedingt beachten

Bei diesem Projekt wird mit Netzspannung gearbeitet. Sie sollten daher wissen, was sie tun. Falls Sie nicht über das entsprechende Fachwissen verfügen, sollten Sie die Arbeiten an den netzspannungsführenden Leitungen einer Fachkraft überlassen.

Wenn die Kabeltrommel nicht richtig funktioniert, können Sie sich über die USB-Leitung mit Hilfe des seriellen Monitors der Arduino-IDE alle Meldungen des Wemos-Moduls anzeigen lassen. Dazu muss jedoch das Elektronik-Gehäuse offen betrieben werden, wodurch netzspannungsführende Teile frei zugänglich sind. Dies darf nur durch Fachleute gemacht werden. Sie handeln hier auf eigene Gefahr!

Kurzinfo

- » Fähigkeiten des Strom-Messmoduls PZEM004
- » Weitergabe der Messdaten mittels ESP8266
- » Zwei Versionen für 10A und 16A Strombelastbarkeit

Checkliste



Zeitaufwand:
2 Stunden



Kosten:
30 bis 35 Euro

Werkzeug

- » 3D-Drucker
- » Bohrmaschine
- » Quetschzange für Kabelendhülsen

Mehr zum Thema

- » Florian Schäffer, Stromverbrauchsmonitor mit ESP32, Make-Sonderheft Energie 2018, S. 76

Material

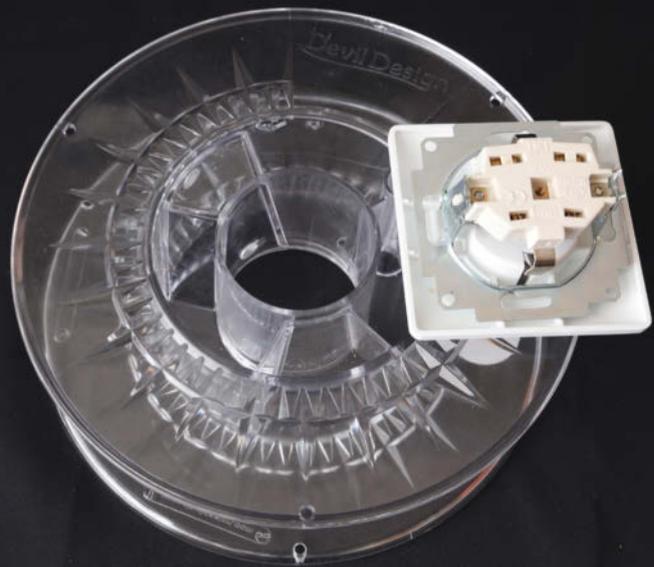
- » Filamentspule mit mind. 53mm Innendurchmesser
- » 10m Kabel 31mm² für 10A / 3 × 1,5mm² für 16A Strombelastbarkeit
- » Schukostecker
- » Schukosteckdose Unterputz
- » Strom-Messmodul PZEM004-10A (für 10A) oder PZEM004-100A (mit Messspule für 16A)
- » Wemos D1 mini
- » Netzteilmodul Eingang: 230V, Ausgang: 5V/700mA
- » Schalltülle ca. 0,5 m
- » Druckfilament
- » Kabelendhülsen

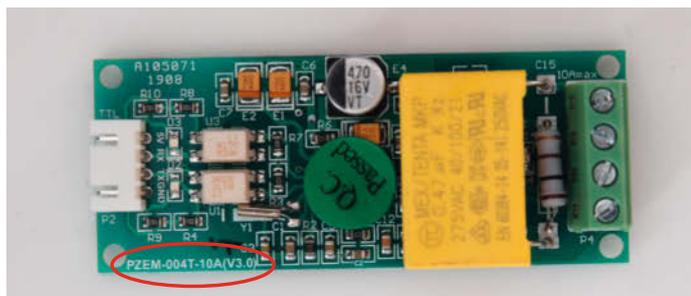
Alles zum Artikel im Web unter make-magazin.de/xwdr

beide, erkannte aber aufgrund der Ähnlichkeit nicht, dass man mir versehentlich zweimal die 100A-Version schickte. Die hat jedoch eine andere Belegung der Schraubklemmen für die Netzspannung und den Verbraucher. Ich schloss ihn als 10A-Ausgabe an und steckte

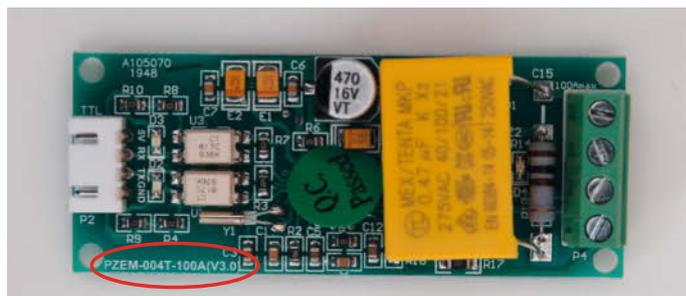
den Netzstecker ein. Prompt hatte ich eine sehr präzise 0-Volt-Stromquelle für den entsprechenden Sicherungs-Stromkreis, sprich: einen Kurzschluss. Erst daraufhin sah ich mir das Modul genau an und entdeckte die Falschlieferrung.

❶ Spule und Steckdose, das muss doch zusammenpassen.





2 Die 10A-Version des Stromsensors ...



3 ... sieht der 100A-Version zum Verwecheln ähnlich.

Zwei Leistungsklassen

Ich plante zwei Versionen der Kabeltrommel: eine mit 10A-Belastbarkeit (das entspricht bei einer ohmschen Last einer Leistung von 2300W). Sie kommt mit einem Kabel von 3x1mm² Querschnitt aus. Davon passen etwa 15 bis 20m auf die 1kg-Spule, je nach Isolationsdurchmesser. Die zweite Version mit bis zu 16A Belastbarkeit (bis zu 3680W) braucht ein Kabel mit 3 x 1,5mm² und die 100A-Version des Sensors. Von diesem dickeren Kabel passen nur etwa 10m auf die Spule.

Dementsprechend gibt es zwei Schaltpläne für die Kabeltrommel. In Bild 4 sehen Sie die Verdrahtung für die 10A-Version. Hier wird der Verbraucherstrom direkt über die Sensorplatine geführt. Achten Sie darauf, Kabelendhülsen über die Aderenden zu schieben, bevor Sie sie festschrauben.

Bild 5 ist der Plan für die 16A-Ausgabe. Hier wird der Strom induktiv mit einem Ringkern-Sensor gemessen, durch den eine(!) Ader der Verbraucherleitung geführt wird. Übrigens: Auch wenn der Sensor wesentlich mehr Strom vertragen könnte: Steckdose, Kabel und Netzstecker sind nur für 16A ausgelegt. Deshalb bitte später beim Gebrauch auf die maximale Belastung achten! Die wird ja passenderweise gemessen.

Zusammenbau

Im folgenden schildere ich den Zusammenbau der 10A-Version. Die 16A-Ausgabe ist sehr ähnlich. Lediglich der Ringkern-Sensor muss zusätzlich entsprechend dem Schaltplan angeschlossen und im Hohlraum der Spule untergebracht werden. Die Elektronik beider

Varianten wird in einem zweiteiligen 3D-Druck-Gehäuse untergebracht 6. Da die Druckzeit recht lange ist, sollten Sie damit anfangen. Ich habe mit PETG gedruckt, PLA ist aber auch möglich. Füllung: 15% bei PETG, 20% bei PLA. Das Oberteil muss mit Stützstruktur gedruckt werden. Die Druckdateien sind über den Kurzinfo-Link erhältlich.

Die Filamentspule hat einen doppelwandigen Kern. Um das Kabel nach innen führen zu können, muss zunächst ein Loch entsprechend dem Kabeldurchmesser (8 bis 10mm) durch beide Wandungen gebohrt werden. Dabei nicht zu stark drücken, sonst bricht etwas. Anschließend entfernen Sie am Kabel etwa 30cm der äußeren Isolierung, führen das Kabel durch die beiden Bohrungen und befestigen zwischen den beiden Kernwänden einen Kabelbinder als Zugentlastung 7.

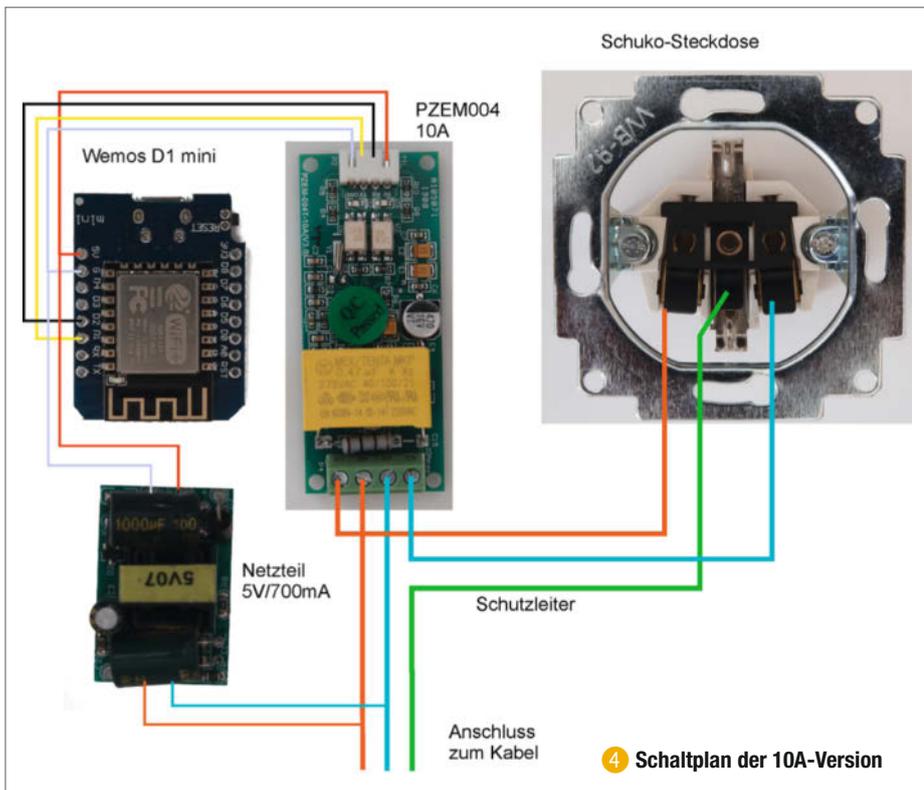
Danach schließen Sie die Leitungsenden an den Steckdoseneinsatz an, stecken diese in die Öffnung der Spule und befestigen sie mit den beiden seitlichen Klemmkralen. Eventuell müssen Sie die Spitzen der Krallen etwas flach biegen, damit die Steckdose komplett in den Hohlraum passt.

Die Zeit des 3D-Drucks nutzen Sie am besten für den Zusammenbau der Elektronik entsprechend des jeweiligen Schaltplans. Lediglich die Adern zum Kabel und zur Steckdose lassen Sie noch frei.

Dann ist es Zeit, den ESP8266 zu programmieren. Das geschieht mit der Arduino-IDE. Der Sketch namens *Kabeltrommel.ino* 8 steht über den Kurzinfo-Link ebenfalls zum Download bereit. Die Arduino-IDE muss mit den Informationen zum ESP-Board ausgestattet sein. Ist das noch nicht der Fall, finden Sie in der IDE unter *Datei* und *Voreinstellungen* das Feld *Zusätzliche Boardverwalter-URLs*. Setzen Sie dort die Adresse

http://arduino.esp8266.com/stable/package_esp8266com_index.json

ein und bestätigen Sie mit *OK*. Anschließend wählen Sie unter *Werkzeuge* und *Board* den *Boardverwalter*. Suchen Sie dort nach *esp8266* und installieren Sie das so gefundene Paket in



4 Schaltplan der 10A-Version

der neuesten Version. Außerdem brauchen Sie die Bibliothek *PZEM004Tv30* (siehe Kurzinformato-Link). Die müssen Sie als ZIP-Datei unter *Sketch/Bibliothek einbinden/.ZIP-Bibliothek hinzufügen* installieren.

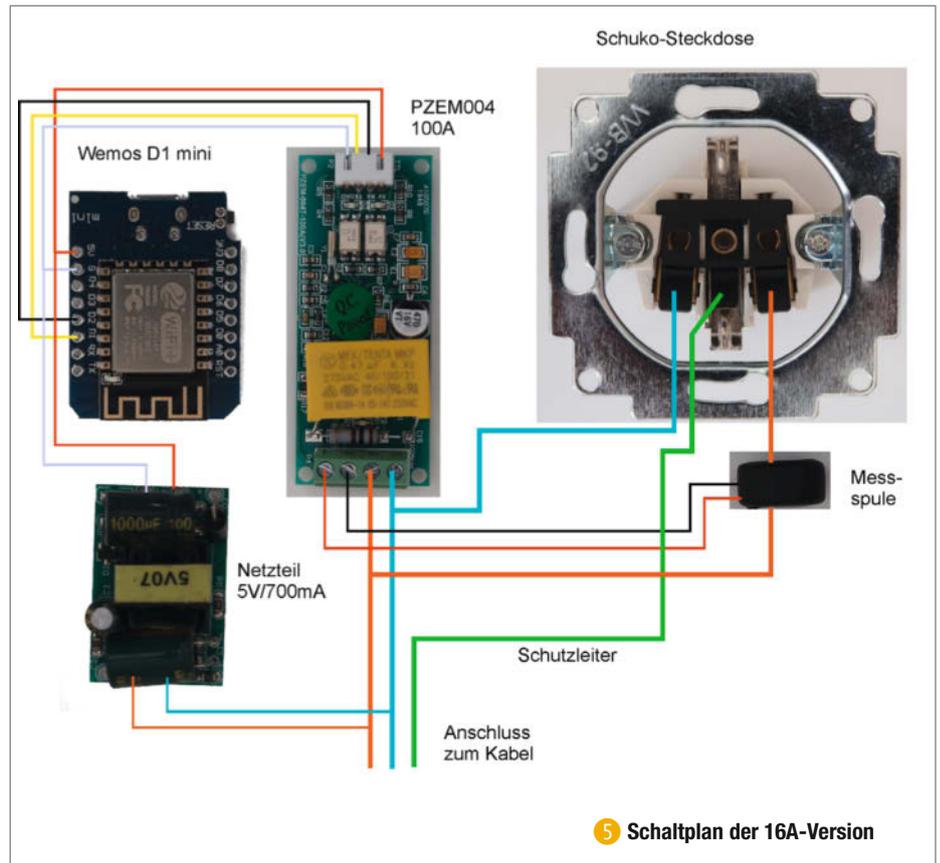
Dann laden Sie den Sketch. In den Zeilen 3 und 4 müssen der Name des WLANs sowie das dazugehörige Passwort eingetragen werden. Nach dem Anschluss des Wemos-D1-mini-Moduls über USB wählen Sie unter *Werkzeuge* als *Board* das *LOLIN (WEMOS) D1 R2 & mini* sowie den richtigen Port.

Jetzt ist alles bereit. Ein Klick auf den Links-pfeil in der Symbolleiste der IDE, und der Sketch wird kompiliert und auf das Wemos-Modul übertragen.

Endmontage

Wenn das Druck-Gehäuse fertig ist, entfernen Sie die Stützstrukturen. Im Inneren des Gehäusedeckels sind drei Fächer, in die der Sensor (Mitte), das Wemos-Modul sowie das Netzteilchen passen. Doch als erstes muss der Gehäuseboden mit zwei M3 x 8-Schrauben und Muttern auf der Spule befestigt werden. Entsprechende Löcher sind im Boden bereits eingedruckt. Die Gegenstücke auf der Spule müssen Sie noch bohren. Also den Boden mit dem Zentriering auf die Spule auflegen, die beiden Bohrlöcher in den Sechskantlöchern anzeichnen und bohren.

Später wird der Deckel ebenfalls mit M3-Schrauben und Muttern an den Boden geschraubt. Dazu müssen vor dem Anschrauben des Bodens an die Spule aber die Muttern in die vier Sechseck-Öffnungen eingelegt werden. Die liegen dann zwischen Boden und Spule. Damit sie nicht herausfallen, schrauben Sie die M3x8-Schrauben locker hinein. Die halten dann die Muttern in Position. Jetzt kann der Boden an der Spule befestigt werden. Führen Sie außerdem die blaue und die braune Ader der Steckdosenleitung durch das



5 Schaltplan der 16A-Version

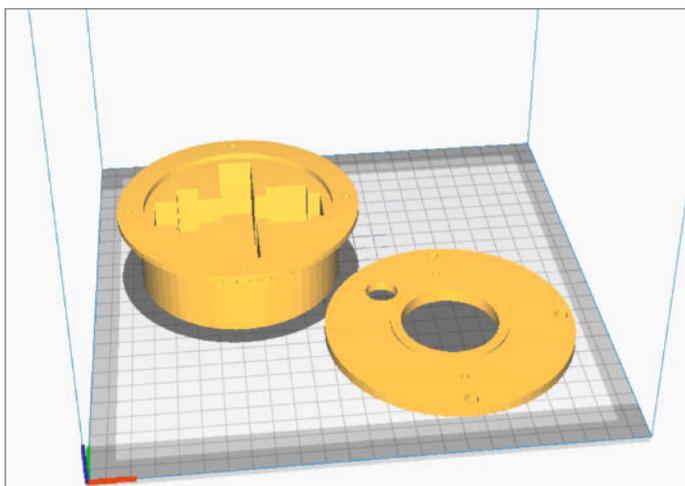
Loch im Boden und schneiden Sie die Leitungen in der Mitte durch 9.

Die vier Adern vom Kabel und der Steckdose schließen Sie dann entsprechend dem Schaltplan an den Stromsensor an. Verstauen Sie dann die Elektronik im Gehäusedeckel 10.

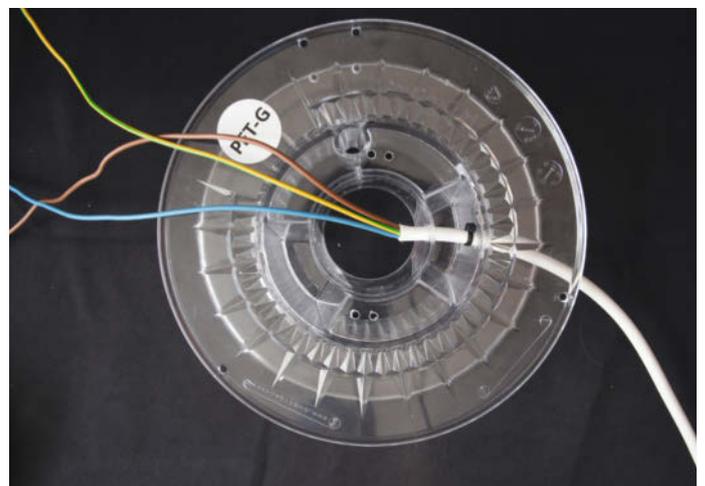
Jetzt ist nur noch der Deckel anzuschrauben und die Kabeltrommel ist fertig, bis auf den Netzstecker am anderen Ende der Leitung. Dazu können Sie jeden beliebigen Schuko-Stecker verwenden. Ich fand im Baumarkt-Sortiment aber ein spezielles Modell: Es ist ein Flach-

stecker mit aufklappbarem Griff. Bei aufgewickelter Leitung kann man den Stecker in die Steckdose stecken und hat so einen praktischen Tragegriff für die Kabeltrommel 11.

Jetzt geht's ans Ausprobieren: Wickeln Sie das Kabel von der Rolle ab, schließen Sie einen Verbraucher an (im Beispiel ist es ein Wäschetrockner) und stecken Sie den Stecker des Kabels ein. Ihr Router verrät Ihnen die IP-Adresse der Kabeltrommel. Geben Sie die im Browser ein. Daraufhin erscheinen die Messwerte 12.



6 Die beiden Teile des Elektronik-Gehäuses



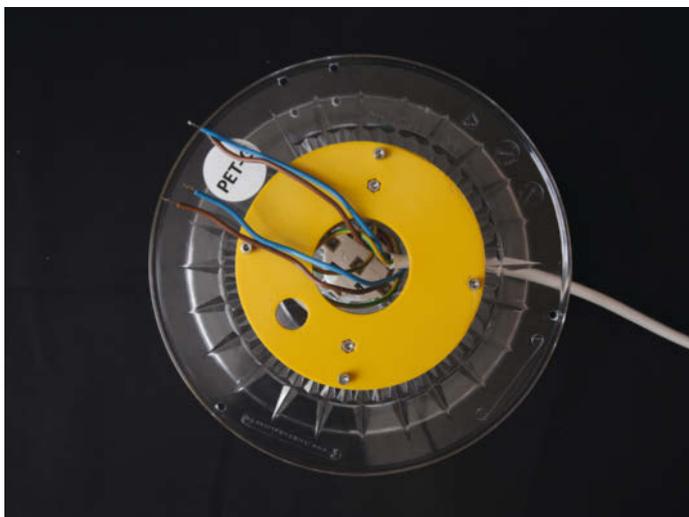
7 Der Kabelbinder als Zugentlastung ist wichtig.

8 Kabeltrommel.ino

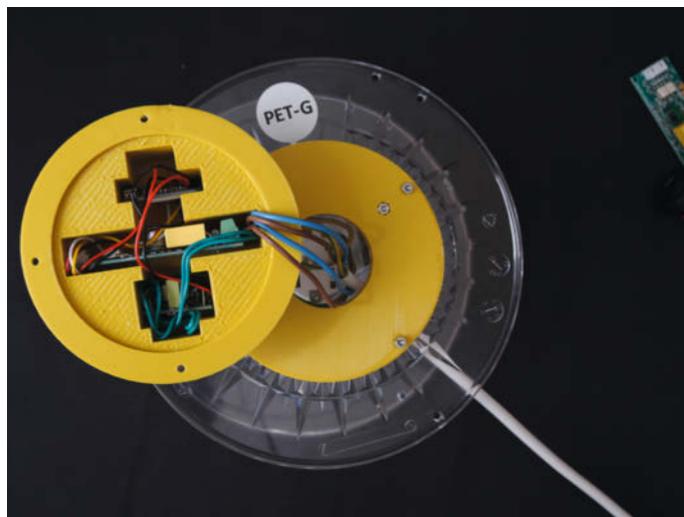
```

01 #include <ESP8266WiFi.h>
02 #include <PZEM004Tv30.h>
03 const char* ssid = "WLAN-Name";
04 const char* password = "WLAN-Passwort";
05 WiFiServer server(80);
06 PZEM004Tv30 pzem(D1,D2);
07
08 void setup() {
09   Serial.begin(115200);
10   Serial.println();
11   Serial.println();
12   Serial.print(F("Connecting to "));
13   Serial.println(ssid);
14   WiFi.mode(WIFI_STA);
15   WiFi.begin(ssid, password);
16   while (WiFi.status() != WL_CONNECTED) {
17     delay(500);
18     Serial.print(F("."));
19   }
20   Serial.println();
21   Serial.println(F("WiFi connected"));
22   // Start the server
23   server.begin();
24   Serial.println(F("Server started"));
25   // Print the IP address
26   Serial.println(WiFi.localIP());
27 }
28 void loop() {
29   WiFiClient client = server.available();
30   float voltage = pzem.voltage();
31   float current = pzem.current();
32   float power = pzem.power();
33   float energy = pzem.energy();
34   float frequency = pzem.frequency();
35   float pf = pzem.pf();
36   client.print("<head><title>Make-Kabeltrommel</title><meta http-equiv='refresh' content='5' /></head>");
37   client.print("<img src='http://www.heise.de/make/icons/make_logo.png' alt='Make:' height=10%><h1>Kabeltrommel</h1><br>");
38   client.print("<table>");
39   client.print("<tr><td><b>Spannung:</b></td><td>"); client.print(voltage); client.print("<br></td></tr>");
40   client.print("<tr><td><b>Strom:</b></td><td>"); client.print(current); client.print("<br></td></tr>");
41   client.print("<tr><td><b>Leistung:</b></td><td>"); client.print(power); client.print("<br></td></tr>");
42   client.print("<tr><td><b>Arbeit:</b></td><td>"); client.print(energy); client.print("<br></td></tr>");
43   client.print("<tr><td><b>Frequenz:</b></td><td>"); client.print(frequency); client.print("<br></td></tr>");
44   client.print("<tr><td><b>Leistungsfaktor:</b></td><td>"); client.print(pf); client.print("<br></td></tr>");
45   client.print("</table>");
46 }

```



9 So muss der Boden auf der Spule befestigt werden.



10 Die Fächer im Deckel passen genau zur Elektronik.



11 Flachstecker mit Tragegriff

Software

Das Programm 8 ist sehr einfach aufgebaut. In den Zeilen 1 bis 6 werden die erforderlichen Bibliotheken geladen, die WLAN-Daten, die Port-Adresse des Kabeltrommel-Servers sowie die am Wemos-Modul für den Stromsensor benutzten PIN-Nummern festgelegt.

In den Zeilen 8 bis 27 werden der Server und die Datenausgabe über die serielle Schnittstelle (via USB) gestartet. Über den USB-Bus kann man sich nämlich für Testzwecke beispielsweise mit dem seriellen Monitor der Arduino-IDE sämtliche Werte anzeigen lassen. Lesen Sie dazu bitte den Warn-Kasten der zweiten Seite dieses Artikels.

Make:

Kabeltrommel

Spannung:	223.00V
Strom:	8.12A
Leistung:	1810.40W
Arbeit:	0.04kWh
Frequenz:	49.90Hz
Leistungsfaktor:	1.00

12 Der Wäschetrockner langt ganz gut zu.

Die Zeilen 30 bis 35 lesen die sechs Messwerte aus dem Sensor aus. In den Zeilen 36 bis 46 wird nun ein HTML-Text an den angeschlossenen Client, also den Browser, geschickt. Der bewirkt, dass die Messwerte in Form einer kleinen Tabelle sauber untereinander angeordnet wie in Bild 12 angezeigt werden. —hgb

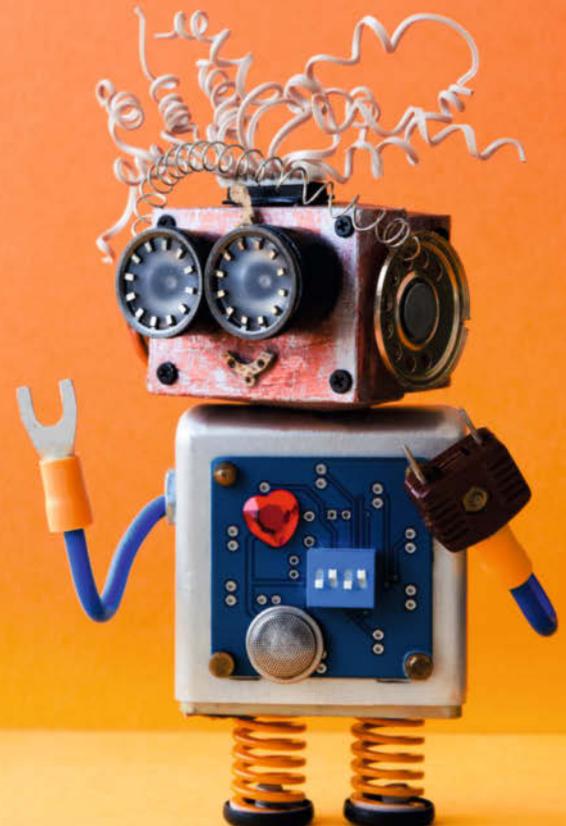
HANNOVER

Gründungswissen kompakt: 18.06. von 11 bis 16 Uhr

MAKER AUFGEPASST!

MACH AUS DEINER IDEE DEIN UNTERNEHMEN.

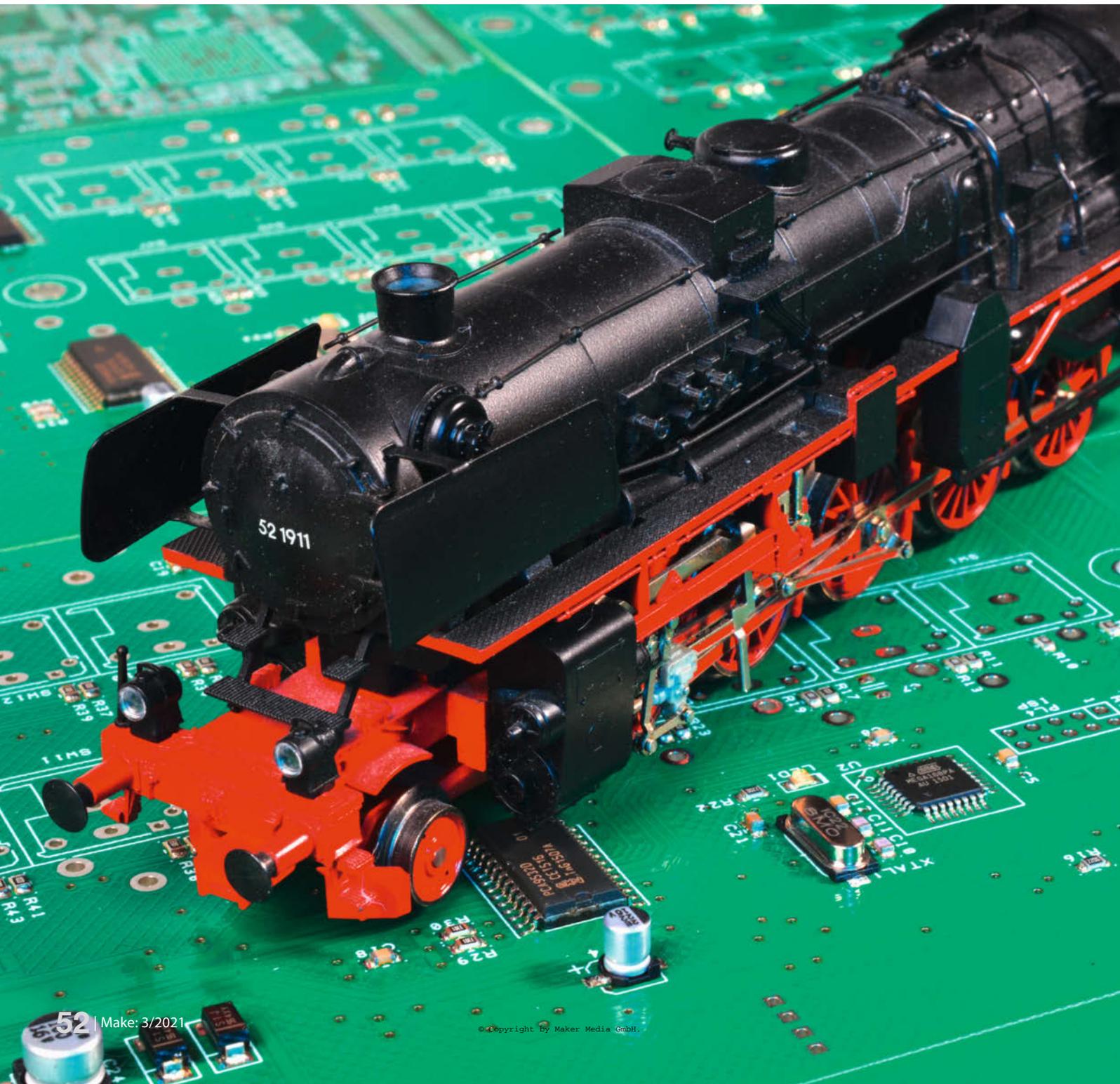
Wir zeigen dir wie. Besuch uns auf der Maker Faire Hannover.



Arduino als Lokführer

Modellbahnanlagen bieten ein zeitloses Bastelvergnügen – sie werden nie ganz fertig. Allerdings ist der Spaß kein preiswerter: Gerade beim Digital-Zubehör schlagen die Hersteller kräftig zu. Wer hier sparen will, darf sich auf das postmaterielle Basteln (sprich: das Arduino-Programmieren) verlegen.

von Till Harbaum



Die Digitalisierung der Modellbahn vor über 35 Jahren eröffnete ungeahnte Möglichkeiten: Der unabhängige Mehrzugbetrieb, Dauer-Zugbeleuchtung, ferngesteuerte Sonderfunktionen, Gleisbildstellwerke am PC mit automatischem Zugbetrieb und nicht zuletzt eine enorm vereinfachte Verdrahtung sprechen für die Digitalisierung, auch wenn die Umrüstung einer großen Anlage Unsummen verschlingt.

Dass sich selbst die einfachen Arduinos als Steuerzentrale eignen, wurde bereits in der c't-Hacks-Ausgabe 1/13 gezeigt, indem eine sogenannte Gleisbox vom Arduino aus per CAN-Bus-Shield angesprochen wurde. Als dem Autor eine kleine Märklin-Sammlung aus der Erbmasse des Opas zufiel, stellte sich auch ihm die Frage: Was damit machen? Die Sammlung enthielt einige Loks, die sich allesamt als digitalfähig herausstellten, aber nur eine sehr einfache zugehörige Fernbedienungseinheit. Die alte Lösung hätte also direkt die zusätzliche Anschaffung einer Gleisbox erfordert, die es zu vermeiden galt. Könnte man nicht elegant mit einem Arduino und einem verbreiteten Motorshield direkt die nötigen Signale für die Loks erzeugen?

Allein Märklin hat über die Zeit ein ganzes Sammelsurium an Protokollen verwendet, von „Motorola“ über „erweitertes Motorola“ bis „MFX“; dazu kommen Protokolle von Drittherstellern wie DCC. Ein genauerer Blick zeigt aber, dass praktisch alle Systeme rückwärtskompatibel sind und sich sämtliche neueren Loks auch mit alter Technik und alten Protokollen ansprechen lassen.

Historisch gewachsen

In der vordigitalen Zeit hat Märklin seine Modellbahnen mit 50Hz-Wechselspannung betrieben. Mit einfachen Trafos wurde eine von 0 bis rund 18V einstellbare Wechselspannung direkt an die Schiene und damit an die Lok angelegt. Je höher die Spannung, desto schneller fuhr die Bahn. Für den Richtungswechsel wurde ein kurzer Spannungsstoß von mehr als 20V auf die Schiene gegeben, der in der Lok einen Schalter umlegte und die Fahrtrichtung änderte, eine Besonderheit der Märklin-typischen Allstrom-Reihenschlussmotoren.

Hier setzen die Digitalsysteme an. Den Motoren der Loks und den Lampen der Beleuchtung ist die Frequenz nämlich vollkommen egal. Sie laufen zur Not auch mit Gleichspannung und sind in keiner Weise auf eine bestimmte Art des Spannungswechsels angewiesen. Diese Tatsache nutzen Digitalsysteme, indem sie den Wechsel der Spannungsrichtung zur binären Kommunikation verwenden – also eine Art Energie-Datenbus. So lässt sich durch Wechsel der Spannungsrichtung eine beliebige Bitfolge auf die Schiene geben, ohne dass sich aus Sicht von Lok und Lämpchen

Kurzinfo

- » Prinzip der digitalen Modellbahn-Steuerung
- » Nachbildung des Motorola-Protokolls mit dem Arduino
- » Arduino-Shields als Treiber

Checkliste

-  **Zeitaufwand:**
2 bis 3 Stunden
-  **Programmieren:**
Arduino-IDE beherrschen
-  **Kosten:**
15 Euro

↓ Alles zum Artikel im Web unter make-magazin.de/xen

Material

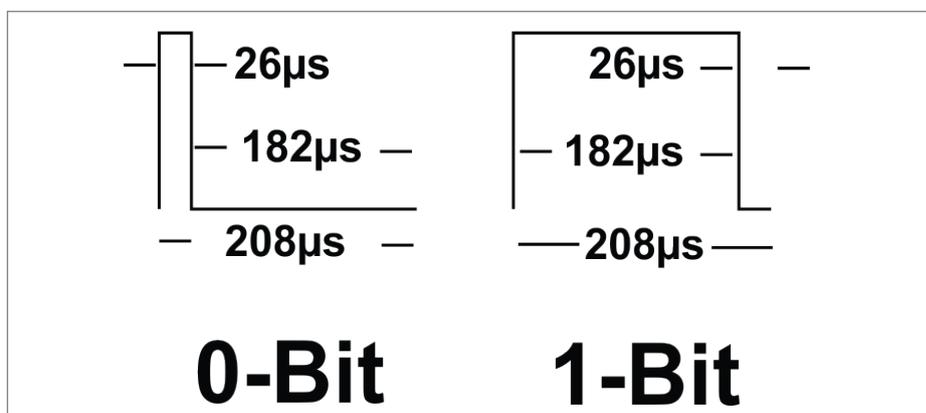
- » **Arduino Micro Pro** oder *Leonardo*, ggf. Nachbau
- » **Motor-Shield** mit mind. 18V Spannungsfestigkeit, z.B. *Debo Motodriver2* von Reichelt
- » **Netzteil** 15V bis 20V, min. 1A

Mehr zum Thema

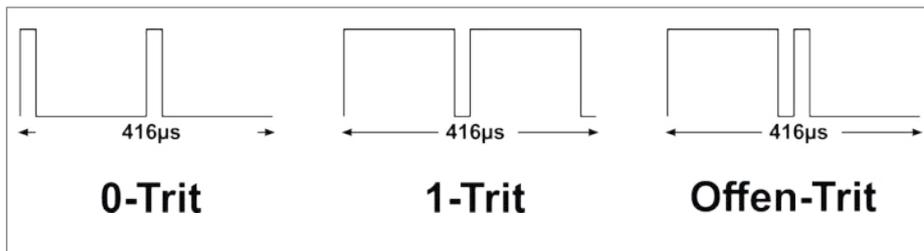
- » Jörg Pleumann, Arduino steuert Märklin-Modellbahn, c't Hacks 1/13, S. 54



Auch der Redakteur musste mal entspannen: Vor über 25 Jahren steuerte das Hypercard-Gleisbildstellwerk auf einem Macintosh Quadra die Modellbahn im Hintergrund.



1 Die Bits oder besser *Halb-Tritts* des Motorola-Formats. Zwei davon bilden ein *Trit*.



2 Motorola-Trits, zusammengesetzt aus je zwei High-Low-Impulsen.



Innenleben einer neueren Digital-Dampflokomotive: Rechts der Motor in klassischer Bauform, aber mit Permanent-Magneten, in der Mitte der Digital-Dekoder.

irgendwas Bedeutsames an der Energieversorgung geändert hat.

Wenn man direkt Nullen und Einsen über die beiden Polaritäten kodiert, bekommt man schnell das Problem, dass bei längeren 1- oder 0-Sequenzen die Länge sehr genau gemessen werden muss. Zusätzlich verfälschen Störungen, wie sie zwangsläufig bei der Übertragung von der Schiene auf Schleifer und Räder der Loks auftreten, die Datenübertragung. Märklin hat sich daher spezieller Fernbedienungs-ICs von Motorola bedient, an kompakte Mikrocontroller wie PIC und AVR war 1984 ja noch nicht zu denken. Die ICs MC145026, MC145027 und MC145028 wurden speziell zur Übertragung kurzer Nachrichten über eine potenziell gestörte einzelne Verbindung konzipiert.

Ein übliches Setup besteht aus Sender MC145026 und einem oder mehreren Empfänger

MC145027/28/29. Statt eines festen High- oder Low-Pegels übertragen diese Chips immer einen High-Low-Puls wie in 1 dargestellt. Die Pulslängen müssen in gewissen Grenzen gehalten werden. Zu sehr abweichende Pulslängen werden als Störung interpretiert und vom Empfänger ignoriert. Die Tatsache, dass jedes Bit einen Signalwechsel enthält, führt zusätzlich dazu, dass sich auch lange konstante 1- oder 0-Bit-Sequenzen eindeutig dekodieren lassen.

Bits und Trits

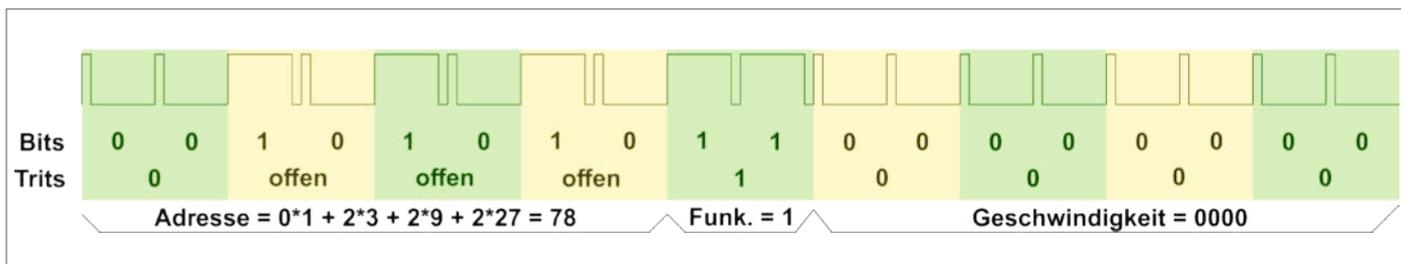
Die Bausteine der (inzwischen von NXP gefertigten) MC14502X-Reihe übertragen statt „richtiger“ Bits ein Signal, das auf drei Zuständen basiert; deren Zeichen werden daher auch Trits genannt. Jedes Trit besteht aus zwei

der oben erwähnten High-Low-Impulsfolgen (Halb-Trits), wobei eine der vier möglichen Kombinationen (kurz-lang, 0-1) ungültig ist 2.

Die Kombination lang-kurz heißt beim Motorola-Format offen, weil sie früher einen offenen (also nicht mit Masse oder Betriebsspannung verbundenen) Adress-Kodiereingang repräsentierte. Durch diesen Trick ließen sich Anschlusspins sparen: Bei vier Eingängen hatte man nun nicht 16, sondern 81 verschiedene Kodiermöglichkeiten. Eine komplette Nachricht eines MC145026-Senders besteht aus insgesamt 9 Trits beziehungsweise 18 Impulsen wie in Abbildung 3 dargestellt, während Bild 4 eine solche Nachricht zeigt, wie sie bei der Benutzung eines Märklin-Handsteuergerätes direkt an der Schiene gemessen wurde.

Diese Aufteilung von Kodierung (Adresse) und Befehl ist willkürlich; Märklin verwendet in den Lokomotiven vier Trits zur Adressierung der anzusprechenden Lok, die übrigen fünf Trits stellen den Befehl (Sonderfunktion und Geschwindigkeit) dar. Nimmt man an, dass der offen-Zustand die Wertigkeit 2 hat und dass das niederwertige Trit zuerst gesendet wird, ergibt sich aus der Folge 0-offen-offen-offen-offen der Dezimalwert $0 \times 1 + 2 \times 3 + 2 \times 9 + 2 \times 27 = 78$. Diese Adresse ist die Standardadresse, die Märklin ab Werk für Dampflokomotiven wie die kleine BR89 verwendet. Andere gängige Adressen sind 72 für Triebwagen, 60 (wie im Oszilloskop-Screenshot zu sehen) für Dieselloks und 24 für Elektroloks. Nur genau diese vier Adressen können auch die ganz einfachen Fernsteuersender von Märklin ansprechen. Die Loks selbst lassen sich aber per internen DIP-Schaltern oder bei neueren Decodern durch spezielle Kommandos auf jede beliebige der 81 mit vier Trits darstellbaren Adressen einstellen.

Die fünf der Adresse folgenden Daten-Trits (Befehl) werden nur binär genutzt, nehmen also nie den Zustand *offen* an. Diese Einschränkung bietet wiederum zusätzlichen Schutz vor Störsignalen. Das Funktions-Trit schaltet üblicherweise die Beleuchtung der Lok an bzw. aus, während die letzten vier Trits wieder mit dem niederwertigsten Trit voran die Geschwindigkeit der Lok von 0 bis 15 vorgeben. Der Wert 1 hat dabei eine besondere



3 Ein komplettes Datenpaket im Motorola-Format besteht aus 9 Trits: 4 für die Adresse, 5 für den Befehl. Letztere sind ausschließlich als 0-0 oder 1-1 kodiert, also binär.

Bedeutung, er entspricht dem Richtungswechsel-Spannungsstoß der analogen Märklin-Bahnen und lässt den Motor noch nicht anlaufen.

Wie der elektromechanische Fahrtrichtungsschalter merkt sich der Decoder übrigens die letzte Fahrtrichtung: Auch nach längerer Stromunterbrechung fährt die Lok unbeirrt in der bisherigen Richtung weiter, ganz wie ihre analogen Schwestern. Der Decoder merkt sich auch die letzte Geschwindigkeit und Sonderfunktion – wichtig, wenn mehrere Loks auf dem selben Gleis fahren sollen: Dann sendet man reihum die Datensätze für jedes einzelne Triebfahrzeug.

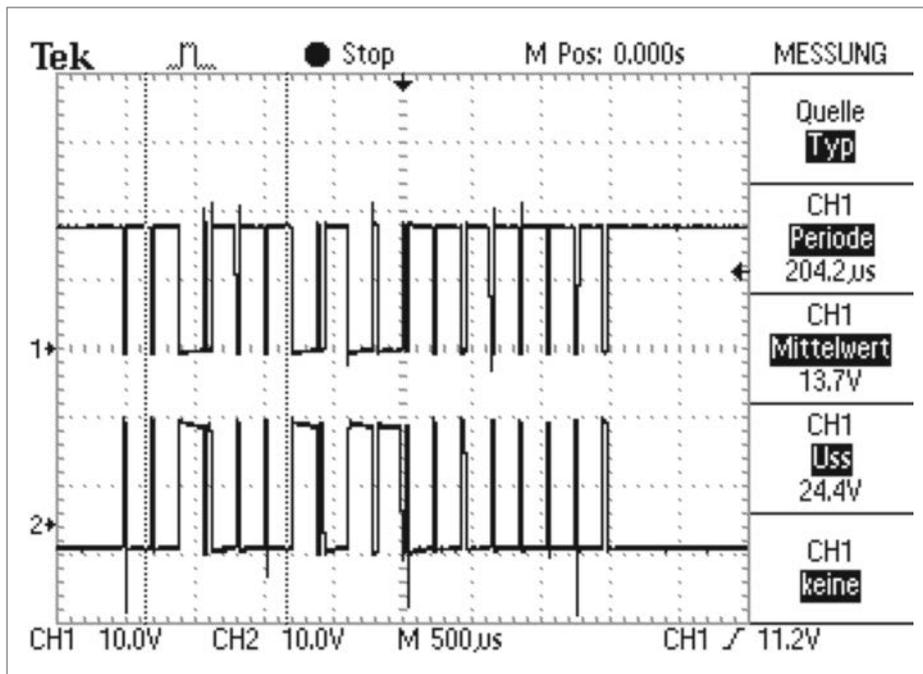
Nachmacher

Zwei Dinge bleiben, um diese Signale mit einem Arduino selbst zu erzeugen: Erstens muss das eigentliche Signalmuster exakt reproduziert werden, und zweitens muss statt der Arduino-üblichen 5-Volt-Signale eine ±15V-Wechselspannung mit ausreichender Leistung (einige Watt) erzeugt werden.

Auf den ersten Blick scheint es naheliegend, das Signal Bit für Bit in Software zu erzeugen, indem man per `digitalWrite()`-Funktion einen Ausgangspin schaltet und per `delay()`-Funktion die passenden Zeiten einhält. Machbar ist das sicher, aber der Arduino ist so mit der reinen Signalerzeugung ausgelastet. Zeit, um zum Beispiel von einem PC Befehle entgegenzunehmen, bleibt da kaum. Zum Glück lässt sich ein großer Teil der Arbeit auf die im Arduino verbaute Peripherie auslagern.

Ein Motorola-Bit beginnt immer mit einer High-Phase von 26µs Länge und endet immer mit einer Low-Phase von 26µs Länge. Dazwischen befinden sich 156µs High oder Low, je nachdem, ob es sich um ein 1- oder 0-Bit handelt. Das ähnelt sehr einem normalen seriellen Signal, das immer mit einem Startbit (0) beginnt und mit einem Stopbit (1) endet. Dazwischen liegen 6 Bits, die jeweils alle 1 oder 0 sind, je nach zu übertragendem Bitmuster. Dass die Polarität das Gegenteil dessen ist, was wir erzeugen wollen, stört uns zunächst nicht, da wir für die Wechselspannung an der Schiene später sowieso auch das invertierte Signal benötigen. Ein UART-Datenformat von 6N1 würde also genau unser Muster erzeugen können **5**. Auch die Bitlänge von 26µs kommt uns sehr entgegen: 26µs entsprechen 38461,54 Bit/s bzw. genau 208 Taktzyklen des mit 16MHz betriebene Arduino.

Die nötige Bitrate lässt sich also durch die serielle UART-Hardware des Arduino exakt reproduzieren. Falls Sie vorhaben, auch Weichen und Signale digital zu steuern, gleich hier ein Hinweis: Die Weichendecoder arbeiten mit dem gleichen Protokoll, aber verdoppelter Datenrate; dafür sind die Zeiten einfach zu halbieren. Weichendecoder-Befehle können an beliebiger Stelle in den Datenstrom eingefügt werden,



4 Einzelnes Datenpaket, Messung direkt an der Schiene mit einem Speicher-Oszilloskop

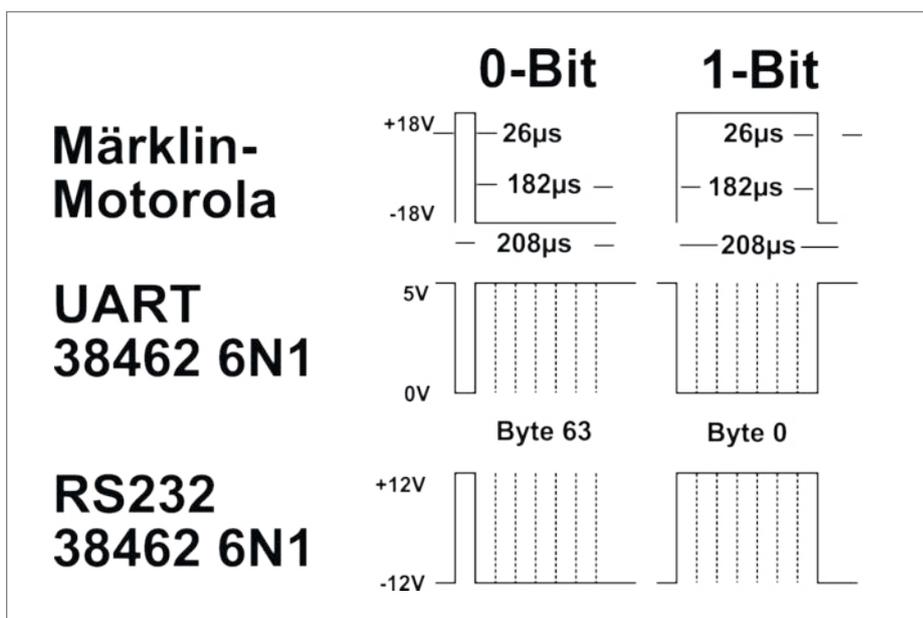
da die Lokdecoder wegen der unpassenden Datenrate nicht darauf reagieren.

USB-Arduino

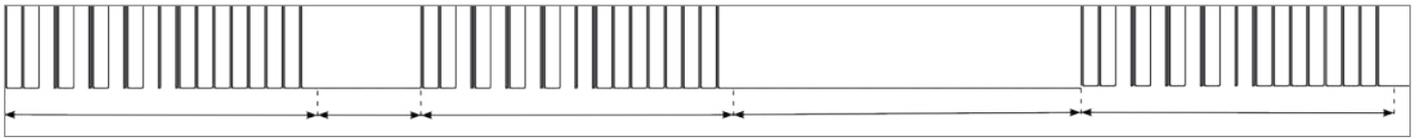
Wir haben für unsere Experimente einen *Arduino Pro Micro* auf Basis des ATmega32U4 verwendet, da dieser für die Kommunikation mit dem PC den internen USB-Anschluss nutzt und den UART unbenutzt lässt. Wir können also den UART zur Kommunikation mit der Lok

nutzen und gleichzeitig per USB mit dem PC Daten austauschen.

Aber der ATmega32u4 enthält zur seriellen Kommunikation nicht nur den Arduino-üblichen, auf reinen Asynchronbetrieb ausgelegten UART (Universal Asynchronous Receiver/Transmitter), sondern einen USART (Universal Synchronous/Asynchronous Receiver/Transmitter). Das bedeutet, dass sich der USART auch für synchrone Betriebsarten wie SPI nutzen lässt. Eine SPI-Übertragung nutzt keine expliziten



5 Durch Aussenden speziell präparierter Bytes können Motorola-Trits mit einem USART im SPI-Modus erzeugt werden.



6 Zur Erhöhung der Störsicherheit wird das gleiche Paket immer zweimal gesendet. Die Pausen zwischen den Impulspaketen sind relativ unkritisch (1...5ms).

Start- und Stoppbits, sondern zusätzliche Signale, um den Übertragungstakt oder den Beginn einer Nachricht zu markieren. Die Datenleitung einer SPI-Verbindung lässt sich daher mit nahezu beliebigen Signalen belegen.

Leider nur „nahezu“, denn zum Beispiel die SPI-Implementierung im ATmega328 des klassischen Arduino ist bei den Datenraten auf ganzzahlige Teiler des Systemtaktes festgelegt – die würden hier nicht funktionieren! Der USART des *Arduino Micro Pro* oder auch des *Leonardo* kann im SPI-Betrieb die Datenrate dagegen mit der gleichen Präzision festlegen wie im asynchronen UART-Modus, allerdings muss man sich zur Programmierung auf eine Hardware-nähere Ebene begeben, da die Arduino-SPI-Bibliothek den SPI-Betrieb des USART nicht unterstützt (siehe Code-Kasten). Das Ergebnis ist in Abbildung 6 zu sehen und entspricht in Signalform und Datenrate exakt dem, was wir benötigen; wie im Original werden die Datenpakete immer zweimal gesendet.

Eine Kleinigkeit fehlt noch: Zwischen den Motorola-Nachrichten werden Pausen eingelegt. Diese zwischen den Datenpaketen mit dem UART zu erzeugen, erfordert etwas Tricksen und Probieren, da der Arduino Teile der UART-Kommunikation im Hintergrund erledigt, und die Übertragung noch im Gange ist, wenn der Sketch bereits weiterläuft. Wir haben die tatsächlichen Pausen mit einem billigen Logic-Analyzer gemessen und manuell angepasst. Da die Pausenlänge zwischen den Paketen ansonsten unkritisch ist, sollte man nicht allzu viele Gedanken darauf verschwenden.

Mehr Dampf

Der Ausgang des Arduino lässt sich leider nicht direkt an die Schiene der Modellbahn anlegen. Nicht nur die Betriebsspannung des Arduino ist mit 5 Volt deutlich niedriger als die ± 15 bis 18V, die die Modellbahn erwartet. Es werden außerdem Ströme von mindestens einem Am-

pere benötigt. Man könnte nun eine leicht modifizierte NF-Endstufe mit 10 bis 30W Leistung benutzen, um die Rechteckspannung zu verstärken (so machte es die Märklin-„central unit“ und der „booster“), es geht aber auch einfacher, wie wir gleich sehen werden.

Eine NF-Endstufe hätte den Nachteil, dass sie eine symmetrische Betriebsspannung (also getrennte +18V und -18V-Spannungen) benötigt; die Ausgangsspannung muss ja die gleichen Werte erreichen können. Wir verwenden stattdessen ein Motor-Shield, das mit einer einzigen Betriebsspannung auskommt, dafür aber zwei gegenphasige Ausgangssignale erzeugt, die beide an Schiene und Mittelleiter angeschlossen werden. Die Lok „sieht“ dann am Schleifer eine Wechsellspannung von $\pm 18V$ bezogen auf den Außenleiter (Schiene, Räder). Kleiner Nachteil: Die Schienen-Masse ist eigentlich keine, sie führt einen Teil der Wechsellspannung; das ist möglicherweise aber erst beim Anschluss von Zubehör (z.B. Gleisbesetzmelder) relevant.

Vorsicht ist bei der Auswahl des Shields geboten. Die besonders kompakten Shields auf Basis des TB6612FNG, DRV8833 oder ähnlich vertragen maximal Spannungen von 10 bis 15 Volt; für einen flotten Zugbetrieb sind aber 18 Volt zwingend nötig. Hier bietet sich der klassische L298 an: Dieser etwas betagte Vollbrücken-Treiber benötigt eine etwas aufwändigere Beschaltung und Kühlung als seine modernen Brüder, kommt aber dafür mit den 18 bis 20V problemlos klar. Einfache Shields auf Basis des L298N bekommt man bereits für rund 5 Euro 7. Von den zwei im L298 vorhandenen Motortreibern benutzen wir nur einen.

Ein Detail fehlt für die Signalerzeugung noch, denn jede Vollbrücke im L298 wird über zwei Eingänge angesteuert, über die sie gegenphasige Signale für die Schienen- und Mittelleiteranschlüsse entgegennimmt. Am Arduino haben wir aber nur einen Ausgang am USART. Die Lösung kommt in Form eines 74HC04- oder 74HC14-Inverters, es sind auch HCT- und LS-Typen geeignet.

Es bleibt ein weiteres Problem zu lösen, da die Eingangsspannung eines üblichen Arduino nur maximal 12V betragen sollte. Bei höheren Spannungen wird der kleine interne Spannungsregler zu heiß und sorgt kurzzeitig für modellbahngerechte Dampferzeugung. Auch hier hilft ein Griff in die Bastelkiste: Der klassische 7805 in TO220-Bauform kommt mit Spannungen über 20V zurecht

Senden mit USART-SPI

Um den ATmega32U4-USART auf das benötigte Datenformat einzustellen, kann man nicht auf die bequemen Arduino-Libraries zurückgreifen, sondern muss die Register hardwarenah programmieren. Ein Blick ins Datenblatt des ATmega16U4/32U4 ist dafür unerlässlich; hier ist insbesondere die Registerbelegung ab Seite 209 wichtig.

Die gesetzten Bits UMSEL10 und UMSEL11 stellen das „USART Control and Status Re-

gister 1C“ UCSR1C auf den Master-SPI-Modus ein, das Bit TXEN1 schaltet das „USART Control and Status Register 1B“ UCSR1B auf Sendebetrieb. In der dritten Zeile setzen wir das Baudraten-Register UBRR1 auf die Bitlänge von 26 μ s. Auch das Aussenden der einzelnen Trit-Hälften durch Schreiben auf UDR1 erfolgt „zu Fuß“ – immer dann, wenn der Sender wieder aufnahmebereit ist. Dies wird durch das „Data Register Empty“-Bit UDRE1 im Status-Register UCSR1A angezeigt.

```
UCSR1C = (1<<UMSEL11)|(1<<UMSEL10); // set spi mode 0
UCSR1B = (1<<TXEN1); // enable transmitter only
UBRR1 = 26*(F_CPU/2000000)-1; // 26us bit length

// wait for transmit buffer empty and send trits
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x7f; // hi-hi-trit
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x7f;
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x7f; // hi-low-trit
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x01;
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x7f; // hi-hi-trit
while ( !(UCSR1A & _BV(UDRE1)) ); UDR1 = 0x7f;

...
```

Neue Online-Kurse für IT-Professionals



Der JavaScript-Kurs für Softwareentwickler

Der Programmierkurs für Einsteiger und Umsteiger

► 77 Videos, 8 Std. Spielzeit



Web Application Security

Das Security-Training für Entwickler und Webmaster

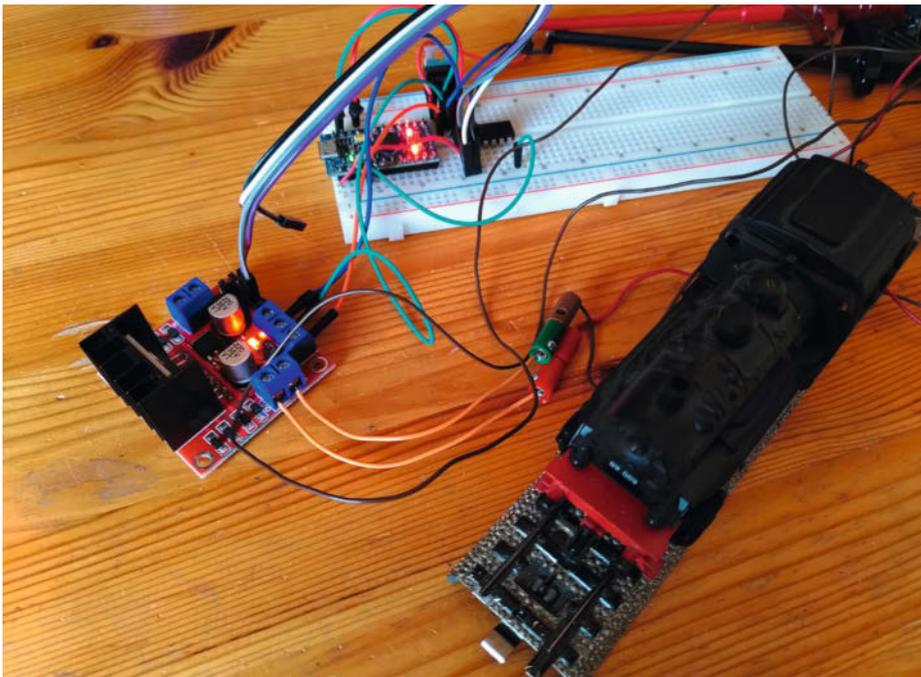
► 82 Videos, 9:30 Std. Spielzeit



C++ 20 – Die Neuerungen im Detail

Concepts, Ranges, Module und Coroutinen

► 48 Videos, 3:30 Std. Spielzeit

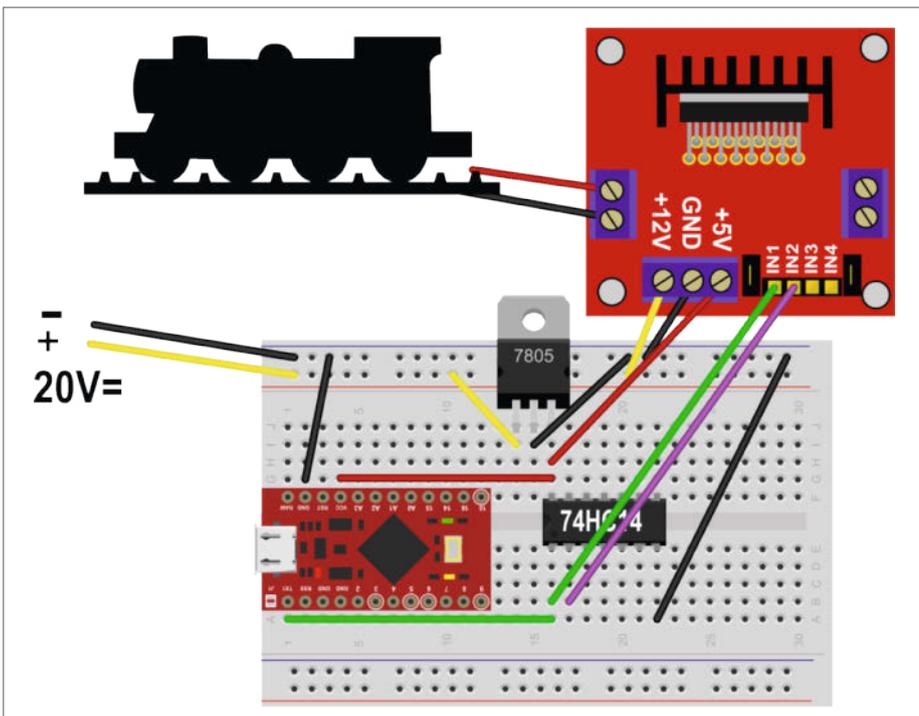


7 Versuchsaufbau mit Arduino Micro und Motor-Shield. Zur Stromversorgung diente hier ein einstellbares Labornetzteil.

und kann die überschüssige Leistung problemlos abführen; sollte er im Dauerbetrieb zu heiß werden, hilft ein kleines Kühlblech.

Die vollständige Schaltung ist in Abbildung 8 zu sehen. Mit einem Arduino-Nachbau, L298-Shield, 7805-Regler und 74HC14-Inverter erhält man für weniger als 15 Euro eine einfache

Möglichkeit, eine kleinere Modellbahn direkt vom Arduino aus zu steuern oder den Arduino als Schnittstelle zwischen PC oder einem Raspberry Pi einzusetzen. Unter dem Link in der Kurzinfo stellen wir Beispiel-Sketches für diverse Anwendungsfälle zur Verfügung. —cm



8 Aufbau auf dem Breadboard mit externem Spannungsregler, Inverter-IC für das gegenphasige Signal und L298-Treiberplatine

Make: Block – Neue Hard- und Software

Fünf Jahre ist es schon her, dass wir ein Klötzchen-Schiebespiel in einen Ikea-Rahmen einbauten. Jetzt gibt es noch einen Super-Klempner dazu. Und einen Bausatz aus Plexiglas erhalten Sie demnächst im heise shop.

von Till Harbaum



In Make 04/16 haben wir aus 300 RGB-LEDs, einem Ikea-Bilderrahmen und dem Arduino ein Spiel der besonderen Art gebaut. Obwohl wir damals das Spiel kräftig aufgebohrt hatten, waren von den 32 Kilobyte Speicher des Arduino nur knapp 50% benutzt. Wir haben uns ein paar Gedanken gemacht, wie sich die zweite Hälfte sinnvoll nutzen lässt. Das Ergebnis steht auf Github zur Verfügung (erreichbar über den Kurzinfo-Link) und kann mit der Arduino-IDE auf den Mikrocontroller im Make:Block beziehungsweise im neuen *GameFrame*-Bausatz (siehe Kasten auf der nächsten Seite) übertragen werden.

Dabei sind wir ein weiteres Mal in der Retro-Kiste fündig geworden und haben uns den Superklempner Mario näher angeschaut. Auch wenn das Original aus dem Pixel-Mittelalter der Konsolengrafik stammt, liegt die Auflösung mit $256 \times 240 = 61440$ Pixeln doch schon deutlich über den 300 Pixeln unserer Hardware. Mit ihren 300 einzeln mit 24-Bit-Farbtiefe ansteuerbaren LEDs übertrifft sie aber immerhin die Farbfähigkeiten des Originals mit seinen 16 aus 52 gleichzeitig darstellbaren Farben.

Kurzinfo

- » Neue Software für Spielerrahmen
- » Super-Klempner und Klötzchenschieben in einem Arduino Nano
- » Spielerrahmen als Bausatz *GameFrame*

Checkliste



Zeitaufwand:
8 Stunden (für Aufbau des *GameFrame*-Bausatzes)



Kosten für den Bausatz:
149,99 Euro

Werkzeug

- » Lötkolben
- » Schere
- » Schraubendreher

Mehr zum Thema

- » Ulrich Schmerold, Tetris-Klon mit LED-Streifen, Make 1/15, S. 12
- » Till Harbaum, Make:Block reloaded, Make 4/16, S. 52
- » Till Harbaum, Make:Block: Das Spiel, Make 5/16, S. 82
- » Bausatz im heise shop

Alles zum Artikel im Web unter make-magazin.de/xgmx

Minimalismus

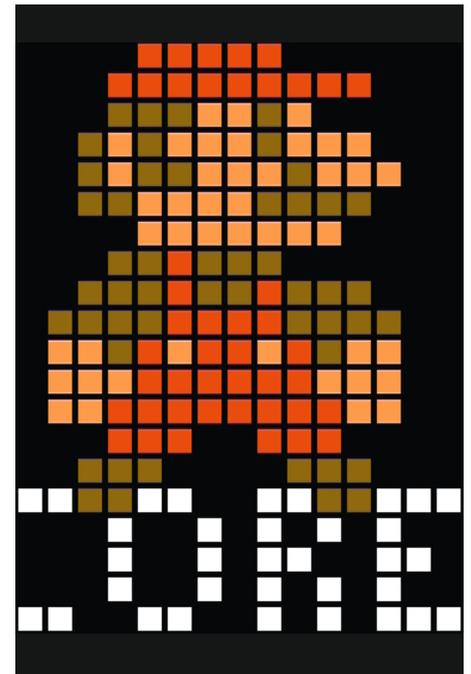
Hätten wir die Mario-Grafik in voller Pracht beibehalten, dann würde die Spielfigur allein den Bildschirm bereits vollständig füllen, wie ein Screenshot des Splash-Screens zeigt 1.

Daher mussten wir die Grafik so weit wie möglich reduzieren. Um das ganze Spielfeld auf den Ikea-Rahmen beziehungsweise den neuen Bausatz *GameFrame* (siehe Kasten auf der nächsten Seite) zu bekommen, musste jedes der aus Tiles und Sprites aufgebauten Grafikelemente des Spiels maximal vereinfacht werden: Jeweils nur noch ein Pixel durften sie groß sein und selbst aus Mario wurde nur noch ein einzelnes rotes Quadrat. Die Frage ist, ob mit soviel Minimalismus noch ausreichend Mario-Spielspaß übrig bleibt. Wir haben es ausprobiert. Wie beim Tetris wollten wir auch bei Mario möglichst viel des Originallooks erhalten. So haben wir durch Einschrumpfen des ersten Mario-Levels 2 auf 1/16 seiner Auflösung und ein wenig manueller Nachbearbeitung einen insgesamt nur 212×13 Pixel großen Level mit überraschendem Wiedererkennungswert geschaffen 3.

Datensparsamkeit

Mit der Reduzierung der Auflösung verringerte sich auch die Datenmenge. Auf diesen Umstand waren wir bei gerade mal 16 Kilobytes verfügbarem Programmspeicher dringend angewiesen, schließlich wollten wir ja auch auf unser Tetris nicht verzichten. Trotzdem wurde es sportlich. Der Level aus 212×13 Pixel zu je 24 Bit hätte über 8000 Bytes benötigt, wenn wir ihn direkt gespeichert hätten. Ohne eine einzige Zeile Spiellogik wäre dadurch schon die Hälfte des zur Verfügung stehenden Speichers verbraucht. Die Daten mussten also noch weiter reduziert beziehungsweise komprimiert werden.

Zur Lösung dieser Aufgabe schrieben wir das kleine Analysetool *png2level*. Es nahm sich das 212×13 -Bild vor und suchte darin nach Wiederholungen und mehrfach verwendeten Mustern. Findet es zum Beispiel eine horizontale Reihe aus braunen Blöcken, merkte es sich lediglich, wo diese Reihe beginnt und wie lang sie ist. Ähnliche Informationen sammelte es über alle anderen Objekte und nur diese Informationen wurden auf dem Arduino gespeichert.



1 In dieser Größe durfte der Super-Klempner nur zu Beginn des Spiels erscheinen.



2 Das Vorbild hätte in dieser Auflösung nie in den Speicher des Arduino gepasst.



3 Jetzt sind es nur noch 213×13 Pixel, mehr Reduzierung war nicht möglich.

GameFrame: Spielerahmen als Bausatz im heise shop

Demnächst ist eine Plexiglas-Version des Spielerahmens als Bausatz im heise shop erhältlich. Der Bausatz besteht aus dem Plexiglas-Rahmen, dem Gehäuse für den Joystick sowie sämtlichen elektronischen Bauteilen wie Arduino nano, Netzteil, Audioverstärker, Lautstärke-Poti und Lautsprecher sowie den benötigten Kabeln und Schrauben. Auch die LED-Streifen liegen bei.

Die Gehäuseteile für den Rahmen und den Joystick sind aus elegantem, schwarzglänzendem Plexiglas per Laser-cut gefertigt. Für die saubere Trennung der leuchtenden Pixel sorgt ein zusammensteckbares Gitter aus stabiler Pappe.

Zusammen mit der Transparentpapier-Mattscheibe sorgt es für ein scharfes Bild.

Auch zusammengebaut kann jederzeit Software auf den im Joystick-Gehäuse untergebrachten Arduino nano übertragen werden, da dessen USB-Buchse von außen zugänglich ist. Die umfangreich bebilderte Bauanleitung zeigt Schritt für Schritt den Zusammenbau. Sie steht zusammen mit der Anleitung zum Überspielen der Software online auf Github (siehe Kurzinfor-Link) zur Verfügung.

Der Bausatz ist demnächst zum Preis von 149,99 Euro im heise shop erhältlich.



So sieht der aufgebaute *GameFrame* aus dem neuen Bausatz aus.

passt hier der maximale-Punkte-in-kürzester-Zeit-Ansatz sehr gut. Ziel ist es, möglichst viele Münzen zu sammeln und *Goombas* in möglichst kurzer Zeit zu plätten.

Apropos Party: Musik bietet unser Spiel natürlich auch. Wie bereits beim Tetris lässt es sich aber zur Ohren- und Nervenschonung im *Setup* abschalten. Zur Erinnerung: Das *Setup* erreicht man, wenn beim Einschalten des Gerätes die Feuertaste gedrückt ist. Dort lässt sich dann die Lautstärke und die Bildhelligkeit ändern.

Die Wahl zwischen *Tetris* und *Mario* erfolgt erst am Start-Bildschirm. Hier kann man durch die Bewegung des Joysticks nach *links* oder *rechts* zwischen beiden Spielen wählen.

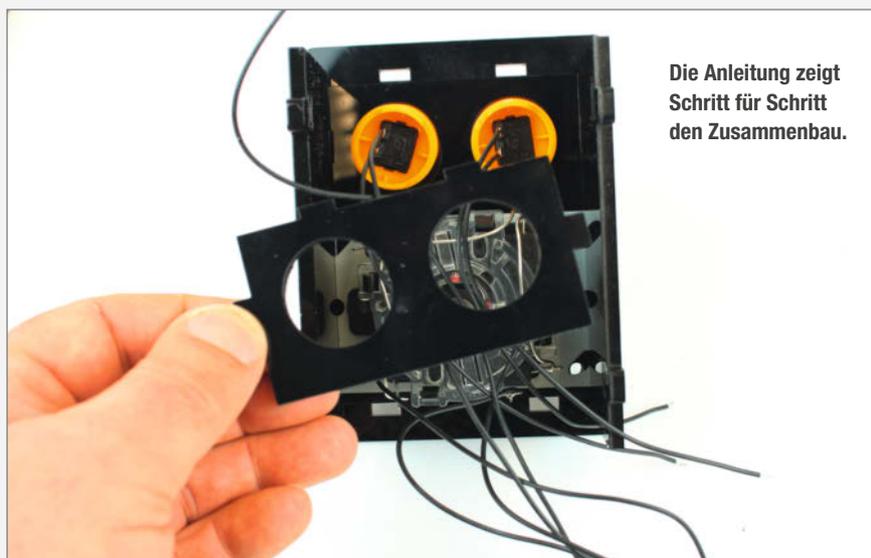
Die HiScores werden für jedes Spiel separat gespeichert, so dass ein gegebenenfalls bereits vorhandener Tetris-HiScore durch das Firmware-Update nicht überschrieben wird.

Stromhunger

Da Mario sehr viel mehr Pixel einschaltet als Tetris, ist der Stromverbrauch deutlich höher. Hier hilft wieder unsere Linux-Simulation aus (siehe *Mehr zum Thema* in der Kurzinfor). Sie sagt für Mario bei maximaler Helligkeit einen Strom von maximal 8,8 Ampere voraus. Betreibt man den Make:Block also wie wir an einem 4A-Netzteil, so sollte man die Helligkeit im *Setup*-Menü höchstens auf etwas unter 50 Prozent einstellen. Im *GameFrame*-Bausatz ist ein entsprechend stärkeres Netzteil enthalten.

Trotz des zweiten Spiels haben wir übrigens immer noch mehr als 7 Kilobytes beziehungsweise 25 Prozent Flash-Speicher frei. Ein paar Optimierungen zum Beispiel bei der Speicherung des Titel-Bildes könnte uns vielleicht nochmal ein gutes Kilobyte verschaffen. Einem dritten Spiel steht daher prinzipiell nichts im Wege. Wer weiß, was noch kommt ...

—hgb



Die Anleitung zeigt Schritt für Schritt den Zusammenbau.

Das Resultat: wenige hundert Bytes abstrakter Informationen, die in der Datei *mario_lvl.ino* liegen. Erst zur Laufzeit baut der Arduino daraus wieder die aktuelle Sicht des Levels zusammen. Bei der Analyse des Level-Bildes am PC wurden auch Eigenschaften der einzelnen Grafikteile bestimmt und gespeichert. So weiß der Arduino nun auch, welche Bestandteile des Levels reine Deko sind (etwa Wolken oder Grünzeug) oder feste Substanz haben und unseren Mario am Fallen beziehungsweise Weiterlaufen hindern (wie Steine, Treppen, Röhren, ...).

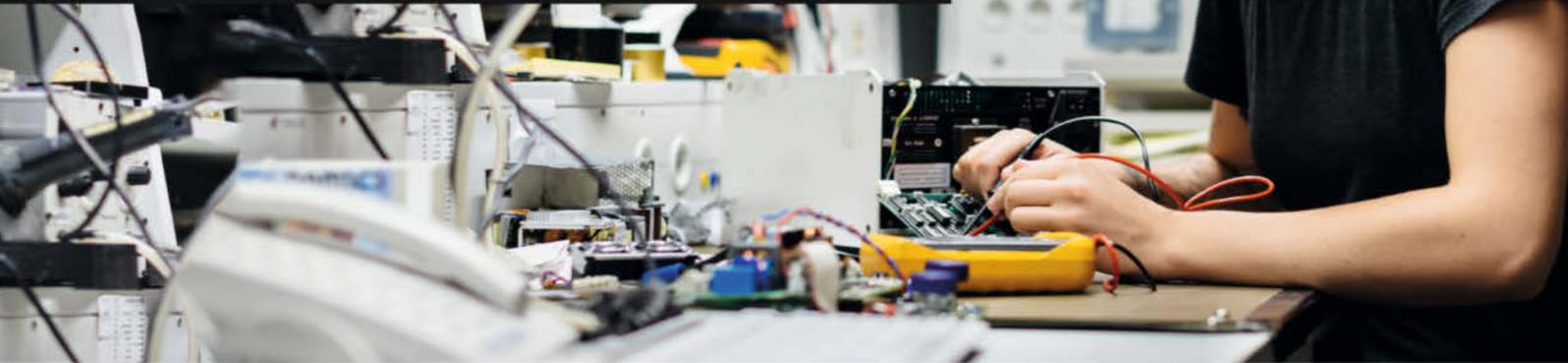
Nur über die Farbgebung einzelner Pixel zwischen den diversen Spielelementen zu unterscheiden, ist nicht ganz einfach. Zur Erleichterung flackern daher zum Beispiel alle Objekte zusätzlich, die Mario sammeln sollte

(wie Münzen oder Pilze). Der RAM-Speicher von nur 2kByte im Arduino stellt übrigens keine zusätzliche Beschränkung dar. Seine Nutzung hat sich sogar gegenüber der reinen Tetris-Variante verringert, da wir die meisten Speicherbereiche nun mehrfach nutzen und der gleiche Speicher vom Titelbild und den beiden Spielen genutzt wird. Es läuft schließlich immer nur eines davon zur Zeit.

Der größte Unterschied zum Originalspiel ergibt sich aber aus der Tatsache, dass wir nur einen Spiel-Level implementiert haben. Während selbst die besten Spieler über 20 Minuten brauchen, um einmal durch das gesamte Original-Spiel zu hetzen, ist man bei uns schon nach wenigen Sekunden am Ziel. Da unser Spielerahmen sowieso eher im Partykeller hängt und auf kurzen Spielspaß ausgelegt ist,

ARROW

We're everywhere
you need us



Besucht Arrow auf der Virtuellen Maker Faire!

Highlights sind:

- Arcade Gaming Station von Infineon
- MAXCO2 Eval-Kit mit Intel®
- FPGA und LT Spice Simulation von Analog Devices



Besucht auch
unsere Webinare:
QR-Code scannen
und fortbilden.

In Partnership with:



Five Years Out

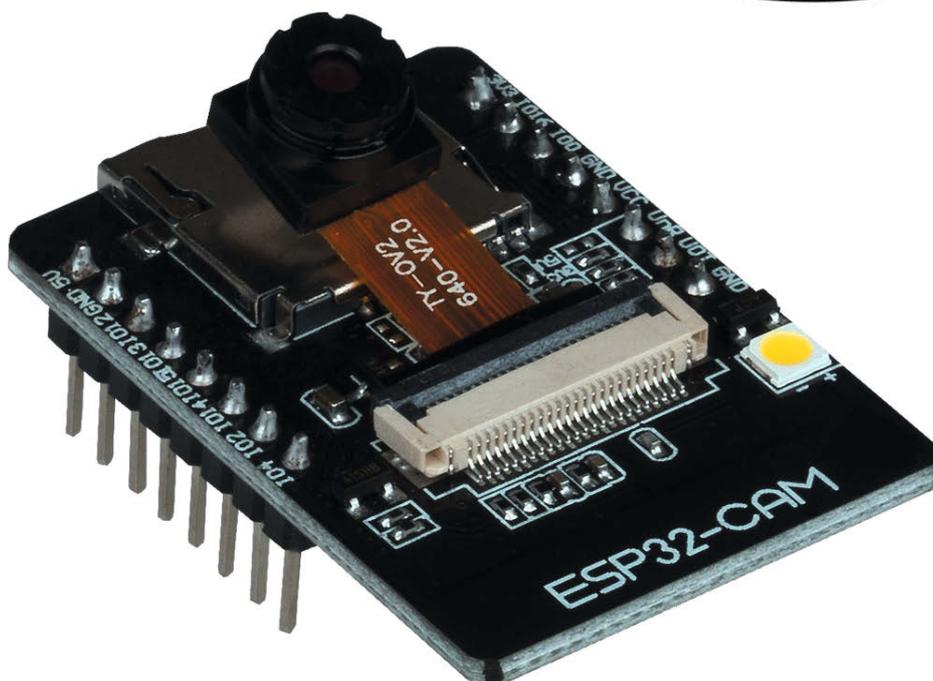
© Copyright by Maker Media GmbH.

arrow.com

Gesichtserkennung steuert Alexa

In diesem Projekt nutzen wir die beliebte ESP32CAM, um Alexa-Routinen zu starten, und zwar auf Basis von erkannten Gesichtern. So wird man mit der Lieblingsmusik begrüßt, wenn man einen Raum betritt.

von Andreas Koritnik



Die ESP32CAM verbindet einen leistungsfähigen ESP32-Mikroprozessor von Espressif mit einem OV2640-Kameramodul (2 Megapixel Auflösung). Die ESP32CAM ist sehr günstig aus China zu beziehen (unter 5 Euro). Aber selbst direkt aus Deutschland geliefert ist sie für ca. 10 Euro zu bekommen. Der Hersteller Espressif hat zusätzlich zum Betriebssystem (FreeRTOS) auch ein Gesichtserkennungsmodul (ESP-FACE) entwickelt, das ein neuronales Netz zur Gesichtserkennung bereitstellt. Das neuronale Netz ist dabei speziell auf die begrenzten Ressourcen von Mikroprozessoren ausgerichtet. Die ausführliche Beschreibung des Erkennungsprozesses erfolgt später im Artikel.

Unser Projekt „Alexa-Gesichtserkennung“ basiert auf dem GitHub-Repository des Nutzers *Robotzero1*, der ein Türöffnungsprogramm mit der ESP32CAM realisiert hatte.

Leider war der Code nicht besonders gut strukturiert und dokumentiert und auch instabil (siehe Kasten *Speicherlecks* am Ende des Artikels).

Ich habe daher einige Änderungen sowie Erweiterungen am Code vorgenommen:

- Zusätzliche Kommentare und Vereinfachungen im Code
- Verwendung von lesbarem HTML/JavaScript-Code in *camera_index.h* (erleichtert das Ändern von Inhalten)
- Der JavaScript-Code wurde geändert, damit er auch mit Apples Safari-Browser funktioniert.
- Gesichtserkennung mit und ohne Client, der über Web-Sockets verbunden ist
- Stammzertifikat und Code hinzugefügt, um URLs für jedes erkannte Gesicht aufzurufen
- Verwenden der eingebauten LED, um anzuzeigen, ob ein Gesicht erkannt wird, und um zusätzliches Licht für eine bessere Erkennung bereitzustellen
- Entfernen von Speicherlecks durch Freigeben genutzter Speicherbereiche

Das Programm wurde ausgiebig getestet und läuft stabil. Es eignet sich durch die detaillierte Kommentierung auch gut als Basis für eigene Projekte. Das fertige Programm ist auf GitHub verfügbar, das Sie über den Link in der Kurzinformatik erreichen.

Vorbereitungen

Die Lösung verwendet den URL-Routine-Trigger-Dienst von www.virtualsmarthome.xyz. Das ist eine einfache Möglichkeit, um Alexa-Routinen zu starten und erinnert an Dienste wie IFTTT.

Um den Dienst zu nutzen, muss man sich mit seinem Amazon-Konto anmelden und außerdem den Virtualsmarthome-Skill in Alexa aktivieren. Dazu muss man Virtual Smart Home den Zugriff auf das Amazon-Profil erlauben **1**.

Kurzinformatik

- » Gesichtserkennung mit ESP32CAM einrichten
- » Online-Dienst *Virtual Smart Home* einrichten
- » Alexa-Skills verknüpfen

Checkliste



Zeitaufwand:

1 Stunde



Kosten:

20 Euro



Programmieren:

Arduino-IDE

Material

- » ESP32CAM Modell AI Thinker
- » USB-zu-Seriell-Wandler plus Jumperkabel

Mehr zum Thema

- » Daniel Bachfeld, Gesichtsteuerung, Make 2/20, S. 16
- » Daniel Bachfeld, Intelligente Webcam für 5 Euro, Make 1/20, S. 28
- » Josef Müller, ESP32CAM liest Wasseruhr, Make 2/21, S. 14

Alles zum Artikel im Web unter make-magazin.de/x8vz

Für jede zu erkennende Person muss eine *Trigger Name* und eine URL definiert werden. Die URL wird automatisch mit zufälligen Werten erzeugt **2**.

Die langen Buchstaben-Zahlenkolonnen dienen der Sicherheit und sollen ausschließen, dass jemand durch Ausprobieren die URL zum Auslösen der Alexa-Routine errät.

Die verschiedenen URLs werden dann vom ESP32 über HTTPS gesendet, nachdem ein definiertes Gesicht erkannt wurde. Eine in Alexa bereits vorhandene Funktion einer virtuellen SmartHome-Türklingel kann mit dem Namen des definierten Triggers verwendet werden, um Routinen für jedes erkannte Gesicht zu starten.

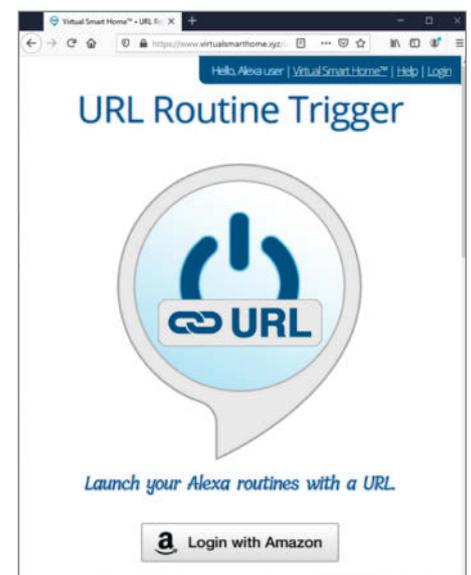
Sketch anpassen

Laden Sie aus dem Github-Repository die Dateien *AlexaFaceDetectionESP32Cam.ino*, *camera_index.h*, *camera_pins.h* und *partitions.csv*, am besten per ZIP-Download, und entpacken Sie alle an einen Ort Ihrer Wahl. Öffnen Sie mit der aktuellen Version 1.8.13 der Arduino-IDE die Datei *AlexaFaceDetectionESP32Cam.ino*. Es sind nur wenige Anpassungen erforderlich: Die Zugangsdaten für das WLAN müssen eingetragen werden und die auf Virtual Smart Home erzeugten URLs müssen definiert werden. Hierzu kopieren Sie einfach die individuellen URLs von der Virtualsmarthome-Website in die vorbereiteten Felder **3**.

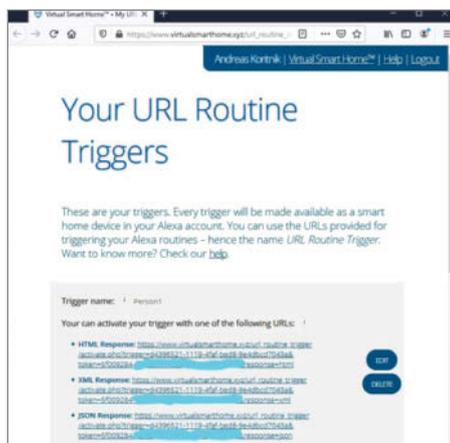
Die Reihenfolge der URLs in der Liste entspricht der Reihenfolge der gespeicherten Gesichter. Mit der aktuellen Konfiguration können bis zu sieben URLs gespeichert werden. Tipp: Man kann mehr als eine Gesichtserkennung pro Person speichern. Das verbessert die Erkennungsgenauigkeit noch weiter.

Zum Übersetzen des Sketches benötigen Sie die Unterstützung für ESP32-Boards ab der Version 1.0.5, mit Version 1.0.4 gibt es Fehlermeldungen. Sofern die Unterstützung bereits installiert ist, müssen Sie gegebenenfalls unter der Board-Verwaltung nur die Version updaten. Ist die Unterstützung noch gar nicht installiert, finden Sie unter dem Link in der Kurzinformatik eine Anleitung.

Zudem muss die Bibliothek *ArduinoWebsocket* ab der Version 0.5.0 installiert sein, die



1 Der Dienst Virtual Smart Home dient als Vermittler zwischen der ESP32CAM und Amazons Alexa-Skills.



2 Die Virtual-Smart-Home-Web-Seite bietet drei unterschiedliche URLs an. Die JSON-Version (dritte URL) ist die bevorzugte Option (kurze Antwort). Es funktionieren aber alle drei Varianten.

sich leicht über den Bibliotheksverwalter nachinstallieren lässt. Zum Übersetzen stellen Sie unter *Werkzeuge/Board* auf *ESP32 Wrover Modul* und beim *Partition Scheme* auf *Huge APP...*

Die Datei *partitions.csv* definiert die Partitionsinformationen 4 für den Flash-Speicher und ermöglicht das Abspeichern von Bildern für die Gesichtserkennung.

Die *fr*-Partition ist für die Speicherung der Gesichts-IDs relevant (knapp 1 Megabyte reservierter Speicher). Mit der aktuellen Arduino-IDE 1.8.13 und ESP32-Board-Version ab 1.0.5 ist es ausreichend, die Partitionsinformationen

einfach im Programmordner des Sketches abzulegen. Das umständliche Ändern der CSV-Dateien im User-App-Ordner ist nicht mehr notwendig.

Wenn das Programm jedoch kopiert/verschoben wird, muss sichergestellt werden, dass sich alle vier Dateien im neuen Ordner befinden: *AlexaFaceDetectionESP32Cam.ino*, *camera_index.h*, *camera_pins.h* und *partitions.csv*.

Die Datei *camera_index.h* enthält die HTML-/Javascript-Webseite und *camera_pins.h* die GPIO-Definitionen für das verwendete Kameramodell. In unserem Fall ist das Modell *CAMERA_MODEL_AI_THINKER* für die ESP32-CAM, das so auch am Anfang des Sketches per *#define* festgelegt ist.

Flashen

Zur Programmierung des ESP32CAM-Moduls wird ein externer USB-zu-Seriell-Adapter (FTDI) benötigt. Zum Programmieren müssen die Pins IO0 an GND, RX vom Adapter an U0T der Kamera sowie TX vom Adapter an U0R angeschlossen werden 5. Zusätzlich schließt man die Stromversorgung des Adapters an die Kamera.

Durch Drücken der *Boot*-Taste an der ESP32CAM wird der Programmiermodus gestartet. Danach kann der Upload-Vorgang in der Arduino-IDE gestartet werden (mit dem korrekt ausgewähltem USB-Serial-Port vom Adapter). Für den normalen Start der ESP32CAM muss man IO0 von GND trennen und erneut die *Boot*-Taste drücken.

Anlernen

Nach dem Programmieren muss die Konfigurationswebseite im Browser mit der im seriellen Monitor der IDE angezeigten IP-Adresse gestartet werden 6. Wenn das Programm richtig funktioniert, können nun die Personen mit Namen hinzugefügt werden. Dazu trägt man einfach den gewünschten Namen in Feld ein (maximal 15 Zeichen) und klickt auf *ADD USER*, um einen Nutzer hinzuzufügen. Für jede Person werden 5 Erkennungsdurchläufe durchgeführt. Es sollte auf ausreichende Beleuchtung geachtet werden, damit die Gesichter später auch erkannt werden. Mit ein bisschen Übung gelingen für die Erkennung geeignete Datensätze.

Wenn der Prozess erfolgreich abgeschlossen wurde, erscheint der neue Name unter *Captured Faces*. Die Gesichtsinformationen werden dauerhaft im Flash-Speicher gespeichert. Es können bis zu sieben verschiedene Gesichter gespeichert werden. Es kann pro Person mehr als ein Gesicht gespeichert werden, zum Beispiel in leicht geänderten Positionen und Winkeln. Das verbessert die Erkennungsrate noch weiter.

Die Reihenfolge der hinzugefügten Benutzer ist für die zu sendende URL relevant, nicht der Name. Die Webseite ist nur zum Verwalten der Gesichter erforderlich (*Hinzufügen / Löschen*). Der Erkennungsprozess ist auch dann aktiv, wenn kein Webclient verbunden ist. Die eingebaute LED wird drei Sekunden lang aktiviert, sobald ein Gesicht erkannt wird (Face Detection). Dabei spielt es keine Rolle, ob dieses Gesicht zu einem bekannten Nutzer gehört. Ein kurzes Blinken mit der LED zeigt einen erfolgreichen Zuordnungsprozess an (Face Recognition), also ob das Gesicht zu einem zuvor aufgenommenen Nutzer passt. Allgemein erkennt der Algorithmus Gesichter ohne Brille und Bart besser. Die anderen Buttons in der Weboberfläche haben folgende Funktionen:

- STREAM CAMERA: Nur das Bild wird angezeigt. Die Gesichtserkennung ist ausgeschaltet.
- DETECT FACES: Generelle Gesichtserkennung ist eingeschaltet.
- DETECT PERSON: Die Erkennung von Personen ist eingeschaltet.
- DELETE ALL: Alle gespeicherten Gesichter werden gelöscht.

Einzelne Gesichter können durch Anklicken des rot/weißen Kreuzes hinter dem Namen gelöscht werden.

Alexa

Nachdem der Virtualsmarthome-Skill in Alexa aktiviert wurde, können nun in der Alexa-App neue Routinen erstellt werden 7. Als Auslöser kann beispielsweise eine virtuelle „Türklingel“ mit dem Namen ausgewählt werden, der bei Virtual Smart Home angegeben wurde.



3 Im Sketch sind nur wenige Anpassungen erforderlich.

4 Partitionsschema				
Name	Art	SubType	Offset	Größe
nvs	Daten	nvs	0x9000	0x5000
otadata	Daten	ota	0xe000	0x2000
app0	App	ota_0	0x10000	0x300000
fr	32	32	0x310000	0xF0000

Genießen Sie jetzt die persönlichen Antworten oder Reaktionen von Alexa, nachdem Ihr Gesicht erkannt wurde. Sie können beispielsweise Ihre Lieblingsmusik nach der Gesichtserkennung abspielen lassen.

Hintergründe

Der URL-Triggerdienst erfordert zwingend eine Verbindung mit HTTPS. Die Nutzung von verschlüsseltem HTTPS gegenüber Klartext-HTTP ist auch aus Sicherheitsgründen notwendig, um zu verhindern, dass ein Dritter Kenntnis der URLs erhält, etwa durch Mitlesen der Kommunikation im Netz. Im Sketch wird hierfür die interne Bibliothek *WiFiClientSecure* verwendet. HTTPS verwendet Zertifikate, um die Echtheit einer Gegenstelle, hier des Virtualsmarthome-Webservers, zu verifizieren. Somit wird ausgeschlossen, dass ein Dritter sich in die Kommunikation einschleust und vorgibt, der Server zu sein (Man-in-the-Middle-Angriff).

Im Browser auf dem PC gibt es dafür extra einen Speicher, der solche Zertifikate und Stammzertifikate von Herausgebern solcher Zertifikate vorhält. In Embedded-Systemen gibt es solche Speicher in der Regel nicht. Stattdessen speichern wir das Zertifikat der Stammzertifizierungsstelle als Text direkt im Code. Das Zertifikat läuft jedoch im September 2021 aus. Es muss dann aktualisiert werden.

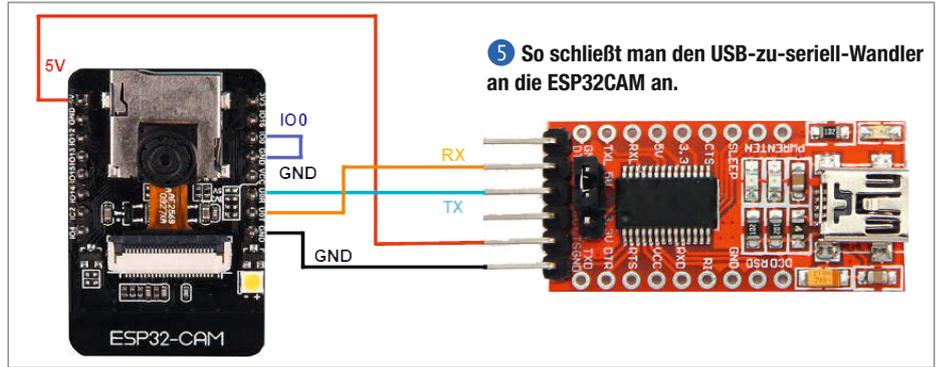
Je nach Browser unterscheidet sich der Weg etwas, um an das Zertifikat zu kommen. Unter Firefox können Sie durch Anklicken des Schlosssymbols und den Punkten *Verbindung sicher/Weitere Informationen/Zertifikat anzeigen* und dem Reiter *DST Root CA* unter dem Punkt *Verschiedenes/PEM (Zertifikat)* jederzeit das aktuelle herunterladen und in den Sketch als Text einbinden.

Ablauf der Gesichtserkennung

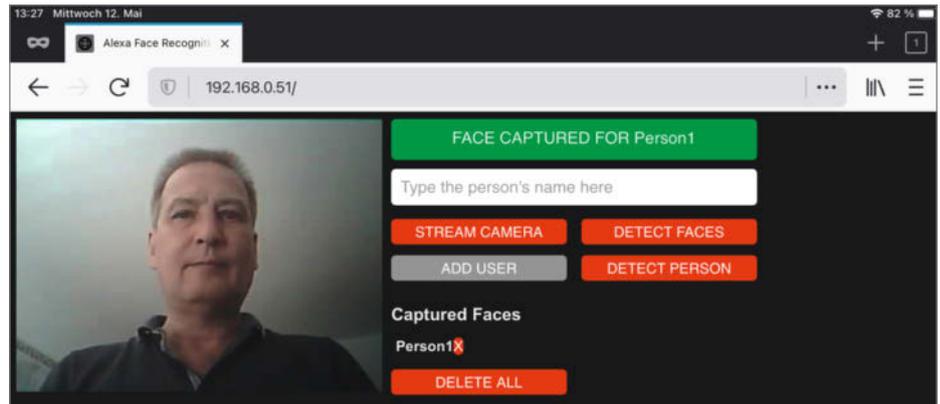
Hintergrundinformationen zur ESP-Face-Komponente von Espressif sind auf GitHub zu finden. Der Link ist im Bereich Kurzinformatik aufgeführt.

Im Folgenden schauen wir uns an, wie das Flussdiagramm im Programm implementiert ist. Für die Gesichtserkennung benötigen wir drei *Include*-Dateien.

Um den Erkennungsprozess zu implementieren, müssen wir einige globale Variablen/Objekte definieren: Der Bildpufferzeiger **fb* enthält später den Zeiger auf das Bild von der ESP-Kamera. Das Format des Bildes ist JPEG (komprimiert). Der Zeiger **detected_face* zeigt auf eine Struktur, die Informationen für ein erkanntes Gesicht enthält. Der Zeiger **image_matrix* zeigt auf eine Struktur, die eine Bitmap des Bildes enthält. Wir brauchen die Bitmap, um Gesichter zu erkennen. Der Zeiger **aligned_face* zeigt auf eine Struktur, die ausgerichtete (= aligned) Informationen für



5 So schließt man den USB-zu-seriell-Wandler an die ESP32CAM an.



6 Die Web-Oberfläche benötigt man nur zum Aufnehmen der Gesichter.

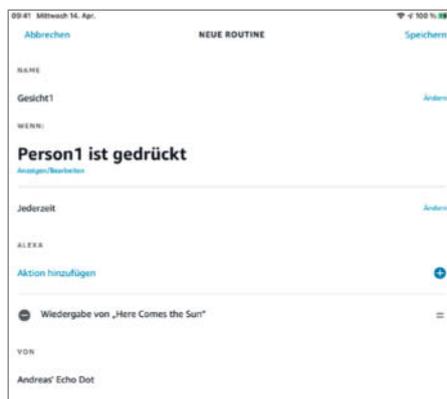
das erkannte Gesicht enthält. Die Ausrichtungs-Informationen sind für den Gesichtserkennungsprozess erforderlich.

Die **face_id* enthält das Ergebnis des Gesichtserkennungsprozesses. Wir werden später die erkannte *face_id* mit den gespeicherten *face_ids* vergleichen, um eine bestimmte Person zu erkennen. Die Struktur *mtmn_config* enthält die Konfigurationsparameter für den Gesichtserkennungsprozess.

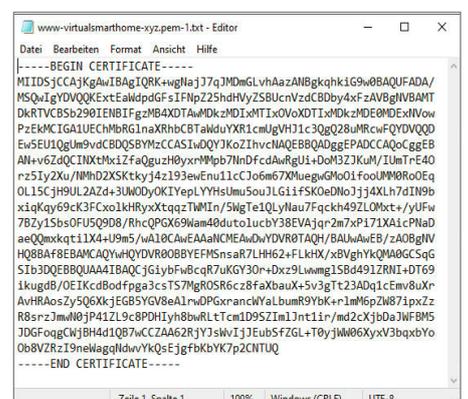
Mit der Funktion *camera_init()* im Sketch wird die Kamera konfiguriert. Wichtig ist die Angabe der korrekten Bildgröße. Sie muss 1/4 VGA, also 320 x 240 Pixel betragen. Sie wird mit der Zeile *config.frame_size = FRAMESIZE_QVGA* konfiguriert. Als nächsten Schritt muss

man einige Vorbereitungsarbeiten in *setup()* durchführen.

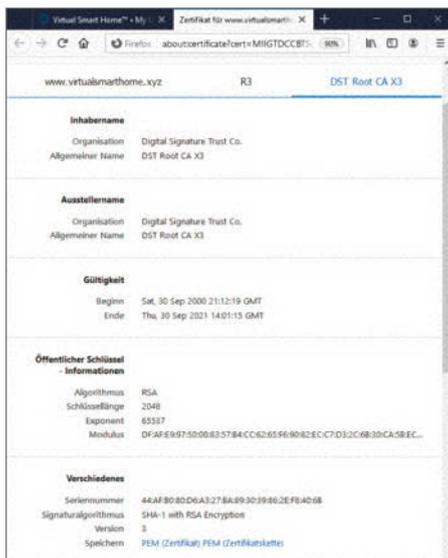
Mit *mtmn_init_config()* setzen wir die Parameter für die Gesichtserkennung auf die in *fd_forward.h* definierten Standardwerte. Dann lesen wir die Gesichter (Namen und Face-IDs) aus dem Flash-Speicher. Dies ist erforderlich, um die *face_id* später zu vergleichen. Nun müssen wir Speicher für die Struktur *image_matrix* reservieren. Diese enthält die Bitmap für die Gesichtserkennung. Die Größe der Bitmap beträgt 1/4 VGA (320 x 240 Pixel). Und das Gleiche machen wir auch für die Struktur *aligned_face*. Ein *aligned_face* hat das Format 56 x 56 Pixel (wie in *fr_forward.h* definiert).



7 In der Alexa-App legt man neue Routinen für Aktionen an.



8 Das Zertifikat als Textdatei, wie es auch im Sketch eingebunden ist



9 Das sogenannte Root-Zertifikat von Digital Signature Trust. Mit ihm lässt sich die Echtheit weiterer Zertifikate prüfen.

Loop()

In der Hauptschleife `loop()` werden wir nun die Gesichtserkennung durchführen. Das erfolgt in zwei Schritten: Im ersten Schritt wird ein generelles Gesicht erkannt (Face Detection). Und im zweiten Schritt wird versucht, eine Face-ID zu berechnen, die zur Wiedererkennung eines Gesichts genutzt werden kann (Face Recognition). Beginnen wir mit der Erkennung eines generellen Gesichts.

Die drei Zeilen in 13 dienen dazu, ein Gesicht zu erkennen. Mit `fb = esp_camera_fb_get()` erhalten wir das Bild von der Kamera. Das Format ist 320×240 Pixel, im JPEG-Format. Für den Gesichtserkennungsprozess benötigen wir das Bild aber als einfache Bitmap. Die Konvertierung erfolgt mit `fmt2rgb888(fb->buf, fb->len, fb->format, image_`

10 Include-Dateien

```
#include <fd_forward.h> // Header file for face detection
#include <fr_forward.h> // Header file for face recognition
#include <fr_flash.h> // Header file for flash storage handling
```

11 Globale Variablen/Objekte

```
// Face detection/recognition variables
camera_fb_t *fb; // Frame buffer pointer for picture from camera
box_array_t *detected_face; // Information for a detected face
dl_matrix3du_t *image_matrix; // Image matrix pointer
dl_matrix3du_t *aligned_face; // Aligned face pointer
dl_matrix3d_t *face_id; // Face ID pointer
face_id_node *face_recognized; // Recognized face pointer
mtmn_config_t mtmn_config; // MTMN detection settings
```

12 Setup

```
// Set MTMN face detection details (default values)
mtmn_config = mtmn_init_config();

// Read faces from flash
read_faces();

// Allocate memory for image matrix
image_matrix = dl_matrix3du_alloc(1, 320, 240, 3);

// Allocate memory for aligned face
aligned_face = dl_matrix3du_alloc(1, FACE_WIDTH, FACE_HEIGHT, 3);
```

13 Face Detection

```
// Get frame buffer pointer from camera (JPEG picture)
fb = esp_camera_fb_get();

// JPEG to bitmap conversion
fmt2rgb888(fb->buf, fb->len, fb->format, image_matrix->item);

// Detect face
detected_face = face_detect(image_matrix, &mtmn_config);
```

14 Face Recognition

```
if (detected_face) { // A general face has been recognised (no name so far)

    // Switch LED on to give more light for recognition
    digitalWrite(LED_BUILTIN, HIGH); // LED on
    led_on_millis = millis(); // Set on time

    if (align_face(detected_face, image_matrix, aligned_face) == ESP_OK) {

        face_id = get_face_id(aligned_face); // Get face id

        if (st_face_list.count > 0) { // if we have faces registered

            face_recognized = recognize_face_with_name(&st_face_list, face_id);

            if (face_recognized) { // Face identified
                face_detected(face_recognized->id_name); // Request URL
            }
        }
        dl_matrix3d_free(face_id); // Free allocated memory
    }
}
```

matrix-> item). Der Pointer image_matrix enthält dann das Bitmap-Bild. Und jetzt versuchen wir, ein Gesicht mit detected_face = face_detect (image_matrix, &mtmn_config) zu erkennen. Die Routine erhält die Bitmap und die Konfigurationsparameter als Eingabe. Die detected_face-Struktur enthält das Ergebnis des Erkennungsprozesses. Wenn ein Gesicht erkannt wurde, ist der Wert von detected_face wahr.

Als nächstes versuchen wir, ein spezifisches Gesicht zu erkennen ¹⁴. Das sind nur noch vier zusätzliche Schritte. Zuerst ermitteln wir die Gesichtsausrichtung mit align_face(detected_face, image_matrix, aligned_face). Die Routine erhält detected_face und die Bitmap in image_matrix als Eingabe. Das Ergebnis wird in der Struktur aligned_face gespeichert. Face Alignment bezeichnet in der Gesichtserkennung die Normalisierung von Bildaufnahmen auf eine einheitliche Größe, Neigung und Position innerhalb eines Rahmens.

Die Face-ID erhält man über die Funktion face_id = get_face_id(aligned_face). Die Face-ID enthält dann die charakteristischen Informationen für ein normalisiertes Gesicht. Wir vergleichen dann die Face-ID

Speicherlecks

Es ist wichtig, die zugewiesenen Speicherblöcke, die für den Erkennungsprozess verwendet werden, mit dl_matrix3d_free(), dl_lib_free() und esp_camera_fb_return() freizugeben. Wenn wir den zugewiesenen Speicher nicht freigeben, würde der verfügbare interne Speicher des ESP32 (Heap genannt) mit der Zeit immer kleiner werden.

Am Ende würde der ESP32 abstürzen und neu starten. Speicherlecks sind häufig der Grund für instabilen Code. Mit dem Aufruf der Unterseite uptime auf dem Webserver der Kamera (<http://192.168.x.y/uptime>)

kann die letzte Startzeit des ESP32 sowie die verbleibende freie Heap-Größe angezeigt werden:

```
Last start: 09.04.2021 13:37:31
Friday
Free Heap: 185108
```

Die Größe des Heaps kann dabei ruhig leicht schwanken. Wichtig ist nur, dass sie nicht kontinuierlich sinkt. Das würde auf ein Speicherleck hindeuten. Die Funktion dl_lib_free() ist allerdings erst ab der Board-Unterstützung 1.0.5 enthalten.

mit gespeicherten IDs: face_recognized = recognize_face_with_name(&st_face_list, face_id). Wenn ein Gesicht erkannt wurde, ist der Wert von face_recognized wahr. Als letzten Schritt erhalten wir den Namen für

das erkannte Gesicht: face_recognized->id_name. Das ist alles, um ein gespeichertes Gesicht zu erkennen. Der ganze Vorgang wiederholt sich kontinuierlich und benötigt weniger als 1 Sekunde. —dab

Noch mehr Stoff für Maker

NEU



Francesco Volpe
Leiterplattendesign mit EAGLE

Mit Autodesk EAGLE und diesem Buch lernen Sie Schritt für Schritt, wie Sie mit EAGLE von der Schaltpläneingabe über das Platinenlayout zu einer funktionierenden Leiterplatte gelangen.

ISBN 9783864905452
shop.heise.de/buch-eagle **36,90 €**



Ralf Jesse
STM32 – Das umfassende Praxisbuch

Hier erhalten Sie einen umfassenden Praxiseinstieg zur Softwareentwicklung für Embedded Systems mit der ARM-Mikrocontrollerfamilie STM32F4xx der Firma STMicroelectronics (STM).

ISBN 9783747501917
shop.heise.de/buch-stm32 **29,99 €**

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/maker-buecher



BESSER STUDIEREN

CAMPUS WILHELMSHAVEN



Ingenieurwissenschaften

JADE-HS.DE/INGENIEURWISSENSCHAFTEN

Unsere Studiengänge / Abschlüsse

- Elektrotechnik* B.Eng./M.Eng.
- **NEU** Ingenieurinformatik M.Sc.
- Maschinenbau* B.Eng./M.Eng.
- Mechatronik* B.Eng.
- Medizintechnik* B.Eng.
- Meerestechnik B.Eng.
- **NEU** Technisches Projektmanagement B.Eng.
- *auch dual

JADE HOCHSCHULE
Wilhelmshaven Oldenburg Eisfleth

SMARTBENCH

Yeti Tool SmartBench

Die große, aber trotzdem zerlegbare und in weniger als fünf Minuten aufzubauende Fräse des englischen Herstellers Yeti Tool wendet sich vornehmlich an Tischlereien, Messe- und Küchenbauer, sie dürfte aber auch in größeren Fablabs gut aufgehoben sein.

von Carsten Meyer

Das in Deutschland von Sauter (www.sautershop.de) vertriebene Gerät ist spezialisiert auf die 2,5D-Bearbeitung von Plattenware wie etwa Küchenarbeitsplatten oder auch ganzer Türen – der Arbeitsbereich umfasst dank der raffinierten Auflage-Vorrichtung 2500mm x 1250mm. Tatsächlich lässt sich die Fräse im Handumdrehen zusammen- und auseinanderbauen und kann platzsparend im Lieferwagen transportiert werden; für den Aufbau sind allerdings zwei Personen ratsam.

Portalfräse mal anders

Die CNC-Fräse besteht aus einem stabilen Aluminium-Untergestell in der Größe eines Tapeziertisches und einer X- Traverse, die in die Linearführungen des Tisches eingehängt wird. Der Antrieb erfolgt durch zwei große Nema-23-Schrittmotoren pro Achse über Zahnstangen; als Wiederholgenauigkeit gibt der Hersteller 0,125 mm an. Die höhenverstellbare X- Traverse gleitet mit Rollen über das Werkstück und unterstützt dieses auch von unten, sodass es seitlich überhängen darf und gleichzeitig auf den Arbeitstisch gedrückt wird.

Während der Y-Verfahr- und Fräsweg natürlich auf die Tischabmessungen beschränkt ist, gibt es eigentlich keine Längenbegrenzung für das Werkstück: Der Tisch ist beidseitig offen. Die Materialdicke darf maximal 150mm betragen, die Tragfähigkeit gibt der Hersteller mit 100kg an. Die massiven Aluminium-Teile der Fräse sind wasserstrahlgeschnitten und dort, wo es auf Präzision ankommt, gefräst. Viele Kunststoffteile stammen dagegen aus dem 3D-Drucker, etwa die Absaugstutzen und verschiedene Abdeckungen. Die rollengeführten Linearschienen sind gut justiert und arbeiten spielfrei. Unsere Befürchtung, dass die nach unten offenen Linearführungen im Betrieb leicht verschmutzen, bestätigte sich im Test dank der effizienten Staubabsaugung nicht.

Da die X-Achse mit ihrem Gewicht teilweise auf dem Werkstück lastet, genügt eine wenig anspruchsvolle Einspannung; bei ganz leichten Arbeiten (etwa beim Fräsen eines Reliefs oder einer Intarsie) reicht sogar das Eigengewicht der Platte selbst, um ein Verutschen zu verhindern. Ansonsten dienen Nuten im Frästisch zum Einspannen, notfalls kann man das Werkstück auch einfach auf den eingesetzten Multiplex-Platten festschrauben oder mit den üblichen Schraubzwingen festklemmen.

Schmutzt nicht

Gut gelöst ist die Staubabsaugung über den transparenten Staubkorb, der das Werkzeug vollständig umschließt. Der mitgeführte Spiralschlauch wird unterseitig einfach an einen Staubsauger angeschlossen. Beim Setzen des Arbeits-Nullpunkts wird man in der Pro-Version von einem Kreuzlaser unterstützt,



Für die Yeti Tool SmartBench sollte genügend Platz in der Werkstatt sein: Die CNC-Fräse ist rund 2,70m lang.

die Maschine rechnet den Offset zur Werkzeugspitze automatisch heraus.

Der Werkzeugwechsel gestaltet sich durch den geschlossenen Staubkorb etwas umständlicher, man muss dafür den Fräsmotor ausbauen. Dank der passgenauen Klemmung mit leicht zugänglicher Anzugsschraube geht das aber recht schnell.

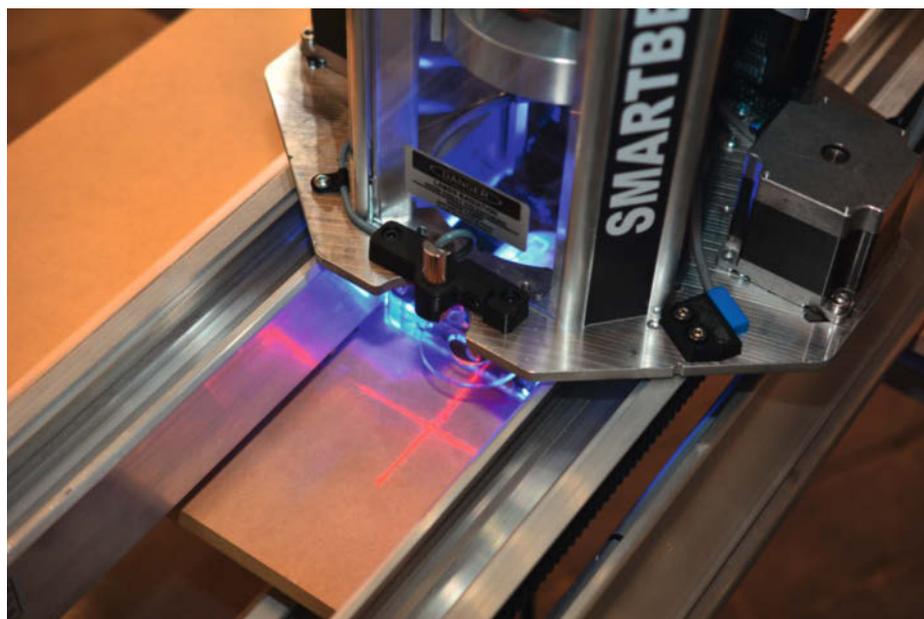
Der Spindelmotor ist bei der uns zur Verfügung gestellten Pro-Version ein 1000W-Modell von Mafell mit Drehzahl-Fernsteuerung über eine separate Steuerleitung. Er ist den bekannten Kress- bzw. AMB-Fräsmotoren recht ähnlich, aber mit einer größeren ER16-Spannzange ausgerüstet. Die Geräusentwicklung hält sich für einen Universalmotor in Grenzen, er ist laufruhig und präzise gefertigt.

Beim „Nullen“ der Z-Achse hilft ein kapazitiv arbeitender Sensor in Form einer runden Scheibe, die seitlich in der Z-Konstruktion steckt und so immer schnell zur Hand ist. Man schiebt sie unter dem Staubfangkorb ein, so dass sie auf dem Werkstück aufliegt. Auf Knopfdruck (oder besser: Touch-Klick) senkt sich die Spindel, bis sie Kontakt zum Sensor hat. Die Steuerung korrigiert die Höhe des Sensors automatisch, so dass sich ein exakter Z-Nullpunkt ergibt.

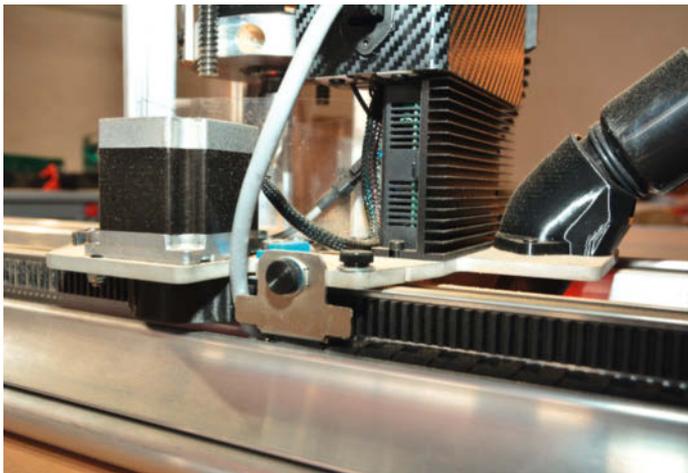
Gut gelöst sind die Schutzeinrichtungen in Form beidseitig der X- Traverse angebrachter Bügel, die auf der gesamten Maschinenbreite ansprechen. Sollte die Maschine gegen ein Spannmittel (z.B. Schraubzwinde) fahren, wird sofort ein Nothalt ausgelöst. Eine Beschädigung des Antriebs oder des Werkzeugs ist damit ausgeschlossen.

Offline-Betrieb

Eine Besonderheit der Maschinensteuerung in Form eines robusten, abnehmbaren Touch-



In der Pro-Version erleichtert ein Kreuzlaser die XY-Positionierung. Gut erkennbar ist der beleuchtete transparente Staubfangkorb.



Der Antrieb erfolgt in X- und Y-Richtung über beidseitige Zahnstangen und insgesamt vier Schrittmotoren.



Beidseitige Schutzbügel (mit roten Kappen) verhindern das Überfahren von Hindernissen.

pads ist, das diese nur „offline“ arbeitet; eine direkte Steuerung der Maschine über G-Codes von einem PC aus ist nicht vorgesehen. Stattdessen überträgt man die mit einer beliebigen CAM-Software vorbereiteten G-Code-Files über USB-Stick oder WLAN auf das Gerät und lässt sie dann abarbeiten.

Da das Gerät unzählige Dateien im internen Speicher halten kann, ist das in der Praxis kein Nachteil; beispielsweise kann ein Lautsprecher-Selbstbauer so auf vorgefertigte Daten für die Ausschnitte der verwendeten Chassis zurückgreifen. Beim stationären Betrieb in der Werkstatt hilft die eingebaute WLAN-Funktion, über die man auf die Daten eines externen Servers (NAS) zugreift.

Im Betrieb wird das Touchpad an der X- Traverse eingehängt und wandert dann in Y-Richtung mit, ebenso wie die unterseitigen Anschlüsse für den Staubsauger und das Netzkabel. Im Betrieb sollte man also darauf ach-

ten, dass unter der Maschine keine Gegenstände gelagert werden, an denen sich Schlauch und Kabel verheddern könnten. Kleiner Nachteil: Auch der seitlich angebrachte Not-Aus-Taster wandert mit.

Die Schrittmotorsteuerung selbst arbeitet mit dem GRBL-Befehlsatz, der den G-Code-Standard nahezu vollständig unterstützt. Mit dem preiswerten ESTLCAM oder dem leistungsfähigen (und vom Hersteller empfohlenen) Vectrix V-Carve lassen sich die G-Code-Dateien problemlos erzeugen. Für „mal eben“-Anwendungen hält das Touchpad noch Funktionen für kreis- und rechteckförmige Fräsungen mit einstellbaren Abmessungen und Rundungen bereit.

Fazit

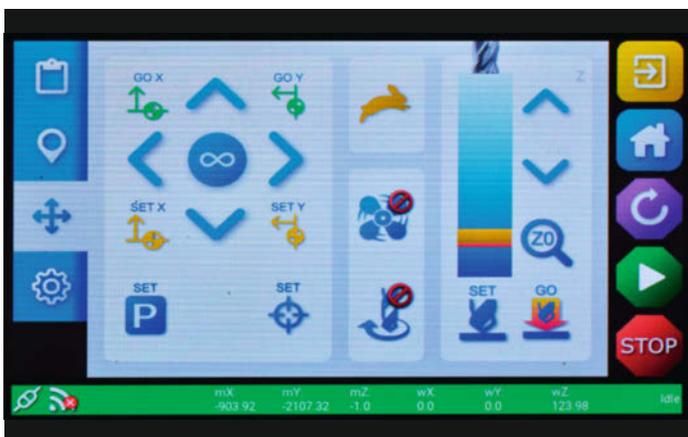
Die Maschine lässt sich dank des Touchpads nach kurzer Einarbeitung auch von (nur

einem CNC-Unkundigen bedienen, so lange nur Plattenware bearbeitet wird.

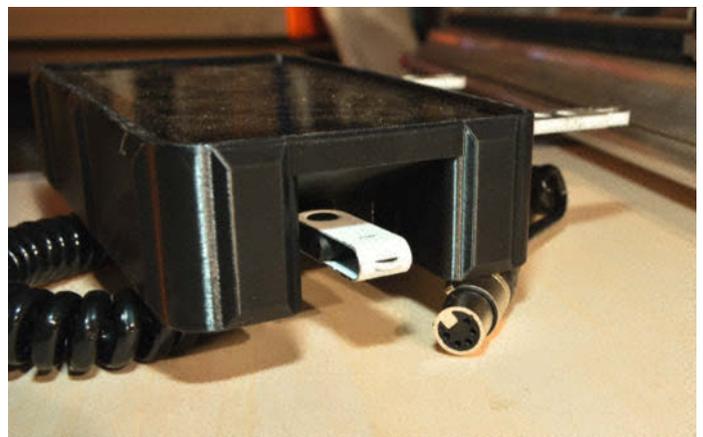
Etwas umständlicher gestaltet sich die Bearbeitung von kleineren Werkstücken: Hierbei muss die X-Traverse beidseitig genau nivelliert werden, da der Vorteil der selbsteinstellenden Rollen-Auflage hier nicht zum Tragen kommt.

Ganz billig ist die große Smartbench-Fräse allerdings nicht: Importeur Sauter ruft für die Basisversion 6698 Euro auf, die uns zur Verfügung gestellte Pro-Version mit Kreuzlaser, doppelter Z-Spindelführung und Echtzeit-Anzeige der aktuellen Motor-Last im Display kostet 7104 Euro. Die Fräse lohnt sich vor allem dann, wenn vorwiegend größeres Plattenmaterial verarbeitet wird, also zum Beispiel für große Fablabs, die öfter mal Möbeltischlernde zu Besuch haben. Für kleine oder „echte“ 3D-Teile ist sie wegen fehlender Spannmöglichkeiten weniger geeignet und auch schlicht überdimensioniert.

—cm



Die Maschinensteuerung in Form eines abnehmbaren Touchpads lässt sich dank großer Symbole auch von Grobmotorikern bedienen.



Das Touchpad wird mit einem robusten XLR-Stecker angeschlossen. An der Unterseite findet sich etwas versteckt der USB-Anschluss für Speichermedien.

Make: Projects

SMART HOME CHALLENGE

präsentiert von



Rolläden
mit Licht-
erkennung



Intelligente
Küche

Auto-
matische
Lüftung

Sprach-
steuerung

**DU HAST DAS SCHLAUESTE
SMART HOME IM GANZEN LAND?**

- Gestalte dein Smart Home
- Verknüpfe Sensoren und Aktoren
- Nutze WLAN zur Kommunikation
- Gewinne einen von **10 Preisen**

STARTE JETZT DEIN SMART HOME-PROJEKT AUF MAKE PROJECTS!

makeprojects.com/de/smart-home-challenge



Shaper Tools GmbH

CNC-Oberfräse Shaper Origin

Am Fließband sorgt die Maschine für groben Vorschub, der Mensch für die Feinarbeit – bei der Origin ist es genau umgekehrt. Will man das? Aber ja! Wäre da nur nicht der Preis ...

von Peter König

Video und Links
im Web unter
make-magazin.de/xntq

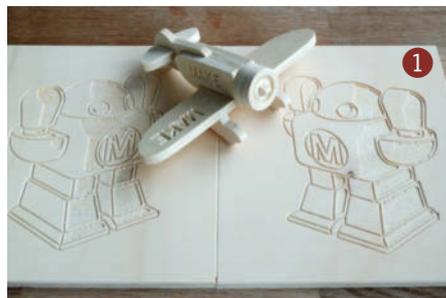
Die *Origin* des Herstellers *Shaper Tools* ist eine smarte Oberfräse, die stets genau weiß, wo sie ist – Position und Ausrichtung im Verhältnis zur Arbeitsfläche sind ihr auf den Zehntelmillimeter genau bekannt. Möglich macht das eine eingebaute Kamera, die auf spezielle Marker achtet – schwarze Rechtecke mit weißem Punktmuster, die ein wenig an Dominosteine erinnern. Diese Marker druckt der Hersteller auf spezielles Klebeband namens *ShaperTape*, das man auf die zu bearbeitende Fläche seines Werkstücks klebt. Dann fährt man mit der Maschine bei ausgeschalteter Spindel systematisch die Arbeitsfläche ab.

Ist dieser Scan abgeschlossen, erscheint auf dem Display der Maschine ein virtuelles Bild der Arbeitsfläche, als würde man aus Sicht der Frässpindel senkrecht nach unten schauen. Jetzt kann man darauf die gewünschten Fräspfade platzieren, die man entweder direkt mit den eingebauten Zeichenfunktionen der Ma-

schine anlegt (Kreis, Rechteck, Linienzüge, Text) oder als Vektorgrafiken im Dateiformat SVG eher traditionell vom eingesteckten USB-Stick holt oder ganz modern übers WLAN aus seinem Online-Account beim Hersteller importiert.

Mini-CNC

Und dann spielt die *Origin* ihren eigentlichen Trumpf aus, denn die Spindel sitzt nicht starr in der Maschine, sondern ist wie bei einer CNC-Fräse in drei Achsen numerisch gesteuert beweglich. Allerdings beträgt der maximale Verfahrenweg in X- und Y-Richtung gerade mal 12,5mm im Durchmesser – das reicht aber aus, dass man zum perfekten Fräsen nach dem Vektorplan die Maschine mit der Hand nur ungefähr dem Pfad entlangführen muss, weil die Maschine in Sekundenbruchteilen automatisch für die Feinpositionierung sorgt, dank des ausgeklügelten Zusammenspiels von Bild-



erkennung für die Positionsbestimmung und der wirklich fix reagierenden Ansteuerung der Motoren für die X- und Y-Achse. Für den Menschen vor der Maschine heißt das in der Praxis: Man schaut die ganze Zeit eigentlich nur auf das Display der *Origin*, schiebt das Gerät nah genug an den nächsten Fräspfad heran, drückt einen Knopf zum Absenken der rotierenden Spindel auf die Arbeitstiefe und schiebt dann

die Fräse so den Pfad entlang, dass der Fräser stets im markierten Kreis auf dem Display liegt. Verrutscht man doch mal, zieht die *Origin* den Fräser so schnell wie möglich zurück, um den Schaden zu begrenzen.

Ja, man kann auch mit dieser Maschine eine Fräsung verpatzen. Umgekehrt erfordert es aber weder viel Mühe noch Übung, sie perfekt hinzubekommen – vorausgesetzt, die Fräser sind scharf, die Drehzahl richtig eingestellt und man achtet darauf, dass die Kamera immer möglichst viele Marker im Blick hat ❶. Das schafft man aber. Wir drehen noch ein Video dazu, das so bald wie möglich nach dem Erscheinen des Hefts über den Link am Ende des Artikels zu finden ist.

Farbcode statt CAM

Verglichen mit einer herkömmlichen Portal-CNC-Fräse fällt der Einstieg in die Arbeit mit der *Origin* sehr leicht, denn zum einen kann man sich alles, was man für den Anfang wissen muss, über drei (deutsch untertitelte) Videos aneignen und es gibt auf der Webseite des Herstellers ausführliche und hervorragend gemachte deutsche Anleitungen zu praktisch jedem Aspekt der Maschine. Zum anderen entfällt verglichen mit der klassischen CNC auch der komplette Brocken des CAM (Computer Aided Manufacturing), bei dem man aus einem CAD-Modell seines Werkstücks mit spezieller Software die Werkzeugbahnen für die CNC-Fräse erzeugt und alle Parameter im Voraus einstellt, die Zustelltiefe, Schrupp- und Schlichtvorgänge, Verfahrenswege und ähnliches.

Zwar beherrscht auch die *Origin* fast alle diese Feinheiten, aber man muss nicht alle Entscheidungen vorab treffen, sondern stellt Frästiefe, Werkzeugdurchmesser, Versatz vom Vorlagenpfad und ähnliche Parameter direkt an der Maschine ein. Dadurch kann man in der Werkstatt flexibler reagieren und etwa beim Taschenfräsen die Tiefe in kleineren Schritten steigern als geplant, falls sich das Holz als zu hart erweist.

Als praktischen Kompromiss zwischen dieser Flexibilität einerseits und dem Wunsch nach Planung vorab kann man den Formen in

Shaper Origin

Hersteller	Shaper Tools, shapertools.com
Leistung Spindel	720 Watt
Drehzahl	10.000 bis 26.000 U/min
Fräserschaftdurchmesser	8mm
Z-Achsen-Hub	43mm
Display	Kapazitives Touch-LCD
Datenübertragung	WLAN, USB-Port
Maße cm (B x T x H)	19,7 x 29,9 x 35
Gewicht kg	6,6
Preis (inkl. Steuer)	3439,10 € Origin, 476 € Workstation (derzeit brutto 119 € Rabatt als Kombination)

seinen SVG-Dateien bestimmte Farben als Füllung und Kontur zuzuweisen, die beim Import gleich die gewünschte Fräsart (Außen- oder Innenkontur, direkt auf der Linie oder als Tasche) vorauswählen. Auf der Fräse kann man im Notfall all das noch mal ändern, aber wer seine Projekte sorgfältig auf dem Rechner vorbereitet, spart sich dadurch eventuell doch ein paar Denkfehler in der Werkstatt.

Mehr als 2D

Mit ihrem aktuellen Betriebssystem fräst die *Origin* jeden Pfad mit wählbarer, aber dann fester Tiefe aus; eine Tasche mit gleichmäßig abfallendem Boden kann man derzeit damit nicht direkt herstellen. Trotzdem erweist sich die Maschine als deutlich vielseitiger, als man auf den ersten Blick meinen mag, insbesondere in Kombination mit der optionalen *Workstation* ❷: Diese Multifunktions-Einspannvorrichtung wird (mit Zwingen oder Schrauben) fest auf der Werkbank befestigt, hält kleine Werkstücke auf unterschiedlichste Weise fest, auch in definierten Winkeln zur Bezugsfläche, womit sich sogar Stirnzapfen für schräge Stuhlbeine fräsen lassen ❸. Im hinteren Teil ist eine waagerechte Bezugsfläche dauerhaft mit Markern bedruckt, sodass man beim Einsatz der Workstation kein *ShaperTape* verbraucht, was mit rund 19 Euro pro 45-Meter-Rolle auch nicht gerade verschenkt wird.

Womit wir beim Preis von über 3000 Euro wären, dem eigentlich einzigen Nachteil der *Shaper Origin*: Für die meisten Marker:innen dürfte sie das Budget sprengen. Leider, muss man sagen, denn es ist leicht, von diesem Gerät begeistert zu sein: Die Arbeit damit ist erstaunlich mühelos und macht Spaß, die Maschine arbeitet zuverlässig und präzise und die Handhabung geht schnell in Fleisch und Blut über. Gegenüber einer klassischen CNC-Portalfräse (also nicht verglichen mit der *Smart Bench* von *Yeti Tool*, S. 68) hat die *Origin* den charmanten Vorteil, dass man mit ihr theoretisch beliebig große Werkstücke bearbeiten und sogar in schon fest eingebaute Arbeitsplatten oder Fußböden CNC-genau Taschen und Aussparungen fräsen kann. Freilich muss man die Maschine die ganze Zeit mit den Händen führen, in dieser Beziehung ist es immer noch eine Oberfräse – die typische CNC-Magie, dass die Maschine nach Knopfdruck alles von selbst macht, gibt es mit der *Origin* nicht. (Aber auch nicht das Bibbern, ob man im CAM-Schritt alles richtig gemacht hat oder ob der Fräser jetzt mit vollem Karacho an ein Spannmittel donnert, das man bei der Pfadplanung nicht berücksichtigt hat). Für Fab-labs und ähnliche Werkstätten ist die *Shaper Origin* allerdings eine Anschaffung, die wahrscheinlich sehr vielen Nutzer:innen ganz neue Möglichkeiten der Holzbearbeitung eröffnen wird. —pek



Shaper Tools GmbH

Shaper Tools GmbH

Stand Up Arcade Machine

Der junge Maker Felix Watzlawek beweist, dass auch in Zeiten von 64bit-Konsolen klassische Arcades immer noch Spaß machen – insbesondere, wenn sie selbst gebaut sind.

von Felix Watzlawek



Heutzutage gibt es Spielekonsolen, mit denen man mehrere tausend Spiele in höchster Qualität spielen kann. Also warum dann solch ein Spielautomat, der nur einen Bruchteil der Leistung mitbringt? Diese Frage lässt sich ganz einfach beantworten: Die heutige Jugend fährt total auf den Retro-Style der 80er und 90er Jahre ab. Es fängt bei der Kleidung an und geht bis zur Einrichtung der Wohnung. Zudem hat mich persönlich ein Arcade-Automat im Internet so begeistert, dass ich daraufhin sofort anfing, das Projekt perfekt durchzuplanen, wobei mich mein Vater unterstützte.

Das Design und die Konstruktion zeigten uns kritische Stellen auf, welche wir dann sofort beheben konnten. Diese Phase war eine der zeitintensivsten, allerdings passte dann später auch alles perfekt zusammen. Ich entschied mich bei der Materialauswahl für MDF- und Multiplexplatten. Die Multiplexplatten benutzte ich jedoch nur für die stabilen Seitenwände, während MDF-Platten für die Monitorabdeckung, die ganze Front- und Inneneinheit benutzt wurden. Nachdem ich alle MDF-Platten zugesägt hatte, wurden die Seitenteile mit der Stich- und Kreissäge in Form gebracht. Die Kanten der Multiplexplatten wurden gleichmäßig verschliffen.

Das ganze Projekt basiert auf einem Raspberry Pi 4 mit 4GiB RAM. Auf diesem läuft *RetroPie*, eine Distribution, welche eine große Anzahl verschiedener Emulatoren bereitstellt (siehe auch Make 4/20, S.8). An den Raspberry Pi schloss ich einen gebrauchten und günstig erworbenen 24-Zoll-PC-Monitor und eine Soundanlage für das perfekte Spielgefühl an. Das Controllerboard samt Joystick und Buttons wurde per USB angeschlossen. Als der Korpus fertig montiert war und die Lage der Elektronik bestimmt, lackierte ich ihn im Rollverfahren mit weißem Lack. Nachdem alles getrocknet war, ging es an das Montieren des Monitors, der Soundanlage und natürlich des Raspberry Pi mit Bedieneinheit. Zu guter Letzt wurden dann die Beleuchtung in Form von RGB-Streifen im beleuchtete Marquee und für die Front angebracht.

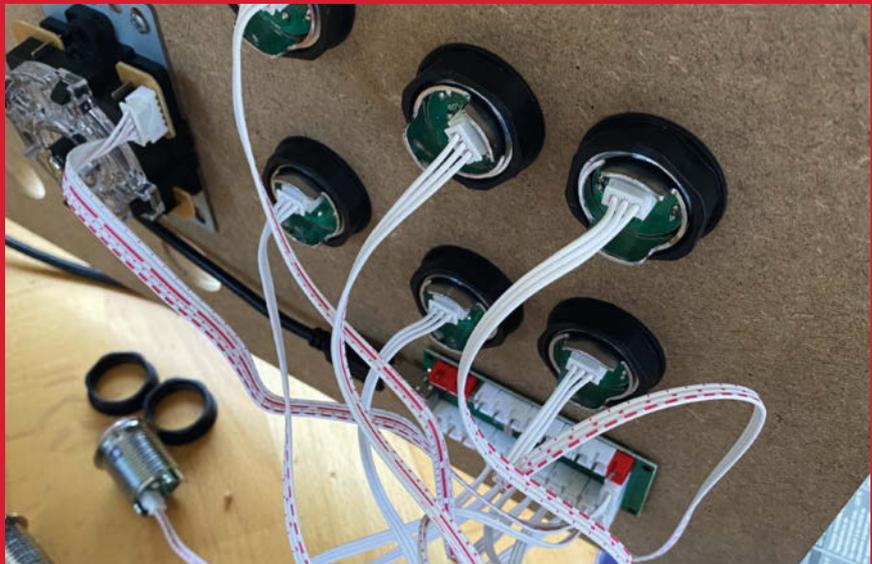
Nun galt es den Automaten „einzuspielen“! Dies übernahmen mein Bruder und ich gemeinsam mit einem schnellen Super-Mario-Kart-Battle. Spiele für den Automaten findet man im Internet und überträgt sie dann per WLAN auf den Raspberry Pi.

Freunde und Verwandte spielen oft mehrere Stunden und kämpfen sich durch Level verschiedener Spiele. Der große Vorteil eines Emulators ist, dass Spiele verschiedener Konsolen auf dem Automaten gespielt werden können, zum Beispiel Nintendo, Playstation 1 und 2, aber auch Spiele des Gameboys. Natürlich ist die Grafikleistung des Raspberry Pi beschränkt, wodurch nur Spiele älterer Konsolen möglich sind. Aber auch die machen einen *mega* Spaß. —caw

► [instagram.com/felixwatzlawek/](https://www.instagram.com/felixwatzlawek/)



Eine gute Planung ist wichtig.



Installation der Arcade Hardware



Gehäuse im Rohbau



Fertig und perfekt ausgeleuchtet

Unendlichkeitsspiegel mit ESP8266

Im Frühling blühen endlich die Blumen wieder – aber leider nicht für immer. Dauerhaft gute Laune macht dieser ESP-gesteuerte *Infinity Mirror* aus dem Lasercutter.

von Lisa Ihde



Rosen sind rot, Veilchen sind blau. Lasercutter schneiden Spiegel, das weiß ich genau! Richtig gelesen, neben gängigen Materialien wie Holz und Acryl kann ein Lasercutter Formen aus Spiegeln schneiden. Das nutze ich, um einen Unendlichkeitsspiegel zu bauen. Diese Spiegel werden oft *Infinity Mirror* genannt, haben etwas Magisches an sich und lassen sich mit einem LED-Streifen selbst bauen. Die Idee ist, dass sich Lichter durch zwei gegenüberliegende Spiegel reflektieren und das immer weiter. Wenn einer der beiden Spiegel halbtransparent ist, kann man durch diesen schauen und der Effekt erinnert an einen unendlichen Tunnel aus Lichtern. Dieser Unendlichkeitsspiegel kann als Blüte an eine Wand geklebt werden, wobei das herausgeführte Kabel wie ein Blumenstiel wirkt.

Für den halbtransparenten Spiegel klebe ich zuerst Blendschutzfolie auf eine dünne Acrylplatte und schneide aus der Platte und einem Spiegel mit dem Lasercutter dieselbe Blütenform aus. Dabei muss die reflektierende Seite nach unten liegen, damit der Laserstrahl sich nicht direkt im Spiegel spiegelt. Die Vorlagen habe ich auf Github zum Download bereitgestellt (siehe Link). Für den Holzrahmen schneidet man ebenfalls mehrmals die gleiche Form, die aufeinandergestapelt und verklebt wird. Außerdem haben einige der Schichten eine kleine Aussparung für das Kabel. Zu guter Letzt können die Blätter aus den Holzresten gecuttet werden.

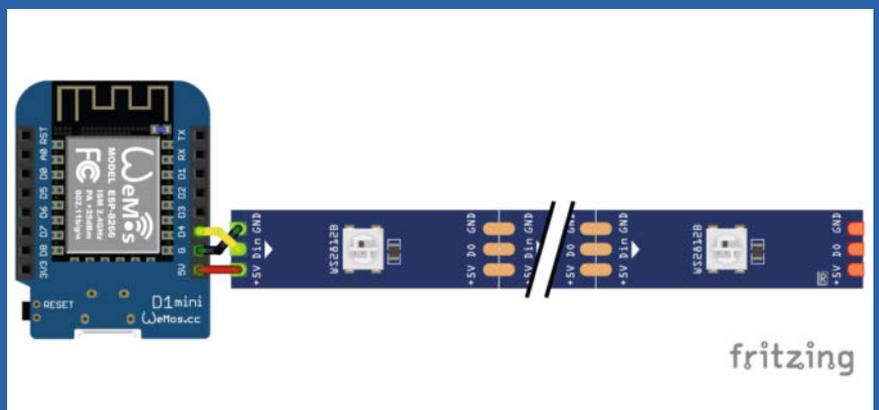
Mit einem Lötkolben bewaffnet wird das drei-adrige Kabel als Verbindung von LED-Streifen und Mikrocontroller befestigt. Danach geht es an das Verkleben des Spiegels und der Holzrahmen: Die drei Schichten mit Aussparung gehören natürlich in die Mitte und der halbtransparente Spiegel wird zunächst nur mit einem Rahmen verklebt – bevor man diesen Deckel schließt, muss erst der LED-Streifen innen befestigt werden. Wer sichergehen möchte, dass die Lötstelle hält, kann Zwei-Komponenten-Klebknete nutzen, um sie zu verstärken.

Um den *Wemos D1 mini* mit der Programmierumgebung von Arduino bespielen zu können, muss die Unterstützung für den Mikrocontroller ergänzt werden. Die Anleitung dazu steht online (siehe Link). Zur Ansteuerung des LED-Streifens nutze ich die *NeoPixel*-Bibliothek von Adafruit. Für einen ersten Test reicht bereits eine der Beispieldateien zur Bibliothek und mit ein paar Anpassungen lassen sich die Farbeffekte an eigene Wünsche anpassen – auch dies erkläre ich online ausführlicher. Der ESP8266 macht es möglich, die LEDs über das Internet anzusteuern; außerdem könnte man mit dem Lasercutter den Spiegel gravieren – beispielsweise mit einem Geburtstagsgruß. —hch

► heise.de/-5999848



Die geschnittenen Teile aus dem Lasercutter



Anschluss des LED-Streifens an den Wemos



Das fertige Unterteil mit eingeklebtem Leuchtstreifen

Maker Education macht Informatikunterricht kreativ

Nur Exceltabellen ausfüllen und am Taschenrechner Tetris spielen? So sieht der Informatikunterricht an Schulen schon lange nicht mehr aus. Aber es reicht nicht, aus Schülerinnen und Schülern die Informatikstudent:innen von morgen zu machen: Sie müssen auch lernen, frei und kreativ mit Technik umzugehen. Wie das aussehen kann, zeigt das Projekt *SDG Arcade*.

Make: Education
für Lehrkräfte

von Mirek Hančl



Wenn man sich den Unterricht in einem Informatik-Abiturkurs vorstellt, weckt das oft Assoziationen von Programmierer:innen in Start-ups: Konzentriert starren alle auf ihre Bildschirme und tippen vor sich hin. Dabei soll im Informatikunterricht gar keine reine Programmierausbildung stattfinden.

Welche Fachinhalte im Informatikunterricht in der gymnasialen Oberstufe vermittelt werden sollen, ist durch die curricularen Vorgaben in den einzelnen Bundesländern festgeschrieben. Beispielsweise enthält ein üblicher Rundumschlag über die vier Semester der Qualifikationsphase in Niedersachsen – also dem 12. und 13. Jahrgang – folgende Themengebiete: *Algorithmik und Datenstrukturen, Objektorientierung, IT-Systeme und Netzwerke, Automaten, Datenbanken und Datenschutz*. Viele Fachbegriffe und damit auch viele Umsetzungsmöglichkeiten. Tatsächlich könnte man die Schulinformatik vollständig auf digitalen Unterrichtsmitteln aufbauen: Apps und Programmiersprachen sind typische Werkzeuge zum Modellieren, Simulieren und Implementieren. Doch als Lehrkraft will man das in Wirklichkeit nicht.

Zugegeben – es ist einfach, mit Laptops oder Tablets auf den Tischen neunzig Minuten in Präsenz zu verbringen und dazu einfach in die passende App zu wechseln. Doch im Unterricht müssen Lehrer:innen neben den fachbezogenen Kompetenzen noch weitere vermitteln – nämlich solche, die auf den Lernprozess selbst abzielen: Kommunikation und Kreativität.

Bleiben wir exemplarisch in Niedersachsen: Neben dem *Strukturieren und Modellieren, Algorithmisieren und Implementieren* sind die prozessbezogenen Kompetenzbereiche *Kooperieren und Kommunizieren* sowie *Kreatives Schaffen und Problemlösen* im Kerncurriculum Informatik für die gymnasiale Oberstufe verankert:

Kooperieren und Kommunizieren

- PK3.1 (Schüler...) kommunizieren unter Verwendung der Fachsprache über informatische Inhalte und stellen diese sachgerecht dar.
- PK3.2 dokumentieren ihre Lösungsansätze und Lösungen mithilfe geeigneter Darstellungsformen.
- PK3.3 begründen Zusammenhänge im Kontext der Informatik.
- PK3.4 organisieren, dokumentieren und reflektieren die gemeinsame Arbeit im Team.

Kreatives Schaffen und Problemlösen

- PK4.1 erweitern gegebene Programme, Algorithmen und Modelle um eigene zusätzliche Funktionalitäten.

Kurzinfo

- » So sehen die curricularen Vorgaben aus
- » Kreative Projekte für den Unterricht finden
- » Konkretes Projektbeispiel: SDG Arcade

Mehr zum Thema

- » Make Education: Newsletter und kostenlose Make-Artikel
- » Elke Schick, Maker Education in der Schule, Make 3/19, S. 64

Alles zum Artikel im Web unter make-magazin.de/xehs

- PK4.2 finden und erläutern Problemstellungen, die mit Hilfe informatischer Kompetenzen gelöst werden können.
- PK4.3 geben unterschiedliche Lösungswege für ein selbst gestelltes oder gegebenes Problem an und entscheiden sich begründet für einen Weg.
- PK4.4 erfinden Produkte oder Verfahren, indem sie informatische Konzepte, Strategien und Methoden in eigenständigen Wegen kombinieren.

Die Umsetzung

Aus den auf Kreativität ausgerichteten Lernzielen lässt sich ableiten, dass Informatikunterricht über das Arbeiten an rein digitalen Projekten hinaus möglich ist – und es auch sein sollte. Dazu kann man Lernziele kombinieren; zum Beispiel Produkte erfinden, in denen informatische Konzepte und Methoden in eigenständigen Wegen kombiniert (PK4.4) und die unter Verwendung der Fachsprache

kommuniziert und sachgerecht dargestellt (PK3.1) werden. All das, während gleichzeitig im Team organisiert, dokumentiert und reflektiert gearbeitet (PK3.4) wird. Fazit: Reiner Frontalunterricht ist hier völlig unangebracht.

Denkbare Umsetzungsmöglichkeiten liefert das Kerncurriculum (KC) in Niedersachsen gleich mit: projektorientierter Unterricht. Dabei „[...] arbeiten die Schülerinnen und Schüler über einen längeren Zeitraum selbstständig an einem Thema. Die Selbstorganisation der Lerngruppe, die Planung der gemeinsamen Arbeit und das Erstellen eines fertigen Produkts, das präsentiert und evaluiert wird, stehen im Mittelpunkt“, so das KC weiter.

Lobenswert ist auch, dass sich die Projektarbeit nicht auf die obligatorischen Themen im KC beschränken muss: „Nach Interesse der Lerngruppe können auch andere Inhalte der Informatik Gegenstand eines Unterrichtsprojektes sein. Die Aufgabenstellung ist dabei so zu gestalten, dass sie kreatives Schaffen und Problemlösen ermöglicht.“



Der besondere Rahmen von projektbasiertem Lernen kann Schüler enorm motivieren.

Mirek Hančl



Die SDG Arcade des Lessing-Gymnasiums Uelzen vereinigt viele einzelne Coding- und Making-Techniken in einem gemeinsamen Artefakt.



Die SDG Arcade ist im Abiturstoff Informatik entstanden – dabei wurden unter anderem ein alter Monitor und Saftkartons recycelt.

Projektbasiertes Lernen ist ein Prozess, ...

Projektbasiertes Lernen berücksichtigt die Interessen und das aktive Tun der Schülerinnen und Schüler nicht nur, sondern fordert sie jedes Mal aufs Neue ein. Dafür kann es keine Generalvorlage geben.

Projektbasiertes Lernen startet immer mit einer gemeinsamen Idee, die oft zufällig gefunden wird. In der Planungsphase werden

die Projektziele abgesteckt, die notwendigen Ressourcen festgelegt und ein Zeitplan erstellt. Die anschließende Arbeitsphase lebt von Erfolgen und Rückschlägen, die gemeinschaftlich evaluiert und durch Problemlösungen iterativ gemeistert werden. Die abschließende Präsentation bietet allen Beteiligten die Möglichkeit, ihre Ergebnisse entweder vor der ganzen Lerngruppe oder sogar der Schulöffentlichkeit vorzustellen und anzupreisen. Dies ist ein sehr individueller Lernprozess.



Nach den 17 Zielen für nachhaltige Entwicklung müssen sich alle Staaten der Vereinten Nationen richten.

... der etwas ruckeln kann

Soweit die Theorie; in der Praxis ist schon die Ideenfindung eine kreative Herausforderung. Auch die Planungsphase zeigt lebhaft, wie viele Faktoren projekthemmend sein können: Zum Beispiel die 45-Minuten-Taktung sowie die räumliche und materielle Ausstattung in der Schule. Und letztendlich muss man den Projektverlauf und die Dokumentation bei der Notengebung berücksichtigen können. Die alleinige Bewertung des Ergebnisses ist nicht angemessen (und in Niedersachsen auch gar nicht gestattet).

Auch wenn die Projektarbeit vor dem ersten Start viele Unsicherheiten aufwirft, werden dafür alle Beteiligten bald von einer Art gemeinsamen Antrieb, nicht nur intrinsisch, erfasst und belohnt. Schnell stellt man fest, dass projektbasiertes Lernen und Kreativität im Informatikunterricht eine besondere Form des Experimentierens und Entdeckens ermöglichen. Sie ist weniger streng naturwissenschaftlich als vielmehr künstlerisch und konstruktivistisch, genau wie beim Making.

Wenn man dabei die weiteren prozess- und fachbezogenen Lernziele, zum Beispiel zur Algorithmik und zur Verwendung von Fachsprache, mitverfolgt, hievt man die Schul informatik auf eine neue Ebene: eine fantastische Ebene des fächerübergreifenden, nachhaltigen Lernens. Die Lerngemeinschaft setzt sich so selbstbestimmt und selbstorganisiert mit einem Thema aus der Welt der Informatik auseinander.

Anknüpfungspunkte zur Maker Education

Sucht man nach geeigneten Werkzeugen und Materialien – vielleicht sogar nach einem passenden Mindset – so ist die *Maker Education* bestens geeignet, um mit ihr die Projektphasen mit der Lerngruppe durchzuführen. Denn Maker Education legt den Fokus auf den Lernprozess und nicht das fertig werden allein. Während die Schüler:innen Dinge erschaffen, können sie sich selbst in kreativer Weise mit einem Lerngegenstand auseinandersetzen und so ihre Vorstellungen, Meinungen und Absichten in einem Produkt vereinen. Diese Möglichkeit motiviert die Lernenden kräftig – das steigert ihre Bereitschaft, sich mit dem Lerngegenstand praktisch auseinanderzusetzen.

Durch Ausprobieren, Tüfteln, Basteln, Bauen und Konstruieren werden beim Making Ideen im Kopf in die Wirklichkeit übertragen. Gemeinsam im Team entstehen großartige Dinge mit Ecken und Kanten, die deutlich machen, dass es nicht um Perfektionismus geht, sondern um Einzigartigkeit. Maker Education kann mit universellen und mit spezialisierten Werkzeugen durchgeführt werden, je nach Ausstattung in der Schule, bei Projektpartnern oder daheim.

Ob Schere, Cutter und Heißklebepistole oder Stichsäge, 3D-Drucker und Lasercutter: Für die Durchführung des Unterrichtsprojekts ist in der Planungsphase ein gemeinsamer Blick in die Werkzeugkiste nützlich, damit es in der Arbeitsphase keine Enttäuschungen gibt. Auch mit alltäglichen Dingen wie Papprollen, Schnur, Joghurtbechern oder dem Upcycling von Rest- und Sperrmüll können einzigartige Gebilde gebaut werden. Mit Filament und einem 3D-Drucker lassen sich Teile passgenau produzieren und mit anderen Werkstoffen kombinieren. Hier wird die Kreativität der Schüler:innen ordentlich gefordert.

Maker Education im Informatikunterricht lässt selbstverständlich das Erstellen von Projekten zu, die ohne jegliche informationstechnologische Funktion sind. Noch motivierender, interaktiver und curricularer wird es aber, wenn Mikrocontroller oder Einplatinencomputer verbaut und so programmiert werden, dass sie Teil des Ganzen sind. Wenn in die Planung und Erschaffung des Artefakts Aspekte der Mensch-Maschine-Interaktion einfließen und die Bedienbarkeit in praktischen Tests erprobt und diskutiert wird, dann erst findet nachhaltiger Informatikunterricht statt: Zum einen haben die Schülerinnen und Schüler den Menschen als Benutzer ihres Artefakts im Fokus ihres Lernprozesses und nicht allein die Technik selbst. Zum anderen haben sie sich und ihr Team während der Projektarbeit im Blick und stärken dadurch ihre kommunikativen und kooperativen Kompetenzen – was nicht nur in der Schulzeit von Vorteil ist.

Wie ein Unterrichtsprojekt im Informatikunterricht eines Abiturkurses aussehen kann, zeigt das Projekt *SDG Arcade*. Es thematisiert bewusst Nachhaltigkeit und bringt sie mit den kompetenzorientierten Vorgaben und Lernzielen des Kerncurriculums in Verbindung.

Spielen für die Nachhaltigkeit

Die *SDG Arcade* ist ein vom Abiturkurs Informatik des Lessing-Gymnasiums Uelzen selbst gebauter und programmierter Spielautomat. Er gamifiziert die *17 Ziele für Nachhaltige Entwicklung* (Sustainable Development Goals, kurz: SDGs) der Vereinten Nationen.

Wenn man im fächerübergreifenden Ansatz die 17 Nachhaltigkeitsziele ausgerechnet in Informatik durchführen soll, liegt nahe, dass man sich mit den Themen Elektroschrott und geplanter Obsoleszenz auseinandersetzen muss. Tatsächlich haben die Schüler:innen das im Projekt *SDG Arcade* auch gemacht.

Für den Spielautomaten konnten die Schüler:innen gebrauchte Hardware upcyclen. Sie haben zum Beispiel einen alten Flachbildmonitor (der kurz vor der Verschrottung stand) und die üblichen Making-Materialien wie Steckbretter, Arduino Nano und Schaltaht



Beim Unterbau wurde nur eine Holzplatte für die Verkleidung gekauft. Der Rest ist wiederverwertet.

aus vergangenen Projekten wiederverwendet. Recycelt wurden auch ein altes Regalbrett, Latten, eine alte Acrylglascheibe eines Gartenhauses, jede Menge Klebefolienreste sowie 17 Saftkartons und eine programmierbare Lichterkette mit RGB-LEDs.

Die Konstruktion des *SDG Arcade* beinhaltet neben dem Making (Verarbeitung von Holz, Plexiglas, Saftkartons, Klebefolie, Lack, Arbeiten mit Bügelsäge, Cutter, Bohrer, Heißklebepistole, Schneideplotter, Elektronik mit Raspberry Pi Zero, Arduino Nano, RGB-LEDs, Joysticks, Arcade-Buttons, Verkabelung, Crimpen...) auch das Programmieren von edukativen Minispielen, sogenannten *Serious Games*. Das Thema waren die 17 Ziele für nachhaltige Entwicklung, gearbeitet wurde in der Programmierumgebung *Arcade MakeCode*.

Das Projekt haben die Schüler:innen mit kollaborativen Tools wie dem webbasierten Vektorgrafikprogramm *Figma* und dem digitalen Whiteboard *Miro* online durchgeführt. Das Team wollte sich durch die Einschränkungen des ersten Corona-bedingten Lockdowns nicht stoppen lassen, die *SDG Arcade* fertigzustellen. Mein Beitrag zum Projekt war es (neben dem Lieferdienst des entstehenden, realen Arcade-Automaten von einem Abiturienten zum nächsten), die Ansteuerung des Arduino Nano vom Raspberry Pi aus den *Serious Games* heraus vorzuprogrammieren.

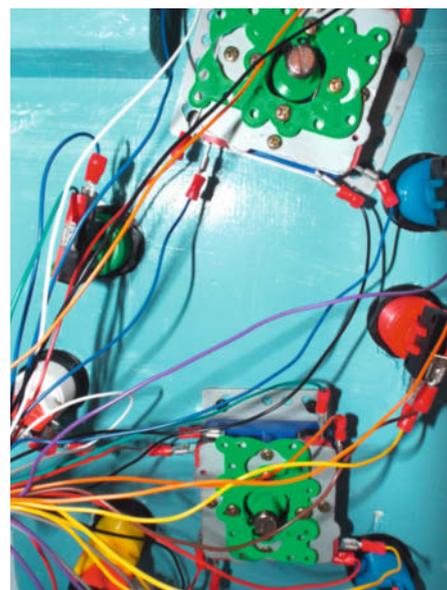
Mithilfe eines Einzelers am Ende jedes Spiels sorgten die Abiturient:innen dann dafür, dass eine RGB-LED das entsprechende SDG-Logo zum Spiel illuminiert, wenn es erfolgreich durchgespielt wurde. Sind alle 16 Spielziele erreicht, leuchtet SDG Nr. 4 „Hochwertige Bildung“ ganz oben am Arcade auf.



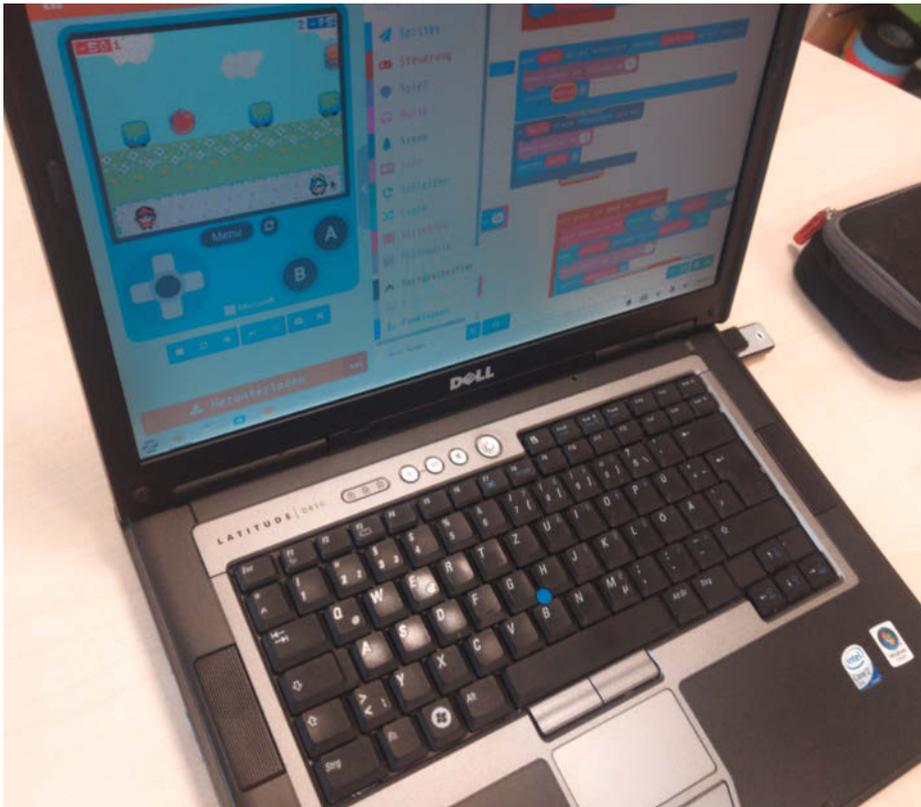
In der Arcade versteckt sich viel Elektronik aus vergangenen Projekten.

Die Abschlussanimation mit Regenbogeneffekt und virtuellem Konfetti bestätigt den Spielern, dass sie sich mit den Zielen für nachhaltige Entwicklung spielend ernst auseinandergesetzt haben.

Meine Erfahrung aus dem Unterrichtsprojekt ist, dass projektbasiertes Lernen und Maker Education, wenn es die Schülerinnen und Schüler erstmal „gepackt“ hat, sicherlich für alle arbeitsintensiver, dafür aber die Lernkurve der Lernenden deutlich steiler und ihre Selbstmotivation merklich größer ist. Zudem könnten viele der durchgeführten Aktionen



In der Projektarbeit lernen die Schüler:innen nicht nur Arcade-Buttons anzuschließen, sondern auch koordiniertes Teamwork.



Mit kollaborativen Tools und der Programmierumgebung Arcade MakeCode haben die Schüler:innen die SDG Arcade entwickelt.

Außerdem ist das Projektergebnis natürlich ein Highlight! Es ist sehr beeindruckend, wenn so ein reales Artefakt im Raum steht: rund einen Meter hoch, bunt, mit Joysticks, Buttons, Bildschirm und 17 blinkenden SDG-Leuchten, wartet es nur darauf, gespielt zu werden. Die SDG Arcade ist seit ihrer Fertigstellung ein Blickfang in der Schule und wird vor dem Informatikunterricht regelmäßig eingeschaltet. Dabei entstehen spannende Diskussionen mit insbesondere den jüngeren Schüler:innen. Fragen wie „Was sind denn die Ziele für nachhaltige Entwicklung?“, „Wer hat den Automaten gebaut?“, „Wie wurden die Spiele programmiert?“ bieten vielfältige Anknüpfungspunkte für Sequels zu den Themen Nachhaltigkeit, Gamification, Fächerübergreif, Maker Education, Projektbasiertes Lernen, und immer zentral im Mittelpunkt: Informatik!

Mit Auszeichnung bestanden

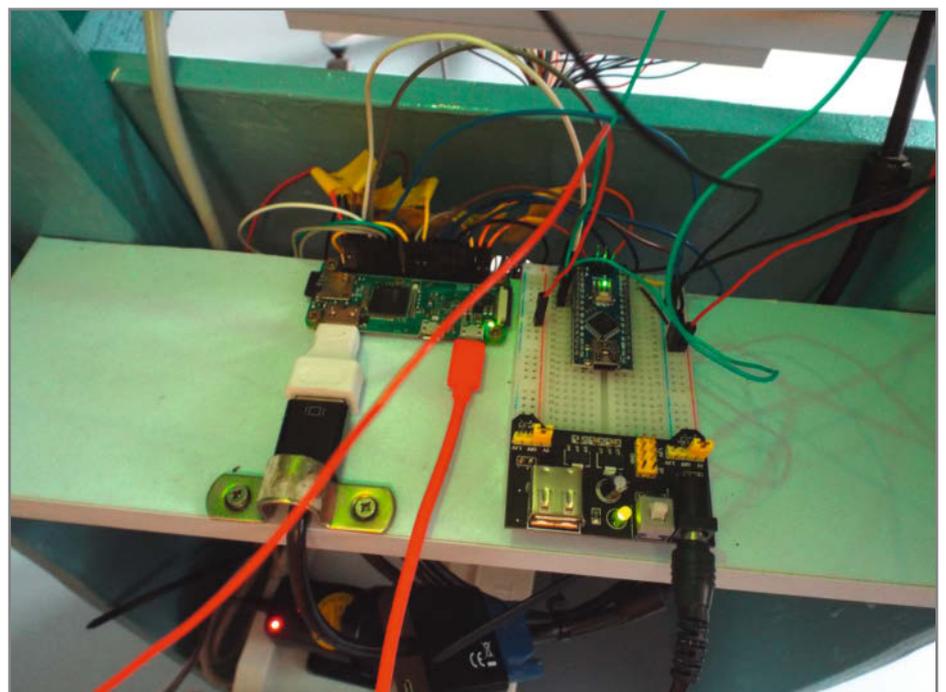
Der SDG Arcade hat den Sonderpreis des Deutschen Multimediapreises 2020 gewonnen, der unter dem Jahresthema *Bits&Bäume* stand. Im Schüler:innenwettbewerb *Hack-TheSummer* der GI belegte eine Teilgruppe der Abiturient:innen mit dem SDG Arcade den zweiten Platz. Ihr Wettbewerbsbeitrag war die Erstellung von freien Bildungsmaterialien (OER) zu SDG Arcade, um besonders jüngeren Schüler:innen das Programmieren und die 17 Nachhaltigkeitsziele der Vereinten Nationen näherzubringen und große und kleine SDG Arcades zu realisieren. Schüler-naher kann Nachhaltigkeit nicht vermittelt werden.

—rehu



Kreativarbeit: Die Bemalung der Arcade

im projektbasierten Lernen gar nicht anderweitig umgesetzt werden. Unter anderem die Elemente des Making, wie Materialkunde und -verarbeitung, ließen sich nur in der Digitalwerkstatt der Schule an Werkbänken und Maschinen praktisch vermitteln.



Bei der Ansteuerung des Arduino Nano vom Raspberry Pi brauchte es etwas Lehrer-Unterstützung.

Die Webinar-Serie von Heise

DevOps in der Praxis

In 5 Online-Trainings zum DevOps-Experten

Lernen Sie in fünf halbtägigen Webinaren, wie Sie zeitgemäße DevOps-Tools und Techniken in Ihrem Unternehmen umsetzen können – angefangen beim Umsetzen von CI/CD-Pipelines und Testautomatisierungen bis hin zum Lösen kultureller Herausforderungen, Security-Verbesserungen mit DevSecOps und einem GitOps-Workflow.

DIE TERMINE:

29. Juni 2021

Wir bauen uns eine Continuous-Delivery-Pipeline

8. Juli 2021

Testautomatisierung in CI/CD-Pipelines

20. Juli 2021

DevOps – Wie schaffe ich die richtige Unternehmenskultur?

27. Juli 2021

DevSecOps: Wie bringe ich Security in meine Entwicklungs-Pipeline?

Fünf umfassende Online-Trainings zu DevOps und Continuous Delivery:

- Aufsetzen des kompletten CI/CD-Workflows
- Automatisierung von Tests und Verbesserung der Sicherheit Ihrer Anwendungen
- Lösen kultureller Herausforderungen zwischen Dev und Ops
- Zeitgemäße Umsetzung mit GitOps-Techniken

5. August 2021

Continuous Deployment mit GitOps

Exklusiver Kombi-Preis: 429 €

Einzelpreis: 139 €



JETZT ANMELDEN

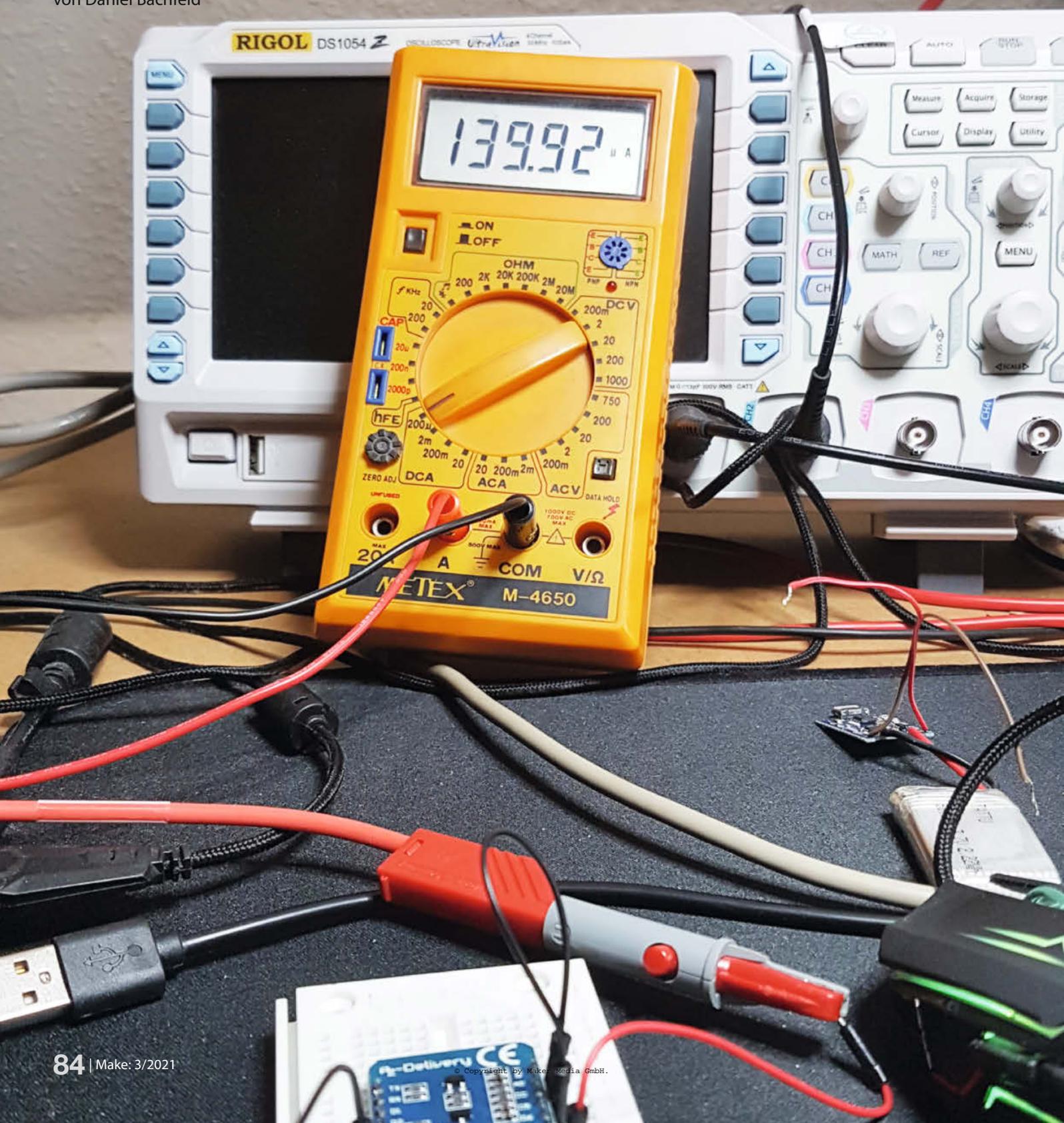
webinare.heise.de/devops/

Jetzt Kombi-Rabatt sichern und über 250 € sparen!

Strom sparen bei ESP-Mikrocontrollern

„Mit Netzteil kann ja jeder“, sagte der Akku zum ESP und entlud sich tief. Damit Ihre IoT-Projekte nicht mit dem Akkubetrieb hadern, zeigen wir ein paar Maßnahmen, um Strom zu sparen und eine lange Laufzeit zu erzielen.

von Daniel Bachfeld



Dank der ESP-Familie von Espressif ist der Einsatz von WLAN und Bluetooth in eigenen IoT-Projekten mit wenigen Zeilen Code machbar. Projekte werden damit für wenige Euro drahtlos und unabhängig von Netzwerkkabeln. Mit *Wireless Electricity* zur Stromversorgung ist man in der Praxis leider noch nicht so weit, abgesehen von den Ladeschalen für Smartphones. Viele Funkprojekte setzen deshalb auf eine Versorgung mit Akkus, im einfachsten Fall mit einer klobigen USB-Powerbank, die im schlimmsten Fall um ein mehrfaches größer als das eingesetzte ESP-Board ist. Das sieht nicht nur doof aus, wenn eine Platine mit dem Kabel an der Bank baumeln, oft ist auch schlicht kein Platz.

Einige ESP-Boards **1** sind bereits mit einer JST-Buchse bestückt, an die man einen einzelnen, kleinen LiPo-Akku anschließen kann, der durch einen Laderegler sogar über die USB-Buchse geladen werden kann. Klingt erstmal ganz gut, in der Praxis zeigt sich aber schnell, dass man den Stromhunger üblicher Boards unterschätzt. Kurzzeitige Stromaufnahmen von 300mA beim Senden und 90mA im Pausenmodus saugen einen Akku überraschend schnell leer. So ist ein Akku mit 1000mAh in knapp 10 Stunden durch mit der Energielieferung.

So lässt sich ein Sensor im entfernten Gartenhäuschen, der Wasserstandsmelder in der Zisterne oder der mobile CO₂-Warner natürlich nicht sinnvoll betreiben. Es gibt aber ein paar Maßnahmen, um den Stromhunger seiner Projekte zu reduzieren: Stromspar-Modi der ESP-Familie und *Muntzing* (zur Herkunft des Begriffs siehe auch die Kurzinfor). Die Stromspar-Modi sind Modi, in denen der ESP32 und der ESP8266 ihr WLAN und die CPU für einige Zeit schlafen legen und so wenig Energie verbrauchen. Beim Muntzing reduziert man die Komponenten in einem elektronischen Gerät auf das notwendigste. Fangen wir mit dem einfachsten an: den Stromspar-Modi.

Schlaf gut

Sowohl der ESP8266 als auch der ESP32 unterstützen drei Stromsparmodi: *Modem-Sleep*, *Light-Sleep* und *Deep-Sleep*. Der ESP32 kann zusätzlich noch in einen Winterschlaf - *Hibernation* genannt - verfallen, in dem ihn nur noch seine Real Time Clock (RTC) wecken kann. Im Modem-Sleep schalten die ESP-Prozessoren ihr Funkmodul aus. Beim ESP8266 ist also WLAN deaktiviert, beim ESP32 zusätzlich auch Bluetooth. In der Tabelle *Sparmodi ESP-Mikrocontroller* sehen Sie, wieviel Strom die Module im jeweiligen Mode noch benötigen.

Welche Einsparungen das im Vergleich zum Betrieb ohne Sleep-Modes bringt, ist allgemein schwer zu beziffern, weil die Funkmodule im aktiven Zustand je nach Situation sehr unterschiedlich viel Strom benötigen. Das

Kurzinfor

- » Diese Stromspar-Modi gibt es
- » So programmiert man sie
- » Weitere Maßnahmen zum Stromsparen

Mehr zum Thema

- » Robert Kränzlein, LoRaWAN-Sensor für den Heizöltank, Make 6/20, S. 66
- » Carsten Meyer, Editorial „Muntzing“ c't Hacks 2/14, S. 3
- » Guido Burger, Projekte für die Hosentasche, c't Hacks 3/14, S. 154

Alles zum Artikel im Web unter make-magazin.de/xe9j

hängt unter anderem von der Umgebung, der aktuellen Sendeleistung, der Geschwindigkeit, der Häufigkeit von Datenübertragungen und anderen Faktoren ab.

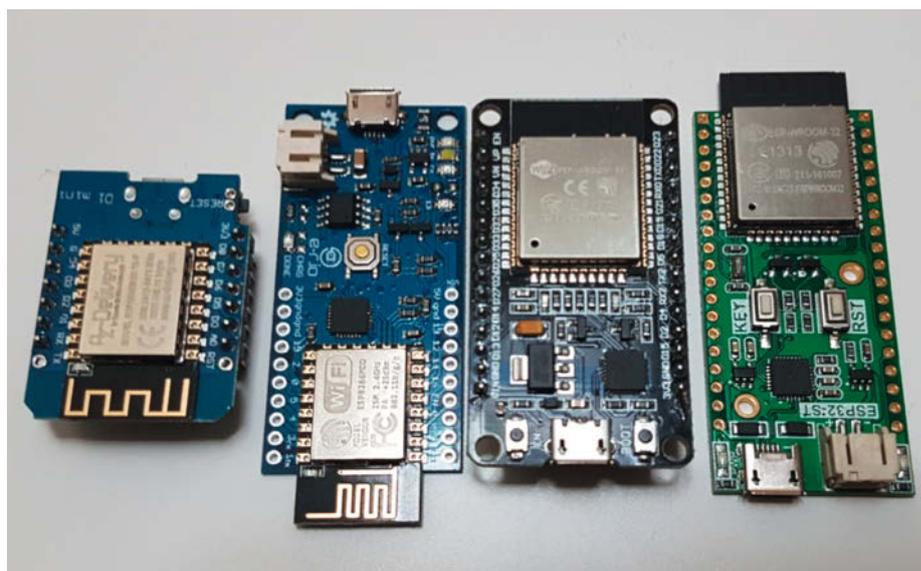
Espressif gibt für den ESP8266 als Stromaufnahme für den Empfang knapp 56mA an, wohingegen das Senden im Mittel maximal 170mA benötigt. Kurzzeitig kann der ESP8266 aber auch 400mA ziehen. Beim ESP32 liegen die Durchschnittswerte zwischen minimal

95mA und maximal 240mA, beim Senden steigt der Strom ebenfalls kurzzeitig an, auf mehrere 100mA.

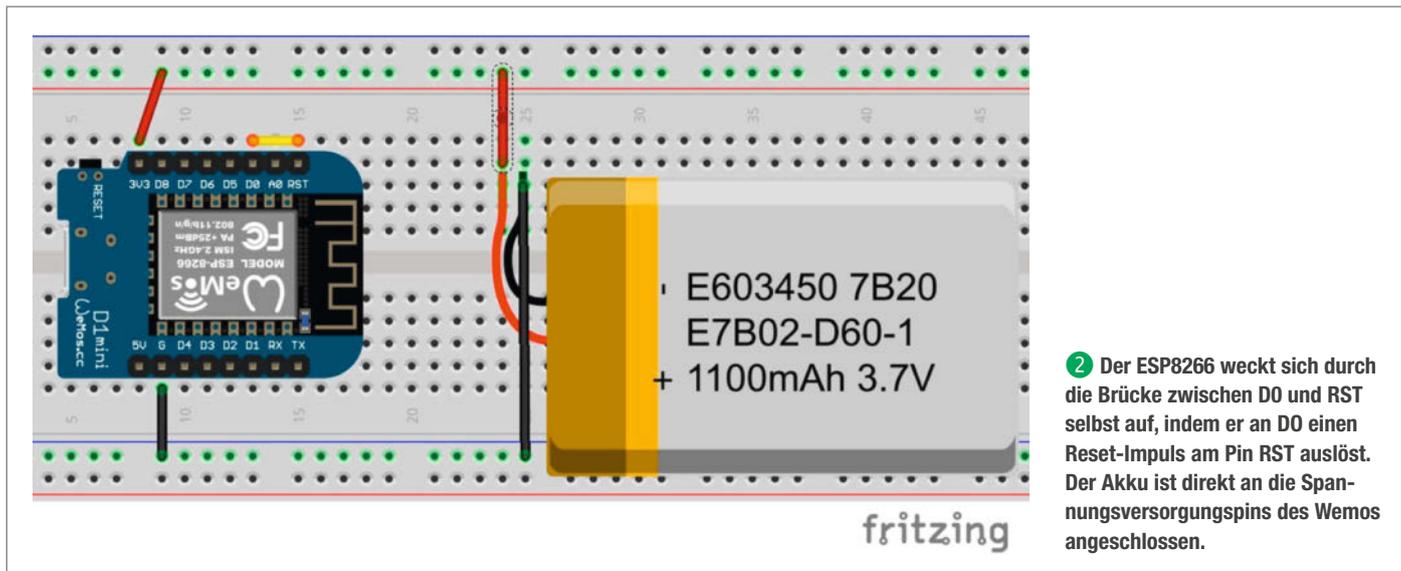
Aber auch mit ausgeschaltetem Funkmodul ist die Stromaufnahme insbesondere beim ESP32 noch ziemlich hoch und für einen Akkubetrieb nicht zu gebrauchen. Deshalb kann man im Light-Sleep zusätzlich noch die CPU in einen Pausenmodus schicken und sie bei Bedarf von verschiedenen internen und ex-

Sparmodi ESP-Mikrocontroller

	Modem-Sleep	Light-Sleep	Deep-Sleep
WLAN	Aus	Aus	Aus
CPU	An	Pausierend	Aus
RTC (Echtzeituhr)	An	An	An
Stromverbrauch ESP8266	ca. 15mA	ca. 0,4mA	ca. 20µA
Stromverbrauch ESP32	4mA (2MHz) - 68mA (240MHz)	ca. 0,8mA	ca. 10µA



1 Boards mit ESP8266 und ESP32. Auch wenn einige mit Anschlüssen für Batterien oder Akkus ausgestattet sind, ist der Betrieb damit kaum sinnvoll.



2 Der ESP8266 weckt sich durch die Brücke zwischen D0 und RST selbst auf, indem er an D0 einen Reset-Impuls am Pin RST auslöst. Der Akku ist direkt an die Spannungsversorgungspins des Wemos angeschlossen.

ternen Ereignissen wieder starten. Da der Inhalt des RAM im Light-Sleep erhalten bleibt, kann die CPU im Programm da weitermachen, wo sie aufgehört hat. Auch dann fließen aber noch 0,8mA beim ESP32 und 0,4mA beim ESP8266.

Tiefschlaf

Im *Deep-Sleep-Mode* wird beim ESP8266 alles außer der internen Echtzeituhr (RTC) abgeschaltet, womit die Stromaufnahme auf

20µA sinkt. Die RTC kann nur einen Timer überwachen und nach einer festgelegten Zeit einen Low-Impuls am Pin GPIO16 (oft D0 auf Boards) ausgeben. In der Regel verbindet man D0 mit RST 2, um einen Reset auszulösen und die CPU und das Funkmodul neu zu starten. Alle Programme laufen komplett von vorne los. Eine Verbindung zum WLAN muss inklusive Anmeldung am Access Point vollständig neu aufgebaut werden. Insofern ähnelt der Modus eher einem jahrelangen Koma: Man muss wieder alles neu lernen.

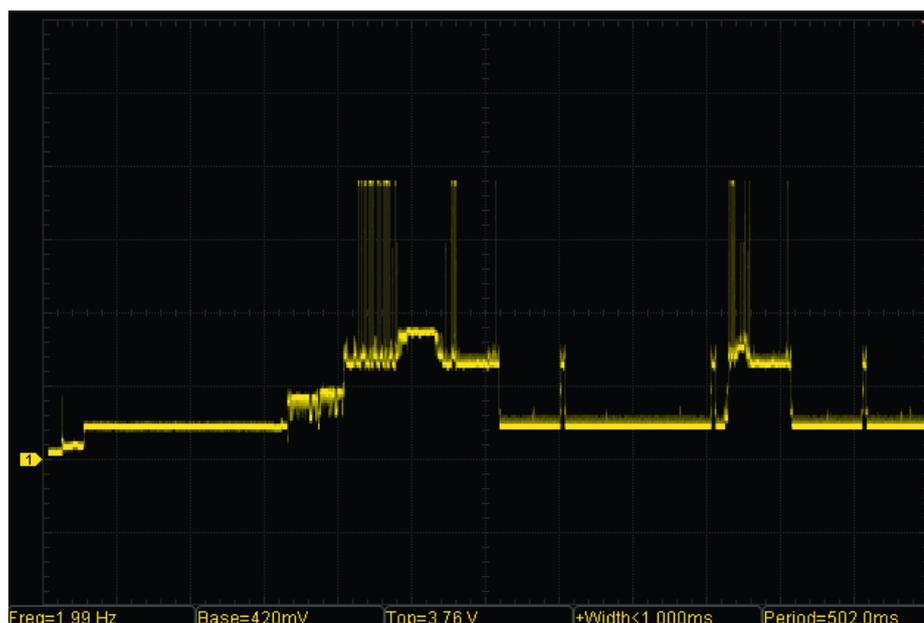
Beim ESP32 läuft es hingegen erheblich flexibler: Man kann mehrere interne und externe Quellen als eine *Wake Up Source* angeben (siehe Tabelle *Wake Up Sources*). Sofern ebenfalls nur die RTC als Wecker läuft, nimmt der ESP32 im günstigsten Fall sogar nur 10µA auf. Alternativ kann aber noch der integrierte Ultra-Low-Power-Coprozessor (ULP) als Wecker fungieren. Er ist sogar in der Lage, Sensoren per I²C abzufragen und bei Bedarf den großen Prozessor zu wecken, etwa um die Daten per WLAN zu versenden.

Der ULP nimmt zwischen 25µA und 150µA auf. Seine Programmierung (in Assembler) ist allerdings vergleichsweise schwierig und würde den Rahmen dieses Artikel sprengen. Wir widmen uns in einem späteren Artikel diesem Thema.

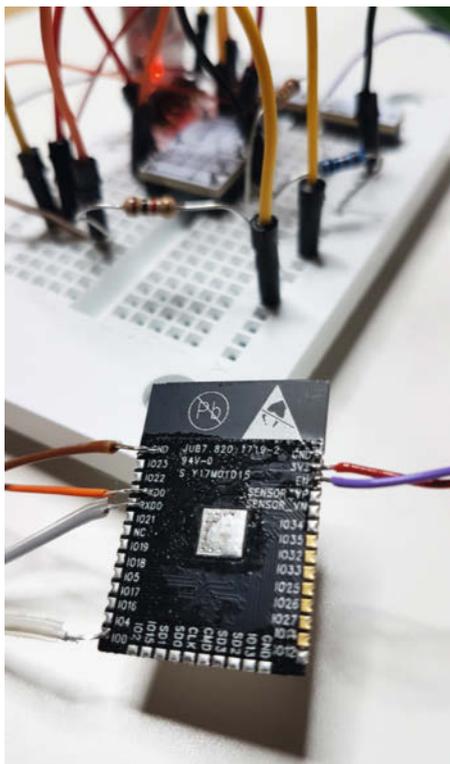
Der oben erwähnte Hibernation-Mode des ESP32 schaltet noch einen Gang runter, ist aber eigentlich kein eigenständiger Mode. Vielmehr schaltet man über Register weitere Teile der RTC ab, sodass sie externe Ereignisse nicht mehr registriert und nur noch der interne Timer als Weckruf funktioniert. Die Stromaufnahme liegt dann bei 5µA. Ein Akku mit 1000mAh könnte in diesem Mode theoretisch $1000/0,005/24 = 8300$ Tage oder fast 23 Jahre lang Energie liefern. Jeder Sendevorgang verkürzt diesen Zeitraum 3.

Wake Up Sources

Quelle	Befehl	Wirkung
Touch	<code>esp_sleep_enable_touchpad_wakeup()</code>	Aufwachen durch Touch-Pin
Timer	<code>esp_sleep_enable_timer_wakeup()</code>	Timer weckt CPU auf
ext0	<code>esp_sleep_enable_ext0_wakeup()</code>	High oder Low an einem GPIO
ext1	<code>esp_sleep_enable_ext1_wakeup()</code>	High oder Low an mehreren GPIOs
ULP	<code>esp_sleep_enable_ulp_wakeup()</code>	ULP weckt CPU auf



3 Wehe, wenn er aufgeweckt: Der Stromverbrauch eines ESP32 kurz nach dem Aufwachen aus dem Deep-Sleep. Zuerst bucht er sich in ein WLAN ein, dann sendet er Daten. Die Peaks erreichen einige 100mA.



4 Ja, man hätte das ESP-WROOM-32-Modul auch auf ein Breakoutboard löten können, um es durchzumessen. Es geht aber auch ohne.

Praxis

Soweit zur Theorie, in der Praxis wird man für Akku-betriebene IoT-Projekte immer den Deep-Sleep- oder gar den Hibernation-Mode nutzen wollen. Im Grunde reichen dafür ein oder zwei Befehle aus: eine Definition, welche Quelle den ESP aufwecken soll und ein Befehl, um ihn in den Schlaf zu schicken. Allerdings muss man bei der Struktur seiner Programme ein klein wenig umdenken.

Fangen wie mit dem ESP8266 an. Er kennt nur eine Quelle, die ihn aus dem Schlaf wach küsst: Ein Timer der internen RTC. Mit dem Befehl `ESP.deepSleep(60 * 1000000)` setzt man den Timer auf 60 Sekunden und versenkt den ESP8266 ins Koma. Da der Timer in Mikrosekunden rechnet, hilft der Umrechnungsfaktor von `1000000` dabei, den Überblick zu bewahren. Die übergebene Zahl war in früheren API-Versionen eine 32-bittige (unsigned) Integerzahl, was die maximale Schlafenszeit 4294967295 Mikrosekunden, also knapp 71 Minuten begrenzte. Die aktuelle Version arbeitet mit 64-Bit, mit der sich Äonen überbrücken ließen. Würde man jedoch als Parameter 0 übergeben, würde kein Timer gesetzt und somit auch kein Signal den Prozessor wachrütteln können.

Das Listing *ESP8266: MQTT mit Deep-Sleep* zeigt exemplarisch, wie ein Wemos D1 Mini eine MQTT-Nachricht verschickt und für 60

ESP8266: MQTT mit Deep Sleep

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

const char* SSID = "SSID";
const char* PSK = "Passwort";
const char* MQTT_BROKER = "192.168.4.1";

WiFiClient espClient;
PubSubClient client(espClient);

void startDeepSleep()
{
    ESP.deepSleep(60 * 1000000);
    yield();
}

void setup_wifi() {
    int failed = 0;
    WiFi.mode(WIFI_STA);
    delay(10);
    WiFi.begin(SSID, PSK);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        failed++;
        if (failed == 10)
            { startDeepSleep();}
    }
}

void reconnect() {
    while (!client.connected()) {
        Serial.print("Reconnecting...");
        if (!client.connect("ESP8266Client")) {
            delay(5000);
        }
    }
}

void setup() {
    Serial.begin(9600);
    setup_wifi();
    client.setServer(MQTT_BROKER, 1883);
    if (!client.connected()) {
        reconnect();
    }
    client.publish("/home/data", "Hello World");
    delay(1000);
    startDeepSleep();
}

void loop() {}
```

Sekunden einschläft. Wacht der ESP auf, baut er eine WLAN-Verbindung (`setup_wifi`) und anschließend eine Verbindung zum MQTT-Server (`client.setServer, client.connect`) auf. Dann sendet er auf einem MQTT-Topic seine Daten und ruft die Funktion `startDeepSleep()` auf. Diese ruft `ESP.deepsleep()` auf und führt noch ein `yield()` aus. Der Befehl `yield()` soll dem einkernigen ESP8266 Gelegenheit geben, zwischendurch interne Arbeiten rund ums WLAN zu erledigen und nicht nur den Sketch abzuarbeiten.

Das Programm versucht, 10-mal eine Verbindung mit dem WLAN aufzunehmen und schläft bei Nichterfolg wieder ein. Aus für uns nicht nachvollziehbaren Gründen haben Espressif-Controller manchmal nach dem Aufwachen Verbindungsprobleme, sodass sie ohne Begrenzung der Verbindungsversuche

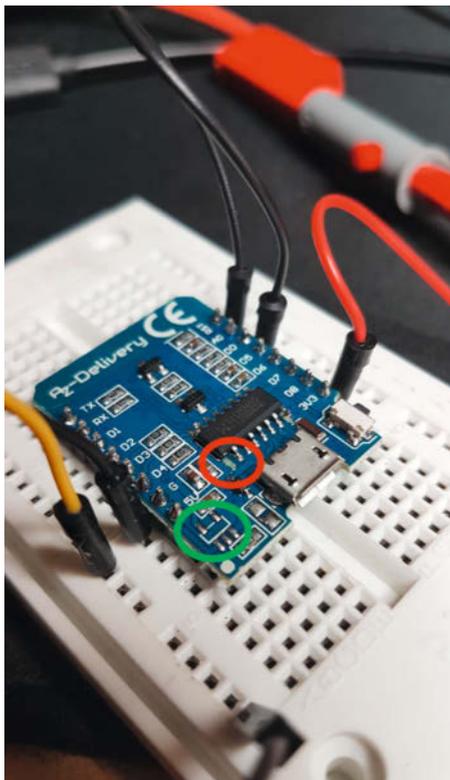
immer wieder probieren und damit den Akku schnell leer saugen würden.

ESP32

Beim ESP32 benötigen wir zwei statt einen Befehl für den Tiefschlaf. Der abgebildete Sketch für den ESP8266 muss nur in wenigen Zeilen angepasst werden:

- Statt `#include <ESP8266WiFi.h>` muss man für den ESP32 `#include <Wifi.h>` schreiben, um WLAN nutzen zu können.
- Wir müssen in `setup()` die Funktion `esp_sleep_enable_timer_wakeup(60 * 1000000)` aufrufen, um als Weckruf den Timer mit einem Intervall von 60 Sekunden festzulegen.

In der Funktion `startDeepSleep()` ändern wir den Befehl `ESP.deepSleep(60 * 1000000)` in



5 Das Wemos-Board wurde mittels Heating Plate seines Spannungsreglers beraubt (grüner Kreis), Der Ground-Pin (roter Kreis) fiel grobmotorischer Lötgewalt zum Opfer.

ESP32: Takt drosseln

Die Kerne des ESP32 laufen standardmäßig mit 240MHz, wenn sie mit der Arduino-IDE programmiert werden. Sie können aber auch auf 160 und 80MHz herunter getaktet werden. Konfiguriert man die internen Taktquellen (PLL, Quarz, RTC) um, so sind auch Frequenzen bis herunter zu 2MHz möglich.

Mit der Zeile `setCpuFrequencyMhz(80)` stellt man den Takt einfach auf 80MHz um und spart über 30mA im Vergleich zum Takt mit 240MHz. Hier lohnt es sich, fürs eigene Projekt weitere Versuche zu machen, ob die Drosselung in den Aufwachphasen in Verbindung mit Deep-Sleep Strom spart.

`esp_deep_sleep_start()`, sodass die Funktion wie im Listing *ESP32: startDeepSleep* aussieht.

Die Strommessung haben wir gleich an einem einzelnen ESP-WROOM-32-Modul 4 vorgenommen, in dem wir einen Widerstand in die Verbindung nach Masse eingebaut haben. Die Programmierung haben wir mit einem externen USB-zu-seriell-Wandler erledigt.

Solange der ESP32 wach war, haben wir den Widerstand mit einem Jumper-Kabel überbrückt. Im Tiefschlaf haben wir das Jumper-Kabel abgezogen und den Spannungsabfall am 1kOhm-Widerstand gemessen: 5,2mV. Nach dem Ohmschen Gesetz entspricht dies einem Strom von 5,2µA. Offenbar ist der ESP32 also nicht einfach nur in den Tiefschlaf gefallen, sondern in den tieferen Hibernation-Schlaf.

Die Erklärung findet sich im Datenblatt von Espressif: Standardmäßig schalten die Befehle `esp_deep_sleep_start()` und `esp_light_sleep_start()` alle Teile der RTC ab, die nicht zum Aufwachen benötigt werden. Da die RTC in unserem Fall keine GPIOs überwachen muss, legt sie diese Teile einfach schlafen.

Sensoren

5µA Stromverbrauch versprechen eine lange Laufzeit, zumindest wenn sonst nichts am ESP32 dran hängt. Kommen Sensoren hinzu, fängt das nächste Elend an: Ein aktuell oft verwendeter Sensor ist der CO₂-Sensor SCD30 von Sensirion. Der verbraucht bei einem vergleichsweise langen Messintervall von über 100 Sekunden knapp 5mA, also Größenordnungen mehr als die ESP-Mikrocontroller.

Abhilfe kann es bringen, Sensoren nur nach dem Aufwachen mit Strom zu versorgen. Indem man ihnen in die Versorgungsleitungen MOSFETs oder JFETs einbaut und diese per GPIO gezielt schaltet, verbrauchen Sensoren während der Schlafphase ebenfalls keinen Strom. Wie man das beispielsweise mit einem JFET 113 und einem Ultraschallsensor macht, zeigt unser Artikel *LoRaWAN-Sensor für den Heizöltank* (siehe Kurzinfo).

Muntzing

Ein Wemos D1 Mini zieht, wenn man das Board über seine Header-bzw. Stiftleiste mit Spannung direkt am 3,3V-Pin versorgt, im Deep-Sleep immer noch 400µA. Ursache dafür sind die anderen ICs auf dem Board, als da wären ein USB-zu-seriell-Wandler CH340 und ein Spannungsregler. Lötet man den CH340 aus, oder trennt die Beinchen der Versorgungsspannung, so sinkt der Verbrauch auf 300µA.

Auch der auf den Boards verbaute 3,3V-Spannungsregler zieht noch einiges an Strom, zumindest die Version (Aufdruck 4a2d) auf den Wemos-China-Klones. Lötet man den Regler aus, spart man weitere 200µA und man landet bei 100µA bis 140µA 5.

Die nun noch verbleibenden µA sind verbleibenden Spannungsteilern und Pull-Up- und Pull-Down-Widerständen geschuldet.

Im kurzzeitigen Sendemodus benötigt das Wemos-Board durchschnittlich 140mA.

Auf jeden Fall lohnt es sich, das jeweils verwendete Board genauer anzuschauen und gegebenenfalls den eventuell verfügbaren Schaltplan zu konsultieren. Favoriten zum



6 Mit einer Mini Heating Plate wird Muntzing zum Vergnügen.

ESP32: startDeepSleep

```
void startDeepSleep()
{
    esp_deep_sleep_start();
}
```

Stromsparen sind immer Power-On-LEDs, die signalisieren sollen, dass das Board mit Spannung versorgt wird. Die LEDs benötigen selbst in den Low-Current-Versionen noch 1mA bis 5mA.

Da die Boards in der Regel in SMD-Technologie aufgebaut sind, benötigt man mindestens einen Lötkolben mit feiner Lötspitze, um einzelne Beinchen zu entlöten und hochzubiegen. In Bild 5 war der Autor jedoch etwas zu ungeduldig, sodass eine Leiterbahn abriß und der Vorgang nur noch semi-reversibel war. Sofern die Boards nur von einer Seite und noch nicht mit Stiftleisten bestückt sind, ist das Entlöten mit einer Mini Heating Plate wie in 6 am einfachsten.

Ab einem bestimmten Punkt lohnt Muntzing aber nicht mehr, weil man eigentlich das Board leerräumt. Dann sollte man überlegen, ob man den ESP8266 nicht besser als Einzelmodul ESP-12 erwirbt und sparsam beschaltet. Breakoutboards 7 gibt es für verschiedene Module und Ausführungen. Gleiches gilt für den ESP32. Zum Programmieren reicht ein günstiger externer USB-zu-Seriell-Wandler.

Power

Zur direkten Stromversorgung haben wir einen LiPo-Akku direkt an GND und 3,3V angeschlossen. Laut Hersteller verträgt der ESP8266 maximal 3,6V und wird mit den maximal 4,2V des Akkus außerhalb der Spezifikation betrieben. In unseren Tests beobachteten

Weitere Tipps zum Sparen

Zeit ist Geld oder Akku-Ladung: Viele unnötige Befehle oder Routinen in Mikrocontrollercodes verlängern die Wachphase unnötig, in der ein ESP viel Strom frisst. Hier ein paar Tipps zum Energiesparen in Software:

- Statische IP-Adressen statt DHCP verwenden
- Statt Hostnamen für Server besser IP-Adressen nutzen
- Timeouts bei Serververbindungen einbauen
- Wenig delays() verwenden
- Ausgaben über serielle Schnittstelle vermeiden
- Sensordaten sammeln und im Block per WLAN verschicken

wir keine Probleme, was nicht heißt, dass keine auftreten könnten. Wer will, kann auch eine Diode in die Versorgungsleitung schalten, um die Spannung zu reduzieren. Allerdings unterschreitet die Akku-Spannung dann früher die für den Prozessor erforderliche Spannung.

Sollten Sie auf die Idee kommen, Akkus in Kombination mit den beliebten Laderegler TP4056 zur Versorgung zu verwenden: Der IC hat im Leerlauf eine typische Stromaufnahme von 5µA.

Haben Sie weitere Tipps zum Stromsparen? Auf welche Laufdauer kommen Sie bei Ihren Projekten? Schreiben Sie uns: info@make-magazin.de —dab

Aufrüsten im Homeoffice



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.

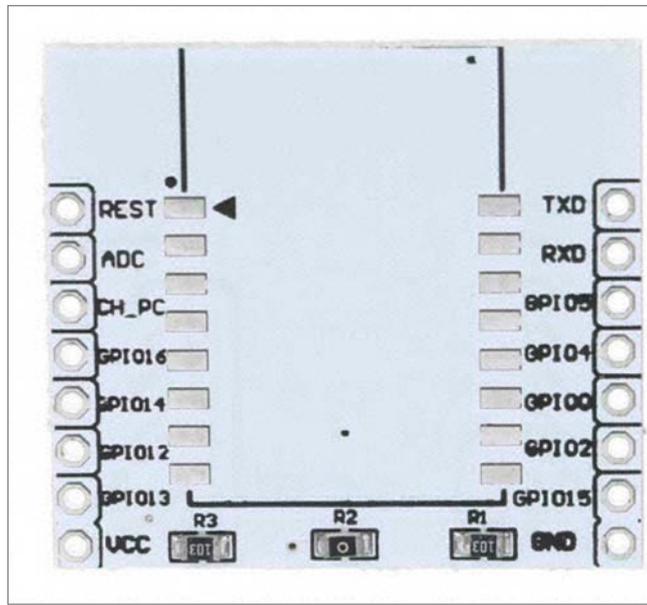
Auch als einzelnes Heft erhältlich. shop.heise.de/ct-hardware-ho21

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.



shop.heise.de/ct-hardware-ho21



7 Ein Breakoutboard zum Bestücken mit einem einzelnen ESP-12-Modul (ESP8266). Damit kann man die notwendigen Elemente selbst nachrüsten.

TV und Heimkino mit Home Assistant, Teil 1

Sprich „Alexa, Kino einschalten!“ und wie von Geisterhand schließen sich die Rollos, während die Lichter allmählich erstrahlen und das TV-Gerät oder der Beamer sowie die Soundanlage aktiviert werden. Schön, nicht wahr? Diese Folge der Smarthome-Reihe bringt Sie diesem Traum ein erstes Stück näher. Wir bringen nämlich zunächst Home Assistant die Steuerbefehle Ihrer Unterhaltungs-Elektronik bei. Im nächsten Teil geht's dann ums Verdunkeln und Sprachbefehle.

von Heinz Behling



Couch-Potatoes, aber auch andere, auf-gepasst: In dieser Folge der Smarthome-Reihe geht es um Unterhaltungselektronik, insbesondere Fernsehen und Heimkino. Wir werden unseren Home Assistant in die Lage versetzen, über Infrarot die gängigen Geräte zu steuern. Die dafür nötige Hardware bauen wir selbst.

Für Lampen und Leuchten werden wir weitgehend preiswerte ZigBee-Steckdosen und -Birnen verwenden. Sie erfahren außerdem, wie Sie diese ohne die teuren Hersteller-Gateways und deren Cloud-Zugänge steuern.

All das können Sie nach diesem Artikel dann schon per Smartphone fernbedienen, Sie müssen ja nur die Home-Assistant-IP-Adresse im Browser des Telefons eingeben und loslegen. Die Anstrengungen dieser Folge dienen aber vor allem als Vorbereitung für den nächsten Teil. Da lernt der Home Assistant nämlich, Ihnen aufs Wort zu gehorchen und Ihr Kino ins richtige Ambiente zu versetzen.

Alexa wird dann nicht nur Ihr TV-Gerät oder Ihren Beamer und die Soundanlage nebst Subwoofer starten, sondern auch langsam das Licht erstrahlen lassen, während ein motorgesteuertes Rollo das Sonnenlicht aussperrt. Ich verrate Ihnen, wie man mehrere schmale, preiswerte Rollos zu einem überbreiten kombiniert. Und nebenbei wird Alexa auch alle anderen Bestandteile des smarten Heimes kontrollieren können.

Kurzinfo

- » **Unterhaltungselektronik via Infrarot mit Home Assistant steuern**
- » **Anpassen der Web-Oberfläche**
- » **Schalten von ZigBee-Lampen und -Steckdosen**

Checkliste



Zeitaufwand:
ab 2 Stunden



Kosten:
ab 50 Euro

Werkzeug

- » Lötkolben
- » Bohrmaschine
- » eventuell 3D-Drucker für Gehäusebau

Mehr zum Thema

- » Heinz Behling, Intelligentes Heim mit Home Assistant, Make 1/21, S. 100
- » Heinz Behling, Heizung unter Kontrolle, Make 2/21, S. 22
- » Heinz Behling, Prime und Netflix mit dem Raspberry Pi 4, Make 3/20, S. 60

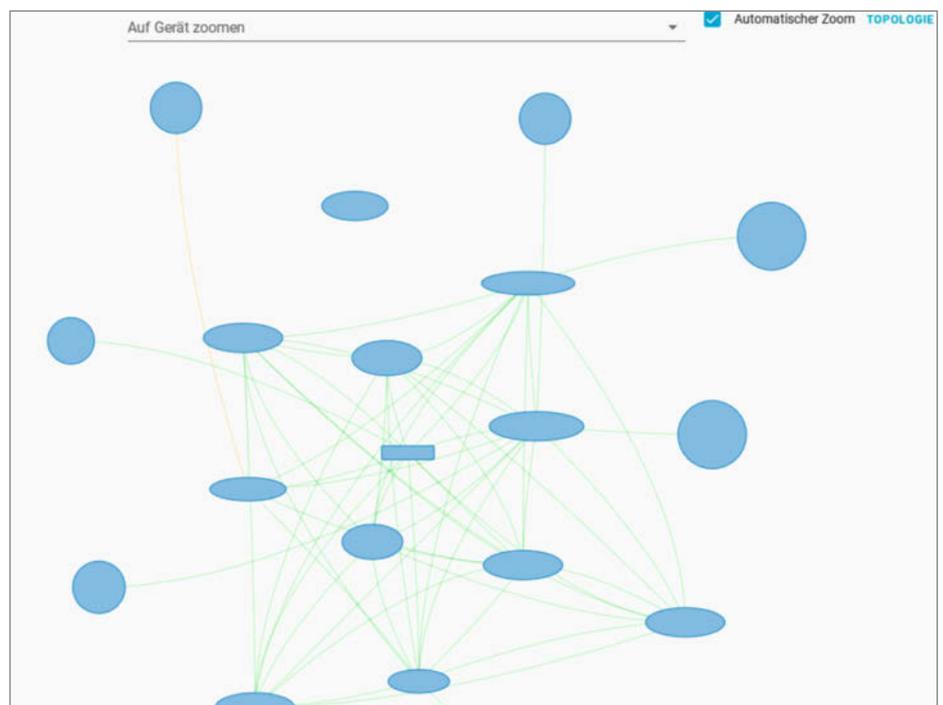
Alles zum Artikel i
m Web unter
make-magazin.de/x5wb

Schalten mit ZigBee

Die meisten Unterhaltungselektronik-Komponenten werden via Infrarot-Fernbedienung ein- und ausgeschaltet. Es gibt aber auch etliche Geräte im Heimkino, zum Beispiel Soundanlage, aktive Subwoofer oder Lampen, die besser mittels Schaltsteckdosen ein- und ausgeschaltet werden. Solche Steckdosen oder auch Leuchtmittel (gerne LED-Birnen genannt) gibt es von vielen Herstellern. Ich möchte da aber nicht nur Komfort, sondern achte auch immer auf die Kosten. Am preiswertesten sind da die *ZigBee*-Komponenten, die es von zahlreichen Herstellern gibt. Darunter sind namhafte Marken wie *Philips* mit seinem *Hue*-System, aber auch Discounter, etwa *Aldi* und *Lidl*, benutzen ZigBee. Schließlich ist da auch noch *Trådfri* von *Ikea* zu nennen.

Alle diese Anbieter haben auch sogenannte Gateways in ihrem Programm und behaupten gerne, dass man unbedingt eines (am liebsten das ihre) zur Steuerung ihrer ZigBee-Geräte braucht. Zusätzlich kommt oft noch eine Steuerung ausschließlich über die Hersteller-Cloud ins Spiel.

Ich kann Sie da beruhigen: Sie brauchen keines der Anbieter-Gateways, sondern lediglich einen ZigBee-USB-Stick. Der kommt auch



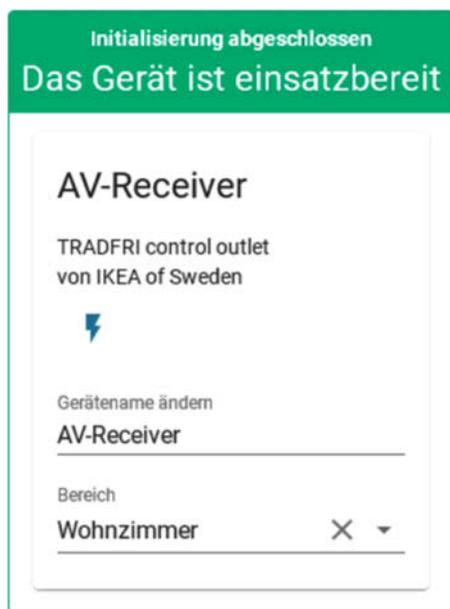
1 Die ZigBee-Integration kann Ihnen die Mesh-Struktur anzeigen. Mit dem Mausekran können Sie in der Grafik zoomen und die Daten einzelner Geräte anzeigen lassen.



2 Wichtig ist die Wahl der Schnittstelle des USB-Sticks.



3 Hinter diesem Loch sitzt der Taster für den Pairing-Modus.



4 Verwenden Sie nur eindeutige Gerätenamen, sonst kommen Sie später irgendwann in Teufels Küche.

ohne jede Cloud aus. Lassen Sie sich auch nicht davon beunruhigen, dass in den technischen Daten der Sticks die Anzahl der zu steuernden Geräte relativ gering ist. Meist sind es nur 20. Allerdings arbeitet das Netzwerk ZigBee (nichts anderes ist es) mit einer Mesh-Struktur 1.

Das bedeutet, dass der steuernde Server (das ist der USB-Stick) nicht mit allen anderen Geräten direkt verbunden sein muss. Jedes ZigBee-Gerät kann als Zwischenstation arbeiten und sich mit anderen Komponenten verbinden und die Befehle des Servers weiterleiten. Das ergibt zwei Hauptvorteile: Zum einen lassen sich so beliebig große Netzwerke aufbauen, solange mit jedem einzelnen Gerät nicht mehr als 20 weitere verbunden sind. Zum anderen erreicht man so trotz geringer Funkleistung recht große Reichweiten. Beides hat mich auch davon überzeugt, nicht die noch preiswerteren, aber in der Reichweite und Funksicherheit begrenzten 433MHz-Funksteckdosen zu verwenden.

ZigBee-USB-Sticks gibt es mehrere. Ich habe den mit etwa 40 Euro etwas teureren Conbee II verwendet. Daneben gibt es auch

zahlreiche Angebote unter der Bezeichnung CC2531 ZigBee-Stick für weniger als 20 Euro. Allerdings enthalten die oft nur den ZigBee-Controller-Chip, aber keinerlei Firmware. Die ist zwar frei im Internet erhältlich, muss aber auf dem Stick installiert werden, wozu ein Programmierer erforderlich ist, der den Preisvorteil dann ausgleicht. Falls Sie sich doch für diese Stick-Version entscheiden, achten Sie darauf, dass der Stick auch die entsprechende Firmware enthält.

Kaufen Sie im Übrigen auch gleich ein mindestens 1m langes USB-Verlängerungskabel mit. ZigBee-Geräte reagieren nämlich allergisch darauf, wenn sie direkt neben einem USB3-Port betrieben werden (vermutlich gibt es da Störeinstrahlungen). Deshalb braucht der Stick Abstand vom RasPi, dessen USB3- und USB2-Buchsen ja direkt nebeneinander liegen. Schließen Sie den USB-Stick über das Kabel vor der Software-Installation an den Raspberry Pi an und starten Sie ihn neu mit Supervisor/System/Core neu starten.

Integration installieren

Softwaremäßig braucht der Home Assistant eine Integration für ZigBee. Die installiert man unter *Einstellungen/Integrationen/Integration hinzufügen*. Das Ziel unserer Träume heißt *Zigbee Home Automation*. Nach einem Klick darauf werden Sie aufgefordert, den Pfad zum Stick anzugeben. Nach einem Klick auf den kleinen Pfeil wird Ihnen die Schnittstelle angezeigt. Wählen Sie die *Conbee-II*-Schnittstelle und klicken Sie auf *absenden* 2.

Kurze Zeit später erscheint die *Success*-Meldung. Dort sollten Sie der Ordnung halber dem Stick einen Bereich zuordnen. Ich habe *Meine Wohnung* gewählt. Nach einem Klick auf *Fertig* ist der Home Assistant schon ZigBee-fähig.

Damit man ein ZigBee-fähiges Gerät steuern kann, müssen Home Assistant und das jeweilige Gerät miteinander gepaart werden. Beim Home Assistant geschieht dies mit *Einstellungen/Integrationen/zigbee konfigurieren/+ Gerät hinzufügen*. Home Assistant bleibt dann für einige Minuten im *Pairing*-Modus.



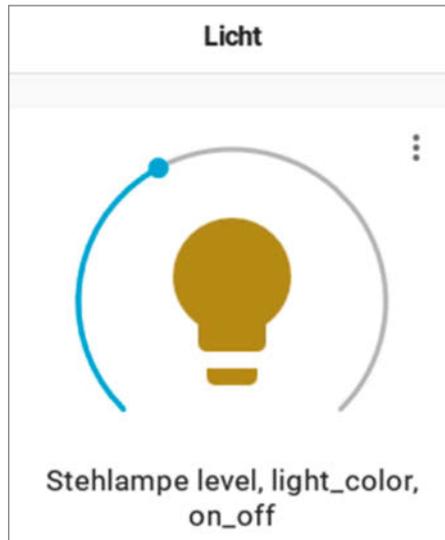
5 Die Einstellungen für den ersten Schalter

Die Vorgehensweise an den Geräten ist leider bei jedem anders: Ikea-Trådfri-Steckdosen haben dazu ein kleines Loch an der Seite, durch das man mit einem Stift einen Taster für einige Sekunden drücken muss **3**.

Die Schaltsteckdose muss dazu unter Strom stehen, also selbst in einer Steckdose sitzen. Trådfri-Lampen müssen zum Pairing zunächst eingeschaltet werden und danach sechsmal kurz hintereinander aus- und wieder eingeschaltet werden. In der Regel ist in der Bedienungsanleitung der Geräte angegeben, wie man in den Pairing-Modus gelangt. Falls dort nichts angegeben ist, wird der Modus oft beim ersten Einschalten des Geräts automatisch gestartet.

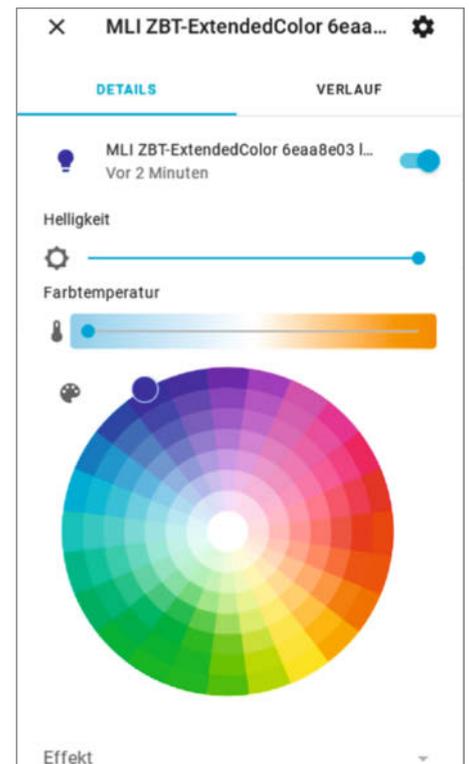
Am Beispiel der Trådfri-Schaltsteckdose für den AV-Receiver machen wir das hier einmal durch: Also Home Assistant und Steckdose in den Pairing-Modus bringen. Kurze Zeit später zeigt Ihnen Home Assistant, dass ein neues Gerät gefunden wurde und fordert Sie auf, dem Gerät einen Namen sowie einen Bereich zu geben **4**.

Der Bereich ist klar: Wohnzimmer. Beim Namen steht als Vorgabe die Gerätebezeichnung. Spätestens beim Hinzufügen einer weiteren Trådfri-Steckdose werden die durch-



6 Mit der Licht-Karte kontrollieren Sie RGB-Leuchten am besten.

nummeriert. Das wird schnell unübersichtlich. Sie sollte daher einen aussagekräftigen Namen erhalten, zum Beispiel *AV-Receiver*. Achten Sie bei der Namensvergabe peinlich darauf, keinen Namen mehrmals zu ver-



7 Hier haben Sie RGB-Leuchten voll im Griff.

Make:markt

ELEKTRONIK / ZUBEHÖR



Code Mercenaries
Hard- und Software GmbH

USB-Controller für: Joystick, Tastatur, Maus, Drehgeber, universeller I/O
Module, Beschleunigungs-/Drehratensensoren, Interface für I2C, SPI im Dongle-Format, I/O und 12 Bit ADC am USB
Komponenten für intelligentes Licht: DMX, IEC62386
Ständig Angebote im Webstore:
www.codemerchs.com/shop

LED-LÖSUNGEN



LED-STUDIEN
professionelle LED-Beleuchtung

LED-Studien.de ist der Spezialist für hochwertige LED-Streifen und Controller.
NEONFLEX, COB-LED, TUNABLE WHITE, RGBW sowie DMX und PIXEL-LÖSUNGEN – auch für große Projekte / komplette Raumbeleuchtungen für Privat und Gewerbe. Wir finden die beste Lösung für Sie!
www.led-studien.de | shop.led-studien.de

BÜCHER / ZEITSCHRIFTEN



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

www.dpunkt.de

MIKROCONTROLLER



christiani.de/arduino-education

Arduino® Education – bietet ein durchgängiges Lernkonzept mit Soft-, Hardware und Support
Leichter Einstieg in die Elektronik und Programmierung mit spannenden Projekten

- Inkl. Online Lernplattform mit verschiedenen didaktischen Unterrichtseinheiten
- Vermittlung der 21st Century Skills



seit 1931

wenden. Das kann (ich spreche aus Erfahrung) später zu einem kaum zu entwirrenden Verwechslungsspiel beim Home Assistant führen. Spätestens, wenn in der nächsten Folge ein Sprachassistent wie Alexa aktiviert wird, würden Sie die mehrfache Namensvergabe verfluchen. Wiederholen Sie das nun für alle Schaltsteckdosen und auch für die Lampen, die künftig per ZigBee gesteuert werden sollen.

Tipp: Es kann übrigens nicht schaden, die ZigBee-Geräte nach dem Pairing mit kleinen Aufklebern zu versehen, die den Gerätenamen und den Bereich enthalten. Andernfalls werden Sie irgendwann (Umzug, Renovierung) feststellen, dass sich die Steckdosen und LED-Birnen alle unheimlich ähnlich sehen...

Oberfläche anpassen

Nun brauchen Sie in der Home-Assistant-Oberfläche noch ein Bedienelement, mit dem Sie das Gerät auch schalten können. Gehen Sie also auf *Übersicht/Wohnzimmer*. Klicken Sie auf die drei Punkte oben rechts und dann auf *Benutzeroberfläche konfigurieren/Karte hinzufügen*. Wählen Sie dann Elemente und stellen Sie die Elemente-Karte entsprechend Ihrer zuvor gemachten Angaben ein **5**.

Falls Sie mehr als eine Steckdose benutzen, können Sie im Feld *Entität* die nächste auswählen und so alle Schalter in einer Karte zusammenfassen. Zum Schluss klicken Sie auf *Speichern*.

Lampen, insbesondere solche mit Farbwechsel (RGB-LEDs) erfordern eine andere Karte, um alles zu steuern. Dazu wählen Sie die Karte *Licht* aus **6**.

Auch danach bitte *Speichern*. Die Karte erscheint in der Home Assistant-Oberfläche. Wenn Sie auf die drei Punkte oben rechts in der Karte klicken, erscheint das Einstellfenster für die Farben **7**.

Wenn Sie nun alle Leuchten und per Steckdose schaltbaren Geräte in Ihrem Heimkino-Raum (sprich Wohnzimmer) auf diese Weise eingerichtet haben, wird es Zeit, sich den per Infrarot bedienbaren Geräten zu widmen, also insbesondere TV-Gerät und Beamer.

Infrarot-Geräte fernsteuern

Als erstes müssen Sie die Hardware zusammenbauen, die Home Assistant zur Steuerung von Infrarot-Geräten braucht. Der entsprechende Schaltplan zeigt, dass das nicht sonderlich schwierig ist **8**.

Anschließend muss die NodeMCU programmiert werden. Beim ersten Programmieren mit Home Assistant muss sie dazu per USB-Kabel direkt an den Raspberry angeschlossen werden. Später kann dann die Software *over the air (OTA)* geändert werden. Das werden wir auch noch benötigen, denn wir müssen mithilfe des IR-Empfängermoduls an der NodeMCU erst die Fernsteuer-codes der Geräte herausfinden.

Ist das Modul angeschlossen, klicken Sie auf *ESPHome* und +. Als *Node Name* geben Sie zum Beispiel *ir_remote* ein (nur Kleinbuchstaben sind zulässig). Nach *Next* wählen Sie den Boardtyp, in diesem Beispiel *NodeMCU*. Noch ein

Next später geben Sie dann den Namen Ihres WLANs (*WiFi SSID*) und das Passwort ein. Jetzt folgen noch Klicks auf *Next/Submit*, und schon erscheint in *ESPHome* das neue Gerät **9**.

Ein paar Befehle braucht die Software aber noch. Nach einem Klick auf *Edit* geben Sie ans Ende des angezeigten Programms noch die Befehle aus Listing **10** ein.

Speichern mit *Save* nicht vergessen. **Tipp:** Schließen Sie das Fenster mit *Close* und wählen Sie dann *Validate*. Dann überprüft Home Assistant den Programmcode und meldet eventuelle Fehler!

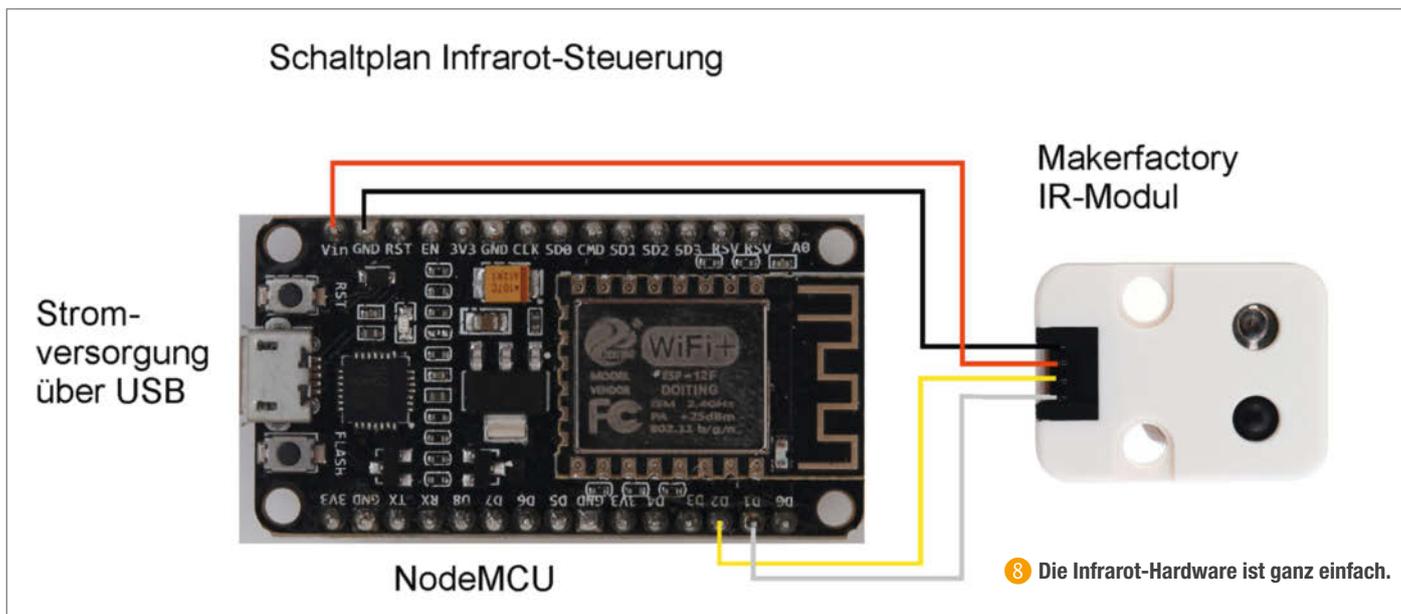
Vor dem Upload auf die NodeMCU klicken Sie auf den Pfeil neben **11**.

Wählen Sie den richtigen USB-Anschluss für die *USB to UART Bridge* **12**.

Jetzt endlich kann per Klick auf *Upload* die Software zum künftigen *ir_remote*-Modul übertragen werden. Das Programm wird kompiliert

und nach etwa einer Minute in der NodeMCU gespeichert. Kurze Zeit später wird auf die Anzeige der Log-Meldungen des Moduls umgeschaltet. Kontrollieren Sie, ob der Empfang des WLAN-Signales angezeigt wird (falls nicht, haben Sie sich zuvor bei der Eingabe des WLAN-Namens oder des Passwortes vertippt).

Doch noch ist das Modul nicht ganz einsatzfähig. Home Assistant übernimmt nämlich Daten von neuen Modulen erst, wenn Sie es auch erlauben. Betrachten Sie einmal das Feld *Benachrichtigungen*. Daneben erscheint eine **1** auf orangem Grund. Ein Klick auf *Benachrichtigungen* nennt auch den Grund: Ein neues Gerät (*device*) wurde gefunden. Per Klick auf *Check it out* gelangen Sie zur Integrationsliste. Die Integration, die von der neuen Entdeckung betroffen ist, besitzt den entsprechenden blauen Titelbalken, hier ist es natürlich *ESPHome* **13**.



Klicken Sie auf *Konfigurieren/Absenden* und wählen Sie den passenden Bereich, also im Beispiel wieder *Wohnzimmer*. Die Bestätigung mit *Fertig* nicht vergessen.

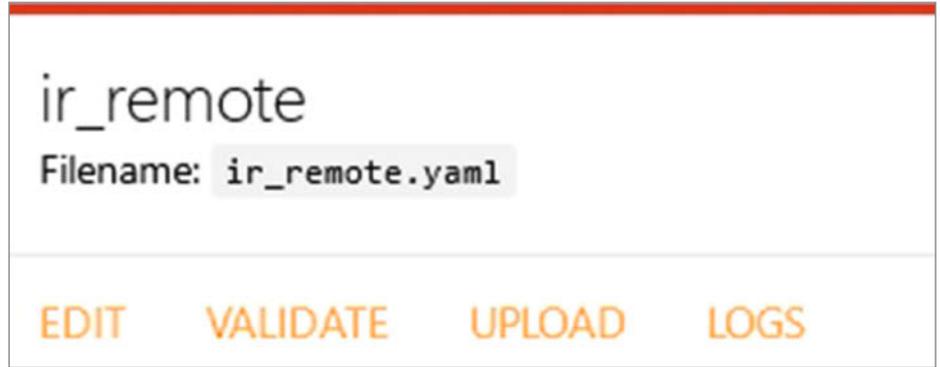
Fernbedienungs-Codes ermitteln

Jetzt geht es zurück zu *ESPHome*. Lassen Sie sich mit einem Klick auf *Logs* die Meldungen des *ir_remote*-Moduls anzeigen. Nun brauchen Sie die Fernbedienungen Ihrer Geräte. Im Folgenden zeige ich Ihnen zwei Beispiele zur Code-Ermittlung. Zunächst der Ein-/Aus-schaltcode meines Beamers: Die Fernbedie-nung ungefähr in Richtung des IR-Moduls halten und die entsprechende Einschalttaste drücken. Auf dem Bildschirm erscheint dann der Code ¹⁴.

Sie können die Code-Zeilen im Log-Fenster mit der Maus markieren und dann per *Kopieren* und *Einfügen* zum Beispiel in einen Texteditor übertragen. So können Sie zuerst alle Codes



¹¹ Hier müssen Sie von OTA auf den USB-Anschluss wechseln.



⁹ Der neue Eintrag in ESPHome

¹⁰ Befehle für ir_remote

```
remote_receiver:
  pin:
    number: D1
    inverted: True
  dump: all

remote_transmitter:
  pin: D2
  carrier_duty_percent: 50
```



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



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

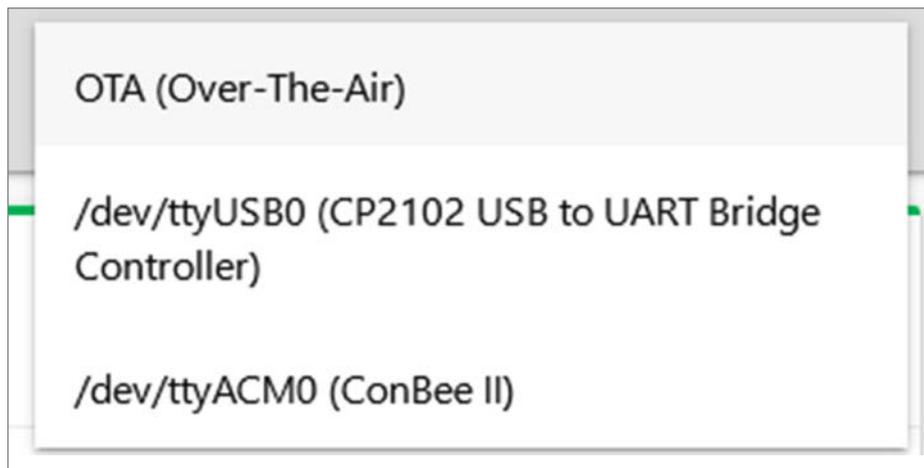


ab 27,99 € · Print | E-Book | Bundle
ISBN 978-3-86490-800-2



ab 27,99 € · Print | E-Book | Bundle
ISBN 978-3-86490-567-4





12 Hier nicht den ZigBee-Stick wählen!

bestimmen, bevor Sie sie in den Programmcode des IR-Moduls übertragen. Notieren Sie sich aber jeweils Gerätenamen und Funktionen dazu! Sendet die Fernbedienung mehrere Codes, reicht es, eine Zeile zu notieren.

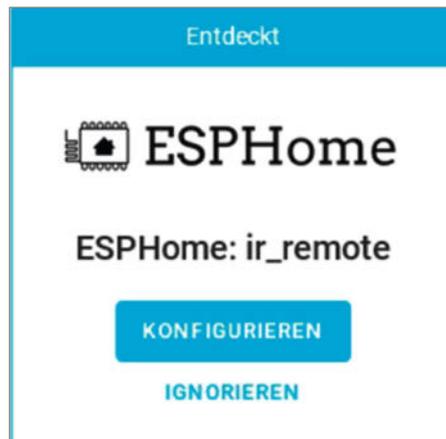
Ermitteln Sie auf diese Weise die Codes aller Funktionen, die Sie über den Home Assistant steuern möchten. **Tipp:** Sinnvoll sind nur die Funktionen, die man später zur Automatisierung des Heimkinos braucht (also zum Beispiel zum gleichzeitigen Einschalten aller benötigten Geräte). Ermitteln Sie daher nur die Codes für Ein-/Ausschaltfunktionen sowie die Wahl der Signaleingänge. Die Programmumschaltung hingegen ist mit der bisherigen Fernbedienung einfacher. Keine Angst, das kann man später immer noch erweitern.

Bei einigen Geräten wird kein Hersteller erkannt, sondern nur der RAW-Code, also die empfangenen Signalpegel angezeigt. Aber auch solche Geräte lassen sich mit Home Assistant steuern.

Kopieren Sie dann die komplette RAW-Code-Zeile. Das war zum Beispiel bei meinem *Medion-TV* der Fall. Es ist übrigens normal, wenn die Codes von Zeile zu Zeile unterschiedlich sind. Sie brauchen trotzdem nur eine der Zeilen 15.

Codes in Programm eintragen

Wenn Ihre Liste komplett ist, können Sie die Software für *ir_remote* erweitern. Auf der Seite https://esphome.io/components/remote_transmitter.html (bitte in einem anderen Tab als den Home Assistant anzeigen lassen) finden Sie wichtige Informationen dazu, insbesondere den Aufbau der herstellerspezifischen Programmbefehle. Ich zeige Ihnen das jetzt wieder an den schon bekannten zwei Geräten. Zunächst der Beamer Marke *Benq*, der ja mehrere Hersteller nannte. Hier habe ich die *LG*-Zeile verwendet. Scrollen Sie auf der



13 ESPHome verlangt, das neue Gerät zu konfigurieren.

genannten Seite runter, bis Sie den Transmit-Befehl des jeweiligen Herstellers finden, im Beispiel also *LG* 16.

Gehen Sie in Home Assistant nun nach *ESPHome* und klicken Sie unter *ir_remote* auf *Edit*. Unter dem letzten schon vorhandenen Befehl setzen Sie die Zeile *switch::*. Das Wort muss ganz am linken Rand beginnen.

Danach folgen dann jeweils Befehlsblöcke für jede einzelne Infrarot-Funktion. Für den Beamer sieht das dann so aus wie in Bild 17.

Als *transmit*-Befehl muss der dem Hersteller entsprechende Befehl verwendet werden. Die Zeilen darunter müssen die zuvor ermittelten und in Ihrer Liste notierten Werte enthalten. Die Zeile *turn_on_action* legt den Einschaltbefehl fest. Bei den meisten infrarot-gesteuerten Geräten ist er gleich dem Ausschaltbefehl. Benutzt Ihrer Hardware unterschiedliche Befehle, müssen Sie noch einen entsprechenden mit *turn_off_action* beginnenden Block hinzufügen.

Achten Sie auf die Bindestriche und die Einrückungen. Sie sind wichtig und müssen genauso verwendet werden.

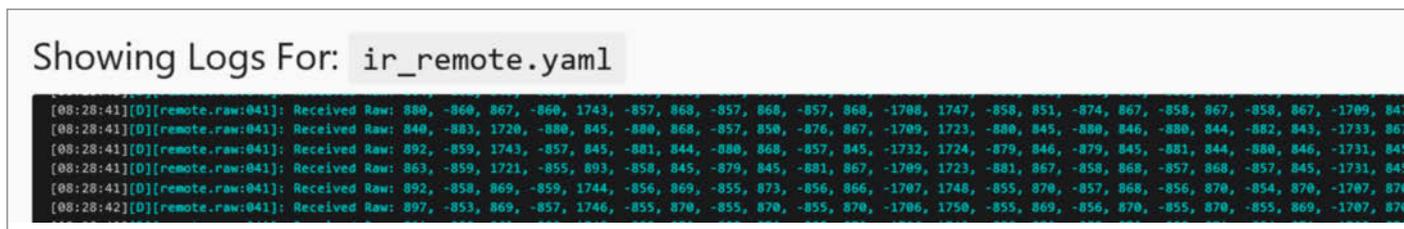
Im zweiten Beispiel mit dem Raw-Code sieht der Befehl nach der *turn_on_action*-Zeile so aus wie in Bild 18.

Setzen Sie nun auf die gleiche Weise die restlichen Befehle ein. Zum Schluss könnte das so aussehen 19.

Anschließend übertragen Sie die Software per *Upload*. Die NodeMCU muss dazu nicht



14 Bei manchen Geräten erscheinen mehrere Code für unterschiedliche Hersteller. Hier können Sie sich für eine Zeile entscheiden.



15 Kann der Gerätehersteller nicht ermittelt werden, erscheint der RAW-Code.

```
remote_transmitter.transmit_lg Action

This action sends an LG infrared remote code to a remote transmitter.

on_...:
  - remote_transmitter.transmit_lg:
      data: 0x20DF10EF # power on/off
      nbits: 32
```

16 Das ist der Befehl für LG-Geräte. Die Werte hinter *data* und *nbits* müssen Ihren Geräten entsprechend angepasst werden.

mehr am Raspi angeschlossen sein. Allerdings braucht sie natürlich Strom. Ein einfaches USB-Netzteil mit 1000mA reicht dazu aus. Platzieren Sie das IR-Modul so, dass es in Sichtweite Ihrer Geräte liegt und die LEDs zum TV-Gerät etc. weisen. Da muss eventuell etwas experimentiert werden. In extremen Fällen, wenn die zu steuernden Geräte sehr weit auseinander stehen und es keinen Punkt für den Infrarot-Sender gibt, der alle Geräte erreichen kann, muss man mit mehreren NodeMCUs nebst IR-Modulen an unterschiedlichen Aufstellungsorten arbeiten. Dabei ist darauf zu achten, dass Sie einen anderen Namen als *ir_remote* verwenden und keinen Steuercode in beiden Modulen verwenden.

Gehäuseentwürfe zum 3D-Druck für das IR-Modul finden Sie über den Kurzinfo-Link.

Jetzt erweitern Sie wieder die Oberfläche. Klicken Sie also wieder auf *Übersicht/Wohnzimmer* und ergänzen Sie die Karte, in der bereits der AV-Receiver steht, um die neuen Schalter (*switches*). Das könnte dann so aussehen wie in Bild 20.

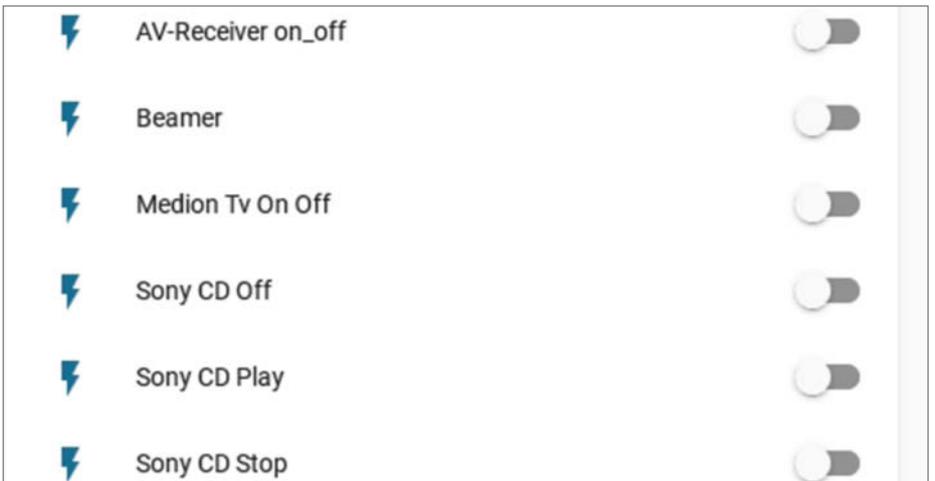
Geschaltet wird mit einem Mausklick (oder Touch am Smartphone) auf die Schaltersymbole. Bei Infrarot-Geräten können Sie allerdings an der Schalterstellung nicht erkennen, ob es ein- oder ausgeschaltet ist, da sie keine Rückmeldung liefern.

So, damit haben wir jetzt einen großen Schritt in Richtung Kino-Automatik getan. Im nächsten Teil verrate ich Ihnen, wie Sie motorische betriebene Verdunklungsrollos installieren und Ihr Heimkino dann mit nur einem Klick oder Sprachbefehl verdunkeln, Sound, Beamer oder TV einschalten sowie gegebenenfalls einen Mediaplayer wie Kodi aktivieren. Um das Kinogefühl so richtig vollständig zu machen, gibt es dann auch noch eine per Home Assistant gesteuerte Popcorn-Maschine. Aber bis dahin haben Sie ja schon etwas zu tun. —hgb

```
Editing: ir_remote.yaml
27   number: 01
28   inverted: True
29   dump: all
30
31 - remote_transmitter:
32   pin: D2
33   carrier_duty_percent: 50
34
35 - switch:
36   - platform: template
37     name: "Sony CD Off"
38     turn_on_action:
39       remote_transmitter.transmit_sony:
40         data: 0x0000A91
41         nbits: 12
42
43   - platform: template
44     name: "Sony CD Play"
45     turn_on_action:
46       remote_transmitter.transmit_sony:
47         data: 0x00004D1
48         nbits: 12
49
50   - platform: template
51     name: "Sony CD Stop"
52     turn_on_action:
53       remote_transmitter.transmit_sony:
54         data: 0x0000101
55         nbits: 12
56
57   - platform: template
58     name: "Beamer"
59     turn_on_action:
60       remote_transmitter.transmit_lg:
61         data: 0x000C40BF
62         nbits: 32
63
64   - platform: template
65     name: "Medion Tv On Off"
66     turn_on_action:
67       remote_transmitter.transmit_raw:
68         code: [854, -866, 1726, -864, 852, -864, 853, -892, 826, -890, 827, -1711, 1729, -835, 882, -1709, 855, -867, 1725, -863, 853]
69         repeat: 10
70         carrier_frequency: 39000Hz
```

17 Die Zeile *switch:* darf nur einmal vorkommen.

19 Achten Sie ganz genau auf Einrückungen!



20 Die Steuerzentrale für die Unterhaltungselektronik

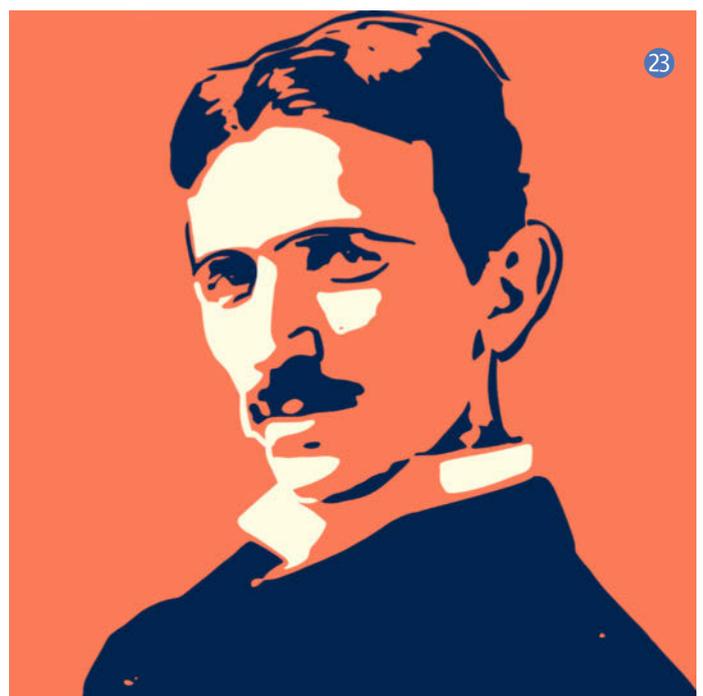
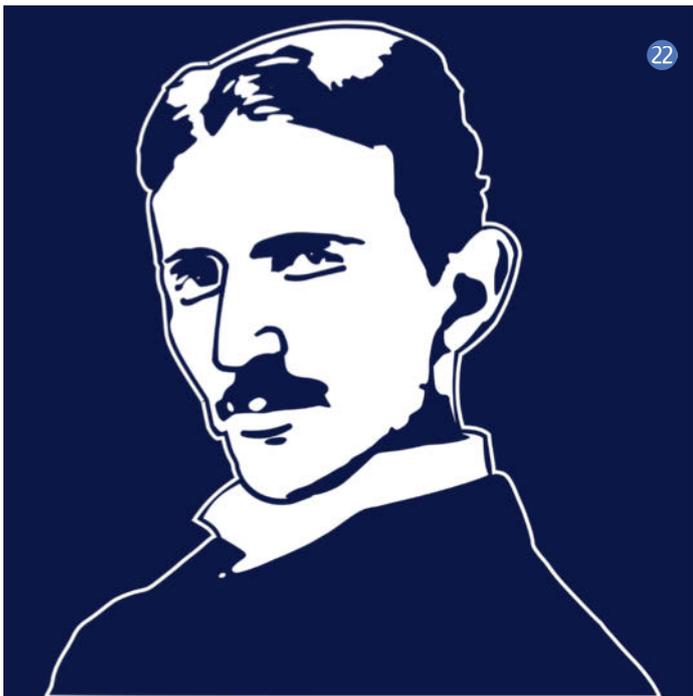
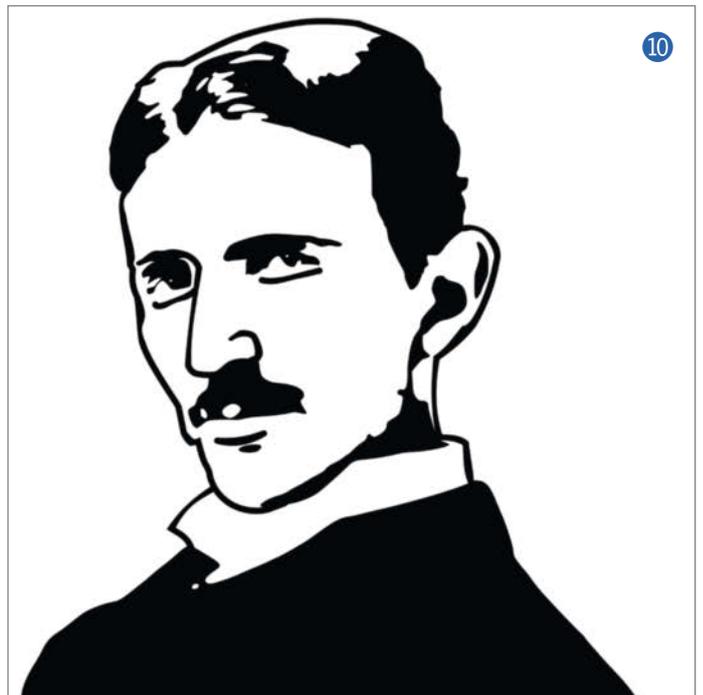
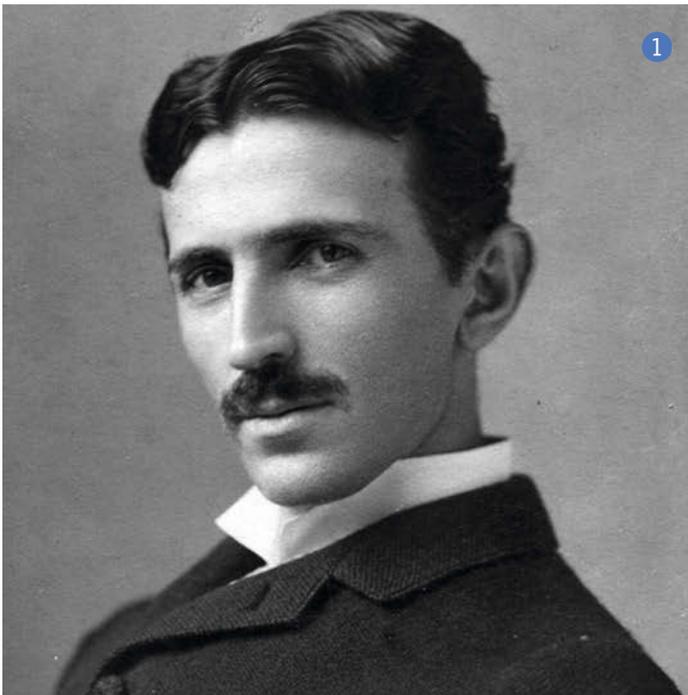
```
remote_transmitter.transmit_raw:
code: [854, -866, 1726, -864, 852, -864, 853, -892, 826, -890, 827, -1711, 1729, -835, 882, -1709, 855, -867, 1725, -863, 853]
repeat: 10
carrier_frequency: 39000Hz
```

18 So sieht ein RAW-Befehl aus.

Grafik-Tricks für Maker

Nach den Vektorgrafik-Grundlagen im vergangenen Heft geht es diesmal ans Eingemachte: Wir machen aus Fotovorlagen echte Vektorkunstwerke für selbstgemachte T-Shirts, Aufkleber oder Sandstrahlschablonen.

von Peter König



In der vorigen Make-Ausgabe haben wir ab Seite 76 Schritt für Schritt gezeigt, wie man Logos und Comics perfekt für Schneidplotter, Lasercutter & Co. vektorisieren und vereinfachen kann – und dabei keine Grundkenntnisse vorausgesetzt. Dabei sind noch ein paar Fragen offen geblieben, die jetzt auf der Agenda stehen: Zum Beispiel, wie man ein Foto geschickt auf schwarze und weiße Flächen mit harten Kanten reduziert, sodass man es einerseits prima aus Transferfolie fürs T-Shirt schneiden kann, das Motiv andererseits erkennbar ist und schick aussieht.

Als Beispiel muss im Folgenden Herr Tesla herhalten – bekannt als Ikone des technischen Exzentrizismus' (und damit so eine Art Maker-Schutzpatron), von selbstgemachten T-Shirts unseres Chefredakteurs sowie von *Brchiographien* (siehe Make 1/20, S.8). Es gibt nur relativ wenige Fotos von Nikola Tesla, eines der bekanntesten stammt von etwa 1890 und das gibt es unter *Public Domain* im Netz 1.

Wie im Artikel in der vorigen Ausgabe beschrieben kann man sich so eine Fotovorlage per *Drag & Drop* auf die Zeichenfläche der kostenlosen Vektorgrafik-Software *Inkscape* laden (falls Sie Inkscape noch nicht installiert haben: siehe Link in der Kurzinfor). Über *Pfad/Bitmap nachzeichnen ...* öffnet man dann das Dialogfenster des eingebauten Vektorisierers in Inkscape. Jetzt kann man hier verschiedene Helligkeitsschwellwerte ausprobieren (nicht vergessen, zwischendrin auf *Aktualisieren* klicken, sonst verändert sich die Vorschau nicht) und sich an den optimalen herantasten. Doch in der Regel wird das Ergebnis nicht überzeugen: In unserem Beispiel ist Herr Tesla bei einem Schwellwert von 0,4 zwar wiederzuerkennen, doch die Kontur links am Gesicht verschwindet komplett, während sich der Schatten an Kinn und Wange in grobkörniges Rauschen auflöst – beim Schneiden von Transferfolie fürs T-Shirt erleidet man mit so einer Vorlage Schiffbruch 2.

Vorweichen in Gimp

Abhilfe gegen das Korn schafft der bereits im vorigen Heft im Detail beschriebene Ausflug in die ebenfalls kostenlose Bildbearbeitung *Gimp*: Damit öffnet man seine Vorlage und lässt erst mal einen *Gaußschen Weichzeichner* über das Bild laufen (zu finden unter *Filter/Weichzeichner*). Der Wert für die *X-Größe* und *Y-Größe* muss dabei gar nicht so groß gewählt werden, in unserem Beispiel reichte 3,5. Anschließend legt man die Grenze für die Tontrennung zwischen schwarz und weiß mit dem Schieberegler fest, der bei Klick auf *Farbe/Schwellwert* erscheint und hier viel bequemer als in Inkscape zu benutzen ist, weil es eine Live-Vorschau für den Effekt gibt. Falls dabei immer noch zu viel Unruhe im Schatten zu sehen ist (oder zu viele Details weichgespült wurden), nimmt man

Kurzinfor

- » Fotos effektiv auf hell-dunkel reduzieren
- » Dunkle Grafiken auf dunklem Grund zur Geltung bringen
- » Tricks mit dem kostenlosen Vektorzeichner *Inkscape*

Checkliste

-  **Zeitaufwand:** je nach Vorlage und eigenem Anspruch von einer halben Stunde bis zu vielen Stunden
-  **Kosten:** keine für die Grafik-Vorbereitung
-  **Computer:** mit Windows, macOS oder Linux

Mehr zum Thema

- » Peter König, Grafik für Maker, Make 2/21, S. 76
- » Peter König, Der einfachste Plotter der Welt, Make 1/20, S. 8

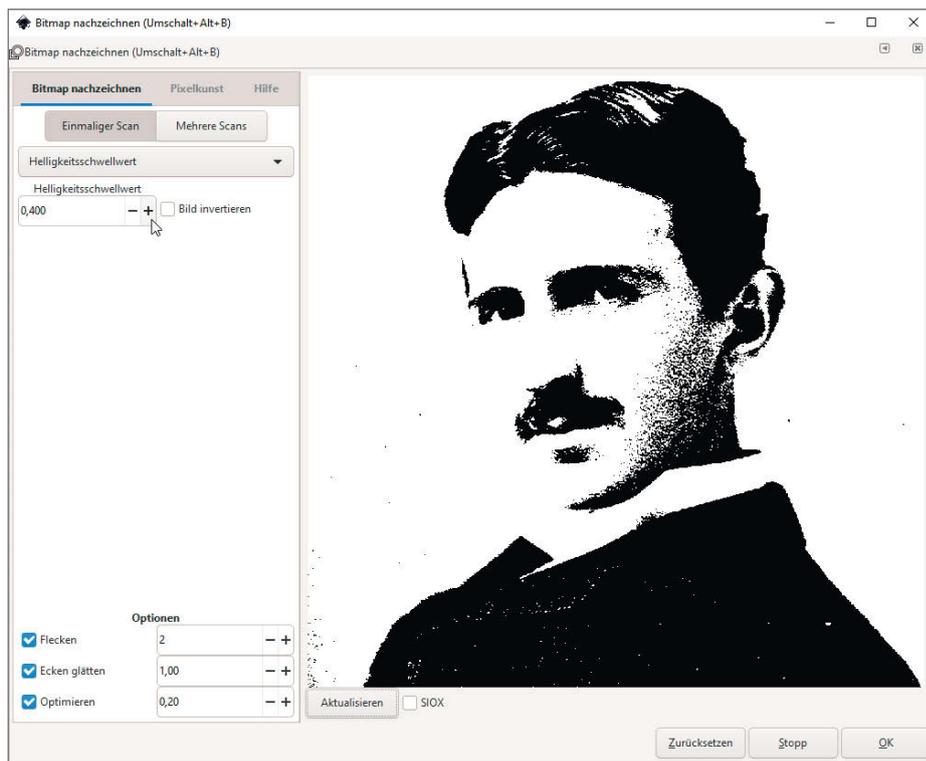
Alles zum Artikel im Web unter make-magazin.de/xwwb

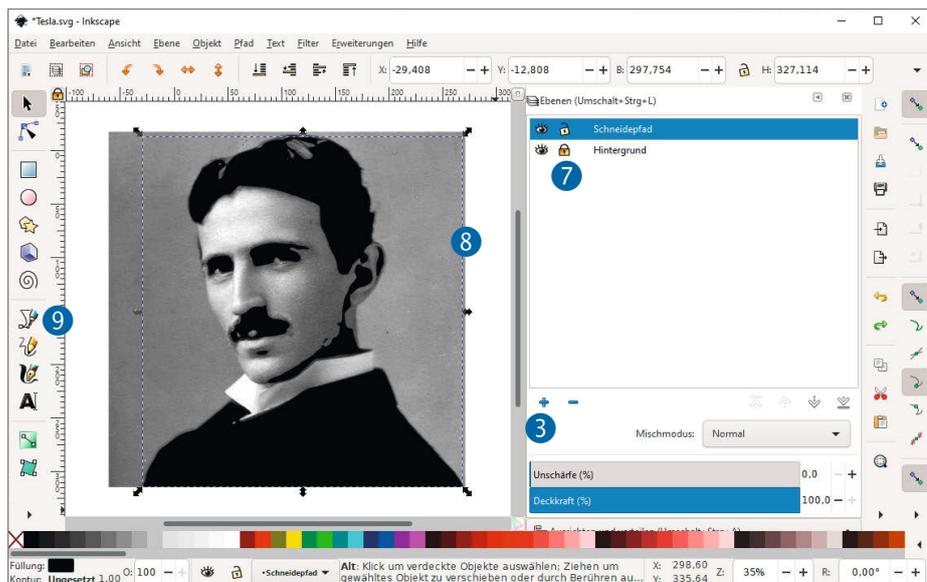
beide Schritte zurück und wählt einen anderen Parameter für den Weichzeichner. Tipp: Es lohnt sich immer, zu notieren, was man da beim letzten Durchgang gewählt hat.

Ist das Ergebnis nach Weichzeichnen und Schwellwert-Anwendung zufriedenstellend, exportiert man das Ergebnis als Pixelbild und holt es wieder in Inkscape an Bord. Dann geht es weiter wie eingangs angerissen und im Artikel in der vorigen Ausgabe ausführlich beschrieben: Vektorisieren, exportieren und ab auf den Plotter damit. Wäre da nur nicht immer noch das Problem, dass Teslas Gesicht links konturlos in den Hintergrund übergeht ...

Abpausen

Dabei hilft tatsächlich kein Automatismus, sondern nur schnödes Abpausen. Mit seinen Vektorpfaden hat Inkscape dafür exakt die richtigen Werkzeuge dabei. Zur Vorbereitung importiert man sowohl das eben vorbereitete Bild mit den schwarzen Schattenflächen als auch das Original in das Vektorzeichenprogramm. Allerdings sollen beide auf unterschiedlichen Ebenen landen. Öffnen Sie den Ebenen-Dialog über *Ebene/Ebenen ...* und fügen Sie darin über das Pluszeichen eine weitere Ebene hinzu 3. Wer will, kann die unten





liegende Ebene noch in *Hintergrund* umbenennen und die obere in *Schneidepfad* oder so ähnlich, das hilft sehr bei der Orientierung. Wichtig ist nur, dass das Originalbild jetzt auf die untere Ebene gepackt wird (Anklicken zum Auswählen und dann nach Rechtsklick

Verschiebe zur Ebene wählen und die richtige herauspicken). Das Schattenbild bleibt in der oberen Ebene.

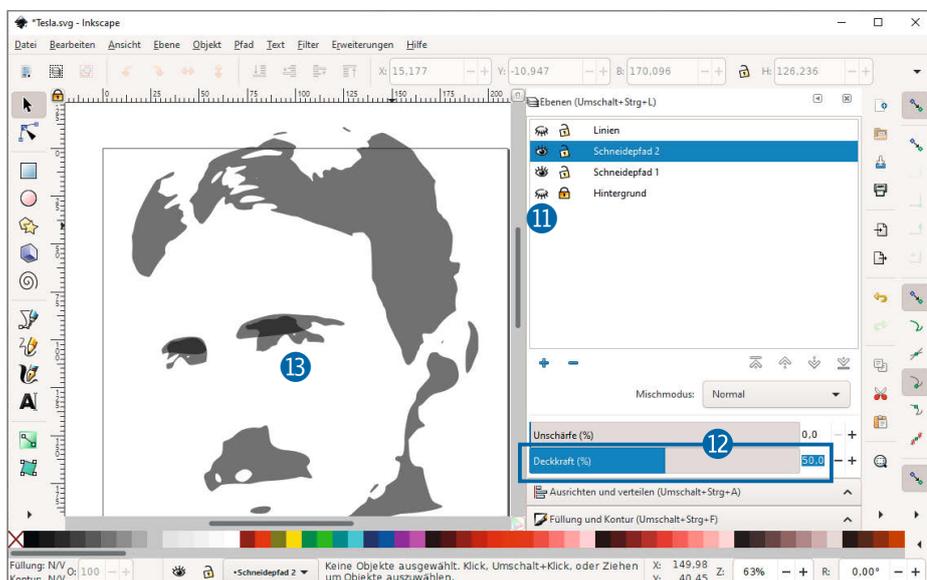
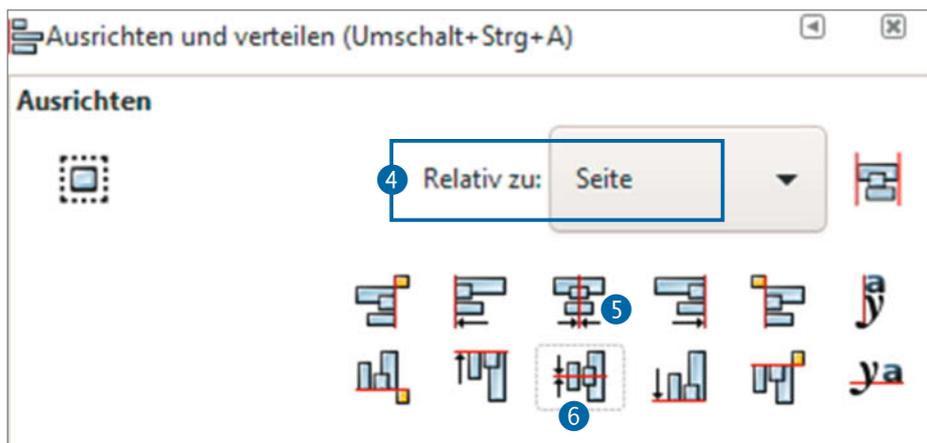
Durch Anklicken mit gehaltener Shift-Taste wählt man anschließend beide Bilder gemeinsam aus und sorgt mittels *Objekt/Ausrichten*

und *Verteilen* dafür, dass sie beide in der Mitte der Seite und damit perfekt übereinander liegen. Dazu ist es wichtig, als Bezugspunkt im *Ausrichten-Dialog Relativ zu: Seite* zu wählen (4), bevor man auf *Vertikal zentrieren* (5) und *Horizontal zentrieren* (6) klickt. Die Ebene mit dem Hintergrundbild sperrt man anschließend durch einen Klick auf das Schloss (7).

Das weichgezeichnete und mittels Schwellwert vereinfachte Bild mit den Schattenformen liegt jetzt oben – vektorisieren Sie es wie gewohnt. Nach Schließen des Dialogs wählt Inkscape das neu erzeugte Vektorobjekt aus, das Pixelbild liegt noch darunter. Mit der Tab-Taste können Sie die Auswahl darauf umschalten. Löschen Sie das Bild mit der Entf-Taste. Anschließend sollten Sie das Originalbild auf der Hintergrundebene sehen, überlagert von der davorliegenden Vektorform (8).

Jetzt können Sie den Bézier-Pinsel (9) schwingen und wichtige Konturen nach der Pausvorlage im Hintergrund nachzeichnen. Weil das mit bewegten Bildern viel besser zu erklären ist, haben wir für den Umgang mit Kurven und Pfaden, Knoten und Anfassern einen Video-Crashkurs produziert, den Sie über den Link in der Kurzinfor finden.

Im Beispiel haben wir nicht nur die Kontur an Teslas Wange ergänzt, sondern unter anderem auch noch die Nase, das Ohr, ein Stück Hals und ein paar Striche um die Augen, die das Konterfei gleich viel detaillierter aussehen lassen (10, siehe Titelseite des Artikels). Wichtig ist aber, dass man auf jeden der gezeichneten Linien anschließend *Pfad/Kontur in Pfad umwandeln* anwendet, um aus Linienzügen mit einer bestimmten Strichstärke eine Fläche zu machen, die ringsum von einem Vektorpfad begrenzt wird. Diese neue Fläche darf gerne etwas mit den bestehenden Flächen überlappen, denn vor dem Export als Schnittdatei sollte man ohnehin beides zusammen auswählen und über *Pfad/Vereinigung* zu einer einzigen Form verschmelzen.



Verfeinern

Manchmal kann man sich eingangs beim Festlegen des Schwellwerts nicht entscheiden: Der größte Teil des Bilds sieht mit Schwellwert X gut aus, aber ein wichtiger Bereich – etwa die Augenpartie – kommt bei Y viel besser zur Geltung. Mit einer kleinen Operation kann man die Sahnepartien aus zwei Vorlagenbildern kombinieren, die aus zwei Schwellwerten für die Schatten erzeugt wurden.

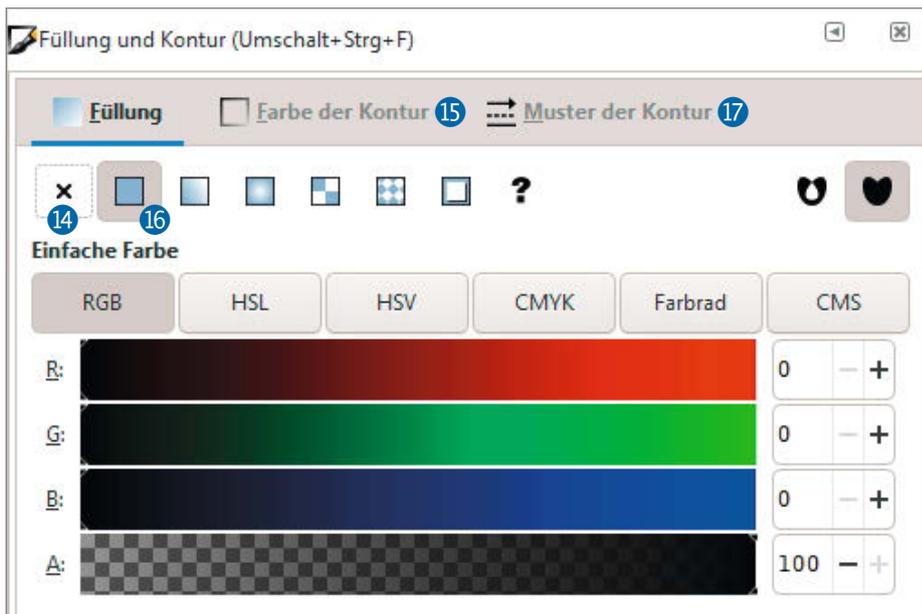
Dazu blendet man die Ebene mit der vorhin vektorisierten Form vorübergehend aus (auf das Augensymbol in der Ebenenübersicht klicken (11)) und legt eine weitere Ebene für das Bild mit dem anderen Schwellwert an. Das importiert man dort hinein und richtet es wieder horizontal und vertikal an der Seite zent-

riert aus, wodurch es deckungsgleich mit den anderen Bildern platziert wird. Dann vektorsieren und das importierte Pixelbild löschen, wie gehabt.

Jetzt zur Fusion der beiden Vektorebenen: Es hilft, hierfür die Hintergrundebene auszublenden, den beiden anderen jetzt wieder eingblendeten Ebenen je eine Deckkraft von etwa 50 Prozent zu geben **12**. Anschließend entfernt man mit dem Knotenwerkzeug (wie im vorigen Heft beschrieben und im Video gezeigt) alle unerwünschten Teile beziehungsweise korrigiert die Formen so, dass sie sich optimal ergänzen **13**. Stimmt der Gesamteindruck, kombiniert man wieder alles mittels *Pfad/Vereinigung* und setzt die Deckkraft wieder auf 100 Prozent.

Weiß auf schwarz

Soll Herr Tesla etwa ein dunkelblaues T-Shirt zieren, bieten schwarze Formen wenig Kontrast. Die Lösung ist hierfür, Teile aus heller Folie auf dunklen Untergrund aufzubringen – mehr dazu war im vorigen Heft zu lesen. Prinzipiell kann man es auch in diesem Fall so machen wie dort beschrieben, wenn man mit der Grafik nochmal ganz von vorne anfängt und dabei konsequent *negativ* denkt. Weil wir uns aber



nicht wiederholen wollen, zeigen wir hier stattdessen einen alternativen Weg. Der bietet sich insbesondere an, wenn man schon viel Mühe in eine Version der Grafik für schwarz auf weiß gesteckt hat, wie wir eben bei unserem Tesla. Dessen Hauptproblem: Er ist von

einer hellen Fläche umgeben, von der er freigestellt werden muss, damit man ihn hell auf dunkel zur Geltung bringen kann.

Voraussetzung für das folgende Freistell-Rezept ist, dass die komplette Figur von einer schwarzen Kontur umgeben ist, dass also kein

Noch mehr Stoff für Maker

2012–2014



2015 Heft 1-6



2016 Heft 1-7



2017 Heft 1-7



2018 Heft 1-7



2019 Heft 1-7

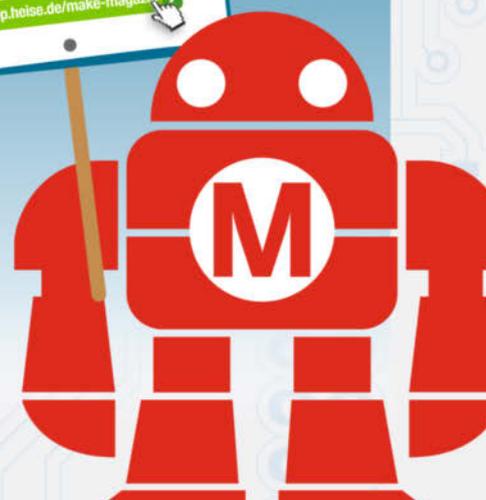


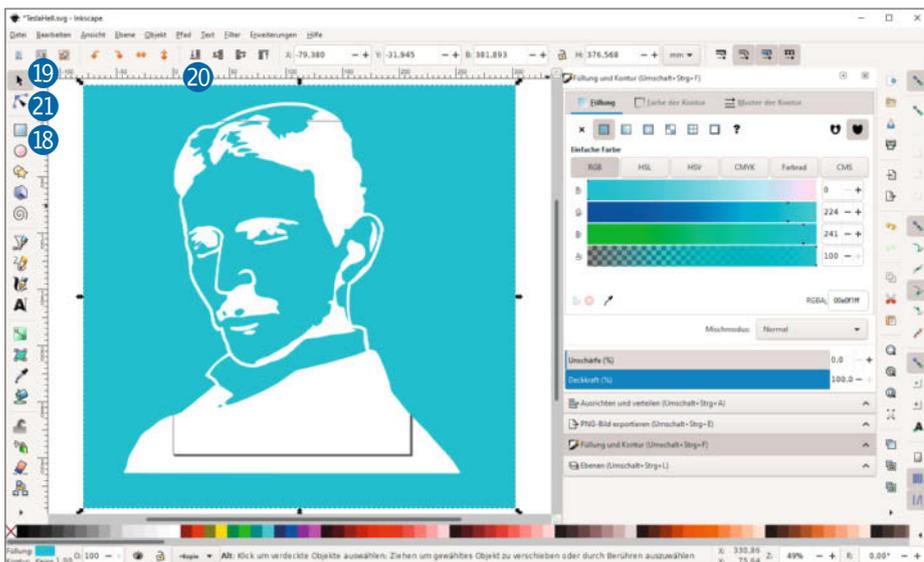
2020 Heft 1-7



Alle Ausgaben
bestellbar im
heise shop
shop.heise.de/make-magazin

Lies sie alle!





heller Bereich im Inneren der Figur direkt mit der hellen Umgebung verbunden ist. Sollte das nicht der Fall sein, zücken Sie nochmals das Bézierwerkzeug. Anschließend legt man mit Strg-C und Strg-V eine Kopie der vorhin erarbeiteten Vektorform an, verteilt Kopie und Original wieder auf zwei Ebenen und bringt sie wie gehabt im Zentrum der Seite zur Deckung.

Der Trick ist jetzt, aus dem Original helle Umrisse für schwarze Flächen zu machen. Das ist ganz einfach: Über *Objekt/Füllung und Kontur ...* die *Füllung* 14 weg-ixen und bei *Farbe der Kontur* 15 eine helle, gerne grelle Farbe nehmen, etwa kreischgrün (nicht weiß, das sieht man zwischendrin nicht). Falls der Farbwähler nicht zu sehen ist, sondern nur die Meldung *Farbe ist undefiniert* oder *keine Farbe*, klickt man vorher noch auf das ausgefüllte Quadrat (*Einfache Farbe* 16); ist die Kontur zu dünn, gibt man ihr bei *Muster der Kontur* 17 noch eine deutlich sichtbare Stärke.

Freistellen

Dann blendet man diese Ebene vorübergehend aus und nimmt sich die Kopie vor. Aus der erzeugt man jetzt Vektorformen für die hellen Flächen. Aktuell besteht diese Grafik aus dunklen Flächen und Löchern für die hellen Stellen, wir brauchen genau das umgekehrte. Dazu zeichnet man ein Rechteck 18 als Hintergrund und gibt ihm eine andere helle, grelle Farbe als Füllung (etwa hellblau). Nach Wechsel zum Auswahlwerkzeug 19 schickt man das Rechteck über einen Klick auf *ganz nach unten absenken* hinter den Tesla 20. Dann Tesla und Rechteck gemeinsam auswählen und mit *Pfad/Differenz* kombinieren. Jetzt hat man ein helles Rechteck mit ausgestanzten Löchern in Tesla-Form.

Jetzt kommt der spannende Teil, denn es gilt, Herrn Tesla von der überflüssigen hellen Form ringsherum freizustellen. Glücklicherweise gibt es dafür einen Trick: Zum Knotenwerkzeug 21 wechseln und den Tesla anklicken, sodass lauter kleine graue Quadrate auf den Pfaden erscheinen. Dann auf irgendein Pfadsegment des umgebenden Rechtecks klicken, um es auszuwählen (die Außenkanten sollten am einfachsten zu treffen sein). Anschließend invertiert man die Auswahl über die Taste ! (Shift-1), dadurch werden *alle anderen* Segmente ausgewählt, nur das zuerst angeklickte nicht. Um das wieder zur Auswahl hinzuzufügen, klickt man es wie üblich mit gedrückter Shift-Taste nochmal an. Jetzt ist der zusammenhängende Pfad der störenden hellen Umgebung komplett ausgewählt und kann schlicht mit der Entf-Taste gelöscht werden. Keine Angst, wir zeigen das in unserem Pfad-Crash-Kurs-Video online auch noch einmal.

Übrig sind jetzt die Flächen, die später weiß auf dem T-Shirt erscheinen, derzeit aber in hellblau auf weiß. Für einen besseren Eindruck

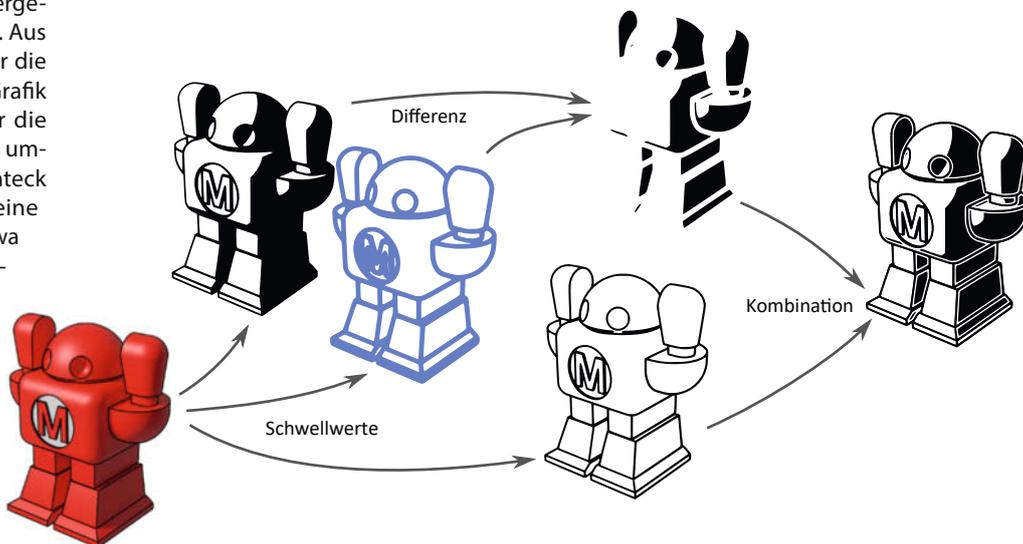
zeichnet man wieder ein Rechteck, gibt ihm eine dunkle Füllung und schiebt es hinter die hellblaue Form, der man dann eine weiße Füllung gibt.

Die hellen Stellen sind jetzt da, aber die dunklen Partien sind noch konturlos, Teslas Haare und sein Jackett verschmelzen mit dem dunklen Hintergrund. Aber dafür haben wir ja noch Umrisse der dunklen Flächen in petto (derzeit noch in kreischgrün). Blenden Sie die Ebene dafür wieder ein und platzieren Sie diese, wenn nötig, im Ebenenstapel per *Drag & Drop* nach oben (sonst liegt sie hinter dem Hintergrundrechteck). Voilà, da sind die nötig hellen Konturen um die dunklen Flächen und Linien im Inneren der Figur, etwa die Augen, einen unnötigen grünen Halo, der die Form nur vergrößert. Alle diese Pfade müssen jetzt noch weg, aber mit dem oben gezeigten Trick ist das schnell erledigt. Sind die übrigen hellen Konturen noch zu dünn, kann man sie über *Füllung und Kontur* breiter machen, dann diese Kontur wieder in eine Vektorfläche umwandeln, weiß füllen, alles zu einer Form kombinieren und dann geht's ans Plotten. Das Ergebnis sieht man auf der Titelseite des Artikels 22.

Weitermachen!

Können:innen bringen ihre Motive nicht nur einfarbig auf T-Shirts, sondern kombinieren helle und dunkle Folie für Highlights und Schatten, wobei der Hintergrund den Mittelton liefert, wie auf dem vierten Bild auf der Titelseite des Artikels zu sehen 23. Mit dem, was Sie jetzt gelernt haben, können wir Ihnen solche Projekte zur Übung selbst überlassen (ja, in der Mathe-Vorlesung fand ich diese Formulierung auch nicht toll, aber Sie kriegen das hin).

Die Grafik unten zeigt ein Rezept, nach dem der 3D-Makey aus dem vorigen Heft noch mehr Struktur in den Schatten bekommt – und einen fast metallischen Effekt. Jetzt aber: Ran an die Vektoren!
—pek



Vielfalt MACHT BEI DER REGION HANNOVER *Karriere*

Hier arbeiten 3.300 Beschäftigte
in ganz unterschiedlichen Berufen

HOCHWERTIGE AUSBILDUNG

SICHERER JOB

Auch in MINT-Bereichen

AUSBILDUNG

Ausbildung zum Elektroniker/
zur Elektronikerin
(m/w/d)

Ausbildung zum Mechatroniker/
zur Mechatronikerin
(m/w/d)

JOBS

Grabungstechniker/
Grabungstechnikerin
(m/w/d)

Bauingenieur/
Bauingenieurin (m/w/d)

und vieles mehr...

JOB ODER AUSBILDUNG

BEI DER REGION HANNOVER – ÖFFENTLICHER DIENST



JETZT ONLINE BEWERBEN:
www.DeineAusbildungMitZukunft.de
www.DaWillIchArbeiten.de



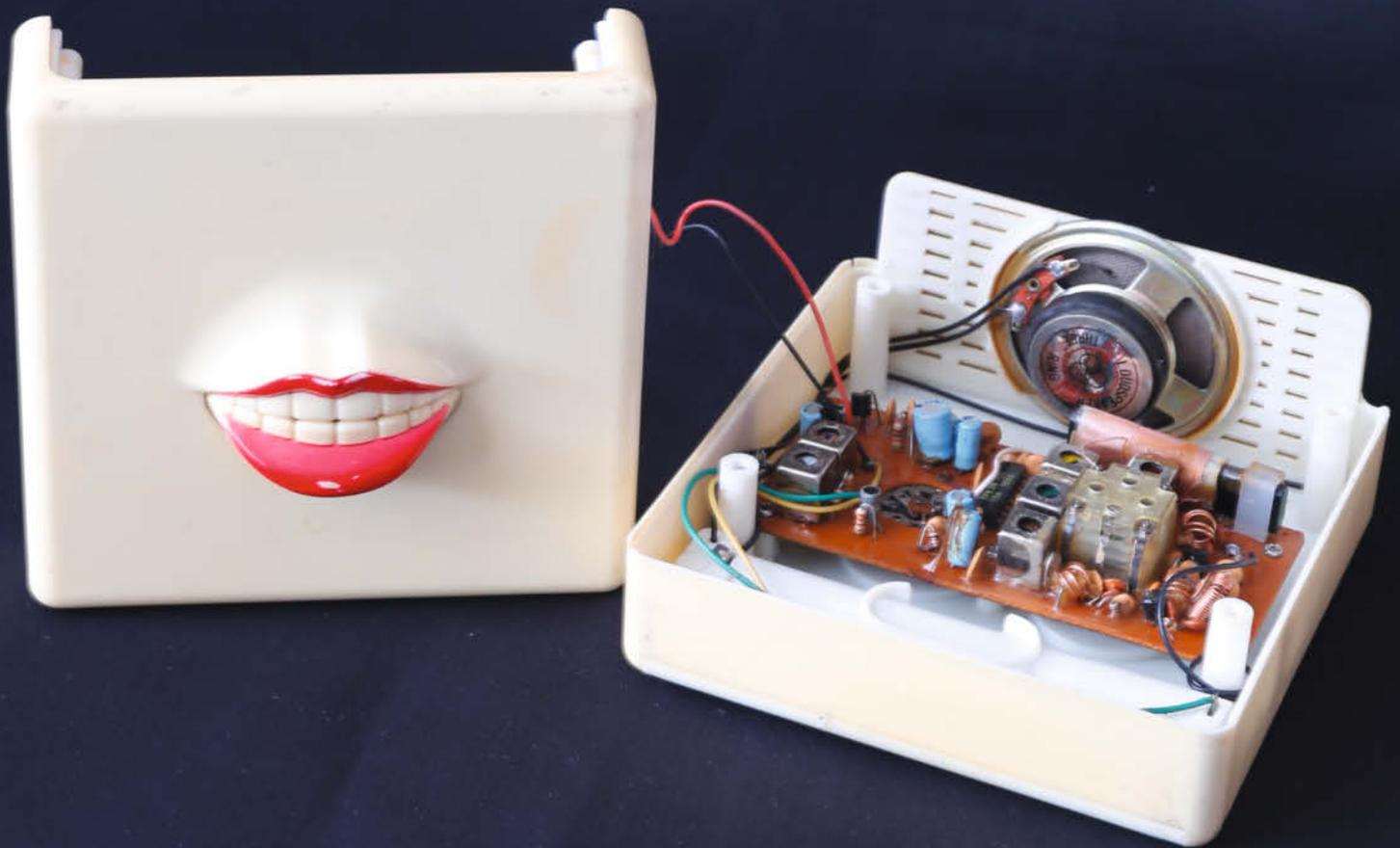
Region Hannover

Blubbermouth

Dieser sprechende Mund ist ein UKW/Mittelwellen-Radio aus den 1980er Jahren. Aber was steckt da an Technik drin?

von Heinz Behling

Alles zum Artikel
im Web unter
make-magazin.de/x55q



Blubbermouth, vielleicht am treffendsten übersetzt mit Labertasche, trägt seinen Namen völlig zu Recht. Der auffallend geschminkte Mund bewegt sich nämlich lippen-synchron zu den Klängen, die das eingebaute Radio spielt (über den Link auf der vorigen Seite können Sie sich ein Video dazu anschauen). Das wirkt natürlich nur bei Wortbeiträgen wie Nachrichtensendungen so richtig echt.

Im Inneren werkelt wie schon erwähnt ein UKW/Mittelwellen-Radio mit Lautsprecher und eher durchschnittlicher Klangqualität **1**. Damit der Antrieb der Lippen reagiert, muss man den Lautstärkeregel auf der Gehäuserückseite schon ziemlich aufdrehen, was den Lautsprecher eher scheppern als klingen lässt. Der Grund dafür: Der Elektromagnet des Lippenantriebs wird vom selben Verstärker angesteuert, der auch den Lautsprecher versorgt, und fordert halt einige Leistung, um sich in Gang zu setzen.

Der Antriebsmechanismus ist sehr einfach: An der Rückseite der horizontal drehbar gelagerten Unterlippe sitzt ein Dauermagnet, der oben auf einem Elektromagneten aufliegt. Die Lippe ist so ausbalanciert, dass sie sich

gerade von selbst schließt. Der Dauermagnet ist recht schwach, sodass er nur wenig am Eisenkern des Elektromagneten haftet. Die Kraft wird noch durch ein dazwischen gelegtes Klebeband verringert. Dies ist der Ruhezustand **2**.

Fließt jedoch ein Strom durch die Spule des Elektromagneten, erzeugt der ein Magnetfeld. Der Strom wird durch Gleichrichtung aus dem Ausgang des Niederfrequenzverstärkers gewonnen und hat daher immer dieselbe Richtung. So entsteht ein Magnetfeld, das die umgekehrte Richtung zum Magnetfeld des Dauermagneten hat. Dadurch stehen sich gleiche Magnetpole gegenüber, was zu einer Abstoßung führt. Folglich bewegt sich der Dauermagnet nach oben und entsprechend die Lippe an der Gerätevorderseite nach unten **3**.

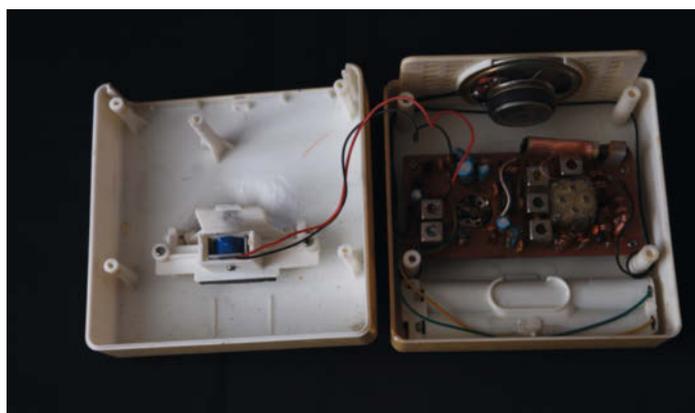
Da der Strom durch die Spule von der Lautstärke abhängt, bewegen sich auch die Lippen entsprechend unterschiedlich weit. Bei Sprachsendungen wirkt das verblüffend echt. Songs erzielen nur dann eine gute Wirkung, wenn die Singstimme relativ laut gegenüber der Instrumentalbegleitung ist. Der Frequenz-

gang des Radios scheint da kräftig nachzuhelfen.

Im Inneren des ehemals weißen, nach 40 Jahren nun aber stark vergilbten Gehäuses befinden sich außer der Radioplatine noch ein Batteriefach für vier AA-Zellen **4**. Neben den Energiespendern findet sich eine Drahtantenne, die für den UKW-Empfang zuständig ist.

Soweit die Technik aus den 80ern. Was kann man aber heute damit anfangen? Meine Idee, die letztendlich auch dazu geführt hat, Blubbermouth bei eBay zu kaufen: der Einbau eines Bluetooth-Lautsprechers und die Ankopplung an einen Sprachassistenten wie Alexa. Wenn dann der Mund auf meine Sprachbefehle reagiert und mit Alexas Stimme meine Anweisungen bestätigt, wird das sicher ein toller Effekt. Der lässt dann nicht nur am Blubbermouth, sondern auch bei so manchem ahnungslosen Besucher die Kinnlade inklusive Unterlippe herabfallen.

Platz ist dazu mehr als genug im Gehäuse. Und die Stromversorgung ist ja auch schon eingebaut. Auch ein zusätzlicher Verstärker, der dann nur den Lippenantrieb versorgt, hätte genug Platz. —hgb



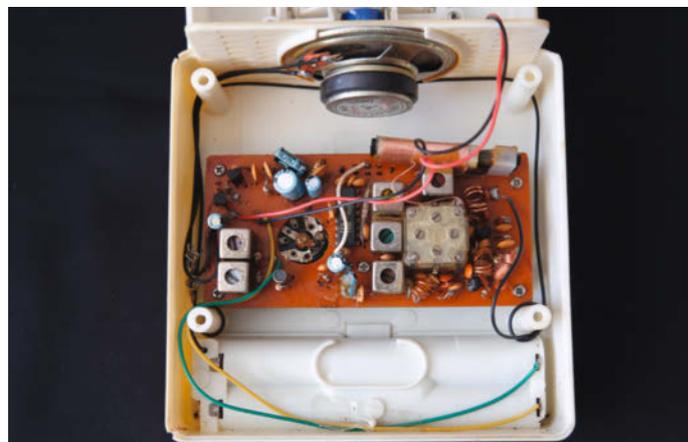
1 Der Lippenantrieb auf dem Gehäusevorderteil ist ein einfacher Elektromagnet. Rechts ist die analoge Radioplatine zu sehen.



2 Im Ruhezustand liegt der Dauermagnet am Hebel der Unterlippe am Elektromagnet an.



3 Fließt ein Strom durch die Spule, wird der Dauermagnet abgestoßen. Die Lippe auf der anderen Seite bewegt sich nach unten.



4 Für künftige Selbstbauprojekte kann man das Batteriefach gut als Stromversorgung nutzen.

3D-Drucken mit Harz: Resine im Praxistest

Warum gibt es so viel unterschiedliches Resin? Welches Resin stinkt am wenigsten? Welches ist besonders umweltfreundlich? Kann man Resine mischen? Und damit vielleicht sogar einen Farbverlauf erschaffen? Die Antwort auf all diese Fragen und viel mehr finden Sie hier, im großen Resintest. Das wichtigste vorab: Ja, man kann glitzerndes Resin drucken.

von Rebecca Husemann



Resindruck ist noch nicht so weit verbreitet wie der klassische FDM-Druck, hat aber einige unschlagbare Vorteile: Man kann mit ihm unfassbar feine Details auf 3D-Druckmodellen fertigen, die jeden Hersteller von Miniaturen vor Neid erblassen lassen. Man hat eine hohe Erfolgsquote: Die wenigen beweglichen Teile eines SLA-Druckers bergen wenig Risiko für Verschleiß und schlechte Kalibrierung. Und man produziert nennenswert weniger Plastikmüll: Das liegt zum einen an der geringeren Menge an Fehldrucken, aber auch daran, dass man eine Flasche Resin rest- und verschwendungslos aufbrauchen kann. Viele gute Gründe, sich genauer mit Resindruck und den verschiedenen Resinen auseinander zu setzen.

Für SLA-Drucker gibt es eine riesige Fülle unterschiedlicher Resine von unterschiedlichen Anbietern. Dabei ist für Resindruck-Anfänger nicht gleich klar, welches das „richtige“ Resin ist und wo die Unterschiede überhaupt liegen. Ich habe auf dem Prusa SL1 und dem Elegoo Mars unterschiedliche Resine getestet, gemischt und wilde Experimente mit ihnen gemacht. Dabei habe ich eine breite Auswahl der bekanntesten Marken, die man gut in Deutschland kaufen kann, unter die Lupe genommen. Ich schaue mir dabei die bekanntesten „Sonderfälle“ an, wie *flexibles Resin*, *Tough Resin*, *Super Low Odor Resin*, *Plant-Based* und *ABS-Like*.

Sicherheitshinweise

Viele Maker:innen machen sich beim Umgang mit Resin Gedanken um den Geruch und die Giftigkeit des Materials. Der Geruch des Resins, so hört man immer wieder, ist äußerst unangenehm. Das ist aber scheinbar wie bei Lakritz: Manche finden es fürchterlich ekelig und andere nehmen den Geruch kaum wahr – ich kann ihn beispielsweise sehr gut ausblenden. Wie sehr einen der Geruch des Resins irritiert, muss man also individuell herausfinden. Wie später im Test gezeigt wird, haben mittlerweile einige Anbieter geruchsarme Resine im Angebot, die dieses Problem lösen können.

Der zweite Faktor ist die Gefährlichkeit des Resins. Da ist was dran: Das Resin kann leicht

Kurzinfo

- » Resintypen in der Übersicht
- » Der richtige Umgang mit Resin
- » Farbverlauf, Marmorieren, Glitzerdrucke

Mehr zum Thema

- » Dominik Laa, Richtig stützen bei Resin-3D-Druck, Make 3/20, S. 102
- » Peter König, SL1 und i3 MK3S, 3D-Drucker von Prusa, Make 1/20, S. 120
- » Dominik Laa, Resin-3D-Drucker im Eigenbau, Make 4/19, S. 106

Alles zum Artikel
im Web unter
make-magazin.de/xxde



So klein und fein kann man mit einem Resindrucker Miniaturen drucken.

te Verbrennungen auf der Haut verursachen. Es gibt Berichte von Menschen, denen das Resin auf Kleidung aus synthetischem Material getropft ist und dort zu einer Reaktion und damit einer chemischen Verbrennung auf der Haut geführt hat. Daher gilt: Immer sauber arbeiten, Handschuhe tragen und am besten auch Atemmaske und Schutzbrille aufsetzen. Latexfreie Handschuhe sind am besten geeignet – in einschlägigen Drogerien erkennt man diese oft daran, dass sie schwarz sind.

Aber seien wir ehrlich: YouTuber und Co. messen wir an einem hohen Standard und toben, wenn sie die Schutzbrille vergessen, aber beim 3D-Druck zu Hause erwischt man sich doch häufig bei einem „Ach, ich schmeiß noch mal schnell 'nen Druck an“, bei dem man sich sicher nicht in Laborkittel und Schutzausrüstung wirft. Daher hier die mindest-notwendige, realistische Ausrüstung, auf die man auch in der größten Eile nicht verzichten sollte: Handschuhe, ein offenes Fenster und ein Waschbecken in erreichbarer Nähe.

Der Resintest

Opakes Resin

Der Klassiker unter den Resinen ist graues, opakes Resin. Hier sollen, vermutlich durch die Pigmentierung, die zuverlässig saubersten Ergebnisse zu erreichen sein. Mit diesem Material starten die meisten Resindruck-Anfänger.

Und hier schneiden alle Hersteller gut ab – Elegoo genauso wie Anycubic und Prusa.

Zum Vergleich mit diesen Marken aus dem mittleren bis höheren Preissegment habe ich das günstige Standard-Resin von *Eryone*, das mit „High Detail“ beworben wird, getestet: Lieber nicht! Beim Versuch, das Stützmaterial

von einer Miniatur zu entfernen, sind die Modelle gebrochen, während das mit anderen Resinen tadellos funktionierte. Fazit ist hier: Die „klassischen“ Marken unterscheiden sich eigentlich nur in Fragen persönlicher Vorlieben, während man von billigem Resin Abstand nehmen sollte.



Das klassische graue Resin ist besonders bei Tabletop-Spielern beliebt, die eigene winzige Miniaturen drucken.

Getestet: Eryone Gray, Anycubic Gray, Prusa Tough Orange, Prusa Tough Maroon, Elegoo Grey

Transparentes Resin

Ebenso beliebt wie opakes Resin ist klares Resin – das birgt allerdings einige Tücken: Farbloses transparentes Resin sieht frisch aus dem Resin-Drucker gefischt aus wie Glas. Man kann kaum glauben, wie perfekt das Ergebnis ist. Nach dem Waschen und Aushärten wird das Material allerdings matt, man könnte es als *frosted* bezeichnen. Ist das zum Waschen benutzte Isopropanol durch vorherige Drucke verfärbt, nimmt das klare Resin gerne einen Schleier der entsprechenden Farbe an.

Jetzt kommt aber der echte Haken: Klares Resin vergilbt mit der Zeit. Wie stark und wie schnell dieser Effekt eintritt, hängt stark von der Marke und der Resinsorte ab. Das transparente Resin von Elegoo war nach dem Aushärten schon deutlich gelb, das klare Resin von Anycubic gilbt erst nach ein paar Wochen und nur sehr leicht nach. Damit geht die Kaufempfehlung klar an Anycubic. (Einige Hersteller bieten im Internet zum Beispiel klare Gehäuse aus dem Resindrucker zum Kauf an – davon würde ich abraten.)

Anders verhält sich dagegen farbiges transparentes Resin, wie *Anycubic Green*. Dieses Resin sieht auch nach Monaten noch knackig grün aus. Vermutlich fällt der Vergilbungseffekt umso sanfter aus, je mehr gelb von vornherein im Farbton ist: Orange-, Rot- und Grüntöne sind scheinbar am wenigsten betroffen.

Legendär ist auch das *Siraya Tech Blu*, das besonders gut aufgelöste Drucke produzieren soll. Dieser Mythos stammt noch aus der ersten Generation Resine – mittlerweile sind die Unterschiede in der Druckqualität nur noch gering. Das *Siraya Tech Resin* ist außerdem etwas flüssiger als andere Resine. So kann man das Druckbecken weitaus leichter reinigen.

Getestet: *Anycubic Clear, Anycubic Green, Elegoo ABS-Like Translucent, Siraya Tech Blue*

Smokey Resin

Dieses ABS-Like Resin in *Smoky Black* von Siraya Tech ist ein Hybrid, das ein sehr charmantes Druckergebnis mit fast transparenten feinen Elementen und fast schwarzen dickeren Druckflächen bietet. Frisch gedruckt ist es absolut umwerfend, beim Aushärten wird es leider matter. Die Versprechungen von Siraya,

dass es weniger spröde sein soll, kann ich so nicht unterschreiben. Mir kommt es zum Beispiel körniger und rauer vor als die normalen Resine von Anycubic. Einen Stresstest mache ich im Abschnitt *ABS-Like und Tough Resine*.

Getestet: *Siraya Tech ABS-Like Resin in Smoky Black*

Mit Wasser abwaschbares Resin

Okay, nach einem Druck mit wasserabwaschbarem Resin in *Clear Blue* von Elegoo stellt sich die Frage: Warum sollte ich jemals wieder mit irgendwas anderem arbeiten? Wirklich. Was für ein Luxus. Normalerweise muss man einen 3D-Druck nach der Fertigung in einem stinkenden Becken voller Isopropanol baden – diesen Schritt kann man mit wasserabwaschbarem Resin durch ein Wasserbad ersetzen. Selbst das Reinigen des Beckens ist so sehr viel komfortabler. Wenn man weniger oder gar kein Isopropanol mehr im Setup benötigt, kann man außerdem ordentlich Geld sparen. Das Resin selbst ist dabei nichtmal teurer als normale Resine.

Das Druckergebnis ist gut und schön sauber. Die Haptik des Drucks unterscheidet sich aber spürbar von anderen Druckmaterialien. Es hat irgendwie mehr Reibung und ist klebriger, nicht so glatt. Wer das Material aber eh weiterbehandeln möchte, dem kann das egal sein.

Getestet: *Elegoo Water Washable Clear Blue*

Geruchsarm

Da viele Menschen den Geruch von Resin unangenehm bis unerträglich finden und für die freigesetzten Stoffe auch noch nicht final geklärt ist, wie viel Einfluss sie auf den Menschen haben, vertreiben verschiedene Anbieter geruchsarme Resine. Ich habe das *Super Low Odor Magenta* von Prusa beschnuppert.

Das Low Odor Resin ist, wie die meisten Prusa-Resine, sehr teuer. Der Geruch, den man aus der geöffneten Flasche erahnen kann, erinnert noch fern an Resin – es ist der bekannte Geruch, nur sehr viel schwächer. Wer genau diesen Geruch aber als stechend empfindet, wird ihn hier sicher rausriechen. Außerdem sind die Dämpfe zwar beim Einfüllen in das Druckbecken schwächer – spätestens beim Ausleeren, Filtern und zurück in die Flasche Füllen verdichtet sich der Geruch aber so stark, dass der Unterschied zu normalem Resin kaum noch ins Gewicht fällt. Das Druckergebnis ist wie bei allen Prusa-Resinen hervorragend.

Getestet: *Prusa Super Low Odor Magenta*

Plant Based

Eco-Resin besteht aus Soja-Öl und ist dadurch weniger ätzend und gesundheitsschädlich als andere Resine. Zudem lässt es



Vergilbung bei Elegoo (links) vs. Anycubic (rechts)

sich wunderbar aus Kleidung entfernen und hinterlässt dort nicht die bereits genannten chemischen Verbrennungen. Man kann es sogar kompostieren. Im Test ist das *Eco UV Resin* von Anycubic zwar etwas teurer als normale Resine, überzeugt aber durch ein fabelhaftes Druckergebnis. Es ist auch etwas viskoser – dadurch eignet es sich gut für Misch-Experimente. Aber, sorry, eine Enttäuschung: Es riecht wirklich nicht besser als die anderen Resine. Dennoch eine große Kaufempfehlung für den heimischen Betrieb, gerade wenn sich in der Nähe Haustiere und Kleinkinder aufhalten können.

Getestet: *Anycubic Eco Resin White*

Flexibles Resin

Flexibles Resin ist in der Theorie eine ganz wunderbare Idee, in der Praxis aber schwer zu verarbeiten. Nur wenige Modelle eignen sich wirklich für den Druck, da feine Strukturen zu flexibel werden und schon beim Druck reißen, massive Teile aber gar nicht erst flexibel werden. Auch Stützstrukturen sind bei flexiblem Resin eine Gedulds- und Zerreißprobe. Für mich war ein Stempel mit individuellem Motiv eigentlich der perfekte Anwendungsfall für das *Liqcreate Premium Flex* – dieses Modell hat sich aber als zu massiv herausgestellt und war ziemlich hart.

Theoretisch kann man Flexibles Resin einfach 1:1 mit normalem Resin mischen, um die Flexibilität zu ändern. Gesagt, getan: Selbes Ergebnis. Das Stempelmotiv ist für diese Resinmischung zudem zu fein und bröseln einfach ab.

Eine Möglichkeit, zu einem adäquaten Ergebnis zu kommen, wäre vermutlich, den Stempelgriff mit normalem Resin zu drucken und beim Motiv auf flexibles Resin zu wechseln. Ich habe aber erstmal die Nase voll von flexiblem Resin, denn der Geruch haut sogar mich aus den Socken – obwohl ich vorher schon an allen anderen Resinen geschnuppert habe.

Getestet: *Liqcreate Premium Flex*

ABS-Like und Tough Resin

ABS-Like Resine sollen sich besonders gut für mechanische Teile eignen. Mit ihnen kann man feine Details weniger gut darstellen, dafür sind die Drucke im Allgemeinen stabiler. Das



Smoky Black ist ein Effektresin, das Grufti-Hezen höher schlägen lässt.

ABS-like Resin von Elegoo soll laut Hersteller „weniger spröde“ sein, sich also zum üblichen Resin wie ABS zu PLA verhalten. Wenn man allerdings weiter recherchiert, gilt das nur für „im Vergleich zu normalem Elegoo-Resin“, das sowieso als recht spröde gilt. Was für eine Mogelpackung. Der Druck läuft gut und das Ergebnis ist stabil. Mehr zahlen würde ich nach der bisherigen Erfahrung für das Gütesiegel *ABS-Like* aber nicht.

Tough ist eine sehr irreführende Bezeichnung für das Resin von Prusa. Es bezeichnet wohl eher den Druck, den das Material aushalten kann, denn Spannung und Bewegung verträgt es nicht: Ein aus Prusa Tough gefertigtes Ersatzteil für ein RC-Auto ist nach kürzester Zeit gebrochen. Hier ist spannend, dass man genau sehen kann, dass es nicht entlang einer Schicht gebrochen ist, sondern wirklich wild geborsten. Ansonsten bietet es ein traumhaftes Druckergebnis, in Prusa-typischen knalligen Farben. Das Resin separiert sich aber stark: Wenn man es einen halben Tag im Becken stehen lässt, muss man es gut umrühren, weil sich am Beckenboden eine weiße Schicht absetzt.

Getestet: *Prusa Tough Maroon, Prusa Tough Orange, Elegoo ABS-Like Translucent, Siraya Tech ABS-Like Resin Smoky Black*



Nicht mehr so tough: Unter Belastung bricht das Resin der Dämpferbrücke direkt.

Experimente

Resine mischen

Man kann Resine miteinander mischen und theoretisch auch aus ihren Belichtungszeiten die optimale Belichtungszeit der neuen Kreation berechnen – in der Praxis haben fast alle

Resine sowieso die gleiche Belichtungszeit. Ich habe eigentlich jedes hier genannte Resin in verschiedenen Verhältnissen schon bunt miteinander gemischt, ohne den Taschenrechner zu zücken und alle Drucke sind gut ge-

worden. Wer auf Nummer sicher gehen will, bleibt am besten bei einer Marke. Bei *ABS-Like* Resinen kann man einfach die längere Belichtungszeit wählen. Und es ist schön, dass es so einfach geht, denn Resin mischen macht Spaß.



Na, Farblehre aus der Grundschule: Was ergibt rot und blau?



Genau, grau!

Farbmix-Resin

Es gibt neben den fertig eingefärbten Resinen auch Basis-Resine, die man mit Pigmenten mischen kann. Wunderbar, dachte ich mir, ich hätte wirklich gerne mal violette Resin. Also habe ich mir, eventuell etwas naiv, rotes und blaues Pigment von 3D Jake bestellt und in das Basis-Resin von 3D Jake gemischt. Das Ergebnis war... grau. So einfach funktioniert das leider nicht.

Die Pigmente an sich machen allerdings wunderschöne, leuchtende Farben und bieten sich auch wunderbar für Farbverlauf-Drucke und Marmorier-Experimente an.

Dazu später mehr. Auch wer gerne viele Farben, aber nicht viele verschiedene Resinflaschen herumstehen haben möchte, kann beruhigt zu den Pigmenten greifen. Daumen hoch!

Getestet: 3D Jake Color Mix Resin Basic mit 3D Jake Resin Colorant rot und blau

Resin Farbverlauf durch Nachkippen

Da man Resine einfach mischen kann und alle SLA-Drucker die Option haben, dass man während des Drucks Resin nachgießen kann, liegt es nahe, auf diese Weise einen Farbverlauf herzustellen. Der Prusa SL1 fährt zum Resin-Auf-

füllen nicht den Druckkopf hoch, dafür rührt er aber nach dem Vorgang das Becken durch.

Und so entsteht der Farbverlauf: Zuerst habe ich weißes Resin in das Becken eingefüllt. Bei 30 Prozent Druckfortschritt habe ich angefangen, in 5-Prozent-Schritten etwas Resin mit einem Plastiklöffel herauszufischen und dafür klares Resin nachzufüllen. Bei der 70-Prozent-Marke habe ich dann das blaue Pigment von 3D Jake eingerührt, das schnell zu einer deutlichen Farbveränderung geführt hat. Leider zu spät; nur die Spitze der großen Kristallzacke ist türkis geworden. Um noch einen kristalligeren Effekt zu bekommen, habe ich nach dem Aushärten die Kristallspitze ins Resin getunkt. Nachträglich aufgetragenes Resin bleibt glänzend, während der Rest leicht nachmattet. Diese Technik führt definitiv zu einem tollen Ergebnis!

Resinverlauf durch Bettwechsel

Der Elegoo Mars fährt, im Gegensatz zum Prusa SL1, den Druckkopf ganz hoch, wenn man den Druck pausiert. Das ist praktisch, denn so kann man schnell das Resin im Becken oder gleich das ganze Becken austauschen. Damit kann man zum einen sehr ebene Farbverläufe erstellen, da man die Gradienten viel gezielter anmischen kann, zum anderen kann man harte Kanten durch einen harten Farb- und Resinwechsel erzeugen.

Resin mit Duft

Wenn man alles Mögliche in Resin einmischen kann ... Funktioniert es dann auch mit Duft? Vom Seifenmachen hatte ich noch Lavendelöl da, das ich vorsichtig und tropfenweise ins



Schick: Ein Kristall mit Farbverlauf.

Siraya Tech Blue gegeben und gut durchgerührt habe. Das roch intensiv, aber angenehm, da das Resin den Duftstoff komplett angenommen hat. Ich mag den Lavendelgeruch und wollte am liebsten ständig am Resin schnuppern – eventuell ist das sogar der beste Hack für alle, die den strengen Resingeruch verabscheuen. Der Druck verlief komplett unproblematisch und das Ergebnis ließ sich leicht vom Stützmaterial befreien. Die Seite mit Stützmaterial fühlt sich etwas schmierig an – eventuell hat sich das Duftöl vermehrt am Boden des Beckens gesammelt. Das ausgehärtete Ergebnis riecht nicht stark nach Lavendel, dafür etwas süßlich. Klappt also!

Resin mit Glimmer

2015 war glitzerndes Resin noch der Aprilscherz eines Herstellers (sehr zu meinem Bedauern). Im Internet behaupten viele 3D-Drucker-Enthusiasten felsenfest, dass es nicht möglich sei, Resin mit Glitzer zu mischen und diese Mischung zu drucken – der Glitzer würde zu viel Licht reflektieren und damit die Schichten nicht richtig aushärten. Andere Resin-

druck-Fans zeigen Fotos von gedruckten Glitzerprojekten und werden dafür der Lüge bezichtigt. Da hilft nur eins: Ausprobieren. Ausschlaggebend ist natürlich, was für einen Glitzer man verwendet. Wer da an groben Glitzer aus der Bastelabteilung denkt, wird sicher ein enttäuschendes Druckerlebnis haben.

Das passende deutsche Fachwort für das, was man ins Resin mischen möchte, ist nämlich „Glimmer“. Hier sind die Partikel so fein, dass ausreichend Resin um sie herum aushärten und sie einschließen kann. Ich habe den Test mit einer Messerspitze Glimmer in Siraya Tech Fast Blue gemacht und ein schönes, schimmerndes Ergebnis bekommen. Die Glimmerpartikel haben sich in einer Ebene ausgerichtet, sodass man sie von vorne nicht sieht, die Büste von der Seite aber glitzert und funkelt. YouTuber und Co-Host des *The Meltzone*-Podcasts (siehe Seite 128) Thomas Sanladerer hat ein sehenswertes Video veröffentlicht, in dem er weitere interessante Experimente mit ganz unterschiedlichen



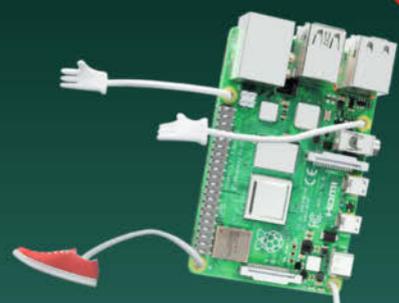
Resin mit Glimmer

Selbermachen leicht gemacht!

NEU



Heft + PDF mit 29% Rabatt



c't PROJEKTE 2021
 Mit den Projekten aus diesem Heft lernen Sie die wichtigsten Komponenten kennen: Raspberry Pi, ESP8266, Arduino Micro und Mega. Außerdem wird Ihnen die richtige Software zum Programmieren, Flashen, Konstruieren und Planen vorgestellt. Verwirklichen Sie Ihre Ideen und starten Ihre eigenen Projekte mit c't Projekte 2021!

shop.heise.de/ct-projekte21

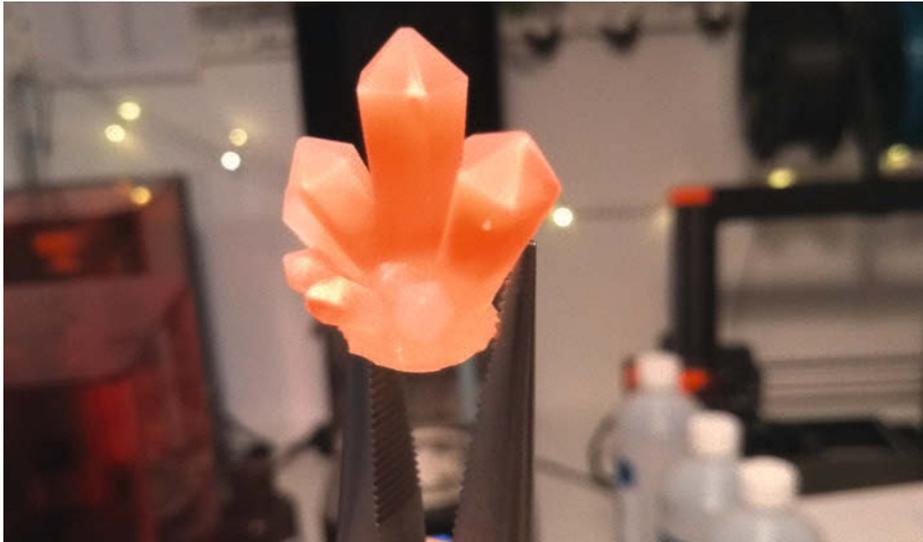
Heft + PDF für nur 19,90 € >

Bilder: Andreas Martini

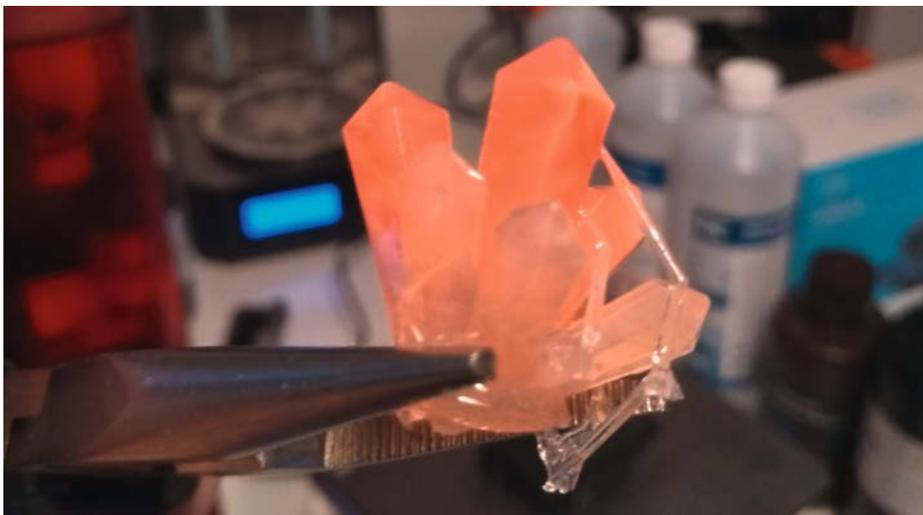
heise shop

shop.heise.de/ct-projekte21 >

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



Marmoreffekt aus Resin und Pigment



Sieht aus wie ein Rosenquarz.



Etwas subtiler ist der Marmoreffekt mit weißem Resin und blauem Pigment.

Glimmer- und Metallpigmenten gemacht hat. Den Link finden Sie in der Kurzinfo.

Marmoriertes Resin

Ein weiteres Experiment aus der Reihe „Das Internet behauptet, das geht gar nicht“: Resin mit einer marmorierten Farbgebung. Die Begründung ist, dass die Selbstmischfunktion und das Auf- und Ab des Druckkopfes im Becken das Resin zu schnell durchmischt, um für eine Marmorierung Zeit zu lassen. Wie stark diese Durchmischung ist, hängt aber stark vom Druckermodell ab – der Prusa SL1 schwenkt regelmäßig das Becken, der Elegoo Mars dagegen nicht. Außerdem spielt das verwendete Resin eine Rolle: Manche unterscheiden sich in ihrer Viskosität so stark voneinander, dass eins dann einfach zum Beckenboden sinkt. Also losgetestet.

Ein interessantes, dezentes Ergebnis hat Anycubic Plant Based in weiß mit Anycubic in transparent-blau und den Pigmenten von 3DJake ergeben: Das Plant Based Resin hat sich nach unten abgesetzt und sich so nur langsam mit dem Rest vermischt. Mit dem Pigment konnte man schöne Schlieren ins Resin ziehen, wie bei der Batik-Technik.

Den zweiten Versuch habe ich mit transparenten ABS-Like von Elegoo und Magenta-Pigment von 3DJake gemacht. Ich habe das Pigment mit einem Zahnstocher aufgenommen und jeweils nach 15 Prozent Druckfortschritt durch das Resin gezogen. Das Ergebnis ist hinreißend und erinnert an Rosenquarz.

Ähnliche Experimenten ohne die Resin Colorants von 3DJake funktionierten zwar auch, der Effekt war aber deutlich schwieriger zu steuern und ich musste regelmäßiger Resin nachgießen.

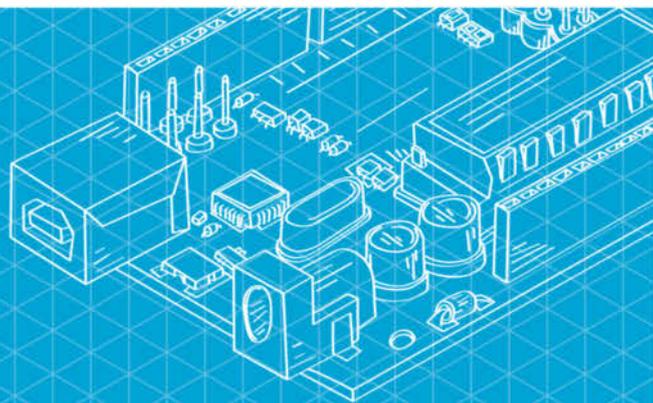
Fazit

Mittlerweile haben Resine ihre Kinderkrankheiten abgelegt: Mit eigentlich allen Marken und Varianten kann man direkt aus der Flasche hervorragende Druckergebnisse erzielen, vorausgesetzt, sie sind für den Anwendungsfall geeignet.

Mein Favorit für die Zukunft ist definitiv das mit Wasser abwaschbare Resin und Eco Resin: Diesen Luxus und den risikoärmeren Arbeitsablauf möchte ich einfach nicht mehr missen. Allgemein ist Anycubic mein Preis-Leistungssieger und wird auch weiter Verwendung finden.

Nur mit Elegoo-Resinen hab ich bisher wirklich schlechte Erfahrungen gemacht: Eine Flasche ist schon kurz vor Ablaufdatum nicht mehr brauchbar gewesen – das Resin wurde eklig schleimig und ist nicht mehr ausgehärtet. Das ist mir mit anderen Resinen nicht passiert: Selbst längst abgelauenes Anycubic Resin ist noch wunderbar druckbar. —*rehu*

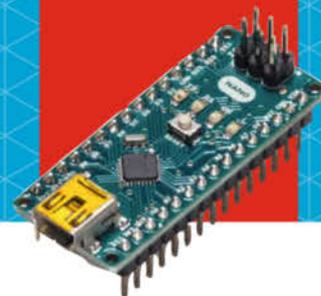
Make:



DAS KANNST DU AUCH!



GRATIS!



2× Make testen und 6 € sparen!

Ihre Vorteile:

- ✓ **GRATIS dazu:** Arduino Nano
- ✓ **NEU:** Jetzt auch im Browser lesen!
- ✓ Zugriff auf Online-Artikel-Archiv*
- ✓ 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

© Copyright by Maker Media GmbH.

Tipps & Tricks

Wer etwas in der Zahnpflege-Abteilung des Badezimmer-schränkchens gräbt, kann auf praktische Helferlein für die Werkstatt stoßen: Interdentalbürstchen machen nicht nur schöne Zähne, sondern auch winzige Bauteile rückstandslos sauber. Zum Beispiel unsere Märklin-Modelleisenbahn von Seite 52.

von Michael Gaus, Miguel Köhnlein

Ein Interdentalbürstchen sieht aus wie eine Flaschenbürste aus dem Puppenhaus. Es ist eigentlich für die Mundpflege zur Reinigung von Zahnzwischenräumen gedacht, eignet sich aber auch prima als Reinigungswerkzeug beim Basteln, da es in kleinste Ritzen passt und man so auch schwer zugängliche Stellen erreichen kann. Am Griffstück ragen zwei miteinander verdrehte dünne Drähte heraus. Je nach Ausführung ist dieser Drahtkern blank oder zusätzlich mit Kunststoff ummantelt. Der Kern ist von kleinen Borsten umschlossen.

Bei Modelleisenbahnen zum Beispiel lassen sich damit Schmutz und Ablagerungen auf Schienen und Rädern entfernen. Bei einer Airbrush-Pistole kann man mit der Interdentalbürste Nadelkanal und Düse reinigen. Ebenso gibt es Einsatzzwecke im Modellbau-bereich, beispielsweise bei der Reinigung von Zahnrädern und Schnecken sowie Gewindengängen.

Praktisch normiert

Die Bürstchen bieten gegenüber Wattestäbchen den Vorteil, dass keine Fusseln am gereinigten Objekt zurückbleiben. Interdentalbürstchen sind in Drogeriemärkten in gerader und in abgewinkelter Ausführung erhältlich. Die Größe ist in die normierten ISO Größen 0 (für kleine Zahnzwischenräume) bis 8 (für größere Zahnzwischenräume) abgestuft und kennzeichnet den passierbaren Lochdurchmesser, in den sich das Bürstchen gerade noch schieben lässt, ohne dass sich der Drahtkern verbiegt (der sogenannte *passage hole diameter*, PHD) sowie den Drahtdurchmesser. Praktisch!

Für jede ISO-Größe ist auch eine entsprechende Farbe definiert worden. Leider halten sich jedoch nicht alle Hersteller an diese Farbkodierung. Die Tabelle zeigt die Farb-



Interdentalbürstchen mit ISO-Größe 2

zuordnung sowie den entsprechenden Drahtdurchmesser.

Das rote Bürstchen mit ISO-Größe 2 in abgewinkelter Ausführung auf dem Foto hat einen PHD von 0,9mm bis 1,0mm und einen Drahtdurchmesser von 0,5mm.

Es gibt verschiedene Formen von Interdentalbürstchen: die Tannenbaumform ist vorne spitz und wird nach hinten breiter. Außerdem gibt es die sogenannte Pinselform und die Flaschenform. Welchen Typ Sie brau-

chen, hängt von der Breite Ihrer Zahnzwischenräume beziehungsweise der Form ihres Bauteils ab. Wer Interdentalbürstchen für die Zähne benutzt, sollte sie nach zehn Anwendungen entsorgen – die Modelleisenbahn hingegen wird sich zwar vor einer länger verwendeten Bürste nicht ekeln, die Borsten nutzen sich trotzdem recht schnell ab. —rehu

► make-magazin.de/xgz5

Machen Sie mit!

Kennen Sie auch einen raffinierten Trick? Wissen Sie, wie man etwas besonders einfach macht? Wie man ein bekanntes Werkzeug oder Material auf verblüffende Weise noch nutzen kann? Dann schicken Sie uns Ihren Tipp – gleichgültig aus welchem Bereich (zum Beispiel Raspberry, Arduino, 3D-Druck, Elektronik, Platinenherstellung, Lasercutting, Upcycling ...).

Wenn wir Ihren Tipp veröffentlichen, bekommen Sie das bei Make übliche Auto-rehonorar. Schreiben Sie uns dazu einen Text, der ungefähr eine Heftseite füllt und legen Sie selbst angefertigte Bilder bei. Senden Sie Ihren Tipp mit der Betreffzeile *Lesertipp* an:

mail@make-magazin.de

ISO Größe	0	1	2	3	4	5	6	7	8
Drahtdurchmesser [mm]	0,4	0,45	0,5	0,6	0,7	0,8	1,1	1,3	1,5

Ein Breadboard ist für Sie kein Schneidebrett?

Dann sind Sie bereit für den Einstieg in die Elektronik!

Exklusiv
im heise shop!



Inklusive
Experimentier-
set

NEU

Make Elektronik Special

Make Elektronik Special bietet einen einfachen und praxisorientierten Einstieg in Transistorschaltungen, die Maker in eigenen Projekten einsetzen können. Das mitgelieferte Experimentierset inkl. Breadboard, Kabeln und 45 Elektronikbauteilen enthält alles, um die gezeigten Schaltungen sofort nachbauen und testen zu können. Dabei machen die speziellen Adapterplatinen den Aufbau zum Kinderspiel. Jede Menge Projekte zum Nachbauen sind ebenfalls dabei. Also – sofort loslegen!

shop.heise.de/make-elektronik21

Heft + Experimentierset für nur
44,95 €



➤ 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**

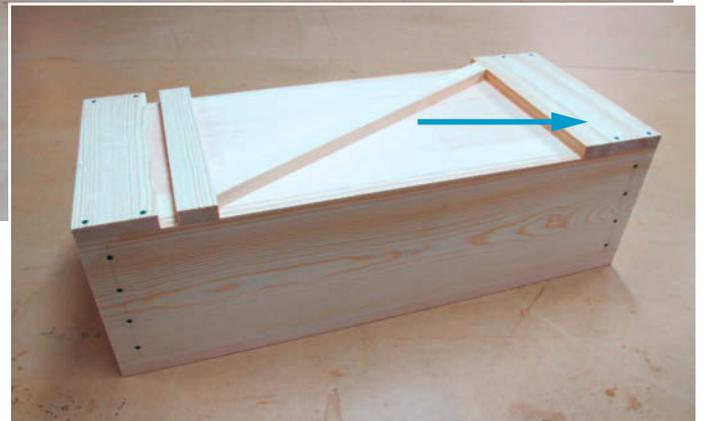
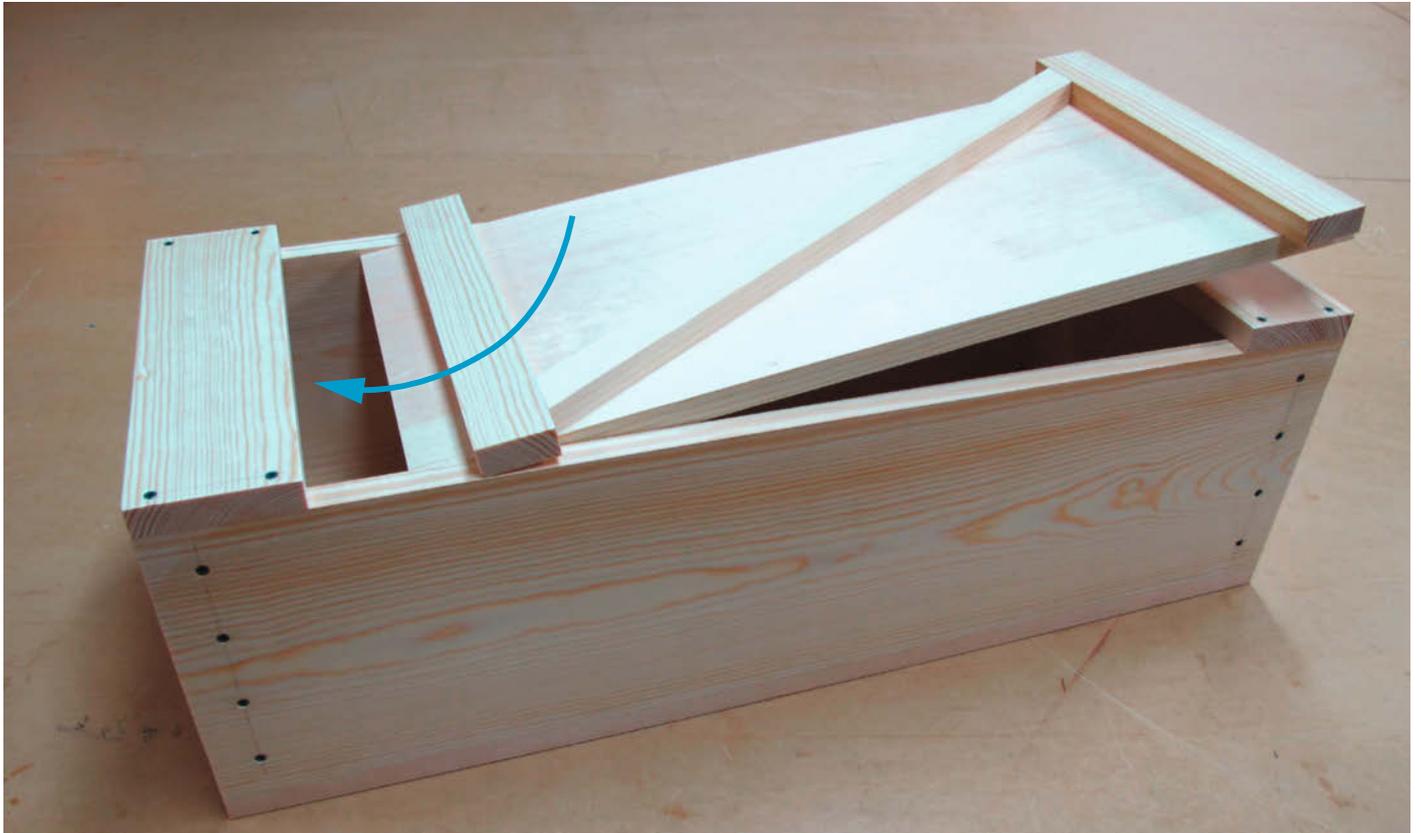
shop.heise.de/make-elektronik21



Werkzeugkiste im japanischen Stil

Diese schicke Werkzeugkiste aus Holz mit integrierten Griffleisten und dem sicher sitzenden, aber komplett abnehmbaren Deckel ist schnell und günstig selbst gebaut. Da die Maße der Konstruktion variabel sind, kann man die Kiste so stabil und groß oder leicht und kompakt bauen, wie man sie für den eigenen Bedarf braucht – ganz gleich, ob man eine empfindliche Roboterkonstruktion, die notwendigsten Handwerkzeuge oder eine amtliche Kettensäge transportieren will.

von Len Cullum (Übersetzung: Peter König)



Die erste Kiste dieser Art, die mir begegnete, diente einer Kreissäge als Aufbewahrung. Die Kiste selbst bestand aus Paneelen, als hätte man einen Hobbyraum entkernt, aus Kiefernholz-Abfällen und Dachpappnägeln. Obwohl sie optisch nicht viel hermachte, war sie stabil und mich faszinierte die Art, wie der herausnehmbare Deckel eingesetzt wurde. Ein paar Monate später erfuhr ich, dass diese Kiste genau einem gängigen Muster der Werkzeugkisten japanischer Zimmerleute entsprach. Da wusste ich: Ich musste selbst so eine Kiste bauen. Und mit ein paar Brettern, die gerade zur Hand waren – ein Zoll dick und einen Fuß breit – bekam ich es hin.

In den vergangenen 15 Jahren reisten meine Werkzeuge in dieser Kiste zu so ziemlich jedem größeren Bauvorhaben. Das einzige Problem: Weil ich die Kiste nach den Maßen meines Kofferraums gebaut habe (anstatt passend zu den Werkzeugen) ist sie eigentlich zu groß für das, was ich bei den meisten Projekten brauche. Deshalb fand ich es an der Zeit, eine neue, kleinere Kiste zu bauen, statt weiterhin das Werkzeug für kleine Projekte in meine alte Umhängetasche zu stopfen.

Größe festlegen

Der erste Schritt ist die Entscheidung, wie groß die Kiste sein soll. Weil ich die Breite meiner alten Werkzeugkiste gut finde, nutzte ich jetzt ebenfalls ein Brett mit einem Fuß Breite (etwa 300mm) und einem Dreiviertelzoll Stärke (etwa 20mm) als Boden. Generell sollte Deine Kiste Dein längstes Werkzeug aufnehmen, in meinem Fall ist das die Säge. Ich fand heraus, dass sie bequem hineinpasst, wenn die Kiste innen etwa 600mm (rund 23 Zoll) lang ist ①.

Als die Länge feststand, war es Zeit, weitere Werkzeuge draufzuschichten. Alle zusammen maßen etwas über 130mm (5 Zoll) in der Höhe ②, und weil man etwas Höhe durch die Dicke des Deckels verliert (und man vorher ohnehin nicht weiß, was später noch alles in die Kiste rein soll), entschied ich mich für 200mm breite Bretter mit ebenfalls 20mm Stärke für die Seiten. Nach ein bisschen Rechnen und einer schnellen Kreideskizze der Anordnung kam ich zu dem Schluss, dass ich alle meine Teile aus zwei vorhandenen Brettern herausbekommen könnte, wenn ich sorgsam säge.

Material auswählen

Du kannst Dir bei Massivholz viel Frustration sparen, wenn Du die besten Bretter aussuchst, die Du finden kannst. Schau nach Stücken, die gerade, flach und frei von Verdrehungen sind. Manchmal bedeutet das, beim Händler jedes Brett in die Hand zu nehmen, was er da hat; manchmal muss man auch mehr Bretter kaufen, um um mangelhafte Stellen herumzuschneiden zu können.

Kurzinfo

- » Nötige Maße der Kiste ermitteln
- » Zuschnitt: Holz geschickt ausnutzen
- » Integrierte Griffe und gut sitzender, abnehmbarer Deckel

Checkliste



Zeitaufwand:
etwa 4 Stunden



Kosten:
20 bis 40 Euro, je nach Größe, Holzart und -qualität



Holzbearbeitung:
Grundkenntnisse: Anreißen, sägen, bohren, leimen, schrauben, ggf. hobeln

Material

- » **Kiefern- oder Tannen-/Fichtenholz** 20mm dick, 30cm breit und mindestens 125cm lang
- » **Kiefern- oder Tannen-/Fichtenholz** 20mm dick, 20cm breit und mindestens 250cm lang (oder je ein Brett dieser Breite mit mindestens 140cm und 115cm Länge)
- » **Senkkopf-Spanplattenschrauben** 4 x 35mm, etwa 50 Stück (ich habe schwarze Schnellbauschrauben verwendet)

Alles zum Artikel
im Web unter
make-magazin.de/xvcc

Werkzeug

- » **Kreissäge** idealerweise Tischkreissäge und Kappsäge mit ausreichend Schneidetiefe für die Breite des Bodens; alternativ Handkreissäge mit Schiene oder selbstgebaute Sägeführung (siehe *Mehr zum Thema*)
- » **Handsäge** Feinsäge oder Japansäge
- » **Bohrmaschine** mit 3mm-Bohrer und Senkkopf
- » **Schraubendreher** oder
- » **Schraubereinsatz** passend für die Bohrmaschine, mit Bits
- » **Holzleim und Pinsel** zum Auftragen
- » **Bandmaß oder Zollstock**
- » **Bleistift** zum Anzeichnen
- » **Winkel** idealerweise verstellbar, zum Anzeichnen von parallelen und senkrechten Linien
- » **Schraubzwingen**
- » **Schleifpapier und/oder Handhobel** zum Glätten der Oberfläche (optional)

Mehr zum Thema

- » Asa Christiana, Zwei Projekte zum Verbessern der Werkstatt, Make Sonderheft 2020 „Loslegen mit Holz“, S. 46

Holz zuschneiden

Weil ich alle Teile auf meine zwei Bretter quetschen wollte, war ihre Anordnung besonders wichtig. Bild ③ zeigt, woher jedes einzelne

Teil in meinem Fall stammt. Je nach Material kann das bei Dir aber anders aussehen, dazu mehr in einem Online-Artikel (Link in der Kurzinfo). Wo immer möglich, versuche ich, alle

Hinweis

Wie bei allen Holzprojekten gilt auch hier: Es gibt viele Wege, eine Aufgabe zu lösen. Die hier gezeigten Werkzeuge und Techniken sollen eine Möglichkeit zeigen, nicht den einzigen richtigen Weg. Welche Werkzeuge, Methoden und Materialien auch immer Dir sinnvoll erscheinen – nutze sie. So habe ich zwar eine Tischkreissäge und Kappsäge zum Zuschneiden benutzt, eine Handkreissäge mit Schiene oder auch – bei entsprechender Übung – eine Stichsäge oder eine Handhobel tun es auch.

Alle genannten Maße beziehen sich auf meine Kiste (und wurden für die deutsche Übersetzung in metrische Maße umgerechnet und gerundet). Online beschreiben wir das Rezept, eine Kiste für die eigenen Wunschmaße zu berechnen und stellen dazu auch eine Tabelle für *Excel* oder *Libre Office/Open Office Calc* zur Verfügung, die Dir viel Rechnerie erspart (siehe Link in der Kurzinfo). Eine zweite Tabellenvariante eignet sich für Kisten, bei denen das Holz für die einzelnen Teile unterschiedlich dick ist.



Teile mit demselben Maß auf einmal zuzuschneiden, was sicherstellt, dass am Ende alles gut fluchtet; dazu gleich mehr.

Die Maße meiner Teile siehst Du in der Tabelle [4](#); wo die Teile bei der fertigen Kiste eingebaut werden, ist auf Bild [5](#) markiert. Die Maserung des Holzes ist dabei jeweils parallel zur angegebenen Länge der Teile.

Ich fing mit dem 300mm breiten Brett an und schnitt mit der Kappsäge an einem Ende etwa einen halben Zentimeter ab, um das raue Ende loszuwerden [6](#); das ergibt eine saubere, rechtwinkelige und verlässlichere Kante. Miss von diesem frischen Ende die Länge des Deckelbretts (B) ab und säge es zu.

Der Boden (A) und die Seiten (D) sind alle gleich lang, also Sorge dafür, dass sie exakt zueinander passen. Ich habe dazu einen Anschlag verwendet: Markiere auf dem Brett für den Boden vom eben genannten Schnitt aus die benötigte Länge (bei mir waren es 680mm) und richte das Brett schon mal so an der Säge aus, wie Du es gleich schneiden willst. Setze ein Abfallstück gegen die Kante auf der ande-

ren Seite und zwinge dieses an der Werkbank fest. Prüfe die Ausrichtung nochmals und säge dann den Boden auf Länge. Nimm dann das Brett für die Seiten, schneide wieder einen halben Zentimeter vom Ende ab, schiebe es dann vorsichtig an den Anschlag heran (nicht mit Wumms, sodass er sich verschiebt!) und schneide nacheinander die beiden Seitenteile zu [7](#).

Zeichne als Nächstes die Länge der oberen Endstücke (F) und Deckelanschlätze (G) an (bei mir sind das 300mm), versetze den Anschlag und schneide zwei Holzstücke dieser Länge aus dem Brett für die Seiten, danach zwei Stücke mit der Länge der unteren Endstücke (E, bei mir 260mm). Lasse den Anschlag an dieser Position, Du brauchst ihn gleich nochmal.

Als Nächstes folgen ein paar Längsschnitte auf der Kreissäge. Bring zunächst das Stück für das Deckelbrett (B) auf die gewünschte Breite (bei mir 259mm) und bewahre das abgeschnittene Stück als Deckelstrebe (C) auf. Schneide dann von einem der Stücke, die Du auf die Länge der oberen Endstücke (F, 300mm) ge-

schnitten hast, zwei 80mm breite Streifen ab. Schließlich schneidest Du das andere Teil dieser Länge in vier je 40mm breite Streifen auf. Mit zwei davon gehst Du zur Kappsäge zurück, nutzt dort den noch festgezugsenen Anschlag, um die beiden Griffe (H) auf Länge zu bringen (die Abfallstücke bitte aufheben!). Und das war's – jetzt bist Du bereit für den Zusammenbau.

Kiste zusammenbauen

Für den Zusammenbau gibt es viele Möglichkeiten. Über die Jahre habe ich Kisten mit Schrauben, Nägeln, Leim, Tischler-Eckverbindungen und Kombinationen aus allen vier zusammengebaut, alles hat gut funktioniert. Gehe es auf eine Weise an, die Dir liegt. Wenn Du Nägel bevorzugst – prima! Wenn Du gerne Schwalbenschwanzzinken schneidest, leg' los!

Die einzige Sache, die ich besonders empfehle: Versuche alle Bretter mit der Seite nach außen anzuordnen, an der die Rinde war, als die Bretter noch im Baum steckten [8](#). Diese Seite des Bretts wird generell glatter bleiben und diese Ausrichtung hilft, die Außenseite der Kiste frei von Ausrissen zu halten, wenn Du sie schleifst oder hobelst.

Zum Verleimen trage einen Tropfen Leim auf und verteile ihn mit einem Borstenpinsel über die Oberfläche [9](#). Verstreiche ihn nicht so dünn, dass er trocknet, bevor Du die Teile zusammenfügst, aber auch nicht so dick, dass Deine Teile sich untereinander verschieben und viel Leim seitlich herausgedrückt wird (*ein wenig ist okay*).

Griffe anbringen

Es ist wichtig, dass die Kanten alle bündig sind, deshalb prüfe vorher, ob alles passt und nimm Dir Zeit, die Teile gegebenenfalls zu korrigieren, bevor Du sie aneinander befestigst. Trage Leim auf und zwinge dann einen Griff und ein unteres Seitenstück leicht zusammen [10](#). Bringe sie in die exakte Position, ehe Du die Zwingen fest anziehst.

Drehe die gezwungenen Teile um. Markiere die drei Löcher mit 20mm Abstand (der halben Breite des Griffs) von der oberen Kante: eines in der Mitte und die anderen beiden jeweils 25mm von den senkrechten Kanten entfernt. Bohre Löcher und Senkungen für die Schraubenköpfe, dann schraube die Teile zusammen [11](#) [12](#) [13](#). Ein unter das andere Ende geschobenes Abfallstück Holz hält die Endstücke beim Schrauben in einer stabilen Position.

Ich zeichne mir Schraubenlöcher immer genau an, weil ich es mag, wenn sie gleichmäßige Abstände haben. Außerdem stellen solche Markierungen sicher, dass ich nicht an dem Brett vorbeischaube, das ich eigentlich befestigen will, und verhindern in den Ecken

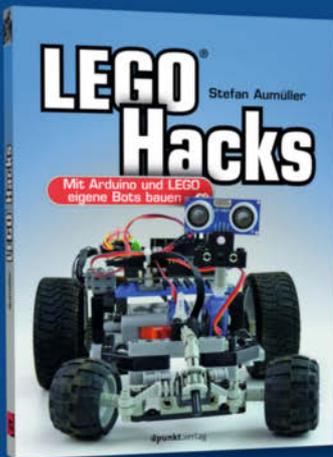
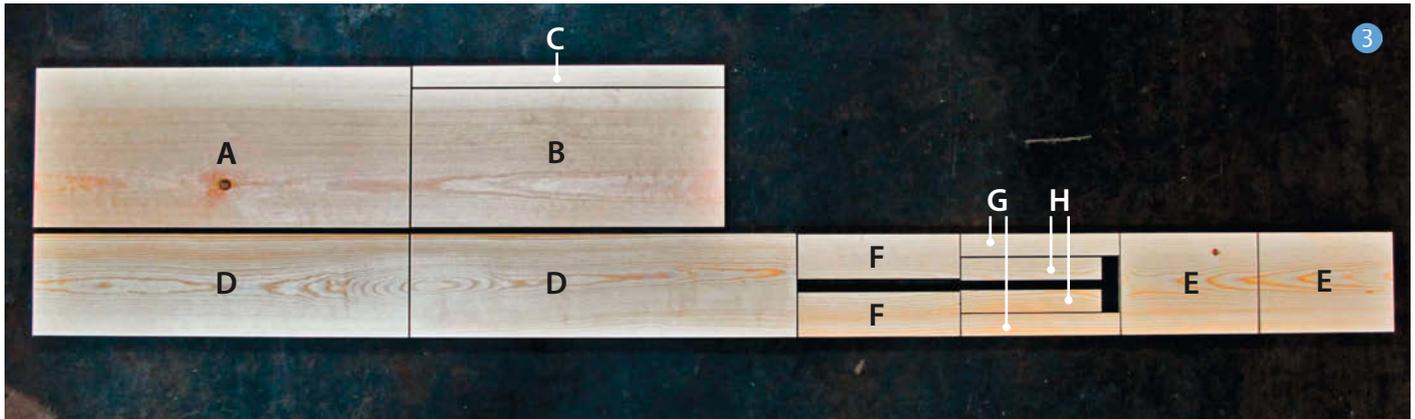
Kollisionen zwischen Schrauben von verschiedenen Seiten.

Korpus bauen

Zeichne auf die Außenseiten der Seitenwände Linien parallel zu den Kanten der Schmalseiten, etwa mit einem einstellbaren Winkel 14°. Der Abstand beträgt das anderthalbfache der Materialstärke, also 30mm. Dann miss von beiden Längsseiten jeweils 25mm und 75mm ab und markiere dort die Schraubenlöcher. Bohre alle 16 Löcher in beiden Seiten vor und bringe

4 Holzzuschnitt

Buchstabe	Bezeichnung	Länge mm	Breite mm	Dicke mm	Anzahl
A	Boden	680	300	20	1
B	Deckelbrett	556	259	20	1
C	Deckelstrebe	556	40	20	1
D	Seite	680	200	20	2
E	unteres Endstück	260	200	20	2
F	oberes Endstück	300	80	20	2
G	Deckelanschlag	300	40	20	2
H	Griff	260	40	20	2



ab 23,99 €
Print | E-Book | Bundle
ISBN 978-3-86490-643-5



ab 25,99 €
Print | E-Book | Bundle
ISBN 978-3-86490-426-4

Für Tüftler
und Erfinder



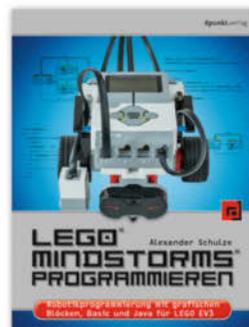
Bundle up!
Print & E-Book nur auf
www.dpunkt.de



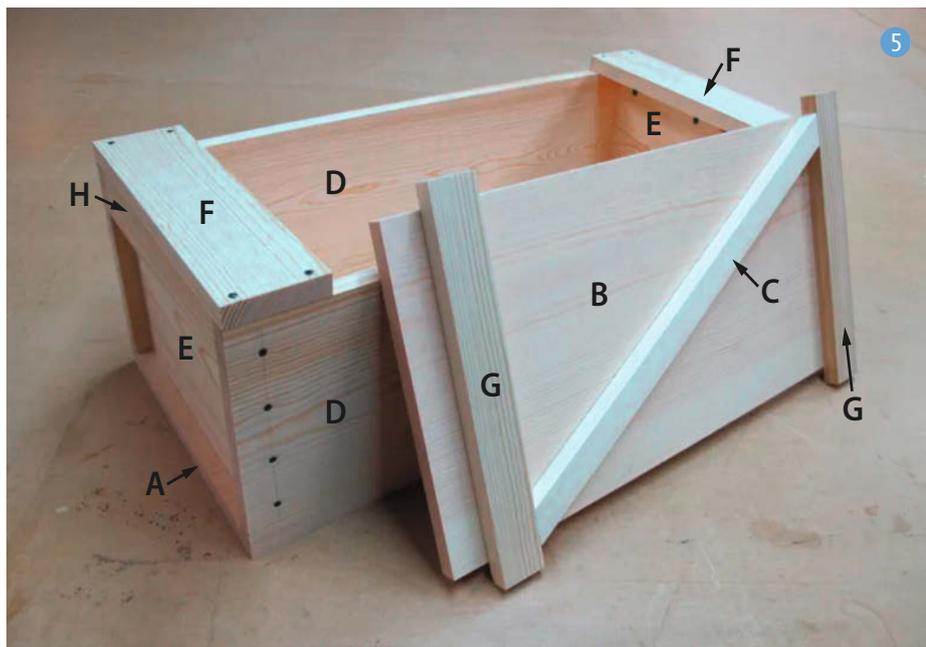
ab 21,99 € · Print | E-Book | Bundle
ISBN 978-3-86490-644-2



ab 21,99 € · Print | E-Book | Bundle
ISBN 978-3-86490-799-9



ab 23,99 € · Print | E-Book | Bundle
ISBN 978-3-86490-741-8



jeweils außen eine Versenkung für den Schraubenkopf an 15.

Trage auf die beiden Seitenkanten eines Endstücks (samt Griff) Leim auf und fixiere es mit Zwingen an der richtigen Stelle zwischen den beiden Seitenteilen. Achte wieder darauf, dass alles gut ausgerichtet ist, bevor Du die Zwingen fest anziehst. Um sicherzustellen, dass die Endstücke unten um das korrekte Maß gegenüber den Seitenteilen zurückspringen, nutze ich die Reststücke von den Griffen als Referenz 16 – Du hast die doch aufgehoben, oder? Tipp: Wenn Du das andere Endstück am offenen Ende der Seiten lose einlegst, hilft das, die Seitenteile am Verschieben zu hindern.

Wenn alles am Platz sitzt, befestige das Endstück mit acht Schrauben. Ich habe zum Schonen des Holzes Zulagen unter die Zwingen gelegt und weil mir die im Weg waren, habe ich zuerst die zwei mittleren Schrauben auf jeder Seite befestigt, dann die Zwingen abgenommen und den Rest festgeschraubt 17 18. Wiederhole die Prozedur am anderen Ende und achte darauf, dass beide Griffe auf der oberen Seite des späteren Korpus sitzen.

Boden anbringen

Drehe den entstandenen Korpusrahmen so, dass die Griffe unten liegen, und lege den Boden auf. Zeichne die Orte für die Schrauben an: von den Längsseiten her jeweils um die halbe Materialstärke (10mm) vom Rand entfernt und dann von den Enden her jeweils mit den Abständen 40mm, 160mm und 280mm. Für die Endstücke zeichne wieder eine Linie mit anderthalbfacher Materialstärke an (30mm); hier sitzt die erste Schraube bei 50mm (um zu verhindern, dass sie mit der unteren

Schraube von der Seite kollidiert) und die mittlere bei 150mm. Dann alles bohren (und Versenkungen anbringen, klar).

Leime und zwinge den Boden, stelle sicher, dass alles richtig sitzt, dann schraube den Boden fest 19. Sei nicht überrascht, falls der Boden ein bisschen über die Seiten herauszragen scheint – das kommt vor. Keiner weiß, warum. Das kannst Du später wegschleifen oder bündig hobeln.

Obere Endstücke

Bringe auf die gleiche Weise wie zuvor – bohren, leimen, zwingen und festschrauben – die oberen Endstücke an 20. Sei besonders behutsam, wenn Du die äußeren Ecken schraubst (oder nagelst), weil sich das Holz hier leicht spaltet. Vorbohren ist daher dringend angeraten (auch falls Du Nägel nimmst!). Meine Schrauben sitzen nur 12,5mm vom Ende entfernt, 20mm wäre sicherer.

Deckel bauen

Damit der Deckel hinterher reibungslos funktioniert, muss alles genau platziert werden. Stelle als erstes sicher, dass das Deckelbrett in der Breite mit etwas Luft zwischen die Seiten passt. Es ist viel einfacher, seine Breite *jetzt* zu korrigieren, bevor man die Deckelanschläge anbringt. Das Deckelbrett sollte etwa ein bis zwei Millimeter *schmäler* als die Öffnung sein und 36mm *länger*.

Zeichne auf der Oberseite des Deckelbretts eine Linie mit 39mm Abstand von einer Schmalseite an – das wird die lange Lasche des Deckels. Trage Leim auf einen der Deckelanschläge auf (aber nicht auf die ganze Breite, halte etwa um eine Materialstärke vor beiden

Enden inne) und zwinge den Anschlag leicht entlang der Linie fest, sodass das Deckelbrett um die 39mm gegenüber dem Anschlag übersteht. Nimm nochmal die beiden Abfallstücke von den Griffen zur Hilfe, um den Anschlag zu zentrieren 21. Zieh die Zwingen fest, drehe das Brett um, bohre, senke und schraube den Anschlag von der Innenseite mit drei Schrauben fest.

Für die kurze Lasche am anderen Ende zeichnest Du die Linie mit 18mm Abstand und wiederholst den Vorgang.

Jetzt kannst Du den Deckel testen: Schiebe den schräggestellten Deckel mit der langen Lasche zuerst in die Öffnung bis zum Anschlag, dann bringe den Deckel in die Waagerechte und schiebe ihn in Richtung der kurzen Lasche – den ganzen Vorgang siehst Du auf der ersten Seite des Artikels auf den drei Fotos dargestellt.

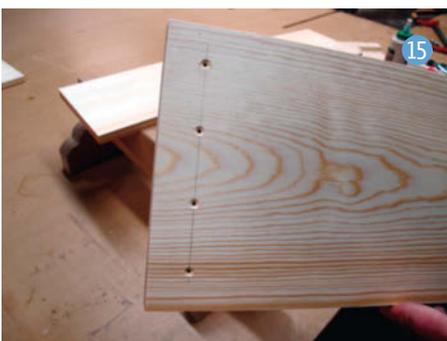
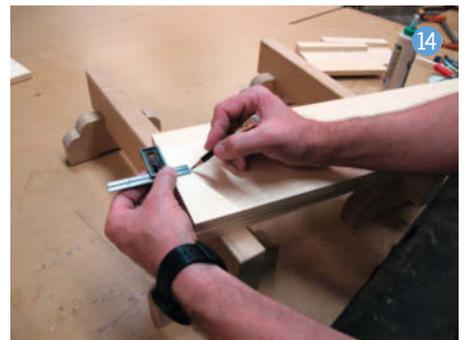
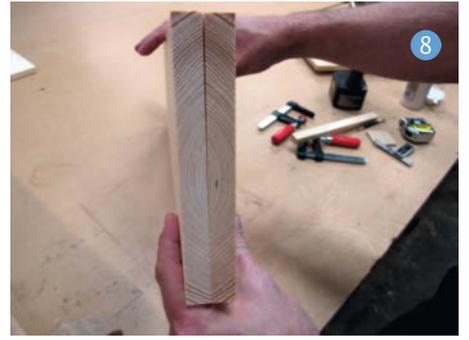
Die diagonale Deckelstrebe ist optional. Ich habe sie angebracht, weil ich das hübscher finde, außerdem macht sie den Deckel ein wenig solider und dient auch als Griff.

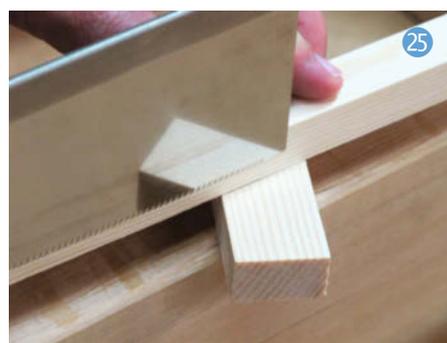
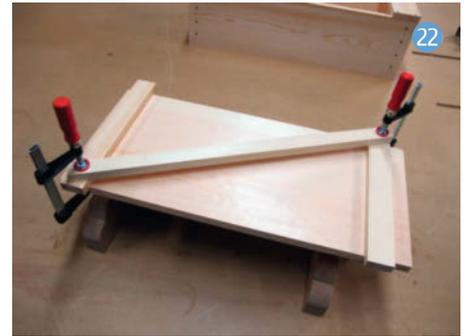
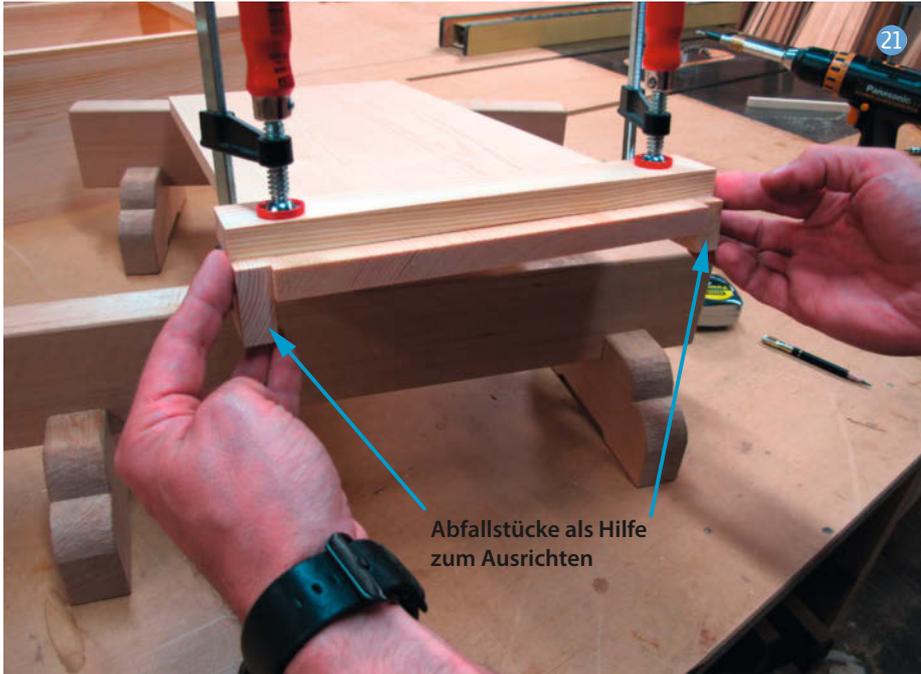
Bringe das vorhin beim Zuschneiden des Deckelbretts gerettete Abfallstück zuerst auf dieselbe Breite wie die Deckelanschläge (40mm). Zum Anbringen lege die Strebe diagonal zwischen die Deckelanschläge und zwinge sie in dieser Position fest 22. Zeichne mit einem spitzen Bleistift eine Linie von unten an 23. Um die Orientierung auch nach Lösen der Zwingen nicht zu verlieren, markiere sowohl die Strebe als auch den Deckel auf einer Seite 24. Das ist wichtig, denn oft passt die Strebe hinterher nur in eine Richtung.

Um mir den Aufriss zu sparen, die Kappsäge auf die schrägen Winkel einzustellen, mache ich diese Schnitte lieber mit einer Handsäge. Dazu setze ich ein gerades Stück Abfallholz entlang der Linie auf, halte es fest und säge daran entlang 25 26. Bevor Du das andere Ende sägst, lege die Strebe nochmal zurück auf den Deckel und prüfe nach, ob die angezeichneten Linien für die nächsten Schnitte noch stimmen. Dann leimen und festschrauben.

Einpacken!

Jetzt bist Du dran. Radiere oder schleife die Anrislinien weg oder lass' sie stehen, bringe den Boden auf die richtige Breite, falls nötig (bei mir war es das), runde die Kanten mit Schleifpapier ab und dann fülle die Kiste mit Werkzeug 27. Oder Büchern. Oder Muffins! Mit meiner höre ich hier jetzt auf, schiebe sie unter die Werkbank neben die größere und warte aufs nächste Projekt. Die Make-Redaktion aber freut sich, wenn Du ihr von Deiner fertigen Kiste per Mail ein paar Fotos schickst – und auch ein paar Infos, aus was für Holz Du sie gebaut hast, wie groß sie ist und was Du darin aufbewahrst. Und jetzt: Viel Spaß beim Bauen Deiner eigenen Kiste im japanischen Stil! —pek





Make-Sonderheft: Loslegen mit Holz

Für viele Maker ist die Arbeit mit Holz ein ungewohntes Metier, weshalb wir uns im Make-Sonderheft „Loslegen mit Holz“ diesem Thema einmal ganz intensiv widmen. In dieser Sonderausgabe von Make erfährst Du, welche Werkzeuge Dir den Einstieg bei minimalem Aufwand maximal erleichtern – und zwar unabhängig davon, ob Du mit handlichen Maschinen schnell zum Ziel kommen willst oder ob der Umgang mit Holz für Dich eine Rückkehr zum echten Handwerk sein soll. Von Lasercuttern und CNC-Fräsen ist natürlich ebenfalls die Rede, für alle, die lieber digital planen und die Maschine den Rest machen lassen wollen. Darüber hinaus gibt es ausführliche Schritt-für-Schritt-Anleitungen für Holzprojekte, die von Werkstatthelfern wie einer einfachen Handkreissägeführung über eine klassische Sägebank samt

Sägebock bis hin zu Hinguckern wie dem raffinierten Vollholz-Lautsprecher ohne Membran und Elektronik reichen, der nur durch den ausgesägten Klangkanal im Inneren die Musik vom hineingesteckten Smartphone verstärkt. Das Make-Sonderheft „Loslegen mit Holz“ ist in gedruckter Form nahezu ausverkauft, aber als PDF für 9,99 Euro im heise shop dauerhaft erhältlich (siehe Link in der Kurzinfo).

Make: Projects

HOLZ CHALLENGE

Mach mit bei der Holz-Challenge!

Wir wollen sehen, was mit Holz alles gemacht werden kann.

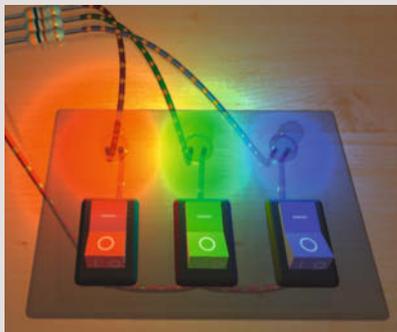
Zeige allen deine Idee rund um das Thema Holz
und sichere dir deine Chance auf einen mega Preis!

Mehr dazu auf Make Projects!

makeprojects.com/de/holz-challenge

iCircuit 3D

Schaltungssimulation mit 3D-Grafik



Wer einer nüchternen Schaltungssimulation wie *LTSpice* nicht allzuviel abgewinnen kann, findet für lötkolbenfreie Experimente vielleicht an dieser App für MacOS und iOS Gefallen: Hier leuchten LEDs realitätsnah, Motoren und Servos drehen und bewegen sich und die simulierten Schaltungen können sogar Töne erzeugen, hörbar gemacht über den iPad- oder iPhone-Lautsprecher.

Die App greift auf die umfangreichen Libraries der 2D-Schaltungssimulation *iCircuit* zurück, die neben Standard-Bauteilen auch eine erfreuliche Anzahl an fertigen, bei Makern beliebten Boards enthält. Der Clou dabei: Selbst das darauf ablaufende Programm, etwa ein Arduino-Sketch, fließt in die Simulation ein – durch die Simulation zwar langsamer als „in echt“, aber immerhin.

Man platziert und verdrahtet die Bauteile auf einem virtuellen Arbeitstisch, die Simulation ist dabei ständig aktiv und passt sich Änderungen im Aufbau laufend an. Das Ganze ist äußerst liebevoll gemacht – in glänzenden Bauteilen spiegelt sich sogar die Umgebung eines typischen Maker-Arbeitsplatzes. Die darunterliegende Technik hat der Programmator Frank A. Krueger in seinem Blog erläutert, etwa wie das Rendering mit *SceneKit* realisiert wurde oder wie eine Physik-Engine Bauteile-Kollisionen überwacht und die Schwerkraft auf dem Arbeitstisch simuliert. —cm

Hersteller	Krueger Systems, Inc.
URL	icircuit3d.appmanuals.com
Preis	16,99 €

Kria K26

FPGA-KI-Modul

Im Gewusel eines Fotos bestimmte Objekte zu erkennen, ist für das Gehirn eine leichte Aufgabe – nicht jedoch für Computer: Je nach Lernmodell ist hierfür eine erhebliche bis immense Rechenleistung erforderlich, die beim Einsatz in der Industrie oft zu übermäßigen Latenzen führt. KI-Entwickler greifen daher immer öfter auf leistungsfähige FPGAs zurück, bei denen ein neuronales Netz sozusagen in Hardware gegossen wird.

Das Design FPGA-basierter KI will Xilinx nun mit dem neuen, durchaus erschwinglichen Modul Kria K26 vereinfachen, das kaum größer als eine Scheckkarte ist und neben einem *Zync UltraScale+*-FPGA auch noch 4GB RAM (4 × 512MB × 16 Bit) und eine Reihe von Kamera-, Display- und Netzwerk-Schnittstellen beinhaltet. Im FPGA-MPSoC mit üppigen 256.000 Logik-Zellen (zum Vergleich: das große Artix-7-FPGA XC7A200 bietet 215.000) und 1200 DSP-Slices ist ein ARM Cortex-A53-Prozessor



mit vier Kernen enthalten, der jene Aufgaben übernimmt, die sich besser sequenziell (also als Programmcode) erledigen lassen.

Das K26-Starterkit umfasst Modul, Basisplatine, umfangreiche Dokumentation und Entwicklungs-Tools. Es beinhaltet zwei IAS-MIPI-Sensor-Interfaces, ein Sensor-Interface für die Raspi-Kamera, vier USB-2.0/3.0-Anschlüsse, einen DisplayPort 1.2a und einen HDMI-Anschluss. Das FPGA-Image (also der Gatter-„Schaltplan“) und die ARM-Firmware kann über einen SDHC-Kartenslot geladen werden. —cm

Hersteller	Xilinx
Vertrieb	Avnet, DigiKey, EBV Elektronik, Mouser
Preis	199 US-Dollar

Xiaomi Cloud Tokens/Map Extractor

Software-Helfer fürs Smarthome

Geräte des chinesischen Herstellers *Xiaomi* sind günstig und beliebt fürs Smarthome, brauchen für die Integration etwa in *Home Assistant* aber einen geheimen Steuerungstoken, was der *Xiaomi Cloud Tokens Extractor* liefert. Die Erweiterung *Xiaomi Cloud Map Extractor* für *Home Assistant* hingegen zeigt die aktuelle Position von *Xiaomi*- und *Roborock*-Saugrobotern an. Das hilft enorm bei der Suche nach dem Staubschlürfer, falls er sich mal wieder in Kabeln unter einem Schrank oder Bett von außen unsichtbar festgefahren hat.

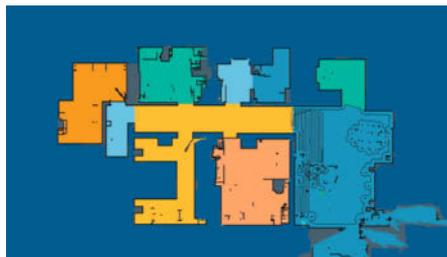
Die Token-Ermittlung ist sehr einfach: Nach dem Download und Start des Windows-Programms (siehe Link in der Tabelle) geben Sie Ihren bei *Xiaomi* registrierten Benutzernamen sowie das Passwort ein. Dann folgt noch die Wahl des länderspezifischen Hersteller-Servers und schon erscheinen alle in der *Xiaomi-Cloud* registrierten Geräte und deren Token. Diese Token können dann in die Konfigurationsdateien diverser Smarthome-Server wie *Home Assistant*, *OpenHAB* oder ähnliche eingebaut werden, um Zugriff auf die Geräte zu bekommen.

Der *Map Extractor* für *Xiaomi*- und *Roborock*-Saugroboter ist eine Erweiterung ausschließlich für *Home Assistant*. Er kann bequem über den *Home Assistant Community Store (HACS)*, aber auch manuell aus dem

Repository heraus installiert werden. Eine Anleitung dazu ist auf der Projektseite vorhanden.

Dort findet man auch die Einträge für die *Home-Assistant*-Konfiguration, die erforderlich sind, um die Erweiterung

mit ihren zahlreichen Möglichkeiten nutzen zu können. So lassen sich nicht nur der aktuelle Roboter-Standort, sondern auch die Kalibrierungspunkte der Wohnungskarte anzeigen. Die sind wichtig, damit der Sauger den Grundriss der Wohnung komplett kennt und alle Bereiche reinigen kann. So lässt sich schnell herausfinden, ob das Auslassen einiger Bereiche durch das Anlegen eines neuen Grundrisses beseitigt werden kann. —hgb



Programmautor	Piotr Machowski
URL	make-magazin.de/x4j6
Preis	kostenlos

Prusament PVB

Mit Isopropanol glättbares 3D-Druck-Filament

PVB-Filamente lassen sich mit unproblematischem Isopropanol glätten. Dies ist sowohl für dekorative Stücke eine tolle Sache, da man nicht schleifen muss, als auch für funktionale Drucke, die transparent sein sollen. Wir haben das *Prusament PVB* ausprobiert und dabei sowohl die Druckbarkeit als auch die Glättung unter die Lupe genommen.

Das Filament kommt in exzellenter Prusa-Qualität gespult und mit genauer Spezifikation, so dass hier keine Überraschungen lauern. Die Druckbarkeit auf unserem Arbeitspferd *Prusa MK2* bei 215°C und einer Standard-Düse mit 0,4mm ist sehr gut und ähnlich unproblematisch wie bei PLA. Es ist allerdings eine Tendenz zum *Curling* von feinen Überhängen sichtbar.

Das Glätten mit Isopropanol haben wir mittels Tauchen, Benetzen, Pinseln und in der Gasphase ausprobiert. Als unser Favorit hat sich hier das mehrfache kurze Tauchen erwiesen, dies ist leicht zu dosieren und zu kontrollieren. Im Bild sieht man zwei *Treefrog*-Modelle von *Thingiverse* in 0,2mm Layern gedruckt, rechts geglättet. Beide Modelle

wurden lackiert, um die Layerstruktur zu verdeutlichen. Man kann sogar die kleinen Warzen auf der Nase erkennen, die ungeglättet in der Layerstruktur verschwinden. Wände mit einzelnen Perimetern oder Schichten bekommt man fast glasklar mit *Smoothing* – so entstanden auch die Tastenkappen im Artikel auf Seite 8.

Das *Prusament PVB* ist ein gut druckbares und leicht zu bearbeitendes Material, mit dem FDM-Drucker wieder neue Möglichkeiten bekommen. Da es recht teuer und nur in transparenten Farben zu bekommen ist, sollte es eher für spezielle Aufgaben eingesetzt werden. Eine Rolle im Sortiment kann aber nicht schaden. —*caw*



Ausprobiert
— von Make: —

Einen ausführlicheren Testbericht mit vielen Bildern lesen Sie online:

► make-magazin.de/x4j6

Hersteller	Prusa Research
URL	shop.prusa3d.com
Preis	25 € / 500g

TronXY XY2-PRO

FDM-3D-Drucker-Schnäppchen

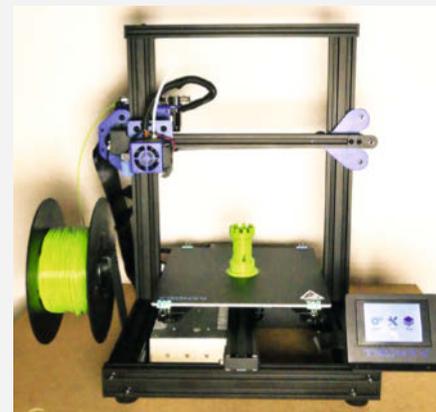
Das ist kein neuer 3D-Drucker – er kam gegen Ende 2019 auf den Markt und kostete damals so um die 400 Euro. Zurzeit gibt es aber immer wieder Angebote bei Amazon und eBay für weniger als 130 Euro, deshalb haben wir noch mal einen genauen Blick auf die Maschine geworfen, um zu prüfen, ob sich dieses Angebot lohnt.

Der XY-2 PRO läuft komplett auf 24V, hat TMC-Stepper-Treiber, einen 32-Bit-Controller mit einer ARM CPU, einen farbigen (und hellen) Touchscreen, einen induktiven Höhen-Sensor, einen Filament-Runout-Sensor und einen soliden Aufbau aus 20mm × 40mm- und 40mm × 40mm-Aluminiumprofilen. Der Bauraum misst 255mm × 255mm × 260mm. Der Drucker wird fast komplett zusammengebaut geliefert, nur ein paar Baugruppen muss man noch zusammenschrauben. Die elektrische Inbetriebnahme ist auf das Zusammenstecken zweier Flachbandkabel reduziert.

Der Extruder ist ein Bowden-Modell, der Single-Drive-Vortrieb ist am linken Rand der X-Achse befestigt. Das Filament wird mithilfe eines Teflon-Schlauchs an den Extruder

gebracht, der Extruder selbst ist recht klein und hat 40 Watt Leistung, die Lüfter sind relativ leise. Der Extruder hat keine Probleme mit PLA, PETG oder ABS, weiche Filamente wie NinjaFlex oder TPU sollten sehr langsam gedruckt werden, sonst können sie im Vorschubrad verklemmen.

Ein paar Knackpunkte gibt es allerdings, etwa die Platzierung des Netzteils unter dem Drucktisch – ohne Schutzabdeckung. Auch die Firmware hat Bugs, aber es gibt eine Marlin-Portierung aus der Community, die zuverlässig funktioniert. So wird der Drucker dank der aktuellen Rabatte tatsächlich zu einem echten Schnäppchen. Besonders die Community, die sich um ihn herum entwickelt hat, ist ein echter Bonus: Im Internet gibt es nicht nur Lösungen für die meisten Probleme, sondern auch Upgrades samt Anleitungen oder eben eine neue Firmware, die die meisten Probleme löst. Besonders hat uns gefallen, dass der Drucker in kürzester Zeit betriebsbereit war, gut druckt und dass es abgesehen von normalen Einlauf- oder Betriebserscheinungen bislang keine Probleme gab. —*Dirk Herrendoerfer/pek*



Ausprobiert
— von Make: —

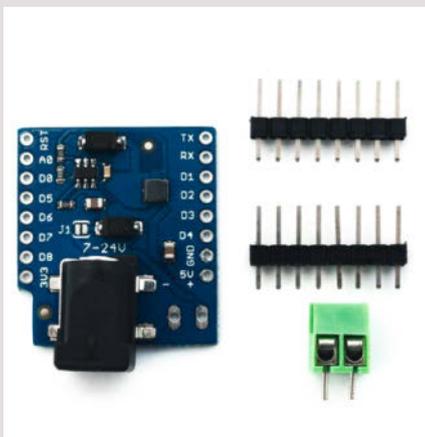
Einen ausführlichen Testbericht mit vielen Bildern lesen Sie online:

► make-magazin.de/x4j6

Hersteller	TronXY
URL	tronxyonline.com
Preis	ca. 130 €

Wemos DC Power Shield

Spannungsregler für ESP8266



Wenn man das ESP8266-Modul *Wemos D1 mini* zur Steuerung anderer Geräte einsetzt, ist dort nicht immer eine der zum Modul kompatiblen Versorgungsspannungen von 5V oder 3,3V verfügbar. Als Folge muss man dann ein zusätzliches Netzteil einplanen.

Diese lästige Schaltungserweiterung fällt nun weg: Der auf die D1 mini aufsteckbare Spannungsregler *Wemos DC Power Shield* verkraftet 7V bis 24V Eingangsspannung und kann dadurch von den meisten Geräten mitversorgt werden. Die daraus gewonnene 5V-Versorgungsspannung wird über die Steckanschlüsse direkt ans Wemos-Modul geleitet. 3,3V stellt der Spannungsregler allerdings nicht zur Verfügung. Diese Spannung wird nach wie vor aber vom D1 mini bereitgestellt.

Die Eingangsspannung kann über einen 5,1mm-Hohlstecker oder über einen zweipoligen Schraubanschluss eingespeist werden. Die Strombelastbarkeit liegt bei 1A und reicht daher nicht nur für das ESP-Modul, sondern bietet auch Reserven für eine zusätzliche Beschaltung.

Das Shield wird mit zwei Pinleisten und den Schraubkontakten geliefert. Etwas Lötarbeit ist aber erforderlich.

—hgb

Bezugsquelle diverse Internethändler, etwa eBay
Preis ca. 4 bis 5 €

Pimoroni Tiny 2040 und SparkFun Pro Micro RP2040

Boards mit dem Pico-Prozessor RP2040

Beide Boards sind deutlich kleiner als der originale *Raspberry Pico* (der zum Größenvergleich mit aufs Bild durfte) und können daher rein vom Platz her weniger GPIOs bieten, doch die Entwickler haben geschickt gewählt: So bieten beide vier ADCs (Analog-Digital-Wandler), wo der *Pico* nur drei hat. Das ermöglicht etwa Projekte mit zwei analogen X/Y-Joysticks. Weiterhin bieten die Boards mit 8MB bzw. 16MB Flash viel mehr Speicherplatz für Programme.

Für beide Boards (und den *Pico*) sind die üblichen Entwicklungsumgebungen (C++, *MicroPython*, *Circuit Python*, demnächst auch *Arduino IDE*) verfügbar. Bei *Adafruit*, den Entwicklern von *Circuit Python*, gibt es für den *Sparkfun RP2040* noch keine Variante mit angepasstem *board*-Modul. Als Abhilfe nimmt man eine *Pico-Firmware* und verwendet dann direkt die GPIO-Nummern laut Datenblatt.

Beide Boards sind allerdings gut doppelt so teuer wie der *Pico*, für den sich aktuell ein Preis von unter vier Euro einzupendeln scheint. Sie bieten aber auch einige Features und Ausstattung, die man beim *Pico* so nicht bekommt.

—caw



Einen ausführlichen Testbericht mit Vergleichstabelle lesen Sie online:

► make-magazin.de/x4j6

Hersteller	Pimoroni (Tiny 2040), Sparkfun (Pro Micro RP2040)
URL	shop.pimoroni.com/products/tiny-2040 sparkfun.com/products/17717
Preis	je ca. 8 €

Zigbee CC2530

Set aus Zigbee-Modul, Entwicklerboard und schwenkbarer Anschraubantenne

Es muss nicht immer ein ESP als Controller sein, wenn man eigene Smart-Home-Projekte entwickeln will. Auf dem Entwicklerboard dieses Sets stehen sämtliche Anschlüsse des Zigbee-Moduls an Steckkontakten zur Verfügung. Außerdem enthält es je zwei Anschlüsse für 5V- und 3,3V-Sensoren, für ein OLED-Display, zwei frei verwendbare sowie einen Resetbutton und eine USB-Schnittstelle, die zur Programmierung und als Stromversorgung benutzt werden kann.

Über eine Hohlbuchse ist außerdem ein externes Netzteil anschließbar. Als unabhängige Stromversorgung sitzt an der Unterseite des Entwicklerboards darüber hinaus ein Batteriehälter für eine AAA-Zelle.

Auf dem Zigbee-Modul thront der CC2530-Chip mit 256KB Flash-Memory. Er bietet 21 GPIO-Anschlüsse, einen 8kanaligen 12-Bit-Analog-/Digitalwandler, eine UART-sowie eine SPI-Schnittstelle. Das Modul braucht im Standby nur 0,4 Mikroampere Strom. Die maximale Stromaufnahme soll bei 20,5mA liegen.



Für das Modul gibt es im Internet einige fertige Firmware-Versionen. Eine Anleitung zum Flashen ist ebenfalls verfügbar. Die Software, mit der das Modul nicht nur geflasht, sondern auch programmiert werden kann, stammt von Texas Instruments und ist ebenfalls frei downloadbar (siehe Link).

—hgb

► make-magazin.de/x4j6

Hersteller	ChdTech
URL	Online-Handel, etwa AliExpress
Preis	ca. 14 €

ThingstAR

Thingiverse-App mit Augmented-Reality-Funktion

Die App *ThingstAR* ist ein *Thingiverse-Client* für Augmented-Reality-fähige iPads (iPad ab 5. Gen., iPad Air ab 3. Gen., alle iPad Pros).

Die Augmented Reality-Funktion (AR) erlaubt, eine STL-Datei aus einem Thingiverse-Projekt durch die iPad-Kamera auf dem Live-Bild scheinbar auf reale Oberflächen zu platzieren. Das Objekt lässt sich dann durch Wischgesten oder Knöpfe drehen, anheben oder vergrößern und so von allen Seiten betrachten. So kann man Modelle auch von hinten oder unten begutachten. Man kann dem Objekt auch eine andere Farbe geben, damit es besser in der jeweiligen Umgebung zu sehen ist. Wie für STL-Dateien typisch, bestehen die AR-Objekte nur aus einem Material.

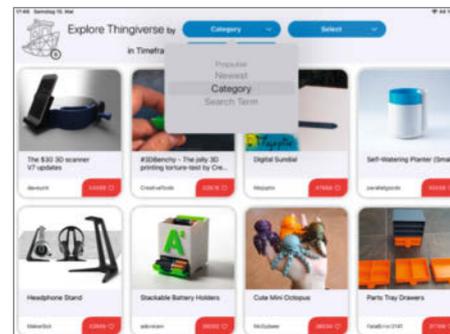
Die AR-Funktion ist gelegentlich nicht perfekt, Objekte verschieben sich schon mal und wandern auf dem Tisch. Um sich ein Objekt genauer anzugucken, dafür reicht es aber definitiv.



Auf der App-Startseite kann man nach bestimmten Begriffen, Kategorien oder besonders populären Projekten suchen. Die App ist – so wie Thingiverse selbst – auf Englisch. Die Suche bietet auch die kostenlose *Thingiverse*-App des Betreibers *MakerBot Industries*, aber diese hat im App Store sehr schlechte Bewertungen; kritisiert wird vor allem die Stabilität. Zur Benutzung wird überdies ein Account gefordert.

ThingstAR benötigt im Gegensatz dazu keinen Account, obwohl man sich auch hier Favoriten speichern kann. Wenn man sich für ein Projekt entschieden hat, kann man die STLs verschicken oder speichern und in weiteren 3D-Programmen importieren, zum Beispiel Fusion360.

Ob man *ThingstAR* benötigt und dafür gut fünf Euro zahlt, muss jeder für sich entscheiden. Ich fand die Benutzung viel angenehmer und flotter als das Browsen auf *Thingiverse* am



Computer. Die Projekte laden sofort (das ist im Browser nicht immer so und stört sehr). Hin und wieder brach das Laden der weiteren Bilder und Projekte zwar ab, das könnte aber auf *Thingiverse* selbst zurückzuführen sein. Vor dem Drucken möchte ich trotzdem lieber den G-Code am Mochter slicen und über USB oder SD-Karte zum Drucker schicken. —stri

Programmierer	Michael Beyer
Bezugsquelle	Apple App Store
Preis	5,49 €

Wappsto:Bit

Erweiterungsplatine für den BBC Micro:Bit

Diese Erweiterung für den britischen Lern-Mikrocontroller *BBC Micro:Bit* führt die GPIO-Anschlüsse als normale Pinleiste im Rastermaß 2,54 mm heraus und fügt für jeden Signalpin je einen für 3,3V und GND hinzu. Außerdem sitzt auf der 100mm x 82mm messenden Platine noch ein ESP32, mit dem der eingesteckte *Micro:Bit* (Version 1 oder 2) über eine definierte Schnittstelle Daten in Form von Zahlen und Zeichenketten austauscht. Und der ESP wiederum stellt über WLAN oder ein optionales Mobilfunkmodul für NB-IoT die Verbindung mit der *Wappsto-Cloud* des Herstellers Seluxit her. Als Frontend stehen dafür kostenlose Apps für Android und iOS zur Verfügung sowie die Webseite *wappsto.com*. Über App oder den Browser kann man sich die vom *Micro:Bit* an den *Wappsto:Bit* weitergegebenen Daten anschauen oder umgekehrt Daten über diesen Weg an den *Micro:Bit* schicken.

Im Browser lassen sich außerdem Dashboards aus Liniendiagrammen und Zeigerinstrumenten zusammenklicken. Für den Kontakt mit weiteren Cloud- und Webdiensten stehen auf *wappsto.com* einige weitere

Software-Module zur Verfügung, denn bei dieser Plattform handelt es sich um einen **Web-App-Store**, daher auch der Name.

Wer die Arbeit mit dem *Micro:Bit* gewohnt ist, bringt dieses Board in wenigen Minuten mithilfe des *Wappsto:Bit* ins Netz, klickt ein erstes Programm zusammen und kann sich dann an seinem Dashboard erfreuen – das gelingt nicht zuletzt dank der ausführlichen (und zum Teil auf Deutsch vorliegenden) Anleitungen auf der Webseite von Seluxit reibungslos. Der *Wappsto:Bit* ist dabei allerdings – ebenso wie der *Micro:Bit* – in erster Linie interessantes Lehrmaterial und als Plattform interessant, wenn etwa im Kontext des Unterrichts mit dem *Micro:Bit* programmiert wird und wenn dieses Board mit den Schülerinnen und Schülern parallel erwachsen werden soll. Dann kann über den *Wappsto:Bit* und die Cloud nach den ersten spielerischen Programmierprojekten der *Micro:Bit* in eine IoT- oder Smart-Home-Umgebung eingebunden werden. In so einem Fall bietet das Paket aus Platine und Cloud schnelle Erfolgserleb-



nisse. Man kann aber bei Bedarf anschließend auch noch tiefer in die Programmierung von eigenen Wapps mit JavaScript einsteigen. —pek

Einen ausführlichen Testbericht mit mehr Bilder lesen Sie online:

► make-magazin.de/x4j6

Hersteller	Seluxit
URL	seluxit.com
Preis	Basic: ca. 52 € NB-IoT: ca. 87 € NB-IoT+ (GPS): ca. 104 €

Ausprobiert
— von Make: —

Ausprobiert
— von Make: —

The Meltzone

3D-Druck-Podcast



Im englischsprachigen Podcast *The Meltzone* wird über 3D-Druck und Neuigkeiten rund um das Thema gesprochen. Die aktuellen Folgen beschäftigen sich mit einem dynamischen Druckbett, das Supportmaterial sparen soll, und der *Thermorph*-Technik, die den Druck durch eine PLA- und TPU-Kombination faltet – so können sich zum Beispiel gedruckte Blütenblätter zu einer schönen Blume einrollen. Es wird über die *Dragon Hotends* gesprochen, die durch ein Patent nicht mehr verkauft werden können, über den neu angekündigte *LulzBot* und die Umweltfreundlichkeit von PLA.

Der Podcast wird von den YouTubern Thomas Sanladerer und Stefan Hermann von *CNC Kitchen* veröffentlicht. Sie quatschen in entspannter Stimmung über alles, was 3D-Druck-begeisterte Maker interessiert und was die beiden auch im Alltag beschäftigt. Der Podcast ist eine Empfehlung, die man gut etwa beim Kochen hören kann, um auf dem Laufenden zu bleiben. In den aktuell 45 Folgen, die seit Ende 2018 erschienen sind, beschränken sie sich nicht auf den bekannteren FDM-Druck, sondern sprechen auch über Metall- und Harz-Druck, außergewöhnliche Drucktechniken, spannende Materialien und Software zum Entwerfen und Slicen. Auf dem YouTube-Kanal des Podcasts kann man ihnen beim Reden zugucken und ihrer sehr guten Audioqualität lauschen. Er ist aber auch zum Hören auf den gängigen Plattformen verfügbar. Die Folgen erscheinen circa zwei Mal im Monat und dauern jeweils ungefähr anderthalb Stunden. —stri

URL <https://themelt.zone>

Python 3 Schnelleinstieg

Programmieren lernen in 14 Tagen /
Einfach und ohne Vorkenntnisse zum Profi

Bücher, mit denen man sich selbst das Programmieren mit Python beibringen kann, gibt es inzwischen viele. Dieses hier ist allerdings ganz auf Effizienz getrimmt – zu einem vergleichsweise günstigen Preis bietet es ein straffes Curriculum in 14 Kapiteln, die von ganz grundsätzlichen Dingen wie *Was ist ein Algorithmus?* bis hin zu Themen wie *agile Softwareentwicklung* und *Test Driven Development* reichen, die man jetzt nicht unbedingt in einem Buch für echte Programmier-Einsteiger erwarten würde.

Das Tempo ist dementsprechend hoch, der Stoff dicht (aber durchweg klar beschrieben), dazu gibt es am Ende jedes Kapitels Übungen zur Vertiefung – insofern ist die Aussage des Untertitels gar nicht verkehrt, dass man für das Durcharbeiten dieses Buches zwei Wochen braucht, wenn man an jedem Tag ein Kapitel durcharbeitet. Wie viel Arbeit dabei pro Tag anfällt, hängt wiederum von den Vorkenntnissen ab: Wer schon andere Programmiersprachen gelernt hat, reißt den Kurs

locker nach Feierabend und am Wochenende ab, ganz ohne Vorkenntnisse sollten pro Tag schon mehr Stunden zur Verfügung stehen, um im Rahmen der zwei Wochen zu bleiben. Vor allem in die – durchaus lebensnahen – Übungen wird der eine oder die andere Zeit investieren müssen, um etwa zur Halbzeit des Kurses aus Webseiten gewünschte Wetterdaten herauszuklauben oder ab Tag bzw. Kapitel 10 GUIs zu bauen. Als Hilfestellung gibt es den Code aller Beispielprogramme und auch Lösungen zu den Übungen zum Download. Da kann eigentlich nicht mehr viel schiefgehen, wenn man sich drauf einlässt. —pek



Autor	Michael Weigend
Verlag	mitp
Umfang	288 Seiten
ISBN	978-3-7475-0328-7
Preis	Buch inklusive E-Book 9,99 €, E-Book 8,99 €

The Secret Life of Components

YouTube-Serie von Tim Hunkin

Der englische Ingenieur, Cartoonist, Autor und Künstler Tim Hunkin startete 1988 mit einer Serie über das „geheime Leben“ von Maschinen („Secret Live of...“) auf *Channel 4* im britischen Fernsehen – und die alten Folgen seiner Serie hat Hunkin *remastered* und auf YouTube gestellt. Jetzt, mit 70 Jahren, hat er außerdem eine neue Staffel im Geiste der alten Serie produziert – diesmal mit dem Titel „The Secret Life of Components“, die einzelne Bauelemente wie Schalter, Federn oder LEDs beschreibt.

Auch in den neuen Folgen geht es um das praktische Verständnis der Mechanik und Physik hinter den Bauelementen. Wie stellt man selbst Federn her? Wie wählt man die Art der Feder aus, warum reagieren Federn so und nicht anders? Welche Schalter und Lager gibt es? Wie kann man sie selbst bauen?

Dabei geht es immer sehr hemdsärmelig und direkt zur Sache: Der beneidenswert rüstige Hunkin springt, macht Klimzüge an geklebten Buchstaben oder steigt mit vielen Kilos schweren Stahlgewichten auf eine Leiter, um Schwingungsversuche mit Federn an der Decke zu machen. Das alles ist sehr informativ,



ohne die polierte, auf YouTube-Algorithmen zugeschnittene, aufgesetzte Fröhlichkeit, „moderner“ YouTuber. Hunkin geht mit merklich Spaß und großem Sachverstand an die Sache heran. Seine Werkstatt ist ein kreatives Chaos und hat so wenig mit einer modernen Industriewerkstatt zu tun, dass einem der eigene Bastelkeller gleich viel aufgeräumter vorkommt. Natürlich wird durchweg Englisch gesprochen, aber dies sehr verständlich und klar, sodass man auch gleich sein Schulenglisch aufpolieren kann. —caw

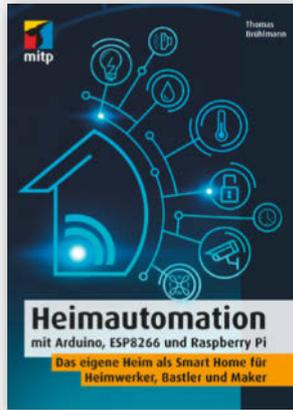
URL youtube.com/c/timhunkin1

Heimautomation mit Arduino, ESP8266 und Raspberry Pi

Das eigene Heim als Smart Home für Heimwerker, Bastler und Maker

Laut Autor richtet sich dieses Buch an Bastler und Maker, die bereits etwas Erfahrung mit Arduino und Raspberry gesammelt haben. Unverständlich, warum die ersten 150 Seiten für Basics wie Installation und Bedienung der Arduino-IDE oder Installation des Betriebssystems auf einer SD-Karte sowie die kurze Vorstellung der Arduino- und einiger weniger ESP-Boards ver(sch)wendet werden.

Die praktische Seite kommt da zu kurz. Bei den meisten Projekten werden als Mikrocontroller Arduinos in verschiedenen Ausführungen bis hin zum Selbstbau-Arduino auf Breadboard benutzt. Um die größte Schwäche dieser Boards auszugleichen – die fehlende drahtlose Netzwerkverbindung – werden 433MHz-Funkmodule bemüht oder auf umständliche Ethernet-Verkabelung hingewiesen. Die in diesem Punkt wesentlich besser geeigneten, weil von Haus aus WLAN-fähigen ESP8266-Boards hingegen werden nur bei wenigen der Beispielprojekte eingesetzt.



Und die zusätzlich mit dem stromsparenden Bluetooth LE arbeitenden und auch sonst leistungsfähigeren ESP32-Module kommen gar nicht vor. Schade, denn damit geht das Buch am aktuellen Trend im Smarthome-Bereich vorbei.

Gut sind die Erklärungen zum Thema Netzwerkprotokoll MQTT, das aktuell in zahlreichen Smarthome-Anwendungen zum Austausch von Daten zwischen den einzelnen Geräten, Sensoren und Aktoren eingesetzt wird. Erweitert wird dies durch Erklärungen zu NodeRED, der grafischen Oberfläche zur Steuerung und Automatisierung von Abläufen mit Hilfe von MQTT.

Die einfachere Möglichkeit, nämlich die Verwendung fertiger Smarthome-Server-Software wie *Home Assistant* oder *OpenHAB* wird nur kurz erwähnt. Im Hinblick auf die Zielgruppe ist das aber schade, denn damit

kämen Bastler und Maker deutlich schneller zum Ziel, statt mit den einzelnen Bibliotheken an zum Teil veralteten Boards mühsam von Hand das Rad immer wieder neu erfinden zu müssen. Das könnten diese Server nämlich mit wenigen Klicks automatisch erledigen.

Das eigentliche Titelthema, die **Hausautomation**, wird nur ganz am Rande in den praktischen Beispielen abgehandelt: So wird etwa beim Stromwächter-Projekt beschrieben, wie man die Daten des Sensor in den MQTT-Node bekommt. Aber dann? Was man damit anfangen soll, bleibt dem Leser überlassen. Da sind die Beispiele nicht zu Ende gedacht.

Insgesamt erscheint dieses Buch wie eine Anleitung für Arduino-Einsteiger, die recht holprig um einige Smarthome-Kapitel ergänzt wurde, ohne auf dem aktuellen Stand der Technik zu sein.

—hgb

Autor	Thomas Brühlmann
Verlag	mitp
Umfang	370 Seiten
ISBN	978-3-95845-672-3
Preis	Buch 29,99 €, E-Book 25,99 €, Bundle 34,99 €

Die Eigenbau-Werkstatt

Schlaue Vorrichtungen und kleine Maschinen aus Multiplex und T-Nut-Schienen

Es ist schon raffiniert und faszinierend, was James Hamilton aka *Stumpy Nubs* da aus einfachem Material und handelsüblichen Beschlägen zusammenbaut – die neun im Detail beschriebenen Projekte reichen vom Oberfräsentisch mit Schiebeschlitzen über den Zyklonabscheider für den Werkstattstaubsauger bis hin zu einer Trommelschleifmaschine und einer kapitalen Bandsäge(!). So entstehen anhand von Stücklisten, Plänen und jeweils einer detaillierten Nachbauanleitung Schritt für Schritt nicht nur Halterungen und Anschläge für fertig gekaufte Maschinen wie Oberfräse, Stich- und Tischkreissäge, sondern auch ausgewachsene Maschinen im Eigenbau – und dabei hauptsächlich aus

Sperrholz oder Multiplexplatte. Da lacht das Herz des Holzwurms.

Die reichlich vorhandenen Bilder zeigen recht klar, was in den einzelnen Bauschritten zu tun ist, wenn auch ein paar durchaus hätten größer sein dürfen. Ein größeres Manko: Man sieht zwar, wie man die Projekte *baut*, aber nicht, wie man sie *benutzt*. Und das erschließt sich aufgrund deren hoher Komplexität keinesfalls von selbst, etwa beim *Ablängschlitten Modell „Super“*, auch wenn man dem Autor sofort glaubt, dass sich damit unter anderem Zapfen, Einhängungen und Fingerzinken schneiden lassen (man würde ihm auch abnehmen, dass sie Spitzen klöppeln und Halbleiter dotieren kann). Im Buch steht an solchen Stellen stets eine URL, unter der es passende Erklärvideos geben soll, doch die läuft ins Leere. Auf dem YouTube-Kanal

von *Stumpy Nubs* ist das versprochene zwar prinzipiell vorhanden, aber zwischen den geschätzt vielen hundert Videos aus den letzten zehn Jahren nicht so einfach herauszupicken. Wird man endlich fündig, erweisen sich die meist etwa zehn Minuten langen Clips zu den Projekten auch als nicht sehr ergiebig, denn die meiste Zeit geht dabei für Plaudereien, Eigenwerbung und den manchmal etwas seltsamen Humor von *Stumpy Nub* drauf. Eigentlich schade, die ausgebufften Konstruktionen aus diesem Buch hätten es verdient, dass man all ihre Vorzüge schneller erkennt, bevor man mit dem Nachbau beginnt.

—pek



Autor	James Hamilton
Verlag	HolzWerken
Umfang	152 Seiten
ISBN	978-3-7486-0426-6
Preis	29,90 €

IMPRESSUM

Make: Nächste Ausgabe erscheint am 5. August 2021

Redaktion

Make: Magazin
 Postfach 61 04 07, 30604 Hannover
 Karl-Wiechert-Allee 10, 30625 Hannover
 Telefon: 05 11/53 52-300
 Telefax: 05 11/53 52-417
 Internet: www.make-magazin.de

Leserbriefe und Fragen zum Heft: info@make-magazin.de

Die E-Mail-Adressen der Redakteure haben die Form xx@make-magazin.de oder xxx@make-magazin.de. Setzen Sie statt „xx“ oder „xxx“ bitte das Redakteurs-Kürzel ein. Die Kürzel finden Sie am Ende der Artikel und hier im Impressum.

Chefredakteur: Daniel Bachfeld (dab)
 (verantwortlich für den Textteil)

Stellv. Chefredakteur: Peter König (pek)

Redaktion: Heinz Behling (hgb), Helga Hansen (hch), Rebecca Husemann (rehu), Carsten Meyer (cm), Andreas Perband (anp), Elke Schick (esk), Carsten Wartmann (caw)

Mitarbeiter dieser Ausgabe: Beetlebum (Comic), Len Cullum, Michael Gaus, Mirek Hančl, Till Harbaum, Dirk Herrendoerfer, Lisa Ihde, Miguel Köhnlein, Andreas Koritinik, Noah Meurer, Stella Risch (stri), Felix Watzlawek

Assistenz: Susanne Cölle (suc), Christopher Tränkmann (cht), Martin Triadan (mat)

Leiterin Produktion: Tine Kreye

DTP-Produktion: Martina Bruns, Martin Kreft (Korrektorat)

Art Direction: Martina Bruns (Junior Art Director)

Layout-Konzept: Martina Bruns

Layout: Nicole Wesche

Fotografie und Titelbild: Andreas Wodrich

Digitale Produktion: Anna Hager, Kevin Harte, Pascal Wissner

Hergestellt und produziert mit Xpublisher:
 www.xpublisher.com

Verlag

Maker Media GmbH
 Postfach 61 04 07, 30604 Hannover
 Karl-Wiechert-Allee 10, 30625 Hannover
 Telefon: 05 11/53 52-0
 Telefax: 05 11/53 52-129
 Internet: www.make-magazin.de

Herausgeber: Christian Heise, Ansgar Heise

Geschäftsführer: Ansgar Heise, Dr. Alfons Schröder

Verlagsleiter: Dr. Alfons Schröder

Stellv. Verlagsleiter: Daniel Bachfeld

Anzeigenleitung: Michael Hanke (-167)
 (verantwortlich für den Anzeigenteil),
 mediadaten.heise.de/produkte/print/
 das-magazin-fuer-innovation

Leiter Vertrieb und Marketing: André Lux (-299)

Service Sonderdrucke: Julia Conrades (-156)

Druck: Dierichs Druck + Media GmbH & Co.KG,
 Frankfurter Str. 168, 34121 Kassel

Vertrieb Einzelverkauf:

VU Verlagsunion KG
 Meßberg 1
 20086 Hamburg
 Tel.: 040/3019 1800, Fax.: 040/3019 145 1800
 E-Mail: info@verlagsunion.de
 Internet: www.verlagsunion.de

Einzelpreis: 12,90 €; Österreich 14,20 €; Schweiz 25.80 CHF;
 Benelux 15,20 €

Abonnement-Preise: Das Jahresabo (7 Ausgaben) kostet inkl. Versandkosten: Inland 77,00 €; Österreich 84,70 €; Schweiz/Europa: 90,65 €; restl. Ausland 95,20 €

Das Make-Plus-Abonnement (inkl. Zugriff auf die App, Heise Magazine sowie das Make-Artikel-Archiv) kostet pro Jahr 6,30 € Aufpreis.

Abo-Service:

Bestellungen, Adressänderungen, Lieferprobleme usw.:
Maker Media GmbH
Leserservice
 Postfach 24 69
 49014 Osnabrück
 E-Mail: leserservice@make-magazin.de
 Telefon: 0541/80009-125
 Telefax: 0541/80009-122

Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden. Kein Teil dieser Publikation darf ohne ausdrückliche schriftliche Genehmigung des Verlags in irgendeiner Form reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden. Alle beschriebenen Projekte sind ausschließlich für den privaten, nicht kommerziellen Gebrauch. Maker Media GmbH behält sich alle Nutzungsrechte vor, sofern keine andere Lizenz für Software und Hardware explizit genannt ist.

Für unverlangt eingesandte Manuskripte kann keine Haftung übernommen werden. Mit Übergabe der Manuskripte und Bilder an die Redaktion erteilt der Verfasser dem Verlag das Exklusivrecht zur Veröffentlichung. Honorierte Arbeiten gehen in das Verfügungsrecht des Verlages über. Sämtliche Veröffentlichungen in Make erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes.

Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.

Published and distributed by Maker Media GmbH under license from Make Community LLC, United States of America. The 'Make:' trademark is owned by Make Community LLC. Content originally partly published in Make: Magazine and/or on www.makezine.com, ©Make Community LLC 2020 and published under license from Make Community LLC. All rights reserved.

Printed in Germany. Alle Rechte vorbehalten.
 Gedruckt auf Recyclingpapier.

© Copyright 2021 by Maker Media GmbH

ISSN 2364-2548

Nachgefragt

Welche Maker:innen oder welches Projekt haben dich besonders nachhaltig beeindruckt?



Lisa Ihde
 Potsdam, zeigt uns auf Seite 76, wie man Unendlichkeitspiegel in Blumenform baut
 Ich bin durch die Makerin bleeptrack auf Generative Art gestoßen und finde die künstlerischen Ergebnisse durch Algorithmen wahnsinnig toll. Ich wünsche mir, dass ich auch mein Makerin-Hobby mit meiner Arbeit kombinieren kann.



Till Harbaum
 Karlsruhe, digitalisiert auf Seite 52 Modelleisenbahnen mit dem Arduino
 Der Maker Benjamin Heckendorn alias Ben Heck begeistert mich schon lange. Seine portablen Retrokonsolen mit ihrer Kombination aus DIY-Elektronik und -Mechanik lassen mich regelmäßig erfürchtigt erblassen.



Stella Risch
 Essen, vormals Make-Praktikantin, jetzt als Werkstudentin fester Teil des Teams
 Mich beeindruckt besonders Billie Ruben. Sie schafft so viele wunderschöne Maker-Projekte mit Charme! Bekannt ist sie z.B. für ihre auf Textil 3D-gedruckten Ketten oder wie lässig sie ihr 3D-gedrucktes Sexspielzeug präsentiert.



Emma Geppert
 Hannover, hat als Auszubildende die Maker Faire am 18. Juni mit vorbereitet
 Besonders cool fand ich das Video von Jeni.Swiss auf Youtube, das ich bei meiner Recherche für Beiträge in der Entertainmentshow gefunden habe. Dort hat er eine komplette Werkstatt in einen roten, alten Ford gebaut.

Inserentenverzeichnis

Arrow Central Europe GmbH, Neu Isenburg	61	Reichtel Elektronik GmbH & Co., Sande	31
dpunkt.verlag GmbH, Heidelberg	95, 119	Rheinwerk Verlag GmbH, Bonn	13
ELEGOO Inc, CN-Shenzhen	25	Shaper Tools GmbH, Leinfelden-Echterdingen	2
Fachverband Elektronik-Design (FED) e.V., Berlin	15	Siemens AG, Nürnberg	43
hannoverimpuls GmbH, Hannover	51	Technische Universität Ilmenau, Ilmenau	37
HBK Essen GmbH, Essen	41	ZUKUNFTINC. e.V., Hannover	132
Jade Hochschule, Wilhelmshaven	67	Make:markt	93
Region Hannover, Hannover	103		

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

Partner



Sponsoren



präsentiert von



WIR SUCHEN KLUGE KÖPFE

Deine **Karriere** bei einem **Marktführer** aus der **Region Hannover** beginnt mit einem Online-Besuch auf **ZUKUNFT-INC.de**

+ JETZT BEWERBEN!



ZUKUNFTINC.-Partner sind:



Region Hannover



ibk IngenieurConsult
Die Lösung.

