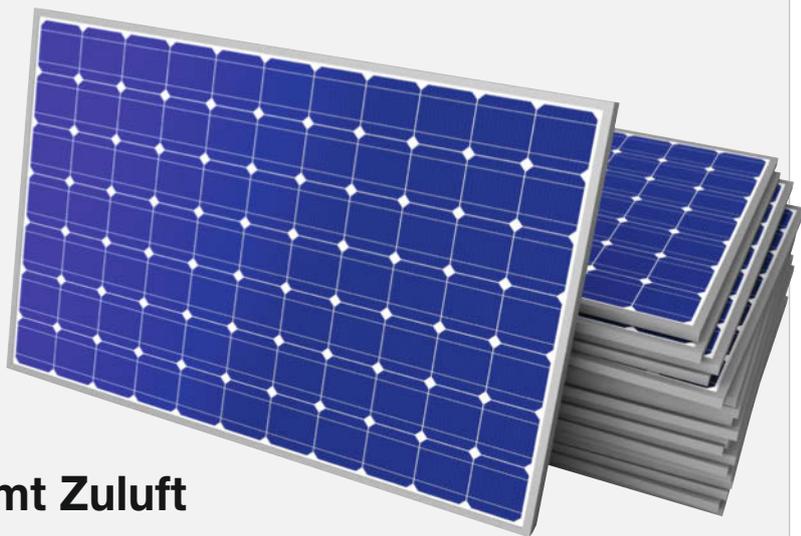




Brandneu:
Raspberry Pico W

Energie erzeugen und sparen

- ▶ Solarpanel als Terrassentisch nutzen
- ▶ So geht's: Solar-Versorgung für Wetterstation
- ▶ Warmwasserpumpen-Hack spart Energie
- ▶ Eigenbau-Wärmetauscher wärmt Zuluft



Projekte

- ▶ Stromausfall-Monitor
- ▶ Dartscheiben-Flutlicht
- ▶ Fliegende Laufschrift



Know-how

- ▶ Arduino-Alternative mit STM32
- ▶ Tricks für Resin-3D-Drucker
- ▶ Maker-Fotostudio einrichten

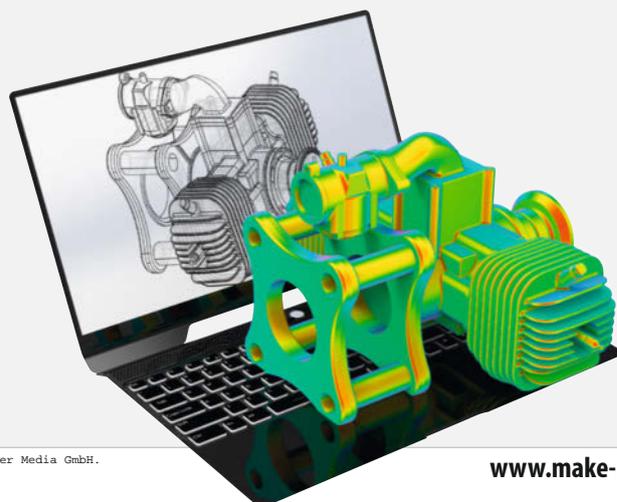
Super8-Hack

- ▶ Digital filmen mit Vintage-Kamera
- ▶ Raspi-Cam als Filmkassette
- ▶ Videos sofort im Browser schauen



3D-Konstruktion

- ▶ Test: Gratis-CAD für Maker
- ▶ Mehrfarbige 3D-Drucke entwerfen
- ▶ Genial: Trick für Senklöcher



4/22
28.7.2022
CH CHF 25.80
AT 14,20
Benelux 15,20
€ 12,90



Größte DIY-Technik- Messe

Hannover

Maker Faire®

Das Format für
Innovation und
Macherkultur

10.–11. Sept.

Hannover Congress Centrum

maker-faire.de

Jetzt 20% Rabatt
mit dem Gutscheincode
MAKE422 sichern!

gültig bis zum 21.08.2022

Wolkiges Angebot

Vor gut zehn Jahren zeigte der CAD-Hersteller *Autodesk* plötzlich ein Herz für Maker und brachte unter dem Namen *123D* eine ganze Latte von Apps für Computer, Smartphones und Tablets heraus, mit denen man nach Herzenslust in 3D konstruieren, modellieren, scannen und sogar elektronische Schaltungen entwickeln konnte. Der Clou: Obwohl hinter all den Apps professionelle Softwaretechnik stand, war das Angebot nicht nur durchweg kostenlos, sondern konsequent so klar gestaltet, dass der Einstieg Hobbyanwendern leicht fiel.

Doch 2017 war überraschend Schluss mit *123D*. Zum Ausgleich bekommen Maker kostenlos das Cloud-gestützte Profi-CAD-Werkzeug *Fusion 360* von Autodesk angeboten. Anders als bei den *123D*-Apps gibt es von *Fusion 360* auch eine Vollversion, im kostenpflichtigen Abo. Die kann natürlich mehr als die Gratis-Ausgabe. Wo aber die Unterschiede im einzelnen liegen, das beschäftigt uns seit Jahren. Es ändert sich und die Aussagen von Autodesk bleiben oft wolkig, wenn es überhaupt welche gibt. Die klaren Zeiten von *123D* sind jedenfalls vorbei.

Anfangs musste man sich für einen *Fusion-360*-Testzugang registrieren und dann versichern, dass man entweder Hobby-Anwender ist oder weniger als 100.000 Dollar Jahresumsatz macht. Dann entpuppte sich dieser Gratis-Zugang als kostenloses „Abo“, das jährlich verlängert werden musste. Später kündigte Autodesk ungewöhnlich klar an, seinen nicht zahlenden Nutzern diverse *Fusion*-Funktionen wie den Export von technischen Zeichnungen als DXF oder CAM-Werkzeuge für CNC-Fräsen mit mehr als vier Achsen zu sperren – das gilt bis heute.

Unterdessen war mein persönliches „Abo“ mal wieder zu Ende, pardon: Mein „Testzeitraum“ sei abgelaufen, so nannte es *Fusion* (ich hatte das bisher irgendwie anders verstanden). Alle Versuche, meinen Make-Zugang über die Autodesk-Webseite oder aus der

Software heraus zu verlängern, schlugen fehl. Da ich auf der Homepage des Herstellers zwischenzeitlich auch nichts mehr über „*Fusion 360* für Privatanwender“ finden konnte, war ich fest davon überzeugt, dass auch dieses Gratis-Angebot von Autodesk klammheimlich den Weg von *123D* genommen hätte ... und alle, die noch vom „kostenlosen *Fusion*“ sprechen, bloß einer zeitlich begrenzten Testversion aufsitzen und das noch nicht geschnallt haben. Deshalb, ich gebe es zu, wäre die Software um ein Haar nicht in unseren Gratis-CAD-Software-Test ab Seite 76 aufgenommen worden.

Glücklicherweise klärte es sich noch rechtzeitig: Doch, man kann *Fusion 360* als Maker weiter gratis nutzen (und auch mein Account funktioniert jetzt wieder). Das Angebot ist auf der Produktwebseite inzwischen gut aufzufinden und es gibt dort auch eine ausführliche Liste der Einschränkungen. Die scheint aber auch nicht ganz korrekt zu sein, denn dort fehlt etwa bei den unterstützten Importformaten STL und DXF ist verummmt als „dxfs“ aufgeführt. Und zur Nutzungsdauer heißt es an einer Stelle, die Software sei „für berechnete Anwender drei Jahre kostenlos für nichtkommerzielle Projekte nutzbar“. Woanders steht, sie sei „eine begrenzte kostenlose Version, die grundlegende Funktionen enthält und die jeweils für drei Jahre verlängert werden kann.“ Nee, Autodesk, klare Kommunikation ist was anderes. Oder macht ihr das bewusst alles so wolkig, weil ihr hofft, dass sich ein Gratis-Nutzer auf einmal in einer „abgelaufenen Testversion“ wiederfindet und versehentlich ein kostenpflichtiges Abo abschließt, um seine Projekte zu retten?

Spätestens in drei Jahren wissen wir, woran wir sind. Wir bleiben dran.

Peter König
Peter König

Sagen Sie uns Ihre Meinung!

mail@make-magazin.de



Energie erzeugen und sparen

Sie könnten natürlich auch die tolle Freie-Energie-Maschine aus einem YouTube-Video nachbauen oder einen garantiert blinkenden Stromspar-Stecker für die Wohnung kaufen – besser aber, Sie halten sich an unsere Anleitungen, die auch wirklich funktionieren: Der Solarpanel-Terrassentisch zum Beispiel liefert genügend Strom für ein Gaming-Notebook, und an schattigen Tagen heizt unser Wärmetauscher kalte Außenluft vor.

- 10** Solarpanel als Terrassentisch nutzen
- 20** Eigenbau-Wärmetauscher wärmt Zuluft
- 28** Warmwasserpumpen-Hack spart Energie
- 32** Solar-Versorgung für Wetterstation

Inhalt

Projekte

Ein unbemerkter Stromausfall kann böse Folgen haben: Vor nicht gestarteten Server-Diensten oder Tiefkühlpizzen mit merkwürdigem Geschmack warnt unser Stromausfall-Monitor. Eher dem Homo ludens gewidmet sind dagegen die Laufschrift-Tafel für Drohnen und Ballons sowie die Dartscheiben-Luxus-Beleuchtung.

- 52** Fliegende Laufschrift
- 58** Dartscheiben-Flutlicht
- 62** Stromausfall-Monitor



- 3** Editorial
- 6** Leserforum
- 8** In eigener Sache: Sonderheft Raspi Pico
- 9** Test: Pi Pico mit WLAN
- 10** Projekt: Der Solartisch und die grüne Steckdose
- 20** Projekt: Eigenbau-Wärmetauscher für Taupunkt-Lüftung
- 28** Projekt: Warmwasser-Sparpumpe
- 32** Projekt: Solar-Wetterstation
- 39** Maker Faire Hannover
- 40** Werkstattberichte: Neues aus der Maker-Szene, Comic
- 42** Projekt: Raspi-Super8
- 52** Projekt: Airsign – Botschaften am Himmel
- 58** Projekt: Dartscheiben-Flutlicht
- 62** Projekt: Stromausfall-Monitor mit dem Pico
- 68** Know-how: Blaue Pille für Arduino-Aufsteiger
- 76** Test: Gratis-3D-CAD für Maker

Know-how

Wer seine Projekte gern online vorstellt, sorgt am besten schon beim Bau für anschauliche Fotos oder Videos – da sind unsere Tipps zu einem kleinen Fotostudio durchaus willkommen. Gern helfen wir auch beim Umstieg auf eine leistungsfähigere Plattform, wenn Ihre stetig gewachsene Firmware nun den kleinen Arduino überfordert.

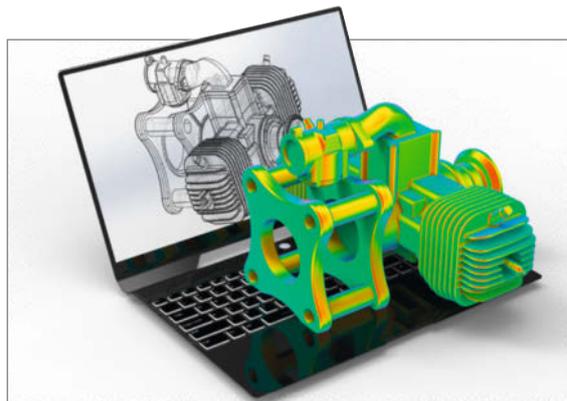
- 68 Arduino-Alternative mit STM32
- 100 Maker-Fotostudio einrichten
- 114 Tricks für Resin-3D-Drucker



3D-Konstruktion

Immer noch nicht die richtige Software für 3D-Konstruktionen gefunden? Unsere Auswahl kostenloser CAD-Programme hilft bestimmt. Wir zeigen auch, wie man ohne großen Aufwand farbige 3D-Objekte entwirft oder Senkungen und Löcher so anlegt, dass sie ohne lästige und schwer zu entfernende Stützen auskommen.

- 76 Gratis-CAD für Maker
- 86 Mehrfarbige 3D-Drucke entwerfen
- 92 Genial: Trick für Senklöcher ohne Stützen



3DConcepts/Shutterstock.com

- 86 **Know-how: Multicolor-Konstruieren für 3D-Druck**
- 92 **Workshop: Hängende Löcher ohne Stützen im 3D-Druck**
- 94 Community-Projekte: Fledermaus-Scanner goes LoRaWAN
- 96 Community-Projekte: Lastenanhänger TrudeTrailer
- 98 Community-Projekte: Masking-Tape-Etikettendrucker
- 100 **Show and Tell: Im Maker-Fotostudio**
- 108 Reportage: Making-Of Posteule
- 114 **Know-how: Tricks für MSLA-Resin-Druck**
- 118 Kurzvorstellungen: XIAO ESP32C3, Programmieren lernen mit Scratch, Sno-M2-FPGA, Modulare Synthesizer mit VCR-Rack 2, AVR-IoT Cellular Mini, BeagleBone AI-64
- 122 Impressum/Nachgefragt

Themen von der Titelseite sind rot gesetzt.

Solarpanels Titelseite: mstanley/Shutterstock.com

Super8-Hack

Auf jedem Flohmarkt findet man sie: Wahre Wunderwerke an Optik und Feinmechanik, die fast verschenkt werden, weil die passenden Filme fehlen. Unsere digitale Filmkassette beschert Super8-Kameras ein neues Leben – ein Raspi-Kamera-Modul als Bildsensor ersetzt nun den Super8-Film und liefert Videos in DVD-Qualität.



- 42 Raspi-Super8-Hack

Leserforum

Nach meinem Geschmack

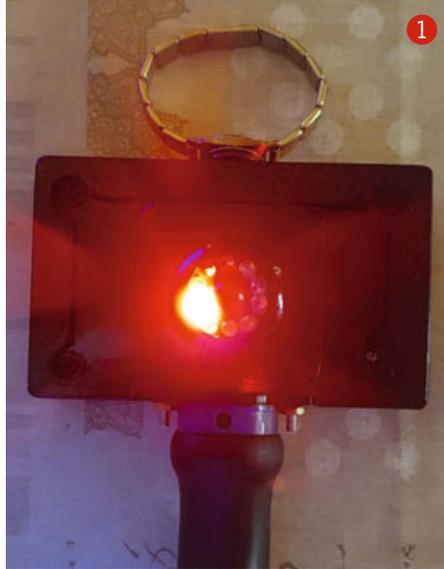
Eigenbau-Geigerzähler mit Alarm,
Make 3/22, S. 8

Das war mal wieder eine Make nach meinem Geschmack! Mein Geigerdetektor mit dem gleichen Bausatz, den ich vor circa einem Jahr gebastelt habe, verwendet ein CD4017-Zähl-IC, um die 10 LEDs anzusteuern.

Auf Bild 1 sieht man ihn in Funktion. Dazu habe ich als radioaktive Quelle die alte Armbanduhr meiner Oma verwendet (Ziffernblatt: Phosphor). Auch sieht man meine ersten Gehversuche mit 3D-Druck ... Der Griff enthält die Batterien und stammt von einem irreparablen Endoskop, der runde Batteriehalter von einer Taschenlampe 2. Da ich bemerkt habe, dass das Gehäuse eine gewisse Abschirmung darstellt, habe ich vor dem Zählrohr eine Ausparung geschaffen 3.

Im Normalfall ist nur ab und wann mal ein Ticker zu sehen. Wenn die LEDs wie blöd im Kreis rennen, wird's Zeit, dass man sich vom Acker macht ...

Wolf



der Schutzanzug oder gar Bunkerbau? Was ist in Sie gefahren? Wollen Sie echt mit der Angst vor einer atomaren Katastrophe oder gar einem Atomkrieg punkten? Was glauben Sie, was mir ein Geigerzähler nutzt, wenn es zum Atomkrieg kommt? Ich bin jetzt 64 und habe trotz kaltem Krieg, trotz Tschernobyl, trotz Fukushima so ein Teil nie gebraucht, denn es ist das unnützlichste Teil, das ein normaler Bürger braucht! Also bitte konzentrieren Sie sich auf die wirklich wichtigen Dinge, wie Umwelt und chemische Analyse von Inhaltsstoffen! Wir werden weitere Infektionen bekommen!

Erfinden Sie eine Dauermaske oder einen effektiven Luftfilter! Aber bitte lassen Sie mich in Zukunft mit Atomkrieg in Ruhe!

Matthias Zwerschke

Unnützlichstes Teil

Als ich gestern die Onlineausgabe der Make 3/22 in der App sah, traute ich meinen Augen nicht! Halbseitig wird der Bau eines Geigerzählers beworben! Kommt in Make 4/22 jetzt

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.

Python vermisst

Make 3/22 generell

Ich habe gerade das neueste Heft gelesen und bin sehr enttäuscht. Warum wird für alle Projekte des Heftes nur C++ genommen? Mein Tipp: Man könnte doch auch beide Sprachen (Python und C++) als Code bereitstellen. Wir Maker sind keine Profi-Programmierer, Python ist sicherlich verbreiteter als C++. Ich lerne doch nicht noch eine Sprache, nur damit ich da mithalten kann. Sicher könnte man die fertigen Programme einfach flashen, da sie ja so lauffähig sind. Ich möchte aber meistens Code als Vorlage für eigene Projekte nutzen, was so natürlich nicht geht.

Michael Petermeier

Viele unserer Autoren programmieren ihre Mikrocontroller-Projekte nur in einer Sprache.



eigentlich ausschließlich C/C++-Projekte, weil alles auf Arduino beruhte und es bis 2014 kaum Projekte für den Pi gab.

Warnschilder gesucht

Leserforum, Make 3/22, S. 6

Für den Aufruf nach Bildern von alten Warnschildern aus der Make 3/22 habe ich dieses Schild für Sie ⁴, was eigentlich in keiner Werkstatt fehlen sollte. Die Quelle ist eine alte Werkszeitschrift aus den 1930er Jahren. Ein Urheber wurde leider nicht genannt. Und falls es an älteren Kollegen und Kolleginnen fehlen sollte, die die alten Schriften lesen können: *Setz die Schutzbrille auf!*

Vielleicht wäre ein Stickerbogen als Beilage in der Make oder der c't ja eine gute Idee, wenn genug Schilder zusammenkommen.

Florian Franzmann

Wahrer Make-Geist

Digitaler Bilderrahmen mit ePaper, Make 2/22, S. 12

Ich muss jetzt mal unbedingt etwas loswerden. Nachdem ich meine argen Probleme beim Zusammenbau des digitalen Bilderrahmens hatte, konnte ich über das Forum Kontakt zu Ihrem Autor Florian Sommer herstellen. So einen tollen Support habe ich noch nie erlebt. Er hat sich persönlich auch im telefonischen Gespräch in der gemeinsamen Fehleranalyse um mein Problem gekümmert und es auch erfolgreich beseitigt. Das ist der wahre Make-Geist, den er aktiv vertritt und praktiziert. Er macht das sicherlich in seiner Freizeit und hilft aus Überzeugung. Schön, ihn kennengelernt zu haben. Ich möchte mich auf diesem Wege auch bei Ihnen bei der Auswahl Ihrer Autoren bedanken und würde mich freuen, wenn die Make-Redaktion auf diesem Wege auch ein herzliches Dankeschön an den Autor sendet.

Jörg Brüsckhe

Vielen lieben Dank für das Lob! Das freut alle Beteiligten sehr und ist ein großer Ansporn für die Zukunft.

Und das ist eben meistens C/C++. Auf dem Raspberry Pi sieht das anders aus und da haben wir zuletzt die Animatronische Posteule in Make 1/22 und 2/22 in Python programmiert. Wir können aus Ressourcengründen leider keine Listings in beiden Sprachen bringen. Im übrigen hatten wir am Anfang von c't Hacks die ersten Jahre

Vielen Dank für dieses besonders schöne Schild! Und ja, wir haben auch schon überlegt, aus den gesammelten Schildern vielleicht wieder ein Poster zu machen, falls genügend zusammenkommen. Dazu brauchen wir aber noch ein paar mehr Einsendungen mit Bild an mail@make-magazin.de.

Programmieren mit dem Pi Pico

Mit dem *Make Pi Pico Special* steigen Sie ein in die Welt der Programmierung von RP2040-ARM-Mikrocontrollern. Wir zeigen in dem 64-seitigen Special, welche Entwicklungsumgebungen es für den Raspberry Pi Pico gibt, wie man sie installiert und wie man sie nutzt. Das Heft ist nur im Bundle mit einem Pico für zusammen 24,95 Euro erhältlich.

Anhand von Praxisbeispielen in Basic, MicroPython und C++ erklärt das Heft, wie

man Sensoren abfragt, OLED-Displays nutzt und serielle Schnittstellen steuert. Mit der *PicoVGA-Bibliothek* kann man sogar VGA-Monitore zur Ausgabe nutzen. Als Highlight beschreibt ein Artikel, wie man Retro-Spiele wie *Pac-Man* und *Doom* auf dem Pico installiert. Und: In Assembler programmieren wir die speziellen PIOs des Prozessors, um eine LED-Kette unabhängig von der CPU zu steuern.



Workshop

Das ist technisch richtig und liegt darin, dass die VGA auf dem Raspberry Pi Pico...
 Das Heft zeigt alle Bauteile, die Sie benötigen, um die Schaltung zu bauen. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können.

DIY-DAC

Die Schaltung implementiert die Digital-to-Analog-Converter (DAC) und eine Treiberstufe für den Lautsprecher. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können.

Workshop

Wie man ein VGA-Signal auf dem Raspberry Pi Pico erzeugt. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können.

Workshop

Wie man ein VGA-Signal auf dem Raspberry Pi Pico erzeugt. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können.

Workshop

Wie man ein VGA-Signal auf dem Raspberry Pi Pico erzeugt. Die Schaltung ist so einfach, dass Sie sie in wenigen Minuten aufbauen können.

Der erste Teil des Hefts widmet sich dem Aufbau und den Hardware-Komponenten des Pico und stellt Entwicklungsumgebungen vor. Im zweiten Teil wird es dann praktisch: mit Temperatursensoren, OLED-Displays und wenigen Widerständen kann man VGA-Signale erzeugen und Retrospiele auf dem Pico spielen, inklusive Sound. Alle Beispiel-Listings stehen zum Download und Nachvollziehen bereit. Das gedruckte Heft mit dem Pico gibt es exklusiv im Heise Shop versandkostenfrei zu bestellen.

Alles zum Artikel im Web unter make-magazin.de/xgg9

Blick ins Heft: Mit wenigen Bauteilen kann man sogar VGA-Monitore ansteuern und Pac-Man spielen.

Hier finden Sie das Make-Magazin

Die Make im Abo zu lesen, ist zwar günstiger als im Einzelverkauf – wer sich aber nicht binden will, findet uns auch im Kiosk, allerdings nicht in jedem. Welche Zeitschriftenhändler und Supermärkte uns führen, finden Sie schnell auf mykiosk.com heraus. Einfach die eigene Postleitzahl und *Make:* (ja, mit Doppelpunkt) eingeben und auf einer Karte erscheinen die Händler im Suchgebiet.

Ist der Weg zu weit, kann man alternativ die Make als gedrucktes Heft im heise Shop bestellen oder als PDF sofort downloaden. Wer gerne auf dem Tablet oder Smartphone liest, kann sich die Make-App installieren und darin das Heft kaufen und wahlweise als PDF oder auf HTML-Seiten lesen.

Alles zum Artikel im Web unter make-magazin.de/xgg9

ENG | FRA | ITA | TUR

MYKIOSK.com

DER SCHNELLSTE WEG ZU MEINER ZEITSCHRIFT!

30625 MAKE: HÄNDLER FINDEN

VERKAUFSTELLEN IN IHRER NÄHE

COMPUTER MAKE: PREIS: 12,90 € AUSGABE: 3/22 ERSCHEINEN: 02.06.2022 ERSCHEINT: 2-monatlich

Pi Pico mit WLAN

Unverhofft kommt oft, zumindest bei Produktvorstellungen der Raspberry Pi Foundation: Ende Juni verkündete sie plötzlich die Verfügbarkeit des Pi Pico W. Mit einem ersten Sample haben wir schon mal rumprobiert.

von Daniel Bachfeld

Alles zum Artikel
im Web unter
make-magazin.de/xv93

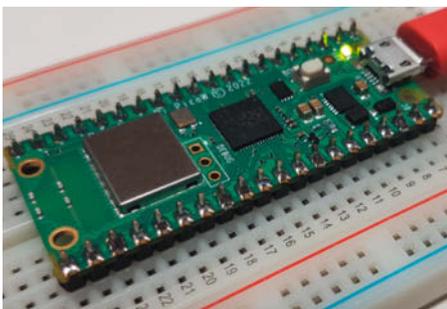


Der Pi Pico W ist im Kern ein herkömmlicher Pi Pico auf Grundlage des ARM-SoC RP2040, dem ein Funk-Modul von Intel zur Seite gestellt wird. Das Modul (CYW43439) ist über eine SPI-Schnittstelle (CLK, Data, CS) an den SoC angebunden und arbeitet im WLAN auf 2,4GHz. Damit lassen sich mit den Code-Beispielen (iperf) der offiziellen und bereits aktualisierten Pico-C-SDK Übertragungsraten über 8MBit/s erreichen, was für die meisten Mikrocontroller-Projekte ausreichen sollte.

Der ebenfalls im neuen Modul vorhandene Bluetooth-5.2-Transceiver wird (noch) nicht benutzt, sehr wahrscheinlich wird er in einer künftigen Firmware-Version unterstützt. Laut Schaltplan wird er vom gleichen GPIO-Pin aktiviert, der auch die WLAN-Einheit anschaltet.

Immerhin hat die Raspberry Pi Foundation pünktlich zum Erscheinen des W-Modells neue Firmwares für MicroPython veröffentlicht, die allerdings (zum Redaktionsschluss) noch die Kennzeichnung *unstable* trugen. Unter MicroPython ist das Einbinden des WLANs mit wenigen Codezeilen erledigt (siehe Kasten). In ersten Tests bockte die WLAN-Firmware jedoch mitunter und ließ über die serielle Schnittstelle keine Verbindung mehr zu. Erst nach dem kompletten Löschen des Flash mit der Firmware *nuke_flash.uf2* und anschließender Neuinstallation der MicroPython-Firmware lief der Pico W wieder, wie er sollte.

Unter C/C++ unterstützt ein zusätzlicher Treiber (*cyw43_driver*) die Steuerung des



Passt auf jedes Breadboard: Der Pi Pico W hat den gleichen Formfaktor wie das Modell ohne WLAN und ist eine interessante Alternative zu Boards mit Modulen von Espressif.

Funk-Moduls, und der hinzugekommene Netzwerkstack lwIP (lightweight TCP/IP Stack) übernimmt die Übertragung der IP-Pakete. Eigentlich ist der Einsatz von lwIP in kommerziellen Produkten laut Lizenz nicht erlaubt, laut Foundation haben die Entwickler für den Pico aber eine Ausnahme gemacht. Ob sich das auf den Pico als Produkt der Foundation oder auf den Einsatz in kommerziellen Projekten durch Anwender bezieht, ist noch unklar.

Der CYW-Treiber und lwIP benötigen regelmäßig ein Stück Prozessorzeit, um diverse interne Arbeiten zu erledigen. Das sollte mit den zwei Kernen des RP2040 zwar kein Problem sein, man sollte nur nicht beide Kerne exklusiv für seine Anwendung blockieren. Ähnliches kennt man bereits von Boards mit ESP-WLAN-Modulen.

In unseren Tests lag der Stromverbrauch bei aktivem WLAN im Schnitt 20mA über dem eines normalen Pico, der mit aufgespieltem MicroPython 30 bis 40mA verbraucht. Beim Verbindungsaufbau und beim Datenaustausch ging der Stromverbrauch in Spitzen sogar auf 400mA.

Interna

Leider sorgt der Anschluss des Funk-Moduls an den RP2040 für ein paar Inkompatibilitäten zum Vorgänger, was sich eventuell in anders reagierender Software äußert: Die GPIOs 24, 25 und 29 dienten vorher der Kontrolle des Spannungsreglers sowie der Steuerung der Onboard-LED. Die Pins hängen nun aber an den SPI-Pins des Moduls. Als Ersatz dienen drei im Funkmodul verfügbare I/O-Pins *WL_GPIO0* (für LED), *WL_GPIO1* (PS) und *WL_GPIO2* (VSYS). Um sie anzusprechen oder abzufragen, gibt es zwar Funktionen in MicroPython und in der Pico-C-SDK, allerdings lassen sie sich nicht von den speziellen PIOs ansprechen. Projekte wie der Strommonitor auf Seite 62 funktionieren daher erstmal nur mit herkömmlichen Picos!

Im CYW43439 stecken zwei ARM-Rechenkern: ein Cortex-M4 im WLAN-Teil und ein Cortex-M3 im nicht genutzten Bluetooth-Teil, die sich jeweils um die Funkkommunikation kümmern. Leider ist deren Firmware in ROMs abgelegt, sodass sich ihre weit über jene der

WLAN mit MicroPython

```
import network
wlan = network.WLAN(network.STA_IF)
wlan.active(True)
wlan.connect('ssid', 'pw')
wlan.ifconfig()
```

Cortex-M0+-Kerne liegende Rechenkraft nicht für eigene Zwecke nutzen lässt – es sein denn, jemand findet einen anderen Weg. Das erinnert ein wenig an den ESP8266, der Anfangs nur als WLAN-Erweiterung für den Arduino diente, aber einen wesentlich leistungsfähigeren Prozessor an Bord hatte. Mittlerweile setzt man gleich ganz auf den ESP, etwa auf Wemos-Boards oder ähnlichen.

Apropos ESP: Gegenüber dem ESP32 hat der Pico W nur 2MB Flash und mit 133MHz nur die halbe Taktfrequenz. Prinzipiell lässt sich der Pico locker übertakten. Welche Implikationen das auf die Kommunikation mit dem Funkmodul hat, müssen Tests zeigen. Einer der großen Vorteile des RP2040 ist, dass er sich als USB-Host einsetzen und mit externen Geräten kombinieren lässt.

Game Changer?

Zwar gibt es bereits alternative Boards mit RP2040 und WLAN-Modul wie das Arduino Nano RP2040 Connect, aber mit einem Straßenpreis von aktuell 7 Euro macht die Raspberry Pi Foundation Boards mit ESP8266 und ESP32 ordentlich Konkurrenz. Dazu kommt das Ökosystem der Foundation mit seinen vielen Entwicklern, Contributoren und der riesigen Community, in der so gut wie kein Problem ungehört oder ungelöst bleibt. Und: Mit dem RP2040 setzt man auf ARM und eine zukunftsfähige Basis statt auf proprietäre Architekturen wie Tensilica-CPU's. Mit dem ESP32-C3 ist Expressif zwar selber schon auf die offene Architektur RISC-V umgeschwenkt, aber das ist im Moment eher noch was für professionelle Anwender.

Ich jedenfalls werde künftig mehr Projekte mit dem neuen Pico umsetzen, allein schon deshalb, weil er ohne Probleme auf Breadboards passt. —*dab*

Der Solartisch und die grüne Steckdose

Eigentlich sollte nur ein Solarpanel aufgestellt werden, mangels Platz wurde daraus der Solartisch. Damit der gewonnene Strom auch am Abend nutzbar ist, kam eine Pufferbatterie dazu. Nur wie bindet man das alles möglichst ohne Elektriker ins Hausstromnetz ein? Wer verbraucht eigentlich den ganzen Strom? Am Ende entstand daraus die Wurzel unserer eigenen kleinen Energiewende: Grüner Strom wird erzeugt, gemessen und die Verbraucher werden entsprechend gesteuert.

von Guido Burger und Klaus-Uwe Gollmer



Woher kommt eigentlich unser Strom? Aus der Steckdose, ja klar! Möglicherweise haben wir einen Vertrag mit dem Energieversorger, der sicherstellt, dass wir *grünen* Strom eingespeist bekommen. Damit ist doch alles gut! Wirklich?

Nein, denn auch ein Ökostromvertrag bedeutet nicht, dass wir dann nur erneuerbare Energie aus der Steckdose geliefert bekommen. Wir müssen viel mehr tun, um der Klimakatastrophe und der durch den Ukraine-Krieg bedingten Energiekrise zu begegnen. Unser Ziel: Die Grundlast weitestgehend über eigenproduzierte erneuerbare Energie z. B. vom Solartisch, Balkonkraftwerk oder Solarpanel auf dem Schuppen abdecken. Zusätzlich die selbst initiierten Verbräuche möglichst in Zeiten legen, in denen der aktuelle Strommix viel regenerative Energie aufweist. Dadurch reduzieren wir unseren CO₂-Fingerabdruck und sparen gleichzeitig in der landesweiten Bilanz erhebliche Mengen an fossilen Brennstoffen. Wichtigste Voraussetzung dazu: Wir müssen mehr über unseren Energiebedarf wissen.

Wir starteten daher mit einer nicht repräsentativen Umfrage **1**, wer seine aktuelle Grundlast im Haushalt kennt. Die Grundlast sind die immer laufenden Verbraucher, die damit für einen nicht unerheblichen, gleichbleibenden Stromverbrauch über das ganze Jahr sorgen. Es sind oftmals viele kleine und gern übersehene Dinge: Steckernetzteile, Fernseher im Stand-by, die Umwälzpumpe der Heizung, automatische Ventile der Fußbodenheizung, aber auch der Kühlschrank oder die Gefriertruhe. Die Antwort ist erschreckend: Nur 27% der Teilnehmer an der Umfrage kennen nach ihrer Aussage die Grundlast in ihrem Haushalt. Wenn wir aber die eigene Grundlast nicht kennen, wie wollen wir dann Maßnahmen ergreifen?

Los gehts: Analyse des Energiebedarfs

Idealerweise sollte uns ein intelligenter Stromzähler des Versorgers detaillierte Informatio-



Arbeiten an 230V!

Wer seinen Solartisch ans Stromnetz anschließen möchte, muss die notwendigen gesetzlichen Bestimmungen und Vorgaben seines Netzbetreibers beachten. Bitte einen Elektriker beauftragen, wenn keine Fachkenntnisse und die entsprechenden Berechtigungen vorhanden sind. Hier verweisen wir auch auf einschlägige Seiten der Netzbetreiber und auf unseren Balkonsolar-Artikel. Alle Links erreichen Sie über die Kurzinfor.

Kurzinfor

- » Schrittweise zur eigenen Energiewende
- » Solartisch bauen als Insel- oder Netzeinspeisung
- » IoT-Steuerung und Überwachung von Stromverbrauch und -verteilung

Checkliste



Zeitaufwand:
1-8 Stunden je nach Ausbaustufe



Kosten:
50-900 Euro je nach Ausbaustufe

Material

» Ausführliche Materialliste Online (QR-Code oder Kurzlink)

Werkzeug

» Ausführliche Liste Online (QR-Code oder Kurzlink)

Mehr zum Thema

- » Sebastian Müller, Photovoltaik auf dem Balkon, Make 1/22, S. 38
- » Carsten Wartmann, Internet-of-Things-Dienste für Maker, Make 3/21, S. 32
- » Guido Burger, Dr. Richard Fix und Klaus-Uwe Gollmer, Der CO₂-Warner für die Schule, Make 5/20, S. 10



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



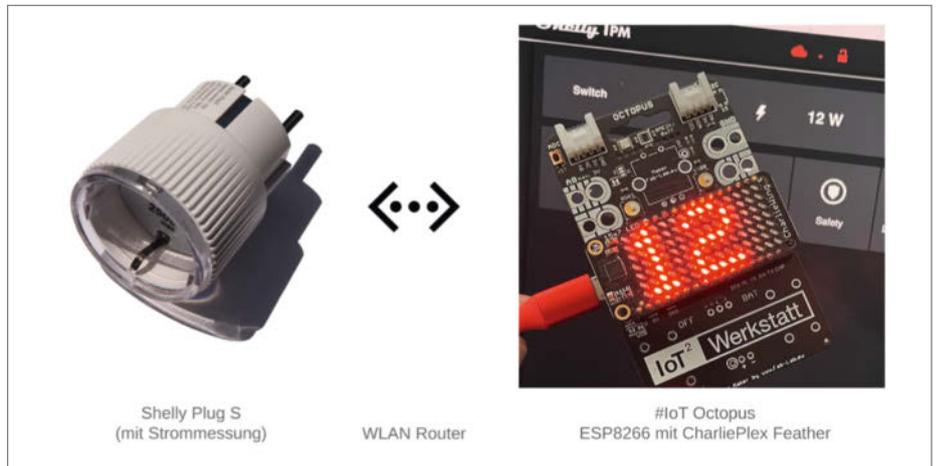
nen zu Zeiten und Verbräuchen liefern. Aber so ein *Smart Meter* ist nur für Haushalte ab 6000 kWh vorgeschrieben und wird aktuell von den meisten Anbietern nicht kostenfrei nachgerüstet. Viele Haushalte werden deshalb noch einen *Ferraris*-Zähler mit Drehscheibe vorfinden. Und selbst bei einem Smart Meter ist nicht sichergestellt, dass man auf seine Daten auch zugreifen kann, da diese ausschließlich zum Energieversorger gehen und dann nicht im Internet bereitgestellt werden. Die sogenannte *S0*-Datenschnittstelle am Zähler selbst wird vom Energieversorger oft deaktiviert.

Hände hoch! Wer kennt seine aktuelle Grundlast (nicht Spitzen durch Waschmaschine etc, sondern was so den ganzen Tag läuft) in der Wohnung/Haus? #Energiewende

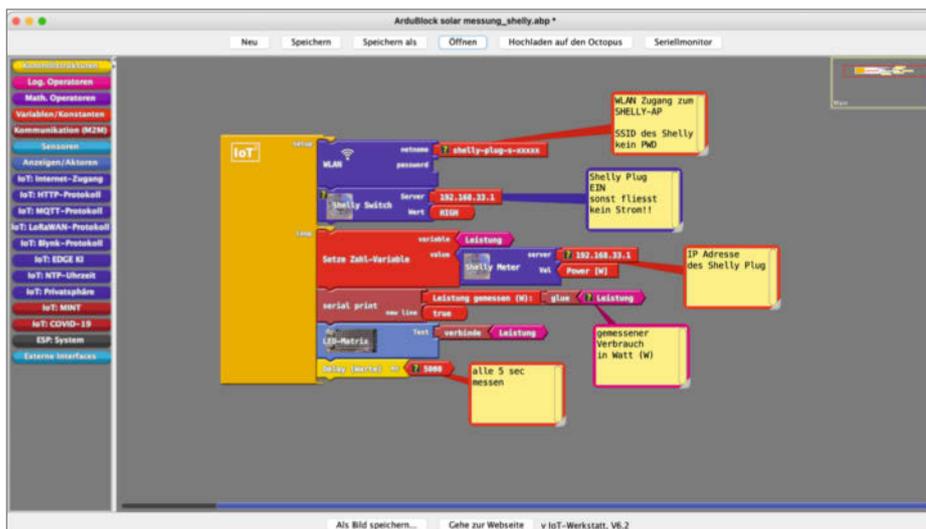
keine Ahnung	48 %
ungefähr	25 %
ja, genau	27 %

204 Votes - Endergebnisse

1 Unsere Twitter-Umfrage



2 Shelly Plug S im Heim-WLAN, verbunden mit dem IoT Octopus



3 IoT²-Werkstatt: Der IoT-Octopus holt mit einem Block die Daten vom Shelly Plug S.

Hier ist dann Eigeninitiative erforderlich. Eine erste grobe Abschätzung ergibt sich mit wenig Aufwand: Wir gehen abends vor dem Zubettgehen und morgens nach dem Aufwachen in den Keller und lesen den Zählerstand ab. Aus der Differenz berechnen wir den nächtlichen Energieverbrauch in kWh. Den teilen wir durch die Zeitdifferenz in Stunden und bekommen so eine Idee für die mittlere Grundlast in kW.

Alternativ können wir hinters Sofa oder hinter den Schrank kriechen und mit einem einfachen Energiemessgerät (Zwischenstecker) die Leistungsaufnahme der einzelnen Geräte notieren. Spätestens beim Kühlschrank werden wir feststellen, dass die Grundlast in Abhängigkeit vom Kompressorbetrieb zeitlich doch sehr stark schwanken kann. Eine einfache Anzeige wird dem also nicht wirklich gerecht.

Hier kommt der erste Einsatz unseres Internets der Dinge (*Internet of Things*, IoT): Smarte

Zwischenstecker mit WLAN-Funktion ermöglichen die automatisierte Speicherung und spätere Auswertung der Energiedaten in der (auch lokalen) Cloud.

IoT²-Werkstatt und Messaufbau

Schon bei der letzten großen Krise, der COVID19-Pandemie, haben wir mit dem Selbstbau von CO₂-Ampeln gezeigt, wie uns IoT bei der Analyse des Lüftungsverhaltens helfen kann und wie wir gleichzeitig noch viel über die MINT-Zusammenhänge der Pandemie lernen können. Zum Einsatz kommt dabei die *IoT²-Werkstatt* vom Umwelt-Campus Birkenfeld der Hochschule Trier, mit ihrer grafischen Programmieroberfläche und dem *IoT-Octopus* (auf ESP8266-Basis). Die Software finden Sie über die Links in der Kurzinfor: eine aktuelle, portable Arduino IDE mit den notwendigen Erweiterungen und Ardublock mit unseren

grafischen IoT-Blöcken. Einfach entpacken und es geht los.

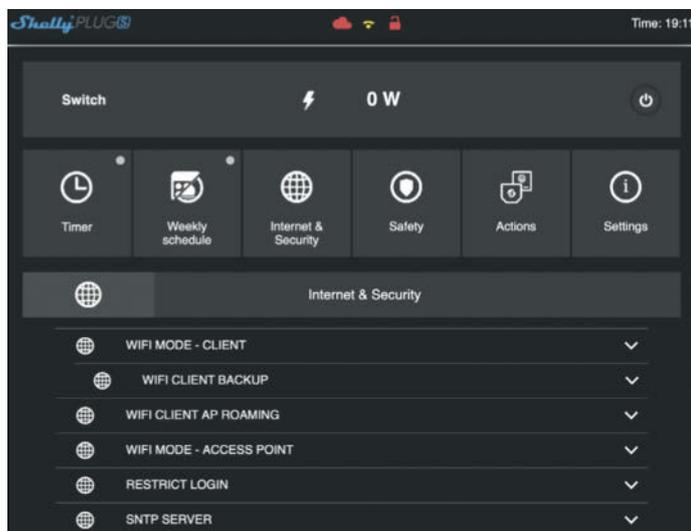
Wir benutzen den WLAN-Stecker *Shelly Plug S*, so haben wir ein mächtiges Duo 2 zur Analyse unseres häuslichen Energiebedarfs.

In der Grundkonfiguration spannt der *Shelly Plug S* ein eigenes WLAN 3 auf, in das sich unser IoT-Octopus einwählen und den aktuellen Leistungsbedarf abrufen kann. Alles ist einfach über die grafische Programmierumgebung der IoT²-Werkstatt umsetzbar.

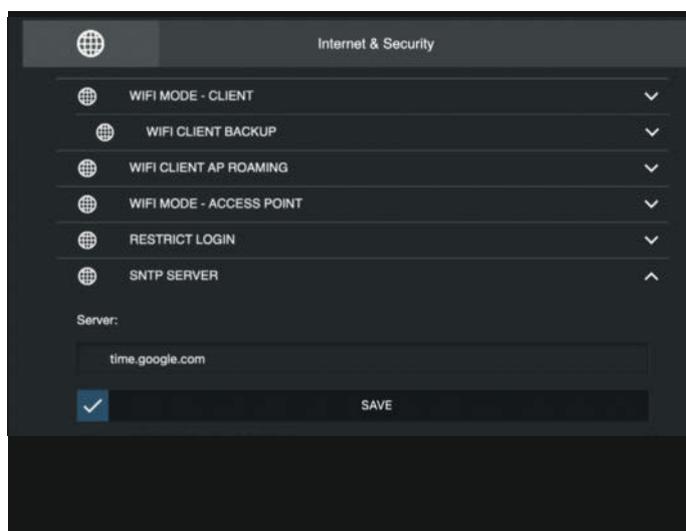
Der *Shelly Plug S* basiert seinerseits auch auf einem ESP8266 und kann sowohl die Leistung (W) als auch die Energie (Wh) messen. Für die Energiemessung ist es wichtig, dass der Stecker einen Zeitserver über den Zugang ins heimische WLAN findet. Ohne diese Verbindung (etwa in einem Gästenetzwerk) kann die Leistung nicht über die Zeit integriert werden und die Energiedaten stehen nicht zur Verfügung. Es wird dann nur die Leistung angezeigt. Die notwendigen Einstellungen können wir vornehmen, indem wir uns mit dem PC in den Shelly-Accesspoint einwählen und im Webbrowser 4 die feste IP 192.168.33.1 aufrufen. Nach der Einbindung in das Heim-WLAN ändert sich diese IP-Adresse in eine von dem Router zugewiesene und ein Zeitserver 5 kann aktiviert werden.

Dem Kühlschrank unter die Haube geschaut

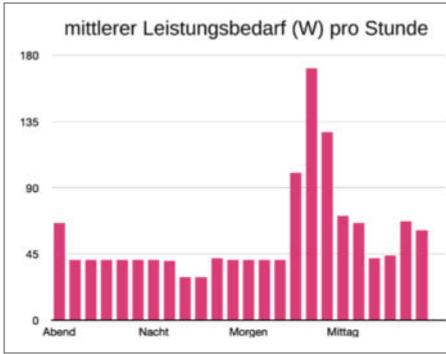
Es ranken sich so einige Fragen und Mythen um den Kühlschrank. Ist die Lampe an, wenn die Tür zu ist? Wie lange hält er die Kälte bei Stromausfall? Wir schauen mal, was unser Strommesser zu sehen bekommt. Ein *Side-by-Side*-Kühlschrank der Energieeffizienzklasse A++ benötigt ganze 1366Wh pro Tag. Bei einem Strompreis von 0,4 Euro pro kWh sind das dann $1,366 \cdot 365 \cdot 0,4 = 200$ Euro im Jahr (365 Tage).



4 Shelly-Plug-S-Zugriff über eine lokal gehostete Webseite



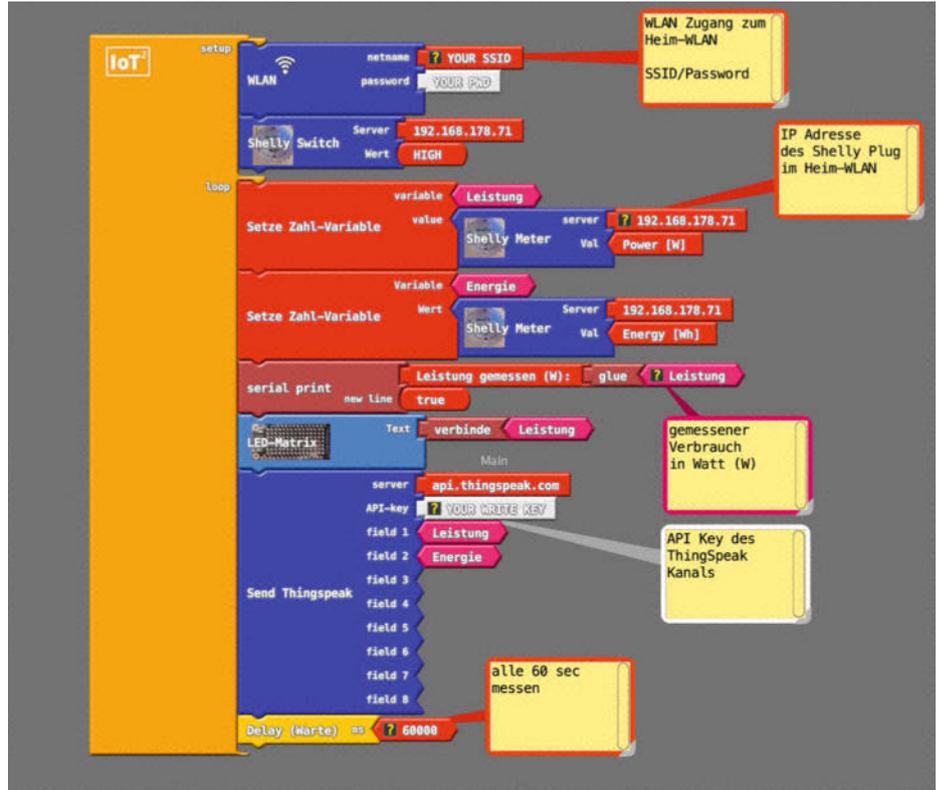
5 Eigenes Heim-WLAN und Zeitserver (default) eingetragen



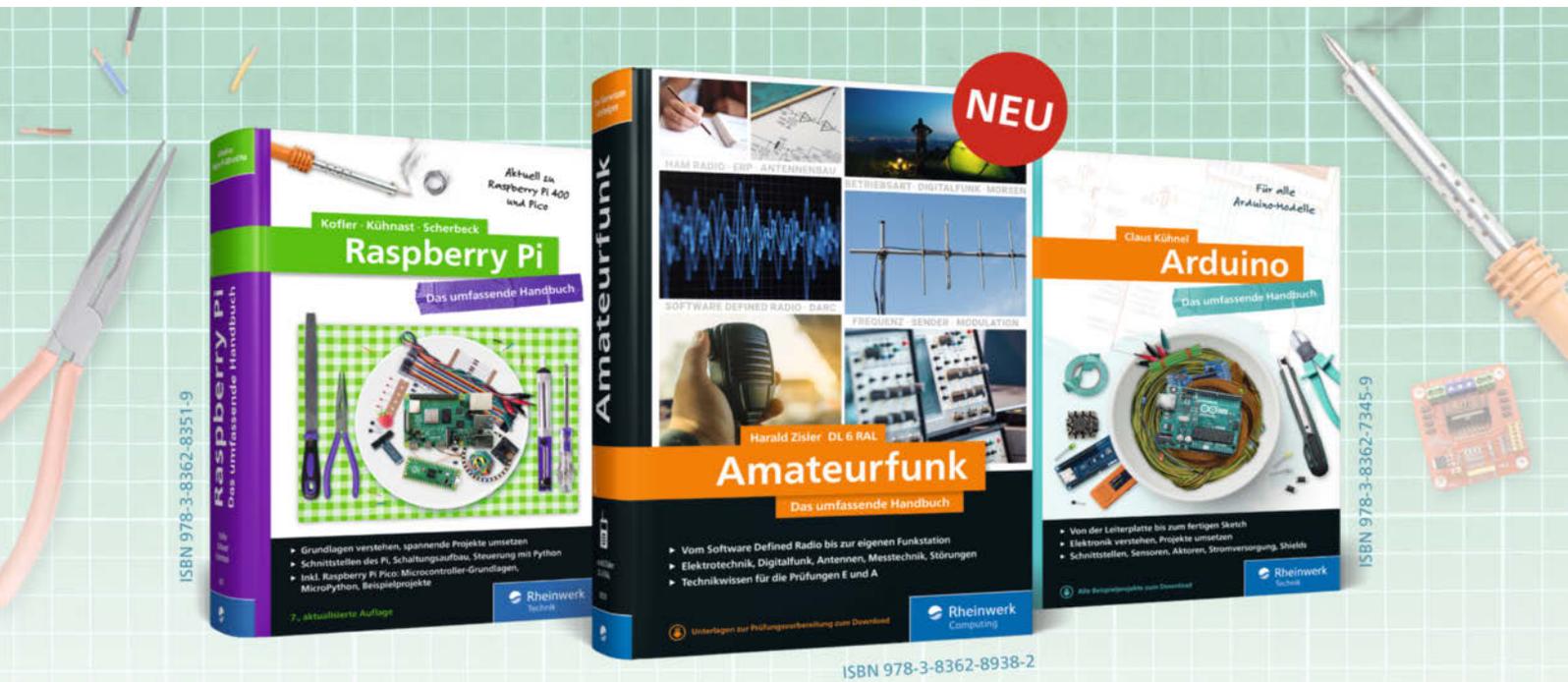
6 Mittlerer Leistungsbedarf Kühlschranks, Aufzeichnung eines Tages

Eine erste Analyse 6 des mittleren Leistungsbedarfs (W) ergibt einen minimalen Wert von 29Wh und einen groben Wert für den Standby-Betrieb von 41Wh. In der Spitze werden 171Wh gemessen. Die Messwerte werden aber über eine Stunde integriert und zeigen daher keine kurzen Lastspitzen.

Befinden sich unser IoT-Octopus und der Shelly bei der Messung im Heimnetz, so können wir die ermittelten Messergebnisse in die ThingsSpeak-Cloud übertragen und dort visualisieren. Die IoT²-Werkstatt unterstützt dabei



7 Der Leistungsbedarf wird auf ThingsSpeak gespeichert, die Shelly-IP ist hier dynamisch.

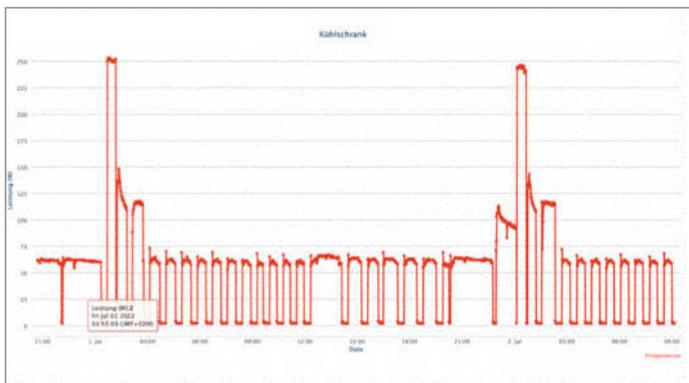


Bücher für Alles-Erfinder

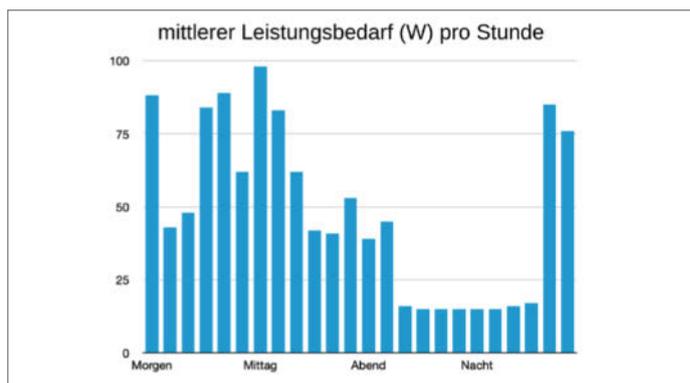
Amateurfunk, Technikprojekte mit Raspberry Pi und Arduino: Unsere Bücher machen den Einstieg in die Maker-Welt zum Vergnügen. Mit allen Grundlagen zu Linux, Programmierung und Elektrotechnik. Seitenweise Zündstoff für Ihre Ideen!

Alle Handbücher auch als E-Book und im Bundle.

www.rheinwerk-verlag.de



8 ThingsSpeak: Leistungsbedarf des Kühlschranks fein aufgelöst



9 Mittlerer Leistungsbedarf des Homeoffice für einen Tag

Wie kann ich grünen Strom bevorzugen?

Schaut man in die Statistik des Umweltbundesamtes, so konnten im Jahr 2021 rund 41% des Bedarfs an elektrischer Energie aus erneuerbaren Energiequellen gedeckt werden. Ob nun Ökostromvertrag oder konventioneller Anbieter, aus der heimischen Steckdose kommt immer der gleiche Strommix, dessen Zusammensetzung im Tagesverlauf erheblich schwanken kann. Physikalisch gibt es nämlich keinen Unterschied, ob die Ladungsträger mittels einer Windkraftanlage oder über ein thermisches Kraftwerk erzeugt wurden. Wo uns die Physik Grenzen gesetzt hat, kann das Internet der Dinge jedoch Licht in die Zusammensetzung des Strommix bringen. Bei der CO₂-Ampel ist es uns gelungen, die gefährlichen Aerosole über die CO₂-Messung quasi bunt einzufärben. Gleiches wird uns nun mit dem heimischen Strom gelingen. Wie kann das funktionieren?

Dazu verwenden wir die öffentliche Datenbasis der Bundesnetzagentur, die sich wiederum auf den *Verband Europäischer Übertragungsnetzbetreiber* stützt. Dorthin melden die verschiedenen Energieproduzenten ihre aktuellen Leistungsdaten in Echtzeit. Diese Marktdaten werden auf der Webseite *SMARD.de* visualisiert.

Hier sieht man deutlich, wie volatil Photovoltaik und Windkraft doch sind. Allein aus den tageszeitlichen Schwankungen ergeben sich Anteile zwischen 30 und 80% am Gesamtangebot im Strommix.

Wichtig für uns ist der aktuelle und prognostizierte Anteil an erneuerbarer Energie.

Gelingt es uns, die heimischen Verbräuche in Phasen des hohen Angebots zu verschieben, so können wir auch den sogenannten *Flutterstrom* sehr effizient nutzen. Wer ein Elektroauto besitzt, aber keine eigene Photovoltaik hat, der lädt vielleicht besser Mittags bei 70% Anteil an *Erneuerbarer Energie* (EE) als Nachts bei 30%.

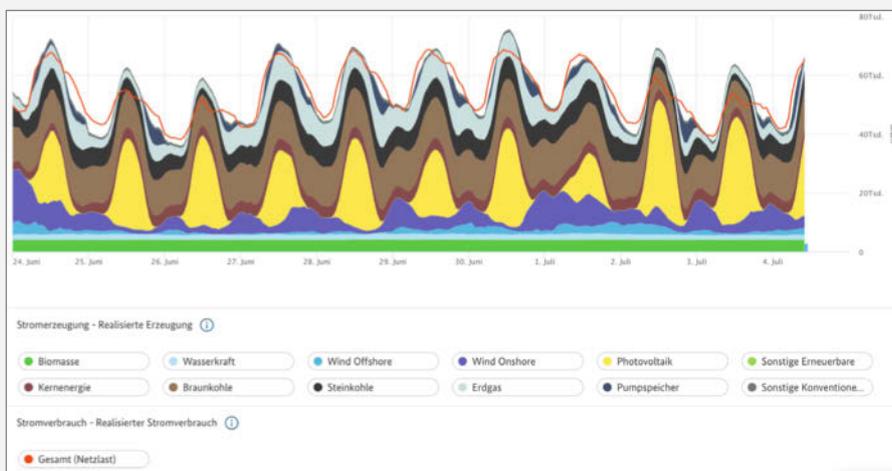
Voraussetzung dafür ist ein IoT-unterstütztes Stromnetz, das sogenannte *Smart Grid*. Die IoT²-Werkstatt wertet die Datenbasis der Bundesnetzagentur kontinuierlich aus und bietet über die Webseite einen schnellen Überblick über die aktuelle Situation.

Natürlich lassen sich diese Informationen auch mittels Blöckchen der IoT²-Werkstatt abrufen und zusammen mit der schaltbaren Shelly-Steckdose zu eigenen Experi-

menten nutzen. Aufgabenstellungen wie: Lade den Akku meines Rasenmähers mit mindestens 60% EE-Anteil, aber spätestens nächstes Wochenende muss der Akku voll sein, sind so zu realisieren.



Wer keinen IoT-Octopus hat, kann den Strommix beim Umwelt-Campus Birkenfeld ablesen.



Marktdaten der Bundesnetzagentur visualisieren Stromangebot und -Nachfrage in Echtzeit

verschiedene Plattformen und Protokolle (MQTT, HTTP-REST) per grafischen Blöckchen **7**. Man kann sich bei *ThingsSpeak.com* kostenlos einen *Channel* einrichten und dort Daten sammeln und visualisieren.

Was passiert nun tatsächlich nachts in unserem Kühlschrank? Wir zoomen in die Messwerte **8**, in dem wir alle 60 Sekunden den Leistungsbedarf (W) messen und machen somit die Steuerung des Kompressors sichtbar: Mittags, wenn gekocht wird und der Kühlschrank öfters geöffnet wird, oder wenn das Gefrierteil neues Gefriergut runtergekühlt, dann entstehen Verbrauchsspitzen. Diese Spitzen sind entscheidend für den späteren effizienten Betrieb unseres Solartisches: Denn hier zählt nicht die mittlere Leistung, sondern Angebot und Nachfrage müssen, wie beim bundesweiten Stromnetz auch, jederzeit zusammen passen. Dies führt dann ggf. dazu, dass die Spitzen zum Teil aus dem Stromnetz gedeckt werden, obwohl in den Pausen ein nicht genutzter solarer Überschuss besteht.

Besuch im Homeoffice

Schauen wir auch nochmal im Homeoffice **9** vorbei. Hier stehen Notebooks, Bildschirme, Router, Kamera und viele kleine Energiefresser. Unser Homeoffice verbraucht erstaunliche 1162Wh pro Tag. In der Nacht haben wir eine Grundlast von 15Wh und tagsüber eine Spitzenlast von 98Wh gemessen.

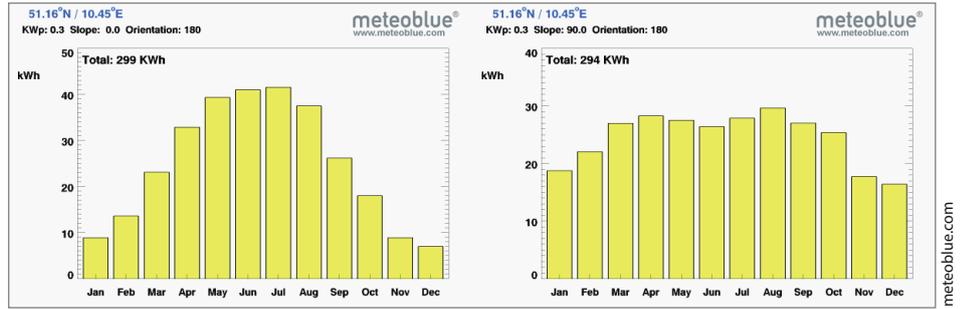
Aber auch unser Analyse-Duo aus IoT-Octopus und Shelly Plug benötigt natürlich Energie. Kalkulieren wir hier mit ca. 1 Watt pro ESP8266 mal zwei, so ergibt sich daraus ein Tagesbedarf von 48Wh für die Analyse- und Komfortfunktionen.

Nachdem wir jetzt Licht ins Dunkel unseres individuellen Energiebedarfs gebracht haben, wollen wir endlich zur heimischen Energie-wende kommen.

Ein Tisch muss es sein

Eines Tages stand nun ein 1,7m x 0,8m großes Solarpanel vor der Tür, die Frage entbrannte, wo soll das hin? Wir haben dafür doch gar keinen Platz. Der Balkon liegt im Schatten, die Terrasse bot Platz und Sonne, aber wo sollte diese große Platte aus Glas und Aluminium mit über 20kg Gewicht befestigt werden? So entstand die Idee, das Panel einfach als Tischplatte zu nutzen.

Aber warum eigentlich ein Tisch? Nicht jeder kann oder darf ein Solarpanel an der Balkonbrüstung montieren. Ein Tisch ist da eine geniale Alternative: Er kann in unterschiedlichen Größen gebaut werden, kann auf der Terrasse oder im Garten genutzt werden. Er benötigt keinen extra Stellplatz, weil er ja ein Möbel ist. Ein Tisch als Balkonkraftwerk ist auch in Mietwohnungen problemlos realisierbar und bei jedem Umzug mitzunehmen. Sollte der Balkon



10 Solarstromausbeute: Solartisch (links) gegen Balkonkraftwerk (rechts), ein Panel 300Wp

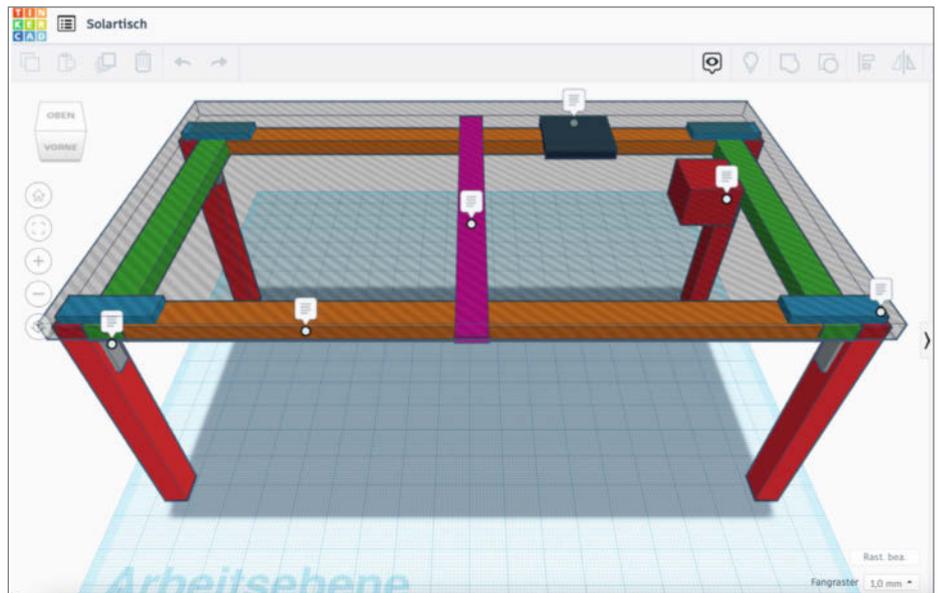
durch einen darüberliegenden Balkon verschattet sein, kann man das Panel während der Energieernte schräg aufstellbar montieren.

Es dreht sich um die Sonne

Wie viel Ertrag bringt unser Tisch übers Jahr? Nach wie vielen Jahren hat sich die Investition rentiert? Um diese zentrale Frage vor dem Bau zu klären, gibt es praktische Tools, die den zu erwartenden Ertrag simulieren.

Wichtig dabei ist der Standort (es gibt Gegenden mit mehr Sonnenschein), die Ausrichtung (Südseite) und der Aufstellwinkel. Natürlich ist es am besten, wenn die Sonne möglichst senkrecht auf unser Panel trifft. Aus diesem Grund sind die Panels schräg aufgestellt. Aber vergleichen wir einmal das klassische Balkonkraftwerk (hängend, Slope 90°) mit dem Solartisch (liegend, Slope 0°).

Man erkennt in **10** gut, dass die Ausbeute beim Solartisch im Sommer besser ist, im



11 Planung des Solartisches in Tinkercad



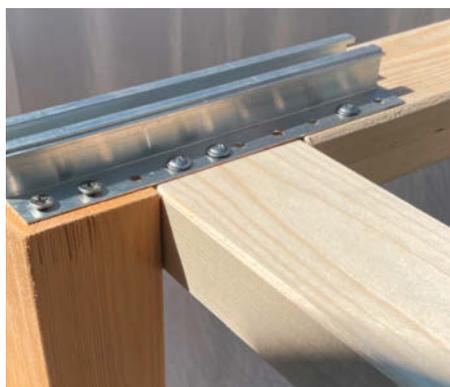
12 Fertiger Rahmen mit Montageschienen und optionalem Schaltkasten



13 Anprobe der Montageschiene, die Positionen für die Holzschrauben unbedingt vorbohren!



14 Montage der Trägerschienen mit Edelstahlschrauben.



15 Die Montageschiene verbindet die Querträger.



16 Die Querlatte und die richtige Menge Unterlegscheiben sorgen für Spielraum



17 Modulklemmen halten das Panel in Position.

Winter aber wegen der flach stehenden Sonne nicht mehr optimal. Bei Nordausrichtung wäre das Verhältnis aber genau umgekehrt. Hier lohnt sich die Simulation der örtlichen Gegebenheiten.

Fast alles im Baumarkt

Zunächst brauchen wir ein Gestell 11 für unseren Solartisch, welches das schwere Solarpanel tragen kann, nicht zu teuer ist und auch mal einem Unwetter standhalten kann. Alle wichtigen Materialien, außer dem Solarzubehör, finden wir im Baumarkt. Hier sind nur Beispiele aufgeführt, es gibt eine Vielzahl an Solarpanels und Wechselrichtern, hier muss

in der Chipkrise auch nach Verfügbarkeit gekauft werden. Das Material für den Tisch kann natürlich ebenfalls variieren. Daher bitte immer Abmessungen und Leistungsdaten überprüfen, nachmessen und an die Panelgröße anpassen.

Das Solarpanel wird später auf diesen Rahmen 12 montiert. Hierzu nutzen wir Montageschienen, welche dem Panel als Auflagepunkte dienen. Die Fixierung erfolgt dann mit vier Modulklemmen. Die Montageschienen geben den notwendigen Abstand, so wird der Wechselrichter nicht auf dem Panel oder Tischrahmen aufliegen. Ein Verteilerkasten oder eine -Dose findet an dem Rahmen einen wettergeschützten Platz. Die Winkelverbinder sorgen für die notwendige Steifigkeit.

Wichtig ist die Lagerung des Solarpanels, diese erfolgt *schwimmend*. Das Panel darf nicht fest verspannt werden, denn durch die hohen Temperaturen von über 60°C dehnt sich das Panel und braucht Freiraum wenn es sich biegt. Dies gelingt in diesem Design durch die Querlatte in der Tischmitte, welche nicht fest mit dem Panel verschraubt wird: sie stützt das Panel, hält es aber nicht fest.

Montage des Tischrahmens

Die Bilder 13 - 18 zeigen die Schritte, wie der Tisch gebaut und montiert wird.

Die Querlatte 16 verhindert, dass das Panel durchhängt, wird aber nicht mit dem Panel verbunden (s. o.). Zur Höheneinstellung (die Latte sollte 2-3 mm Spiel zum Panel haben) verwenden wir Unterlegscheiben aus Edelstahl. Diese werden zusammen mit einer Schraube zwischen Latte und dem Rahmen fixiert.

Das Bild 18 des Tisches von unten zeigt die kodierten MC4-Stecker, welche polrichtig mit dem Wechselrichter verbunden werden (immer prüfen, es soll falsch gepolte Kabel geben). Vom Wechselrichter geht es dann direkt zur Steckdose des Hausstromnetzes. Eine *Wieland*-Steckdose wird von den Energieversorgern meist vorgeschrieben. Der abgebildete Verteilerkasten ist optional.

Wenn alles montiert und verkabelt wurde, wird das Solarpanel montiert und wir haben einen schicken Solartisch!

Regenschirm und dicke Beine

Der Solartisch 19 sollte auch bei Wind, Schnee und Regen draußen stehen können. Dies haben wir im Design durch schwere Beine und einen stabilen Rahmen berücksichtigt. Die Alukante des Panels nutzen wir als Tropfkante für das Regenwasser. Schneelast und Hagel sind gemäß den Spezifikationen des Herstellers kein Problem.

Das Solarpanel ist mindestens für IP65 ausgelegt, für die Verkabelung und den optionalen Verteilerkasten sollte auch entsprechend zertifiziertes Material verwendet werden.

Starker UV-Strahlung und Feuchtigkeit muss das verwendete Material ebenso widerstehen können.

Wie warm wird ein Solartisch? Wir haben Temperaturen von etwa 60°C gemessen ²⁰, also auch nicht mehr als ein beliebiger dunkler Tisch in der Sonne. Die Besonderheit dabei: Das Solarpanel ist systembedingt sehr dünn, d. h. die Wärmekapazität ist beschränkt und es kühlt schnell wieder auf Umgebungstemperatur ab, wenn man den Tisch zum Essen beschattet.

Die Hinterlüftung muss unbedingt erhalten bleiben! Sie sorgt für eine gute Leistung des Panels. Je heißer ein Solarpanel wird, desto geringer ist sein Wirkungsgrad. Als Beispiel: Der Temperaturkoeffizient von kristallinen Panels liegt bei 0,4% Verlust pro Grad Celsius (über 25 Grad) – demnach liefert ein 70°C heißes 100W Panel nur noch 82W.

Prinzipiell hat der Solartisch zwei mögliche Betriebsarten: Wollen wir den Strom ins Hausstromnetz (Balkonkraftwerk-Prinzip) einspeisen oder ihn unabhängig davon im Garten (Insellösung) nutzen?

Solartisch als Kraftwerk

Der Solartisch wird wie eine Balkonsolaranlage an das Hausstromnetz angeschlossen. Hierzu benötigt man einen Wechselrichter für den Betrieb im Hausstromnetz. Wer sich hier nicht auskennt, greift am besten zu fertigen Komplettsets. Meist ist der Wechselrichter dann schon mit passendem Anschluss für die *Wieland*-Steckdose verkabelt, wie bei der *LightMate G*. Wichtig ist hier der Netz- und Anlagenschutz (*NASchutz*), dieser trennt im Fall einer erheblichen Netzstörung eine PV-Anlage vom Stromnetz.

Stecker in die Dose, oder?

Wir müssen für die Steuerung der Verbraucher wissen, wann und wie viel Solarstrom wir produzieren. Im Folgenden beziehen wir uns auf den Betrieb des Solartisches als Balkonkraftwerk. Hierzu gibt es zwei Möglichkeiten.

Einfach einen *Shelly Plug S* als Zwischenstecker zwischen Solartisch und Hausstromnetz setzen. Oder bei der empfohlenen *Wieland-Dose* einen *Shelly 1 PM* (kompatibel zu *Shelly Plug S*) als Unterputzversion ²¹ nutzen. Diesen kann man aber auch, wie hier in den Beispielen, in einen Verteilerkasten zusammen mit einem Leitungsschutz, oder allein in einer günstigen Verteilerdose aus dem Baumarkt unterbringen.

In jedem Fall muss der Wetterschutz gewahrt werden, aber auch die korrekte Verdrahtung sollte vom Fachmann geprüft werden, bevor die Anlage in den Betrieb geht.

Energiespeicher

Wir hatten uns eingangs den Stromverbrauch von Kühlschrank und Homeoffice angesehen.

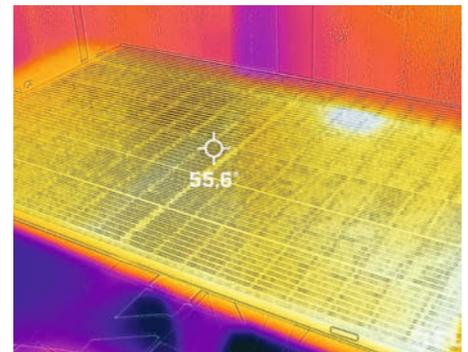


18 Kabelführung vom MC4-Stecker zum Wechselrichter, dann zum Verteilerkasten

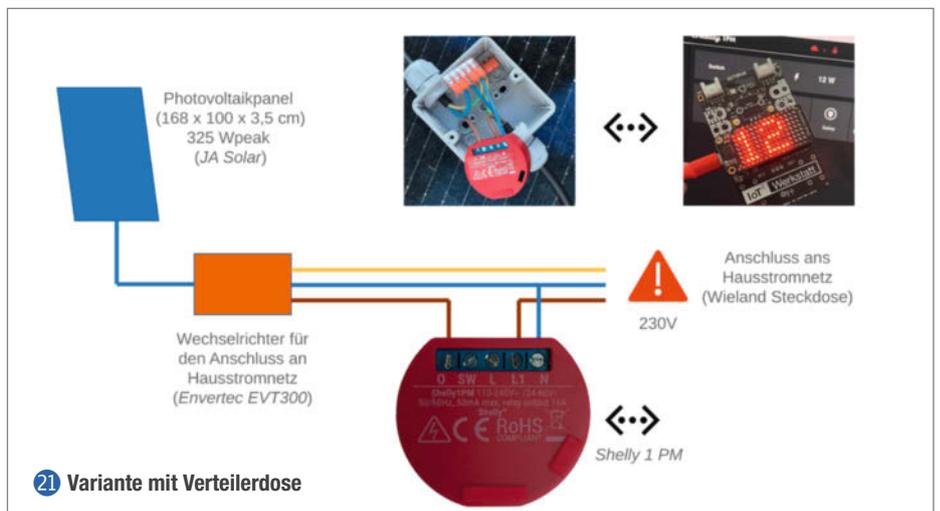


19 Allwetterfeste Konstruktion des Panels. Das Holz muss behandelt werden.

Zumeist waren die Verbrauchsspitzen tagsüber. Aber was, wenn ich nachts fernsehen will? Oder wenn unser Balkonkraftwerk mehr Strom produziert, als gerade benötigt wird? Zur perfekten Energiewende gehört noch ein Energiespeicher, den wir mit unserer überschüssigen Energie laden und später nutzen können. Hier bietet der Markt interessante Powerstationen wie die *EcoFlow RIVER mini wireless*, eine Pufferbatterie mit 230V Ausgang. Praktischerweise kann diese dann auch als mobile Energiequelle beim Camping, als Notstromaggregat bei Stromausfall oder als *USV* genutzt werden. Dieses Modell ist eine gute Einsteigerlösung. Ähnliche Geräte gibt es auch mit viel mehr Kapazität.



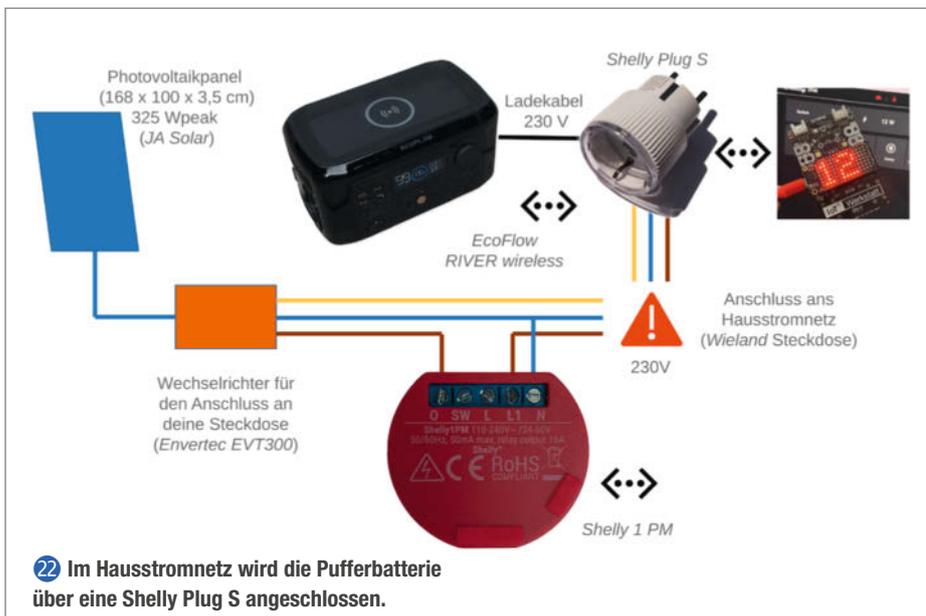
20 Wärmebild des Solartisches. Die Position des Wechselrichters ist an dem weißen Hotspot zu erkennen.



Da unser Solartisch direkt mit dem Hausstromnetz ²² verbunden ist, können wir die Pufferbatterie direkt am zu puffernden Verbraucher platzieren. Ein *Shelly Plug S* ver-

bindet das 230V-Ladekabel nur dann mit dem Hausstromnetz, wenn genügend Energie vorhanden ist, die ansonsten eventuell kostenlos an den Energieversorger abgegeben

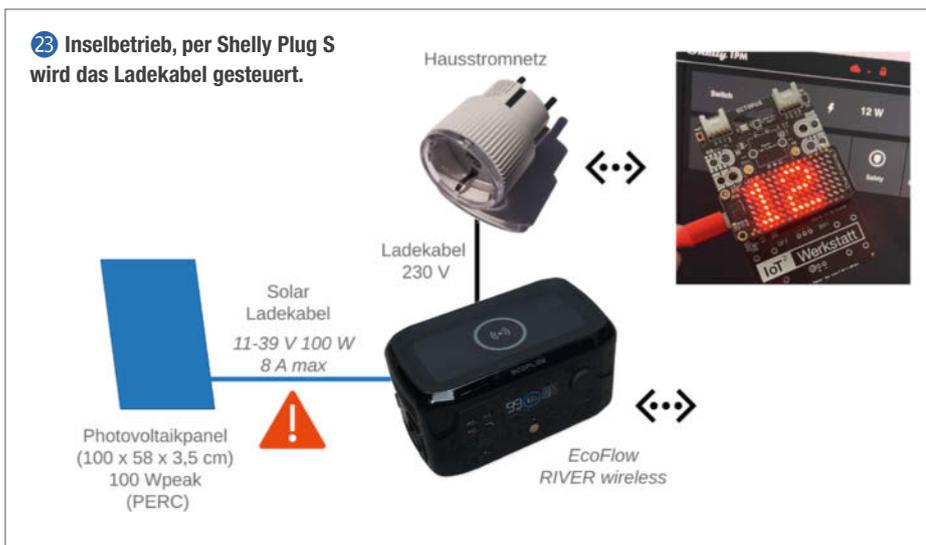
würde. Im Beispiel schaltet der IoT-Octopus den Ladeprozess ein, wenn mehr als 180W Solarleistung ins Hausstromnetz eingespeist werden. Der Strom kann dann vom Verbraucher aus dem Batteriepuffer entnommen werden. Dieses Konzept hat den Vorteil, das nach Bedarf mehrere unterschiedliche große Pufferbatterien benutzt werden können. Ferner können diese auch mit EE-Strom aus dem Netz geladen werden, wenn mal kein Solarstrom lokal produziert werden kann (siehe Kasten). Auch die Pannelleistung muss nicht, wie bei Inselbetrieb, auf die Größe des Solarladers der Pufferbatterie angepasst werden. In der IoT²-Werkstatt gibt es auch ein ArduBlock für den Zugriff auf den Ladezustand der Pufferbatterie. Alle Beispiel-Programme für den IoT-Octopus erhalten Sie über die Links in der Kurzinfo.



Solartisch als Insel

Wer zum Beispiel nur die Gartengeräte oder sein E-Bike aufladen möchte, kann den Solartisch auch im Inselbetrieb ²³ nutzen. Hier entfällt der Wechselrichter für die Verkabelung zum Hausstromnetz und eine Elektroinstallation im 230V-Netz.

Im Inselbetrieb muss das Solarpanel zum Solarcharger der verwendeten Powerstation, hier eine *EcoFlow RIVER wireless* passen. Dieses spezielle Modell kann direkt mit bis zu 39V und 8A bei aber insgesamt maximal 100W geladen werden. Die Insel ²⁴ kann so direkt 230V aus den Pufferakku generieren. Hier steht eine Kapazität von 210Wh zur Verfügung, welche mit 300W (600Wp) abgerufen werden kann. Zusätzlich stehen USB-C/A/PD und eine 12V-Dose zur Verfügung. Ein E-Bike, Akku-Rasenmäher oder auch die Teichpumpe lassen sich so problemlos laden bzw. betreiben. Smartphones und andere Kleingeräte lädt man direkt an USB oder mit dem eingebauten kabellosen Lader. Die niedrigschwellige Einsteiger-Variante!



Die grüne Steckdose im Eigenbau

Die Integration von schaltbaren Steckdosen mit Strom-Messfunktion, Solartisch und Energiespeicher im Internet der Dinge ermöglicht die einfache Umsetzung eigener Steuerungsstrategien – sei es im Hausstromnetz oder im Inselbetrieb. Jeder Haushalt hat spezifische Randbedingungen (Balkonausrichtung, elektrische Verbraucher, Nutzungsverhalten), die sich so möglichst optimal integrieren lassen. Die Zeiten von „Strom kommt aus der Steckdose“ sind vorbei. Starten wir die Energiewende als Graswurzelprojekt mit einer grünen Steckdose, die den Strom aus der Steckdose bunt färbt und Verbraucher bei grünem Strom einschaltet. —caw



24 Beispiel für den Inselbetrieb mit einer EcoFlow-river-Powerbank

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.



Eigenbau-Wärmetauscher für das Taupunkt-Lüftungssystem

Wer falsch lüftet, holt sich oft Kälte oder Feuchtigkeit ins Haus. Gegen die Feuchtigkeit hilft ein Taupunkt-Lüftungssystem, wie es in Heft 1/2022 vorgestellt wurde. Doch gegen ein Auskühlen des Kellers durch das Lüften nützt dies nichts. Abhilfe schafft die Kombination des Lüftungssystems mit einem Wärmetauscher. Unser Projekt zeigt, wie man diesen günstig und mit wenig Aufwand selber bauen kann.

von Peer Steldinger



Der erste Artikel zum *Taupunkt-Lüftungssystem* auf Basis eines *Arduino Nano* und zweier *DHT22-Sensoren* erschien in der *Make 1/22*. Ein solches System lüftet nur dann, wenn die Luft im Keller mehr Wasser enthält als die Außenluft, sodass vorhandene Feuchtigkeit aus dem zu lüftenden Raum transportiert wird. Als Maß für die in der Luft vorhandene Feuchtigkeit wird die *Taupunkttemperatur* verwendet. Das ist die Temperatur, auf die die Luft abgekühlt werden muss, damit die relative Luftfeuchtigkeit auf 100% ansteigt – daher der Name *Taupunkt-Lüftungssystem*. Gelüftet wird nur dann, wenn der Taupunkt innen höher ist als draußen, wenn das Lüften also nicht noch mehr Feuchtigkeit nach innen holt.

Oft schaltet ein solches *Taupunkt-Lüftungssystem* die Lüftung gerade dann ein, wenn es draußen deutlich kälter ist als drinnen. Der Grund ist, dass kalte Luft oft deutlich weniger Wasser enthält als warme, selbst wenn man dies nicht erwartet. Für das Lüften ist die absolute Luftfeuchtigkeit wichtig und nicht die relative Luftfeuchtigkeit, die zwar eher dem menschlichen Empfinden entspricht, aber eben sehr stark von der Lufttemperatur abhängt. Ein Beispiel: Tritt bei 0–5°C Bodennebel auf, d.h. die relative Luftfeuchtigkeit beträgt 100%, so kann man die Feuchtigkeit der Außenluft deutlich spüren und sogar sehen. In diesem Fall entspricht der Taupunkt direkt der Temperatur. Im Vergleich dazu nimmt man 19°C warme Zimmerluft bei 40% relativer Luftfeuchtigkeit als angenehm bis eher zu trocken wahr. Tatsächlich liegt der Taupunkt bei 19°C und 40% Luftfeuchtigkeit aber bei ca. 5,1°C, die Raumluft ist also absolut gesehen feuchter als die neblige Außenluft.

Das *Taupunkt-Lüftungssystem* senkt daher nicht nur die Luftfeuchtigkeit in Innenräumen, sondern fast immer auch die Raumtemperatur. Der Code für das System aus *Make 1/22* sieht zwar eine Mindesttemperatur für den Keller vor, unter der das System nicht mehr aktiv wird. Auf den Artikel hin haben aber viele *Make-Leser* geäußert, dass sie gerade im Herbst und im Winter die Kälte eher gerne draußen lassen und sie sich nicht noch aktiv ins Haus holen wollen. Ist die kalte Luft einmal im Haus, muss die Heizung einspringen. Das Ergebnis sieht man dann auf der Heizkostenabrechnung.

Soll man deswegen auf regelmäßiges Lüften verzichten? Natürlich nicht! Aber um Heizkosten zu sparen, braucht man einen Weg, den Wärmeverlust beim Lüften zu verhindern oder zumindest zu beschränken. Tatsächlich gibt es eine überraschend einfache Methode, wie man zwar die feuchte Innenluft gegen trockenere Außenluft tauschen kann, dabei aber die Wärme weitgehend erhalten bleibt. Der Trick ist, die ausströmende Luft zu nutzen, um die einströmende Luft zu erwärmen. Dies lässt sich mit Hilfe eines sogenannten *Wärmetauschers* erreichen.

Kurzinfo

- » Funktionsprinzip und Arten von Wärmetauschern
- » Eigenbau-Wärmetauscher für das Taupunkt-Lüftungssystem wärmt Außenluft passiv vor
- » Wirkungsgrad von bis zu 85 Prozent

Checkliste

-  **Zeitaufwand:**
zwei Wochenenden
-  **Kosten:**
120 Euro für den Wärmetauscher, 80 Euro für die hier benutzten Lüfter und Rohre, evt. zusätzliche Kosten bei Wanddurchbruch

Werkzeug

- » Feinsäge ohne Rücken
- » Stypoporschneider
- » Heißkleber
- » Sanitär-Silikon

Material

- » 84 Aluminiumbleche 15cm × 15cm, 0,5mm dünn
- » Styroporbox 49 Liter, Innenmaß 40,4cm × 40,4cm × 30,3cm
- » PVC-Hartschaumplatte 42cm × 30cm, 19mm dick
- » PVC-Hartschaumplatte 42cm × 30cm, 10mm dick
- » 25,2m doppelseitiges Dichtungsklebeband 10mm breit, 3mm dick
- » 2 Lüftungsschutzgitter für die inneren Öffnungen, mit Insektenschutz
- » Material für die Leitungen nach draußen je nach baulichen Gegebenheiten

Mehr zum Thema

- » Ulrich Schmerold, Das Taupunkt-Lüftungssystem, *Make 1/22*, S. 22
- » Ulrich Schmerold, Logging-Funktion für das Taupunkt-Lüftungssystem, *Make 2/22*, S. 82
- » Dieter Hoffmann, Daniel Bachfeld, Elke Schick, *Alternative Energie*, *Make Sonderheft 2018 „Energie“*, S. 3

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



tauschers erreichen. Dieser tauscht nicht nur verbrauchte Raumluft gegen frische Außenluft, sondern sorgt zusätzlich noch dafür, dass die Wärme der Abluft auf die Zuluft übertragen wird. Daher passt der weniger gebräuchliche Name *Wärmeübertrager* eigentlich besser.

Der große Vorteil ist, dass ein Wärmetauscher, abgesehen von den nötigen Lüftern, keine Energie verbraucht, da die einströmende Luft nur passiv erwärmt wird. Statt die kostbare Wärme mit der Abluft nach draußen verpuffen zu lassen, wird diese Wärmeenergie einfach zurückgewonnen. Ohne ein solches Energie-recycling wären moderne Niedrigenergiehäuser gar nicht denkbar.

Was ist also naheliegender, als das *Taupunkt-Lüftungssystem* aus dem Heft 1/22 um einen selbstgebauten Wärmetauscher zu erweitern?

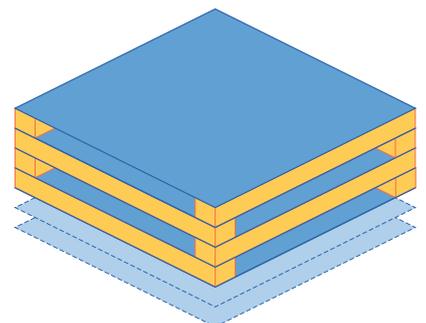
Arten von Wärmetauschern

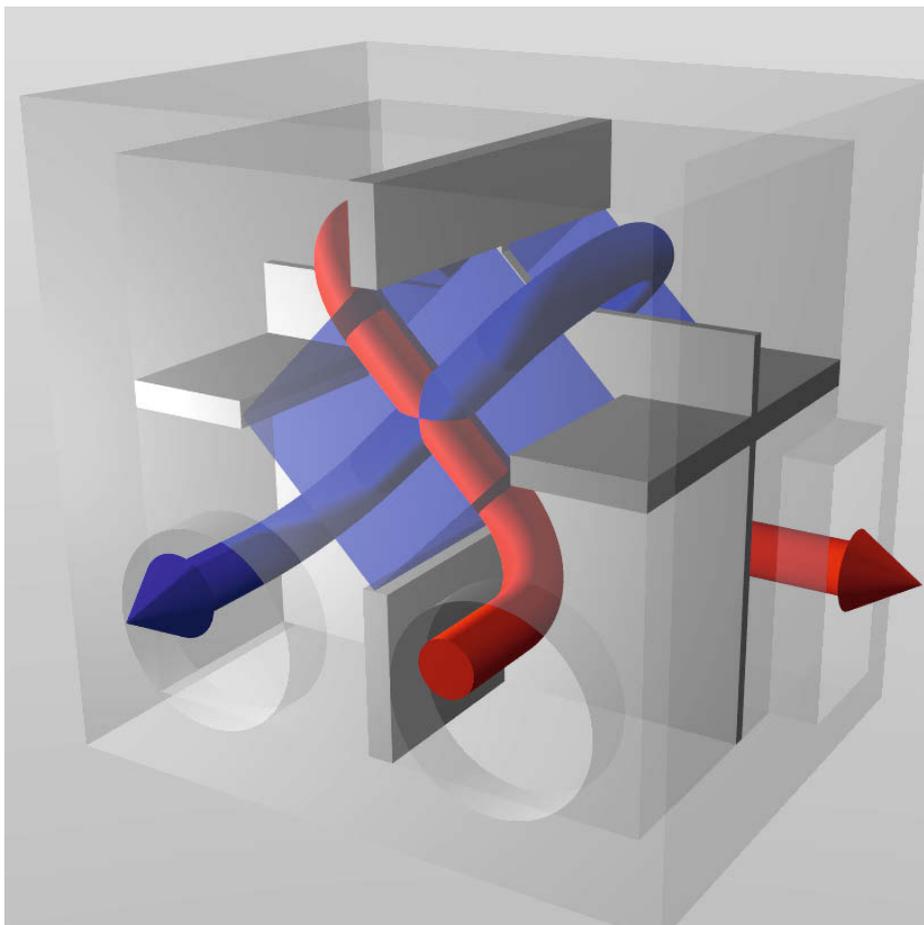
Wärmetauscher finden heutzutage in vielen Geräten Anwendung – immer dann, wenn Wärme von einem Medium auf ein anderes übertragen werden soll. Ein Heizkörper ist daher im Grunde nichts anderes als ein

Wärmetauscher, genauso wie auch der Kühler eines Autos. Aber auch Kühlschränke, Klimaanlage und moderne Wäschetrockner kommen nicht ohne Wärmetauscher aus.

Schema Wärmetauschermodule

Die Wärmetauschermodule entstehen durch Stapel von dünnen Aluminiumplatten (blau), die mit Dichtungsklebeband (gelb) aufeinander geschichtet werden, wobei von Blech zu Blech jeweils um 90 Grad gedreht wird.





1 Schematische Darstellung des Funktionsprinzips des hier umgesetzten Wärmetauschers

Das Funktionsprinzip eines Wärmetauschers ist dabei sehr einfach: Zwei Medien strömen aneinander vorbei, sodass das kühlere Medium das wärmere abkühlt und umgekehrt von diesem erwärmt wird. In unserem Fall handelt es sich bei den beiden Medien um die Zuluft und die Abluft, es geht also um einen *Luft-Luft-Wärmetauscher*.

Man unterscheidet zwischen *direkten*, *indirekten* und *halb-indirekten* Wärmetauschern. Bei einem *direkten* Wärmetauscher werden das warme und das kalte Medium miteinander vermischt und diese müssen nach der dabei erfolgenden Wärmeübertragung wieder voneinander getrennt werden. Das ist dann einfach, wenn ein Medium ein Gas ist und das andere eine Flüssigkeit. Möchte man aber die Wärme zwischen zwei Luftströmen übertragen, lässt sich dieses Prinzip nicht anwenden.

Bei einem *halb-indirekten* Luft-Luft-Wärmetauscher durchströmen die warme und die trockene Luft zeitlich nacheinander denselben Weg und die Wärmeübertragung erfolgt über ein dabei durchlaufendes wärmespeicherndes Material. Die einfachste Ausführung besteht aus einem bidirektional betriebbaren Ventilator, der die Luft in kurzen, ständig wechselnden Intervallen von innen

nach außen und von außen nach innen durch eine Metallgitterstruktur drückt. Der Wirkungsgrad von halb-indirekten Wärmetauschern ist aber begrenzt.

Die bessere Alternative sind *indirekte* Wärmetauscher, bei denen beide Luftströme voneinander getrennt verlaufen, aber eine möglichst große Kontaktfläche zu den zwischen ihnen verlaufenden, gemeinsamen Trennwänden haben. Über die Kontaktfläche wärmt die ausströmende Raumluft die einströmende Außenluft auf, kühlt dabei selbst ab und geht als abgekühlte Abluft nach draußen. Die vormalige Außenluft strömt dagegen als aufgewärmte Zuluft in den Raum.

Strömen die beiden Luftströme dabei in entgegengesetzter Richtung aneinander vorbei, handelt es sich um einen sogenannten *Gegenstromwärmetauscher*. Bei vollständiger Wärmeübertragung erreicht die Zuluft dabei die Raumlufttemperatur und die Abluft wird auf dieselbe Temperatur wie die Außenluft abgekühlt. Ein solcher idealer Wirkungsgrad von 100% Wärmeübertragung ist jedoch in der Praxis nicht erreichbar.

Verlaufen die beiden Luftströme dagegen parallel, also in gleicher Richtung, erreichen die beiden Luftströme eine mittlere Temperatur.



2 Anbringen der Klebestreifen auf den Aluplatten in einer Reihe

Bei identischer Strömungsgeschwindigkeit und Luftmasse entspricht diese genau dem Mittelwert der Raum- und der Außentemperatur. Damit erreicht ein solcher *Gleichstromwärmetauscher* selbst im Idealfall nur einen Wirkungsgrad von 50%.

Kreuzstromwärmetauscher stellen einen Mittelweg zwischen Gleichstromwärmetauschern und Gegenstromwärmetauschern dar: Die beiden Luftströme verlaufen senkrecht zueinander, was man als Kombination eines Gleichstrom- und eines Gegenstromanteils auffassen kann. Der bestmögliche Wirkungsgrad beträgt hier dementsprechend 75%. Ein großer Vorteil von Kreuzstromwärmetauschern ist deren einfache Bauweise.

Der hier vorgestellte Wärmetauscher arbeitet nach einer anderen Variante des Kreuzstromprinzips: Um den Wirkungsgrad gegenüber einem reinen Kreuzstromwärmetauscher zu erhöhen, werden zwei Kreuzstromwärmetauscher im Gegenstromprinzip hintereinander verwendet 1. So könnten im Idealfall 75% des nach dem ersten Wärmetauscher noch vorhandenen Temperaturunterschieds zurückgewonnen werden und der damit theoretisch erreichbare Wirkungsgrad steigt auf $75\% + 75\% \cdot 25\% = 93,75\%$.

Die genaue Abstimmung eines Wärmetauschers auf gegebene Lüfter ist jedoch nicht einfach und so ist es nicht verwunderlich, wenn unser Selbstbau diese theoretischen Werte nicht erreicht. Aber der Nachbau lohnt sich trotzdem auf jeden Fall.

Aufbau

Ein Kreuzstromwärmetauscher ist im Grunde sehr einfach aufgebaut: Es reicht, viele dünne quadratische Platten so mit etwas Abstand zueinander zu einem Stapel zu verbinden, dass die Luft zwischen je zwei Platten nur von einer der vier Seiten zur gegenüberliegenden Seite strömen kann. Diese Strömungsrichtung muss dabei von Schicht zu Schicht senkrecht zueinander versetzt sein. So strömt an jeder der Platten sowohl die Zuluft als auch die Abluft entlang. Die Platten selbst sollten idealerweise gute Wärmeleiter sein. Ich nutze hierfür 84 Aluminiumbleche der Größe 15cm × 15cm mit einer Dicke von 0,5mm. Aluminium hat eine sehr gute Wärmeleitfähigkeit, ist korrosionsbeständig und im Vergleich zu den noch besseren Wärmeleitern Kupfer, Silber und Gold sehr günstig.

Mit etwas Glück findet man Alubleche bereits in den passenden Maßen zugeschnitten in Baumärkten oder Bastelgeschäften. Ich habe auf fertig zugeschnittene Alubleche zurückgegriffen, die im 12er-Set für 15 Euro angeboten wurden. 7 Sets ergeben dann die nötigen 84 Bleche. Man kann aber auch größere Aluminiumplatten kaufen und selbst zuschneiden. Dazu eignen sich Metallsägen, also Sägen mit möglichst kleinen Sägezähnen. Es gibt auch spezielle Sägeblätter für Aluminium. Wer eine elektrische Säge verwendet, sollte diese nicht zu schnell laufen lassen, da sich Aluminium sehr schnell erwärmt. Viele Baumärkte bieten auch einen Zuschneideservice für die verkauften Alubleche an. Es lohnt sich, hier nach günstigen Angeboten Ausschau zu halten, da die Alubleche die Hauptkosten des Wärmetauschers darstellen.

Klebebandverbindung

Die Bleche verbinde ich mit Hilfe von doppelseitigem Dichtungsklebeband der Breite 10mm und Stärke 3mm. Dabei verklebe ich auf jedem Blech an zwei gegenüberliegenden Seiten durchgehend je einen Streifen Klebeband. Dies gelingt besonders einfach, wenn man mehrere der Bleche zu einer Reihe auslegt und das Klebeband so über die gesamte Reihe aufbringt **2**. Anschließend werden die einzelnen Bleche dann mit Hilfe eines Cutters wieder voneinander getrennt. Bei 84 Blechen der Seitenlänge 15cm sind $84 \cdot 0,3\text{m} = 25,2\text{m}$ Dichtungsklebeband nötig. Je 42 der Bleche werden nun immer abwechselnd um 90° ge-



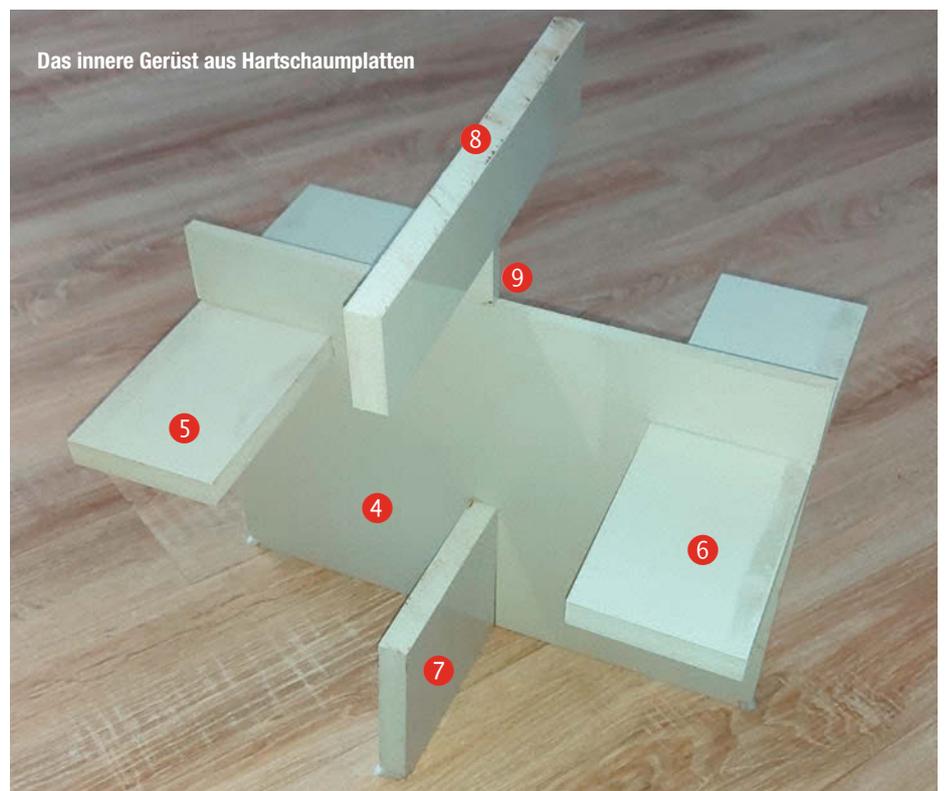
3 Die ersten Plattenstapel entstehen.

dreht aufeinander zu einem würfelförmigen Stapel verklebt **3**, siehe auch *Schema Wärmetauschermodule*.

Die beiden eigentlichen Wärmetauschermodule sind damit schon fertig. Die Module sind nun ca. 15cm hoch ($42 \times 0,5\text{mm} + 43 \times 3\text{mm}$). Entlang jeder Durchströmungsrichtung weisen sie jeweils 21 Schlitze der Größe 13cm × 3mm auf. Dies entspricht jeweils einer Querschnittsfläche von $21 \times 13\text{cm} \times 0,3\text{cm} = 81,9\text{cm}^2$. Die gemeinsame Kontaktfläche zwischen den beiden Luftströmungen beträgt nun wegen der Breite der Klebebänder $84 \times 13\text{cm} \times 15\text{cm} = 1,638\text{m}^2$.

Gehäuse

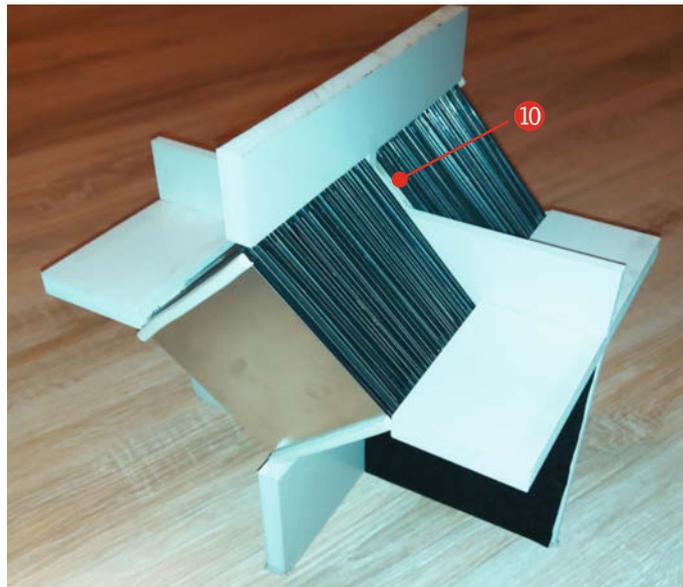
Nun müssen die Module noch so in ein Gehäuse eingebaut werden, dass die Luftströme über verschiedene Kammern durch beide Wärmetauschermodule geleitet werden, ohne sich dabei zu durchmischen. Auch sollten die Kammern im Wärmetauscher gut gegeneinander und nach außen isoliert sein, damit die abgekühlte Abluft dem Raum bzw. der Zuluft nicht wieder Wärme entzieht und die erwärmte Zuluft nicht die Abluft erwärmt. Ich habe mich für die Verwendung einer Styroporbox als Gehäuse entschieden, da dieses einen



Das innere Gerüst aus Hartschaumplatten



11 Der fertige Wärmetauscher von innen bei abgenommenem Deckel



10 Die Hartschaum-Dreiecke isolieren die beiden Wärmetauschermodule voneinander.

hervorragenden Wärmedämmer darstellt. Außerdem nimmt Styropor keine Feuchtigkeit auf und ist resistent gegenüber Pilzbefall – in Anbetracht dessen, dass in einem Wärmetauscher Kondenswasser anfallen kann, ist dies ein großer Vorteil gegenüber einem Gehäuse aus Holz.

Geeignete Styroporboxen werden als Thermoboxen oder Isolierboxen für den Transport von zu kühlenden Lebensmitteln günstig in verschiedensten Größen angeboten. Ich nutze eine 49-Liter-Box mit den Innenmaßen $40,4\text{cm} \times 40,4\text{cm} \times 30,3\text{cm}$ und einer Wandstärke von $3,8\text{cm}$ (Außenmaße $48,2 \times 48,2 \times 38,2\text{cm}$).

Innerhalb der Box verwende ich Trennwände aus PVC-Hartschaum, da diese ebenfalls gut isolieren, leicht zu bearbeiten sind und die für die Trennwände nötige hohe Stabilität bieten. Eine Hartschaumplatte in Größe $42\text{cm} \times 30\text{cm}$ (DIN A3) der Stärke 19mm und eine weitere gleicher Größe, aber in 10mm Stärke reichen aus.

Für den Zuschnitt der Hartschaumplatten zeigt der Zuschnittplan (Download über den Link in der Kurzinfor) die nötigen Schnittkanten für die beiden Platten. Bei Verwendung einer anderen Styroporbox müssen die Schnittmaße entsprechend angepasst werden. Der Zuschnitt erfolgt nun mit Hilfe einer Feinsäge ohne Rücken. Als Erstes säge ich die kleinen, in der 10mm starken Platte eingezeichneten Dreiecke aus. Diese benötige ich später noch. Anschließend kürze ich die 10mm -Platte auf das Maß $40,4\text{cm} \times 30\text{cm}$ und säge auf den kurzen Seiten in $22,5\text{cm}$ bis $24,5\text{cm}$ Höhe je einen 2cm breiten Einschnitt der Tiefe $9,5\text{cm}$. An der unteren langen Seite säge ich einen Einschnitt der Breite 2cm und Tiefe $12,8\text{cm}$ 4.

Aus der 19mm -Platte säge ich zwei „Bretter“ der Größe $30\text{cm} \times 9,6\text{cm}$ 5 6 und ein Brett der Größe $30 \times 12,4\text{cm}$ 7. Von dem übrigen Brett der Größe $30\text{cm} \times 10,4\text{cm}$ 8 entferne ich längs bis auf einen mittigen Steg von 1cm Breite 9 noch einen Streifen der Breite $3,7\text{cm}$, wie im Zuschnittplan gezeigt wird.

Die $9,5\text{cm}$ breiten Bretter 5 6 setze ich nun mittig in die seitlichen Schlitz der 10mm -Platte 4. Das $12,4\text{cm}$ breite Brett 7 wird in den unteren Schlitz eingesetzt. Nun klebe ich mit Heißkleber das T-förmige Brett mittig so von oben auf die 10mm -Platte, dass es senkrecht über dem $12,4\text{cm}$ langen Brett verläuft 8 9. Dort, wo die beiden Bretter aufeinandertreffen, klebe ich seitlich die beiden Dreiecke an,



12 Der fertige Wärmetauscher mit den vier Öffnungen

damit die beiden Wärmetauschermodule vollständig voneinander getrennt sind **10**. Zum Abdichten der Fugen zwischen den Wärmetauschermodulen und den Hartschaumplatten nutze ich Bad-Silikon. Das ganze kommt später so in die Styroporbox, dass deren Deckel die Vorderseite bildet und deren Boden die Rückseite **11**.

Schließlich schneide ich mit einem Styroporschneider die vier Öffnungen für die ein- und ausströmende Zu- und Abluft in die Styroporbox. Bei meinem Aufbau liegen die beiden Öffnungen zum Innenraum hin auf der Vorderseite des Deckels der Box und eine Öffnung für die Zuluft von außen auf der Rückseite, dem ursprünglichen Boden der Box **12**. Die vierte Öffnung, durch die die Abluft nach draußen gelangen soll, liegt an einer der Seitenwände, da ich den Wärmetauscher in einer Raum-Ecke installiere, sodass die Zu- und die Abluft durch zwei zueinander senkrechte Wände verlaufen (siehe Titelbild des Artikels). Die ersten drei Öffnungen können kreisrund mit dem Radius der Lüftungsrohrabdeckungen geschnitten werden. An der Seite ist jedoch nicht genügend Platz, um die vierte Öffnung mit demselben Radius auszuführen. Stattdessen



13 Wer sich für Wanddurchbrüche entscheidet, braucht eine 1500-Watt-Bohrmaschine mit Kernbohrer.



Maker & MINT Welt



Snapmaker
3-in-1 3D Drucker,
Fräser & Laser



iRobot
Coding Roboter



Arduino
Boards

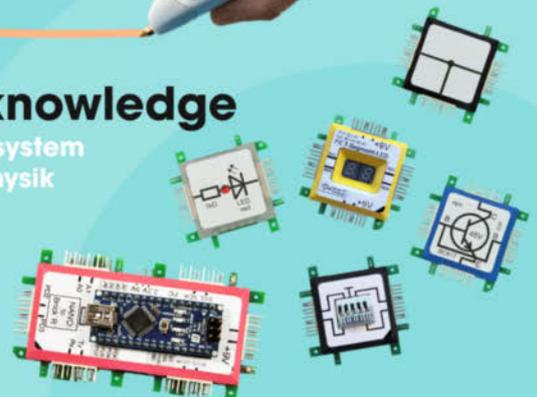


Photon
Coding, Künstliche
Intelligenz

3Doodler
MINT 3D Stift



Brick'R'knowledge
Experimentiersystem
Elektronik & Physik



MatataLab
Coding ab 4 Jahren





14 Der Lohn der Mühen: Die fertige Kernbohrung

verwende ich hier einen Adapter von 15cm Durchmesser auf 150er Flachkanal, da dessen rechteckige Seite mit den Maßen 90mm x 220mm seitlich genau passt. Die genaue Position der Öffnungen (Rückseite oder seitlich) lässt sich aber je nach Installationsort frei wählen.

Zwei statt ein Lüfter

Das in 1/22 vorgestellte Lüftungssystem steuert nur einen Lüfter an. Der Wärmetauscher benötigt jedoch zwei Lüfter, einen für die Zu- und einen für die Abluft. Wer nun mit dem Arduino zwei Relais ansteuern möchte, je



15 Eingebautes Lüftungsrohr mit innenliegendem Ventilator

eines für einen Lüfter, kann in das Problem hineinlaufen, dass es durch ein zeitgleiches Schalten beider Relais schnell zu Spannungseinbrüchen kommt, die den Arduino in die Knie zwingen. Einfacher ist es, beide Lüfter an nur einem Relais anzuschließen. Für den Arduino macht das keinen Unterschied. Wichtig ist dabei, dass die Lüfter parallel und nicht in Reihe angeschlossen werden.

Zu- und Abluftkanäle

Der eigentliche Wärmetauscher ist nun fertig. Mehr Aufwand macht der Einbau der Lüfter, da nun zwei Durchbrüche nötig sind und die



16 Öffnen des Ventilators zum Verlängern des Anschlusskabels

Luftströme durch den Wärmetauscher geleitet werden sollen. Ein einfacher Durchlass in einem Kellerfenster reicht hier also nicht aus. Möchte man die Zu- und die Abluft durch ein Kellerfenster laufen lassen, bräuchte man zwei Durchlässe und müsste diese mit Lüftungsrohren oder -schläuchen mit dem Wärmetauscher verbinden. Das sähe nicht schön aus und es würde auch nur noch wenig Licht durch das Fenster lassen.

Ich habe mich stattdessen entschieden, 15cm-Lüftungsrohre durch die Kellerwand zu verlegen. Der für die beiden Bohrungen erforderliche Aufwand darf dabei aber nicht unterschätzt werden. Man benötigt eine spezielle Diamant-Bohrkrone für Kernbohrungen. Wichtig ist, dass die Bohrkronen einen Bohrchdurchmesser von nicht genau 150mm, sondern ca. 155mm hat, damit die 150mm-Lüftungsrohre auch tatsächlich passen.

Wichtig: Für das Bohren mit solchen Bohrkronen reicht eine normale Bohrmaschine nicht aus. Man sollte schon eine spezielle Kernbohrmaschine oder zumindest eine 1500W Bohrmaschine nutzen **13**. Die wenigsten haben eine solche Ausrüstung zu Hause, man kann sich die Geräte aber im Baumarkt mieten.

Wenn man keine Kernbohrmaschine mit Wasseranschluss nutzt, sollte man die Bohrkronen regelmäßig in Wasser tauchen, um ihren Verschleiß zu reduzieren. Auch ist es wichtig, dass man die Bohrmaschine ohne Schlagwerk und nur bei relativ niedriger Geschwindigkeit laufen lässt. Geduld zahlt sich hier aus **14**. Unbedingt empfehlenswert ist gerade bei Wänden ab einer Stärke von 20cm auch die Verwendung eines an der Wand anzubringenden Bohrständers, da eine so große Kernbohrung von Hand nur mit viel Erfahrung durchführbar ist und eine beim Blockieren der Bohrkronen ausschlagende Bohrmaschine zu ernsthaften Verletzungen führen kann. Bei dem vergleichsweise hohen Aufwand und den auch nicht zu vernachlässigenden Gerätemietkosten lohnt es sich, darüber nachzudenken, für die Bohrungen einen Fachmann zu beauftragen.

Als Lüfter verwende ich Einschub-Rohrventilatoren, die in die Lüftungsrohre eingesetzt werden und somit unsichtbar in der Wand verschwinden **15**. Ich nutze das Modell *Calimaero INLINE 150*, welches mit einer Lüftungsleistung von bis zu 320m³/h eher groß ausgelegt ist. Leider ist dessen Anschlusskabel nur ca. 20cm lang. Durch Lösen einer Schraube am Ventilatorkopf kann man jedoch die Verbindung des Anschlusskabels freilegen und dieses durch ein längeres ersetzen **16**.

Außen werden die Lüftungsrohre durch aufsteckbare Lüftungsgitter mit integriertem Insektengitter abgeschlossen. Dieselben Lüftungsgitter nutze ich auch für die beiden Lüftungsdurchbrüche an der Vorderseite des Wärmetauschers. Hier drehe ich die Gitter so

voneinander weg, dass sich die einströmende und die ausströmende Luft möglichst nicht vermischen.

Energiebilanz

Der Energieverbrauch des Wärmetauschers wird während des aktiven Lüftens maßgeblich durch die Lüfter bestimmt. In meinem Fall verbraucht jeder Lüfter nominell 18 Watt. Dies deckt sich mit einem gemessenen Gesamtverbrauch von ca. 37 Watt für zwei Lüfter und die Regelungsschaltung. Bei ausgeschalteten Lüftern sinkt der Verbrauch auf 0,5 Watt für den Arduino, das Display und das Relais. Schaltet man zusätzlich noch die aktive Hintergrundbeleuchtung des LCDs aus, sinkt der Verbrauch weiter auf knapp 0,2 Watt.

Die fertig installierte Lüftungsanlage mit Wärmetauscher steht bei mir auf einem Regalbrett in der Ecke eines Kellerraumes. Mit großen Erwartungen habe ich nach Inbetriebnahme den Wirkungsgrad der Lüftungsanlage ermittelt. Der Wirkungsgrad hängt dabei nicht nur von dem Aufbau des Wärmetauschers, sondern auch von der Stärke der

verwendeten Ventilatoren ab. So führt die Verwendung von starken Ventilatoren zu einem schnellen Luftdurchlauf, sodass zwar die feuchte Luft schneller gegen trockenere Luft getauscht werden kann, dabei aber weniger Wärme an diese überträgt. Nutzt man dagegen schwache Ventilatoren (z.B. in Kombination mit einem kleineren Durchmesser des Lüftungsrohrs), so erhöht dies den Wirkungsgrad der Wärmeübertragung, verlangsamt jedoch den Luftaustausch.

Mein Aufbau erwärmt mit den genannten Ventilatoren die Zuluft bei einer Außentemperatur von 2,9°C und einer Raumtemperatur von 18,1°C auf 11,8°C. Der Wärmetauscher erreicht somit einen Wirkungsgrad von ca. 58%. Im Verhältnis zu der Leistung der hier verwendeten Lüfter ist der Wärmetauscher also vergleichsweise klein dimensioniert. Ohne den Wärmetauscher zu verändern lässt sich der Wirkungsgrad deutlich steigern, wenn man die Lüfterleistung mit einem Dimmer reduziert. Ich habe dafür testweise ein 230V/2000W-Spannungsreglermodul zwischen das Relais und die Ventilatoren geschaltet. Reduziert man hiermit die Gesamt-Leistungsaufnahme

auf 28W, so erwärmt der Wärmetauscher 3,1°C kalte Außenluft bei 18,0°C warmer Raumluft auf 15,8°C, der Wirkungsgrad des Wärmetauschers steigt also auf ca. 85%.

Ausblick

Der Wärmetauscher läuft nun seit März 2022 bei mir in dem Kellerraum, den ich auch zum Wäschetrocknen nutze. Dabei ist die Raumtemperatur auch in Frostnächten nie unter 14°C gefallen. Ich denke aber schon darüber nach, wie man in das Gehäuse eine größere Wärmetauscherfläche unterbringen kann, um so den Wirkungsgrad weiter zu erhöhen. —pek

Anmerkung der Redaktion: Der Kreuzwärmetauscher baut direkt auf ein Projekt auf, das in früheren Make-Ausgaben vorgestellt wurde. Über solche Weiterentwicklungen und Erweiterungen freuen wir uns sehr! Zögern Sie also nicht, uns eine Mail an mail@make-magazin.de und Bilder zu schicken, wenn Sie selbst eines unserer Projekte in Ihrem Sinne perfektioniert haben.



VSM
WE KNOW ABRASIVES

**VSM Schleifmittel.
Erfolgreich in aller Welt.
Zuhause in Hainholz.**

- > Qualität aus Hannover seit über 155 Jahren
- > In mehr als 70 Ländern der Welt aktiv

www.vsmabrasives.com

**Besuchen Sie uns
auf der Maker Faire**

MADE IN GERMANY

Warmwasser-Sparpumpe

Heizkessel mit Brauchwasser-Erwärmung sind in der Regel mit einer Pumpe für den Warmwasser-Zirkulationskreislauf ausgerüstet, die regelmäßig anspringt, damit auch an entfernten Zapfstellen stets heißes Wasser entnommen werden kann. Diese Komfort-Funktion spart zwar Wasser, verbraucht aber unnötig Energie. Unsere Spar-Schaltung reduziert die Verschwendung auf ein Minimum.

von Uwe Rohne



Ist es Ihnen auch schon so ergangen: Sie wachen als erster Ihrer Familie im Hochsommer auf und hören im Keller die Heizung laufen. Wie kann das sein? Sie sind der Erste und somit hat auch noch niemand geduscht oder warmes Wasser gezapft. Verantwortlich für das notwendige Aufheizen Ihres Kessels sind mit hoher Wahrscheinlichkeit die Wärmeverluste Ihrer Zirkulationsleitung.

Ohne eine Zirkulationsleitung liefe bei langen Rohrleitungen, die man sehr oft in Einfamilien- und Reihenhäusern zwischen Heizkessel und der jeweiligen Zapfstelle findet, zunächst sehr viel kaltes Wasser, bis endlich das Warmwasser den Hahn erreicht hat. Denn auch bei einer guten Isolierung der Rohre nimmt das Wasser in den Rohren mit der Zeit die Temperatur in der verbauten Wand an. Um hier also den Komfort zu erhöhen und den Wasserverbrauch bis zum Erreichen der Wunschtemperatur zu minimieren, werden Zirkulationsleitungen verwendet ①.

Komfort sparen

Eine kleine Pumpe lässt Heißwasser vom Heißwasseranschluss des Kessels beziehungsweise Boilers durch die Rohre zirkulieren. Der Rücklauf ist in etwa in der Mitte des Boilers angeschlossen. So wird erreicht, dass permanent warmes Wasser zur Verfügung steht. Bei sehr alten Installationen wird dieses sogar ohne Pumpe gemacht; dabei verlässt man sich auf die Physik, da warmes Wasser nach oben steigt und so eine natürliche Umwälzung stattfindet. Da dies dauerhaft geschieht, geht hier am meisten Energie flöten; für solche Anlagen eignet sich unsere Schaltung nicht.

Bei Verwendung einer Zirkulationspumpe wird diese entweder dauerhaft betrieben oder mit einer Schaltuhr (extern oder intern im Heizkessel) periodisch eingeschaltet. Ist sie zu Nachtzeiten abgeschaltet, gibt es in dieser Zeit auch nicht sofort warmes Wasser.

In den Zeiten, wo die Zirkulation läuft, jedoch kein warmes Wasser benötigt wird,



Vorsicht, Netzspannung!

Der Einbau unserer Pumpensteuerung erfordert Erfahrungen bei der Verdrahtung netzspannungsführender Teile und die Beachtung einschlägiger VDE-Vorschriften. Lassen Sie den Einbau von einem Fachmann vornehmen, wenn Sie diesbezüglich unsicher sind. Das Hantieren mit Netzspannung ist lebensgefährlich, erst recht im Umfeld gut geerdeter Rohre!

Kurzinfo

- » Funktion der Zirkulationspumpe für Brauchwasser-Entnahme
- » Reduzierung des Brennstoffverbrauchs durch intelligente Pumpensteuerung
- » Realisierung mit Mikrocontroller-Board und Temperatursensor

Checkliste

- Zeitaufwand:**
3 Stunden
- Kosten:**
20 Euro
- Löten:**
Bestücken einer Lochraster-Platine
- Hochspannung:**
VDE-gerechtes Arbeiten mit Netzspannung

Material

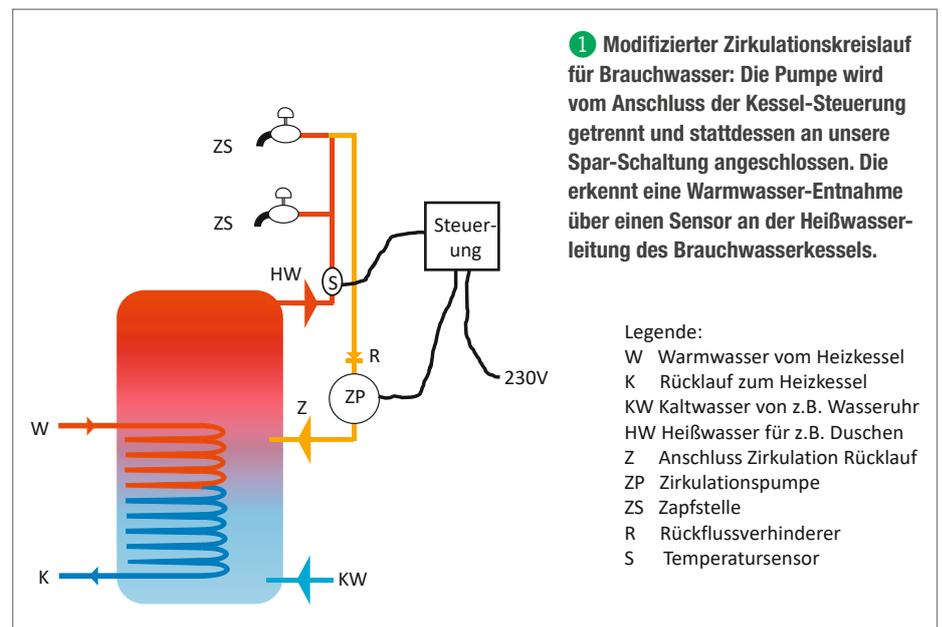
- » Lochrasterplatine oder selbst geätzte Platine
- » ESP8266-12E oder F oder Wemos D1 Mini oder Arduino Nano
- » Netzteil-Modul 230V/3,3V, z.B. Meanwell IRM-01-3.3 bei Wemos USB-Steckernetzteil
- » Temperatursensor Dallas DS18B20
- » Solid-State-Relais 230V/10A
- » Passive Bauteile und Steckverbinder laut Schaltplan
- » USB-Programmieradapter (nicht notwendig bei Wemos D1 und Arduino)

Alles zum Artikel im Web unter make-magazin.de/xw58

werden (trotz Dämmung) lediglich die Wände aufgeheizt. Es entstehen erhebliche Wärmeverluste. Hier setzt nun unser Selbstbauprojekt an. Wenn es gelingt, die Pumpe mit einem intelligenten Schalter zu steuern, läuft diese nur bei tatsächlichem Bedarf und spart damit auf Dauer sehr viel Energie ein – sowohl Strom für den Betrieb der Pumpe als auch Brennstoff zur Erwärmung des in den Rohren befindlichen Wassers.

Im Detail

Für unsere Steuerung ② wird an der Warmwasserleitung etwa 25 bis 30cm vom Boiler entfernt zyklisch die Temperatur gemessen. Wenn diese binnen 5 Sekunden um mindestens 0,1 K (K=Kelvin, eine Temperaturänderung von 0,1 °C) steigt, ist an einer Zapfstelle mehr als rund 50ml Wasser entnommen worden. Die Pumpe wird gestartet



und ein Timer für die nächsten 36 Minuten gesetzt.

Unser intelligenter Pumpenschalter wird also dadurch aktiviert, dass wir einen halben Zahnputzbecher Wasser zapfen; das funktioniert überall im Haus an jeder Warmwasserzapfstelle. Jetzt heißt es einige Minuten zu warten, bis die Leitungen einmal komplett von der Pumpe durchgespült wurden und das warme Wasser entnommen werden kann. An

diese Routine gewöhnt man sich sehr schnell. Welche Einsparungen sich damit erzielen lassen, haben wir im Kasten unten dargelegt.

Schaltung und Einbau

Ein ESP8266 (ein Arduino würde ebenfalls funktionieren) misst jede Sekunde über den Sensor DS18B20 die Temperatur **3**. Die absolute Temperatur ist dabei uninteressant, da

uns nur eine relative Temperaturerhöhung von mehr als 0,1K interessiert. Da auch der Boiler selbst Wärmeverluste hat, wird er in größeren Zeitabständen vom Heizkessel aufgeheizt. Darauf darf unser Sensor nicht hereinfallen. Daher ist er in einem Abstand von 25 bis 30cm Abstand vom Boiler an der Heißwasserleitung zu installieren, sodass tatsächlich nur eine Wasserentnahme detektiert wird.

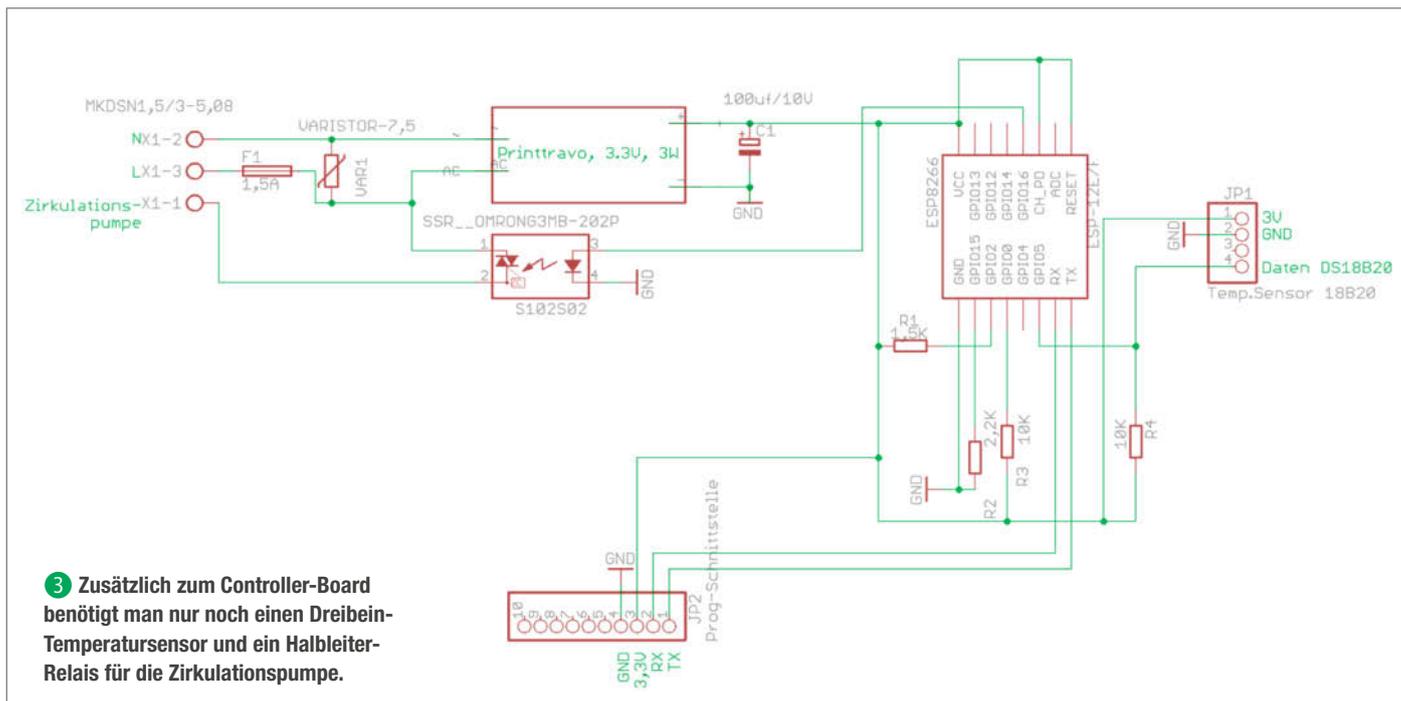
Der Prozessor bewertet bei einer Änderung die jeweils letzten fünf Sekunden und stellt damit zuverlässig auch kleinste Mengen gezapften Wassers fest. Wird dies erkannt, schaltet ein SSR (*Solid State Relay*, Halbleiterrelais) die Pumpe ein. Alternativ wäre auch ein Optokoppler mit Triac oder ein normales Relais mit vorgeschaltetem Transistor und Freilaufdiode möglich.

Nach sechs Minuten, egal was der Temperatursensor inzwischen misst, wird die Stromversorgung zur Umwälzpumpe unterbrochen und ein 30-Minuten-Timer gestartet; solange Warmwasser entnommen wird, muss die Pumpe nicht laufen. Das Spiel beginnt danach von vorn, es sei denn, man fährt in Urlaub: Stellt die Steuerung über 24 Stunden keine Wasserentnahme fest, wird ein kurzer Zwangszyklus durchgeführt, damit die Leitung einmal mit Warmwasser durchspült wird.

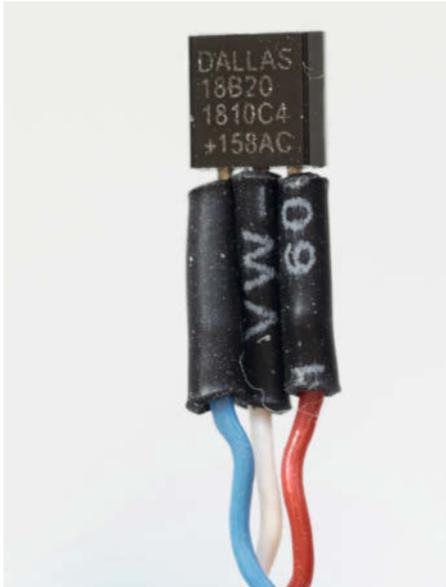
Wir haben in der Umsetzung ein ESP8266-Modul mit 3,3-Volt-Trafo und SSR verwendet, weil der ESP8266 sehr preiswert ist und gerade zur Hand war. Genauso denkbar ist die Verwendung eines *Wemos D1Mini* mit USB-Stromversorgung oder ein schlichter *Arduino Nano*; hierfür sind dann natürlich die Ports in der



2 Provisorische Installation an der Heizungsanlage des Autors: Die Schaltung ist in der Verteilerdose untergebracht, links erkennt man die Leitung zum Sensor (hier unter der Isolierung verborgen).



3 Zusätzlich zum Controller-Board benötigt man nur noch einen Dreibein-Temperatursensor und ein Halbleiter-Relais für die Zirkulationspumpe.



4 Der Temperatursensor DS18B20 liefert keine analoge Ausgangsspannung, sondern digitale Daten über den 1-Wire-Bus von Dallas Semiconductor. Das nötige Protokoll liefert die über den Bibliotheks-Manager zu installierende Arduino-Library DallasTemperature.

Software anzupassen. Die Wireless-Funktionen des ESP8266 benutzen wir nicht; erfahrene Programmierer könnten hier aber zum Beispiel eine Fernwartbarkeit über WLAN nachrüsten. Das Kompilieren mit der Arduino-IDE und das Flashen des Controllers war bereits in diversen Make-Artikeln Thema, eine Anleitung finden Sie unter dem Link im Infokasten.

Beim ersten Test der Schaltung bitte doppelt überzeugen, dass der DS18B20-Sensor **4** nicht versehentlich mit 230V in Kontakt ist und seine Anschlüsse nach der Montage auch nicht das Wasserrohr berühren; sie sollten mit dünnem Schrumpfschlauch isoliert werden. Nach dem Einschalten läuft zunächst der 30-Minuten-Timer an, es heißt also warten. Es

Legionellen

Viele Boiler heizen den Warmwasserspeicher zur Entkeimung wenigstens einmal pro Woche auf 70 bis 80°C auf. Aus diesem Grund ist in unserer Steuerung ein Laufen der Zirkulationspumpe einmal in 24 Stunden einprogrammiert. Das heißt, bei einem 14-tägigen Urlaub läuft die Zirkulation 14 mal und die thermische Desinfektion einer Heizung zweimal. Ob dieses wirklich ausreichend ist, kann von uns nicht garantiert werden.

Einsparungspotential

Im Haus des Autors wird auf ca. 15m Länge bei 28mm Außendurchmesser der Warmwasserleitung und 15mm des Rücklaufs alle 30 Minuten für vier Minuten die Umwälzpumpe für die Zirkulation eingeschaltet, egal ob jemand da ist oder auch nicht. Bei 15m Länge sind das bei ca. 25mm Innendurchmesser 7,4 Liter Wasser und für den Rücklauf noch einmal ca. 1,7 Liter. Somit sind gesamt 9,1 Liter Wasser alle 30 Minuten um ca. 35 Kelvin aufzuheizen. Diese Zirkulation lief über den Heizkessel gesteuert für 18h pro Tag, also 36 mal. Mit Einsatz der Steuerung läuft die Zirkulation noch ca. 15 mal pro Tag (Praxismessung über mehrere Tage).

Um 1 Liter Wasser um 1 Kelvin zu erwärmen, werden 1,16Wh benötigt, in unserem Fall also 9,1 Liter um 35 Kelvin (=°C) zu erhitzen $9,1 \times 35 \times 1,16 = 369\text{Wh}$. Eine Einsparung von 21 Pumpenzyklen macht somit 7,758kWh/Tag, bei 365 Tagen also 2832 kWh pro Jahr. Bei ca. 13 Cent pro kWh Gas ist somit eine Einsparung von 368 Euro pro Jahr zu erwarten. Bei 0,22kg CO₂ je kWh Erdgas ergibt sich eine Einsparung von 623kg CO₂ pro Jahr. Die Verluste durch das Aufheizen der Wasserleitungsrohre ist bei dieser Rechnung noch nicht berücksichtigt, ebenso nicht der Stromverbrauch der Pumpe selbst.

empfiehlt sich, für erste Tests eine Lampe statt der Umwälzpumpe anzuschließen. Sind die 30 Minuten um, den Sensor für einige Sekunden zwischen Daumen und Zeigefinger nehmen. Das reicht, um den Sensor zu erwärmen, und unsere Lampe leuchtet für 6 Minuten.

Die fertige Schaltung lässt sich zum Beispiel in eine Aufputz-Verteilerdose aus dem Baumarkt einbauen. Den DS18B20-Sensor haben wir in der Ausführung ohne Gehäuse gewählt, da so die schnellste Reaktion auf Temperaturänderungen erreicht wird. Nach vorsichtigem Wegschieben der Wärmeisolierung (gegebenenfalls etwas aufschneiden) wird der Sensor

mit einem Kabelbinder auf dem Heißwasserrohr fixiert. Danach die Isolierung vorsichtig zurückschieben.

Perfektionisten werden Pumpe, Schaltung und den alten Pumpenanschluss mit Schuko-Steckverbindern versehen **5**, sodass im Fehlerfall rasch auf die alte Verdrahtung umgesteckt werden kann. Beachten Sie beim Aufbau die VDE-Bestimmungen, mit 230V-Verdrahtungen nicht vertraute Maker sollten vom Nachbau Abstand nehmen. Die Firmware für den Controller, Schaltplan und Platinenlayout (Eagle-Format) finden Sie wie üblich unter dem Link im Info-Kasten. —cm

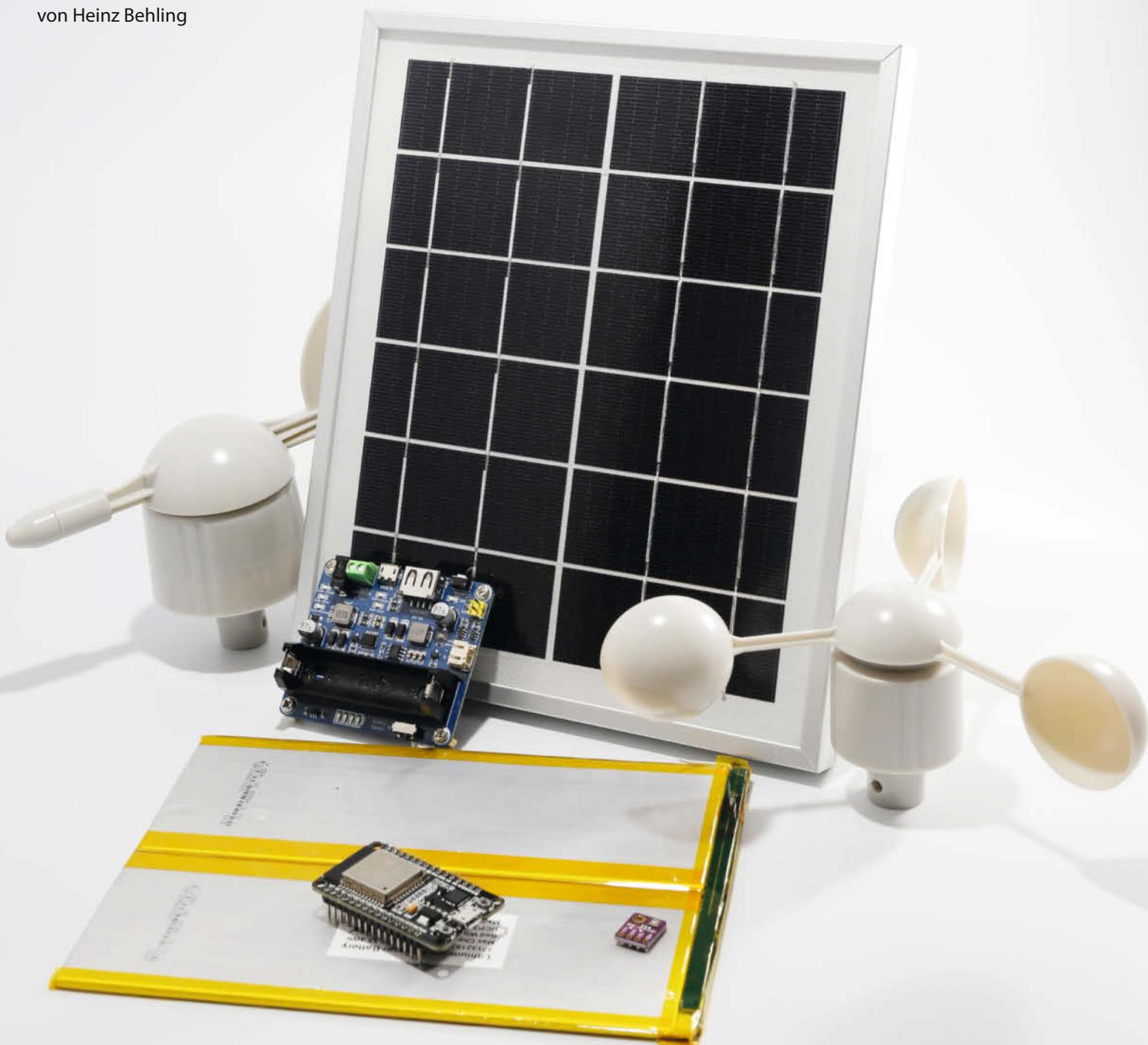


5 Beim Musteraufbau haben wir für den Pumpen-Anschluss Schuko-Steckverbinder vorgesehen, sodass zum Test eine 230V-Glühlampe angeschlossen werden kann; auch die Pumpe bekam einen Schukostecker verpasst.

Solarstrom für Mikrocontroller, Teil 1

Energie wird immer kostspieliger, dabei liefert die Sonne jede Menge davon gratis. Man muss sie nur anzapfen. Selbst bei Kleinverbrauchern wie Arduino, ESP und ähnlichen Mikrocontrollern kann sich das lohnen, besonders, wenn man für Projekte wie eine Wetterstation im Freien auch noch ein langes Stromkabel einspart.

von Heinz Behling



Mikrocontroller wie ESP- oder Arduino-Boards eignen sich sehr gut zur Messwerterfassung. Ihr geringer Strombedarf, der auch aus Akkus getilgt werden kann, macht sie auch ideal für Geräte im Freien, weiter entfernt von einer Steckdose. Doch beliebig lang halten Akkus das auch nicht durch. Da wäre eine völlig autarke Stromversorgung über eine kleine Solar-Anlage am besten. Bei Aufnahmeleistungen von nur wenigen Watt ist das auch auf kleinem Raum und mit geringen Kosten machbar.

Fertiglösung von der Stange

Am einfachsten ist es mit einer Lösung von der Stange, deren Einzelteile es fertig zu kaufen gibt und die nur noch zusammengesteckt werden müssen. Von *Waveshare* gibt es beispielsweise diesen Solar-Laderegler **1**. Mehr über den Laderegler lesen Sie im Kasten *Laderegler*.

Außerdem liefert der Hersteller dazu passende Solarpaneele mit einer Leistung von 5W **2**, zusammen kostet das etwa 25 Euro. Dazu kommt dann nur noch der Preis eines Akkus.

Dieser Artikel zeigt im ersten Teil am Beispiel einer Ganzjahres-Wetterstation, die im 24-Stunden-Betrieb Daten wie Temperatur, Luftdruck, Luftfeuchte und weitere erfasst, wie man solch eine solar stromversorgte Anlage aufbaut und was es dabei zu beachten gibt. Im zweiten Teil in der nächsten Make geht es dann um die dauerhafte Protokollierung der Messdaten auf einer Speicherkarte und ein originelles Anzeigegerät für den Innenraum in Form eines Wetterhäuschens.

Bedarfsrechnung

Doch zunächst zur Auslegung der Station: Hier ist Sparsamkeit Trumpf. Die Mikrocontroller unterscheiden sich deutlich in ihrem Strombedarf (siehe Kasten *Strombedarf gängiger Mikrocontroller*).

Für die Wetterstation ist eine WLAN-Verbindung erforderlich. Das priorisiert die ESP-Boards. Aber auch eine Lösung aus zwei Boards wäre denkbar: So könnte ein kleiner Arduino im 24-Stunden-Betrieb die Messungen vornehmen und auf einer SD-Karte zwischenspeichern. Das ergäbe eine Dauerstrombelastung von unter 25mA – plus dem, was die Sensoren aufnehmen. Dazu käme dann aber noch ein WLAN-fähiges Board, das seine Zeit fast nur im *Deepsleep*-Modus verbringt. In regelmäßigen Zeitabständen erwacht es jedoch für einige Minuten, um die Daten dann nach außen auf einen anderen Computer oder in ein Cloud-System zu senden. Der Strombedarf dafür wäre wegen der geringen Einschaltdauer vernachlässigbar, erkaufte mit dem Nachteil, nicht permanent auf die Daten zugreifen zu können. Die Entscheidung bei diesem Projekt fiel daher stattdessen auf

Kurzinfo

- » Solar-Stromversorgung mit Akkupufferung
- » Energiesparen bei Mikrocontrollern
- » Solar-Wetterstation zum Selbstbauen

Checkliste



Zeitaufwand:

4 Stunden (zzgl. Druckzeit für 3D-Teile)



Kosten:

60 Euro für Temperatur, Luftdruck und Luftfeuchte, (zzgl. 50 Euro für Windmessung)

Material

- » Wemos Lolin32 lite
- » Solar Power Manager
- » Solar-Paneel 6V/5W
- » Wettersensor BME280
- » LiPo-Akku 10000mAh
- » Windgeschwindigkeitssensor WH-SP-WS01
- » Windrichtungssensor WH-SP-WD
- » Kabel-Installationsrohr 25mm mit 90-Grad-Bögen

Werkzeug

- » 3D-Drucker, Schraubendreher, Lötkolben

Mehr zum Thema

- » Daniel Bachfeld, Strom sparen bei ESP-Mikrocontrollern, Make 3/21, S. 84
- » Florian Schäffer, Umweltsensoren für Mikrocontroller, Make 1/19, S. 8
- » Florian Schäffer, Akkutechnik, Make Sonderheft 2018, S. 68

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

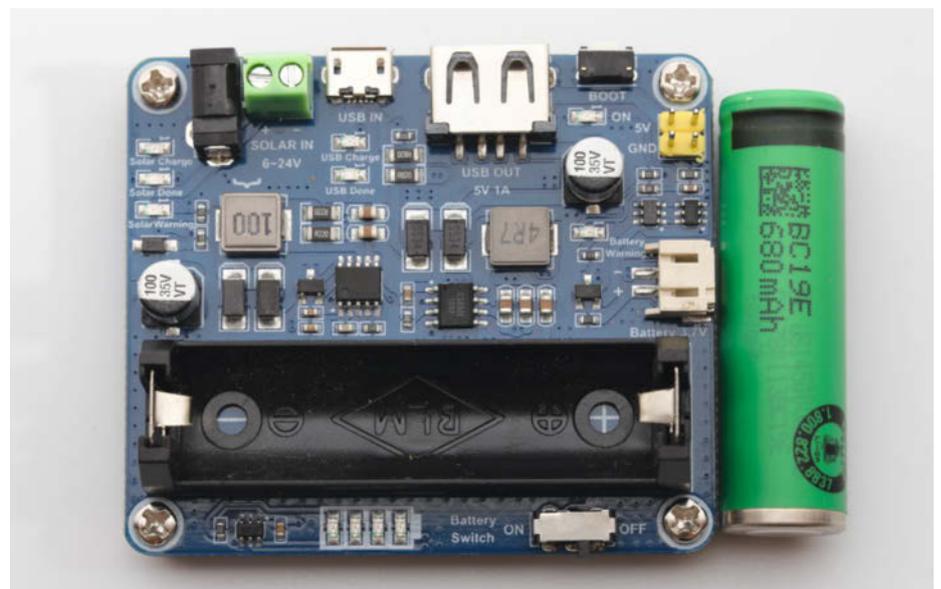


ein *Lolin32-lite*-Board **3**, das mit etwa 45mA Stromaufnahme nur wenig über den Arduino-Boards liegt.

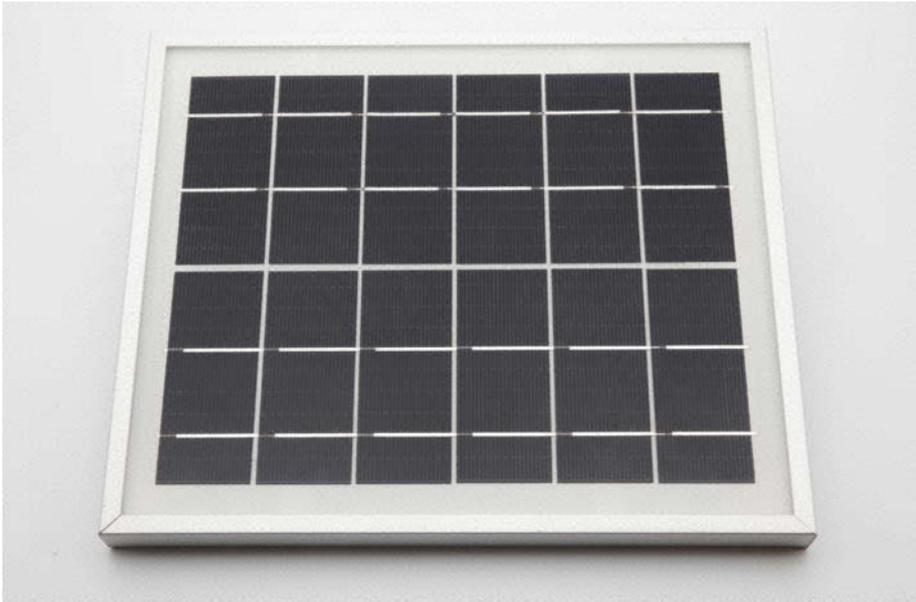
Daraufhin galt es herauszufinden, ob die Solaranlage ausreicht und den passenden Akku zu berechnen: Die Station soll 24 Stunden am Tag ganzjährig laufen, also auch

im Winter, in dem die Sonne nur wenige Stunden scheint. Das ergibt eine Strommenge von $24 \times 45\text{mA}$, also 1080mAh.

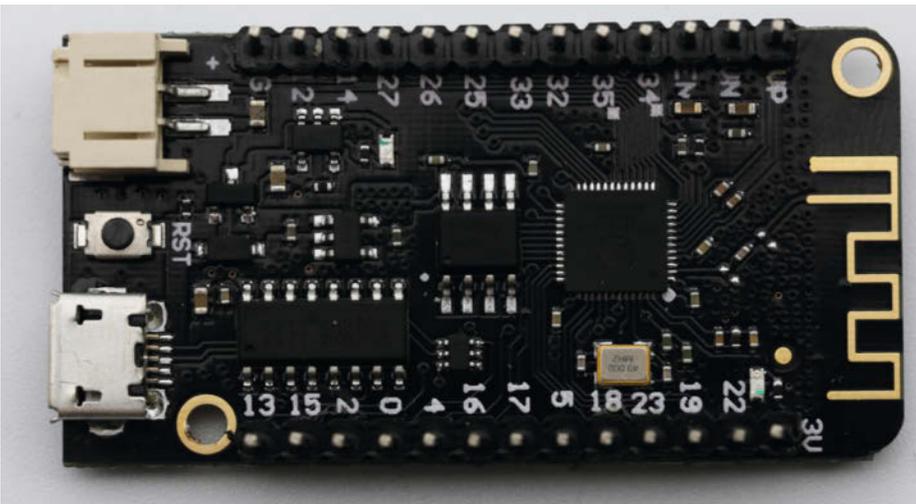
Die hier benutzte Waveshare-Anlage kann bei vollem Sonnenschein etwa 800mA Strom bei 6V Spannung liefern (ergibt 5W Leistung). Versuche ergaben, dass auch bei schlechter



1 Der Laderegler besitzt eine Fassung für eine Akkuzelle vom Typ 14500. Die ist aber mit 680mAh viel zu klein. Die weiße Buchse an der rechten Seite erlaubt jedoch auch den Anschluss anderer 3,7V-Akkus.



2 Das Solarpaneel ist nur etwa 25cm × 19cm groß.



3 Das Lolin32-lite-Board

Beleuchtung (während eines Gewitterschauers und auch in der Morgen-/Abenddämmerung) immer noch etwa 20mA geliefert wurden, bei leicht bewölktem Himmel waren es etwa 200 bis 300mA. Ich schätze daher, dass im Winter etwa 6 Stunden lang durchschnittlich 200mA und im Sommer etwa 13 Stunden lang circa 500mA geliefert werden können. Im Winter liefert die Anlage also täglich 1200mAh. Das reicht, um den 24-Stunden-Bedarf der Station zu decken. Somit passt die Solar-Anlage.

Der Akku muss als absolutes Minimum den Strombedarf einer Winternacht decken, also 18 Stunden Dämmerung und Dunkelheit × 45mA, ergibt 810mAh. Ein Akku von 1000mAh scheint also auf den ersten Blick auszureichen. Doch dann hätte das System keinerlei Reserven etwa zur Überbrückung von Zeiten, in denen Schnee auf dem Solar-Paneel liegt oder wegen anhaltender dunkler Bewölkung die Stromausbeute der Solar-Anlage geringer ist. Daher sollte der Akku dieser Anlage etwa zehn Tage ohne Nachladen durchhalten können. Somit ergab sich ein 10.000mAh-Akku 4.

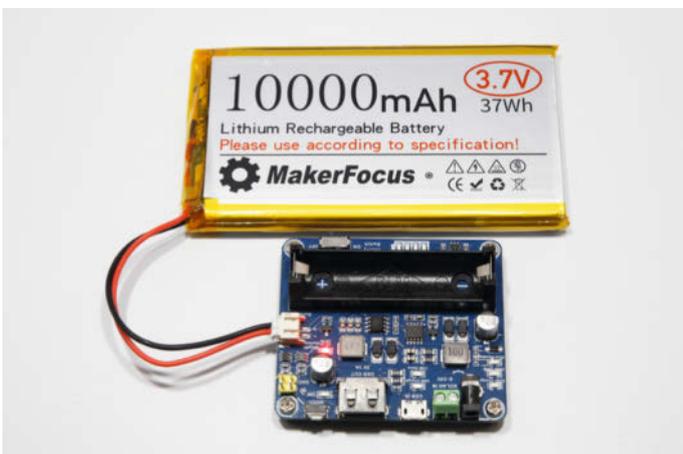
Ich spreche hier vom relativ winterschwachen Hannover als Standort der Anlage. Die Akku-Größe kann an anderen Wohnorten, etwa schneesicheren Gebirgslagen, eventuell zu knapp sein.

Schaltung Solar-Anlage

Die Schaltung ist denkbar einfach: Solar-Paneel, Laderegler und Akku müssen lediglich zusammengesteckt werden, deshalb gibt es hier keine Zeichnung, sondern nur ein Foto 5.

Das Paneel wird wasserdicht vergossen in einem Alu-Rahmen geliefert. Ich habe eine passende Rückwand für den Rahmen per 3D-Druck angefertigt, in dem nicht nur der Laderegler und der Akku, sondern auch der Mikrocontroller Platz finden 6.

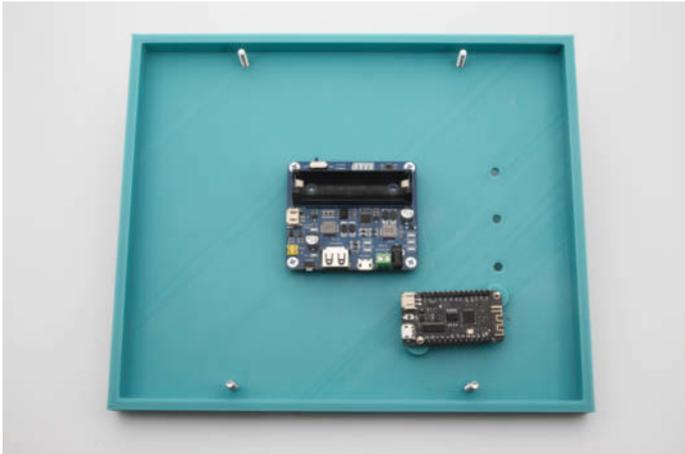
Der Alu-Rahmen hat an der Rückseitenkante vier Löcher, die ich mit M4-Gewindenieten



4 Beim Bestellen des Akkus darauf achten, dass er einen PH2.0-Stecker hat. Dann passt er in die Buchse des Ladereglers.



5 Akku und Paneel in den Laderegler einstecken, das reicht für die Schaltung der Solar-Anlage.



6 Die 3D-Druck-Rückwand bringt mit 230mm Breite so manchen Drucker an seine Grenzen.



7 Der Alu-Rahmen wird mit Nietgewinden ausgestattet.

ausgestattet habe. Die Löcher mussten dazu auf 6mm Durchmesser aufgebohrt werden. Dabei muss man vorsichtig vorgehen, damit das Paneel nicht beschädigt wird 7.

Aufbau Wetterstation

Die Wetterstation soll mit Sensoren für Lufttemperatur, -feuchtigkeit, -druck sowie für die Windgeschwindigkeit und -richtung ausgestattet werden. Für die ersten drei Größen wird ein kombinierter Sensor Typ *BME280* benutzt 8, der mittels I²C mit dem Controller-Board verbunden wird.

Die Sensoren für den Wind sind der *WH-SP-WD* für die Windrichtung und der *WH-SP-WS01* für die Windgeschwindigkeit, beide werden unter dem Markennamen *Snowinsprin* verkauft 9.

Damit zur Verdrahtung: Der Schaltplan 10 zeigt, was wohin kommt. Bitte beachten: Zwei Widerstände zu je 10 Kiloohm werden direkt unter das Controllerboard gelötet.

Zum Controller führt nur das Kabel vom *BME280* und vom Windrichtungssensor. Der Richtungssensor hat an seiner Unterseite eine Buchse, in die der Stecker des Windgeschwindigkeitssensors gehört 11. Das Geschwindigkeitssignal wird dann über das Kabel des Richtungssensors weitergeleitet.

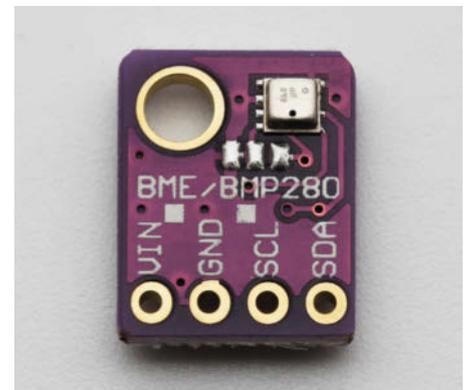
Eine große Bedeutung bei einer Wetterstation hat der mechanische Aufbau. Diese Station soll auf meinem Balkon platziert werden. Dazu habe ich eine Halterung per 3D-Druck angefertigt, die an einem Rohr der Balkonbrüstung angeschraubt wird 12. Daran sitzt dann nicht nur das Solar-Paneel, sondern auch die drei Sensoren werden dort befestigt.

Die Halterung muss aber in der Regel an die lokalen Verhältnisse angepasst werden. Die Druckdateien dafür erhalten Sie über den Link bzw. den QR-Code in der Kurzinfo. Beim Zusammenbau der Halterung habe ich mit

dem Anschrauben der Paneel-Halterung an den Rohrhalter begonnen 13.

Danach kommt die kleinere Rohrhalterung dran. Sie ist für ein 25mm dickes PVC-Kabelrohr bestimmt, auf dem die beiden Windsensoren Platz bekommen. In diese Rohrhalterung müssen zunächst von hinten je eine M3-Mutter in die vier sechseckigen Löcher gesteckt werden. Von der anderen Seite dreht man dann je eine M3×30-Schraube durch die Rohrschelle in die jeweilige Mutter 14. Die Rohrschellen aber noch nicht ganz festziehen. Das geschieht erst, wenn das Kabelrohr drinsteckt.

Jetzt das Kabelrohr einstecken und mit zwei 90-Grad-Bögen versehen. In die nach oben zeigenden Öffnungen der Winkelstücke



8 Die Platine des BME280 hat nur etwa Fingernagelgröße.

Strombedarf gängiger Mikrocontroller

Der Strombedarf von Arduino- und ESP-Boards liegt nur bei einigen Milliampere. Die Unterschiede können aber bei Solar-Betrieb sehr wichtig sein. Die Tabelle unten zeigt den Strombedarf

gängiger Board jeweils im Normal- und im Ruhe-Modus. Zu beachten ist, dass Arduino-Boards im Gegensatz zu ESPs keine WLAN-/Bluetooth-Verbindung bereitstellen.

Board	Strombedarf Normalbetrieb	Strombedarf Schlafmodus
Arduino Pro mini	22,1mA	3,6mA
Arduino Nano	24,7mA	7,4mA
Arduino Uno	47,6mA	32,6mA
Arduino Mega 2560	70,2mA	29,4mA
Wemos D1	74,1mA	4,9mA
Wemos D1 mini	70mA	0,17mA
Wemos D1 mini Pro	70,4mA	0,31mA
Wemos D1 mini Pro V2.0	72,2mA	0,16mA
Wemos D1 mini lite	70,4mA	0,04mA
Wemos Lolin32 lite	45,4mA	1,28mA

Quelle: arduino-projekte.info

9 Die Windsensoren, hier schon auf den 3D-gedruckten Befestigungen



kommen die 3D-gedruckten Adapter für den Windgeschwindigkeits- und den Richtungs-sensor 15.

Jetzt ist die Rückwand des Solar-Panels dran. Dazu werden die vier serienmäßigen M3-Schrauben des Reglers durch 25mm lange Versionen ersetzt 16.

Bevor Sie nun mit der Verkabelung beginnen, sollten Sie die Software *Solarwetter.ino* auf Ihren Computer überspielen und mithilfe der Arduino-IDE auf das Lolin-Board übertragen. Die Software muss in zwei Zeilen angepasst werden: In Zeile 9 ist WLAN-SSID durch den

Namen Ihres WLANs zu ersetzen. In Zeile 10 ersetzen Sie WLAN-Passwort durch das für Ihr Funknetz gültige Passwort 17.

Nun beginnt die Verkabelung entsprechend dem Schaltplan. Ich habe die einzelnen Kabel am Controllerboard angelötet. Die Spannungsversorgung des Lolin32-lite erfolgt mit einem kurzen USB-Kabel. Der Akku wird mit doppelseitigem Schaumstoff-Klebeband in die Rückwand geklebt. Achten Sie auf die Polung des Akku-Anschlusses: Das rote Kabel muss am Pluspol des Ladereglers sein. Falls nicht, die Kabel im Stecker tauschen. Wenn

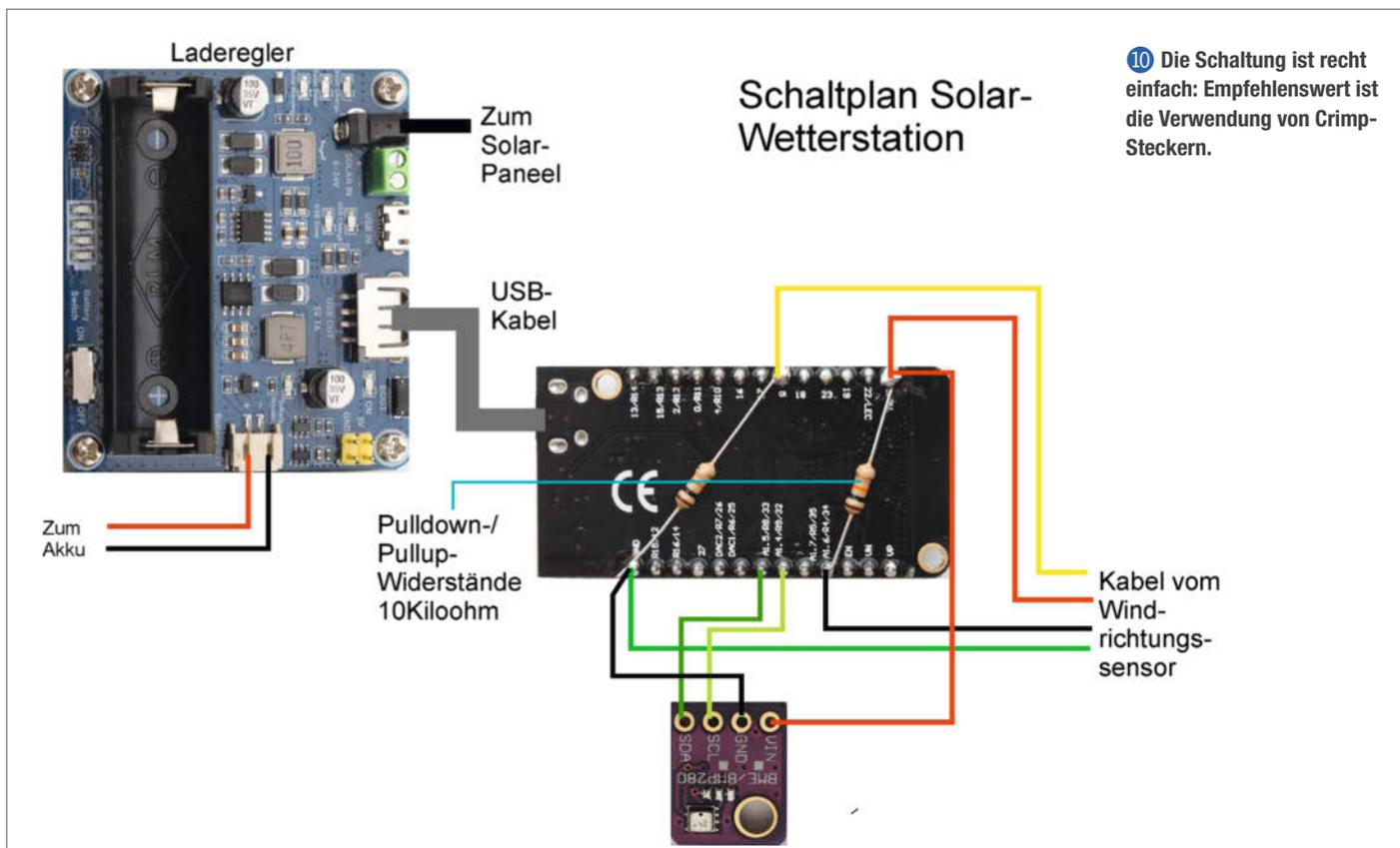
alles eingebaut ist, schließen Sie zum Test das Solar-Panell an und beleuchten Sie es. Auf dem Laderegler darf die LED *Battery Warning* nicht aufleuchten. 18.

Somit ist die Wetterstation bereit, an ihren Einsatzort zu kommen. Der Thermo-Sensor sollte an einem schattigen Ort befestigt werden. In den Druckdateien finden Sie einen kleinen Wetterschutz, der ihn vor direkter Sonnenstrahlung und Regen schützt 19. Den Sensor können Sie darin anschrauben und mitsamt seinem Häuschen dann an geeigneter Stelle befestigen.

Schauen Sie jetzt auf der Web-Oberfläche Ihres Routers nach, welche IP-Adresse Ihre Wetterstation (Name: *esp32-arduino*) erhalten hat. Geben Sie die im Browser Ihres Computers oder Smartphones ein und Sie erhalten die aktuellen Wetterdaten 20. Falls sich die Wetterstation nicht meldet, ist wahrscheinlich der Akku noch nicht geladen. Dann müssen Sie noch einen Tag warten.

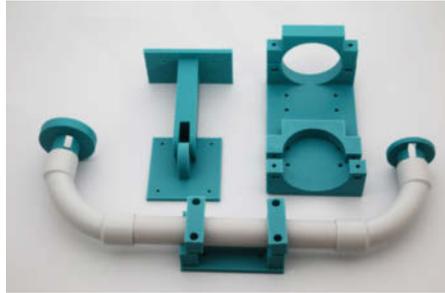
So gehts weiter

Die Wetterstation ist nun schon voll funktionsfähig. Aber man kann alles noch verbessern. In der nächsten Make erfahren Sie, wie man die Wetterdaten dauerhaft auf SD-Karte protokollieren kann. Außerdem wird es eine originelle Anzeigeeinheit für die Daten geben. Ich sage nur: Wetterhäuschen. Lassen Sie sich überraschen. —hgb





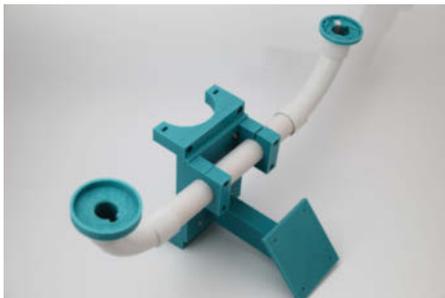
11 In diese Buchse gehört das Kabel des Windgeschwindigkeits-Sensors.



12 Die selbstgedruckte Halterung aus 3D-Druck-Teilen und Kabelrohr



13 Links die Halterung fürs Balkonrohr, rechts der Halter fürs Solar-Panel



15 Die beiden tellerförmigen Adapter tragen später die Windsensoren.



14 Die Muttern in den kleinen Rohrhalter müssen zuerst eingeschraubt werden.



16 Diese langen Schrauben werden durch die Panel-Rückwand gesteckt.

Make:markt

MIKROELEKTRONIK



AZ-Delivery

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

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

METALLBAU



MakerBeam: Mini T-Nut Alu-Profil

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



Was Maker schon alles geschaffen haben!

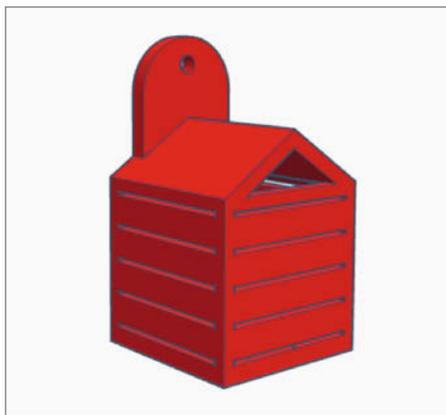
Die Antwort und viele Beispiele finden Leser in unseren Zeitschriften
 „Space – das Weltraum Magazin“,
 „Wissen 2022“ und dem „Urknall“ vieler Computer- und Make-Enthusiasten – dem „Retro Gamer“.

www.emedia.de

```
Solarwetter3 | Arduino 1.8.15
Datei Bearbeiten Sketch Werkzeuge Hilfe

Solarwetter3 $
1 #include <Wire.h>
2 #include <Adafruit_Sensor.h>
3 #include <Adafruit_BME280.h>
4 #include <WiFi.h>
5
6 const int dirPIN = 34;
7 const int RecordTime = 3;
8 const int SensorPin = 5;
9 const char* ssid = "WLAN-SSID";
10 const char* password = "WLAN-Passwort";
11 int dirValue = 0;
12 int dirTime;
```

17 Ohne WLAN kein Zugriff auf die Wetterdaten



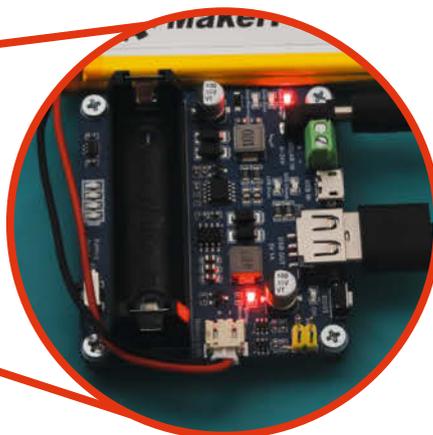
19 Das Häuschen des Thermosensors

Make:

Solar-Wetter

Temperatur:	28.00 Grad Celsius
Luftfeuchtigkeit:	33.97 %
Luftdruck:	1010.43 hPa
Windrichtung:	Sued-Ost
Windgeschwindigkeit:	0.00 km/h

20 Warm, trocken und windstill meldet die Wetterstation.



18 Die Solar-Charge-LED muss jetzt leuchten. Die Battery-Warning-LED zeigt hier hingegen einen verpölpelten Akku an, also umpolen!

Laderegler und MPPT

Der hier verwendete Laderegler verwendet **Max-Power-Point-Tracker-Technik (MPPT)**. Dies ermöglicht, die Energieausbeute eines Solar-Paneels besser zu nutzen.

Das Paneel liefert bei guter Beleuchtung 6V und 800mA Strom, entspricht 5 Watt. Der Akku darf aber nur mit einer maximalen Spannung von 4,2V geladen werden. Also ist dazwischen eine Regelung notwendig. Einfache Laderegler leiten den Strom der Solarzellen nicht kontinuierlich an den Akku, sondern verbinden Zelle und Akku immer nur impulsweise für wenige Millisekunden. Die Spannung am Akku steigt nicht schlagartig auf die Spannung des Paneels an, sondern braucht etwas Zeit. Sobald seine maximale Ladespannung erreicht wird, kappt der Regler die Verbindung wieder. Je mehr Strom der Akku aufnehmen kann (also je leerer er ist), umso länger dauern diese Ladeimpulse. Dies ist die sogenannte **PWM-Regelung (Pulsweitenregelung)**. Nachteil: Da die meiste Zeit kein Strom fließt, wird nur ein sehr geringer Anteil der von den

Solarzellen bereitgestellten Energie zum Laden des Akkus benutzt. Häufig bleiben dann von dem im Beispiel genannten 800mA im Mittel weniger als 100mA übrig.

Bei einem MPPT-Laderegler wird hingegen die Belastung der Solarzelle so geregelt, dass das Produkt der von ihr gelieferten Spannung und dem Strom (das entspricht der elektrischen Leistung) möglichst groß ist. Solche Regler sind imstande, den Lastwiderstand der Solarzelle zu ändern. Der Regler steuert den Widerstand ständig so, dass sich die größtmögliche Leistung ergibt. So kann es durchaus sein, dass unser Solar-Paneel so stark belastet wird, dass die Spannung auf 5V absinkt. Liefert die Zelle dann aber 1100mA Strom, ergibt sich eine Leistung von $5V \times 1,1A$, also 5,5 Watt.

Die Spannung, die dann zum Akku fließt, wird dann nicht per PWM, sondern mit einem richtigen Spannungswandler auf den besten Wert für den Akku angepasst, also beispielsweise 4,2V. Die liegen dann



Mit diesem Mäuseklavier stellt man die richtige Spannung ein.

aber nicht als kurze Impulse, sondern dauernd am Akku an. Bei 4,2V wäre dann ein Strom von 5W/4,2V, also 1190mA möglich. Damit diese einfache Arbeitspunktanpassung über die Spannung funktioniert, muss der Laderegler auf die maximale Spannung der Solarzelle eingestellt sein. Unser Regler besitzt dafür kleine Schalter an der Unterseite.

Moderne MPPT-Laderegler berücksichtigen aber noch weitere Werte, etwa die Sonneneinstrahlung, die Temperaturen von Akku und Solarzelle. Dadurch lässt sich vor allem bei größeren Anlagen erheblich mehr Energie aus der Sonne herausholen.



Maker Faire Hannover

Ein feuerspeiender Vogel, brennende Laufräder, aber auch Know-how rund um Energiegewinnung mit Photovoltaik – die Maker Faire lockt im September mit einem breiten Programm nach Hannover.

von Kristina Fischer

Endlich wieder live und in echt: Auf der *Maker Faire Hannover* am 10. und 11. September kommen in den drei Ausstellungshallen des *Hannover Congress Centrums* sowie im angrenzenden Fontänen-Garten wieder Maker aus ganz Deutschland und einigen europäischen Ländern zusammen und zeigen, was in den vergangenen Monaten in ihren Werkstätten an faszinierenden Projekten entstanden ist.

Ein feuriger Blickfang wird das Wesen *Garuda* sein: halb Maschine, halb Vogel, aus der indischen Mythologie stammend und doch hydraulische Maschine, die sich bis zu 6 Meter hoch entfalten und ihre 15 Meter spannenden Flügel ausbreiten kann. Für besonders Mutige steht das 5 Meter große Feuerrad *SteelWheel* des *Propane Punk Circus* bereit, in dem man auch laufen kann – willkommen im Hamsterrad. Mit den beliebten R2D2-Robotern des *R2 Builders Club e.V.* sowie unterschiedlichsten Cosplay-Charakteren des *Northern Outpost e.V.* kann man Bekannte aus dem *Star Wars*-Imperium treffen, aber es sind auch noch andere Parallelwelten zu entdecken. Die *Dark Gallery* bietet besonders spektakuläre Lichtinstallationen – was genau, ist noch geheim.

Auf der Außenbühne im Stadtpark gibt es Unterhaltung und Musik. Dort sind auch von YouTube bekannte Holz- und Stahlhandwerker aktiv: So werden *Der Kompressorschrauber* und *thisa* während der Veranstaltung einen Kofferranhänger zu einem vollwertigen Camper umbauen, wobei sie ihr Materialbudget selbst auf 300 Euro begrenzt haben.

Das eigentliche Highlight der *Maker Faire* sind aber wie immer die vielen unterschiedlichen Maker mit ihren vielseitigen Projekten. In diesem Jahr stehen die Themen *Energie sparen* und *Energie gewinnen* ganz besonders im Fokus. So kann man sich direkt vor Ort bei Make-Autor Guido Burger über die DIY-Solar-tische von Seite 10 informieren und – nach Voranmeldung – zusammen mit dem Make-Autor Sebastian Müller ein eigenes Sonnenkraftwerk für den Balkon bauen. Energie ist auch Thema im begleitenden Vortragsprogramm. Bei *Glückskee* der Uni Hannover gibt es Experimente für autonome Pflanzenzucht für die ISS zu entdecken und die *Hobbybrauer Hannover* zeigen, wie man mit angewandter Biochemie und Technik zum eigenen Bier kommt. *Das FabLab München* präsentiert



diverse Open-Hardware-Projekte, unter anderem ein 3D-Drucker für Metall. Und an vielen Stellen kann man als Besucherin und Besucher selber mitmachen, ob in der Workshop-Area oder direkt an vielen Ausstellerständen.

Unter dem Motto *Inspiration für den MINT-Unterricht: Learning by Making* gibt es am Maker-Faire-Samstag einen kostenlosen Lehrer-Fortbildungstag mit abschließender Podiumsdiskussion.

Wer kurz entschlossen noch als Maker ausstellen will: Wir haben noch Platz und den Anmeldeschluss bis einschließlich 12. August verlängert (siehe Link unten). Auf Besuchertickets gibt es mit dem Code „MAKE422“ bis zum 21. August noch 20 Prozent Rabatt im Online-Ticketshop.

—pek

► maker-faire.de/hannover

Raumsuche in Ulm

Nach den Querelen um den neuen Nutzungsvertrag des Ulmer *Ver-schwörhauses* (s. a. Werkstattberichte in Make 3/22), ist der Verein nun auf der Suche nach neuen, möglichst barrierefreien Räumlichkeiten.

verschwoerhaus.de

Maker-Termine

oktolog – Art & Maker Camp

13. – 20. August

Alter Schlachthof Wels
waschaecht.at/oktolog

VVoid Camp

18. – 21. August

Zeltplatz Messerschmiedmühle, Perlesreut
tickets.vvoid.camp

Hacken Open Air

30. August –

3. September

Gifhorn

hackenopenair.de

There is no Camp

1. – 4. September

Vogtland

thereisno.camp

Hacks on the Beach

9. – 11. September

Strandpavillon Solitude, Flensburg

chaostreff-flensburg.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

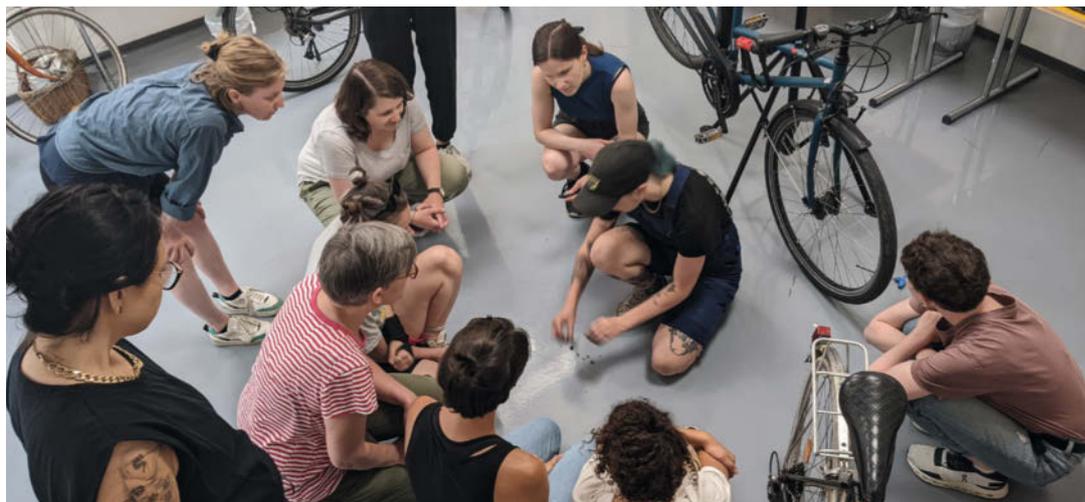


Bild: HappyLab Wien / Joanna Kowolik

Frauen an die Werkbank

Beim Female* Maker Month standen die Makerinnen im Vordergrund

Frauen stellen die Hälfte der Bevölkerung, aber in Hacker- und Makerspaces liegt ihr Anteil meist darunter. Im Wiener *HappyLab* will man das ändern und hat den Mai zum *Female* Maker Month* ausgerufen. Einen Monat lang gab es unter dem Motto „Frauen an die Werkbank“ zahlreiche Angebote – exklusiv von und für Frauen. Workshops, ein Werkstatttrundgang und eine Ausstellung luden Interessierte ein, das *HappyLab* und seine Maschinen kennenzulernen und selbst tätig zu werden. Immer mit dabei: Vorbilder aus der lokalen Maker-Community wie die Fahrradladen-Besitzerin Ana Powdrill, die im Workshop „Repair and Ride“ die Grundlagen der Radreparatur vermittelte. Einen Einstieg in elektronische Schaltungen, aber auch zu Nachhaltigkeitsfragen gab es bei „Ethische Hardware“ mit Stefanie Wuschitz vom feministischen Kollektiv *Mz* Baltazar's Lab*.

Über die Veranstaltungen hinaus war die Werk-schau *Eyes on female* makers* zu sehen. Ausgestellt wurden Arbeiten von Makerinnen des *HappyLabs*, wie etwa Silberschmuck aus dem Lasercutter und ein Sprechfunktrainer. Einblicke in noch unfertige Projekte wie eine Massageliege für Schwangere und einen E-Scooter aus Holz gab es für die Teilnehmerinnen des Werkstatttrundgangs, mit dem der Monat eröffnet wurde. Insgesamt 100 Frauen waren am Ende im *HappyLab* zu Gast und das Feedback durchweg positiv. Auch in Zukunft soll es daher Veranstaltungen exklusiv für Makerinnen geben. Geplant sind regelmäßige Vernetzungstreffen, um den Austausch zu fördern, sowie weitere Workshops nur für Frauen. —hch

► happylab.at/de_vie/femalemakers

Ideenwerkstatt für Studierende

In Wismar wird jetzt in der StartUpYard gewerkelt

Seit dem vergangenen Sommer ist an der Hochschule Wismar die *StartUpYard* für die Studierenden geöffnet. In der interdisziplinären Ideenwerkstatt können sie an eigenen Ideen tüfteln und zu digitalen und anfassbaren Prototypen umsetzen. 3D-Drucker, Lasercutter, Schneidplotter und Nähmaschinen helfen bei der Umsetzung. Da die *Yard* vor allem von Studis genutzt wird, waren sie von Anfang an bei der Gestaltung dabei und haben etwa die Raumaufteilung bestimmt, die *Couchlandschaft* selbst gebaut und sich erfolgreich für eine barrierefreie Toilette eingesetzt. In dem aktuellen Sommersemester wurden nun die Außenanlagen von der Gruppe *UrbanYardening* neu gestaltet.

Neben Workshops rund um die Maschinen gibt es in der *Yard* auch übergreifende Weiterbildung, etwa zu Stress- und Zeitmanagement und beim *CouchTalk* erzählen Berufstätige aus ganz unterschiedlichen Bereichen, wie es nach dem Studium weitergehen kann. Ein weiterer Schwerpunkt sind die Wettbewerbe,

bei denen Projekte entwickelt und sogar finanziert werden können. Bei der ersten Runde des *KickStart*-Ideenwettbewerbs arbeiteten sechs Teams ein halbes Jahr an ihren Prototypen – dabei entstanden eine App für studentische Nachhilfe, eine wartungsarme Duschkabine, Schmuck mit variablem Grundkonzept, eine alternative Schütte für Unverpacktläden und ein Bügelbrett, das auch Massageliege sein kann. Der Bewerbungsschluss für die zweite Runde endet übrigens am 31. Juli. —hch

► hs-wismar.de



Bild: Hochschule Wismar / Sara R. Scholl

Mach's auf

Eine Wiener Initiative bringt Inklusion und Nachhaltigkeit zusammen

Wenn Besucher an der Tür des Wiener Hackspaces *Metalab* klingelten, konnte Oliver Suchanek bis vor kurzem nicht die Tür öffnen – denn das Vorstandsmitglied ist gehörlos. Inzwischen gibt es daher zusätzlich eine (selbstgebaute) Lichtklingel. Damit auch andere Makerspaces und Werkstätten für gehörlose und schwerhörige Menschen besser zugänglich werden, hat Suchanek das Forschungsprojekt *Mach's auf!* gestartet. In einer Reihe an Vorträgen und Workshops soll erarbeitet werden, welche Barrieren es bei der Inklusion der Gehörlosen-Community in Makerspaces gibt und wie diese abgebaut werden können. Ein weiteres Ärgernis, so Suchanek, sind etwa Video-Tutorials ohne Untertitel. Neben dem besseren Zugang zur Technik stehen noch Nachhaltigkeit und Reparaturkultur im Fokus.

Seit Mai gibt es im Metalab bereits ein MeetUp in Österreichischer Gebärdensprache (ÖGS), um die Gehörlosen- und die Maker-Communities miteinander zu vernetzen. Auch beim *Netzpolitischen Abend* war das Projekt zu Gast – natürlich mit ÖGS-Dolmetschung. Nach zwei Jahren soll schließlich ein Handbuch mit Vorschlägen und Best Practices erscheinen. Weitere Veranstaltungen werden auf der Webseite und über Instagram angekündigt. *Mach's auf!* ist eine Kooperation der TU Wien, des Metalabs und equalizent, einer Bildungsfirma für Gebärdensprache. Es ist eines von sechs Projekten, die vom *Action for Sustainable Future hub* ausgewählt wurden und mit bis zu 100.000 Euro gefördert werden. —hch



► machs-auf.at

Förderung in Coburg

Das *Creapolis* in Coburg wird für weitere fünf Jahre über die Bund-Länder-Initiative „Innovative Hochschule“ gefördert. creapolis-coburg.de

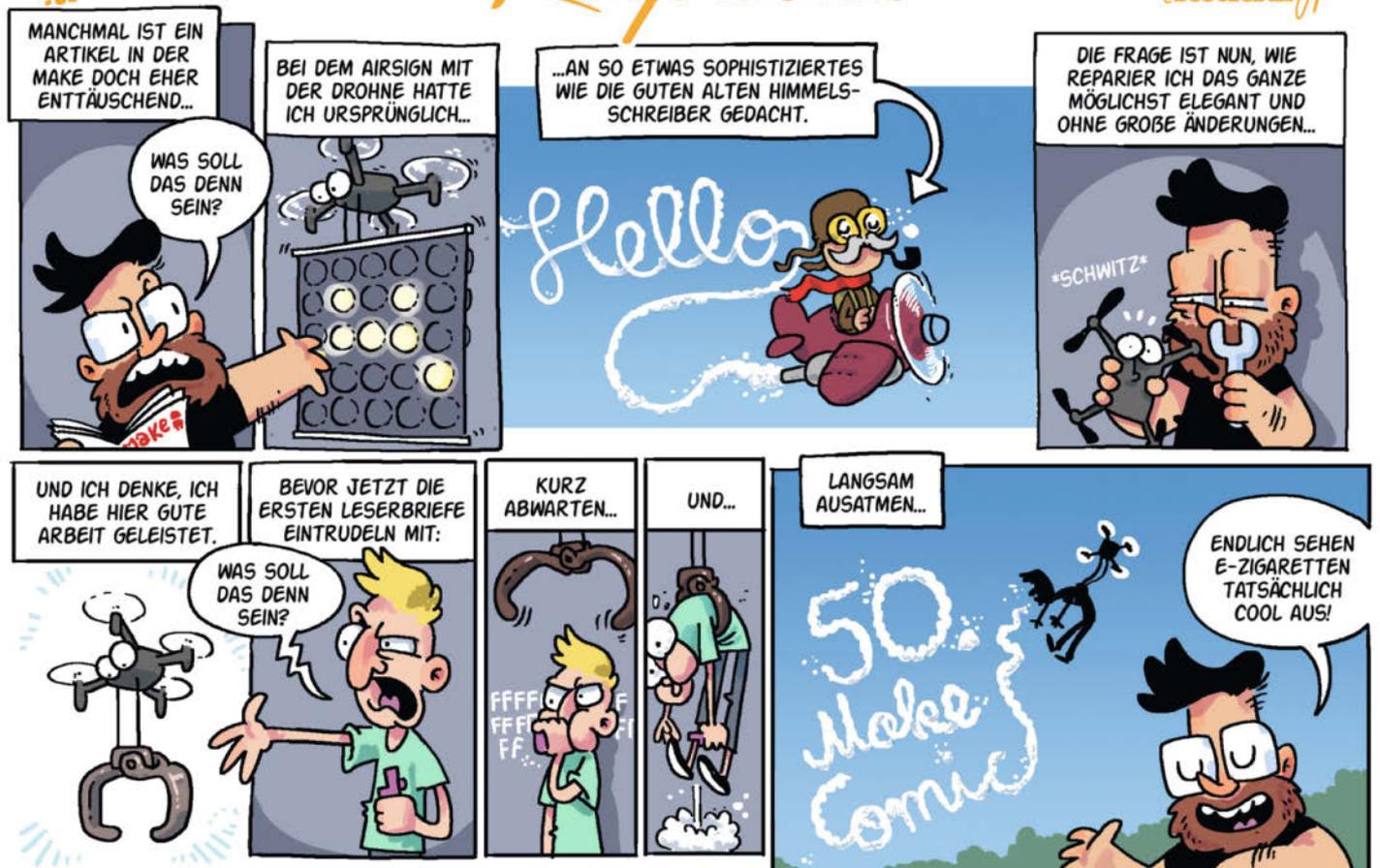
Makerspace in Pirmasens

Die Hochschule Kaiserslautern hat am Standort Pirmasens einen neuen Makerspace für Studierende eröffnet. hs-kl.de

50 AirSigns, aber wirklich.

Kolophonium

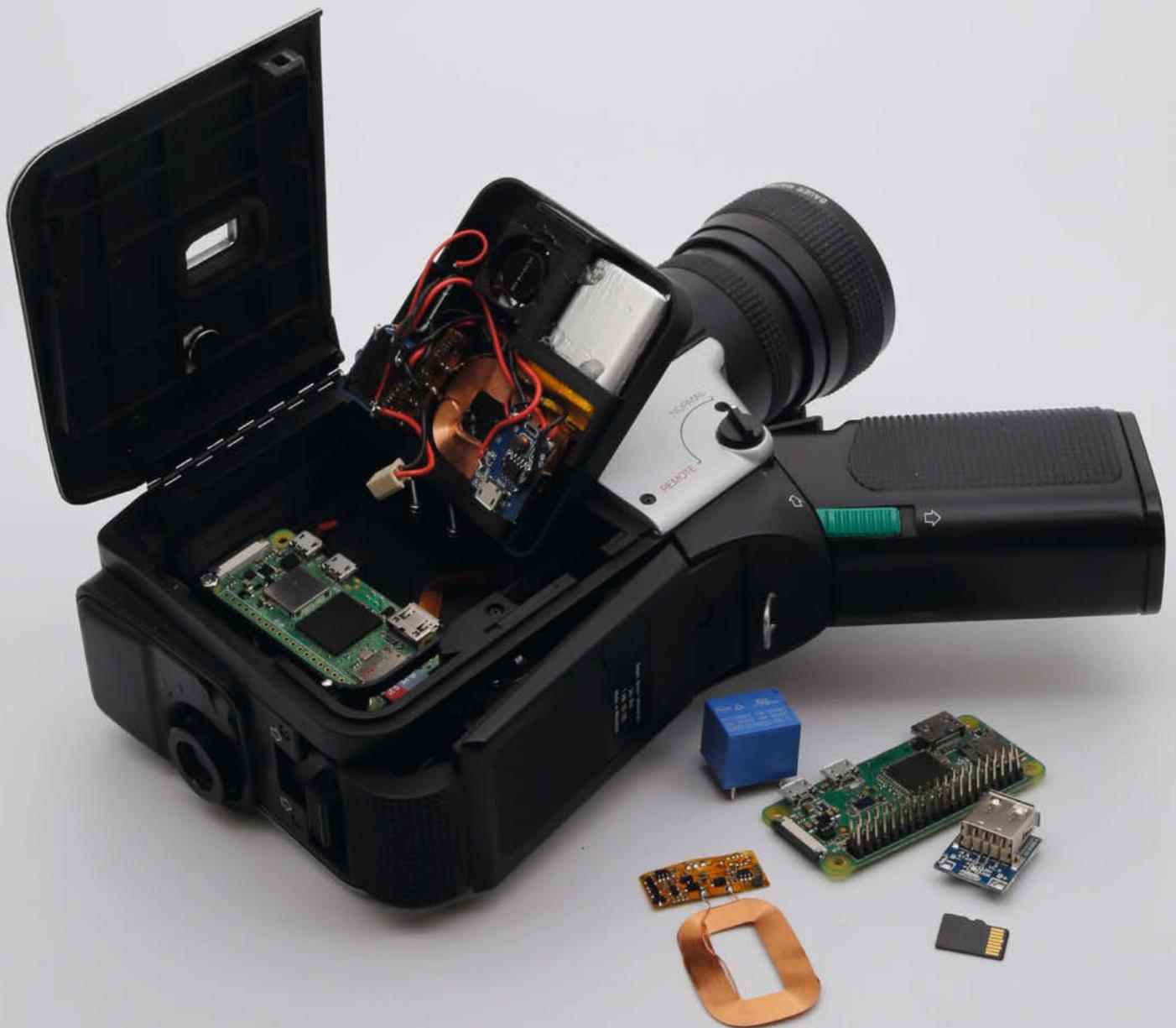
von und mit @beetlebum



Raspi-Super8

Schlechte Film-Versorgungslage und übermächtige Smartphone-Konkurrenz haben Super8-Kameras zu teurem Schrott gemacht. Dabei haben die oft gute, lichtstarke Objektive mit Motorzoom und ähnlichen Feinessen. Grund genug, eine elektronische Filmkassette zu entwickeln, die den Feinmechanik-Kunstwerken mittels Raspi Zero 2 wieder Leben einhaucht.

von Heinz Behling



Womit man doch früher als Hobby-Kameramann/-frau zufrieden war: Filmkassetten mit nur etwas mehr als drei Minuten Aufnahmezeit, stromhungrige Scheinwerfer für Innenaufnahmen (1000W waren damals üblich) und zeitaufwändiges Einsenden des Films in ein Entwicklungslabor. Dazu kam dann noch die komplizierte Vorführung mittels Projektor und Leinwand im abgedunkelten Raum.

Der preiswerte Einstieg war damals bei vielen Hobby-Filmern die Kamera, die gab es schon ab 100 DM im Versandhaus. Solche einfachen Teile hatten aber meist ein relativ schlechtes Objektiv, was häufig zum Kauf eines besseren Aufnahmeegeräts führte. Auch davon gab auch eine Menge mit Motorzoom, Zeitlupen- und Zeitrafferaufnahmen und sogar mit Sound. In der Spitzenzeit Anfang der 70er-Jahre waren etwa 1000 Modelle auf dem Markt.

Ob Billig-Kamera oder semiprofessionelles Equipment für vierstellige Preise, allen gemein war das teure Aufnahmematerial: Eine Filmkassette kostete mindestens 10 DM zuzüglich Entwicklungskosten und das für drei Minuten Filmspaß! Lediglich bei Kodak war die Entwicklung im deutlich höheren Filmpreis enthalten. Heutzutage gibt es wieder Filmkassetten zu kaufen, doch liegen die Preise inzwischen bei 35 bis 60 Euro. Das leisten sich nur sehr engagierte Filmer. Kein Wunder also, dass Super8-Filmen in den 80er Jahren des vorigen Jahrhunderts zunächst durch Video-Camcorder und später durch Smartphones völlig verdrängt wurde.

Seitdem liegen die Kameras als Edelschrott irgendwo herum. Besonders bei teuren Exemplaren tut das dem einst so stolzen Filmer weh. Versuche, sie wenigstens noch auf eBay zu Geld zu machen, schlagen meist fehl (oder die Kameras wechseln für eher symbolische Beträge den Besitzer), und so bleiben die Kameras im Schrank oder wandern in den Elektroschrott.

Kurzinfo

- » Super8-Filmkassette mit Raspberry Pi Zero 2
- » MotionEye als Aufnahme-Software
- » Einstellungen per Web-Oberfläche auf dem Smartphone

Checkliste



Zeitaufwand:
4 Stunden zzgl. 3D-Druckzeit



Kosten:
50 Euro (zzgl. Preis für Raspberry und Super8-Kamera)



3D-Druck:
Druck der Ersatz-Filmkassette

Material

- » Raspberry Pi Zero 2 W
- » Super8-Kamera
- » Mikro-SDXC-Card
- » Mini-Kameramodul für Raspberry Zero mit 15cm Kabel
- » Akku-Laderegler ST 6815
- » LiPo-Akku 3,7V, 1000mAh 9mm x 30mm x 40mm, z.B. Eremit 903040
- » Reedkontakt
- » Neodyn-Magnet rund, 10mm x 2mm
- » Relais SRD-05VDC-SL-C
- » Qi-Empfänger und Ladegerät
- » Schrauben M2 x 16 mit Müttern und Unterlegscheiben
- » Zwei-Komponenten-Kleber (2K) schnell aushärtend, 2-5 Minuten
- » Druckfilament schwarz (PLA oder PETG)

Werkzeug

- » Feinmechaniker-Schraubendreher
- » kleine Rundfeile

Mehr zum Thema

- » Heinz Behling, Super8-Tonfilm-Kamera, Make 4/20, S. 88
- » Heinz Behling, Fernseh-Start, Make 2/18, S. 66
- » Video: Rundgang bei Heise mit der Super8-Kamera

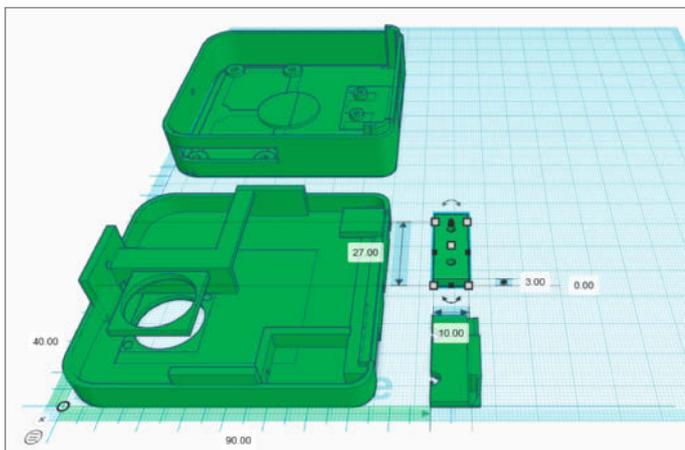


Alles zum Artikel im Web unter make-magazin.de/x55c

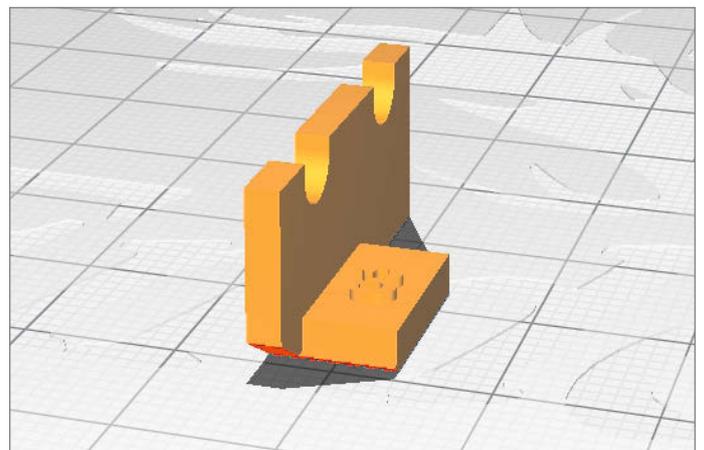


Es wurmt mich gewaltig, voll funktionsfähige Geräte zu entsorgen. Deshalb dachte ich über eine Lösung für das Problem der teuren Versorgung mit Verbrauchsmaterial nach.

Zufällig fiel mir auf, dass das kleine Fenster in einer Super8-Kamera, durch das der Film belichtet wurde, ziemlich genau der Sensorgröße eines 2-Megapixel-Kameramoduls für



1 Die vier Teile der Filmkassette wurden mit Tinkercad konstruiert.



2 So gedruckt wird die Klebefläche für den Kamerasensor völlig glatt.



3 Diese Teile gehören in den Deckel (von links): Akku, Relais, Lüfter, Reed-Kontakt, Laderegler und Qi-Empfänger.

den Raspberry Pi entspricht. Könnte man da nicht einen Raspberry Zero nebst Kameramodul in ein 3D-Druck-Gehäuse in Form einer Super8-Kassette einbauen?

Der Umgang damit sollte natürlich genauso einfach sein wie ursprünglich: Daher musste die Kassette keinerlei elektrische Verbindung zur Kamera benötigen, eine eigene interne Stromversorgung haben und die Filmaufnahme wie bisher durch den Auslöser der Kamera gestartet werden. Und die Videos sollten sich einfach betrachten lassen, am besten auf einer Web-Oberfläche, damit moderne Menschen sich die Filme auf ihrem Smartphone anschauen können. Nach Investition von einigem Hirnschmalz ergab sich dann dieses Projekt.

Stromversorgung

Klar, in der Kassette sollte ein Akku die Stromversorgung übernehmen. Ein Raspberry Zero 2 braucht mit Kameramodul etwa 500mA an Strom. So sollte ein 1000mAh-LiPo-Akku daher für etwa zwei Stunden Betrieb reichen. Das ist ja immerhin etwa 36mal so viel wie die Aufnahmezeit einer Originalkassette.

Der Akku muss auch geladen werden. Ein kleiner Laderegler sollte dabei sowohl das Aufladen überwachen als auch 5V für die Versorgung des Raspi bereitstellen. Und das Laden sollte ohne von außen zugängliche Anschlussbuchse geschehen, denn das erleichtert die Anordnung der zahlreichen Teile im engen Innenraum der Kassette. Daher kam ich auf Qi, die drahtlose Ladetechnik, die vor allem bei Smartphones verwendet wird.

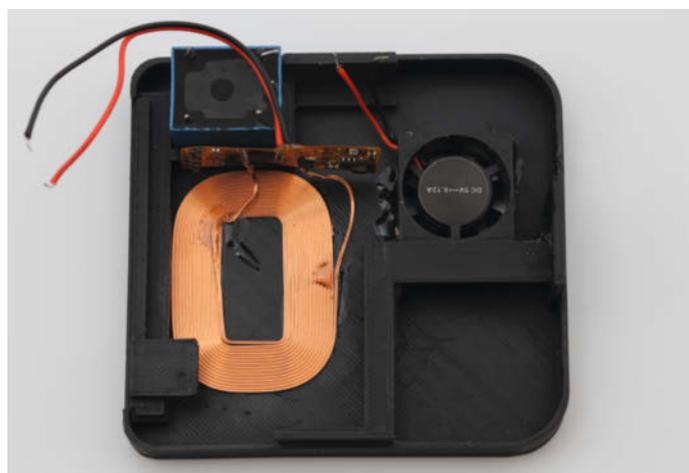
Dritter Teil der Stromversorgung: der Ein-/Aus-Schalter. Auch hier wollte ich auf einen von außen zugänglichen Schalter verzichten. In den Kassettenfächern der Kamera geht es eng zu. Da bleibt kein Platz für einen abstehenden Schaltknopf. Als eleganteste Lösung erschien mir schließlich ein Magnetschalter (Reed-Kontakt) im Inneren der Kassette, der



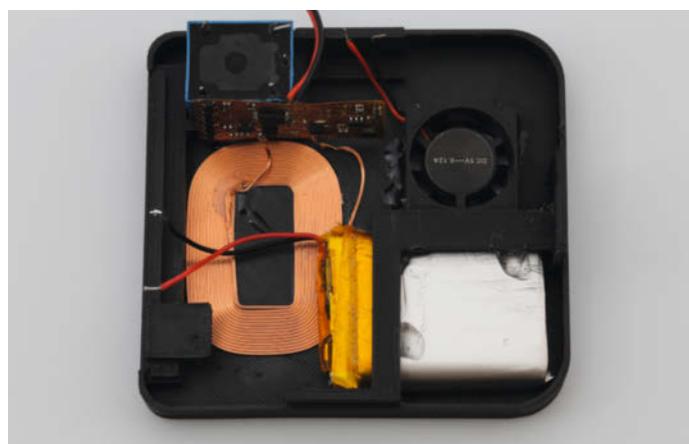
5 Beim Lüfter muss der Kleber sparsam dosiert werden, damit der Rotor nicht verklebt.



4 Der Deckel muss wegen der Halterungen unbedingt mit Stützstrukturen gedruckt werden.



6 So sollte die Spule liegen, damit sie später nicht vom Akku verdeckt wird, was zu Ladeproblemen führen kann.



7 Das Sichern des Akkus gegen Verschieben ist wichtig, damit er nicht die Spule verdeckt. Dann könnten einige Qi-Ladegeräte Probleme bekommen, den Empfänger zu erkennen.

von einem am Kameradeckel befestigten kleinen Magneten eingeschaltet wird. Der Raspberry würde dann also nach dem Einlegen der Kassette und Schließen des Deckels ein- und durch Öffnen wieder ausgeschaltet.

Ein paar Experimente ergaben da noch ein Problem: Selbst Reedkontakte, die laut Datenblatt einen Strom von 1A-Schalten verkraften sollten, klebten nach dem ersten Schaltvorgang dauerhaft zusammen und schalteten auch nach Entfernen des Magneten nicht mehr ab. Grund dafür dürfte wohl ein zwar kurzer, aber sehr hoher Stromstoß beim Einschalten des Raspi sein, der die winzigen Kontakte regelrecht zusammenschweißst. Lösung: Der Reedkontakt sollte den Raspi über ein Relais einschalten. Hier wäre auch ein Transistor (MOSFET) möglich, doch fand sich keiner in meinen Teilebeständen.

Auslöse-Übertragung

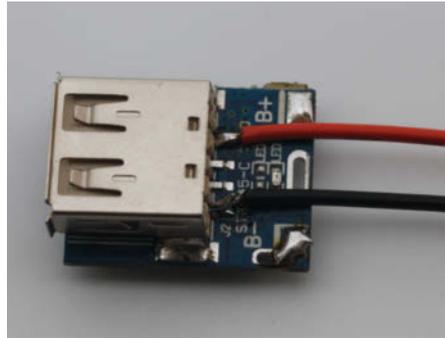
Wie aber bekommt man ohne elektrische Verbindung das Auslösesignal von der Kamera zum Raspberry in der Kassette? Da half mir mein Nistkasten aus der Make 2/18. Dessen Software *Motioneye* konnte bewegungsgesteuert Videos aufnehmen. Das sollte sich doch nutzen lassen. Zur Erklärung: Vor dem Bildfenster der Kamera sitzt eine rotierende Blende, die immer, wenn der Film transportiert wird oder die Kamera steht, den Lichtgang blockiert. Ein Öffnen der Blende erfolgt also nur, solange der Auslöser der Kamera gedrückt wird. Das dann auf den Kamerasensor fallende Licht sollte *Motioneye* als Bewegung erkennen können und so die Videoaufnahme starten, ganz ohne elektrische Verbindung. Und selbst, wenn die Kamera bei der Aufnahme auf einem Stativ stehen und sich nichts im Blickfeld bewegen würde, sollte schon durch die ständig sich öffnende und schließende Blende die Bewegungserkennung weiterhin reagieren und die Kamera nicht abschalten.

Web-Oberfläche

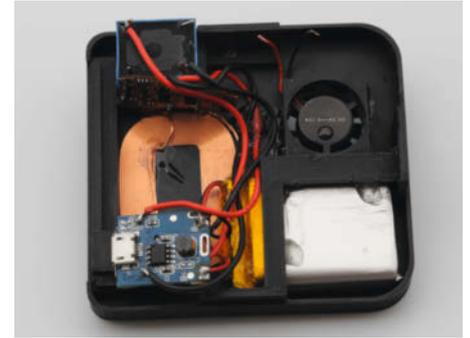
Mit der Wahl von *Motioneye* lösten sich auch einige andere Probleme: Es enthält eine Web-



11 An einer Seite des Reed-Kontakts wurde massiver Schaltdraht verwendet: Er hält den Kontakt sicher in Position.



8 Diese beiden Litzen sollten mindestens 0,4mm² Querschnitt haben.



9 Der Laderegler auf seiner Halterung

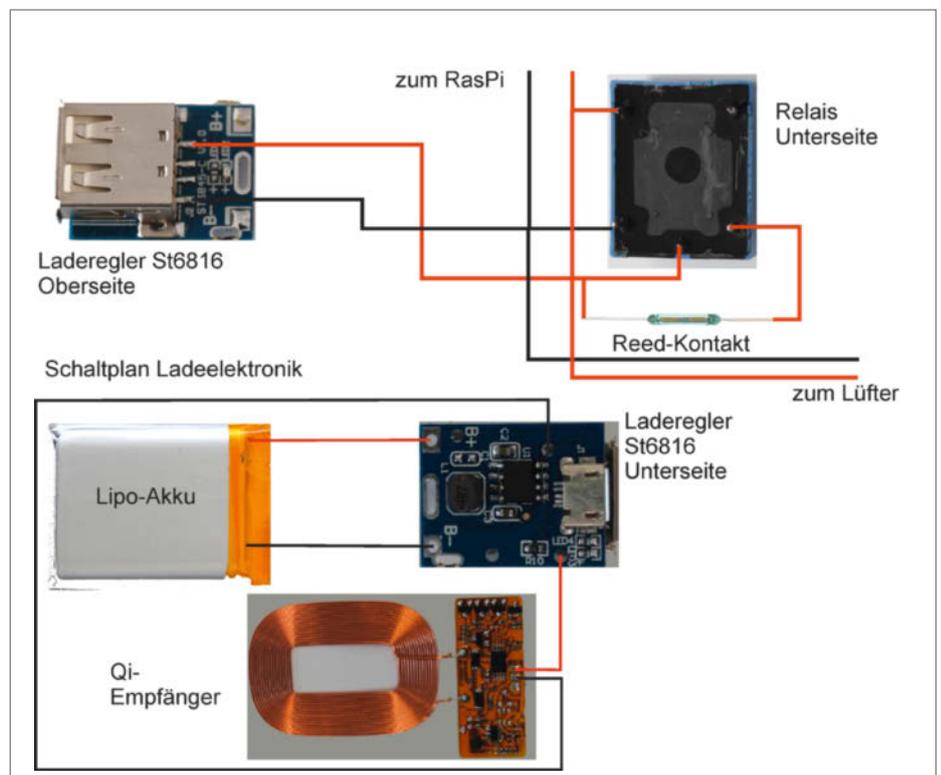
Oberfläche, auf der das Live-Video bild und alle bereits aufgenommenen Videos betrachtet werden können. Außerdem sind dort alle Einstellmöglichkeiten vorhanden. Und auch ein Samba-Server ist enthalten, der dann den Datenaustausch mit anderen Computern ermöglichen sollte.

Somit war nun die Wahl der Komponenten abgeschlossen und es ging an die Konstruktion. Die Anleitung im Folgenden zeigt Ihnen, wie alle Teile Platz in der engen Kassette finden und die Software konfiguriert werden muss, damit die *Make8*-Kassette sich mit einem WLAN verbindet, das dann auch per Smartphone den Zugriff auf die Filme erlaubt.

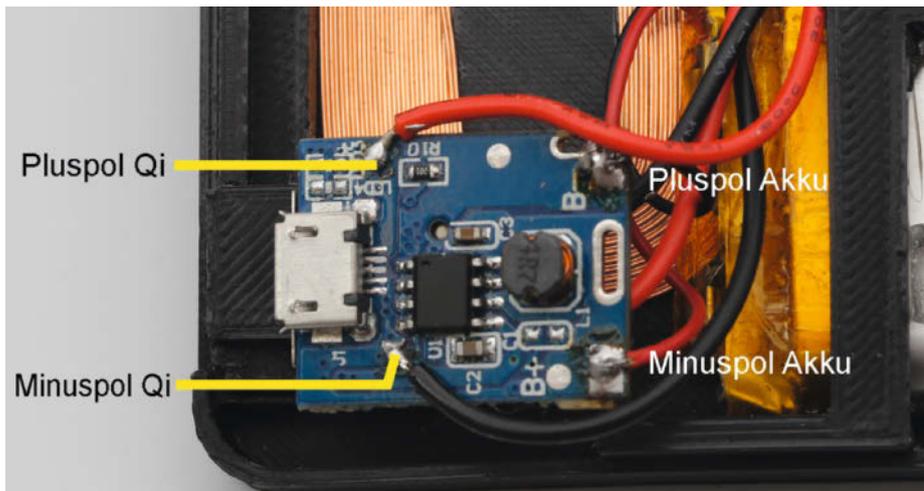
Druckvorschriften

Die Kassette besteht aus nur vier Teilen 1:

1. Unterteil: Hier sitzt der RasPi und das Kameramodul.
2. Deckel: Er enthält die gesamte Stromversorgung aus Qi-Empfänger, Laderegler, Akku, Reedkontakt, Relais und den Lüfter.
3. Sensorhalter: Dieses Teil hält den von seinem Objektiv befreiten Kamerasensor so, dass die lichtempfindliche Sensoroberfläche genau in der Bildebene der Kamera liegt.
4. Haltestück: Diese kleine Platte fixiert mithilfe zweier Schrauben und Muttern den Sensorhalter.



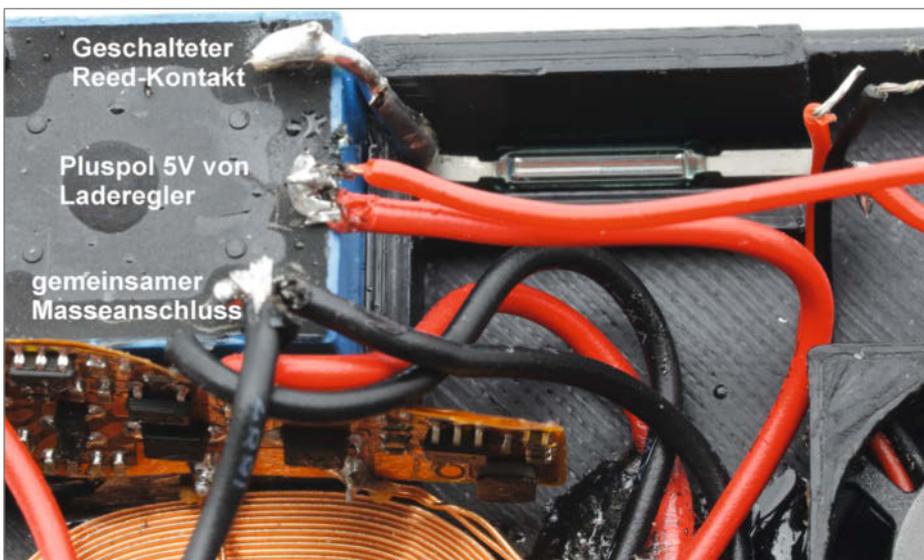
10 Der Schaltplan sieht zweigeteilt aus: Oben steht der Versorgungskreis für den Raspberry, unten die Ladeelektronik für den Akku. Die Verbindung ist der Laderegler, der in beiden Teilen vorkommt.



12 Die Drähte zum Akku und zum Qi-Empfänger am Laderegler

Gedruckt wurde in zwei Durchgängen: Unterteil, Deckel und Haltestück wurden aus PETG (PLA ist auch geeignet) mit 0,2mm Layerhöhe, Stützstrukturen und 100 Prozent Füllung gedruckt. Der massive Druck ist notwendig, weil

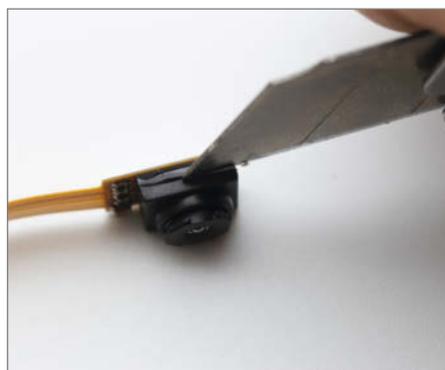
sich andernfalls die Kassette beim Herausnehmen aus der Kamera stark verwindet, was die Komponenten im Inneren übelnähmen. Alle Druckteile müssen aus schwarzem Filament gefertigt werden. Wegen des offenen Licht-



13 Die Verbindungen am Relais



14 Die Buchse für den Raspi: Parallel zu deren Anschlüsse sollten Sie auch die beiden Drähte des Lüfters anschließen.



15 Beim Entfernen des Objektivhalters kann ein Teppichmesser hilfreich sein.

gangs könnte es andernfalls zu farbigen Lichtstreuungen auf den Sensor kommen.

Der Sensorhalter sollte einzeln gedruckt werden: Layerhöhe möglichst fein (0,6mm) in der im Bild 2 gezeigten Anordnung. Die Auflagefläche, auf die später der Sensor geklebt wird, zeigt nach unten und wird so völlig glatt.

Die Oberfläche des Sensors muss auf einen halben Millimeter genau in der Bildebene der Kamera liegen, damit man im Einstellbereich des Kameraobjektivs scharfe Bilder erhält. Um Abweichungen bei Sensoren und Kameras ausgleichen zu können, gibt es vom Sensor sechs Varianten, die sich jeweils um 0,5mm im Abstand der Sensorauflagefläche vom Rand der Kassette unterscheiden. Hier hilft leider nur Probieren. Drücken Sie am besten alle sechs Halter und testen Sie später beim Zusammenbau, welcher für Ihre Kamera-/ Sensorkombination passt.

Ich hatte hier im ersten Entwurf einen verschiebbaren Halter vorgesehen. Leider war der so weit verschiebbar, dass bei einem meiner Versuche der vor dem Sensor befestigte Infrarotfilter den Lichtschacht der Kamera berührte. Das führte zum Zerschneiden des hauchdünnen Filters. Auf diese Weise kam ich ungewollt zu einem Infrarot-Sensor. Vielleicht baue ich (sobald es wieder Raspberrys zu vertretbaren Preisen gibt) mir noch eine zweite Kassette in Infrarot-Version für Nachtaufnahmen.

Deckelinhalt

Die meisten Elektronik-Teile müssen im Deckel untergebracht werden 3.

Ich habe den Deckel so konstruiert, dass er Halterungen für die meisten Teile enthält 4. Allerdings passen die Teile nur dann hinein, wenn eine bestimmte Reihenfolge eingehalten wird.

Als erstes werden Relais und Lüfter eingeklebt. Das habe ich mit schnell (in zwei Minuten) aushärtendem Zwei-Komponenten-Kleber gemacht. Achten Sie dabei auf die Lage der Relais-Kontakte. Der Lüfter muss die Luft von außen nach innen blasen 5.

Dann folgt der Qi-Empfänger. Dessen Spule muss so nah wie möglich am Relais liegen. Die Folienplatine steht senkrecht dazu und wird ans Relais geklebt 6.

Dann wird der Akku in seine Halterung geschoben und ebenfalls mit ein paar Tropfen Zwei-Komponenten-Kleber gesichert 7.

Nun muss der Laderegler vorbereitet werden. An der großen USB-Buchse löten Sie zwei Litzen an: Die rote für den Pluspol, die schwarze für den Minuspol der 5V-Ausgangsspannung 8.

Für den Laderegler gibt es ebenfalls eine Haltevorrichtung im Deckel, eine kleine Lasche, die in die USB-Buchse passt 9.

Nachdem der Regler so befestigt wurde, kann die Verkabelung in Angriff genommen

werden. Der Schaltplan 10 zeigt, wie alles verbunden wird.

Vor dem Einkleben des Reed-Kontakts sollten die Drähte angelötet werden. Vorsicht: Biegen Sie die Anschlüsse des Kontakts nicht unmittelbar an seinem Gehäuse. Es besteht aus hauchdünnem Glas und bricht schnell 11.

Akku und Qi-Empfänger werden mit Löt-punkten am Laderegler verbunden. Dabei unbedingt die Polarität beachten 12.

Der Reedkontakt wird mit dem Schaltdraht direkt am Relais angelötet. Auf der dreipoligen Seite des Relais sitzen außerdem der gemeinsame Massekontakt für die gesamte Schaltung (gleichzeitig ein Anschluss der Relais-Spule) und der geschaltete Ausgang des Reed-Kontakts (der zweite Spulenanschluss) 13.

Jetzt ist noch ein kurzes (!) Kabel mit Buchse vom geschalteten Pol des Relais und dem Masseanschluss notwendig. An die Buchse wird später der Raspi im anderen Teil der Kassette angeschlossen. Das Kabel sollte nur etwa 5cm lang sein, damit es später nichts in der geschlossenen Kassette blockiert 14.

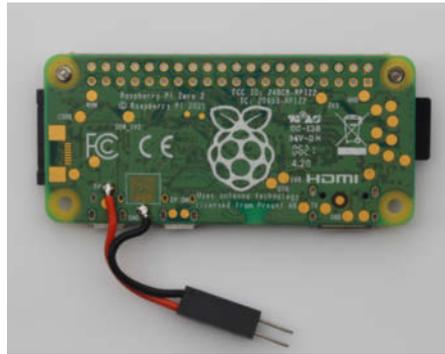
Nachdem alles verlötet ist, sollten die Kabel mit ein paar Tropfen 2K-Kleber am Deckel fixiert werden, damit sie später nicht in den Lichtgang geraten oder den Lüfter blockieren können.

Unterteil

Im Unterteil geht es aufgeräumter zu, denn dort sitzen nur der Raspi und der Kamera-sensor. Beide Teil müssen vor dem Einbau noch vorbereitet werden. Am Sensor entfernen Sie das Objektiv inklusive Halter. Das geht am besten mit einem Messer, das genau an der Klebefuge des Objektivhalters angesetzt wird 15. Mit vorsichtigem Druck kann man den Halter dann abhebeln. Vorsicht: Direkt dahinter liegt der aus dünnem Glas bestehende IR-Filter, der nicht zerbrechen sollte.

Der Raspi erhält ein kurzes Kabel für die Versorgungsspannung. Es wird an praktischerweise mit 5V und GND gekennzeichnete Löt-punkte an der Platinenunterseite angelötet 16. Da kann man also nicht viel falsch machen. Am anderen Ende des einige Zentimeter langen Kabels muss ein zur Buchse im Deckel passender Stecker sitzen.

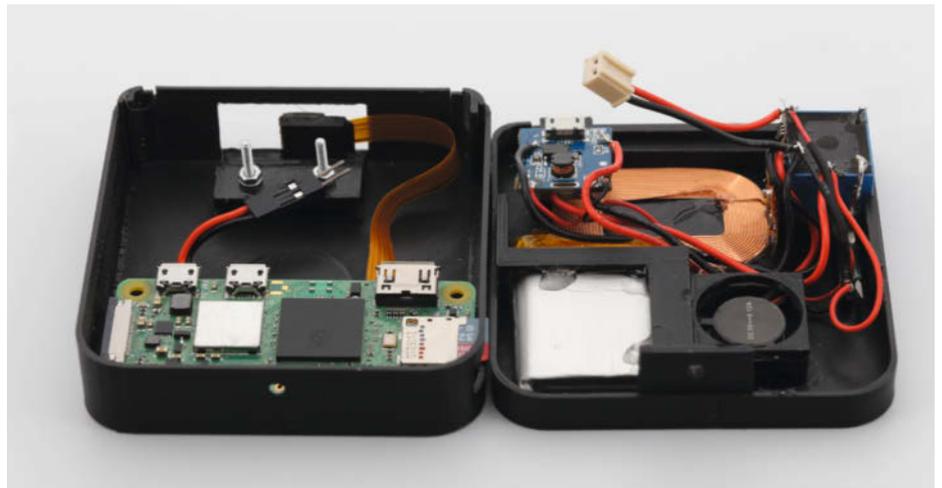
Haben Sie bereits alle Sensorhalter gedruckt? Sie sind mit Ziffern gekennzeichnet (0 bis 5). Bei meiner Kamera/Sensorkombination war der mit der Nummer 4 der richtige. Kleben Sie den Sensor auf die Vorderseite des Halters. An der Rückseite des Sensors befindet sich dafür ein doppelseitiges Klebeband. Die Verbindung damit ist ausreichend stabil für den Fokus-Test, lässt sich aber bei Bedarf wieder lösen. Erst, wenn der richtige Sensorhalter gefunden wurde, wird der Sensor mit 2K-Kleber endgültig (!) befestigt. Diesen Test können wir aber erst später nach Installation der Soft-



16 Die Spannungsversorgung des Raspi



17 Mehr Teile kommen nicht ins Kassettenteil.



18 Beide Kassettenteile sind nun voll bestückt.

ware vornehmen. Den Sensorhalter schrauben Sie dann mit zwei M2-Schrauben und dem Haltestück am Kassettens-Unterteil fest. Falls die Schraubenöffnungen im Sensorhalter etwas knapp sind, feilen Sie sie mit einer kleinen Rundfeile etwas nach. Verbinden Sie das Sensorkabel mit dem Raspi 17. Die Kamera-Elektronik sitzt auf deren Folienkabel und hat auf der Rückseite ebenfalls doppelseitiges Klebeband. Damit befestigen Sie sie an der Unterseite des Raspberry.

Der Raspberry Zero wird dann mit M3 x 4-Schrauben im Unterteil festgeschraubt. Zwei

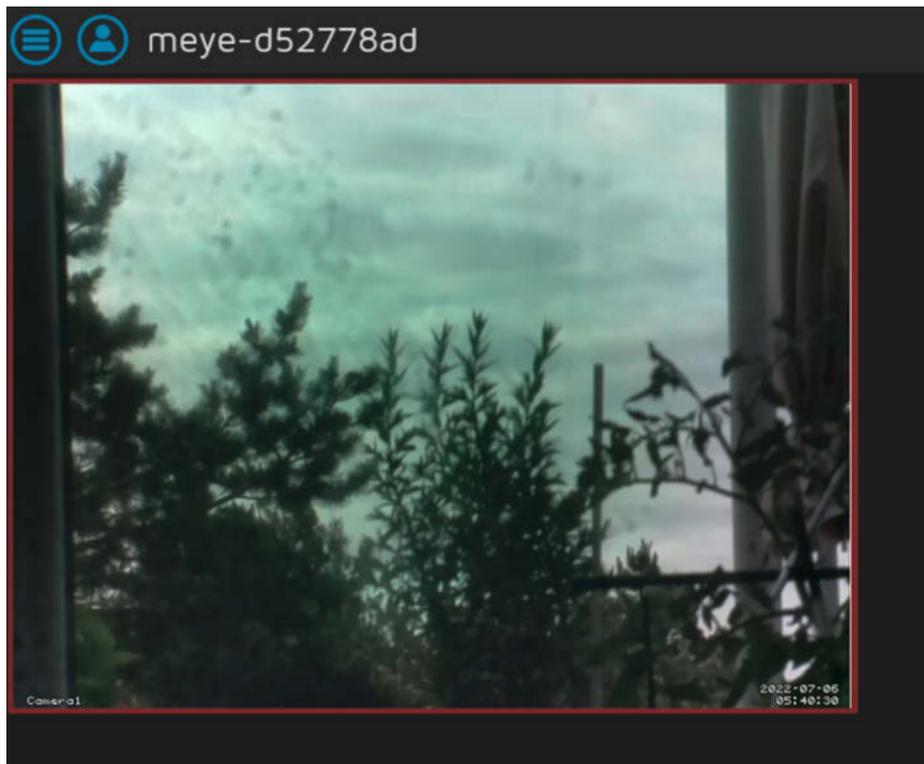
Schrauben reichen dafür schon aus, auch wenn vier Befestigungslöcher vorgesehen sind. Stecken Sie auch einmal die Speicherkarte in den Raspi: Sie darf nicht über die Kassette hinausragen, sonst wird sie beim Einlegen in die Kamera beschädigt 18.

Software-Installation

Bevor wir mit der Software-Installation beginnen: Legen Sie den Kassettendeckel jetzt auf ein Qi-Ladegerät. Es sollte mit seiner LED-Beleuchtung (Farbwechsel oder Blinken) nach



19 Notieren Sie sich den Namen und die IP-Adresse des Raspi.



20 Wenn es Ihnen gelingt, die Blätter scharf einzustellen, haben Sie den richtigen Sensorhalter.

spätestens einer Minute anzeigen, dass es mit dem Laden begonnen hat. Außerdem blinkt dann eine rote LED am Laderegler im Deckel. Falls nicht, haben Sie entweder einen Verdrahtungsfehler oder die Spule des Empfängers liegt nicht mittig auf dem Ladegerät.

Während der Ladezeit können Sie die Speicherkarte für den Raspi vorbereiten. Die Software *MotionEyeOS* erhalten Sie über den

Kurzinfo-Link. Legen Sie die Speicherkarte in den entsprechenden Slot Ihres Computers. Installiert wird die Software mit dem *Pi Imager*. Darin wählen Sie unter *Betriebssystem* den Punkt *Eigenes Image*. Dann klicken Sie sich bis zur zuvor über-spielten Image-Datei *motioneyeos-raspberrypi3.img-gz* durch. Dann stellen Sie die Speicherkarte ein und starten per Klick auf *Schreiben* die Programmierung der Speicherkarte.



22 Der Magnet sollte in etwa so positioniert werden.



21 Fertig! Sieht doch wie eine richtige Super8-Kassette aus.

Danach müssen Sie die Karte kurz aus dem Computer nehmen und gleich wieder einstecken, damit sie wieder beim System angemeldet wird. Jetzt gilt es, den WLAN-Zugang einzurichten. Dazu müssen Sie in der Boot-Partition der Speicherkarte eine Text-Datei mit dem Namen *wpa_supplicant.conf* anlegen. Die muss folgenden Inhalt bekommen:

```
country=DE
update_config=1
ctrl_interface=/var/run/wpa_supplicant
network={
scan_ssid=1
ssid="WLAN-Name"
psk="WLAN-Passwort"
}
```

Vorsicht, Filmführung

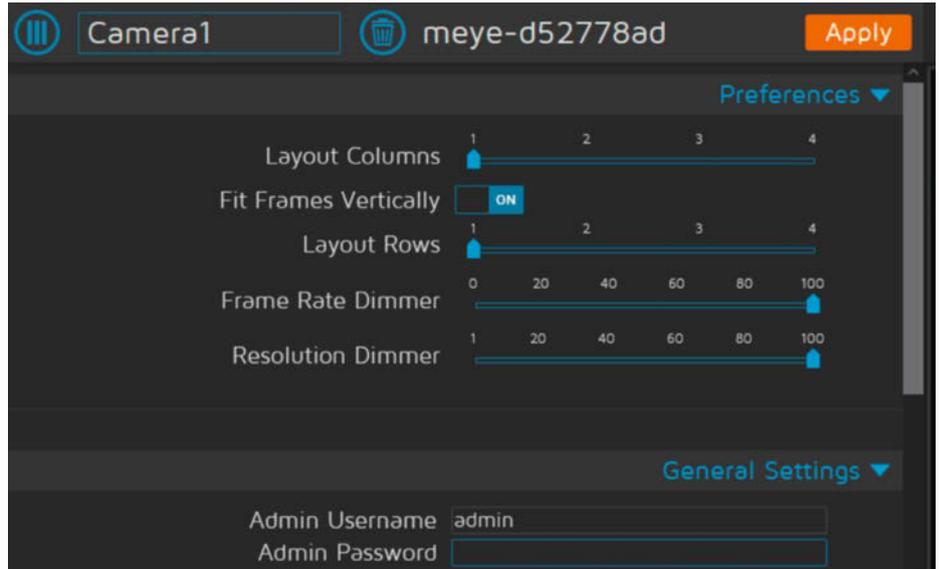
In den meisten Kameras ist eine hervorstehende Filmführung am Bildfenster angeschraubt. Die wird in vielen Fällen so weit in unsere Kassette hineinragen, dass sie den Kamerasensor beschädigen kann. Daher sollte diese Führung entfernt werden. Meist ist sie mit zwei oder drei kleinen Schrauben befestigt. Die Teile sollten Sie aufbewahren, falls Sie später doch noch einmal richtigen Film verwenden möchten.



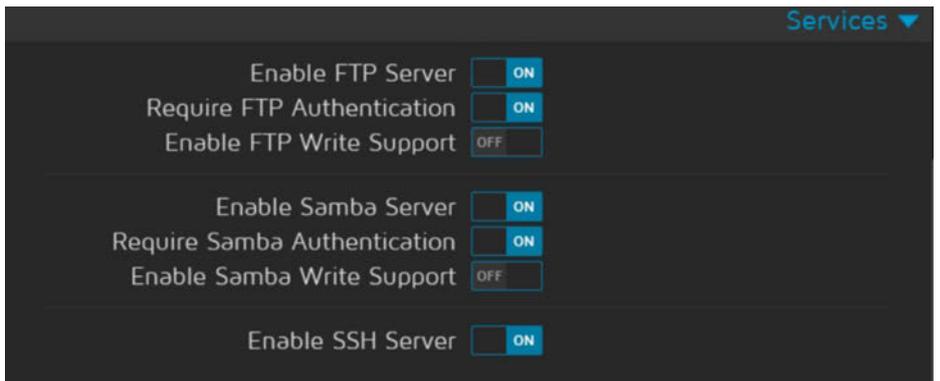
Für WLAN-Name und WLAN-Passwort müssen Sie die entsprechenden Werte einsetzen. Nach dem Speichern auf der Karte setzen Sie die Karte in den Raspi. Bevor Sie jetzt die Kassette zum ersten Mal in die Kamera einlegen, lesen Sie den Text im Kasten *Vorsicht, Filmführung*. Für den folgenden Test setzen Sie das Kassettenunterteil in die Kamera ein (in die Kamera müssen Batterien eingelegt sein). Den Raspberry versorgen Sie jetzt über ein passendes USB-Kabel mit Strom. Er bootet daraufhin. Nach etwa einer Minute sollte das abgeschlossen sein und der Raspi Kontakt mit dem WLAN haben. In der Web-Oberfläche Ihres Routers kontrollieren Sie, welchen Namen und welche IP-Adresse der Raspi erhalten hat. Sein Name beginnt mit *meye-* gefolgt von einer Buchstaben/Zahlenkombination. Stellen Sie im Router ein, dass der Raspi immer dieselbe IP-Adresse zugewiesen bekommt. Bei einer Fritzbox sieht das zum Beispiel so wie in Bild 19 aus.

Erster Test

Geben Sie im Browser Ihres Computers nun die IP-Adresse des Raspis ein. Daraufhin sollte das motionEye-Fenster mit dem Live-Kamerabild erscheinen. Stellen Sie das Objektiv der Kamera auf unendlich ein und halten Sie die Kamera auf ein mehr als 20m entferntes Objekt, zum Beispiel auf einen Baum im Garten. Erhalten Sie ein scharfes Bild, eventuell nach kleineren Korrekturen mit der Schärfereinstellung des Kamera-Objektivs? Falls Ja, dann haben Sie den richtigen Sensorhalter benutzt. Lassen Sie sich dabei nicht von der Bewegungsunschärfe täuschen: Eine Super8-Kamera besitzt keine Bildstabilisierung wie viele Smartphones. Daher verwackeln Sie mit Sicherheit das Bild. Ein Stativ oder die Ablage der Kamera auf einem Tisch kann da helfen.



23 Das Einstellungsmenü



24 Diese Einstellungen braucht man zum Zugriff von anderen Computern aus.

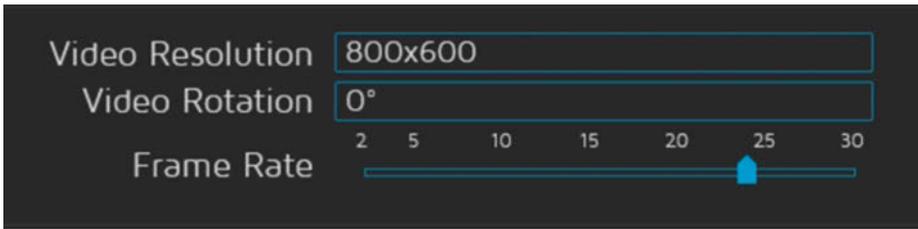
WAZER

Erster DESKTOP-WASSERSTRAHLSCHNEIDER der Welt.

ab 9.990,- € zzgl. MwSt.

Treffen Sie uns:
Maker Faire, Hannover
10.09. - 11.09.2022





25 Das ist in etwa DVD-Bildqualität.

Lässt sich die Schärfe nicht einstellen, müssen Sie einen anderen Sensorhalter ausprobieren. Das heißt, Sie müssen den Sensorhalter aus dem Kassetten-Unterteil ausbauen, vorsichtig den Sensor lösen und dann mit dem doppelseitigen Klebeband am nächsten Sensorhalter befestigen, wieder einbauen und erneut testen, bis das Bild in der Unendlichkeitseinstellung scharf wird 20.

Wundern Sie sich bei diesem Test nicht, wenn merkwürdige schwarze, unbewegliche Flecken auf dem Bild zu sehen sind: Das ist höchstwahrscheinlich ein Fingerabdruck von Ihnen auf dem IR-Filter. Das kann man aber später sehr gut mit einem weichen Mikrofaser-tuch beseitigen.

Herzlichen Glückwunsch, alles funktioniert. Sobald das Ladegerät nun die vollständige Ladung anzeigt (dauert etwa eine Stunde), können Sie die beiden Kassettenteile zusam-

menbauen: Dazu den Stecker des Raspberry in die Buchse des Deckels einstecken (Polung beachten) und den Deckel auf das Unterteil aufsetzen. An der Rückseite der Kassette befindet sich ein Loch, durch das man eine kurze M3-Schraube (6mm Länge) mit flachem Kopf eindrehen kann. Sie schafft sich selbst ein Gewinde im Unterteil und fixiert beide Teile aneinander. Allerdings könnte es sein, dass sie mit Teilen Ihrer Kamera kollidiert. In diesem Fall sollten Sie auf die Schraube verzichten und die Kassette verkleben 21.

Bleibt noch unser genialer Schalter. Kleben Sie den Magneten in den Deckel der Kamera. Er sollte bei geschlossenem Deckel über der Mitte der oberen Kassettenkante liegen. Ich habe ihn mit dickerem doppelseitigen Klebeband befestigt 22.

Wenn Sie nun den Deckel schließen, sollte sich die Kassette einschalten. Das kann man

akustisch kontrollieren: Der Lüfter läuft an (er saugt Luft aus dem Kassettenfach ins Innere der Kassette, die dann über die Öffnung am Kartenslot wieder nach außen entweicht) und äußert sich durch leises Rauschen.

Einstellungen

Nun müssen Sie noch ein paar Einstellungen in motionEye vornehmen: Das Menü erreichen Sie per Klick auf den Kreis mit den drei waagerechten Strichen. Als Erstes sollten Sie das Kamerabild vergrößern: Standardmäßig ist motionEye so eingestellt, dass es drei Kamerabilder nebeneinander darstellt. Stellen Sie die *Layout Columns* auf 1. Außerdem sollten Sie ein Admin-Passwort eingeben. Achtung: Die meisten geänderten Einstellungen werden erst nach einem Klick auf *Apply* übernommen 23! Eventuell ist dann ein Reboot notwendig und Sie müssen sich mit dem Benutzernamen *admin* und dem zuvor gewählten Passwort anmelden.

Damit Sie später Videos auch auf einen anderen Computer überspielen können, aktivieren Sie im *Services*-Menü den *Samba-Server* und *SSH* 24.

Unter Video Device wählen Sie eine Auflösung von 800 x 600. Grund dafür: Bei höherer Auflösung würde der Arbeitsspeicher des Raspi (512MB) rasch volllaufen, da die Bilder nicht schnell genug auf die Speicherkarte geschrieben werden können. Diese Bildqualität ist aber immer noch mehr als DVD-Videoqualität. Stellen Sie in diesem Bereich auch die *Frame Rate* auf den Wert ein, den Ihre Kamera benutzt. Standard bei Super8 waren 18 Bilder/s. Teurere Kameras konnten auch 24 Bilder/s 25. Stimmen die Bildraten von Kamera und motionEye nicht überein, flimmert das Videobild. Das kann übrigens auch ganz reizvoll sein, erinnert es doch an Stummfilmzeiten.

Jetzt muss noch der Auslösemechanismus eingestellt werden: Erinnern Sie sich? Das soll mit der Bewegungserkennung (*motion detection*) geschehen. So wie in Bild 26 sollte es aussehen, damit es funktioniert.

Vergessen Sie nicht den Klick auf *Apply*! Danach kann es losgehen mit dem Super8-Filmen. Betrachten können Sie die Videos nach einem Doppelklick auf das Live-Videobild und das dann erscheinende Wiedergabe-Symbol.

In der nächsten Make zeige ich dann, wie Sie motionEye so einrichten, dass die Kassette als WLAN-Accesspoint arbeitet. So können Sie auch unterwegs immer mit dem Smartphone auf die Kassette zugreifen. Bis dahin können Sie ja schon einmal sämtliche Familienmitglieder und Haustiere filmen. Online präsentiere Ihnen ich den ersten Super8-Rundgang durch den Heise-Verlag. Vielleicht können Sie uns ja auch das ein oder andere kurze Katzenvideo oder Ähnliches zusenden. Viel Spaß beim Filmen! —hgb



26 So startet die Videoaufnahme beim Drücken des Auslösers.

Mac & i Wissen erfahren

Apple School Manager

Tablets sind aus der modernen Schule nicht mehr wegzudenken. Apples iPads sind robust, wartungsarm, intuitiv und iOS wird für Bildung optimiert.

10%
Frühbucher-
rabatt sichern!

WEBINAR

Apple School Manager gratis beantragen und registrieren

Sie möchten Ihre Schule oder Klasse mit iPads ausstatten? Apples **kostenlose Software** Apple School Manager hilft Ihnen dabei. Unser Live-Webinar zeigt, wie Sie diesen beantragen und registrieren. Wir geben Tipps, die Sie beachten sollten, damit die Einrichtung der Software von Grund auf gelingt.

05.10.2022 | 29.11. 2022
mac-and-i.de/webinar-1



WORKSHOP

Apple School Manager beherrschen

Nach dem Download des Apple School Manager zeigen wir Ihnen, wie Sie diesen konfigurieren, nutzen und mit einem Mobile Device Management System verknüpfen. Sie lernen, wie Sie die Struktur Ihrer Bildungsorganisation im School Manager abbilden. Außerdem beschäftigen wir uns mit der **Verwaltung von Accounts** und Rollen und deren Aufgaben.



12. – 13.10.2022 | 07. – 08.12.2022
mac-and-i.de/workshop-1

Für den optimalen Einstieg in den Apple School Manager empfehlen wir die Teilnahme an Webinar und Workshop.

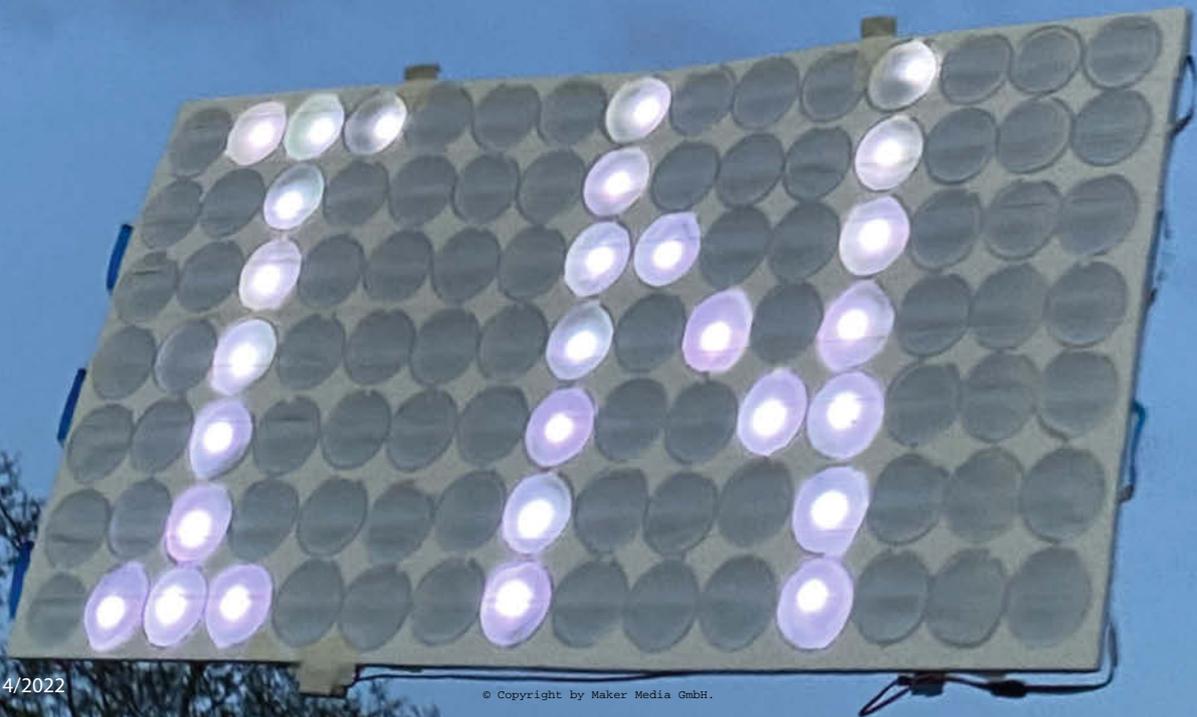
**JETZT
TICKETS SICHERN**



Airsign

Botschaften an den Himmel schreiben: Das geht mit Airsign. Die Textanzeige fliegt mit Drohne oder Heliumballons und wird vom Boden aus per WLAN über ein Webinterface gesteuert.

von Dr. Armin Zink



Für ein Geburtstagskind wollte ich bunte Glückwünsche gut lesbar am Himmel überbringen. Aus 35 Muffin-Cups und einigen WS2812-LEDs habe ich daher fliegende Buchstaben gebaut – eine Tafel mit Leuchtdioden, die Texte beliebiger Länge anzeigt. So übermittelt man Herzensbotschaften, die lange im Gedächtnis bleiben! Damit das *Airsign* fliegen kann, muss es sehr leicht konstruiert sein, denn Gasballons oder eine Drohne können nur wenige Gramm heben. Es ist also notwendig, auf das Gewicht der Komponenten zu achten.

Das *Airsign* gibt es in zwei Varianten: Der einzelne Buchstabe leuchtet in den Abmessungen 50 × 70 Zentimetern groß auf einem Holzgestell und ist aus 50 bis 100 Metern Entfernung sehr gut sicht- und lesbar. Außerdem habe ich eine größere Variante als Laufschrift entworfen, die mehr Platz benötigt. Bis zu drei Buchstaben kommen auf einer 100 × 50 Quadratzentimeter großen Styroporplatte unter. Hier ist das Gewicht größer und die Pixel sind kleiner. In beiden Modellen verwende ich den Mikrocontroller *Wemos D1 Mini Pro*, dessen WLAN eine Reichweite von circa 100 Metern hat. Der kleine Akku einer alten Drohne dient als Stromversorgung. Damit ist ein Betrieb von zehn Minuten problemlos möglich.

Außer Glückwünschen kann das *Airsign* natürlich beliebige andere Nachrichten anzeigen und zum Beispiel als Plakat bei einer Demonstration dienen. Über den *Wemos* wird das *Airsign* aus der Ferne gesteuert und der Text kann während des Betriebes vom Boden aus verändert werden. Wer mag, klebt die leichte Konstruktion an Fenstern oder Balkonen an, stellt sie auf Vordächern ab oder hängt sie an Kirchtürmen auf. Um Nachrichten in anderen Sprachen anzuzeigen, habe ich einen zweiten Zeichensatz mit kyrillischen Buchstaben eingebaut. Ebenso gibt es Steuerzeichen für farbige Texte und Flaggen.

Technik

Das *Airsign* ist im 8-Bit Retro-Style gehalten, mit einer Matrix von fünf Spalten und sieben Zeilen, also 35 Punkten. Jeder Punkt (oder Pixel) besteht aus einer WS2812-LED, die einen Muffin-Cup beleuchtet. Die Papierförmchen dienen als Diffusoren und sind aus großer Entfernung viel deutlicher erkennbar als eine einzelne LED. Die Ränder der Muffin Cups verhindern außerdem, dass das Licht andere Punkte trifft. So sind die Punkte scharf gegeneinander abgegrenzt. Überraschenderweise ist die Helligkeit der LEDs kein Problem: Eine LED pro Cup reicht aus, um die Anzeige am dunklen Himmel zu erkennen. Bei Sonnenschein sieht man natürlich nichts.

Ich habe das Board *Wemos D1 Mini Pro* zur Steuerung verwendet, da es sehr klein und leicht ist und die kostenlose Arduino-Program-

Kurzinfo

- » ESP-Funksteuerung von WS2812-LEDs
- » Buchstaben und Laufschriften zeichnen
- » Superleichter Aufbau mit Holzbrett und Styropor

Checkliste

-  **Zeitaufwand:**
ein Wochenende
-  **Kosten:**
80 Euro
-  **Löten:**
einfache Lötarbeiten
-  **Elektronik:**
Grundkenntnisse
-  **Programmieren:**
Arduino-Grundkenntnisse
-  **Holzbearbeitung:**
Holz und Styropor zuschneiden

Styropor-Version

- » Styroporbrett 1000 × 500 × 20mm

Werkzeug

- » Elektronische Küchenwaage
- » Lötkolben
- » Draht
- » Skalpell
- » Akkubohrer

Mehr zum Thema

- » Peter Tschulik, LED-Laufschrift mit ESP32, Make 4/21, S. 8
- » Luca Zimmermann und Helga Hansen, ESP-Boards mit der Arduino-IDE programmieren, Online

Material

- » *Wemos D1 Mini Pro*
- » *S1 Lipo Micro-USB-Lader* mit 5-Volt-Ausgang
- » *1600mAh Lipo-Akku* oder ähnlich
- » *WS2812-LED Streifen* 30LEDs/m, indoor
- » *Kreppband* 50mm oder breiter
- » *Plastikschnüre*
- » *Drohne* DJI Mini 2 oder vergleichbares Modell
- » *große Luftballons* und *300 Liter Helium* (alternativ)

Holz-Version

- » *35 Muffin Cups*
- » *1m Holzleiste* 3 × 3mm
- » *Klingeldraht*
- » *Tesa-Band*





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



mierumgebung ihn gut unterstützt. Es gibt verschiedene *D1-Mini*-Versionen: Die Rückseite unseres Modells *Mini Pro* enthält keine Bauteile und kann deshalb mit doppelseitigem Klebeband montiert werden. Der *Mini Pro* erzeugt einen eigenen Access Point, startet einen *Mini-Webserver* und nimmt den gewünschten Text auf einer Webseite entgegen. Nach Eingabe des Textes erledigt er auch gleich die Anzeige. Die Buchstaben werden nacheinander mit variabler Geschwindigkeit und Farbe auf den 5 × 7 Punkten angezeigt.

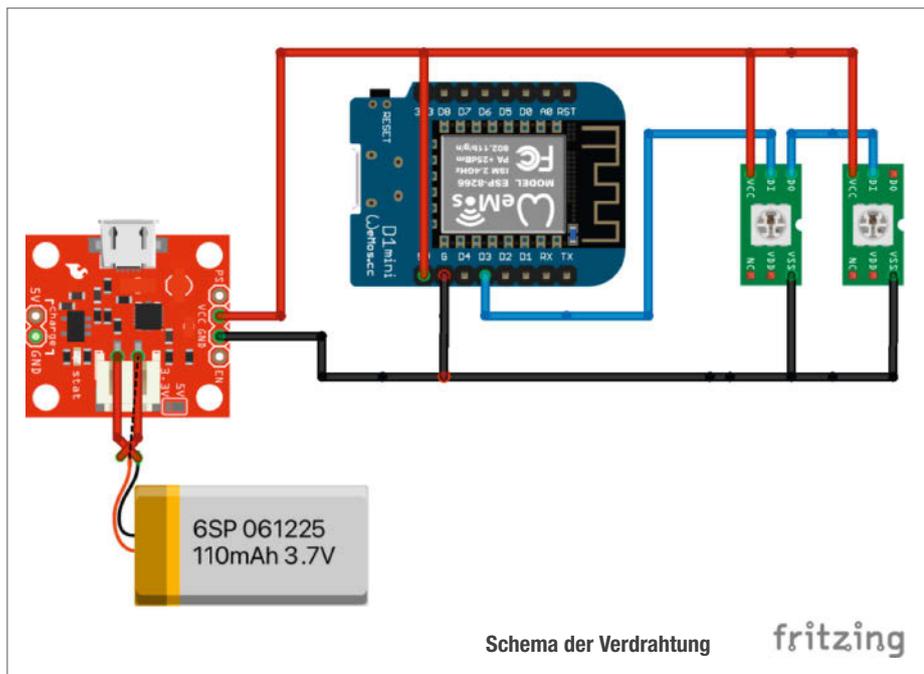
Elektrik

Der Schaltplan des Projektes ist sehr übersichtlich. Der Akku ist mit dem Batterie-Eingang

des Lipo-Lademoduls verbunden. Je nach Akku und Modul kann das per Stecker oder Lötverbindung geschehen. Der 5-Volt-Ausgang des LiPo-Laders speist die LEDs und den *Wemos-D1*. Als einzige Steuerleitung verbinde ich den Pin *D3* mit *DI* (Data-In) der ersten LED. Hier lötet man den Verbindungsstecker an, der üblicherweise mit den LEDs geliefert wird. Jeder *DO* (Data-Out) einer LED wird zum nächsten *DI* weitergeleitet.

Holz-Konstruktion

Das erste *Airsign* habe ich aus Holzleisten mit Querschnitt 3 × 3mm gebaut. Ich habe sie im Baumarkt auf der *SB-Schrauben-Waage* gewogen, um die leichtesten Leisten zu finden.



Ich verwende sie für die obere und untere Querstange. An den Enden bohre ich zwei Löcher für die Befestigung und dann fünf Löcher für die vertikale Verdrahtung im Abstand von zehn Zentimetern. Vor dem ersten Loch sind fünf Zentimeter Abstand vom Rand nötig, denn an dem überstehenden Teil wird später das Klebeband für die Muffin-Cups angebracht. Zuerst habe ich außerdem breitere Stäbe (5 × 2mm) als vertikale Leisten für die LEDs verbaut, damit war das Airsign im Test aber zu schwer.

LED-Verdrahtung

Die LEDs stammen aus einem WS2812-Selbstklebe-Streifen. Ich habe sie jeweils an der gestrichelten Linie an den Lötmarkierungen abgeschnitten und mit 10 Zentimeter langen Kabelstücken verbunden. Da das Gewicht zur Stabilisierung des Flugs am besten weit unten hängt, habe ich Controller und Akku unten am Holzrahmen montiert. Dementsprechend fängt man links unten mit der Verbindung zu den LEDs an und verlegt die Leitungen in einer Schlange über die Leisten.

Beim Zusammenbau ist auf die korrekte Orientierung von *Data-Out*, *Data-In* und der Versorgungsspannung achten. Kaputte LEDs später zu wechseln ist sehr mühsam. Vor dem Anschluss also bitte nochmal prüfen! Anschließend erfolgt der Anschluss an den ESP und ein erster Test. Ich habe den üblichen 3-Pol-Stecker am Controller und den LEDs verwendet, um die Controller wechseln zu können und Tests zu erleichtern – auch wenn man ohne Stecker noch etwas Gewicht sparen könnte. Als Test kann man ein beliebiges WS2812-Demoprogramm verwenden, zum Beispiel den *Strandtest.ino* aus der Adafruit-Bibliothek für Neopixel. Hinweise zur Programmierung folgen später im Artikel.

Montage der Muffin Cups

Nach dem Test kann man die Muffin Cups aufkleben. Zuerst befestigt man sechs Klebestreifen aus transparentem Packband zwischen den Querlatten – einmal komplett von oben nach unten. Ich habe es nur einseitig geklebt, um Gewicht zu sparen. Die Muffin Cups versieht man mit zwei Streifen Kreppklebeband, das man leicht entfernen und neu anbringen kann. Ich klebe die Cups mittig über den LEDs auf den Klebestreifen. So treffen sich die Klebestreifen nicht. Das Kreppband kann man leicht umkleben, bis alles gleichmäßig aussieht. Die Böden der Cups sollten einen Abstand von drei Zentimetern von den LEDs haben, sodass sie ganz beleuchtet werden können.

Stromversorgung

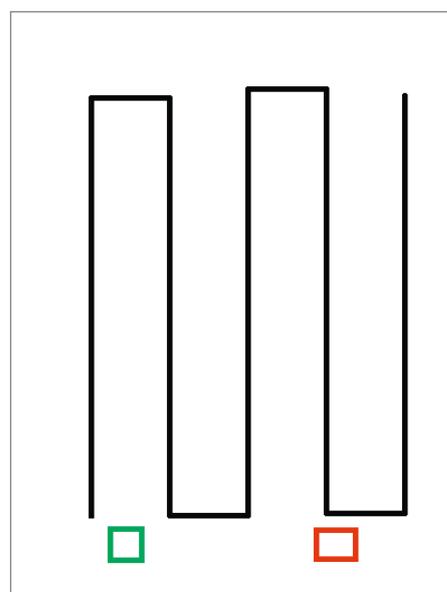
Für die Spannungsversorgung im Betrieb habe ich zuerst eine kleine Powerbank aus-



Von links nach rechts: Controller Wemos D1, Lader und Akku. Hier sieht man den Stecker für die LEDs und einen kleinen Stecker für die Stromversorgung, den ich aus Stifteleisten hergestellt habe.



Die Leisten habe ich an den Enden durchbohrt und mit Klingeldraht fixiert, um ein Gitter für die 35 LEDs zu bilden.



Schematische Anordnung der LEDs auf dem Gitter: Die Position des ESP-Controllers ist in grün, die Position des Akkus in rot eingezeichnet.

probiert. Diese war mit 61 Gramm allerdings viel zu schwer. Deswegen nutze ich einen alten Drohnen-Akku. Die Leistung reicht für die 35 LEDs und einige Minuten Betrieb aus. Zusätzlich ist ein Akku-Ladegerät notwendig. Ich habe ein Modell mit Micro-USB-Stecker, das auch einen 5-Volt-Ausgang für den Betrieb der LEDs und des Wemos D1 Mini hat. Es gibt allerdings Lipo-Lader, die keinen 5V-Ausgang haben. Hier muss man beim Bestellen etwas aufpassen.

Software

Als Basis für die Software habe ich das Arduino-Beispiel *ESP8266 Web Server* verwendet. Für die Steuerung der LEDs kommt die Bibliothek *Adafruit Neopixel* zum Einsatz. Anleitungen zum Einbinden des Wemos D1 Mini Pro (ESP 8266) und der Bibliotheken in die Arduino IDE finden Sie über den Downloadlink in der Kurzinfo, ebenso meinen Sketch. Das Beispiel habe ich um einen Zeichensatz (Font) erweitert, sodass es Buchstaben in verschiedenen Farben sowie Flaggen anzeigen kann. Der Rest ist kurz gehalten. Man kann es also leicht erweitern. Es beginnt mit Name und Passwort für den WLAN Access Point:

```
#define SSID "MUFFIN"
#define PASSWORD "12345678"
#define DELAYVAL 500
```

Anschließend werden in `void setup()` mit `pixels.begin()` die Neopixel initialisiert und der Access Point gestartet: `while (!WiFi.softAP(SSID, PASSWORD, 6, false, 15))`. Auch die Hauptschleife `void loop()` ist nur zwei Zeilen lang.

```
yield();
ledbuffer_
printString(textMessage,col);
```

Die Zeichensätze

Den größten Teil des Programms nehmen die Fonts in Anspruch: ein klassischer 5 x 7-ASCII-Font und ein kyrillischer Font. Am besten lesbar sind die Großbuchstaben.

Um die Zeichen anzuzeigen, muss man ihre einzelnen Punkte den richtigen LEDs zuordnen. Der Einfachheit halber sind in unserem Zeichensatz alle Zeichen gleich breit und bestehen aus 5 Spalten mit je 7 Pixeln. Alternativ kann man Zeichensätze mit variablem Abstand verwenden, bei denen etwa ein *i* nicht so breit abgebildet wird wie ein *W*.

Für die Anzeige werden alle Zeichen aus einem String einzeln gelesen und die passenden Einträge im Font gesucht. Dies geschieht über den Befehl `pgm_read_byte()`, mit dem ein Byte aus dem Programmspeicher gelesen wird. Abhängig vom ASCII-Code des Zeichens werden nacheinander fünf Bytes gelesen. Jedes dieser Bytes enthält eine 1 in der Zeile,



Hochgeklappter Muffin-Cup, darunter die LED-Verdrahtung – hier noch auf Holzleisten, wdie sich aber als zu schwer erwiesen.

in der ein Pixel leuchten soll. Die 0 besteht aus fünf Spalten, die ich hier der Anschaulichkeit halber um 90 Grad gedreht habe. Die beleuchteten Pixel sind als 1, die unbeleuchteten als 0 gezeichnet. Das höchstwertige Bit 8 ist immer 0 und wird nicht verwendet, da das Zeichen zur Darstellung nur 7 Pixel benötigt.

```
00111110
01010001
01001001
01000101
00111110
```

Die Routine `ledbuffer_printChar()` zeichnet einen Buchstaben.



ledbuffer_printChar()

```
void ledbuffer_printChar(char ch, t_color ccol) {
    uint8_t d;
    for (uint8_t x = 0; x < 5; x++) {
        if (cyrillic)
            d = pgm_read_byte(6 + (ch - 32) * 5 + x + Cyrillic5x7);
        else
            d = pgm_read_byte(6 + (ch - 32) * 5 + x + System5x7);

        if ((d & 64) == 64)
            ledbuffer_setPixel(x, 0, ccol);
        else ledbuffer_setPixel(x, 0, CBLACK);
        if ((d & 32) == 32)
            ledbuffer_setPixel(x, 1, ccol);
        else ledbuffer_setPixel(x, 1, CBLACK);
        if ((d & 16) == 16)
            ledbuffer_setPixel(x, 2, ccol);
        else ledbuffer_setPixel(x, 2, CBLACK);
        if ((d & 8) == 8)
            ledbuffer_setPixel(x, 3, ccol);
        else ledbuffer_setPixel(x, 3, CBLACK);
        if ((d & 4) == 4)
            ledbuffer_setPixel(x, 4, ccol);
        else ledbuffer_setPixel(x, 4, CBLACK);
        if ((d & 2) == 2)
            ledbuffer_setPixel(x, 5, ccol);
        else ledbuffer_setPixel(x, 5, CBLACK);
        if ((d & 1) == 1)
            ledbuffer_setPixel(x, 6, ccol);
        else ledbuffer_setPixel(x, 6, CBLACK);
    }
    ledbuffer_update(true);
    delay(speed);
}
```

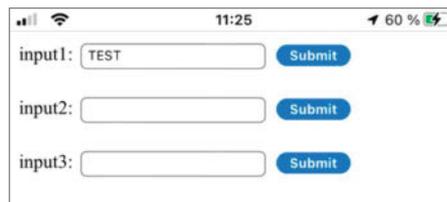
Bedienung

Der Wemos D1 Mini startet den Access Point *MUFFIN*, mit dem man sich nach Eingabe des Passwortes (hier 12345678) verbindet. Danach öffnet man in einem Webbrowser die Adresse 192.168.4.1 für das Webinterface zur Steuerung. Dort gibt man in der Zeile 1 den Text ein. Werden nur Leerzeichen eingegeben, bleibt das Display aus. Als Steuerzeichen für die Farben dienen <R> für Rot, <G> für Grün, für blau und <W> für Weiß. In der zweiten Zeile stellt man die Wartezeit zwischen den Buchstaben ein. Verwenden Sie Werte zwischen 300 für schnell und 700 für lange Zeiten zwi-

schen den Buchstaben. 500 ist der Standard. Für Text in der Zeile 3 wird der kyrillische Font genutzt.

Ballon-Aufstieg

Um das Airsign fliegen zu lassen, habe ich eine Drohne und mehrere mit Helium gefüllte Ballons ausprobiert. Laut Wikipedia benötigt man 1000 Liter Helium, um 1 Kilogramm in die Luft zu heben. Für die rund 200 Gramm des Airsigns sind also 200 Liter an Ballonvolumen nötig. Ich habe zunächst einen Standardballon mit Ballongas gefüllt. Er konnte fünf Gramm tragen, aber nur sehr wenige Stunden. Für 200 Gramm



Das Webinterface zur Airsign-Steuerung

bräuchte man somit circa 40 Ballons. Daher habe ich größere Luftballons mit 60 bis 80 Zentimeter Durchmesser genommen und mit den 300 Litern aus einer Flasche gefüllt. Die Ballons habe ich an einer weiteren Holzleiste angeknötet, die an der Oberseite des Airsigns angebracht wird.

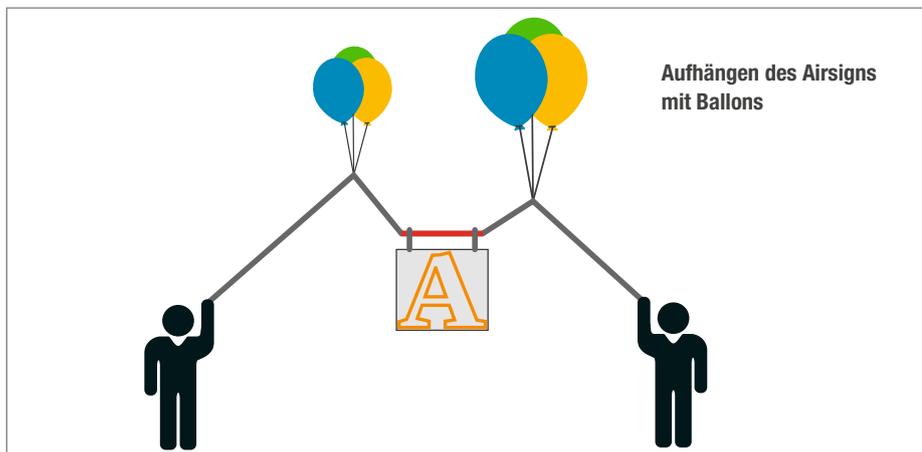
Drohne

Ich habe es außerdem mit der DJI Mini2 probiert, die zusätzlich nochmal annähernd ihr Eigengewicht von 250 Gramm tragen kann. Dazu habe ich das Airsign mit vier Nylon-schnüren an der Drohne befestigt. Mein erster Prototyp mit einer Powerbank und vielen Holzleisten wog rund 250 Gramm. Die Drohne hob zwar ab, aber sie wurde sehr warm und der Akku leerte sich sehr schnell. Mit Akkupack und nur den nötigsten Leisten reduzierte sich das Gewicht auf 140 Gramm, womit die Drohne gut klarkam.

Der Scroller

Einige Freunde wünschten sich nun noch eine Laufschrift mit mehr Zeichen. Damit die Pixel nicht zu klein werden und lesbar bleiben, musste das Airsign deutlich größer werden. Für den Prototyp der Laufschrift-Version habe ich ein 30mm dickes Styroporbrett verwendet. Es ist viel schwerer als die Holzleisten, aber robuster. Als Diffusor nutze ich Kreppband mit 50 Millimetern Breite. Außerdem habe ich fertige LED-Streifen genommen, die ein klein wenig mehr wiegen, aber einfacher und zuverlässiger zu nutzen sind. Ein 70 Zentimeter langer Streifen an WS2812-LEDs mit 30 LEDs pro Meter wiegt 8 Gramm, während meine Spalte aus 7 LEDs und 70 Zentimetern Kabel nur 6 bis 7 Gramm wiegt.

Die Pixel haben einen Abstand von 6,6 Zentimetern. Damit ergeben sich für ein Styroporbrett mit den Abmessungen 1 Meter x 50 Zentimeter dann 15 x 7 (105) Pixel mit je sechs Zentimetern Durchmesser. Das reicht für einen Laufschrift von fast drei Zeichen und ist sehr gut lesbar. So kann man allerdings nur jede zweite LED verwenden. Das habe ich in der Software berücksichtigt. Beim Verlegen als Schlange verliert man ebenfalls vier LEDs pro Zeile. In der Software gibt es eine entsprechende Zuordnung, die man leicht auf sein eigenes Design anpassen kann.



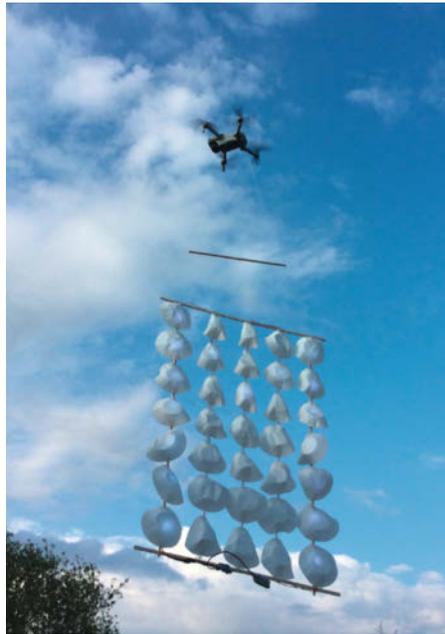
Kleiner Pico – großer Spaß!



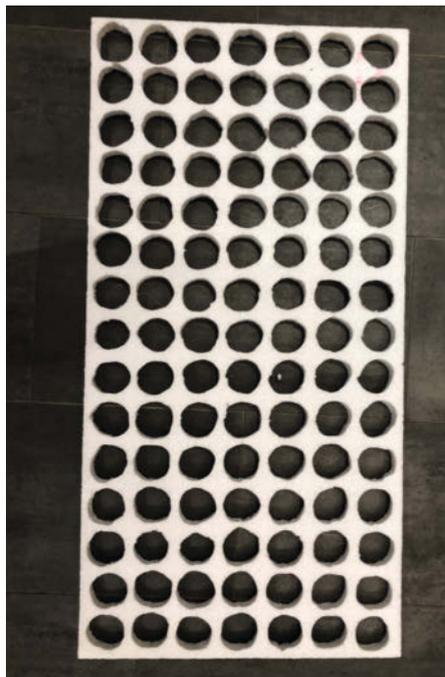
Befestigung der Schnüre an der Drohne

Die Löcher im Brett habe ich ausgerechnet, angezeichnet und mit einem Skalpell geschnitten. Nach dem Ausschneiden wiegt das Styropor nur sensationelle 38 Gramm, nach dem Aufkleben des Kreppbandes leider schon wieder 80 Gramm. Durch zusätzliche Löcher könnte man hier noch Gewicht sparen. Nach Montage der LEDs, des Controllers und des Akkus brachte die Konstruktion trotzdem nur 160 Gramm auf die Waage. Das reichte für einen erfolgreichen Testflug mit der Mini2.

Mein neues Airsign ist so deutlich grösser und etwas schwerer als das 5 x 7-Airsign und die Pixel sind kleiner, sodass sich die maximale Lese-Entfernung verringert. Dafür ist die Ablesbarkeit als Lauftext besser. Letztendlich ist die Lösung auch sehr universell, da man mit der Schlange aus LEDs auch leicht andere Größen erstellen könnte. Wie bei allen Projekten war die Lernkurve am Anfang hier sehr steil und laufend hatte ich neue Einfälle, um das Projekt zu verbessern. Sicherlich haben Sie beim Nachbauen auch einige gute Ideen – schicken Sie die gern an mail@make-magazin.de, ebenso Bilder eigener Airsigns. —hch



DJI Mini2 mit dem 140 Gramm schweren Airsign – das geht.



Das vorbereitete Styroporbrett mit Löchern



Das fertige Airsign in 100 x 50cm Grösse: Mit dem Styropor-Rahmen ist es stabiler als das leichte Holzgestell. Es eignet sich auch prima als Raumdekoration.



Inkl. Raspberry Pi Pico RP2040

Exklusiv im heise Shop

Mit dem **Make Special Pi Pico** steigen Sie ein in die Welt der Programmierung von ARM-Mikrocontrollern. Make zeigt in dem 64-seitigen Special, welche Entwicklungsumgebungen es für den Rasperry Pi Pico gibt, wie man sie installiert und wie man sie nutzt:

- ▶ Den Pi Pico programmieren mit Basic, MicroPython oder C++
- ▶ Sensoren und Mini-Displays nutzen
- ▶ NeoPixel per PIO-Funktion kontrollieren
- ▶ **inkl. Raspberry Pi Pico, Modell RP2040 Mikrocontroller-Board**

Heft + Raspberry Pi Pico für 24,95 €



shop.heise.de/make-pico

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

Das Dartscheiben-flutlicht

Dart in der Nacht und in dunklen Kellern wird mit dieser Dartscheiben-Beleuchtung möglich. Auf verschiedene Boards anpassbar und demontierbar dank 3D-Druck.

von Alexander Moser



Als Corona im Sommer 2021 größere Treffen in geschlossenen Räumen stark einschränkte, zog es viele in Ihrer Freizeit ins Freie. Auch einige Dartspieler hatten ihre Spiel-treffen in öffentliche Parks verlegt und einfach eine Dartscheibe an einen Baum gehangen. Wenn dann aber die Sonne unterging, musste das Spiel beendet werden, da die Scheibe nun nicht mehr zu erkennen war. Um auch nach Sonnenuntergang weiterspielen zu können, wäre also eine Art Flutlicht für die Beleuchtung des Boards praktisch. Wie in einem Stadion sollen die Pfeile gleichmäßig mit wenig Schattenwurf ausgeleuchtet werden. Mittels einer 3D-CAD Software könnte man leicht zu diesem Zweck einen Halter für das Flutlicht entwickeln und mit einem 3D-Drucker ausdrucken: gesagt, getan.

Professionelle Dartboards haben einen Durchmesser von 45cm. Der Halter wurde passend zu diesem Durchmesser konstruiert. Zusätzlich kann an ein Dartboard ein Außenring aus weichem Kunststoff, ein sogenanntes *Surround*, angebracht werden. Dieser Außenring fängt Dartpfeile auf, die das Board verfehlt haben. Mit einem Surround kommt das Dartboard dann auf einen Außendurchmesser von 72cm. Auch hierauf ist die Konstruktion vorbereitet und das Surround passt ebenfalls in den Halter für das Flutlicht.

Die elektrischen Teile für das Projekt sind leicht und preiswert zu beschaffen. Als Stromquelle dient eine USB-Powerbank, die man vielleicht schon im Haushalt hat oder andernfalls heutzutage schon im Supermarkt an der Kasse findet. Als Faustformel reicht eine Powerbank mit 2,6Ah (Akkukapazität) für erfahrungsgemäß zwei Stunden Licht (ca. 500mA Strombedarf der LEDs).

Lichtquelle

Die Lichtquelle besteht aus zwei 50cm langen, selbstklebenden LED-Streifen. Die bekommt man entweder im Internet oder auch im Baumarkt um die Ecke. Sehr praktisch sind z. B.

Kurzinfo

- » Gute Ausleuchtung für eine Dartscheibe
- » Sparsame LED-Beleuchtung
- » Befestigung an Stangen und Pfosten

Checkliste



Zeitaufwand:

eine Stunde für den Zusammenbau



Kosten:

30 Euro (ohne Dartboard)



3D-Druck:

Drucker mit großem Bauraum (Ender 3), Druckzeit ca. 60 Stunden

Werkzeug

- » **Haushaltswerkzeug** (Schere, Cutter, Schraubendreher)
- » **3D-Drucker** großer Bauraum mit 250mm Höhe

Mehr zum Thema

- » Matthias Mett, Gehäusebau mit FreeCAD, Teil 1, Make 5/21, S. 116
- » Gerd Michaelis und Stella Maria Risch, 3D-Drucke clever kleben, Make 6/20, S. 114
- » Rebecca Husemann, Slicer Shootout, Make 4/20, S. 90
- » David Traum, Smarte Beleuchtung für die Treppe, Make 2/22, S. 34
- » Video: Aufbau und Benutzung

Material

- » **Dartboard** und Surround
- » **2 x LED-Streifen** mit USB-Anschluss, wetterfest und selbstklebend
- » **Powerbank** ab 2,6Ah
- » **Klebeband** Panzertape, Riggertape
- » **selbstklebendes Klettband** oder Doppelklebeband
- » **ca. 500g Filament**

Alles zum Artikel im Web unter make-magazin.de/x5s



die LED-Streifen von Hornbach, welche schon einen USB-Anschluss nebst Schalter im Kabel mitbringen. Hat man eine Powerbank, die man an- und abschalten kann, so reichen LED-Streifen ohne Schalter. Wasserfest gekapselte LED-Streifen sind praktisch und können einem verirrtten Pfeil besser widerstehen.

3D-Druck

Der 3D-Ausdruck der vier Bögen fordert den Maker und seinen Drucker schon etwas mehr. Mit dem Maßen 100 x 193,55 x 249,61 mm passt die Konstruktion gerade so eben auf einen Ender 3 oder einen Anycubic Vyper und muss



Beleuchtung an dem Board



Beleuchtung mit installiertem Surround



LED-Streifen, hier eine Variante ohne Schalter



Die drei zu verklebenden Stellen



Ein fertiger Bogen nach etwa 20 Stunden Druck

in der Slicing-Software richtig positioniert werden damit er komplett in den Bauraum passt. Der Druck der vier Bögen dauerte bei mir insgesamt etwa 60 Stunden. Je nach Dru-

cker und den verwendeten Einstellungen kann man hier sicher noch etwas optimieren. Dies ist eine lange Zeit, aber dank des recht sauberen 3D-Drucks auch ohne spezielle Holz- oder Metallwerkstatt in der Wohnung machbar.

Die fertige .STL-Datei steht als Download (Links in Kurzinfo) zur Verfügung und kann mit oder ohne Stützkonstruktionen gedruckt werden. Ohne Stützkonstruktionen wird der Druck zwar oben am Überhang durch das *Bridging* etwas unsauber, aber in der Regel klappt es trotzdem, so wird die Druckzeit gesenkt und weniger Filament verbraucht. Insgesamt werden vier Ausdrücke des Bogens für ein Dartboard benötigt. Auch eine .3mf-Datei (vom Prusa Slicer) passend eingerichtet für die *Ender 3* Drucker ist im Download zu finden.

Zusammenbau

Nach dem Ausdruck der vier Bögen werden jeweils zwei davon mit Klebeband zusammengefügt. Dies erwies sich in der Praxis als haltbarer und besser zu handhaben als andere Klebmethoden, zudem ist es so leicht wieder zu demontieren. Zwei Klebebänder verbinden jeweils die Streben und eines den Innenkreis der Bögen.

Um die Bögen an dem Dartboard zu befestigen, kann man entweder doppelseitiges

Klebeband oder selbstklebendes Klettband benutzen. Bei Verwendung des Klettbands hat man den Vorteil, dass man die Bögen später wieder leichter abnehmen kann, um z. B. das Board zu transportieren oder um die Bögen auf einem neuen Dartboard weiter zu nutzen.

Die beiden Bögen werden mittig zur Scheibe, links und rechts an dem Board angebracht. Die LED-Streifen sind ebenfalls selbstklebend und werden an der Führung der Bögen angebracht. Die Powerbank kann zum Beispiel mit Klettband an der Rückseite des Dartboards befestigt werden. Nun noch die LED-Streifen mit der Powerbank verbinden, das Flutlicht einschalten und die Partie Dart kann losgehen. Im Video zum Artikel sehen Sie den kompletten Aufbau, die Ausleuchtung des Boards und eine kurze Demopartie.

Sollte kein Baum zur Stelle sein, finden Sie bei den Druckdateien einen Halter für eine alternative Befestigung: Mit zweien davon kann man das Board an einer Stange mit vier Zentimeter Durchmesser befestigen, die dann z. B. an einem Dreibeinstativ oder einem Pfeiler befestigt wird. Andere Stangendurchmesser, etwa für Zaunpfähle, Pfosten von Hinweisschildern und andere Befestigungsarten sind sicher leicht durch Modifikation des Halters zu realisieren, wir freuen uns über Rückmeldungen per Mail. —caw



Alternative Befestigung an einer Stange



Die gleichmäßig ausgeleuchtete Scheibe



Die Heise-Konferenz
für Speichernetze und
Datenmanagement

13. und 14. Oktober 2022
in München

Jetzt Early-
Bird-Rabatt
nutzen!

Der Treffpunkt für Storage-Anwender

Die storage2day bietet Ihnen aktuelle **Informationen** und **Inspiration** sowie viel Raum für **anregende Gespräche** und **Networking** mit Gleichgesinnten.

Bringen Sie sich auf den aktuellen Stand bei Themen wie Storage-Architekturen, Security, Ransomware, Backup & Recovery, Storage-Performance, NVMe, DAOS ...

Lernen Sie, wie Sie Ihre Speicherkonzepte effizienter und sicherer gestalten, und schauen Sie mit uns auf die Trends von morgen.

www.storage2day.de

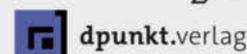
Platinsponsoren



Goldsponsoren



Veranstalter



Stromausfall-Monitor mit dem Pico

Eine vereiste Tiefkühltruhe, Lücken in den Server-Logs, abgestürzte Rechner: Gab es einen Stromausfall während der Abwesenheit? Eine simple Schaltung mit einem Raspberry Pi Pico und ausgefeiltem Python-Code kann einen Stromausfall feststellen und dokumentieren und so vor Ungemach schützen oder als Basis für eigene Entwicklungen ähnlicher Anwendungen dienen.

von Hans-Martin Hilbig



Vielleicht kennen Sie das: Man kommt nach einer mehrwöchigen Reise wieder nach Hause. Die Rückfahrt war lang, anstrengend und man freut sich auf die Tiefkühl-Pizza im heimischen Gefrierschrank, um schnell etwas im Magen zu haben. Jedoch ist die Pizza im Tiefkühlfach von einer Eisschicht überzogen. Gab es einen Stromausfall während ich nicht zu Hause war? Wenn ja, wann war er und wie lange dauerte er? Die Antworten auf diese Fragen entscheiden dann darüber, ob man die Pizza besser gleich in den Mülleimer entsorgt oder sie doch noch im Ofen aufbackt. Wie auch die *Stiftung Warentest* regelmäßig in ihren Kühlschranks-Tests notiert, fehlt ein solcher Netzausfallmonitor in den meisten Geräten. Also greifen wir zur Selbsthilfe.

Hardware-Minimalismus

Um nicht mit Kanonen auf Spatzen zu schießen, sollte der Hardware-Aufwand so gering und das Projekt so kostengünstig wie möglich gehalten werden **1**. Die Wahl fiel auf den *Raspberry Pi Pico*, ein Board, das man aktuell schon für unter vier Euro kaufen kann. Der im Pi Pico verwendete RP2040-Mikrocontroller verfügt über mehr als genug Bordmittel, um mit ein bisschen Software-Aufwand einen komfortablen Stromausfall-Monitor zu realisieren.

Lediglich die (Not-)Stromversorgung muss sichergestellt werden: Dies übernimmt ein alter USB-Lader (als permanente Stromversorgung) aus dem Maker-Fundus zusammen mit zwei AA- oder AAA-Batterien für die Zeit des Stromausfalls. Endlich hat man eine Anwendung für schwache Ladestecker, die als Akkulader als zu langsam aussortiert wurden. Der Pico begnügt sich mit nur etwa 30mA Stromaufnahme.

Diese unterbrechungsfreie Stromversorgung dient vor allem dazu, das RTC-Modul (**Real Time Clock**) des Mikrocontrollers für

Kurzinfo

- » Netzstromausfälle detektieren und aufzeichnen
- » Duale unterbrechungsfreie Stromversorgung des Pi Pico
- » MicroPython-Tipps zu Interrupts, PIOs und Stromsparen

Checkliste



Zeitaufwand:
1 Stunde



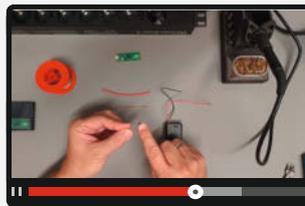
Kosten:
unter 10 Euro

Material

- » Raspberry Pi Pico nicht Pi Pico W
- » Schottky-Diode etwa 1N5817
- » Batteriehalter für zwei AA- oder AAA-Zellen nebst Kabel, mit Schalter
- » Stiftleisten abgewinkelt, 2,54mm Raster
- » Jumperkabel Dupont-Buchsen
- » Doppelklebeband mit Schaumkern
- » USB-Kabel USB-A nach Micro-USB
- » USB-Lader als Netz-Stromversorgung, 500mA

Mehr zum Thema

- » Make Pi Pico Special inkl. Raspberry Pi Pico, Juni 2022, im Heise Shop
- » Carsten Wartmann, Extra-Tastatur für Videokonferenzen, Make 3/21, S. 8
- » Andreas Perband, Raspberry Pi Pico, Make 1/21, S. 8



Werkzeug

- » Lötkolben und Zubehör
- » übliches Werkzeug Zangen, Seitenschneider, Schraubendreher ...

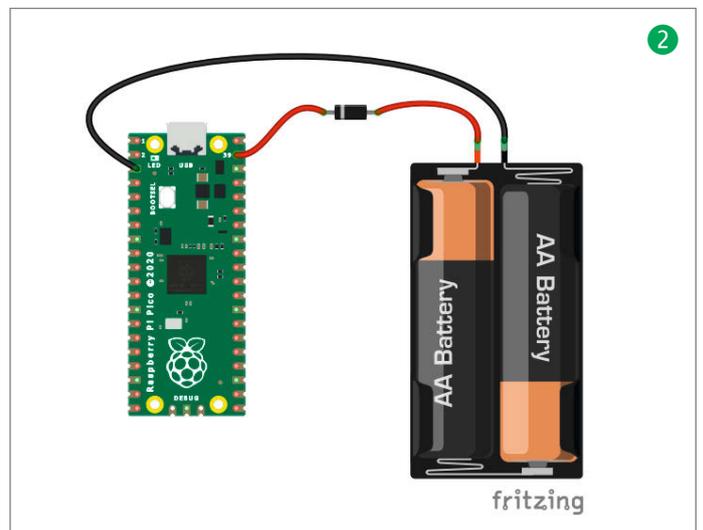
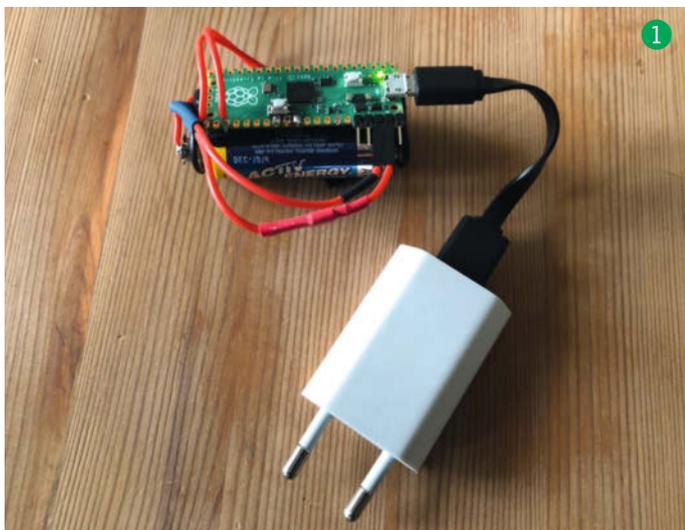
Alles zum Artikel im Web unter make-magazin.de/xzj7



Datum und Uhrzeit unter Spannung zu halten. Auf externe Displays oder Uhrzeit-Puffer-Platinen, wie sie reichlich für den Pi Pico angeboten werden, wurde bewusst verzichtet.

An der Software wurde hingegen nicht gespart. Während erfahrene Arduino-Maker meist mit Mikrocontroller-Funktionen wie RTCs, Interrupts und ADCs bestens vertraut

sind, haben Python-Programmierer oft nur geringe Erfahrung mit der Hardware-nahen Programmierung. MicroPython unterstützt aber ebenfalls die Verwendung der speziellen im RP2040 integrierten Module wie PIO, Interrupts, den Onboard-Flash-Speicher und sogar die Verwendung des zweiten CPU-Kerns des RP2040.



Stromausfälle

Wir haben in Europa ein recht stabiles Stromnetz. Allerdings steigen die Vernetzung und die Abhängigkeiten immer weiter, was dazu führt, dass bereits kleinere Störungen in den Netzen sich kaskadenartig ausweiten können. Krisenexperten rechnen mit einem großen Blackout in den nächsten fünf Jahren. Die Bundesnetzagentur verzeichnete einen durchschnittlichen Ausfall von etwa 10 Minuten pro Stromkunde im Jahr

Die Schaltung

Die Schaltung ② kann kaum einfacher sein. Egal, ob Sie diese auf einem Breadboard aufbauen, Steckerleisten und Jumper-Kabel verwenden oder löten, alles führt hier zum Ziel. Für einen mehr permanenten Aufbau wird allerdings eine schaltbare Batteriebox und das Löten empfohlen.

Bei der Polung der Diode ③ müssen Sie acht geben: Der Pluspol der Batteriebox muss an die Seite ohne Ring angeschlossen werden. In der Schaltung führt dann die positive Spannung an Pin 39 des Pico, der Übersichtlichkeit halber (keine kreuzenden Kabel im Schema) ist der Minuspol der Box hier an Pin 3 des Pico angeschlossen. Dies ist ebenso ein Masse-Pin (GND) wie Pin 38, den man in der Praxis wohl eher verwenden wird.

Die Schaltung ist so einfach, weil während der Entwicklung des Pico schon ähnliche Sze-

nen 2020. Damit gehört Deutschland zwar absolut zu den Ländern mit der höchsten Versorgungssicherheit. Dennoch: In der momentanen Energiekrise und angesichts der Diskussion, woher Strom in naher Zukunft kommen soll, ist auch die Angst vor künftig öfter drohenden Stromausfällen verständlich. Aber auch beim Camping oder in Ferienhäusern in anderen Ländern kann ein wenig Kontrolle nicht schaden.

narien angedacht waren. So ist ein Analog-Digital-Wandler (ADC, **A**nalog **D**igital **C**onverter) bereits mit der Versorgungsspannung verbunden und der Mikrocontroller kann diese so messen.

Der Spannungsregler auf dem Board kann eine Spannung bis zu 1,8V hinunter in die für den Pico nötigen 3,3V umsetzen, ideal für Batteriebetrieb. Weiterhin gibt es intern Anschlüsse, die eine Abfrage erlauben, woher der Pico mit Strom versorgt wird (USB oder extern). Weitere Informationen dazu lesen Sie in den Online-Ergänzungen zum Artikel.

Inbetriebnahme in Thonny

Der Stromausfall-Monitor-Code basiert auf *MicroPython 1.19* und der Entwicklungsumgebung *Thonny* ④. Der Pi Pico wird zunächst per USB an einen PC angeschlossen und alle drei notwendigen Python Skripte auf dem

Pico gespeichert (*main.py*, *brownout.py* und *pio_led_msg.py*, Download über Link in der Kurzinfor).

Thonny versorgt beim Start von *main.py* automatisch die RTC (Echtzeituhr) des RP2040 mit dem korrekten Datum und der Uhrzeit (Kasten *Zeiteinstellung*) vom Computer. Das genaue Vorgehen in Thonny und die Installation der Firmware wird in dem Video zum Artikel erklärt.

Ist *main.py* gestartet, so blinkt die interne LED des Pico viermal alle zwei Sekunden, der Pico ist im Initialisierungsmodus. Als Nächstes verbinden Sie die AA-Batterien mit den VSYS- und GND-Pins des Pico. Die LED des Pi Pico blinkt weiterhin viermal lang im Zwei-Sekunden-Rhythmus und signalisiert so den Initialisierungsmodus des Monitors unter USB-Versorgung (Kasten *LED-Codes*). Nun kann das Pi-Pico-Board vom USB-Port des Computers getrennt werden und läuft im Batteriebetrieb.

Dieser neue Betriebszustand wird durch drei kurze (50ms) Lichtblitze der Pi-Pico-LED im Zwei-Sekunden-Rhythmus angezeigt. Im dritten und letzten Schritt wird der Stromausfall-Monitor mit einem USB-Ladestecker verbunden.

Der USB-Ladestecker wird in eine passende und zu überwachende Haushaltssteckdose gesteckt und damit ist der Netzausfallmonitor betriebsbereit, was durch einen Blinkimpuls alle zwei Sekunden angezeigt wird.

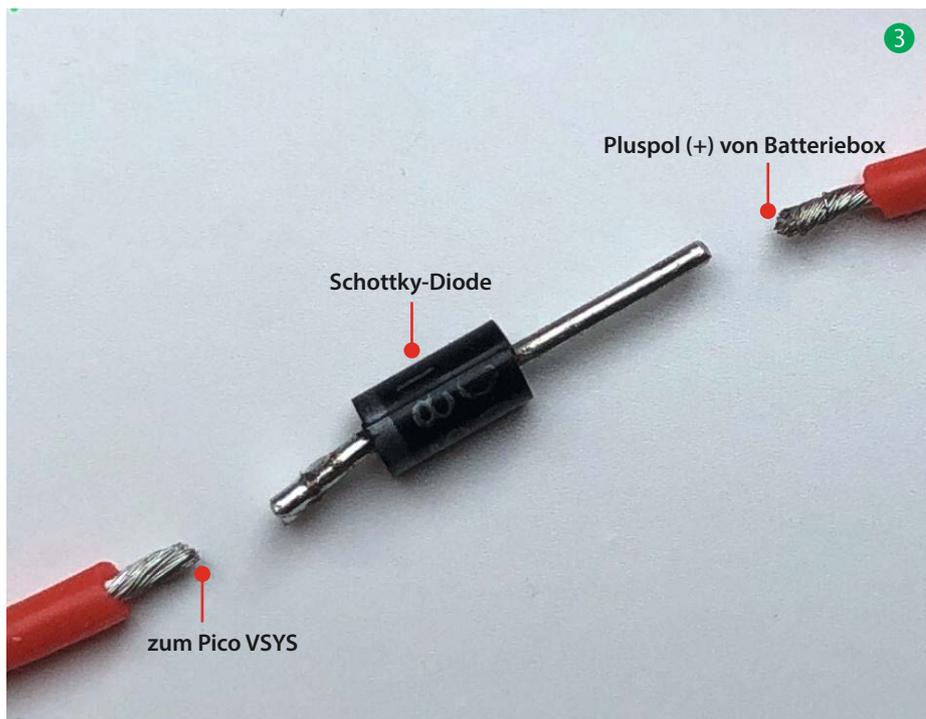
Der Betriebszustand wird weiterhin über die LED des Pico angezeigt. Datum und Uhrzeit des Anschließens an die Steckdose werden in einer Datei im Flash-Memory des Pico gespeichert. In dieser Datei werden alle Stromausfälle, der jeweilige Betriebszustand des Monitors sowie die Batteriespannung der Batteriebox zum Zeitpunkt des Stromausfalls protokolliert. Als Datei-Format wurde ein durch Kommas getrenntes Text-Format (**.csv**: **c**omma **s**eparated **v**alues) gewählt, das man in ein Tabellenkalkulationsprogramm wie *Microsoft Excel* einlesen und analysieren kann oder sich einfach in Thonny anschaut.

Fertig?

Wer nur einen einfachen Netzstrom-Monitor als Nachbauprojekt sucht, ist hier fertig. Wer tiefer in die Materie einsteigen möchte, bekommt in den nächsten Abschnitten, im Online-Teil und im Video (Links in der Kurzinfor) weitere Informationen.

Ans Eingemachte

Die Software des Netzausfall-Monitors besteht aus drei voneinander getrennten Python-Skripten. Das eigentliche Haupt-Skript *main.py* wurde bewusst kurzgehalten und die benötigten Funktionen in zwei Python-Module ausgelagert, um eine Wiederverwendung in anderen Projekten zu erleichtern. Im Nach-



LED-Codes

- » Initialisierung nach dem Start: vier 200ms lange LED-Impulse im Zwei-Sekunden-Takt
- » Normalmodus, Netzspannung liegt an: ein 200ms langer LED-Impuls im Zwei-Sekunden-Takt
- » Zeitinformation fehlt: zwei 200ms lange LED-Impulse im Zwei-Sekunden-Takt
- » Batteriespannung ist niedrig: drei 200ms lange LED-Impulse im Zwei-Sekunden-Takt
- » Netzspannung ausgefallen: ein 50ms kurzer LED-Blitz im Zwei-Sekunden-Takt
- » Stromausfall-Zähler: die Anzahl kurzer LED-Blitze nach einem langen LED-Impuls zeigt die Anzahl der bisher erkannten Stromausfälle an

folgenden wird auf alle drei Skripte eingegangen und darauf, wie diese mit der Hardware zusammen arbeiten.

LEDs und Pulsfolgen

Die Art mit einer LED Betriebszustände zu signalisieren, kennen auch Multicopter-Piloten von den ESCs (Motorreglern) und Flightcontrollern: Statt aufwändiger Displays wird der Betriebszustand über eine LED oder eine Tonfolge signalisiert, die je nach Modus aus einer Reihe kurzer oder langer Licht- bzw. Tonfolgen besteht. Solche Impulsfolgen kann man mit Mikrocontrollern auf verschiedene Arten realisieren:

1. Direkt per GPIO: Die CPU verschwendet Zeit in einer Schleife und tut nichts anderes, als die LED an- und auszuschalten.
2. Mittels Timer-Funktionen für die Impuls- und Pausenzeit: Die CPU kann in der Zwischenzeit sinnvolle weitere Aufgaben übernehmen.
3. Mithilfe einer programmierbaren State Machine: Das ist eine Art Coprozessor (PIO im Pico), der unabhängig vom Hauptprozessor arbeitet.

Hier soll die dritte Methode benutzt werden, um unsere Pulse zu generieren: Für unseren Zweck sicher übertrieben, aber es geht um den Lerneffekt, und wer wollte nicht einmal einen Coprozessor auf einem Mikrocontroller benutzen.

Im RP2040 gibt es tatsächlich acht solcher State Machines, verteilt auf zwei sogenannte PIO (Programmable I/O Module). Das Python-Skript `pio_led_msg.py` enthält den dafür notwendigen Code. Die State Machine wird in PIO-Assemblercode programmiert.

Als Schnittstelle zwischen PIO-Code und MicroPython-Code dienen die `put()`- und `get()`-

```

1 # Release candidate 2, 17.Juni 2022
2
3 import brownout
4 import pio_led_msg
5 from machine import lightsleep, freq
6 from time import sleep
7
8 sleep(1) # warte 1 Sekunde mit dem Programmstart, nur fuer den Fall, i
9 freq(5000000) # der System-Clock wird auf 50MHz reduziert,
10 mydis = pio_led_msg.led_msg() #erzeugt das display Objekt zur Statusan:
11 mybrownout = brownout.brownout() #erzeugt das brownout Objekt zum Erkennen un
12
13 while True:
14     status = mybrownout.check_status() #holt den momentanen Status vom Stromar
15     if mydis.done(): # Zeige den neuen Status ueber Pi Pico LED an, falls d
16         if mybrownout.in_power_loss(): # Stromausfall erkannt?
17             mydis.msg(0,status) # ändere Statusanzeige in kurze Lichtblit:
18             # Kein Stromausfall:
19             mydis.msg(status,mybrownout.power_loss_events) # Statusanzeige übr
20             lightsleep((500 + mybrownout.power_loss_events *350)+2000) # legt die Cl
21
Kommandozeile
>>>
>>>
MicroPython v1.19.1 on 2022-06-18; Raspberry Pi Pico with RP2040
Type "help()" for more information.
>>>
MicroPython (Raspberry Pi Pico)

```

Funktionen, die Werte aus Python-Variablen in den Speicher (Tx-Fifo, First In, First Out) der State Machine exportieren oder Daten aus dem Rx-Fifo der PIO in die Python-Welt importieren. Die State-Machine läuft unabhängig von den eigentlichen CPU-Kernen. Der PIO-Assembler (pioasm) kommt mit nur neun Instruktionen aus, die es aber in sich haben. Lesern, die sich intensiv mit den PIOs beschäftigen wollen, empfehle ich, sich zunächst den ergänzenden Online-Artikel anzuschauen.

Während die PIO-Einheit über die LED des Pico den Betriebszustand des Stromüberwachungsmonitors und die Anzahl der bisherigen Stromausfälle morst, kann die CPU des RP2040 sich also unabhängig davon mit der eigentlichen Überwachung der Netzspannung beschäftigen oder in den `lightsleep()`-

Zustand (siehe Online-Teil zum Stromsparen) gehen.

Überwachung der Netzspannung

Alle Funktionen zur Überwachung der Netz- und Batteriespannung sowie zum Speichern der Daten im Flash des Pi Pico sind im `brownout.py` Modul enthalten.

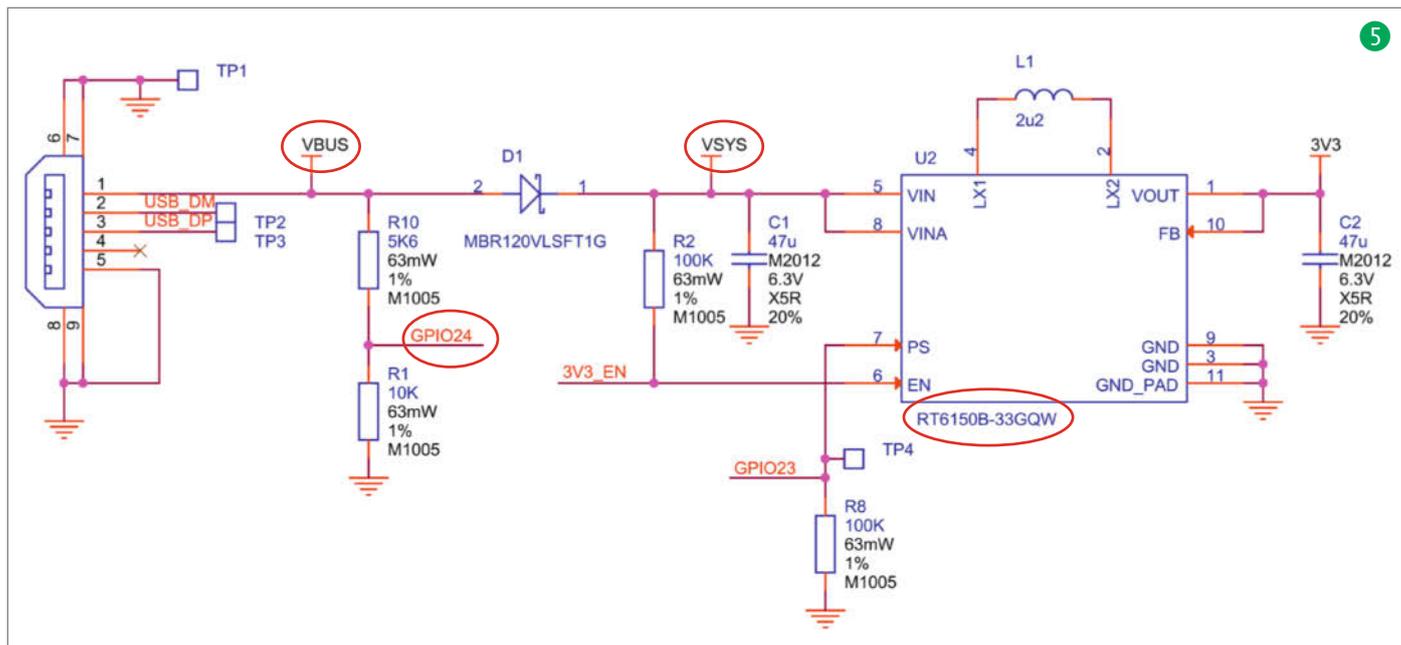
Der Pi Pico wird normalerweise mit 5V vom USB-Stecker (VBUS) versorgt 5, der Spannungsregler macht daraus die für den RP2040 nötigen 3,3V. Fällt die USB-Versorgung wegen Stromausfall aus, wird der Pico über die zwei Batterien an VSYS versorgt. Das Power-Management-IC (RT6150B) sorgt für geregelte 3,3V-Versorgung des Pi Pico über

Zeiteinstellung

Damit die richtige Zeit im Pico vorhanden ist, muss das `main.py`-Skript von Thonny aus gestartet werden. Dabei wird die Uhrzeit vom Computer mit übertragen. Der Raspberry Pi Pico hat keine batteriegepufferte Echtzeituhr, die weiterläuft, wenn der Strom komplett weg ist. Solange der Monitor am Netzstrom hängt, können Sie die Batterien

in der Box auswechseln, falls Sie eine Batteriewarnung sehen (dreimal blinken im Netzmodus, dreimal blitzen im Batterie-modus).

Eine autonome Inbetriebnahme ohne Computer wäre nur durch das Einbinden einer externen, dauernd mit Strom versorgten Echtzeituhr möglich.



Raspberry Pi Pico Datasheet

einen breiten Eingangsspannungsbereich von 5,5V bis hinunter auf 1,8V, ideal für eine Versorgung aus einer breiten Auswahl von Stromquellen.

In die Zuleitung zur Batterie wird, wie vorhin in der Bauanleitung gezeigt, eine Schottky-Diode (2) gelötet, um einen unerwünschten Stromfluss vom USB-Port in die Batterie zu verhindern, solange das Pico-Board vom USB-Stecker versorgt wird. Im Schaltplan der Pico-Stromversorgung (5) finden Sie übrigens ebenfalls eine Schottky-Diode (D1), die hier verhindert, dass eine an VSYS angelegte Spannung in den USB-Port zurückfließt.

Über die Schottky-Diode fallen etwa 0,3 Volt ab, daher dürfen die beiden Batterie-Zellen nur bis zu einer Schlussspannung von 1,05V pro Batterie betrieben werden, da ansonsten der Spannungswandler nicht genügend Spannung (1,8V Minimum) bekommt.

Glücklicherweise sind auf dem Pico-Board sowohl der GPIO-Pin 24 und ein ADC-Pin (Analog-Digital-Wandler, GPIO29) mit VSYS verdrahtet, um den RP2040 über den Zustand seiner Versorgungsspannung zu informieren und ihn über unser Programm abzufragen. GPIO24 liegt über einen Spannungsteiler (R1/R10) an der USB-Stromversorgung (VBUS, Pin 40) des Pi Pico an. Wird der Pico mit USB-Spannung versorgt, signalisiert GPIO24 dies mit einer logischen Eins, bei Stromausfall ändert sich dies zu einer logischen Null und kann so einfach im Programm abgefragt werden.

Spannungsprüfer – der ADC

Wie im Bild (6) zu sehen, ist der Kanal 3 des ADC (GPIO29_ADC3) über einen Spannungsteiler (R5/R6) mit dem VSYS-Pin des Pico

verbunden. Auf diese Weise kann man sehr einfach auch die Spannung im Betrieb messen. Um den tatsächlichen Spannungswert zu erhalten, muss die Ganzzahl (16Bit, 0-65535), die vom ADC zurückkommt, umgerechnet werden. Dabei ist sowohl der 1:3-Spannungsteiler als auch die Referenzspannung von 3,3V einzubeziehen. Der Berechnungsfaktor wird in *brownout.py* in Zeile 25 als `self.cf = 3.3*3/65535` definiert.

Bei jedem Wechsel der Betriebsart (siehe nächster Abschnitt), wird auch die VSYS-Spannung in der Flash-Datei protokolliert. Eine Spannung unterhalb von 2,0V im Batteriebetrieb ändert den Status des Stromausfall-Monitors auf *Low-Bat* und wird über drei LED-Impulse signalisiert: Wenn der Pico per USB-Spannung versorgt wird, sind dies lange Pulse, unter Batteriebetrieb kurze Blitzer. Im Online-Teil des Artikels wird auch betrachtet, wie lange der Monitor bei Netzausfall mit zwei Batterien hält und wie man durch Softwaretricks die Laufzeit bei Stromausfall auf über 100 Stunden streckt.

Netzspannungsausfall-Alarm!

Damit sowohl der Beginn als auch das Ende eines Stromausfalls erkannt wird, muss ein Eins-Null-Übergang (Stromausfall) und auch ein Null-Eins-Übergang (Strom wieder da) an GPIO24 erkannt werden. Man könnte den GPIO24-Eingang in einer Python-Schleife permanent abfragen, aber es gibt eine viel elegantere Lösung: Eine Zustandsänderung von GPIO24 löst einen Interrupt (Unterbrechung des Hauptprogramms) im Python-Code (*brownout.py*) aus, unabhängig davon, was der

Pico gerade macht. Zwei Dinge müssen in Python dafür programmiert werden:

1. Dem betreffenden GPIO-Pin wird eine Interrupt-Funktionalität zugewiesen. Dort wird bestimmt, ob eine steigende (*Pin.IRQ_RISING*, Zeile 28) oder eine fallende (*Pin.IRQ_FALLING*) Flanke einen Interrupt auslösen soll. Für den Stromausfall-Monitor werden die fallende (Beginn des Stromausfalls) und die steigende (Stromausfall beendet) Flanke als Auslöser benötigt. Dies wird durch die ODER-Verknüpfung beider Bedingungen erreicht.
2. Die Funktion, die beim Auslösen eines Interrupts aufgerufen werden soll (*isr_event()*)

Die Funktion aus Punkt 2 nennt man auch *Interrupt Service Routine (ISR)*. In MicroPython gibt es für ISRs zwei wesentliche Regeln zu beachten:

1. Die ISR sollte so einfach und kurz wie möglich gehalten werden, um andere Interrupts nicht zu blockieren.
2. Dynamische Speicherverwaltung, wie sie zum Beispiel bei Listen, Dictionaries, bei Fließkomma-Mathematik oder beim Dateizugriff in Python vorkommt, müssen vermieden werden.

In unserem Fall müssen bei einem Stromausfall die Uhrzeit zum Zeitpunkt des Interrupts, der Zustand von GPIO24 und die Batteriespannung über den ADC abgefragt werden und alle Daten in den Flash Speicher des Picos geschrieben werden (Verstoß gegen Regel 2). Allein für das Schreiben ins Flash braucht der Pico über zehn Millisekunden und man würde damit auch gegen Regel 1 für ISRs verstoßen.

Als Ausweg benutzen wir einen Aufruf der `micropython.schedule()`-Funktion: Damit

Blaue Pille für Arduino-Aufsteiger

Ein ARM-Kern mit 72MHz, 64KByte Flash, 20KByte RAM und reichlich Schnittstellen – der STM32F103C8T6-Controller auf dem sogenannten Blue-Pill-Board bietet zum gleichen Preis ungleich mehr als ein Arduino-Nachbau. Wir zeigen, wie man die Maker-freundliche Platine mit verschiedenen Entwicklungsumgebungen zum Leben erweckt.

von Carsten Meyer



Manchmal darf es auch etwas mehr sein: Mit der 32 Bit breiten ARM-Cortex-M3-Architektur des STM32F103C8T6 und einer Taktfrequenz von 72MHz überrundet die sehr preiswerte *Blue-Pill*-Platine einen ATmega328P-Arduino (16 MHz Taktfrequenz) gleich mehrfach. Der mit 64KByte Flash doppelt so große Programmspeicher und vergleichsweise üppige 20KByte RAM tragen der Tatsache Rechnung, dass 32-Bit-Programme mehr Speicherplatz benötigen; der Mehrbedarf relativiert sich natürlich, wenn vorwiegend Werte mit 16 oder 32 Bit Breite zu verarbeiten sind, was ein 8-Bitter immer in mehrere Teilschritte zerlegen muss.

Daneben beinhaltet das Board eine mit einem separaten Quarz auf 32.768 MHz getaktete Echtzeituhr (RTC), allerdings ohne Stützbatterie. Im Normalbetrieb verwendet der Mikrocontroller Logikpegel von 3,3V, es sind auf einigen Pins aber auch 5V Eingangspegel zulässig. Ansonsten gibt sich das Board erfrischend bescheiden **1**: Auf eine üppige Ausstattung mit allerlei Peripheriebausteinen wie zum Beispiel auf den STM-Discovery-Boards hat der chinesische Hersteller verzichtet. Selbst ein Programmieradapter oder ein Interface-Baustein für die serielle Schnittstelle sucht man vergeblich.

Das muss kein Nachteil sein: Für den jeweiligen Anwendungsfall nicht benötigte Bauteile muss man auch nicht bezahlen; das Board ist als China-Import deutlich unter drei Euro zu bekommen. Außerdem blockieren überflüssige Peripherie-Chips oft I/O-Pins, I²C- oder SPI-Schnittstellen, die man extern gut gebrauchen könnte.

Alle an die DIL-Stiftleisten herausgeführten I/Os lassen sich frei benutzen, bis auf ganz wenige Ausnahmen: Die Port-Pins PC14 und PC15 sind mit dem Uhrenquarz belegt, was aber nicht stört, solange man die RTC nicht verwendet; in der Regel ist die parasitäre Kapazität des Quarzes (einige zehn Picofarad) zwischen PC14 und PC15 zu vernachlässigen. PA11 und PA12 sind zur USB-Buchse geführt, sie lassen sich softwaremäßig als USB-Schnittstelle einrichten. Für die Behandlung des USB-Protokolls muss dann allerdings die Firmware sorgen. Wird USB nicht gebraucht, ist nur noch der Pull-down-Widerstand R10 (4,7kOhm) an PA12 zu beachten; sollte er stören, kann man ihn auch auslöten.

Die zwei gelben Jumper stellen den Boot-Modus ein (siehe Tabelle), normalerweise stehen beide auf „0“: Damit startet die Programmausführung der eigenen Firmware aus dem Flash-Speicher. Sollte man versehentlich die zur Kommunikation mit dem *Programmer* (siehe unten) nötigen SWD-Pins in der Firmware abgeschaltet haben, setzt man *BOOT 0* (nahe der DIL-Stiftleiste) zum erneuten Flashen auf „1“. Der Controller startet dann nicht mit der (fehlerhaften) Firmware, sondern mit einem internen ROM-Bootloader, der die SWD-Pins

Kurzinfo

- » Bluepill-Board: Eigenschaften und Besonderheiten
- » Compiler und Entwicklungsumgebungen für ARM-Einsteiger
- » „Hello World“ für verschiedene IDEs
- » Mit HAL oder hardwarenah programmieren – Vor- und Nachteile

Checkliste



Zeitaufwand:
1 bis 5 Stunden



Kosten:
15 Euro

Material

- » Blue-Pill-Board mit STM32F103C8T6
- » ST-Link-Adapter für USB
- » Jumper-Kabel female/female
- » USB-Steckernetzteil mit Micro-USB-Kabel (optional)

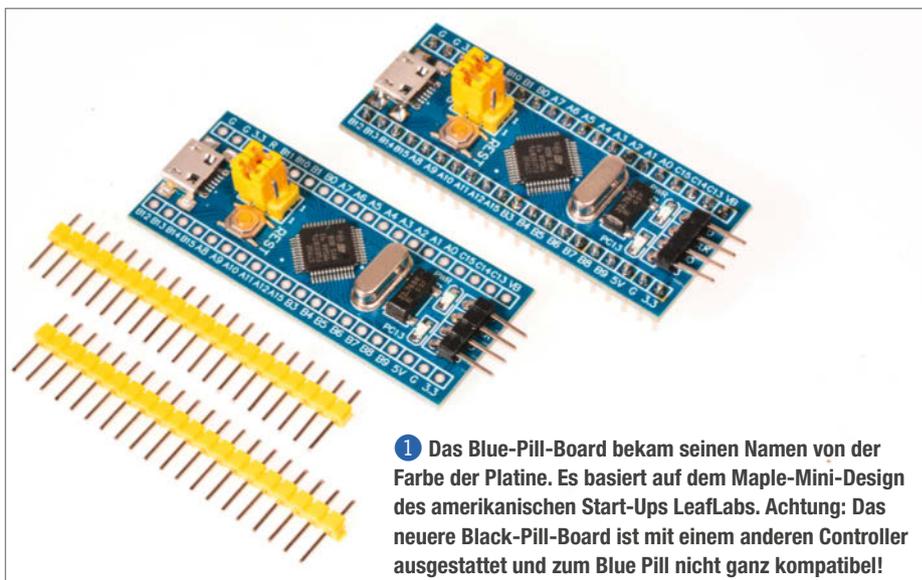
Mehr zum Thema

- » Maik Schmidt, STM32F4-Discovery: Programmieren bis aufs blanke Metall, Make: 3/13, S. 132
- » Carsten Meyer, Chipkrisenmanagement, Make: 3/22, S. 78

Alles zum Artikel im Web unter make-magazin.de/x7nz

wieder in den Programmiermodus setzt. Nach dem Hochladen der Firmware stellt man ihn auf „0“ zurück. Stehen beide Jumper auf „1“, würde ein temporär ins RAM geladenes Programm gestartet; das dürfte nur in Ausnahmefällen sinnvoll sein.

Die Blue Pill enthält einen kleinen Low-Drop-Spannungsregler, der die Betriebsspannung von 3,3V aus den 5V des USB-Ports oder dem 5V-Pin aufbereitet. Einige Ausführungen bieten eine 800mA-Ausführung, erkennbar am SOT-223-Gehäuse mit dem großen Lötpad

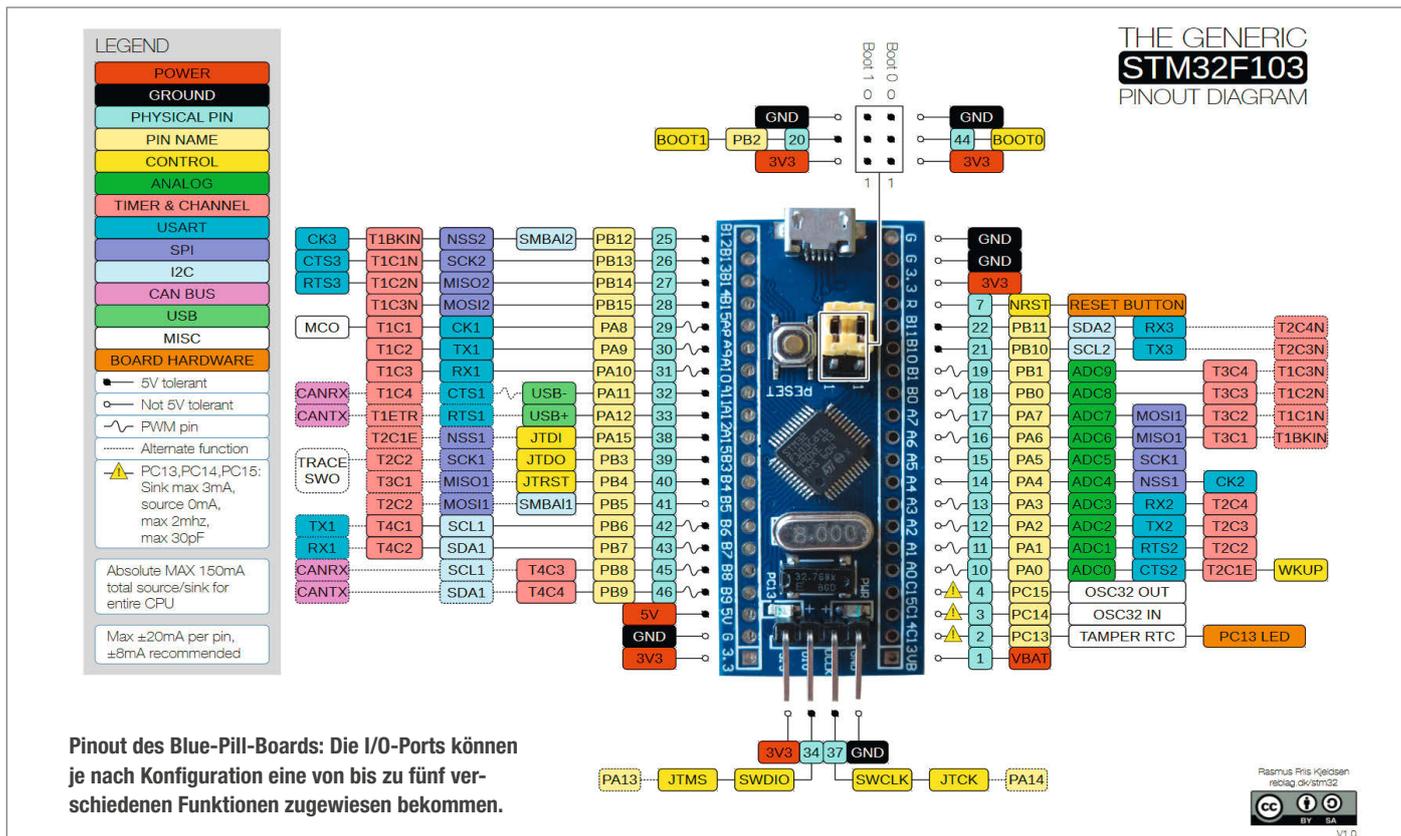


1 Das Blue-Pill-Board bekam seinen Namen von der Farbe der Platine. Es basiert auf dem Maple-Mini-Design des amerikanischen Start-Ups LeafLabs. Achtung: Das neuere Black-Pill-Board ist mit einem anderen Controller ausgestattet und zum Blue Pill nicht ganz kompatibel!

Boot-Mode-Jumper

BOOT 0 (nahe Stiftleiste)	BOOT 1 (nahe Taster)	Modus
0	(egal)	Main Flash (default)
1	0	System ROM
1	1	Int. SRAM

Im Normalbetrieb stehen beide Jumper auf "0".



oben; unsere Boards waren allerdings nur mit einem winzigen RT9193-3.3 ausgestattet, der maximal 175mA verträgt. Die 3,3V-Leitung darf hier nur mit wenigen (zusätzlichen) Milliampere belastet werden – mehr als 50mA (etwa für externe LEDs) sollte man dem Spannungsregler nicht zumuten. Es ist übrigens durchaus zulässig, die 3,3V auch extern zuzuführen, etwa über ein kräftigeres Regler-IC.

Lebenszeichen

Einziges Peripherie-Bauteil (so man es so nennen darf) ist die an PC13 angeschlossene LED

(bei unseren Platinen grün). Ihre Anode ist mit der 3,3V-Betriebsspannung verbunden, sodass sie ein Low-Pegel an PC13 zum Leuchten bringt. In unseren Beispielprogrammen wird sie als „Lebenszeichen“ verwendet: Die blinkende LED ist ja sozusagen das „Hello World“ der Mikrocontroller. Ist diese (keineswegs zu unterschätzende) Hürde erst einmal geschafft, ist der Weg zur eigenen Anwendung nicht mehr weit.

Das Blue-Pill-Board ist im Prinzip ein Nachbau des *Maple Mini* vom amerikanischen Start-Up *LeafLabs*. Das Original war noch mit einem Bootloader ausgestattet, sodass man es ohne Programmieradapter über USB flashen

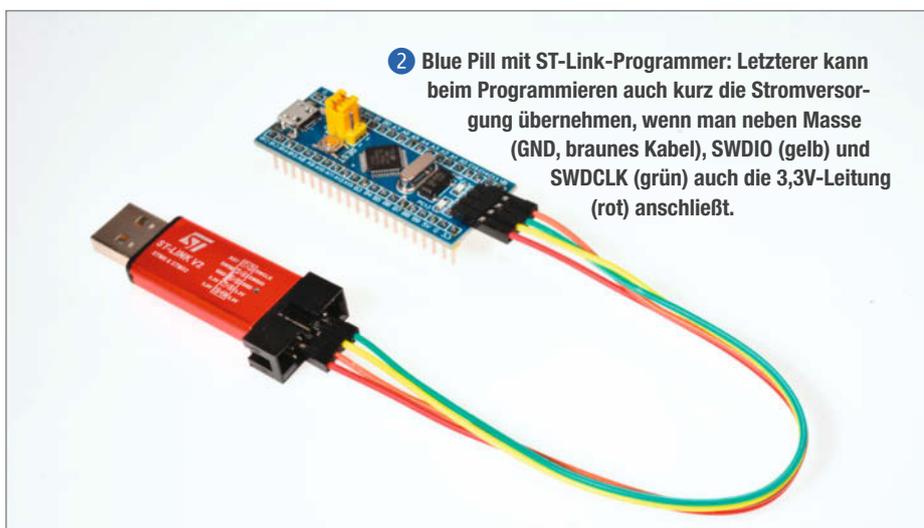
konnte. Dieser Vorteil wurde allerdings mit rund 20KByte Flash-Speicher für den Bootloader erkauft, der für das Anwendungsprogramm verloren war. Der im Vergleich zum Arduino erheblich größere Bootloader muss hier ja auch das komplette USB-Protokoll abwickeln, da kein separater Interface-Baustein enthalten ist.

Die Nachbauten verzichten auf den Bootloader (der sich allerdings nachrüsten lässt – siehe Link im Info-Kasten) und stellen dem Anwender lieber den kompletten Flash-Speicher zur Verfügung. Zum Aufspielen der eigenen Firmware (oder auch des nachträglichen Bootloaders!) benötigt man deshalb zwingend einen Programmieradapter (Programmer) 2. Das erfordert zum Glück keine größere Investition: Passende ST-Link-Programmer findet man bei eBay & Co. schon für 3 Euro, wenn man ein paar Wochen auf die Lieferung aus China warten kann.

Wir bezogen unseren Programmer von *Funduino*; dort war er zwar etwas teurer, lag aber schon nach zwei Tagen im Briefkasten, kam in einem schmacken Metallgehäuse und war richtig beschriftet. Einige ST-Freunde berichten in einschlägigen Foren über Adapter, die nur deshalb nicht funktionierten, weil das Pinout falsch aufgedruckt war – das ist das typische Risiko bei Fernost-Bestellungen.

Flash-Hardware

Großer Vorteil der ST-Link-Programmer ist, dass sie sich auch als Debug-Interface für die



schrittweise Abarbeitung von Programmen aus der IDE heraus verwenden lassen. Sie kommunizieren mit dem STM32-Controller über lediglich zwei (dedizierte) Pins; es gehen sonst keine I/Os für das Debug-Interface verloren. Die Programmier-Pins SWDIO und SWDCLK sind (neben den Betriebsspannungs-Anschlüssen 3V3 und GND) an vier Stiftleisten-Pins auf der Platinen-Oberseite herausgeführt. Zum Flashen des Controllers verbindet man sie einfach mit den gleichnamigen Pins des Programmers.

Wer noch ein Nucleo- oder Discovery-Board von ST besitzt, kann den dort enthaltenen ST-Link-Programmer von der restlichen Schaltung abklemmen und ihn als eigenständiges Gerät benutzen. Vorteil: Ein solches Interface ist Update-fähig und uneingeschränkt zu allen ST-Softwaretools kompatibel.

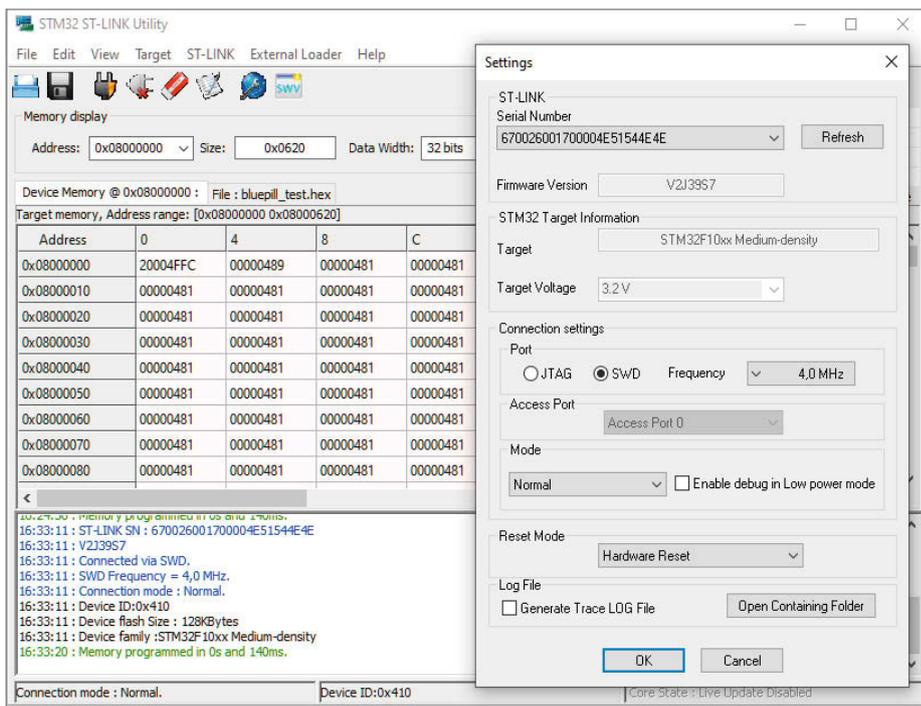
Nach dem Einstöpseln des Programmers in eine freie USB-Buchse des Rechners (es empfiehlt sich ein USB-Verlängerungskabel A-A) wird die rote Betriebsspannungs-Indikator-LED des Boards leuchten, sofern man wie im Bild auch die 3,3V-Leitung angeschlossen hat. Der PC wird sich nun beschweren, dass er keinen Treiber für den ST-Programmer findet.

Tools und Treiber installieren

Den ST-Link-Treiber *STSW-LINK009* von ST Microelectronics brauchen Sie fortan, auch wenn sie nicht deren IDE zur Firmware-Erstellung verwenden. Sie finden ihn unter dem Link im Info-Kasten, ebenso wie Bezugsquellen für die im folgenden besprochenen Tools und Entwicklungsumgebungen. ST bietet den *STM32CubeProgrammer* und das *ST-LINK Utility* als Programmier-Tools, die eigenständig und auch ohne die ST-Entwicklungsumgebung einsetzbar sind – wenn es etwa darum geht, den STM32F103 mit einem bereits fertigen Hex-File zu füllen.

Der *STM32CubeProgrammer* funktioniert leider nicht mit einigen billigen ST-Link-Nachbauten, aber garantiert mit den Originalen des Herstellers und der auf den ST-Entwicklungsboards verbauten ST-Link-Elektronik. Das ist nicht allzu tragisch, das ältere *ST-LINK Utility STSW-LINK009* tut es genauso. Das Tool erkennt den STM32F103 nach einem *Target Connect* (immer notwendig, wenn ein Baustein oder der Programmer neu angeschlossen wurde) automatisch. Unter *Target/Settings* sollte der ST-Link-Programmer mit einer (langen) Seriennummer erscheinen, stellen Sie hier den Port SWD mit 4MHz Takt wie in 3 gezeigt ein.

Auch wenn Sie später Ihre Firmware mit anderen ARM-Entwicklungsumgebungen (dazu später) erstellen wollen, empfehlen wir trotzdem, zunächst die aktuelle (und leider recht füllige) *STM32CubeIDE* zu installieren – deren Pinout- und Clock-Wizard erleichtert die grundlegende Konfiguration des Chips ungemein.



3 Im Unterschied zum neuen STM32CubeProgrammer funktioniert das klassische STM32 Link Utility auch mit den chinesischen Programmer-Nachbauten.

Takt-Mysterien

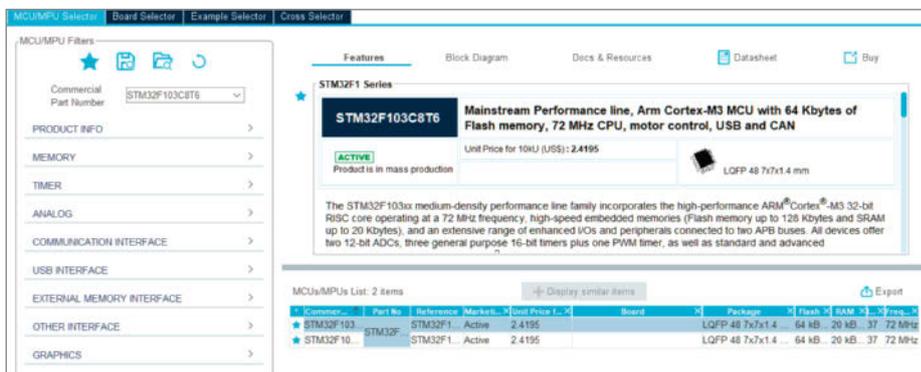
Wie viele andere ARM-Chips auch besitzt der STM32F103 eine geradezu erschlagende Anzahl an Takt- und Pin-Konfigurationsregistern. Während man beim AVR einfach ein paar Fuses setzte, um die Taktquelle richtig einzustellen, sind beim STM32 eine Unmenge von Register-Bits richtig zu setzen, damit Prozessorkern und eingebaute Peripherie mit den passenden Frequenzen beliefert werden. Dies geschieht erst in der Firmware, die also eine geeignete Initialisierung ausführen muss – oft unbemerkt vom Anwender, weil der Compiler entsprechende Routinen automatisch einbindet.

AVR-Kenner werden sich zunächst über den 8MHz-Quarz als Taktquelle wundern – wo doch der Hersteller 72MHz versprochen hat. Tatsächlich läuft der Baustein nach einem

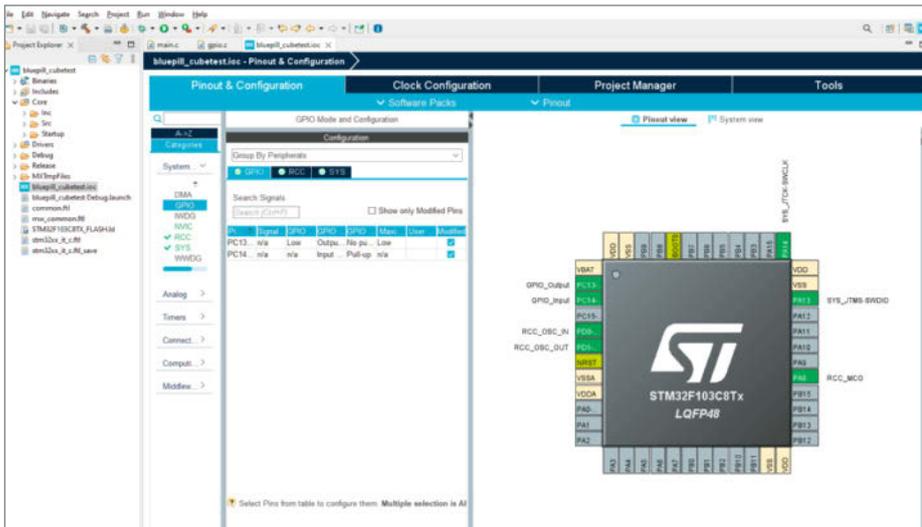
Hardware-Reset zunächst mit beschiedenem Takt los, die endgültige Taktrate legt die Firmware in ihrer Initialisierungsroutine selbst fest.

Das Geheimnis hinter dem einstellbaren Takt ist eine Schaltung namens *Phase Locked Loop* oder kurz *PLL*, bestehend aus einem spannungsgesteuerten Oszillator, einem Frequenzteiler und einem Frequenz-Vergleicher. Das Ganze ergibt einen Frequenz-Vervielfacher mit über Register einstellbarem Multiplikator. Stellt man den auf Faktor 9, ergeben sich die versprochenen 72MHz.

Über weitere Register-Bits kann man noch festlegen, ob der Systemtakt aus einem internen RC-Oszillator abgeleitet wird oder von der PLL kommt, ob er vor oder nach der PLL noch geteilt werden soll und welche Teilfaktoren für die Ports und Timer gelten sollen. Und dann gibt es ja noch die unzähligen Zweit- und



4 Im Projekt-Wizard wählt man zunächst den gewünschten Chip, dann den Projektnamen und die zu verwendende Sprache (C oder C++).



5 Im Pinout-Konfigurator weist man den Controller-Pins die gewünschte Funktion zu. Graue Pins werden nicht benutzt.

Drittfunktionen der I/O-Pins, die sich als Timer- oder PWM-Ausgang, als I²C- oder SPI-Bus oder als UART-Pins verwenden lassen – was natürlich auch über viele, viele Register eingestellt werden muss.

Diesen Wust an Registern kann man „zu Fuß“ konfigurieren, wenn man sich mit dem Prozessor-Datenblatt drei Tage in ein stilles Kämmerlein einschließt, oder man lässt es sicher und bequem von der CubeIDE erledigen – dazu kommen wir gleich.

STM32CubeIDE einrichten

Die ST-Entwicklungsumgebung ist kostenlos, aber ein ziemliches Trumm, das auf der mäch-

tigen Eclipse-IDE basiert und einige Minuten zur Installation benötigt. Wir beschreiben hier die Windows-Version, es gibt sie aber auch für MacOS und Linux; die GUIs unterscheiden sich nur marginal. ST verlangt eine Anmeldung auf deren Webseite, auch beim Download von weiteren Tools. Beim ersten Start der IDE legt man das Stammverzeichnis für eigene Projekte fest.

Nach dem Start der IDE legen Sie mit dem Projekt-Wizard ein neues Projekt an 4; tragen Sie im MCU/MPU Selector unter Commercial Part Number den Typ STM32F103C8T6 ein, wählen Sie unter den nun unten vorgeschlagenen Bausteinen den ersten und klicken Sie Next. Die nun folgende Dialogbox verlangt einen

Projektnamen (wir wählten *bluepill_cubetest*), dann klicken Sie ohne weitere Änderungen auf *Finish*. Sie landen dann wieder in der IDE, nun aber mit einem leeren Projekt-Framework.

Im Dateibaum links erkennen Sie die Dateien, die der Compiler für das Projekt verwendet. Ihr *main.c* finden Sie unter *Core/Src*. Doppelklicken Sie zunächst aber auf die automatisch angelegte Datei *bluepill_cubetest.ioc* mit dem blauen MX-Icon. Es öffnet sich ein Editor für die Pin-Konfiguration 5 mit einem Abbild des Controllers; hier legen wir fest, welche Pins benutzt werden und welche Funktion sie erfüllen sollen. Beige und olivgrüne Pins haben besondere Funktionen und können nicht geändert werden.

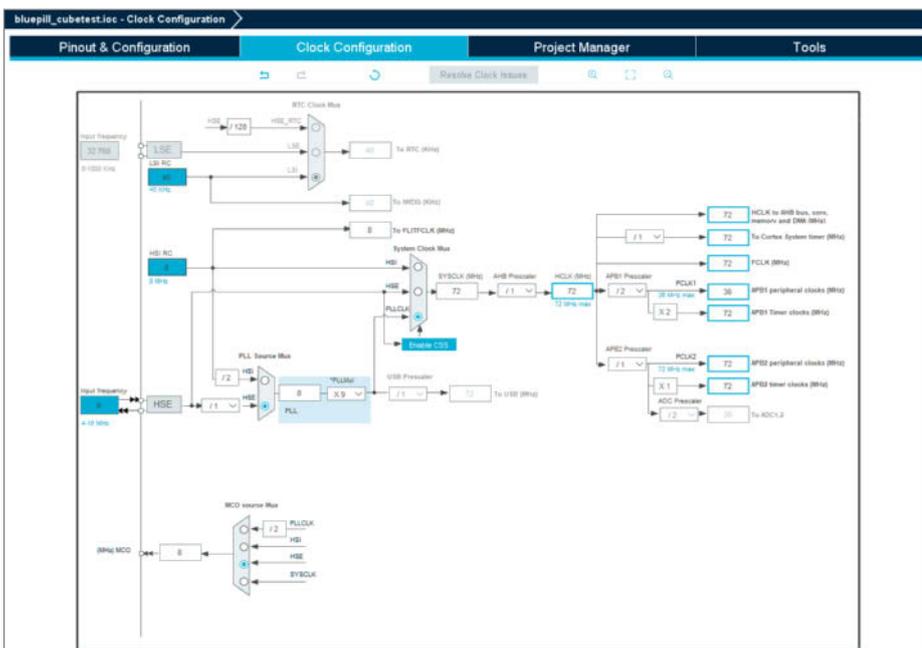
Klicken Sie auf Pin PC13 und weisen Sie in der nun aufklappenden Dropdown-Liste die Funktion *GPIO_Output* zu. Ähnlich verfahren Sie für Pin PD0 als *RCC_OSC_IN* und PD1 als *RCC_OSC_OUT*; diese Pins sind vom Taktgeber-Quarz belegt. Ändern Sie die bereits vorbelegten Pins PA13 und PA14 nicht, hier ist ja der Program/Debug-Adapter angeschlossen. Kleine Anmerkung: Im Prinzip dürfen auch diese Pins als I/Os benutzt werden, man verliert dann aber die Möglichkeit, die Firmware mit der IDE zu debuggen und muss zum nächsten Flashen mit den Boot-Jumpers (siehe oben) jonglieren. Stellen Sie nun noch den *RCC Mode HSE* auf *Crystal/Ceramic Resonator*; der 32-kHz-RTC-Oszillator *LSE* bleibt auf *Disable*.

Wechseln Sie nun auf den Reiter *Clock Configuration* 6; hier erkennt man, dass am HSE-Eingang nun 8MHz anliegen sollen (default, so lassen). Wichtigstes Feld ist das interne *HCLK*-Signal in der Mitte. Hier tragen Sie 72 für 72MHz ein. Die IDE stellt nun (nach Bestätigung, dass eine andere Taktquelle verwendet werden soll – ja, bitte!) PLL, Taktauswahl und Teiler so ein, dass die gewünschte Frequenz erreicht wird: Der Schalter *System Clock Mux* wechselt auf den Eingang *PLLCLK*, der PLL-Multiplikator *PLLMul* auf 9.

Auf Wunsch kann man nun die Taktteiler für Timer und I/Os (*ABP1* und *ABP2 Prescaler*) heraufsetzen, sodass diese (stromsparend) mit niedrigerem Takt laufen; für unser Beispiel-Programm ist das aber nicht nötig. Bei den Timern kann dies manchmal sinnvoll sein, um längere Zeiten zu ermöglichen. Ist alles wie auf dem Screenshot eingestellt, sichern Sie die *IOC*-Einstellungen mit *CTRL-S*. Die IDE passt nun einige der Quellen auf die neuen Voreinstellungen an.

Lampe an, Lampe aus

Um die LED nun endlich blinken zu lassen, müssen Sie das vom Projekt-Wizard vorgefertigte *main.c* nur noch um folgende zwei Zeilen ergänzen, und zwar im Abschnitt */ * USER CODE BEGIN 3 */* der Hauptschleife, die wie üblich mit *while(1)* beginnt:



6 Im Clock-Konfigurator wählt man die Taktquellen für Prozessorkern (SYSCLK) und RTC, gegebenenfalls den PLL-Multiplikator und die Teilfaktoren für die Peripheriebausteine.

```
while (1)
{
  /* USER CODE BEGIN 3 */
  HAL_GPIO_TogglePin(GPIOC,
    GPIO_PIN_13);
  HAL_Delay(200);
  // Delay für Blinken
  /* USER CODE END 3 */
}
```

Die automatisch erzeugte Clock-Initialisierung finden Sie in *main.c* in der Routine *SystemClock_Config(void)*, die I/O-Konfiguration in *gpio.c*. Ein Klick auf den grünen Run-Button oben kompiliert nun Ihr Elaborat und schickt es direkt an den ST-Link-Adapter; das oben erwähnte separate *ST-LINK Utility* ist dafür nicht notwendig. Die LED blinkt!

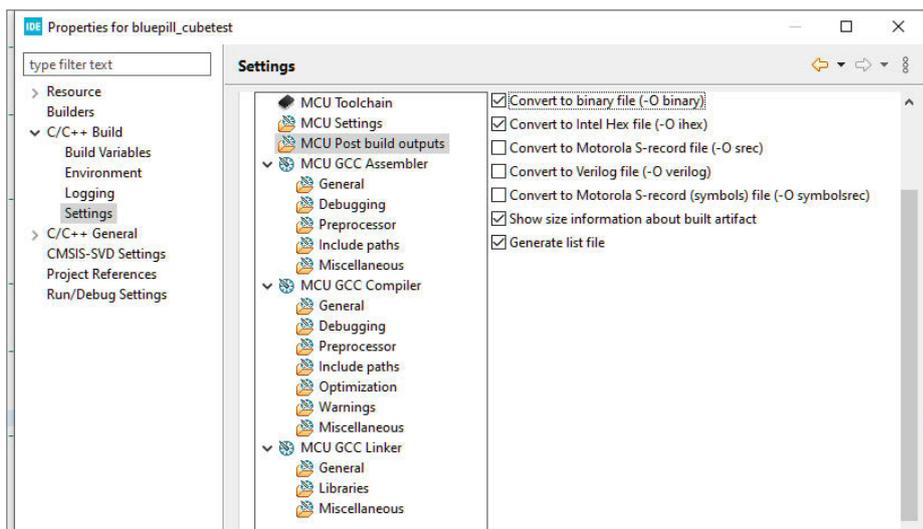
Eigentlich gibt es nun nichts zu debuggen, das Programm funktioniert ja wie erwartet. Probieren Sie es trotzdem einmal aus: Klicken Sie oben auf das grüne Käfer-Icon; die Ansicht wechselt nun in den Debug-Modus, die Firmware wird neu gestartet und Sie finden sich im Editor auf der Zeile *HAL_Init()* wieder – an diesem ersten Befehl der *main*-Routine wurde Ihr Programm angehalten. Mit F6 (*Step Over*) oder einem Klick auf den gelben Pfeil links oben können Sie sich nun von Befehlszeile zu Befehlszeile hangeln. F8 (*Resume*) setzt die normale Ausführung fort.

Nun wird es spannend: Klicken Sie, während die LED fleißig blinkt, neben *HAL_Delay(200)* in die hellblaue Breakpoint-Leiste links. Es erscheint hier ein Stecknadel-Symbol, und die Programmausführung stoppt an dieser Stelle. Sie können nun die Programmausführung wie eben erwähnt in Einzelschritten abarbeiten oder in normalem Tempo wieder aufnehmen. F5 (*Step Into*) springt, so vorhanden, in eine Subroutine, um dort die Einzelschritte fortzuführen, während F6 (*Step Over*) einen Funktionsaufruf als Ganzes auffasst.

Besonders interessant ist die Überwachung von Variablen: Mit einem Rechtsklick auf eine Variable fügen Sie diese über den Pop-up-Menüpunkt *Add Watch Expression...* der Variablen-Watchlist (rechte Seite) hinzu. Jedesmal, wenn Ihr Programm angehalten wurde (egal, ob über Breakpoint oder Einzelschritt-Ausführung), werden hier die aktuellen Werte aufgelistet – so lange sie im aktuellen Kontext erreicht werden können. Will heißen: Variablen einer Subroutine sind nur erreichbar, wenn die Programmausführung sich auch ebendort befindet.

Der Debugger bietet noch weitaus mehr Möglichkeiten, die den Rahmen dieses Artikels sprengen würden und an die man sich am besten durch Ausprobieren herantastet. Die IDE-Hilfefunktion steht Ihnen dabei unter dem Stichwort *Debug View* zur Seite.

Ist das Debuggen abgeschlossen, erstellt man mit *Project/Build all* (CTRL-B) die endgültige Produktions-Datei. In der Grundeinstellung



7 Damit die STM32CubeIDE auch Hex- und Binärfiles erzeugt, ist eine kleine Änderung in den Projekt-Properties nötig.

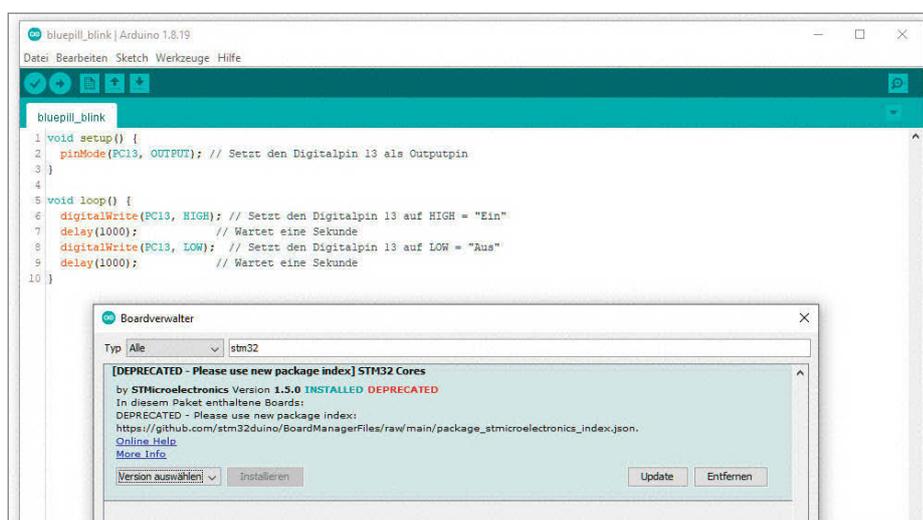
erzeugt die CubeIDE allerdings nur eine *.elf*-Datei für den CubeProgrammer; das alte *ST-LINK*-Utility möchte aber ein Intel-Hex- oder Binärfile. Wählen Sie in *Project/Properties* unter *MCU Post build outputs* zusätzlich die Checkboxen *Convert to binary file* und *Convert to Intel Hex file* an 7 und lassen Sie *Build all* nochmals laufen. Die Elaborate finden Sie dann im Unterverzeichnis *Release* Ihres Projekts.

Das entstandene Hex-File enthält keine *Debug-Hooks* (Einsprung-Informationen für den Debugger) mehr, das Programm wird deshalb (etwas bis deutlich) schneller ausgeführt als durch direktes Starten in der IDE. Sie können die Hex-Datei dann mit dem *ST-LINK Utility* laden und nach *Connect* auf den STM32F103 flashen. Beachten Sie, das das Tool den Speicherinhalt nicht automatisch aktualisiert! Nach Änderungen und einer Neukompilierung muss die Datei auch neu geöffnet werden.

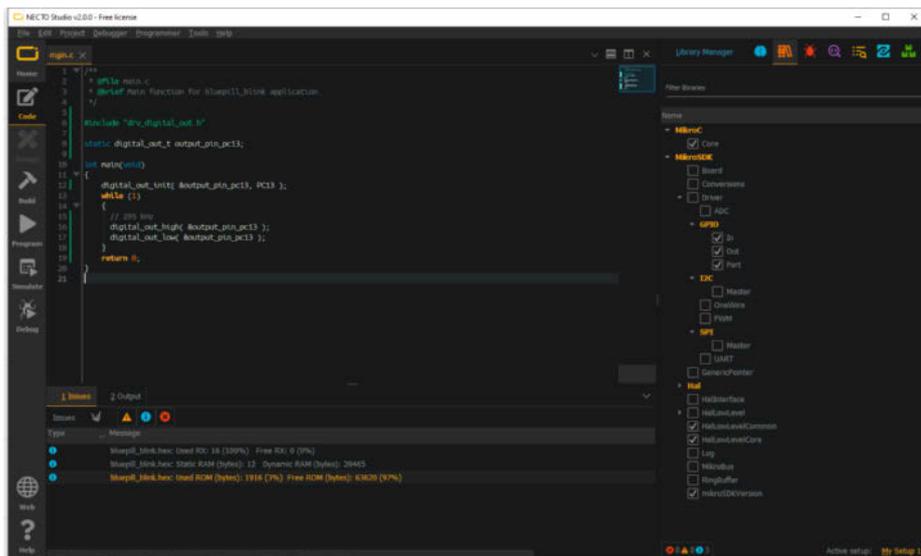
HAL oder nicht?

Wie die Arduino-IDE bieten die ST-Libraries die Möglichkeit, Zugriffe auf Hardware-Register zu abstrahieren. Wenn man in der Arduino-IDE zum Beispiel ein schlichtes *digitalWrite(13, LOW)* aufruft, setzt der Compiler im Maschinencode eine Reihe von Befehlen ein, die auf den jeweils eingesetzten Mikrocontroller abgestimmt sind. Das gleichwertige *HAL_GPIO_WritePin(GPIOC, GPIO_PIN_13, 0)* der CubeIDE macht es genauso; mit dem *Hardware Abstraction Layer* (HAL) programmiert man also hardware-übergreifend und muss den Code nicht bei einem Wechsel des Mikrocontrollers anpassen.

Nachteil: Der Umweg über eine HAL-Routine oder ein -Makro dauert länger als ein direkter Register-Zugriff. Mit folgender Schleife ergibt sich beispielsweise auf dem 72-MHz-



8 Installieren des STM32 Cores auf der Arduino-IDE: Beachten Sie, dass Versionen oberhalb von 1.5 nur noch den CubeProgrammer als Flash-Tool unterstützen.



9 Der NECTO-C-Compiler von MikroE unterstützt viele gängige Controller-Familien. An das dunkle Design werden sich Visual-Studio-Nutzer schnell gewöhnen.

ST32F103 eine Frequenz von 1,9MHz (836kHz im Debug-Modus) am Ausgangs-Pin PC13:

```
while (1)
{
    HAL_GPIO_WritePin(GPIOC,
        GPIO_PIN_13, 0);
    HAL_GPIO_WritePin(GPIOC,
        GPIO_PIN_13, 1);
}
```

Demgegenüber erreicht folgende Schleife, die direkt auf die Port-Register zugreift, rund 11MHz (4MHz im Debug-Modus):

```
while (1)
{
    GPIOC->ODR = 1 << 13;
    GPIOC->ODR = 0;
}
```

Ein erheblicher Unterschied also – der direkte Zugriff ist mehr als viermal schneller. Einen Haken hat die Sache allerdings: HAL_GPIO_WritePin() verändert isoliert nur ein Bit, während GPIOC->ODR = xx das gesamte Port-Register setzt, also 16 Bits gleichzeitig. Der Performance-Unterschied ergibt sich vorrangig aus der Tatsache, dass HAL_GPIO_WritePin() zunächst den alten Port-Zustand lesen muss, das gewünschte Bit ausmaskiert und dann neu setzt.

Der ST32F103 unterstützt (im Unterschied zu seinen größeren Brüdern) leider kein Bit Banding im I/O-Bereich, bei dem jedes einzelne Bit der I/O-Register über eine eigenständige Speicheradresse ansprechbar ist; das wäre dann gleichwertig zu einem HAL_GPIO_WritePin(), aber viel schneller. Es gibt allerdings die sogenannten Bit Set/Reset Register (BSRR),

die gezielt nur ausgewählte Bits eines Ports verändern:

```
while (1)
{
    // Set Bit 13
    GPIOC->BSRR = 1 << 13;
    // Reset Bit 13
    GPIOC->BSRR = 1 << (16 + 13);
}
```

Diese Routine blinkt nun mit satten 12MHz an PC13, ändert am Port C aber wirklich nur Bit 13 und lässt die anderen unangetastet. BSSR-Register sind mit 32 Bit doppelt so breit wie die normalen Port-Ausgangsregister ODR. Das Schreiben eines 1-Bits im unteren BSSR-Word setzt das Port-Bit, während das Schreiben eines 1-Bits im oberen BSSR-Word das Port-Bit löscht. 0-Bits bewirken dagegen gar nichts.

Im Zweifelsfall ist also abzuwägen, ob man die schlankere, schnellere Variante der direkten I/O-Zugriffe wählt oder die kompatible, bequemere HAL-Variante. Der Vollständigkeit halber sei erwähnt, dass es mit dem Projekt STM32duino sogar Bestrebungen gibt, die klassischen Arduino-Libraries auf der STM32CubeIDE einzusetzen, womit es dann bei einem digitalWrite(..) bliebe.

Sparta-Version

Wer unbedingt die Arduino-IDE weiter einsetzen und auf die luxuriösen Debug-Möglichkeiten verzichten möchte, kommt mit dem Nachinstallieren einer Boardverwalter-Erweiterung relativ einfach zu einem stabilen Entwicklungssystem. Tragen Sie zunächst https://github.com/stm32duino/BoardManagerFiles/main/STM32/package_stm_index.json als zusätzliche Boardverwalter-URL in die Voreinstellungen der Arduino-IDE ein und suchen Sie im Boardverwalter dann nach STM32. Installieren Sie die STM32 Cores in Version 1.5 – die neueren verlangen als Flash-Tool den STM32CubeProgrammer, der aber wie oben erwähnt mit vielen China-Nachbauten des ST-Link-Adapters nicht funktioniert 8. Wenn Sie einen originalen (oder mit dem STM32CubeProgrammer funktionierenden) ST-Link-Adapter verwenden, dürfen Sie natürlich auch die neueste Version (derzeit 1.9) installieren.

Wählen Sie sodann als Board den neuen Menüeintrag Generic STM32F1 series und als Board part number den Eintrag BluePiIII F103C8. Mit dem STM32 Core müssen Sie praktisch nicht umdenken und sich auch nicht um die umständliche Takt-Konfiguration kümmern; lediglich an die ausführlichen Pin-Bezeichnungen (zum Beispiel PC13 statt einfach 13) müssen Sie sich gewöhnen. Die kürzestmögliche Schleife sieht also so aus:

Clock-Konfiguration für STM32F103

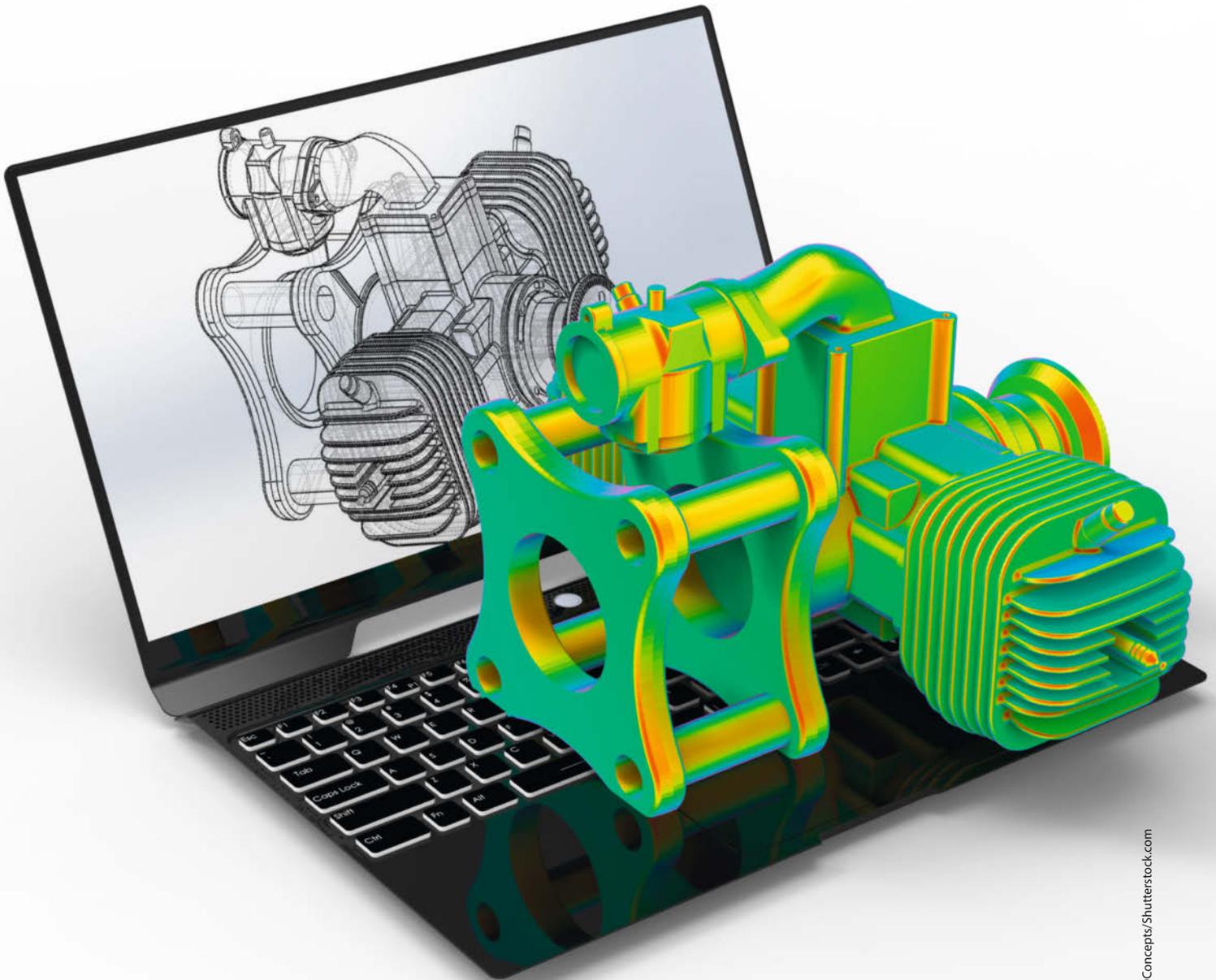
Project Setting	Eingestellter Wert
Internal high-speed clock enable	internal 8 MHz RC oscillator OFF
External high-speed clock enable	HSE oscillator ON
External high-speed clock bypass	HSE oscillator not bypassed
Clock security system enable (CSS)	Clock detector OFF
PLL enable	PLL ON
System clock switch	PLL selected as system clock
Division factor	SYSCLOCK not divided
APB1 low-speed prescaler	HCLK divided by 2
APB2 high-speed prescaler	HCLK not divided
ADC prescaler	PCLK2 divided by 4
PLL entry clock source	Clock from PREDIV1 as PLL input
PLL multiplication factor	PLL input clock × 9
USB prescaler	PLL clock is not divided
Microcontroller clock output	No Clock

ADC Prescaler und APB Prescaler können für stromsparenden Betrieb heraufgesetzt werden. Je nach Exemplarstreuung ist auch ein höherer PLL-Multiplikator möglich (etwa Faktor 12 für 96 MHz).

Gratis-3D-CAD für Maker

Wer in den 3D-Druck oder das CNC-Fräsen einsteigt, braucht über kurz oder lang eine Software, um die eigenen Ideen in präzise 3D-Dateien zu verwandeln. Da es davon erstaunlich viel kostenlos gibt, stehen Maker eigentlich vor einem Luxusproblem: Wie soll man das richtige Werkzeug wählen, wenn alle zu viel bieten, um jedes selbst auszuprobieren? Unser Tipp: weiterlesen! Dann finden Sie ganz sicher Ihr richtiges Gratis-3D-CAD-Programm.

von Peter König, Heinz Behling, Johannes Börnsen, André Burkhardtmaier,
Matthias Mett, Ralf Stoffels, Florian Stolz und Carsten Wartmann



Wie viel Planung man für seine Hobbyprojekte braucht, dazu dürfte es so viele Meinungen wie Maker geben – während die einen sich nur bei *Extreme Basteling* so richtig entspannen können, wollen andere in ihrer kostbaren Freizeit keine Umwege gehen und planen vorab jedes Detail minutiös (und genießen es). Aber spätestens, wenn digitale Werkzeuge wie 3D-Drucker, Fräsen oder Lasercutter ins Spiel kommen, geht es ohne Software für präzise Konstruktion einfach nicht mehr: Aus einer vagen Idee oder Skizze lässt sich nun mal kein *G-Code* für die Maschinensteuerung erzeugen, die KI dafür ist noch nicht erfunden.

Aber auch wer Möbel tischlert und die Teile dafür auf herkömmliche Weise auf der Kreissäge zuschneidet, profitiert davon, sein Projekt vorab in CAD-Software zur Probe zu bauen: Man bekommt nicht nur einen ersten Eindruck, wie das Ganze im fertigen Zustand aussehen wird. Die passende Software liefert außerdem auf Knopfdruck bemaßte Pläne zum Drucken oder auch Stücklisten für den Zuschnitt. Wer will, kann meist auch probieren, ob sich Türen, Klappen oder Schubkästen so nutzen lassen, wie gedacht, da passieren sonst auch gerne mal böse Überraschungen ...

Die gute Nachricht: Für erstaunlich viel leistungsfähige bis professionelle 3D-Software muss man nichts bezahlen. Zumindest nicht, wenn man keine kommerziellen Interessen mit seinen Projekten verfolgt. Der Wermutstropfen: Kaum ein Gratis-3D-Programm ist für alle Arten von Maker-Projekten ideal ausgestattet; alle haben Stärken und Schwächen, eignen sich für manche Vorhaben besonders gut und für andere gar nicht.

Praxiserprobt

Deshalb ist es vier Jahre nach unserem letzten Gratis-CAD-Überblick in der *Make*-Ausgabe 4/18 (S.90) dringend geboten, hier erneut eine Orientierungshilfe zu bieten. Allerdings machen wir heute einiges anders als damals: So konzentrieren wir uns diesmal auf CAD mit 3D-Fähigkeiten und lassen die reinen 2D-Zeichenprogramme außen vor (auch wenn man mit einigen der hier gezeigten 3D-Programme sehr gut in 2D konstruieren kann). Falls Sie aber zu den reinen 2D-CAD-Programmen mehr wissen wollen, schicken Sie uns eine Nachricht an mail@make-magazin.de; bei genügend Interesse liefern wir dazu einen eigenen Artikel nach.

Wir haben diesmal in den Reihen unserer Autoren und in der Redaktion für jedes interessante CAD-Programm einen Fürsprecher gefunden, der schon viel mit der betreffenden Software oder dem Webdienst gearbeitet hat, dadurch die jeweiligen Talente, Paradedisziplinen und Schwächen sehr gut kennt – und sich vielleicht sogar als Fan bezeichnen würde. Auf den folgenden Seiten finden Sie deshalb neun – durchaus persönlich gefärbte –

Kurzinfo

- » **Kostenlose Software für Eigenkonstruktionen vom 3D-Druck-Projekt über CNC und Möbelbau bis zur Nachbearbeitung von 3D-Scans**
- » **Open-Source-Programme, Gratis-Versionen professioneller Werkzeuge, Webdienste und lokal installierte Software**
- » **Im Vergleich: Blender, DesignSpark Mechanical, FreeCAD, Fusion 360, Onshape, OpenSCAD, SketchUp Make, Solid Edge Community Edition, Tinkercad**

Mehr zum Thema

- » Die ausführliche Liste mit weiteren Artikeln zum Thema gibt es über die Kurz-URL oder den QR-Code.

Alles zum Artikel
im Web unter
make-magazin.de/x3st



Kurzporträts kostenloser CAD-Werkzeuge, gefolgt von einer Vergleichstabelle und der Bewertung fünf spezieller *Use-Cases*. Links zu Downloads und Registrierungsseiten finden Sie über die URL in der Kurzinfo, ebenso zu erwähnten Videos und Kanälen sowie weiteren Artikeln von uns zum Thema.

Geschenke mit Haken

Alle im Folgenden vorgestellten Programme kann man (nach derzeitigem Stand) dauerhaft kostenlos nutzen. Bei der Open-Source-Software *Blender*, *FreeCAD* und *OpenSCAD* liegt das auf der Hand und bleibt auch ganz sicher so. Aber bei den übrigen sechs Angeboten in diesem Artikel handelt es sich durchweg um Produkte von Firmen mit Gewinnerzielungsabsicht, die keineswegs verpflichtet sind, ihre Programme oder Dienste bis in alle Ewigkeit an die Maker-Community zu verschenken.

Wie wir bei der Recherche für diesen Artikel festgestellt haben, ist es gerade bei den Gratis-Versionen der professionellen CAD-Schwergewichte manchmal gar nicht so leicht herauszufinden, ob und wo es das Angebot für Heimanwender (noch) gibt. Und natürlich könnten die Hersteller das auch sang- und klanglos einstellen. Bei Cloud-Werkzeugen wie *Fusion 360* und *Tinkercad* von *Autodesk* oder *Onshape* könnten im schlimmsten Fall dann sogar die eigenen Konstruktionen auf Nimmerwiedersehen in den Orkus verschwinden.

Der Ehrenrettung für die Industrie halber sei gesagt, dass dies in den letzten Jahren eher selten vorgekommen ist (siehe auch Editorial, Seite 3). Beschnitten wurde manches Angebot aber schon. So war *Autodesk* bei *Fusion 360* früher mal deutlich großzügiger: Die Funktionen waren praktisch nicht eingeschränkt und man durfte mit der Gratis-Version als Start-up sogar noch ganz ordentlich Geld verdienen, ohne einen Cent an den Hersteller zu bezahlen.

Von *SketchUp* gibt es seit Version 2017 keine neue *Make*-Variante mehr, denn die wurde ursprünglich unters Volk geworfen, als die Software Google gehörte. Der Konzern wollte auf diese Weise Freizeitarchitekten animieren, damit den 3D-Globus *Google Earth* mit Gebäudemodellen aufzupeppen. Die letzte *Make*-Version kann man zwar noch herunterladen, wenn man weiß, wo. Und auch die Installation funktioniert nach wie vor. Allerdings erfordert es mittlerweile einen Trick, um noch an Modelle aus der riesigen Online-3D-Galerie zu kommen, weil die auf eine neuere Version des Dateiformats umgestellt wurde (Video siehe Link in der Kurzinfo).

Deutsch oder Englisch

Wer nicht gerade eine Ausbildung als Bauzeichner oder ein Maschinenbau-Studium hinter sich hat, für den ist die Einarbeitung in eine komplexe 3D-CAD-Software ein ambitioniertes Vorhaben (und gerade deshalb möchte man eigentlich nicht riskieren, dass einem der Hersteller das mühsam erschlossene Werkzeug wieder aus der Hand reißt). Manchen fällt der Einstieg leichter, wenn der nicht auch noch auf Englisch passieren muss. Alle neun vorgestellten Tools gibt es wahlweise mit deutscher Bedienoberfläche und auch zu allen gibt es Tutorials auf Deutsch vom Hersteller oder Nutzern im Netz, als Video oder sogar in Buchform.

Das englischsprachige Angebot ist allerdings durch die Bank weg umfangreicher. Auch die User-Community, die man im Zweifelsfall um Hilfe fragen kann, vergrößert sich deutlich, wenn man seine Fragen auf Englisch stellt. Das ist auch der Grund, warum etwa bei *Blender* auch in deutschen Tutorials eigentlich immer die englische GUI verwendet wird – zumal viele Fachausdrücke aus der 3D-Welt auch hierzulande unübersetzt gebraucht werden, etwa *Mesh* oder *Subdivision Surface*.

Blender

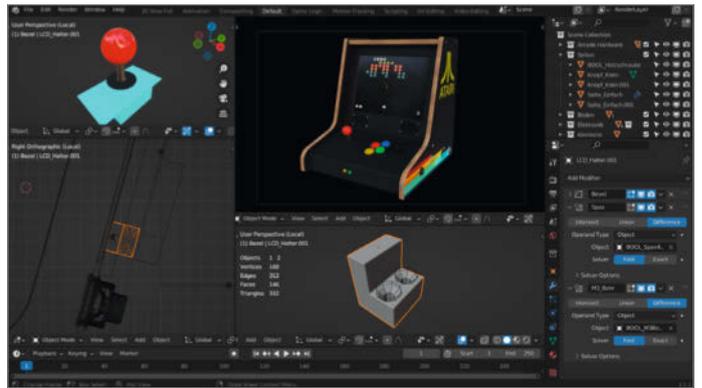
Ich bin ein *Blender*-Fachidiot: Ich arbeite seit über 20 Jahren für und mit Blender, daher tue ich mich mit allen anderen Konzepten von 3D-Konstruktion und Modellierung schwer. Bei Blender arbeite ich direkt im 3D-Raum mit mehreren Ansichten. Das Objekt baut sich vor meinen Augen auf und ich muss nicht von einer abstrakten 2D-Skizze umdenken. Blender ist anders, aber für visuell-dreidimensional denkende Maker prima geeignet. Es ist nicht nur kostenlos, sondern wird durch die GPL-Lizenz auch immer kostenlos und frei bleiben.

Aber Blender ist nicht für Gelegenheitsbenutzer gemacht, Assistenten und intelligente Fangoptionen sucht man vergebens. Einsteiger müssen erst mal einen hohen Lernaufwand in Kauf nehmen. In der Entwicklung von Blender wird allerdings sehr großer Wert auf eine nachhaltige Bedienung gelegt: Auch wenn ständig neue Funktionen hinzukommen, gehen die etablierten Arbeitsweisen und Bedienkonzepte nicht verloren; was man für eine Funktion gelernt hat, ist mit der nächsten konsistent und alle bereits gelernten Tastendrücke funktionieren analog. Die wichtigsten Werkzeuge werden mit der linken Hand an der Tastatur aufgerufen und die rechte Hand bleibt an

der Maus. Durch seine hohe Konfigurierbarkeit wird Blender zu einem maßgeschneiderten Arbeitswerkzeug, praktisch jeder Aspekt der GUI kann angepasst werden.

In der riesigen Auswahl von Add-Ons findet man oft auch eine Funktion, die einem einen großen Teil der Routinearbeit abnimmt. Profis programmieren in Python ganze Arbeitsabläufe und bauen parametrisierbare Normteile, die sich immer wieder verändern lassen. Dass Blender auf Windows, macOS und Linux absolut identisch funktioniert und aussieht, lieben Profis in Studios ebenso wie Heimanwender. Blender ist portabel, vom USB-Stick läuft er (ja, *der* Blender, alte Geschichte) ohne Installation praktisch auf jedem PC.

Die Grenze von Blender liegt darin, dass die Software bei der Modellierung ausschließlich mit Oberflächengittern (Meshes) arbeitet – einen



mathematisch sauberen Geometrie-Kernel wie bei Profi-CAD-Paketen gibt es nicht, was die präzise Konstruktion für subtraktive CNC-Verfahren wie Fräsen und Lasern erschwert. Klassische 3D-CAD-Formate liest oder schreibt Blender nicht. Dafür geht Blender mit sehr komplexen Mesh-Modellen, wie sie in 3D-Scans und bei hoch detaillierten Figuren vorkommen, souverän um. Und dass man seine Kreationen rendern, animieren und simulieren kann, ohne die Software zu wechseln, ist auch nicht zu verachten. —Carsten Wartmann (caw)

DesignSpark Mechanical

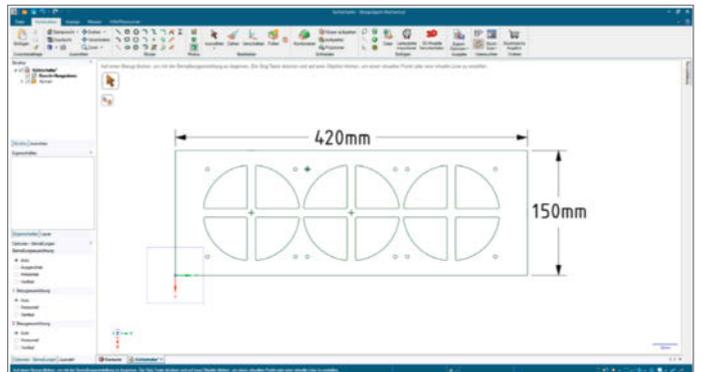
Bereits in der Version 5.0 liegt das nach Online-Registrierung (unter Angabe von Name, Wohnort und beruflicher Position) kostenlos nutzbare *DesignSpark Mechanical* von *RS Components* vor. Obwohl ein 3D-CAD-Programm, nutze ich es vor allem als 2D-Konstruktions-Software für Lasercut-Projekte. Das hat zwei Hauptgründe: die permanente und übersichtliche Kontrolle über sämtliche Maße der Konstruktion sowie die problemlose Ausgabe der Ergebnisse – nicht nur im eigenen RSDOC-Dateiformat, sondern auch als von wohl allen Lasercut-Programmen verwertbare DXF-Dateien (AutoCAD). Daneben beherrscht es nur die Dateiformate PDF, glTF und SKP (SketchUp) sowie die Bildformate JPG und PNG. Öffnen kann es DXF-Dateien leider nicht, dafür aber SketchUp-Files, STL und STEP-Dateien, letztere aber nur schreibgeschützt.

Zweidimensionale Konstruktionen werden aus Linien, Rechtecken, Kreisen, Vielecken und Kurven zusammengesetzt. Die Maße werden dabei entweder per Ziehen mit der Maus oder ganz genau per Werteingabe festgelegt. Die Konstruktion von 3D-Projekten ist zwar auch möglich, aber für meine Anwendung alles andere als komfortabel – insbesondere,

wenn es sich um Formen mit komplexeren Hohlräumen handelt. Denn im Wesentlichen werden dabei 2D-Objekte einfach in die dritte Dimension aufgezogen.

Als Hilfe stellt der Hersteller zwar eine Online-Bibliothek mit mehr als 75.000 fertigen 3D-Bauteilen. Die sind aber vor allem aus dem Bau-, Pneumatik- und Elektronikbereich und waren etwa bei Carsten Meyers *MaXYposi* (Make 1/17) hilfreich; ich kann sowas in meinen Projekten kaum verwenden. Da bleibt dann nur die händische Konstruktion über, und das ist mit anderen Programmen deutlich einfacher (ja, ich bin ein *Tinkercad*-Fan!). Design-Spark Mechanical kann durch einige wenige kostenpflichtige Module erweitert werden, mit denen sich beispielsweise Gewinde konstruieren oder Barcodes einfügen lassen.

Hilfreicher sind die Online-Hilfen in Text- und Videoform sowie einige Online-Tutorials, die sich vor allem an Einsteiger wenden und



daher leicht verständlich sind. Falls es trotzdem noch unlösbare Probleme gibt, dem hilft der direkte Link zum E-Mail-Support weiter. Allerdings erfolgt der nur in Englisch.

Einigen Funktionen merkt man die Entstehungsgeschichte des Anbieters *RS Components* an: Ursprünglich wurde die Firma als Lieferant von Radio- und anderen elektronischen Bauteilen gegründet. Das erklärt zum Beispiel das Sortiment in der Download-Bibliothek, aber auch die Möglichkeit, in der Software Leiterplatten zu importieren und anhand der Bestückungslisten Angebote für die Bauteile einzuholen. —Heinz Behling (hbg)

FreeCAD

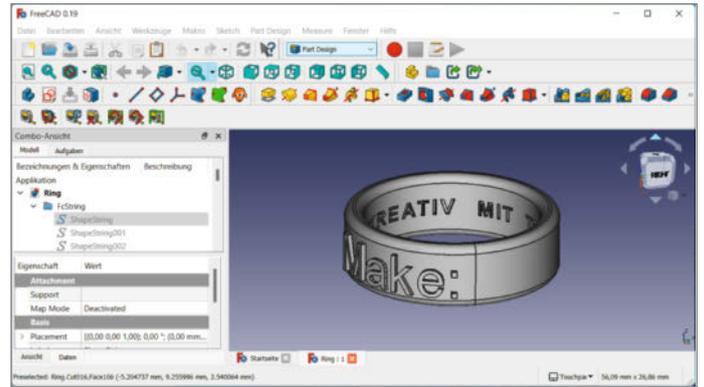
Mit dem kostenlosen Open Source-Programm *FreeCAD* gibt es eine klassische parametrische CAD-Software, die sich ideal für den Maker-Bereich eignet. Die Software ist verfügbar für die verschiedensten Betriebssysteme und hat sehr geringe Hardwareanforderungen. Mit *FreeCAD* konstruiert man direkt mit 3D-Objekten oder mit zweidimensionalen Skizzen, die sich durch Aufpolstern zu 3D-Objekten wandeln. Mit seinem Konzept der *Workbenches* enthält *FreeCAD* eine Reihe von vorgegebenen Arbeitsumgebungen, mit denen man verschiedene Aufgaben erledigen kann. So gibt es *Workbenches* für alle möglichen Einsatzgebiete wie das dreidimensionale Zeichnen oder die Erstellung von zweidimensionalen technischen Zeichnungen aus einem 3D-Objekt. Aber auch außergewöhnliche Einsatzzwecke wie Roboterprogrammierung oder Bahnberechnungen für CNC-Maschinen sind in der Standardinstallation von *FreeCAD* enthalten.

Durch zahlreiche Import- und Exportmöglichkeiten lässt sich *FreeCAD* als Multifunktions-tool verwenden. Falls mal ein Werkzeug benötigt wird, das nicht in *FreeCAD* enthalten ist, finden sich im Add-on-Manager eine große

Menge weiterer kostenloser Tools, die sich nachinstallieren lassen. Dort kann man beispielsweise Zusammenbau-Workbenches hinzufügen, mit denen sich 3D-Objekte zusammensetzen und animieren lassen. Außerdem ist in *FreeCAD* eine Python-Programmierungsumgebung enthalten, mit der man das Programm automatisieren kann.

Damit lassen sich auch Makros aufzeichnen, wie man es aus Office-Programmen kennt.

Da *FreeCAD* von einer Open-Source-Community programmiert wird, dauern Weiterentwicklungen manchmal etwas länger als bei professionellen Programmen. Die Benutzeroberfläche besteht noch aus einfachen Symbolleisten, wie man sie aus älteren Programmen kennt. Mit Add-ons kann man auch hier verschiedene Oberflächenelemente nachrüsten und diese an den eigenen Geschmack anpassen.



In *FreeCAD* lassen sich die meisten Konstruktionsprobleme lösen, wenn auch manchmal mit ein oder zwei Klicks mehr als bei kommerziellen Programmen. Wenn man sich bereits mit CAD-Programmen auskennt, findet man sich hier schnell zurecht. Anfänger können sich gut einarbeiten. Für die eigene Werkstatt bringt das Programm viele Möglichkeiten mit, weshalb die *Make* seit etwa zwei Jahren diverse Tutorial-Artikel zu *FreeCAD* veröffentlicht hat (siehe Link). —*Matthias Mett*



nobufil

PREMIUM RECYCLED MATERIALS
www.nobufil.com



Filamente und Spulen aus recycelten Polymeren.



Einfache Verarbeitung auf unterschiedlichen Druckern.



Ausgiebig getestet für eine hervorragende Qualität.



Hergestellt in der EU aus lokal beschafften Materialien.



Fusion 360

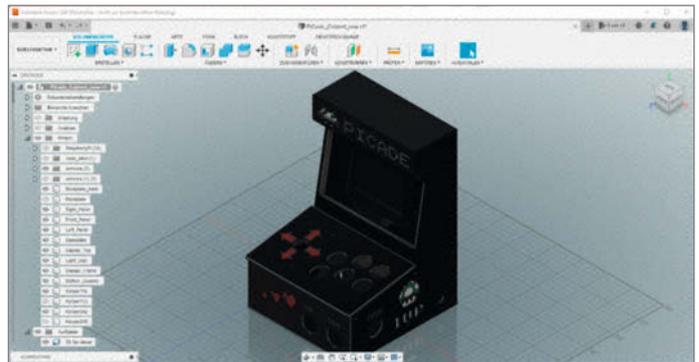
Zwischen dem kostenlosen *Tinkercad* aus eigenem Hause – was aber recht wenig mit einem professionellen CAD-Programm zu tun hat – und Profi-Software, die schnell mal einen vierstelligen Betrag kostet, möchte Autodesk die Lücke mit *Fusion 360* und dessen flexiblem Abo-Modell schließen. Dabei bietet der Hersteller Privat Anwendern wie Makern und anderen Hobbykonstrukteuren ohne kommerzielle Ziele das umfangreiche CAD-Programm sogar ganz ohne Kosten an. Die Gratis-Lizenz muss man lediglich alle drei Jahre verlängern. Der Haken? Natürlich wird hier der Funktionsumfang im Vergleich zu den teuren Abos leider etwas eingeschränkt, was für den Einstieg in die CAD-Welt aber gar nicht so stark ins Gewicht fällt.

Wer *Fusion 360* zum ersten Mal startet, wird überrascht sein, weil man nicht gleich mit allerrhand Werkzeugen und Funktionen erschlagen wird, was den Einstieg sehr angenehm macht. Durch die große Community rund um *Fusion* lassen sich viele Tutorials oder Einsteiger-Videos finden, mit deren Hilfe man in relativ kurzer Zeit schon ein recht komplexes Teil selbst konstruieren kann. Für eine schöne Aufbereitung des eigenen Projekts bietet *Fusion 360* auch eine

Rendering-Funktion an, die das konstruierte Teil attraktiv in Szene setzt. Allerdings benötigt man hierfür in der kostenfreien Version einen leistungsstarken Rechner – gerendert wird in diesem Fall lokal, im teuren Abo kann hingegen der Clouddienst von Autodesk diesen Job übernehmen.

Apropos: *Fusion 360* ist ein cloudbasiertes Produkt, das eine Internetverbindung voraussetzt. Offline zu arbeiten ist zwar eingeschränkt möglich, jedoch liegen sämtliche Projektdaten auf dem Cloudspeicher von Autodesk, was bei einigen verständlicherweise etwas Bauchweh auslöst. Wer damit aber kein Problem hat, ist mit *Fusion 360* super bedient und kann die Funktionen eines professionellen CAD-Programms für seine eigenen Projekte komplett kostenfrei nutzen.

Um zum Schluss das fertige Modell weiterzuverarbeiten bietet *Fusion 360* viele ver-



schiedene Exportformate (etwa STL, OBJ oder DXF) und Tools an, um daraus ein 3D-druckbares Modell zu erstellen oder die Daten für eine Maker-typische 3-Achsen-CNC-Fräse vorzubereiten.

Für mich persönlich ist *Fusion 360* ein tolles Tool um auf einfachste Weise auch mal komplexere Teile zu konstruieren und für den 3D-Drucker vorzubereiten. Durch den Cloudspeicher habe ich meine Projekte zudem überall griffbereit, um daran weiterzuarbeiten. —André Burkhardtmaier

Onshape

Im Gegensatz zu vielen anderen der hier vorgestellten CAD-Programme handelt es sich bei *Onshape* um eine Software, die zu hundert Prozent im Browser läuft. Rechenaufwändige Operationen werden auf den Servern ausgeführt, sodass die Anforderungen an den verwendeten Rechner moderat bleiben. Es gibt *Onshape* auch als Apps für iOS und Android. Auch hier übernimmt das Endgerät nur die Anzeigefunktion.

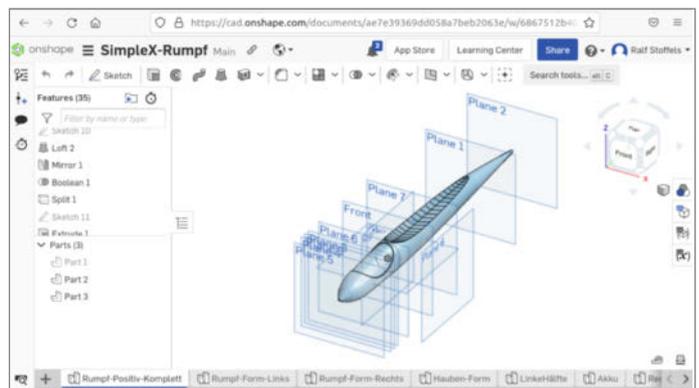
Die hier vorgestellte freie Version unterscheidet sich von der kommerziellen Standardversion (Basislizenz für 1500 Euro pro Jahr) im Wesentlichen nur dadurch, dass alle Konstruktionen frei für jeden zugänglich sind. Eine Verwendung für kommerzielle Projekte schließt sich somit aus und ist bei der freien Lizenz auch nicht erlaubt.

Für den Hobbyanwender ermöglicht dieses Lizenzmodell den Zugang zu einer professionellen parametrischen CAD-Software, die auf Windows, Mac und Linux problemlos läuft. Auch ein fliegender Wechsel zwischen diesen Systemen funktioniert einwandfrei, da die Dokumente online gespeichert sind.

Insbesondere als Nutzer dieses freien Zugangs muss man sich natürlich bewusst sein, dass die Firma *PTC* jederzeit die Spielregeln

ändern kann – von einer Einschränkung der Funktionalität bis hin zur kompletten Sperrung des freien Zugangs. Ich konstruiere dennoch seit einigen Jahren mit *Onshape* und erfreue mich des intuitiven Bedienkonzepts und den mächtigen Freiformfunktionen – und lebe mit dem Damoklesschwert, dass über mir hängt.

Onshape ist eine parametrische CAD-Software, die die Konstruktionsschritte in einer Liste darstellt. Nimmt man eine Änderung in einem der Schritte vor, dann werden alle folgenden Schritte, sofern sie eine Abhängigkeit haben, entsprechend neu berechnet. Üblicherweise definiert man Objekte über eine zweidimensionale Skizze, aus der man die 3D-Objekte extrudiert. Durch boolesche Operationen erzeugt man Ausschnitte und Löcher, sofern sie nicht schon in der Ausgangsskizze gezeichnet sind.



Mir gefallen an *Onshape* insbesondere die Funktionen zur Erzeugung eines komplexen 3D Objektes entlang einer vorgegebenen Kurve. So importiere ich zum Beispiel zwei Tragflügelprofile einer Modellflugzeugtragfläche als DXF und erzeuge eine 3D-Form der Tragfläche als Übergang vom ersten zum zweiten Profil entsprechend der gewünschten Tragflügelgeometrie. So entstehen bei mir Negativformen von GFK-Rümpfen und Tragflächen, die ich dann mit dem 3D-Drucker herstelle. —Ralf Stoffels

OpenSCAD

Es lässt sich trefflich darüber streiten, ob *OpenSCAD* überhaupt in die Kategorie der CAD-Programme aufgenommen werden sollte, da es nicht wie üblich eine interaktive grafische Darstellung der Konstruktion hat, sondern dem Anwender eine angepasste Programmiersprache zur Definition der Objekte an die Hand gibt. Das Resultat der programmatischen Konstruktion ist dabei aber jederzeit in einer frei drehbaren 3D-Darstellung auf dem Bildschirm überprüfbar.

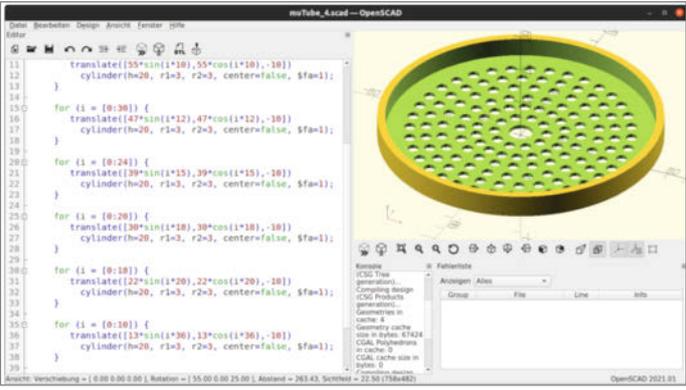
Ich breche hier eine Lanze für *OpenSCAD*, weil es trotz dieser ungewöhnlichen Eingabeschnittstelle viele Eigenschaften hat, die man von professioneller parametrischer Konstruktionssoftware kennt und sogar manches darüber hinaus. So können alle Konstruktionsparameter als Variable ausgedrückt und Abhängigkeiten einfach beschrieben werden. Berechnungen wie $L=L_1+L_2$ sind kein Problem und helfen den Überblick zu bewahren. Natürlich sind – wie in jeder Programmiersprache – Kommentare erlaubt (und erwünscht), die helfen, die Konstruktionschritte zu dokumentieren.

Bauteile lassen sich in einzelne Module aufteilen, denen man Parameter übergeben kann,

so wie man es bei Funktionen anderer Programmiersprachen gewöhnt ist. Somit stehen gute Werkzeuge zur Vereinfachung und Modularisierung zur Verfügung. Aufgrund dieser Eigenschaft finden sich viele frei parametrisierbare Modelle im Netz wie Zahnräder und Gewinde, die sich auf den jeweiligen Anwendungsfall anpassen lassen. Eine Suche nach „*OpenSCAD*“ auf *Thingiverse* vermittelt hierzu einen guten Eindruck.

OpenSCAD stößt dort an seine Grenzen, wo man mal schnell eine Kante ausrunden möchte oder eine Freiformfläche aus Schnittzeichnungen erstellen will. Man kann in der grafischen Darstellung keine Objekte markieren und dann bearbeiten, wie man es von anderer CAD-Software kennt.

Dafür spielt es seine Stärke aus, wenn sich Strukturen in einer Konstruktion oft wieder-



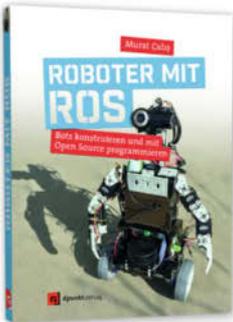
holen und sich sogar berechnen lassen, wie die vielen Löcher im Lautsprecherdeckel aus einem meiner Make-Projekte (siehe Screenshot). Hier habe ich nicht nur die Möglichkeit von `for()`-Schleifen genutzt, sondern die Positionen der Löcher mit Winkelfunktionen berechnet, was nicht unbedingt zum Standardumfang anderer CAD-Software gehört. Wer also eine Affinität zum Programmieren besitzt und einfache Objekte entwerfen möchte, wird *OpenSCAD* bald schätzen. —*Ralf Stoffels*



Erober dir die Welt der 3D-Modellierung!

Autodesk Fusion 360 ist deine virtuelle Vorstufe für den 3D-Drucker oder die CNC-Fräse. Du verwandelst Ideen in Modelle, prüfst und korrigierst sie und lässt sie schließlich Wirklichkeit werden. Dieses Buch zeigt dir in praxisnahen und unterhaltsamen Tutorials, wie du mit Fusion 360 massive, hohle, natürlich-organische und parametrisch angelegte Projekte umsetzt.

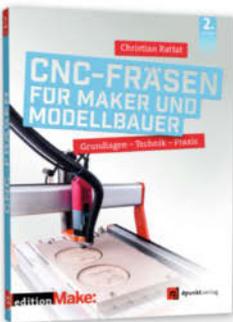
2. Auflage 2022 · 382 Seiten · 34,90 €
ISBN 978-3-86490-866-8



308 Seiten · 34,90 €
ISBN 978-3-86490-567-4



296 Seiten · 36,90 €
ISBN 978-3-86490-545-2



314 Seiten · 32,90 €
ISBN 978-3-86490-752-4

© Copyright by Maker Media GmbH.



dpunkt.verlag

Tinkercad

Ganz gleich, ob motivierter Schülerpraktikant oder gestandene Kollegin aus einer 3D-fernen Abteilung bei Heise – wer in der Make-Redaktion auftaucht und gerne mal „irgendwas mit 3D-Druck“ probieren will, wird von uns als erstes zu *Tinkercad* geschickt. Denn niederschwelliger geht der Einstieg ins Konstruieren in 3D kaum: Die bunte Bauklötzchenwelt läuft komplett im Browser, ohne spezielle Anforderungen an Hardware-Ausstattung und Betriebssystem. Beim Konstruieren bedient man sich per Drag&Drop einer Palette von bunten Grundkörpern wie Quadrern, Zylinder, Kugeln, Buchstaben und vieles mehr, die sich präzise auf die Wunschmaße bringen und frei im Raum platzieren und drehen lassen.

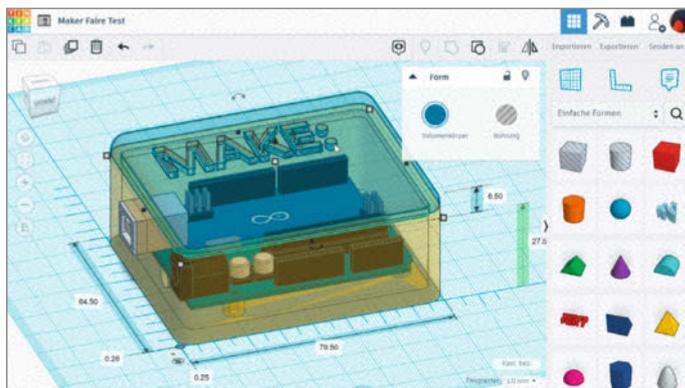
Die einzelnen Objekte definiert man entweder als *Volumenkörper* oder *Bohrung* und kann dann mehrere miteinander kombinieren – auf diese Weise bietet Tinkercad die beiden elementarsten booleschen Operationen *Ver-einigung* und *Differenz*, wodurch sich aus den wenigen Grundelementen erstaunlich komplexe Konstruktionen erzeugen lassen, etwa die *Super8*-Tauschkassette auf Seite 42. Gleichzeitig stellt Tinkercad stets sicher, dass die Topologie der Konstruktion stimmt: Was man aus diesem Webdienst hinterher als STL- oder

OBJ-Datei exportiert, hat immer eine waserdichte Oberfläche, keine doppelten Kanten oder verdrehte Flächennormalen.

Das macht Tinkercad zur idealen Spielwiese für den Entwurf von 3D-Druck-Objekten. Möbel hingegen würde ich damit nicht planen wollen, aber das sehen andere anders (siehe Seite 10).

Und obwohl ich echter Tinkercad-Fan der ersten Stunde bin und den Webdienst schon 2011 in der *c't* kurz vorgestellt habe, greife ich in den letzten Jahren aber doch meist zu *Blender*. Die Einfachheit von Tinkercad bringt dann eben doch auch Einschränkungen mit sich, so kann man darin nicht gezielt eine scharfe Objektkante auswählen und abrunden.

Dass meine 3D-Modelle zwingend auf Servern von Autodesk liegen, stört mich hingegen nicht – was ich damit konstruiere, ist keinesfalls patentverdächtig oder geheim und



ich kann außerdem entscheiden, welche Projekte ich öffentlich zugänglich mache und welche ich für mich behalte. Nur einmal habe ich geflucht, dass Tinkercad ein Webdienst ist: Auf der *Maker Faire Hannover 2018* plante ich, live im Vortragsprogramm zu zeigen, wie einfach man damit ein Arduino-Gehäuse bauen kann. Am Vorabend wollte ich das alles nochmal durchspielen und sah dann im Browser nur den dünnen Hinweis, man sei für 24 Stunden im Wartungsmodus ... die dann glücklicherweise am nächsten Morgen vorbei waren.
—Peter König (pek)

Meister aller Klassen

Na, haben Sie schon Ihren Favoriten gefunden? Oder können Sie sich (noch) nicht entscheiden? Kein Problem, nach diesen Einzelvorstellungen nehmen wir im Folgenden *alle Neune* nochmal mit einer anderen Fragestellung unter die Lupe – nämlich, wie sie sich für fünf *Use-Cases* eignen, die typisch für Projekte aus dem Maker-Alltag sind. In der Tabelle auf der folgenden Doppelseite haben wir dabei konkrete Noten vergeben, in den drei Stufen ⊕ *gut*, ○ *mittel* und ⊖ *schlecht*.

Konstruktion von Grund auf

Mit diesem Use-Case ist gemeint, was die Software bietet, wenn man von null an ein passendes Bauteil für sein Projekt konstruiert, etwa ein maßgeschneidertes Verbindungsstück mit Winkeln, Zungen, Schraubenlöchern und Steckverbindungen oder auch ein komplettes Gehäuse mit Öffnungen, Ausschnitten und Montagehülsen.

Für unseren Test ist dies das Pflichtprogramm und mit allen neun Kandidaten kommt man hier zum Ziel, wenn auch manchmal auf die etwas eigenwillige Weise, wie das jeweilige Programm funktioniert: Bei *Blender* ist man stets am *Mesh* (dem Polygonnetz der 3D-Oberfläche) dran, bei *SketchUp* zeichnet ein virtueller Bleistift Linien in den Raum und die

Software füllt geschlossene Polygonzüge automatisch mit Flächen. Einzig *OpenSCAD* und *Tinkercad* fallen hier gegenüber dem restlichen Feld etwas ab, allein deshalb, weil sich deren bewusste Reduktion der Werkzeuge im Alltag durchaus bemerkbar macht.

Konstruktion für 3D-Druck

Manche 3D-Software bietet speziell für den 3D-Druck noch Werkzeuge, um etwa Eigenschaften des Oberflächennetzes wie *manifold* oder *watertight* zu prüfen oder nach verdrehten Flächennormalen zu fahnden. Außerdem sollte das Tool für diesen Einsatz die einschlägigen Exportformate beherrschen, um für 3D-Druck gewappnet zu sein. Für die Benotung dieses Use-Cases (wie auch beim folgenden CNC-Szenario) gilt: *Gut* bewertet wird, wer den 3D-Druck aktiv unterstützt, *mittel*, wer ihm zumindest keine Steine in den Weg legt, etwa durch ungeeignete Geometrie oder Dateiformate – das wäre dann *schlecht*.

Die gute Nachricht: Wer Futter für seinen 3D-Drucker konstruieren will, bekommt zumindest von keinem Kandidaten Knüppel zwischen die Beine geworfen. Besonders gut schneiden hier *Solid Edge* mit einem eigenen 3D-Druck-Arbeitsbereich und *Blender* mit seinem speziellen Add-on *3D Printing Toolbox* ab. *Tinkercad* punktet hier schlicht durch die Tatsache, dass

alle Konstruktionen daraus garantiert als fehlerfreie Meshes exportiert werden, ohne dass man sich um Volumen und Normalen kümmern muss wie etwa bei *SketchUp*, bei dem es Prüfwerkzeuge dafür als Extension gibt. *FreeCAD* und *Fusion 360* erlauben beim STL-Export, selbst die Mesh-Dichte einzustellen.

Konstruktion für CNC

Wer für die Fräse konstruiert, braucht einen Export seiner 3D-Daten in einer geeigneten Form für die CAM-Stufe (**C**omputer **A**ided **M**anufacturing), die Werkzeugbahn-Aufbereitung vor dem Fräsen. Manche CAD-Software bringt praktischerweise bereits ein eigenes CAM-Tool mit oder es lässt sich als Plug-in nachrüsten.

Allerdings nicht bei vielen: *FreeCAD* bringt einen eigenen Arbeitsbereich für die Erzeugung von Werkzeugpfaden mit, *Fusion 360* hat ein eigenes CAM-Modul, wobei das in der kostenlosen Version gegenüber jenem für zahlende Kunden beschnitten ist (Details siehe Link in der Kurzinfor). Bei *Solid Edge* ist ausgerechnet der CAM-Bereich in der *Community Edition* nicht enthalten, immerhin exportiert die Software – wie auch *Onshape* – 3D-Modelle in geeigneten Dateiformaten, um damit in separater CAM-Software was anfangen zu können. Bei den übrigen Kandidaten ist nicht mal das der Fall,

weil deren Export-Dateien bereits Polygone statt gekrümmter Flächen enthalten.

Konstruktion für klassische Fertigung

Sei es der eingangs zitierte Möbelbau oder der Fertigungsauftrag an einen Dienstleister – manchmal braucht man ganz *old school* digitale oder gedruckte technische Zeichnungen und Stücklisten. Schließlich kann kaum ein 3D-Drucker im Bastelkeller Schrankwände drucken oder Teile aus Edelstahl formen; trotzdem wollen manche auch solche Projekte präzise am Rechner planen.

FreeCAD bietet hier unter anderem *Workbenches* für die Ableitung technischer Zeichnungen aus einem 3D-Modell (*TechDraw*), für den Zusammenbau und die Animation von Objekten (*Assembly-Workbenches* als Add-on) sowie ein Tabellentool für Stücklisten (*Spreadsheet*). *SketchUp* ist vor allem in Kombination mit dem Plug-in *OpenCutlist* eine große Hilfe für Möbelbauer, das wohl nur von spezieller

Tischlersoftware noch übertroffen wird. Mit *SketchUp* kann man faktisch maßstabgerechte Zeichnungen erstellen und Türen und Deckel auf Funktion prüfen, auch wenn das der Hersteller nicht als Funktion vorgesehen hat.

Ebenfalls gut geeignet für diesen Use-Case sind *Onshape* und *Solid Edge*, wobei dessen *Community Edition* in technische Zeichnungen ein Wasserzeichen einfügt. Bei *Fusion 360* gibt es in der Gratisversion hier mehr Einschränkungen, denn 2D-Ansichten kann man nur als Einzelblatt drucken und nicht als DXF exportieren und für detaillierte Bemaßungen braucht es ein bezahltes Abo. *DesignSpark Mechanical* schließlich bietet eine Stücklistenfunktion, die sich aber auf Bauteile aus dem integrierten Online-Katalog bezieht.

Nachbearbeitung

Oft muss man das Rad nicht neu erfinden (oder konstruieren), es reicht, etwas Passendes aus dem Netz bei *Thingiverse* oder *Printables* herunterzuladen, das eine oder andere Maß an-

zupassen, ein zusätzliches Loch zu bohren (in der Datei, nicht am realen Objekt!) oder auch die Oberfläche eines 3D-Scans zu glätten. Außerdem ist für dieses Szenario wichtig, welche verbreiteten 3D-Dateiformate ein Tool so öffnen oder importieren kann, dass man an die einzelnen Elemente herankommt. Die Bewertung der Tabelle bezieht sich auf beides; kann ein Programm das eine *gut* und das andere *schlecht*, kommt ein *mittel* heraus.

Da viele Gratis-Ausgaben kommerzieller Programme und Dienste gerade bei den Dateiformaten eingeschränkt sind, bieten sich als „CAD-Konverter“ vor allem *FreeCAD* auf der Open-Source-Seite und *Onshape* sowie *Solid Edge* auf der Industrieseite an. Wer vor allem STL-Dateien für den eigenen 3D-Druck anpassen will, hat natürlich viel mehr Auswahl beim Werkzeug, denn alle Tools im Test holen solche Objekte an Bord.

Blender ist bei der Bearbeitung von *Meshes* – seien es STL-Dateien oder detaillierte 3D-Scans – in seinem Element; zur Abwertung für diesen Use-Case führte nur die fehlenden Im-

Gratis-3D-CAD für Maker

Produkt	Blender	DesignSpark Mechanical	FreeCAD	Fusion 360
Version ¹	3.2.1	5.0 (2020 R2)	0.20	2.0.13377
Anbieter/Hersteller	Blender Foundation	RS Components/Space-Claim (Ansys)	FreeCAD-Team	Autodesk
Typ	Lokale installierte Open-Source-Software	Lokal installierte Software, erfordert Registrierung bei Installation und alle 30 Tage Internetverbindung	Lokale installierte Open-Source-Software	Clouddienst mit lokalem Client, Gratis-Lizenz für 3 Jahre, verlängerbar
Plattform	Windows/macOS/Linux	Windows	Windows/macOS/Linux	Windows/macOS
Preis Vollversion	–	–	–	ab 396 €/Jahr
Einschränkungen zur Vollversion ²	–	–	–	Kein kommerzieller Einsatz, weniger Dateiformate, nur 10 aktive/bearbeitbare Projekte ³ , CAM eingeschränkt, Rendering nur lokal, kein Teamwork
3D-Importformate	DAE, ABC, STL, FBX, glTF, OBJ, X3D	RSDOC, OBJ, STL, SKP, STEP (schreibgeschützt)	u.a. ² DAE, STL, OBJ, STEP, IGES	123DX, 3MF, F3D, F3Z, FBX, IGE, IGES, IGS, OBJ, STE, STEP, STP, STL
2D-Importformate ⁵	SVG, DXF ⁴	–	u.a. ² DXF, DWG, SVG	DXF
3D-Exportformate	DAE, ABC, STL, FBX, glTF, OBJ, X3D	RSDOC, glTF, STL, 3D-PDF	u.a. ² DAE, STL, OBJ, STEP, IGES	3MF, F3D, F3Z, FBX, IAM, IPT, OBJ, SKP, SMT, STEP, STP, STL
2D-Exportformate ⁵	SVG, DXF ⁴	DXF	u.a. ² DXF, DWG, SVG	–
Erweiterungen	Sehr viele Add-ons (gratis oder kostenpflichtig)	Kostenpflichtige Erweiterungen für Dateiformate und Zeichenfunktionen	Add-on-Manager für kostenlose Makros und Arbeitsbereiche	Plug-ins aus dem Autodesk App Store, teils gratis, teils kostenpflichtig, Dateiformate nicht erweiterbar
Sprache Skripte /Add-ons	Python, C++	–	Python	–
Use-Cases				
Konstruktion von Grund auf	⊕	⊕	⊕	⊕
Konstruktion für 3D-Druck	⊕	○	○	○
Konstruktion für CNC	⊖	⊖	⊕	⊕
Konstruktion für klass. Fertigung	⊖	○	⊕	⊖
Nachbearb. 3D-Dateien/Scans	○	⊖	⊕	⊖

¹ bei Redaktionsschluss ² Auswahl, vollständige Liste siehe Link in der Kurzinfo ³ unbegrenzt inaktive ⁴ mit Plug-in ⁵ ohne Rastergrafiken

port- und Exportfunktionen für CAD-Formate. *FreeCAD* wandelt STL-Importe auf Wunsch in bearbeitbare Objekte um, für 3D-Scans gibt es mit *Surface* und *Points* wieder eigene Workbenches für Oberflächen und 3D-Punktwolken. *SolidEdge*, *Fusion 360* und *SketchUp* importieren allesamt STL-Dateien und erlauben deren Bearbeitung – aber wenn man nur mal eben schnell ein neues Bohrloch setzen und ein anderes füllen will, ist *Tinkercad* oft die einfachere Wahl, wobei es hierbei eine harte Obergrenze von 25MB pro Datei für den Import gibt und der Webdienst in der Vergangenheit beim Import vereinzelt das Mesh auf eigene Faust vereinfacht hat. *Onshape* und *OpenSCAD* importieren Meshes zwar, bieten für deren Bearbeitung allerdings keine Werkzeuge an.

Die Qual der Wahl

Immer noch unentschieden? Dann hilft vielleicht das: „Wer noch nicht genau weiß, wohin die persönliche Reise als Hobby-3D-Konstrukteur geht, liegt bei seinen ersten Schritten mit

Tinkercad eigentlich immer richtig“ – das ist ein wörtliches Zitat aus dem Artikel *Gratis-CAD für Maker* aus der *Make* 4/18 und ist bis heute absolut wahr, jedenfalls wenn in erster Linie 3D-Druck-Projekte auf der ToDo-Liste stehen. Ebenso einfach ist die Auswahl für Leute, die genau wissen, was sie haben oder nicht haben wollen: Streichen Sie in der Tabelle einfach alle Produkte aus, die Ihre K.O.-Kriterien verletzen und schauen Sie, was übrig bleibt. So bleibt für Linuxer mit Cloud-Allergie als parametrische Konstruktionssoftware nur *FreeCAD*. Es gibt schlimmere Schicksale.

Wer keine Killerkriterien kennt, kann sich auch von der Frage leiten lassen, ob man sich lieber in *eine* Software ganz tief einarbeiten will, um sie für alle Konstruktionsaufgaben zu nutzen, oder bereit ist, für unterschiedliche Aufgaben mehrere Programme zu lernen, zu dem Preis, dass man im Kopf zwischen verschiedenen Bedienkonzepten umschalten muss. Rundum weitgehend gerüstet ist man mit den vier großen CAD-Paketen *FreeCAD*, *Fusion 360*, *Onshape* und *Solid Edge Commu-*

nity Edition, wobei auch die alle ihre blinden Flecken haben. So kann man mit *Onshape* keine reinen 2D-Konstruktionen entwickeln und muss bei *FreeCAD* damit leben, dass das Entwicklungstempo, die Ergonomie und der Support eben nicht den professionellen Ansprüchen genügen, die kommerzielle Anbieter ihren Kunden bieten können (und müssen).

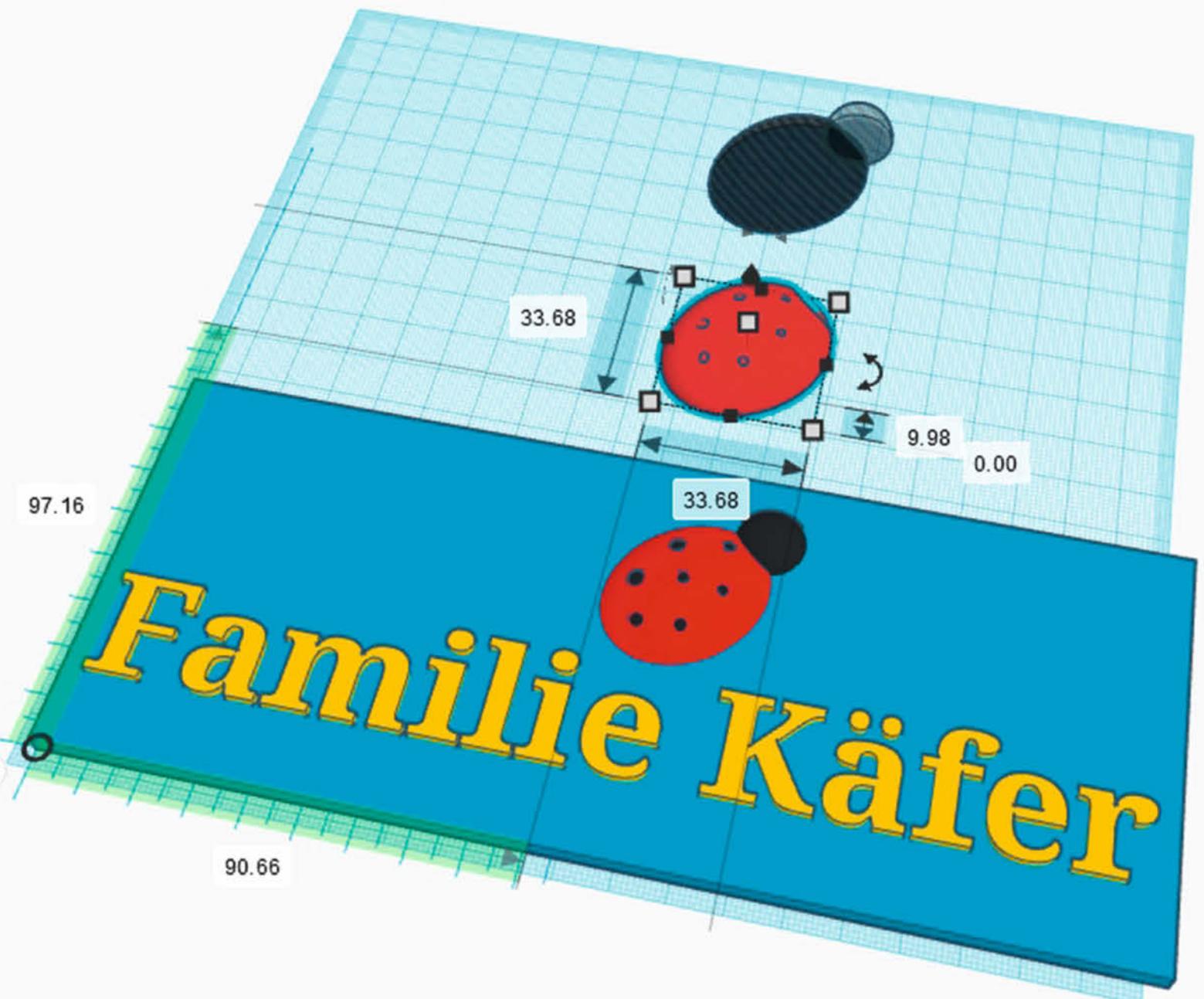
In der Praxis benutzen viele Maker am Ende zwei Werkzeuge: *FreeCAD* und *Blender*, *Fusion* und *Tinkercad* oder auch ein 3D-Programm mit GUI und *OpenSCAD*. Letzteres ist gerade für Spezialfälle ein wunderbares (Zweit-) Werkzeug, etwa wenn man parametrisierbare 3D-Dateien weitergeben will: Schreibt man alle Parameter in Variablen an den Anfang der Datei und kommentiert sie verständlich, können daraus auch viele andere Nutzen ziehen, die nicht gelernt haben, in *OpenSCAD* zu programmieren. Ein Paradebeispiel für so ein Skript wäre eine *OpenSCAD*-Datei, die eine anpassbare STL-Negativform für die Senklochbohrungen ab Seite 92 erzeugt. Haben wir Sie animiert? Hoffentlich! —pek

Onshape	OpenSCAD	SketchUP	Solid Edge Community Edition	Tinkercad
1.148	2021.01	Make 2017	2022	–
PTC	Community	Trimble	Siemens	Autodesk
Webdienst	Lokale installierte Open-Source-Software	Lokal installierte Software	Lokal installierte Software mit jährlichen Updates, Gratis-Lizenz für 3 Jahre, verlängerbar	Webdienst
Browser	Windows/macOS/Linux	Windows/macOS	Windows	Browser
ab 1500 US-\$/Jahr	–	ab 299 US-\$/Jahr	ab 102 €/Jahr	–
Kein kommerzieller Einsatz, nur öffentliche Projekte, kein Teamwork, kein fotorealistisches Rendering, kein Release-Management	–	Kein kommerzieller Einsatz, weniger Dateiformate, keine booleschen Operatoren, wird nicht mehr aktualisiert	Kein kommerzieller Einsatz, Dateien nicht kompatibel mit Vollversion, kein CAM-Modul, Wasserzeichen in exportierten Zeichnungen	–
Parasolid, ACIS, STEP, IGES, Catia v4/v5/v6, Solidworks, Inventor 9, Creo, JT, Rhino, STL, OBJ, NX, Solid Edge, glTF	STL, OFF, AMF, 3MF	SKP, 3DS, DAE, KMZ, STL ⁴	STEP, Parasolid, JT, NX, ACIS, AutoCAD, Catia, IGES, Inventor, ProE, SolidWorks, SDRC, STL, IFC, OBJ, XML	STL, OBJ (max. 25MB)
DWG, DXF	DXF, SVG	–	u.a. DXF, DWG, PDF	SVG (max. 25 MB)
Parasolid, ACIS, STEP, STEP AP242, IGES, Solidworks, JT, PVZ, Rhino, OBJ, STL	STL, OFF, AMF, 3MF	DAE, KMZ, STL ⁴	Parasolid, JT, XGL, ACIS, Catia, IFC, IGES, STEP, 3MF, OBJ, FBX, STL, XML, 3D-PDF, Universal 3D	STL, OBJ, glTF
DWG, DXF	DXF, SVG, PDF	–	PDF	SVG
Installierbar über den Online-AppStore (meist kostenpflichtig)	Viele parametrisierbare Modelle gratis im Netz	Plug-ins aus dem <i>Extension Warehouse</i> , teils gratis, teils kostenpflichtig	API für Makros, auch kostenlose aus der Community	–
–	–	Ruby	VB.Net, C#, C++, Python	–
⊕	○	⊕	⊕	○
○	○	○	⊕	⊕
○	⊖	⊖	○	⊖
⊕	⊖	⊕	⊕	⊖
○	⊖	⊖	⊕	○
– nicht vorhanden oder keine Angabe	⊕ gut ○ mittel	⊖ schlecht		

Multicolor-Konstruieren für 3D-Druck

Mehrfarbig in 3D drucken, das ergibt ungeahnte Möglichkeiten. Aber egal, ob das mit dem preiswerten Farbumbau aus der vorigen Make oder mit teuren mehrfarbfähigen Druckern geschieht: Erst muss konstruiert werden. Hier erfahren Sie, wie das ganz einfach geht.

von Heinz Behling



Familie Käfer schwelgt nach dem Einzug ins neue Eigenheim im Glück. Nur ein originelles Türschild fehlt noch. Der Familienrat einigte sich auf ein Schild mit blauem Hintergrund, auf dem in gelben Lettern der Name sowie ein Marienkäfer als gleichzeitiges Glücks- und Familiensymbol prangen soll. Das ergibt also vier Farben:

- blau: Schildplatte (Hintergrund)
- gelb: Schrift
- rot: Käferflügel
- schwarz: Kopf und Punkte des Käfers

Eine Skizze war schnell angefertigt: Das Schild sollte die Maße 20cm x 8cm bekommen und auf der rechten Seite den Familiennamen enthalten, links daneben den Käfer.

Da das Bankkonto aber durch Hausbau und Möbelkauf arg strapaziert war, war Sparsamkeit angesagt. So sollte der eigene 3D-Drucker zum Einsatz kommen, den man ja mithilfe der Anleitung in der vorigen Make preisgünstig farbtauglich gemacht hatte. blieb noch die Aufgabe, das Schild mit dem Computer zu konstruieren. Auch da bot sich eine kostengünstige Lösung an: die Konstruktion per *Tinkercad* (siehe auch Seite 83). Diese Online-Software erfordert lediglich eine Registrierung, ist einfach zu bedienen, bietet alle Möglichkeiten, die man fürs Multicolor-Konstruieren braucht und kostet nichts! In Tinkercad setzt man die Konstruktionen aus geometrischen Grundformen zusammen, also zum Beispiel Quadern, Kugeln, Zylindern und viele mehr. Das ist besonders für unerfahrene 3D-Konstrukteure ein bequemer Einstieg.

Und so kamen die Glückskäfer zu ihrem Schild. Übrigens: Im Folgenden wird die grundlegende Vorgehensweise gezeigt und Sie erhalten Tipps für gute Druckergebnisse. Die genaue Schritt-für-Schritt-Anleitung können Sie online als Video (siehe Kurzinfor) sehen. Die Anleitung ist nicht nur für den erwähnten Multicolor-Umbau des *Creativity Ender 3* geeignet, sondern kann für alle mehrfarb-fähigen 3D-Drucker verwendet werden, da es letztendlich völlig gleichgültig ist, ob ein 3D-Drucker die Farben nachträglich per Filzstift aufträgt oder mehrere Filamente in den erforderlichen Farben benutzt. Auch für Geräte, die Farben durch Mischung aus Filamenten mit den Grundfarben erzeugen (meist Cyan, Magenta und Gelb, evtl. noch Schwarz und Weiß), ist diese Art des Konstruierens geeignet.

Tinkercad vorbereiten

Nach der Registrierung auf Tinkercad wählen Sie *einen neuen Entwurf erstellen*. Daraufhin erscheint die Arbeitsfläche, die an Millimeterpapier erinnert **1**. Wichtig beim Farbdruck: Alle Teile jeder Druckfarbe müssen später dem Slicer als jeweils eigene Datei übergeben werden. Für dieses Beispiel wird es also am Ende je eine STL-Datei für Blau, Gelb, Rot und

Kurzinfor

- » Vierfarbiges Türschild konstruieren mit Tinkercad
- » Besonderheiten für Farb-3D-Druck
- » Tipps für besseres Farbdrucken

Checkliste

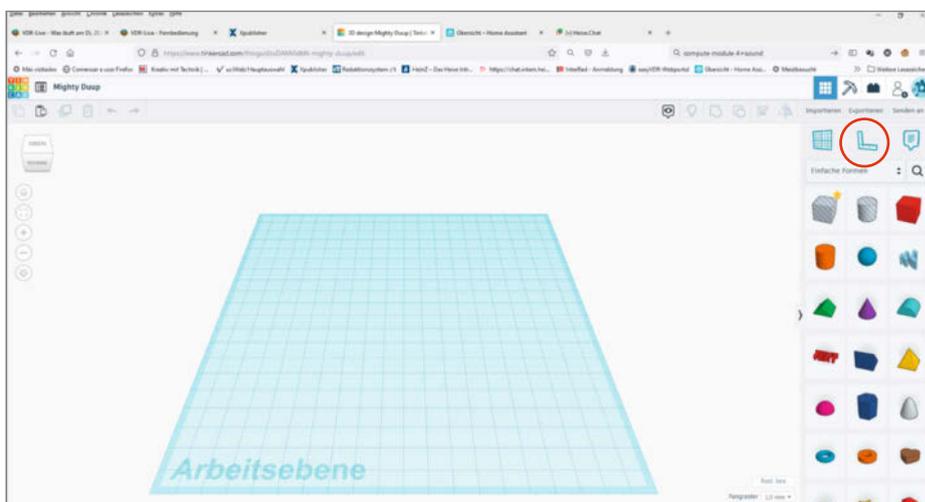


Zeitaufwand:
1 bis 2 Stunden (für dieses Beispiel zzgl. Druckzeit)

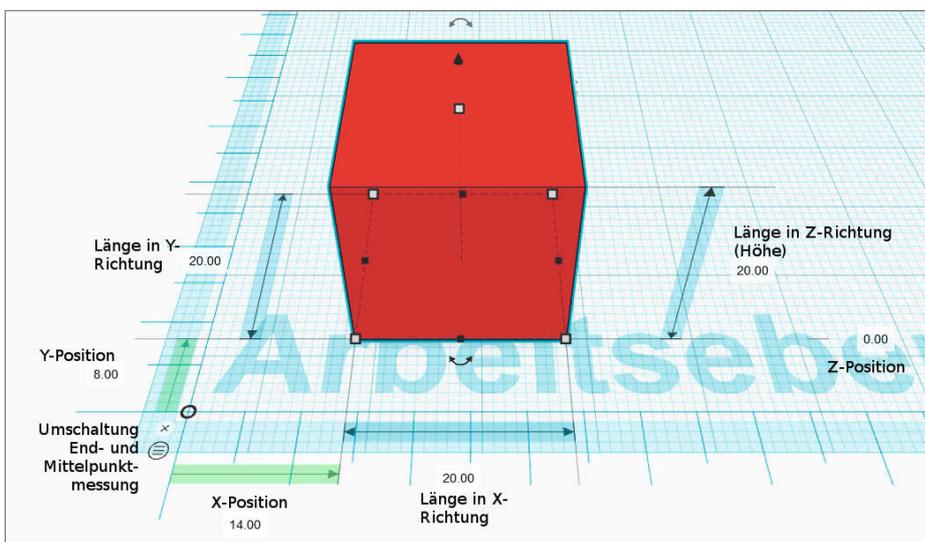
Alles zum Artikel im Web unter [make-magazin.de/xv39](https://www.make-magazin.de/xv39)

Mehr zum Thema

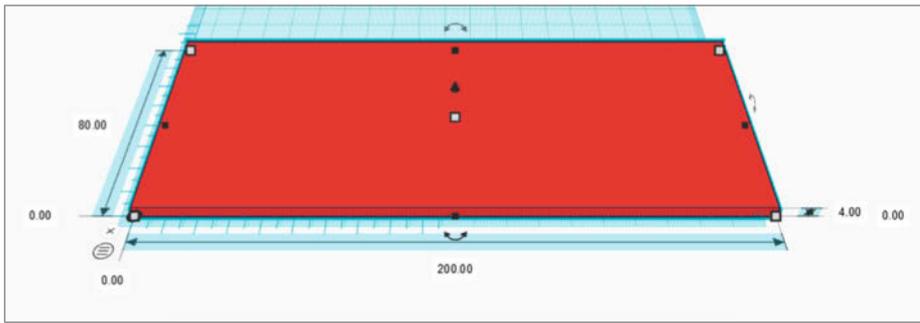
- » Carsten Wartmann: 3D-Druck-Veredelung, Make 5/21, S. 120
- » Heinz Behling: Farb-3D-Drucker, Make 3/22, S. 92
- » Video: Klick für Klick zum Türschild



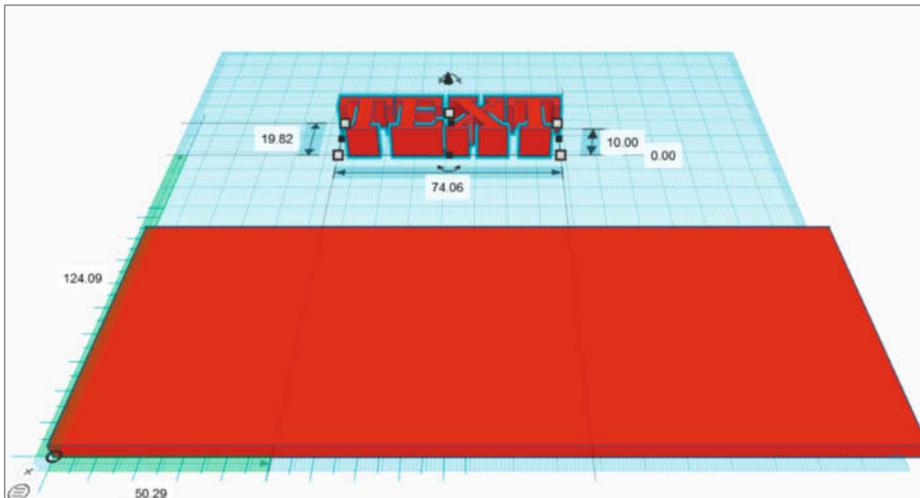
1 Nach Anlegen eines neuen Entwurfs in Tinkercad sollten Sie den kryptischen Namen oben links ändern und das Lineal einblenden.



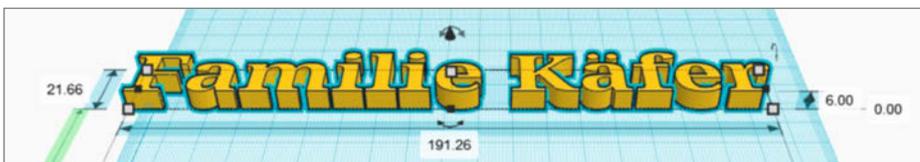
2 Das bedeuten die Werte, die auf der Arbeitsfläche angezeigt werden.



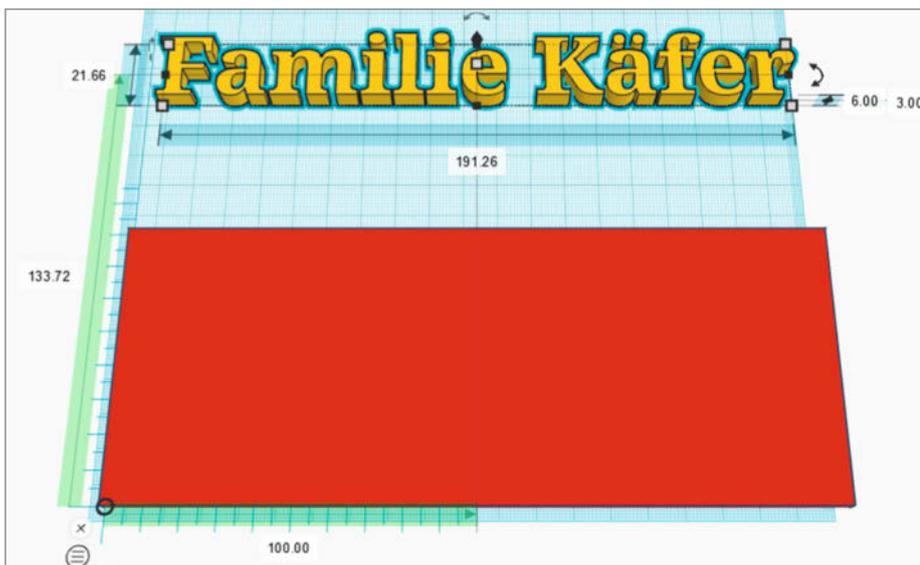
3 Die Maße der Schildplatte



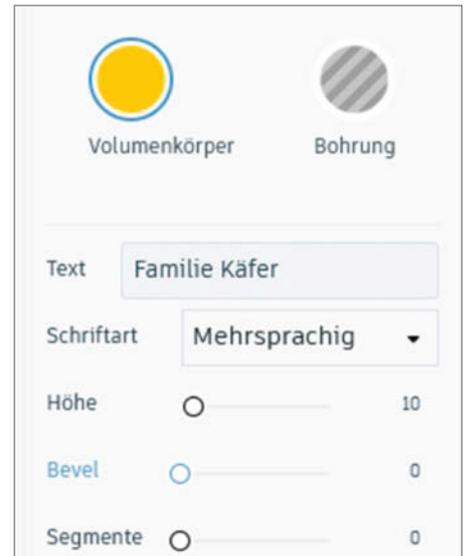
4 Als Nächstes bringen wir den Text in Form.



6 Die Position des Textes ist im Moment noch nicht wichtig, nur die Maße.



7 Der Name ist nun zentriert und hat die richtige Größe.



5 Die hier gewählte Farbe erscheint nur auf dem Bildschirm. Die Druckfarbe wird erst im Slicer gewählt.

Schwarz geben. Bei der Konstruktion müssen wird diese Einzelteile jedoch so anordnen, wie sie später auch im fertig gedruckten Objekt liegen. Allerdings dürfen wir die Einzelteile nicht miteinander gruppieren und so verschmelzen. Doch dazu später noch mehr.

Um die Teile exakt positionieren zu können, müssen Sie das Lineal in die Arbeitsfläche einsetzen: Ziehen Sie dazu das Winkel-Lineal mit der Maus auf die untere linke Ecke der Arbeitsfläche. Von nun an werden Ihnen, sobald Sie einen Teil der Konstruktion einfügen oder anklicken, die entsprechende Position und die Maße angezeigt. Außerdem sollten Sie den von Tinkercad automatisch vergebenen Entwurfsnamen (im Beispiel *MightyDump*) durch einen weniger kryptischen ersetzen (draufklicken und neuen Namen eingeben).

Systematisches Vorgehen

Beim Farbdruck dürfen sich keine Teile unterschiedlicher Farben durchdringen. Für unser Beispiel heißt das: In dem Raum, den die Buchstaben des Textes (Familiename in gelb) einnehmen, darf kein Teil des Hintergrunds (Schild in blau) liegen. Andernfalls wäre nämlich unklar, in welcher Farbe an den Durchdringungsstellen gedruckt werden soll und der Slicer würde einen Fehler melden. Für die Arbeit beim Konstruieren heißt das in diesem Beispiel, dass wir dort, wo der Text steht, in die Schildplatte entsprechende Ausschnitte in Form der Buchstaben einfügen müssen. Ähnliches gilt für die schwarzen Punkte und den Käferkörper sowie den Käfer und den Hintergrund. Hier zeigen wir diesen Vorgang einmal konkret für den Text, alles Weitere zeigen wir im Video.

Beginnen wir mit dem Hintergrund: Ziehen wir also einen Quader mit der Maus auf die Arbeitsfläche **2**.

Dem geben wir nun die Maße der Schildplatte, also 200mm in X-, 80mm in Y- und 4mm in Z-Richtung. Außerdem setzen wir diese mit der unteren linken Ecke auf Position X:0, Y:0 und Z:0 **3**.

Tipp: Sobald Sie die Maße eines Teiles deutlich ändern, gibt es durch die perspektivische Betrachtung in Tinkercad einen deutlichen Verzerrungseffekt, oder das Teil kann sogar aus dem Blickfeld heraus wandern. Mit der Taste **6** im Zahlenblock der Tastatur rückt man es wieder in die Mitte des Blickfelds und die Verzerrungen verschwinden.

Jetzt fügen wir den Text ein, aber bitte zunächst nicht auf die Schildplatte, sondern darüber oder daneben. Das erleichtert die Arbeit **4**.

Ist ein Objekt in Tinkercad gewählt, erscheint ein kleines Fenster mit den Objekteigenschaften. Für das Textobjekt geben Sie darin den Text ein und wählen die Anzeigefarbe **5**.

Auf der Arbeitsfläche sehen Sie sofort das Ergebnis. Achten Sie darauf, dass die Maße des Textes in X- und Y-Richtung zur Schildfläche passen. Zur Höhe: Die Schrift soll in diesem Beispiel später 2mm aus der Schildfläche herausragen. Zusammen mit der Höhe der Platte ergibt das eine Gesamthöhe von 6mm für den Text **6**.

Der Text soll später mittig in der unteren Hälfte des Schildes stehen. Um es zu zentrieren, müssen wir durch einen Klick auf den kleinen Kreis mit den drei Querstrichen auf den Mittelpunktbezug umstellen. Wenn Sie danach auf die Schildfläche klicken, wird Ihnen deren Position in X-Richtung mit **100** angezeigt. Klar, denn die Breite des Schildes ist ja

200mm, die Mitte muss also bei 100mm liegen. Klicken Sie dann auf den Text und ändern Sie dessen X-Position ebenfalls auf **100** **7**.

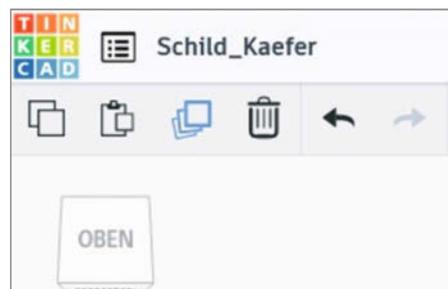
Jetzt müssen wir das Ausschneiden der Textkontur aus der Schildplatte vorbereiten. Dazu legen wir per Klick auf das Symbol **8** eine Kopie des Textobjekts an.

Auf dem Bildschirm tut sich scheinbar nichts. Der Grund: Original und Kopie liegen noch exakt übereinander. Drücken Sie die Cursor-hoch-Taste, und die Kopie wandert in Y-Richtung nach oben. Klicken Sie auf die Kopie und im Eigenschaftsfenster auf *Bohrung* statt *Volumenkörper*, sodass die Schrift gestreift dargestellt wird **9**.

Verschieben Sie dann diese Kopie mit der Cursorortaste nach unten auf die Schildfläche: Achtung: Vermeiden Sie seitliche Verschiebungen, denn der Text ist ja bereits zentriert **10**!

Nun markieren Sie zusätzlich die Schildplatte per Mausclick bei gedrückter Shift-Taste und tippen dann *Strg* und *G*. Das gruppiert die beiden Teile, fügt sie also zu einem Teil zusammen. Da der Text als Bohrung definiert war, wird er quasi von der Schildplatte subtrahiert. Das sieht dann so aus wie in Bild **11**.

Nun positionieren Sie die noch verbliebene Textkopie auf dieselbe Y-Position (25mm).



8 Mit dem blau markierten Symbol kopiert man das gewählte Objekt.

Diesmal darf aber nicht gruppiert werden! Um einen Eindruck vom späteren Aussehen zu bekommen, setzen Sie noch die Farbe der Schildplatte auf blau **12**.

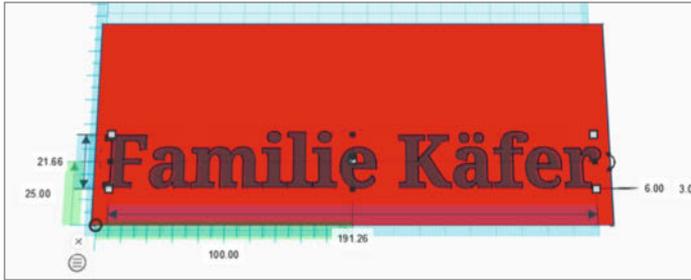
Jetzt ist Kreativität gefragt: Den kleinen Glückskäfer konstruieren Sie nun selbst aus Halbkugeln **13**. Falls Sie dazu Hilfe brauchen: Im Video zeigen wir Ihnen Schritt für Schritt, wie der kleine Krabblerschild kommt.

Einzel speichern

Jetzt müssen wir aus Tinkercad heraus die Teile einzeln auf dem PC speichern. Dabei wird in



9 Die Kopie wird als Stanzform für die Schildplatte benutzt.



10 Wenn die Textkopie platziert ist, merken Sie sich die Y-Position.



11 Der Platz für den Text ist ausgestanzt.



12 Jetzt ist das Schild schon zweifarbig.



13 Das fertige Schild

den Dateien nicht nur die Form des Objekts, sondern auch seine Position gespeichert. Das ist wichtig, damit der Slicer nach der Farbwahl alles wieder zum Gesamtobjekt zusammenbauen kann.

Das Speichern geht aber recht einfach: Markieren Sie einen Teil, zum Beispiel den Käferkörper, und wählen Sie *exportieren*. Speichern Sie dann die STL-Datei unter einem aussagekräftigen Namen, beispielsweise *Schild_Kaefer_body_rot.stl*. Speichern Sie die restlichen drei Teile auf gleiche Weise. Sie haben dann vier Dateien auf dem PC:

- *Schild_Kaefer_body_rot.stl*
- *Schild_Kaefer_kopfpunkte_schwarz.stl*

- *Schild_Kaefer_text_gelb.stl*
- *Schild_Kaefer_platte_blaue.stl*

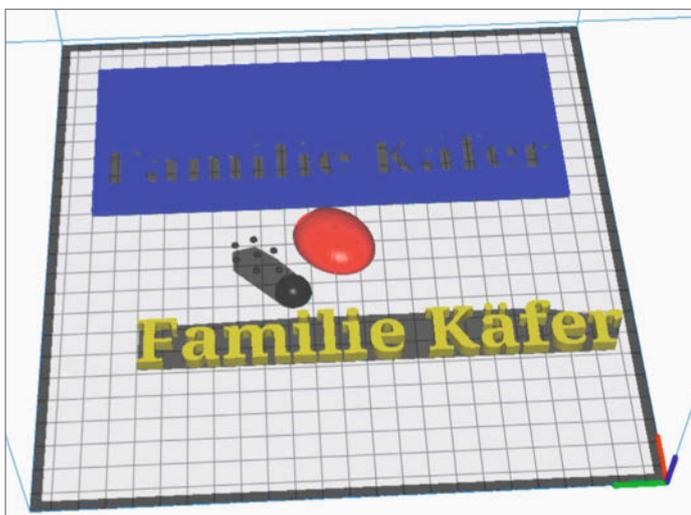
Die weitere Arbeit übernimmt nun der Slicer, hier also *Cura*. Laden Sie darin die vier Dateien und ordnen Sie ihnen jeweils die richtige Druckfarbe zu (Anleitung dazu in der vorigen Make) 14.

Anschließend wählen Sie *alle Teile* und klicken dann auf *Modelle zusammenführen*. Aufgrund der bereits erwähnten Positionsdaten in den STL-Dateien setzt *Cura* nun alles passgenau zusammen und zeigt dann, wie das Schild gedruckt aussehen wird 15.

Nun kann die Datei zum Drucker geschickt werden und nach ein paar Stunden können Sie Ihr Werk betrachten.

Eines sollten Sie noch beachten, falls Sie den Farbdruck-Umbau aus der Make 3/22 verwenden: Durch den Farbauftrag per Filzstift kann der Zusammenhalt der Druckschichten verringert sein. Sie sollten daher mit solchen Geräten nichts drucken, was kleine filigrane Teile enthält, die weit herausragen. Solche Teile brechen sehr leicht ab. Bei Druckern, die mehrere Filamente verwenden, ist diese Gefahr deutlich geringer.

Nun aber dürfen Sie Ihrer Kreativität freien Lauf lassen. Es muss ja nicht unbedingt ein Türschild sein. Schaffen Sie bunte 3D-Kunstwerke. Vielleicht senden Sie uns auch ein Bild Ihrer Schöpfungen? Wir würden uns freuen!
—hgb



14 Erst im Slicer wird die Druckfarbe festgelegt.



15 Jetzt kann gedruckt werden.

Neuer Input für Maker

Make Elektronik Special

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

Heft + Experimentierset für 44,95 €

shop.heise.de/make-elektronik21



Inklusive Experimentierset und Breadboard

Make Micropython Special

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

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

Heft + PDF für 19,90 €

shop.heise.de/make-micropython



Make Picaxe Special

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

Heft + Nano-Axe-Board für 24,95 €

shop.heise.de/make-picaxe



Inklusive Nano-Axe-Board mit PICAXE-08M2

3D-Druck: Hängende Löcher ohne Stützen

Senkungen kopfüber zu drucken erfordert Stützen für die überhängenden Löcher. Es ist jedoch mühselig, diese später aus den engen Stellen zu entfernen. Wenn wir die Lochkonstruktion vorher ein wenig anpassen, gelingt der Druck aber auch ohne Stützen.

von Ákos Fodor



1 Das Dummy-Gehäuse mit Senkungen für M3-Zylinderkopfschrauben

Manchmal ist es wie verhext: Ganz gleich, wie ich ein 3D-Modell auf dem Druckbett drehe und wende, es findet sich keine vernünftige Ausrichtung, mit der ich auf Stützen verzichten könnte. Diese verlängern jedoch die Druckzeit, verbrauchen zusätzliches Material und sind, wenn es eng wird, nur mit Mühe zu entfernen. Gerade neulich stieß ich zu diesem Thema auf einen *Reddit*-Beitrag, in

dem sich der Nutzer *mrflib* beklagte, Michelangelo habe seinen *David* Anfang des 16. Jahrhunderts (offensichtlich) nicht mit FDM im Hinterkopf gemeißelt. In der Tat benötigt die Skulptur stehend oder liegend fast noch einmal so viel Material für die Stützen. Das lässt sich jetzt nicht mehr ändern. Bei unseren eigenen Entwürfen können wir jedoch versuchen, Stützen konstruktiv zu vermeiden. Im Fall von

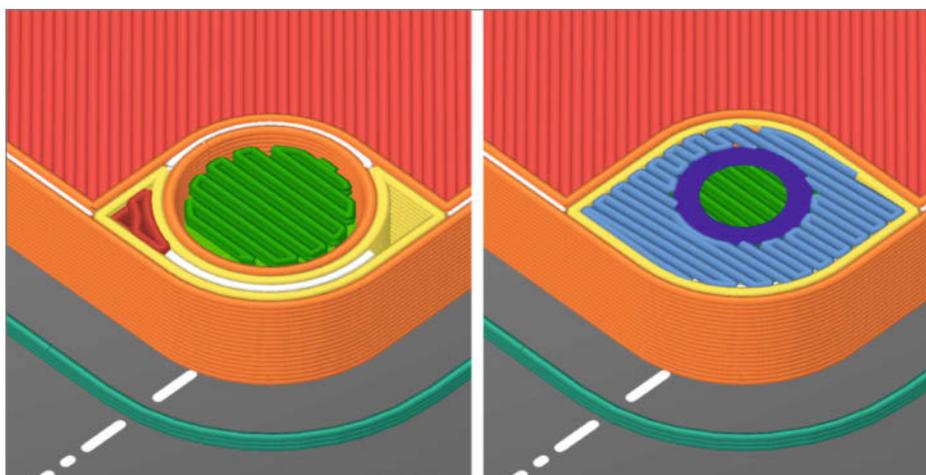
Senkungen für Zylinderkopfschrauben ist es mit einem einfachen Trick sogar möglich, vollständig auf Stützen zu verzichten – selbst wenn die Löcher hängend, also mit der Senkung nach unten gerichtet, gedruckt werden.

Zur Veranschaulichung verwende ich ein funktionsloses Dummy-Gehäuse, bestehend aus zwei Schalen 1. Diese werden mithilfe von M3-Zylinderkopfschrauben zusammengehalten. Daher habe ich das Modell mit Durchgangslöchern und Senkungen versehen, in denen die Zylinderköpfe bündig mit dem Gehäuse abschließen. Wenn ich die Schalen in einen Slicer importiere und mit der Gehäuseöffnung nach oben auf der Druckplattform platziere, erkennt das Programm beim Slicen zu Recht die Überhänge in den Senkungen und bietet mir Stützen an 2. Lasse ich sie weg, wird die erste Schicht der Löcher in der Luft gedruckt und ob das Ergebnis nachher zu gebrauchen ist, hängt vom Zufall ab. Also muss ich die Senkungen im Vorfeld ein wenig anpassen.

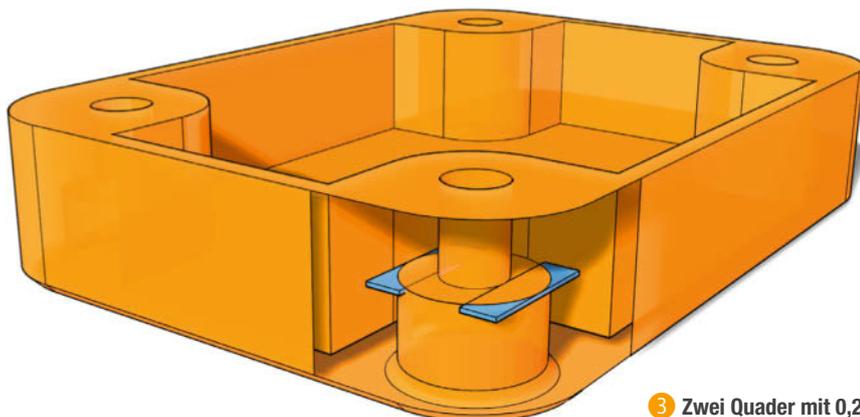
Senkungen überbrücken

Um die hängenden Löcher zu tragen, verwendet diese Technik *Bridging*. Allerdings wird nicht die gesamte Senkung überbrückt, sondern nur die am Loch anliegenden Flächen, und zwar schichtweise. Das führt zu sauber gedruckten Löchern, die unmittelbar einsatzbereit sind. Im Folgenden erkläre ich euch die notwendigen Schritte anhand eines exemplarischen Lochs:

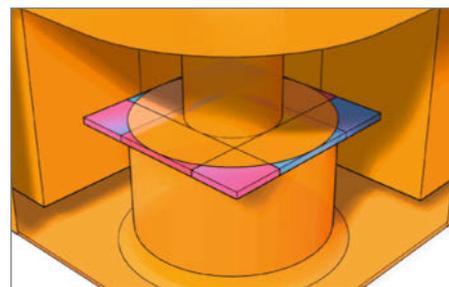
- 3 Fügt in CAD links und rechts vom Durchgangsloch flache Quader ein, die an den Längsseiten sowohl das Loch als auch die Senkung berühren. Die Quader sollten die spätere Druckschichthöhe haben (hier 0,2mm) und in der Senkung unterhalb des Lochs sitzen, sodass sie an der Deckelfläche der Senkung aufliegen.
- 4 Dupliziert beide Quader und dreht sie um 90 Grad. Jetzt sieht das Gebilde aus wie



2 Im Prusa-Slicer: Wenn ich die Stützen (grün) ignoriere, wird das Loch (blau) in der Luft gedruckt.



3 Zwei Quader mit 0,2mm Höhe unterhalb des Lochs



4 Dupliziert und um 90 Grad gedreht entsteht eine Art Rahmen.

ein Rahmen mit einem ausgesparten Quadrat in der Mitte.

- 5 Wählt jetzt zwei gegenüberliegende Quader aus und skaliert sie nach unten auf die doppelte Schichthöhe. Bei 0,2mm sollten sie danach 0,4mm hoch sein.
- 6 Vereint die Quader mit der restlichen Geometrie mithilfe einer booleschen Operation.

Wenn ihr das Modell anschließend im Slicer prüft, könnt ihr sehen, dass die Senkung schrittweise geschlossen wird: erst an den Seiten, dann mit einer quadratischen Ausspa-

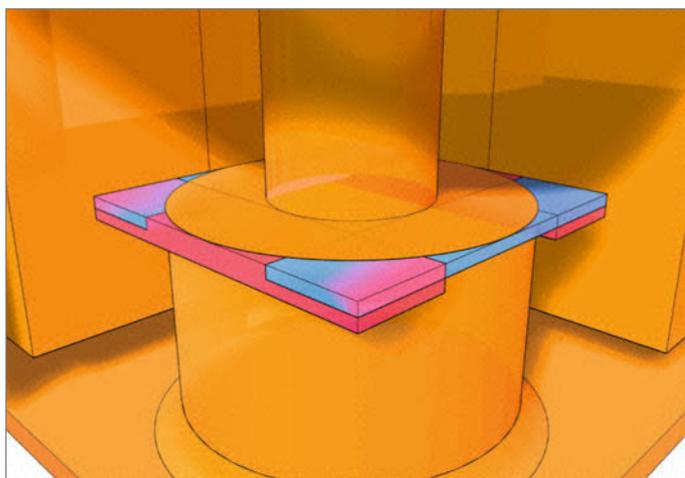
rung, auf der anschließend das Durchgangsloch aufgebaut wird 7. Genauso wird das Modell anschließend auch gedruckt.

Einfacher mit Werkzeug

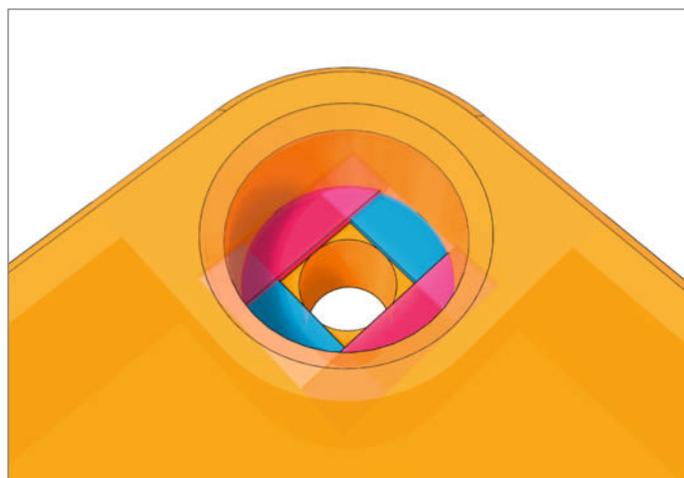
Diese Methode bei jedem CAD-Modell und Loch von Grund auf zu wiederholen, wäre ein aufwendiges Unterfangen. Daher habe ich ein paar Stanzwerkzeuge gebaut, die ihr für die Größen M3, M4 und M5 in unserem Github-Repository (siehe Link) herunterladen könnt. Die 3D-Modelle liegen dort als STL

und in diversen CAD-Austauschformaten. Platziert die Werkzeuge an eurem Modell dort, wo ihr ein Durchgangsloch plant und subtrahiert es anschließend mit einer booleschen Operation. Die Werkzeuge sind so konstruiert, dass die erstellten Löcher sich anschließend auch mit 0,3mm Schichthöhe drucken lassen. Falls ihr tiefere Löcher benötigt, extrudiert einfach die abschließenden Flächen des Werkzeugs. Viel Spaß beim Locher! —akf

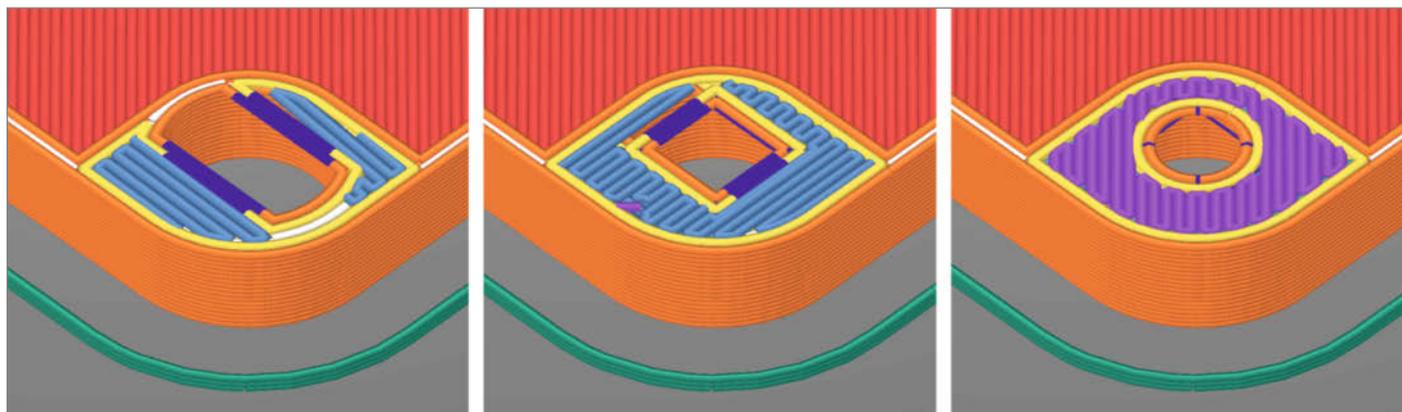
► make-magazin.de/x9fu



5 Die Höhe zweier gegenüberliegender Quader muss nach unten verdoppelt werden.



6 Von unten betrachtet sind die gestuften Überbrückungen vor dem Vereinen gut zu erkennen.



7 Der Slicer überbrückt die Senkung jetzt schrittweise, sodass wir keine Stützen mehr benötigen.

Fledermaus-Scanner goes LoRaWAN

Den Fledermaus-Scanner kennt ihr vielleicht aus den Heften 6/20 und 4/21 als stationäre und mobile Variante. Jetzt gibt es eine neue Erweiterung mit LoRaWAN, die es ermöglicht, Messdaten bequem aus der Ferne auszuwerten.

von Ralf Stoffels

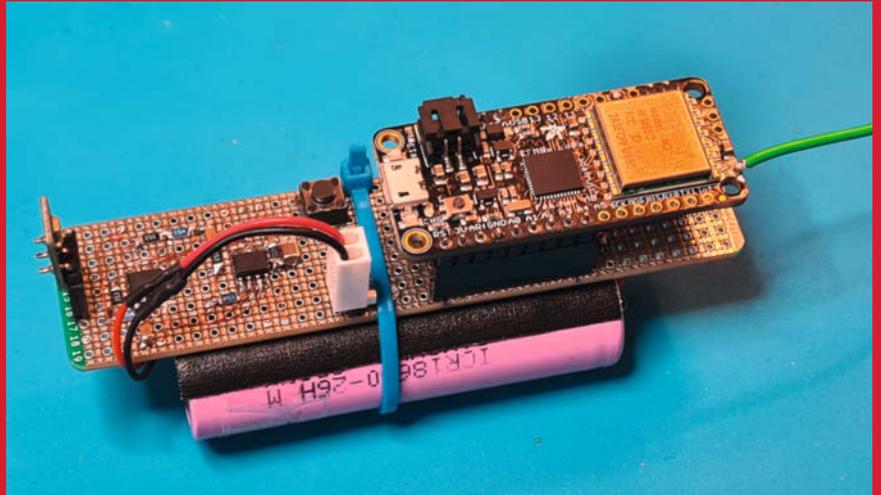


Die Lebensräume von Fledermäusen zu beobachten, gibt Umweltbiologen Aufschluss über Veränderungen in unserem Ökosystem. Das geht beispielsweise mithilfe meines Fledermaus-Scanners, der zukünftig in der Stadt Calw bei flächendeckenden Langzeitbeobachtungen unterstützen soll. Als Teil eines größer angelegten Naturschutzprojektes hilft er dabei, die Vorkommen verschiedener Fledermausarten zu erkennen und zu dokumentieren, die sich aufgrund ihrer charakteristischen Frequenzen voneinander unterscheiden lassen. Damit es einfacher ist, die erfassten Aufzeichnungen auszuwerten, habe ich den Entwurf so angepasst, dass der Scanner in der dritten Version die Messergebnisse per LoRaWAN an einen Server übermitteln kann. Das bot sich an, weil es im erweiterten Stadtgebiet von Calw eine gute LoRa-Abdeckung gibt. Dadurch erübrigt sich die physische Interaktion, außer der Scanner muss geladen oder gewartet werden.

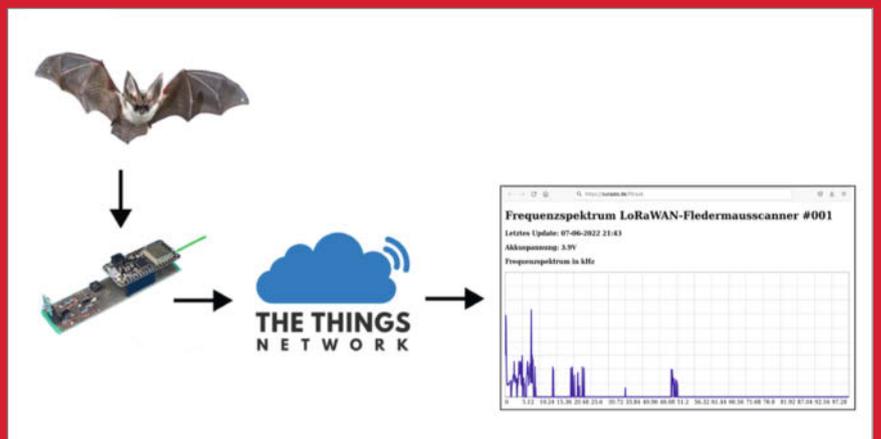
Das Prinzip des Entwurfs beruht darauf, mit einem schnellen *Analog/Digital-Wandler* (ADC) das ganze Frequenzband von 0-90kHz abzutasten und dann die empfangenen Frequenzanteile mittels *Fast-Fourier-Transformation* (FFT) zu ermitteln. Auch in dieser Version habe ich das *Feather M0+* Board von *Adafruit* verwendet, dessen *SAMD21 ARM-Prozessor* einen schnellen ADC hat – mit bis zu 500.000 Samples pro Sekunde. Die Boardversion mit integriertem LoRaWAN-Modem ist perfekt für dieses Projekt. Für die Übertragung an meinen Webserver verwende ich das *The Things Network*. Da es jedoch nur Datenpakete von maximal 54 Bytes unterstützt, die auch nur einmal pro Minute gesendet werden dürfen, ist eine Übertragung des kompletten Spektrums mit 1024 16-Bit-ADC-Werten nicht möglich. Das *Feather M0+* Board rechnet daher per FFT das Spektrum aller Signale aus, die innerhalb einer Minute empfangen wurden, sortiert anschließend die Frequenzanteile nach Amplitude und sendet dann nur die 12 höchsten Werte per LoRaWAN. Über einen Zeitraum von mehreren Stunden kann man so dennoch reichlich Spektraldaten erfassen. Derzeit werden bei mir die Daten eines einzelnen Fledermaus-Scanners grafisch angezeigt. In Zukunft plane ich, weitere verteilte Sensoren einzubinden.

Falls ihr neugierig seid, welche Fledermäuse in eurer Nähe unterwegs sind, findet ihr die Anleitung für den LoRaWAN-Fledermaus-Scanner auf meiner Website. Im September 2022 werde ich diesen und andere LoRaWAN-Projekte auf der *Maker Faire Hannover* vorstellen. —akf

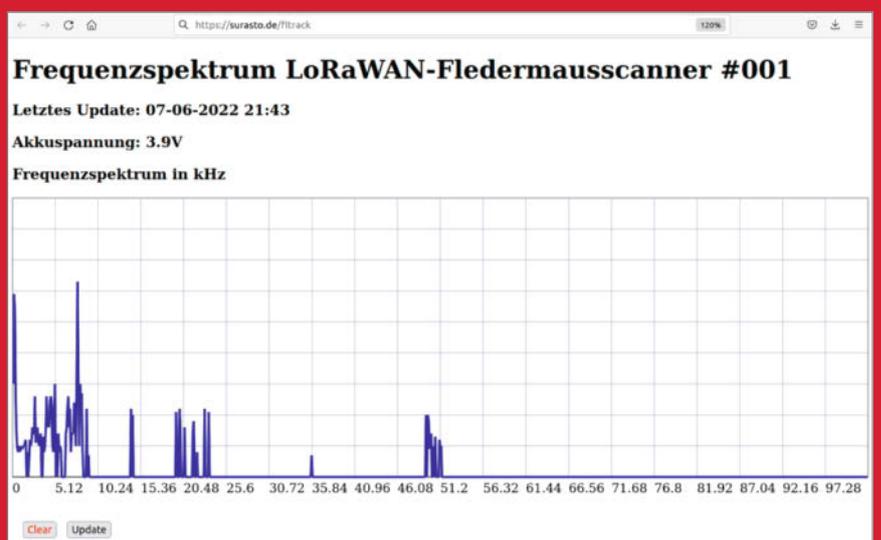
► make-magazin.de/x3dx



Die Verstärkerschaltung mit Feather M0+ LoRa



Die aussagekräftigsten Messergebnisse werden mithilfe des The Things Network an einen Server gesendet.



In einer grafischen Darstellung lassen sich die Fledermausarten anhand spezifischer Frequenzen erkennen.

TrudeTrailer: DIY-Fahrrad-Lastenanhänger

Große Lastenräder und -anhänger sind praktisch und umweltfreundlich. Sie kosten jedoch schnell mehrere tausend Euro. Dass es auch günstiger geht, zeigt die Welcome Werkstatt e.V. mit der kostenfreien TrudeTrailer-Bauanleitung.

von Ákos Fodor



Bilder: Marvin Dunze

Sie huschen lautlos und emissionsfrei durch die Städte, transportieren fröhliche Hunde, Kinder oder ausladende Einkäufe – manchmal auch alles auf einmal. Die Rede ist von Lastenrädern und Fahrrädern mit Anhängern. Sie sind praktisch, schonen die Umwelt und erfordern nicht einmal einen Führerschein. Allerdings gibt es einen Haken: Wer so ein Gefährt kaufen möchte, gibt schnell mehrere tausend Euro aus. „Wieso nicht einfach selber bauen?“, haben sich die Mitglieder der Hamburger *Welcome Werkstatt e.V.* gefragt und unter der Werkstattleitung von Ole Burmester gemeinsam eine Alternative entwickelt: *TrudeTrailer*, einen Fahrrad-Anhänger in Übergroße zum Selberbauen. Das Projekt liegt mit den reinen Materialkosten knapp unter 1000 Euro und bietet dank seiner modularen Bauweise genügend Spielraum für individuelle Anpassungen. Dem

XYZ-Cargo-Prinzip von *N55* und *Till Wolfer* folgend und von den Fahrradanhängern der Marke *Carla Cargo* inspiriert, besteht *TrudeTrailer* Grundgerüst aus speziell verschraubten Aluminium-Profilen, dessen konstruktive Stärke in den besonderen Eckverbindungen steckt. So stabilisiert sich das Gerüst in alle Richtungen selbst. Um das System zu veranschaulichen, zieht Wolfer auf der Website von *XYZ-Cargo* den Vergleich zu Panzersperren, die nach demselben Prinzip gebaut, nur eben verschweißt und nicht geschraubt werden.

Dank ihrer leichten Bauweise wiegt *TrudeTrailer* zarte 34kg und lässt sich aufgrund der Kastenform sogar platzsparend hochkant verstauen, wenn sie nicht benötigt wird. Die Ladefläche von 1,65m × 0,65m (ca. 8 Kästen Mineralwasser) trägt mühelos Gewichte bis 150kg und größere Objekte lassen

sich dank des offenen Rahmens einfach mit zusätzlichen Haltegurten stabilisieren. Dem nächsten Sofa-Transport steht also nichts im Wege. Damit der Anhänger sich bei jedem Wetter sicher manövrieren lässt, sind zwei Auflaufbremsen in Form hydraulischer Scheibenbremsen verbaut sowie signalgebende Vorder- und Rücklichter – also alles StVZO-konform.

Wer ein Fahrrad besitzt und schon länger über einen passenden Anhänger nachdenkt, kann sich die *TrudeTrailer*-Bauanleitung kostenfrei auf der Website des Vereins herunterladen, inklusive der benötigten CAD-Daten für den 3D-Druck, Zeichnungen und Teilleisten. Mit Sicherheit freut sich die offene Werkstatt auch über einen persönlichen Besuch. —akf

► make-magazin.de/xtnn



Das XYZ-Prinzip stabilisiert *TrudeTrailer*, auch ohne Schweißgerät.



Wenn das Fahrrad bremst, bremst auch *TrudeTrailer*.

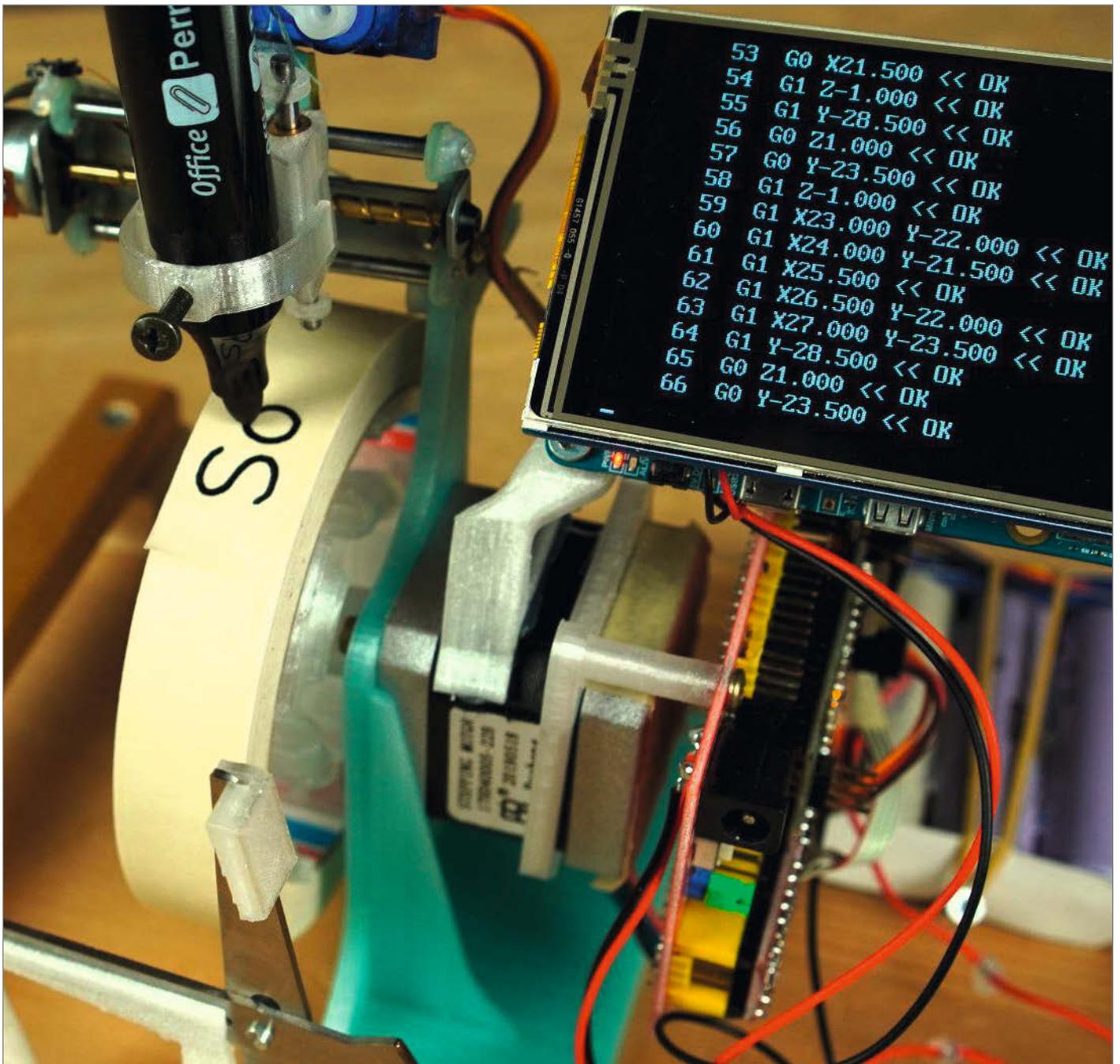


Wetterfest mit hydraulischen Scheibenbremsen

Masking-Tape- Etikettendrucker

Masking-Tape oder Kreppband wird gerne zum Beschriften von Gegenständen und Schubladen benutzt, ist es doch billig und leicht wieder ablösbar. Mit diesem Plotter für Klebeband werden die Aufkleber zusätzlich noch gut lesbar und auch die Herstellung von großen Stückzahlen verliert ihren Schrecken.

von Matthias Martin



Etikettendrucker und die assoziierten Skandale sind seit dem Dymo-DRM-Theater in aller Munde. Ich hatte, wie wohl alle Bastler, so manche Schublade und Kiste mit Werkzeug und Bauteilen darin, welche ein leserliches Etikett benötigten.

So richtig konnte ich mich mit den kommerziellen Angeboten nie anfreunden – zu pixelig, zu klein, zu viel Plastik, zu teures Verbrauchsmaterial. Da kam mir dann eines Tages der Geistesblitz mit dem Kreppband-Plotter: Eine Rolle des besagten Bandes wird direkt von einem Nema-17 Schrittmotor auf der X-Achse rotiert (Schreibrichtung) und in Y-Richtung genügt der Verfahrensweg eines Mini-Schrittmotors aus einem DVD-Laufwerk. Die Z-Achse wird wie bei so vielen Pen-Plotter-Basteleien mit einem RC-Servo bewegt.

Die Benutzerschnittstelle ist Menü-basiert mittels Bash und dem *dialog*-Programm. Das Herzstück des Software-Stacks ist das Programm *hf2gcode* von Andreas Weber: Es wandelt, wie der Name schon besagt, eingegebenen Text in *G-code*. *hf2gcode* würde zwar auch auf einem *ATmega*-Mikrocontroller laufen, aber ich wollte schnell vom Fleck kommen und habe kurzerhand einen Odroid (C1) der ersten Generation aus meinem Fundus genommen, denn Texteingabe ist nicht so einfach mit Arduino und Co. zu bewerkstelligen und ich wollte möglichst schnell vorankommen.

Mechanisch gab es zwei Herausforderungen: Erstens wollte ich eine kompakte Y-Achse und zweitens eine leichtgängige und präzise Z-Achse. Um diese Ziele zu erreichen, habe ich mir, inspiriert von DVD-Laufwerken und dem *Axidraw*-Stiftplotter, lineare Führungen mit Silberstahlwellen und gesinterten Bronze-Gleitlagern gebaut. Die Z-Achse war besonders herausfordernd, weil der geringste Verzug zu einem Klemmen führt und dann der Stift nicht mehr unter seinem eigenen Gewicht nach unten fährt.

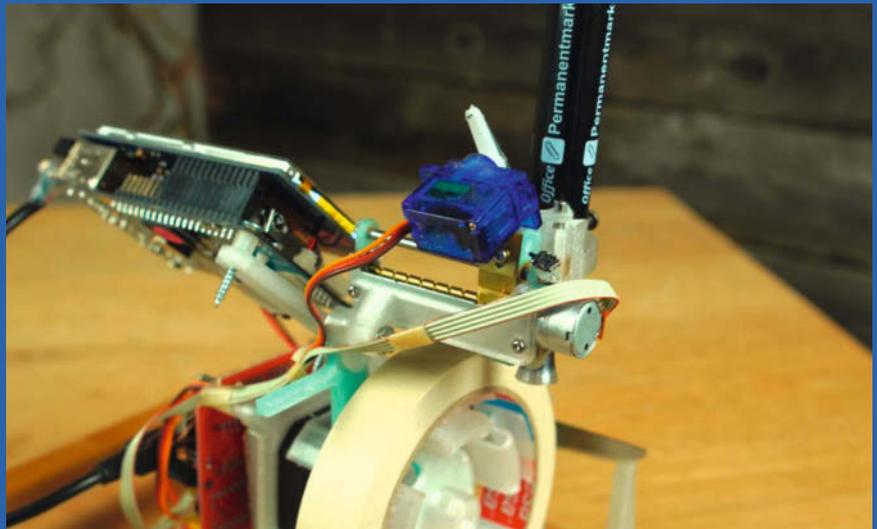
Die Schwierigkeit bei der Y-Achse war der Nachbau des filigranen Mitnehmers aus Kunststoff, welcher in das Gewinde der Spindel aus dem DVD-Laufwerk eingreift. Ich hab dieses winzige Teil in *Blender* modelliert und mit 0,1mm Schichtdicke ausgedruckt - und es hat zu meiner Verwunderung funktioniert!

Die X-Achse war einfach, der Rollen-Halter ist ein einfacher Rotationskörper und beim verantwortlichen Schrittmotortreiber habe ich das Microstepping auf das Maximum (16x) gesetzt und somit genug Auflösung erreicht.

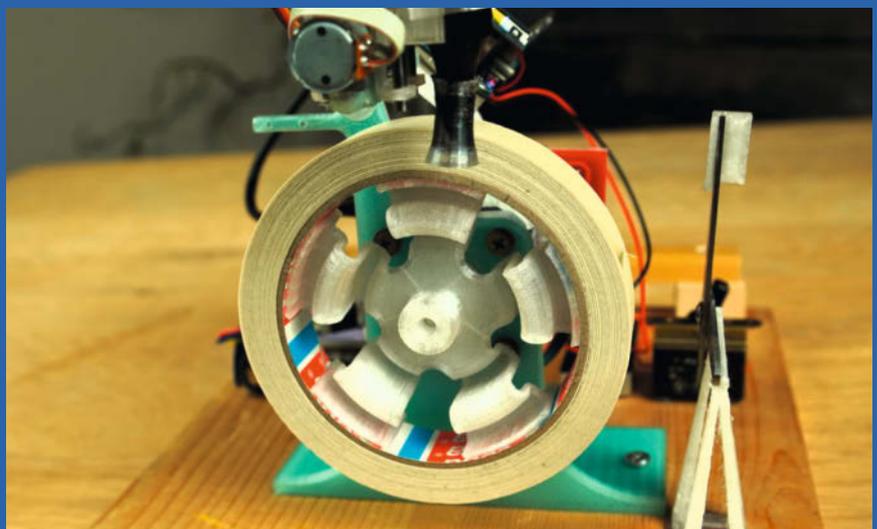
Die meisten Teile der Maschine habe ich in *FreeCAD* modelliert, aber bei gewissen Teilen habe ich einfach mit einem Klecks Heißkleber eine aufwendige Konstruktion vermieden, so geschehen z. B. beim Z-Servo.

Wer die Maschine nachbauen oder modifizieren möchte, findet dazu alle benötigten Dateien auf meiner *Hackaday*-Seite. —caw

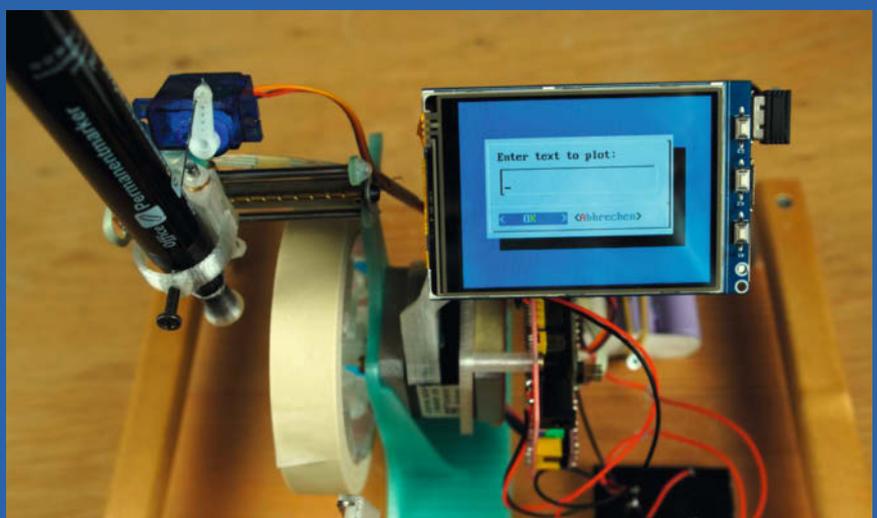
► hackaday.io/project/185279



Y-Achse und Mechanik zur Stiftführung



X-Achse und Rollenhalter



Interface zur Texteingabe

Im Maker-Fotostudio

Das Projekt ist fertig und wird allgemein bewundert – und erst dann fällt auf, dass man keine Fotos von interessanten Details des Innenlebens oder von Zwischenschritten des Baus gemacht hat. Besser, man macht sich gleich zur Routine, das Entstehen seiner Maker-Projekte zu dokumentieren. Meiner Erfahrung nach ist der Aufwand dafür gar nicht so groß, wie man vielleicht zuerst denkt – passende Vorbereitung und Ausrüstung erleichtern die Sache dabei deutlich.

von Daniel Springwald



Wenn man leidenschaftlich gerne bastelt, gibt es im Freundeskreis oder in der Mittagspause immer wieder Gelegenheiten, über das Hobby zu sprechen. Oft ist es dann aber schwierig, nur mit Worten zu berichten, woran man gerade werkelt. Wollte ich Anfang der 1990er Jahre jemandem beschreiben, wie meine computergesteuerte Laser-show nebst selbst gebauter Nebelmaschine genau aussieht, halfen bestenfalls ein Blatt Papier und ein Stift für eine spontane Skizze. Fotos hingegen waren damals noch ausschließlich analog, mit vergleichsweise hohen Kosten und Zeitaufwand für die Entwicklung und Abzüge. Dementsprechend überlegte man sich gut, was man fotografierte. Für Filmaufnahmen stand im besten Fall eine Videokamera mit VHS-Band zur Verfügung, bei vielen aber auch gar nichts.

Wer zeigen will, braucht Fotos

Seitdem ist es glücklicherweise einfacher und günstiger geworden, Aufnahmen vom eigenen Bastelprojekt zu machen. Wenn alles fertig zusammengebaut ist, merkt man dann aber doch oft, dass man gar nicht festgehalten hat, wie das Gerät entstanden ist oder wie es von innen aussieht. Dabei sind solche Aufnahmen viel wert, nicht nur, wenn man sein Projekt anderen tiefergehend erläutern will, sei es im echten Leben oder online. Auch für den Hausgebrauch lohnt sich die Dokumentation: Vielleicht möchte man später mit etwas zeitlichem Abstand ein zweites Exemplar bauen oder bewährte Teile in ein neues Projekt übernehmen. Dann kann es ohne Fotos vom Innenleben teilweise schwer werden, die Verkabelung nachzuvollziehen – vor allem, wenn das Zerlegen des bereits fertigen Gerätes schwierig oder sogar unmöglich ist.

Aus all diesen Gründen habe ich im Laufe der Zeit damit begonnen, beim Basteln möglichst viele Detailfotos vom Entstehungsprozess zu machen. Wenn man gerade mitten *im Fluss* ist, sammeln sich aber auf dem überfüllten Schreibtisch gerne mal die Werkzeuge, Kabel und das sonstige Material der letzten Arbeitsschritte an. Ein davon gemachtes Foto fördert dann vielleicht das Bild des kreativen Bastlers (wenn man das denn möchte), wirkt aber ansonsten eher unordentlich als vorzeigbar. Die vielen Objekte im Bild lenken zudem vom eigentlichen Motiv ab ❶.

Der gute Vorsatz, Arbeitsfortschritte zu dokumentieren, ist zwar schnell gefasst – in der Praxis bedeutete es für mich aber zunächst eine ziemliche Umstellung: Denn „mittendrin“ den Schreibtisch komplett aufzuräumen, nur um ein ansprechendes Foto vom Baufortschritt zu machen, erschien mir zunächst doch recht aufwändig. Nach etwas Eingewöhnung konnte ich aber als Benefit zusätzlich ein

Kurzinfo

- » Tipps für günstiges Equipment wie Kamera, Stative und Licht
- » Fotohintergründe von der Schneidematte bis zum Roll-Up-Banner
- » Ausblick: Videos drehen für Maker

Mehr zum Thema

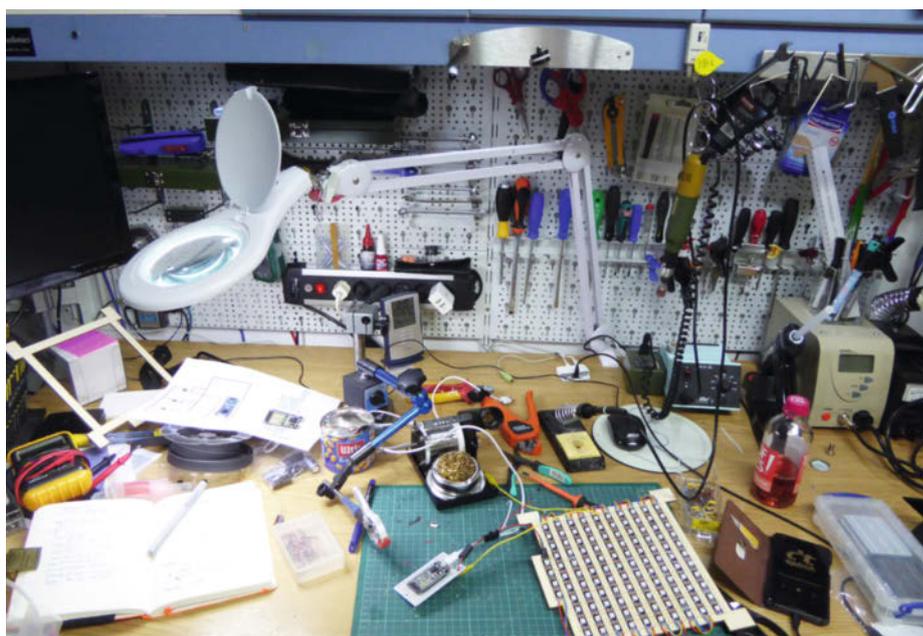
- » Carsten Wartmann, Flexible Stative im Eigenbau, Make 1/22, S. 68
- » Carsten Meyer, Automatik-Blitzgerät, Make 4/21, S. 102
- » Achim Berg, Neues Leben für alte Computermonitore, Make 5/17, S. 68, **gratis im Volltext online**
- » Florian Fusco, Weiches Videolicht, Make 2/16, S. 128, **gratis im Volltext online**
- » Florian Fusco, Die Schlüssel der Schönen, c't Hacks 3/14, S. 90, **Gratis-PDF online**
- » Thomas Saur, Bessere Fotos mit LED-Taschenlampen, c't Hacks 1/12, S. 87
- » Burkhard Fleischer, Licht richtig aufs Objekt führen, **kostenloser Online-Artikel**
- » Florian Schäffer, Grundlagen: Ins rechte Licht gesetzt Make 2/17, S. 32, **gratis im Volltext online**
- » Video: Mehr zu den Themen Licht, Hintergründe und Stativpositionen

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



entspannteres Arbeiten feststellen: Weniger Suche auf dem Schreibtisch und ausreichend Platz zum Hantieren macht einfach mehr Spaß – vor allem, wenn man dann beim Hin- und Herschieben auf dem vollen Schreibtisch nicht mehr aus Versehen das Material in den noch heißen Lötkolben schiebt.

Für ein gutes Bild kann es sich also durchaus lohnen, alles etwas herzurichten. Ansonsten sollten aber der praktische Aufwand und vielleicht Investitionen für das Foto möglichst so gering sein, dass sie keine Hürden darstellen. Die wenigsten Maker werden in der Wohnung einen freien Raum oder eine ungenutzte



❶ Auf einem gefüllten Tisch ist das Hauptmotiv schwer zu entdecken.



2 Tischstativ mit Smartphone-Adapter

3 Die zwei Teile einer Schnellwechselplatte

Ecke haben, die sie langfristig und exklusiv zum Fotografieren einrichten wollen. Dennoch möchte man schnell bereit für die Aufnahme sein – also mit möglichst wenig Auf- und Abbau. Zu guter Letzt möchte man anschließend keine (größeren) Teile der Fotoausrüstung störend herumstehen haben – alles sollte optimalerweise nachher „unters Bett“ passen. Meine im Folgenden geschilderten Erfahrungen helfen hoffentlich möglichst vielen, den für sie praktikablen Weg zur routinemäßigen Fotodokumentation der eigenen Projekte zu finden.

Das nötige Equipment wie Lampen und Stative sind natürlich nicht zum Nulltarif zu

bekommen. Wer nicht viel ausgeben will, kann die nötigen Investitionen auf Maker-Art durch Eigenbau und *Upcycling* drücken. Mehr dazu stand bereits in vielen vergangenen Make-Ausgaben (siehe *Mehr zum Thema* in der Kurzinfor). Viele dieser Artikel kann man mittlerweile kostenlos im Volltext online lesen, zu erreichen über die Kurz-URL im Kurzinfor-Kasten.

Kamera und Stativ

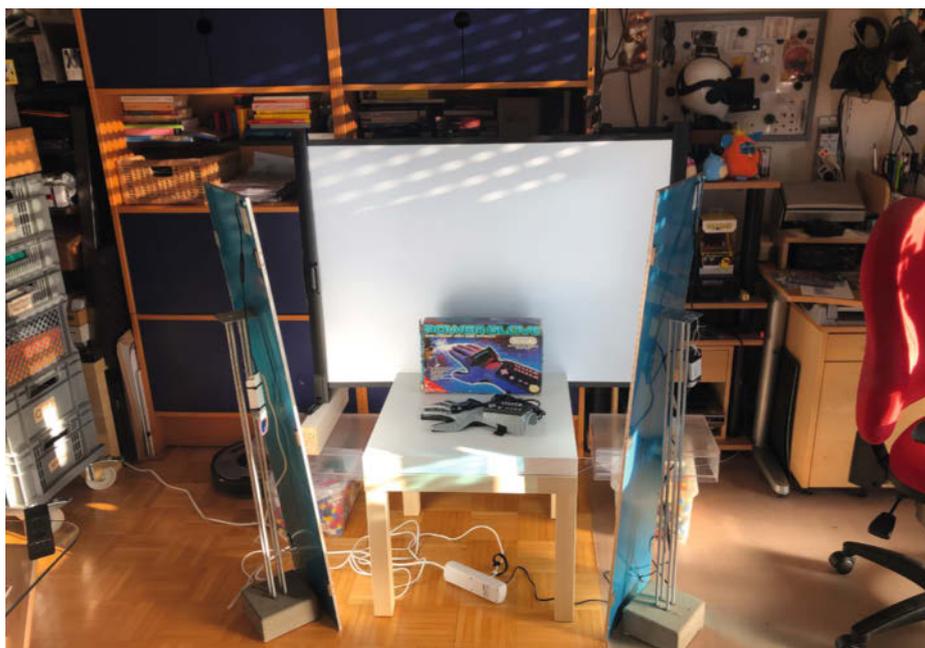
Wer keine Digitalkamera besitzt, muss nicht unbedingt eine kaufen, um Fotos der eigenen Basteleien zu machen. Ein aktuelles Mittel-

klasse-Smartphone reicht meist völlig, da die darin verbauten Kameras inzwischen sehr gute Qualität liefern. Ich selbst verwende meist eine kleine Digitalkamera, die ich sonst eher im Urlaub nutze und die von der Größe her noch in die Hosentasche passt.

Eine empfehlenswerte erste Anschaffung für das Fotografieren am Basteltisch ist ein Stativ. Das verringert zum einen das Verwackeln der Bilder und zum anderen kann man damit bei geringerem Licht immer noch gute Ergebnisse erzielen. Für die reine Schreibtischnutzung bietet sich zum Beispiel ein kleines, dreibeiniges Tischstativ an, das bereits ab rund 10 Euro zu bekommen ist. Mit entsprechendem Adapter für ein paar Euro kann man Fotostative ebenfalls für ein Smartphone verwenden 2.

Beim Handy kann das Schießen eines Fotos trotzdem noch verwackelt sein, wenn durch Tippen auf den Touchscreen oder den Lautstärke-Knopf ausgelöst wird. Um den stabilisierenden Effekt des Stativs nicht wieder zu verspielen, empfiehlt sich daher die Verwendung des Selbstauslösers in der Kamera-App.

Sofern man zum Basteln den Küchentisch oder Schreibtisch umfunktioniert und anschließend wieder in den Ursprungszustand zurückversetzen möchte, sind normale Stative die erste Wahl. Steht ein expliziter Bastelplatz oder ein Werkstatttisch zur Verfügung, habe ich mit festen Stativ-Halterungen gute Erfahrungen gemacht. Je nach Raumbeschaffenheit kann man diese an die Tischplatte, Regale oder Wand klemmen bzw. schrauben. Der Vorteil dabei ist, dass das Stativ beim Basteln keine Stellfläche benötigt, nicht im Weg steht und nicht für jedes Foto auf- und abgebaut werden muss. Zudem bekommen die Fotos der einzelnen Entstehungsphasen des Werkstücks immer den gleichen Blickwinkel, sodass die



4 Improvisiertes Foto-Setup mit umfunktionierten Decken-LED-Lichtern

Aufnahmen auch als Zeitraffer verwendet werden können.

Ist der Raum nicht allzu hoch und wenn man sich zudem an ihrem Anblick nicht stört, kann man die Stativhalterungen an der Decke befestigen. Hier gibt es für deutlich unter hundert Euro teleskopische, schwenkbare Halterungen zu kaufen, auch zu sehen im ergänzenden Video zu diesem Artikel (siehe Link in der Kurzinformatio). Gerade in einem engen, vollgestellten Bastelkeller ist das meiner Erfahrung nach eine gute Lösung. Denn diese Halterungen stören bei der Arbeit am wenigsten, da sie schnell nach oben geschoben werden können und keine Stellfläche benötigen.

Der Nachteil einer fest montierten Kamerahalterung ist, dass man nur einen einzigen, festen Blickwinkel auf das Projekt bekommt. Selbst wenn man mehrere Stative auf diese Weise angebracht hat, ist die Kamera immer fest auf jeweils einem davon verschraubt. Ein Losschrauben kann dann je nach Montagewinkel umständlicher und zeitraubender sein, als ein normales Bodenstativ umzustellen. Nach dem anschließenden Wechsel zurück und erneutem Festschrauben der Kamera sitzt diese oft mit leicht unterschiedlichem Winkel darauf. Das macht den Vorteil des kontinuierlichen, festen Blickwinkels wieder zunichte.

Zur Lösung dieses Problems bieten sich statt der herkömmlichen Verbindung mit Stativschrauben sogenannte *Schnellwechsellplatten* (auch Kameraplatten genannt) an. Diese bestehen aus zwei Teilen, je eines wird an Stativ und Kamera befestigt **3**. Durch einen schnell zu bedienenden Verriegelungsmechanismus können beide dann immer wieder im selben Winkel miteinander verbunden werden. Da in den Schnellwechsellplatten wiederum eine Standard-Stativschraube steckt, kann man sie auch für alles andere benutzen, was ein Stativgewinde hat, vom eben erwähnten Smartphone-Adapter bis hin zu manchen Leuchten (dazu gleich mehr).

Schnellwechsellplatten gibt es von verschiedenen Herstellern in unterschiedlichsten Preislagen und Qualitäten. Sehr hochwertig – aber auch kostspielig – sind dabei die Produkte von *Manfrotto*. Meiner Erfahrung nach genügen aber die Platten der günstigeren Hersteller, selbst wenn es dort bei der Passgenauigkeit durchaus mal etwas mehr Spiel gibt. Für den hier genannten Zweck wäre es eher attraktiv, darauf zu achten, dass man die Platten nicht nur zwingend als Paar erwerben kann – denn man benötigt zwar mehrere Platten für die Stative, aber nur eine für die Kamera.

Licht

Auf den ersten Blick bietet es sich an, das normale Tageslicht für Fotos zu verwenden: Es ist kostenfrei und immer verfügbar – könnte



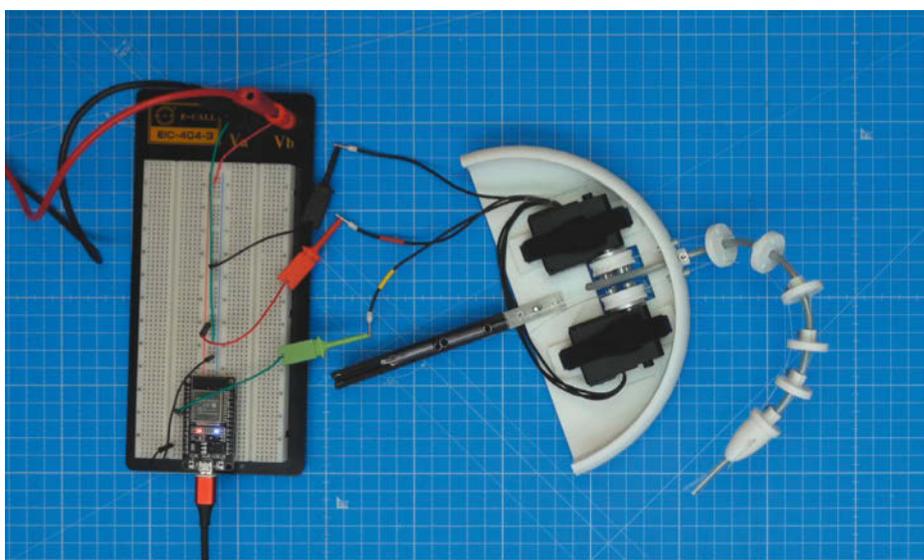
5 Fotolicht lässt sich selbst in einer vollen Werkstatt gut und preisgünstig unterbringen.

man meinen. Für ein einzelnes Foto vom Werkstück kann Sonnenlicht tatsächlich prima passen, aber bei einer Folge von Fotos über den Entstehungsprozess kann die Tageszeit oder eine Wolke das Bild komplett anderes wirken lassen. Dann ist Sonnenlicht eher unkontrollierbar und unzuverlässig. In einem Bastelkeller oder bei Arbeiten in den Abendstunden entfällt die Option „Tageslicht“ zudem automatisch.

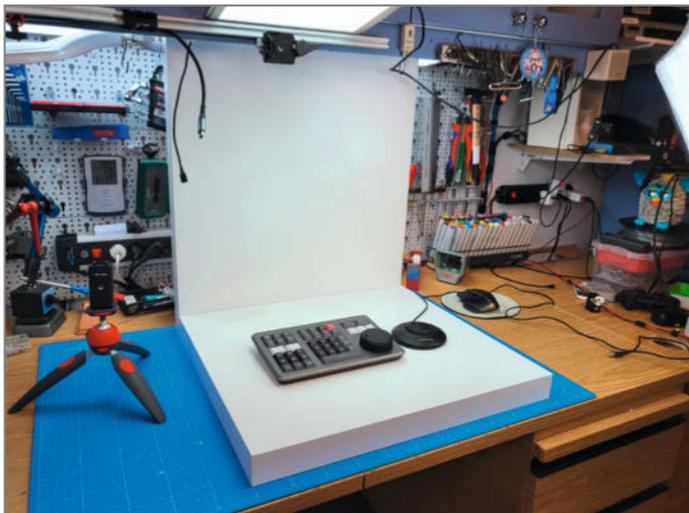
Oft kann es stattdessen attraktiv sein, den Raum zu verdunkeln und Tageslicht – wenn überhaupt – nur sehr begrenzt und indirekt beleuchtend zu nutzen. Dazu muss man nicht

unbedingt professionelles Fotolicht kaufen – normale Leuchten oder die Schreibtischlampe können ein guter Anfang sein. Ich hatte zum Beispiel das Glück, dass innerhalb meiner Verwandtschaft eine Anzahl LED-Flächenlichter zur Deckenmontage nicht gebraucht wurden und ich diese nach ein paar Jahren des Einstaubens geschenkt bekommen habe. Seitdem verwende ich sie immer wieder als mobile Leuchten für Fotos von meinen Projekten **4**.

In echtes Fotolicht (bzw. Videolicht) zu investieren, ist davon abgesehen bereits für



6 Auch eine große Schneideunterlage kann ein guter Hintergrund sein.



7 Zwei „Lack“-Tischplatten als Hintergrund



8 Schaumstoffplatten mit verschiedenfarbigen Seiten

wenig Geld möglich (zu selbstgebauten Alternativen siehe Artikelliste in der Kurzinfo). Am günstigsten sind hier Systeme mit Leuchtstofflampen, die aber beim Stromverbrauch nicht optimal sind und zudem sehr vorsichtig behandelt werden müssen. Hier kann man Sets mit zwei oder drei Leuchten (inklusive Softbox und passenden Stativen), verpackt in einer Tragetasche, schon für deutlich unter 100 Euro erwerben. Teurer, aber leistungsfähiger und technisch aktueller sind Ausführungen mit LEDs. Bei diesen kann man meist die Helligkeit dimmen, was bei den Leuchtstofflampen nicht möglich ist. Bei teureren LED-Modellen liegen meist noch Farbfilter bei oder man kann die Farbe sogar über die LED-Elektronik einstellen.

In den günstigen Preisklassen muss man sich aber darüber im Klaren sein, dass die me-

chanische Qualität nicht ansatzweise mit der Ausrüstung für Profifotografen vergleichbar ist. Ein regelmäßiges Auf- und Abbauen oder gar Transporte werden diese Produkte meiner Erfahrung nach nicht lange überleben. Für die statische Montage im Bastelkeller kann die Qualität aber vollkommen ausreichen 5.

Erfreulicherweise kann man Fotostative ebenfalls für Leuchten verwenden. Die Gewindegrößen für Kameras und Fotolichter sind zwar in der Regel unterschiedlich, können aber durch entsprechende Adapter angepasst werden. Wenn man jede Leuchte dann zusätzlich auf eine Schnellwechselplatte schraubt, kann man alles unkompliziert und beliebig miteinander kombinieren.

Den LED-Blitz des Smartphones – bzw. bei einer echten Kamera den eingebauten Blitz – verwende ich nach Möglichkeit übrigens nicht, da hier das Objekt meist von vorn überstrahlt wird und im schlimmsten Fall noch harte Schatten auf den Hintergrund geworfen werden.

Hintergründe oder Backdrops

In diesem Artikel geht es vor allem um Fotografieren von Projekten, die auf einem Schreibtisch oder einer Werkbank Platz finden. Bei so kleinen Objekten ergeben sich viele Möglichkeiten, den Hintergrund des Fotos mit wenig Aufwand zu variieren. Wer hingegen etwas in der Größe eines selbst konstruierten Elektromotorrads fotografieren möchte, muss für Beleuchtung und Hintergrund andere Entscheidungen treffen, für die hier (buchstäblich) kein Platz ist.

Kann das Werkstück zwischendurch nicht vom Tisch bewegt werden, weil es zum Beispiel gerade zerlegt ist, kann man direkt auf einer farbigen Schneideunterlage arbeiten 6. Diese gibt es in unterschiedlichen Farben und bis zur Größe DIN A2, was in der Regel sowohl

zum Basteln als auch zum Fotografieren groß genug ist. Wenn die Unterlage durch die Arbeit viel verschmutzt wird, kann man stattdessen eine große Papierblock-Unterlage verwenden und das oberste Blatt je nach Verschmutzung einfach von Zeit zu Zeit abreißen.

Kann das Motiv hochgehoben werden, habe ich mit den Platten von zwei weißen IKEA-Lack-Tischen gute Erfahrungen gemacht: Eine Platte wird als Unterlage verwendet und die zweite senkrecht dahinter aufgestellt 7. Die entstehende Schattenkante empfinde ich dabei nicht als störend, sondern je nach Beleuchtung und Fotowinkel sogar als aufwertenden Effekt, weil der Raum ein wenig definiert wird (zu sehen auf Bild 2). Beide Tische kosten zusammen knapp 16 Euro und benötigen verstaub anschließend kaum Platz.

Wenn es noch unempfindlicher und farbiger sein soll, kann man zwei Hartschaumstoffplatten verwenden. Im gezeigten Beispiel 8 haben diese auf jeder Seite unterschiedliche Farben, sodass man damit gleich mehrere Farbszenarien erzielen kann.

Sind Motive etwas höher, bieten sich sogenannte Roll-Up-Banner an 10. Diese sind je nach Anbieter und Ausführung schon für unter 50 Euro erhältlich und ursprünglich als Werbeaufsteller für Messen oder Veranstaltungen gedacht. Zusammengerollt sind sie sehr klein, kommen ausgerollt aber problemlos auf bis zu 2,20 Meter Höhe bei 2 Metern Breite. Ihrer Natur als Werbemittel geschuldet kann man bei der Bestellung direkt sein eigenes Fotomotiv für das Banner hochladen. So bekommt man sehr preiswert einen großen Fotohintergrund mit individuellem Motiv wie zum Beispiel dem eigenen Logo. Die Tauglichkeit als Fotohintergrund haben einige Hersteller inzwischen erkannt und bieten bereits fertig konfektionierte Roll-Up-Banner mit abstrakten oder farbigen Hintergründen an 9.



9 Ein Roll-Up-Banner findet überall Platz ...

Wer eine mobile Beamerleinwand besitzt, kann diese ebenfalls als weißen Hintergrund für seine Fotos verwenden ¹¹. Gleiches gilt für einen *Greenscreen*, den man während der Pandemie vielleicht für Videokonferenzen angeschafft hat. Ich selbst habe ein Modell von *Elgato* im Einsatz, das technisch wie ein Roll-Up-Banner konzipiert ist und mich in Punkt Knitterfreiheit, Mechanik und Preis überzeugt hat. Ist man bereit, etwas mehr Platz und Geld investieren, kann man bereits für unter 200 Euro einen abrollbaren Hintergrund zur Wand- oder Deckenmontage mit bis zu 3 Metern Breite bekommen. Auch ein weißes Rollo kann hier gute Dienste leisten.

Zu weiteren Details und Eindrücken zu den Aspekten Hintergründe, Stativpositionen und Licht gibt es ein ergänzendes YouTube-Video, zu erreichen über den Link in der Kurzinfo.

Show & Tell

Die Fotos sind jetzt gemacht, aber man hat sie nicht immer dabei. Zu Besuch bei Freunden stellt sich meiner Erfahrung nach jedenfalls oft heraus, dass ich gerade genau die zum jeweiligen Gespräch passenden Fotos noch nicht oder nicht mehr auf dem Smartphone gespeichert hatte. Daher habe ich begonnen, meine Projekte öffentlich in Form eines Blogs zu dokumentieren, auf das ich und andere an jedem Ort und zu jeder Zeit zugreifen können.

Als positiver Nebeneffekt bringt es Aufmerksamkeit und Feedback. Oft entsteht darüber hinaus Kontakt zu anderen Makern und Gleichgesinnten, was dann wiederum Ansporn für weitere Projekte oder Veröffentlichungen sein kann. Auch Artikel in Zeitschriften oder Vorträge auf *Maker Faires* bzw. in lokalen Makerspaces können das eigene



¹⁰ ... und bietet sich bei etwas höheren Objekten an.

Bastlernetzwerk vergrößern und wertvolle Anregungen liefern. Für all das sind aber natürlich die – bei mir inzwischen routinemäßig beim Bauen entstehenden – Fotos die zwingende Voraussetzung.

Video-Aufnahmen und YouTube

Die reine Text-/Bilddarstellung eines Blogs stößt gerade bei interaktiven oder beweglichen Projekten oft an ihre Grenzen. Der

logische nächste Schritt war daher für mich, Videos zu publizieren. Innerhalb der verwendeten Blog-Software kann das durchaus eine Herausforderung sein, daher bin ich zum Hosting auf *YouTube* übergegangen. In die Artikel meines Blogs einbinden kann ich die so gespeicherten Videos dennoch.

Während man ein Foto in der Regel zwar beschneiden, aber nicht unbedingt besonders bearbeiten muss, kommt man bei eigenen Videos kaum um eine Videoschnittsoftware herum. Hier konnte ich in der Vergangenheit

TECHNIKUNTERRICHT MACHT ENDLICH SPAß!



Make:Education

Mit **Make Education** erhalten Sie jeden Monat kostenlose Bauberichte und Schritt-für-Schritt-Anleitungen für einen praxisorientierten Unterricht:



Für alle weiterführenden Schulen



Fächerübergreifend



Digital zum Downloaden



Monatlicher Newsletter

Jetzt kostenlos downloaden:

make-magazin.de/education



12 Jog-Dial-Geräte helfen beim Videoschnitt: DaVinci Speed Editor (links) und Contour ShuttleXpress (rechts)

einige Erfahrung mit professionellen Programmen (z.B. *Premiere* oder *Vegas*), sowie Consumerprodukten wie *Magix Video Maker* sammeln. Mein aktueller Favorit für den Videoschnitt ist *DaVinci Resolve*, das als kostenfreie Version erhältlich ist (Download siehe Link in der Kurzinfor). Diese kann sowohl privat als auch kommerziell verwendet werden und verfügt über einen beeindruckenden Funktionsumfang. Bestimmte Profifunktionen sind zwar der kostenpflichtigen Studio-Version vorbehalten, meiner Erfahrung nach stößt man mit einem Maker-Video aber nicht an diese Grenzen. Für den Einstieg gibt es direkt vom Hersteller umfangreiches (wenn auch durchweg englisches) Lehrmaterial, ebenfalls gratis.

Möchte man etwas häufiger Videos bearbeiten, kann sich die Anschaffung eines *Jog-Dial*-Eingabewerkzeugs lohnen **12**. Statt mit Maus oder Tastatur bewegt man sich damit mittels Drehung eines Rades durch die Filmclips. Neben dem spürbar erhöhten Komfort spart man damit viel Zeit. Gute Erfahrungen habe ich dabei mit dem *Contour ShuttleXpress* gemacht, der für knapp 50 Euro erhältlich ist. Wenn man mit dem Sprung ins (Semi-)Profilage liebäugelt, kann man sich alternativ den vielfach teureren *DaVinci Speed Editor* zulegen, der hochwertiger konstruiert ist und mehr Möglichkeiten bietet. Abgesehen vom Preis würde ich Gelegenheitsnutzern den *Speed Editor* aber auch deshalb nicht empfehlen, da

das Gerät primär für die *Cut*-Ansicht statt für den *Edit*-Modus gedacht ist und seine Vorteile eher bei der Sichtung umfangreicheren Materials ausspielt. Zudem arbeitet der *Speed Editor* ausschließlich mit dem Schnittprogramm *DaVinci Resolve* zusammen. Mir persönlich dreht das Rad beim *Speed Editor* zu frei, während es beim *ShuttleXpress* hingegen spürbar einrastet und damit einzelne Frames haptisch fühlbar macht.

Wenn man als Maker von Foto zu Video „upgraded“, ergeben sich so viele neue Möglichkeiten, dass man damit leicht einen eigenen Artikel nur zu diesem Thema füllen könnte – falls Sie sich so einen für eine der kommenden Ausgaben wünschen, schreiben Sie der Redaktion bitte eine Mail an mail@make-magazin.de. Zwei Anregungen möchte ich hier aber dennoch kurz geben:

Man kann die Kamera bereits beim Basteln bzw. Auf- und Abbau mitlaufen lassen. So hat man zum Beispiel später Videomaterial für Zeitraffer-Szenen. Wenn dabei unerwartete Unfälle wie magischer, blauer Rauch gefilmt werden, kann man das Video dadurch aufheichern und es wirkt authentischer und persönlicher.

Wer noch eine zweite Kamera oder ein Smartphone zur Verfügung hat, kann mit zwei Kameras gleichzeitig filmen und dann später im Schnitt zwischen den Blickwinkeln wechseln. Das kann eine längere Szene abwechslungsreicher erscheinen lassen, als wenn man sie nur aus einer einzigen Perspektive zeigt.

Zum Einstieg in das Thema Videoschnitt und vertiefende (Profi-)Techniken bietet YouTube zahlreiche Videos. Ich persönlich finde dabei den YouTube-Kanal von Andreas Abb sehr unterhaltsam und gut gemacht.



11 Eine mobile Beamer-Leinwand als Hintergrund

Mitmachen

Hat man erst einmal mit dem Publizieren der Bastelprojekte begonnen, spornen Feedback und Networking dazu an, weiterzumachen. Für mich persönlich hat der beschriebene Weg vom Blog über YouTube dazu geführt, dass ich viele meiner Baupläne, Software und Modelle der Maker-Community als Open Source zur Verfügung stelle. Neben den erwähnten Fotos und Videos gehören dazu eingescannte Zeichnungen oder *Fritzing*-Schaltpläne – also alles, was anderen oder später mir das Nachvollziehen des Aufbaus ermöglicht. Ohne all dies hätte ich sicher nicht meine Projekte auf einer Maker Faire ausgestellt oder angefangen, Artikel wie diesen hier zu schreiben. Ich kann daher jeden nur ermutigen, ebenfalls diesen ersten Schritt zu gehen und zum Beispiel über Social Media oder auf YouTube einfach mal etwas über sein aktuelles Projekt zu posten. Und natürlich auch der Make-Redaktion davon zu berichten.

—pek

**JETZT
KOSTENLOS
TESTEN**

DIE NEUE LERNPLATTFORM FÜR IT-PROFESSIONALS

Wir machen IT-Weiterbildung digital

IT-Kurse aus der Praxis

Lerne in Online-Kursen und -Trainings, wie Techniken funktionieren und wie du Aufgaben löst.

Triff erfahrene IT-Experten

Profitiere von der Erfahrung unserer IT-Experten und hole dir hilfreiches Praxiswissen aus erster Hand.

Lerne, wie es für dich passt

Nutze das Kursangebot überall und auf jedem Gerät und lerne immer dann, wenn du es brauchst.

Übungen zum Ausprobieren

Probiere das Gelernte selbst aus – mit Beispielaufgaben, Coding-Segmenten und Praxisübungen.

Überprüfe dein neues Wissen

Teste das Gelernte mit interaktiven Quizzes und löse die Programmieraufgaben deiner Trainer spielerisch.

Individuelle Lernumgebung

Lerne in deinem eigenen Tempo, inklusive Notizen, Transkript und Fragen-Modul.

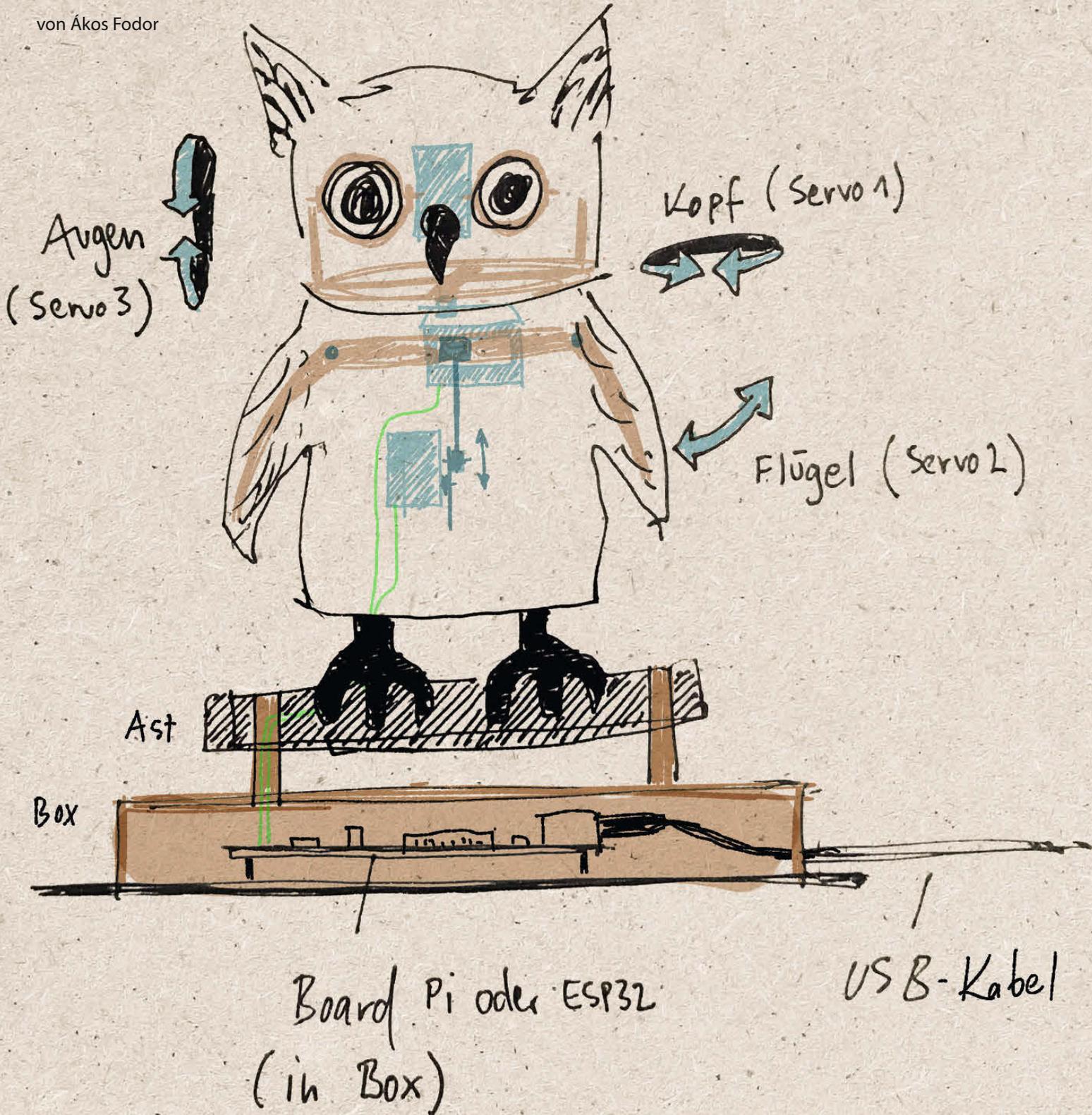
Hier geht's zu deiner Weiterbildung: heise-academy.de



Making-Of Posteule

Wie man eine animatronische Posteule baut, habe ich bereits in den Make-Ausgaben 1/22 und 2/22 gezeigt. Dieser Erfahrungsbericht wirft einen Blick auf ihre Entstehung und die damit verbundenen Herausforderungen und Entscheidungen, die mir unterwegs begegnet sind.

von Ákos Fodor



Animatronik fasziniert mich seit meiner Kindheit. Ich habe mir schon oft vorgenommen, selbst so ein Projekt umzusetzen. Aber andererseits gab es auch immer eine passende Ausrede. Kaum bin ich bei der Make, schubst mich mein Chefredakteur ins kalte Wasser und sagt: „Gute Idee, mach mal! Du hast acht Wochen Zeit.“

Da stand ich nun, mit einer Skizze in der Hand und dem festen Ziel, eine Plüschhülle zum Leben zu erwecken. Sie sollte als Einstieg in das Thema Animatronik dienen, nützlich und erweiterbar sein – und vor allem nachbausicher, denn ich baute sie ja nicht nur für mich allein. Ein erstes Projekt dieser Art unter erschwerten Bedingungen zu bewältigen, schreckte mich nicht ab. Ich war zuversichtlich, dass mein Produktdesign-Studium und die zehn Jahre CAD- und 3D-Druck-Erfahrung mir helfen würden, die kommenden Hürden zu überwinden. Nur bei der Elektronik war ich mir nicht sicher, aber optimistisch genug, um loszulegen. Also bestellte ich die benötigten Komponenten und begann schon einmal zu programmieren.

Mithilfe der *IMAPClient*-Bibliothek für den Raspberry Pi stellte ich schon zu Beginn des Projektes sicher, dass der E-Mail-Abruf funktionierte. Später würde ich die Bewegungen der Eule und Sounds damit verknüpfen. Als mich die ersten Komponenten erreichten, begann ich damit, die Plüschhülle vorsichtig auseinanderzunehmen. Das fühlte sich auf eine verquere Art martialisch und endgültig an. Und als die Augen und die leere Plüschhülle vor mir lagen, hegte ich auch kurz Zweifel, ob sie je wieder niedlich aussehen würde. Daher wollte ich so schnell wie möglich das Gesicht rekonstruieren.

Vorbereitungen

Zuerst musste ich jedoch einige Vorbereitungen treffen, wie die in sich zusammengefallene Plüschhülle auszumessen. Dazu verwendete ich ein einfaches Trinkglas, das ich in den Plüsch steckte und maß dann den Durchmesser an den Öffnungen sowie die Höhe von Kopf und Rumpf. Jetzt wusste ich, welche Volumina mir zur Verfügung standen. Danach maß ich mit dem Messschieber die eingetragenen Elektronik-Komponenten und baute sie im CAD-Programm *Rhinoceros* nach **1**. Dazu zählten der Servo-Controller, der Audioverstärker, die Servos, der Lautsprecher samt Löchern und Anschlüssen sowie die DeLock-Kupplung – das Modell eines Raspberry Pi 3B hatte ich noch aus einem vorherigen Projekt übrig. Natürlich wäre es einfacher gewesen, fertige 3D-Daten aus dem Internet zu laden. Da ich mich jedoch selten auf deren Präzision verlassen kann, wollte ich aufgrund der knappen Zeit böse Überraschungen vermeiden.

Kurzinfo

- » Entstehung eines nachbausicheren Projekts
- » Hintergründe und Lösungsansätze
- » Stolpern, aufstehen, Krönchen richten und ins Ziel laufen

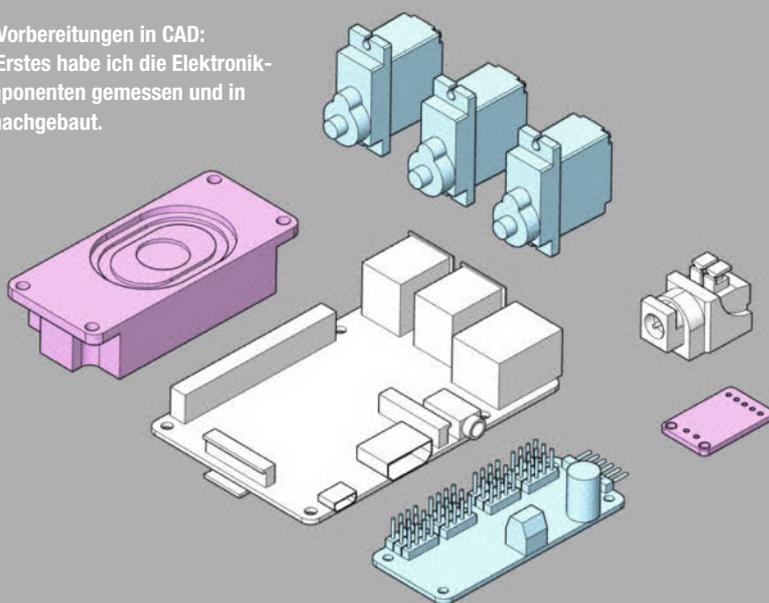
Mehr zum Thema

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

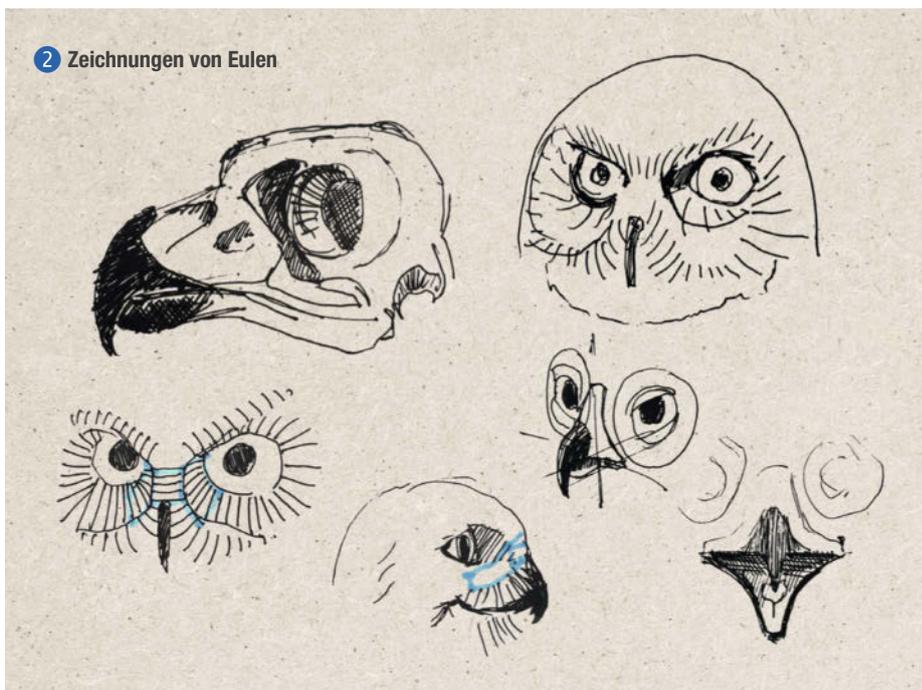
Alles zum Artikel
im Web unter
make-magazin.de/x5yy



- 1** Vorbereitungen in CAD:
Als Erstes habe ich die Elektronik-Komponenten gemessen und in 3D nachgebaut.



- 2** Zeichnungen von Eulen





3 Aus zwei Augen und einem Schnabel entsteht schnell ein Gesicht.



4 Das leichte Design des Schädels entwickelte sich Schritt für Schritt.

Anatomie der Posteule

Um meiner räumlichen Vorstellungskraft auf die Sprünge zu helfen, suchte ich mir zunächst anatomisches Bildmaterial von Eulenschädeln und zeichnete sie mit und ohne Gefieder, denn ich musste ungefähr wissen, welche Abstände die Augen zum Schnabel haben oder wie weit dieser herausstehen sollte 2. Dann begann ich mit der Gesichtsrekonstruktion in CAD. Da Eulen ihre Augen von Natur aus nicht bewegen können, musste ich zum Glück erstmal nur darüber nachdenken, wie die Augenlider und der Schnabel auf- und zuklappen würden. Um dem Gesichtsausdruck etwas Intelligenz zu verleihen, drehte ich die Augen um ein paar

Grad, sodass die Eule etwas kritisch, aber nicht böse schaute 3. Zahnstocher und Schaschlikspieße hatten mir als Achsenverbindungen bereits in der Vergangenheit gute Dienste geleistet. Daher verwendete ich sie auch hier. Vor allem, wenn geringe mechanische Kräfte wirken, sind sie völlig ausreichend für Prototypen und lassen sich mit einfachem Werkzeug schnell zurechtschneiden. Außerdem sind sie günstig und können leichter in Löcher gezwängt werden als etwa vergleichbare Stäbe aus Edelstahl.

Nachdem ich mit dem Gesicht zufrieden war, konstruierte ich als Nächstes den Schädel. Das am Ende in den Artikeln beschriebene Stecksystem, aus dem dieser besteht, existiert

seit den ersten Entwürfen. Ich wollte möglichst viel Flexibilität bewahren, um Elemente während der Entwurfsphase leicht verschieben zu können. Die genaue Anordnung der Augen wurde z.B. durch die Löcher im Plüsch bestimmt und ich musste sie einige Male versetzen, bis sie richtig saßen. Um den Materialverbrauch und die Druckzeiten der Prototypen gering zu halten, beschloss ich, ein Gerüst zu entwerfen, das leicht und stabil war 4. Es bot sich an, dieses aus mehreren, liegend gedruckten Teilen zusammenzustecken. Für den finalen Schädel hatte ich ursprünglich eine realistischere Anmutung geplant. Bei dem, was mich später noch erwarten sollte, bin ich rückblickend jedoch froh, das Design so gelassen zu haben.



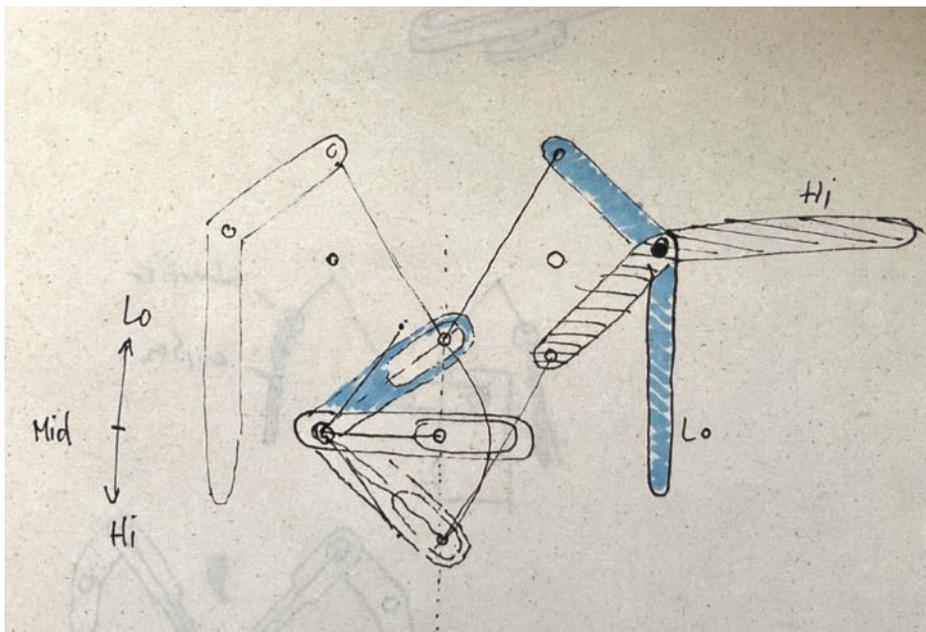
5 Entwicklungsstufen des Rumpfs: Die filigranen Teile erforderten einige Tests und Anpassungen.

Weniger Platz als gedacht

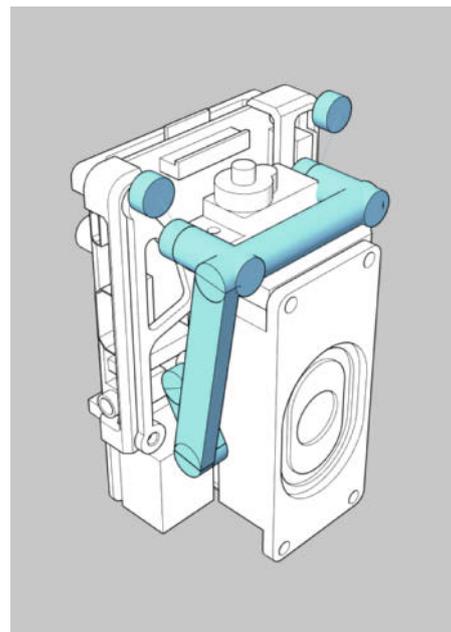
Im Rumpf begegneten mir bei der Anordnung der Elektronik-Komponenten die ersten Platzprobleme. Denn wo der eine Servo perfekt saß, hatte der Lautsprecher plötzlich keinen Platz mehr, drückte an einer anderen Stelle gegen den Raspberry Pi oder versperrte woanders die Achse eines Servos. Besonders in dieser Phase zahlte es sich aus, dass ich die Komponenten als CAD-Modelle vorbereitet hatte, denn an vielen Stellen entschied ich wenige Millimeter über die endgültige Anordnung.

Nachdem sich mein Kopf und die Komponenten ein paarmal im Kreis gedreht hatten, fand ich ein passendes Layout und begann, die Löcher der einzelnen Platinen konstruktiv miteinander zu verbinden.

Wie beim Schädel wollte ich auch hier möglichst viel Material einsparen, allerdings eher aus Platzmangel. Außerdem sollte später alles in ein schützendes Gehäuse passen. Je kompakter ich also baute, desto mehr Luft hätte ich später für eventuelle Korrekturen und die Kühlung 5. Mit der Stabilität im Blick konstruierte ich einen Großteil der Durchgangs-



6 Die ursprüngliche Idee für den Flügelmechanismus: Der Servo in der Mitte bewegt die Flügel links und rechts.



7 Ich tastete mich mit Volumenkörpern an die Konstruktion heran.

löcher rechtwinklig zum Druckbett, damit die Ebenen später nicht Gefahr liefen, von Schrauben auseinandergerissen zu werden. Gerne hätte ich auch auf Stützkonstruktionen beim Druck verzichtet. Da jedoch so viele Parameter zu beachten waren, nahm ich in Kauf, dass ich welche benötigen würde.

Ein Servo, zwei Flügel

Die erste mechanische Herausforderung begegnete mir mit dem Mechanismus, der die Flügel bewegen sollte 6. Aus Platzmangel hatte ich den antreibenden Servo um 90 Grad drehen müssen, sodass er jetzt zur Seite und nicht, wie geplant, nach vorne zeigte. Zudem befand er sich auf Bauchhöhe und mehrere Zentimeter von den Flügeln entfernt, umliegend blockiert von den übrigen Komponenten. Ich musste also mit einer entfernten und einseitig ausgerichteten Achse eine symmetrische Bewegung für zwei Flügel erzeugen 7. Dafür verlagerte ich zuerst mit einer Verlängerung die Achse des Servos nach oben in den Brustbereich. Danach spiegelte ich die Bewegung mit einer Art Brücke auf die andere Seite und verband sie mit beiden Flügeln. Weil die Brücke jedoch zu viel Spiel hatte, stabilisierte ich sie mithilfe einer Führungsschiene. Als die Flügel nach ein paar Varianten 8 reibungslos flatterten, schickte ich vor Freude sofort ein Video an meine Kollegen.

Ein knappes Rennen

Von den ursprünglichen acht Wochen bis zum Druckschluss waren jetzt nur noch zwei übrig. Die Mechanik wirkte solide und ich war zu-



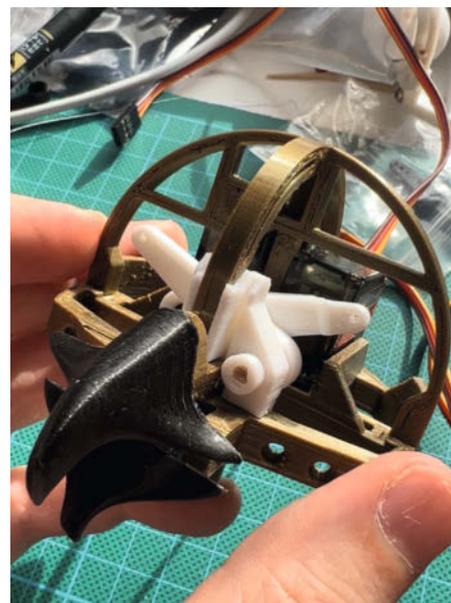
8 Einige Entwicklungsstufen des Flügel-Mechanismus

versichtlich, dass sie funktionieren würde. Auch erste Tests mit der Elektronik verliefen zufriedenstellend. Es gab allerdings noch einige Feinheiten zu klären. Und ich musste noch Grafiken erstellen und den Aufbau beschreiben. Offenbar hatte ich mich mit dem Zeitaufwand verschätzt.

Also beschlossen mein Chefredakteur und ich, den Artikel aufzuteilen und den bisherigen Stand zu veröffentlichen. Die Programmierung sollte in der darauffolgenden Ausgabe erscheinen. Am 10. Februar war es dann so weit. Der erste Teil meines Projektes erschien. In der Make! Auf der Titelseite! Unglaublich! Jetzt fehlte nur noch der zweite Teil, aber was sollte schon schiefehen?

Der Teufel steckt im Detail

Nach ein paar euphorischen Tagen setzte ich mich wieder an den Entwurf. Da erwachte es mich eiskalt: Die Augenlider ließen sich nicht wie erwartet ansteuern. Anscheinend hatte



9 Der erste Scheren-Mechanismus

ich mich in meinem Gestaltungswahn selbst in eine missliche Situation manövriert, denn die Schrägstellung der Augen führte zu ebenso verdrehten Rotationsachsen bei den

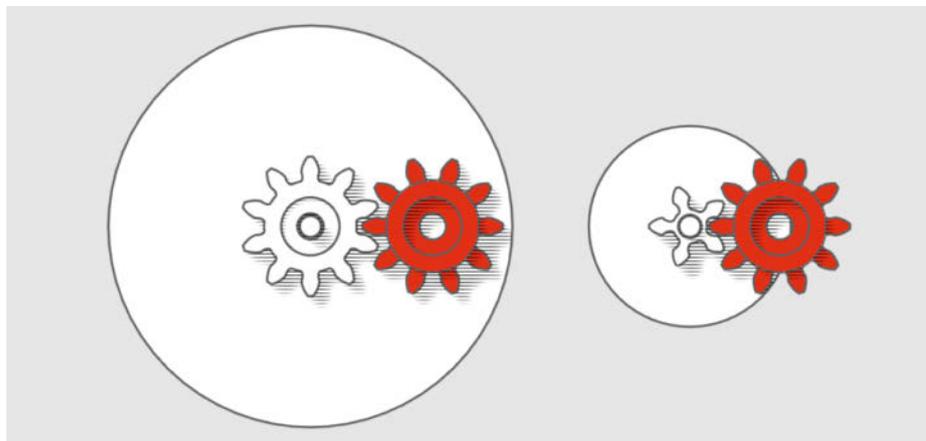
Augenlidern, die sich auch noch gespiegelt bewegten. Als ich das eine Lid mit dem Servo verband, stellte ich fest, dass der Abstand zwischen Lid und Servo-Arm variierte, wäh-

rend sich beide bewegten. Dadurch verkeilte sich der verbindende Draht, der Servo fiepte und lief heiß. „Alles halb so wild“, sagte ich mir, „es ist nur ein mechanisches Problem“, und setzte mich umgehend an eine neue CAD-Konstruktion. Dafür musste es doch eine Lösung geben, ohne bestehende Teile zu ersetzen. Einige Leser hatten schließlich schon mit dem Nachbau begonnen.

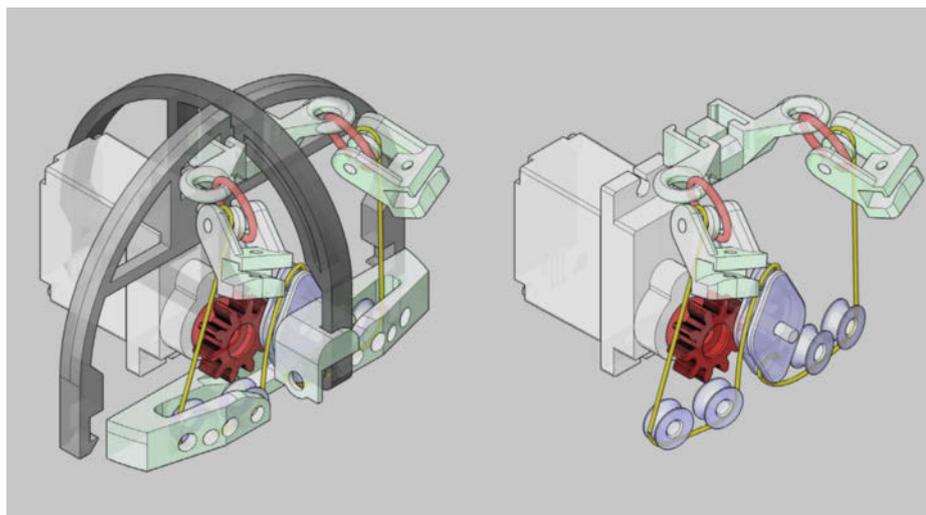
Die Zeit lief. Der Raum hinter den Augen war so klein, dass meine Konstruktionen ständig mit den Schädel-Bauteilen kollidierten oder nur für ein Auge funktionierten. Ein Balken wie beim Flügel-Mechanismus ließ sich aufgrund dessen nicht auf die Augen übertragen. Um die Bewegungen zu spiegeln, erschien mir ein Scheren-Mechanismus am vielversprechendsten. Dieser würde mithilfe einer Mittelachse die Bewegung des einen Lids auf das andere übertragen ⁹.

Und tatsächlich: Beide Augen bewegten sich, aber das eine hing immer leicht hinterher, weil der gespiegelte Scheren-Arm zu viel Spiel hatte. Außerdem gestaltete sich das Verdrahten schwieriger als gedacht und war nicht ansatzweise nachbausicher. Ich entwickelte meine Idee also weiter, nur leider wurde der Entwurf dadurch noch komplizierter. Als ich meinem Kollegen Heinz von den Herausforderungen berichtete, fragte er mich, ob ich schon über eine Lösung mit Gummibändern nachgedacht hätte. Diese könnten doch die Augen geschlossen und auf Spannung halten. Das klang genial und viel einfacher als meine bisherigen Überlegungen. Es fehlte nur etwas, das in die andere Richtung zog, um sie zu öffnen. Vielleicht ein Faden mit einer Rolle, die ihn aufwickelt? Diesen Trick hatte ich vor einigen Jahren bei der *Stan Winston School* gelernt. Einen Versuch war es wert. Ich musste nur noch lösen, wie ich die Fäden umlenken würde.

Als Erstes übersetzte ich mit Zahnrädern die seitlich sitzende Servo-Achse 1:1 auf eine Rolle, die ich mittig zwischen den beiden Augen angeordnet hatte. Da mein Servo sich jedoch nur 180 Grad weit drehte, bewegte sich die Rolle auch nur halb und ragte aufgrund ihres Durchmessers außerdem nach unten in den Schnabel-Mechanismus. Also verkleinerte ich die Rolle, sodass ich ihren vollen Umfang nutzen konnte. Jetzt drehte sie sich einmal um sich selbst, wenn der Servo eine 180-Grad-Drehung vollzog ¹⁰. Nachdem ich noch die Anzahl und Größe der Zahnräder optimiert hatte, griffen Servo und Rolle nahtlos ineinander. Ich freute mich und rannte in das nächste Hindernis: Obwohl die Fäden wenig Raum beanspruchten, musste ich den Schädel vollständig zerlegen, um sie zu befestigen ¹¹. Und als die Augenlider sich auch hier nicht richtig bewegten, wurde ich langsam unruhig. Die Zeit lief davon. Ich hatte noch drei Wochen bis Druckschluss des zweiten Teils.



10 Links die große Rolle, rechts die kleinere Version, die sich doppelt so weit dreht wie die Servo-Welle (rot).



11 Das Rollensystem, mit dem der Faden umgeleitet wurde, war zu kleinteilig.



12 Der Scheren-Mechanismus veränderte sich mit jedem Test.

13 Optimierung des Scheren-Mechanismus:
Durch das Spiegeln der Scheren konnte ich die Mittelachse verstärken. (Perspektive von oben, die Augen schauen nach links)

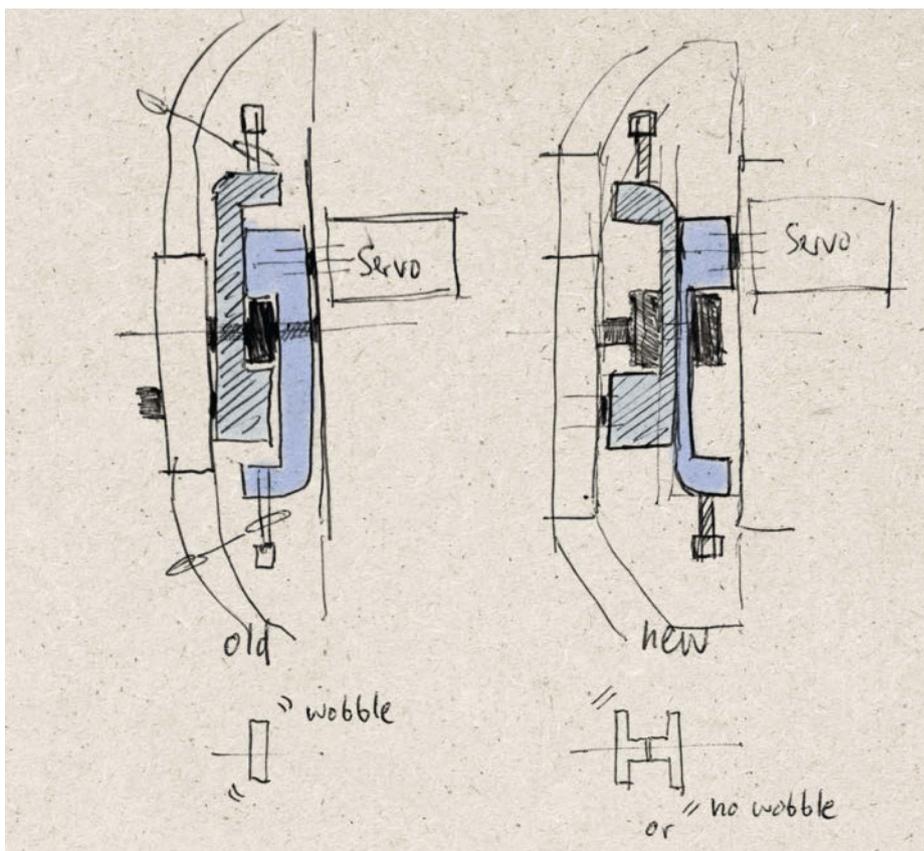
Vorwärts, rückwärts? Nein, beides!

Mein Scheren-Mechanismus hatte bis jetzt am besten funktioniert. Es kostete mich jedoch einiges an Überwindung, den Schritt zurück zu wagen und auch als möglichen Schritt nach vorn zu begreifen. Schließlich hatte ich ja nicht ohne Grund nach Alternativen gesucht. Umso schöner war es, als ich erkannte, dass meine Fehlversuche mir die fehlenden Puzzleteile lieferten **12**. Ich war jetzt nicht nur schneller, sondern löste die Anbindung des Servos an die Augenlider mit einem gedruckten Bauteil, anstatt mit Draht und stabilisierte das hängende Auge mithilfe einer verbesserten Ausrichtung der Scheren **13** und einer zusätzlichen Führungsschiene, die ich hinter dem Schnabel befestigte. Diese Idee hatte bei den vorherigen Anläufen nicht funktioniert. Jetzt passte sie.

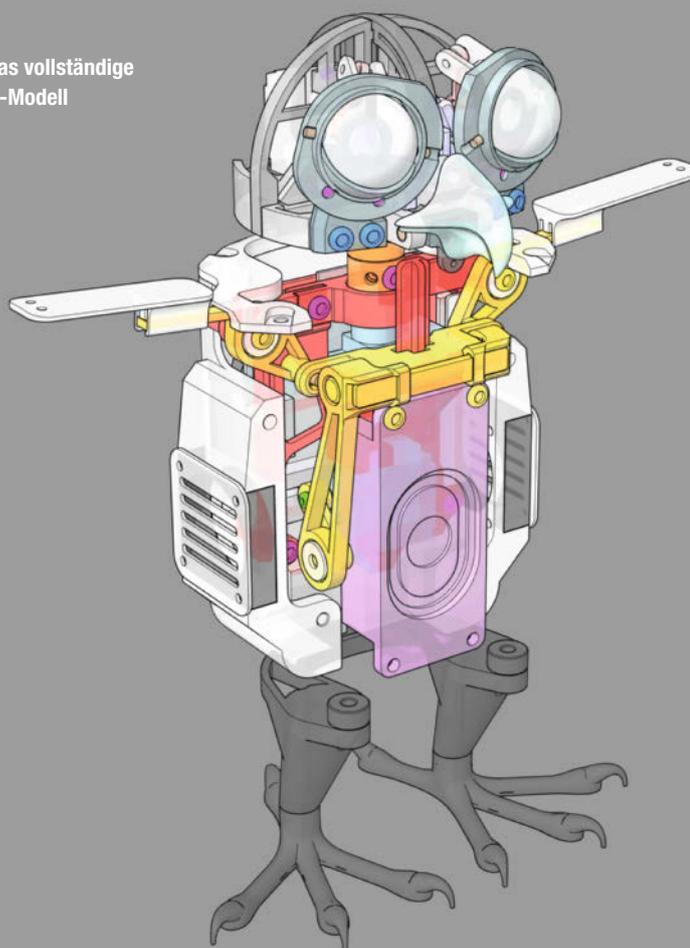
Die Bewegung lief also endlich zuverlässig. Nur der Zusammenbau erforderte unfassbares Fingerspitzengefühl, weil die Bauteile so klein waren und während des Zusammenschraubens ständig durch die Gegend flogen. Ich wollte mich davon aber diesmal nicht bremsen lassen. Also konstruierte ich noch eben ein kleines Werkzeug, das beim Zusammenschrauben die filigranen Teile zusammenhielt und drei winzige Hände ersetzte. Perfekt! Jetzt konnte ich mich endlich den übrigen Aufgaben widmen, die in der Zwischenzeit geduldig auf mich gewartet hatten. Das Gehäuse war seit der ersten Anleitung weitestgehend fertig. Es fehlten jedoch noch die Beine in CAD sowie die endgültige Verkabelung und Verkleidung samt Anleitung. Zwei Wochen hatte ich noch bis Druckschluss. Das beschränkte zwar den Experimentierspielraum, genügte jedoch für eine Punktlandung.

„Fertig!“, sagt die Eule

Als am 7. April die Make 2/22 erschien, war ich ziemlich erschöpft, aber auch sehr glücklich. Ich hatte es geschafft, die Eule war fertig **14**. Sie bewegte sich, flatterte und piepste – einfach so. Und wenn ich ganz ehrlich bin, vielleicht vor allem, weil sie fertig werden musste. Denn allen Herausforderungen zum Trotz hatten bislang anscheinend nur ein kleiner Schubs und ein gesetzter Rahmen gefehlt. —akf



14 Das vollständige Eulen-Modell



Tipps & Tricks

MSLA-Druck ist beliebt für hoch detaillierte Modelle und Prototypen. Das Druckerhandling, die Reinigung, Aushärtung und die Entsorgung von Abfall wird jedoch oft unterschätzt. Unser Artikel gibt ein paar Hilfestellungen.

von Carsten Wartmann

Werkzeuge und Hilfsmittel, die beim Drucker mitgeliefert werden, sind oft von minderer Qualität und nur ausreichend für einen Probe-Druck. Vor dem Loslegen mit dem neuen Drucker sollten daher bessere Werkzeuge und Materialien bereitliegen und auch in Zukunft immer in ausreichender Menge vorhanden sein, um frustfrei zu drucken.

Sicherheit

Weil man es nicht oft genug sagen kann: MSLA-Druck mit Harz (*Resin*) hat ein hohes Gefährdungspotential für Mensch und Umwelt, daher diese Ratschläge, auch wenn diese natürlich in jedem Handbuch zum Drucker stehen sollten.

Handschuhe in passender Größe und Beschaffenheit: Nitril-Handschuhe **1** sind am besten, sie sitzen wie eine zweite Haut und es lässt sich gut mit ihnen hantieren. Sie lassen sich leicht ohne Puder anziehen und sind angenehm zu tragen, auch wenn man natürlich darunter etwas schwitzt. Latex und ähnliche



Handschuhe lassen zu viel von den Allergenen und Reizstoffen des ungehärteten Harzes durch.

Man kann Handschuhe auch mehrfach nutzen, wenn man sie vor dem Ausziehen mit einem in Isopropanol getränkten Tuch abwäscht, um Restspuren von Harz zu beseitigen. Neben der Gesundheitsgefahr möchte man keine Harz-Fingerabdrücke auf Drucken, der Einrichtung, Werkzeugen und auf dem Drucker hinterlassen. Im Zweifel aber lieber ein paar frische Handschuhe aus der Box holen: Das spart Frust, schont die Gesundheit und sorgt für sauberes Arbeiten.

Schutzbrillen schützen die Augen vor Harzspritzern, Isopropanol und vor Splintern beim Entstützen von bereits gehärteten Drucken. Aber eine Schutzbrille sollte an sich in jeder Werkstatt immer griffbereit sein, Bequemlichkeit kann hier sprichwörtlich ins Auge gehen.

Ein **Atemschutz** schützt die Lunge und auch die Nase vor Geruchsbelastung in der Werkstatt. Bei Atemschutzmasken sollte der Filter *Klasse A* oder *AP* (Gase und Partikel) sein und insbesondere bei längerer Exposition durch Harz-Dämpfe getragen werden. Natürlich muss die Maske gut und dicht sitzen und der Filter nach seiner Standzeit (bei der Gelegenheit auch den Filter im Drucker wechseln) getauscht werden. Halbmasken sind

besonders für Brillenträger besser geeignet und auch insgesamt angenehmer zu tragen und aufzusetzen.

Um nicht nur die eigenen Atemwege zu schützen, bietet es sich an, den Arbeitsbereich gut zu lüften, was aber im Winter oder mangels Fenstern auch ein Problem werden kann. Dann sollten Sie über eine Abluftanlage oder wenigstens einen Filter für die Raumluft nachdenken.

Ein **Luftfilter** für den Raum, z. B. der günstige Ikea-Förnünftig **2** mit zusätzlichem Aktivkohlefilter (wichtig!) ist eine gute Ergänzung für jede Werkstatt und die Auswahl jenseits von Ikea ist durch die Pandemie auch sehr groß geworden. Wenn ich in meiner Heimwerkstatt mit Resin drucke, so ist der Ikea-Filter bei Stufe 2 ausreichend, um den Raum in ein paar Stunden komplett von Harz-Gerüchen zu befreien. Auch beim Löten und staubigen Arbeiten eine echte Erleichterung.



Material

Nicht nur Einsteiger unterschätzen leicht den Materialbedarf für den Harzdruck, denn neben dem teuren Druckmaterial summiert sich der Verbrauch von Handschuhen bis Isopropanol doch erheblich auf.

Isopropanol, mit am besten 99,9% Reinheit, benötigt man kanisterweise: Es gibt auch mit Wasser waschbare Harze, aber das kontaminierte Wasser ist teils problematischer zu entsorgen als Isopropanol und darf natürlich ebensowenig in die Kanalisation oder Natur gelangen!

Kleine Gebinde bis 5l sind einfacher zu benutzen, aber überproportional teuer. Für das Umfüllen in kleinere Flaschen zum besseren Handling und zum Auffüllen der Waschbehälter sollte man sich einen zum Kanister passenden **Ausgießer** oder ein **Hahnventil** ³ kaufen. Durch die hohe Kohäsion des Kanistermaterials und des Isopropanol ist sonst eine Sauerei vorprogrammiert.

Küchentücher benötigt man andauernd, auch bei vorsichtiger Arbeitsweise. Oft reicht ein halbes Tuch: immer in Rollrichtung zerreißen. Harzgetränkte Tücher mit UV-Licht aushärten, sobald möglich (siehe UV-Taschenlampe). Achtung, den **FEP-Film** des Harztanks höchstens vorsichtig mit Küchentüchern austupfen, nicht reiben. Noch besser: Den Stoff von alten T-Shirts benutzen. Auf keinen Fall FEP mit Isopropanol reinigen, sondern mit warmem Wasser, Spülmittel und **Mikrofasertuch**, damit werden auch trübe Filme wieder klar.

Werkzeug

Hier nun ein paar Hilfsmittel und Werkzeuge, auf die ich nicht mehr verzichten möchte:

Ein geschärftes **Palettenmesser** kann bei sehr fest sitzenden Drucken (wie auch beim **FDM-Druck**) Abhilfe schaffen, ohne den Druck durch rohe Gewalt, die bei den recht stumpfen Plastikspateln nötig ist, zu beschädigen. Dabei muss das Palettenmesser ⁴ aber richtig angeschliffen sein und auch richtig herum angesetzt werden, um die Druckbettfläche nicht zu beschädigen! Angeschliffen wird nur von einer Seite, die Unterseite wird nur entgratet bzw. poliert, damit sie plan auf dem Druckbett aufliegt und die Gefahr von Kratzern minimiert ist. Dann vorsichtig und fast waagrecht unter den ersten Layer des Drucks gehen und vorsichtig vorarbeiten. Bei sehr großflächigen Drucken können ein paar Spritzer Isopropanol helfen. Dies unterwandert den Druck und hilft beim Ablösen.

Mit einer **UV-Taschenlampe** kann man Harz schnell härten. In Harz getränkte Tücher werden dann fest und lassen sich als Restmüll entsorgen. Gleiches gilt für Harzspritzer auf dem Tisch oder Reste von Harz. Ausgehärtetes Harz ist nicht mehr besonders problematisch und darf im Hausmüll entsorgt werden. Solch eine Lampe ist auch praktisch, wenn man



Druck-Harz dazu benutzt, um FDM-Drucke zu glätten oder etwas zu reparieren. Natürlich sollte man mit dem UV-Licht aufpassen und bei den Arbeiten damit eine geeignete Schutzbrille tragen, sowie das UV-Licht nicht auf unbedeckte Haut einwirken lassen. Achten Sie auch darauf, dass die UV-Lampe nicht benutzt wird, wenn der Drucker offen ist oder noch ungewaschene Drucke auf dem Tisch stehen.

Eine kleine **Zerstäuber-Flasche** ist praktisch, um schnell Isopropanol auf ein Tuch oder eine Oberfläche zu bekommen: Das geht viel schneller und sauberer als mit einer normalen Schraubverschlussflasche und hilft nicht nur hier in der 3D-Druck-Werkstatt, sondern auch beim Reinigen von Platinen nach dem Löten und in vielen anderen Fällen.

Lackfilter oder Trichter mit Einsätzen. Das Filtern ⁵ von wiederzuverwendendem Harz, in dem möglicherweise ausgehärtete Teile des Drucks oder der Stützen schwimmen, ist sehr wichtig. Diese Partikel können den Druck misslingen lassen oder noch schlimmer den FEP-Film punktieren und damit für eine

riesen Sauerei sorgen, die im schlimmsten Fall den Drucker beschädigt. Die Firmware vom *Prusa SL1(S)* bietet seit Neuestem auch einen Reinigungsdruck an.

Um Harz sowie Isopropanol für das Spülen zu sparen, lässt man das Harz zur weiteren Verwendung in den Drucktank tropfen. Druck oder bastelt man sich eine **Abtropfhilfe** ⁶, so kann die Druckplattform schräg aufgehängt werden und das Harz tropft so viel besser und schneller ab. Bei dem gesamten Prozess hilft eine höhere (Raum-)Temperatur, die das Harz flüssiger macht.

Hat man mehrere Harztanks, so spart man viele Resin-Wechsel. **Tankabdeckungen** sorgen dafür, dass weder Verschmutzungen noch UV-Licht das Harz unbrauchbar machen können. Für die meisten Drucker findet man solche Abdeckungen online als Modell. Auf den gängigen 3D-Daten-Plattformen findet man zusätzlich noch viele weitere praktische 3D-druckbare Helferlein. Die Links zu einigen Modellen finden Sie am Ende des Artikels.





Isopropanol reinigen

Mit der Zeit wird das verwendete Isopropanol immer mehr mit gelöstem Harz verunreinigt. Benutzt man dieses noch weiter zum Spülen von Drucken, so können sich auf der Oberfläche beim UV-Härten unschöne Flecken bilden. Es hilft, zuerst grob in schon älterem Isopropanol zu spülen und dann in sauberem Isopropanol. Aber das Isopropanol muss man irgendwann dann doch entsorgen.

Um dem zu entgehen, kann man das Isopropanol reinigen: Es wird unter ständigem Rühren mit UV-Licht bestrahlt ⁸, dabei fällt gelöstes Harz aus und kann später ausgefiltert werden. Dieses Belichten kann entweder in UV-durchlässigen Behältern in der Curing-Station erfolgen oder in Metallbehältern, wenn man eine UV-Lampe hineinscheinen lässt. Auch ein Tag in der Sonne funktioniert mit UV-durchlässigen Behältern. Hierbei sollte man aber darauf achten, dass sich in den Flaschen ein hoher Druck aufbauen kann und für einen Druckabbau sorgen, z. B. durch einen nicht ganz geschlossenen Deckel.

Nach dem Belichten muss das behandelte Isopropanol mit einem dichten Papierfilter (Kaffeefilter in altem Filtereinsatz oder Laborfilter) filtriert werden, um die ausgefallenen Harz-Partikel herauszubekommen. Das kann einige Zeit dauern und wenn sich ein Filter zugesetzt hat, sollte man ihn besser gleich durch einen neuen ersetzen. War das Isopropanol sehr verunreinigt, so kann es auch praktisch sein, es einen Tag lang ruhen zu lassen, damit sich die Schwebstoffe absetzen und dann vorsichtig zu dekantieren und anschließend zu filtern.

Wie generell beim Umgang mit brennbaren Flüssigkeiten sollte man hierbei auf gute Ventilation achten und natürlich Zündquellen fernhalten. —caw

Harztemperatur

Die Druckbarkeit von Harzen steigt mit geringer werdender Viskosität (wenn sie flüssiger werden). Das erreicht man durch eine höhere Harztemperatur. Umgekehrt kann man den Effekt bei zu geringen Raumtemperaturen beobachten: In einer kalten Werkstatt oder im Keller steigt die Gefahr von Fehldrucken. Hochleistungsharze für den Profimarkt und belastbarere mechanische Bauteile sind nochmals zähflüssiger, was das Problem verstärkt.

Die Lösung ist, die Harze zu erwärmen: Dies senkt die Viskosität deutlich, die Druckbarkeit steigt. Einige Belichtungs- und Waschstationen wie z. B. die CW1 von Prusa bieten daher an, Harzflaschen zu temperieren ⁷. Im Druckraum ist die Temperatur meist ausreichend hoch durch die von der UV-LED abgestrahlte Wärme.

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 ...).

Alles zum Artikel im Web unter make-magazin.de/xxb1



Wenn wir Ihren Tipp veröffentlichen, bekommen Sie das bei Make übliche Autorenhonorar. Schreiben Sie uns dazu einen Text, der ungefähr eine Heftseite füllt und legen Sie selbst angefertigte Bilder bei. Senden Sie Ihren Tipp mit der Betreffzeile *Lesertipp* an:

► mail@make-magazin.de



Herbstcampus

Die Nürnberger Konferenz mit der Lernatmosphäre

6. bis 8. September 2022
TH Nürnberg

Bis 3. August
Frühbucher-
Rabatt
Jetzt buchen!

Der Herbstcampus ist der **Geheimtipp** schlechthin für Softwareentwickler, Softwarearchitekten und IT-Projektleiter im Enterprise-Umfeld. Die Nürnberger Fachkonferenz bietet IT-Profis aus ganz Deutschland ein **professionelles und lockeres Umfeld**, um aktuelle Trends kennenzulernen, aber auch um ganz bewusst Themen **abseits des Mainstreams** zu entdecken.

Highlights des Herbstcampus

- ✓ Lernen Sie aktuelle Trends der Softwareentwicklung kennen.
- ✓ Wählen Sie aus rund 40 Vorträgen und 4 ganztägigen Workshops Ihr Programm aus.
- ✓ Besuchen Sie die Konferenzausstellung mit wichtigen Unternehmen der Region und der Branche.
- ✓ Profitieren Sie von der anregenden Lern-Atmosphäre und hohem Wohlfühlfaktor.
- ✓ Genießen Sie die die Rund-um-die-Verpflegung über zwei Tage - inkl. Get-together am ersten Konferenzabend.

www.herbstcampus.de

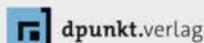
Veranstalter



Gold-Sponsor



Silber-Sponsoren

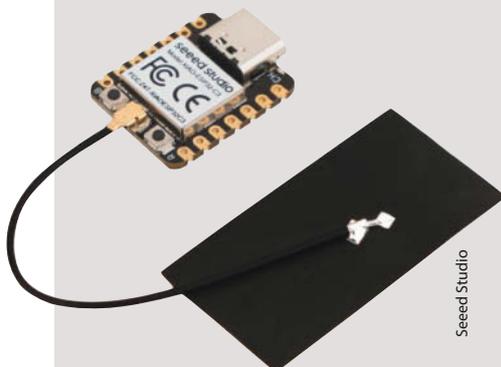


© Copyright by Maker Media GmbH.



XIAO ESP32C3

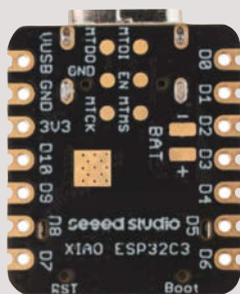
Funk-Mikrocontroller-Modul



Seeed Studio

Nur daumennagelgroß (21mm × 17,5mm) ist das neue Funk-Mikrocontroller-Modul XIAO ESP32C3 von Seeed Studio mit dem aktuellen 32-Bit-Prozessorkern in RISC-V-Architektur von Espressif, der dank vierstufiger Pipeline mit bis zu 160MHz getaktet werden kann. Das Modul bietet an Schnittstellen 11 GPIOs (teilweise mit PWM), die auch als UART, I²C, I²S, SPI oder ADC (4 Kanäle) konfigurierbar sind. Das HF-Frontend unterstützt WiFi/Bluetooth-Verbindungen über 100m, aber auch Bluetooth 5 und Bluetooth Mesh im Low-Power-Modus. Für eigene Programme stehen 400KByte SRAM und 4MByte Flash-Speicher zur Verfügung.

Der Hersteller hebt den geringen Stand-By-Stromverbrauch von nur 44µA und die eingebaute Ladeelektronik für Lilon-Zellen hervor. Die Platine enthält praktischerweise Taster für Reset und Programmierung per Bootloader sowie eine Buchse für die externe Antenne. Das Modul kostet bei Vorbestellung 5 US-\$ und soll ab Mitte August am Lager sein. —cm



Hersteller	Seeed Studio
URL	make-magazin.de/xpxw
Preis	5 US-\$

Warp2 Charger

Open-Source-Wallbox mit ESP32

Die Wallbox Warp2 Charger ist ein Open-Source-Projekt von Tinkerforge, basiert auf einem ESP32 und ist in drei Versionen erhältlich. Firmware, Dokumentation und weitere Informationen der Made-in-Germany-Box sind auf der Github-Projektseite erhältlich (siehe Link).

Die Basic-Ausgabe beschränkt sich auf die reinen Ladefunktionen. Es gibt sie mit 11 und 22kW Ladeleistung sowie 5 oder 7,5m langem Ladekabel (Ladestecker Typ 2) für jeweils 50 Euro Aufpreis. Die Smart-Version enthält zusätzlich eine Web-Oberfläche, die unter anderem den Ladeverlauf des angeschlossenen Fahrzeugs anzeigt. Außerdem ermöglicht ein eingebauter RFID-Leser eine Benutzerverwaltung. In Verbindung mit einer Photovoltaik-Anlage ist auch ein Überschussladen möglich. Die Pro-Variante wird mit einem geeichten Stromzähler sowie einer Logbuch-Funktion zur Abrechnung der Ladekosten geliefert.

Allen Versionen gemein ist das Gehäuse (280mm × 215 mm × 95 mm) mit einer Front aus Edelstahl und der wahlweise Anschluss an 230V-Einphasen-Wechselstrom oder ans dreiphasige Drehstromnetz. Die Box ist gemäß IP54 spritzwassergeschützt und damit auch zur Montage



Tinkerforge

im Außenbereich geeignet. Integriert ist eine 6mA-Fehlerstromerkennung im Gleichstromteil.

Zusätzlich gibt es eine aus rostfreiem Edelstahl gefertigte Säule, an die zwei Wallboxen montiert werden können. Die Boxen werden dabei von innen mit Schrauben befestigt, der Zugang zum Säuleninneren kann mit einem Schloss gesichert werden. Das soll vor Diebstahl schützen. —hgb

► make-magazin.de/xpxw

Hersteller	Tinkerforge
URL	tinkerforge.com/de/shop/warp/wallbox.html
Preis	ab 700 €

Positron V3

Transportabler Open-Source-3D-Drucker druckt über Kopf

Dieser 3D-Drucker passt mit seinem Packmaß von 200mm × 200mm × 80mm genau in einen handelsüblichen Umkarton für eine Filamentrolle und lässt sich bequem im Rucksack transportieren. Vor Ort dauert der Aufbau dann nur eine Minute. Danach kalibriert sich das Gerät mithilfe eines Infrarotsensors selbst und druckt mit einem Bauraum von 180mm × 185mm × 180mm und bis zu 250mm/s schnell.

Wenn der 3D-Drucker aufgeklappt ist, fällt sofort der umgedrehte Druckkopf ins Auge. Mit der gläsernen Druckplattform über dem Hotend wirkt der Positron fast wie ein Harzdrucker, er befördert jedoch PLA- oder PETG-Filament – nach oben. Das funktioniert sehr gut, wie im Video zu sehen (siehe Link). Selbst Überhänge druckt der Positron mühelos.

Dass er den Druckkopf nach unten versetzt hat, begründet sein Entwickler damit, dass dies Platz spart und den Drucker gleichzeitig stabilisiert. Je tiefer sich der Druckkopf befindet, desto weniger Vibrationen sei der 3D-Drucker ausgesetzt. Um die größtmögliche Druckfläche auf kleinem Raum zu nutzen, sind die X- und Y-Achsen diagonal zur Grundfläche angeordnet. Da sein Entwickler selbst keine



KRALYN/Github.com

Produktion des Positron plant, stellt er allen Interessierten sämtliche Baupläne des Open-Source-Projekt auf GitHub kostenfrei zur Verfügung. —akf

Einen etwas ausführlicheren Online-Artikel inklusive Video zum Drucker finden Sie online (siehe Link).

► make-magazin.de/xpxw

Entwickler	KRALYN Design
URL	github.com/KRALYN/PositronV3
Lizenz	Open Source

BeagleBone AI-64

Hochleistungs-Kleinstcomputer

Der Open-Hardware-Spezialist *BeagleBoard* hat den *BeagleBone AI-64* auf den Markt gebracht, mit dem man auch für aufwändige AI- und Machine-Learning-Anwendungen gewappnet sein soll. Das Board will mit seinem 2GHz schnellen *Jacinto TDA4M* von *Texas Instruments*, einem *Cortex-M72*-Zweikerner mit Vision-Engine und AI-Beschleuniger die Lücke zwischen kleinen SBCs wie dem *Raspberry Pi* und Desktop-PCs schließen.

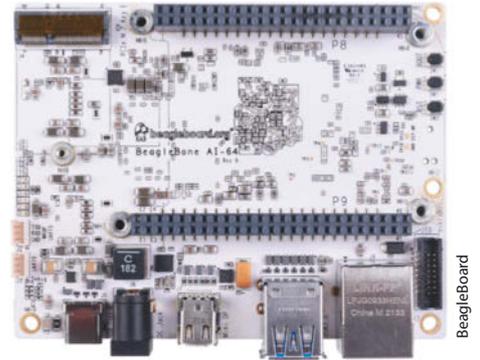
Die Eignung im Bereich künstlicher Intelligenz sollen ein *C7x*-Vektor-Signalprozessor als *Deep Learning Matrix Multiply Accelerator (MMA)* mit 8 TOPs und zwei Fließkomma-DSPs mit zusammen 40 GFLOPs unterstreichen,

ebenso *Depth and Motion Processing Accelerators (DMPACs)* und drei *Arm-Cortex-R5*-Zweikerner als Koprozessoren. Die Grafikausgabe erledigt eine 3D-fähige *PowerVR-Rogue-GPU 8XE GE8430*, die mit 750MHz läuft. Als Hauptspeicher stehen 4GByte LPDDR4-RAM zur Verfügung.

Wie üblich sind Hardware-Design und Software als Open Source offengelegt und stehen in einem Git-Repository zur Verfügung (siehe Link); das Debian-Linux-Betriebssystem ist bereits lauffähig vorinstalliert.

—cm

► make-magazin.de/xpxw



BeagleBoard

Hersteller	BeagleBoard
Vertrieb	z.B. Farnell, RS Components
Preis	195 €

Programmieren lernen mit Scratch

Für Kinder und Erwachsene – ab 10 Jahre

Das Buch führt zuerst in die grafische, blockorientierte Sprache *Scratch* ein, um dann anhand dieser das Programmieren allgemein zu lehren. Die Wahl von *Scratch* als Einstieg ist nicht ungeschickt, bringt dessen Entwicklungsoberfläche doch viele Werkzeuge wie Bild- und Soundeditoren mit, die man sonst extra lernen oder installieren müsste.

Die ersten drei Kapitel folgen dem Prinzip vieler *Scratch*-Bücher und -Videos: Man erstellt im Grunde eine – vielleicht interaktive – Geschichte oder Animation, wobei mehr Erzähltalent denn Programmierung nötig ist. Aber man lernt dabei recht detailliert die Werkzeuge von *Scratch* kennen, vor allem, wie Block-Editor und Zeichenfläche funktionieren. Wenn man nur nach dem Inhaltsverzeichnis geht, kommen einem die fast 80

Seiten für diesen Teil viel vor, allerdings werden immer wieder geschickt Vorbereitungen für die folgende *eigentliche* Programmierung getroffen: So findet der Leser zwischendurch Tipps zu gutem Programmierstil mit Kommentaren und sinnvoll benannten Objekten.

In Kapitel 4 geht es dann los mit Simulationen und Spielen wie dem Klassiker *Pong* oder einem Rennautospiel. Zwischendurch geben Kästen weitere Information und kleine *Challenges*, die etwa kleine Codebrocken zeigen, aus denen man dann den funktionierenden aussuchen muss.

In Kapitel 5 geht es um Geschichten und Abenteuer. Man wähnt sich wieder im Kapitel 1, aber weit gefehlt: Im Unterbau geht es um Listen, Elemente, Indices und Ablaufsteuerung, verpackt in interaktive Geschichten und ein *Escape-Room*-Spiel.

In Kapitel 6 kommen dann die Wissenschaftler unter den



Kids auf ihre Kosten: Biologie des Tarnens, reale Physik von Mondfähren und Planeten, Lautstärkemessung und Diagramme in Echtzeit, Gassimulation. Stoff, mit dem man sich in der Schule beliebt machen kann.

Das Buch verwendet durchgehend Illustrationen, die wenigen Pixelbilder tauchen nur dort auf, wo es auch um Pixelgrafiken geht. Manche Grafiken wirken leicht naiv, aber zusammen mit der Farbgebung ergibt sich am Ende doch ein ruhiges, aber nicht langweiliges Layout. Das quadratische Format von 22cm bringt viel Buch vor oder hinter die Tastatur.

—caw

Autor	Michael Weigend
Verlag	mitp Verlag
Umfang	224
ISBN	9783747504406
Preis	19,99 €



Sno-M2

FPGA-Modul von Sparkfun



Der in der Maker-Szene bekannte US-Hersteller *Sparkfun* hat in Zusammenarbeit mit dem FPGA-Spezialisten *Alorium* ein neues Prozessormodul im *MicroMod*-Formfaktor vorgestellt, das statt eines Mikrocontrollers einen MAX10M16-Gatterbaustein der Intel-Tochter *Altera* enthält.

Die *MicroMods* verwenden einen M.2-Anschluss im *E-Key*-Format, auf das auch viele WLAN-Module für PCs setzen. Die Module lassen sich in eigenen Entwicklungen ebenso einsetzen wie auf den *Sparkfun*-Entwicklungskits. *Alorium* steuert einen schnellen AVR-Softcore bei, sodass sich das Modul über die *Arduino*-IDE-Software programmieren lässt und sich zunächst wie ein normaler (wenn auch nun bis zu 32MHz schneller) *ATmega*-Mikrocontroller verhält.

Besonderer Clou dabei ist aber, dass man beliebige Peripherie gleich mit in das FPGA integrieren kann; die 16.000 Logikzellen, 45 Multiplizierer und insgesamt 549Kbit frei konfigurierbares Block-RAM bieten (FPGA-Erfahrung vorausgesetzt) genügend Platz für eigene Ideen.

Wer sich nicht mit der (anfängs recht gewöhnungsbedürftigen) *FPGA*-Programmierung herumschlagen will, kann auf vorgefertigte *IP-Module* (sogenannte *Xcelerator Blocks* oder *XBs*) von *Alorium* zurückgreifen, etwa zur direkten *Neo-Pixel*-Ansteuerung, für schnelle Fließkomma-Arithmetik oder zum Aufbau von *Servo*-Regelkreisen. —*cm*

Hersteller	Sparkfun
URL	sparkfun.com/products/18030
Preis	50 US-\$

AVR-IoT Cellular Mini

Entwicklungsboard mit 5G-Verbindung

Für die drahtlose Schmalband-Vernetzung von Sensoren und Aktoren stellt *Microchip* nun ein Entwicklungsboard auf Basis des *AVR128DB48*-Mikrocontrollers her. Das Board arbeitet im 5G-Netz und soll damit Lücken in *LoRa*-Netzwerken schließen.

Das 8Bit-Board bietet unter anderem sechs analoge Eingänge, sieben digitale IO-Pins sowie eine I²C- und zwei serielle Schnittstellen. Per I²C auslesbare Temperatur- und Farbsensoren sitzen bereits auf der Platine. Die Stromversorgung erfolgt über eine USB-C-Buchse oder einen Akku (3,7V Lithium). Die Ladeelektronik ist auf der Platine im *Adafruit-Feather*-Format vorhanden. Mitgeliefert wird eine externe Funkantenne.

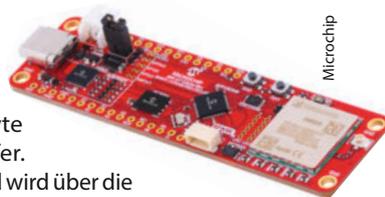
Der Datenaustausch erfolgt über die Cloud des Herstellers, eine entsprechende *SIM*-Karte für den Funknetz-Zugang gehört zum Board. Sie ist 90 Tage lang gebührenfrei

und erlaubt in dieser Zeit bis zu 150 Megabyte Datentransfer.

Das Board wird über die *Arduino*-IDE programmiert. Die erforderlichen Bibliotheken und Beispielprogramme stehen über ein *GitHub*-Repository bereit. Damit ist beispielsweise der Datenaustausch über *HTTPS* und *MQTT* an gängige *SmartHome*-Systeme möglich. Eine ausführliche Dokumentation der Platine gibt es online (siehe Link). —*hgb*

► make-magazin.de/xpxw

Hersteller	Microchip
URL	make-magazin.de/xpxw
Preis	57 €



Modulare Synthesizer mit VCV-Rack 2 entdecken

Das *VCV-Rack-System* ist eine in der Grundversion kostenlose *Open-Source*-Software, die ein komplettes *Eurorack*-Synthesizer-System auf einem Computer simuliert. In wenigen Minuten installiert, kann man dann direkt losstöpseln, wobei ein kleines Tutorial hilft. Ohne Vorwissen steckt man dann aber fest und muss weitere Hilfe suchen.

Hier springt das von *Erik Bartmann* geschriebene Buch ein, das auf 320 farbigen *Din-A4*-Seiten so ziemlich alles zu *VCV-Rack2* behandelt. Auch musikalischer und mathematischer Hintergrund der Klangerzeugung kommen nicht zu kurz.

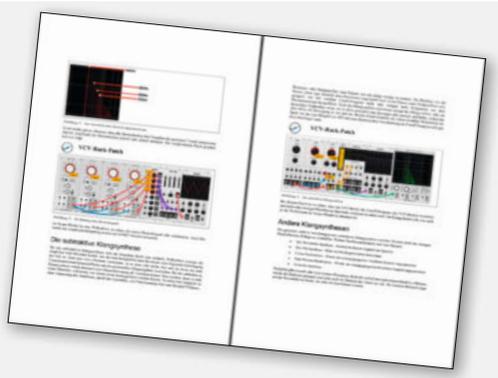
Die Themen werden aufeinander aufbauend behandelt, zwischen den Abschnitten befinden sich immer wieder Kästen, die neue

Begriffe und Konzepte einführen, Tipps oder Links geben. Der Leser kann diese Kästen überspringen und so stören sie den Lesefluss nicht.

Nach den Grundlagen von *VCV-Rack* geht es in den weiteren Kapiteln detaillierter um Themen wie *Klangsynthese*, *LFOs*, *Filter*, *VCA* und alles, was ein modulares System ausmacht. Dabei werden natürlich für jedes Thema *Patches* zum Nachstecken präsentiert, an denen man das Gelernte ausprobieren kann.

Das Buch ist klar aufgebaut, man sollte aber kein liebevoll designtes Layout

erwarten. Das ist aber auch nicht nötig, die Abbildungen sind von guter Qualität, der Text ist gut geschrieben und verständlich. Noch zwei Warnungen: Es ist durchaus möglich, nach *Synthesizern* süchtig zu werden. Vielleicht noch wichtiger: Auch wenn man alle Module versteht, wird man nicht zwingend den nächsten großen Hit produzieren. —*caw*



Autor	Erik Bartmann
Verlag	Bombini Verlags GmbH
Umfang	320 Seiten
ISBN	9783946496311
Preis	34,95 €

CM4-POE-UPS-BASE

Baseboard mit USV und PoE für Raspberry Pi Compute Module 4

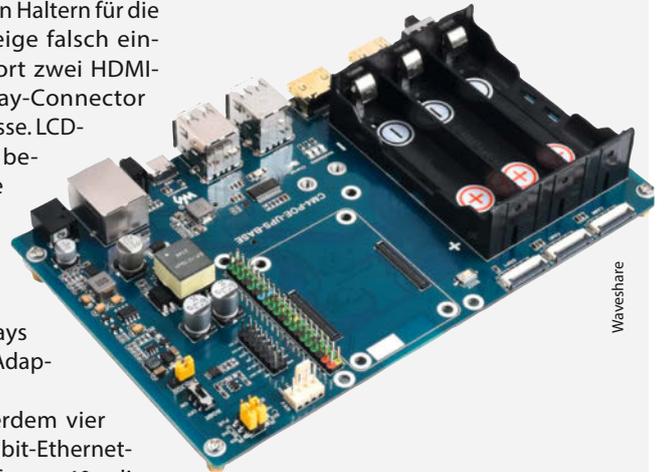
Das neue Baseboard des Herstellers *Wave-share* für das *Raspberry Pi Compute Module 4* bietet dank eines Halters für 3 Lithium-Akkus vom Typ 18650 unterbrechungsfreie Stromversorgung. Zur normalen Energieversorgung und zum Laden stehen USB und *Power over Ethernet* (PPoE) zur Wahl – oder man nutzt eine Stiftbuchse mit einem zulässigen Eingangsspannungsbereich von 7 bis 36V.

Per Linux-Software (per Download beim Hersteller erhältlich) kann ein Ausfall der Stromversorgung erkannt und bei Unterschreiten einer Mindestladung der Akkus der Raspi automatisch heruntergefahren und ausgeschaltet werden.

Auf der Unterseite der 175mm × 108mm großen Platine sitzt ein Slot für eine M.2-SSD inklusive Schraubbefestigungen für Module der Größe 2230, 2242, 2260 und 2280. Auch die Batterie der RTC, der SD-Kartenslot sowie die selbstheilenden Sicherungen für die Akkuzellen finden sich hier.

Auch die Oberseite des Baseboards hat einiges zu bieten: Neben Haltern für die Akkuzellen mit LED-Anzeige falsch eingesetzter Akkus sitzen dort zwei HDMI-Anschlüsse, ein LC-Display-Connector sowie zwei Kameraanschlüsse. LCD- sowie Kameraanschlüsse besitzen jeweils 22 Kontakte mit 0,5mm Abstand wie der Kameraanschluss des Raspberry Pi Zero. Zur Verbindung mit herkömmlichen Raspi-Displays und Kameramodulen sind Adapter notwendig.

Das Board bietet außerdem vier USB-2-Buchsen, einen Gigabit-Ethernet-Anschluss, die Raspi-konforme 40polige GPIO-Stiftleiste sowie einen 4poligen Lüfter-Stecker. Die ausführliche Dokumentation für das Board ist online erhältlich (siehe Link).
—hgb



WaveShare

Hersteller	WaveShare
URL	make-magazin.de/xpxw
Preis	72 US-\$ ohne Versand bis 98 € mit Versand

DIE WEBINAR-SERIE VON HEISE

Product Owner im Unternehmen

Lernen Sie, wie Sie als **Product Owner effektiv arbeiten** können, **Ihre Rolle im Unternehmen** verstehen, und wie Sie die wichtigsten **Konzepte und Techniken meistern**.

DIE TERMINE:

07. September 2022

Die Rolle des Product Owners verstehen

14. September 2022

Als Product Owner in den Scrum-Events

21. September 2022

User Stories, Akzeptanzkriterien und Schneiden von Anforderungen

28. September 2022

Erfolgreiche Zusammenarbeit mit Stakeholdern

05. Oktober 2022

Die Bedürfnisse der Nutzer verstehen und berücksichtigen

Exklusiver Kombi-Preis: 595,-

Einzelpreis pro Webinar: 169,-

 heise Academy



Jetzt Kombi-Rabatt sichern und 250,- sparen:
webinare.heise.de/product-owner/

IMPRESSUM

Make: Nächste Ausgabe erscheint am 22. September

Redaktion

Make: Magazin
Postfach 61 04 07, 30604 Hannover
Karl-Wiechert-Allee 10, 30625 Hannover
Telefon: 05 11/53 52-300
Telefax: 05 11/53 52-417
Internet: www.make-magazin.de

Leserbriefe und Fragen zum Heft: info@make-magazin.de

Die E-Mail-Adressen der Redakteure haben die Form xx@make-magazin.de oder xxx@make-magazin.de. Setzen Sie statt „xx“ oder „xxx“ bitte das Redakteurs-Kürzel ein. Die Kürzel finden Sie am Ende der Artikel und hier im Impressum.

Chefredakteur: Daniel Bachfeld (dab)
(verantwortlich für den Textteil)

Stellv. Chefredakteur: Peter König (pek)

Redaktion: Heinz Behling (hgb), Ákos Fodor (akf), Helga Hansen (hch), Carsten Meyer (cm), Carsten Wartmann (caw)

Mitarbeiter dieser Ausgabe: Beetlebum (Comic), Johannes Börsnes, Guido Burger, André Burkhardtmaier, Kristina Fischer, Klaus-Uwe Gollmer, Hans-M. Hilbig, Matthias Martin, Matthias Mett, Alexander Moser, Uwe Rohne, Daniel Springwald, Peer Stelldinger, Ralf Stoffels, Florian Stolz, Dr. Armin Zink

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 Titellbild: Andreas Wodrich

Digitale Produktion: Kevin Harte, Thomas Kaltschmidt, Pascal Wissner

Hergestellt und produziert mit Xpublisher:
www.xpublisher.com

Verlag

Maker Media GmbH
Postfach 61 04 07, 30604 Hannover
Karl-Wiechert-Allee 10, 30625 Hannover
Telefon: 05 11/53 52-0
Telefax: 05 11/53 52-129
Internet: www.make-magazin.de

Herausgeber: Christian Heise, Ansgar Heise

Geschäftsführung: Ansgar Heise, Beate Gerold

Anzeigenleitung: Michael Hanke (-167)
(verantwortlich für den Anzeigenteil),
mediadaten.heise.de/produkte/print/
das-magazin-fuer-innovation

Leiter Vertrieb und Marketing: André Lux (-299)

Service Sonderdrucke: Julia Conrades (-156)

Druck: Dierichs Druck + Media GmbH & Co.KG,
Frankfurter Str. 168, 34121 Kassel

Vertrieb Einzelverkauf:
DMV DER MEDIENVERTRIEB GmbH & Co. KG
Meßberg 1
20086 Hamburg
Telefon: +49 (0)40 3019 1800
Telefax: +49 (0)40 3019 1815
E-Mail: info@dermedienvertrieb.de
Internet: dermedienvertrieb.de

Einzelpreis: 12,90 €; Österreich 14,20 €; Schweiz 25,80 CHF;
Benelux 15,20 €

Abonnement-Preise: Das Jahresabo (7 Ausgaben) kostet
inkl. Versandkosten: Inland 77,00 €; Österreich 84,70 €;
Schweiz/Europa: 90,65 €; restl. Ausland 95,20 €

Das Make-Plus-Abonnement (inkl. Zugriff auf die App, Heise Magazine sowie das Make-Artikel-Archiv) kostet pro Jahr
6,30 € Aufpreis.

Abo-Service:

Bestellungen, Adressänderungen, Lieferprobleme usw.:

Maker Media GmbH
Leserservice
Postfach 24 69
49014 Osnabrück
E-Mail: leserservice@make-magazin.de
Telefon: 0541/80009-125
Telefax: 0541/80009-122

Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden. Kein Teil dieser Publikation darf ohne ausdrückliche schriftliche Genehmigung des Verlags in irgendeiner Form reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden. Alle beschriebenen Projekte sind ausschließlich für den privaten, nicht kommerziellen Gebrauch. Maker Media GmbH behält sich alle Nutzungsrechte vor, sofern keine andere Lizenz für Software und Hardware explizit genannt ist.

Für unverlangt eingesandte Manuskripte kann keine Haftung übernommen werden. Mit Übergabe der Manuskripte und Bilder an die Redaktion erteilt der Verfasser dem Verlag das Exklusivrecht zur Veröffentlichung. Honorierte Arbeiten gehen in das Verfügungsrecht des Verlages über. Sämtliche Veröffentlichungen in Make erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes.

Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.

Published and distributed by Maker Media GmbH under license from Make Community LLC, United States of America. The 'Make:' trademark is owned by Make Community LLC. Content originally partly published in Make: Magazine and/or on www.makezine.com, ©Make Community LLC 2020 and published under license from Make Community LLC. All rights reserved.

Printed in Germany. Alle Rechte vorbehalten.
Gedruckt auf Recyclingpapier.

© Copyright 2022 by Maker Media GmbH

ISSN 2364-2548

Nachgefragt

Energieversorgung ist gerade ein brisantes Thema in Deutschland. Auf welches Gerät würdest Du als Erstes bei Stromknappheit verzichten?



Guido Burger
Villingen-Schwenningen, sammelt auf Seite 10 Energie mit einem Solartisch
Auf keines – Wer sich seinen Stromverbrauch genau anschaut, kann selbst mit wenig Sonnenenergie viel Unabhängigkeit erreichen: Kühlen wir mit Sonnenenergie! Aber muss denn jedes Gerät heute einen Akku haben?



Hans-Martin Hilbig
Tiefenbach, kommt mit dem Pi Pico auf Seite 62 Stromausfällen auf die Spur
Der zweite Kühlschrank im Keller! Dank des Strommonitors werfe ich in Zukunft keine Tiefkühl-Pizzen mehr weg. Dann reicht auch ein Kühlschrank.



Alexander Moser
Hannover, steht dank Flutlicht (Seite 58) beim Dart nicht im Dunkeln
Am ehesten könnte ich auf meinen Kühlschrank verzichten. Im Winter kann ich ja vieles auf dem Balkon lagern. Für Aufgaben wie Surfen, E-Mail und FTP reicht ein Raspi bei 4 Watt anstatt eines PC bei 80 Watt.

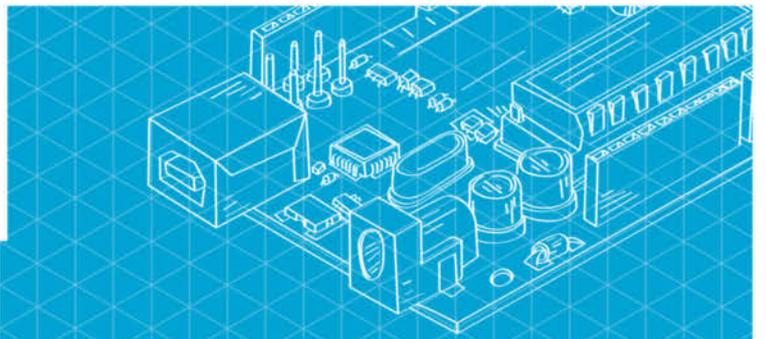


Martina Bruns
Garbsen, sorgt als Jr. Art Directorin der Make für eine grafische Linie
Der Stand-by-Betrieb wird schon, wo es geht, ausgeschaltet. Vor Kurzem habe ich die elektrische Brot-schneidemaschine in den Schrank verbannt und bin wieder auf Handbetrieb umgestiegen. Da war ich der Zeit wohl voraus.

Inserentenverzeichnis

ALLNET Computersysteme GmbH, Germering	25	nobufil GmbH, A-Krems an der Donau	79
AVM Computersysteme Vertriebs GmbH, Berlin	124	Rheinwerk Verlag GmbH, Bonn	13
Cameo Laser Franz Hagemann GmbH, Stuhr	7, 49	VSM Vereinigte Schmirgel- und Maschinen-Fabriken AG, Hannover	27
dpunkt.verlag GmbH, Heidelberg	81	ZUKUNFTINC. e.V., Hannover	19
Leipziger Messe GmbH, Leipzig	89	Make:markt	37

Make:



DAS KANNST DU AUCH!



GRATIS!



2x Make testen und über 9 € sparen!

Ihre Vorteile:

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

Für nur 16,10 € statt 25,80 €

* Für die Laufzeit des Angebotes.

Jetzt bestellen: make-magazin.de/miniabo



WLANSINN

Schneller. Weiter. Besser.



avm.de/WLANsinn