

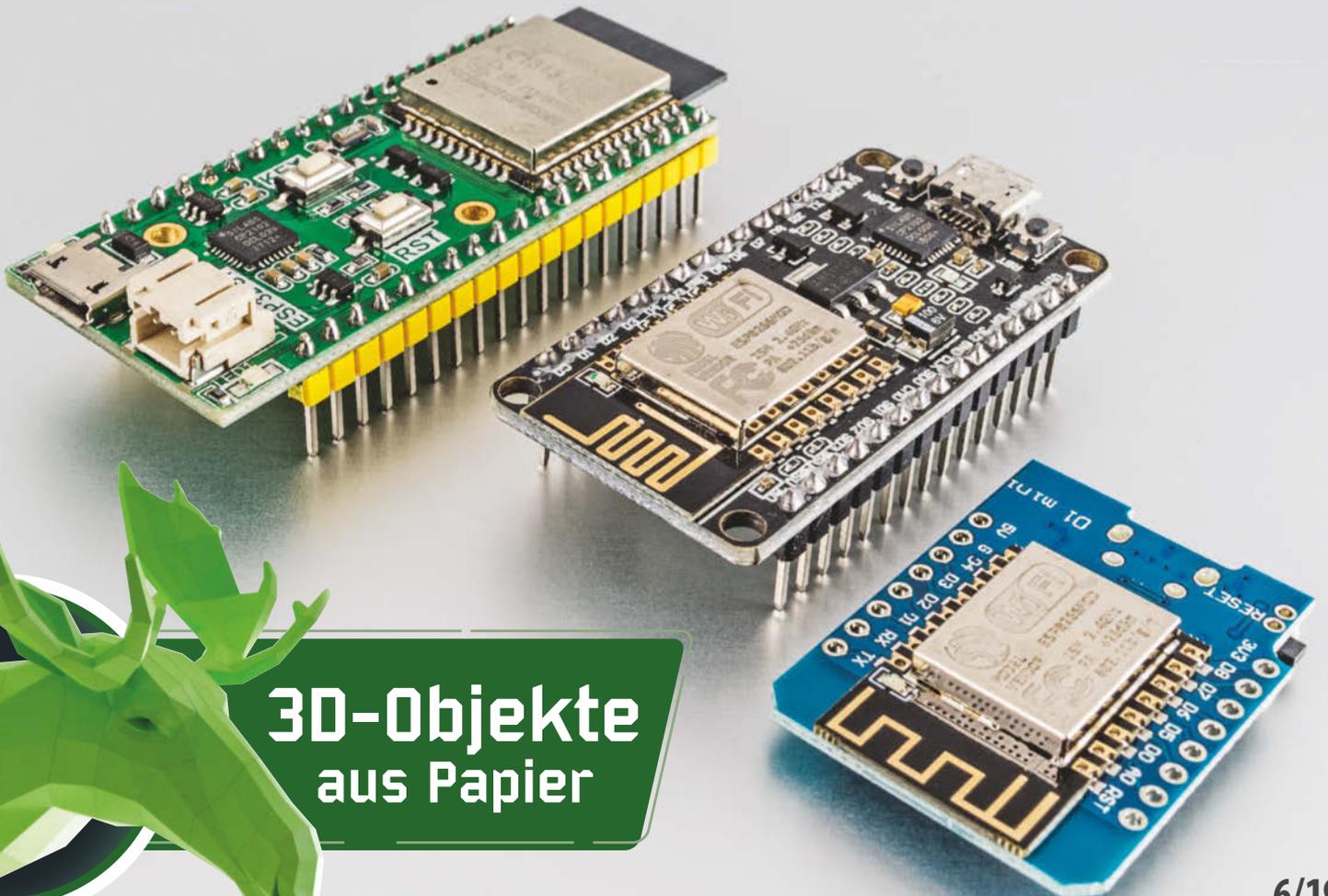
Open Window Alarm
Schützt Räume
vor Kälteeinbruch



Klein, drahtlos, stromsparend, nur 5 Euro

Die IoT-Allerwacker

Einsteigerprojekte mit ESP8266 und ESP32



3D-Objekte
aus Papier

- ▶ **Riesenspaß für viele Stunden**
Roboterbasteln für Kinder
- ▶ **Der Glasfasertrick**
Kleidung leuchten lassen
- ▶ **Touch-Steuerung programmieren**
Grafische Oberfläche fürs Raspi-Projekt
- ▶ **Feiern wie die NASA**
Startrampe fürs sichere Silvesterfeuerwerk

6/19

CH CHF 18,00
AT, Benelux,
IT, ES € 11,90

€ 10,90



Maker Faire®

TECHNOLOGIE. INNOVATION. COMMUNITY.

Where it's cool to be smart

Präsentieren Sie Ihr Unternehmen auf der Maker Faire, entdecken Sie neue Talente, Ideen und bereichern Sie Ihre Unternehmenskultur.

Die Maker Faires zelebrieren Technologie, Innovation und Community auf eine einmalige Art und Weise. Werden Sie Aussteller und atmen Sie die Unternehmenskultur von morgen!



Ihr Ansprechpartner

Jens Ahlers

Manager Sales & Strategy

Tel.: +49 511 5352-133

jeah@maker-media.de

Werden Sie Aussteller!

Lernen Sie alle Vorteile kennen:

www.maker-faire.de/Aussteller

© Copyright by Maker Media GmbH



Inzwischen sind wir drei: die Redakteurinnen der Make
Rebecca Husemann, Helga Hansen und Elke Schick (v.l.n.r)

Sichtbar sein

Zu den Aufgaben einer Zeitschriften-Redakteurin gehört es wenig überraschend, Artikel zu redigieren. Bei der Produktion dieses Heftes stutzte ich aber. Meine zwei Autorenmanuskripte waren tatsächlich Autorinnenmanuskripte – eine Premiere. Das zeigt sich, noch eine Premiere, auch in unserer Impressumsrubrik „Nachgefragt“. Dort finden Sie zum ersten Mal fünf Autorinnen. Gerne würde ich an dieser Stelle zur Tagesordnung übergehen. Vielleicht einen kurzen Tusch erklingen lassen und dann kommentarlos weiter schreiben. Oder irgendwas basteln, was mir mehr Spaß macht, als Gender Gaps zu erklären. Leider ist das aber immer noch ein Thema. Technik gilt in Deutschland als Männersache und unter Makern sieht es oft so ähnlich aus wie im Maschinenbau oder der Elektrotechnik: Zu Meetups erscheinen mehr Leute, die Sebastian heißen, als Frauen und ich wurde schon für die Sekretärin meiner Kollegen gehalten.

Dabei war und ist das Geschlechterbild nicht überall so. Bis in die 60er Jahre war Programmiererin in den USA ein Frauenberuf und auch heute gelten in Malaysia Computertechnik und Informatik als Frauensache. Dort bestimmt der Arbeitsort – drinnen oder draußen – das Geschlechterklischee. So, wie es gerade ist, muss es also nicht bleiben. Eine einfache Lösung, um die Klischees hierzulande

aufzulösen, gibt es allerdings nicht. Im Gegenteil: meist ist das viel Arbeit. Wenn es weniger Makerinnen gibt, bedeutet das für uns als Redaktion mehr Aufwand, sie zu finden. Oft sind ihre Zusagen zögerlicher. Einer der Gründe: Unter den vielen Autoren sticht man, beziehungsweise frau, als Autorin hervor und muss mit besonderer Aufmerksamkeit rechnen.

Also helfen Sie uns gern, Makerinnen sichtbar zu machen, ohne sie in ein unangenehmes Rampenlicht zu stellen. Bieten Sie Workshops für Mädchen in ihrem Fablab an, damit keines den Weg alleine gehen muss. Laden Sie Frauen-Meetups und Verbände ein, Ihren Makerspace kennenzulernen. Fragen Sie nach, warum die Frauen nicht wiederkommen würden, und suchen Sie nach Lösungen, von der Kinderbetreuung bis zur Bus-Anbindung. Und wenn Sie ein schönes Projekt sehen, geben Sie der Makerin unsere Mail-Adresse.

Helga Hansen

Helga Hansen

Sagen Sie uns Ihre Meinung!

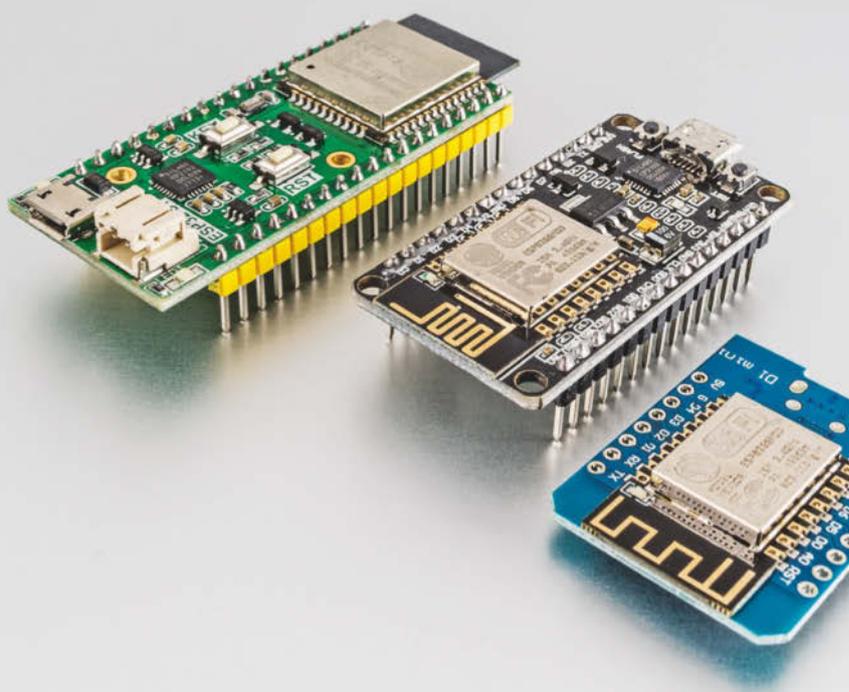
mail@make-magazin.de



3D-Objekte aus Papier

Seite 28

Bild: Paperwolf, Wolfram Kampffmeyer



Die IoT-Allerköner

Sie sind stromsparend, so klein, dass sie sich überall einbauen lassen, drahtlos per WLAN und Bluetooth angebunden und kosten gerade mal 5 Euro: Mikrocontroller-Boards mit ESP32 oder ESP8266 sind ideale Steuerplatinen für eigene IoT- und Smart-Home-Projekte. Wir zeigen den Einstieg und zwei Anwendungen.

- 8 ESP32 und ESP8266
- 12 Mobiler Entfernungsmesser mit WLAN
- 16 Lovebox, die moderne Liebesbotin

Inhalt

Hightech selber machen

Glasfasern als Lichtleiter in Stoffe einweben und elektrische Signale aus menschlichen Muskeln ableiten – geht sowas denn auch mit Maker-Mitteln? Klar! Unsere Projektberichte zeigen im Detail, wie es funktioniert und wie der Nachbau klappt.

- 50 Kleidung durch Glasfasern leuchten lassen
- 56 Biosignale aus Armmuskeln steuern Spiel



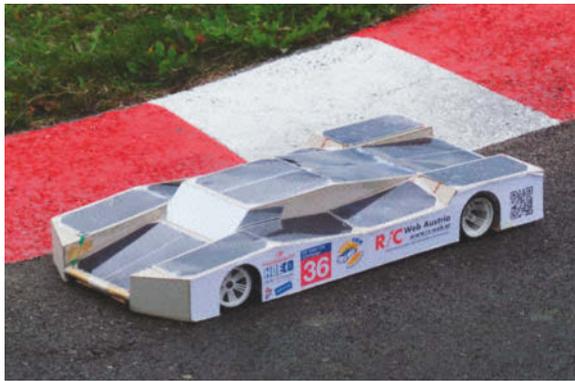
- 3 Editorial
- 6 Leserforum
- 8 **Die IoT-Allerköner: ESP32 und ESP8266**
- 12 Mit ESP32: Mobiler Entfernungsmesser mit WLAN
- 16 Mit ESP8266: Lovebox, die moderne Liebesbotin
- 24 Werkstattberichte: Neues aus der Szene, Comic
- 26 3D-Kurs für Maker: Konstruieren mit Blender 2.8
- 28 **Pepakura: 3D-Objekte aus Papier**
- 38 Was uns inspiriert: Wasserwelten, Pyro-Skateboard, Kunst mit Mustern
- 42 **Schützt Räume vor Kälteeinbruch: Open Window Alarm**
- 48 **Feiern wie die NASA: Startrampe fürs sichere Silvesterfeuerwerk**
- 50 **Der Glasfasertrick: Kleidung leuchten lassen**
- 56 Biosignale steuern Spiel: Muscle Laser Run

Elektrisch mobil

Rennen mit ferngesteuerten Autos machen Spaß und dank Solarzellen auf der Karosserie kann man (fast) klimaneutral rasen – für die Leistungsspitzen puffert ein Superkondensator die Energie. Der versorgt auch die Kindergeburtstag-tauglichen Bürstenroboter.

62 RC-Rennwagen mit Solarantrieb

78 Roboterbasteln für Kinder



Werkstatt

Unsere Anleitungen zeigen, wie der Raspberry Pi eine Touchscreen-GUI auf Maß und der Creality Ender 3 einen Druckbettsensor bekommt. Außerdem erklären wir, was Lithium-Ferrophosphat-Akkus können.

98 Pi mit Touchscreen steuern

110 Druckbettsensor für 3D-Drucker

116 Lithium-Ferrophosphat-Akkus



- 62 Ferngesteuerter Rennwagen mit Solarantrieb
- 70 Datenlogger für den Stratosphärenflug
- 78 **Make Family: Roboterbasteln für Kinder**
- 88 Community-Projekte: Riesen-Schachuhr, Alexa im Röhrenradio, LED-Uhr im Lego-Raumschiff
- 94 Tipps & Tricks
- 96 Reingeschaut: Sound-Glückwunschkarte
- 98 **Touch-Steuerung für Raspi selber programmieren**
- 110 3D-Druck: Druckbettsensor beim Ender 3 nachrüsten
- 116 Grundlagen: Lithium-Ferrophosphat-Akkus
- 121 Lieblingswerkzeug: Knete als Löthilfe
- 122 Kurzvorstellungen: Wärmebildkamera, I/O-Baustein, Mikrocontroller, Papercraft, Roboter, Retro-Joystick, 3D-Drucker
- 126 Bücher: KiCAD, Kunststoffe, Holz mit Dampf biegen, KI für Kinder, Akkutechnik
- 128 Impressum / Nachgefragt
- 130 Vorschau auf Make 1/20

Themen von der Titelseite sind rot gesetzt.

Schutz vor Kälteeinbruch

Im Hinblick auf Heizkosten und potenzielle Einbrecher sollte man in der kalten und dunklen Jahreszeit auf unnötig lang geöffnete Fenster achten – oder das diesem Sensor überlassen. Der ist schnell und billig nachgebaut und die Batterie hält einen Winter lang durch.

42 Open Window Alarm



Leserforum

Berichtigungen

Leider haben sich in der Make-Ausgabe 6/19 drei Druckfehler einschlichen: Zum einen ist in der 3D-Druckertabelle auf Seite 18 bei Creality Ender 3 die falsche Angabe hineingerutscht, das Gerät habe einen Direkt-Extruder – stattdessen hat der Ender einen Bowden-Extruder, wenn auch einen speziellen, bei dem der Vorschubmotor zusammen mit der Z-Achse nach oben und unten wandert. Zum anderen fehlen auf dem Verdrahtungsplan zur Followeranzeige auf Seite 62 die Verbindungen zwischen den oberen beiden 7-Segment-Anzeigen-Modulen und der Spannungsversorgung, die richtige Version gibt es online: heise.de/-4552830. Schließlich fehlt auf Seite 82 der Hinweis auf die Bildquelle: *Ivan Marjanovic/Shutterstock.com*. Wir möchten uns für diese Versehen entschuldigen.

Vertrauen in Technik

Sieben Highlights der Maker Faire Hannover, Make 5/19, S. 26

Der DeLorean war auch für mich als Besucher der Maker Faire Hannover am 17. 8. ein echter Hingucker. Allerdings war ich dann schon ziemlich verwundert, als der Besitzer, umringt von ein paar Dutzend Zuschauern, seinen DMC-12 mit einer Modellbaufernsteuerung über das Freigelände steuerte ¹. Wenn man Multicopter schwerer als 250g

schon als zu gefährlich zum Überfliegen von Menschenansammlungen einstuft, möchte ich mir nicht ausmalen, was die 1350kg und 97kW des Autos anrichten würden, wenn ein Übertragungsproblem der Funkfernsteuerung oder ein fehlerhaftes Servo den DeLorean mal richtig in Schwung gebracht hätte. Das Vertrauen der Veranstalter in die Technik scheint jedenfalls groß gewesen zu sein.

H. M. Hilbig

Es bestand tatsächlich in der Tat kein Grund zur Sorge, denn es handelt sich nur äußerlich um eine Modellbaufernsteuerung – im Inneren des Gehäuses stecken gleich drei Sender für drei unabhängige Bremssysteme mit jeweils eigenen Empfängern sowie einer Totmannschaltung. Versagt ein Empfänger, greift der zweite ein; versagt ein Servo, wird immer noch der zweite angesteuert. Das Failsafe-System ist bidirektional. Wenn eine der drei Bremsen gar kein oder kein plausibles Signal mehr bekommt, wird automatisch gestoppt und auch noch die Benzinzufuhr abgeschaltet. Die Geschwindigkeit des DeLorean ist auf 8km/h abgeriegelt, der Wagen wird bei Veranstaltungen aber ohnehin nur mit dem Standgas des Automatikgetriebes gefahren und stets auf Sicht sowie mit Helfer im Publikum. Es sind noch viele weitere Sicherheitssysteme an Bord und es kam in den vergangenen Jahren bei vielen Veranstaltungen noch nie zu einer kritischen Situation.



Selten Alltagsnutzen

Falsche Frage, Leserbrief von Frank Nerstheimer, Make 5/19, S. 6

Obwohl ich eigentlich schon mit neun Jahren, was nun auch schon wieder 34 Jahre her ist, zum „Maker“ wurde, lese ich die Make und ähnliche DIY-Zeitschriften nur sporadisch. Der Grund: Die Projekte mit echtem Alltagsnutzen sind sehr selten! Und oft empfinde ich sie als unvollständig. Möglicherweise liegt das daran, dass die Ausgangssituation nicht klar beschrieben ist und somit auch die finale gebrauchsfertige Lösung nicht definierbar ist oder zu sein scheint.

Für mich ist etwa Audio ein großes Thema. Will ich einen Vorverstärker, Kopfhörerverstärker oder eine kleine Aktivbox bauen, greife ich gezwungenermaßen auf Anleitungen aus den letzten Jahrzehnten zurück. Aber wie komme ich zu einer eigenen Audio-Wifi-Streaminglösung? Wie kann ich entsprechende fertige Systeme auf Seiten der Software und der Hardware modifizieren? Mit viel Interesse verfolge ich derzeit die Umprogrammierung billiger chinesischer Wifi-Dimmer und -Schalter, um diese vom Cloud-Zwang zu befreien. In den meisten Geräten findet sich ein Standard-Bauteil – der ESP8266/ESP32. Aber kann man das Vorgehen nicht auch auf Audio-Lösungen übertragen? Wo ist der Bauvorschlag für eine DIY-Steuerung mit einem ESP32 und 7-Zoll-Touchdisplay oder ein paar Tastern und LEDs, um „intelligente“ IoT-Geräte von der Stange von ihren speziellen Apps zu befreien, sodass man sie ohne Smartphone bedienen kann? Oft ist die verwendete Hardware identisch und ein kleines Touch-Bedienfeld vorhanden – hier wäre denkbar, dass es eine allgemeine Lösung geben könnte.

Mark Haacke

Kontakt zur Redaktion

Leserbriefe bitte an:

mail@make-magazin.de

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



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



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

Korrekturen

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

DAS EXKLUSIVE REICHELTT RASPBERRY PI 4 B ALL-IN-BUNDLE



THE COMPATIBLE PART OF YOUR PROJECT

3 Modelle. #1 Am Markt. Version 4

- mit Raspberry Pi 4 B mit 1/2/4 GB Arbeitsspeicher
- einem hochwertigen und trendigen Gehäuse
- einem leistungsstarken USB-C-Netzteil
- einer 16 GB microSD-Karte (Class 10) mit vorinstalliertem NOOBS
- einem 4-teiligen Raspberry Pi Kühlsatz
- einem microHDMI auf HDMI Kabel, 1 m



NEU

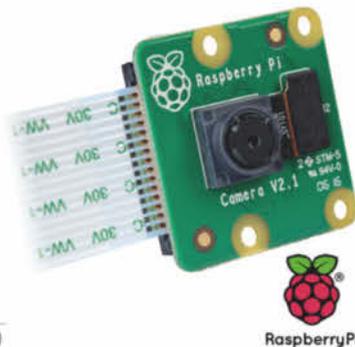
ab **64,⁵⁰**

Bestell-Nr.:		Speicher
RPI 4B 1GB ALLIN	64,50	1 GB
RPI 4B 2GB ALLIN	74,50	2 GB
RPI 4B 4GB ALLIN	84,50	4 GB

Raspberry Pi – 8 MP-Kamera

Hochauflösende Videokamera für den Raspberry Pi. Der Anschluss der Kamera erfolgt über die 15-polige serielle MIPI-Schnittstelle (CSI-2) auf Ihrem Raspberry Pi.

- Foto: 3280 x 2464 Pixel
- Video: 1080p @ 30 fps
- Bildsensor: 1/4"
- Kabellänge: ca. 150 mm
- Maße: 20 x 25 x 10 mm



PREIS-TIPP

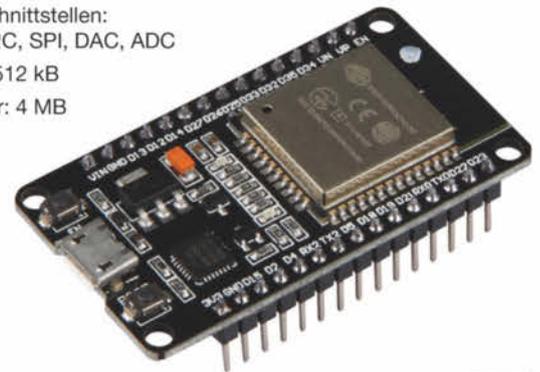
Bestell-Nr.: **RASP CAM 2** **24,50** (~~28,50~~)



ESP32 WiFi- und Bluetooth-Modul

Der NodeMCU ESP32 bietet neben WLAN auch noch eine Bluetooth Schnittstelle. Dank des integrierten CP2102 können Sie das Board einfach per microUSB programmieren.

- Prozessor: Tensilica LX6 Dual-Core
- Taktfrequenz: 240 MHz
- Datenschnittstellen: UART, I2C, SPI, DAC, ADC
- SRAM: 512 kB
- Speicher: 4 MB



Bestell-Nr.: **DEBO JT ESP32** **9,99**



Temperatur- & Feuchtigkeitssensor

besonders geeignet für Raspberry und Arduino

- Luftfeuchtigkeitsmessungen bis zu 2% Genauigkeit
- Temperaturmessungen von -40 ... 80°C bis zu 0,5°C Genauigkeit
- 0,5 Hz Abtastrate (einmal alle 2 Sekunden)



Bestell-Nr.: **DEBO DHT 22** **4,35**

ÜBER 110.000 ARTIKEL
ÜBER 10.000 NEUHEITEN

ENTDECKEN SIE UNSEREN AKTUELLEN KATALOG!

Jetzt kostenlos anfordern ►
www.reichelt.de/katalog



- Top Preis-Leistungs-Verhältnis
- über 110.000 ausgesuchte Produkte

- Zuverlässige Lieferung – aus Deutschland in alle Welt

Bestellservice: +49 (0)4422 955-333

www.reichelt.de

reichelt
elektronik – The best part of your project

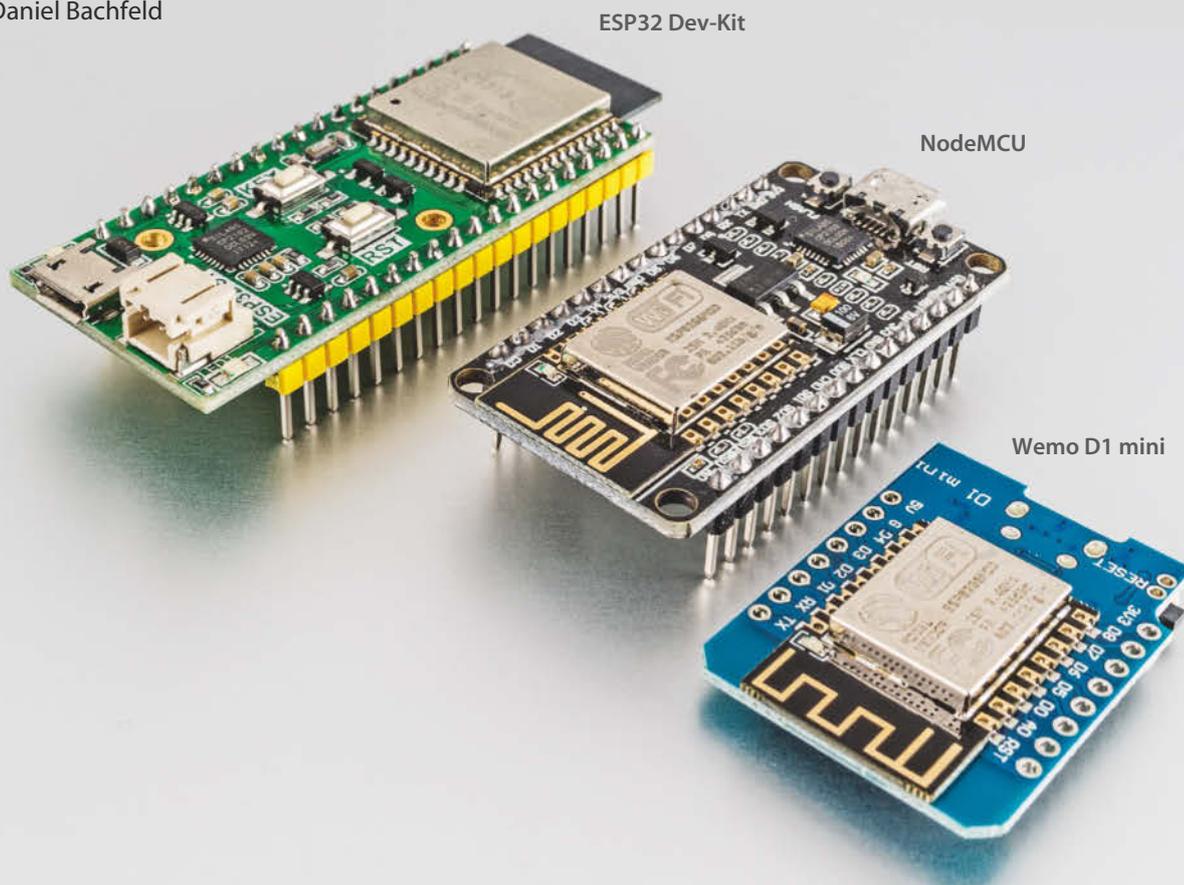
Es gelten die gesetzlichen Widerrufsregelungen. Alle angegebenen Preise in € inkl. MwSt. zzgl. Versandkosten für den gesamten Warenkorb. Es gelten ausschließlich unsere AGB (unter www.reichelt.de/agb, im Katalog oder auf Anforderung). Abbildungen ähnlich. Druckfehler, Irrtümer und Preisänderungen vorbehalten.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

TAGESPREISE! Preisstand: 18. 11. 2019

Die IoT-Allerwähltesten: ESP32 und ESP8266

WLAN, Bluetooth, A/D-Wandler, PWM, digitale Ein- und Ausgänge:
Der ESP8266 und sein großer Bruder ESP32 haben viel zu bieten.
Wir zeigen, was!

von Daniel Bachfeld



ESP32 Dev-Kit

NodeMCU

Wemo D1 mini

Make ESP32 Special

Das erste Make ESP32 Special erklärt die Hardware-Grundlagen des populären Wifi-Mikrocontrollers, hilft beim Programmierereinstieg und bringt IoT-Projektideen mit. Von der Einrichtung in der Arduino-IDE bis zum WLAN-Server für eine Temperaturüberwachung erklären wir im neuen Make ESP32 Special den Umgang mit dem Wifi-Mikrocontroller. Das Heft ist ab sofort exklusiv und versandkostenfrei im heise shop erhältlich: Das Bundle mit einem ESP32-NodeMCU-Entwicklerboard gibt es für 24,95 Euro zu bestellen.

Der erste Teil des 68 Seiten starken Hefts widmet sich den Grundlagen: Wir zeigen, wie Sie den ESP32 schnell über die beliebte Arduino-Programmierungsumgebung ansteuern. Was steckt in der Hardware der Wifi-Mikrocontroller? Das erklären wir und auch, wie Sie den ESP32 ins WLAN bringen oder per Bluetooth Daten austauschen. Anschließend geht es um weitere Funktionen des Boards vom Hallsensor bis zu Digital-Analog-Wandlern und wie Sie den BASIC-Interpreter freischalten.

Das gedruckte Heft mit dem ESP32 gibt es exklusiv im heise shop für 24,95 Euro versandkostenfrei. Ohne Mikrocontroller ist es als PDF für 14,90 Euro erhältlich. Weiterhin ist im heise shop der 2. Teil des Make: Arduino Special verfügbar. —hch



Der Arduino Uno und der Nano waren lange Zeit die führenden Plattformen beim Basteln mit Mikrocontroller-Boards. Der Hersteller Espressif hat mit seinen Mikrocontrollern ESP8266 und ESP32 die Maker-Community ordentlich aufgemischt und wird mittlerweile (zumindest aus der Sicht der Make-Redaktion) in mehr Projekten eingesetzt als der Arduino. Im Wesentlichen liegt das an drei Dingen: Beide Mikrocontroller sind (auf Development-Boards) günstiger als Original-Arduinos, sie bieten erheblich mehr Speicher und Funktionen und sie lassen sich trotzdem weiterhin mit der populären Arduino-IDE programmieren.

Kurz zur Historie: Bereits im Jahr 2015 erfüllte der eigentlich für industrielle Zwecke entwickelte ESP8266 die lang gehegten Wünsche vieler Maker nach Netzwerkfunktionen, die sich mit dem Arduino nur durch teure Zusatz-Shields realisieren ließen. Anfangs fungierte der ESP8266, trotz seiner im Vergleich viel potenteren Hardware, nur als günstiges WLAN-Modul für viele Arduino-Projekte. Per serieller Schnittstelle kann man Daten (mit AT-Befehlen wie bei Modems) über das WLAN-Modul **1** ins Netz schicken oder von Servern abholen. Das Modul arbeitet auf Wunsch auch als Access Point mit Webserver-Funktion.

Nach und nach wuchs die Softwareunterstützung mit dedizierten Bibliotheken für die Arduino-IDE durch die Community (esp8266.com) und Espressif, sodass der ESP8266 immer öfter auch als autark arbeitendes Modul ohne Arduino zum Einsatz kam. Als dann auch noch immer mehr Anbieter das Modul auf Prototyping-Boards zusammen mit Spannungsregler, USB-Anschluss, Stiftleiste, Tastern und LEDs verbauten, war der Weg frei für einen breiten Einsatz auch unter Einsteigern.

2017 folgte der ESP32 mit zusätzlichen Funktionen wie Bluetooth, Hallensensor und weiteren Funktionen. Seither finden sich in vielen Projekten rund um Smart Home, IoT und Robotik die Produkte von Espressif.

Details

Die Blockschaltbilder **2** und **3** zeigen die vorhandenen Funktionen des ESP8266 und des ESP32. Ersterer hat einen 32-Bit-RISC-Prozessorkern (Tensilica Xtensa Diamond Standard 106Micro) mit bis zu 160MHz Takt. Das WLAN erreicht maximal 72Mbit/s. Insgesamt bietet der ESP8266 17 GPIOs mit 3,3V-Pegel, die sich auch als Schnittstellen nutzen lassen: 2 x UART, 2 x SPI, 1 x I2C, 1 x I2S (Audio), 1 x IR sowie 1 x SDIO zum Anschluss von SD-Karten und anderen ICs. Dazu kommen ein 10-Bit-A/D-Wandler und 4 x PWM. In der Praxis kommt der ESP8266 auf Entwicklungsboard vormontiert daher, beispielsweise auf

Kurzinfo

- » Vergleich ESP8266 und ESP32
- » Besondere Funktionen des ESP32

Alles zum Artikel im Web unter make-magazin.de/x7yg

Mehr zum Thema

» Markus Ulsass, Arduino ins WLAN, Make 6/15, S. 18

dem Wemos D1 Mini oder dem NodeMCU. Ursprünglich bezeichnete NodeMCU (Kostpunkt 5 Euro) eine Firmware für den ESP8266, um ihn in der Skriptsprache Lua programmieren zu können. Das zum Projekt dazugehörige Evaluierungsboard wird auch NodeMCU genannt und ist mittlerweile eine Art Standard für die Bauweise geworden **4**.

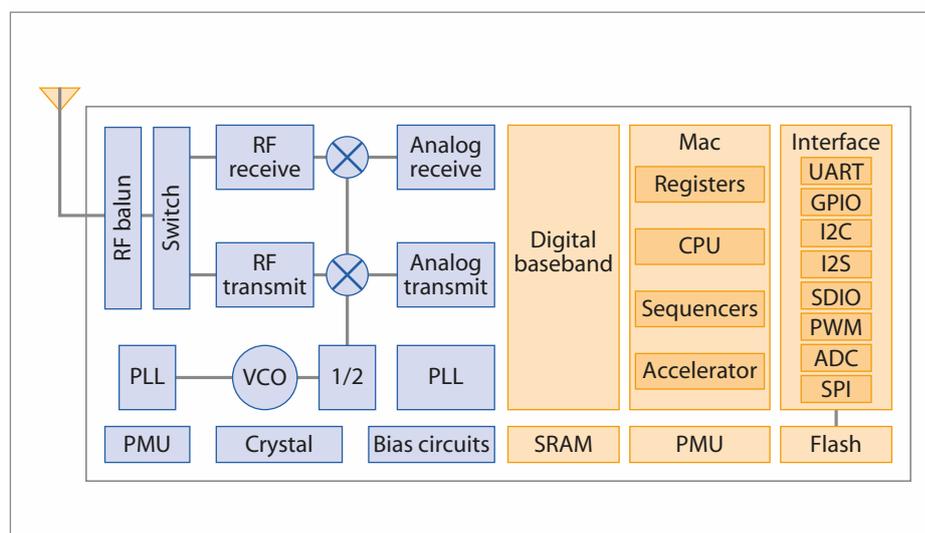
Im Unterschied zum Arduino ist in den ESP8266-Modulen bereits eine Firmware mit einem Echtzeitbetriebssystem (RTOS) vorinstalliert, das unter anderem das Management der WLAN-Funktionen übernimmt, den TCP/IP-Stack verwaltet und die Stromsparfunktionen steuert. Das interne RAM ist mit 160kByte zwar erheblich größer als im Arduino Uno (2kByte), wird aber in Teilen bereits vom RTOS belegt. Nutzt man den ESP8266 als WLAN-Client, stehen für eigene Programme nur noch 50kByte RAM zur Verfügung. Einen Flash-Speicher besitzen ESPs grundsätzlich nicht, man muss ihn über externe Bausteine hinzufügen. In der Regel sind 4MByte verbaut, was selbst für komplexere Projekte mehr als genug ist.

Um das NodeMCU-Board mit der Arduino-IDE programmieren zu können, muss

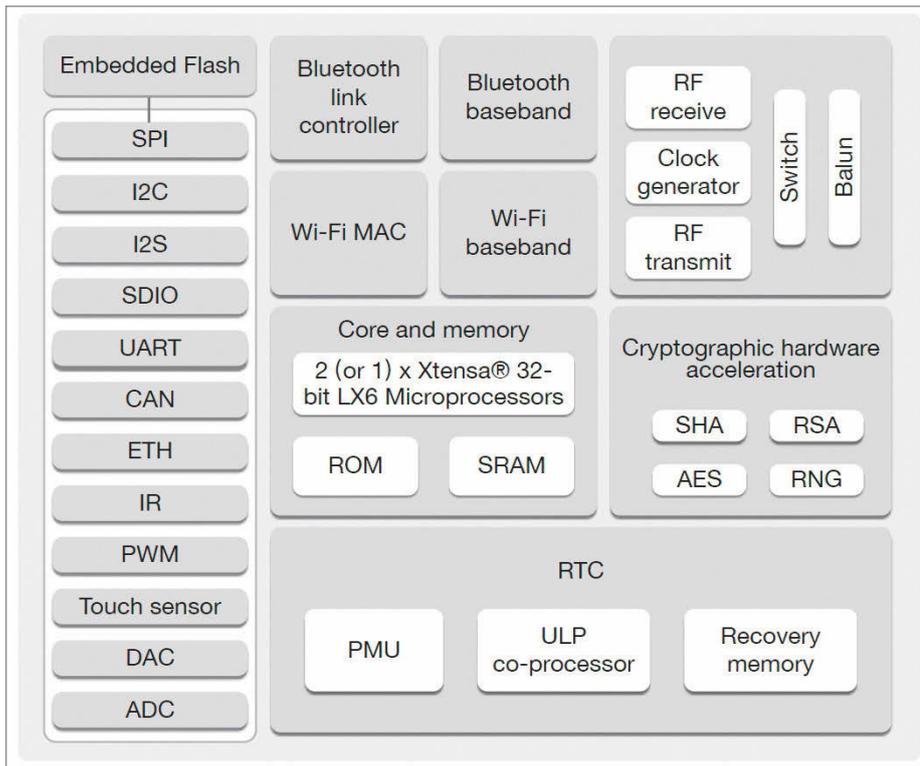
man die vom Hersteller zur Verfügung gestellten Bibliotheken und Hardwarepakete über den Boardmanager installieren. Wie das geht, finden Sie unter dem Link. Je nach benötigter Funktion fügt man im eigenen Programm die Header-Dateien hinzu, beispielsweise `#include <ESP8266WiFi.h>`, wenn man WLAN-Funktionen benötigt, oder `#include <i2s.h>`, wenn man mit dem Audio-



1 Genau genommen steht die Bezeichnung ESP8266 nur für den Chip. Im Kombination mit weiterer Hardware und anderen ICs werden dann Module draus, etwa das ESP-12.



2 Der ESP8266 bringt WLAN mit und interessante GPIO-Funktionen.



3 Der ESP32 hat sogar mehr Funktionen als ein Raspberry Pi, letztlich aber kein natives USB.

bus experimentieren möchte. In der Arduino-IDE sind unter dem Menüpunkt „Datei“ viele gut dokumentierte Beispiele zu finden, wie man die internen Funktionen, Schnittstellen und GPIOs nutzt.

Doppelt hält besser

Die enorme Funktionsvielfalt bringt den Prozessor des ESP8266 bei schlechter Programmierung allerdings in Bedrängnis. Blockierende Programmschleifen verhindern unter Um-

ständen, dass der Prozessor Hintergrundarbeiten für den Betrieb des WLAN und des TCP/IP-Stack erledigen kann. Das führt mitunter zu Abstürzen und mysteriösen Resets. Das Problem lässt sich zwar mit speziellen Funktionsaufrufen wie `yield` in eigenen Programmen lösen, beim ESP32 setzte Espressif aber von vornherein auf einen Doppelkern, um solche Probleme zu verhindern. Einer, der sich dediziert um WLAN (auf dem ESP32 mit 150MBit/s), TCP/IP und andere Dinge kümmert, und einer, der sich mit dem User-Pro-

gramm beschäftigt. Der Schritt zum Doppelkern war auch notwendig, weil der ESP32 zusätzlich die Kommunikation über Bluetooth 4.2 und LE, Ethernet und CAN unterstützt.

Zusätzlich hat der ESP32 noch einen Ultra-Low-Power-Co-Prozessor (ULP) spendiert bekommen. Der kann auch dann noch mit GPIOs und I2C arbeiten, wenn der große Bruder schläft. Der Stromverbrauch liegt bei geringen 0,8mA. Damit kann er beispielsweise Sensoren abfragen und etwa bei Überschreiten gemessener Schwellwerte den großen Bruder aufwecken, der dann das WLAN hochfährt und darüber Daten versendet.

Zusätzlich zu den bereits im ESP8266 vorhandenen Funktionen wurde der ESP32 mit einem 10-Bit-D/A-Wandler mit 2 Kanälen, einem integrierten Temperatursensor sowie einem Hallsensor und Touch-fähigen Eingängen ausgestattet. Der D/A-Wandler lässt sich zur Steuerung externer Bauelemente einsetzen, etwa zur Ansteuerung von Transistoren und MOSFETs, oder um eine Folge von Audiosamples abzuspielen.

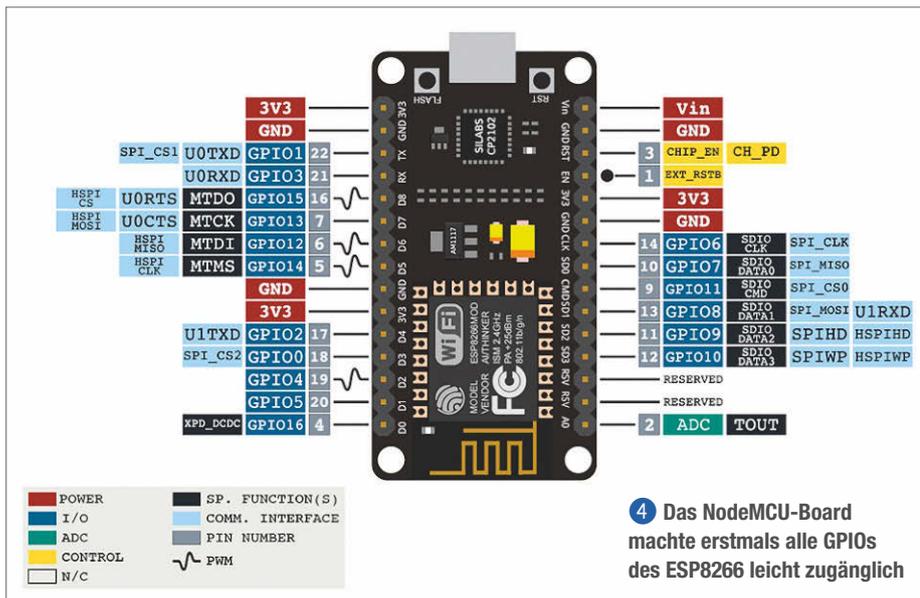
Mit dem Hallsensor lassen sich Magnetfelder in der Umgebung messen. In der Praxis kann der ESP32 damit feststellen, ob ein mit einem Magneten versehenes Objekt in der Nähe ist oder nicht. In der Industrie nutzt man das unter anderem in Autos, um festzustellen, ob der Gurt geschlossen ist. Im Haushalt könnte man es für Türen und Fenster nutzen, um festzustellen, ob diese offen oder geschlossen sind.

Eine weitere Spezialität, die den ESP32 von der Konkurrenz abhebt, ist die eingebaute Unterstützung für Berührungssensoren (Touch-Sensoren). Er nennt gleich zehn kapazitive Eingänge 5 sein Eigen und damit lassen sich neben vielen nützlichen Anwendungen auch schnell ein paar Spaß-Projekte umsetzen, etwa ein Synthesizer mit Obst oder Knete als Eingabelemente.

Der Temperatursensor im Inneren des ESP32 ist leider nicht sonderlich genau und liegt in einem Bereich des Chips, in dem er bei hoher Last die Abwärme seiner Kollegen mitbekommt und auch misst. Für absolute Messungen taugt er daher ohne eine ordentliche Kalibrierung nicht, aber er kann unter Umständen verwendet werden, um Schwankungen in der Umgebungstemperatur zu erkennen.

Für Lichtprojekte ist der integrierte LED-PWM-Controller praktisch. Mit ihm lässt sich die Intensität von LEDs auf sechzehn Kanälen auf vielfältige Weise steuern. Bei Bedarf erzeugen die sechzehn Kanäle sogar unterschiedliche Schwingungsformen, um RGB-LEDs zu kontrollieren.

Neben diversen Timern, die für hohe Flexibilität bei der Erzeugung schicker Muster sorgen, helfen Funktionen zum Ein-, Aus- und Überblenden von LEDs bei der kreativen Beleuchtung.



4 Das NodeMCU-Board machte erstmals alle GPIOs des ESP8266 leicht zugänglich

Bild: Acrobot, CC-BY-SA

Geheim

Weil auch in industriell eingebetteten Systemen das Thema Sicherheit immer wichtiger wird, hat Espressif den ESP32 mit Verschlüsselungsfunktionen ausgestattet. Die lassen sich natürlich auch zur Absicherung eigener Hobby-Projekte einsetzen, etwa Smart Home. Starke kryptographische Algorithmen stellen aber selbst für eine vergleichsweise leistungsstarke CPU, wie sie auf dem ESP32 ihren Dienst verrichtet, eine große Herausforderung dar.

Bei der symmetrischen Verschlüsselung greift der Tensilica-CPU deshalb ein AES-Beschleuniger unter die Arme. Der Advanced Encryption Standard (AES) gehört zu den wichtigsten symmetrischen Verschlüsselungsalgorithmen und der ESP32 kommt mit Schlüsseln der Länge 128, 192 und 256 Bit klar.

Zur asymmetrischen Verschlüsselung setzt man häufig noch den RSA-Algorithmus ein. Hier bietet der ESP32 zwar keine direkte Unterstützung, immerhin hilft er aber bei der zugrunde liegenden Mathematik. Die basiert auf modularer Arithmetik mit sehr großen ganzen Zahlen und dabei müssen diese Zahlen unter anderem multipliziert werden. Der ESP32 beherrscht die Multiplikation von Zahlen mit einer Länge von bis zu 2048 Bits und erlaubt die modulare Potenzierung von Zahlen mit einer Länge von bis zu 4096 Bits. Damit lässt sich der RSA-Algorithmus mit den heutzutage empfohlenen Schlüssellängen in Software implementieren – eine solche Implementierung gehört auch zur Standard-Bibliothek.

Community

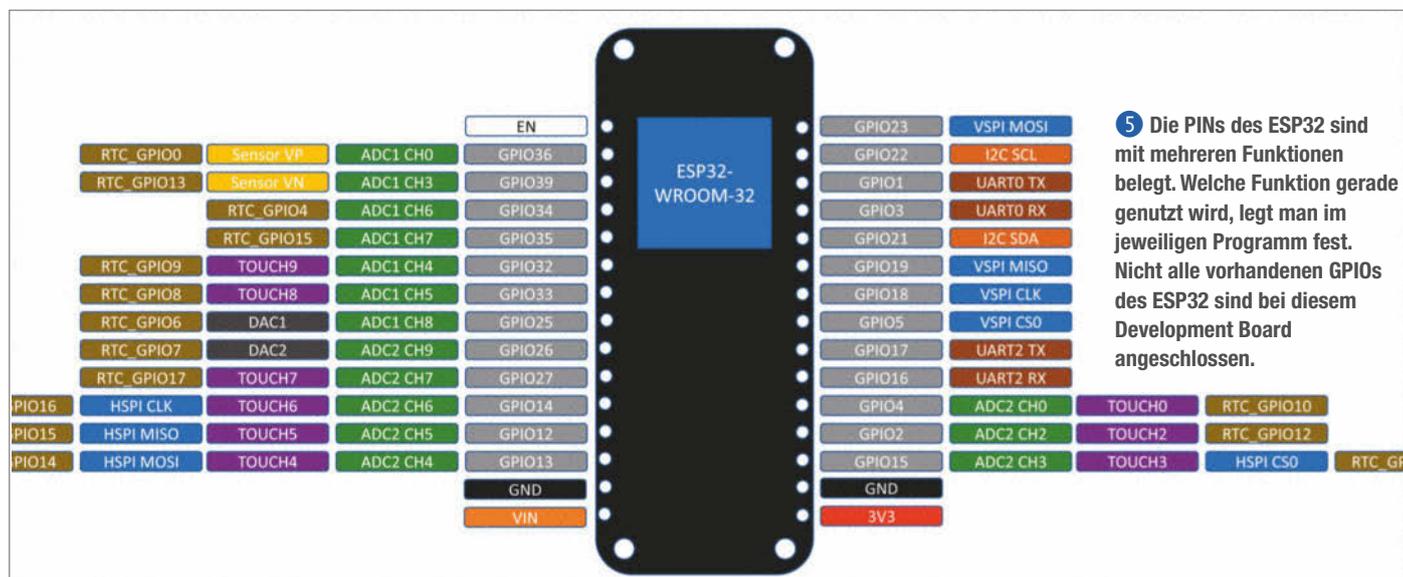
Sowohl mit dem ESP32 als auch dem ESP8266 wird jedes Projekt für wenig Geld netzwerkfähig. Im Moment scheint die Ver-

Mikrocontroller

	ESP32	ESP8266	Arduino Uno
CPU	Xtensa Dual-Core 32-bit LX6	Xtensa Single-core 32-bit L106	ATmega 328P
max. Taktfrequenz CPU	240MHz	160MHz	16MHz
SRAM	520KB	160KB	2KB
Flash	meistens 4MB	meistens 4MB	32KB
GPIOs	36	17	14
SPI-Schnittstellen	4	2	1
I2C-Schnittstellen	2	1	1
I2S-Schnittstellen	2	1	–
UART-Schnittstellen	3	2	1
CAN 2.0-Schnittstellen	1	–	–
SD/SDIO/MMC Host-Controller	1	–	–
SDIO/SPI Slave-Controller	1	1	–
ADC-Kanäle	18 (12 Bit)	1 (10 Bit)	6 (10 Bit)
IR-Funktionalität	ja (Hardware)	ja (größtenteils Software)	–
Ethernet MAC-Schnittstelle	1	–	–
GPIOs für Touch-Sensoren	10	–	–
Hall-Sensor	1	–	–
Temperatur-Sensor	1	–	–
DAC-Kanäle	2 (8 Bit)	–	–
WLAN	802.11 b/g/n	802.11 b/g/n	–
Bluetooth	Bluetooth 4.2 und BLE	–	–
Betriebsspannung	2,3V bis 3,6V	2,3V bis 3,6V	7V bis 12V

breitung des ESP8266 und damit auch die Community größer zu sein als die für den ESP32. Das liegt womöglich auch daran, dass die Integration für den ESP8266 direkt von der Community entwickelt wurde, während die für den ESP32 vom Hersteller Espressif selbst kam. Ist man auf der Suche nach der Lösung für ein Problem, stößt man beim

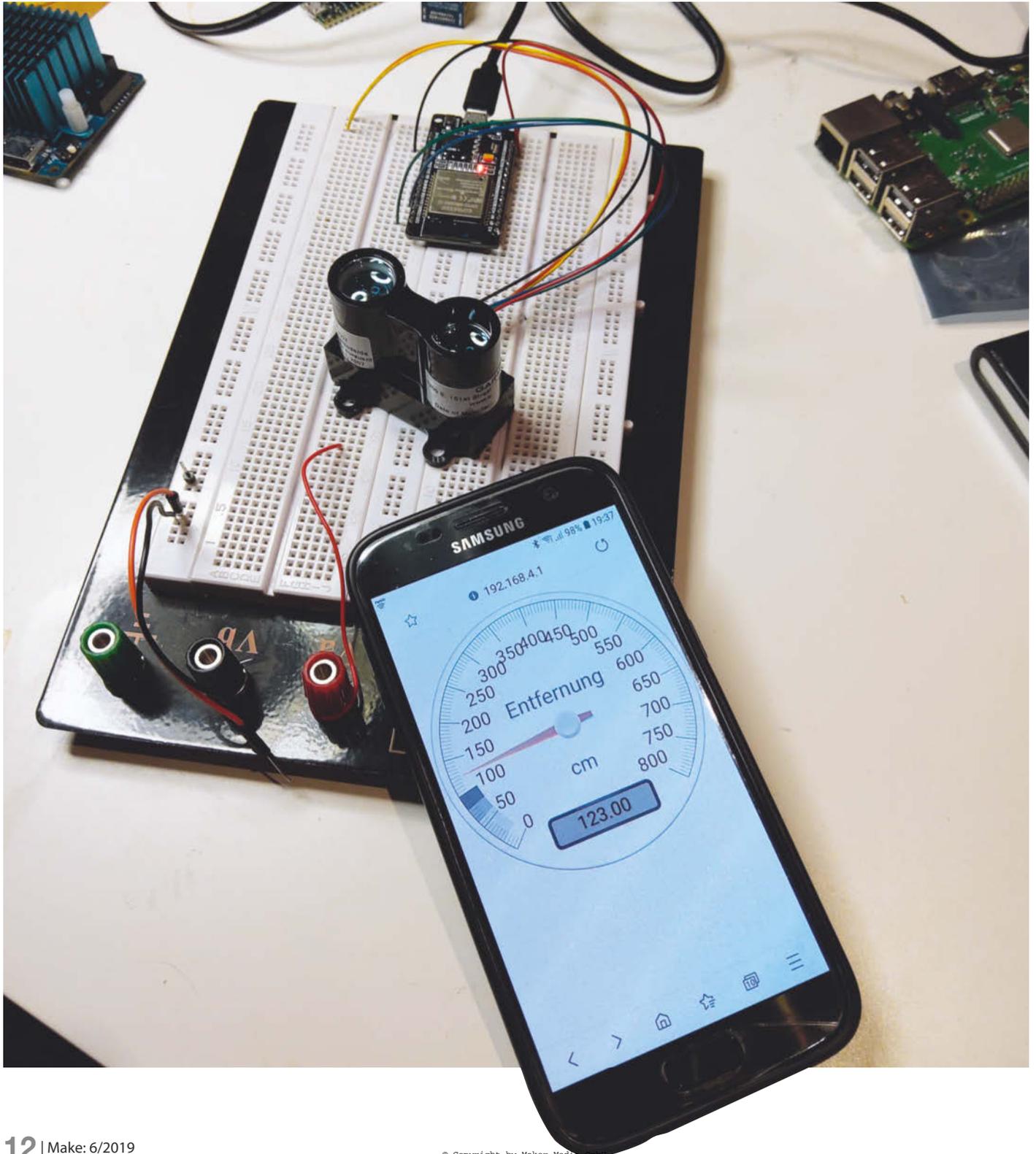
ESP8266 eher auf Unterstützung oder Beschreibungen. Für viele Anwender scheint der ESP32 zu „oversized“ fürs eigene Projekt, weshalb die Wahl auf den kleineren Bruder fällt. Nichtsdestotrotz lohnt eine Beschäftigung mit dem ESP32. Auf den folgenden Seiten finden Sie Praxisprojekte für den ESP32 und den ESP8266. Viel Spaß! —*dab*



Laserentfernungsmesser

Ob zum konkreten Messen einer Entfernung oder zur Überwachung eines Abstands: Mit unserem leicht nachzubauenden webbasierten Laserentfernungsmesser ermitteln Sie zentimetergenau Distanzen und zeigen sie auf PCs, Smartphones oder Tablets an.

von Daniel Bachfeld



Distanzsensoren auf Basis von Infrarot oder Ultraschall sind zwar günstig, aber meist recht ungenau, relativ langsam und ihre Reichweite liegt bei unter 4 Metern. Wer weiter, schneller und genauer messen will, greift auf LIDARs (laser detection and ranging) zurück, die sich dank fertiger Bibliotheken und von jedem Mikrocontroller unterstützten Schnittstellen in Windeseile in eigene Projekte integrieren lassen. In unserem Beispiel nutzen wir einen WLAN-fähigen ESP32, um die Messwerte des LIDAR über einen selbst programmierten Simple-Webserver grafisch zur Verfügung zu stellen, sodass – geräteunabhängig – jeder Browser die Daten abrufen und anzeigen kann. Eine kurze Einführung in die Arbeitsweise eines LIDAR finden Sie im Kasten „LIDAR-Technik“.

Der hier eingesetzte LIDAR **1** stammt vom Hersteller Garmin, sonst eher für Navigations-Geräte und smarte Uhren bekannt. Das Modul hat insgesamt 6 Pins: Vcc (rotes Kabel), Power Enable (orange), Mode (gelb), SCL (grün), SDA (blau) und GND (schwarz). Vcc erwartet eine Spannung zwischen 4,5 und 5,5 Volt, die der ESP32 an seinem Pin Vin bereithält. Normalerweise dient Vin dazu, eine Spannung an den Spannungsregler des ESP32 einzuspeisen, falls er nicht über seinen USB-Port versorgt wird. Hier nutzen wir den Port quasi als Ausgang, da er (über eine Schutzdiode) an der 5V-Versorgung des USB-Ports hängt.

Pins

Die Pins *Power Enable* und *Mode* können wir ignorieren und müssen sie nicht anschließen. Sie sind intern bereits mit Pull-up-Widerständen verschaltet, sodass sie uns nicht in die Quere kommen. *SDA* und *SCL* sind die Leitungen des I2C-Bus, über die der LIDAR mit 400kHz seine Daten sendet oder Befehle empfängt. Die Verschaltung des ESP32 DevKit mit dem LIDAR ist in **2** zu sehen.

Garmin stellt eine Bibliothek (LIDARLite.h) für die Arduino-IDE zur Verfügung, die praktischerweise Funktionen zur Initialisierung, Konfiguration und Abfrage der Messwerte enthält. Die Bibliothek steht über Github zum Download zur Verfügung (siehe Link), die sich über den Bibliotheks-Manager per ZIP-Import einfach integrieren lässt. Damit ist ein erster

ESP32 und die Arduino-IDE

Eine Anleitung, wie man die Arduino-IDE für die Unterstützung des ESP32 fit macht, finden Sie unter dem Link in der Kurzinfor!

Kurzinfor

- » LIDAR verstehen
- » Anschluss an den ESP32
- » Webserver programmieren

Checkliste



Zeitaufwand:
etwa 1 Stunde



Kosten:
etwa 150 Euro



Programmieren:
Bedienung der Arduino-IDE



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

Material

- » Garmin Lidar Lite v3
- » ESP32-Board Dev Kit
- » evtl. Breadboard
- » Jumperkabel
- » Smartphone
- » USB-Powerbank

Mehr zum Thema

- » Markus Knapp, Distanzsensoren, Make Sonderheft Robotik, S. 98

lidarlite.ino

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

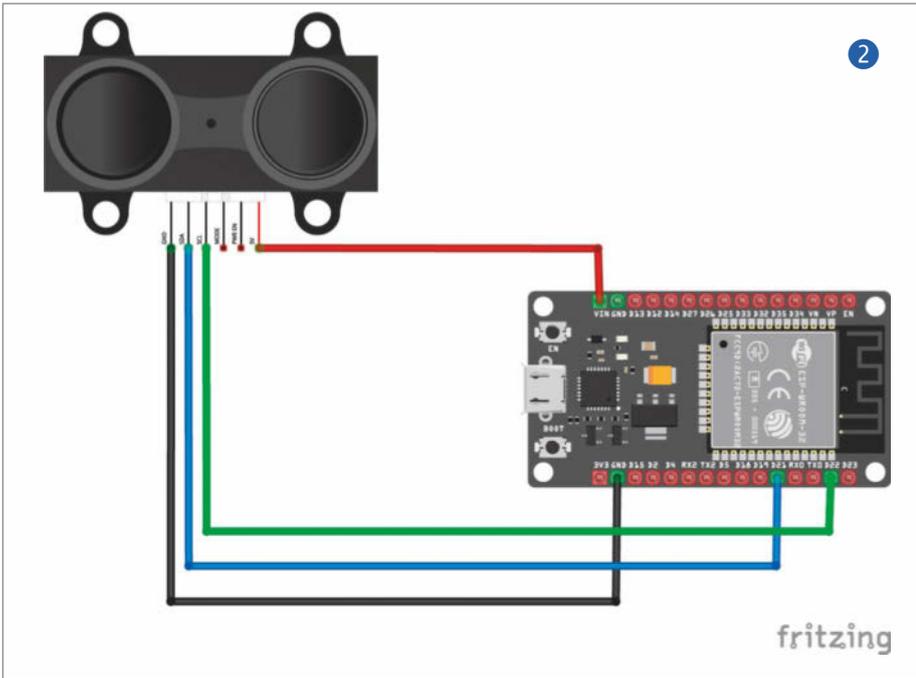
LIDARLite myLidarLite;

void setup()
{
  Serial.begin(115200);
  myLidarLite.begin(0, true);
  myLidarLite.configure(0);
}

void loop()
{
  Serial.println(myLidarLite.distance());
}
```

1





Sketch für den LIDAR schnell mit wenigen Zeilen gebaut, wie in Listing lidarlite.ino zu sehen ist. Der Sketch ist selbsterklärend, anzumerken ist noch, dass sich verschiedene Modi mit dem Befehl `myLidarLite.configure()` einstellen lassen, in denen der LIDAR jeweils besser im Nah- oder Fernbereich arbeitet oder langsamer und dafür genauer misst.

Webserver

Das kurze Beispiel würde natürlich keinen Artikel in der Make rechtfertigen. Deshalb nutzen wir die Fähigkeiten des ESP32 und ergänzen den kurzen Sketch noch um einen Webserver und einen Access Point, um die Daten nicht einfach nur über die serielle

Schnittstelle auszugeben. Listing lidar-webserver.ino zeigt den Sketch. Trotz der wenigen Zeilen schafft er es, einen Access Point aufzuspannen, einen Webserver zu starten und die LIDAR-Daten grafisch anzuzeigen. Die Magie versteckt sich hinter den Bibliotheken `Wifi.h`, `ESPAsyncWebServer.h` und `SPIFFS.h`.

`Wifi.h` übernimmt alles rund ums WLAN. Mit `ESPAsyncWebServer.h` ist zudem ein Multisession-fähiger Webserver schnell gebaut. Dazu müssen die genannte und die Bibliothek `AsyncTCP` zur Arduino-IDE hinzufügen. Beide stehen auf Github (siehe Link) zur Verfügung und werden ebenfalls als ZIP-Datei importiert – wie zuvor die LIDAR-Bibliothek.

Dateisystem

Beim genauen Studieren des Sketches wundern Sie sich womöglich, wo die vom Server aufgerufenen Dateien `gauge.min.js` und `index.html` liegen. Zu den bisher verwendeten Bibliotheken wird am Anfang des Programms noch die `SPIFFS`-Bibliothek (Serial Peripheral Interface Flash File System) eingebunden. Sie ist nach der Installation der ESP32-Unterstützung verfügbar und macht es möglich, Teile des Flash-Speichers des ESP32 als Dateisystem zu betrachten und dort Dateien zu erzeugen, zu lesen und zu modifizieren. Die Bibliothek wird in der `setup`-Funktion durch den Aufruf von `SPIFFS.begin` initialisiert. Doch wie kommen die Dateien in den ESP32?

Um den Speicher auch aus der Arduino-IDE heraus nutzen und beschreiben zu können, dient das Tool „Arduino ESP32 Filesystem Uploader“. Es erweitert die IDE um die Möglichkeit, beliebige Dateien in den Flash-Speicher des ESP32 zu kopieren. Dazu muss man das Plug-in herunterladen (siehe Link) und auf dem eigenen PC im Arduino-Ordner im Unterordner `tools` ablegen (Neustart der IDE nicht vergessen). Nun ist im Menü Werkzeuge ein neuer Eintrag namens „ESP32 Sketch Data Upload“ zu sehen.

Damit lädt man die Dateien `index.html` (siehe Listing `index.html`) und `gauge.min.js` vom PC auf den ESP32. Die beiden Dateien müssen im Ordner des Sketches `lidar-webserver.ino` im Unterordner `data` liegen.

Statt die Dateien mit viel Gewurschtel in den Quelltext einzubetten, werden sie vom ESP32 jetzt einfach mit der Klasse `SPIFFS` aus dem Flash-Speicher gelesen. Den Content-Type ermittelt die `AsyncWebServer`-Bibliothek in diesem Fall automatisch. Die Datei `gauge.min.js` enthält dabei ein JavaScript samt Bibliothek, mit der sich grafische Anzeigen wie Thermometer oder Tachos im Browser anzeigen lassen. Die Datei `index.html` setzt die Parameter für die Anzei-

lidar-webserver.ino

```
#include <WiFi.h>
#include <ESPAsyncWebServer.h>
#include <SPIFFS.h>
const char* SSID = "MeinESP32";
const char* PASSWORD = "testpasswort";
#include <LIDARLite.h>
LIDARLite myLidarLite;
AsyncWebServer server(80);

void setup(){
  Serial.begin(115200);
  if(!SPIFFS.begin(true)){
    Serial.println("Dateisystem konnte nicht initialisiert werden.");
    return;
  }
  myLidarLite.begin(0, true);
  WiFi.softAP(SSID, PASSWORD);
  Serial.print("IP-Adresse: ");
  Serial.println(WiFi.softAPIP());
  server.on("/gauge.min.js", HTTP_GET, [](AsyncWebServerRequest* request)
  {
    request->send(SPIFFS, "/gauge.min.js");
  });
  server.on("/", HTTP_GET, [](AsyncWebServerRequest* request) {
    request->send(SPIFFS, "/index.html", String(), false, replaceVariable);
  });
  server.begin();
}

void loop() {}
String replaceVariable(const String& var) {
  if (var == "ENTFERNUNG")
    return String(myLidarLite.distance(false));
}
```

ge, etwa Größe des Elements, Wertebereich, Einheiten, Skalierung und Beschriftung, und ruft dann das JavaScript auf. Der Platzhalter %ENTFERNUNG wird vom Webserver-Programm (genauer gesagt von der AsyncWebServerRequest-Klasse) durch die LIDAR-Messwerte ersetzt.

Inbetriebnahme

Sind nun alle Bibliotheken und Tools installiert, die Dateien über den Filesystem-Uploader auf den ESP32 geladen und der Sketch fehlerfrei übersetzt und ebenfalls hochgeladen, nimmt man das Smartphone oder Tablet zur Hand und wählt sich ins WLAN „MeinESP32“ ein (Passwort testpasswort). Im Browser ruft man <http://192.168.4.1> auf und sollte nun eine Seite wie in 3 sehen. Der Inhalt wird alle 3 Sekunden aktualisiert und zeigt die gemessene Entfernung einmal als Dezimalzahl und einmal als Tachostand an. Sollte keine Seite aufgebaut werden, kann es helfen, die mobile Datenverbindung auf dem Smartphone auszuschalten. Zur Kontrolle, ob der Sketch auch läuft, wie er soll, und ein WLAN aufgespannt hat, kann man am PC auch die Meldungen über den seriellen Monitor anzeigen lassen.

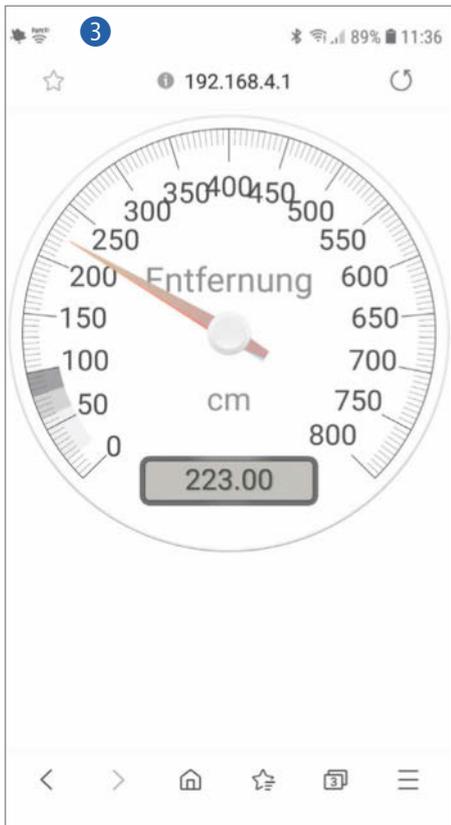
Apropos Anzeige: Die Parameter für die Tacho-Anzeige lassen sich leicht in der Datei `index.html` anpassen. Benötigen sie eine größere Anzeige, ändern Sie einfach `data-width='1000'` und `data-height='1000'`. Er-

```
index.html
<!doctype html>
<html>
  <head>
    <title>ESP32 Entfernungsmesser</title>
    <meta http-equiv="refresh" content="3" >
  </head>
  <body>
    <canvas id='lidar'
      data-width='1000'
      data-height='1000'
      data-type='radial-gauge'
      data-title='Entfernung'
      data-units='cm'
      data-max-value="800"
      data-major-
ticks="0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800"
      data-value='%ENTFERNUNG%'>
    </canvas>
    <script src='gauge.min.js'>
  </script>
  </body>
</html>
```

warten Sie Messwerte jenseits der 800 Zentimeter, ändern Sie `data-max-value= 800` und passen die Zeile `data-major-ticks= 0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800` entsprechend der neuen Einteilung an. Die Änderungen nimmt man in der auf dem PC liegenden Datei vor und lädt sie anschließend über das Upload-Tool wieder auf den

ESP32. Weitere Ideen für Gauges finden Sie unter <http://canvas-gauges.com/documentation/examples/>.

Benötigen Sie keine mobile Version des Entfernungsmessers, können Sie den ESP32 auch einfach stationär ins lokale WLAN einbuchsen. Wie das geht, sehen Sie in den Beispielen der Arduino-IDE. Der Rest des Codes bleibt gleich. —*dab*



LIDAR-Technik

Um die Entfernung zu einem vor ihm liegenden Objekt zu messen, schickt der Laser im einfachsten Fall einen Lichtimpuls aus und ein Prozessor misst die Zeit, bis er den reflektierten Impuls wieder empfängt (Time of Flight, ToF). Anhand der Zeit und der bekannten Lichtgeschwindigkeit berechnet der Prozessor nun die Entfernung. Da das Licht immerhin 300.000km/s schnell ist, muss man gute Messelektronik haben, um Laufzeiten im – je nach Entfernung – Pico- oder Nanosekundenbereich zu messen. Bis vor wenigen Jahren kostete solch ein Modul noch mehrere hundert Euro und war für die meisten Hobby-Projekte einfach zu teuer.

Vor rund 4 Jahren kamen für rund 100 Euro die ersten erschwinglichen Module „LIDAR Lite“ auf den Markt, die ein vereinfachtes Konzept einsetzen, das sich kostengünstiger in Silizium gießen lässt. Ver-

einfacht gesagt sendet der Laser kontinuierlich moduliertes Licht im Mehrfachpulsbetrieb aus und misst die Phasenverschiebung zwischen ausgesendeten und empfangenen Pulsen. Dies entspricht ebenfalls einer Zeit und ist somit proportional zur Entfernung. Seit dieser Vereinfachung findet man LIDARs auch in vielen Hobby-Projekten rund um Robotik und Drohnen, um rechtzeitig Hindernisse zu erkennen.

Das hier verwendete Modul Lidar Lite v3 des Herstellers Garmin kostet knapp 130 Euro und eignet sich durch das geringe Gewicht (22g) und die kleinen Abmessungen (20 x 48 x 40 mm) für mobile Anwendungen. Seine Stromaufnahme ist mit 135mA nicht gerade niedrig, aber für den Einsatz unterwegs immer noch akzeptabel. Seine maximale Reichweite liegt bei 40m, wobei der Fehler +/-2,5cm beträgt. Pro Sekunde kann der LIDAR bis zu 500 Messwerte sammeln und über seinen I2C-Bus an einen Mikrocontroller senden.

Lovebox

die moderne Liebesbotin

Advent, Advent, ein Lichtlein brennt. Erst eins, dann zwei, dann drei, dann vier und plötzlich steht Weihnachten vor der Tür. Wie jedes Jahr startet die Suche nach schönen Geschenken für die Liebsten. Mit der Lovebox schlagen die Herzen höher und der Einkaufsstress hat ein Ende.

von Lisa Ihde



Was sind neben Wollsocken und selbstgebackenen Plätzchen die Klassiker unter dem Weihnachtsbaum? Natürlich selbstgebastelte Geschenke. Als kleines Kind reichte ein Bild oder ein gefalteter Fröbelstern, um ein persönliches Geschenk herzuzaubern. Heute darf es etwas komplexer sein: Die Lovebox ist eine Kombination aus kreativem Basteln und ein wenig Programmieren. Auf ihrer Vorderseite hat die kleine Holzschachtel ein bewegliches Herz. Wenn der Deckel der Lovebox angehoben wird, kommt ein kleiner Bildschirm zum Vorschein. So kann man positive und liebevolle Notizen an die Liebsten senden, die die Box über WLAN empfängt. Geht eine Nachricht ein, beginnt sich das Herz auf der Vorderseite der Lovebox zu drehen. Es bewegt sich so lange, bis der Empfänger oder die Empfängerin den Deckel der Box anhebt, um die geheime Nachricht auf dem Bildschirm zu lesen **1**. Die Lovebox ist somit eine Art moderner Liebesbote.

Erfinder der Lovebox ist Jean Gregoire, der für anderthalb Jahre zum Forschen in die USA zog, während seine Verlobte in Frankreich blieb. Er wollte ihr trotz der großen Distanz seine Liebe zeigen und baute in einem FabLab die Lovebox. Mittlerweile hat das Projekt richtig Form angenommen und wird hochwertig produziert. Weitere Informationen dazu sind unter en.lovebox.love zu finden. Die Lovebox kann aber ganz einfach selber gebaut werden. Unsere Lovebox wird ein Gehäuse aus dem Lasercutter erhalten und nutzt zur Anzeige ein OLED-Display. Das Herz auf der Vorderseite bewegt ein Servomotor. Außerdem verbauen wir einen Helligkeitssensor, um festzustellen, ob der Deckel der Box geöffnet wurde. Die Steuerung übernimmt ein ESP8266-Mikrocontroller. Schließlich müssen wir die Nachrichten noch jeweils für das kleine Display aufbereiten. Für die Kommunikation über das Internet nutzen wir den Codesnippet-Dienst Gist von Github. Nach der Erklärung der verwendeten Komponenten folgt am Ende eine Schritt-für-Schritt-Anleitung zum Nachbau.

Mehr Liebe mit IoT

Die Lovebox eignet sich nicht nur für Liebesbotschaften, sondern zum Beispiel auch für motivierende Sprüche, die jeden Tag gesendet werden können. Die Nachrichten werden dabei kabellos übertragen, denn das Herzstück bildet das WeMos-D1-Mini-Board mit einem WLAN-fähigen ESP8266-Modul. Neben den Nachrichten können sogar kleine Zeichnungen versendet werden. Das Board ist nur wenige Zentimeter groß und somit fast deckungsgleich mit dem OLED-Display mit 0,96 Zoll (ca. 2,4cm). Es verfügt über eine Auflösung von 128 × 64 Pixeln. Der Servomotor

Kurzinfo

- » Nachrichten über Github Gist senden
- » Binäre Bilder für OLED-Display erstellen
- » Elektronik im Mini-Kästchen stapeln

Checkliste



Zeitaufwand:
etwa 2 bis 3 Stunden



Kosten:
circa 12 Euro



Löten:
einfache Lötarbeiten



Maschinen:
Lasercutter



Programmieren:
Arduino-Grundkenntnisse

Werkzeug

- » Schere
- » Holzleim
- » LötKolben
- » Lötzinn
- » Abisolierzange
- » Lineal

Alles zum Artikel
im Web unter
make-magazin.de/xf4z

Material

- » Holz (3mm dick, ca. A5-groß) für das Gehäuse und das Herz
- » Mikrocontroller WeMos D1 Mini mit Netzteil
- » Servo-Motor (SG90)
- » Helligkeitssensor
- » Widerstand 10kΩ
- » OLED-Display (128 × 64 Pixel, SSo1306)
- » Lochrasterplatine
- » Pinleiste
- » Draht oder Jumper-Kabel
- » transparentes Acryl (1mm dick, 4 × 4cm) für das Display
- » schwarze Blendschutzfolie, wie sie zum Sonnenschutz an Autoscheiben genutzt wird
- » rote Acrylfarbe

Mehr zum Thema

- » Lukas Scheffler, Digitaler Bilderrahmen mit Messenger-Funktion, Make 5/19, S. 32
- » Wasili Adamow, Patrick Beedgen, Marc Hassenzahl, Kirstin Kohler, Eva Lenz, Thies Schneider, Digitales Familientreffen, Make IoT Special 2016, S. 104



1 Wenn eine neue Nachricht empfangen wurde, bewegt sich das rote Herzchen auf und ab, bis der Deckel geöffnet wird.



2 Diese Folie wird meist als Blindstreifen für Autofensterscheiben verkauft, um vor Sonnenstrahlen zu schützen.

Holz für das Gehäuse und ein kleines Stück einer durchsichtigen und sehr dünnen Acryl-Platte. Die Platte wird mit Blendschutzfolie 2 beklebt, um dahinter das OLED-Display zu verstecken. Diese Folie wird beispielsweise in Autos an den Fensterscheiben genutzt, um sich vor Sonnenstrahlen zu schützen. Bei uns dient die Acryl-Platte mit aufgeklebter Folie als ein halbtransparenter Spiegel. Außerdem wirkt das Display optisch größer, als würde es bis zu den Rändern der Box reichen.

Nachrichten senden

Wie kommen nun die Nachrichten auf das Display der Lovebox? Wie schon erwähnt besitzt der Mikrocontroller ein WLAN-fähiges ESP8266-Modul. Darüber kann er HTTPS-Requests an einen Server stellen und checken, ob eine neue Nachricht eingetroffen ist und abgerufen werden kann. Für die Übertragung brauchen wir ein privates Github Gist, welches in der Regel zum Teilen von Code-Schnipseln genutzt wird. Im Unterschied zu öffentlichen Gists werden private Gists nicht in der Gist-Übersicht angezeigt und sind auch nicht über die Suche auffindbar. Sobald man die URL kennt, kann man aber alles mitlesen. Unser Gist verwalten wir über gist.github.com. Dort kann man ein Dokument anlegen und mit Inhalt füllen. Diesen Inhalt fragt die Lovebox dann ab. Der Inhalt kann jederzeit geändert werden. Wie bei Github üblich, sind die Änderungen als Revisionen nachlesbar.

Dafür benötigen wir im Dokument drei Informationen: eine Identifikationsnummer, einen Nachrichtentyp und die Nachricht selbst. Die erste Zeile im Dokument dient zur Identifikation und kann eine beliebige Zahl sein. Wenn sie sich von der aktuell gespeicherten unterscheidet, weiß die Lovebox, dass eine neue Nachricht vorliegt. Somit wird die zweite Zeile angeschaut, welche für den Nachrichtentyp steht. Wenn diese ein „t“ enthält, erwartet der Mikrocontroller einen Text und zeigt die Nachricht auf dem Display mit der Methode `drawStringMaxWidth()` an. Diese Nachricht ist ab der dritten Zeile zu finden, sie kann aber länger als eine Zeile sein. Die ersten zwei Zeilen bestimmen also die Darstellung und die nächsten enthalten die eigentliche Nachricht. Wenn ein anderes Zeichen in der zweiten Zeile steht, etwa ein „b“, wird ein Bild erwartet. Bilder müssen wir für die Darstellung allerdings in Folgen aus Nullen und Einsen umwandeln.

3 Screenshot des Konverters von Dcode, der Bilder in einen String aus Nullen und Einsen verwandelt.

Einstellungen Lovebox

```

1 #include <ESP8266WiFi.h>
2 #include <WiFiClientSecure.h>
3 #include <EEPROM.h>
4 #include <Servo.h>
5 #include "SSD1306Wire.h"
6
7 const char* ssid = "WLAN";
8 const char* password = "Passwort";
9 const String url = "Gist-Dokument-Url";
    
```

Am Beginn des Codes werden die benötigten Bibliotheken zum Ansteuern des Servos, des Displays und der Nutzung von WLAN eingebunden.

ist noch kleiner als das Display und das Wemos-Board. Daher passt alles in eine kleine Box.

Zum Ansteuern des OLED-Displays existieren verschiedene Bibliotheken. Meist wird die SSD1306-Bibliothek in Verbindung mit der GFX-Bibliothek von Adafruit genutzt. Leider funktioniert aber das Encoding von einigen Zeichen, wie etwa Umlauten, nicht und man kann keine individuelle Schriftart wäh-

len. Wir nutzen daher die Bibliothek *ESP8266 and ESP32 Oled Driver for SSD1306 display* von Daniel Eichhorn und Fabrice Weinberg. Diese bringt verschiedene Funktionen mit, beispielsweise das Anzeigen von X-Bitmaps, also monochromen Rastergrafiken. Weiteres kann man in der Projekt-Doku auf Github nachlesen (Link siehe Kurzinfo).

Neben den bisher erwähnten Komponenten benötigen wir für die Lovebox ein wenig





Für den Bau benötigen wir schließlich noch das Holzkästchen. Als Vorlage gibt es vier Vektorgrafiken mit den Schnittplänen, die im Internet heruntergeladen werden können (siehe Link in der Kurzinfo) 5. Die Dateien können dann beispielsweise mit dem Open-Source-Grafikprogramm Inkscape angepasst werden, um die Box größer oder kleiner zu bauen oder eine andere Deko als das Herz zu verwenden.

Software vorbereiten

Zur Programmierung des Mikrocontrollers nutzen wir die Arduino-Programmierungsumgebung (Download siehe Kurzinfo, dort gibt es auch eine Anleitung zur Installation). Um den WeMos D1 Mini nutzen zu können, müssen wir noch den Treiber installieren. Dazu fügen wir unter *Datei/Voreinstellungen/Zusätzliche Boardverwalter-URLs* die folgende Adresse hinzu:

http://arduino.esp8266.com/stable/package_esp8266com_index.json. Anschließend muss der Boardverwalter mit *Werkzeuge/Board/Boardverwalter* geöffnet und dort ESP8266 in das Suchfeld eingegeben und installiert werden. Nun erscheint unter *Werkzeuge/Board* das *LOLIN(WEMOS) D1 R2 & mini* und kann ausgewählt werden.

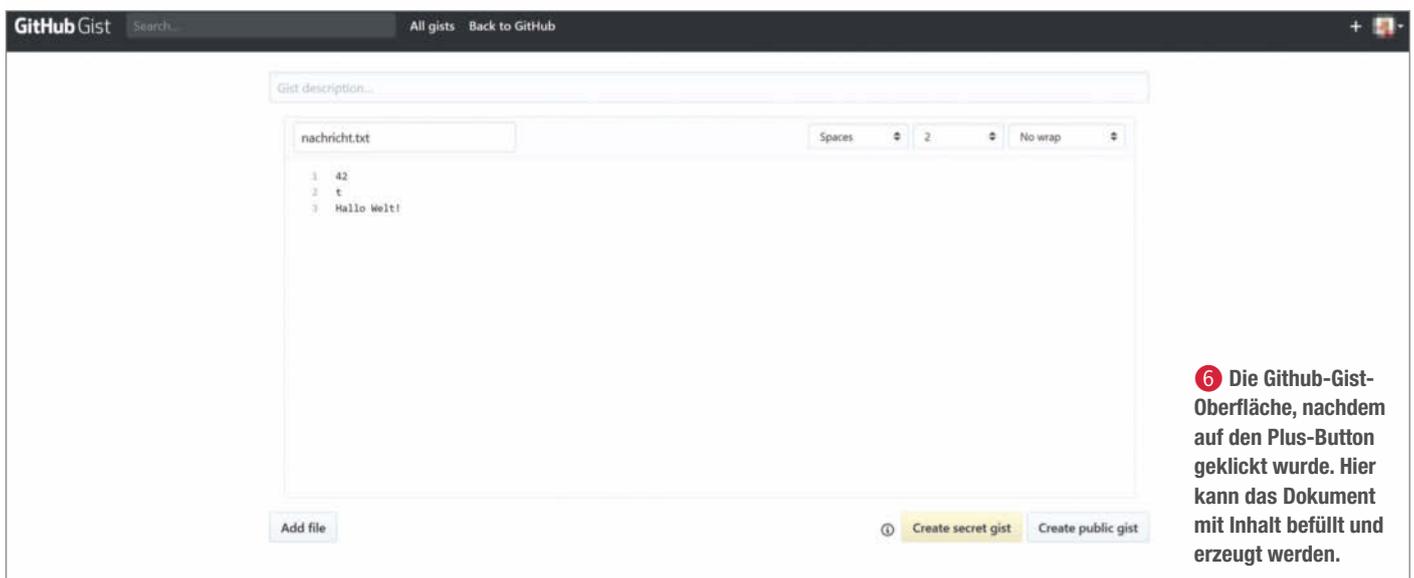
Ähnlich funktioniert die Installation der erwähnten Bibliothek *ESP8266 and ESP32 Oled Driver for SSD1306 display*. Dazu müssen wir „SSD1306“ im Suchfeld unter *Sketch/Bibliothek einbinden/Bibliotheken verwalten...* eingeben und dann die entsprechende Library installieren. Die Bibliothek zum Ansteuern des Servo-Motors wird bei der Installation von der Arduino-IDE mitgeliefert.

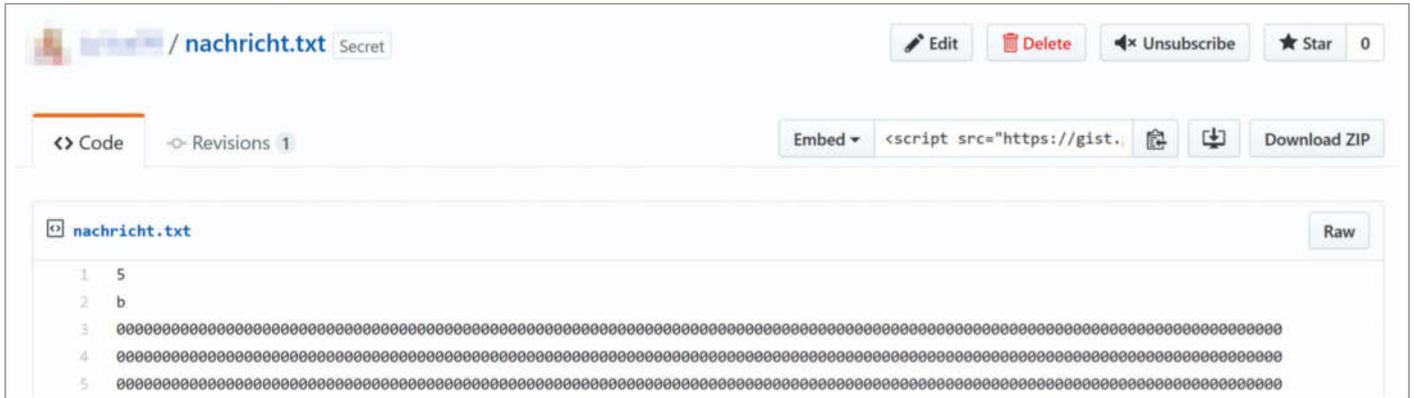
Für die Nutzung von Github Gist wird ein Github-Account benötigt. Wer einen hat, kann sich damit auf gist.github.com einloggen. Nach dem Einloggen erscheint links neben dem Profilbild in der oberen Status-

leiste ein Button mit einem Plus. Durch einen Klick darauf öffnet sich die Oberfläche, um ein neues Dokument anzulegen 6. Als Dateiname kann *nachricht.txt* eingetragen werden und als Inhalt die drei Bestandteile: Identifikationsnummer, Nachrichtentyp und Nachricht. Also könnte dort beispielsweise in den jeweiligen Zeilen stehen: 42, t und Hallo Welt!. Danach wird der Button *Create secret gist* geklickt und schon ist eine Nachricht online, die die Lovebox abrufen könnte.

Code anpassen

Den kompletten Code für die Lovebox gibt es online zum Herunterladen (siehe Link in der Kurzinfo). Für den Einsatz müssen auf jeden Fall drei Anpassungen vorgenommen werden. Diese sind die Angaben für `const char* ssid`, `const char* password` und `const String url` ab Zeile 7 im Listing. Die ersten beiden Änderungen sind der Name und das Passwort des WLANs, wo die Lovebox später stehen soll. Zu Testzwecken sollten natürlich erst mal die Daten des WLANs eingetragen werden, wo gerade die Lovebox entwickelt wird. Die `url` ist der Pfad zum eigenen Gist-Dokument. Um die Adresse zu finden, muss in der Übersicht auf gist.github.com auf die vorbereitete *nachricht.txt* geklickt werden. Dadurch öffnet sich die Seite wie in Abbildung 7 zu sehen. Hier muss man auf den Button *Raw* klicken und dann aus der Suchleiste die URL kopieren und von der URL den Part zwischen `raw/` und `nachricht.txt` entfernen. Somit erhält man die URL zum Gist-Dokument an sich und nicht nur zu einer Version davon. Sie sieht ungefähr so aus: `https://gist.Githubusercontent.com/username/hashcode/raw/nachricht.txt`. Hiervon entfernen wir noch





7 Nach dem Erstellen eines Gist-Dokuments kann dieses nicht nur betrachtet werden. Durch einen Klick auf den Edit-Button kann man den Inhalt verändern, wenn eine neue Nachricht gesendet werden soll. Mit dem Raw-Button öffnet sich das Dokument einzeln und man erhält die URL zum Gist.

den Anfang, sodass dann nur der Part /username/hash-code/raw/nachricht.txt der Variable url zugeordnet wird.

Wenn später die aktuelle Nachricht für die Lovebox geändert werden soll, muss man auf den Button Edit klicken, der auch in Abbildung 7 zu sehen ist. Mit dem Klick auf Update Secret Gist wird die Änderung gespeichert.

Ausblick

Mit der fertigen Lovebox ist die Vorfreude auf Weihnachten doppelt so groß, denn die Liebsten werden sich bestimmt riesig freuen.

Das Unternehmen Lovebox hat noch eine passende App entwickelt, wodurch die Nachricht über die App an die Lovebox gesendet werden kann. Außerdem erscheint in

der App ein Regen aus Herzen, wenn an der Lovebox das Herzchen manuell gedreht wird. So etwas ließe sich ebenfalls selbst programmieren. Mit der read()-Methode vom Servo-Motor könnte herausgefunden werden, ob am Herz manuell gedreht wurde, und diese Info könnte genutzt werden, um ein POST-Request über das ESP8266-Modul zu versenden.

ENTWICKELN, BAUEN, AUSPROBIEREN!



LED-WEIHNACHTSBAUM

- Adressierbare RGB LEDs
- Für Arduino und Micro:bit
- Inkl. Arduino-Board

Art.-Nr. 2195092



3D-DRUCKER STARTER-KIT RF100 V2

- Starter-Kit inkl. Einhausung und Filament PLA
- Fertig montiert und sofort einsatzbereit
- Keine Vorkenntnisse nötig

Art.-Nr. 2183196



MATRIX KIT INKL. AUFBEWAHRUNGSKOFFER

- RGB LED Matrix Modul mit 64x64 Pixel Auflösung
- Die perfekte Erweiterung für Ihren Raspberry PI 4B
- Anzahl LED's: 4096

Art.-Nr. 2158679

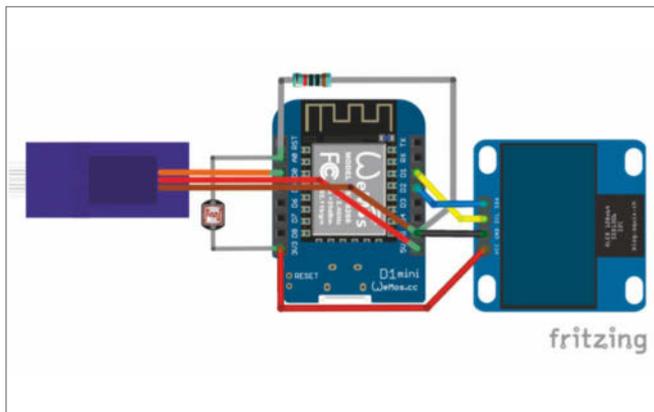


Alle Produkte unter conrad.de/makerfactory

Realisiere Dein Projekt - Start next level conrad.de/maker

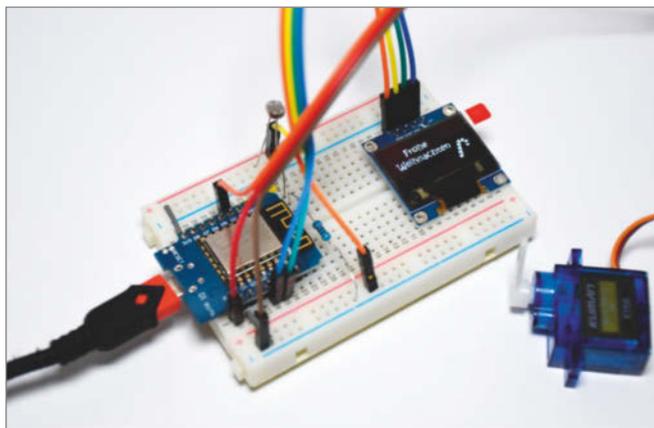
Zusammenbau

Nun bauen wir endlich die Lovebox zusammen. Damit die Elektronik später kompakt in das Gehäuse passt und alle Komponenten stabil an der richtigen Stelle sitzen, werden diese auf Lochrasterplatten verlötet. Doch zuvor sollte alles mit Jumperkabeln auf einem Breadboard gesteckt werden, um den Aufbau zu testen und gegebenenfalls Defekte bei der Hardware festzustellen. Zusammengesteckt ist der Austausch von Teilen deutlich einfacher. Daher folgt das Lötten erst, nachdem der Code erfolgreich getestet wurde.



Aufbau testen

Zuerst verbinden wir die Komponenten mit dem Mikrocontroller. Dafür schließen wir den Helligkeitssensor an den analogen Pin A0 sowie einen 10kOhm-Widerstand und verbinden diese mit GND und 3.3V wie in der Abbildung. Der Servo-Motor besitzt drei Kabel, deren Farben leider nicht standardisiert sind. Im Zweifelsfall lohnt ein Blick ins Datenblatt. Bei diesem Servo wird das orangene Kabel mit dem Signal an den Pin D0 angeschlossen, das braune an GND und das rote Kabel an 5V. Die letzte Komponente bildet das OLED-Display, welches vier Pins hat. Zwei davon sind für die Spannungsversorgung: VCC wird mit dem 5V-Pin des Mikrocontrollers verbunden und GND mit GND. Die Kommunikation zwischen dem Display und dem ESP erfolgt über I²C, welches durch die Verbindungen zu SDA und SCL am Display erfolgt. Dabei wird SDA mit dem Pin D2 und SCL mit dem Pin D1 am Mikrocontroller verbunden.



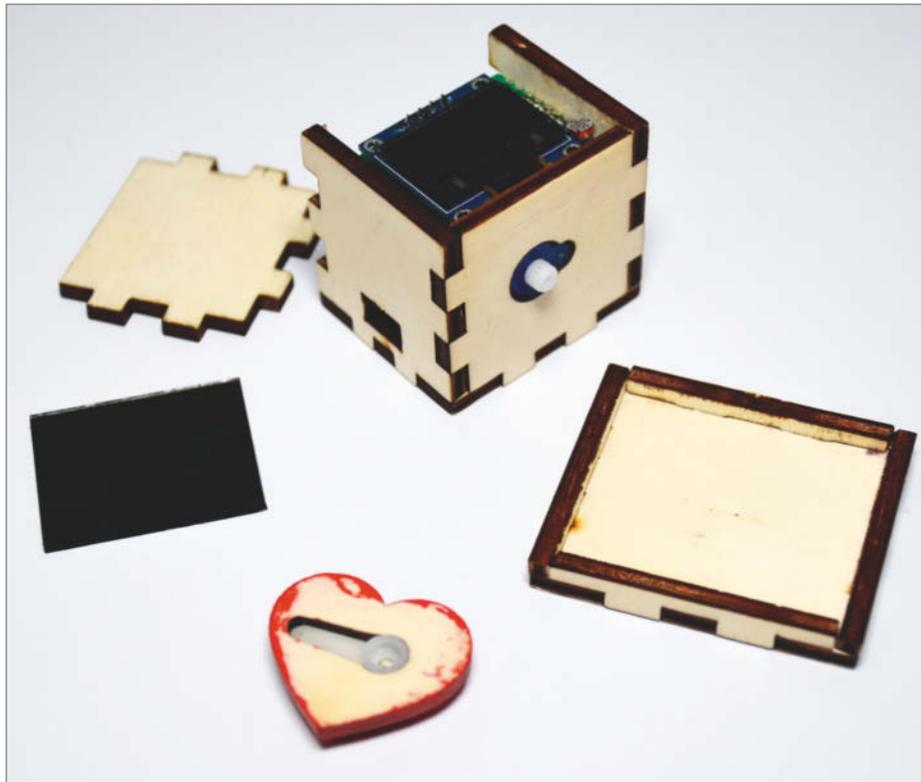
Code testen

Nach dem Aufbau schließen wir den Mikrocontroller mit einem Micro-USB-Kabel an den Rechner an und öffnen die Arduino-IDE. Hier muss das Board *LOLIN(WEMOS) D1 R2 & mini* ausgewählt werden. Außerdem überprüfen wir unter *Werkzeuge/Port* den passenden COM-Port. Nun kann durch ein Klick auf *Hochladen* der Code auf den Mikrocontroller gespielt werden. Nach dem erfolgreichen Übertragen können wir die Lovebox testen und Nachrichten im Gist eintragen. Wenn die Botschaften auf dem Display erscheinen, kann die Box endgültig zusammengebaut werden.



Lötten

Beim Verlötten darf man kreativ sein. Hiervon ist abhängig, wo im Gehäuse die Löcher für die Spannungsversorgung und den Servo-Motor gebohrt werden müssen. Wichtig ist nur, dass der Schaltplan eingehalten wird. Auf einer Lochrasterplatte verlötet wir das OLED-Display mit dem Helligkeitssensor und auf einer zweiten Platine das Mikrocontroller-Board mit den Verkabelungen und dem Widerstand. Mit Hilfe von Pinleisten können die Kabel vom OLED-Display zur unteren Lochrasterplatte am Rand entlang geführt werden, sodass dazwischen genug Platz für den Servo-Motor bleibt. Somit dienen die Pinleisten als Abstandhalter zwischen der Lochrasterplatte mit dem Mikrocontroller-Board am Boden und der Lochrasterplatte mit dem OLED-Display und dem Sensor oben. Je nachdem wie viele Pinleisten vorhanden sind, können diese auch genutzt werden, um das Mikrocontroller-Board unten nicht direkt festzulöteten. Somit kann bei einem Defekt leicht das Board ausgetauscht werden, ohne mühsam alles abzulöten. Sobald alles befestigt ist, sollte geprüft werden, ob noch alles funktioniert. Nun kann man auch den Code mit den WLAN-Einstellungen für das endgültige Zuhause hochladen.



Gehäuse herstellen

Für das Gehäuse müssen die Seitenteile, der Deckel und die Herzen aus dem 3mm-Holz und das Display aus dem 1mm-Acryl mit einem Lasercutter geschnitten werden. Hierbei sollte der Verschnitt passend zum Lasercutter angepasst werden und die Positionen für die Löcher mit der fertig gelöteten Elektronik verglichen werden. Dann kleben wir das kleine Herz auf das größere Herz und malen die Vorderseite rot an. Schließlich kann das weiße Servo-Horn in die Rückseite des Herzens gesteckt werden. Danach wird mit einer Schere ein Stück von der Größe des Displays aus der Blendschutzfolie geschnitten und auf das Acryl-Display geklebt. Der letzte Schritt ist der Zusammenbau des Gehäuses und des Deckels. Nun können wir die Hardware in das Gehäuse einsetzen und das Servo-Horn montieren. —hch

Smarte Gadgets

Sensoren & Aktoren

BBC micro:bit

ePaper-Displays

Calliope mini

Arduino UNO

PORTOFREI AB 15 € BESTELLWERT

shop.heise.de/hardware

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

★ **Spezialitäten**

ATmega 328P-PU	3,60
ATtiny 13A-PU	1,40
ATtiny 85-20PU/-SU	2,20
DS 18B20+	2,30
FlexJumper 10M/M (40x)	4,50
FlexJumper 20F/F (10x)	2,40
LCD2x16-LED blau/gnge	4,50
PT 2399	2,20
PT 4115 B	1,50
RFM 95W-868 (LoRa)	8,90
Solarzelle 5,5V/600mW	4,50
Solarzelle 6V/1000mW	5,60
ST-Link/V2 (LC)	7,40
USB-I2C+SPI DongleKit	33,32
WS2812 (8x) Striplight	2,20
YX 8018	0,50

★ **Module**

18650 BatteryShield V3	4,50
AD 9833 DDS-Modul	7,80
ADC-Modul 16-bit	8,70
Bluetooth-Modul (HC05)	9,80
DHT 22-Einzelement	8,50
DHT 22-Modul	9,50
DRV 8825 MicrostepMod	7,20
ePaper 2.13" HAT sw/rt	19,90
ESP32-Modul	7,50
HC-12 Modul 433MHz	7,50
I2C-Modul für LCD2x16	3,80
LevelConverter 8-Kanal	3,50
Lilon/LiPo-Lademodul	2,80
MicroSD-Card Modul	3,50
Mini-Thermostat 12V	9,90
MOSFET-MiniModul/LL	2,80
MP3 Soundmodul	4,80
Neun-Achsen-Modul	9,50
OLED-Modul 1.3"ws I2C	9,90
PIR-Modul 4-12V(KC7783)	4,80
PWM-Generator 0-150kHz	7,80
RS485-TTL Modul	3,50
RealTimeClock/DS3231	3,90
RFID-Modul RC522	6,50
Sechs-Achsen-Modul	8,90
StepDown-Modul Adj/Fix	2,80
StepDown-Modul USB 3A	6,50
StepUp-Kabel (USB:1.2V)	3,80
StepUp-Modul (Mini)	2,80
StepUp-Modul (100W)	8,90
TMC2130-Modul/SPI	12,90
USB-TTL Modul 3,3/5V	7,80
Voice Recording Modul	4,50

★ **RasPi - Lernkoffer**

RPI Set JOYPI 238,00

Komplettes Set mit 7"- Touch- Farbdisplay und Kamera im Deckel, diversen Anzeigen, Sensoren und Aktoren, fest verbaut in einem schönen Koffer. Ebenfalls im Lieferumfang: 32 GB-Micro-SD-Karte mit passender Linux-Distribution, Funktastatur, IR-Fernbedienung, Servo, RFIDs, Schrittmotor, Netzteil u.v.a.m. Geeignet für den RasPi 2, 3 und 4.

Löt- und brückenfreies Konzept (GPIO-Patching via DIP-Switches). Passender Kurs zum Koffer im Web gratis auf der Hersteller-Homepage, dort auch ausführliche Anleitung mit Codebeispielen und deutsch-/englischsprachiger Schnellstartanleitung.

Weitere Informationen: www.segor.de

★ **Boards**

ESP32-DevKit	13,90
Mini Pro (3,3V)	7,50
Mega 2560 R3	22,50
Mega 2560 R3/CH340	13,90
- Prototype Shield	9,60
Nano V3.0 (CH340)	4,80
- Sockel (ScrewShield)	4,50
NodeMCU-DevKit	10,80
NodeMCU-DevKit CH340	8,90
Raspberry Pi Zero WH	17,90
Raspberry Pi3 Mod A+	29,95
Raspberry Pi4 /1GB	45,00
Raspberry Pi4 /4GB	69,90
- Gehäuse Acryl/trp	5,90
- Display 7" Touch	75,80
Teensy 3.6	39,00
Teensy 4.0	24,90
Tiny85 microUSB Board	4,80
Uno R3	15,90

★ **Sensoren**

Blitzdetektor-Modul	29,50
BME280 Combosensor	7,90
BME680 Combosensor	19,50
CO2-Sensor(MH-Z19B)	28,00
Distanzsensor HC-SR04	5,90
Lichtsensor-Modul 16bit	6,50
Puls/Sauerstoff-Sensor	7,60
Stromsensor-Modul I2C	7,30
Time-of-Flight Modul	6,80
Wägesensor-Modul	4,50

★ **Shields**

DataLoggerShield+RTC	8,90
Ethernet Shield W5100	17,90
LCD KeypadShield	13,50
Motor Shield	16,80
Multi-Function Shield	6,90
ScrewShield	12,50

★ **Feinstaub-Sensor SDS 011**

SDS 011 PM2.5/PM10 29,90

Das Herz des Feinstaubsensoren ist ein integrierter Halbleiterlaser, der die schwebenden Feinstaubpartikel (0,3-10µm) in einer stockdunklen Meßkammer seitlich „anleuchtet“ und so optisch erfassbar macht. Ein kleiner Lüfter zieht während des Meßvorganges Luft von außen in die Kammer.

Eine Aufbauanleitung für eine Meßstation mit weltweiter Netzwerkanbindung finden Sie z.B. unter: luftdaten.info/feinstaubsensor-bauen

Preise in € • Staffelpreise im Online-Katalog • Stand: 18.11.2019 • Änderungen vorbehalten.

elektronische Bauteile

Kaiserin-Augusta-Allee 94 • 10589 Berlin
Tel: (030) 43998-43 • Fax: -55 • www.segor.de

www.segor.de

Umzug in Hannover

Das LeineLab Hannover ist umgezogen und nun in der Paul-Dohrmann-Schule, Burgweg 5 zu Hause. Interessierte können jeweils am Freitagabend zum Offenen Treffen kommen.

leinelab.org

Maker-Termine

Makerfest Luxembourg
16./17. Dezember 2019
Forum Geesseknäppchen
bee-creative.lu/makerfest

36C3 Chaos Communication Congress
27.–30. Dezember 2019
Leipziger Messe
events.ccc.de

Fashion Tech
14.–16. Januar 2020
Kraftwerk Berlin
fashiontech.berlin

Hack im Pott
28. Februar – 1. März 2020
hackimpott.de
Falkenzentrum Süd

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 unseren Kalender ein oder schicken Sie uns eine E-Mail an

mail@make-magazin.de



Call for Makers

Jetzt schon für die Maker Faires im Jahr 2020 anmelden

Für die 2019 ist die Maker-Faire-Saison vorbei: Anfang November lockten die Mini Maker Faires in Salzburg und Sindelfingen noch einmal große und kleine Bastelbegeisterte, die fleißig selber dreheln oder mit einem DIY-Gerät Radioaktivität hinterherspüren konnten. Wer es zu keiner Veranstaltung geschafft hat oder die nächste schon herbeisehnt, kann sich auf 2020 freuen. Die ersten Termine stehen fest und man kann sich bereits beim Call for Makers anmelden

Los geht es am 28. und 29. März 2020. Da starten wir erneut mit einem Doppelpack, denn die Maker Faire Sachsen und die Maker Faire Ruhr teilen sich diesen Termin. In Sachsen geht es zum vierten Mal in der Stadthalle Chemnitz rund, während die Maker Faire in der Dortmunder Arbeitswelt Ausstellung DASA ihren fünften Geburtstag feiert.

Für unsere sechste Maker Faire Berlin gibt es mit dem 18. und 19. April nicht nur einen neuen Termin, sondern auch eine neue Location. Die Arena Berlin ist zentral am Spreeufer gelegen und bietet als ehemaliges Industriegelände einen passenden Hintergrund für kreative Technikprojekte. Ob Vortrag, Workshop oder Stand mit den eigenen Projekten – für jeden Maker ist das passende Format dabei. Das Gleiche gilt auf der Maker Faire Vienna, die am 16. und 17. Mai erneut in der METASTadt Wien gastiert. Last, but not least freuen wir uns auf die Maker Faire Hannover: am 12. und 13. September im bekannten Hannover Congress Centrum. —hch

► maker-faire.de/call-for-makers

Neueröffnung in Bayreuth

Ein Wochenende rund um die Zukunft der Reparatur

Anfang Oktober feierte das Fablab Bayreuth nach seinem Umzug die Neueröffnung. Nach einer Feierstunde mit Gästen aus Wirtschaft und Politik und Projektvorstellungen aus dem Fablab wurden die neuen Räume der breiten Öffentlichkeit vorgestellt. Das Fablab ist nun in der Medicusstraße 3, auf dem Gelände der Schlaeger Kunststofftechnik zu finden – nicht weit entfernt vom alten Standort. Aus den bisherigen Räumlichkeiten in den Wedlich-Hallen musste das Fablab nach sechs Jahren ausziehen, weil deren Eigentümer und Fablab-Sponsor Christian Wedlich Insolvenz angemeldet hatte.

Auch in den neuen Räumen warten eine Reihe an 3D-Druckern, ein Lasercutter, eine Stickmaschine, verschiedene Schneidplotter und eine Transferpresse jetzt auf Besucher. Für alle Geräte werden regelmäßig Einführungsveranstaltungen angeboten, hinzu kommen weitere Workshops. Seit diesem Monat können im Fablab auch Kindergeburts-



Bild: Fablab Bayreuth

tage gefeiert werden. Vom Einstieg in die Programmierung bis zum Basteln mit Plottern, 3D-Druckern und Lasercuttern sind viele kreative Veranstaltungen möglich. Interessierte können jeweils am Mittwoch und Samstag Abend beim OpenLab vorbeischauen. —hch

► fablab-bayreuth.de

Neue Projekte für offene Werkstätten

Gezielte Weiterbildung und ein Open-Source-Zugangssystem

Einmal im Jahr lädt der Verbund der Offenen Werkstätten (VOW) seine Mitglieder zum Netzwerktreffen, um sich auszutauschen und neue Projekte zu starten. Als Tagungsort diente in diesem Jahr das Sandershaus in Kassel, das Werkstatt, Bar, Hostel und Räume für Geflüchtete vereint. Dort wurden gleich mehrere neue Projekte vorgestellt. „Reparieren verbindet“ heißt das Projekt, das an „3D-Druck & Reparatur“ anschließt. In dessen Verlauf waren Möglichkeiten für den Einsatz von 3D-Druck in Repair Cafés untersucht worden. Die Ergebnisse sollen jetzt in sechs Workshops in verschiedenen Werkstätten in der Praxis erprobt und vertieft werden. Außerdem soll das Projekt die Entstehung und Vernetzung von Gruppen fördern, die 3D-Druck in der Reparatur einsetzen.

Künftig wollen die VOW-Mitglieder auch ein eigenes Weiterbildungsprogramm entwickeln, das sich auf die Vermittlung von Didaktik-Kenntnissen konzentrieren soll. Als erster Schritt ist eine Bestandsaufnahme der Anleitungspraxis in Werkstätten und vorhandener Materialien zur Weiterbildung geplant.



Bild: Verbund offener Werkstätten

Schließlich unterstützt der Verbund die Entwicklung eines Open-Source-Zugangssystem. Es soll flexibel auf die Bedürfnisse verschiedener Werkstätten angepasst werden können und die Möglichkeit zur Abrechnung von Maschinennutzung bieten. Wer sich für eine der Initiativen interessiert und mitarbeiten möchte, findet auf den Projektseiten jeweils Kontaktadressen. —hch

► offene-werkstaetten.org/seite/wissenschaftsforschung

Eröffnung in Löbau

Ende November eröffnete die Digitalwerkstatt Löbau-Lebt in der Bahnhofstr. 26 in Löbau ihren neuen Makerspace Geistesblitz.

loebaulebt.de/makerspace

Umzug in Kassel

In Kassel ist die Werkstatt Hammertime umgezogen. Anfang November wurde in der neuen Location am Hauptbahnhof, Franz-Ulrich-Straße 16, gefeiert.

hammertimekassel.de



3D-Kurs für Maker: Konstruieren mit Blender 2.8

Allein mit Text und Screenshots kann man den Umgang mit 3D-Software nur mühsam erklären. Deshalb startet Make jetzt eine Video-Serie zur Arbeit mit dem Open-Source-Paket Blender.

von Peter König

Blender ist eine kostenlose, aber leistungsfähige 3D-Software für Windows, macOS und Linux. Makern gibt sie einen umfangreichen Werkzeugkasten für Eigenkonstruktionen an die Hand, um damit etwa Vorlagen für den 3D-Drucker zu erschaffen oder aus dem Web heruntergeladene Dateien an die eigenen Wünsche anzupassen.

Blender steht im Ruf, Einsteigern hohe Hürden in den Weg zu stellen. Nicht ganz zu Unrecht – jedenfalls bis im August die Version 2.8 mit komplett umgestalteter Bedienoberfläche zum Download freigegeben wurde, die den Einstieg für 3D-Neulinge und den Umstieg für Nutzer anderer Software deutlich erleichtert.

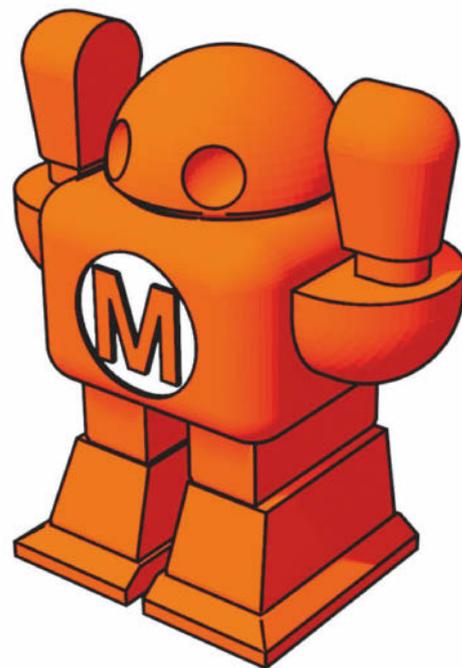
Wer also den Einstieg in die Arbeit mit Blender bisher gescheut oder sogar aufgegeben hat, kann jetzt (noch mal) frisch einsteigen – es lohnt sich. Als Starthilfe hat Make zusammen mit dem Blender-Tutor und Buchautor Carsten Wartmann eine Video-Tutorial-Serie bei Vimeo gestartet (siehe Link am Ende des Artikels). Darin wollen wir anhand verschiedener kleiner, in sich abgeschlossener Maker-Projekte zeigen, wie man

Blender fürs Konstruieren eigener Werkstücke produktiv nutzen kann.

Die Serie besteht derzeit aus einigen Video-Lektionen zwischen 12 und 20 Minuten, die man einzeln für wenig Geld kaufen, jederzeit online anschauen und auch herunterladen kann. Kostenlos gibt es vorab eine ausführliche Übersicht für Umsteiger, in der Carsten Wartmann zeigt, was sich zwischen den Blender-Versionen 2.79 und 2.8 geändert hat – das ist eine ganze Menge.

Speziell für Maker

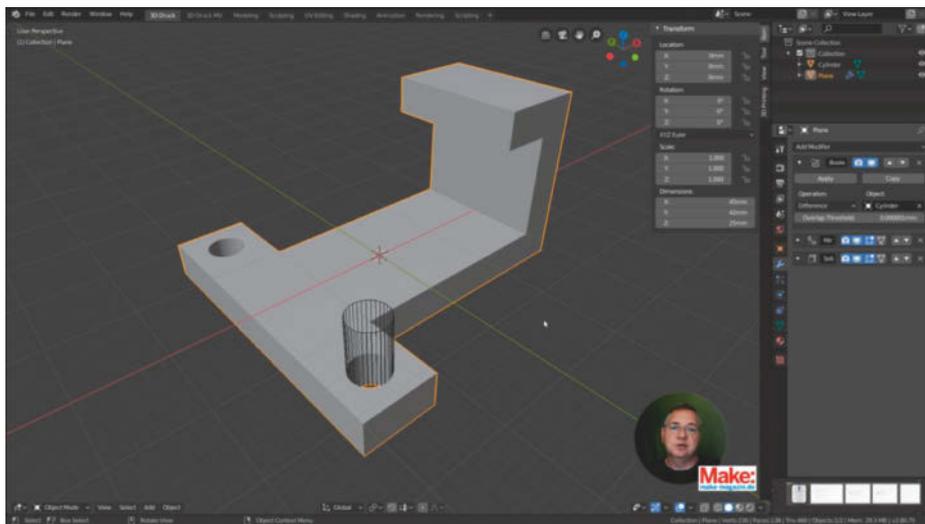
Unsere Tutorial-Serie ist keine umfassende Einführung in alle Aspekte von Blender – sie konzentriert sich ganz bewusst auf die Konstruktion von 3D-Modellen mit dem Ziel, diese später in reale Objekte umzusetzen, etwa auf einem 3D-Drucker oder einer CNC-Fräse. Deshalb bleiben viele Gebiete, auf denen Blender eine Menge zu bieten hätte, aber Makern wenig nützen, konsequent außen vor – etwa Beleuchtung, Oberflächen- und Materialgestaltung, Rendering, Animation, Physik-Simulation und Videoschnitt.



Anders als andere Blender-Kurse führt unsere Serie deshalb auch nicht zu Beginn einmal systematisch und komplett (und stundenlang) durch alle Menüs und Views der Bedienoberfläche. Wir wollen lieber schnell mit konkreten Projekten loslegen und beschränken uns daher pro Folge auf wenige Werkzeuge. Unser Ziel ist es, in Zukunft regelmäßig neue Video-Tutorials nachzuliefern, sodass nach und nach ein immer breiteres Spektrum der Einsatzmöglichkeiten von Blender abgedeckt wird. Dabei nehmen wir über das Forum auf unserer Webseite zur Tutorial-Serie gerne Ihr Feedback und Ihre Fragen, Ihre Wünsche und Anregungen zu weiteren Projekten und Themen entgegen. Auf der Webseite gibt es auch alle Beispieldateien aus den Videos zum Download sowie Links zu weiterführenden Quellen rund um Blender.

Für uns als Make-Redaktion ist ein solches Video-on-Demand-Angebot ein Experiment. Sie können durch Ihr Feedback mit beeinflussen, in welche Richtung die Tutorial-Serie in den zukünftigen Folgen gehen wird. Wir freuen uns daher über Ihre Kommentare und Ihre Anregungen. Und jetzt: Viel Spaß beim Anschauen und beim Bauen mit Blender!

—pek



In Full HD, zum Herunterladen und mit dem Tutor rechts unten in der Ecke – die neuen Blender-Video-Tutorials von Make.

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

ELEKTRONIK / ZUBEHÖR



An alle Technikbegeisterte, Bastler, Selbermacher und Maker:
Jetzt bei Conrad: Viele neue Produktideen der Marke MAKER FACTORY in den Bereichen Elektronik, Robotik, Einplatinencomputer und Entwicklungs-Kits.
Jetzt gleich entdecken unter:

www.conrad.at



Router mit Originalfirmware? Ungeöffnete Geräte, statt Einblicke zu nehmen und hinter die Fassaden zu schauen? Pah.
Wir lieben die Maker-Bewegung und sind mit Feuer und Flamme dabei. Auf Euch warten mehr als 50 000 Artikel aus Elektronik, IT & mehr, besucht uns auf

www.reichelt.de



SEGOR-electronics liefert seit 1978 ein umfangreiches Sortiment an elektronischen Bauteilen. Speziell für Maker, Entwickler und Ingenieure.
Besuchen Sie uns. Entweder online oder direkt vor Ort in unserem Ladengeschäft in Berlin-Charlottenburg, Kaiserin-Augusta-Allee 94

www.segor.de



Frische Brise benötigt?
ICY BOX bietet neben Speicherlösungen und DockingStationen nun auch eine große Auswahl an Schutzgehäusen für Raspberry Pi® 2, 3 und 4 an.

Dank der innovativen Gehäusedesigns stets mit optimaler Kühlung um den Pi® kühl zu halten und zu verhindern, dass es zu temperaturbedingten Leistungsabfällen kommt. Je nach Bedarf mit oder ohne Lüfter, aus hochwertigem Aluminium oder aus Acryl gefertigt.

Ausschnitte für Anschlüsse wie Strom, Video, Audio, USB, LAN, microSD-Karte und GPIO sind selbstverständlich bereits vorhanden, so dass ein einfacher Zugriff gewährleistet ist.

www.raidsonic.de

EDUCATION



Physik Experimente vom einfachen Stromkreis bis zur anspruchsvollen Hochfrequenz für Jung & Alt!
Die Themen der Brick'R'knowledge Sets: Standard Elektronik & Physik, Programmierung, Logik & Flip-Flops, Erneuerbare Energien, Internet of Things, Messtechnik & Home Automation.
Landingpage: www.brickrknowledge.de

BÜCHER



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

LASER



Der einfache Weg zu lasergeschnittenen Teilen. Vom Einzelstück bis zur Serie. Für Tüftler, Maker und Profis.

Bei uns kann sich jeder seine eigenen Teile aus Acrylglas oder Sperrholz schneiden lassen. Einfach eine Anfrage mit Datei an info@plexilaser.de schicken oder das Online-Formular benutzen.

www.plexilaser.de

3D-DRUCK



Filament aus Joghurtbechern!

Nachhaltig – Sozial

PLA-Filament aus 100% recyceltem Joghurtbechermaterial. Mit Ökostrom in Berlin zu Filament verarbeitet und von den Mosaikwerkstätten für Behinderte umgespult, verpackt und versandt.

www.3dk.berlin

Make:markt

Der **Make:markt**. Nur 150,00 Euro je Ausgabe für eine Basisanzeige.

Weitere Informationen erhalten Sie unter:

jeah@maker-media.de



Pepakura

Die Nase voll vom 3D-Druck? Für diese 3D-Modelle reicht ein 2D-Drucker. Wir geben einen Einblick in die Pepakura-Szene und die Software, die aus 3D-Objekten Schnittmuster aufs Papier zaubert.

von Rebecca Husemann

Bild: Tom Forsyth, Owen Giffers/leeve

Pepakura bezeichnet 3D-Modelle, die mit einfachen Werkzeugen aus Papier gebaut werden und für deren Konstruktion die Pepakura-Software zum Einsatz kommt. Die Technik wird gerade in der Cosplay-Szene zunehmend beliebter, da das Arbeiten mit Karton und Papier besonders für Einsteiger empfehlenswert ist. Selbst wenn man keinen Zugriff auf spezielle Werkzeuge und Maschinen hat, kann man mit Pepakura kostengünstig Rüstungen, Waffen und andere Kostüm-Elemente bauen. Alles, was man braucht, ist viel Geduld, Papier, ein Drucker, Kleber und eine Schere ①.

Pepakura? Heißt das nicht Papercraft?

Pepakura und Papercraft sind zwei Begriffe, die oft synonym verwendet werden. Richtig ist das nicht: In Wirklichkeit ist Papercraft der Sammelbegriff für alle Projekte, die als Basismaterial Papier oder Pappe verwenden. Die meisten Papercraft-Disziplinen haben gemeinsam, dass man sie in kurzer Zeit lernen kann. Zu ihnen zählt beispielsweise das einfache Basteln von Grußkarten, Origami, Scrapbooking, Scherenschnitte und sogar das Schöpfen von Papier. Eine weitere Disziplin der Papercrafts ist der Modellbau. Beim Papier-Modellbau erschafft man dreidimensionale Formen aus zweidimensionalen Bauteilen. Durch Falzen und Verkleben fügt man sie zu einem dreidimensionalen Modell zusammen. Unterstützt wird der Bau durch eine Anleitung oder Explosionszeichnungen. An dieser Stelle kommt der Begriff Pepakura ins Spiel: Pepakura ist eine Form des Papier-Modellbaus.

Wobei, halt – eigentlich ist Pepakura nur der Name des Computerprogramms, mit dem man dreidimensionale Formen in ein zweidimensionales Schnittmuster verwandeln kann. Mit der Zeit hat Pepakura sich als Dachbegriff für diese Auffalttechnik durchgesetzt. Heutzutage meint Pepakura alle Modellbau-Schnittmuster für polygonale dreidimensionale Skulpturen, unabhängig davon, wie sie entstanden sind.

Modellbaubögen, wie die Software Pepakura sie berechnet, gibt es schon seit dem 16. Jahrhundert zu kaufen. Damals mussten noch viele kluge Köpfe über die perfekte sogenannte *Abwicklung* grübeln. (Das ist immer noch eine beliebte Hausaufgabe für arglose Designstudierende.) Heutzutage können intelligente Programme diese Aufgaben präziser und schneller lösen. Bei den Motiven der Karton-Modellbaubögen handelte es sich bis in die Neunziger primär um historische Autos und Gebäude, ferner auch Schiffe und Flugzeuge. Die Zielgruppe waren eher verschrobene Bastler und Schulkinder. Seit es Farbdrucker in den meisten Haushal-

Kurzinfo

- »Einstieg in die Pepakura-Szene
- »Papiermodelle schneiden und kleben
- »Materialkunde

Alles zum Artikel
im Web unter
make-magazin.de/x692

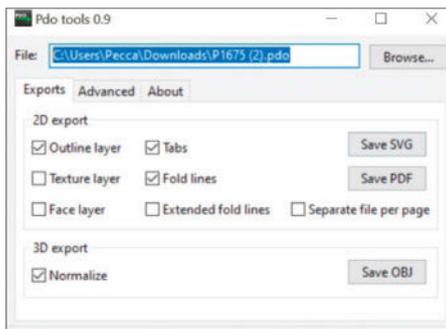


① Die nötigen Materialien haben die meisten Menschen bereits zuhause.



② Dieses Chamäleon hat Erschaffer *Paperwolf* so entworfen, dass es mit wenigen Polygonen möglichst lebendig wirkt.

Bild: Paperwolf, Wolfram Kampffmeyer



3 Mit der Anwendung *PDO tools* kann man aus Pepakuradateien Vektorgrafiken exportieren.

ten gibt und jeder einfach selbst Druckbögen herstellen kann, stammen die Motive aus der Populärkultur. Besonders beliebte Druckobjekte sind Figuren aus Comics und Videospiele. Am Anfang handelte es sich noch um einfache geometrische Formen, von interessierten Laien mit kostenfreien CAD-Programmen in mühsamer Kleinarbeit generiert. Mittlerweile ist das Internet voll von detailgetreuen und äußerst komplexen Figurinen und Superhelden-Rüstungen. Mit der Pepakura-Software und dem Einzug von Schneidplottern in die heimische Werkstatt entwickelte sich die Szene weiter. Sie zieht jetzt auch Menschen an, die gar nicht unbedingt selbst basteln wollen. Es macht ihnen viel mehr Spaß, Pepakuravorlagen für andere zu entwickeln. Besonders attraktiv macht die meisten Pepakura-Modelle ihre Low-Poly-Optik. Je gezielter die Menge an Polygonen und ihre Platzierung bei einem Modell ge-

wählt wird, desto ästhetischer wird das Ergebnis. Das nehmen viele Künstler als Herausforderung 2.

So funktioniert das Programm Pepakura

Pepakura ist ein Programm der japanischen Firma Tama Software. Die Software ist immer noch aktuell, auch wenn der Retro-Look der Webseite das nicht vermuten lässt. Den Pepakura Designer 4 kann man in der Version 4.1.6 installieren. Die Software läuft nativ unter Windows, auf Linux in Wine – und unter macOS gar nicht. Es gibt das Programm in zwei Ausführungen: Die kostenfreie Version besteht aus dem *Pepakura Viewer* und dem *Pepakura Designer* mit eingeschränkten Funktionen. Mit der kostenpflichtigen Version kann man sämtliche Funktionen des *Pepakura Designers* nutzen. Der Produkt-Key kostet 38 US-Dollar. Für die meisten Pepakura-Einsteiger reicht die kostenfreie Version. Sie ermöglicht es, Pepakuras proprietäres Format *.pdo* zu öffnen und sich fertige Schnittmuster anzusehen. Aus dem Pepakura-Viewer kann man die Dateien direkt ausdrucken. Da endet allerdings schon die Auswahl an kostenfreien Funktionen. Auch wenn das zunächst sehr einschränkend klingt, reicht es für die meisten User. Online findet man Pepakura-Dateien für unzählige 3D-Modelle, mit denen man ohne weitere Modifikationen losschnippeln kann. Wenn man ein passendes Motiv gefunden hat, muss man es nur im Pepakura Viewer öffnen und auf *Drucken* klicken. Schon hält man den Modellbaubogen in den Händen.

Spannender wird es, wenn man eine eigene Abwicklung von einem 3D-Modell erstellen oder eine bestehende Pepakura-Datei verändern möchte. Die kostenpflichtige Variante von Pepakura bringt die notwendigen Features mit. Sie ermöglicht es beispielsweise, 3D-Modelle im *.obj*- und *.stl*-Format zu importieren. Viele Videospiele-Schmieden stellen ihre 3D-Modelle online zur Verfügung und auch Objekte von Webseiten wie thingiverse.com bieten sich für den Import an. Importiert man eine 3D-Datei, führt das Programm automatisch die Abwicklung durch. Allerdings wird die Abwicklung nur so gut wie das vorhandene 3D-Modell. Daher ist dieses Feature nur für diejenigen interessant, die Übung in der Bearbeitung von 3D-Modellen mit Programmen wie Blender haben. In der kostenpflichtigen Version kann man dazu sein Ergebnis auch speichern und exportieren – dies ist im kostenfreien Designer nicht möglich.

Auch für alle, die ihr Schnittmuster von einem Schneidplotter schneiden lassen wollen, ist die Pro-Version sinnvoll. Nur mit dieser Edition kann man PDFs, EPS, DXF und SVGs exportieren, also echte Vektordateien. Der kostenfreie Pepakura-Viewer tut zwar so, als könnte er PDFs exportieren, in Wirklichkeit handelt es sich aber nur um schlecht verkleidete PNGs – komplett mit Trenchcoat und falschem Schnurrbart. Diese PNGs sind tatsächlich von so schlechter Qualität, dass man sie nicht mal händisch in Vektorgrafiken umwandeln möchte. Leider überzeugt der SVG-Export auch nur bedingt: Manchmal fehlen an einigen Kanten die Schnittlinien. Dazu bestehen ge-

2 Das Walross ist so auf seine markantesten Formen reduziert, dass sie von den Polygonen nur noch angedeutet werden – trotzdem erkennt man es sofort.

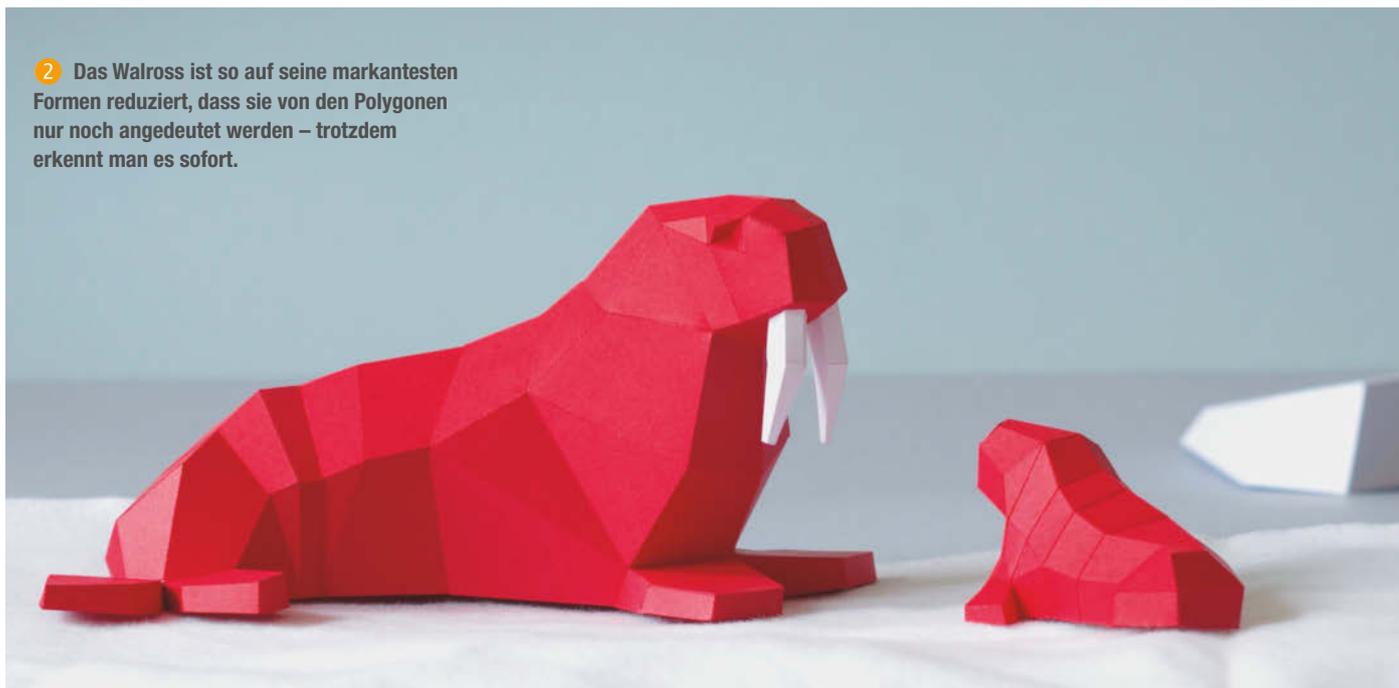


Bild: Papenwolf, Wolfram Kampfmeier

strichelte Linien aus vielen einzelnen Elementen, statt einer einzigen Vektorlinie.

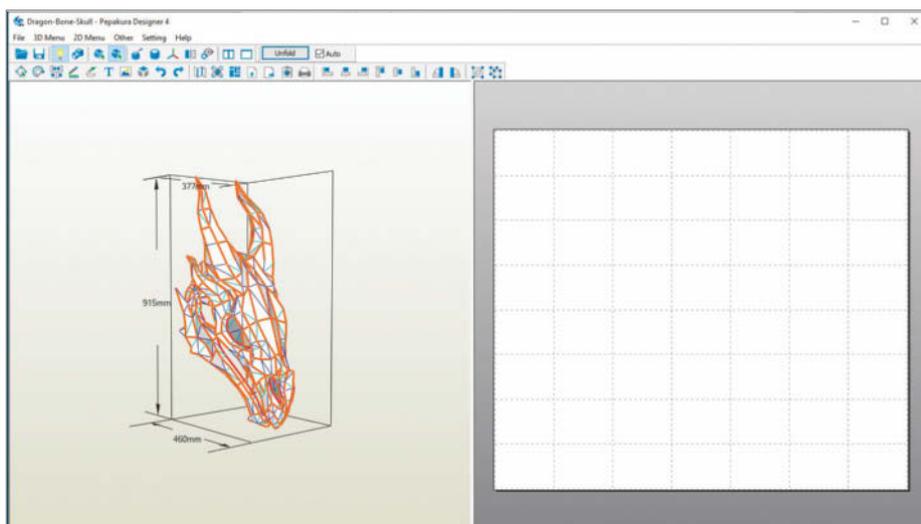
Das Internet bietet für diese Hürde eine Lösung: Dávid Pethes hat ein kostenloses Tool gebaut, das .pdo-Dateien in Vektorgrafiken umwandelt. Die Anwendung heißt schlicht *PDO tools*. Sie bringt verschiedene Optionen für den Export mit. Man kann beispielsweise Farbebenen ausblenden und die Falzlinien an- und ausschalten **3**.

Wie eine Schnittdatei entsteht

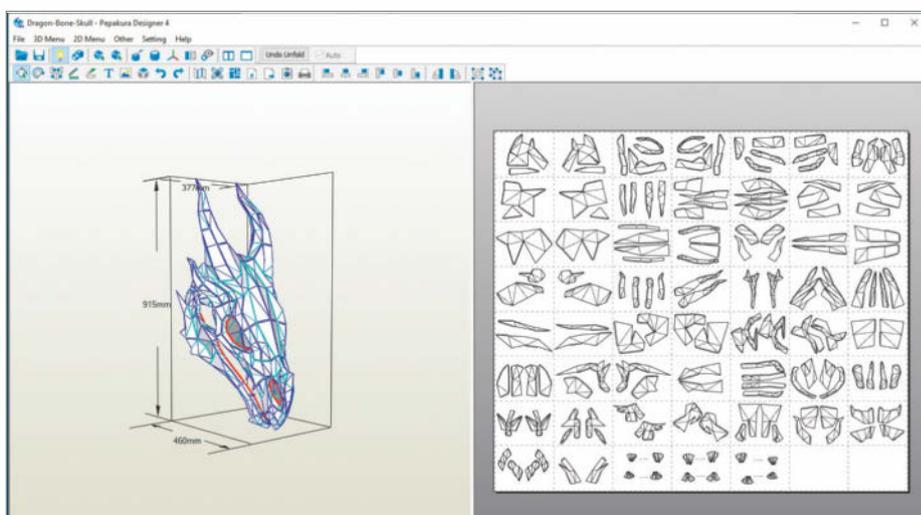
Jetzt werfen wir einen Blick in den Pepakura Designer. Der Screen ist zweigeteilt: Im linken Fenster sieht man die 3D-Ansicht, auf der rechten Seite einen 2D-Arbeitsbereich in Form eines DIN-A4-Bogens. In den Einstellungen definiert man das Papierformat. Die Pepakura-Software nimmt ein 3D-Modell, zerteilt es in handliche Stücke und wickelt es ab. Dann verteilt sie die einzelnen Elemente automatisch auf mehrere Seiten, damit sie optimal druckbar sind. Das Programm fügt Klebelaschen und Berg- und Talfalten hinzu. Dazu nummeriert es die einzelnen Teile und Kanten durch. Wenn man sich in den Einstellungen umschaut, sollte man direkt die Funktion *Print Page Number* aktivieren. Schnittlinien und Falzlinien sind durch die Linienart gut zu unterscheiden: Durchgezogene Linien muss man schneiden, gestrichelte Linien werden gefaltet. Die gestrichelten Linien teilen sich in Berg- und Talfalten, dazu später mehr.

Öffnet man ein 3D-Objekt in Pepakura, um es in eine Abwicklung umzuwandeln, muss man zunächst die Ausrichtung des Modells im 3D-Raum kontrollieren **4**. Nach dem Import sieht man das 3D-Objekt links im 3D-Fenster. An dieser Stelle lohnt es sich, vor der Abwicklung schon mal Hand anzulegen: Mit der *open edge*-Funktion kann man im *Edit Mode* die Kanten auswählen, an denen auf jeden Fall ein Schnitt sitzen soll – und damit Stellen als zusammenhängend markieren, die besonders prominent sind. (Ein Schnitt sollte zum Beispiel aus ästhetischen Gründen nicht unbedingt durch das Auge einer Figur führen.) Besonders geeignet für zusätzliche Schnittlinien sind Symmetrieachsen. Dann kommt der spannende Teil. Mit einem Klick auf *unfold* beginnt der Aufklappprozess. Pepakura generiert aus der 3D-Grafik eine Vektorgrafik. Stellt man die Entfaltung auf *auto* ein, arrangiert Pepakura die Schnittstücke von alleine. Dann passt jedes Element auf eine DIN-A4-Seite, ohne Anschnitte. Schon erscheint auf der Arbeitsfläche das 2D-Schnittmuster **5**. Obacht: Die Größe des 3D-Modells kann riesig sein, bitte immer überprüfen und anpassen.

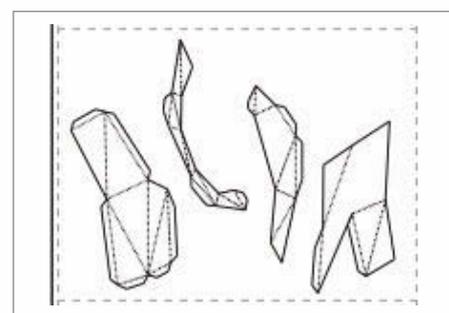
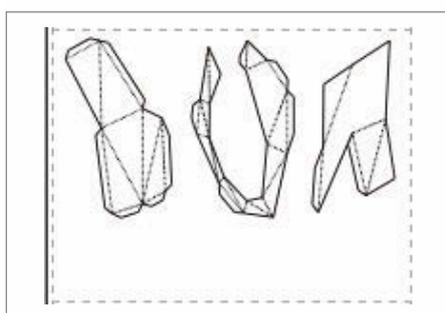
Die entstandene Abwicklung sollte man genau unter die Lupe nehmen. An dieser Stel-



4 Das 3D-Modell kann man nach dem Import kontrollieren und prüfen, ob die Größe stimmt.



5 Pepakura zerteilt das 3D-Modell automatisch und verteilt den Schnittbogen auf DIN-A4-Seiten.



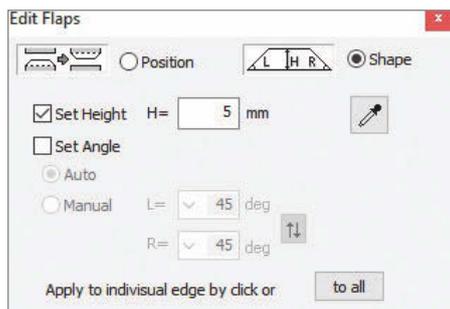
6 So sieht ein Teilstück vor dem Zerschneiden aus ...

... und so hinterher

le entscheidet sich, wie mühsam der Falt- und Klebeprozess wird. Wenn man viele Überlappungen und kleine Einkerbungen sieht, wird es vermutlich recht komplex. Diese Art von Fehlern kann man in Pepakura manuell korrigieren. Sieht die 2D-Abwicklung insgesamt sehr kleinteilig und kompliziert aus, kann es daran liegen, dass das 3D-

Modell zu viele polygonale Flächen hat. Das kann man nicht in Pepakura ändern. Hier bleibt nur die Möglichkeit, in einem 3D-Modellierungs-Programm wie Blender die Flächen zu vereinen und das Modell so zu vereinfachen.

Viele kleine Problemstellen kann man vor dem Druck korrigieren. Mit einem Linksklick



7 Bei besonders kleinen und sehr großen Modellen lohnt es sich, die Größe der Klebelaschen manuell anzupassen.

auf ein einzelnes Modellelement im 2D-Viewer hebt man diese Stelle im 3D-Viewer farblich hervor. So kann man erkennen, welcher Abschnitt hinterher welchen Teil des Modells bildet. Mit Drag&Drop kann man die einzelnen Abschnitte des Schnittmusters auf dem Druckbogen verschieben. Das ist praktisch, wenn man ein komplexes Modell hat, bei dem benachbarte Teile auf einem Bogen stehen sollen. Ist ein einzelnes Element zu groß für ein DIN-A4-Blatt, kann man sie weiter zerteilen. Dazu verwendet man das *Divide/Connect Faces*-Werkzeug **6**. Dieses Werkzeug sollte man auch benutzen, wenn ein Element so verschachtelt ist, dass sich zum Beispiel die Klebelaschen überschneiden. Anschließend kann man die Größe der Klebeflächen justieren. Bei sehr kleinen Modellen legt Pepakura die Klebeflächen automatisch sehr klein an. Das kann beim Kleben Schwierigkeiten bereiten. Daher sollte man für sich individuell herausfinden, welche Klebelaschen-Breite zum Modell und der eigenen Fingerfertigkeit passt. Um die Klebelaschen manuell zu verändern, kann man im 2D-Fenster einen Rechts-

klick machen und die Funktion *edit flaps* auswählen **7**. Jede problematische Stelle, die man zu diesem Zeitpunkt repariert, spart beim Zusammenbau viel Zeit und Nerven. Falls man das Schnittmuster mit einem Laser-cutter zuschneiden möchte, sollte man die Kantenummerierung ausschalten. Sonst schneidet der Cutter die Zahlen mit aus.

Einige Pepakura-Modelle bringen eine farbige Textur mit. Möchte man diese nicht mitdrucken, kann man sie in den Einstellungen ausblenden.

Woher bekomme ich Pepakura-Dateien?

Die offizielle Pepakura-Webseite bietet viele Dateien zum Download an. Auch eine einfache Google-Suche mit Schlagwörtern wie „Pepakura Cat“ fördert erstaunlich viele Ergebnisse zutage. Bei dieser Suche landet man meistens auf Pinterest – und das ist tatsächlich eine hervorragende Quelle, auf der man mehrere Tage mit Stöbern verbringen kann. Auf der Plattform Etsy bieten dagegen viele Gestalter Pepakura-Dateien zum Kauf an. Diese haben den Vorteil, dass die Ersteller sehr darauf achten, dass die Schnittmuster leicht zu bauen sind. Dazu gibt es auf Etsy viele vorge-schnittene Papiersets für den schnellen Einstieg. Wer sehr spezielle Fragen zu Pepakura hat, kann sich an das Pepakura-Subreddit auf Reddit wenden. Das ist zwar nicht mehr besonders aktiv, hat aber schon viele interessante Fragen zum Thema beantwortet. Diverse Tutorials findet man auf YouTube, auch zu sehr technischen Problemstellungen.

Materialien

Auch wenn Pepakura für den schnellen Druck mit Farbdruckern konzipiert wurde,

sollte man sich nicht täuschen lassen: Normales Druckerpapier ist für Pepakura nicht geeignet. Papiere und Karton sollten eine Grammatur zwischen 160g/m² und 200g/m² vorweisen. Je größer das Modell werden soll, desto stabiler muss das Material sein. Wählt man eine zu leichte Grammatur, könnte das Modell einfach kollabieren. Bei sehr großen Objekten, wie zum Beispiel einer lebensgroßen Iron-Man-Rüstung, kann eine innere Stützstruktur nötig sein. Die bedruckten Pepakura-Bögen sollte man so ordentlich wie möglich bereitstellen. Am besten legt man sie in der richtigen Reihenfolge auf einen Stapel am Arbeitsplatz.

Pepakura-Bogen von Hand schneiden

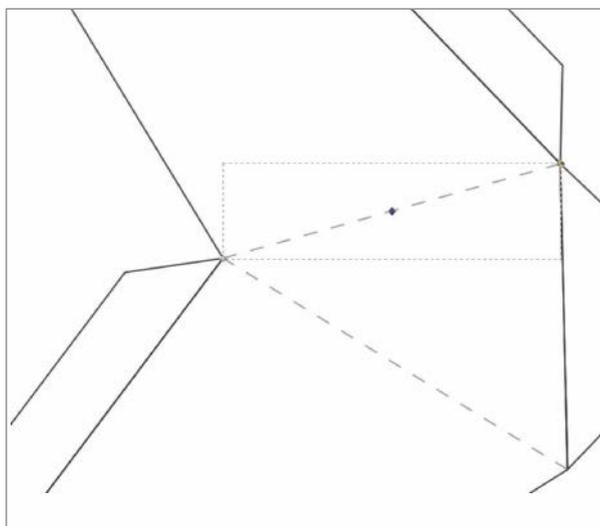
Viele finden es meditativ, ihre Pepakura-Bögen von Hand auszuschneiden. Plappert dazu noch ein nettes Hörbuch, arbeitet man in einen hochproduktiven Flow. Andere kriegen bei der kleinteiligen Geduldsprobe nach wenigen krummen Schnitten einen Schreianfall und beschließen, das Hobby umgehend wieder aufzugeben. Dieser Abschnitt richtet sich an den ersten Typus.

Traditionell wird Pepakura von Hand geschnitten, mit Cutter, Skalpell und Schere **8**. Das liegt in den Ursprüngen der Technik: Pepakura wollte 3D-Modelle für alle die zugänglich machen, die sich keine teuren Modellbaubögen und Spezialgeräte leisten können. Für besonders exakte Ecken und Aussparungen sollte man zum Schneiden ein Skalpell verwenden. Bei runden Formen sollte man dagegen zur Schere greifen. Am wichtigsten ist, dass das Werkzeug scharf ist. Für Skalpell und Cutter liegen im Bestfall zu jedem Zeitpunkt Ersatzklingen bereit. Pepakura-Modelle sind sehr präzise geometrische



Bilder: Zim & Zou

8 Designer wie Zim & Zou schneiden selbst solch kleinteilige Modelle komplett von Hand zu.



9 Diese gestrichelte Linie könnte ein Schneidplotter so noch nicht schneiden. Pepakuras Vektorgrafiken muss man zwar immer auf Richtigkeit kontrollieren, dafür ist man bei den meisten Schneidplottern nicht an das DIN-A4-Format gebunden.

Formen, bei denen viele Seiten ineinandergreifen müssen – da ist nichts ärgerlicher, als ausgefranste, unsaubere Kanten. Besser als eine Schere sind übrigens zwei Scheren: Eine mit langen Scherenklingen und eine kurze für Feinarbeiten.

Die beste Vorgehensweise ist, zuerst benachbarte Elemente auszuschneiden und diese direkt zusammenzukleben. Man sollte vermeiden, alle Teile auf einmal auszuschneiden. Das geht nur dann gut, wenn man sehr diszipliniert ist und genug Platz hat, um alle in der richtigen Reihenfolge auszubreiten. Wir empfehlen dazu auch noch eine große, selbstheilende Schneidmatte als Unterlage. (Schneidmatten vertragen übrigens keine Hitze – wer also plant, sein Pepakuramodell mit Heißkleber zu verbinden, sollte sie vorher in Sicherheit bringen.)

Schneiden mit Schneidplottern

Hier kommt die Lösung für die Ungeduldigen: Maschinelle Fertigung. Mit einem Schneidplotter wird einem die schlimmste Arbeit abgenommen. Bei dickeren Materialien bietet sich zusätzlich auch ein Lasercutter an.

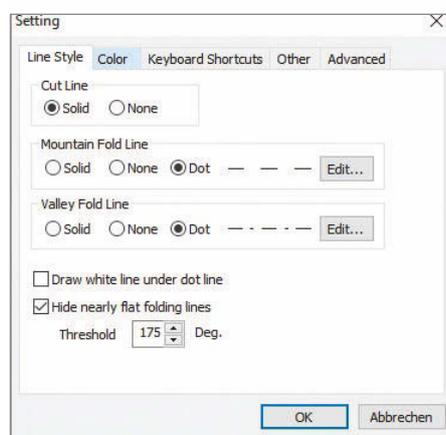
Wer einen Silhouette-CAMEO-Schneidplotter hat, der kann für 15 US-Dollar eine Zusatzsoftware erwerben, die Druckdateien und eine dazugehörige Vektorgrafik erzeugt. Das ist in der Theorie der schnellste Weg, um die Dateien zu drucken und automatisiert zuschneiden zu lassen.

Für alle anderen Geräte ist der einfache Export als SVG oder EPS die richtige Wahl. Mit Vektorprogrammen wie Inkscape und Illustrator kann man die Dateien öffnen und überprüfen, ob alle Elemente richtig exportiert wurden 9. Zusätzlich muss man die gestrichelten Linien noch in echte Pfade umwandeln. Wenn man die Datei schon geöffnet

hat, bietet es sich an, in jedes Element Stege einzubauen. Bei modernen Schneidplottern mit geradem Schneidbett ist das nicht nötig, aber viele industrielle Schneidplotter bewegen das Papier auf einer gebogenen Unterlage. In diesem Fall könnten die ausgeschnittenen Abschnitte während des Schneidvorgangs aus dem Bogen fallen. Ergänzend ist auch eine leicht klebende Unterlage für das Papier eine gute Anschaffung. Farbige Motive mit Texturen kann man bei älteren Schneidplottern leider auch vergessen. Der Schneidplotter müsste die gedruckte Form präzise umfahren – dafür kann man die meisten Geräte nicht genau genug ansteuern.

Falten

Mit Pepakura erstellte Druckbögen haben zwei Arten von Falzlinien: Gleichmäßig gestrichelte Linien sind Bergfalten. Faltet man sie, befindet sich die Linie an der Außenkante des Knicks. Talfalten sind im Wechsel kurz



10 Bei Pepakura sind Berg- und Talfalten durch unterschiedlich gestrichelte Linien gekennzeichnet.

ALLNET MAKER & MINT WELT

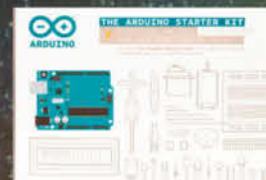
Makeblock

Programmierbare Roboter



ARDUINO PRODUKTE

Baukästen Informatik & Physik



www.brickrknowledge.de

Brick'R'knowledge
Experimentiersystem
Elektronik & Physik

NEU!

Bio Feedback Set

Von Herzschlag bis Mind Control



MatataLab

Coding ab 4 Jahren



3Dsimo

4-in-1 3D Stift



www.maker-store.de

www.allemachenmint.de



11 Bei besonders kleinen Klebelaschen ist Feingefühl angesagt – da dürfen Hilfsmittel wie Pinzetten und Holz- oder Wattestäbchen zum Einsatz kommen.

und lang gestrichelt. Beim Falten verschwindet die Linie im Inneren des Knicks 10. Das Falten ist der Prozess, bei dem man am meisten Fehler machen kann. Man sollte sehr genau darauf achten, alle Falzlinien in die richtige Richtung zu knicken, sonst passen zum Schluss die einzelnen Elemente eventuell gar nicht zusammen. Hat man das Motiv mit einem Schneidplotter zugeschnitten, findet man auf dem Bastelbogen vorperforierte Falzlinien. In diesem Fall ist das Falten ein Kinderspiel. Wer das Modell hingegen ausgedruckt und von Hand geschnitten hat, der macht sich jetzt ans Falzen. Das geht am besten ganz klassisch mit Falzbein und Stahllineal. Die Falzung mit dem Falzbein sorgt dafür, dass das Papier an der Knickstelle nicht aufbricht. Dazu legt man einfach das Stahllineal fest aufs Papier und zieht mit dem spitzen Ende des Falzbeins eine Rille.

Software-Alternativen

Wer kein Geld für die Software ausgeben kann oder will oder keinen Windows-Rechner besitzt, kann beispielsweise zu **Slicer for Fusion 360** von Autodesk greifen. Die Software lässt sich kostenlos herunterladen und läuft unter Windows und macOS. Sie bereitet importierte STL- und OBJ-Dateien für unterschiedliche Fertigungsverfahren auf – unter anderem als Oberflächenabwicklung – und exportiert 2D-Schnittvorlagen als Vektorgrafik, er-

fordert aber etwas Einarbeitung. Wer ohnehin gerne mit **Blender** arbeitet, kann diesem Open-Source-3D-Paket über *Edit/Preferences* und die Schaltfläche *Add-ons* die Erweiterung *Import-Export: Export Paper Model* hinzufügen und anschließend Abwicklungen von 3D-Objekten samt Klebelaschen über das Menü *File/Export* als PDF speichern. Allerdings ist die beschriebene Aufbereitung von Modellen mit Pepakura deutlich komfortabler.

Kleben

Beim Kleben scheiden sich die Geister: Einige schwören auf Sekundenkleber, andere auf einfachen Bastelkleber (wie die Flinke Flasche von Uhu), die nächste Gruppe liebt schnelltrocknenden Holzleim und ein paar Abenteuerlustige wagen sich an den Heißkleber. Wie immer liegt die Wahrheit irgendwo dazwischen: Den Kleber wählt man am besten nach dem Material und der weiteren Verarbeitung. Bei kleinen Teilen aus leichtem Material mit winzigen Klebeflächen ist Sekundenkleber die beste Wahl. Für größere Modelle aus festem Karton sind Holzleim und Bastelkleber gleichermaßen geeignet. Hauptsache, sie sind schnelltrocknend. Für besonders dicken Karton oder EVA-Foam ist Heißkleber optimal – er trocknet schnell und bildet bei großen Klebeflächen zuverlässig eine feste Verbindung.

Bei der Klebermenge gilt: So viel wie nötig, so wenig wie möglich. Das Material Papier verzeiht es durchaus, wenn man eine Klebestelle wieder löst, um mehr Kleber hinzuzufügen. Dagegen hinterlässt überflüssiger Kleber unschöne Verfärbungen, wenn er aus der Klebestelle quillt. Für besonders kleine Klebeflächen kann man den Kleber vorsichtig mit einem Schaschlikspieß oder einem kleinen Spatel auftragen. Dabei bietet es sich an, das Objekt mit einer Pinzette zu halten 11. Ein Vorteil von Pepakura: Beim Kleben muss man sich nur wenig Gedanken über die Winkel und die Geometrie machen. Normalerweise passen die Klebeflächen nur auf eine einzige Weise zusammen, ohne Falten zu werfen.

Unabhängig davon, welchen Kleber man benutzt: Wichtig ist, dass man in gut gelüfteten Räumen arbeitet.



2 Mit Pepakura kann man sich den Traum vom Erdferkel als Haustier erfüllen.

Bild: Papervolf, Wolfram Kampfmeyer

Härten

Viele Pepakura-Modelle sind Dekorationsobjekte, die auf dem Schreibtisch stehen oder kahle Wände verschönern. Andere werden hingegen zu ambitionierten Rüstungen von noch ambitionierteren Cosplayern. In diesem Fall ist einfache Pappe nicht stabil genug. Um die Pappe zu härten, kann man sie mit Gießharz behandeln. Das Harz wird zu diesem Zweck dünn mit einem Pinsel auf der Außenseite des Modells verstrichen. Das sollte man am besten an der frischen Luft machen, um keine giftigen Dämpfe einzuatmen. Bei der Verarbeitung sollte man eine Atemmaske und Handschuhe tragen. Die Schutzschicht aufzubringen kann etwas trickreich sein: Sobald die Pappe feucht wird, verliert sie an Stabilität. Es dauert einige Minuten, bis das Harz fest wird. In dieser Zeit muss man das Modell möglichst gut abstützen.

Die Behandlung des Papiers mit Gießharz bietet ganz neue Möglichkeiten zur Weiterverarbeitung. Man kann Unebenheiten mit Spritzspachtel und Filler ausgleichen, die Oberfläche schleifen, polieren und abschließend lackieren. So bearbeitete Pepakura-Mo-

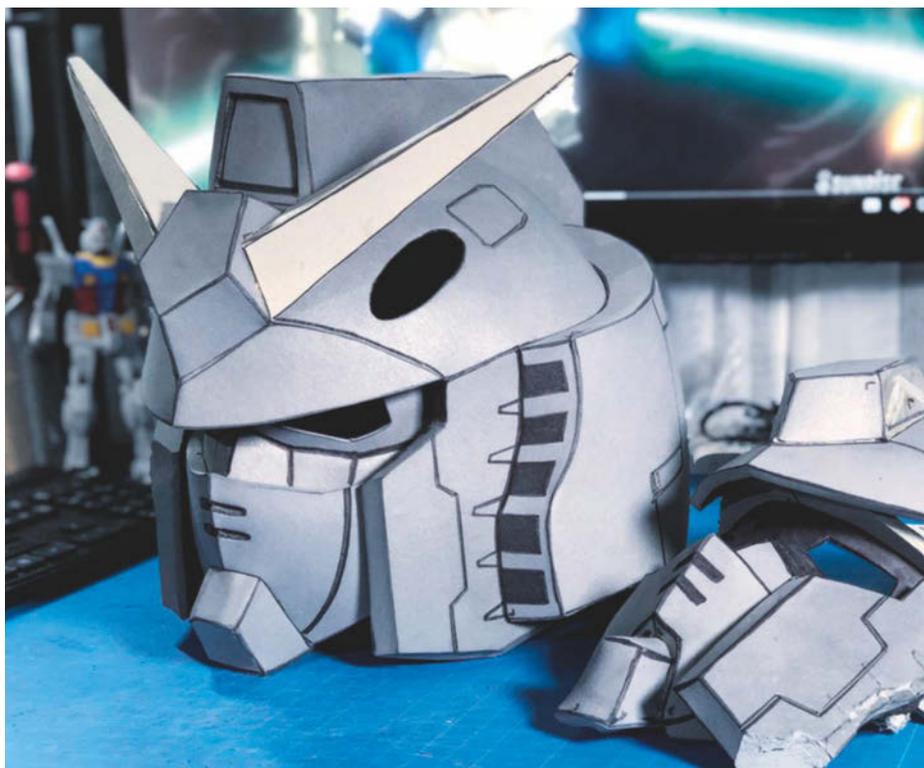


Bild: Heroes Workshop

12 Dieser beeindruckende Gundam-Helm wurde vom Heroes Workshop mit EVA-Foam umgesetzt.

Für LEGO®-Freunde



P. »Sariel« Kmieć

Das »inoffizielle« LEGO®-Technic-Buch

Kreative Bautechniken
für realistische Modelle

2. Auflage

2017, 428 Seiten

€ 29,90 (D)

ISBN 978-3-86490-433-2



A. Ehle

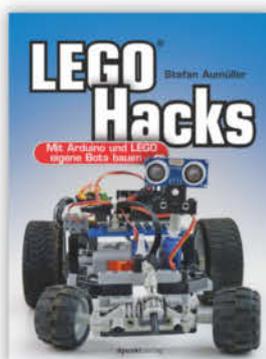
LEGO®-Modelle beleuchten

Belebe deine LEGO-
Konstruktionen mit Licht
und Lichteffekten

2020, 356 Seiten

€ 32,90 (D)

ISBN 978-3-86490-687-9



S. Aumüller

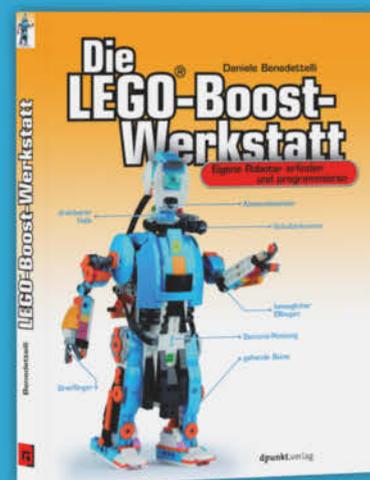
LEGO® Hacks

Mit Arduino und LEGO
eigene Bots bauen

2020, 320 Seiten

€ 29,90 (D)

ISBN 978-3-86490-643-5



D. Benedettelli

Die LEGO®-Boost-Werkstatt

Eigene Roboter erfinden
und programmieren

2020, 272 Seiten

€ 26,90 (D)

ISBN 978-3-86490-644-2

13 Tom Forsyth und Owen Gildersleeve haben den Star-Wars-Roboter K-2SO nicht nur aus Papier gebaut, sondern ihn auch noch mit leuchtenden Augen und Kopfbewegungen ausgestattet.



delle können optisch ohne Weiteres mit 3D-gedruckten Objekten mithalten.

Andere Materialien

Ursprünglich war Pepakura zwar für Papier gedacht, Maker beschränken sich aber längst nicht mehr auf dieses Medium. Insbesondere die Cosplay-Szene verschiebt ständig die Grenzen des Möglichen und sucht nach immer neuen Materialien für Experimente. Ein besonders beliebter Werkstoff ist EVA-Foam, aus dem leichtgewichtige Schwerter und Rüstungsteile entstehen 12. Um EVA-Foam zuzuschneiden, druckt man das Schnittmuster auf Papier aus und überträgt die Formen mit einem Stift. Doch auch die ständig wachsende Schneidkraft von Haushalts-Schneidplottern eröffnet neue Möglichkeiten: Wenn ein Schneidplotter Leder, Stoffe, Filz, dünnes Holz und Aluminium schneiden kann, kann man auch daraus 3D-Modelle falten. Und nimmt man noch ein paar Motoren und LEDs dazu, leuchten und bewegen sich die Modelle 13.

Und los geht's

Wer direkt loslegen möchte, hat verschiedene Möglichkeiten: Geübte Selbsterbauer können direkt auf die Jagd nach den coolsten Pepakura-Dateien gehen. Will man erstmal nur in das Hobby reinschnuppern, kann man vorgeschchnittene Bastelbögen online bestellen, die von Designern mit viel Liebe zum Detail entwickelt wurden (siehe auch Seite 123). Für den Start sollte man sich auf jeden Fall keine zu ambitionierten Ziele setzen. Statt des lebensgroßen Killerroboters bietet sich ein kleines, gut aufbereitetes Projekt an.

—rehu



**WIR MACHEN
KEINE WERBUNG.
WIR MACHEN EUCH
EIN ANGEBOT.**

ct

ct.de/angebot

Jetzt gleich bestellen:

 ct.de/angebot

 +49 541/80 009 120

 leserservice@helse.de

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

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

**Als Willkommensgeschenk erhalte ich eine Prämie nach Wahl,
z. B. einen Bluetooth Kopfhörer.**

© Copyright by Maker Media GmbH.



Was uns inspiriert





Bilder: Masaki Seki

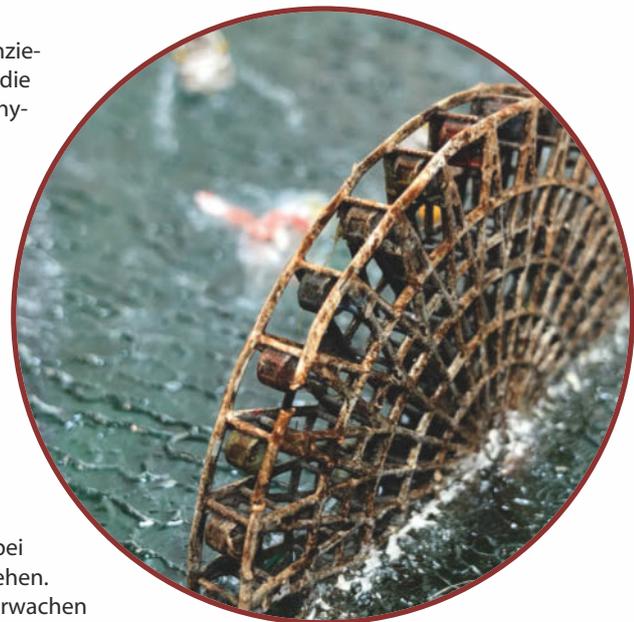
Wasserwelten

Stadtlandschaften unter Wasser, die langsam zu Trümmern am Meeresboden zerbröckeln, sind ein beliebtes Motiv in Endzeitfilmen. Irgendwie fasziniert uns das Zusammenspiel aus Wasser, bekannten Orten und städtischem Verfall. Wie viele andere fühlt sich auch der japanische Künstler Masaki Seki zu diesen Bildern hingezogen. Seine Umsetzung ist aber alles andere als gewöhnlich.

Seine Unterwasser-Modellbaulandschaften zeigen überschwemmte Städte, zerfallende Freizeitparks und gespenstisch beleuchtete Wolkenkratzer mit mustergültig umgesetzten Details. Diese Detailverliebtheit scheint kein Ende zu nehmen: Wer genau hinschaut, erkennt mit jedem Blick neue Elemente.

Ein weiterer Aspekt der Anziehungskraft ist die Art, wie die Modelllandschaften der Physik zu trotzen scheinen, wenn etwa die aufpeitschenden Ozeanwellen einfach in der Luft hängen bleiben, wie eingefroren. Das Geheimnis dahinter ist zugleich der schwierigste Teil der Arbeit, so Seki. Nach dem akribischen Formen, Zurechtschnitzen und Anmalen der Landschaften gießt er das Harz, das schließlich das Wasser darstellen soll. Dabei dürfen keine Blasen entstehen. Mit ein wenig Feinschliff erwachen die Kunststücke schließlich zum Leben.

—Caleb Kraft/hch



► [instagram.com/masaki1006](https://www.instagram.com/masaki1006)



Bilder: Mike Warren

Auf der Suche nach dem Feuer

Skateboarding ist immer ein leicht gefährliches Hobby. Man stürzt sich aus dem Lauf heraus auf eine Unterlage, die vom Design her beweglich ist. Hinzu kommt, dass die meisten Skateboard-Tricks Variationen von Sprüngen und Drehungen des Boards beinhalten, die es beinahe unmöglich machen, darauf wieder zu landen. Die Unsicherheit hat Autodesk-Designer Mike Warren aufgegriffen und auf die Spitze getrieben, indem er eine Apparatur unter sein Board geklemmt hat, die als rudimentärer Flammenwerfer funktioniert. Nun kann er beim Skaten mit seinem Fuß einen Button drücken, um den flüssigen Brennstoff unter dem Board freizusetzen, der schließlich von einem Grillfeuerzeug entzündet wird.

„Ich habe das Projekt umgesetzt, weil ich gerne Ideen zusammenwerfe, um auf wilde Dinge zu kommen“, erklärt Warren. „Dieses Projekt entwickelte sich von ‚Ob ich das wohl zum Laufen bekomme?‘

zu ‚Wann ist es zu viel Feuer?‘, und die Antwort ist: ‚Es ist niemals zu viel Feuer.‘“ Überraschenderweise hat Warren keine Ahnung von der Sportart und kann nicht einmal richtig Skateboard fahren. „Ich hatte die Vision von etwas, das noch nie gemacht wurde, und habe damit angefangen. Auch wenn ich das fertige Projekt liebe, kommt meine Erfüllung vom Lösen der Probleme während des Bastelprozesses. Es ist die Reise, nicht das Ziel.“

Das Resultat ist jedenfalls umwerfend. Wie in einer Szene aus *Zurück in die Zukunft* hinterlässt Warrens Board einen Sog aus brennenden Spuren und eroberte so, unpraktisch und gefährlich wie es ist, die Bewunderung zahlreicher Draufgänger – bis Instructables die Nachbauanleitung entfernte.

—Caleb Kraft/hch

► michaelsaurus.com



Beinahe unsichtbär

Beim Blick auf unbelebte Objekte zeigen sich manchmal unerwartete Gesichter, wenn etwa ein Wasserhahn wie eine Nase erscheint oder britische Steckdosen erschrocken in den Raum schauen. Dieses Phänomen hat einen Namen: Pareidolie. Darauf setzt auch die Londoner Künstlerin Debbie Lawson beim Erschaffen ihrer faszinierenden Skulpturen, geht dabei aber in eine etwas andere Richtung.

„Aus Mustern und Texturen scheinen mich immer Bilder anzuspringen und wenn ich sie einmal gesehen habe, kann ich sie nicht mehr Nicht-sehen“, so Lawson. „Das Gleiche passiert mir bei Möbeln. Sie scheinen mir oft wie ein bestimmtes Tier oder ein menschliches Merkmal. Wenn so eine Idee erst mal in meinem Kopf ist, arbeite ich damit, um das Tier oder Merkmal hervorzuarbeiten.“ In einer Serie an Kunstwerken erweckt sie Bären aus Orientteppichen heraus zum Leben, inspiriert von ihren Reisen durch die Welt.

„Ich war entzückt von der Geschichte und Verbreitung der Orientteppiche, die ich in Staaten der ehemaligen Sowjetunion gesehen habe, wo sie seit Jahrhunderten von Hand hergestellt werden, ebenso wie in Teilen von Europa“, berichtet sie.

Für den Bärenbau benötigte sie ein Set aus zwei identischen Teppichen. Mit Kaninchen draht, Klebeband und sehr viel Kraftaufwand brachte sie den Bär in Form, bevor schließlich einer der Teppiche als „Haut“ zum Einsatz kommt. Der zweite bleibt als Hintergrund intakt. Das fertige Kunstwerk lässt einen auf den ersten Blick wundern, ob man gerade richtig gesehen hat. Oder sich das Bild vielleicht nur eingebildet hat? Genau so, wie es sich zuvor vor Lawsons geistigem Auge abgespielt haben mag.

—Caleb Kraft/hch

► debbielawson.com

Was inspiriert Dich?

Wir freuen uns über Vorschläge an:

mail@make-magazin.de



Bild: Debbie Lawson



Open Window Alarm

Im Hinblick auf Heizkosten und zwielichtige Eigentumsveränderer (Einbrecher) sollte man in der kalten und dunklen Jahreszeit auf unnötig lang geöffnete Fenster achten – oder diesen Sensor nachbauen.

von Armin Joachimsmeier

Auf die Dauer fand ich es nervig, abends trotz Heizung in ein kaltes Badezimmer zu kommen, nur weil ich oder meine Kinder morgens vergessen hatten, das Fenster nach dem Lüften wieder zu schließen. Und nach dem Verlassen des Hauses bei offen gelassenen Fenstern hatte ich den ganzen Tag über immer ein schlechtes Gewissen wegen des erhöhten Sicherheitsrisikos, das Diebe anlocken kann.

Schon für den letzten Winter habe ich mir daher diesen batteriebetriebenen Fenster-Offen-Sensor gebaut. Er meldet den durch ein offenes Fenster verursachten Temperaturabfall per lautstarkem Alarm, mahnt so zum Schließen, ist preiswert, hält ein Jahr mit einer Batterie und alles läuft ganz ohne Cloud und Home Automation.

Der Detektor gibt fünf Minuten nach Öffnen des Fensters Alarm. Damit die Nachbarn nicht zu sehr genervt werden, geht er irgendwann in ein Intervall über – die Sache ist also praxistauglich. Die Detektoren lege ich einfach im Herbst auf die Fensterbänke und erst im Frühjahr werden sie wieder abgeräumt. Es ist in der Zwischenzeit kein Batteriewechsel nötig. Die Energie eines kleinen Akkus oder zweier AAA-Batterien reicht für mindestens eine Saison.

Um den Detektor einfach und preiswert zu halten, habe ich möglichst einfach zu beschaffende oder ohnehin rumliegende Teile verwendet. So stammt der Alarm-Lautsprecher aus einem alten Billig-Kopfhörer, der Reset-Taster und die Schutzdiode sind aus Schrott-Geräten ausgelötet und der Akku schließlich stammt aus einem alten Handy. Neben der Vermeidung unnötigen Energieverbrauchs wird so das Projekt durch Recycling noch umweltschonender. Falls kein Alt-Akku vorhanden ist, kann der Sensor aber auch mit Batterien versorgt werden.

Sogar auf einen extra Temperatursensor habe ich verzichtet, da jeder ATmega- und ATtiny-Controller schon über einen eingebauten verfügt. Der ist zwar recht simpel, die Genauigkeit von einem Grad reicht für diesen Zweck aber völlig aus und der absolute Temperaturwert interessiert für diese Anwendung ohnehin nicht. Man muss nur im Auge behalten, dass eine Temperaturänderung ungefähr eine Minute braucht, um durch das Gehäuse zum Chip vorzudringen. Die Eigenerwärmung des Chips spielt hier keine Rolle, da er die meiste Zeit schläft. Wenn er zur Messung aufwacht, reicht die Zeit nicht aus, dass seine Verlustwärme den Sensor beeinflusst.

Vorbereitung der Arduino IDE

Programmiert werden kann das Board mit der Arduino IDE. In dieser muss noch die Digispark-Unterstützung geladen werden. Eine Anleitung dazu finden Sie online (siehe Kurzinfo-Link). Im Wesentlichen ist der Board-Treiber zu installieren und in der Arduino IDE unter *Datei* und *Voreinstellungen* als *Zusätzlicher Boardverwalter-URL* die Adresse

```
http://digistump.com/package_digistump_index.json
```

einzutragen. Danach wählt man unter *Werkzeuge* und *Board* den *Boardverwalter* und installiert dort die *Digistump AVR Boards*.

Anschließend kann man als Board das *Digispark (1mhz – No USB)* auswählen **1**. In diesem Fall ist die Frequenz egal, da das Programm sowieso den internen Timer ausschaltet.

Schließlich öffnen Sie ein neues Programmfenster mit *Datei* und *Neu*. Das dann erscheinende leere Programm wird mit dem Programmtext aus dem Repository gefüllt. Oder Sie kopieren das Verzeichnis *OpenWindowAlarm* aus dem github-Repository (siehe Link) in den Sketch-Ordner und öffnen es mit

Datei und *Sketchbook*. Mit **Ctrl** und **U** wird das Ganze dann auf das Board hochgeladen. Achtung: Das Board erst einstecken, wenn die Aufforderung dazu erscheint, also nach dem Übersetzen. Und keine Panik, man hat 60 Sekunden Zeit dafür.

Strom sparen

Die eigentliche Herausforderung an dem Projekt lag darin, das Board so zu modifizieren, dass es eine Saison mit der Energie einer Batterie oder eines alten Handy-Akkus auskommt. Man kann jeden AVR-Arduino per Software in einen Sleep-Modus versetzen, in dem er nur ein paar Mikroampere braucht. Allerdings gilt das nur für den Controller, nicht für den Rest der Schaltung auf den Boards. Der verbraucht nach wie vor relativ viel Strom (6mA), zum Beispiel die Power-LED, der Spannungsregler für VIN (externer 9V-Batterieeingang) oder der USB-Anschluss.

Als Basis für den Stromspar-Arduino habe ich mich für das Digispark-MINI-Board mit dem ATtiny85 entschieden **2**. Das ist auch schon im Normalzustand recht genügsam. Das Board gibt es in zwei Ausführungen. Ich bevorzuge die Version, die man direkt in den USB-Anschluss stecken kann (ohne Micro-USB-Buchse). Sie kostet 3,50 Euro bei Lieferung aus Deutschland oder, bei entsprechend längerer Versanddauer, 1,50 Euro aus China.

Das Board kann man einfach direkt in einen USB-Anschluss stecken und mit der Arduino IDE programmieren. Man benötigt keinen Programmier-Adapter oder USB-Kabel dafür.

Umbau

Jetzt muss man das Digispark-Board nur noch so umbauen, dass es statt 6 Milli- nur noch 26 Mikroampere braucht. Das passiert in 3 Schritten:

Power-LED

Sie wird stillgelegt. Das geht auf mehrere Arten. Man könnte zum Beispiel die Power-LED oder den dazugehörigen Vorwiderstand einfach von der Platine hebeln oder ablöten. Ich bevorzuge allerdings das Durchtrennen der Leiterbahn, dann kann man die LED später durch eine Lötbrücke wieder verwenden. Das spart **2mA**.

Spannungsregler

Der Spannungsregler (das ist das größte Bauteil auf dem Board) für eine externe Stromversorgung mit mehr als 5 Volt wird nicht benötigt, da der Arduino direkt mit den Batterien oder dem Akku verbunden wird. Trotzdem braucht er wertvollen Strom. Der Regler wird

Kurzinfo

- » Stromsparbetrieb für Digispark-Board
- » Internen Controller-Temperatursensor benutzen
- » Sleep-Modus

Checkliste



Zeitaufwand:
2 Stunden



Kosten:
2 bis 7 Euro



Programmieren:
Arduino IDE ans Laufen kriegen



Löten:
Kabel an SMD-Bauteilen anlöten,
SMD Bauteil auslöten

↓ Alles zum Artikel
im Web unter
make-magazin.de/xj1

Material

- » Digispark MINI ATtiny85
- » Mini-Taster
- » Alter Kopfhörer oder Piezo-Summer
- » Universaldiode 1 A, beispielsweise 1N4001
- » Alter Handy-Akku oder AAA-Batterie-Halter
- » Female-Pin-Header zum Beispiel aus alten IDE-Festplatten- oder Floppy-Kabeln

Werkzeug

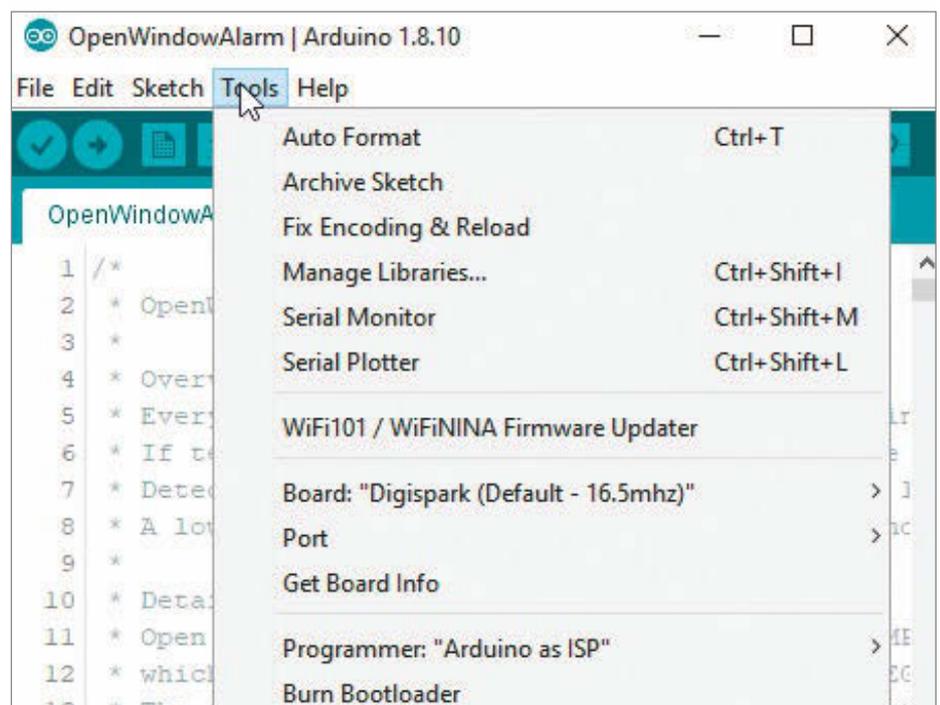
- » LötKolben klein, auch USB-LötKolben sind geeignet
- » Cutter
- » Lupe optional

daher einfach ausgelötet, den Platz kann man dann für den Reset-Taster gut brauchen. Am besten geht das Auslöten, wenn man unter eines seiner Beinchen eine Stecknadel steckt und dann damit das Beinchen beim Erwärmen anhebt. Danach wird noch der große Anschluss mit dem LötKolben erwärmt und der

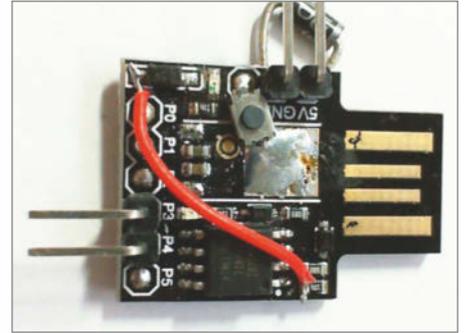
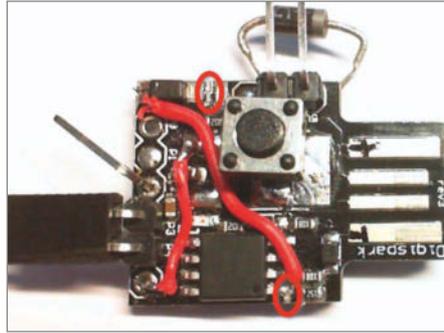
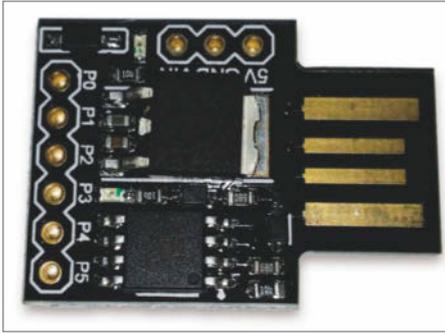
Spannungsregler kann weggeschoben werden. Das spart weitere **1,5mA**.

USB-Pullup-Widerstand

Man braucht ihn, um das Board zu programmieren. Dafür reicht es aber aus, ihn mit der



1 Hier müssen Sie als Board einen Digispark ohne USB wählen.



2 Das Digispark-Board vor der Umrüstung (links). **3** Das Board (rechts) nach der Umrüstung: Oben und unten die aufgetrennten Leiterbahnen, quer die neue USB-Pullup-Verbindung und links die Verbindung von P5 zu VIN für den Reset-Taster. Oben ist die Schutzdiode gut zu sehen, sie dient mir auch als Hinweis, wie der Akku anzuschließen ist.

4 Hier eine Version mit einem kleineren Reset-Taster. Die Verbindung von P5 zu VIN ist hier auf der Rückseite ausgeführt.

5V-Leitung des USB-Anschlusses zu verbinden, statt ihn an der ATtiny-Stromversorgung zu lassen.

Zuerst wird die Leiterbahn vom Widerstand zur ATtiny-Stromversorgung unterbrochen. Dann wird ein Kabel vom jetzt freien Ende des Widerstands zur USB-5V-Seite (am Platinenrand) der Entkoppel-Diode gelötet. Die richtige Seite kann man auch mit einem Durchgangsmesser herausfinden. Sie ist direkt mit einem der USB-Kontakte verbunden. Diese Maßnahme spart jetzt die restlichen **2,5mA**.

Reset-Taster

Da ich nicht jedes Mal die Stromversorgung unterbrechen möchte, um den Alarm zurückzusetzen, habe ich noch einen Reset-Taster nachgerüstet. Der wird zwischen dem Anschluss P5 und Masse angeschlossen. Dazu kann man gut den nun freien Platz des Spannungsreglers in der Boardmitte verwenden.

Die Seite des Tasters, die an P5 angeschlossen wird, habe ich auf die jetzt nicht

mehr verbundene VIN-Lötflächen des Spannungsreglers gelötet **3** und die dann per Kabel mit P5 verbunden. Den Taster habe ich aus einem alten Gerät ausgelötet. Schön ist, wenn man einen mit nur zwei Beinchen erwischt. Falls Sie einen vierbeinigen verwenden: Messen Sie durch, welche Pins intern permanent miteinander verbunden sind. Knipsen Sie jeweils einen davon mit einem Seitenschneider ab und isolieren Sie die Reste.

Verpolungs-Schutzdiode

Zusätzlich habe ich noch zwischen GND und 5V eine Verpolungs-Schutzdiode eingebaut **3** und **4**. Dafür genügt jedes Exemplar aus der Bastelkiste, das einigermaßen groß ist, zum Beispiel eine 1N4001. Schließt man die Stromversorgung versehentlich verkehrt herum an, fließt der volle Strom durch diese Diode. So wird zwar die Diode heiß und die Batterie entladen, doch das merkt man schnell. Es wäre doch zu schade, wenn man aus Versehen den Akku falsch herum an-

schließt und der Controller auf dem Board raucht ab.

Alarm

Als Alarm-Lautsprecher kann man alte Kopfhörer benutzen **5**. Piezo-Summer oder Buzzer mit Gehäuse **6** sind auch schön laut, da darf man nur keine aktiven nehmen. Die passiven sind auch billiger. Ab 16 Ohm Impedanz ist alles okay, 8-Ohm-Kleinlautsprecher gehen also nicht. Einfache Piezo-Scheiben ohne jedes Gehäuse sind ebenfalls ungeeignet, weil zu leise.

Den Lautsprecher habe ich an zwei Female-Pinheader gelötet **7**. Die Pinheader kann man aus alten Floppy- oder IDE-Kabeln gewinnen. Das ergibt eine Steckverbindung zu den an P3 und P4 des Boards angelöteten Male-Pinheadern. Eine Steckverbindung ist nötig, weil sich das Board mit dem angeschlossenen Lautsprecher nicht programmieren lässt. Alternativ könnte man ihn nach dem Aufspielen des Programms fest anlöten, dann hält es.

Listing 1: Konstanten zur Einstellung des Alarms

```
const uint8_t OPEN_WINDOW_ALARM_DELAY_MINUTES = 5; // Wait time between detection and activation of alarm
const int OPEN_WINDOW_ALARM_FREQUENCY_HIGH = 2200; // Should be the resonance frequency of speaker/buzzer
const int OPEN_WINDOW_ALARM_FREQUENCY_LOW = 1100;
const int OPEN_WINDOW_ALARM_FREQUENCY_VCC_TOO_LOW = 1600; // Use a different frequency
/*
 * Temperature timing
 */
const uint16_t TEMPERATURE_SAMPLE_SECONDS = 24; // Use multiple of 8 here
const uint8_t OPEN_WINDOW_SAMPLES = (OPEN_WINDOW_ALARM_DELAY_MINUTES * 60) / TEMPERATURE_SAMPLE_SECONDS;
const uint8_t TEMPERATURE_COMPARE_AMOUNT = 2;
const uint8_t TEMPERATURE_COMPARE_DISTANCE = 8; // 3 minutes and 12 seconds
// Array to hold enough values to compare TEMPERATURE_COMPARE_AMOUNT values
// with the same amount of values TEMPERATURE_COMPARE_DISTANCE positions before
uint16_t sTemperatureArray[(TEMPERATURE_COMPARE_AMOUNT + TEMPERATURE_COMPARE_DISTANCE +
TEMPERATURE_COMPARE_AMOUNT)];
/*
 * Temperature values
 */
const uint16_t TEMPERATURE_DELTA_THRESHOLD_DEGREE = 2; // 1 LSB = 1 Degree Celsius
```

Stromversorgung

Die Schaltung kann mit alten Handy-Akkus oder 2 normalen Batterien betrieben werden. An die alten Handy-Akkus habe ich Kabel mit Steckverbindern gelötet. Dadurch kann man sie nicht mehr im Handy aufladen, es sei denn, man lötet dort einfach 2 Pins an oder man baut sich eine andere Lademöglichkeit. Knopfzellen, wie zum Beispiel CR2032, sind prinzipiell auch möglich. Es ist wichtig, hier keine Billigware zu nehmen.

Wenn man die Alarmlautstärke testen will, verbindet man den Anschluss P0 mit Masse und löst dann einen Reset aus.

Auf github (siehe Kurzinfo-Link) sind weitere Bilder zum Zusammenbau des Detektors und die Software verfügbar.

Software

Der Programmablauf ist ganz einfach. Alle 24 Sekunden (`TEMPERATURE_SAMPLE_SE-`

`CONDS`, der Wert muss ein Vielfaches von 8 sein) wacht der ATtiny auf, misst die aktuelle Chip-Temperatur und vergleicht sie mit der gemessenen Temperatur von vor gut drei Minuten (`TEMPERATURE_COMPARE_DISTANCE`). Wenn die Temperatur um mehr als zwei Grad (`TEMPERATURE_DELTA_THRESHOLD_DEGREE`) gesunken ist, wird das Fenster als offen erkannt. Je nach Kälte der hereinströmenden Luft dauert das zwischen ein paar Sekunden und circa einer Minute.

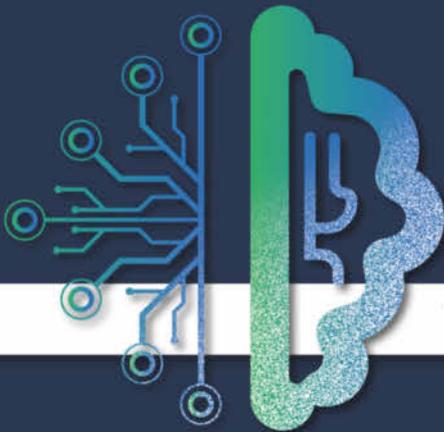
Der Vergleich mit dem drei Minuten alten Wert ermöglicht es, Temperaturänderungen von unter zwei Grad pro drei Minuten, zum Beispiel beim Abstellen der Heizung, zu ignorieren.

Nach der Erkennung wird fünf Minuten (`OPEN_WINDOW_ALARM_DELAY_MINUTES`) gewartet. Wenn dann die Temperatur wieder ein Grad niedriger ist, wird der Alarm ausgelöst. Ist die Temperatur gleichgeblieben oder wieder gestiegen, geht das Programm davon aus, dass das Fenster wieder geschlossen ist, und unterdrückt den Alarm.

In dem bereitgestellten Programm sind dann noch Verfeinerungen eingeflossen, wie mehrfache Temperaturmessung mit Over-



5 Die Lautsprecher aus einem alten Kopfhörer reichen völlig.



Künstliche Intelligenz und maschinelles Lernen – Was ist das und wo fange ich an?

Webinar am 14. Januar 2020, 11 Uhr

99,00 Euro inkl. MwSt.

Was ist Künstliche Intelligenz aktuell, was ist damit möglich und wo liegen die Grenzen? Diese und viele andere Fragen beantworten wir in diesem Webinar. Lernen Sie außerdem, welche verschiedenen Technologien sich unter dem Sammelbegriff KI befinden. Was ist der Unterschied zwischen maschinellem Lernen und Deep Learning? Anhand von Beispielen zeigen wir Ihnen zusätzlich, wie die meisten aktuellen KI Algorithmen angewendet werden und führen Sie in einige der wichtigsten Begrifflichkeiten ein. Lernen Sie, wie Ihr Unternehmen im Bereich KI effizient loslegt, ohne die Orientierung zu verlieren. Wir zeigen Ihnen auch Best Practices und besprechen Herausforderungen, denen man sich beim Start stellen muss.

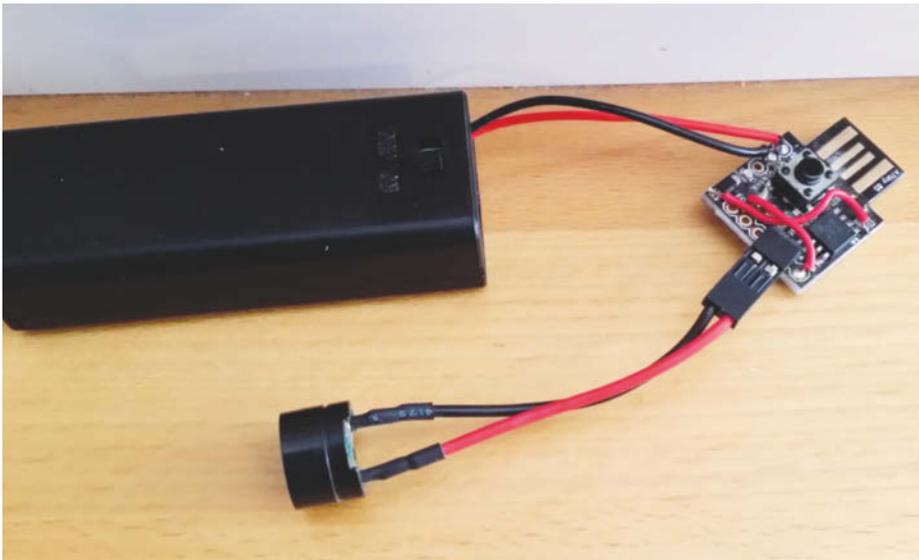
www.heise-events.de/ki_web



Referent: Philipp Brauhart



Referent: Moustapha Karaki



6 Eine Version ohne Schutzdiode mit fest angelötetem AAA-Batteriehalter und gestecktem Piezo-Summer

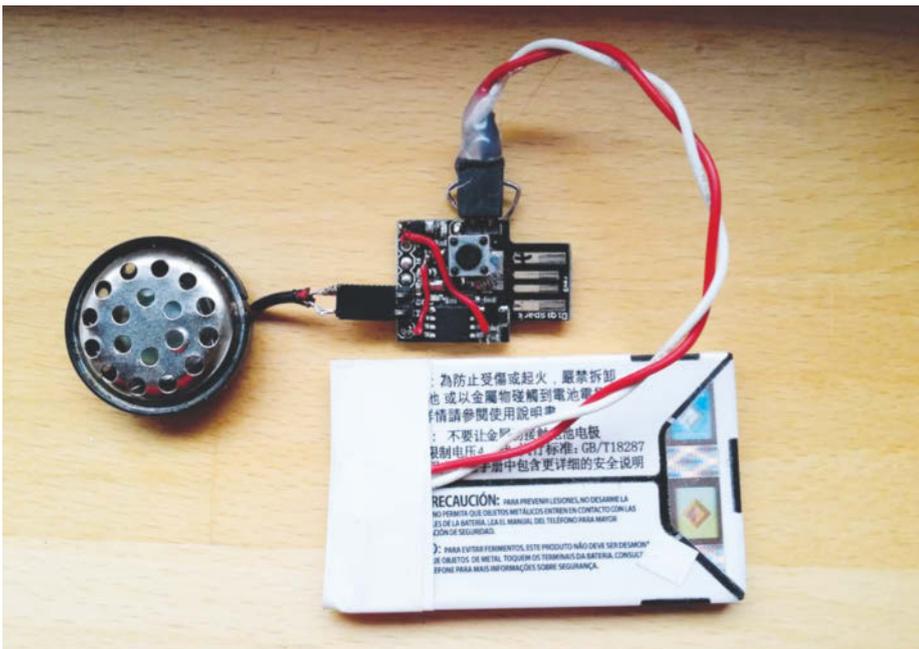
sample, eine Batterieüberwachung und das Auslaufenlassen des Alarms.

Wer die Software anpassen möchte, sollte dies leicht machen können. Im Programmtext befinden sich dazu zahlreiche Kommentare.

Zu Beginn finden Sie die bereits genannten Konstanten, mit denen die wichtigen Werte für Alarmdauer, Messungen und Frequenzen festgelegt werden (siehe Listing1). Die fünf Minuten Alarm-Verzögerung sind beispielsweise in der Konstanten namens `OPEN_WINDOW_ALARM_DELAY_MINUTES` definiert.

Die Alarm-Frequenzen lassen sich auch leicht anpassen. Optimal ist es, damit die Resonanzfrequenz des Lautsprechers zu treffen, dann wird es besonders laut. Hier hilft ausprobieren, eine der zahlreichen Smartphone-Apps zur Lautstärkemessung kann da sehr hilfreich sein.

Der Abstand zwischen den jeweils zu vergleichenden Temperaturmessungen wird nicht direkt als Sekundenanzahl, sondern als Vielfaches des zuvor im Programm festgelegten Zeitabstandes zwischen zwei Messungen (`TEMPERATURE_SAMPLE_SECONDS = 8`) festgelegt.



7 Der fertige Detektor mit Stromversorgung per Handy-Akku

Tipps zur Platzierung

Das Board liegt am besten direkt am Fenster im Schatten des Rahmenprofils. Mir ist es schon passiert, dass das Board so auf der Fensterbank lag, dass bei schönem kaltem Wetter durch das Aufmachen des Fensters die Sonne auf das Board schien und dadurch die Abkühlung der Raumtemperatur nicht mehr erfasst wurde.

legt. `TEMPERATURE_COMPARE_DISTANCE = 8` bedeutet daher einen Zeitabstand von 8 mal 24 Sekunden. Das sind 192 Sekunden oder 3 Minuten und 12 Sekunden.

Los geht's ...

Beim Einschalten oder nach einem Reset dauert es erst mal fünf Sekunden, in denen gar nichts passiert und in denen das Board programmiert werden kann. Danach blinkt die LED fünfmal und dann piepst der Lautsprecher für 100 Millisekunden.

Die erste Temperaturmessung passiert acht Sekunden später. Dabei blitzt die LED kurz auf. Danach wird alle 24 Sekunden gemessen, wobei die LED jedes Mal kurz aufblitzt. Die Aufblitzzeit ist übrigens die Zeit, die der ATtiny für das Postprocessing nach dem Einlesen der Temperatur benötigt, bevor er sich wieder schlafen legt. Man bekommt hier einen Eindruck, wie schnell ein 1MHz-Prozessor ein Programm abarbeitet.

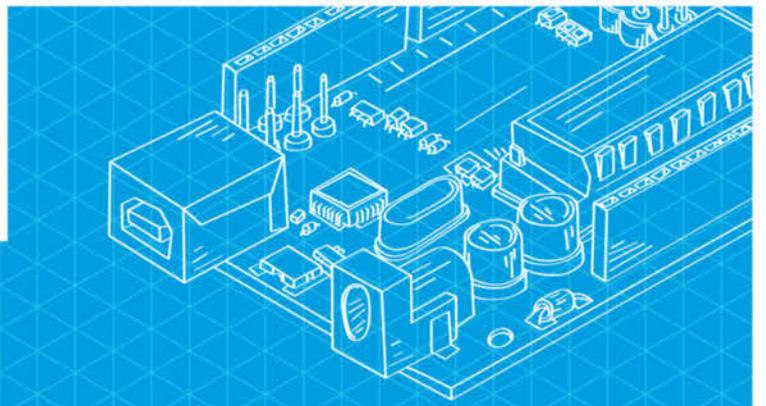
Wenn ein offenes Fenster erkannt wurde, wird das durch einen kurzen Knacks des Lautsprechers bei jeder Messung zurückgemeldet. Auch das Aufblitzen der LED dauert dann länger.

... oder auch nicht

Die ersten fünf Minuten nach dem Power-on kalibriert sich der ATtiny selber. Wenn er da schon einen Temperaturabfall bemerkt, wird der ignoriert, da er zum Beispiel vom Temperaturabfall nach dem Programmieren des Chips herrühren könnte.

Sollte das Programm nur starten, falls das Board im USB-Anschluss steckt, ist wahrscheinlich eine inkompatible micronucleus-Version aufgespielt. Ich habe das zwar noch nicht erlebt, aber es hilft, die micronucleus-Version `upgrade-t85_no_pullup.hex` aufzuspielen, die erforderlichen Dateien und ein Aufrufskript sind im github-Repository enthalten. Auch wenn einem das 5-Sekunden-Warten nach einem Reset stört, kann man diese Version aufspielen. Und nun genießen Sie Ihr warmes Badezimmer. —hgb

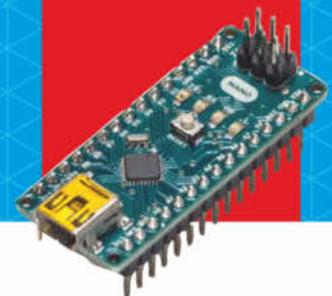
Make:



DAS KANNST DU AUCH!



GRATIS!



2x Make testen und 6 € sparen!

Ihre Vorteile:

- ✓ **GRATIS dazu:** Arduino Nano
- ✓ **NEU:** Jetzt auch im Browser lesen!
- ✓ Zugriff auf Online-Artikel-Archiv*
- ✓ Zusätzlich digital über iOS oder Android lesen

Für nur 15,60 Euro statt 21,80 Euro.

* Für die Laufzeit des Angebotes.

Jetzt bestellen: make-magazin.de/miniabo

© Copyright by Maker Media GmbH.

Feuerwerksraketen Startrampe

Unsachgemäß abgefeuerte Silvesterraketen können ins Auge gehen – davon weiß jede Notfallambulanz zu berichten. Unsere Startrampe verhindert Fehlstarts und vagabundierendes Feuerwerk.

von Guido Körber

Zum Jahreswechsel schnell anstoßen und dann zum Feuerwerk nach draußen – aber was benutzt man als Startrampe für die Raketen? Der Schampus ist möglicherweise noch nicht leer getrunken, und bei dem Gewicht von aktuellen Raketen mit ihren dicken Effektsätzen kippt eine Flasche auch schnell um. An der dann resultierenden Panik, wenn sich die Rakete horizontal in Bewegung setzt, ist nichts lustig – zumal es mit dem Reaktionsvermögen ethanolschwangerer Zuschauergruppen auch nicht zum Besten gestellt ist.

Eine ganz schlechte Idee ist es, die Rakete in den Boden zu stecken. Es ist nicht sichergestellt, dass die Antriebsleistung in der Lage ist, die Haftreibung des Bodens zu überwinden, und wenn, wird in jedem Fall die gewünschte Steighöhe reduziert. Eine bodennahe Zündung des Effektsatzes erhöht nicht etwa den Spaß am Feuerwerk, sondern führt ebenfalls zu erheblicher Gefährdung aller Personen, die sich im näheren Umfeld aufhalten.

Ein Getränkekasten zur Stabilisierung der als Startrampe dienenden Flasche verhindert zwar das Umkippen, zum Schluss stellt man dann auch noch fest, dass der Ruß der startenden Raketen an der Kiste eine ziemliche Sauerei hinterlassen hat. Ohnehin dreht sich eine in die Flasche gestellte Rakete nach Murphy immer mit der Lunte weg vom Anwender, sodass der Leitstab beim Anzünden im Weg ist. Die niedrige Lage verleitet dazu, sich beim Anzünden über die Rakete zu beugen – eine ganz schlechte Idee, wenn man die Brenndauer der Lunte bis zur Zündung überschätzt oder ein Produktionsfehler vorliegt, der zu einer Frühzündung führt. Der Autor hat als Sportschütze keine Angst vor Feuerwerk, aber einen gesunden Respekt. So wie man nicht in den Lauf einer Waffe guckt, beugt man sich auch nicht über eine Rakete, die man gerade starten möchte.

Launchpad

Das führte letztlich zu der Frage, ob man das nicht besser machen könnte – und prakti-

scherweise lagen ein paar Einzelteile herum, die sich dafür anbieten. Drei Stück Flach-eisen, die von einem alten Schrank übrig waren, bilden ein Dreibein und eine Regal-schiene die Führungsschiene für die Rake-ten. Etwas Bindedraht sorgt dafür, dass die Raketen sicher, aber gut beweglich in der Führungsschiene sitzen. Die Spreizung für die hinteren Beine und die Position der Füh-rungsschiene werden mit einem Stück Seil gesichert.

Die entstandene Vorrichtung hat mittler-weile einige Jahresenden hinter sich, und es stand nun eine gründliche Überholung an. Da die Raketen einiges an Ruß auf der Start-rampe hinterlassen, lagerte diese meistens draußen, was auf Dauer den Eisenteilen nicht so gut bekam. Die Konstruktion mit dem Seil war auch eher nicht optimal und verrutschte öfter mal. Daher wurde aus der Überholung dann ein Neubau.

In der Raumfahrt wird viel Aluminium verwendet, also wird die Startrampe Mark II auch aus Aluminium gebaut. Die Regal-schiene hatte stets den Nachteil, dass durch die gesenkten Schraubenlöcher die Füh-rungsstäbe der Raketen beim Einfädeln daran immer mal etwas hakten. Darum wird jetzt ein U-Profil verwendet (20 x 10mm), das sich auch auf eine besser geeignete Länge von 120cm zuschneiden lässt. Drei flache Profile werden als Beine auf 85cm Länge zu-gesägt.

Alle drei Flachprofile bekommen 2cm von einem Ende entfernt eine 6,5mm-Bohrung, ebenso das U-Profil 50cm vom oberen Ende entfernt. Zwei der Flach-Profile erhalten auf der anderen Seite ebenfalls eine 6,5mm-Bohrung, die besonders sauber entgratet wird: Hier wird das Seil für die Einstellung der Brei-te durchgefädelt, das sich nicht durchscheu-ern soll.

Das vordere Bein des Dreibeins muss etwas abstehen, dazu ist etwas Biegearbeit nötig. Den Winkel kann man etwas variieren; damit kann man einstellen, wie weit die Ra-kete geneigt wird. Das sollte aber nicht über-



Bild: NASA

trieben werden, eine zu flach startende Rakete fliegt eine ballistische Kurve, was bei der Artillerie Sinn macht, aber beim Feuerwerk nur wieder unnötige Gefährdung von Unbeteiligten zur Folge hat.

Hat man gerade keinen Schraubstock oder eine andere Biegevorrichtung zur Hand, reicht auch die Tischkante und eine Schraubzwinde. Das Ende mit der Bohrung wird damit leicht um etwa 15 Grad nach unten gebogen. Am unteren Ende des U-Profils kommen noch zwei Löcher rein, sodass man eine Drahtseilklemme durch das Profil stecken kann. Die Muttern liegen dann innen, die eigentliche Klemme hinter dem Profil. So kann die Position des U-Profils sauber fixiert werden.

Countdown

Alle vier Teile werden dann mit einer M6-Schraube verschraubt. Einige Unterlegscheiben, die möglichst gerade noch so in das U-Profil passen, dienen später gleichzeitig als Stütze für den Führungsstab der Raketen. Verwenden Sie eine Sicherungsmutter, damit die Teile zum Verstauen noch beweglich bleiben; zwei gekonterte M6-Muttern tun es zur Not auch.

Jetzt führt man ein Stahlseil durch die Klemme am U-Profil und dann durch die Bohrung eines der hinteren Beine, das Seil wird dann mit einer Klemme fixiert. Wenn man dann die Spreizung der hinteren Beine eingestellt hat, längt man das Seil passend

Kurzinfo

»Unfallverhütung

»Startrampe aus Alu- und Blechprofilen

»Anwendung in der Praxis

Checkliste



Zeitaufwand:
eine Stunde



Kosten:
0 bis 10 Euro für Material



Maschinen:
Standbohrmaschine oder Akkuschauber



Werkzeug:
Feile, Schraubendreher, Seitenschneider

Material

- »Alu-Flachprofil 2 bis 3mm stark, 20mm breit
- »Alu-C-Profil min. 10 × 10mm lichte Weite innen
- »Schraube M6 20mm lang, Sicherungsmutter und Unterlegscheiben

Alles zum Artikel
im Web unter
make-magazin.de/x4yb

ab und fixiert es am anderen Bein. Nun muss noch die Führungsschiene sauber in senkrechte Lage gebracht und mittels der Seilklemme fixiert werden.

Als letzter Schritt werden noch ein paar Ringe Bindendraht um das U-Profil gelegt, damit die Rakete sicher steht. Damit die Drahringe nicht verrutschen, kann man

kleine Kerben in das Profil feilen. Das verdrötelte Ende des Drahts sollte man flach umbiegen, damit man später nicht daran hängen bleibt. Damit steht dem sicheren Spaß beim Feuerwerk nichts mehr im Weg. An dieser Stelle wünschen wir unseren Lesern einen guten und sicheren Rutsch ins neue Jahr!
—cm



Die benötigten Einzelteile und Werkzeuge dürften sich in jedem Bastelkeller finden.



Neue (links) und alte Startrampe, beide aus Blech- und Profilresten gefertigt.



Glasfasern in Textilien

Leuchteffekte machen jedes Projekt zum Hingucker. Besonders schick verteilen Glasfasern bunte Lichter. Wie sie funktionieren und worauf man beim Einsatz achten muss, zeigen wir am Beispiel des Highlight-Skirt.

von Eva Ismer



Auch Glasfasern lassen sich in Makerprojekten einfach einbauen. Ein Vorteil im Vergleich zu den beliebten LED-Streifen: Sie verteilen das Licht deutlich gleichmäßiger und lassen sich überall kürzen. Damit ähneln sie elektroluminiszierenden Schnüren, die allerdings eine Hochspannungsversorgung benötigen, deren Inverter meist hörbar piepen. Dagegen reicht eine einfache, knopfzellenbetriebene LED als Lichtquelle für eine Faser völlig aus.

Der Überbegriff für lichtleitende Verbindungen ist übrigens Lichtwellenleiter. Meistens bestehen sie aus Kunststoff oder Glas. Prinzipiell sind nur die aus Glas gefertigten Lichtwellenleiter auch Glasfasern. Trotzdem werden die Begriffe Glasfaser und Lichtwellenleiter häufig synonym verwendet, manchmal werden sie auch Lichtfasern genannt. Ursprünglich wurden sie für die verlustarme Übertragung großer Datenmengen in der Nachrichtentechnik entwickelt. Beim Einsatz in Maker-Projekten steht aber überwiegend die Übertragung von sichtbarem Licht im Vordergrund. Dabei ist es in den meisten Fällen unerheblich, aus welchem Material der Lichtwellenleiter hergestellt ist. Für Experimente mit sichtbarem Licht reichen die kostengünstigen Kunststofffasern aus.

Das Beispielprojekt für die Integration von Glasfasern ist hier der Highlight-Skirt, den ich im Rahmen eines E-Textile-Kurses entworfen habe. Im Unterrock habe ich einen Lichtwellenleiter aus Kunststoff eingesetzt. Für die Ansteuerung nutze ich den Mikrocontroller Ada-

Kurzinfo

- » Aufbau und Funktion von Glasfasern
- » Glasfaser und LEDs verbinden
- » Einsatz in E-Textilien

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

Material

- » Die Materialliste finden Sie unter dem Link.

Mehr zum Thema

- » Elke Schick, Die Prinzessin auf LEDs, Make 3/15, S. 16
- » Helga Hansen, Schöner leuchten, Make 4/15, S. 21
- » Boris Kourtoukov, Body Boards – tragbare Mikrocontroller, Make 4/15, S. 29

fruit Gemma sowie ein spezielles Board zur Verbindung von Lichtwellenleiter und LED, das Lulu-Board. Seine ausladende Form verdankt der Reifrock 3D-gedruckten Teilen und an der Schulter sind weitere LEDs vernäht ①.

Worauf muss ich beim Einsatz achten?

Neben einem Lichtwellenleiter benötigt man auch eine Lichtquelle, beispielsweise eine LED. Die letztliche Leuchtkraft hängt vor allem von der Intensität der Lichtquelle ab: Je höher die Leistung der LED, desto intensiver leuchtet der Faserstrang am Ende und an den Seiten.

Hochleistungs-LEDs und LEDs mit einem engen Abstrahlwinkel funktionieren besonders gut. Außerdem werden in textilen Projekten häufig individuell adressierbare und einnähbare RGB-Farbpixel und -streifen verwendet, wie die Neopixel von Adafruit. Dabei muss man die entstehende Wärme durch die Lichtquelle beachten: die Wärmeentwicklung der verwendeten LED muss getestet werden, bevor sie in das Textil eingenäht wird. Dies gilt insbesondere bei Textilien mit Kunststofffasern, die leicht entflammbar sind.

Die Lichtwellenleiter aus Kunststoff gibt es in verschiedenen Ausführungen. Zum einen die endemittierenden Lichtwellenleiter, die in der Regel sehr dünn und in Bünd-



① Der Highlight-Skirt in Aktion



② Erster Test eines seitlich emittierenden Lichtwellenleiters

Herstellung und Geschichte

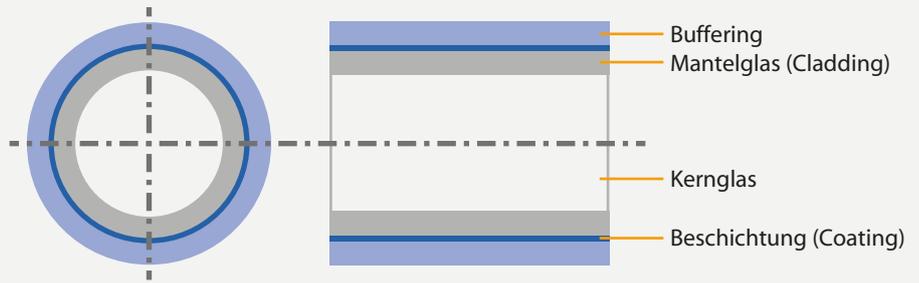
Die Herstellung der eigentlichen Glasfaser gelang erstmals im 18. Jahrhundert in der Kunstglasbläserei im Thüringer Wald. Dort wurden mithilfe des historischen Stabziehverfahrens lange dünne Fäden gezogen. Diese Glasfasern wurden zunächst jedoch nur zu Verzierungs Zwecken genutzt.

Im Jahr 1870 experimentierte der Wissenschaftler John Tyndall mit Strahlen. Er zielte einen Lichtstrahl auf einen frei fallenden, gekrümmten Wasserstrahl und zeigte, dass das Licht mit dem Wasser „floss“. Das Phänomen der Totalreflexion war noch nicht bekannt – dabei werden Wellen an Grenzflächen fast vollständig reflektiert, wenn sie sich im optisch dichteren Medium befinden. Tyndall konnte sein Experiment damals also nicht wissenschaftlich erklären, demonstrierte aber die physikalische Grundlage der modernen optischen Nachrichtentechnik, wenn auch mit einem Wasserstrahl statt einer Glasfaser.

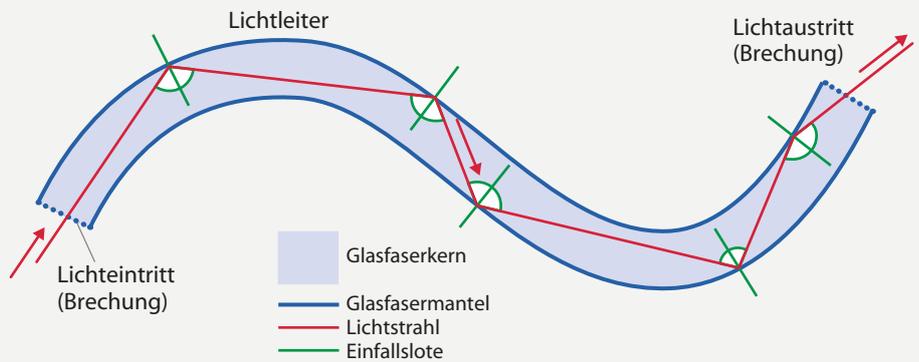
In den 1950er Jahren wurden Lichtwellenleiter dann erstmals erfolgreich zur Beleuchtung innerer Organe in der Medizintechnik eingesetzt und im Jahr 1965 gelang der Durchbruch für die Glasfaser in der Nachrichtentechnik. Heute setzen wir sie für Beleuchtungszwecke, in der Medizintechnik, der Nachrichtentechnik und der Netzwerktechnik ein.

Wie funktionieren Lichtwellenleiter?

Lichtwellenleiter sind ultraklare Stränge aus Kunststoff oder Glas, die aus einem zentralen Kern bestehen, der von einer Umhüllung und einer Schutzschicht um-



Quer- und Längsschnitt eines Glasfaserkabels



Lichtbrechung im Glasfaserkabel

geben ist. Das in den Kern einer Faser eingespeiste Licht bleibt bis zum Austritt aus dem gegenüberliegenden Ende eingeschlossen. Auf diese Weise können die Fasern Licht mit sehr geringem Verlust übertragen und sogar um Ecken biegen, wobei der maximale Biegeradius des Lichtwellenleiters zu beachten ist. Das Licht bleibt im Kern, da die Umhüllung einen etwas höhe-

ren Brechungsindex aufweist als der Kern (siehe Grafik oben).

Bereits ein gebogener, transparenter Plexiglasstab kann als Lichtwellenleiter genutzt werden. Das Licht wird dank der Totalreflexion an der Grenzfläche zur umgebenden Luft im Plexiglasstab geführt (siehe Grafik unten).

deln gruppiert sind (Durchmesser: 0,25 bis 3mm). Die Verkleidung dieser Fasern hält alle Lichtsignale in sich und strahlt helle Lichtpunkte am Ende des Strangs aus. An den Seiten tritt wenig Licht aus. Zum anderen gibt es die seitlich emittierenden Lichtwellenleiter, die in der Regel etwas dicker und flexibler sind (2 bis 12mm). Die Verkleidung der Fasern lässt etwas Licht heraus, um ein gleichmäßiges Glühen zu erzeugen ②. Schließlich gibt es auch Varianten, die an den Seiten extra beschädigt sind, um Licht austreten zu lassen, oder Bündel mit vielen kleinen Fasern. Erhältlich sind Lichtwellenleiter

auf eBay und verschiedenen Webshops. Oft heißen sie dort auch Lichtfasern.

Verbindung von Lichtwellenleiter und Lichtquelle

Der knifflige Teil ist, die LED so an den Fasern zu befestigen, dass das Licht direkt in die Fasern geleitet wird und kein Licht entweichen kann. Dabei gibt es verschiedene Möglichkeiten, eine Verbindung der beiden zu optimieren. Wichtig ist zunächst, dass die Schnittfläche der Faser gerade abgeschnit-

ten oder sogar poliert ist. Gute Ergebnisse lassen sich mit einer erhitzten Klinge erzeugen. Zum Befestigen empfehle ich Schrumpfschlauch als Verbindungsstück und Adapter aus dem 3D-Drucker.

Schrumpfschlauch eignet sich für den Einsatz von herkömmlichen, länglich bedrahteten LEDs. Der Schrumpfschlauch wird vollständig über das Kunststoffgehäuse der LED und das Ende des LWL gezogen. Beim Zusammenfügen muss man darauf achten, dass der Lichtwellenleiter beim Anbringen des Schlauches nicht schmilzt. Eine Option ist das Verkleben mit dem Industrieklebstoff E6000, der im

Internet leicht erhältlich ist. Heißkleber ist in keinem Fall eine gute Idee. Alternativ kann man auch ein kleines Loch in die Spitze der LED bohren, die Ummantelung der Glasfaser entfernen und den Faserkern direkt in die LED stecken. Eine Anleitung dazu haben die Kollegen der US-Make online veröffentlicht (siehe Link in der Kurzinfo).

Für die Neopixel habe ich einen passgenauen Adapter entworfen und 3D-gedruckt. Er wird direkt auf die Lichtquelle gesteckt und verbindet sie mit dem Lichtwellenleiter, sodass kein Licht an der Seite austritt. So kann die Faser optimal an der Lichtquelle positioniert werden und maximale Helligkeit ist gewährleistet. Für das beste Ergebnis druckt man am besten mit hoher Auflösung und minimalem Support. Dazu sollte möglichst schwarzes Filament benutzt werden, um den Streuverlust der Lichtquelle so gering wie möglich zu halten **3**.

Die Lulu-Boards

Eine weitere Herausforderung bei dem folgenden Projekt war die Integration der Lichtwellenleiter in Textilien. Dabei galt es nicht nur, den Strang optimal mit der Lichtquelle zu verbinden, sondern zusätzlich eine praktikable und optisch ansprechende Lösung für das Aufnähen und Einbinden in den Stromkreis zu finden.

Vor dieser Herausforderung standen auch schon Maurin Donneaud, Hannah Perner-Wilson und Mika Satomi, die viele E-Textilien produzieren. Sie begannen daher, eine Hardwarelösung für die Verbindung von Lichtfasern mit LED-Lichtquellen zu entwickeln. Das Design sollte robust, langlebig und komfortabel sein. Hinzu kamen Anforderungen für die Einbettung in E-Textilien: Statt Kabeln werden leitfähige Fäden als elektrische Verbindungen genutzt. Gewebe und Fasern sind weich im Vergleich zu Breadboards und durch das Tragen am Körper im täglichen Leben werden sie verschlissen.

So entstand Lulu: eine Schnittstelle zwischen LED-Lichtquelle, Lichtwellenleiterstrang oder -bündel und textilem Stromkreis. Nach vielen Prototypen **4** gibt es derzeit vier Bauformen, die allerdings immer noch in der Beta-Phase sind. Auf den Boards ist die LED jeweils im rechten Winkel aufgelötet, um das Licht auf die Faser richten zu können. Diese wird wiederum in eine Hülse aus Metall gesteckt oder mit Schrumpfschlauch fixiert. Der Lulu-Star ist ein dreieckiges Board, während Lulu-Daisy ein hoher PCB-Streifen mit Einkerbungen zum Vernähen ist **5**. Außerdem plant das Lulu-Team mit Lulu-MCU und Lucette zwei Varianten mit integrierten Mikrocontrollern. Ich habe beim Projekt nachgefragt und den Lulu-Daisy-Konnektor für mein E-Textile-Projekt „Highlight-Skirt“ als Beta-Testerin bekommen.

Das Beispiel Highlight-Skirt

In dem Projekt kombiniere ich die Welt der Photonik, der Lehre des Lichtes, mit den Möglichkeiten der Makerszene und den dort gängigen innovativen Fertigungstechnologien sowie der Modeindustrie. Den Rock habe ich im Rahmen der Fabricademy als Abschlussprojekt entwickelt, einem Kurs der Fab Foundation, in dem es um neue Technologien in der Textilindustrie, Modebranche und dem Wearable-Markt geht.

Der Highlight-Skirt besteht aus einem Reifrock mit drei 3D-gedruckten Ringen, die von einem transparenten Unterrock aus Vinyl umspielt werden. Im Saum des Unterrockes befindet sich einer der integrierten Lichtwellenleiter aus Kunststoff, der mithilfe eines Lulu- und eines Gemma-Boards in Intervallen in verschiedenen Farben beleuchtet wird **6**. Über dieser Konstruktion liegt ein asymmetrisch geschnittenes Kleid aus braunem Kunstleder. Durch die Asymmetrie ist der Unterrock nur halb verdeckt und man kann die Konstruktion und die Beleuchtung gut erkennen. Der obere Teil des Kleides ist mit einnähbaren, im Farbwechsel leuchtenden RGB-Farbpixeln versehen, die ebenfalls von einem Gemma-Board angesteuert werden.

Die Entwicklung habe ich in drei Kernbereiche unterteilt: 3D-Druck und Konstruktion, Entwurf und Design des Kleides sowie Elektronik und Programmierung der Mikrocontroller.

3D-Druck und Konstruktion

Im Reifrock stecken drei von mir konstruierte und 3D-gedruckte Ringe aus PLA, um die

3 Mein Adapter zum Verbinden von Glasfaser und LED (auf dem Lulu-Board)



5 Der Lulu-Daisy-Konnektor mit LED und Pin-Einkerbungen

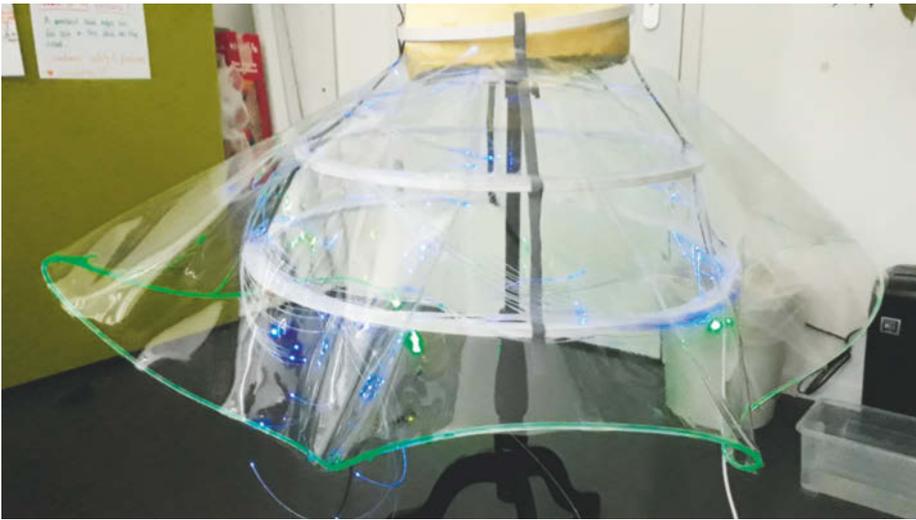


Bild: Flickr/Plusea (CC BY 2.0)



4 Prototypen der Lulu-Entwicklung

Bild: Flickr/Plusea (CC BY 2.0)



6 Prototyp des Unterrockes mit verschiedenen Lichtwellenleitern



7 Um den Rock tragen zu können, habe ich einen Ring mit Scharnieren entworfen und gedruckt.



8 Die Neopixel und der Gemma-Controller auf der Schulter

charakteristische, ausladende Form zu erreichen. Die Teile habe ich mit einem BigRep-3D-Drucker hergestellt. Der kleinste Ring ist im Unterschied zu den anderen beiden oval, um optimalen Tragekomfort an der Hüfte zu ermöglichen. Er hat ein eigens konstruiertes Scharnier zum Öffnen und Schließen und Aussparungen an den Seiten für das spätere Anbringen des Unterrockes durch Druckknöpfe 7. Die drei Ringe habe ich mithilfe von Klettband zu einem dreidimensionalen Reifrock zusammengefügt.

Auch einen Adapter zum Verbinden von LED und Glasfaser habe ich konstruiert und 3D-gedruckt. Alle Druckdateien habe ich zum Herunterladen und Selbstdrucken online bereitgestellt (siehe Link in der Kurzinfo).

Entwurf und Design des Kleidungsstückes

Das Kleid ist eine Kooperation mit der Modedesignerin Anke Schönberner aus dem Wildauer Mode-Atelier „fashion school“. Sie hat das Unikat aus hellbraunem Kunstleder maßgeschneidert. Es besteht aus einem einzigen Schnittteil, das unter dem Arm vernäht ist. Das asymmetrisch verlaufende Kleid verdeckt nur einen Teil der Unterkonstruktion des Rockes, um die eingebaute Elektronik sichtbar zu machen. Der transparente Unterrock aus Vinyl reflektiert die integrierten Glasfasern und verstärkt den Farbeffekt.

Elektronik und Programmierung

Mein Ziel war es, mithilfe programmierbarer RGB-Farbpixel seitlich emittierende Lichtwellenleiter an mehreren Bereichen des Reifrockes zum Leuchten zu bringen. Ein Programm steuert die Intervalle und Intensität der Farben. Des Weiteren habe ich zwei Neopixel mit elektrisch leitendem Garn in den oberen Teil des Kleides eingenäht 8.

Die Pixel werden jeweils von einem Gemma-Board gesteuert. Das Adafruit Gemma ist ein portables Mikrocontroller-Board auf Basis des ATtiny85. Es eignet sich für den Einsatz von textilen Projekten, da das Board mit seinen großen runden Anschlüssen direkt in Kleidung eingenäht werden kann. Außerdem verfügt es über einen kleinen Schalter zum An- beziehungsweise Abschalten. Wie andere Mikrocontroller kann es über USB an einen Rechner angeschlossen und über die beliebte Arduino-Programmierungsumgebung bespielt werden. Für die erste Programmierung reicht eines der Beispielprogramme, die in der Arduino-Bibliothek für die Neopixel enthalten sind. Anleitungen zum Installieren und Nutzen der Arduino-Software finden Sie unter dem Link in der Kurzinfo.

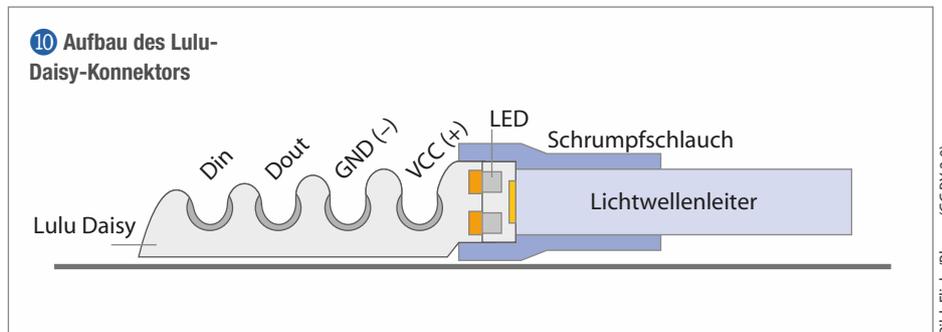
Zum Vernähen der Komponenten benötigt man ein elektrisch leitendes Garn. Die meisten sind mit einer Legierung aus verschiedenen Metallen ummantelt, etwa Silber, Kupfer oder Zinn. Der Kern besteht in der Regel aus Baumwolle oder Polyester. Ich habe Karl-Grimm-Garn, einen Silberfaden mit Kunststoffkern, verwendet.

An der Schnittstelle der LED-Lichtquelle und des Lichtwellenleiters habe ich schließlich den Lulu-Daisy-Konnektor eingesetzt ⁹. Der Streifen verfügt selbst über eine integrierte RGB-LED und wird mit elektrisch leitendem Garn direkt auf das Textil genäht. Da ein Neopixel über vier Pins verfügt (VCC, GND, Data in und Data out), sind diese im Lulu-Daisy-Konnektor als Aussparungen zum Annähen gestaltet ¹⁰. Für die perfekte Verbindung von der Lulu-LED zum Faser-Strang dient der 3D-gedruckte Adapter aus schwarzem PLA. Dieses Teil hat vier kleine Löcher und kann so ebenfalls auf dem Stoff vernäht werden. Ich habe ihn im Saum des Reifrockes genutzt.

Im obersten Ring des Reifrockes befindet sich ein weiterer, seitlich emittierender Lichtfaser-Strang. Dieser Ring kann separat von der Reifrockkonstruktion getragen werden. Er könnte zum Beispiel im Straßenverkehr auf dem Fahrrad als leuchtender Sicherheitsring dienen, um von anderen Verkehrsteilnehmern und Autofahrern besser gesehen zu werden. Der Lichtwellenleiter wird über ein RGB-Farbpixel, das von einem Gemma angesteuert wird, mit Licht versorgt. Als Upgrade werde ich an dieser Stelle zukünftig noch einen Abstandssensor integrieren, der für einen Farbwechsel sorgt, wenn sich beispielsweise ein Fahrzeug nähert.



⁹ Gemma-Board, Lulu-Konnektor und Lichtwellenleiter



¹⁰ Aufbau des Lulu-Daisy-Konnektors

Bild: Flickr/Plusea (CC BY 2.0)

Die drei Teilbereiche ergeben zusammen den Highlight-Skirt – ein experimentelles und anwendungsbezogenes E-Textiles-Projekt, das ich mit großartiger Unterstützung des ViNN: Labs, dem Makerspace der TH Wil-

dau, realisieren konnte. Es zeigt, wie vielfältig die Anwendung von Lichtwellenleitern in Textilien sein kann und welches Potenzial in der Kollaboration von Fashiondesign und der Welt der Photonik steckt. —hch

Boxtronik +++ Konfiguriere die Box +++ Und aus Deiner Schaltung wird ein





Für Deine Elektronik

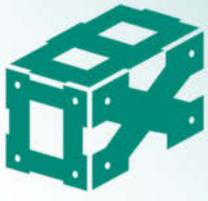
Eine total nützliche Box!

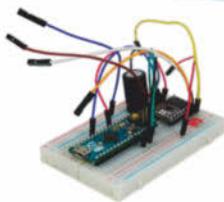


Für beliebige Steckverbinder



Mit Nutzer-Schnittstelle









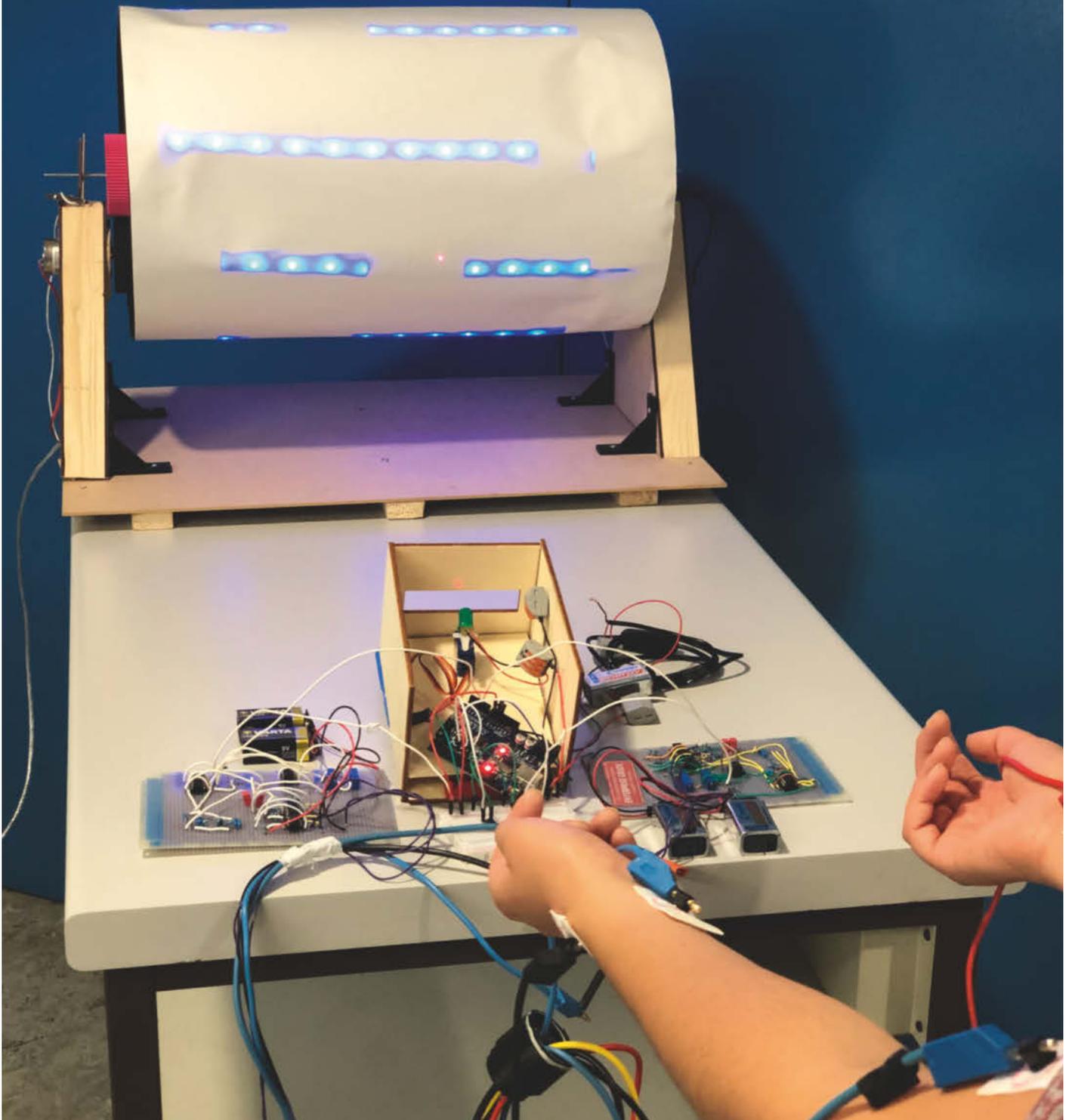


fertiges Gerät +++
<https://bloess.berlin> +++
Telefon +49 30 57703695-0

Muscle Laser Run

Spannen Sie mal Ihren linken Unterarm an. Hat es geklappt? Ein Gedanke reicht, um unsere Körper Erstaunliches leisten zu lassen – zum Beispiel eine Muskelkontraktion. Diese Muskelkontraktion wird durch elektrische Impulse ausgelöst, die man Biosignale nennt. Mit dem Bau unseres Laserparcours *Muscle Laser Run* haben wir Biosignale spielerisch erkundet.

von Nicole Bäß, Anne Bias und Selina Häuser



Was entsteht, wenn neun Medizinphysik-Studierende und ein Maschinenbaustudent 30 Stunden Zeit haben, um einen Prototyp zu entwickeln? Beim Makeathon unter dem Motto *Make Light* auf der *Laser World of Photonics 2019* in München konnten wir beweisen, wie gut Medizinphysik und Maschinenbau zusammenpassen. Gemeinsam haben wir das Spiel „Muscle Laser Run“ entwickelt. Ziel des Spiels ist es, einen Laserstrahl nur mithilfe von Muskelkontraktionen der Arme durch einen Parcours zu steuern – ganz ohne Anfassen.

Biosignale messen und verstärken

Das Konzept scheint zunächst simpel: Wir lesen die Muskelanspannung an beiden Armen aus und übersetzen sie als Anweisung, um einen Laserstrahl nach links oder rechts zu bewegen. Spannt man den richtigen Arm zum richtigen Zeitpunkt an, bewegt sich der Laser durch das Labyrinth wie in einem Computerspiel. Der Schwierigkeitsgrad steigt, sobald die Muskeln ermüden und der Spieler ungenauer wird.

Die Umsetzung benötigt dann natürlich medizinisches Verständnis. Um die Steuerung des Spiels zu bauen, müssen wir zuerst die Muskelkontraktion in den Armen messen können. Eine Muskelkontraktion ist als Biosignal messbar. Der Begriff Biosignale bezeichnet alle elektrischen Signale, die man an Lebewesen messen und überwachen kann. Aber wie genau funktioniert das? Nervenzellen besitzen konstante Ladungen. So kommt es bei einer Reizübertragung zu einer kurzzeitigen Änderung der Polarisation. Dies ist bei 1 im roten Bereich zu erkennen. Dabei misst man mit Klebeelektroden, wie viele Zellen zwi-

schen zwei Messpunkten – im Bild als graue Pfeile dargestellt – positiv und wie viele negativ sind. Das entspricht dem Verhältnis von polarisierten zu unpolarisierten Zellen. Da das Verhältnis nur relativ messbar ist, ist eine Referenz oder auch Masse erforderlich. Aus diesem Grund ist zur Aufnahme von Aktionspotentialen eine dritte Messstelle notwendig, die Masseelektrode. Die Messelektroden werden dabei parallel zur Richtung der Nervenbahnen positioniert. Wir greifen das Signal wie in 2 am Arm ab.

Analoge Filterung und Verstärkung des Biosignals

Um das Biosignal verwenden zu können, müssen wir es nun filtern und verstärken. Dafür verbinden wir Hochpässe, Instrumentenverstärker und einen Notch-Kerfilter nach der Schaltung 3. Zunächst erreicht das Signal den Hochpass. Der Frequenzbereich eines Biosignals liegt zwischen 0,5 Hertz und 100 Hertz. Signalanteile außerhalb dieses Bereiches führen zu Artefakten oder Signalstörungen. Dabei kann es sich zum Beispiel um Atmung, Handysignale und ähnliche Störfaktoren handeln. Der Hochpass agiert als Filter: Er lässt hohe Frequenzen passieren und unterdrückt Frequenzen unterhalb der Grenzfrequenz ($f_g = 1/2\pi RC$). Er besteht aus einem Kondensator und einem Widerstand. Das gefilterte Signal oszilliert um 0 Volt.

Ein Instrumentenverstärker ist eine Schaltung, die aus mehreren Operationsverstärkern besteht. Instrumentenverstärker werden meist in EKG-Messgeräten als Messwertverstärker eingesetzt, um die Signale aufzubereiten. Der Vorteil der Instrumentenverstärker-Schaltung ist, dass Spannungsdif-

Kurzinfo

- » Biosignale messen und verstärken
- » Laser mit Biosignalen steuern
- » Unendliches Labyrinth konstruieren

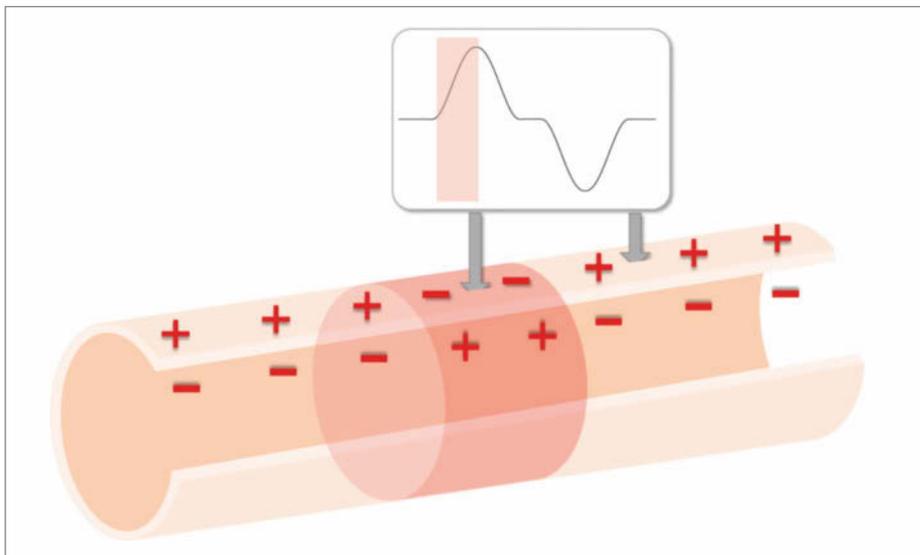
Checkliste

-  **Zeitaufwand:** ein Wochenende
-  **Kosten:** etwa 250 Euro
-  **Löten:** Bauteile auf Platine löten
-  **Holzbearbeitung:** Holzplatten mit dem Lasercutter zuschneiden
-  **3D-Druck:** Zahnräder drucken
-  **Programmieren:** Arduino Uno mit der Arduino-IDE

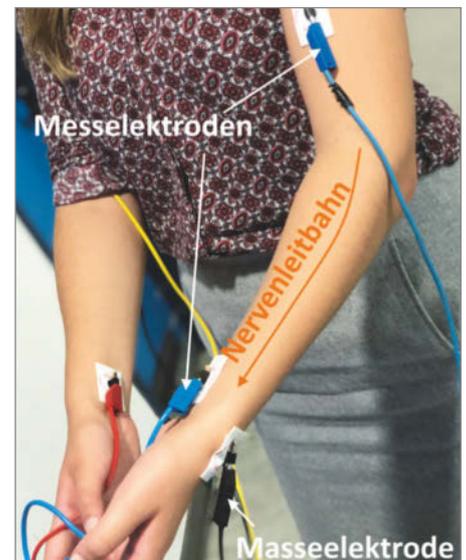
Material

Die Materialliste finden Sie unter dem Link.

↓ Alles zum Artikel im Web unter make-magazin.de/xx7h



1 Schematische Darstellung der Polarisationsänderung von Zellen entlang einer Nervenbahn



2 Position der Elektroden am Arm

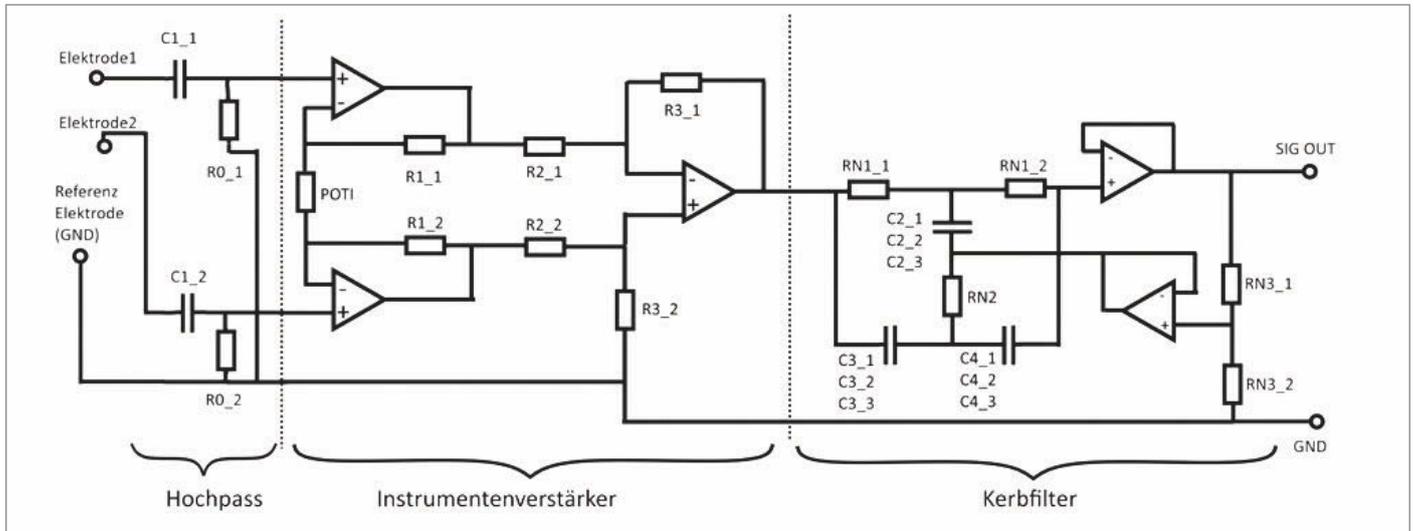


Bild: Josef Kauer-Bonin

3 Alle im Schaltplan verwendeten Bauteile sind in der Materialliste in den Download-Links zu finden.

ferenzen verstärkt werden, aber nicht der Stromfluss. So werden Gleichtaktsignale nicht berücksichtigt. In dieser Schaltung ist die Gesamtverstärkung durch ein Potentiometer einstellbar und berechnet sich nach $V=(1+(2 \cdot R1)/POT1) \cdot R3/R2$. Der Nachteil der Schaltung ist, dass auch die Spannungsversorgung der Operationsverstärker, also ein 50-Hertz-Netzbrummen, verstärkt wird. Dann folgt der Notch-Kerbfiler. Er unterdrückt eine diskrete Frequenz nach $f_G=1/(2\pi \cdot RN1 \cdot C3)$ und damit das Netzbrummen.

Es gibt verschiedene Möglichkeiten, um den Schaltplan nachzubauen: Zum Beispiel kann man alle Bauteile auf einer Rasterplatte mit Draht verlöten 4. Der EAGLE-Schaltplan steht zum Download zur Verfügung (Link in der Kurzinfo). Einfacher geht es, wenn man die Platine kostengünstig von einem PCB-

Hersteller produzieren lässt. Die Gerber-Datei gibt es ebenfalls zum Download in der Kurzinfo. Die Versorgung des Boards ist mit -5 Volt und +5 Volt oder auch -9 Volt und +9 Volt durch gegeneinander verschaltete Batterien möglich.

Laser mit Biosignalen steuern

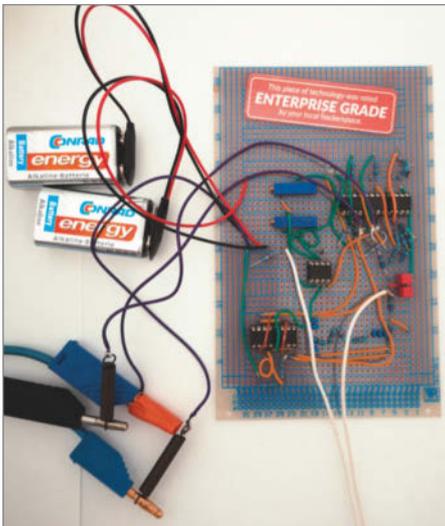
Achtung: Laserschutz beachten. Die Laserleistung von 1mW darf nicht überschritten werden.

Nun können wir Biosignale an einem Arm messen und verstärken. Wir löten eine zweite Platine für den zweiten Arm, damit uns zwei Signale zur Verfügung stehen. Im nächsten Schritt wollen wir mit den beiden Biosignalen einen Laser durch ein Labyrinth steuern:

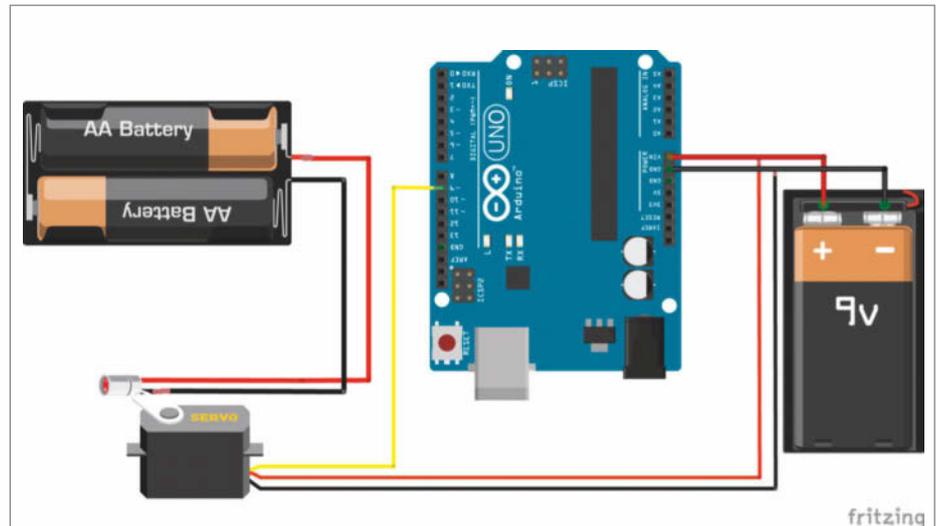
Durch das Biosignal vom linken Arm bewegt sich der Laser nach links und mit dem Signal des rechten Arms bewegt sich der Laser nach rechts. Zu diesem Zweck befestigen wir den Laser auf einem Servomotor. Dann steuern wir den Servomotor mit dem Arduino an. Als Laser verwenden wir eine Laserdiode, die mit 3 Volt betrieben werden kann. Dies ist nötig, da der Schaltkreis 5 mit einer 3-Volt-Batterie versorgt wird.

Programmierung

Mit dem Quellcode in 6 werten wir das analoge Biosignal aus. Da das Biosignal stark verrauscht ist, wird es erst ab einem bestimmten Schwellwert weiterverarbeitet. Auf diese Weise kann man den Servomotor recht präzise mit der Anspannung eines Muskels steuern.



4 Rasterplatte mit verlöteten Bauteilen zum Filtern des Biosignals



5 Lasersteuerung durch Biosignale mit einem Arduino Uno

ern. Dabei sollte man den Schwellwert individuell nach der Signalstärke festlegen. Außerdem ist es möglich, die Geschwindigkeit und die maximale Auslenkung des Servomotors zu variieren.

Alternative Steuerfunktion

Wenn der Spieler erschöpft ist und die Muskelkraft nachlässt, kann er statt der Biosignale auch zwei Taster verwenden **7**. Dazu ersetzt man die analogen Pins des Arduinos, die durch die Platinen belegt sind, mit digitalen Pins. Außerdem muss man den Quellcode anpassen. Quellcode und Schaltplan gibt es ebenfalls im Link in der Kurzinfor.

Unendlich langes Labyrinth konstruieren

Jetzt können wir einen Laser bewegen, aber wie wird daraus ein Spiel? Dazu bauen wir ein unendliches Labyrinth aus LEDs auf einer sich drehenden Rolle. Die leuchtenden Stellen bilden die Mauern des Labyrinths. Die dunklen Bereiche bilden den Weg, auf dem man den Laserstrahl mithilfe der eigenen Biosignale lenken muss **8**.

Als Drehachse für die Labyrinth-Rolle verwenden wir eine 42cm lange Holzlatte. In beide Enden der Achsen bohren wir mittig ein Loch, um anschließend 7cm lange Metallstangen einzusetzen. Danach schneiden wir drei Kreisscheiben mit einem Lasercutter zu. Die Laserdatei befindet sich auch unter den Downloadlinks. Auf den beiden äußeren Holzkreisen bringen wir einen Klettstreifen an. Das Gegenstück des Klettstreifens befestigen wir am linken und rechten Rand einer Papierbahn. Daneben kleben wir jeweils eine Bahn der Länge nach mit Duck Tape ab. Das LED-Band heften wir in Bahnen mit 12cm Abstand auf das Papier. Nun muss man entscheiden, wo und wie groß die Lücken im Parcours sein sollen. An den entsprechenden Stellen klebt man Duck Tape zwischen LED-Streifen und Papier. Anschließend klebt man Duck Tape von hinten auf die gesamte erste LED-Bahn. Diesen Vorgang wiederholen wir für acht weitere Bahnen. Im Anschluss kleben wir die gesamte Innenseite mit schwarzem Duck Tape ab und führen die restliche Länge des LED-Streifens zur Mittelachse. Die Stromversorgung der LEDs **9** soll nicht mitrotieren, damit sich kein Kabel aufwickelt. Dafür führen wir die Versorgung über die elektrisch leitende Achse zu einer Halterung. Die Masse des LED-Streifens führt man dazu mit einer Krokodilklemme an eine Metallstange der Achse. Einen weiteren Ausgang des LED-Bandes verbindet man mit einer Krokodilklemme mit der anderen Metallstange. Die Rolle kann man nun mit den Klettstreifen auf den Holzreifen spannen.

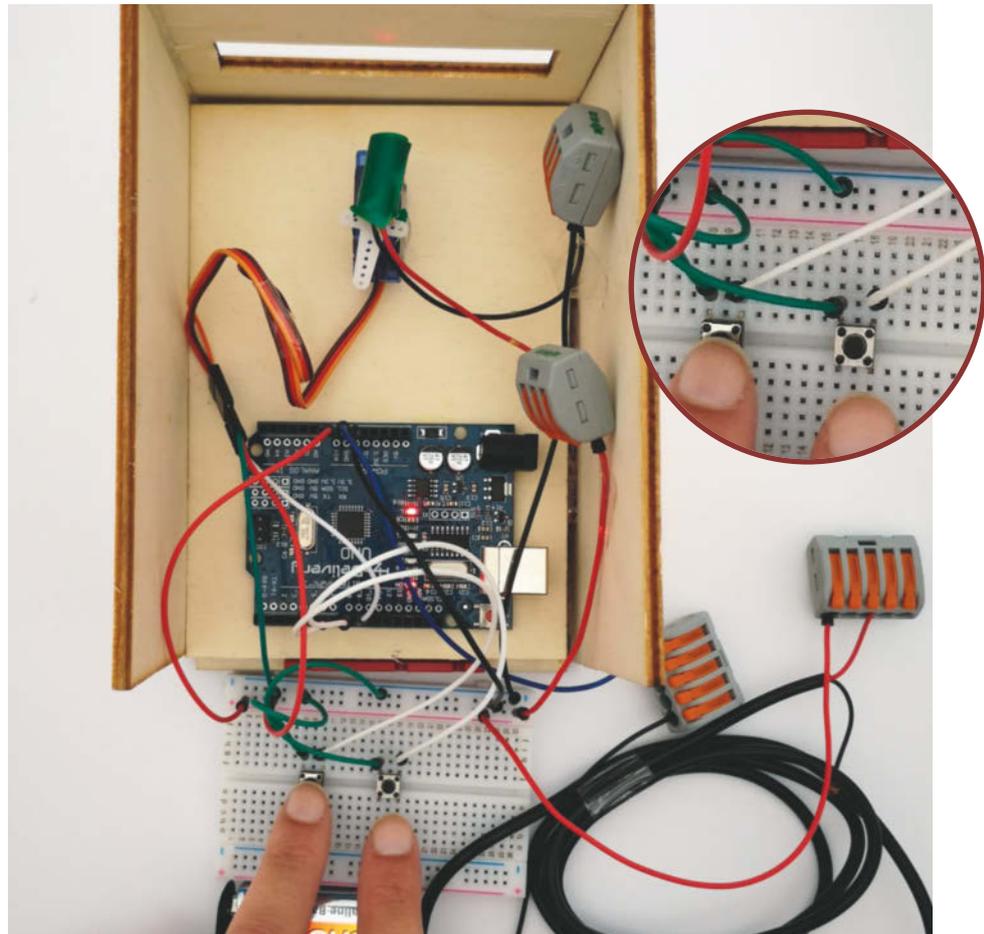
6 Listing 1

```

1 #include <Servo.h>
2
3 Servo myservo;
4   int LINKS = A3;
5   int L = 0;
6   int RECHTS = A2;
7   int R = 0;
8   int pos = 90;
9   int thresholdL = 5;
10  int thresholdR = 150;
11
12 void setup() {
13   Serial.begin(9600);
14   myservo.attach(9);
15 }
16
17 void loop() {
18   L = analogRead(LINKS);
19   R = analogRead(RECHTS);
20
21   if ( L >= thresholdL && L >= R && pos <= 130){
22     myservo.write(pos);
23     delay(20);
24     pos += 1;
25   }
26
27   else if ( R >= thresholdR && R >= L && pos >= 70){
28     myservo.write(pos);
29     delay(20);
30     pos -= 1;
31   }
32 }

```

Die erste Zeile bindet die Bibliothek für den Servomotor ein. Zeilen 3–8 deklarieren die Grundposition und die Pinbelegung. In den Zeilen 9 und 10 definieren wir die Schwellwertbedingungen. Ab Zeile 17 lesen wir Daten vom Analogpin aus und geben in den Zeilen 21–24 Bedingungen vor: Wird der Schwellwert überschritten und ist das Signal in A3 > A2, dann bewegt sich der Motor um +1° (nach links), ist das Signal in A2 > A3, dann bewegt sich der Motor um -1°, also nach rechts.



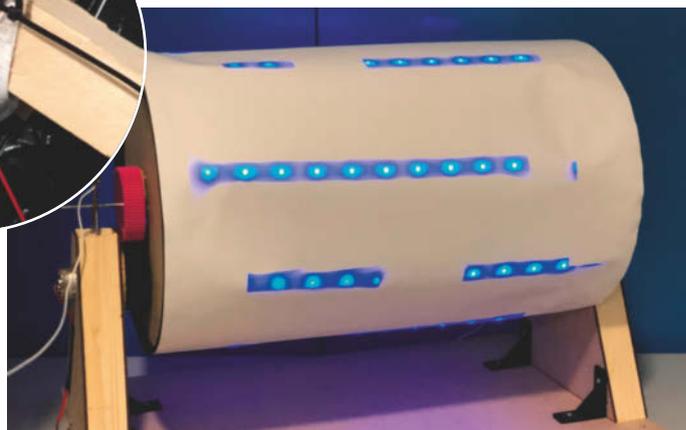
7 Lasersteuerung mit Taster



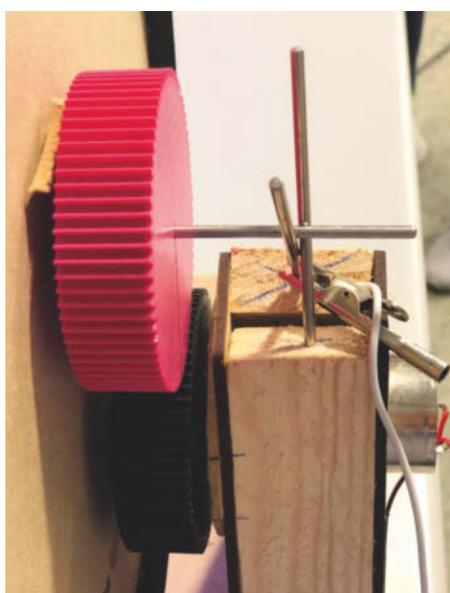
8 Parcours-Rolle von innen



9 Stromversorgung der LEDs



8 Parcours-Rolle von außen



10 Antrieb mittels Elektromotor und Zahnrädern

Antrieb

Die Rolle soll rotieren können, also benötigen wir einen Elektromotor als Antrieb. 3D-gedruckte Zahnräder ermöglichen dabei die Kraftübertragung mit einem Verhältnis von 1:1 auf die Rolle **10**. Um den Elektromotor zu betreiben, baut man ihn wie in **11** auf. Der Elektromotor wird mit einer 12-Volt-Batterie versorgt. Damit wir die Drehgeschwindigkeit variieren können, benutzen wir einen Arduino. (Der Programmcode ist in den Links in der Kurzinfo.) Der Arduino steuert einen Feldeffekttransistor. Dieser steuert mittels Pulsweitenmodulation die effektive Spannung im Motor. Hier kann man über den seriellen Monitor die Geschwindigkeit durch die Eingabe eines Wertes zwischen 0 und 1024 regulieren. Dafür ist allerdings eine ständige Verbindung zu einem Notebook nötig. Als Pulsweitenmodulation bezeichnet man ein ständiges An- und Abschalten des

Stroms. Da sich im Elektromotor Spulen befinden, entstehen dabei Induktionsspitzen. Diese ungewollten Spannungen können die Elektronik zerstören. Die parallel geschaltete Freilaufdiode verhindert diesen Effekt.

Halterung

Damit sich die Rolle frei drehen kann, ist ein Gestell nötig. Das konstruieren wir schnell aus zwei Holzplatten mit eingesetzten Metallstangen **12**. Die Kabelenden des LED-Bandes führen wir an je ein Ende der Metallstange: So wird der Strom durch die Metallstangen der Halterung an die Metallstangen der sich drehenden Achse übergeben **8**. Um den Elektromotor zu befestigen, schneiden wir vier Sichtblenden mit dem Lasercutter zu. Den Elektromotor können wir nun in die Aussparung setzen. Um diesen vollständig zu fixieren, setzen wir eine weitere Platte vor die Verkleidung und schrauben

EKG-Messung

Das Herz ist ein ganz besonderer Muskel. In diesem Organ werden elektrische Impulse erzeugt, die den Herzschlag auslösen. Genauer bildet sich dieser Impuls im sogenannten Sinusknoten. Das ist ein Bereich im rechten Vorhof, der den Impuls auf die Muskulatur der beiden Vorhöfe überträgt. Das daraus bedingte Zusammenziehen der Muskeln presst das Blut in die Herzkammern. Erreicht der Impuls den sogenannten AV-Knoten, ziehen sich die Herzkammern zusammen und das Blut kann in die Körpergefäße transportiert werden. Das elektrische Biosignal bestimmt also den Takt des Herzens und wird bei der Elektrokardiographie (EKG)

gemessen. Es gibt Information über den Herzrhythmus, die Frequenz und die Erregungsbildung, -ausbreitung und -rückbildung des Herzens.

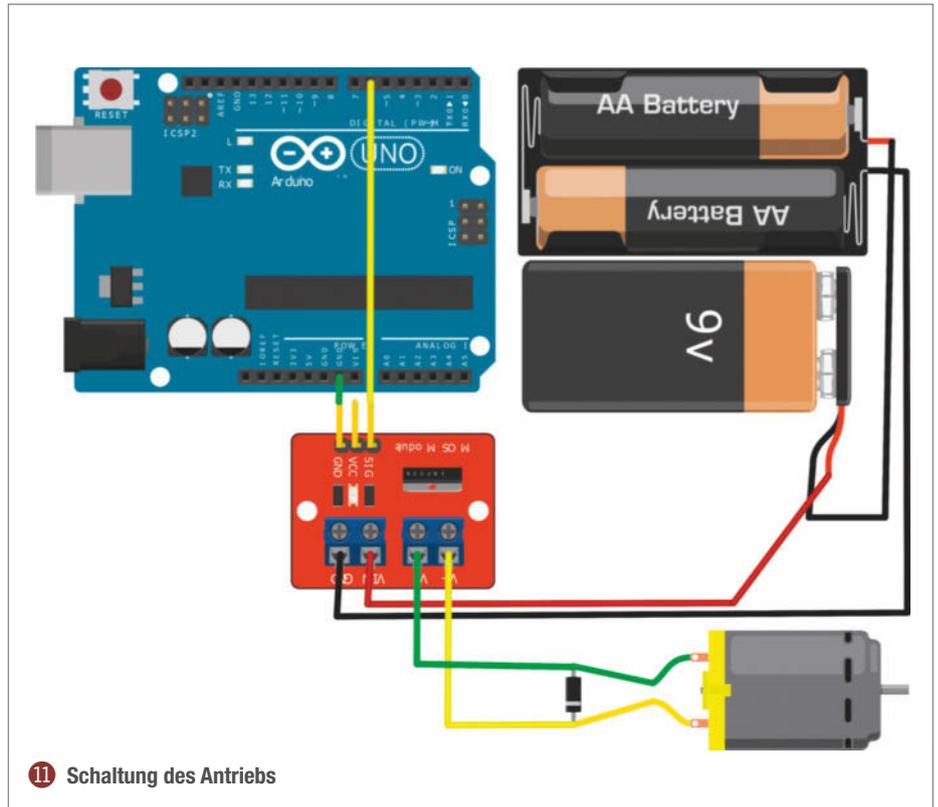
Die Messpunkte zur Registrierung der EKG-Signale sind in sogenannten Ableitungen festgelegt. Existierte diese Standardisierung nicht, gäbe es eine unüberschaubare Vielfalt an unterschiedlichen EKGs. Für viele Ableitungen verwendet man selbstklebende Einmal-Elektroden, die meist mit einer Art Kontaktgel beschichtet sind. Der Vorteil liegt hier in der großen Kontaktfläche. Dank ihr kann man die elektrische Aktivität des kompletten

Herzens messen. Im Gegensatz dazu verwendet man für die Messung von Aktivitäten einzelner Zellen mikroskopisch kleine Nadelelektroden. Die Elektroden bestehen meist aus Metallen wie Silber oder Zinn. Das Material und die Geometrie der Elektroden beeinflussen die Übertragung und den Frequenzbereich der messbaren Signale. Sind also keine Einmal-Elektroden parat, kann man zum guten Silberlöffel aus Omas Besteckkasten greifen. Befeuchtet man zusätzlich die Hände mit Wasser, ist die elektrische Leitfähigkeit des menschlichen Körpers größer: Es ist möglich, dass man tatsächlich ein elektrisches Biosignal abgreifen kann.



12 Seitengestell ohne Verkleidung

sie fest. Anschließend füllen wir das Loch des zweiten Zahnrades mit Heißkleber und setzen es auf den Elektromotor. Die gesamte Vorrichtung ist aus Stabilitätsgründen auf einer großen Holzplatte mithilfe von Winkeln fixiert. Jetzt kann sich die Rolle auf dem Gestell drehen und das Spiel losgehen: Elektroden ankleben, Verkabelung kontrollieren, den Laser auf die Rolle richten und die Rolle in Bewegung setzen. Wer es schafft, am längsten im Labyrinth zu bleiben, gewinnt. —*rehu*



11 Schaltung des Antriebs

Der Makeathon

Wir haben am Makeathon teilgenommen, nachdem wir das Wahlpflichtmodul Optoelektronik an der Beuth Hochschule für Technik Berlin belegt hatten. In diesem Kurs trainierten die Dozent*innen unseren Teamgeist und Einfallsreichtum durch Wettbewerbe in Kleingruppen: So traten wir gegeneinander an, um den höchsten Turm ausschließlich mithilfe von Papier, Klebeband und einer Schere zu konstruieren, und konkurrierten um das Schiff mit der größten Traglast, das wir aus einer Holzlatte und verschiedenen Klebstoffen bauten.

Auf dem Makeathon kam unserer Gruppe die Idee, Biosignale mal anders einzusetzen. Da wir uns auf der *Laser World of Photonics* Messe befanden, wollten wir einen ansteuerbaren Laser in unser Projekt integrieren. Außerdem wollten wir einen spielerischen Gedanken bewahren, woraus die Idee des Muscle Laser Run entstand. Wir teilten unser Projekt in kleinere Aufgaben auf und uns nach Kompetenzen und Interessen zu: Zwei Personen widmeten sich in den 30 Stunden dem Abgreifen der Biosignale. Eine Studentin telefonierte mit unzähligen Apotheken, Kliniken und Univer-

sitäten, um uns die benötigten Klebeelektroden zu beschaffen. Sie war erfolgreich und wir konnten wir uns mit der Motorsteuerung der Rolle beschäftigen. Drei Personen übernahmen die Konstruktion der Rolle, wobei eine vorwiegend den Lasercutter betreute. Ebenfalls drei Studierende beschäftigten sich mit der Lenkung des Lasers, während ein Kommilitone unsere 3D-Druckaufträge entgegennahm und realisierte. Mit der Zeit verschoben sich die Aufgaben, weil manche Unterprojekte schneller erledigt waren als andere. Dadurch vergrößerten sich einige Gruppen und andere lösten sich auf. Dies verlief sehr dynamisch und unproblematisch.

Dank der Organisatoren und Kooperationspartner*innen standen uns 3D-Drucker, Werkbänke, Lasercutter, jegliche Hardware und weitere denkbare Materialien zur Verfügung. In diesem Bastelparadies tobten wir uns während des 30-stündigen Wettbewerbs so sehr aus, dass einige aus unserer Gruppe komplett auf Schlaf verzichteten. Andere schliefen lediglich für kurze Zeit auf der Messe, um dringend benötigte Energie zu tanken. Nach Ablauf der Zeit wurden alle Projekte vor den rest-



Unser Team besteht aus Marie Böhm, Selina Häuser, Nicole Bätz, Marcel Treis, Tina Heßelmann, Maximilian Leven, Anne Bias, Sebastian Gerke, Jonas Strobel und Wjatscheslaw Liublin. Hier sieht man uns nach dem erfolgreichen Pitch unseres Prototyps.

lichen Teilnehmerinnen und Teilnehmern, vor zufälligem Publikum der Messe und einer Jury präsentiert. Dies war der Abschluss unseres ersten Makeathons, den wir überglücklich und vollkommen ausgepowert verließen.

Elektrorenner mit Solarantrieb

Auch wenn Photovoltaik (noch?) nicht für mobile Anwendungen im Großen geeignet ist, kann man ferngesteuert und im Maßstab 1:8 bei Sonnenschein mit dieser Energiequelle schon sehr flott unterwegs sein. Im oberösterreichischen Wels werden sogar jährlich Rennen ausgetragen – und wer will, ist kommendes Jahr mit dabei.

von Paul Srna

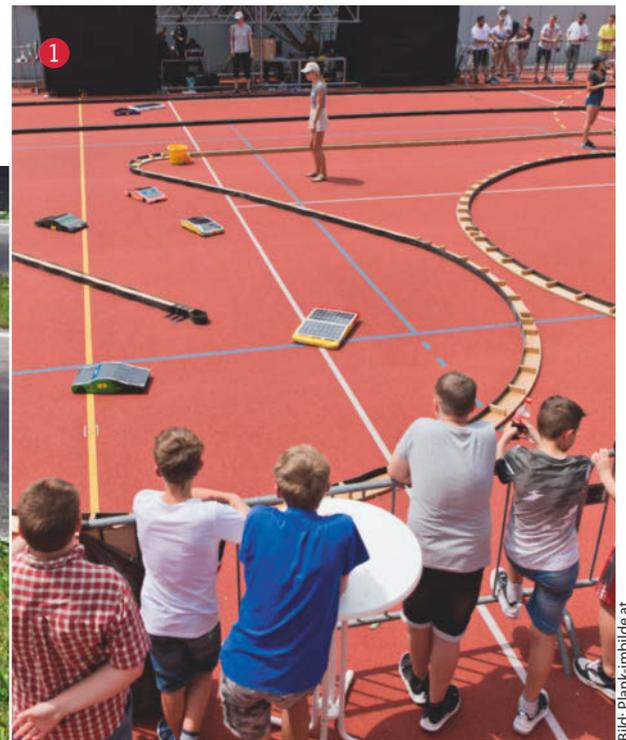
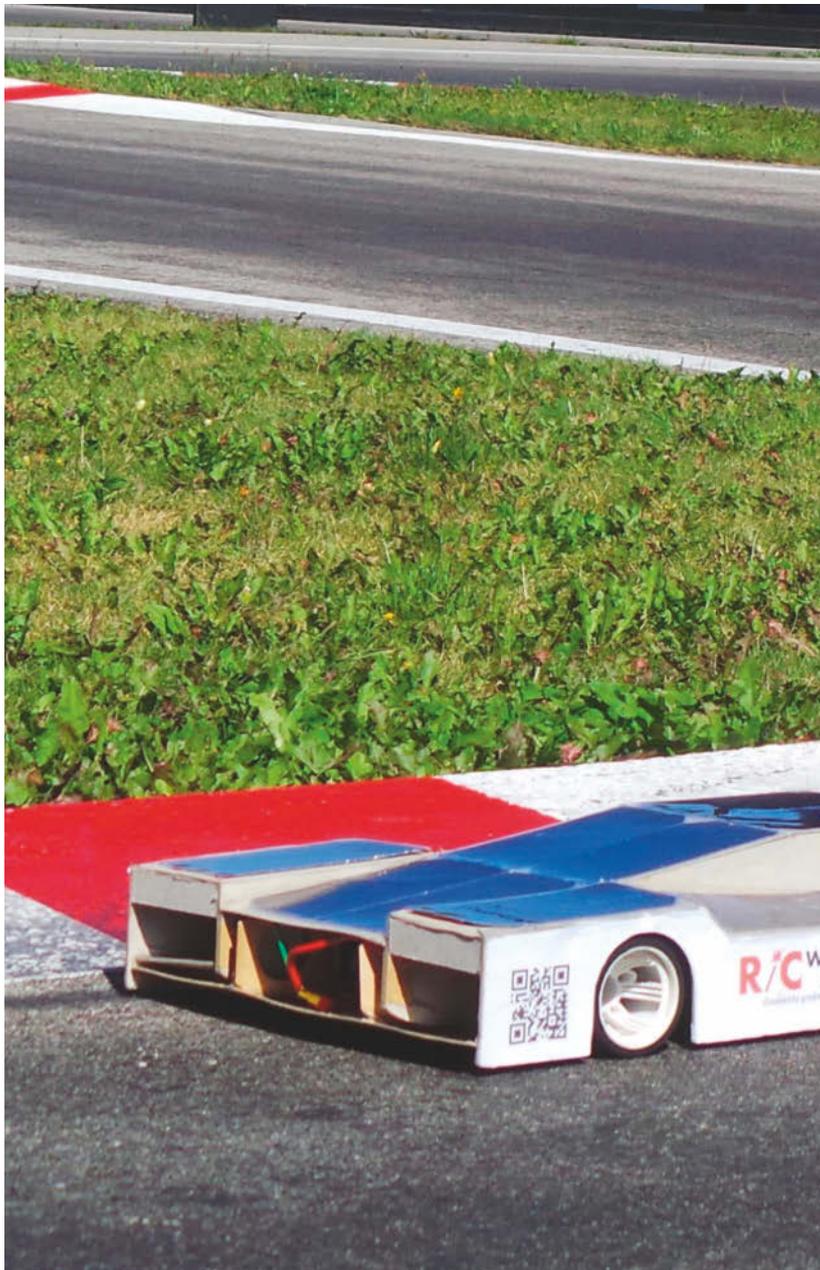


Bild: Plank-imbilde.at

Als Lösung für einen Teil des Klimaproblems ist die Elektromobilität in aller Munde. Eine Variante, die – weitgehend – ohne Ladestationen auskommt, weil sie ihre Energie aus Photovoltaik-Modulen bezieht, soll hier vorgestellt werden. Zugegeben, das funktioniert derzeit noch nicht für den Alltagsgebrauch und das Familienauto. Aber schon seit 1987 legen bei der *World Solar Challenge* mit Solarenergie betriebene Rennfahrzeuge eine Strecke von rund 3000km quer durch Australien zurück. Neben den reinen Rennfahrzeugen wurde dann 2013 auch eine Klasse für Fahrzeuge mit mehreren Passagieren ins Reglement aufgenommen.

Mehrere Kategorien gibt es auch bei der *RC-SolarCar Challenge*, einer ähnlichen Veranstaltung, allerdings für RC-Cars. Nein, nicht im australischen Outback. Die Fachhochschule Oberösterreich in Wels veranstaltet solche Modellautorenrennen seit 2012 am Tag der Sommersonnenwende auf dem Sportplatz des benachbarten Gymnasiums 1.

Die Herausforderung

Bei den ersten Rennen war die große Herausforderung, mit dem vorgegebenen Solarpanel ein Modellfahrzeug überhaupt zum Laufen zu bringen. In den folgenden Jahren wurde dann eine Zwischenspeicherung der Energie erlaubt und das Solarpanel ist seither nur mehr in der Größe vorgegeben. Dadurch können verschiedene Konzepte erprobt werden und die Modelle haben auch immer häufiger ein fahrzeugähnliches Aussehen und entwickeln sich weg von der einst dominierenden „fahrenden Platte“.

Die grundsätzliche Aufgabe ist aber geblieben: mit einer nominellen Leistung von maximal $45W_{Peak}$ (bei voller Sonneneinstrahlung) mit einem elektrisch betriebenen Modellfahrzeug in 20 Minuten möglichst viele Runden auf dem vorgegebenen Rundkurs 2 zu absolvieren.

Da wegen der Zählerlektronik mit Antenne über der Ziellinie und Transponder im Fahrzeug nur ganze Runden gezählt werden können, fährt man nach Ende der 20 Minuten noch die letzte Runde fertig; aus der Rundenzahl und der Überzeit ergibt sich dann die Rangfolge. Das genaue Reglement für die RC-SolarCar Challenge der FH Wels findet man auf der Website der Hochschule (siehe Link in der Kurzinfor).

Der Rennablauf

Der Ablauf des Rennens entspricht ziemlich genau dem eines Formel-1-Rennens mit Training, Qualifying und Rennen. Eingeteilt werden die Teilnehmer in zwei Kategorien – Schüler/Jugendliche unter 20 Jahre (kurz U20) und Hochschüler/Hobymodellbauer

Kurzinfor

» Photovoltaik-Modul als Energiequelle für ein RC-Car

» Zwischenspeicherung der Energie in Gold Caps

» Lasergeschnittenes Leichtbau-Chassis

Checkliste



Zeitaufwand:

zwei bis drei Wochenenden



Kosten:

etwa 250 Euro (ohne Fernsteuerung)



Elektronik:

Fernsteuerkomponenten und Energiespeicher verbinden, Solarzellen löten



Maschinen:

Lasercutter, Foliergerät, Lötstation, Heißklebepistole

» **8 Solarzellen** Sun Power monocristalline flexible solar cells 125mm x 125mm, 0,57V/3,3W, halbiert

» **4 Gold Caps** Samxon Doppelschicht-Kondensatoren 22F, 2,5V

» **Stützzakku** 2s LiPo 600mAh

» **Vorder- und Hinterachsteile** XRay-X10- oder CRC-Pan-Car-Teile

» **Reifen und Felgen** Contact 1/10-Pan-Car-Räder

Plattenmaterial

» **Flugzeugsperholz** 1,5mm dick, 1000mm x 300mm

» **Flite Test WR Foam Board oder Graupner Vector Board** 3mm dick, 71mm x 97mm (vor der Vorderachse) und 365 x 97 (Mittelteil)

» **Pappelsperholz** 4mm dick, 300mm x 600mm

» **Finnpappe oder Graupner** 2mm dick, 700mm x 1000mm

» **Kraftplex** 1,5mm dick, 272mm x 72mm, leicht gebogen

Alles zum Artikel im Web unter make-magazin.de/x1q3

Material

Antrieb und Elektronik

» **Fernsteuer-Sender** etwa Sanwa MT-4

» **Empfänger** etwa Sanwa RX-461

» **Lenkservo** etwa Hitec HS-82MG

» **Brushless-Motor** Robitronic Platinum 21.5T oder LRP X20 21.5T

» **Brushless-Regler** Robitronic Cube BL oder LRP SPX brushless



Sicherheitshinweise:

- LiPo-Akkus nicht über 4,2V pro Zelle laden und nicht unter 3,4 V pro Zelle entladen.
- Ein kleiner LiPo-Wächter mit Unterspannungssignal hilft dabei – falls der elektronische Regler keinen Unterspannungsschutz eingebaut hat.
- Akkus **nie** unbeaufsichtigt laden (auch nicht mit dem Solarpanel) und besser in einem *Safety Bag* transportieren und lagern.

(kurz HH). Dem freien Training, bei dem die Fahrzeuge auf ihre Funktion geprüft und letzte Abstimmungsarbeiten durchgeführt werden können, folgt das 15 Minuten dauernde Qualifying. Dabei wird die Startreihenfolge aufgrund der schnellsten gefahrenen Runde ermittelt. Die Hauptrennen dauern dann jeweils 20 Minuten plus Überzeit und werden in Gruppen zu je zehn Startern gefahren. Für die drei schnellsten in jedem dieser Rennen gibt es schöne Pokale und sie dürfen im abschließenden Rennen um den Titel des *SolarCar Champion* antreten, das ebenfalls 20 Minuten dauert.

Das technische Reglement

Im Wesentlichen wird die minimale (250mm x 500mm) und maximale Größe (450mm x 750mm) des Solarpanels beziehungsweise des Fahrzeugs vorgegeben und – nach Kategorien getrennt – eine maximale Speicherkapazität festgelegt. Dabei bekommt die U20 mit 6Wh etwas mehr zugesprochen als die Studenten/Modellbauer, die mit maximal 5Wh auskommen müssen. Für Schulklassen und interessierte Gruppen wird auch jedes Jahr ein Bausatz angeboten, der außer der Fernsteuerung alle notwendigen Materialien

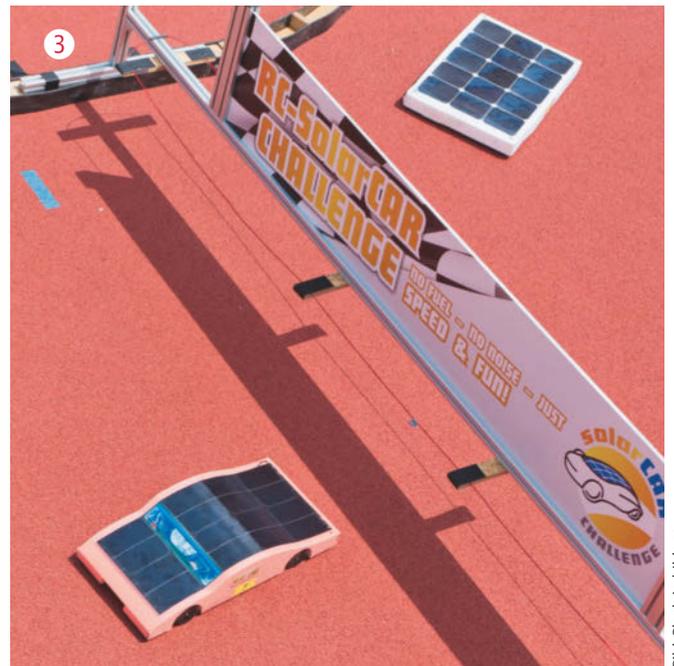
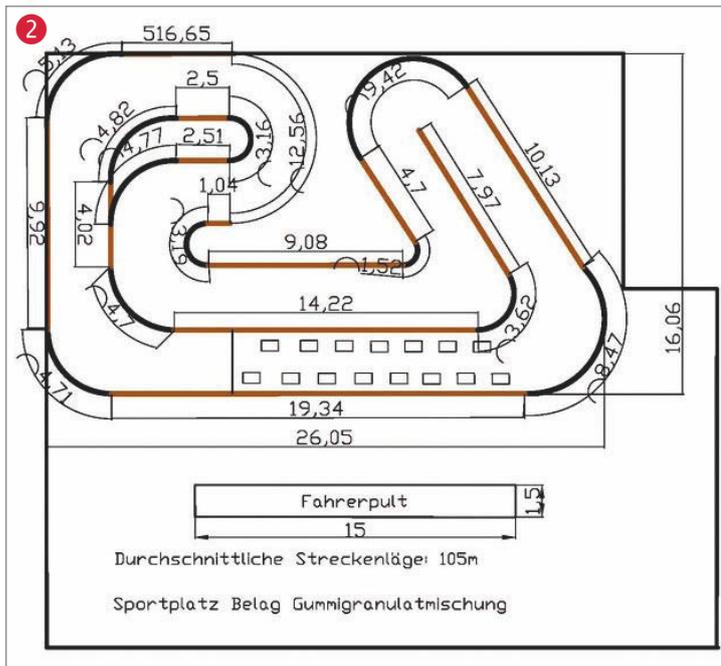


Bild: Plank-imbilde.at

enthält, die zum Bau eines funktionsfähigen Modells notwendig sind 3.

Wieder am Start

Nachdem wir bereits seit der ersten RC-Solar-Car Challenge an diesem Wettbewerb teilnehmen, sollte es auch 2019 wieder ein Modell werden, das ein paar neue Ideen enthält. „Wir“ sind ein Dreierteam aus Modellbauern und Modellautofahrern: Thomas Landauf, unser Pilot, Martin Bayer, der Chefkonstrukteur, der sein Wissen zur Fahrwerkstechnik und seine gut ausgerüstete Werkstatt beisteuert, und ich, Paul Srna, als Ideenlieferant und treibende Kraft.

Den Schritt weg vom Schuhkarton-Design hatten wir schon vor Jahren eingeleitet und

ein Brushless-System aus Motor und Regler als Antriebstechnologie war uns als Modellbauern auch seit längerem ebenso vertraut wie die Speicherung der elektrischen Energie in LiPo-Akkus. Unser Ehrgeiz lag also in der Saison 2019 in der weiteren Steigerung der Effizienz.

Erste Überlegungen

Im Brainstorming wurde beschlossen, zwei verschiedene Konzepte auszuprobieren. Eine Version unseres Wagens sollte mit dem kleinen Solarpanel (250mm x 500mm und 25W_p Leistung) auskommen, die andere Version nutzte die maximale Panelfläche (450mm x 750mm und 45W_p Leistung) mit einem entsprechend großen Modell. Ob maximale

Photovoltaik-(PV-)Leistung und mehr Gewicht oder weniger PV-Leistung in einem leichteren, wendigeren Modell besser funktionieren würde, sollte im praktischen Versuch geklärt werden.

Nachdem wir im Vorjahr mit unseren 3D-gedruckten Chassis die ersten drei Plätze in der Kategorie HH und auch den Gesamtsieg erringen konnten, wollten wir das bewährte Fahrwerkskonzept beibehalten, aber Gewicht einsparen. Die Wahl fiel schließlich auf Flugzeugspertholz und Leichtstoffplatten mit den bewährten Achs- und Antriebsteilen von 1/10 Pan Cars, einer Klasse von Modellrennwagen für garantiert flache Rennstrecken. Optisch sollte – wie bereits 2018 erfolgreich umgesetzt – eine gewisse Ähnlichkeit mit echten Rennfahrzeugen der Klasse der *Le-Mans-Prototypen* bestehen, die sich von Formel-1-Fahrzeugen am auffälligsten durch die geschlossenen Radkästen abheben 4.

Konstruktion des Chassis

Beim Antrieb wurde auf die bewährte Kombination eines 21.5T-Brushless-Motors mit einem elektronischen Regler gesetzt, wie er auch im Modellrennsport bei 1/10-Formelfahrzeugen verwendet wird. Vorderachsaufhängung, Reifen und Felgen und die Hinterachse mit Kugeldifferential stammen ebenfalls aus dem Bereich dieser 1/10-Elektrorenner. Die Länge und Breite des Modells wurden an die Anzahl der Solarmodule und die geplante Karosserieform angepasst.

So entstand vorerst am PC das Modell eines Wannenchassis mit Seitenteilen 5 und Querspannten 6 aus Flugzeugspertholz, einer

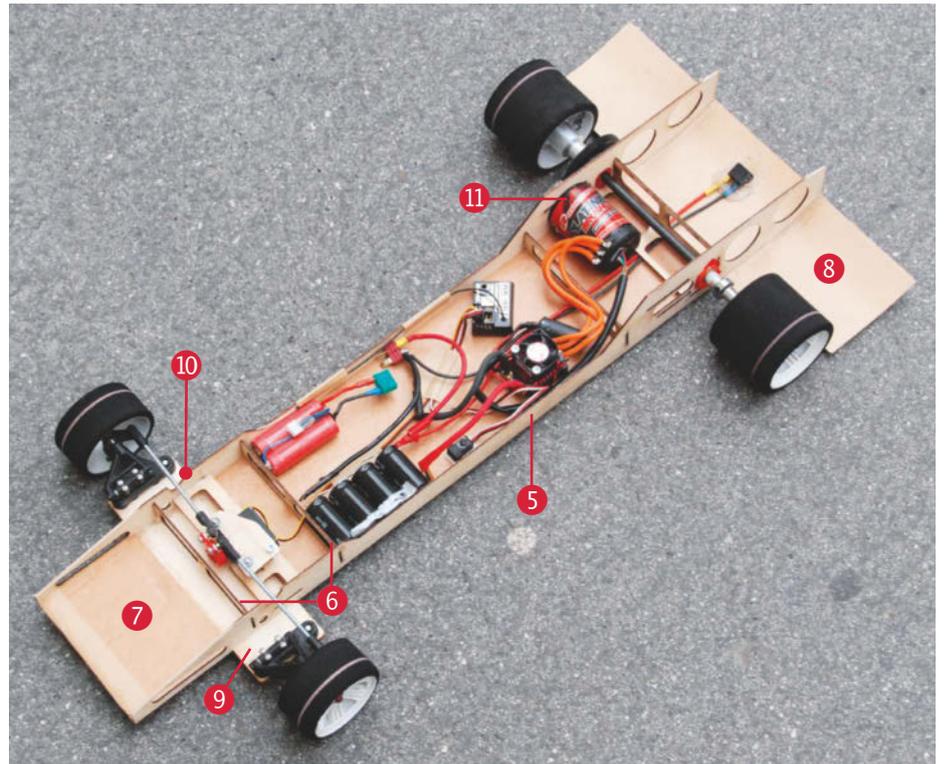


Bodenplatte **7** aus *Flite Test Board*, einem speziellen Schaumstoff aus dem Flugmodellbau, und einem Diffusor **8** aus Kraftplex, der durch seine Sogwirkung für eine bessere Straßenlage sorgt. Die Vorderachse sollte auf einem Träger aus Pappelsperholz **9** montiert werden, die Hinterachse mit den verstellbaren Achsträgern in Ausschnitten der Seitenteile. Weitere Ausschnitte waren für die Spurstangen **10**, für die Befestigung des Motors **11** und als Gewichtseinsparung vorzusehen. Zusätzlich war natürlich die Befestigung der Karosserie zu bedenken und mit den Seitenteilen abzugleichen. Entstehen sollte das Chassis auf dem Lasercutter.

Solarpanel und Karosserie

Der größte Unterschied der beiden Modelle lag, bedingt durch die verschiedenen PV-Module – auch im wahren Sinn des Wortes –, in den Abmessungen. In diesem Artikel werde ich mich auf das kleinere Modell beschränken, weil die Anfertigung der lasergeschnittenen Teile dafür weniger Anforderungen an die Größe des Lasercutters stellt.

Auch die Karosserie sollte mit dem Lasercutter geschnitten werden. Neben der Anmutung eines Fahrzeugs für die 24 Stunden von Le Mans war eine zweite Anforderung, dass der Wagen eine ausreichende Stabilität für den „harten Rennbetrieb“ bietet. Abgesehen vom unterschiedlichen Fahrkönnen der Teilnehmer und den daraus resultierenden Crashes sollten auch selbst verschuldete Ausrutscher in die harten Banden ohne bleibende Schäden zu überstehen sein. Wie gesagt, es war nicht unsere erste Teilnahme an der RC-Solarcar Challenge ...



Elektrische Komponenten

$25W_{\text{Peak}}$ aus dem Solarpanel und 5Wh aus dem Speicherakku sind eine mehr als überschaubare Energiemenge für 20 Minuten Fahrzeit, wenn man weiß, dass bei RC-Rennen mit Elektromodellen Akkus mit rund 60Wh für 5 Minuten Renndauer eingesetzt werden. Eine gut überlegte Auswahl im Sinne von Effizienz, Leistung und Gewicht ist also gefordert. Die Kombination aus Brush-

less-Motor und -Regler aus einem 1/10-Formel-1-Modell hatte sich bewährt, daher wurde als Speicher ein 2s-LiPo-Akku mit 7,4V und einer Kapazität von 600mAh gewählt. Rechnerisch ergibt sich damit eine Kapazität von knapp 4,5Wh. Zusätzlich sollten noch Doppelschichtkondensatoren mit einer Kapazität von rund 250Ws eingebaut werden (siehe Kasten), damit die Stromspitzen beim Beschleunigen besser abgefangen werden und der Akku weniger belastet wird.

Energie, Ladung und Kapazität

Akkus und Superkondensatoren liefern gleichermaßen Energie für den Antrieb des Solarrenners und beide zusammen dürfen laut Reglement des Rennens nicht mehr als 5 Wattstunden (Wh) Energie speichern können. Die maximale Ladung von Akkus wird allerdings in Amperestunden (Ah), die Kapazität von Kondensatoren in Farad angegeben.

Für die Umrechnung von Amperestunden in Wattstunden muss man die Spannung des Akkus berücksichtigen, denn $1\text{Wh} = 1\text{A} \cdot \text{V} \cdot \text{h}$. Der hier verwendete 2s-LiPo-Akku hat eine Spannung von 7,4V und mögliche Ladung von 600mAh oder 0,6Ah. Daraus ergibt sich eine gespeicherte Energie von $0,6\text{A} \cdot \text{h} \cdot 7,4\text{V} = 4,44\text{Wh}$.

Die Formel für den Zusammenhang zwischen Kondensatorkapazität und der gespeicherten Energie lautet:

$$W = 1/2 \cdot C \cdot U^2$$

Dabei steht W für die Energie in Wattsekunden (Ws), C für die Kondensatorkapazität in Farad und U für die Spannung in Volt. Im konkreten Fall hat die Kondensatorbank eine Kapazität von 5,5 Farad bei einer maximalen Spannung von 9,2 Volt. Das ergibt sich daraus, dass vier Kondensatoren mit je 22 Farad und 2,3 Volt in Reihe geschaltet werden, was die Gesamtkapazität gegenüber dem einzelnen Kondensator viertelt, die maximale Spannung aber vervierfacht.

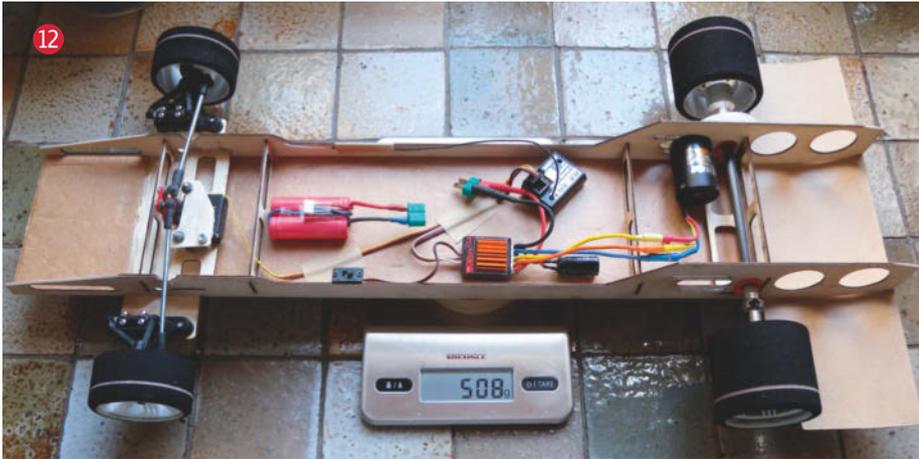
Daraus ergibt sich eine maximal in den Kondensatoren gespeicherte Energie von:

$$W = 1/2 \cdot 5,5\text{F} \cdot (9,2\text{V})^2 = 232,76\text{Ws}$$

Das entspricht 0,065Wh und damit bleibt die in Kondensatorbank und Stützakku speicherbare Energie deutlich unter der im Reglement festgeschriebenen Höchstgrenze von 5Wh.

Aber wie kommt man von Farad auf Wattsekunden? Farad ist definiert als Ampere-sekunde durch Volt (A·s/V), was für die Einheiten der oben genannten Formel ergibt:

$$W_s = \text{F} \cdot \text{V}^2 = \text{A} \cdot \text{s} / \text{V} \cdot \text{V}^2 = \text{A} \cdot \text{s} \cdot \text{V}. \text{Kommt also alles hin.}$$



Um diese Speicherbank aufladen zu können, müssen die Solarzellen des PV-Moduls mindestens die Ladespannung des Akkus liefern. Daraus ergibt sich deren notwendige Anzahl, weil diese in Reihe geschaltet werden. Konkret haben wir daher 16 Zellen mit einer Spannung von je 0,57V vorgesehen. Aus Platzgründen sollten für das kleinere Modell deshalb 8 Solarzellen mit den Abmessungen 125mm × 125mm halbiert werden – sonst passen sie schlecht auf die Flächen der Karosserie.

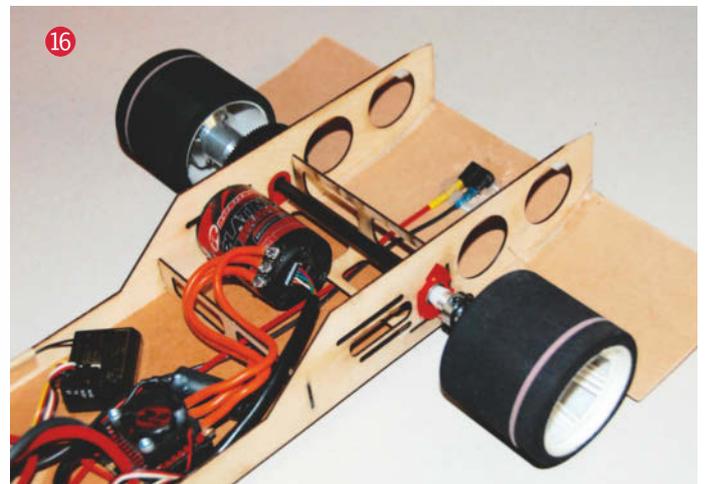
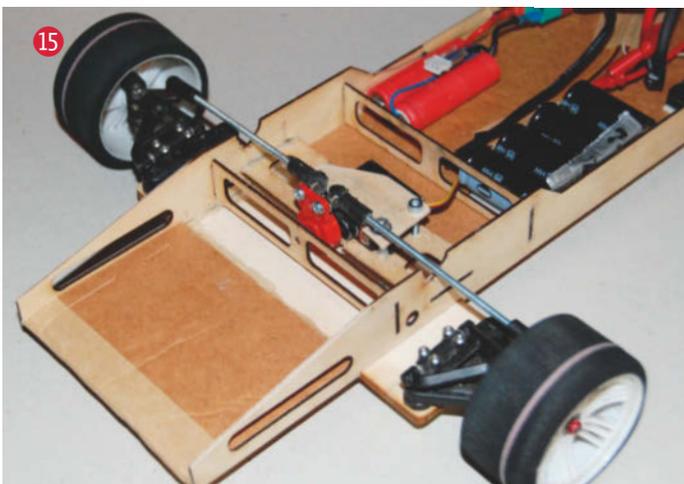
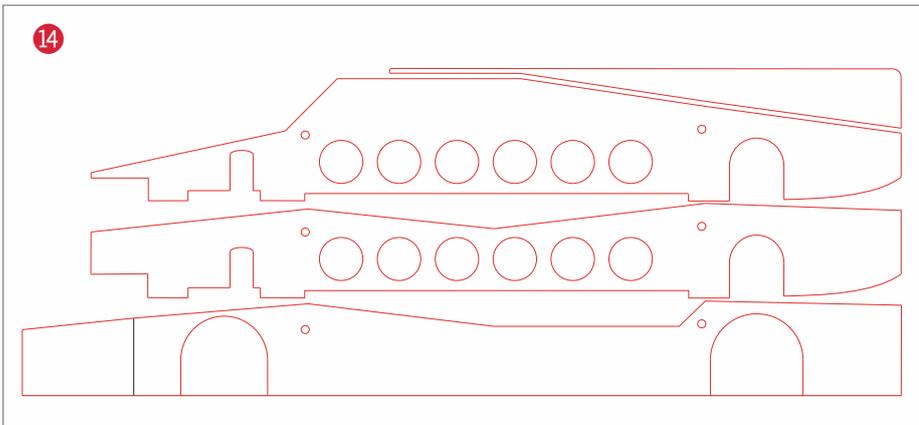
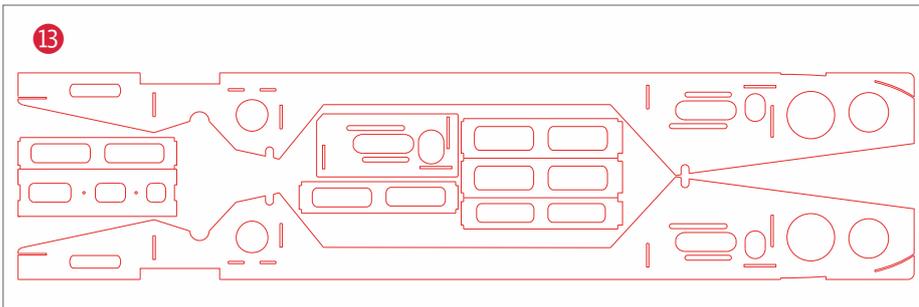
Erste Probleme

Damit standen wir vor dem ersten größeren Problem! Wie teilt man Solarzellen in zwei Teile, ohne sie unbrauchbar zu machen und ohne große Verluste? Die erste Idee war natürlich Laserschneiden. Leider ist das mit den uns zur Verfügung stehenden Lasercuttern (im HappyLab Wien oder Salzburg) nicht möglich. Nächste Idee: mit Wasserstrahl schneiden! Allerdings hätte die Firma, die das mit einem Präzisionsschneidergerät machen könnte, deutlich über unserem Budget liegende Umrüstkosten verlangen müssen – also auch keine Lösung.

Schließlich haben wir uns zu der eigentlich brutalen mechanischen Methode des Ritzens mit einem Hartmetallstichel und anschließenden Abknicken über einer harten Kante entschlossen (Vorgangsweise wie bei Fliesen). Das hat mit wenig Verlusten (Bruch durch ungeschicktes Hantieren) auch ganz gut funktioniert. In diesem Zusammenhang sei erwähnt, dass die Solarzellen sehr empfindlich sind und auch beim Verlöten entsprechende Vorsicht erfordern. Dazu später noch ein paar Tipps.

Abmessungen und Gewichte

Aus der Anzahl und Anordnung der Module hat sich ergeben, dass die Karosserie aus



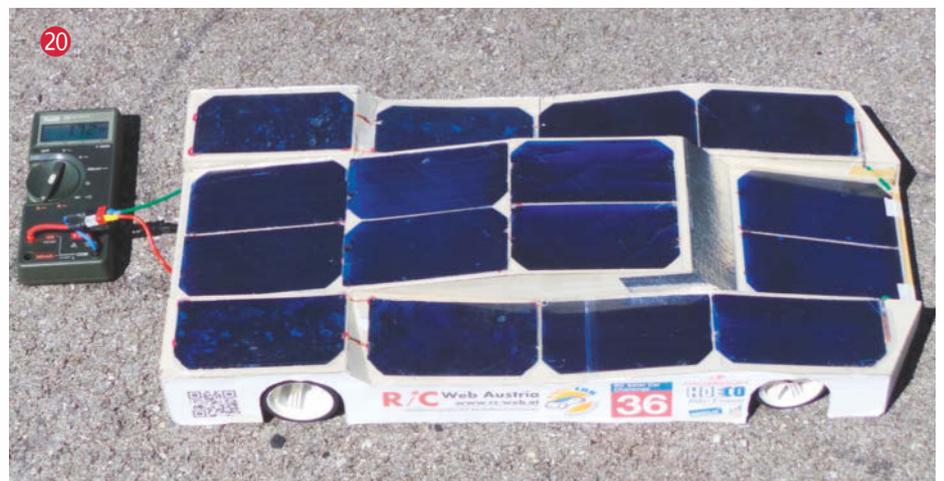
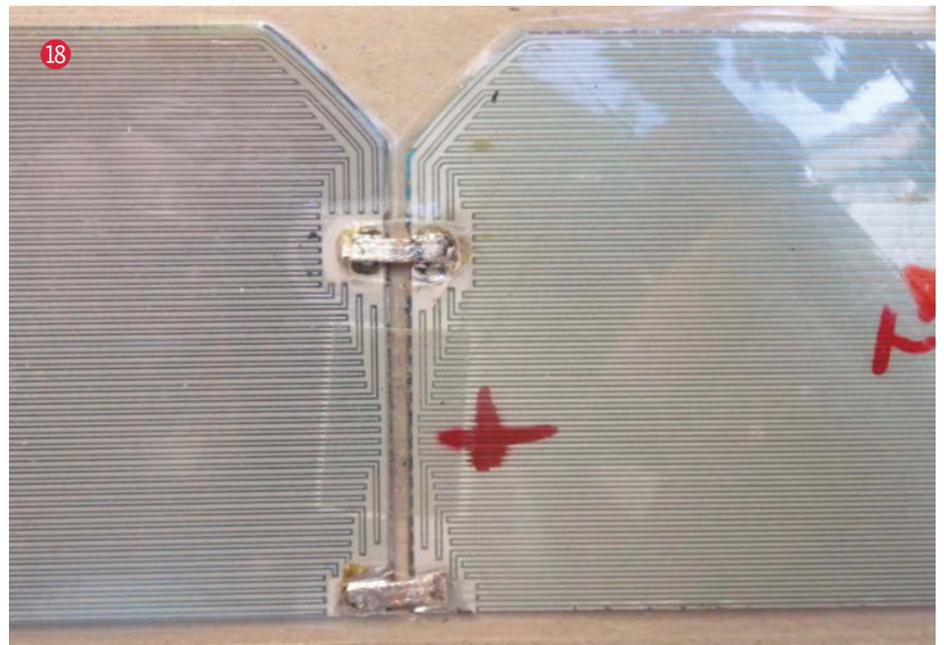
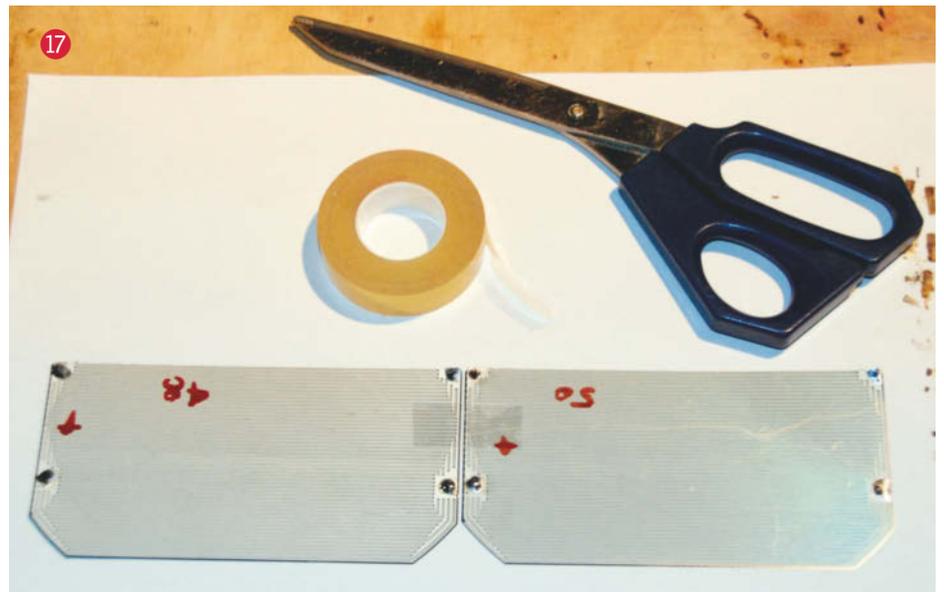
einem 130mm breiten Mittelteil mit zwei Seitenkästen von je 70mm Breite bestehen und die Gesamtlänge 560mm betragen sollte. Auf dieser Fläche konnten die Solarzellen weitgehend waagrecht verteilt und damit die fast optimale Ausbeute erreicht werden. Theoretisch sollten sich die Verluste durch die verschiedenen Einstrahlwinkel bei Richtungsänderung im Bereich von etwa 10 Prozent bewegen.

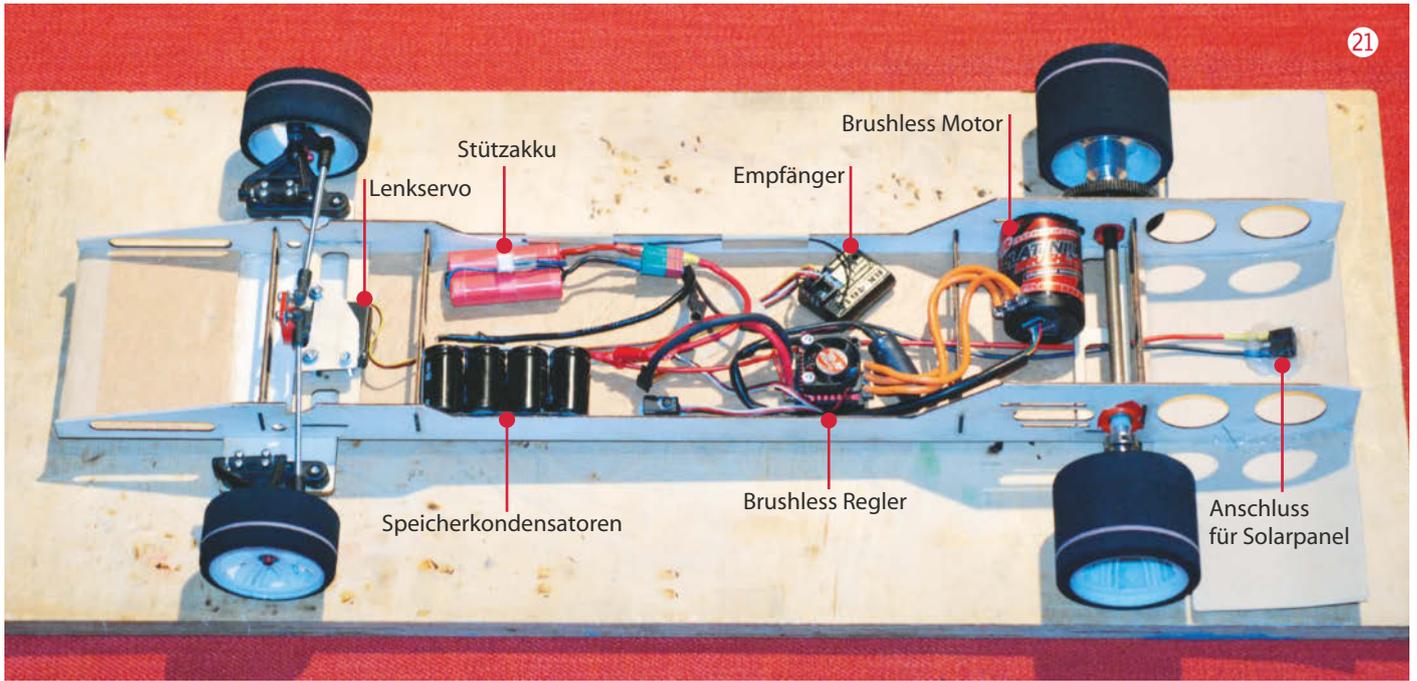
Die Chassisbreite von 100mm und die Achsbreite gemessen von der Außenkante der Reifen von 268mm hat sich aus den verwendeten Achsteilen ergeben. Der Radstand wurde mit 360mm festgelegt. Die Vorderreifen haben die Dimension 57mm x 30mm, die Hinterreifen 60mm x 50mm. Das angestrebte Gesamtgewicht von 1000 Gramm sollte mit der Wahl der Materialien – Flugzeugsperrholz 1,5mm für die Seitenteile und Querstreben des Chassis, die Bodenplatte aus Foam Board, die Karosserie aus 2mm-Finnpappe und dazu ein Solarpanel aus 16 halben Zellen – machbar sein ¹².

Zusammenbau des Chassis

Die Chassisteile ¹³ aus Flugzeugsperrholz, der Träger für die Vorderachse aus 4mm starkem Pappensperrholz, der Diffusor aus Kraftplex und die Teile für die Karosserie ¹⁴ wurden mit einem Trotec Laser Speedy 100 (Arbeitsfläche 610mm x 305mm) im HappyLab Wien geschnitten, die Bodenplatte aus dem Foam Board einfach mit Stahllineal und Stanleymesser.

Die lasergeschnittenen Chassisteile wurden mit einem Mikrofasertuch von Schmauchspuren gereinigt und dann mit Weißleim verklebt, die Karosserieteile wurden von innen mit Heißkleber verbunden. Die Trägerplatte für die Vorderachse ¹⁵ kann ebenfalls mit Heißkleber am Chassis befestigt werden. Für die Zeit der Aushärtung sollte das Chassis auf einer möglichst planen Platte





(Glasplatte, dickes Sperrholz oder Ähnliches) fixiert werden, damit sich kein Verzug einstellt, der später zu unterschiedlichem Fahrverhalten in Links- und Rechtskurven führen kann. Die Vorderachsträger werden mit Beilagscheiben auf die passende Bodenfreiheit eingestellt und mit Senkkopfschrauben montiert.

Die Hinterachse ¹⁶ wird mit den üblichen Lagerschalen von 1/10 Pan Cars im Chassis befestigt. Sie bestehen im Kern aus einem Exzenter, mit dem man die Bodenfreiheit des Wagens an den Reifendurchmesser anpassen und genauer einstellen

kann. Bitte auch darauf achten, dass die Achsen etwas Spiel haben und nicht in den Lagern klemmen.

Das Solarpanel

Wer nicht zu einem fertigen PV-Modul (etwa aus dem Bausatz der FH Wels) greifen will, sondern wie wir sein Solarpanel selbst zusammenbauen möchte, sollte sich an ein paar Grundregeln halten. Ganz wichtig: Die Solarzellen sind sehr dünne Siliziumplättchen und ähnlich zerbrechlich wie Glas. Daraus ergibt sich, dass man die Zellen – viel-

leicht abweichend von der ersten Intuition – zum Ritzen oder Schneiden und auch zum Verlöten nicht auf eine weiche, sondern auf eine harte Unterlage legen sollte. Denn schon ein unvorsichtiger Druck auf eine Zelle auf einer zu weichen Unterlage kann zum Bruch der Zelle führen und sie unbrauchbar machen. Da Fingerabdrücke zumindest un schön sind, ist auch die Verwendung von (Baumwoll-)Handschuhen angeraten.

Wir haben Solarzellen der Größe 125mm x 125mm von SunPower verwendet, die gängigere Größe ist allerdings 156mm x 156mm. Nach dem Ritzen und Knicken sollten die Solarzellen zum besseren Schutz gegen Beschädigung mit einem Laminiergerät foliert werden. Dazu haben wir die einzelnen Zellen zuerst mit Klebeband auf der Rückseite auf den passenden Abstand fixiert ¹⁷. Anschließend wurden auf der Rückseite der Folientasche die Lötunkte angezeichnet und ausgeschnitten.

Vor dem Folieren sollte man sicherheits halber mit einem Voltmeter kontrollieren, ob alle Zellen die Teilung gut überstanden haben. Solarzellen geben zwar bei künstlichem Licht wesentlich weniger Leistung ab als bei Sonnenlicht, defekte Zellen liefern aber gar nichts und sind somit leicht zu identifizieren. Nach dem Folieren sind die Zellen dann deutlich robuster und überstehen sogar leichtes Biegen ohne Schaden.

Es kann also ans Verlöten der Zellen gehen. Mit entsprechender Vorsicht und möglichst wenig Zugabe von Lötzinn werden die 16 Zellen in Serie geschaltet. Dazu werden zuerst bei den vier Zellen eines Streifens jeweils + mit – verbunden ¹⁸. Draht mit



Bild: FH Wels

1mm² reicht für die relativ geringen Ströme (bei unserem Panel etwa 2A) locker aus. Dann werden die seitlichen Paneele mit dem mittleren verbunden (auf ausreichende Drahtlängen zur Anpassung an die Karosserie achten) und die Enden der seitlichen Paneele mit dem Anschlusskabel versehen. Eine abschließende Kontrolle sollte etwas mehr als 8V Leerlaufspannung ergeben ¹⁹. Unter senkrecht auftreffenden Sonnenstrahlen müsste der Kurzschlussstrom (bei wolkenlosem Himmel) rund 2A betragen ²⁰. Damit wäre die wichtigste Hürde zum erfolgreichen Solarbetrieb geschafft.

Einbau der Elektronik

Der Rest sollte für Maker mit etwas Elektronik- oder Modellbauerfahrung kein Problem darstellen. Der Brushless-Motor wird am rechten Seitenteil mit der Verstärkungsplatte oder einer Aluscheibe als Kühlkörper befestigt und das Spiel zwischen Ritzel und Hauptzahnrad eingestellt (im Zweifelsfall hilft ein eingeklemmter Papierstreifen).

Der elektronische Regler wird mit dem Motor verbunden und mit doppelseitigem Klebeband auf die Chassisplatte geklebt.

Stützakku, Speicherkondensatoren und Empfänger folgen ebenso. Der Lenkservo wird mit einer Klemmplatte hinter der Vorderachse so befestigt, dass der Servo Saver mittig sitzt – das ist ein spezielles Steuerhorn, das Stöße abfängt, um den Servo bei Unfällen vor Schäden zu schützen. Spurstangen aus einer 3mm-Gewindestange mit Kugelköpfen vom RC-Car bilden die Verbindung zu den Achsschenkeln.

Verschaltung

Die Verbindung der einzelnen Komponenten ²¹ ist einfach und logisch: Das Solarpanel wird polrichtig mit dem Akku und den in Serie geschalteten Kondensatoren verbunden. Eine weitere Steckverbindung wird für den Anschluss des Reglers benötigt – im Ruhezustand können dann Akku und Regler vom Solarpanel getrennt und so eine Tiefentladung des Akkus verhindert werden. Das Lenkservo wird mit dem Empfänger verbunden. Nach dem Binden des Empfängers mit dem Sender sollten alle Einstellungen noch einmal überprüft werden – mit einem vorge-ladenen Akku geht das auch ohne Sonne.



Start frei!

Bleibt zum Abschluss nur der Wunsch: Gute Fahrt bei Sonnenschein und wolkenlosem Himmel! Wir haben uns beim Rennen 2019 dann doch für die größere Solarfläche und damit fast die doppelte Leistung entschieden, unser großes Modell eingesetzt und damit den 3. Platz unter mehr als 30 Teams erreicht ²². Und inzwischen ist auch eine neue Karosserie aus Hartschaum entstanden ²³, nach gelaserten Schablonen mit Heißdraht geschnitten. Sehen wir uns bei der nächsten RC-SolarCar Challenge in Wels am 17. Juni 2020? —pek

Immer gut für neue Ideen.

Sparen Sie 10% im Abo und sammeln wertvolles Know-how:

- 6 Ausgaben kompaktes Profiwissen für nur 55,80 €
- Workshops und Tutorials
- Tests und Vergleiche aktueller Geräte
- Sparvorteile mit Gutscheinen und Sonderaktionen
- Bequeme Zustellung direkt nach Hause
- Inklusive HD-Actioncam



Ihr Geschenk

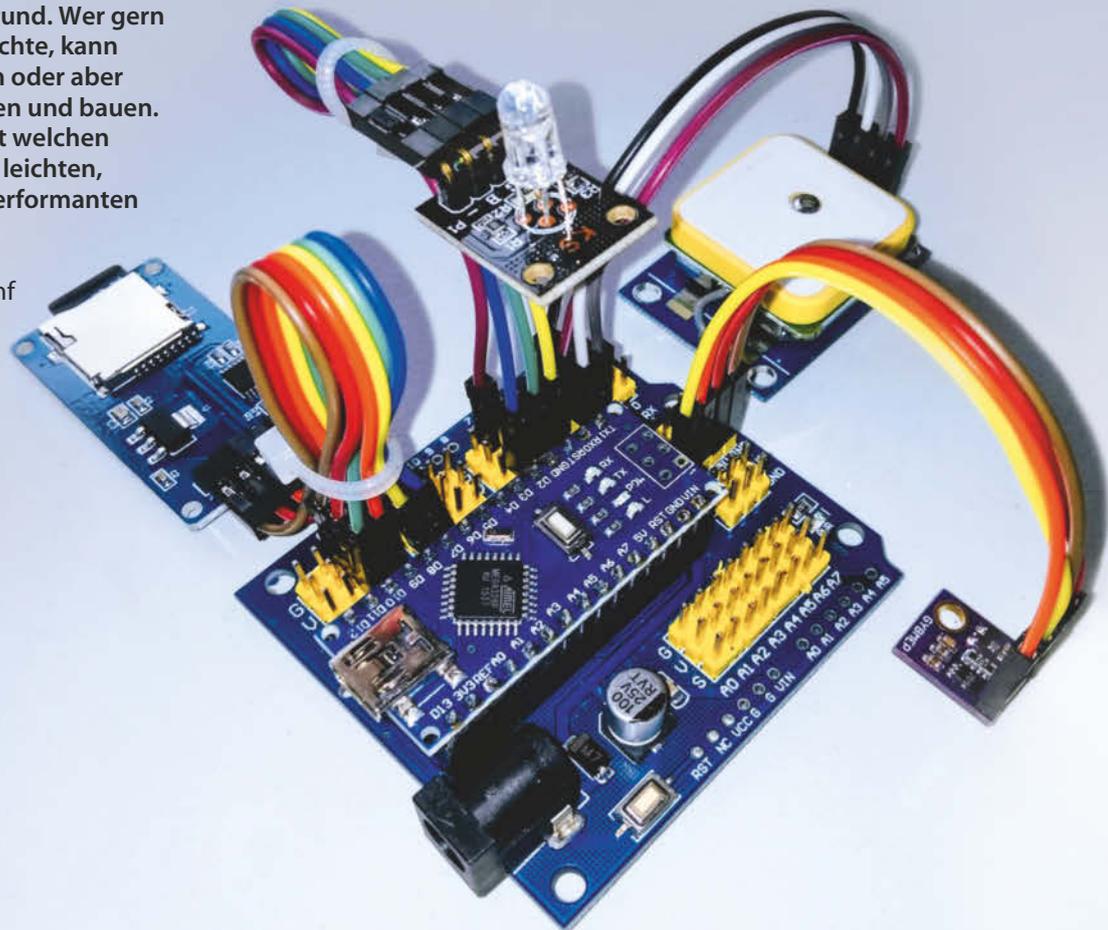
Jetzt bestellen:

ct-foto.de/abo

Datenlogger für den Stratosphärenflug

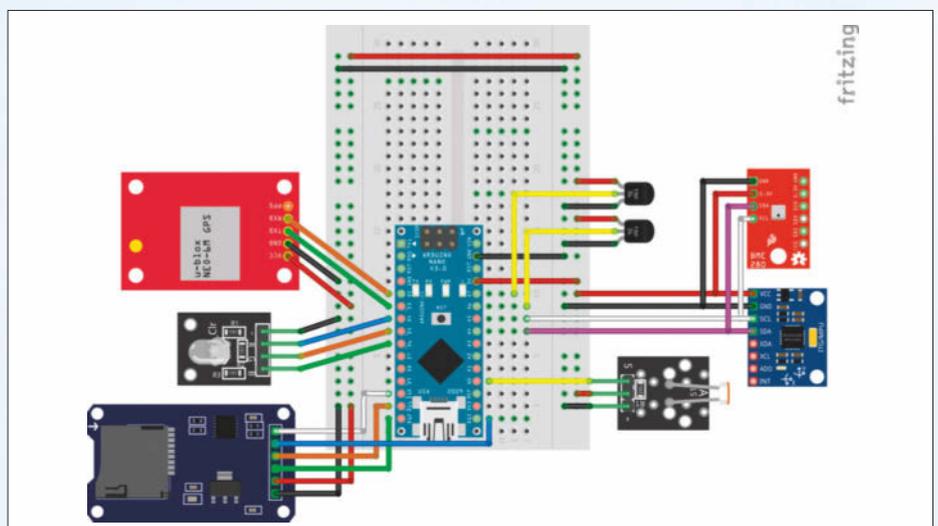
Einen Stratosphärenballon aufsteigen zu lassen ist ein beliebtes DIY-Projekt. Dabei stehen meist die spektakulären Videos im Vordergrund. Wer gern zusätzlich Messdaten erfassen möchte, kann natürlich ein Fertigprodukt kaufen oder aber selbst einen Datenlogger entwerfen und bauen. Im Folgenden beschreiben wir, mit welchen Komponenten man einen kleinen, leichten, kostengünstigen, robusten und performanten Logger selbst aufbauen kann.

von Simon Ilgenfritz und Tobias Lohf



Die Basis des Loggers bildet ein Sensor Shield, auf das man einen Arduino Nano V3 aufsteckt. Für die Sensorverdrahtung reichen Jumperkabel aus – kein Platinenätzen, wenig Löten, maximale Flexibilität. Fällt ein Sensor aus, so kann man ihn einfach entfernen und ersetzen. Den Logger kann man außerdem bei Bedarf um weitere Sensoren ergänzen. Der Minimalaufbau eines Stratosphären-Loggers umfasst ein SD-Modul, ein GPS-Modul, eine Statusleuchte und einen Umweltsensor.

Beispiel-Schaltplan in Breadboard-Darstellung mit zusätzlichem Beleuchtungssensor, Gyroskop, Temperaturfühlern und IR-Sensor (erstellt mit fritzing)



Kurzinfo

- » Sensoren für Stratosphärenflug mit Arduino Nano V3 steuern
- » Codingbeispiele für GPS bei Höhenanwendungen

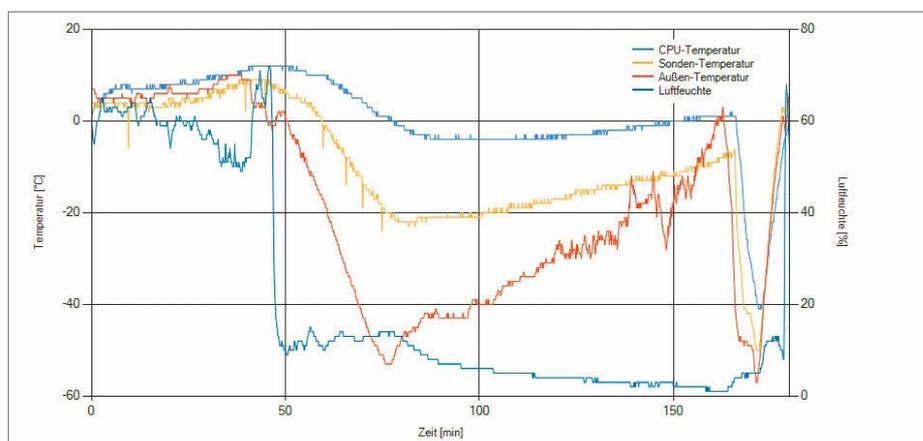
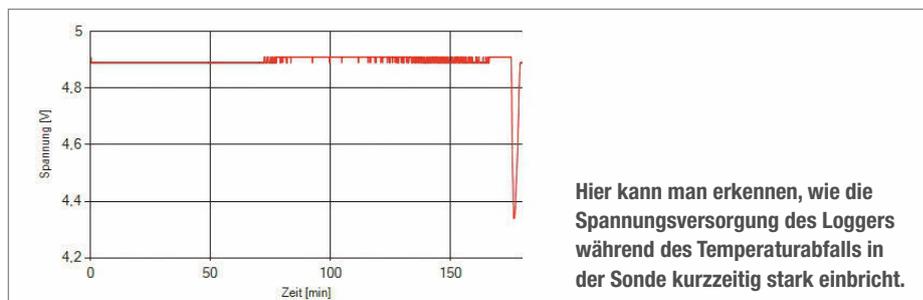
Alles zum Artikel
im Web unter
make-magazin.de/xrzd

System- überwachung

Das System ist in der Höhe extremen Bedingungen ausgesetzt: Temperaturen bis -60°C , vollkommen trockene Luft, heftige Winde und starke elektromagnetische und kosmische Strahlung (v. a. IR-, UV- und Gammastrahlung). Hinzu kommen der freie Fall nach dem Platzen des Ballons mit rund 200km/h und der Aufschlag auf den Boden mit rund 30km/h .

Selbst bei gut isolierten Sonden ist die Belastung für die Elektronik extrem hoch. Da macht es Sinn, das System selbst zu überwachen, um nach dem Flug anhand der Aufzeichnungen auswerten zu können, ob alles nach Plan verlaufen ist. Der Nano bietet hierfür die Möglichkeit, Systemzeit, Systemspannung und CPU-Temperatur auszulesen. Die Funktion von Isolierhülle, Logger und Batterie kann man über die oben genannten Parameter gut bewerten.

Weiterführende Informationen zum Auslesen der internen Spannung findet man im Arduino-Forum – einen Thread zu diesem Thema haben wir in die Links zum Artikel aufgenommen.



Beispielcode zum Auslesen der CPU-Temperatur

```
float Systemtemperatur = 0;
//Zyklischer Aufruf der Funktion in der Loop
void Temperatur_Ueberwachung()
{
  unsigned int wADC;
  double t;
  //Einstellen der internen Referenzspannung und des Multiplexers
  ADMUX = (_BV(REFS1) | _BV(REFS0) | _BV(MUX3));
  //Freigabe AD-Wandler
  ADCSRA |= _BV(ADEN);
  //Warten, bis sich die Spannung stabilisiert hat
  delay(20);
  //Starten des AD-Wandlers
  ADCSRA |= _BV(ADSC);
  //Warten, bis der AD-Wandler die Konvertierung abgeschlossen hat
  while (bit_is_set(ADCSRA,ADSC));
  //Lesen des Registers "ADCW" (ADCL und ADCH werden berücksichtigt)
  wADC = ADCW;
  //Umrechnen der Temperatur mit Default-Werten
  //Diese können im Rahmen einer Kalibrierung individuell angepasst werden
  Systemtemperatur = (wADC - 324.31) / 1.22;
}
```

Stromquelle

Um böse Überraschungen zu vermeiden, sollte man eine bis -40°C kältefesteste Batterie verwenden. Hierfür eignen sich zum Beispiel die 9V-Blockbatterie Varta Lithium oder die Energizer Ultimate Lithium.

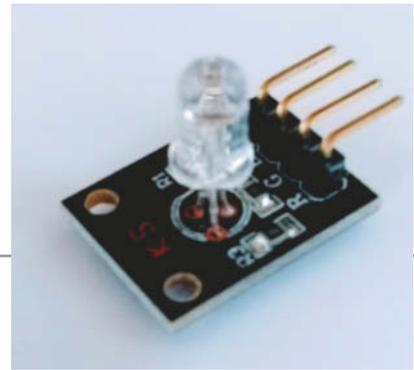
Statusleuchte

Es ist nicht besonders komfortabel, beim Balloonstart einen PC zur Überprüfung des Loggers mitschleppen zu müssen. Um das zu vermeiden, kann man als Minimallösung für die Funktionsprüfung eine Statusleuchte einbauen. Wir empfehlen hierfür die RGB-LED Keyes KY-016.

Ob man sich nun für unsere Empfehlung oder ein anderes Modell entscheidet: Eine RGB-LED sollte man in jedem Fall einsetzen. So kann man durch verschiedene Farben und Modi anzeigen lassen, ob sich der Logger in der Initialisierungsphase befindet, ob gerade ein Log-Intervall abgeschlossen wurde oder ob ein Fehler vorliegt. Geeigneter Quellcode managt, welche der drei LEDs jeweils angesteuert wird, wie oft sie blinkt und wie lange sie ein- und ausgeschaltet sein soll.

Beispielhafte Ansteuerung der RGB-LED

```
const int Led_Rot = 4;
const int Led_Gruen = 5;
const int Led_Blau = 6;
//Beispiel-Aufruf: Statusausgabe(5, LED_Rot, 500, 1000);
void Statusausgabe(int Anzahl, int LED, int Dauer, int Pause)
{
  //Wiederholen des Blinkens
  for(int i=0; i < Anzahl; i++)
  {
    //LED ein für vorgegebene Dauer
    digitalWrite(LED, true);
    delay(Dauer);
    //LED aus für identische Dauer
    digitalWrite(LED, false);
    delay(Dauer);
  }
  //Wartezeit nach der Sequenz
  delay(Pause);
}
```



Das Arduino-LED-Modul RGB-LED Keyes KY-016 eignet sich perfekt für den Datenlogger.

SD-Modul

Mit einem Mikro-SD-Modul, das per SPI mit dem Arduino kommuniziert, kann man die erhobenen Daten einfach speichern und auswerten. Die Daten sollte man im CSV-Format ablegen und zum Beispiel durch Kommas zwischen den Messwerten und Zeilenumbrüche zwischen den Logzyklen trennen.

Dateinummerierung

Soll der Logger mehrere Messvorgänge nacheinander ausführen, ohne dass die Daten von der SD-Karte zwischenausgewertet werden, sollte man die Dateien fortlaufend nummerieren. Andernfalls werden alle Daten in ein File geloggt, was die Übersicht beim Auswerten deutlich erschwert. Hierfür bietet sich die Bibliothek „EEPROM.h“ an. Sie ermöglicht es, einen Dateiindex aus dem nichtflüchtigen Speicher nach dem Start des Loggers zu laden, zu inkrementieren und abzuspeichern.

Zeichenketten

An sich funktioniert das Loggen auf SD-Karte sehr verlässlich. Probleme kann es aber geben, wenn Log-Strings sehr lang sind. Das ist bei GPS-Datensätzen durchaus der Fall. Durch die sich verändernde Länge der Log-Strings kann es passieren, dass sich zur Laufzeit Speicherbereiche überlappen und die Strings „beschädigt“ werden. Im ungünstigsten Fall schlägt der Schreibzugriff auf die SD-Karte sogar komplett fehl. Wir empfehlen daher, keine zu langen Log-Strings zu ver-

Das Mikro-SD-Modul von kwmobile ist kostengünstig und mit dem Arduino kompatibel.



Verwaltung der Dateinummerierung im Festwertspeicher

```
#include <EEPROM.h>
float Dateiindex = 0.0f;
//Einmaliger Aufruf der Funktion bei Programmstart im setup
void Erhoehe_Dateiindex()
{
  //Lesen des letzten Index aus dem Festwertspeicher
  EEPROM.get(0, Dateiindex);
  Serial.println("Alter Datei-Index: :" + String(Dateiindex, 0));
  //Inkrementieren des Index
  Dateiindex = Dateiindex + 1;
  //Begrenzen des Index auf das Intervall [1..100]
  if (Dateiindex > 100)
  {
    Dateiindex = 1;
  }
  //Schreiben des neuen Index
  EEPROM.put(0, Dateiindex);
  //Überprüfen des Erfolgs des Schreibvorgangs
  EEPROM.get(0, Dateiindex);
  Serial.println("Neuer Datei-Index: :" + String(Dateiindex, 0));
}
```

wenden und den Arbeitsspeicher des Mikrocontrollers bis maximal zu 75% auszunutzen.

Der Trick besteht darin, immer abwechselnd einen relativ kurzen Teil-Log-String zu

bilden, diesen auf die Speicherkarte wegzuschreiben und diesen Vorgang mehrfach zu wiederholen. Im Quellcode schaut das konkret so aus:

Zusammenfügen der Daten, Teil 1

```
Daten_Zeichenkette =
String(Systemzeit) + "," +
String(Spannung, 2) + "," +
String(Systemtemperatur, 1) + "," +
String(Temperatur_GY521, 1) + "," +
String(Aussen_Temperatur, 1) + "," +
String(Aussen_Luftfeuchte, 1) + "," +
String(Aussen_Luftdruck, 2) + ",";
//Ausgabe & Wegschreiben der Daten
Log("DATA.CSV", false);
```

Zusammenfügen der Daten, Teil 2

```
Daten_Zeichenkette =
String(Analogwert_0, 1) + "," +
String(Analogwert_1, 1) + "," +
String(Analogwert_2, 1) + "," +
String(Analogwert_3, 1) + "," +
String(Temperatur_2, 1) + "," +
String(Temperatur_3, 1) + ",";
//Ausgabe & Wegschreiben der Daten
Log("DATA.CSV", false);
```



ODER
AUTONOME DROHNE?

Neugierig geworden?

**Testen Sie jetzt 3 Ausgaben
Technology Review und sparen
Sie über 9 Euro.**

Lesen, was wirklich zählt in
Digitalisierung, Energie, Mobilität,
Biotech.



**Bestellen Sie jetzt unter
trvorteil.de/3xtesten**

 trvorteil.de/3xtesten

 +49 541/80 009 120

 leserservice@heise.de

GPS

Es gibt mittlerweile sehr viele gut funktionierende GPS-Module für den Arduino. Aber gerade bei Höhenanwendungen gibt es einige Kniffe, die über Erfolg oder Misserfolg der GPS-Aufzeichnung bei einer Ballonmission entscheiden können.

COCOM-Limit

Um zu verhindern, dass mit frei erhältlichen GPS-Modulen Waffensysteme gesteuert werden können, regelt das COCOM-Limit, dass sich zivile GPS-Module ab einer Ge-

schwindigkeit von 1900 km/h und ab einer Höhe von 18km deaktivieren müssen. Einige Hersteller interpretieren diese Forderung als UND-Verknüpfung, andere als ODER.

Bei den Empfängern des Typs u-blox NEO-6M ist die Bedingung als UND implementiert – die Empfänger deaktivieren sich nur, wenn Geschwindigkeit UND Höhe die Grenzen überschreiten. Da die Aufstiegs-geschwindigkeit unseres Stratosphärenballons aber nur bei 5–6m/s liegt, bleibt das GPS-Modul aktiv. Die maximal erfassbare

Daten im GPGLL-Format

Bedeutungen der Platzhalter im Datensatz

Ausrichtung des
Breitengrads
(N = Norden,
S = Süden)

Ausrichtung des
Längengrads
(E = Osten,
W = Westen)

\$GPGLL,HHMMSS.ss,BBBB.BBBB,b,LLLL.LLLL,l,...

aktuelle Uhrzeit
(UTC)

Breitengrad
in Grad und
Minuten

Längengrad
in Grad und
Minuten

**Technology
Review**
Das Magazin für Innovation

Setzen des Flugmodus am GPS-Modul

```
void Initialisiere_GPS()
{
//Starten der seriellen Kommunikation zum GPS-Board
GPS.begin(9600);
//Das 9. Feld (Wert 0x06) setzt den Flugmodus
//Pedestrian Mode = 0x03
//Automotive Mode = 0x04
//Sea Mode = 0x05
//Airborne < 1G Mode = 0x06
uint8_t setNav[] = {
0xB5, 0x62, 0x06, 0x24, 0x24, 0x00, 0xFF, 0xFF, 0x06, 0x03, 0x00, 0x00,
0x00, 0x00, 0x10,
0x27, 0x00, 0x00,
0x05, 0x00, 0xFA, 0x00, 0xFA, 0x00, 0x64, 0x00, 0x2C, 0x01, 0x00, 0x00,
0x00, 0x00, 0x00,
0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x16, 0xDC };
//Verschicken eines Arrays gemäß UBX-Protokoll an das GPS-Modul
for(int i=0; i<sizeof(setNav)/sizeof(uint8_t); i++)
{
GPS.write(setNav[i]);
}
GPS.println();
}
```

Auslesen von GPGGA-Datensätzen

```
//Ist das Modul verfügbar?
if (GPS.available() > 0)
{
//Einlesen von Datensätzen, bis zum 1. GPGGA-Datensatz
if (!Daten_Zeichenkette.startsWith("$GPGGA") ||
//Ignorieren ungültiger GPS-Datensätze: Das Zeichen "$" soll nur
genau 1x am Anfang stehen
!Daten_Zeichenkette.lastIndexOf("$") == 0)
{
//Einlesen beliebiger GPS-Pakete
Daten_Zeichenkette = GPS.readStringUntil('\n');
}
}
```

Höhe liegt bei 50km – das ist ideal für Stratosphärenballons.

Flight Mode

GPS-Module verfügen in der Regel über verschiedene Betriebsarten, für die optimierte Algorithmen aufgerufen werden. Ab Werk sind die meisten GPS-Module auf den „Pedestrian“-Modus eingestellt. Das bedeutet: Optimierung auf niedrige Geschwindigkeit, exakte Position und gleichbleibend niedrige Höhe. Die Höhenmessung endet hier bei 12km. Bei Stratosphärenballons liegt der Fokus aber natürlich auf großer Höhe. Daher sollte im Quellcode der Modus auf „Airborne < 1G“ gesetzt werden: Die Optimierung erfolgt auf niedrige Geschwindigkeit und große Höhe.

Am GPS-Modul lassen sich noch sehr viele andere Parameter einstellen. Weiterführende Infos findet man in den Links zum Artikel.

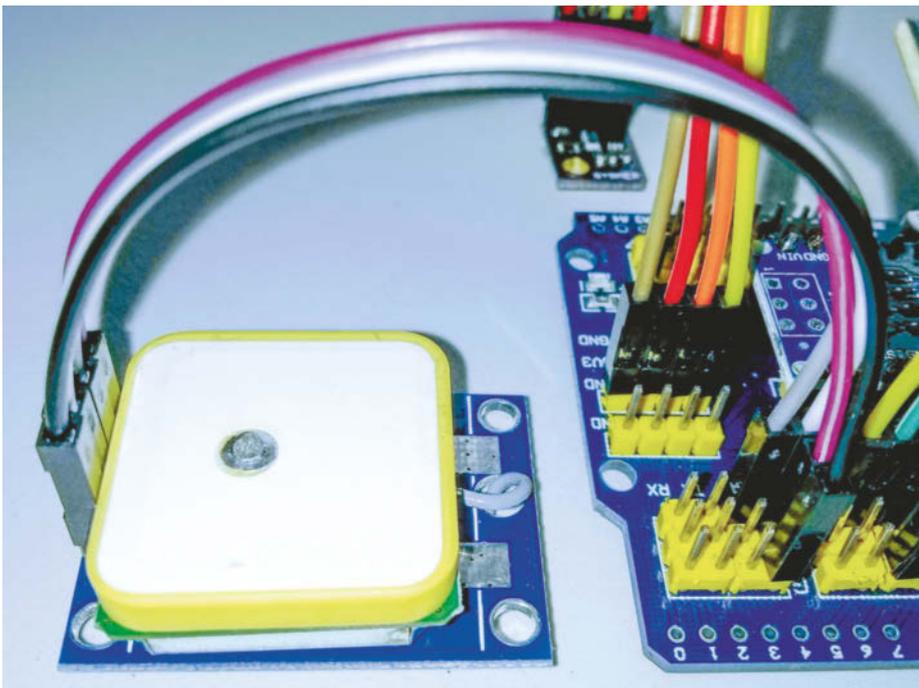
Antenne

Die GPS-Module arbeiten sehr zuverlässig. Voraussetzung ist allerdings, dass die Antenne korrekt ausgerichtet ist (Lötrose nach oben) und nicht verschattet wird. Für Ballonanwendungen reichen aus Gewichtsgründen kleine, leichte Keramikantennen aus. Styroporsonde und Ballon bestehen im Wesentlichen aus Luft beziehungsweise Helium und dämpfen damit elektromagnetische Wellen kaum. Zudem erfolgt der Start in der Regel bei gutem Wetter. Die Atmosphäre bietet dann ideale Empfangsbedingungen. Während des Aufstiegs wird die Atmosphäre immer dünner und der Empfang somit noch besser.

Protokoll

GPS-Module können Meldungen verschiedenster Protokolle ausgeben. Das u-blox-NEO-6M-Modul zum Beispiel gibt unter anderem Meldungen des Typs NMEA 0183 GPGGA aus. Dabei handelt es sich um einen Standard für die Kommunikation zwischen Navigationsgeräten auf Schiffen, der von der National Marine Electronics Association (NMEA) entwickelt wurde. Dieser Standard wird inzwischen auch für die Kommunikation zwischen GPS-Empfängern und PCs oder mobilen Endgeräten genutzt.

Die Daten von NMEA 0183 basieren auf ASCII und können jeweils bis zu 80 druckbare Zeichen enthalten. Die GPGGA-Datensätze beginnen mit einem Dollar- oder einem Ausrufezeichen, danach folgen unter anderem Uhrzeit, Breiten- und Längengrad im Klartext. Auch die Höhe wird ausgegeben, wodurch die Daten für Ballonanwendungen geeignet sind. Das Filtern der Meldungen kann zur Laufzeit erfolgen.



Das GPS-Modul mit korrekt ausgerichteter Antenne

Außensensor

Neben der Erfassung von GPS-Daten besteht die Kernaufgabe des Loggers in der Aufzeichnung von Umweltparametern. Um die Eigenschaften der umgebenden Luft zu erfassen, eignen sich besonders gut Sensoren des Typs Bosch BME280. Sie bieten eine gute Preisleistung und eine hervorragende Performance. Neben Temperatur und Luftfeuchte kann auch der Luftdruck per I2C-Bus abgefragt werden. Alternativ würden sich auch die beiden Klassiker DHT22 (Temperatur und Luftfeuchte) und DS18B20 (Temperatur) anbieten, allerdings liefern sie weniger Daten.

Laut Datenblatt ist der BME280 für eine minimale Temperatur von -40°C und einen minimalen Luftdruck von 300mbar geeignet. Bei ersten Tests lieferte er allerdings verblüffende Ergebnisse: Temperaturen bis -58°C und Drücke bis 12mbar wurden stabil ausgegeben. Die bei den Testflügen gemessenen Werte (Temperatur, Luftdruck, Luftfeuchte) decken sich sowohl mit den Modellen als auch mit Messungen aus anderen Flügen mit anderen Sensoren. Der Sensor nahm hierbei keinen Schaden.

Höhenberechnung

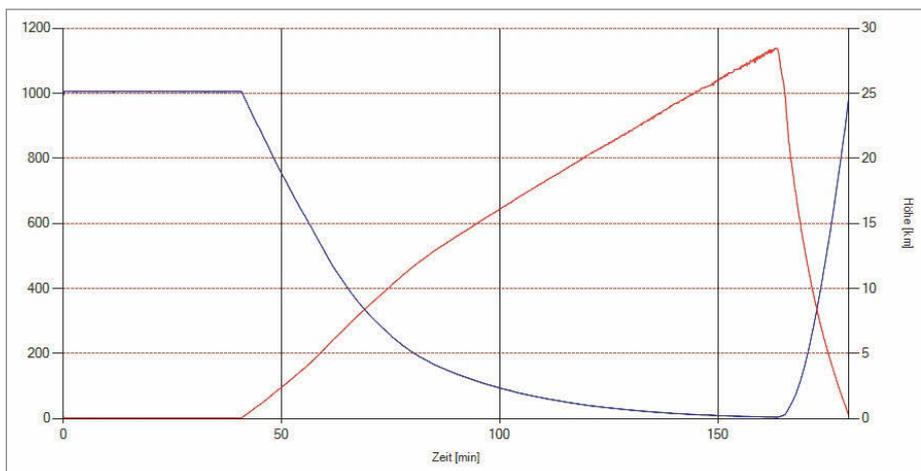
Schön, dass das GPS-Modul die Höhe ausgibt. Aber noch besser, dass man eine Kontrollgröße hat: den Luftdruck. Daraus lässt sich nach der internationalen barometrischen Höhenformel näherungsweise der Höhenverlauf ableiten. Die Formel beschreibt, wie dicht die Atmosphäre der Erde auf welcher Höhe ist. Sie liefert vor allem für die Troposphäre bis zu einer Höhe von ca. 11km verlässliche Ergebnisse.

Analogensoren

An den Analogeingängen kann man insgesamt bis zu sechs Sensoren anschließen – die Anschlüsse A4 und A5 werden bereits für den I2C-Bus benötigt. Um die Analogwerte umzurechnen, kann man die vordefinierte map-Funktion nutzen. Diese arbeitet allerdings mit Ganzzahlen, was unter Umständen zu einem Genauigkeitsverlust führen kann.

Um präzise Messwerte zu erhalten, sollte man die Werte vom AD-Wandler in Gleitkommazahlen überführen und sofern benötigt mit einer Steigung und einem Offset versehen.

Die meisten Analogensoren, wie zum Beispiel Gassensoren, arbeiten als Spannungsteiler mit einem in Reihe geschalteten Serienwiderstand. Das heißt konkret, dass man hier Spannungsschwankungen nicht herauszurechnen braucht. Der vom AD-Wandler ausgegebene Wert spiegelt wirklich das Spannungsteilerverhältnis wider.



Verlauf des Luftdrucks und daraus errechnete barometrische Höhe

$$p(h) = 1013,25 \cdot \left(1 - \frac{0,0065 \frac{\text{K}}{\text{m}} \cdot h}{288,15 \text{ K}} \right)^{5,255} \text{ hPa}$$

$$= p_0 \cdot \left(1 - \frac{0,0065 \frac{\text{K}}{\text{m}} \cdot h}{T_0} \right)^{5,255}$$

Mit der internationalen Höhenformel kann man den Luftdruck berechnen, ohne weitere Faktoren, die bei anderen Formeln notwendig sind, zu kennen.

Bild: Wikipedia

Umrechnen von Analogwerten

```
float Berechne_Analogwert(int Analogwert, float Steigung, float Offset)
{
  //1. Skalieren des Analogwerts auf den Messbereich des AD-Wandlers von 0...1023
  //2. Berücksichtigen der Steigung (z.B. 0...1023 = 0...100%)
  //3. Berücksichtigung der Nullpunktverschiebung
  return float(Analogwert) / 1023.0 * Steigung + Offset;
}
```

Umrechnung Analogwert in Prozent

```
//Skalieren der Sensorwerte auf 0 - 100% für spätere Nachverarbeitung
Analogwert_0 = Berechne_Analogwert(analogRead(14), 100, 0);
```

Umrechnung Analogwert Temperaturfühler in Grad Celsius

```
//Temperatursensoren TMP36:
//Skalierung auf 100°C/V bzw. 10mV/°C
//Nullpunktverschiebung um -50°C
Temperatur_2 = Berechne_Analogwert(analogRead(20), Spannung * 100, -50);
```

Vektorielle Addition der drei Gravitations-Komponenten

```
//Default: Skalierung auf 2G: 32768 / 2 = 16384
float Skalierung_Beschleunigung = 16384.0;
//Default: Skalierung auf 250°/s: 32768 / 250 = 131
float Skalierung_Drehwinkel = 131.0;
//Umrechnen der gelesenen Werte
AcX=Wire.read()<<8|Wire.read();
AcX = AcX / Skalierung_Beschleunigung;
```

Addieren der drei Gravitations-Komponenten

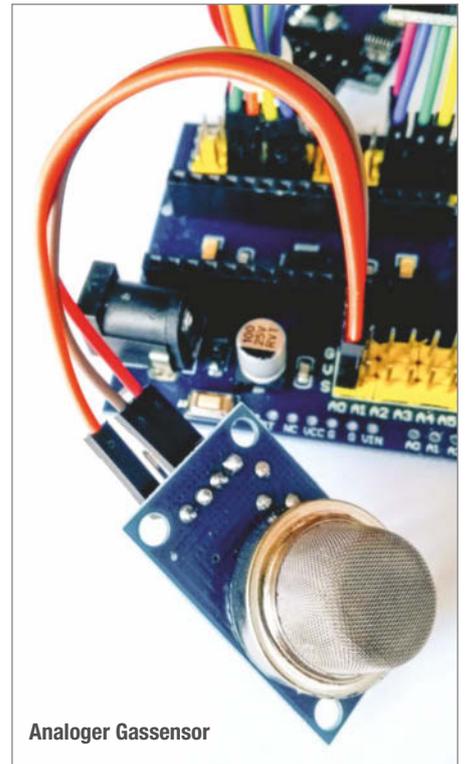
```
float AcX,AcY,AcZ,AcTot;
AcX=Wire.read()<<8|Wire.read();
AcX = AcX / Skalierung_Beschleunigung;
AcY=Wire.read()<<8|Wire.read();
AcY = AcY / Skalierung_Beschleunigung;
AcZ=Wire.read()<<8|Wire.read();
AcZ = AcZ / Skalierung_Beschleunigung;
AcTot = sqrt(pow(AcX, 2) + pow(AcY, 2) + pow(AcZ, 2));
```

Wie man bei Gassensoren die gemessenen Analogwerte in ppm-Werte umrechnen kann, ist sehr gut in Make 1/2019 beschrieben.

Zusatzsensoren

In der Make 1/2019 wurde darüber hinaus eine ganze Reihe an Umweltsensoren vorgestellt. Einige davon bieten sich perfekt für die Stratosphärenerkundung an. Während des Aufstiegs fliegt der Ballon durch die Ozonschicht. In einer Höhe von circa 20km herrscht die maximale Ozon-Konzentration. Hier wird, vereinfacht gesagt, UV-Strahlung in Wärme umgewandelt und dabei IR-Strahlung freigesetzt. Oberhalb der Ozonschicht nimmt dann die UV-Strahlung drastisch zu.

Über einen **Ozonsensor** wie dem MQ-131 lässt sich direkt die Ozon-Konzentration und damit die Stärke der Ozonschicht vermessen. Ein **IR-Sensor** – beispielsweise der Keyes KY-026 – kann indirekt über die Wärmestrahlung der Ozonmoleküle die Stärke der Ozonschicht vermessen. Der rapide Anstieg der UV-Strahlung oberhalb der Ozonschicht kann mit einem **UV-Sensor**, zum Beispiel dem UVM30A, erfasst werden. Darüber hinaus lassen sich beim Aufstieg des Ballons natürlich auch mit einem **Hallsensor** die Abschwächung des Erdmagnetfelds (Keyes KY-024), die Zunahme von Gammastrahlung (GDK101), die Verteilung der Feinstaubbelastung (Sharp GP2Y-1010AU0F), die Verteilung von CO₂ (MG-811) und vieles mehr messen. Hier können Bastler kreativ werden!



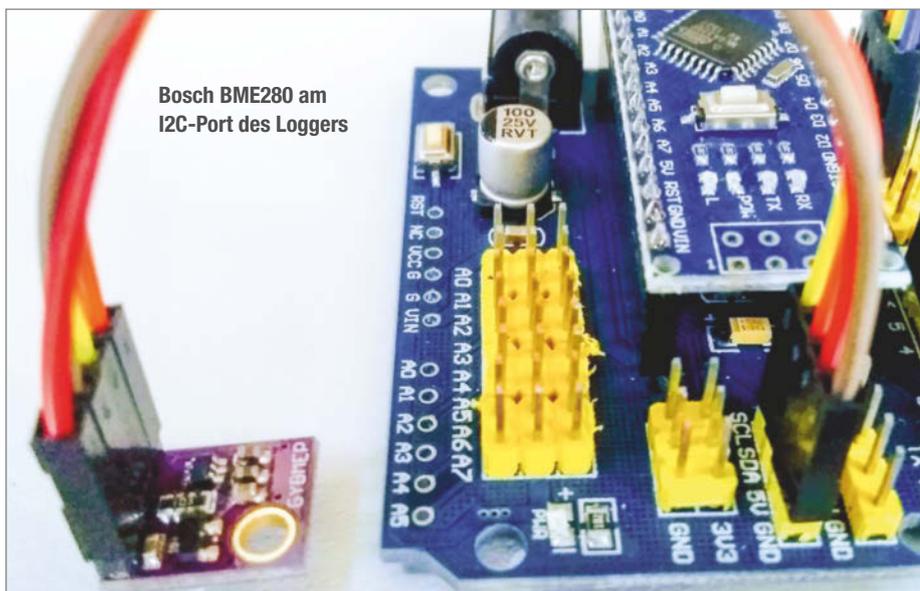
Analoger Gassensor

Gyroskop und Beschleunigungssensor

Mithilfe eines MPU-6050-Sensors auf einem GY-521-Breakout können Beschleunigung und Drehung jeweils in X-, Y- und Z-Richtung und die Temperatur der Sonde erfasst werden. Ab Werk sind die Sensoren auf maximal 2G und 250°/s eingestellt. Die per I2C-Bus übertragenen Werte muss man entsprechend skalieren.

Welche nützlichen Informationen liefert der Sensor? Neben der Sondentemperatur erfasst er auch Daten, mit denen man nachvollziehen kann, wie die Sonde windbedingt pendelt (Beschleunigung) und rotiert (Gyroskop). Zudem kann man die Abnahme der Gravitation mit steigender Höhe erfassen – sie liegt in der Größenordnung von 1%. Nach dem Platzen des Ballons gibt es eine Phase des nahezu ungebremsten freien Falls, ehe sich der Fallschirm entfalten kann. Dabei ist der Sensor relativ zur Sonde schwerelos und misst eine Gravitation nahe Null. Dieses Phänomen kennt man von Astronauten beim Parabelflug. Es lässt sich auswerten, wie lange es im nahezu luftleeren Raum dauert, bis der Fallschirm sich komplett entfaltet hat und seine Bremswirkung einsetzt.

Je nach Missionsziel kann man je alle drei Komponenten von Beschleunigung und Drehung in X-, Y- und Z-Richtung mitschreiben. Häufig reicht es aber aus, die drei Komponenten vektoriell zu addieren und nur den resultierenden Betrag zu loggen.



Bosch BME280 am I2C-Port des Loggers

Diagnose-Display

Alternativ zum Gyroskop kann man am zweiten I2C-Port auch ein Display anschließen. Dieses LC-Display kann beispielsweise über 4 Zeilen à 20 Zeichen verfügen. Damit bietet sich die Möglichkeit, alle Daten kurz vor dem Start noch mal zu checken. Gerade für die Analogensensoren kann das durchaus hilfreich sein.

Anschließend wird das Display natürlich abgesteckt und verbleibt am Boden, um Gewicht und Energieverbrauch nicht unnötig in die Höhe zu treiben.

Für eine gute Überschaubarkeit sollte man die Displayausgabe je Log-Zyklus einmal durchschalten und nur je eine Gruppe von Sensordaten ausgeben. Die GPS-Daten sind zum Beispiel ziemlich umfangreich und nehmen bereits alleine fast das gesamte Display ein. Die Log-Strings können zeilenweise gestückelt werden.

Für eine gute Überschaubarkeit sollte man die Displayausgabe je Log-Zyklus einmal durchschalten und nur je eine Gruppe von Sensordaten ausgeben. Die GPS-Daten sind zum Beispiel ziemlich umfangreich und nehmen bereits alleine fast das gesamte Display ein. Die Log-Strings können zeilenweise gestückelt werden.

```

Aufteilen von Log-Strings für ein
4x20-Display

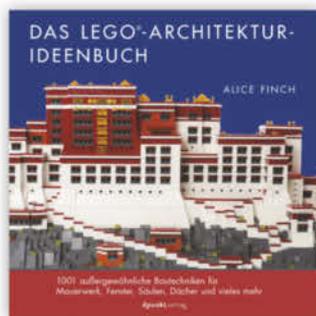
void LCD_Ausgabe()
{
  //Löschen des Display-Inhalts
  lcd.clear();
  //Durchlaufen aller 4 Zeilen
  for (int i=0; i<=3; i++)
  {
    //Anwählen des Starts der einzelnen Zeilen
    lcd.setCursor(0,i);
    //Ausgabe von Teilstrings, die je 20 Zeichen lang sind
    lcd.print(Daten_Zeichenkette.substring(i * 20, (i + 1) * 20));
  }
}
    
```

Unterstützung

Für alle, die bei ihrem Stratosphärenlogger-Projekt ein wenig Unterstützung in Anspruch nehmen wollen, gibt es auf der Homepage von Stratoflights ein umfassendes kostenfreies Download-Paket. Dieses umfasst neben dem Arduino-Quellcode und einer fertigen Windows-Software zum Auswerten der Messdaten eine ausführliche Betriebsanleitung mit Materialliste, Montage- und Inbetriebnahmeanleitung und Schaltplan. Online-Tools helfen zudem beim Planen der Helium-Menge und der Flugroute —esk



Für Maker & Bastler



D. Knox

Roboter selbst bauen

13 Bot-Anleitungen für Maker

2018, 160 Seiten
€ 19,95 (D)
ISBN 978-3-86490-537-7

A. Finch

Das LEGO®-Architektur-Ideenbuch

1001 außergewöhnliche Bautechniken für Mauerwerk, Fenster, Säulen, Dächer und vieles mehr

2019, 232 Seiten
€ 24,90 (D)
ISBN 978-3-86490-642-8

T. Bartoschek · D. Fehrenbach · J. Fehrenbach · M. Pesch · L. Steinmann

Das senseBox-Buch

12 Projekte rund um Sensoren, Umwelt und IoT

2019, 180 Seiten
€ 22,90 (D)
ISBN 978-3-86490-684-8

G. Wostrack

Digitale Modellbahn selbstgebaut

CANguru-Steuerung mit ESP32 in Arduino-Umgebung

2020, 202 Seiten
€ 29,90 (D)
ISBN 978-3-86490-711-1

**Make:
Family**



Alles über Bürstenroboter

Bürstenroboter sind schon seit Jahren ein Hit für große und kleine Maker. Man kann sie schnell bauen, sie sind preiswert und können leicht abgewandelt und verziert werden. Wir stellen hier ein paar Konzepte für die Roboter vor, die nicht immer eine Bürste, aber auf jeden Fall einen Vibrationsmotor enthalten müssen.

von Carsten Wartmann

Kurzinfo

- » **Bewegungsprinzip** Vibrationsmotor mit Untersatz, der Vibration überträgt
- » **Unwucht an Motor bauen**

Check



Zeitaufwand:
circa eine Stunde



Kosten:
je nach Motor/Akku/Bürste
5–15 Euro

Mehr zum Thema

- » Elke Schick, Simple Bots: Der Skitter, Make 6/18, S. 84
- » Elke Schick, Maker Education in der Schule, Make 3/19, S. 64

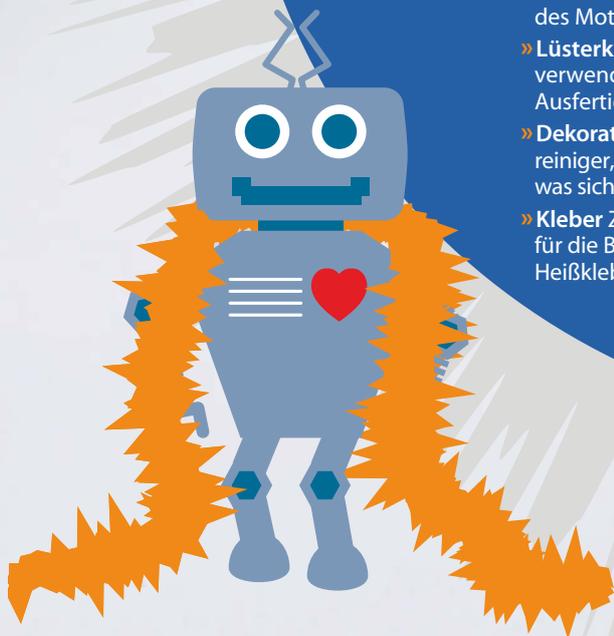
Material

- » **Bürste** von der Zahn- bis zur Spülbürste
- » **Elektromotor** Auswahl je nach Bürstengröße und nach Gewicht der Stromversorgung
- » **Stromversorgung** wird auf Basis des Motors ausgewählt
- » **Lüsterklemmen** optional, wenn verwendet, dann möglichst kleine Ausfertigung
- » **Dekoration** Wackelaugen, Pfeifenreiniger, Federn und alles andere, was sich durch die Vibration bewegt
- » **Kleber** Zwei-Komponenten-Kleber für die Befestigung der Bauelemente, Heißkleber für die Dekoration

Material Partyroboter

- » **Vibrationsmotor DC 3V**
10 × 2,7mm
- » **Batteriehalter**
2 × AA mit Schalter
- » **Supercap Superkondensator**
3V, 5F, 10 × 20mm
- » **Buchsen- und Stiftleisten**
2,54mm-Raster
- » **Kabel**

Alles zum Artikel
im Web unter
make-magazin.de/xz4c



Eigentlich sind die sogenannten Bürstenroboter gar keine Roboter, sondern nur eine Art von Fahrzeugen. Sie bestehen aus einem Vibrationsmotor mit Stromversorgung und einem Untersatz, der durch den Motor in Bewegung versetzt wird. Man kann sie nicht steuern oder programmieren. Trotzdem – oder vielleicht gerade deswegen – sind sie sehr beliebt und wirklich gut als Einstiegsprojekt für Elektronikbastler geeignet.

Ein Vibrationsmotor ist ein Motor, der mithilfe einer Unwucht an der Achse Vibrationen erzeugt. Solche Motoren kannst du fertig kaufen, zum Beispiel in

der besonders kleinen Form, die in Mobiltelefonen verwendet wird. Aber auch fast jeder Motor aus einem defekten batterie- oder akkubetriebenen Spielzeug ist geeignet.

Wenn du wissen willst, wie ein Vibrationsmotor funktioniert, mache einmal folgendes Experiment: Nimm einen schweren Gegenstand, den du gut festhalten kannst, in beide Hände und drehe dich schnell im Kreis. Du wirst merken, wie das Gewicht des Gegenstands versucht, dich nach außen zu ziehen, wenn du dich nicht kräftig dagegenstemmst.

Genauso wird auch ein Motor mit einer Unwucht von deren Gewicht nach

außen gezogen. Da die Unwucht an der Welle des Motors angebracht ist – also dem Teil des Motors, der sich dreht –, wird der Motor ständig in wechselnde Richtungen gezogen. Er wackelt hin und her. Setzt du einen solchen Motor auf einen Gegenstand, dann wird dieser Gegenstand von dem Motor hin und her bewegt – er vibriert. Ein schwerer, großer und flacher Gegenstand wird von dieser Vibration nicht sehr stark bewegt. Dinge wie Bürsten hingegen, die auf vielen kleinen Füßen stehen, kommen leichter in Bewegung. Der Zug der Unwucht sorgt dafür, dass sie beginnen, sich zu bewegen.

Die verschiedenen Roboterklassen

Zahnbürsten-Roboter

Die Basis dieser kleinen Roboter ist ein Zahnbürstenkopf. Als Motor eignen sich besonders kleine Vibrationsmotoren aus Mobiltelefonen. Diese Motoren kann



man kostengünstig im Fünfer- oder Zehnerpack kaufen. Als Energiespeicher kommen Knopfzellen (nicht sehr umweltfreundlich) oder besser Superkondensatoren in Frage.

Obwohl sie so klein sind, kann man mit diesen Gesellen eine Menge anstellen. Jede Änderung an der Biegung oder Länge der Borsten, Position und Lage des Motors verändert die Eigenschaften des Bots. Mit ein paar einfachen Utensilien – wie Wackelaugen und Pfeifenreinigern – kann man wahre Kunstwerke schaffen.



Zahnbürstenroboter gibt es auch als Bausätze zu kaufen. Hier reichen die Ausführungen von einfachen Teilesets, zu denen man noch eine Zahnbürste beisteuern muss, bis zu High-Tech-Kits, die spezielle Bürsten und Spritzgussteile mitbringen. Die besonders ausgefeilten Sets kommen mit zwei Motoren und Bürsten. In diesem Fall können die fertigen Roboter auch gesteuert werden.

Spülbürsten-Roboter

An sich unterscheidet nur die Größe diese Modelle von den Zahnbürstenrobotern. Die Bauteile sind aber wesentlich günstiger. Zusätzlich kann man fast alle Motoren aus kaputtem motorisier-

ten Spielzeug benutzen und zudem noch normale Batterien oder besser Akkus. Wie bei allen Arten der Vibrobots können nicht nur Bürsten als Beine verwendet werden.



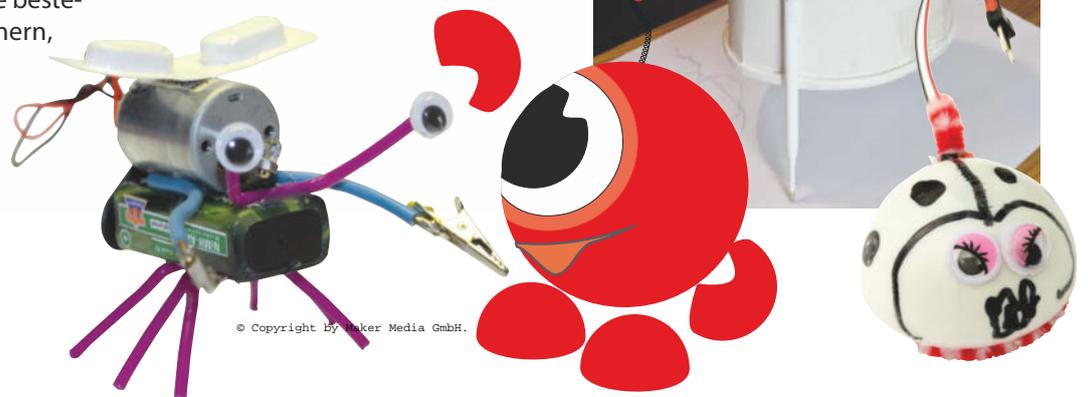
Offene Klasse aka Wobblebot

Mit einem Vibrationsmotor kann man nicht nur Borsten in Bewegung bringen. Alles, was zwar stabil ist, aber keine große Standfläche hat, ist auch geeignet. Beispiele sind Radiergummistücke, Pappbecher, halbe Bälle oder selbstgebaute Drahtbeine. Aufgrund ihrer Bewegungsform werden diese Roboter auch Wobblebots genannt.

Eine besondere Klasse innerhalb dieser Kategorie sind die Artbots: Sie bestehen aus Plastik- oder Pappbechern, an die mehrere Malstifte oder Kugelschreiber so angeklebt sind, dass die Spitzen der Stifte die Füße des Bots sind. Wenn man sie einschaltet, malen sie

interessante Muster auf eine geeignete Unterlage (nicht den Küchenfußboden!).

Der ursprüngliche Wobblebot baut allerdings darauf, dass ein Stück Radiergummi auf der Motorwelle befestigt wird, das – im Gegensatz zu den Motoren der anderen Roboter – den Boden berührt. Der Roboter wird so halb fahrend und halb springend durch die Gegend bewegt.



Partyroboter

Der Ausgangspunkt dafür, mich nochmals intensiver mit den Zahnbürstenrobotern zu beschäftigen, war der Wunsch meines Sohns (10), eine Geburtstags-Forscher-Party zu machen. Konkret wünschte er sich „irgendwas mit Robotern“. Aha. Einen billigen Plastik-Roboter zu kaufen, der 5 Minuten Spaß macht, kam bei unserer bastelaffinen Familie nicht in Frage. „Richtige“ Roboter wären zu kompliziert und zu teuer gewesen. Also mussten wir selbst etwas „maken“!

Auf dem Vintage Computer Festival Berlin hatte ich Zahnbürstenroboter mit Supercaps gesehen, das sollte es sein. Die Supercaps sind wiederaufladbar und daher nachhaltiger als Knopfzellen. Dadurch haben sie auch nach der Party noch Spiel- und Bastelwert. Für das Aufladen benötigt man zwei AA-Batterien. Die einfachste Lösung war hier, die Batterien in einem Batteriehalter mit Schalter unterzubringen. Wenn die Supercap-Kondensatoren entladen waren, mussten die Kinder sie mit dem ausgeschalteten Batteriehalter verbinden, den Batteriehalter einschalten, bis zehn zählen und ihn dann wieder ausschalten.

Nur wie hält man den Bau der Roboter auf einem Geburtstag unter Kontrolle? Löten geht mit so vielen Anfängern absolut nicht. Die Drähte nur zu klemmen oder zu verzwirbeln erschien mir nicht haltbar genug. Und außerdem sollte die Hauptsache doch das Verschönern und Ausprobieren der Bots sein. Die Technik so vorzubereiten, dass sie nur noch auf eine Bürste gepappt werden muss, fand ich aber eine zu geringe Herausforderung für die Kinder.

Meine Lösung bestand dann darin, vorher einige Komponenten so vorzubereiten, dass sie später nur noch gesteckt und aufgeklebt werden müssen.

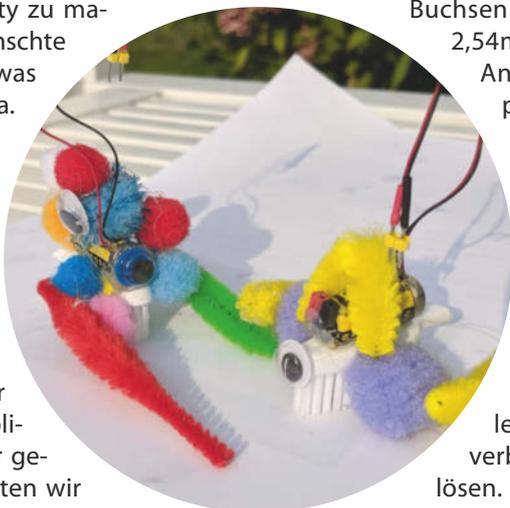
So mussten sich die Kinder zwar auch mit der Elektronik beschäftigen, aber nicht selbst löten. Für die Umsetzung des Stecksystems verwendete ich Buchsen und Stifte mit 2,54mm-Raster. An die Anschlüsse der Superkondensatoren lötete ich Buchsen an. An die Anschlüsse der Batteriehalterung und der Motoren jeweils Stifte. So konnten die Kinder die Bauelemente leicht miteinander verbinden und wieder lösen.

Beim Kauf der Superkondensatoren muss man verschiedene Dinge beachten. Die zulässige Spannung (V/Volt) der Kondensatoren sollte nicht unterhalb der benötigten Spannung für den Motor liegen. Die Spannung der Batterien zum Aufladen darf nicht über

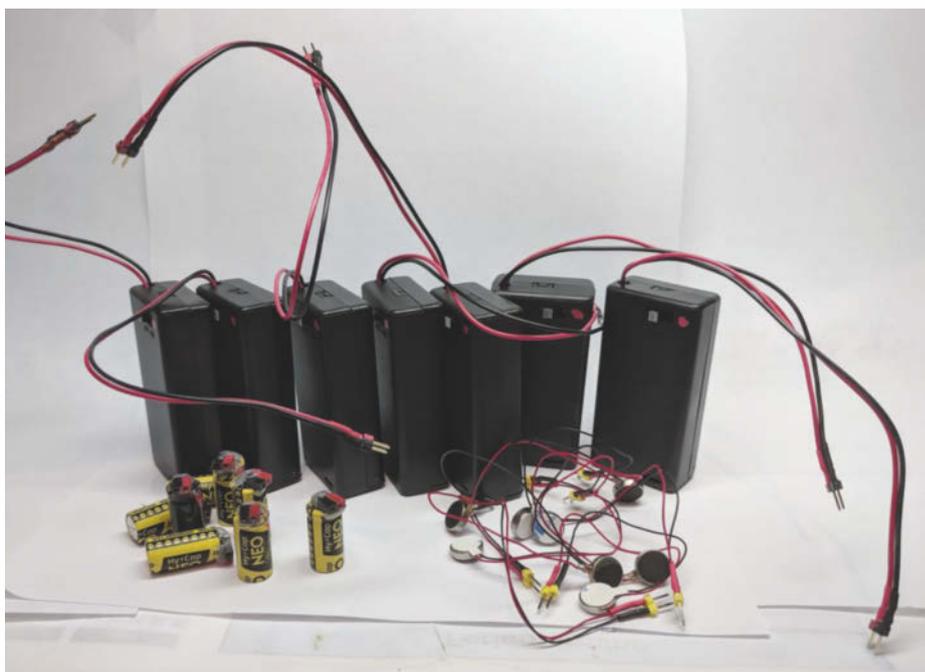
der der Kondensatoren liegen. Ihre Kapazität sollte höher als 4 Farad (F) sein, sonst sind sie zu schnell wieder entladen. Außerdem sollten die Kondensatoren nicht zu groß sein.

Vorbereitung

Mein Sohn und ich haben die Lötarbeiten in circa 2 Stunden an einem freien Nachmittag erledigt. Mit Lackmarkern sowie rotem und schwarzem Schrumpfschlauch haben wir die Polung der Komponenten markiert. Mit diesen Markierungen und der klaren Ansage, dass die Komponenten kaputt gehen, wenn man sie verkehrt herum anschließt, haben alle Supercaps die Party überlebt. Eine alternative Lösung wären kleine verpolssichere Stecker/Buchsen oder gegebenenfalls auch eine Diode gewesen. Dies konnte ich aber mangels Zeit nicht testen. Aber ein System nicht zu sicher zu machen kann auch das Gefühl für Verantwortung und die Physik dahinter stärken, und das ist ja auch wichtig. Zum Bau der Komponenten brauchst du etwas Löterfahrung.



Zwei stark bepöschelte Partyroboter

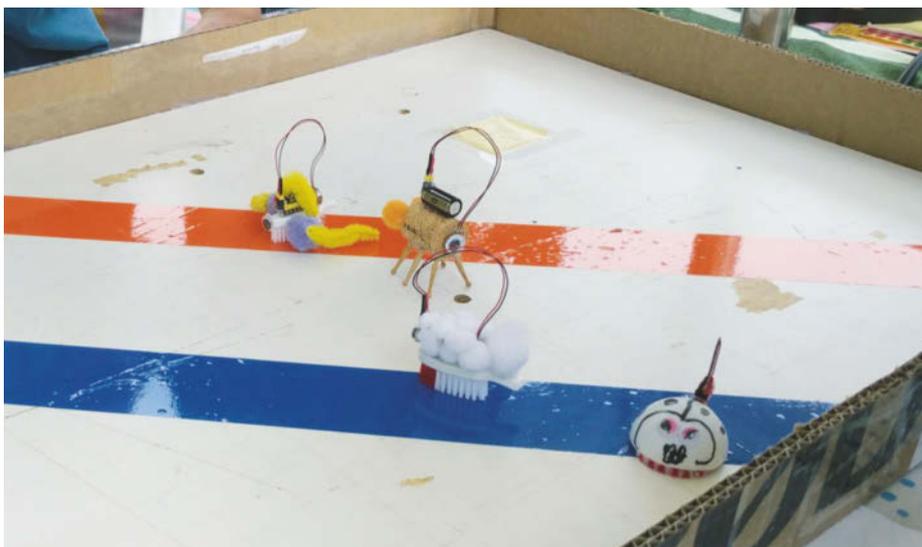


Die Elektronik-Komponenten der Partyroboter: vorne links die Superkondensatoren, rechts die Vibrationsmotoren und hinten die Batteriehalter

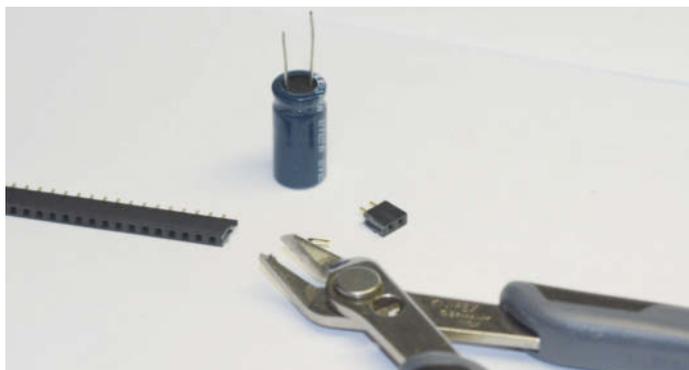
Die Party

Auf unserer Party haben 10 Kinder gebastelt. Und alle haben am Ende einen sehr individuellen und funktionalen Roboter auf der Bahn gehabt! So wuselig die Party vorher war, beim Basteln und meinen Erklärungen waren alle mit Eifer bei der Sache. Besonders viel Spaß machte es allen, die Roboter zu verzieren und zu verschönern. Anschließend wurden die Roboter auf unsere Bahn zum „Battle“ oder zum Schaulaufen gesetzt.

Eine ausführlichere Anleitung für den Bau der Roboter und der Arena habe ich online als PDF zur Verfügung gestellt. Es ist zusammen mit vielen anderen Anleitungen für Bürstenroboter in den Links zu diesem Artikel zu finden.

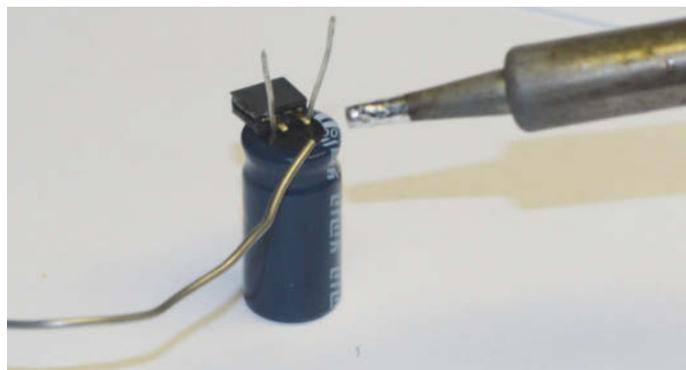


Tanzfläche oder Kampfarena: Die fertigen Roboter können hier gegeneinander antreten.



Buchsen und Kondensatoren vorbereiten

Schneide zwei Buchsen mit einer Kneifzange ab. Eventuell musst du sie etwas glattschleifen. Biege die Beinchen des Kondensators etwas auseinander, damit die Anschlüsse der Buchse genau dazwischenpassen.



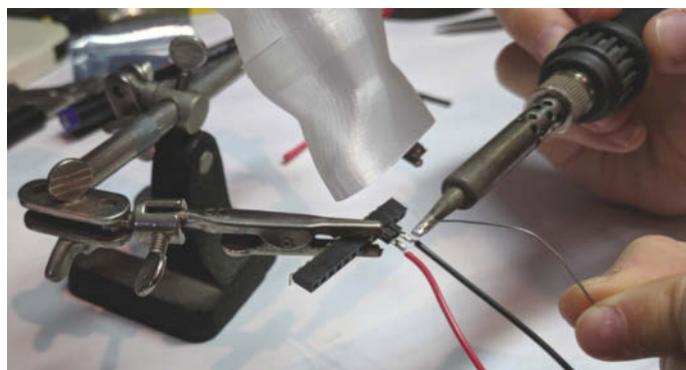
Löten

Jetzt verbindest du die Anschlussstifte der Buchsen mit den Anschlüssen der Kondensatoren, indem du sie mit Lötzinn überziehst. Löte zügig, aber nicht hektisch, damit der Kondensator nicht zu heiß wird.



Sichern und markieren

Nun markierst du die Buchse, die am Pluspol angeschlossen ist, mit Lackstift rot. Dann knipst du die überstehenden Drähte ab und gibst danach etwas Heißkleber darauf.



Stifte anlöten

Um die Stifte anzulöten, klemmst du am besten eine Buchsenleiste in eine dritte Hand und steckst dort die Stifte hinein. Danach ziehst du Schrumpfschlauch über die Anschlusskabel des jeweiligen Bauelements und lötest die Kabel an die Stifte.

Material, Werkzeug und Bautipps



Material

Wenn du Glück hast und deine Eltern sogenannten Elektroschrott aufheben, hast du viel vom benötigten Material vielleicht schon zuhause. Aus altem und/oder kaputtem elektrischen Spielzeug kann man oft Motoren und Batteriehalter ausbauen.

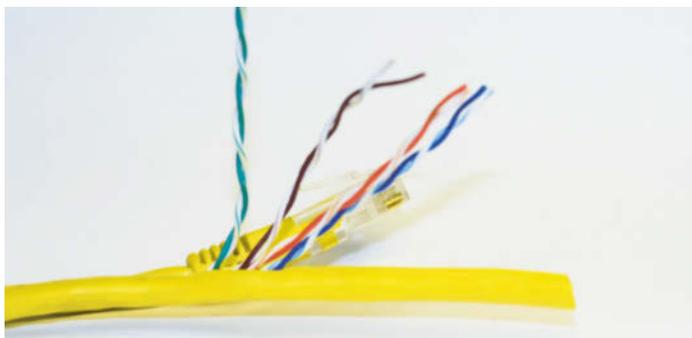
Die elektronischen Bauteile sind in den Spielzeugen und anderen Geräten

meistens in Gehäuse gebaut, die du aufschrauben und vielleicht auch aufbrechen musst. Setze Dich mit einem Erwachsenen zusammen, der dir zeigt, welche Bauteile du gebrauchen kannst und wie du sie ausbaust.

Wie bei den Wobblebots schon erwähnt, bist du bei der Wahl der Basis deines Roboters nicht auf Bürsten

beschränkt. Hierfür kann fast alles dienen, was nicht zu schwer ist und die Schwingungen des Motors nicht absorbiert.

Für die Dekoration brauchst du natürlich Wackelaugen, Pfeifenreiniger, Pommes, Schnüre, Glitzerfolie und buntes Papier. Lasse deiner Kreativität freien Lauf!



In der Plastikhülle von Kabeln befinden sich ein oder mehrere Metalldrähte.



Alle nicht-elektronischen Bestandteile unserer Roboter

Werkzeug

Dein wichtigstes Werkzeug ist wahrscheinlich eine Heißklebepistole! Ich empfehle die Modelle mit den 7mm dicken Klebesticks, weil diese schon bei niedrigen Temperaturen schmelzen. Den Kleber für Heißklebepistolen gibt es auch mit Glitzer oder farbig, so kannst du den Roboter beim Kleben gleich verschönern. Diesen Heißkleber kannst du wieder gut abpulen und so deine Designs sehr oft ändern und verbessern.

Wenn du größere Bürstenroboter mit stärkeren Motoren baust, reicht Heißkleber für die Befestigung des Motors nicht aus. Du solltest den Heißkleber an deinem Motor dann, wenn du die ideale Position gefunden hast, durch Zweikomponenten-Kleber ersetzen. Der braucht zwar länger zum Trocknen, hält dafür aber die Rüttelbewegungen auf Dauer besser aus.

Zudem solltest du eine Auswahl von Zangen (Seitenschneider, Flachzange), Scheren, Messer und dergleichen zur Verfügung haben. Du brauchst sie, um das Dekomaterial und die Kabel zurecht-

zuschneiden und am Kabel die Drähte freizulegen. Es ist wichtig, dass du dir die Handhabung aller Werkzeuge zuerst von einer erwachsenen Person zeigen lässt.



Bautipp: Kabel verbinden

Der beste Weg, um Kabel sicher zu verbinden, ist das Lötten. Aber Lötten will gelernt sein. Wenn du kannst, hole dir hier Hilfe von einem Erwachsenen, der dies beherrscht! Aber auch wenn niemand in deinem Umfeld lötten kann, steht deinem Roboter nichts im Wege. Für erste Versuche kannst du Lüsterklemmen (die kleinsten, die man be-

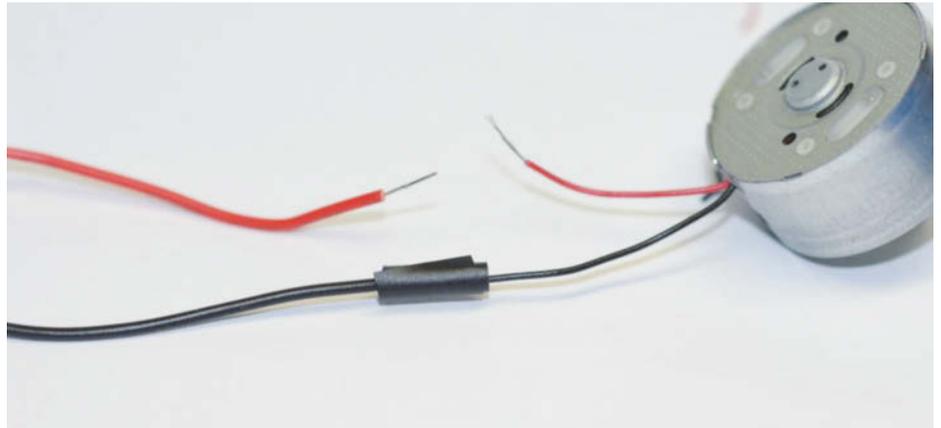
kommen kann) verwenden oder die Drähte zusammenzwirbeln und dann mit Klebeband oder Schrumpfschlauch isolieren.

Sehr wichtig für einen guten Kontakt ist es, die Kabel gut von der Plastikisolierung zu befreien. Dies geht am einfachsten und sichersten mit einer Abisolierzange. Alternativ kannst du auch mit

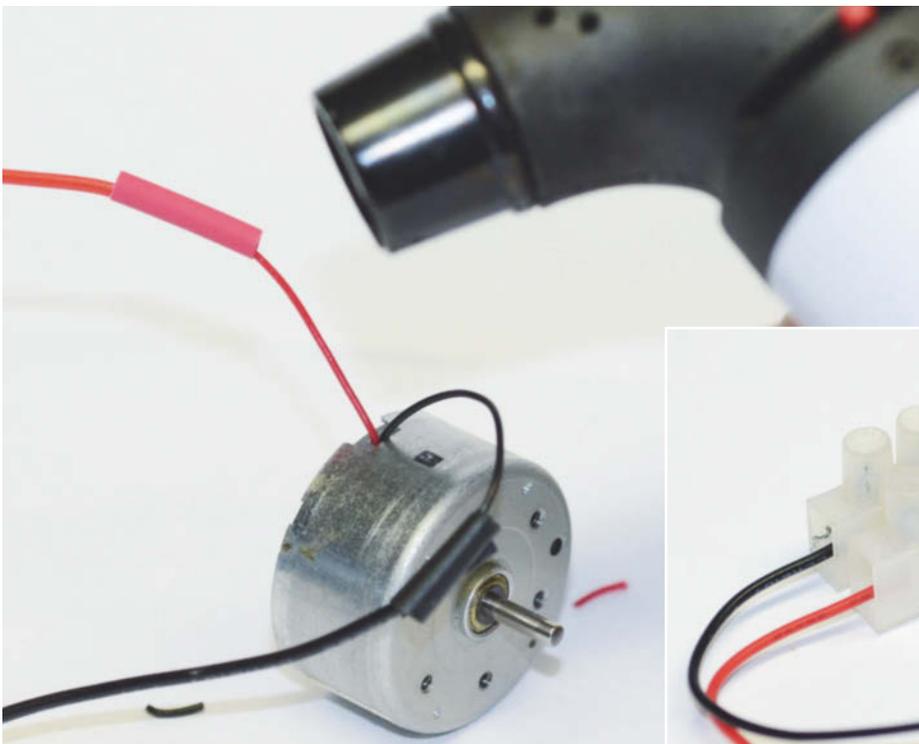
einem Seitenschneider vorsichtig die Isolierung einkneifen und dann am Kabel ziehen. Oder du schneidest die Plastikhülle mit einer Schere ein. Ziel ist es, die Isolierung zu entfernen, ohne die Leiter zu beschädigen. Am besten übst du an Kabeln aus den kaputten Elektronik, die du bei deinen Eltern eingesammelt hast.



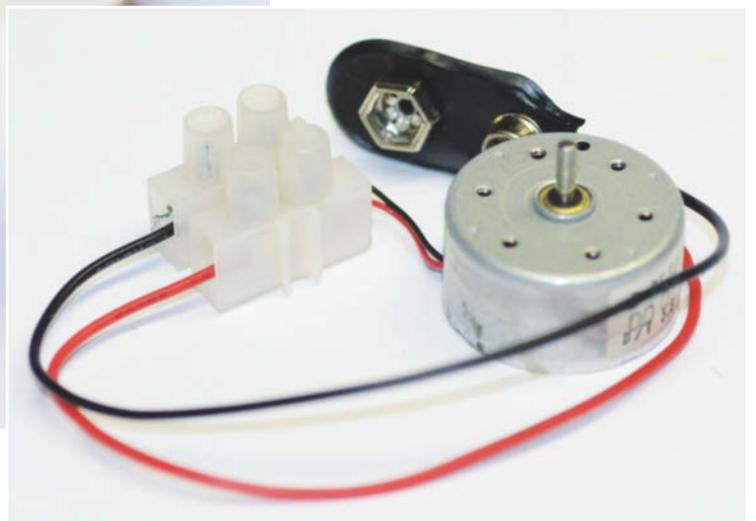
Verzwirbeln ist eine einfache Art, Kabel zu verbinden.



Verzwirbelte Kabel kannst du mit Klebeband sichern, sodass sie länger halten und keine Kurzschlüsse ausgelöst werden.



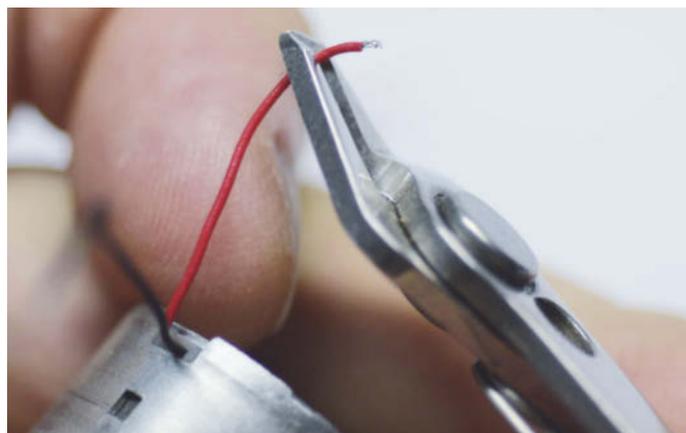
Wenn du deine Kabelverbindung mit Schrumpfschlauch schützt, musst du den Schlauch kurz erhitzen, damit er sich fest um die Kabel legt.



Lüsterklemmen sind zwar etwas klobig, halten aber sehr gut.



Zum Freilegen der Kabel kann man eine Abisolierzange ...



... oder einen Seitenschneider verwenden.

Bautipp: Vibrationsmotor bauen

Wenn du Motoren aus alten Elektrogeräten verwendest, handelt es sich dabei meist nicht um Vibrationsmotoren, sondern um einfache Elektromotoren ohne Unwucht. Du kannst sie in Vibrationsmotoren umbauen, wenn du eine Unwucht an der Motorwelle befestigst. Da die Welle ein dünner Metallstab ist und die Unwucht mit der ganzen Kraft des Motors bewegt wird, ist das etwas kompliziert. Auch hier kann eine Lüsterklemme gute Dienste leisten.

Eine Lüsterklemme hat ja zwei nebeneinander liegende Anschlüsse, in die man eigentlich Kabel schieben und sie mit den Schrauben befestigen soll. Wenn du einen dieser Anschlüsse über den Motorschaft schiebst und festschraubst, hast du automatisch ein Gewicht, das nicht mittig auf dem Motorschaft sitzt. Deine Unwucht ist fertig!

Oft ist die Motorachse nicht lang genug, um die Lüsterklemme mit Plastikummantelung sicher zu befestigen. In dem Fall musst du die Plastikhülle ganz entfernen oder zumindest vorne kürzen. Wenn du kleine Lüsterklemmen hast, kannst du auch gut eine zweireihige Klemme verwenden. Dann hast du eine sehr starke Unwucht.

Du kannst aber auch ein anderes Gewicht verwenden und es mit Draht, Kabelbindern und/oder Kleber befestigen. Das kann eine Schraube sein oder ein paar Münzen oder ein kleiner Stein. Sichere diesen Gegenstand gut!.

Die Befestigung mit einer Lüsterklemme ist recht sicher, aber auch hier sollte man einen Klecks Heißkleber oder Schraubensicherung verwenden, um die Schrauben gegen das Lösen durch die Vibration zu sichern. —esk



ACHTUNG

Die Motoren erreichen eine hohe Drehzahl! Teile, die sich von der Motorachse lösen, können mit hoher Geschwindigkeit durch die Gegend fliegen. Daher verbieten sich scharfkantige Gewichte von selbst. Außerdem solltest du besonders bei Tests eine Schutzbrille tragen.



Lüsterklemmen-Unwucht ohne Plastikhülle



Lüsterklemmen-Unwucht mit Plastikhülle

secIT by Heise

HANNOVER 2020

Seien Sie dabei und profitieren Sie als Besucher von neuesten IT-Security Trends, Produkten oder Software-Lösungen.

Fachvorträge auf
2 Bühnen

Ausstellungsfläche auf rund
3.400 m²

bereits am Vortag
Schulungsseminare

kostenfrei und fachbezogen
Partner-Workshops

kostenfrei und informativ
Partner-Expert-Talks

unabhängige
redaktionelle Workshops

HIGHLIGHT:

Krypto-Experte
aus den USA

Bruce Schneier

Workshop
und
Keynote

Die Partner der secIT Hannover



Weitere Informationen und Anmeldung unter

sec-it.heise.de

Der Treffpunkt für Security-Anwender und -Anbieter!

25. – 26. März 2020
Hannover

PROGRAMM-HIGHLIGHTS:

Podiumsdiskussion:

Offensive Cyberverteidigung alias „Hackback“ mit: Andreas Können (BfV), Manuel Atug (HiSolutions), Ninja Marnau (CISPA Saarland), Tobias Haar (LL. M., MBA Rechtsanwalt)

Comedy-Hacker Tobias Schrödel:

Was bei stern TV nicht gezeigt werden durfte.

Prof. Dr. Gina Wollinger

(Kriminologisches Forschungsinstitut Niedersachsen e.V.) zeigt die Ergebnisse der repräsentativen KFN-Sicherheitsstudie aus 5.000 befragten Unternehmen.

Cyber Risk Perception & Resilience –

Wie wir Risiken wahrnehmen und welche Risiken tatsächlich bestehen
// Volker Kozok (Referent BMVg - Bundeswehr)

Hack'n Secure – Wie hackt man IoT-Geräte?

// Sascha Herzog (Technischer Geschäftsführer CEO & CTO Nside) und Mirko Ross (Gründungsgesellschafter und CEO der digital worx GmbH)

Abschluss-Keynote:

Bruce Schneier, Krypto-Experte aus den USA



Veranstalter



organisiert von

© Copyright by Maker Media GmbH



Eventpartner



Die Riesenschachuhr

Wie man Kinder zum Schachspielen motiviert? Mit Bewegung!
Damit es noch mehr Spaß macht, haben wir außerdem eine
Raspi-gesteuerte Riesenuhr gebaut.

von Thomas Klaube



Unser Schachverein bietet in Stuttgart für Kinder ab fünf Jahren wöchentliche Trainings an. Manchmal ist es nicht leicht, insbesondere die Kleinsten sprichwörtlich „in Schach“ zu halten. Doch sobald wir unser Riesenschachfeld aufbauen, sind alle Kids gebannt – ganz anders als beim drögen Theorieunterricht. Daher spielen wir oft Blitzschach mit den Riesenfiguren. Wenn die Uhr ein paar Meter entfernt steht, kommt richtig Bewegung ins Spiel: Ziehen, laufen, drücken, laufen, ziehen, zwischendurch mal nachdenken. Bei Blitzschach spielt der Faktor Zeit eine entscheidende Rolle. Die Uhr muss jederzeit für beide Spieler gut sichtbar sein. Daher braucht man für Riesenschach auch eine Riesenschachuhr. Leider gibt es so etwas nicht zu kaufen, bleibt also nur: selbst bauen. So entstand eine Raspi-gesteuerte Uhr mit zwei Zeit-Anzeigen und großen Buttons.

Leider waren Flipdot-Anzeigen zu teuer, Gleiches gilt für 7-Segment-Anzeigen, die oft an Tankstellen verwendet werden. Daher haben wir zwei 20cm hohe 7-Segment-Anzeigen mit Sketchup entworfen und bei einem 3D-Druck-Service drucken lassen – eine für den Weiß- und eine für den Schwarzspieler. Die 3D-Druckmodelle stehen zum Download zur Verfügung. Als Leuchtmittel haben wir uns für die wenig verbreiteten SK6812-LED-Pixelstreifen entschieden. Sie haben die gleichen Abmessungen wie Neopixel, können über einen Raspberry Pi mit den gleichen Libraries angesteuert werden und verbrauchen weniger Strom.

Alle wichtigen Funktionen übernimmt der Raspi: die Ansteuerung der Displays, die Konfiguration der Betriebsmodi sowie die Überwachung der Riesentaster. Wir haben einen Raspberry Pi 2B benutzt. Geeignet sind auch neuere Pis, sogar ein Zero mit aufgelöteter GPIO-Pinleiste funktioniert. Die Verbindung zum Display und den beiden Riesentastern erfolgt via RJ45-Kabel. Zusätzlich benötigen wir noch vier Taster zum Einstellen der Uhr. Aus diesem Grund haben wir den Raspberry Pi in ein großes Gehäuse gepackt. Strom erhält das Projekt von einer Powerbank. Sie hat 24000mAh und drei USB-Ausgänge für die beiden Displays und den Raspberry Pi. Mit einer geladenen Powerbank kann man die Uhr gut sechs Stunden betreiben. Schließlich haben wir noch eine Halterung aus zwei schwarz lackierten Sperrholzplatten sowie ein Gestell aus zwei Sonnenschirmständern gebaut. Die Abmessungen aller Komponenten sind so gewählt, dass die Teile in einen mobilen Koffer passen, der mit Schaumstoff ausgekleidet ist. —rehu

► heise.de/-4550190

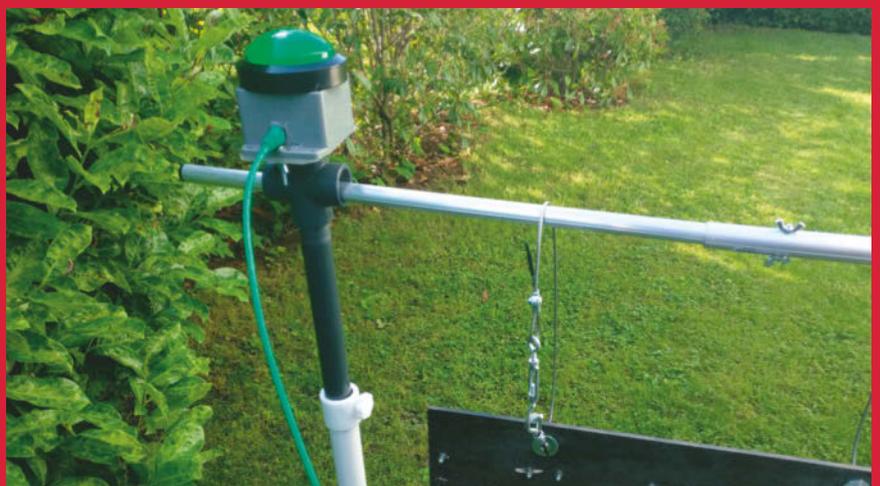
► sg-vaihingenrohr.de/2019/10/10/eine-schachuhr-fuers-freischach



Die 3D-gedruckten Module der 7-Segment-Anzeige werden auf Sperrholzplatten geschraubt.



Der Raspberry Pi und die Taster zum Stellen der Uhr sind in einem großen Gehäuse untergebracht.



Einmal auf den Riesentaster gedrückt, dann geht's los.

Alexa im Röhrenradio

Mit einem Amazon Echo Dot und ausgedienten
PC-Lautsprecherboxen kann man einem defekten
Röhrenradio neues Leben einhauchen.

von Wolfgang Ziegler



Sie verstauben in Kellern, in Abstellräumen oder auf Dachböden: Röhrenradios, wie sie einst die Wohnzimmer unserer Eltern und Großeltern schmückten. Unter Bastlern und Sammlern erfreuen sie sich nach wie vor großer Beliebtheit. Leider sind die Ersatzteile für kaputte Geräte, wie etwa Elektronenröhren, heute kaum noch im Handel erhältlich. Auch das Röhrenradio in diesem Projekt war lange Zeit bloßes Dekorationsobjekt im heimischen Wohnzimmer. Mangels kreativer Ideen und dank seines defekten Innenlebens fristete es ein stummes Dasein.

Ein Amazon Echo Dot, den ich aus einer Laune heraus erworben habe, sollte sich schließlich als Initialzündung für dieses Projekt erweisen. Als ich den Echo Dot erhalten habe, zeigte sich schnell, dass der eingebaute Lautsprecher denkbar ungeeignet für den Musikgenuss war. Aus diesem Grund habe ich ihn kurzerhand mit etwas betagten, aber klanglich noch einwandfreien PC-Lautsprechern (Creative Inspire 280) verbunden. Die Klangqualität war damit erträglich. Ästhetisch konnte man jedoch mit der Kombination aus Schreibtisch-Boxen und Echo-Dot im Wohnzimmerregal keinen Blumentopf gewinnen. Aber da gab es ja noch dieses wunderschöne Röhrenradio.

Dies war der perfekte Anwendungsfall, um dem stummen Röhrenradio neues Leben einzuhauchen. Zunächst dachte ich, ich könnte den eingebauten Lautsprecher des Röhrenradios statt der PC-Lautsprecher benutzen. Leider entsprach die Tonqualität nicht meinen Erwartungen (Mono-Signal und lautstarkes Knacken). Außerdem schien mir die Position des Lautsprechers als die ideale Stelle, um den Echo Dot im Röhrenradio einzubauen: Dieser wäre elegant hinter dem Stoffbezug des Radios verborgen und der farbige Leuchtring ließe sich bei Aktivierung dennoch gut ausmachen. Die Mikrofone wären nicht blockiert und Spracheingabe somit problemlos möglich. Kurzerhand habe ich also den eingebauten Lautsprecher entfernt. Anstelle des Lautsprechers habe ich den Echo Dot mit einer 3D-gedruckten Konstruktion direkt hinter dem Stoffbezug des Radios platziert. Die Optik war überzeugend – guten Sound hatte ich so allerdings immer noch nicht.

Also ging es den PC-Schreibtisch-Boxen ebenfalls ans Eingemachte. Ich habe ihren Verstärker und die Stromversorgung im Inneren des Röhrenradios untergebracht und sie wieder an den Echo Dot angeschlossen. Die eigentlichen Lautsprecher bekamen neue 3D-gedruckte Gehäuse. Diese fügen sich nun optisch besser im Wohnzimmerregal ein. Sie flankieren das Röhrenradio für angenehmen Stereoklang und Musikgenuss. —*rehu*

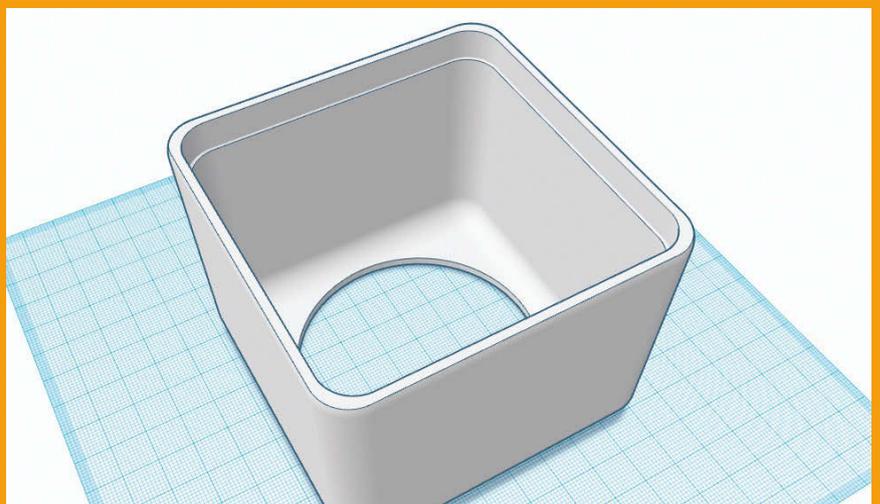
► wolfgang-ziegler.com/blog/alexatube-radio



So ist der Echo Dot im Innern des Röhrenradios befestigt.



Der Echo Dot leuchtet durch den Stoffbezug des Röhrenradios.



3D-Modell der neuen Lautsprecher-Gehäuse

LED-Uhr im Lego-Raumschiff

Mit einem LED-Set von eBay und einem Raspberry Pi lässt sich jedes Lego-Set in eine schicke Uhr verwandeln, die jede Viertelstunde die Zeit anzeigt.

von Tobias Hübner



Erinnern Sie sich an ihr erstes Lego-Set? Meins war ein Raumschiff, das Ende der 70er Jahre erschien und bis vor kurzem in Einzelteile zerlegt auf dem Dachboden lag. Kürzlich holte ich es wieder hervor. Einige Tage später sah ich, dass Drittanbieter bei eBay für viele Lego-Sets Bausteine mit winzig kleinen LEDs anbieten. Damit lässt sich nicht nur das Modell beleuchten – die LEDs lassen sich auch mit dem Einplatinenrechner Raspberry Pi ansteuern. Da ein Pi Zero in das Set hineinpasst, überlegte ich, mit den LEDs jede Viertelstunde die Zeit anzuzeigen, ähnlich wie das Läuten von Kirchenglocken.

Zur vollen Stunde leuchten nun alle LEDs auf, um 13 Uhr einmal, um 14 Uhr zweimal und so weiter. Zu jeder Viertelstunde zeigen die grünen LEDs zunächst die aktuelle, volle Stunde, danach leuchten die roten LEDs: einmal um viertel nach, zweimal um halb und dreimal um viertel vor. Die Sets werden für verschiedene Lego-Projekte angeboten, die auf Ebay über den Suchbegriff „Lego LED“ zu finden sind. Meist handelt es sich um weiße LEDs in farbigen, transparenten Lego-Steinen, so dass der Austausch in andere transparente Steine sehr einfach ist. Sie sind so klein, dass sie sich ohne Umbauten zwischen zwei Teile klemmen lassen. Auch die Kabel sind so dünn und robust, dass sie keinen Schaden nehmen. Für mein Projekt habe ich vier kleine LEDs für die Flügel bzw. Seiten des Raumschiffs sowie ein 3-LED-Stück für die Innenbeleuchtung verwendet.

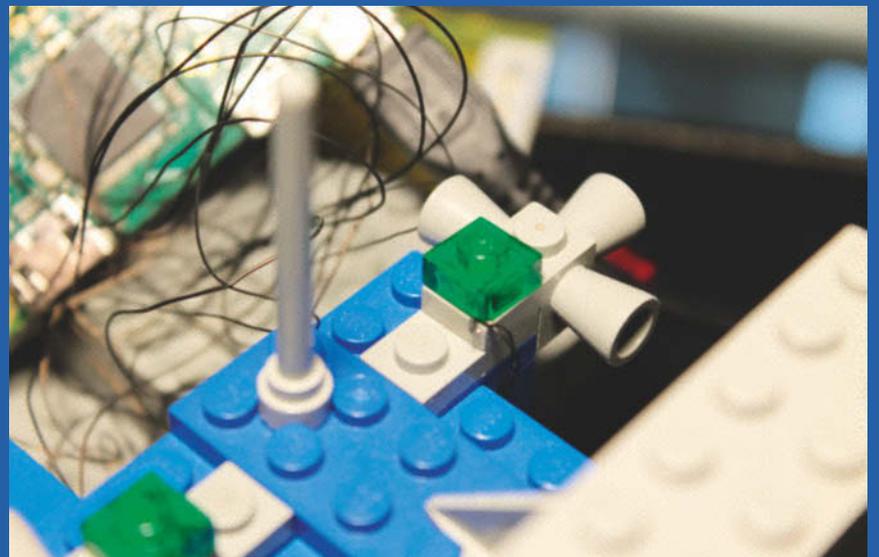
Zur Ansteuerung habe ich sie mit einem Pi verbunden. Am einfachsten geht dies mit dem Pi Zero W, der statt Shiftleisten 40 Löcher in der Platine besitzt, an die Kabel gelötet werden können. Außerdem unterstützt er WLAN, sodass sich die Uhr über das Internet aktualisieren lässt. Eine eingebaute Uhr haben Raspberry Pis nicht, nach jedem Neustart vergessen sie, wie spät es ist. Noch ein Tipp, da die Pole der LEDs meist nicht markiert sind: Um herauszufinden, welches Kabel zum Plus- und welches zum Minus-Pol führt, habe ich ein Stück der Ummantelung abisoliert und die Enden an die Pole einer Knopfzellen-Batterie gehalten. Leuchtet die LED, merkt man sich, welches Kabel an den Plus- und welches an den Minuspol gehalten wurde. Leuchtet sie nicht, wechselt man die beiden Kabel – keine Angst, die LED wird dabei nicht beschädigt.

Die Uhr habe ich schließlich in Python auf dem Raspberry programmiert. Dafür nutze ich die Bibliothek *datetime*. Darüber kann ich die aktuelle Uhrzeit abrufen und abspeichern. Mehr Erläuterungen zum Bau und zur Programmierung sowie den kompletten Code gibt es online. —hch

► heise.de/-4582951



Die LEDs sind winzig und passen in die Lego-Noppen.



Dank der dünnen Kabel lassen sich die LED-Steine einfach einbauen.



Der Raspberry Pi Zero W passt perfekt in mein Lego-Raumschiff.

Tipps & Tricks

Potigriffe aus LEDs, Beruhigung von Lüftern nicht nur in 3D-Druckern, platzsparende SD-Karten-Adapter und Ursachenforschung bei Schmorstellen in 3D-Gedrucktem sind diesmal Themen unserer Tippecke.

Achse für Spindeltrimmer

Ein Spindeltrimmer lässt sich viel präziser einstellen als ein herkömmliches Potentiometer. Meist sind zwischen 10 und 25 Gänge vorhanden, also volle Umdrehungen. Die Einstellung erfolgt über eine Stellschraube per Schraubendreher oder speziellem Abgleichstift, sodass man nicht mal eben ohne Werkzeug den Wert verändern kann. Dies kann einerseits von Vorteil sein, da so der Widerstandswert nicht unabsichtlich verändert werden kann, andererseits aber auch von Nachteil, wenn man mal eben den Wert ändern möchte und erst Werkzeug suchen muss.

Damit man den Widerstandswert auch ohne Werkzeug verstellen kann, kürzt man einfach die Anschlussdrähte einer 3mm-LED und lötet diese dann vorsichtig oben an die Verstellerschraube, schon hat man eine Drehachse mit farbigem Knopf. Dadurch kann die Einstellung auf dem Steckbrett bequemer und schneller erfolgen. Hinweis: Laut Datenblatt haben Spindeltrimmer meist nur eine spezifizizierte Lebensdauer von 200 Drehvorgängen.

Miguel Köhnlein und Michael Gaus

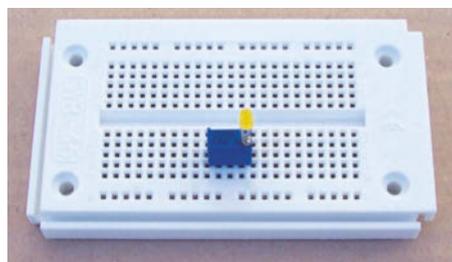
Creality Ender 3 beruhigen

In 3D-Druckern wie dem Creality Ender 3 sitzen eine ganze Reihe von Geräuschquellen. Da sind insbesondere die Schrittmotoren. Wie man die beruhigt, lesen Sie auf Seite 125. Sobald die Triebwerke schallgedämpft sind, fallen aber die Lüfter des Druckers unangenehm auf, denn deren Ansaugeräusche sind unangenehm schrill.

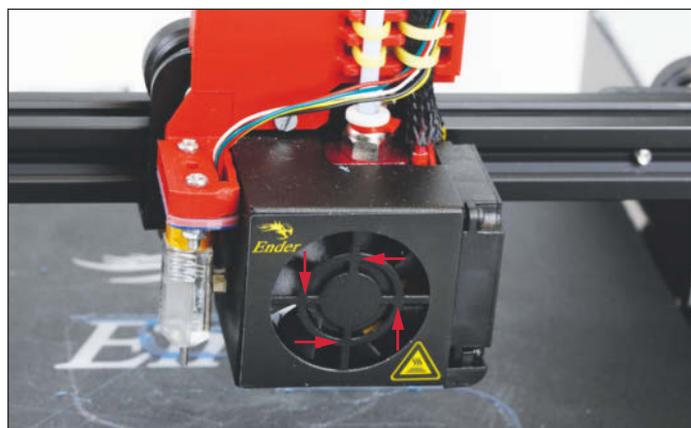
Ursache dafür ist das Design des Druckers: Die Lüfter am Druckkopf und am Deckel des Elektronikfaches sind mit kleinen Gittern ab-

gedeckt, die an jeweils vier Stellen vom Gehäuse getragen werden. Unmittelbar hinter dem Gitter dreht der Rotor des jeweiligen Lüfters. Sobald eine Kante eines Lüfterblattes hinter solch einem Blechsteg ankommt, kann es keine Luft mehr ansaugen, sondern erzeugt einen stärkeren Unterdruck. Tritt die Kante wieder hinter dem Steg hervor, verringert sich der Unterdruck wieder. Druckschwankungen in der Luft sind aber nichts anderes als Schall. Der siebenblättrige Rotor des Lüfters rotiert mit etwa 4000 Umdrehungen pro Minute. Bei vier Stegen am Gitter ergeben sich so 112 000 Luftdruckschwankungen pro Minute oder ein Geräusch mit einer Grundfrequenz von 1867Hz. Das erklärt den relativ schrillen Klang der Lüfter.

Diese Schallquelle ist aber einfach und sogar mit dem Bordwerkzeug des Druckers zu beseitigen: Schneiden Sie die Gitter mit dem Seitenschneider direkt an der Gehäusekante ab, sodass eine kreisrunde Öffnung übrig bleibt. Hantieren Sie dabei aber vorsichtig, um das Lüfterrad nicht zu beschädigen. Danach hat diese Lärmquelle schon einiges an Energie verloren und Sie erhalten einen großen Teil Ihrer Ruhe zurück. Übrigens gilt dies natürlich für alle ähnlichen



Eine LED als Drehknopf



Diese vier Stege kann man mit einem Seitenschneider durchtrennen. Achten Sie darauf, die Lüfterblätter nicht zu beschädigen.



Die selbstgedruckte Lüfterabdeckung des Netzteils lenkt die Schallabstrahlung nach hinten.

gestylten Luftöffnungen, nicht nur an 3D-Druckern.

Auch am Netzteil sitzt solch ein Gitter. Das Blech ist hier aber dicker, sodass man zum Entfernen eine Säge oder eine kleine Trennscheibe braucht. Außerdem muss das Netzteil geöffnet werden. Vorsicht: Im Inneren können auch bei gezogenem Netzstecker noch aufgeladene Kondensatoren hohe Spannungen führen! Dieser Lüfter ist aber auch dann noch deutlich hörbar. Mit einer selbstgedruckten Kühlluftführung (Link für die Druckdateien auf Thingiverse, siehe Link) kann man den Schall einfacher dämpfen und nach hinten abstrahlen lassen.

Eine noch etwas stärkere Beruhigung des Druckers erreicht man nur durch andere Lüfter. Im Internet gibt es eine Reihe entsprechender Angebote: Sie brauchen für Elektronik und Druckkopf Axial-Lüfter mit 40mm Breite und 10mm Höhe sowie 24V Versorgungsspannung. Achten Sie auf die Lautstärke: 18dB ist ein guter Wert für einen leisen Lüfter.

Heinz Behling

Schmorstellen beim 3D-Drucker

Falls Ihre 3D-gedruckten Objekte merkwürdige, braun verbrannte Fehlerstellen aufweisen, muss das nicht immer an einer zu hohen Drucktemperatur liegen. Oft ist lediglich die Druckdüse locker. Dann quillt geschmolzenes Filament am Gewinde der Düse vorbei nach außen. Während es die Düse hinabläuft, verkohlt es teilweise und wird so braun. Hat sich genug an der Düse gesammelt, tropft es auf das bereits Gedruckte herab und bildet dort die Fehlerstellen.

Die Abhilfe ist einfach: Heizen Sie den Drucker auf und ziehen Sie die Düse mit einem Schraubenschlüssel (liegt meist dem Bordwerkzeug bei) fest. Anschließend reinigen Sie die noch heiße Düse mit einer Messing-Drahtbürste von noch anhaftendem Kunststoff – fertig.

Heinz Behling

Gekürzter Micro-SD-Adapter für Raspberry Pi

Viele Bastler dürften noch einen Raspberry Pi der ersten Generation mit SD-Karte statt Micro-SD-Karte verwenden, schließlich wurde dieser in großen Stückzahlen verkauft. Auch als die neueren Modelle erschienen sind, dürften die wenigsten alten Pis entsorgt worden sein. Ein Nachteil dieser ersten Modelle ist, dass die SD-Speicherkarte



ziemlich weit über den Sockel hinausragt. Selbst wenn der Raspberry in ein passendes TEK-BERRY-Gehäuse eingebaut wird, ragt die Karte noch circa 1cm über die Außenkante des Gehäuses hinaus.

Als Alternative kann ein spezieller kurzer Micro-SD-Adapter verwendet werden, bei dem die Micro-SD-Karte um 90 Grad gedreht eingeschoben wird. Wird dieser Adapter in den SD-Kartenslot eingeschoben, so schließt er bündig mit der Außenkante des Gehäuses TEK-BERRY ab, sodass nichts übersteht. Durch eine Einkerbung an der Außenkante des Adapters kann man ihn auch leicht wieder mit dem Fingernagel aus dem Slot herausziehen.

Auch für Arduino-Shields mit SD-Slot eignet sich der Adapter, wenn es aus Platzgründen auf möglichst wenig Überstand der Karte ankommt. Erhältlich ist der Adapter zum Beispiel bei Watterott (siehe Link).

Miguel Köhnlein und Michael Gaus



Alles zum Artikel im Web unter make-magazin.de/xnk3

Machen Sie mit!

Bei der Gestaltung dieser Tipps- und Tricks-Seiten können Sie mitmachen. Kennen Sie auch einen raffinierten Trick? Wissen Sie, wie man etwas besonders einfach macht? Dann haben Sie die Chance, eine Make-Tasse und 50 Euro zu verdienen. Schicken Sie uns Ihren Tipp gleichgültig aus welchem Bereich (zum Beispiel

Raspberry, Arduino, 3D-Druck, Platinenherstellung, Lasercutting usw. ...). Wenn wir ihn veröffentlichen, erhalten Sie die versprochene Belohnung. Ihr Tipp

sollte aber nicht länger als eine Drittel Heftseite sein und maximal ein von Ihnen angefertigtes Bild enthalten. Senden Sie Ihren Tipp mit der Betreffzeile *Lesertipp* an:

► mail@make-magazin.de

... und Sie erhalten bei Veröffentlichung die Make-Tasse plus 50 Euro!



Sound-Glückwunschkarte

Der Geburtstagssaurier ist los in Form einer singenden Glückwunschkarte. Mit dem Soundmodul und dem elektrischen Unterkieferantrieb kann man aber auch nach der Party noch einiges anfangen.

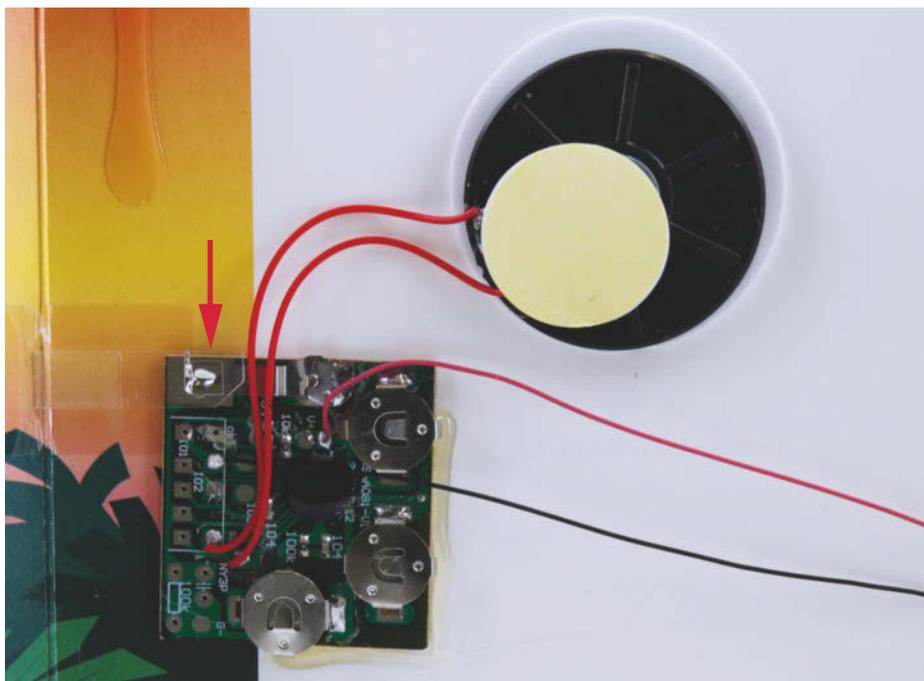
von Heinz Behling



Beim Kindergeburtstag war diese Karte sicher ein Knüller, denn aufgeklappt ertönt Musik und der Saurier trällert ein Geburtstagsständchen. Passend dazu bewegt sich sein Unterkiefer auf und ab. Bewirkt wird dies durch ein aus drei Knopfzellen gespeistes Soundmodul, das sich hinter dem Dinokopf im Karteninneren befindet. Im geschlossenen Zustand befindet sich eine Kunststoffolie zwischen zwei Kontakten auf dem Soundmodul und unterbricht so die Stromzufuhr. Beim Öffnen wird sie zur Seite bewegt und das Soundmodul eingeschaltet. Über den sehr flachen, aber umso lauter Lautsprecher ertönt die Musik, kurze Zeit später wird auch der Motor für den Unterkieferantrieb eingeschaltet. Er sitzt in einem ebenfalls sehr flachen Gehäuse und treibt über ein Schneckengetriebe ein großes Zahnrad mit Zapfen an. Auf diesem Zapfen sitzt der Unterkiefer des Dinos, der sich daher auf- und abbewegt.

Die Technik in der Karte ist viel zu gut, um in den Abfall zu wandern, sondern schreit geradezu nach neuen Aufgaben: Befestigt man zum Beispiel am Zapfen des Motors einen langen Faden, an dessen anderem Ende ein Stück Papier oder Stoff sitzt, und versorgt das Ganze über Batterie mit Strom, dann hat man ein ideales Spielzeug, um den Stubentiger längere Zeit zu beschäftigen.

Im Modellbau ist so ein Antrieb überall dort einsetzbar, wo es um Hin- und Herbewegung (beziehungsweise auf und ab) geht, beispielsweise bei der Betätigung von Weichen und Schranken auf einer Modellbahnanlage, fahrscheinwendenden Besatzungsmitgliedern auf Modellbooten und Ähnlichem. Auch der platzsparende, aber recht lautstarke Lautsprecher kann in diversen Modellen sicher angewendet werden.



Das Soundmodul wird durch einen Folienschalter beim Öffnen der Karte eingeschaltet.

Kinder können aus Soundmodul und Lautsprecher auch eine Alarmanlage für ihre Schatztruhe bauen: Dazu beides in die Truhe (oder Spardose, Schublade usw.) einbauen und den Folienschalter mit dem Deckel so verbinden, dass er beim Öffnen den Stromweg freigibt. Und schon plärrt der Saurier los und die Umgebung weiß Bescheid: Schatzräuber sind unterwegs.

Der Geburtstagsong passt zwar nicht ganz zum Diebstahlsalarm. Das Soundmodul enthält aber noch einige ungenutzte Anschlüsse. Wir konnten zwar im Internet keinerlei Dokumentation dazu finden. Aber

durch Ausprobieren müsste sich herausfinden lassen, ob sich darunter wie bei vielen anderen Modulen auch Anschlüsse für ein Mikrofon und für einen Aufnahmeschalter befinden. Das ergäbe dann noch wesentlich mehr Anwendungsmöglichkeiten. Falls Sie diesbezüglich etwas herausfinden, teilen Sie Ihr Wissen doch bitte im Heftforum zu diesem Artikel (Adresse siehe Link). —hgb

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



Hinter dem Unterkiefer sitzt ein flacher Motor mit Schneckengetriebe.

Touchscreen GUI mit dem Raspberry

Touchscreens sind perfekt, um Programme mit einer grafischen Oberfläche unkompliziert zu bedienen. Ein eigenes Raspberry-Pi-Projekt ist die Gelegenheit, um ein personalisiertes Touch-Interface zu entwickeln. Hier zeigen wir, wie es geht.

von Carsten Romahn



Eine der vielen Möglichkeiten, eine grafische Benutzeroberfläche (GUI) zu gestalten, stelle ich hier vor. Ein Hinweis vorab: Die Aussage „Eine der vielen Möglichkeiten“ gilt für die meisten Punkte auf den nächsten Seiten, da es nicht *den einen* perfekten Weg gibt. Die Qual der Wahl beginnt schon bei der Auswahl der Hardware. Statte ich meinen Raspberry Pi mit einem eigenen Touchscreen aus? Steuere ich ihn über ein Webinterface oder über eine eigene App auf dem Smartphone? Ein Touchscreen hat den Vorteil, dass man den Raspberry Pi unabhängig von zusätzlicher Hard- und Software bedienen kann. Eine App ist charmant: Sie macht nicht nur zusätzliche Hardware überflüssig, Geräte wie Smartphones und Tablets sind auf Touchsteuerung optimiert.

Die Touchscreen-Modelle

Entscheidet man sich für ein Display, ergibt sich schon die nächste Frage: Genügt ein resistiver Touchscreen? Oder ist ein kapazitiver Touchscreen geeigneter? Einfache resistive Touchscreens mit 3,5 Zoll sind schon für unter 20 Euro zu bekommen. Außerdem kann man sie gut mit einem Stift bedienen. Bei der Steuerung mit den Fingern machen sie allerdings nur mit den einfachsten Bedienelementen Freude – kommen als Steuerelemente Gesten wie *Wischen* oder *Ziehen* ins Spiel, ist ein kapazitives Display die bessere Wahl.

Die Betriebssysteme

Darauf folgt die Frage, welches Betriebssystem auf dem Raspberry laufen soll. Die Entscheidung für eine Linux-Distribution wie Raspbian bietet die beste Hardware-Unterstützung und ist unter den Pi-Nutzern am weitesten verbreitet. Windows 10 IoT Core ist allerdings eine interessante Alternative, besonders wenn man mit Cloud- und Server-

Kurzinfo

- » Touchscreen für den Raspberry Pi programmieren
- » Grafische Oberflächen mit Qt gestalten
- » Manuelle PIN-Steuerung mit ComboBox



diensten von Microsoft arbeiten möchte. Für dieses Projekt entscheide ich mich für Raspbian, da ich die Software direkt auf dem Raspberry Pi entwickeln will – das wäre unter Windows 10 IoT nicht möglich.

IDE und GUI

Jetzt steht „nur noch“ die Wahl der Programmiersprache, des GUI-Frameworks und der Entwicklungsumgebung (IDE) an. Es gibt eine unglaublich große Anzahl an verschiedenen GUI-Toolkits. Allerdings sind viele von ihnen auf die Nutzung mit der Maus optimiert. Die interessantesten Frameworks für die Bedienung mit den Fingern sind *Kivy* und Qt's *Quick Controls*. Beide Frameworks sind nicht auf ein bestimmtes Betriebssystem beschränkt und können auf allen gängigen Plattformen genutzt werden. Beide bringen eine Open-Source-Lizenz mit, wobei für Qt auch eine kommerzielle Lizenz erhältlich ist. Kivy programmiert man in Python und Qt in C++. Für Qt gibt es allerdings zusätzliche Bindings zu einer Vielzahl von verschiedenen Sprachen. So kann man Qt durch das PySide2-Binding auch mit Python verwenden. Meine Entscheidung fällt auf Qt, da mit QtCreator eine recht komfortable Entwicklungsumgebung zur Verfügung steht. Wer nicht direkt auf dem Raspberry Pi entwickeln

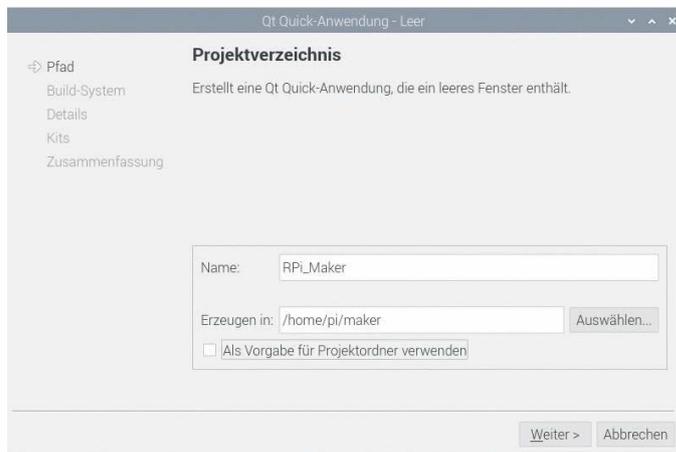
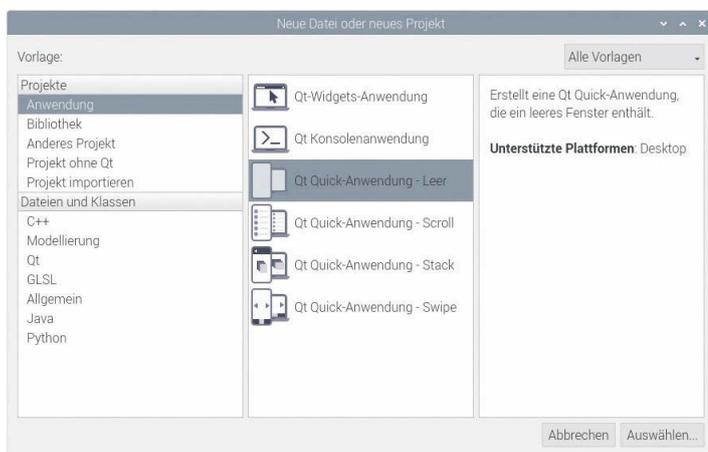
möchte, für den bietet sich das sogenannte Cross-Compiling an. Hierbei wird das Programm auf einem vollwertigen Linux Rechner entwickelt und kompiliert. Das fertige Programm überträgt man dann auf den Raspberry Pi. Eine gute Anleitung gibt es unter den Download-Links in der Kurzinfo.

Installation von Qt auf dem Raspberry Pi

Der einfachste Weg zur Installation von Qt auf dem Raspberry Pi geht über den Feed:

```
sudo apt-get install qt5-default
qtcreator qtdeclarative5-dev
qtquick5* qml-module*
```

Touchscreens eignen sich gut zur Steuerung von Programmen, zur Entwicklung dieser sind sie allerdings zu klein. Eine Lösungsmöglichkeit für dieses Problem wäre ein zusätzlicher Monitor am Raspberry Pi. Allerdings unterstützt der Raspberry Pi bis zur Version 3 immer nur einen Bildschirm, daher müsste man den Pi für jeden Wechsel zwischen Monitor und Touchscreen neu starten. Praktischer ist es, wenn man zusätzlich einen Server für das Remote Desktop Protocol installiert. Über die Serveranwendung lässt sich der Desktop des Minirechners dann auf jeden anderen Computer streamen. Man installiert den Server folgendermaßen:

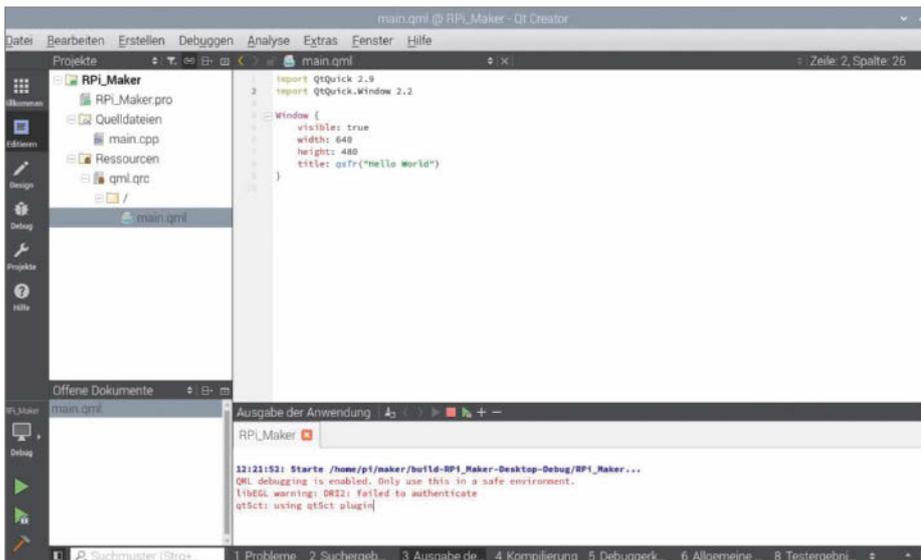


1 Die grafische Oberfläche wird als Quick-Anwendung angelegt.

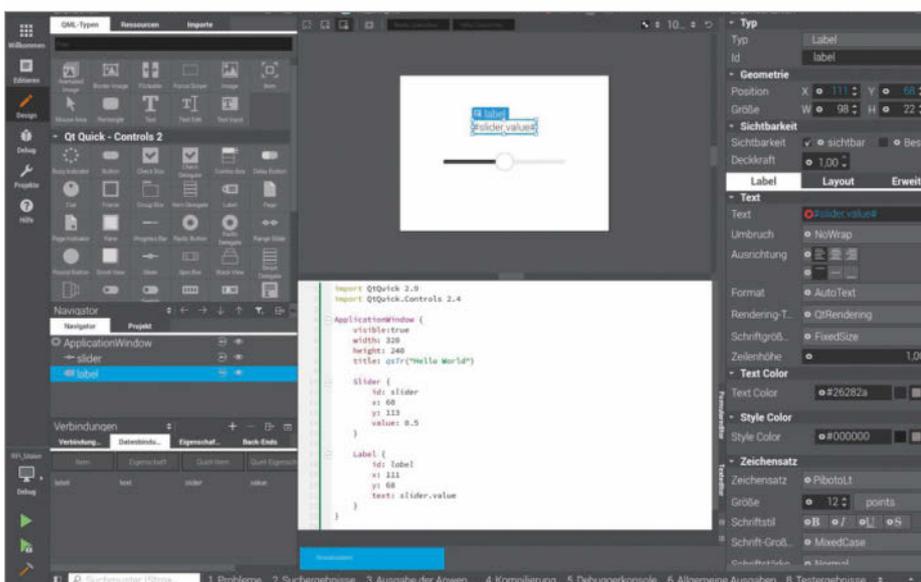
2 Man beginnt mit einer leeren Quick-Anwendung.



3 Man wird von einem klassischen „Hello World“ begrüßt.



4 Die Ausgabe funktioniert schon mal.



5 Mit dem Designer kann man die grafische Oberfläche auch per Drag & Drop gestalten.

```
sudo apt-get install xrdp
```

Die passenden Clients gibt es für Windows (vorinstalliert: Remotedesktopverbindung), Linux (RDesktop) und für MacOS, Android und Apple iOS ist es der Microsoft Remote Desktop.

Los geht's

Im QtCreator lege ich ein neues Projekt (Datei->Neu...) an. Im Popup-Menü entscheide ich mich für eine leere Qt Quick-An-

wendung 1. Quick steht für Qt User Interface Creation Kit. Es ist die Basis für Steuerelemente, die auf Steuerung durch Touch optimiert sind. Im folgenden Fenster lege ich nur noch einen Namen und ein Verzeichnis fest, dann klicke ich in allen folgenden Fenstern weiter 2.

Qt erstellt nun ein fast leeres Projekt, das ich mit Erstellen->Ausführen starte 3 und daraufhin mit einem leeren Fenster mit dem Titel „Hello World“ belohnt werde 4.

Ein Blick in den Projektexplorer zeigt, dass Qt das Programm in zwei logische Bereiche

aufteilt. Der imperative Teil befindet sich in main.cpp und definiert die eigentliche Anwendung in C++. Die Benutzeroberfläche selbst wird dafür im deklarativen Teil in der Sprache QML beschrieben und enthält keinerlei C++. Diese Trennung erlaubt es, auch dann denselben QML-Code weiterzuverwenden, wenn man im Hauptteil die Programmiersprache beispielsweise von C++ nach Python ändert. Die Bereiche von C++ und QML können komfortabel mit dem integrierten Editor bearbeitet werden, da dieser mit einer Syntaxhervorhebung und Autovervollständigung aufwartet. Beim QML-Teil geht es sogar noch bequemer: Mit dem Designer lassen sich Bedienflächen einfach mittels Drag & Drop zusammenstellen und anpassen. Der Designer erzeugt dabei recht übersichtlichen Code 5, den man nach Belieben im Editor bearbeiten kann.

Leider stößt ein Raspberry Model 3B hier an seine Grenzen, die Bedienung des Designer-Features wird sehr zäh. Allerdings ist – mit etwas Übung – die Erstellung der GUI-Elemente mit dem Editor eh schneller. So sind in 5 lediglich acht Zeilen notwendig, um einen Slider und ein Label zu erzeugen und beide auf der Benutzeroberfläche zu positionieren. Gleichzeitig wird eine Datenverbindung etabliert: Sie sorgt dafür, dass jede Änderung der Eigenschaft value des Steuerelements slider den Text des Labels automatisch neu setzt. Datenverbindungen gehören zu den mächtigsten Elementen von Qt Quick. Sie erleichtern die Erstellung von dynamischen Benutzeroberflächen erheblich. Zudem kann man in den Datenverbindungen auch JavaScript-Ausdrücke benutzen. So wäre in 5 auch der Ausdruck text: if(myslider.value>0.5){ "High" }else { "Low" } und dessen Kurzform text:(myslider.value>0.5)? "High": "Low" gültig.

Signal/Slot-System in Qt

Ähnlich komfortabel kann man Qts zentrales Signal/Slot-System nutzen. Signale sind „Botschaften“, die bei Eintreten eines Ereignisses abgegeben werden. Ein Slot ist eine normale Funktion, die man mit einem Signal verknüpfen kann. Jedes Mal, wenn das Signal abgegeben wird, wird anschließend der verbundene Slot aufgerufen. In QML stehen viele Signale zur Verfügung. Neben clicked ist <Eigenschaft>Changed eines der wichtigsten Signale, das ausgelöst wird, sobald sich eine bestimmte Eigenschaft verändert. Mit on<Signal>: kann man einen Signal-Handler einrichten, der JavaScript-Code ausführt. In 6 könnte man die Datenverbindung auch durch eine Signal/Handler-Verbindung ersetzen. Ein wichtiger Unterschied ist, dass in 6 die Datenverbindung zwischen Slider

6 Listing 1

```

1 Slider{
2     id:myslider
3 }
4 Label{
5     id:label
6     text:myslider.value
7 }
8 Button{
9     onClicked:label.text="Stopp!"
10 }

```

7 Listing 2

```

1 Slider{
2     id:myslider
3     onChangeed:label.text=value
4 }
5 Label{
6     id:label
7     text:"Noch nichts passiert!"
8 }
9 Button{
10    onClicked:label.text="Stopp!"

```

und Label unterbrochen wird, sobald man den Button drückt und dem Label den Text „Stopp!“ zuweist. In **7** ist die Verbindung robuster. Der Button kann zwar den Text des Labels ändern, aber sobald sich die Eigenschaft `value` ändert, wird wieder der Wert des Sliders im Label angezeigt.

Praktisches Beispiel

Da nun das Handwerkszeug bereitsteht, geht es endlich an die Arbeit. Ich möchte eine modulare Benutzeroberfläche erstellen, über die man mit den GPIOs interagieren kann. Im ersten Schritt will ich zwei Module realisieren. Mit Modul 1 sollen sich die GPIOs ein- und ausschalten lassen, zudem soll deren Status angezeigt werden. Modul 2 soll dazu dienen, Servos zu testen. Die Oberfläche will ich im Querformat realisieren und in einen Navigations- und einen Arbeitsbereich aufteilen **8**.

Da die Anwendung im Vollbildmodus laufen soll, beginne ich mit dem Ausschalter – ansonsten könnte man die Anwendung nur mit einer Tastatur schließen. Für den Schalter scheint das Steuerelement `Button` eine gute Lösung. Allerdings nutze ich eine weitere Stärke von Qt und erstelle mein eigenes Steuerelement. So habe ich die volle Kontrolle über das Design. Ausgangspunkt ist das Bild **9** **11**, über das ich eine `MouseArea` lege. Bevor ich **9** „Power.png“ nutzen kann, muss ich das Bild dem Projekt als Ressource hinzufügen. Dazu mache ich einen Rechtsklick auf `qml.qrc` und wähle „Hinzufügen existierende Datei...“. Zur Positionierung nutze ich das relative Anchorsystem (siehe Kasten „Positionierung“).

Den Mauszeiger der Anwendung verstecke ich, indem ich über die Anwendung eine deaktivierte `MouseArea` lege und den Mauszeiger mit `cursorShape: Qt.BlankCursor` ausblende **10**.

Nun geht es an den Navigations- und Arbeitsbereich: Ich will eine Anwendung erstellen, bei der man zwischen mehreren Seiten hin- und herwechseln kann. Qt bietet dazu eine Reihe verschiedener Möglichkeiten. Möchte man die Navigationsleisten an der

10 Listing 3

```

1 import QtQuick2.7
2 import QtQuick.Controls2.4
3
4 ApplicationWindow {
5     id:root
6     visible:true
7     visibility:"FullScreen"
8
9     Image {
10        id:powerButton
11        anchors.bottom:parent.bottom
12        anchors.left:parent.left
13        anchors.margins:20
14        width:parent.width*0.2
15        height:width
16        source:"img/Power.png"
17        fillMode:Image.PreserveAspectFit
18        MouseArea{
19            anchors.fill:parent
20            onClicked:Qt.quit()
21        }
22    }
23    MouseArea {
24        anchors.fill:parent
25        enabled:false
26        cursorShape:Qt.BlankCursor
27    }
28 }

```



8 Nach diesem Schema gestaltet man die grafische Oberfläche.



9 Für den Ausschalt-Button kann man einfach eine Grafik einbinden.

Die vier Basissysteme zur Positionierung

QML bietet vier Basissysteme zur Positionierung von GUI-Elementen, die sich in weiten Teilen kombinieren lassen: Zunächst gibt es die **manuelle Positionierung** über die Koordinaten `x` und `y`. Die Größe legt man mit `height` und `width` fest. Beim Positionieren mit **Ankern** werden die Kanten (links, rechts, oben, unten) eines Elementes an den Kanten oder dem Mittelpunkt des Eltern- und eines Geschwisterelementes festgelegt. Mit **Positionierern** (Column, Row, Grid und Flow) werden die Steuerelemente automatisch positioniert, in der Reihen-

folge ihrer QML-Datei. Hierbei legt man nur die Koordinaten durch die Positionierer fest. Zuletzt funktionieren **Layouts** (ColumnLayout, GridLayout, RowLayout, StackLayout) ähnlich wie Positionierer und sind besonders interessant für Anwendungen, deren Größe nicht festgelegt ist. Für jedes Element kann man eine minimale, eine maximale und eine bevorzugte Größe festlegen. Entsprechend dem zur Verfügung stehenden Platz weist das Layout dann jedem Element automatisch seine Position und Größe zu.

12 Listing 4

```

1 ListModel {
2     id: lModel_pages
3     ListElement{ titel:"GPIO"; source:"PageGPIO.qml"; img_source:"img/io.png"}
4     ListElement{ titel:"Chart"; source:"PageChart.qml"; img_source:"img/chart.png"}
5     ListElement{ titel:"Servo"; source:"PageServo.qml"; img_source:"img/servo.png"}
6     ListElement{ titel:"Settings";source:"PageSettings.qml";img_source='img/setting.png'}
7 }

```



11 Die grafische Oberfläche bekommt zuerst einen Ausschalter.

Ober- oder Unterkante positionieren, dann ist eine Umsetzung mit einer *TabBar* in Kombination mit einem *StackLayout* oder *SwipeView* eine gute Option. Diese kann man über eine Datenverbindung kombinieren. Ich entscheide mich für *ListView* für den Arbeitsbereich und den Navigationsbereich. Ein *ListView* wird im Wesentlichen durch zwei Eigenschaften definiert. Mit `model:` wird festge-

legt, was, und mit `delegate:`, wie es dargestellt werden soll. Die mit `delegate:` festgelegte Komponente kann jedes beliebige Steuerelement beinhalten. Für den Navigationsbereich bieten sich Buttons an. Die Eigenschaft `model:` erwartet ein Array, wobei für jedes Element im Array eine unter `delegate:` definierte Komponente erzeugt und dargestellt wird. So kann ich mit `model:6` und `delegate: Button{text:index}` sechs Knöpfe erstellen, die von 0 bis 5 beschriftet sind. Für mein Programm möchte ich `delegate:` für jedes Element nicht nur eine Zahl, sondern noch ein paar zusätzlich Informationen mitgeben. Mit *ListModel* lässt sich ein Array erstellen, das jeder Komponente ein *ListElement* mitgibt. Ein *ListElement* ist eine einfache Liste, wobei jedes Element aus dem Paar `role: data` besteht. Hört sich alles etwas umständlich an – wie man am Code [12](#) sieht, ist es aber gar nicht so kompliziert.

13 Listing 5

```

1 ListView{
2     id:screen_buttons
3     model:lModel_pages
4     anchors.top:parent.top
5     anchors.bottom: powerButton.top
6     anchors.left:powerButton.left
7     anchors.right:powerButton.right
8     snapMode: ListView.SnapToItem
9     orientation:Qt.Vertical
10    delegate: Button {
11        width:parent.width
12        text: model.titel
13        flat:screens.currentIndex == index?false:true
14        icon.source:img_source
15        onClicked: screens.currentIndex = index;
16    }
17 }

```

Im Navigationsbereich [13](#) bekommt jeder Button einen Titel und gegebenenfalls ein Image vom Model mitgeteilt. Insgesamt entstehen so vier Knöpfe, die jeweils einen Slot `onClicked` bekommen. Zudem wird für die Eigenschaft `flat:` eine Datenverbindung zu dem *ListView* `screens` erstellt. Sie sorgt dafür, dass der Knopf der ausgewählten Seite optisch hervorgehoben wird. Im Arbeitsbereich [14](#) wird mit dem Element Loader für jede Komponente dynamisch eine QML-Datei geladen. Mit diesem Konstrukt können nun beliebig viele Seiten eingebunden werden. Für jede neue Seite ist lediglich ein Eintrag im *ListModel* notwendig. Allerdings müssen die Seiten dem Ressourcensystem bekannt sein (Datei->Neu...->Qt/QML-Datei). Man kann mit der Navigationsleiste zwischen den Seiten wechseln [15](#). Da *ListView* auf einem sogenannten *Flickable* basiert, ist es auch möglich, mittels „wischen“ zwischen den Seiten zu navigieren. Sollte im Navigationsbereich der Platz nicht zur Anzeige aller Buttons reichen, kann man die Ansicht einfach mit dem Finger scrollen. [14](#)

14 Listing 6

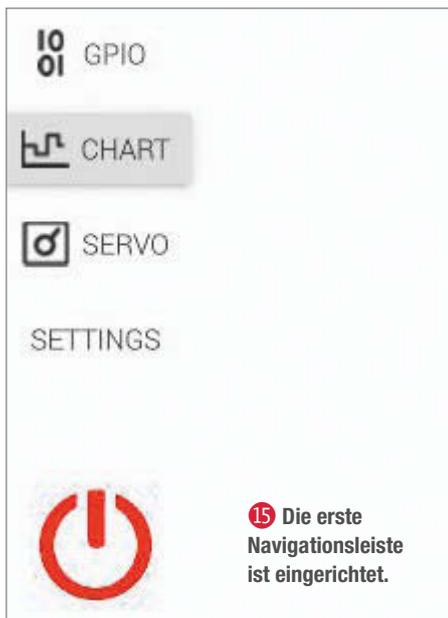
```

1 ListView{
2     id:screens
3     model:lModel_pages
4     anchors.right:parent.right
5     anchors.top: parent.top
6     anchors.bottom :parent.bottom
7     anchors.left:screen_buttons.right
8     anchors.margins: 5
9     highlightMoveDuration: 0
10    snapMode: ListView.SnapToItem
11    orientation:Qt.Vertical
12    highlightRangeMode: ListView.StrictlyEnforceRange
13    delegate: Loader {
14        width: screens.width
15        height: screens.height
16        source: model.source
17    }
18 }

```

Unter der Haube

Jetzt will ich den einzelnen Modulen Leben einhauchen. Dazu benötige ich – neben der Benutzeroberfläche – eine Schnittstelle, die



15 Die erste Navigationsleiste ist eingerichtet.

die Hardware ansteuert. Hierfür verwende ich die bereits auf dem Raspberry installierte Bibliothek *WiringPi*. Um die Bibliothek nutzen zu können, ist es notwendig, sie mit LIBS += L/usr/local/lib -lwiringpi in der

16 Listing 7

```

1 ApplicationWindow {
2 ...
3     signal digitalWrite(int pin, int value)
4     signal digitalRead(int pin)
5     signal pinMode(int pin,int mode)
6     property int value
7 ...}
    
```

Projektdatei bekannt zu machen. Nun kann man die Bibliothek mit `#include wiringPi.h` im C++ Bereich einbinden. Es ist nicht möglich, direkt aus dem QML-Bereich auf die Bibliothek zuzugreifen. Für die Kommunikation zwischen QML und C++ gibt es mehrere Möglichkeiten: Die primitivste Lösung wäre, in `main.qml` ein Signal innerhalb von `ApplicationWindow` zu definieren und dieses Signal nach der Qt-typischen Slot/Signal-Mechanik an *WiringPi* über eine eigene Klasse weiterzugeben. Die Rückgabewerte von *WiringPi* würden dann als Property im QML-Bereich zur Verfügung gestellt. Das Ganze sähe dann so aus: 16 17 18.

Deutlich eleganter ist der Ansatz, in C++ eine Klasse zu erstellen, die in QML Eigen-

schaften und Slots zur Verfügung stellt. Hierfür registriert man die Klasse mit `qmlRegisterType<myGPIO>(Wire, 1,0, GPIO);` in `main.cpp`. Dann kann man sie in QML-Dateien mit `import Wire 1.0` importieren und als GPIO-Element in den Code einbinden. Aber eins nach dem anderen: Zunächst erweitere ich das Projekt um eine leere Klasse (`Datei->Neu.;` C++Klasse). Im folgenden Menü lege ich den Klassennamen fest und wähle *QObject* als Basisklasse aus 19.

Qt erstellt daraufhin eine leere Klasse mit Header und Source-Datei und macht sie in der Projektdatei bekannt. Für die Kommunikation zwischen QML und C++ ist es notwendig, die Schnittstelle zu beschreiben. Hierfür definiere ich mit dem Makro `Q_PROPERTY`,



< Komed, Köln / 4. Februar 2020 >

Die Konferenz für Frontend-Entwicklung • WORKSHOP-PROGRAMM (AUSZUG)

Schwarze Magie in TypeScript: Deep Dive (Peter Kröner)

- > Die dunklen Ecken von TypeScript
- > Fortgeschrittene Features des Typsystems
- > Praktisches Formulieren fortgeschrittener Typ-Konstruktionen wie Mapped Types und Conditional Typs
- > Praktische Lösungen für häufige Type-Script-Probleme



Peter Kröner

Build a production ready PWA with Angular and Firebase (Önder Ceylan)

This workshop will focus on:

- > State of the PWA development
- > Displaying Add to Home Screen guidelines
- > Configuring & Extending Angular service worker
- > Hosting your PWA on Firebase
- > Auditing your PWA with Lighthouse



Önder Ceylan

CSS (endlich) lieben lernen (Nico König)

- > CSS als eine Zutat für Web-Apps
- > Weiterentwicklung mit Flexbox, Grid und Custom Properties
- > Die wichtigsten Konzepte und Features von CCS
- > Umsetzung in handliche Katas



Nico König

15 \$
16
17
18
19
20
21 \$
22
23
24
25
26 \$
27
28
29
30
31 #de
32 win
33 \$
34 \$
35 \$
36 \$
37
38 i
39
40
41

17 Listing 8

```

1 int main(int argc, char *argv[]){
2 ...
3 QObject *fenster =engine.rootObjects()[0];
4 MyGPIO myGPIO(fenster);
5 QObject::connect(fenster, SIGNAL(digitalWrite(int,int)),&myGPIO, SLOT(_digitalWrite(int,int)));
6 QObject::connect(fenster, SIGNAL(digitalRead(int)),&myGPIO, SLOT(_digitalRead(int)));
7 QObject::connect(fenster, SIGNAL(pinMode(int,int)),&myGPIO, SLOT(_pinMode(int,int)));
8 ...}

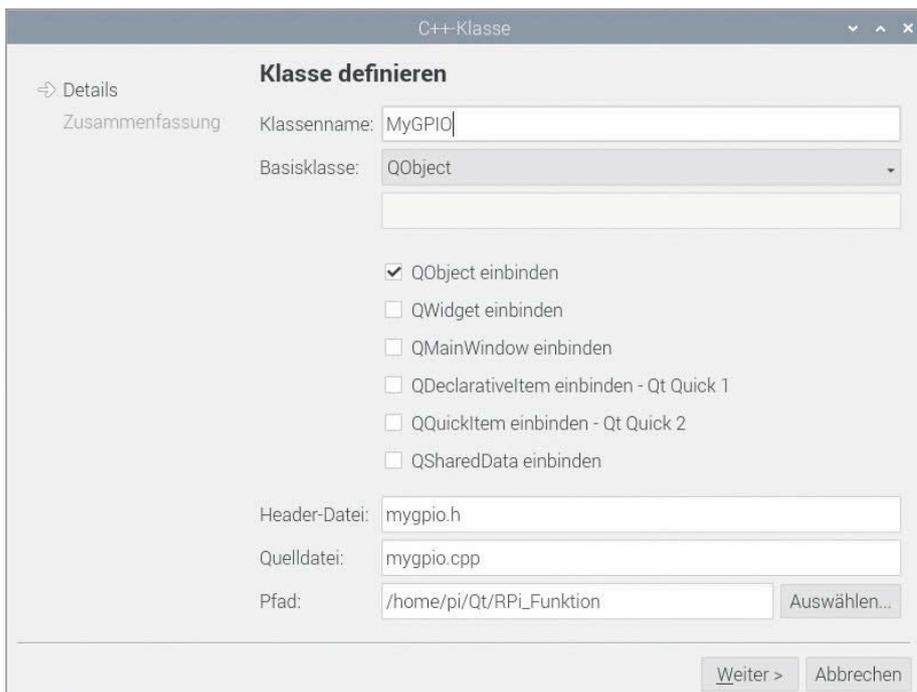
```

18 Listing 9

```

1 class MyGPIO: public QObject
2 {
3     Q_OBJECT
4     QObject *fenster;
5 public:
6     MyGPIO(QObject *ptr_fenster) {fenster=ptr_fenster; wiringPiSetupGpio(); }
7 public slots:
8     void _pinMode(int pin, int mode){pinMode(pin,mode);}
9     void _digitalWrite(int pin,int value){ digitalWrite(pin,value);}
10    void _digitalRead(int pin){fenster->setProperty ("value",digitalRead (pin));}
11 };

```



19 Man kann in C++ Klassen erstellen, die man in QML-Dateien einbinden kann.

20 Listing 10

```

1 class myGPIO: public QObject
2 {
3     Q_OBJECT
4     Q_PROPERTY( int pin READ pin WRITE setPin NOTIFY pinChanged )
5     Q_PROPERTY( int mode READ mode WRITE setMode NOTIFY modeChanged )
6     Q_PROPERTY( int value READ value NOTIFY valueChanged )
7     Q_PROPERTY( int state READ state WRITE setState NOTIFY stateChanged )

```

21 Listing 11

```

1 public:
2     int pin() const;
3     void setPin(const int &);
4 signals:
5     void pinChanged()

```

welche Eigenschaften QML mit welchem Datentyp zur Verfügung gestellt und wie auf sie zugegriffen wird. Im Code 20 wird zum Beispiel die Eigenschaft `pin` vom Typ Integer bekannt gemacht. Hinter `READ` und `WRITE` stehen die Namen der Funktionen, mit denen QML die Eigenschaft lesen und schreiben kann.

Nur `READ` ist verpflichtend. Hinter `NOTIFY` ist das Signal definiert, das bei einer Änderung von `pin` gesendet wird. Das Signal muss man nur dann definieren, wenn für diese Eigenschaft eine Datenverbindung in QML möglich sein soll. Die Funktionsprototypen für die Eigenschaft `pin` sieht man in 21. Neben den Eigenschaften kann man auch Aufzählungen bekannt machen. Hierfür steht das Makro `Q_ENUMS()` zur Verfügung 22.

Allerdings stellt *WiringPi* keine Callback-Funktion zur Verfügung, die auf Änderungen des logischen Zustands eines Pins reagiert. Daher ist es notwendig, den Zustand immer wieder abzufragen. Hierfür ergänze ich das `GPIO()`-Element um einen Timer 24, der in einem festgelegten Intervall den öffentli-

22 Listing 12

```

1 Q_ENUMS(mode)
2 Q_ENUMS(state)
3 public:
4     enum mode { Input=0, Output=1, PWM_OUTPUT=2 };
5     enum state{ Low=0, High=2};

```

23 Listing 13

```

1 GPIO{
2     pin:21
3     mode: GPIO.Output
4     state: GPIO.High
5 }

```

Nachdem die Funktionen implementiert sind, kann man die GPIO-Pins komfortabel in QML ansprechen. Um zum Beispiel GPIO21 als Ausgang festzulegen und einzuschalten, ist nur **23** notwendig. Auf ähnliche Weise kann man auch den aktuellen Zustand eines Pins abfragen.

24 Listing 14

```

1 GPIO{
2     pin:21
3     mode: GPIO.Input
4     onValueChanged: console.debug(value)
5     Timer {
6         interval: 100
7         repeat: true
8         running:gpio.pin?true:false
9         onTriggered: gpio.updateValue()
10    }
11 }

```

25 Listing 15

```

1 void myGPIO::updateValue(){
2     if (m_value!=digitalRead (m_pin)){
3         m_value=digitalRead (m_pin);
4         emit valueChanged();
5     }
6 }

```

26 Listing 16

```

1ListModel {
2     id:ListModel_GPIO
3     ListElement {key: "Leer"; pos:0; pin:0; reserved: false; type:"nothing"}
4     ListElement {key: "GPIO 2"; pos:3; pin:2; reserved: false; type:"I2C"}
5     ListElement {key: "GPIO 3"; pos:5; pin:3; reserved: false; type:"I2C"}

```

27 Listing 17

```

1ComboBox{
2     model:ListModel_GPIO
3     textRole: "key"
4     property int pin:ListModel_GPIO.get(currentIndex).pin
5     property int lastIndex :0
6     property string filter:""
7     signal add()
8     signal remove()
9     delegate:ItemDelegate {
10        id:delegateItem
11        width: controlPin.width
12        height: visible?controlPin.height:0
13        visible: (filter =="" || type=="nothing"|| type===filter)?true:false
14        contentItem:Text {
15            text:key+" ["+pos+"]";
16            anchors.fill: parent
17            anchors.leftMargin: 10
18            verticalAlignment: Text.AlignVCenter
19            font.pixelSize: parent.font.pixelSize
20            color: (reserved==1&&pin>0)?Material.accent:Material.foreground
21        }
22    }
23    Component.onDestruction:ListModel_GPIO.get(lastIndex).reserved = false
24    onCurrentIndexChanged: {
25        if (ListModel_GPIO.get(currentIndex).reserved=== true)currentIndex=0
26        ListModel_GPIO.get(lastIndex).reserved= false
27        ListModel_GPIO.get(currentIndex).reserved= true
28        if (lastIndex==0 && currentIndex>0) add()
29        if (currentIndex==0 && lastIndex>0) remove()
30        lastIndex=currentIndex
31    }
32 }

```

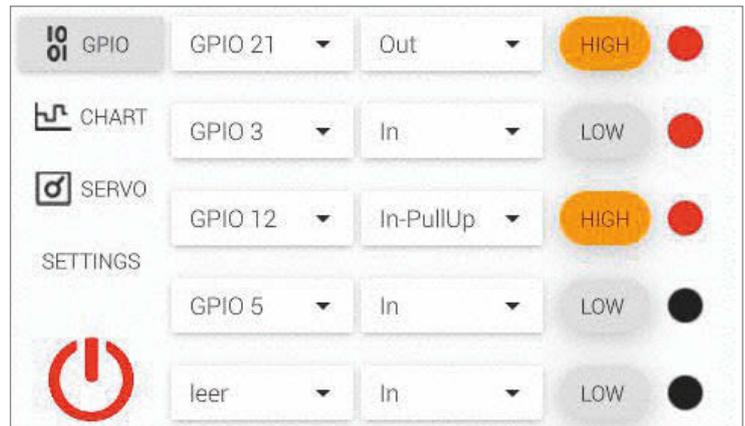
28 Listing 18

```

1 import QtQuick 2.8
2 import QtQuick.Controls 2.4
3 import Wire 1.0
4 ListView {
5     id: listView
6     spacing: 10
7     anchors.fill: parent
8     model: ListModel {ListElement {leer:0}}
9     delegate: component_gpio
10 }

```

30 Ab diesem Punkt sieht man auf dem Touchscreen die verschiedenen PINS und ihre Belegung.



chen Slot `updateValue()` auslöst. Bei der Implementierung von `updateValue()` sollte man – wie bei allen anderen Funktionen – darauf achten, dass das entsprechende Changed-Signal nur dann ausgelöst wird, wenn sich die Eigenschaft wirklich geändert hat. Sonst hält man die QML-Engine unnötig auf Trab. 25

Weiter geht es mit der GUI

Als Erstes lege ich eine Liste an, die alle notwendigen Informationen über die Pins enthält. Dann platziere ich sie im `root`-Element. Diese Liste unterscheidet zwischen der physikalischen Position (`pos`) und der internen Bezeichnung (`pin`). Mit `reserved` kann man Pins von der Auswahl ausschließen. So kann man verhindern, dass ein Pin versehentlich ausgewählt wird – zum Beispiel, wenn dieser

bereits vom Display belegt ist. Im Feld `key` steht der Name, mit dem ich den Pin auswählen möchte 26. `type` verwende ich später, um die Liste nach bestimmten Pins zu filtern.

Bei allen Modulen sollen GPIOs ausgewählt werden. Daher ist es sinnvoll, ein angepasstes Steuerelement bereitzustellen. Dazu bietet sich eine `ComboBox` an. Damit ich das Steuerelement in mehreren Modulen nutzen kann, speichere ich es als `ComboBoxGPIO.qml` im Projekt 27. So kann ich es wie jedes andere Steuerelement verwenden.

Die wesentliche Änderung befindet sich im Slot `onCurrentIndexChanged`. Hier stelle ich sicher, dass nur Pins ausgewählt werden können, die nicht mit `reserved:true` gekennzeichnet sind. Zudem wird der ausgewählte Pin reserviert und somit für eine erneute Auswahl blockiert. Beim Blockieren

eines zusätzlichen Pins löst das Signal `add` und beim Freigeben das Signal `remove()` aus. Außerdem realisiere ich über die Eigenschaften `height:` und `visible:` einen Filter, damit ich für alle Module das gleiche `ListModel` benutzen kann. Dazu sollen für das Servomodul nur GPIOs angezeigt werden, die auch Pulswellenmodulation beherrschen.

Modul GPIO

Für das Modul `GPIO` geht es mit einem einfachen `ListView` los 28. Als Datenmodell dient ein `ListModel` mit einem weitestgehend leeren `ListElement`. Für jeden weiteren Eintrag in das Modell wird eine neue Zeile angehängt. Der Weg über das `ListModel` ist etwas umständlich, aber Modelle, die auf einem `JSArray` basieren, kann man nicht dynamisch er-

29 Listing 19

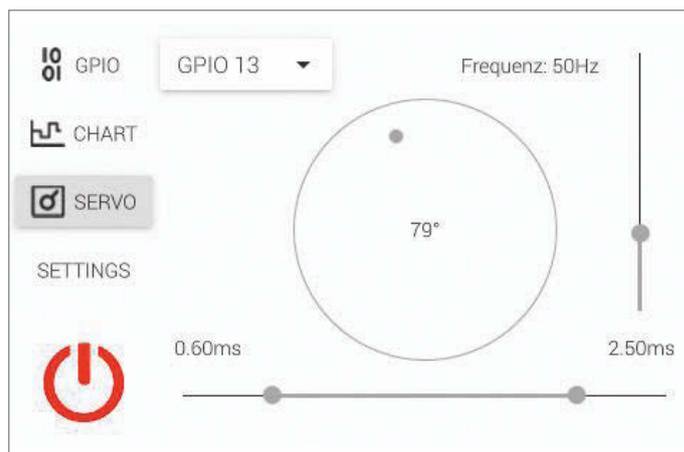
```

1 Component {
2     id: component_gpio
3     Row {
4         spacing: 10
5         ComboBoxGPIO {
6             id: cb_pin
7             onAdd:
8                 listView.model.append(listView.model.get(0))
9             onRemove: listView.model.remove(index)
10        }
11        ComboBox {
12            id: cb_mode
13            property int mode: model.get
14                (currentIndex).mode
15            model: ListModel {
16                ListElement { key:
17                    "IN"; mode: GPIO.Input }
18                ListElement { key: "IN
19                    mit Pull"; mode: GPIO.InputPullUp }
20                ListElement { key:
21                    "OUT"; pin: 20; mode: GPIO.Output }
22            }
23            textRole: "key"
24        }
25        RoundButton {
26            id: b_state
27            width: height * 1.5
28        }
29        Rectangle {
30            anchors.verticalCenter:
31                parent.verticalCenter
32            height: parent.height / 2
33            width: height
34            radius: height / 2
35            border.color: "gray"
36            border.width: 1
37            color: gpio.value ? "red" : "black"
38        }
39        GPIO {
40            id: gpio
41            pin: cb_pin.pin
42            mode: cb_mode.mode
43            state: b_state.checked
44        }
45        Timer {
46            interval: 100
47            repeat: true
48            running: gpio.pin ? true : false
49            onTriggered: gpio.
50                updateValue()
51        }
52    }
53 }

```



31 Die Anzeige ist um ein kleines Diagramm mit Pin-Zuständen ergänzt.



33 Mit dem Slider kann man testen, in welchem Frequenzbereich die Steuerung der Servos zuverlässig funktioniert.

weitem. So würde zwar `model:6` ein `ListView` mit sechs Zeilen erstellen, aber der Versuch, das Modell zur Laufzeit zu verändern, würde zum Programmabsturz führen.

Als `delegate`: verwende ich die `ComboBox` für die Pinauswahl und natürlich das in C++ erstellte `GPIO`-Objekt. Dazu kommt noch eine `ComboBox`, mit der man auswählen kann, ob der Pin als `Output`, `Input` oder `Input` mit `PullUp` fungieren soll. Nun fehlt noch ein einfacher Schalter, um zwischen `High` und `Low` zu wechseln, und eine Anzeige für den tatsächlichen Zustand (`High/Low`). Für die Anzeige wähle ich ein Rechteck, dessen Farbe ich mit der Eigenschaft von `gpio.value` verbinde 29. Auf das 3,5-Zoll-Display passen so fünf Steuereinheiten

32 Listing 20

```
1 void myGPIO::setServo(double winkel, unsigned int freq, double
  low, double high){
2     const unsigned int resolution=1023;
3     double tick = double(1000)/(resolution*freq);
4     int position=int ((low+(high-low)*winkel)/tick);
5     pwmSetRange (resolution);
6     pwmSetClock (18750/freq);
7     pwmWrite(m_pin, position);
8 }
```

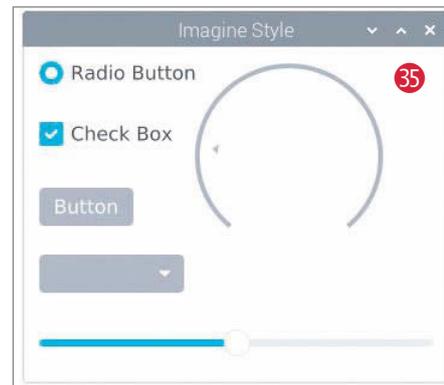
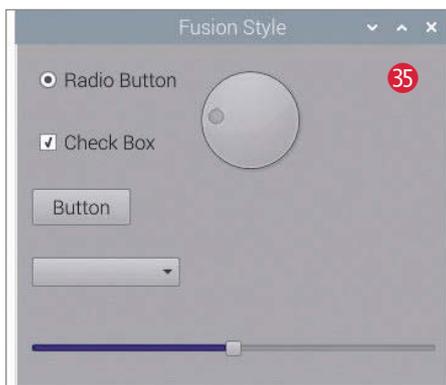
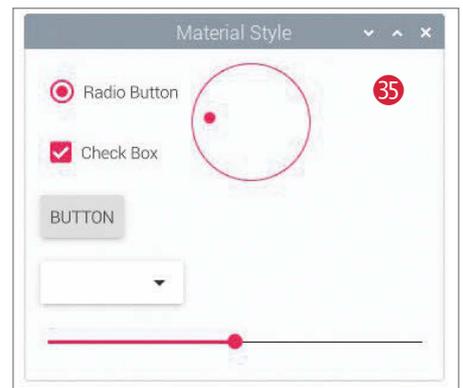
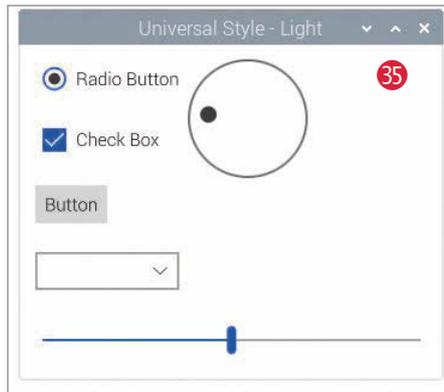
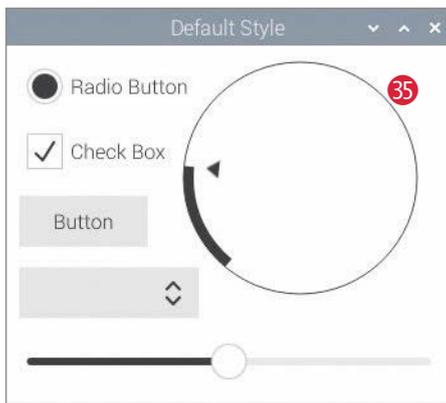
gleichzeitig 30. Da die Seite aber auf einem `ListView` basiert, der die Ansicht für jeden Eintrag automatisch um eine Zeile erweitert, lassen sich beliebig viele `GPIOs` gleichzeitig auswählen.

Modul Chart

Im Modul `QtCharts 2.0` findet sich das Steuerelement `ChartView`, mit dem man sehr einfach verschiedene Diagramme erstellen

34 Listing 21

```
1 Page {
2     ComboBoxGPIO{
3         id:cb_servo
4         filter:"PWM"
5     }
6     Dial {
7         id: dial_servo
8         to: 180
9         onValueChanged: gpio_servo.update()
10    }
11    Slider {
12        id: s_freq
13        orientation: Qt.Vertical
14        from: 10
15        to:150
16        value: 50
17    }
18    RangeSlider{
19        id:r_servo_range
20        from:0.1
21        to:3.0
22        snapMode: "SnapAlways"
23        stepSize: 0.05
24        first.value: 1
25        second.value:2.5
26        first.onValueChanged:{dial_servo.value=
dial_servo.from;gpio_servo.update();}
27        second.onValueChanged:{dial_servo.value=dial_
servo.to;gpio_servo.update();}
28    }
29    Label {
30        text: dial_servo.value.toFixed(0) +"-∞"
31    }
32    Label {
33        text: "Frequenz: " + s_freq.value.
toFixed(0) + "Hz"
34    }
35    Label {
36        text:
r_servo_range.first.value.toFixed(2)+"ms"
37    }
38    Label {
39        text:
r_servo_range.second.value.toFixed(2)+"ms"
40    }
41    GPIO{
42        id:gpio_servo
43        pin:cb_servo.pin
44        function update(){gpio_servo.setServo
(dial_servo.value/ dial_servo.to,
s_freq.freq, r_servo_range.first.
value, r_servo_range.second.value);
45    }
46    }
47 }
48 }
```



kann. Das lädt zu einer kleinen Spielerei ein. Mit einem Diagramm lasse ich mir für einen Pin den logischen Zustand über die Zeit auftragen ³¹. Hierfür nutze ich eine Datenreihe vom Typ *LineSeries* und fülle sie mit einem Timer in einem festgelegten Intervall mit Datenpunkten. Die Konstruktion ist ziemlich simpel und die Leistungsfähigkeit begrenzt. Mit dem Raspberry Pi schafft man so eine Auflösung von gut 25 Hertz.

Modul Servo

Mit dem Servomodul soll es möglich sein, einen analogen Servo anzusteuern. Der Raspberry Pi 3B besitzt zwei GPIOs, die PWM-fähig sind. Damit eignen sie sich für die Ansteuerung eines Servos. Mit `filter:"PWM"` werden in der ComboBox nur die zulässigen Pins angezeigt. Zur Steuerung des ausgewählten Pins nutze ich ein Dial-Element für

die Position des Servos im Bereich von 0 bis 180° ³³. Die Endpunkte, die ja von Servo zu Servo etwas variieren können, bestimme ich mit einem *RangeSlider*. Üblicherweise werden analoge Servos mit einem PWM-Signal von 50 Hertz angesprochen. Dennoch habe ich einen Slider hinzugefügt, mit dem man die Frequenz im Bereich 10–200 Hertz variieren kann. So ist es möglich, zu testen, in welchem Frequenzbereich die Steuerung zuverlässig funktioniert. Ergänzt wird die Benutzeroberfläche mit ein paar Labels, die die tatsächlichen Werte der vier Parameter anzeigen. Für die Ansteuerung ist es notwendig, die Klasse `MyGPIO` zu erweitern. Die Klasse um die notwendigen Eigenschaften zu ergänzen wäre konsequent, ist mir aber zu aufwendig. Aus diesem Grund ergänze ich die Klasse lediglich um den Slot `void setServo(double winkel, unsigned int freq, double low, double high)`. Da ich die Ei-

genschaften nicht definiere und es daher nicht möglich ist, entsprechende Datenverbindungen zu den Steuerelementen zu nutzen, muss ich selber festlegen, wann `setServo(...)` ausgeführt werden soll ³². Übersichtshalber realisiere ich das über die Funktion `„update()“` im Element `GPIO`. Sie wird aufgerufen, sobald sich die Werte des *RangeSliders* oder des Dials ändern. Die *WiringPi*-Bibliothek benötigt Root-Rechte, wenn sie die Funktionen `pwmSetRange` und `pwmSetClock` ausführen soll. Daher ist es von nun an notwendig, das Programm mit Root-Rechten zu starten ³⁴.

Settings

Auf dieser Seite soll sich das Erscheinungsbild der Benutzeroberfläche anpassen lassen. Qt bietet hier fünf verschiedene Styles an ³⁵.

Der Default-Style ist auf Leistung optimiert, der Fusion-Style soll ein Desktop-Look&Feel geben und beim Imagine-Style basieren die Steuerelemente auf Bildern, die sich auswechseln lassen. Der Material-Style basiert auf Google's und der Universal-Style auf Microsoft's Designvorgaben und haben jeweils ein helles und ein dunkles Thema. Zudem lassen sich bei beiden die Farbakzente ändern. Ich entscheide mich für den Material-Style und lege diesen in `main.cpp` mit `qputenv(QT_QUICK_CONTROLS_STYLE , material);` fest. Die Auswahl des Themas

³⁷ Listing 22

```
1 app.setOrganizationName("maker");
2
3 app.setOrganizationDomain("heise.de");
4
5 app.setApplicationName("RPI_Maker");
```

und der Akzentfarbe realisiere ich mit zwei ComboBoxen **36**.

Die Boxen fülle ich mit jeweils einem *List-Model*. Zudem füge ich noch einen Schalter vom Typ *Switch* hinzu, mit dem ich den Vollbildmodus an- und ausschalten kann. Damit nicht bei jedem Programmstart erneut das Design ausgewählt werden muss, möchte ich die Auswahl speichern und beim Programmstart wieder automatisch herstellen. Was sich aufwendig anhört, geht mit dem Element Settings quasi von alleine. Lediglich Firmenname, Domain und Anwendungsname müssen in `main.cpp` festgelegt werden **37**.

Und fertig

Anschließend definiert man im Settings-Element für jede zu sichernde Eigenschaft ein Alias. Fertig! Nun wird jede Änderung in den Eigenschaften `root.theme`, `root.accent`, `root.fullscreen` und `screens.currentIndex` automatisch gespeichert. Beim Neustart des Programmes wird der letzte Wert wiederhergestellt. Das Settings-Element befindet sich in der Bibliothek `Qt.Labs.set-`

38 Listing 23

```
1 import Qt.Labs.settings 1.0
2 ApplicationWindow {
3     id:root
4     Settings{
5         id:settings
6         property alias theme: root.theme
7         property alias accent: root.accent
8         property alias fullscreen: root.fullscreen
9         property alias page: screens.currentIndex
10    }
```

`tings`. Wie der Name bereits signalisiert, ist es leider nicht garantiert, dass nachfolgende Qt-Versionen diese Bibliothek ebenfalls enthalten werden **38**.

Anpassen

Eine Stärke von *QtQuick* ist, dass nichts so aussehen muss, wie von Qt vorgegeben. Man kann Steuerelemente selbst kreieren – so, wie ich es mit dem Power-Button gemacht habe. Es ist aber auch möglich, be-

stehende Steuerelemente zu modifizieren. So hätte ich zum Beispiel für den Power-Button auch mit einem Button-Steuerelement starten und dessen Hintergrund ersetzen können: `background: Image{ source: img/Power.png }`. In ähnlicher Weise kann man die meisten Steuerelemente modifizieren. Dem Erscheinungsbild der Steuerelemente sind – außer der eigenen Kreativität und der Rechenleistung des Minicomputers – quasi keine Grenzen gesetzt. —*rehu*

Tools für Maker



PORTOFREI
AB 15 €
BESTELLWERT

 **heise shop**

shop.heise.de/tools

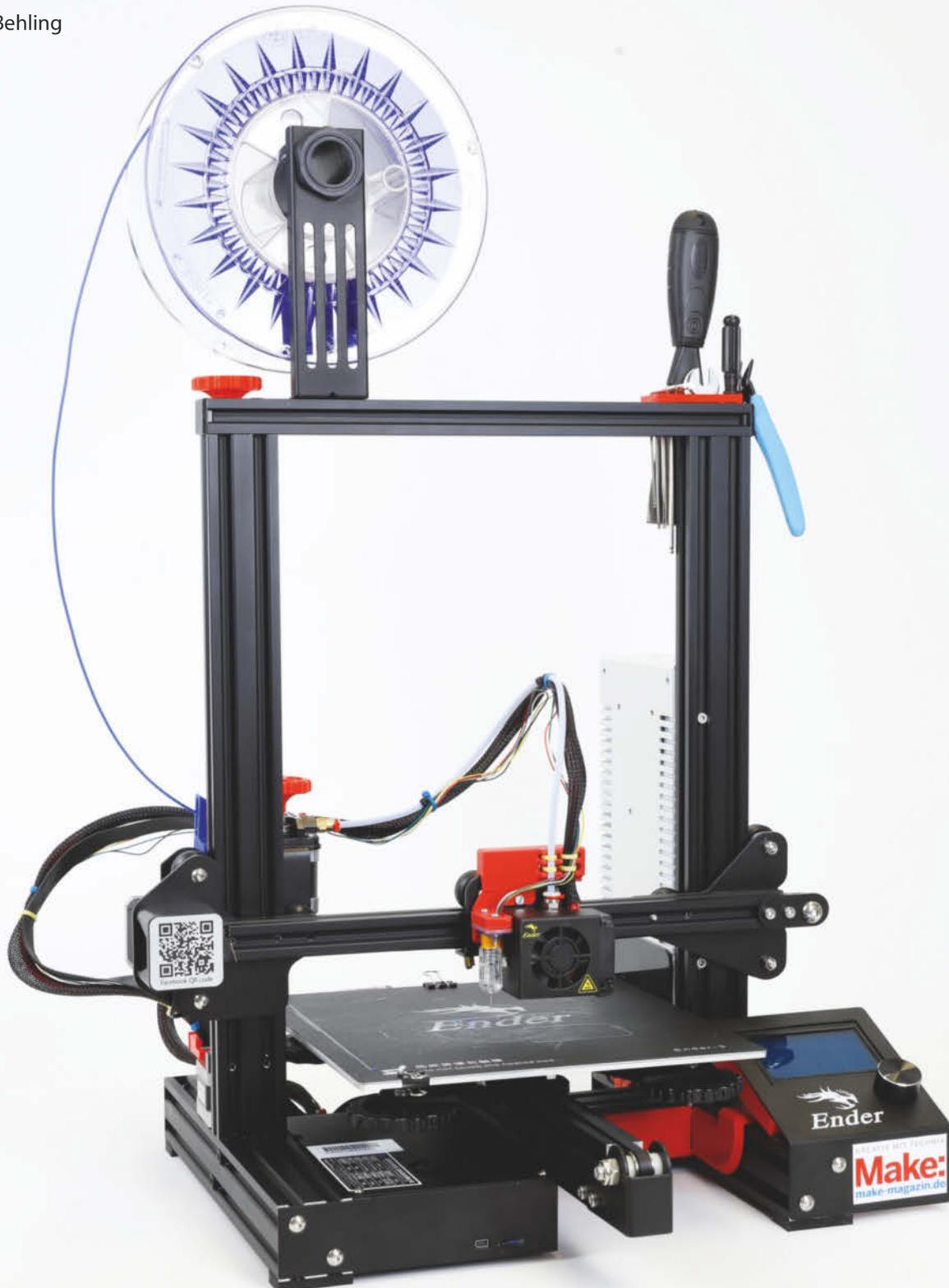
Bestellen Sie ganz einfach online unter shop.heise.de oder per E-Mail: service@shop.heise.de

© Copyright by Maker Media GmbH.

Druckbettensor nachrüsten

Die Bedienung eines 3D-Druckers ist im Grunde recht einfach, bis auf die immer wieder mal nötige genaue Einstellung des Druckbetts. Ein kleiner zusätzlicher Sensor macht das beim Creality Ender 3 zum Kinderspiel.

von Heinz Behling



dealerweise ist das Druckbett eines 3D-Druckers so eingestellt, dass seine Oberfläche völlig parallel zur Bewegungsebene des Druckkopfes liegt und bei Druckhöhe Null die Düsen spitze das Druckbett gerade nicht mehr berührt. Diesen Zustand auf der gesamten Fläche stellt man beim sogenannten Leveling mit vier Stellschrauben unterhalb des Druckbetts ein. Hat man das einmal mit der notwendigen Genauigkeit erreicht, ist das aber meist kein Dauerzustand: Zum einen sind die Stellschrauben beim Creality Ender 3 sehr groß und leichtgängig **1**, sodass sie schnell versehentlich verstellt werden. Zum anderen hat der eingebaute Mikroschalter **2**, der dafür sorgen soll, die Position immer wieder zu finden, eine recht geringe Wiederholgenauigkeit. Daher kann es beim nächsten Druckvorgang vorkommen, dass die Druckbettposition einige Zehntelmillimeter abweicht. Viel zu viel, denn beim 3D-Druck wird mit Druckschichten um 0,1mm gearbeitet. Steht dann das Druckbett zu tief, haftet der Druck nicht auf dem Bett, ist es zu hoch, kann kein Filament aus der Düsenöffnung austreten. In besonders schlimmen Fällen zerkratzt die Düse sogar das Druckbett. Daher ist häufiges und lästiges Nachjustieren nötig.

Abhilfe bringt da der Einbau eines Sensors, dessen Wiederholgenauigkeit erheblich besser ist. Der hier verwendete BLTouch hat eine Genauigkeit von 0,01mm, die preiswerteren Nachbauten (3D Touch) sind erfahrungsgemäß ein wenig schlechter, aber immer noch erheblich besser als das Original.

Und noch besser: Dadurch, dass der Sensor nicht am Druckerrahmen, sondern am Druckkopf befestigt wird, kann er das Druckbett an mehreren Stellen abtasten und so eventuelle Unebenheiten des Druckbetts messen. Mit der richtigen Drucker-Firmware

Kurzinfo

- » Druckbett-Sensor in 3D-Drucker Creality Ender 3 einbauen
- » Automatic Bed Leveling in der Firmware Marlin einschalten
- » Firmware auf Drucker installieren

Checkliste

-  **Zeitaufwand:**
etwa 2 Stunden
-  **Kosten:**
etwa 40 Euro
-  **Programmieren:**
Bedienung der Arduino-IDE
-  **3D-Druck:**
Druck des Sensorhalters
-  **Löten:**
einfache Lötarbeiten

Material

- » Sensor-Set von Creality inklusive USB-Programmer zum Aufspielen des Bootloaders alternativ:
- » Sensor BLTouch oder ähnlich
- » Adapter zum Sensoranschluss (s. Bezugsquellen)
- » Kabelbinder

Werkzeug

- » Bordwerkzeug des Creality Ender 3
- » Crimp-Zange

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

Mehr zum Thema

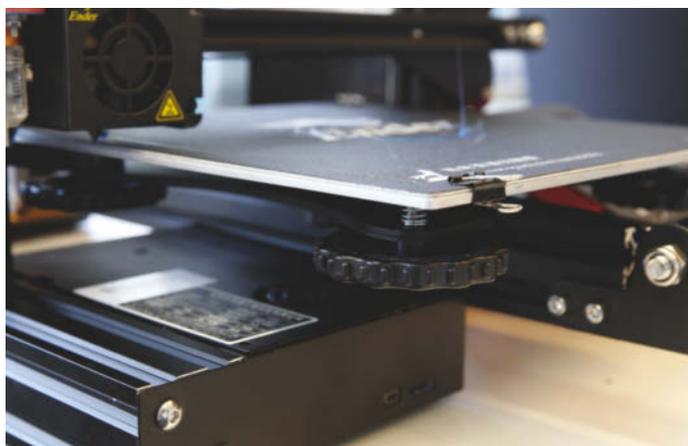
- » Heinz Behling, Bootloader auf 3D-Drucker Creality Ender 3 installieren, online
- » Heinz Behling, 3D Touch-Sensor wiederbeleben, Make 2/19, S. 96

(Marlin) berücksichtigt der Creality dann beim Druck diese Ungenauigkeiten und passt die Position des Druckkopfes über dem Druckbett laufend entsprechend an. Dies ist die sogenannte ABL-Funktion (**Automatic Bed Leveling**). Nebenbei erhält der Drucker so auch noch eine Firmware, die die Temperaturen von Heizbett und Druckkopf überwacht und bei Ausfall einer der Thermosensoren die Heizungen ausschaltet und so Schlimmeres verhütet. Sie müssen jedoch

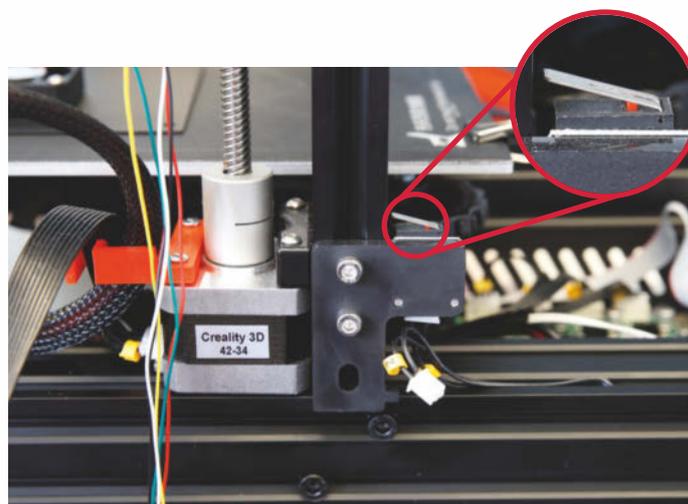
einen kleinen Preis dafür zahlen: Der Beeper unter dem Druckerdisplay, den viele ohnehin für lästig halten, wird für immer verstummen, da wir seine Steuerleitung zweckentfremden müssen.

Halter drucken

Zuerst laden Sie sich die Druckdatei für den Sensorhalter und die Distanzscheiben auf Ihren Computer (über den Kurzinfo-Link zu



1 Die leichtgängigen Stellschrauben für die Druckbetthöhe sind allzu schnell verstellt.



2 Der Mikroschalter ist nicht sehr genau. Er wird durch den neuen Sensor ersetzt und kann ausgebaut werden.

So funktioniert der Sensor

Der Sensor hat am unteren Ende eine Tastnadel. Durch den im Sensor eingebauten Servo, dessen Steuerung über das dreipolige Kabel durch den Drucker erfolgt, wird sie bei einer Messung nach unten ausgefahren. Wird sie bei Berührung ein wenig in den Sensor gedrückt, schnell die Nadel blitzartig nach oben. Das wird durch einen kleinen Magneten am oberen Ende der Nadel erreicht, dessen Anziehungskraft auf eine eiserne Stellschraube am oberen Ende des Sensors wirkt. Befindet sich die Nadel unten, reicht seine Kraft nicht aus. Erst durch die geringe Abstandsänderung beim Berühren des Druckbetts kann er die Nadel anheben. Dieser Abstand hat eine hohe Wiederholgenauigkeit. Zugleich schließt sich im Sensor ein Schaltkontakt, der über ein zweipoliges Kabel mit der Drucker elektronik verbunden ist und das Signal des bisherigen Mikroschalter-Höhensensors ersetzt.

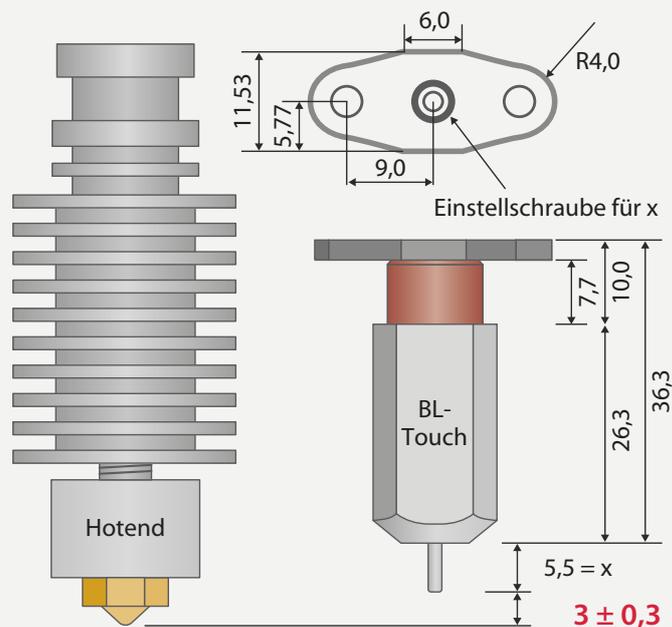
Damit die Tastnadel beim Drucken nicht im Weg ist, muss ihr Ende oberhalb der Düsen spitze stehen. Deshalb muss der Sensor auf einer bestimmten Höhe montiert sein. Diese Maße (siehe Zeichnung) empfiehlt der Sensor-Hersteller.

Beim Messen muss die Spitze der Tastnadel tiefer stehen als die Düsen spitze des Druckkopfes. Das wird durch den Servoantrieb im Sensor bewirkt. Das bedeutet aber, dass für den Sensoranschluss eine zusätzliche Verbindung zur Ansteuerung

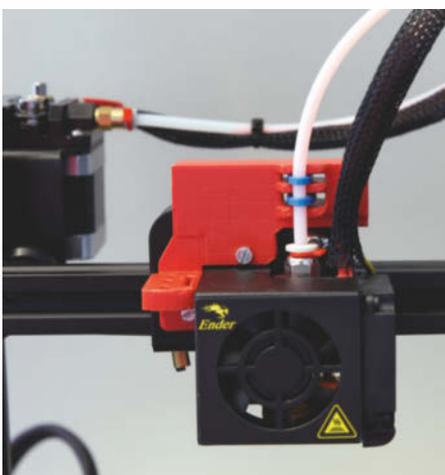
des internen Servos erforderlich ist. Das Elektronik-Board des Creality Ender 3 hat jedoch keinen freien Anschlusspin dafür. Daher muss hier ein Trick helfen: Die Leitung, die den Beeper im Display-Gehäuse steuert, wird aufgetrennt und als Servoleitung „umgenutzt“. Diese Auftrennung kann man zwar per Seitenschneider im Flachkabel durchführen, das zum Display führt. Sauberer ist aber ein Adapter, der

zwischen Board und Display-Kabel eingesteckt wird. Er leitet die entsprechende Verbindung um und führt sie auf eine Steckerleiste, an die dann der dreipolige Stecker der Sensorleitung (das ist die Servosteuerung) gesteckt wird.

Dass der Beeper dann nicht mehr funktioniert, wird von vielen Ender-Besitzern eher als Vorteil gewertet.



Die Maße des Sensors: wichtig sind die Abstände der Tastnadel.



3 So muss der selbstgedruckte Sensorhalter am Druckkopf sitzen.

erreichen) und drücken den Halter und Scheiben. Als Einstellungen empfehlen wir 0,1mm Schichthöhe, Füllung 20 Prozent mit Stützstruktur. Nach dem Entfernen der Stützen und Säubern der Kanten und Flächen schrauben Sie den Halter an den Druckkopf 3.

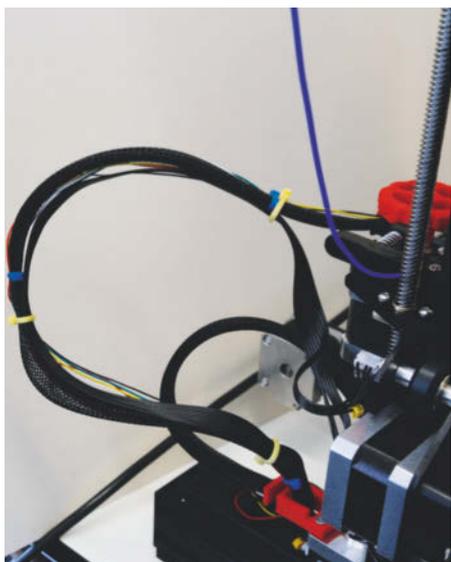
Bei dem in den Bezugsquellen (Kurzinfolink) genannten Sensor-Nachbau wird ein Kabel in passender Länge mitgeliefert, an das Sie noch die Stecker (mitgeliefert) für den Anschluss an das Elektronik-Board des Druckers crimpen müssen. Falls bei Ihrem Sensor nur ein kurzes Kabel beigelegt ist, müssen Sie es verlängern. Dazu schneiden Sie das kurze Kabel durch und löten Schaltlitze entsprechender Länge dazwischen. Auf diese Weise haben Sie an beiden Enden wieder die passenden Stecker. Achten Sie darauf, dass Sie die fünf Drähte nicht verwech-

seln, andernfalls kann der Sensor später beschädigt werden.

Verlegen Sie nun das Kabel vom Druckkopf zum Elektronikgehäuse. Befestigen Sie es mit Kabelbindern am Druckkopf und am Kabelbaum sorgfältig, damit es später beim Drucken nirgends hängen bleiben kann 4. Die dünnen Drähte reißen nämlich sehr leicht. Am Sensorhalter können Sie auch den Kabelbaum des Druckkopfes sowie den Filamentschlauch mit Kabelbindern befestigen. Falls Sie es besonders schön machen möchten, fädeln Sie das Sensorkabel in den Textilschlauch des Kabelbaums.

Anschluss

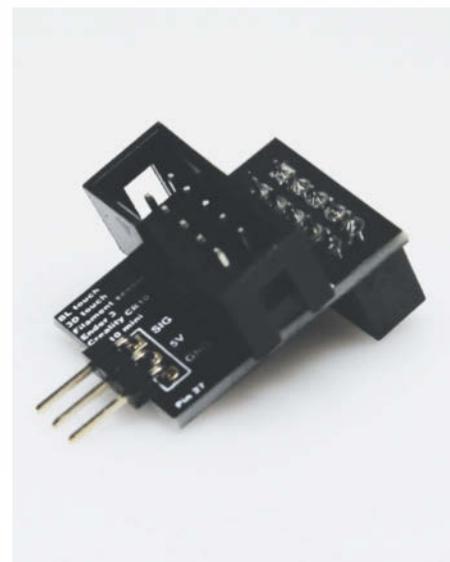
Öffnen Sie dann das Elektronikfach 5. Ziehen Sie das Displaykabel vom Elektronik-



4 Wichtig: Vergessen Sie nicht, die wenig reißfesten Sensorleitungen auch hinten gut zu befestigen.



5 Zum Öffnen des Elektronikfachs müssen drei Schrauben entfernt werden. Die dritte ist nur erreichbar, wenn der Drucktisch ganz vorn steht.



6 Dieser Adapter lenkt die Leitung vom Beeper zum Sensor um und wird zwischen Displaykabel und Mainboard gesteckt.

board ab und stecken Sie stattdessen den Sensor-Adapter **6** ein und stecken Sie das Displaykabel in die Buchse auf dem Adapter **7**.

Schrauben Sie nun den nicht mehr benötigten Mikroschalter an der linken Säule des Druckers ab **2**. Der neue Sensor hat zwei Kabel: Das zweipolige ist die Verbindung mit dem Schaltkontakt und ersetzt den bisherigen Mikroschalter. Stecken Sie ihn daher in die Buchse, in der bislang das Kabel des Mikroschalters steckte. Das dreipolige Kabel führt das Signal zur Steuerung des sensorinternen Servos zum Bewegen der Nadel (siehe Kasten „So funktioniert der

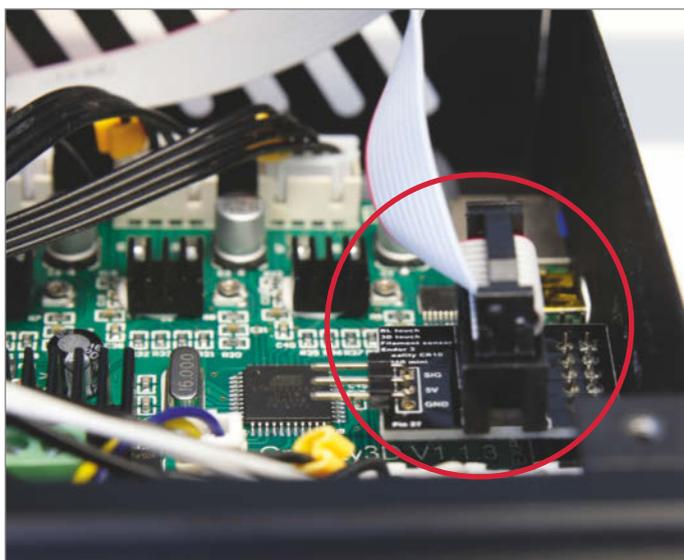
Sensor“). Es kommt auf die Pinleiste des Adapters **8**. Beachten Sie anhand der Farben die richtige Polung der Kabel. Nun können Sie das Elektronikfach wieder schließen. Die längere der drei Schrauben kommt nach hinten.

Jetzt ist es Zeit, den Sensor an den Druckkopf zu schrauben. Wichtig: Im Ruhezustand, wenn die Nadel also eingefahren ist, muss die Nadelspitze 3mm über der Düsenspitze stehen **9**. Insbesondere die Nachbauten des Sensors variieren in der Länge. Deshalb müssen Sie diesen Abstand mit Distanzscheiben zwischen Sensor und Halter einstellen. Die Druckdateien dafür

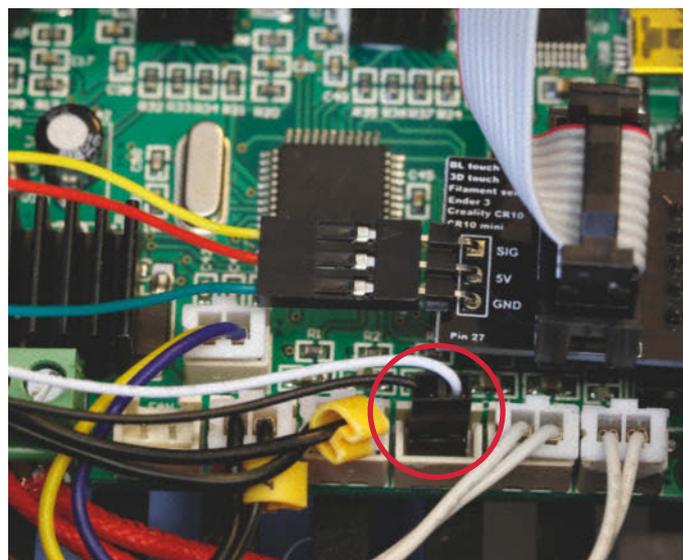
(in 1 und 1,5mm Stärke) gibt es ebenfalls zum Download. Bevor Sie den Sensor festschrauben, stecken Sie dessen Kabel ein!

Das letzte Level

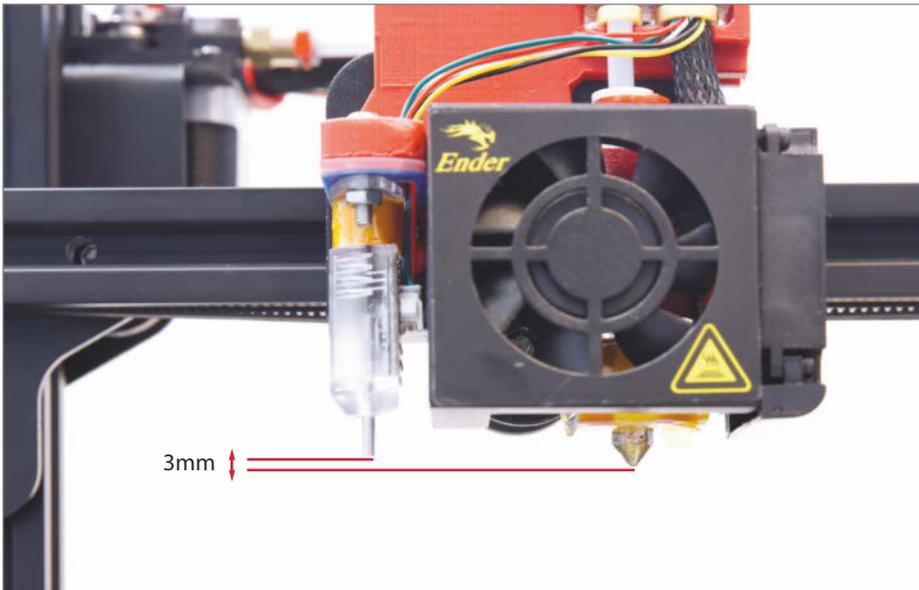
Stellen Sie nun das Druckbett noch einmal ein. Legen Sie ein Stück normales 80g-Briefpapier auf den Drucktisch und fahren Sie den Druckkopf von Hand in der Mitte des Tisches so weit nach unten, bis die Düsenspitze gerade auf dem Papier aufliegt. Das können Sie spüren, wenn Sie das Papier bewegen. Bewegen Sie den Druckkopf an die Ecken des



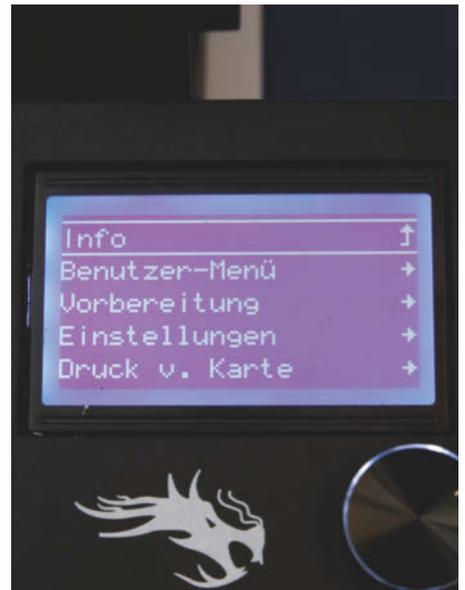
7 So muss der Sensor-Adapter zwischen Displaykabel und -buchse eingesteckt werden.



8 Schließlich stecken Sie die beiden Stecker der Sensorleitung ein. Achten Sie auf die Reihenfolge der Farben!



9 Im Ruhezustand (Nadel oben) muss die Nadelspitze 3mm über der Düsen Spitze stehen. Diesen Abstand stellen Sie mit den selbstgedruckten Distanzscheiben ein.



10 Mit ein paar kleinen Änderungen bringen Sie der Firmware Deutsch bei.

Tisches und stellen Sie die Höhe des Tisches mit den Stellschrauben so ein, dass der Widerstand überall gleich ist. Das Papier hat eine Stärke von etwa 0,1mm. Somit steht die Düsen Spitze nun in allen Ecken ebenfalls 0,1mm über der Druckplatte.

Damit sind die mechanischen Arbeiten abgeschlossen. Nun geht es daran, die neue Firmware auf dem Drucker zu installieren. Dies geschieht mit der Arduino-IDE (bitte downloaden und installieren) und einer USB-Verbindung zum Drucker. Leider besitzen die bislang im Handel erhältlichen Ender 3 keinen Bootloader, also jene Software, die man zur Firmware-Installation braucht. Sie müssen also zunächst diesen Loader installieren. Eine Anleitung dazu gibt es auf den Make-Internetseiten (siehe Kurzinfo-Link).

Konfiguration

Als Firmware benutzen wir Marlin, das von Daniel Brooke Peig bereits für den Ender 3 und den Sensor angepasst wurde. Laden Sie es auf Ihren Computer (siehe Kurzinfo-Link) und entpacken Sie das Archiv. Falls Sie mit englischen Menüs am Drucker leben können, machen Sie mit dem Abschnitt „Installation“ weiter. Möchten Sie dem Drucker Deutsch 10 beibringen, sind noch ein paar Änderungen an der Firmware nötig. Normalerweise wird zur Änderung der Menüsprache in Marlin nur der Eintrag

```
#define LCD_LANGUAGE en
entsprechend geändert, zum Beispiel in
#define LCD_LANGUAGE de
```

In unserem Beispiel genügt das aber nicht, denn durch die zusätzlichen Befehle zur Steuerung des Sensors ist die Firmware so groß geworden, dass die englische Version gerade noch in den Speicher des Druckers passt. Die deutschen Menüs nehmen aber mehr Platz in Anspruch. Daher haben wir die deutschen Menütexte etwas gekürzt und noch ein paar andere, unnötige Bestandteile entfernt. Die Änderungen befinden sich in den Dateien *configuration.h* und *language_de.h*, die ebenfalls zum Download bereitstehen. Überspielen Sie die Dateien und entpacken Sie sie in das Marlin-Verzeichnis der zuvor entpackten Firmware.

Installation

Starten Sie die Arduino-IDE. In der Grundversion kennt die IDE das Board des Ender 3 nicht. Das kann man ihr jedoch mit dem Boardverwalter beibringen. Zunächst tragen Sie in den Voreinstellungen (zu finden im Datei-Menü) als zusätzlichen Boardverwalter die Adresse aus Bild 11 ein. Diese Adresse finden Sie auch in der Readme-Datei auf GitHub. Danach klicken Sie sich über *Werkzeuge* und *Board* zum *Boardverwalter* durch und installieren dort das Sanguino-Board 12.

Nun fehlt nur noch die Bibliothek zur Ansteuerung des LC-Displays. Mit Klicks auf *Sketch*, *Bibliothek einbinden* und *Bibliotheken verwalten* gelangen Sie in den Bibliotheksverwalter. Suchen Sie dort nach *U8GLib* und installieren Sie die Bibliothek.

Nun haben Sie alles beisammen. Schließen Sie nun den Ender über ein USB-Kabel

an den Computer an. Normalerweise sollte die Stromversorgung über den USB-Anschluss für das Ender-Board ausreichend sein, erkennbar an der normalen Anzeige auf seinem Display. Erscheint dort aber nichts oder nur kryptische Zeichen, schalten Sie zusätzlich auch den Drucker an. Öffnen Sie nun in der Arduino-IDE die Datei *Marlin.ino*, die sie im zuvor entpackten Marlin-Verzeichnis der Firmware finden. Im *Werkzeuge*-Menü wählen Sie dann Board, Prozessor und COM-Port 13.

Jetzt fehlt nur noch ein Klick auf das Symbol 14. Nach kurzer Wartezeit ist die Firmware übertragen und der Ender startet neu. Auf seinem Display erscheint nun der Info-Bildschirm von Marlin.

Z-Versatz

Zwei Dinge sind nun noch zu erledigen: Dem Drucker muss mitgeteilt werden, wie hoch die Düsen Spitze über dem Druckbett steht, wenn der Sensor anspricht (Z-Versatz). Und in Ihrem 3D-Druckprogramm müssen Sie den Befehl zum Start der ABL-Funktion einfügen. Zunächst der Z-Versatz. Legen Sie wieder ein Stück Papier auf den Drucktisch. Über das Drucker Menü *Vorbereiten* führen Sie ein *Home* durch.

Sobald das erledigt ist, bewegen Sie die Z-Achse ebenfalls per Drucker Menü (*Vorbereiten*, *Bewegen*, *Z-Achse* und *0,1mm*) und Drehknopf langsam nach unten, bis das Display für Z den Wert 0 anzeigt. Die Düse liegt jetzt aber noch nicht auf dem Papier auf. Drehen Sie den Knopf langsam weiter, bis die Düse das Papier berührt. Ziehen Sie vom

nun angezeigten Z-Wert noch die Papierstärke ab. Zeigt der Drucker also beispielsweise $-0,7\text{mm}$ an, ergibt das nach Abzug von $0,1\text{mm}$ Papierstärke insgesamt $-0,8\text{mm}$. Notieren Sie sich diesen Wert.

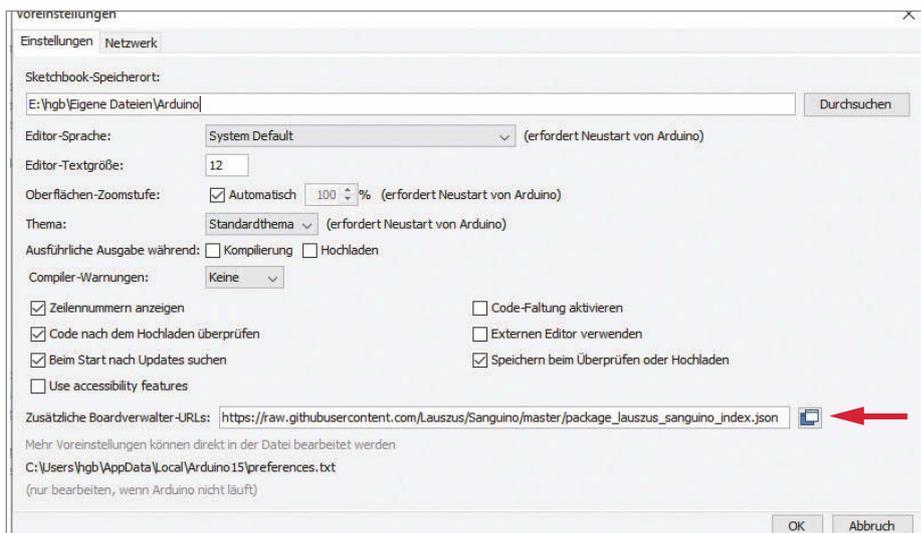
Gehen Sie dann im Menü zurück bis zu *Einstellungen, Bewegung* und *Sondenversatz Z*. Dort geben Sie nun mit dem Drehknopf den zuvor notierten Wert ein, also beispielsweise $-0,8\text{mm}$. Ein Druck auf den Knopf bringt Sie eine Menüebene zurück. Dort wählen Sie dann *Konfig. speichern*. Damit ist der Z-Versatz dauerhaft auf dem Board gespeichert.

Startskript

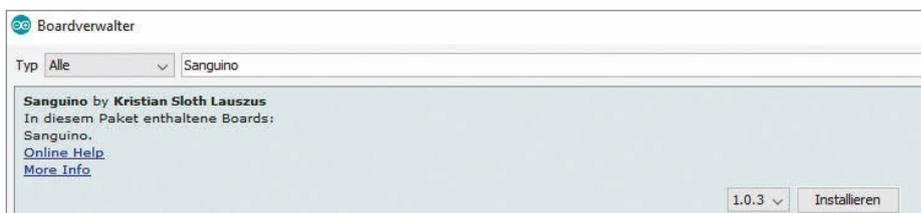
Der Drucker führt von sich aus nur das Homing mit dem Sensor aus. Um auch das Abtasten des Druckbetts und den Höhenausgleich zu benutzen, muss man es ihm mit dem Gcode `G29` mitteilen. Das muss vor jedem Druckvorgang geschehen, denn beim Ausschalten oder erneutem Homing vergisst der Drucker das zuvor gemessene Höhenprofil des Druckbetts. Das ist durchaus sinnvoll, denn beim Abnehmen und Wiederaufsetzen der Druckplatte kann sich das Profil geändert haben, etwa weil Sie das Druckbett anders positionieren. Den Gcode bauen Sie ins Startskript Ihres 3D-Druckprogramms ein, zum Beispiel Cura. Gehen Sie dort in die Druckereinstellungen, wählen Sie das Feld *Extruder* und setzen Sie ans Ende des Startskripts die Zeile

`G29`

ein **15**. Von nun an können Sie wie gewohnt drucken. Vor dem Druck tastet der Drucker nun aber 25 Messpunkte auf dem Druckbett ab und merkt sich die jeweilige Höhendifferenz. Das dauert ein wenig, macht aber bei



11 Die Informationen über das Board des Creality Ender 3 muss sich die Arduino-IDE erst aus dem Internet holen.

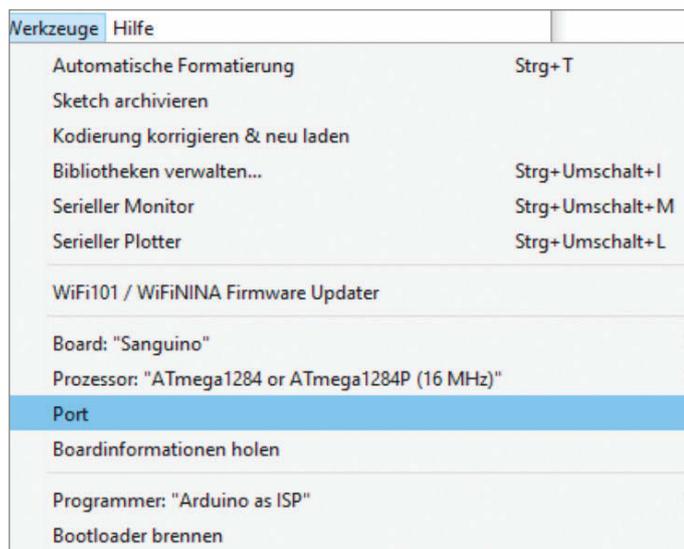


12 Sanguino ist der Name des Ender-3-Boards.

den ohnehin meist stundenlangen Druckzeiten kaum etwas aus. Falls es sie aber einmal bei einem kurzen Druckauftrag stört, können Sie den Befehl entweder kurzzeitig aus dem Startskript entfernen oder ihn nachträglich in der von Cura angelegten Gcode-Datei löschen.

Damit sind Sie nun im Besitz eines Druckers, aus dem Sie die höchstmögliche Ge-

naugigkeit herausholen können. Seine Bemühungen können Sie übrigens beobachten: Beim Druck gleicht er die Höhenunterschiede des Druckbetts durch minimale Bewegungen der Z-Achse aus. Die senkrecht stehende Gewindespindel wird sich also bei jeder horizontalen Druckkopfbewegung etwas hin- und herdrehen. Und nun viel Spaß damit!
—hgb



13 Stellen Sie das Board, den Prozessor sowie den richtigen COM-Port ein.



14 Wenn alle Vorarbeiten abgeschlossen sind, starten Sie den Kompilervorgang mit einem Klick auf den Pfeil.

```

Start G-Code

G1 X10.4 Y200.0 Z0.28 F5000.0 ;Move to side a little
G1 X10.4 Y20 Z0.28 F1500.0 E30 ;Draw the second line
G92 E0 ;Reset Extruder
G1 Z2.0 F3000 ;Move Z Axis up
G29
    
```

15 Der G29-Befehl startet die automatische Bed-Level-Funktion vor dem Druck.

Lithium-Ferrophosphat-Akkus

Aus einer nicht unberechtigten Angst heraus scheuen sich viele Maker, die „gefährlichen“ Lithium-Akkus in eigenen Projekten einzusetzen. Die relativ junge LiFePO₄-Technik verspricht eine problemlose Anwendung von Hochleistungs-Akkus – stimmt das?

von Roman Radtke



Die Batterie- und Akku-Technologie hat seit der Erfindung der Voltaschen Säule im Jahr 1800 regelrechte Quantensprünge vollzogen, und selbst dem weniger geneigten Endverbraucher dürften inzwischen Begriffe wie Alkaline, NiMH, Li-Ion oder Li-Po bekannt sein. Letztere allerdings auch wegen einiger spektakulärer Unglücke mit abgebrannten Mobilgeräten – wir erinnern uns an das Galaxy-7-Desaster von Samsung.

Von daher rührt wahrscheinlich auch die Scheu vor der Lithium-Technik. Tatsächlich – das sei an dieser Stelle vorausgeschickt – bedarf es auch nur relativ wenig Ungeschick, um einen Lithium-Akku abbrennen, platzen oder wenigstens sterben zu lassen. Dabei kommt ein Lithium-basierter Akku dem perfekten Energiespeicher schon recht nahe: Bezogen auf Gewicht und Volumen bietet er jede Menge Kapazität und einen guten Wirkungsgrad. Im Unterschied zu den bis vor einiger Zeit noch recht verbreiteten NiMH-Akkus können moderne Lithium-Zellen bis zu fünfmal mehr Energie speichern und haben eine geringe Selbstentladung, verlieren somit bei ungenutzter Lagerung nur wenig an gespeicherter Energie.

Leider weisen die meisten gängigen Lithium-Zellen einen eklatanten Nachteil auf: Werden sie beschädigt, nicht gemäß ihrer Spezifikation behandelt oder herstellereitig nachlässig gefertigt, können sie zu einer Gefahr werden, da die Separatoren zwischen den Zellen für hohe Energiedichten extrem dünn sein müssen und auch die enthaltene Chemie nicht ohne ist.

Kommt es intern zu einem Kurzschluss, so kann sich ein solcher Akku lokal stark erhitzen, wobei intern Sauerstoff freigesetzt wird. Durch diesen Prozess wird auch die Umgebung der Schadstelle weiter geschädigt, was zur Folge hat, dass sich der Schaden weiter ausbreitet und sich die gesamte Zelle unkontrolliert erhitzt (thermal runaway), was zu einem Brand oder sogar einer Explosion des Akkus führen kann.

Eine mögliche Lösung liefert eine veränderte Zellchemie, bei der die Brandgefahr so gut wie ausgeschlossen ist. Bereits im Jahre 1996 wurde als vielversprechendes Material Lithium-Eisen-Phosphat (LiFePO4) für die positive Elektrode vorgeschlagen.

Der Neue

Prinzipiell ist der Lithium-Eisen-Phosphat-Akku ein direkter Verwandter des Lithium-Ionen-Akkus. Während bei Letzterem Lithium-Cobalt-(III)-Oxid verwendet wird, kommt bei der neuen Technologie Lithium-Eisen-Phosphat (LiFePO4) zum Einsatz.

Die ersten Versuche mit LiFePO4 als Material für die positive Elektrode waren nicht allzu erfolgreich, da es nur eine geringe elek-

Kurzinfo

- »Vor- und Nachteile von LiFePO4-Akkumulatoren
- »Ladetechniken und Lebensdauer
- »Anwendung in der Praxis

Alles zum Artikel im Web unter make-magazin.de/xy3b

trische Leitfähigkeit aufwies, was eine geringe Leistungsdichte zur Folge hatte; diese Zellen waren also nicht in der Lage, große Ströme abzugeben, was sie für viele Zwecke ungeeignet machte. Mehr Erfolg hatte man, nachdem man die Kathode optimierte und ihre Durchlässigkeit erhöhte; der Urahn des heutigen LiFePO4-Akkus war geboren. Kleine Anmerkung: Bei einer elektrochemischen Zelle definiert die Kathode die Elektrode, an der die sogenannte Reduktionsreaktion stattfindet, also der Pluspol, die Anode der Minuspol!

Die Elektrode eines LiFePO4-Akkus besteht aus einer Kohlenstoff-Modifikation, in der das für die Reaktion notwendige Lithium eingelagert ist. Durch diese Zellchemie ergibt sich eine Spannung von ca. 3,2V bis 3,3V, also etwas weniger als die rund 3,7V bei einem LiPo-Akku. Auch die Energiedichte dieser Zellen ist mit 90–120Wh/kg den anderen beiden Zelltypen unterlegen (LiPo ca. 130–150Wh/kg, Li-Ion ca. 120–180Wh/kg), was jedoch durch die große Eigensicherheit bei vielen Anwendungen wieder wettgemacht wird.

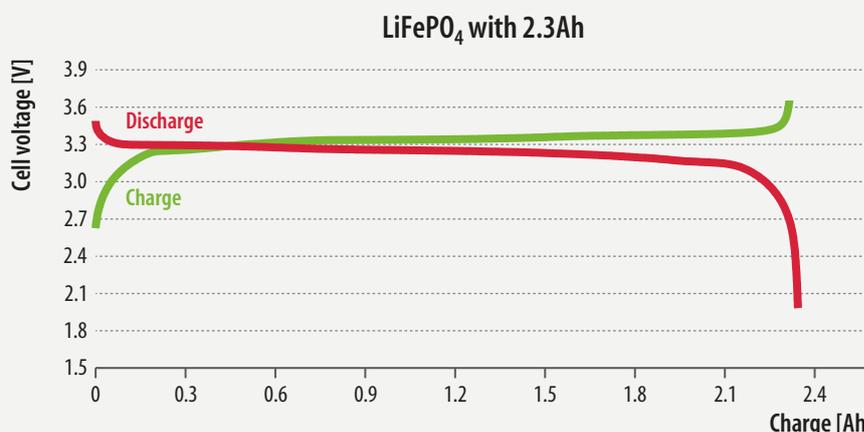


Verwechslungsgefahr: LiFePO4-Zellen, hier in Bauform 26650, unterscheiden sich kaum von gängigen Li-Ion-Akkus.

Letzteres ist in der Tat der größte Vorteil der LiFePO4-Akkus: Selbst bei voll geladenem Akku ist nach bisherigen Erkenntnissen ein thermisches Durchgehen ausgeschlossen – und das sogar, wenn der Akku mechanisch beschädigt wird. Ein weiterer nennenswerter

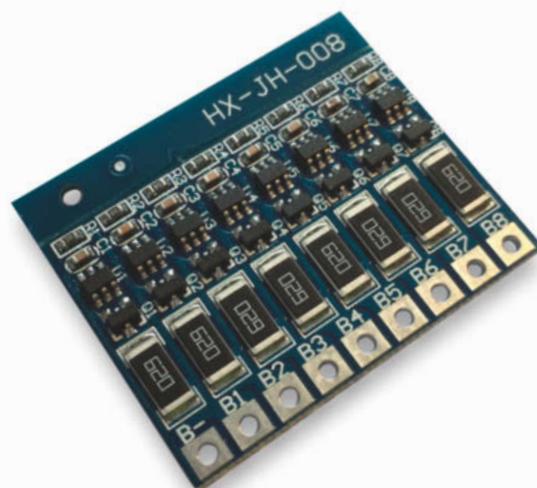
Lade-/Entladekurven

Über einen weiten Ladegrad verlaufen die Spannung beim Laden (grün) als auch beim Entladen (rot) ziemlich konstant





Das Ladegerät muss aufgrund der geringeren Zellspannung unbedingt ein spezielles Programm zum Laden von LiFePO4-Akkus aufweisen. Die dünneren Kabel zum Akku führen zum Balancer-Anschluss des Ladegerätes.



Die meisten modernen Ladegeräte benutzen integrierte Ladecontroller – hier eine diskrete Version.

Vorteil ist die Möglichkeit, Zellen mit einer Leistungsdichte von bis zu 3000W/kg zu konstruieren. Diese lassen sich dauerhaft mit sehr großen Strömen entladen, ohne dass die Zelle geschädigt wird, was sie besonders für stromhungrige Anwendungen prädestiniert.

Lebensdauer

Auch was das Aufladen betrifft, brauchen sich LiFePO4-Zellen nicht hinter anderen Akku-Typen zu verstecken – es gibt Zellen, die sich kurzzeitig mit Spitzenströmen bis zum Sechsfachen der Eigenkapazität aufladen lassen; sie können in weniger als einer halben Stunde auf über 95% ihrer Kapazität geladen werden. Auch der Energienutzungsgrad, welcher das Verhältnis zwischen eingelegener und entnommener Energiemenge wiedergibt, ist verhältnismäßig groß: Liegt dieser zum Beispiel bei einem Bleiakku bei nur etwa 80%, so glänzt hier der LiFePO4 mit deutlich über 90%.

Erfreulich für den Nutzer ist auch die flache Lade- beziehungsweise Entladekurve der Akkus. Im gesamten Betriebsbereich liegt die Spannung auf einem recht konstanten Level, was die Nutzung der enthaltenen Energiemengen ohne aufwendige externe Schaltungen vereinfacht.

Einen Memory-Effekt, wie man ihn beispielsweise von NiCd-Akkus kennt, ist nicht vorhanden. Gemäß einer Studie des schweizerischen Paul-Scherrer-Instituts und der Forschungsabteilung des japanischen Autobauers Toyota zeigen LiFePO4-Akkus einen anderen, jedoch reversiblen Effekt, welcher dort ebenfalls als „Memory Effekt“ bezeichnet wird: Bereits nach der ersten Ladung tritt eine minimale Erhöhung der Ladespannung auf, welche sich vermutlich durch eine unterschiedliche Verteilung von geladenen und

ungeladenen Partikeln in der Elektrode erklären lässt. Beachtet man in der Ladeelektronik diese leicht veränderte Spannungslage, so lässt sich immer noch dieselbe Energiemenge in den Akku laden; darüber hinaus verschwindet der Effekt, wenn man dem Akku nach der Entladung etwas Zeit lässt, um sich zu erholen.

Licht und Schatten

Die Umweltbilanz dieser Akkus ist nicht die schlechteste, was vor allem an zwei Faktoren liegt: zum einen die sehr große Zyklenfestigkeit als auch ein hoher Nutzungsgrad des in der Zelle eingeschlossenen Alkalimetalls, weshalb nur rund halb so viel Lithium eingesetzt werden muss wie bei LiPo- oder Li-Ion-Zellen. Der Verzicht auf Kobalt dürfte sogar noch wichtiger sein, da es sich in Pflanzen ansammeln kann und in hohen Dosen schädlich für Weidetiere ist; daneben ist die Gewinnung aufwendig und teuer.

Preislich liegen (mechanisch) gleich große Li-Ion- und LiFePO4-Zellen heute gleichauf (etwa 8 Euro für die Bauform 18650), die Nennkapazität einer LiFePO4-18650 ist allerdings deutlich geringer (LiFePO4: 4,6Wh = 1400mAh bei 3,2V statt Li-Ion: 9,6Wh = 2600mAh bei 3,7V). Die niedrigere Zellspannung stellt beim Ersatz von Bleiakkus kein Problem dar, denn ein vierzelliger LiFePO4-Akku hat eine ähnliche Spannungslage wie ein konventioneller sechszelliger Blei-Akku (12V).

Li-Ion-Einzelzellen können dagegen nicht ohne weitere Maßnahmen durch LiFePO4-Ausführungen ersetzt werden. Ältere Batterie-Management-Systeme, spezielle Ladegeräte und Balancer-Module sind auf eine Ladeschlussspannung von 4,2V und Entladeschlussspannungen von 3,0V abgestimmt.

Bei einer Umrüstung muss die komplette Elektronik zusammen mit den Zellen ersetzt werden, was im Einzelfall nicht unerhebliche Zusatzkosten verursacht.

Der flache Spannungsverlauf, der für viele Anwendungen wünschenswert ist, erschwert leider die Erkennung des Lade- beziehungsweise Entladezustandes; auch hier ist für Li-Ion-Zellen ausgelegte Elektronik nicht geeignet. Ein weiterer Nachteil betrifft ihre ausgeprägte Kälteempfindlichkeit. Zwar liegt die Temperatur, bei welcher man Energie aus einem LiFePO4 entnehmen kann, je nach Hersteller im Bereich von -20°C bis 60°C, was für die meisten Anwendungen unkritisch ist; geladen werden sollte ein LiFePO4-Akku jedoch nicht bei Temperaturen unter 0°C, wobei auch ein Laden mit großen Strömen bei Temperaturen über 40°C zu einer beschleunigten Alterung führen kann.

Geben und Nehmen

Auch LiFePO4-Akkus sollte man innerhalb der Spezifikationen im Datenblatt behandeln, damit sie nicht kränkelnd und vorzeitig dahinwelken. Der zulässige maximale Strom wird, wie bei Lithium-Akkus üblich, meist in „C“ angegeben, wobei der Wert den Strom in Bezug auf die Zellkapazität angibt: eine 60C-Zelle mit 1,5Ah darf beispielsweise mit $60 \times 1,5A = 90A$ entladen werden. Auch eine Tiefentladung unter die Entladeschlussspannung muss vermieden werden, da es sonst ebenfalls zu einem Kapazitätsverlust und einer verminderten Lebensdauer kommen kann. Wann immer möglich, sollte man nicht die letzten mAh beim Ladevorgang „reinquetschen“ und auch nicht die maximal erreichbare Entladediefe ausnutzen.

Wird die Zelle überladen, kommt es zu einem Überdruck, der sich über ein Ventil ent-

lädt und das (giftige!) Elektrolyt in die Umgebung entlässt – dass die resultierende Schädigung irreversibel ist, liegt auf der Hand. Generell sollten für Zellen ohne Elektronik nur LiFePO₄-geeignete Ladegeräte zum Einsatz kommen, die manuelle Ladung mit einem Labornetzgerät ist aber durchaus auch möglich.

Als Ladeverfahren kommt nur ein Verfahren infrage, bei dem anfangs der maximale Ladestrom begrenzt und nach Erreichen der Ladeschlussspannung diese durch eine Spannungsregelung nicht überschritten wird; in der Technik wird dies Verfahren als CCCV bezeichnet (Constant Current Constant Voltage). Am Ende des Ladevorgangs sinkt die Stromaufnahme des Akkus immer weiter, bis dieser fast keinen Strom mehr aufnimmt.

Zunächst muss man ohne angeschlossenen Akku die Ausgangsspannung auf die im Datenblatt angegebene Ladeschlussspannung einstellen – und zwar auf ein paar Millivolt genau, im Zweifelsfall also mit einem Digital-Multimeter nachmessen. Die Strombegrenzung des Netzteils wird nun nach Anschließen des Akkus auf den maximalen Ladestrom eingestellt.

Während des Ladevorgangs wird die Spannung langsam auf die Ladeschlussspannung ansteigen; das Netzteil wird dann vom stromgeregelten (CC) in den spannungsgeregelten Modus (CV) wechseln. Sinkt der Strom auf weniger als 0,1C bei voller Spannung, so ist der Akku geladen und kann abgeklemmt werden.

Auf diese Weise lassen sich zumindest Einzelzellen laden; bei Akkupacks funktioniert das nur, wenn sie ein BMS (Battery Management System, überwacht die einzelnen Zellen im Pack) integriert haben, ansonsten muss ein Ladegerät mit einem Balancer-Anschluss zum Einsatz kommen. Denn nur so ist sichergestellt, dass alle Zellen innerhalb des Verbundes auf die gleiche Spannung geladen werden.

Balance-Akt

Bereits ab etwa 10 Euro erhält man geeignete Ladegeräte mit integriertem Balancer. Ein von mir erprobtes Ladegerät, das so ziemlich alle Features aufweist, ist das ISDT Q6 Pro (Bezugsquelle siehe Info-Kasten). Der Eigenbau eines geeigneten Ladegerätes erscheint aufgrund der günstigen Preise kaum sinnvoll; wer sich trotzdem daran versuchen will, findet im Internet diverse Ladeschaltungen.

Da die Spannung einer Einzelzelle für viele Anwendungen zu niedrig ist, ist es üblich, mehrere der Zellen in Reihe zu schalten. Die Anzahl der in Reihe geschalteten Zellen wird in „S“ (wie „seriell“) angegeben. Will man Kapazität und Strombelastbarkeit erhöhen, schaltet man die Zellen parallel, wobei

hier die Anzahl mit einem „P“ (wie „parallel“) gekennzeichnet wird; ein 4S3P Pack besteht somit aus 12 Einzelzellen.

Selbst gleiche Zellen aus modernsten Fertigungsanlagen weisen geringe Fertigungstoleranzen auf; schaltet man die Zellen ohne weitere Maßnahmen in Reihe, würde die Zelle mit der geringsten Kapazität zuerst ihre Ladeschlussspannung erreichen, bei weiterem Laden also Schaden nehmen, während die anderen Zellen noch nicht ganz voll sind. Um das zu verhindern, verwendet man Ladegeräte mit Balancer-Funktion (und demzufolge zusätzlichen Anschlüssen zum Abgreifen der Spannung jeder einzelnen Zelle). Um die Handhabung zu vereinfachen, weisen Akkupacks, wie sie zum Beispiel in Elektrofahrrädern verwendet werden, oft fest verbaute Battery-Management-Systeme auf, welche dafür sorgen, dass alle Zellen des Packs stets innerhalb der Herstellervorgaben geladen und betrieben werden.

Werden die Zellen nicht genutzt, wirkt sich (wie bei Li-Ion-Akkus auch) ein mittlerer Ladezustand positiv auf die Lebensdauer aus. Sind die Zellen knackevoll geladen, verringern die verstärkt ablaufenden chemischen Prozesse die Lebensdauer, eine leere Zelle kann sich aufgrund der Selbstentladung nach einiger Zeit tiefentladen.

Passend gemacht

Weit verbreitet sind besonders die von der Bauform mit den Li-Ion-Akkus vergleichbaren zylindrischen Zellen: Auch hier gibt es die „klassischen“ Ausführungen wie den 18650- und den 26650-Akku, wobei die Zahl die Abmessungen der Zelle widerspiegelt.



Bild: Walimex

Der LiFePO₄-Akku von Walimex für Hochleistungs-Blitzanlagen erleichtert Fotografen den On-Location-Einsatz – im wahren Wortsinn.

Beispielsweise hat eine 18650-Zelle einen Durchmesser von 18mm Durchmesser bei einer Länge von 65mm. Diese Bauformen werden oft für portable Consumer-Geräte eingesetzt und weisen Kapazitäten im Bereich einiger Ah auf. Größere Bauformen wie die 38140 (38mm x 140mm) kommen eher in Elektrofahrzeugen und in der Industrie zum Einsatz.

Neben den zylindrischen Zellen sind auch prismatische (rechteckige) Zellen verbreitet. Die eignen sich besonders für den Einsatz in größeren Akkupacks oder Photovoltaik-Energiespeichern. Zellen ohne eine harte äußere Hülle werden als Folienzelle bezeichnet und lassen sich in fast jeder beliebigen Größe herstellen. Da sie ohne ein Gehäuse mechanisch relativ empfindlich sind, stellen sie nur ein industrielles Zwischenprodukt dar, vor der eigentlichen Nut-



Bild: Liontron

Akkus von Liontron: nicht billig, aber „Rundum-Sorglos-Pakete“ zum Ersatz von Bleisammlern



Wichtig bei der Verwendung eines derart leistungsstarken Akkus ist eine Verwendung dicker Kabel und eine Absicherung möglichst dicht am Pluspol. Gut zu erkennen auch der Temperatursensor am Minus-Pol zur Optimierung der Ladekennlinie des Solar-Ladereglers.

zung muss man sie zum Schutz fest in ein Gerät oder ein Gehäuse einbauen.

In der Praxis

Aufgrund der vielen positiven Eigenschaften haben die LiFePO4-Akkus inzwischen weite Verbreitung gefunden: vom Einsatz von Einzelzellen zum „Dampfen“ über die Verwendung in kleineren Zellverbänden beim Modellbau oder in Elektrowerkzeugen bis hin zu

gigantischen Zellenblöcken mit Kapazitäten von vielen tausend Amperestunden als Energiespeicher in Solaranlagen oder U-Booten. So kommen in der chinesischen Yuan-Klasse Akkus von Winston Battery mit 32MWh zum Einsatz, ThyssenKrupp Marine Systems plant Ähnliches.

Outdoor-Freuden

Um die Zielgruppe der Segler, Camper und Wohnmobilsten für die LiFePO4-Technik zu begeistern, gibt es spezielle „pinkompatible“ Akkus, die Blei-Akkus ohne weitere Maßnahmen ersetzen können. Einen besonders smarten Vertreter liefert der Hersteller Liontron. Er bietet eine Lebensdauer von bis zu 3000 Zyklen sowie eine geringe Selbstentladung von nur rund 3% im Monat (zum Vergleich: Bleiakku 4 bis 6%, NiMH-Akku 15 bis 25%). Auch was die Entladetiefe angeht, ist ein solcher Akku den gängigen Bleisammlern deutlich überlegen, da fast die gesamten 100Ah des Akkus nutzbar sind. Die extreme Strombelastbarkeit von kontinuierlich 200A lässt es sogar zu, über einen Wechselrichter Haartrockner oder Winkelschleifer zu versorgen.

Scheidungsgrund

Nachdem ich die geringe Lebensdauer von Bord-Bleiakkus satt hatte, lag die Entscheidung nahe, den erschöpften Bleiakku meines Wohnmobils durch einem LiFePO4-Akku von Liontron zu ersetzen – immerhin eine Investition von rund 1000 Euro. Da der Akku uneingeschränkt mit Blei-Bordbatterien kompatibel ist, erübrigte sich jede Veränderung an der bisher bestehenden Elektroinstallation.

Ein deutlich spürbarer Vorteil des Akkus ist, dass er jede Menge Volumen und Gewicht spart: Der 100-Ah-Liontron wiegt bei Abmessungen von 345 × 215 × 175mm nur rund 14kg. Das ist besonders beim Einsatz in Wohnmobilen bis 3,5 Tonnen, bei denen man auf jedes Kilo achten muss, ein großer Vorteil.

Mit leistungsfähiger Lichtmaschine reichen wenige Stunden Fahrt, um den Akku voll zu laden. Ähnlich sieht es bei der Verwendung von Solarmodulen aus – der Liontron nimmt an Strom, was er kriegen kann, bis er komplett geladen ist. Besonders in den Bergen, wo tagsüber die Kühlbox und nachts die Heizung viel Strom verbrauchen, ist das ein unschätzbare Vorteil. Das i-Tüpfelchen ist schlussendlich das Feature, den Akku mittels eines Mobiltelefons über das im BMS integrierte Bluetooth-Modul zu überwachen.

Auch der Liontron-Akku reagiert empfindlich auf Minusgrade: Laut Datenblatt darf man ihn nur im Bereich von -5°C bis +45°C laden. Doch selbst wenn man mit einem kalten Fahrzeug losfährt, wird durch die verbauten Elektronik der Ladestrom so lange begrenzt, bis sich der Akku im beheizten Innenraum genügend erwärmt hat, um den vollen Ladestrom aufzunehmen. Den einzigen wirklichen Nachteil stellt somit der deutlich höhere Anschaffungspreis dar, was allerdings durch die höhere Lebensdauer relativiert wird.

Auch für andere größere Geräte, wie zum Beispiel mobile Roboter oder elektrische Rollstühle, bieten große LiFePO4-Akkus die perfekte Basis, weniger dagegen als Starterbatterie für Motorräder und Autos: Schönwetterfahrer kommen zwar noch in den Genuss ihrer Vorteile, ihre Kaltstartleistung kann aber mit der von modernen AGM-Bleiakkus (Absorbing Glass Mat, Elektrolyt ist in einer Glasfasermatte gebunden) nicht mithalten.

Natürlich wird im Bereich der LiFePO4-Akkus ständig weiter entwickelt, um die Zellen noch stärker und universeller zu machen. Derzeit kommen mehr und mehr Yttrium-dotierte Zellen (LiFeYPO4) auf den Markt, die einen erweiterten Betriebstemperaturbereich aufweisen. Mit Kosten von rund 800 Euro für eine einzige gespeicherte Kilowattstunde muss man auch hier rechnen. Eine leistungsfähige Alternative zum LiFePO4-Akku könnte in naher Zukunft der LiHV-Akku werden – das ist im Prinzip ein Lithium-Polymer-Akku, bei dem die Anode durch ein Silizium-Graphene-Gitter ersetzt wurde. Seine Hochstrom-Eigenschaften sind dem LiFePO4 ebenbürtig, allerdings weist er eine höhere Zellenspannung (3,8V, daher HV = High Voltage) und Kapazität auf. Im Modellbau-Bereich wird schon eifrig damit experimentiert.

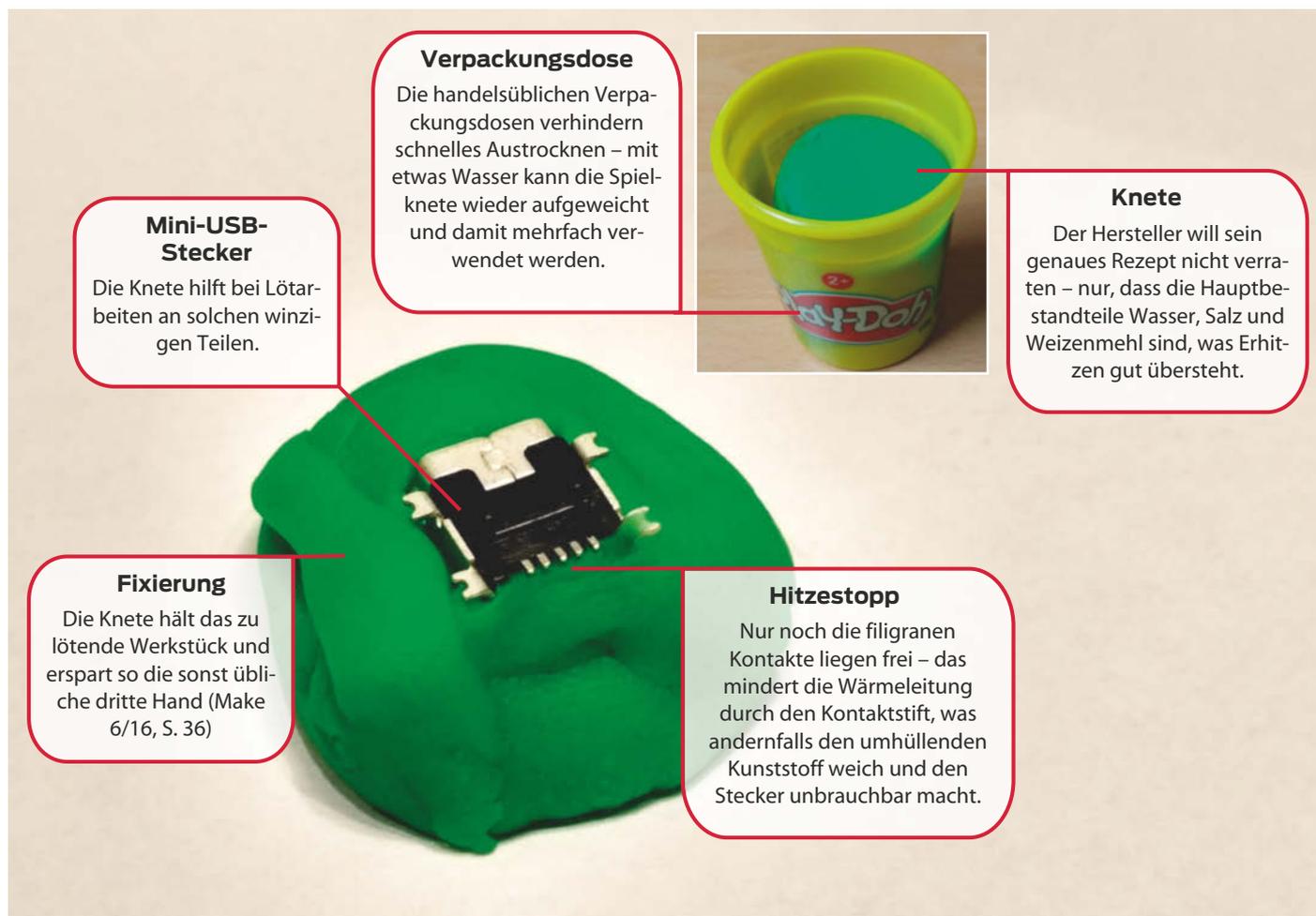
—cm



Besonders sinnvoll an der Liontron-App ist die Anzeige der verbleibenden Akkukapazität und des derzeit fließenden Stroms.

Knete als Löthilfe

von Martin Neuhaus

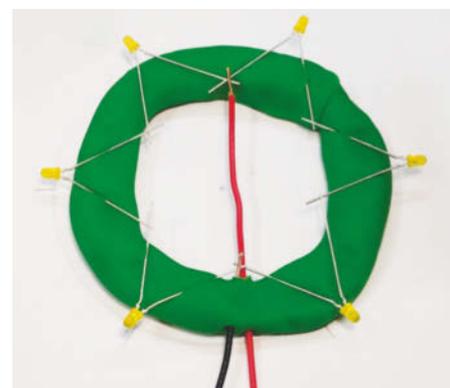


Bereits vor einigen Jahren las ich einen Artikel zur Anwendung einer knetbaren, Kitt-ähnlichen Masse, die beim Schweißen als Hitzeblockade auf das lackierte Blech gedrückt wird, um die entstehende Wärme auf den lokalen Reparaturbereich zu begrenzen. Ähnliche hitzebeständige Modelliermassen gibt es für Hartlötanwendungen – einerseits, um auch dort die Wärmebeaufschlagung zu begrenzen, andererseits, um die Benetzung mit Lot auf den erforderlichen Bereich zu beschränken. Ein großer Vorteil der knetbare Masse ist dabei, dass man sie zusätzlich als flexible Montagehilfe nutzen kann. Das spart die dritte Hand und schützt vor schmerzhaften Verbrennungen!

Diese Eigenschaften nutze ich im kleineren Maßstab für filigranere Elektroniklötarbeiten und habe dafür keine spezielle

Masse im Einsatz, sondern bediene mich an der bunten Spielknete meiner Kinder. Meine Anwendungen sind etwa Reparaturen an Miniaturkontaktstiften von Steckern, deren Vergoldung dafür sorgt, dass das Lötzinn bereits nach kurzer Erwärmung mit dem LötKolben zur Kontaktfläche steigt und bei Benetzung das ganze Teil ruiniert.

Wird stattdessen der Stecker in Knete gepresst, schaut nur noch der Lötbereich heraus und die empfindliche Seite wird geschützt. Fliegende Verdrahtungen lassen sich ebenfalls einfacher erstellen, wenn die Bauteile in einer ausgerollten Knetwurst eingedrückt und positioniert werden. Da kommen dann zusätzlich auch Kindheitserinnerungen auf.



Anwendungsbeispiel ähnlich dem LED-Weihnachtsstern aus Make 12/14: Anstatt eine Schablone zu bohren, werden die LEDs in einem Ring aus Knete angeordnet.

—pek

Ausprobiert
— von Make: —

DANIU HY-18

**Günstige
Wärmebildkamera**



Die handliche Kamera hat eine Auflösung von 32×24 Pixeln und eine Messfrequenz von 8Hz. Das helle und kontrastreiche 2,4-Zoll-Display zeigt das Infrarot-Bild und die wichtigsten Messwerte (minimale und maximale Temperatur im sichtbaren Bereich sowie Temperatur im Zentrum) an und markiert die entsprechenden Messpunkte. Das reicht aus, um heiße Komponenten auf einer Platine aufzuspüren, ein Loch in einer Fensterdichtung zu finden oder die Qualität der Isolation von Heizungsrohren zu bewerten.

Der Akku hält etwa 2,5 Stunden und wird über den USB-Mini-Anschluss geladen, über den man sich die Daten auch auf den PC holen kann. Billig wirkt nur das Gehäuse: Es gibt keine Beschriftung der Tasten, keine Schutzfolie vor dem Display und auf der anderen Seite ragt die Linse des Sensors aus dem Gehäuse. Andererseits macht ja gerade ihr Preis diese Kamera so attraktiv.

Einen ausführlichen Testbericht lesen Sie online (siehe Link).

—Dirk Herrendorfer/pek

► make-magazin.de/xj18

Hersteller DANIU
URL <https://ban.ggood.vip/gcmv>
Preis ca. 115 €

Smartwatch Bangle.js

JavaScript fürs Handgelenk

Smartwatches erfreuen sich immer größerer Beliebtheit, lassen sich meist aber nur schwer um eigene Apps erweitern. Davon war Gordon Williams, der Vater des Espruino-Projekts, wenig begeistert und hat eine in JavaScript programmierbare Smartwatch entworfen – genauer gesagt hat er eine bestehende Uhr angesehen, einem Reverse Engineering unterzogen und neue Espruino-basierte Firmware geschrieben. Das Ergebnis wurde als Bangle.js in wenigen Tagen auf Kickstarter finanziert und wird künftig im Espruino-Shop erhältlich sein.

Die wasserdichte Smartwatch hat Bluetooth, 64kB internen RAM und 512kB Flash. Dazu kommt ein 16-Bit-LCD-Display mit zwei Touchzonen. Die üblichen Smartwatch-Features wie GPS, Pulsmessung, Vibrationsmotor und Lautsprecher sind auch dabei. Mit ihrem 350mAh-Akku überlebt die Uhr eine Woche im Standby. Dank des Espruino-Interpreters lassen sich alle Hardware-Kompo-



Bild: Espruino

Ausprobiert
— von Make: —

nenten mit eigenen Programmen nutzen. Als besonderes Schmankehl gibt es TensorFlow Lite für Mikrocontroller obendrein.

Einen ausführlichen Testbericht lesen Sie online (siehe Link). —Maik Schmidt/rehu

► make-magazin.de/xj18

Hersteller Espruino
Vertrieb shop.espruino.com
Preis ab 55 €

IO-Warrior 28

USB-I/O-Baustein mit einfacher Ansteuerung

Steuerungsaufgaben über USB kann man natürlich mit einem selbst programmierten Arduino erledigen. Wem der Aufwand dafür zu hoch und der erzielbare Datendurchsatz zu gering ist, sollte zum IO-Warrior 28 greifen, den es als Breadboard-fähiges Modul oder als nackten QFN-Chip gibt. Das Modul stellt 19 I/O-Leitungen am USB zur Verfügung; davon lassen sich einige Pins als I²C-Schnittstelle oder Analog-Eingänge umprogrammieren, womit sich vielfältige Einsatzmöglichkeiten (nicht nur) in der Mess- und Regeltechnik ergeben.

Der IO-Warrior meldet sich als HID (Human Interface Device) am Rechner und benötigt deshalb keinen eigenen USB-Treiber. Die Ansteuerung der Pins und Funktionen erfolgt über den gleichen Report-Mechanismus, wie ihn auch Tastaturen oder Joysticks verwenden. Zur Anbindung an Hochsprachen dient eine DLL. Beispiele für C, Delphi (Object Pascal), MathLab und LabVIEW (hier leider nicht ganz aktuell) werden mitgeliefert.

I/O-Pins sind immer „Open-Drain“-Ausgänge, schalten also nur nach Masse. Dadurch vereinfacht sich die Einstellung der Datenrichtung: Pins, die man auf „high“ gelegt hat, sind automatisch auch Eingänge. Als HID liefert der IO-Warrior eigentlich nur



Ausprobiert
— von Make: —

Änderungen der Eingangspegel an den Rechner; eine DLL-Funktion liest aber auch explizit den gerade aktuellen Zustand. Aufpassen muss man beim Einsatz des integrierten Vierkanal-AD-Wandlers: Kanalzahl und Samplingrate (bis 30kHz) lassen sich nur ändern, wenn man ihn vorher abschaltet. —cm

Hersteller CodeMercenaries
URL www.codemercs.com
Preis 16,90 €

Vogelflug

Fünf Papervögel im Papercraft-Set

Aus diesem Papercraft-Kit entstehen fünf Vögel aus festem Papier, die zusammen einen Flügelschlag-Zyklus abbilden. Die Modelle sind angeschnitten und eignen sich daher, um sie als Skulptur an die Wand zu hängen.

Jeder Vogel besteht aus nur zwei Bögen Papier etwa im A4-Format und wird – dank kluger Faltungen – überraschend groß. Die Modellbau-Bögen sind vorgeschritten, durchnummeriert und an den Falzkanten perforiert. Der Karton wartet mit einer angenehmen Naturpapier-Haptik auf. Trivial ist der Aufbau nicht. Absolute Papercraft-Anfänger sollten sich vorm Bau die Zeit nehmen, die Anleitung ausgiebig zu studieren, bis sie die Konstruktion verstanden haben. Klebt man einfach munter drauflos, geht es schief. Besonders schnell kann es passieren, dass man ein einzelnes Element des Vogels spiegelverkehrt faltet und verklebt. Das bemerkt man leider erst zu spät, nämlich dann, wenn man versucht, es mit den anderen Teilen zusammzusetzen. An dieser Stelle wäre eine ausführlichere Anleitung angebracht. Dafür ist die Freude groß, sobald einzelne Formen zusammen plötzlich einen Schwanz oder Flügel bilden. Der Aufbau eines einzelnen Vogels kann ein bis zwei Stunden dauern – für Einsteiger, die sich erst mit der Technik vertraut machen müssen, sogar länger.



Das Ergebnis kann sich dafür sehen lassen: Die Konstruktion der Vögel ist offenkundig durchdacht. Die polygonalen Flächen und Kanten sind so gewählt, dass die Vögel – trotz ihrer reduzierten Form – organisch und lebendig wirken. Wer sich an Pepakura versuchen möchte, kann sich sicher sein, dass er hinterher mit dem Vogelflug-Kit ein sehenswertes Ergebnis in den Händen hält.

Sein Erschaffer, Wolfram „Paperwolf“ Kampffmeyer, hat seit 2010 noch viele weitere polygonale Papercraft-Modelle von Tieren im Angebot – von Fuchs bis Walross. —*rehu*

Hersteller	Paperwolf
URL	www.paperwolf.de
Preis	50 €

Ausprobiert
— von Make: —

LOBOT xArm

Roboterarm mit Digitalservos und Controller

Unter den erschwinglichen „Education“-Roboterarmen sticht das getestete Banggood-Angebot etwas hervor: Statt mit billigen Modellbau-Servos ist es mit kräftigen Digitalservos ausgerüstet, die über einen Ein-Draht-Bus gesteuert werden und eine genauere Positionierung erlauben. Dem Bausatz liegt eine im Fuß untergebrachte Controller-Platine zur Fernsteuerung über WLAN oder USB sowie ein Gamepad zur Teach-in-Programmierung bei.

Im Unterschied zu vielen anderen chinesischen Angeboten sind Aufbau, Betrieb und Protokolle brauchbar dokumentiert. Der Zusammenbau ist in zwei Stunden erledigt, allerdings erfährt man im Video-Tutorial nicht, in welcher Reihenfolge die adressierten Digitalservos eingebaut werden

müssen (Servo 1 ist der Greifer, dann fortlaufende Nummerierung). Die durch das Bussystem eigentlich simple Verdrahtung wird durch schlecht verarbeitete Stecker erschwert.

Mehrere Möglichkeiten zur Steuerung stehen zur Verfügung: über USB und dem Remote-Programm (Windows), über das Drahtlos-Gamepad oder per WLAN über eine App. Besonderer Clou ist, dass man die Servo-Motoren abschalten kann, um den Arm vorsichtig manuell in die zu programmierende Position zu bringen. Die Gelenkwinkel werden dann aus den Servos ausgelesen und als Programmschritt abgespeichert. Die Positioniergenauigkeit dürfte im Bereich einiger Millimeter liegen, die Traglast ist mit 200g angegeben. —*cm*

Ausprobiert
— von Make: —



Hersteller	Lobot
URL	www.lewansoul.com/product/detail-27.html
Preis	ca. 200 €

DIY-Retro-Joystick

Platine zum Selbstkostenpreis



Bild: Stephan Eckweiler



Zwei Knöpfe, ein Joystick und Anschlüsse für alte Konsolen (D-Sub) wie neue Rechner (USB) – das Retro-Projekt von Stephan Eckweiler begeisterte Anfang des Jahres die Make-Leserschaft (Ausgabe 3/19, S. 86) und den LED-Hersteller Emilum, der für die Entwicklung einer Joystick-Platine seine Hilfe anbot. Die fertige Platine gibt es zum Unkostenbeitrag jetzt bei Emilum zu bestellen.

Einzelne Platinen bekommt man für 26,40 Euro (inklusive Steuern und Versand nach Deutschland und Österreich). Kommen genügend Bestellungen zusammen, wird der Preis voraussichtlich noch einmal sinken, daher sind auch Sammelbestellungen ausdrücklich erwünscht. Die Platinen gibt es nicht in einem Shop, sondern ausschließlich über die Mail-Adresse unten in der Tabelle. Die Bezahlung erfolgt über Paypal.

Neu auf der Platine ist übrigens ein zweiter logischer Feuerknopf, der besonders die Amiga- und MAME-Fraktion erfreuen dürfte. Die Platine ist weiterhin ein Open-Source-Projekt und alle Dateien zum Nachbauen, inklusive dem Platinendesign, stehen auf Gitlab zum Herunterladen zur Verfügung (siehe Link). —*ch*

► make-magazin.de/xj18

Hersteller	Emilum
Mail	retrojoystick@emilum.com
Preis	26,40 €

MKR WAN 1310

LoRa-fähiger Mikrocontroller von Arduino

Möchte man kleine Datenpakete mit möglichst geringem Energieaufwand über große Entfernungen funken, ist LoRaWAN das Protokoll der Wahl. Da viele per LoRaWAN angebundene IoT-Knoten wie Sensorstationen in der Wildnis möglichst sparsam mit dem Energievorrat in der Batterie umgehen müssen, soll sich das neue Arduino-Board MKR WAN 1310 bei richtiger Konfiguration mit 104µA Stromaufnahme begnügen.

Als Prozessor kommt ein sparsamer SAM D21 von Microchip zum Einsatz und für die Verbindung das LoRa-Modul CMWX1ZZABZ von Murata. Ebenfalls an Bord sind ein ECC508-Kryptochip für die verschlüsselte Speicherung von Zugangsdaten und Zertifikaten sowie 2MByte SPI-Flashspeicher etwa für Konfigurationsdateien oder als Datenpuffer. Das Board soll sich sowohl über

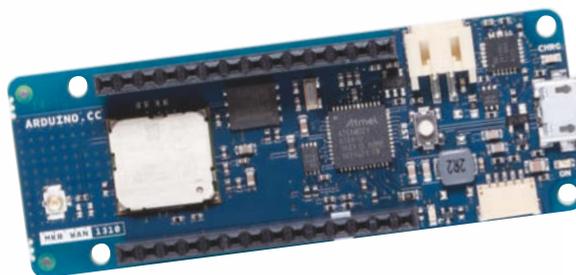


Bild: Arduino

Arduinos eigene IoT-Cloud bei *Arduino Create*, mittels eines selbst betriebenen LoRa-Gateway von Arduino, aber auch über bestehende LoRaWAN-Infrastruktur wie *The Things Network* (siehe Make 4/18, S. 46) anfunken lassen. —*pek*

Hersteller	Arduino
URL	store.arduino.cc/mkr-wan-1310
Preis	33 €

Personal Robotics

Laufroboter aus dem Lasercutter

Wer gerne einen zweibeinigen Roboter bauen möchte und Zugriff auf einen Lasercutter hat, findet auf Wikifactory mit den beiden Open-Source-Modellen von *Personal Robotics* zwei interessante Plattformen: Der Kopffüßler *BOB-303* (siehe Bild) baut auf dem bekannten *Otto*-Muster auf (siehe Make 2/18, S. 90), ist mit drei Servos pro Bein aber besser zu Fuß. Er wird durch einen Raspberry Pi Zero gesteuert und ist mit einem Distanzsensoren sowie einer Kamera ausgestattet. Wer will, kann ihn von einem externen Computer mittels ROS fernsteuern, auf dem dann die aufwendigeren Algorithmen für Bildverarbeitung oder die Selbstlokalisierung erledigt werden.

Mechanisch deutlich ambitionierter ist die große Schwester *lilli* (sprich: *Lilli*), die aufgerichtet 75cm hoch ist und der 25 Servos ebenso viele Freiheitsgrade verleihen. Paten bei der Konstruktion standen die beiden DIY-Humanoiden *Halley* und *Poppy*, die jedoch beide aus dem 3D-Drucker stammen. Für *lilli* bekommt man zwar die Schnittdateien für den Lasercutter und eine Bauanleitung für den Aufbau des Chassis, aber derzeit gibt

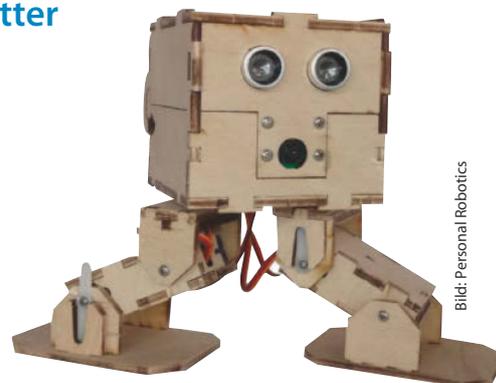


Bild: Personal Robotics

weder ein empfohlenes Hardware-Rezept noch eine fertige Software – es handelt sich um eine offene Plattform, die aus der Community heraus mit Leben gefüllt werden darf und soll.

Personal Robotics ist ein Ein-Mann-Projekt von Per R. Ø. Salkowitsch. Wer in diesem Jahr auf den Maker Faires in Wien oder Rom war, konnte BOB-303 und *lilli* dort schon live erleben und ihren Schöpfer ebenso. —*pek*

Hersteller	Per R. Ø. Salkowitsch
URL	wikifactory.com/@personalrobotics
Preis	kostenlos

Longer Orange 10

LCD-SLA-3D-Drucker

Dieser Drucker härtet UV-empfindliches Kunstharz durch selektive Belichtung mittels eines Matrix-LED-Boards durch ein LCD aus. Dessen Auflösung von 854 × 480 Pixeln bei 98mm × 55mm Größe ergeben eine Auflösung von 0,115mm. Das ist nicht herausragend, aber brauchbar und deutlich feiner als ein FDM-Druck. Die maximale Objekthöhe beträgt 140mm. Das Gehäuse ist aus Metall und sehr stabil, die Mechanik zweckmäßig, allerdings dürfte der Lüfter gerne leiser sein.



Die verwendeten Matrix-LEDs verursachen zwar ein leichtes Verwaschen der Ränder des projizierten Bildes, aber der kurze Abstand zwischen LED und Harz sowie die diffuse Belichtung härten das Material sehr schnell aus und verhindern Einschlüsse flüssigen Harzes zwischen belichteten Pixeln. Im Test gelang jeder Druck mit dieser Maschine, das ist auch bei Resin-3D-Druckern keine Selbstverständlichkeit.

Die übersichtliche Anleitung ist einsteigerfreundlich und erklärt auch ein paar Kleinig-

Ausprobiert
— von Make: —

keiten, die andere Hersteller gerne weglassen, etwa dass die Reste des Harzes korrekt entsorgt werden müssen oder wie Support-Strukturen funktionieren. Auch die Software zur Druckvorbereitung (Slicer) ist auf Nutzer zugeschnitten, die lieber in sechs Schritten von der STL-Datei zum Druck geführt werden, als an hunderten Parametern zu schrauben.

Insgesamt sticht der Drucker in keiner einzigen Disziplin zwischen den Konkurrenz-

produkten auf dem Markt hervor, aber die Kombination stimmt: Er ist derart einfach zu bedienen, dass es Spaß macht, damit „mal eben schnell“ was zu drucken.

Einen ausführlichen Testbericht lesen Sie online (siehe Link). —Dirk Herrendoerfer/pek

► make-magazin.de/xj18

Hersteller	Longer3d.com
Bezugsquellen	BangGood, GearBeast
Preis	etwa 250 €

Creality Silent Mainboard V1.1.5

Mainboard für 3D-Drucker Creality Ender 3

Der Creality Ender 3 gehört zwar zu den leiseren 3D-Druckern, störend ist seine Geräuschentwicklung aber trotzdem. Vom Hersteller kommt nun ein neues Mainboard für das Gerät, das dank anderer Treiber-ICs für die Steppermotor-Ansteuerung einen deutlich leiseren Betrieb verspricht.

Da das Board von der Größe und der Lage der Anschlüsse völlig kompatibel zur Standardversion ist, kann man es schnell tauschen. Mit der vorinstallierten Marlin-Firmware (Version 1.1.5) war der Beruhigungs-Effekt im ersten Testlauf bereits sehr deutlich, das Stepper-Geräusch geht völlig im Rauschen der Lüfter unter (dazu auf Seite 96 noch ein Tipp).

Die Firmware arbeitet jedoch nicht mit BLTouch- beziehungsweise 3DTouch-Sensoren zusammen (siehe Seite 108), weswegen wir sie durch die aktuelle Version 1.1.9 ersetzen. Dabei zeigte sich ein weiterer Vorteil des Boards: Es enthält bereits den zum Firmware-Wechsel erforderlichen Bootloader. Diese Modifikationen machen allerdings Änderungen



Ausprobiert
— von Make: —

in der Konfigurationsdatei und beim Lüfteranschluss nötig – mehr dazu in der ausführlichen Online-Version unseres Tests (siehe Link unten).

Alles in allem hält das Board, was es verspricht: Der Drucker wird angenehm leise. Der Preis von etwa 40 Euro ist dafür durchaus angebracht. —hgb

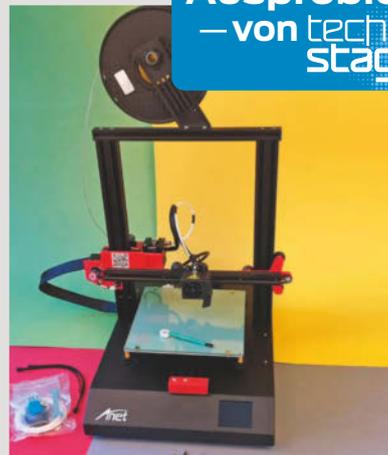
► make-magazin.de/xj18

Hersteller	Creality
URL	www.creality.com
Preis	40 €

Anet ET4

FDM-3D-Drucker

Ausprobiert
— von tech-stage



Mit dem ET4 verspricht der Hersteller Anet einen günstigen 3D-Drucker mit Touchscreen-Steuerung, Filament- und bei Bedarf ansteckbarem Level-Sensor für das beheizte Druckbett. Der Druckraum fällt mit 22cm × 22cm × 25cm großzügig aus. Der Filament-Sensor erkennt, falls das Material reißt oder zu Ende geht, die Maschine pausiert dann den Druck.

Vor dem ersten Druck muss man die Maschine noch montieren und die elektronischen Komponenten zusammenstecken. Im Test unserer Kollegen von Techstage.de war das zwar nach einer Viertelstunde erledigt – bis der Drucker aber schließlich lief und gute Werkstücke produzierte, mussten sie noch knapp zehn Mannstunden Arbeit investieren, wobei sie auf diverse schlampig gebaute Details, Fehlkonstruktionen oder schlicht Bugs stießen. Ihren ausführlichen Testbericht mit vielen Bildern lesen Sie online.

Trotz der guten Ausstattung und seines potenziell ordentlichen Druckbildes kann man den ET4 nur sehr bedingt und für erfahrene Nutzer empfehlen. Wer seinen Drucker als reines Werkzeug betrachtet, sollte sich ein anderes Produkt mit weniger Problemen suchen. —pek

► make-magazin.de/xj18

Hersteller	Anet
Vertrieb	Amazon
Preis	ab 190 €

Kunststoffe für Modellbauer



Kunststoffe sind für den Modellbau sehr interessant. Einen Überblick über ihre Eigenschaften, Einsatz- und Verarbeitungsmöglichkeiten will dieses Buch geben.

Zu Beginn erhält man gute Infos zu Kunststoffarten und deren Eigenschaften, wie Rohstoffe, Festigkeit, Gewicht, Formbarkeit oder Chemikalienbeständigkeit. Dann folgen vor allem Verarbeitungstipps, etwa zum Verkleben unterschiedlicher Materialien. Oft ist der Autor aber unkonkret bei seinen Angaben. So hilft die Bezeichnung „aliphatische Klebstoffe“ ohne Handelsnamen nicht weiter.

Den Platz für die zahlreichen Verweise auf Herstellerangaben hätte man besser für die darin enthaltenen Infos oder zumindest die Nennung der Hersteller nutzen können. Bei Bezugsquellen bleibt das Buch ebenfalls vage. So soll es Farben für Nylon im Baumarkt geben. Uns gelang es nicht, sie dort zu finden.

Für die wichtigen Informationen erspart das Buch also keine Internet-Recherche. Warum soll man es dann kaufen? —hgb

Autor	Alex Weiss
Verlag	Verlag für Technik und Handwerk
Umfang	164 Seiten
ISBN	978-3881807692
Preis	22,00 €

Das große Modellakku-Buch

Grundlagen und Praxistipps

Schwirrt Ihnen auch schon der Kopf von Begriffen wie CCCV, Lithium-HV, Ladereise, Lazy-Battery-Effect, LiFePO4 und 4S3/6C? Jede neue Akku-Technologie wirft Fragen auf und verunsichert potenzielle Anwender. Das große Modellakku-Buch will als Rundumschlag aufklären und liefert eine Maker-gerechte Mischung aus Theorie und praktischer Anwendung. Es wendet sich zwar vornehmlich an Modellbauer, aber die sind ja nun mal auch Hardcore-Maker.

Den Kapiteln merkt man an, dass der Autor vom Fach ist – unter anderem hat er kommerzielle Batterie-Messgeräte entwickelt. Ungenauigkeiten oder vage Formulierungen leistet er sich nicht, stattdessen weiß er durchaus populärwissenschaftlich zu schreiben. Gelungen sind die einzelnen Übersichten und Tabellen



zu den Akku-Familien – man weiß gleich, was man von der jeweiligen Technologie zu erwarten hat.

Auch die Praxis kommt nicht zu kurz: So gibt es reich bebilderte Tipps, wie man Akkupacks selbst konfektioniert, also verlötet und einschrumpft. Natürlich ist manches eher für Fernsteuerungsfreunde interessant, etwa die Auswahl von Akkus für Sender- und Empfängersysteme. Ein eigenes, gut recherchiertes Kapitel ist dem Thema Ladege-

räte und Ladetechniken gewidmet. Alles in allem ein sehr empfehlenswertes Buch auch für Nicht-Modellbauer. —cm

Autor	Ulrich Passern
Verlag	vth Verlag für Technik und Handwerk
Umfang	130 Seiten
ISBN	978-3-88180-495-0
Preis	19,90 €

KiCad wie ein Profi

Für Hobby-Elektroniker, die nur ab und zu eine Platine entwerfen, ist das freie ECAD-Programmpaket KiCad eine gute Alternative zur kommerziellen Konkurrenz. Obwohl es kostenlos ist, strotzt es nur so vor Funktionen und Peter Dalmaris will mit seinem Buch den Einstieg erleichtern.

Nach einer kurzen Einführung in die wichtigsten Eigenschaften von Platinen entwickelt er eine kleine Leiterplatte, um eine LED zum Leuchten zu bringen. Das triviale Beispielprojekt reicht aus, um einen erheblichen Teil der KiCad-Funktionen kennenzulernen. Haarklein erklärt der Autor jeden Schritt, angefangen bei der schematischen Darstellung des Schaltplans über die Erstellung des Layouts bis hin zum dreidimensionalen Modell. Dabei achtet er auf möglichst effiziente Arbeitsabläufe.

Nach diesem Experiment wagt er sich an drei größere Projekte: eine Stromversorgung für Breadboards, einen Raspberry Pi-HAT mit einem DHT22-Sensor zur Messung von Tem-



peratur und Luftfeuchtigkeit und schließlich einen kompletten Arduino-Klon.

Diese Projekte beschreibt er nicht mehr ganz so ausführlich wie das erste, sondern konzentriert sich auf deren Besonderheiten und wie man ihnen mit KiCad begegnet. Wichtige Details, wie zum Beispiel die Beschriftung einer Platine, vergisst er nicht.

Der Rest des Buchs ist eine Sammlung von Rezepten, die selten genutzte KiCad-Funktionen erläutern. Unter anderem lernt man hier, wie man neue Bauteile erstellt und Platinen mit außergewöhnlichen Formen entwirft. Insgesamt ist Dalmaris ein gutes Handbuch gelungen, das allerdings ein besseres Lektorat verdient hätte.

—Maik Schmidt/pek

Autor	Peter Dalmaris
Verlag	Elektor
Umfang	503 Seiten
ISBN	978-3-8957-6341-0
Preis	39,80 € (Buch), 29,52 € (PDF)

Holz & Dampf

Techniken und Projekte des Holzbiegens

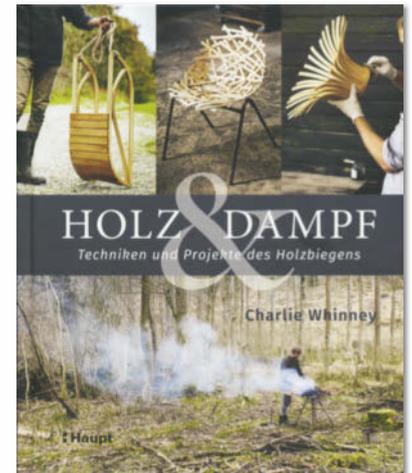
In heißem Dampf erhitztes Holz lässt sich biegen und hält nach Abkühlung diese Form – das Prinzip kennt man etwa von den berühmten Bugholzstühlen. Weniger bekannt dürfte sein, dass sich die Technik auch mit einfachsten Mitteln selbst nutzen lässt. Wie das geht, zeigt der britische Autor Charlie Whinney Schritt für Schritt und mit vielen Fotos anhand von 14 Projekten, die beim Kleiderbügel aus einem Ast beginnen und im Bau eines Schlittens, eines Schaukelstuhls und eines Boots gipfeln. Zusätzlich beschreibt er, wie man sich Hilfsmittel wie Biegeschablonen oder eine

Dampfkammer aus einem Stück Abflussrohr und einem Tapetenablösegerät baut.

Praxisnah sind die Projekte nach der Arbeitsumgebung eingeteilt: Die einfachsten Anleitungen setzen nicht viel mehr als ein Lagerfeuer voraus und in der Küche entstehen in einer Schmorpfanne als Dampferzeuger vor allem dekorative Kleinigkeiten. Bei den anschließenden Projekten für die „Mini-Werkstatt“ kommen Holzstücke von maximal knapp 30cm Länge zum Einsatz, die man in der Mikrowelle oder einer selbst gebauten Dampfschale vorbereitet, bevor man sie zu Lampenschirmen oder einer Sessel-Sitzschale verarbeitet. Die eingangs erwähnten großen Projekte erfordern mehr

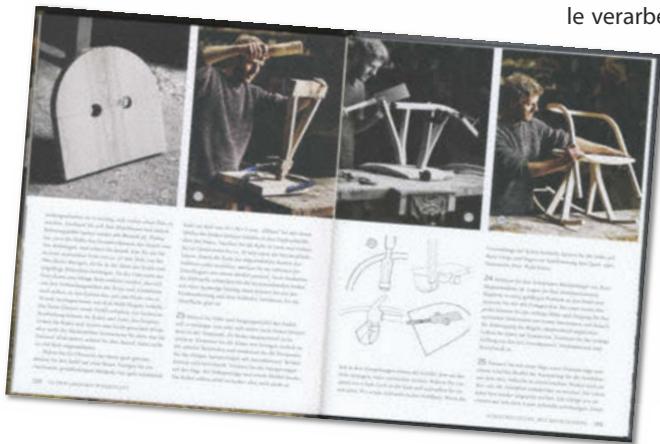
Platz und aufwendigere Schablonen – Vorlagen hierfür gibt es zum Download auf der Webseite des Verlags.

Während die einfachen Projekte wenig Werkzeug und Vorkenntnisse voraussetzen und auch mit Kindern umsetzbar sind, braucht man für die ambitionierteren Bauvorschläge etwas Routine in der Holzbearbeitung und



idealerweise auch Werkzeug wie Hobel, Axt, Kreis- oder Bandsäge. Wer gerne mit Holz arbeitet, findet in diesem Buch viel Inspiration und erweitert sein Repertoire vielleicht um eine bisher eher exotische Technik. —pek

Autor	Charlie Whinney
Verlag	Haupt Verlag
Umfang	144 Seiten
ISBN	978-3-258-60208-0
Preis	28 €



Hello Ruby

Wenn Roboter zur Schule gehen

Im vierten Buch ihrer „Hello Ruby“-Reihe unternimmt Linda Liukas den Versuch, die Entwicklung von künstlicher Intelligenz kindgerecht zu vermitteln. Wie bei ihren vorherigen Büchern beginnt die Autorin mit einer schön illustrierten Geschichte, die die Kernpunkte des technischen Themas berührt. Im zweiten Teil des Buchs stellt sie die grundlegenden Konzepte von maschinellem Lernen in Form von erklärenden Texten und Übungsaufgaben dar.

Mit ihrer Herangehensweise, Kindern technisches Wissen durch Geschichten, wiedererkennbare Figuren und Übungsaufgaben zu vermitteln, hat Linda Liukas sicherlich den richtigen Weg gewählt. Zwischen den einzelnen Bänden dieser Reihe gibt es allerdings klare Qualitätsunterschiede. Das Thema des vorliegenden Buchs ist komplex und die Autorin schafft es nicht so gut wie in den beiden vorangegangenen Büchern (*Die Reise ins Innere des Computers* und *Expedition ins Internet*) das Thema zu vereinfachen.

In vielen Erklärungen bleiben Fachbegriffe unerklärt stehen, die Übungsaufgaben bleiben oberflächlich oder sind zu ungenau. Das liegt daran, dass die Autorin viel zu viele Begriffe einführt. Konzepte wie BLOBs, der Turing-Test oder bestärkendes Lernen für KIs gehen weit schon über das Vorstellungsvermögen vieler Erwachsener hinaus und können auch nicht so weit heruntergebrochen werden, dass sie Grundschulkindern gut vermittelt werden kön-

nen. Dennoch wird auch für dieses Buch die Altersempfehlung ab fünf Jahren ausgesprochen. Die grundsätzlichen Unterschiede zwischen menschlichem und maschinellem Lernen und ihre Auswirkungen auf die Entwicklung von Maschinen und unseren Alltag hätten als Thema für dieses Buch ausgereicht. Und diese Unterschiede erklärt Liukas auch gut. Leider geht diese Qualität des Buchs durch die Überfrachtung verloren. —esk

Autorin	Linda Liukas
Verlag	Bananenblau
Umfang	96 Seiten
ISBN	978-3-946829-34-8
Preis	16,80 €



Impressum

Make: Nächste Ausgabe erscheint am 20. Februar 2020

Redaktion

Make: Magazin
 Postfach 61 04 07, 30604 Hannover
 Karl-Wiechert-Allee 10, 30625 Hannover
 Telefon: 05 11/53 52-300
 Telefax: 05 11/53 52-417
 Internet: www.make-magazin.de

Leserbriefe und Fragen zum Heft: info@make-magazin.de

Die E-Mail-Adressen der Redakteure haben die Form xx@make-magazin.de oder xxx@make-magazin.de. Setzen Sie statt „xx“ oder „xxx“ bitte das Redakteurs-Kürzel ein. Die Kürzel finden Sie am Ende der Artikel und hier im Impressum.

Chefredakteur: Daniel Bachfeld (dab) (verantwortlich für den Textteil)

Stellv. Chefredakteur: Peter König (pek)

Redaktion: Heinz Behling (hgb), Helga Hansen (hch), Carsten Meyer (cm), Rebecca Husemann (rehu), Elke Schick (esk)

Mitarbeiter dieser Ausgabe: Nicole Bätz, Beetlebum, Anne Bias, Marcel Dierig, Michael Gaus, Selina Häuser, Dirk Herrendoerfer, Tobias Hübner, Lisa Ihde, Simon Ilgenfritz, Eva Ismer, Armin Joachimsmeier, Thomas Klaube, Miguel Köhnlein, Guido Körber, Caleb Kraft, Tobias Lohf, Martin Neuhaus, Roman Radtke, Carsten Romahn, Maik Schmidt, Paul Srna, Carsten Wartmann, Wolfgang Ziegler

Assistenz: Susanne Cölle (suc), Christopher Tränkmann (cht), Martin Triadan (mat)

DTP-Produktion: Nicole Judith Hoehne (Ltg.), Martina Bruns, Martina Fredrich, Jürgen Gonnermann, Birgit Graff, Angela Hilberg, Wolfgang Otto (Korrektorat), Astrid Seifert, Dieter Wahner

Art Direction: Martina Bruns (Junior Art Director)

Layout-Konzept: Martina Bruns

Layout: Nicole Wesche

Fotografie und Titelbild: Andreas Wodrich, Melissa Ramson

Verlag

Maker Media GmbH
 Postfach 61 04 07, 30604 Hannover
 Karl-Wiechert-Allee 10, 30625 Hannover
 Telefon: 05 11/53 52-0
 Telefax: 05 11/53 52-129
 Internet: www.make-magazin.de

Herausgeber: Christian Heise, Ansgar Heise

Geschäftsführer: Ansgar Heise, Dr. Alfons Schröder

Verlagsleiter: Dr. Alfons Schröder

Stellv. Verlagsleiter: Daniel Bachfeld

Anzeigenleitung: Michael Hanke (-167) (verantwortlich für den Anzeigenteil), www.heise.de/mediadaten/make

Leiter Vertrieb und Marketing: André Lux (-299)

Service Sonderdrucke: Julia Conrades (-156)

Druck: Firmengruppe APPL echter druck GmbH, Delpstraße 15, 97084 Würzburg

Vertrieb Einzelverkauf:

VU Verlagunion KG
 Meißberg 1
 20086 Hamburg
 Tel.: 040/3019 1800, Fax.: 040/3019 145 1800
 E-Mail: info@verlagsunion.de
 Internet: www.verlagsunion.de

Einzelpreis: 10,90 €; Österreich 11,90 €; Schweiz 18,00 CHF; Benelux, Italien, Spanien 11,90 €

Abonnement-Preise: Das Jahresabo (7 Ausgaben) kostet inkl. Versandkosten: Inland 65,10 €; Österreich 66,50 €; Schweiz/Europa: 72,10 €; restl. Ausland 88,20 €

Das Make-Plus-Abonnement (inkl. Zugriff auf die App, Heise Select 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 2019 and published under license from Make Community LLC. All rights reserved.

Printed in Germany. Alle Rechte vorbehalten. Gedruckt auf Recyclingpapier.

© Copyright 2019 by Maker Media GmbH

ISSN 2364-2548

Nachgefragt

Auf Seite 42 zeigen wir, wie man einen Sensor baut, der daran erinnert, nach dem Lüften das Fenster rechtzeitig zu schließen.

Was für ein Helferchen würdest Du Dir gerne mal selber bauen?

Lisa Ihde

Potsdam, hat die Lovebox auf Seite 16 gebaut.

Gerne würde ich Alexa sagen: „Bitte räume den Geschirrspüler aus!“ Damit dann ein kleiner Roboter in die Küche fährt und die Maschine leert. Aber die Gefahr ist groß, dass bei der Umsetzung von solch einem Projekt einiges an Geschirr zerbricht.

Eva Ismer

Berlin, zeigt auf Seite 50, wie Lichtleiter in Textilien kommen.

Ich hätte gerne ein smartes Kleidungsstück, das sich bei Bewegung auflädt, den Strom speichert und dann etwa mein Handy aufladen kann. Alternativ fände ich auch einen Wecker mit integrierter Glücksinjektion super.

Anne Bias, Nicole Bätz und Selina Häuser

Berlin, zeigen gemeinsam ab Seite 56, wie man mit Biosignalen aus angespannten Armmuskeln ein Spiel steuern kann.

Um einschläfernde Papierberge bei wissenschaftlicher Recherche zu umgehen, soll unser Helferchen jeden komplexen Zusammenhang im Schnelldurchlauf oder Schritt für Schritt visuell darstellen können. Die Animationen würden den aktuellen Stand der Forschung jederzeit als virtuelles Erlebnis ins eigene Wohnzimmer holen. So wird Lernen schnell zum Hobby!



Inserentenverzeichnis

ALLNET Computersysteme GmbH, Germering	33	dpunkt.verlag GmbH, Heidelberg	77
BERNHARD Kunststoffverarbeitings GmbH 3dk.berlin, Berlin.....	19	Reichelt Elektronik GmbH & Co., Sande	7
Blöss GmbH, Berlin	55	segor electronics, Berlin	23
Conrad Electronic SE, Hirschau.....	21	Make:markt	27

NEU
+portofrei

Jetzt durchstarten mit dem ESP32!

Als Heft oder Digital: shop.heise.de/esp-special



INKLUSIVE:



Original ESP32-Modul

➤ Generell portofreie Lieferung für Heise Medien- oder Maker Media Zeitschriften-Abonnenten oder ab einem Einkaufswert von 15 €. Nur solange der Vorrat reicht. Preisänderungen vorbehalten.

© Copyright by Maker Media GmbH.

 **heise shop**

shop.heise.de/esp-special



Im nächsten Heft

Die Make-Ausgabe 1/20 erscheint am 20. Februar 2020



Bild: Pixel B/Shutterstock.com

Umweltsensorkits

Ob Feinstaubbelastung oder Geräuschpegel – Veränderungen der Luft- und Umweltqualität treiben derzeit viele um. Wer die Bedingungen auf seinem Balkon oder im Garten überwachen möchte, muss inzwischen nicht mehr bei Null anfangen. Diverse Bausätze vereinfachen den Bau der eigenen Messstation. Wir nehmen sie unter die Lupe.

Wir behalten uns Änderungen des Inhalts der kommenden Make-Ausgabe vor. Was auf dieser Seite steht, zeigt den aktuellen Planungsstand, mehr als zwei Monate vor Druckschluss der kommenden Ausgabe. Wenn uns bis dahin ein Thema spannender und dringender erscheint, werfen wir unsere Pläne schon mal um – auch wenn wir nicht vorhaben, das häufiger zu tun.

Spracherkennung ohne Cloud

Sprachassistenten wie Siri, Alexa & Co. hören aufs Wort, leiten aber auch alles in eine unkontrollierbare Cloud weiter, wo das Gesagte analysiert und – wer weiß – vielleicht auch weitergegeben wird. Wir sagen Ihnen, wie Sie Spracherkennung oder -steuerung nutzen können, ohne mit dem Internet verbunden zu sein.



Bild: BrA82/Shutterstock.com

E-Bike im Eigenbau

Ein Vater möchte seinen Sohn fürs Basteln begeistern und verspricht ihm, dass sie gemeinsam sein Wunschprojekt umsetzen. Heraus kommt ein durchaus praxistaugliches E-Bike mit Akkuschauberantrieb und Arduino-Steuerung.



Gesucht: Reinventing the Past

Wir suchen Projekte, die die Fähigkeiten alter 8-Bit-Systeme und anderer Vintage-Hardware aus der frühen Home-Computer-Ära mit Boards von heute nachbauen – keine Emulatoren auf dem Raspberry Pi, sondern echte Mini-computer mit Bildschirm und Tastatur aus Mikrocontrollern.



Bild: creatifolio/Shutterstock.com



I WANT
YOU

Berlin

Maker Faire®

Vernetzen • Austauschen • Präsentieren

18. - 19. April 2020

Arena Berlin/Alt-Treptow

Melde dich **bis zum 9. Februar**
als **Maker** für eine kostenfreie
Standfläche an.

maker-faire.de/berlin

© Copyright by Maker Media GmbH.

Make:

IHR MAKERLEIN KOMMET...



Das perfekte Maker-Geschenk!

Ihre Vorteile:

- ✓ 7x im Jahr Maker-Ideen verschenken
- ✓ Inklusive Geschenk-Gutschein
- ✓ Conrad-Gutschein für Sie oder den Beschenkten
- ✓ Versandkostenfrei

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

© Copyright by Maker Media GmbH.