

Rabattaktion: 40-teiliges Sensoren-Starterset für 54,90 € (statt 64,90 €)



SPASS MIT TECHNIK

Einfache Computerprojekte zum Selbermachen

Lego-Roboter bauen & steuern

Maschinelles Lernen mit Lego Mindstorms
Spielzeugroboter mit dem Raspberry Pi

In Minecraft coden

3D-Klötzchenwelt als Lernhilfe

Songs arrangieren

Mit Google Song Maker experimentieren

Mit Licht malen

Tolle Fotos mit der Taschenlampe

Text-Adventure programmieren

Einsteigerfreundliches Abenteuerspiel gestalten

Fidget-Spinner selber machen

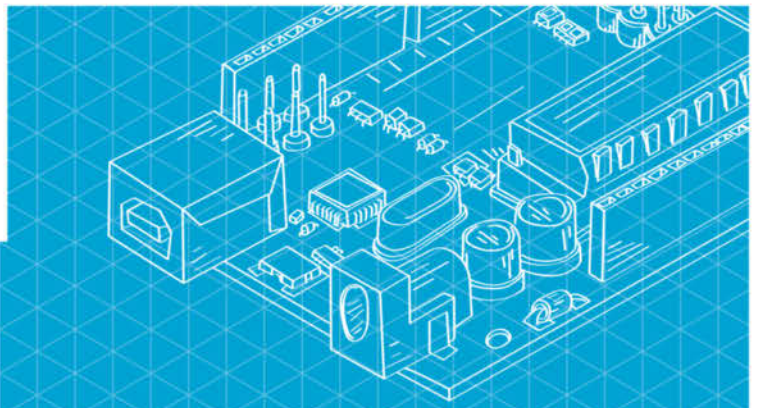
Konstruieren und mit dem 3D-Drucker ausdrucken

Cooler Kreativ-Projekte

Trickfilme animieren • Comics gestalten • Podcasts produzieren



Make:



DAS KANNST DU AUCH!



GRATIS!



2x Make testen und 6 € sparen!

Ihre Vorteile:

- ✓ Neu: Jetzt auch im Browser lesen!
- ✓ Zugriff auf Online-Artikel-Archiv*
- ✓ Zusätzlich digital über iOS oder Android lesen
- ✓ Versandkostenfrei

Für nur 15,60 Euro statt 21,80 Euro.

* Für die Laufzeit des Angebotes.

Jetzt bestellen: make-magazin.de/miniabo



Bild: Albert Hulm

Liebe Leserin, lieber Leser,

mehr als einen Computer und ein paar Zeilen Code braucht es manchmal nicht, um echte Nerds zum Lächeln zu bringen. Wir haben in diesem Heft 25 Anleitungen gebündelt, die Neugier und Spaß an (Computer-)Technik wecken. Sie zeigen unter anderem, wie Sie auf dem iPad Comics gestalten (Seite 66), mit dem Smartphone Stop-Motion-Filme drehen (Seite 56), mit einer Taschenlampe Lichtbilder zaubern (Seite 70) und im Browser Podcasts und Musik produzieren (Seite 92).

Viele der vorgestellten Projekte kosten nichts und lassen sich in wenigen Stunden realisieren – optimal, um auch mal gemeinsam mit dem technikaffinen Nachwuchs Regentage zu überbrücken. Anleitungen, die sich komplett ohne oder mit wenig Vorkenntnissen realisieren lassen, haben wir im Inhaltsverzeichnis mit einem Teddy gekennzeichnet; so musizieren Sie etwa mit Googles Songmaker ohne einen einzigen schiefen Ton und ohne jemals ein Instrument in der Hand gehalten zu haben (Seite 102). Und einen PC-Lüfter-angetriebenen Katamaran kann jeder in Eigenregie nachbauen, der einen ausrangierten Computer, eine Säge und etwas Styropor zur Hand hat (Seite 28).

Anleitungen, deren Umsetzung etwas Vorerfahrung voraussetzen, erkennen Sie im Inhaltsverzeichnis am Computermännchen. Darunter fallen auch einige unserer Programmier-Artikel – denn der Einstieg in die Welt von Python, Processing & Co. ist deutlich einfacher, als viele vermuten: Rund 40 Zeilen Code reichen bereits, um etwa unser Zeichenprogramm "Mal-o-Mat" zu coden (Seite 106) oder mit Lua ein Labyrinthspiel im angesagten Pixel'n'Sprites-Design zu erschaffen (Seite 148). Und wenn in Ihren Adern Maker-Blut fließt, greifen Sie zu Lego, Einplatinencomputern und den mit einem Doktorhut gekennzeichneten Artikeln. Damit konstruieren Sie Roboter oder tauchen in die Welt des maschinellen Lernens ein.



leicht



mittel



schwer

Viel Spaß beim Lesen und vor allem beim Ausprobieren wünscht Ihnen





Martin Reche

Inhalt







Roboter bauen und steuern

Lego und fischertechnik haben diverse Roboterbausätze im Angebot. Wir zeigen, was diese können und wie Sie mit den richtigen Hacks noch mehr aus den Sets herausholen.

-  6 Roboterbausatz Boost im Test
-  10 Lego Move-Hub am Raspberry Pi betreiben
-  14 Spielzeug-Roboter per Raspberry Pi steuern
-  20 Neuronale Netze mit Lego Mindstorms






3D-Druck und andere Maker-Ideen

Mit leicht verfügbaren oder selbst hergestellten Bauteilen und Einplatinencomputern lassen sich tolle Projekte realisieren. Wir haben eine Auswahl für Sie zusammengestellt.

-  28 Katamaran mit PC-Lüfter-Antrieb
-  30 Elektronik aus leitfähiger Knetmasse
-  36 Fidget Spinner aus dem 3D-Drucker
-  42 Arduino aus Minecraft ansteuern

Kunst aus dem Computer

Man muss kein Photoshop-Profi sein, um dem Rechner visuelle Leckerbissen zu entlocken. Mit unseren Workshops können Sie schnell großartige Ergebnisse erzielen.

-  46 Fotobearbeitung mit dem kostenlosen Paint.Net
-  56 Trickfilme mit dem Smartphone produzieren
-  62 Mit Wordseye aus Texten Bilder erzeugen
-  66 Mit dem iPad eigene Comics gestalten
-  70 Lichtmalerei mit Taschenlampen

Kreative Audio-Projekte

Schon mit einfachen Mitteln lassen sich Audio-Projekte mit dem Computer arrangieren. Lesen Sie, wie Sie Klangbasteleien vom Podcast bis zum fertigen Song am PC umsetzen können.









-  80 Musikstücke im Browser arrangieren
-  86 Erste Schritte mit der Musik-Software Sonic Pi
-  92 Podcasts selber produzieren
-  98 Audioschnipsel aufnehmen und klanglich verfremden
-  102 Melodien mit Googles Song Maker arrangieren





Programmieren leicht gemacht

Programmieren kann jeder! Wie das geht, zeigen wir Ihnen mit unseren vielfältigen Einstiegen in die fantastische Welt des Codens – von Processing über Lua bis Logo.

-  106 Ein Malprogramm mit Processing coden
-  112 Upgrade für das Projekt Mal-o-Mat
-  118 Spaßige Bash-Experimente mit Raspbian
-  124 Aus Programm-Code Bilder zaubern
-  130 Schicke Grafiken im Browser zeichnen
-  136 Ein Text-Adventure in Python programmieren
-  142 Mit Mods in Minecraft coden
-  148 Ein Labyrinthspiel mit Pico-8 programmieren

Zum Heft

- 3 Editorial
- 154 Aktion
- 154 Impressum

Aktion: Sensorkit zum Sonderpreis



Wir haben zusammen mit dem heise shop ein besonderes Angebot geschnürt. Auf Seite 154 lesen Sie, wie Sie das Sensorkit X-40 von joy-it für Raspi, Arduino & Co. zum günstigen Vorteilspreis bekommen.

Sven Hansen

Roboterbausatz Boost im Test



Bisher waren Legos Roboterbausätze teuer und lockten mit ihrem Ingenieurs-Charme bestenfalls Erstsemester an. Boost soll nun selbst die Jüngsten an das Programmieren heranführen.

Lust auf Lego? Das Set 17101 – Code-Name „Boost“ – hat einiges zu bieten. Herzstück ist der Move-Hub – ein per Bluetooth LE ansprechbarer Controller, an den sich bis zu zwei zusätzliche Aktoren/Sensoren anschließen lassen; er wird mit sechs AAA-Batterien bestückt. Zwei Motoren, ein Gyrosensor und eine RGB-LED sind im Move-Hub bereits integriert. Ein weiterer Motor und ein kombinierter Farb-/Entfernungssensor sind im Set enthalten. Der Move-Hub bildet die Basis für fünf spannende Grundmodelle vom klassischen Roboter über eine Technokatze

bis zur Legofabrik. Die Modelle lassen sich per App in Bewegung setzen und programmieren.

In der Packung liegen 843 bunte Legosteine. Schon die Farbzusammenstellung zeigt, dass Lego nicht in die Geschlechterfalle tappen möchte: Leuchtendes Hellblau und Orange sollen Boost für Jungen wie Mädchen gleichermaßen attraktiv machen.

Eines fehlt: die sonst obligatorische gedruckte Bauanleitung. Der Boost-Bausatz lässt sich ausschließlich in Kombination mit einem Smartphone oder Tablet nutzen. Alle Anleitungen sind in der für Android und iOS erhältlichen App enthalten. Auf kleineren Smartphones sind sie zwar schlecht zu lesen, aber die Kids wird es nicht stören. Ein Tablet bietet mehr Komfort beim Lesen.

Das Handy oder Tablet mit Boost-App zeigt die Bauanleitung an und stellt die nötige Rechenleistung bereit. Über Bluetooth steuert es Aktoren und liest die Informationen aus den Sensoren aus. Zugleich verleiht es den Modellen die nötige Stimme und sorgt für Interaktivität: Das Mobilgerät gibt passende Sounds aus und leiht den Modellen das interne Mikrofon als Ohr. Alle Boost-Modelle erwachen nur in Kombination mit der App zum Leben – trennt man die Verbindung, bleiben sie stehen.

Lego Boost ist ein enger Verwandter des WeDo-2.0-Kits, das von Lego ursprünglich im Rahmen seiner Education-Reihe für den Einsatz an Schulen konzipiert wurde. Schnell hatte dieser Bildungsbausatz auch bei der außerschulischen Kundschaft seine Fangemeinde gefunden – mit Boost will Lego das Konzept nun in den Massenmarkt bringen. Während sich die WeDo-App noch mit einer recht spröden Lernerfläche präsentierte, hat Lego für Boost einen starken Partner mit Erfahrung in der Programmierung von Spiele-Apps ins

Boot geholt. Das Ergebnis aus der Entwicklerschmiede Funday Factory kann sich sehen lassen.

Die liebevoll gestaltete Boost-App führt Kinder und Erwachsene behutsam an die Möglichkeiten des Sets heran. Den Kern bilden die fünf Basismodelle, die man nacheinander erstellen kann: den klassischen Roboter Vernie, das Kettenfahrzeug M.T.R. 4, die Mini-Fabrik Auto Builder, die Katze Frankie und die coole Lego-Klumpfe Guitar4000. Die App führt den Nutzer zunächst sehr kleinschrittig durch die Bauanleitung jedes Modells und dessen individuelle Fähigkeiten. So wird der kombinierte Farb-/Abstandssensor bei der Gitarre etwa verwendet, um die Distanz des Schlittens vom Korpus und damit den gewünschten Akkord zu bestimmen. Bei der Katze Frankie steckt er im Maul und lässt sie verschiedene Objekte anhand der Farbe erkennen oder auf einer Mundharmonika aus farbigen Legosteinen spielen.

Auch die fünf Boost-Modelle sind mit viel Liebe zum Detail gestaltet – für den Aufbau benötigt man je rund zwei Stunden. Für den besonderen Kick sorgen die in der App hinterlegten Sounds: Vom Happy-Birthday-Song der Katze über die Western-Sprüche des Cowboy-Roboters bis hin zur Furz-Gitarre ist alles dabei. Der Speicherhunger der App von rund 300 MByte ist so erklärbar.

Jedes Modell lässt sich Level für Level freispielen. Bevor man Roboter Vernie beispielsweise das Schießen beibringt, lernt man seinen Kettenantrieb kennen. In einem anderen Level geht es darum, die Sprachausgabe des Roboters zu kontrollieren. Am Ende steht jeweils ein Programmiermodus, in dem sich das Modell durch das Verschieben von einfachen Symbolen per Drag & Drop animieren lässt. Die Programmiermöglichkeiten sind klar auf

die Fähigkeiten des jeweiligen Modells beschränkt, was jüngeren Mitspielern ebenfalls schnelle Erfolge garantiert. Die objektorientierte Programmiersprache umfasst auch einfache Schleifenelemente und Zufallsvariablen.

Besonders witzig ist der „Auto Builder“. Die hochautomatisierte Fabrik spuckt auf Knopfdruck kleine Legoroboter aus, die ein Greifarm von einem vorbestückten Brett mit Bauteilen pflückt und stapelt. Die Fertigungstoleranzen sind minimal und der Produktionsprozess gerät durch gelockerte Legosteine leicht auf Abwege. Aber selbst die Unfälle sind unterhaltsam, und die Faszination ist nur umso größer, wenn dann doch mal ein perfekter Mini-Robo vom Band fällt.

Die zwei fahrbaren Modelle Vernie und M.T.R. 4 kann man über die App

fernsteuern. Passend zum Raupenfahrzeug liegt ein bedruckter Parcours in der Packung, auf dem der M.T.R. 4 allherhand Aufgaben erfüllen muss.

Hat man die fünf Boosts durchgespielt, bleiben noch drei Grundmodelle, die man für die Animation älterer Lego-Sets oder fürs Freispiel verwenden kann. Eine undekorierte Seifenkiste dient als Basis für eigene Fahrzeug-Kreationen. Aus dem watschelnden Vierfüßler lassen sich im Handumdrehen Fantasietiere zaubern. Ein motorbetriebenes Tor wird je nach Fantasie zum Eingang einer Raumfahrtzentrale oder führt auf Zuruf in Ali Babas Räuberhöhle.

Ab und zu wird das Spiel jäh unterbrochen, denn Boost frisst Batterien. Waren die Aktoren in Bewegung, zeigte unser Messgerät im Labor rund 1,7 Ampere an. Mit handelsüblichen Batterien



Schrittweise führt die Boost-App die Kinder durch den Aufbau der einzelnen Modelle. Die objektorientierte Programmier-Umgebung bildet den Abschluss.



Das Kettenfahrzeug M.T.R. 4 muss auf dem mitgelieferten Spielplan allerlei Aufgaben erfüllen. Wie der Roboter Vernie lässt es sich über die App frei bewegen.

kamen wir auf eine Spieldauer von knapp 2 Stunden. Kappt man die Bluetooth-Verbindung, schaltet der Hub nach zwei Minuten automatisch ab.

Generell ist Boost beim Anzeigen des Batterielevels recht zappelig: Wenn zwei Motoren gleichzeitig anlaufen, sackt die Spannung kurzzeitig ab und die App fordert oft zu früh zum Batteriewechsel auf. Will man das Set viel bespielen, ist also der Kauf von 12 guten Akkus und passendem Ladegerät angeraten. Auch das Netzteil fürs Tablet sollte bei längeren Bau-Sessions nicht zu weit entfernt sein.

Das Freispielen aller Modelle beschäftigt große wie kleine Bastler locker

15 Stunden. Danach ist das System in Sachen Programmierung allerdings am Ende. Im Unterschied zum WeDo 2.0 gibt es keinen wirklich freien Programmiermodus. Jedes Modell hat je nach seinen Fähigkeiten ein festes Set aus Objekten. Ein direktes Auslesen von Sensoren oder das freie Ansteuern der Aktoren ist nicht vorgesehen. Genau diese Limitierung hebt unser ab Seite 10 beschriebener Boost-Hack auf. Er zeigt, wie man den Hub über die Bluetooth-Verbindung eines Raspberry Pi direkt ausliest und steuert.

Fazit

Lego Boost kombiniert die Freude am Steinchenstecken mit spielerischen Programmieransätzen und liefert stundenlange Unterhaltung für die ganze Familie. Als Ersatz des klassischen Batterieblocks wünscht man sich ein optionales Powerpack mit Lilon-Akku, denn

das ständige Nachlegen von Batterien nervt.

Der Hersteller empfiehlt Boost für Kinder von 7 bis 12 Jahren. Das ist recht eng gefasst. Tatsächlich haben auch jüngere Kinder – zur Not mit Hilfe der Eltern – viel Spaß mit den bewegten Legobausätzen. Durch die inzwischen hinzugekommenen App-Versionen fürs Smartphone und für ältere Android-Versionen hat sich zudem das Feld möglicher Bediengeräte erweitert. Die Kleinen müssen nun nicht mehr unbedingt das Familientablet blockieren, ein altes Handy genügt.

Die obere Altersgrenze lässt sich mit dem Hacking-Artikel locker nach oben schieben. Sollte dem Teenager das Herumschubsen von Logikblöcken auf dem Touchscreen nur noch ein Gähnen entlocken, lässt sich das Set mit einem Raspi gehörig aufbohren. Vom Mauschubser zum Hacker ist es so nur ein Katzensprung. (sha) **ct**

LEGO BOOST

Robotik-Bausatz

Hersteller	Lego, lego.com
Kommunikation	Bluetooth LE
Stromversorgung	6 x AAA-Batterien
Preis	120 €

04. Juli 2018 – Areal Böhler – Düsseldorf

IT-SECURITY JOBMESSE

im Rahmen der

Cyber Security Challenge Germany 2018



Nach wie vor sucht die IT-Sicherheitsbranche händeringend nach Nachwuchs- und Fachkräften. Aus diesem Grund werden wir auch in diesem Jahr wieder eine Jobmesse im Rahmen der CSCG veranstalten. Wir heißen alle jungen Sicherheitstalente, Studenten aus dem Fachbereich IT-Sicherheit sowie erfahrene Security-Experten herzlich willkommen.

- Treffen Sie zahlreiche **attraktive Unternehmen** und nutzen Sie die Gelegenheit, um Ihren Wunscharbeitgeber vor Ort persönlich kennenzulernen.
- Über kurze Vorträge in unserem **Recruiting-Forum** können Sie einen ersten Eindruck der Unternehmen und Einblicke in deren Arbeit gewinnen!
- Testen Sie Ihre Fähigkeiten und nehmen Sie am **CTF-Hacking-Wettbewerb** teil! Tauschen Sie sich mit Profis aus und gewinnen Sie tolle Preise.
- Außerdem bieten wir vor Ort ein kostenfreies **Bewerbungsfoto-Shooting** an und stellen Ihnen die Fotos anschließend in digitaler Form kostenfrei zur Verfügung.

Kostenlose Anmeldung unter www.heise-events.de/cscg

Aussteller:



Veranstalter:



Partner:



Kooperationspartner:



Dr. Till Harbaum

Lego Move-Hub am Raspberry Pi betreiben



Der Move-Hub aus dem Roboterbausatz von Lego lässt sich eigentlich nur per App von Smartphones und Tablets aus steuern. Wir zeigen, wie man Boost in Kombination mit einem Raspberry Pi zu einer offenen und universellen Plattform erweitert.

Mit dem Boost System führt Lego endlich einen vergleichsweise günstigen Roboterbausatz im Portfolio. Im Unterschied zu den teureren Robotik-Sets ist er durch die feste Anbindung an die Boost-App allerdings weniger flexibel einsetzbar.

Ab Seite 14 zeigen wir, wie sich die Spielzeug-Controller von Fischertechnik und Lego WeDo 2.0 über Bluetooth mit dem Raspberry Pi steuern lassen. Da liegt es ziemlich nahe, dieses Projekt als Basis zur Öffnung des Boost-Systems zu nutzen.

WeDo 2.0 nutzt dieselben Steckverbindungen zum Anschließen der Sensoren und Aktoren wie der Boost Move-Hub und kommuniziert wie dieser per Bluetooth LE. Doch die Unterschiede unter der Haube fallen überraschend groß aus. Die mechanische Ähnlichkeit des „Move-Hub“ getauften Boost-Controllers mit dem WeDo 2.0 weckt dagegen die Hoffnung, dass auch auf Bluetooth-Seite große Ähnlichkeit herrscht.

Der offensichtlichste Unterschied zwischen beiden Systemen besteht darin, dass Lego nun eine herstellereigenspezifische Bluetooth-Adresse nutzt, statt

wie bei WeDo 2.0 nur die Kennung des Chipherstellers (Texas Instruments) einzusetzen. Der Boost-Hub hingegen meldet sich mit einer Adresse nach dem Muster „00:16:53:XX:XX:XX“ und gibt sich so klar als Lego-Device zu erkennen. Die ersten drei Bytes der insgesamt sechs Bytes langen Bluetooth-Adresse bilden die Firmenkennung (Organizationally Unique Identifier OUI). Sie wird von der IEEE an Firmen vergeben, die dann die übrigen drei Bytes frei verwenden können. Dienste wie „https://macvendors.com“ erleichtern die Suche nach Herstellern.

Rigorous hat Lego die Interna der Bluetooth-Kommunikation verbogen. Der Move-Hub nutzt keine der Standard-Bluetooth-Charakteristiken. Mit üblichen BLE-Diagnose-Tools lassen sich ihm daher kaum Information entlocken. Standard-Tools wie Nordics nRF-Connect bringen nicht einmal den Batteriestand in Erfahrung. Stattdessen implementiert der Hub eine Lego-spezifische Charakteristik für den gesamten Datenaustausch. Lego verwendet hier eigene UUIDs und entleiht sie nicht – wie bei WeDo 2.0 – bei Nordic Semiconductor. Damit besteht zumindest keine Verwechslungsgefahr mehr mit anderen Geräten mit Nordics Code.

Leider liefert Lego zum Boost anders als beim WeDo 2.0 keinerlei Dokumentation zum Kommunikationsprotokoll. Da der Datenaustausch zwischen dem Controller und der Lego-App unverschlüsselt abläuft, kann man ihn mit Geräten wie dem Ubertooth-One jedoch problemlos belauschen. Projekte wie „BOOSTreveng“ sammeln auf diesem Weg gewonnene Informationen und machen sie für eigene Programmierungen nutzbar. Eine ausführliche Linksammlung finden Sie unter ct.de/w5nx.

Die Bedeutung der verschickten Datenpakete muss man sich nun aus dem Datenfluss selbst zusammenreimen. Manches ist relativ einfach zu interpretieren, beispielsweise das Längenfeld, das jedem Kommando voransteht. Viele andere Werte im Datenstrom haben jedoch keine direkt erkennbare Bedeutung. Erst durch Manipulation der Werte und anschließendes Beobachten der Reaktion des Move-Hub lassen sich die wesentlichen Werte erraten.



KEINE GARANTIE

Es ist nicht komplett auszuschließen, dass der Move-Hub des Boost-Systems auf unpassende Kommandos problematisch reagiert.

Steuert man den an einem Port angeschlossenen Sensor beispielsweise mit Motorbefehlen an, könnten Spannungen auf den Sensor geschaltet werden, die man ihm nicht zumuten darf.

Es lässt sich auch nicht ausschließen, dass man im Move-Hub hinterlegte Mechanismen zum Firmware-Update aktiviert, die den Controller dauerhaft beschädigen oder unbrauchbar machen.

Während unserer ausführlichen Experimente kam es allerdings nie zu Problemen. Daher halten wir die Wahrscheinlichkeit eines Totalausfalls für gering.

Voraussetzungen

Da der Move-HUB per Bluetooth-Low-Energy (BLE) kommuniziert, muss auch der eingesetzte Raspberry diese Bluetooth-Variante beherrschen. Für den eingebauten Bluetooth-Chip des Pi 3 und 3B+ ist das der Fall, bei älteren Raspis mit externen USB-Dongles klärt das Skript ble_test.py, ob die Voraussetzungen gegeben sind. Ebenfalls unter ct.de/w5nx finden Sie eine Sammlung von Skripten und Programmen, die Sie für unser Boost-Projekt benötigen.

Die eigentlichen Bluetooth-Versuche haben wir in Python realisiert. Sie verwenden die PyGatt-Bibliothek, die eine relativ neue Version des Linux-Bluetooth-Stacks BlueZ voraussetzt. Die nötigen Installationen auf dem Raspberry Pi kann man unserem Script python-gatt-install.sh überlassen.

Grundsätzlich hat sich an den Voraussetzungen seit den WeDo- und Fischertechnik-Experimenten nichts geändert und das damalige Setup ist für den Move-Hub unverändert tauglich. Ein Raspberry Pi 3 mit Standard-Raspbian – natürlich standesgemäß im Lego-Gehäuse verpackt – ist der Ausgangspunkt für unser Projekt. Grundkenntnisse in Python sollten vorhanden sein.

Bluetooth-Geflüster

Will der Bluetooth-Host den Controller veranlassen, Motoren und Sensoren zu aktivieren, schickt er eine Nachricht. Der Controller sendet seinerseits Nachrichten mit Ergebnissen oder Ereignissen zurück, beispielsweise wenn Sensoren

und Aktoren an- und abgesteckt werden oder ihren Status melden.

Frisch verbundene Sensoren liefern zunächst keine Messwerte. Erst muss man sie beziehungsweise die Ports, in denen sie stecken, explizit aktivieren. In unserer kleinen Programmsammlung finden Sie entsprechende Beispiele. Besonders hilfreich beim Experimentieren ist das Programm lego_boost_monitor.py. Es gibt alle Ereignisse im Klartext aus und aktiviert sofort jeden eingesteckten Sensor, sodass man die Reaktionen des Boost-Systems direkt sieht. Außer Informationen des Farb- und Distanzsensors zeigt das Programm auch Werte der Boost-internen Spannungs- und Stromüberwachung oder des Neigungssensors an.

Neigungssensor und Motor des WeDo 2.0 lassen sich auch am Move-Hub betreiben. Die Lego-eigene App unterstützt das zwar nicht, tatsächlich meldet sich der Boost-Controller aber zu Wort, sobald ein WeDo-2.0-Sensor oder -Motor angesteckt wird, und spuckt eine entsprechende Meldung aus.

Die WeDo-2.0-Sensoren lassen sich mit den gleichen Kommandos aktivieren, die man für die Boost-eigenen Sen-

BYTE 4

Hex-Code	Funktion
01	externer Anschluss „C“
02	externer Anschluss „D“
37	interner Motor „A“
38	interner Motor „B“
39	beide internen Motoren „A+B“
32	interne Farb-LED
3a	interner Neigungssensor
3b	interne Strommessung
3c	interne Spannungsmessung

soren nutzt. Wer beide Systeme sein Eigen nennt, kann sie also auch kombinieren. Es ist anzunehmen, dass man den Stecker in Zukunft in weiteren Produkten sehen wird und der Move-Hub auch mit zukünftigen Erweiterungen umgehen kann.

Unser Monitor-Script enthält alle nötigen Routinen, um auch die Sensoren und Motoren aus dem WeDo-2.0-Kasten zu nutzen und gibt entsprechende Neigungs- und Distanz-Werte aus, sobald man einen Sensor an den Move-Hub steckt.

Der WeDo 2.2 lässt sich wie der Boost-Motor ein- und ausschalten und in Richtung und Geschwindigkeit beeinflussen. Anders als dieser verfügt er jedoch nicht über eine Bewegungsrückmeldung. Versucht man dennoch, den aktuellen Drehwinkel des Motors abzufragen, erhält man die Fehlermeldung „05 00 05 41 06“.


Sie begegnet einem immer dann, wenn man Sensoren oder Aktoren mit für sie nicht vorgesehenen Codes anspricht oder den Boost-Hub mit lückenhaften Nachrichten konfrontiert.

Obwohl sich schon jetzt alle Basisfunktionen des Boost-Systems steuern lassen, verbergen sich sicher noch unentdeckte Schätze im Datenfluss. So lassen sich die Sensoren offensichtlich in unterschiedliche Betriebsarten versetzen. Der Farbsensor reagiert zum Beispiel auf entsprechende Kommandos, indem er die Farbe seiner LED verändert oder statt weniger, fester Farbwerte hoch aufgelöst RGB-Werte liefert.

Forscher gesucht

Einige der möglichen Betriebsarten haben wir bereits im Quellcode des `lego_boost_monitor.py` als Kommen-

tar dokumentiert. An vielen Stellen ist die Reaktion des Sensors beziehungsweise die genaue Bedeutung seiner Antwortnachrichten noch unbekannt. Hier besteht also noch Experimentierbedarf. Unser Praxiskasten soll helfen, sich an die Raspi-Steuerung des Move-Hubs heranzutasten.

Die Experimente mit dem Boost sind natürlich nicht auf den Raspberry Pi beschränkt, sondern lassen sich prinzipiell auf jeden Bluetooth-LE-fähigen Linux-Rechner übertragen. Das schließt Linux-Desktop-PCs ebenso ein wie den TXT-Controller von Fischertechnik oder den Lego-EV3 mit passendem Bluetooth-Adapter. Der Boost ließe sich also auch als günstige Sensor- und Aktorenerweiterung des EV3 nutzen. (sha) 

Links und Skripte zum Boost-Hack:
[ct.de/w5nx](https://github.com/w5nx)

Der Farbspiegel

In unserem Anwendungsbeispiel „Farbspiegel“ erkennt das Legomodell Farbe und Distanz eines vor ihm positionierten Objekts. Da der Boost eine interne Farb-Leuchtdiode hat, die sich frei ansteuern lässt und sein Farbsensor die Farberkennung beherrscht, ist das relativ einfach möglich.

Um den Farbsensor in Betrieb zu nehmen, muss der Raspberry ihn zunächst durch das passende Kommando aktivieren. Steckt der Sensor am Port C, geschieht das durch die Byte-Sequenz:

```
0a 00 41 01 08 01 00 00 00 01
```

Während der Farberkennung sendet der Boost kontinuierlich den erkannten Farbwert an den Pi. Eine mögliche Antwort besteht aus der Bytefolge:

```
08 00 45 01 09 00 ff 06
```

Das erste Byte 08 (Längenfeld) gibt die Gesamtlänge der Nachricht an. Die Bedeutung des zweiten Bytes ist unbekannt – es hat den Wert 00. Es könnte sich um das High-Byte des Längenfeldes handeln. Das dritte Byte (45) gibt an, dass es sich um einen Sensorwert handelt, der gemeldet wird, und das vierte Byte (01) schließlich, dass es um den Sensor in Port 1 geht.

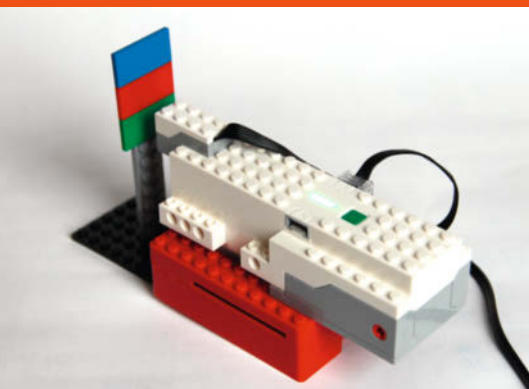
Wichtig: Dieser Port ist am Boost mit „C“ beschriftet. Steckt man den Sensor stattdessen in den mit „D“ beschrifteten Port, müsste an dieser Stelle in der Nachricht 02 stehen. Bisher bekannte Werte für Ports entnehmen Sie unserer Tabelle.

Der vom Raspi (rot) gesteuerte Move-Hub erkennt beim Farbspiegel-Beispiel Distanz und Farbe eines Objektes.

Der Move-Hub unterscheidet grundsätzlich zwischen den eigentlichen Sensoren und den Ports, an denen sie angeschlossen sind. Dabei wird in der Bluetooth-Kommunikation kein Unterschied zwischen den externen Ports C und D und den internen Ports gemacht, auch wenn sich die internen Sensoren natürlich nicht an-, ab- oder umstecken lassen. Aus diesem Grund haben wir den internen Ports Namen gegeben, die sich an den dort fest verdrahteten Sensoren orientieren. Nichtsdestotrotz unterscheidet der Hub auch bei internen Sensoren zwischen Port und Sensor.

Die folgenden zwei Bytes (09 und 00) geben den Farbcode und die Entfernung an. Der Wert 09 steht dabei wie beim WeDo 2.0 für Rot, der Wert 00 für minimale Entfernung. Das erkannte Objekt war also rot und befand sich direkt vor dem Sensor.

Die Farbwerte wie auch sämtliche anderen uns bekannten Konstanten sind im Source-Code unseres Monitor-Skripts dokumentiert. Die Bedeutung der verbleibenden zwei Bytes (ff 06) ist unbekannt. Dies ist nur ein Beispiel für die Dinge, die noch der Erforschung bedürfen.

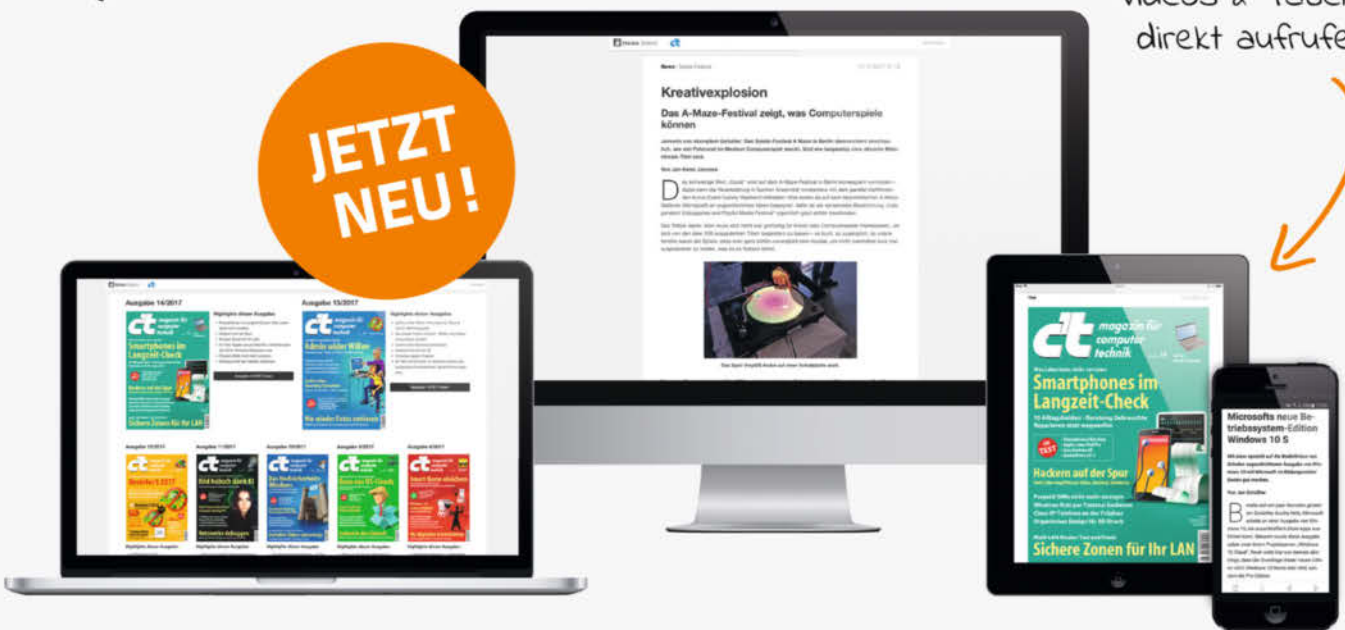


UNABHÄNGIG IN JEDER HINSICHT: c't als Heft und digital auf jedem Gerät lesen

In jedem Browser &
auf jedem Gerät c't lesen

Videos & Tabellen
direkt aufrufen

**JETZT
NEU!**



Lesenzeichen & Kommentare
synchronisieren

Optimale Darstellung für
mobile Endgeräte


Testen Sie jetzt c't online und stöbern Sie
3 Monate lang in allen verfügbaren Ausgaben.


Mit Willkommensgeschenk: z.B. ein Kino- und Snack-Gutschein

Zum Angebot: ct.de/digital-erleben

Bitte bei Bestellung über Telefon oder E-Mail angeben: 1CEA1713

**JETZT
TESTEN!**

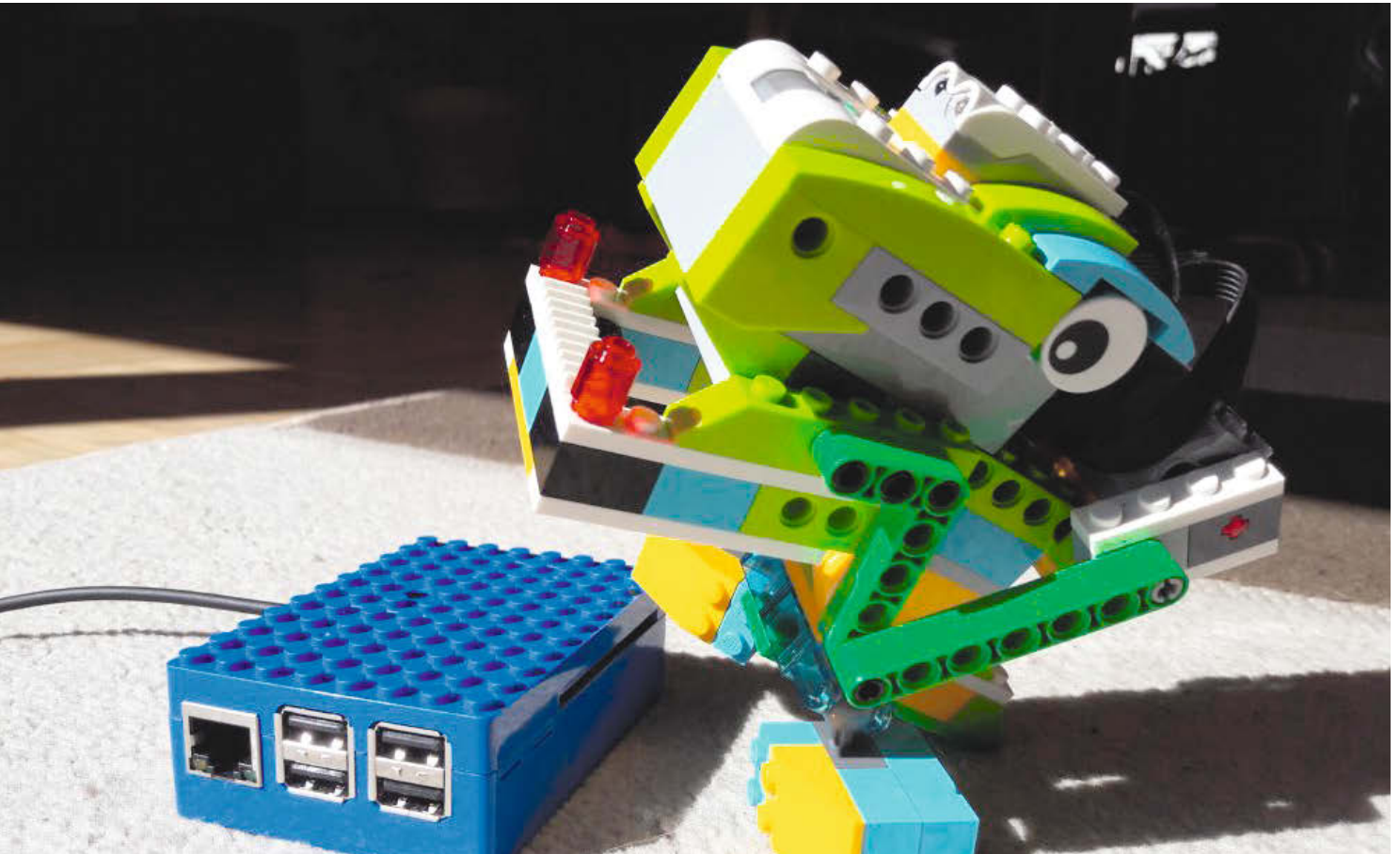
 ct.de/digital-erleben

 +49 541/80 009 120

 leserservice@heise.de

Dr. Till Harbaum

Spielzeug-Roboter per Raspberry Pi steuern

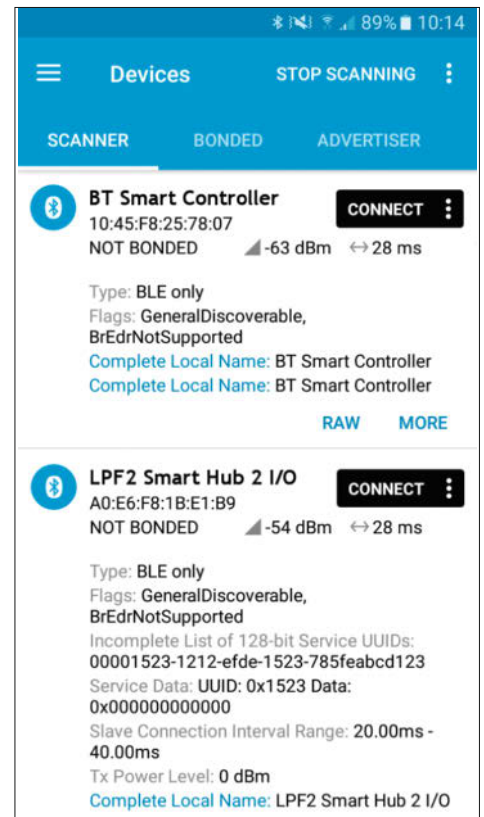


Roboter-Bausätze von Lego und Fischertechnik ermöglichen auch handwerklich weniger versierten Nutzern den Einstieg in die Robotik. Die herstellereigenen Controller sind jedoch teuer und nicht sonderlich leistungsfähig. Als alternative Steuereinheit bietet sich ein Raspberry Pi mit Bluetooth an.

Der Raspberry Pi eignet sich auch für Robotik-Experimente. Ganz einfach klappt das mit Konstruktionsbaukästen etwa von Lego oder Fischertechnik. Beide Hersteller haben mit dem „Lego EV3“ respektive dem „Robotics TXT“ auch eigene Linux-Controller im Angebot. Die kosten allerdings um die 300 Euro und erreichen trotzdem nicht ansatzweise die Leistung eines aktuellen Raspberry Pi 3. Beide Hersteller bieten zum etwa halben Preis aber auch Sets mit einfacheren Controllern an, die sich dank Bluetooth am Raspberry Pi betreiben lassen und ihn auf diesem Weg mit den Sensoren und Aktoren des jeweiligen Baukastenroboters verbinden.

Legos WeDo-2.0-Controller und der neue Boost-Controller nutzen dabei ebenso Verbindungen per Bluetooth Low Energy (BLE) wie der BT-Smart-Controller und das Bluetooth-Control-Set von Fischertechnik. Beide Hersteller folgen damit dem Trend weg vom PC und hin zu Mobilgeräten. Die Intelligenz bei diesen Systemen steckt dabei im Tablet oder Smartphone, wo per Drag-and-Drop programmiert wird und die eigentliche Programmausführung stattfindet. Doch auch der Raspi kann Roboter per BLE steuern. Die dafür nötigen Linux-Programme muss man dann allerdings selbst schreiben, beispielsweise in der Programmiersprache Python.

Die App nRF Connect listet die Verbindungen zu den Controllern von Lego und Fischertechnik mit den wesentlichen Funktionen auf.



setztes Tool finden Sie ebenfalls unter ct.de/w7z7.

Kommunikationsprofil

Die eigentliche Kommunikation über BLE GATT erfolgt durch das Lesen und Schreiben der namensgebenden Attribute. Unter Linux gibt es dafür `gatttool`. Durch Eingabe von `gatttool -I` startet man einen interaktiven Modus, in dem man manuell Geräte verbinden und mit ihnen kommunizieren kann. Die

Ein erster Blick

BLE hat sich vor allem bei Smartphones etabliert und zugehörige Verbindungstools sind in den App-Stores zu finden. Ich habe das kostenlose „nRF Connect“ von Nordic Semiconductors verwendet, um die beiden Spielzeug-Controller einer ersten Analyse zu unterziehen. Dank des von BLE genutzten Generic Attribute Profils (GATT) lassen sich die für die Steuerung notwendigen Funktionen und Fähigkeiten der Controller leicht abfragen.

Neu ist bei BLE gegenüber klassischem Bluetooth, dass ein Gerät ohne bestehende Verbindung durch sogenannte Undirected-Advertisements außer seinem Namen auch detaillierte Informationen über seine angebotenen Dienste verbreiten kann. Dafür werden Bluetooth-üblich 128-Bit lange Kennungen (sogenannte UUIDs) verwendet, die Dienste und einzelne Funktionen eines BLE-Gerätes eindeutig identifizieren. Der WeDo-2.0-Hub meldet während des Scans einen Dienst mit der UUID 00001523-1212-efde-1523-785feabcd123 und ist darüber als WeDo-Hub zu erkennen, selbst wenn der Benutzer den Gerätenamen per Lego-App verstellt.

Der Fischertechnik BT-Smart-Controller ist etwas weniger mittelsam und verbreitet ungefragt lediglich seinen Namen. Da der Name bei Fischertechnik nicht veränderbar ist, lässt er sich zur Geräteerkennung nutzen. Bezieht man zusätzlich den herstellerspezifischen Teil der Bluetooth-Adresse ein, so ist auch er sicher zu erkennen.

BLE am Raspberry Pi

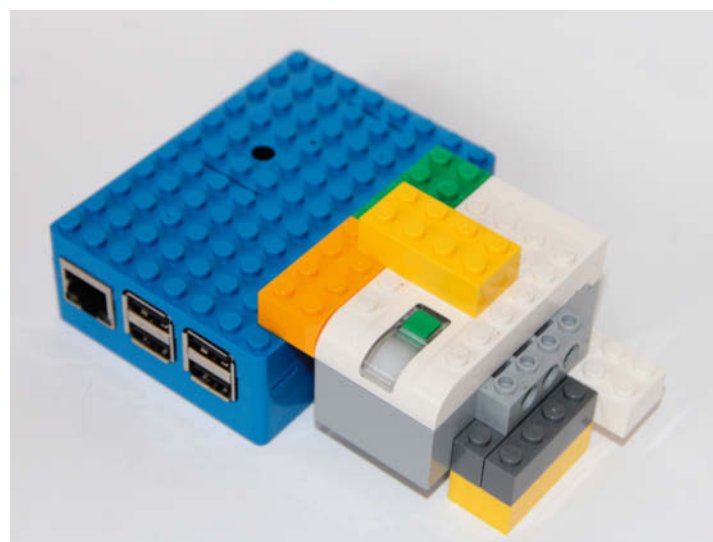
Da beide Controller ausschließlich den Low-Energy-Standard von Bluetooth 4.0 nutzen, muss der verwendete Raspberry Pi diesen Standard ebenfalls unterstützen. Beim eingebauten Bluetooth-Con-

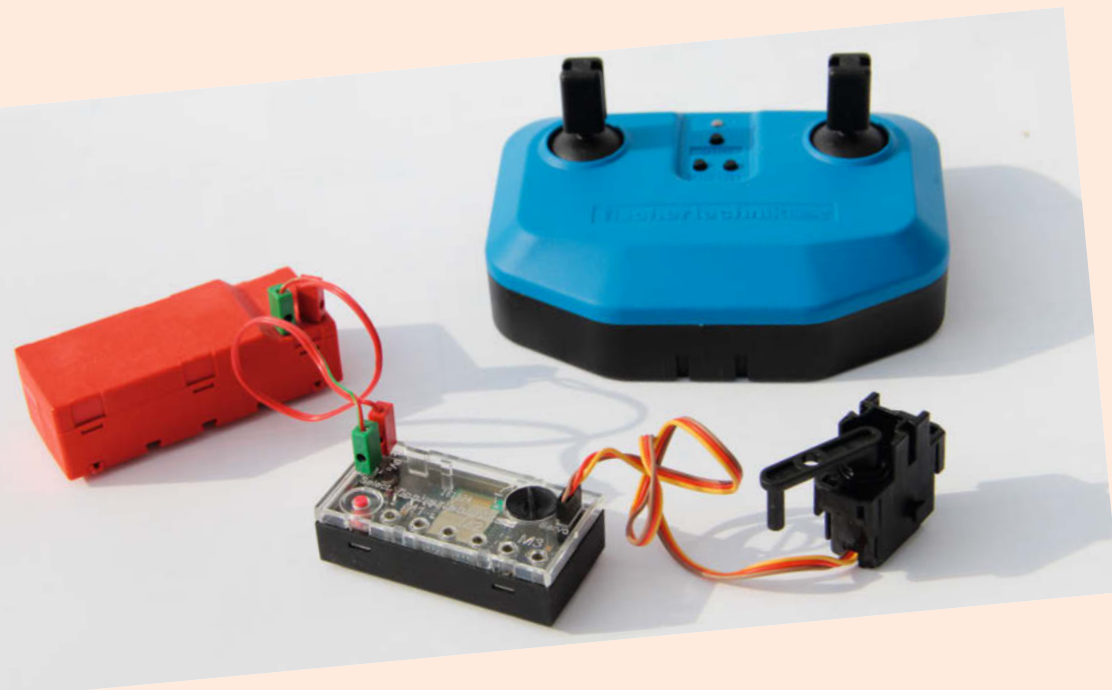
troller des Raspberry Pi 3 und 3B+ ist das der Fall. Bei nachgerüsteten USB-Dongles an älteren Raspis muss man neben Bluetooth-4.0-Fähigkeiten explizit auf Unterstützung des Low-Energy-Standards achten. Unter ct.de/w7z7 stellen wir eine kleine Sammlung von Programmen und Skripten bereit. Das Linux-Script `ble-test.sh` stellt fest, ob der Bluetooth-Adapter die nötigen BLE-Fähigkeiten mitbringt.

Obwohl der BLE-Standard bereits ein paar Jahre alt ist, unterstützt ihn Linux-Blue nur rudimentär. Das wichtigste Werkzeug ist, außer den auch für klassisches Bluetooth zuständigen Programmen „`hciconfig`“ und „`hcitool`“, das BLE-spezifische „`gatttool`“. Diese Programme sind bei aktuellen Distributionen wie Raspbian vorinstalliert.

Unter dem Link haben wir auch einen Patch für `hcitool` hinterlegt, der dessen „`lescan`“-Kommando zum Suchen nach BLE-Geräten erweitert, sodass es wie `nRF-Connect` neben Bluetooth-Adresse und Gerätenamen weitere Informationen anzeigt. Unsere Beispiele laufen auch ohne diesen Patch, aber er hilft unter anderem beim Identifizieren des Lego-WeDo-Hub via UUID. Ein fertig für den Raspi über-

Im Lego-Gehäuse mit angestecktem WeDo-Hub dient der Raspberry Pi als sehr leistungsfähige und flexible Robotik-Steuerung.





Das Racing-Set von Fischer-technik kann ebenfalls über das Control-Set (vorn) per Bluetooth mit dem Raspi gesteuert werden (siehe Beispiele im Download-Archiv).

meisten Befehle kann man gatttool aber auch direkt als Parameter mitgeben. Im Download befinden sich drei einfache Beispiel-Skripte: batterie.sh, ft_bt_smart_led_blink.sh und lego_wedo_led_blink.sh. Sie verwenden hcitool und gatttool, um die Controller zu erkennen und ihren Batteriestand abzufragen respektive die in beiden Controllern vorhandenen LEDs blinken zu lassen.

Der BLE-Zugriff auf die Controller folgt dabei immer dem gleichen Muster. Zunächst muss die Bluetooth-Adresse des Gerätes gefunden werden. Das geschieht mithilfe des hcitool auf Basis der ersten drei Bytes der Bluetooth-Adresse (OID), des Bluetooth-Namens oder der eindeutigen Kennungen (UUIDs). Ist die Adresse bereits bekannt, kann die Suche per hcitool entfallen. Alle unsere Beispiele erlauben es daher, die Bluetooth-Adresse des anzusprechenden Controllers auch direkt als Parameter anzugeben. Über die Bluetooth-Adresse des Controllers baut gatttool die eigentliche Verbindung auf.

Nach dem Verbindungsaufbau werden die UUIDs der Dienste des Controllers abgefragt und im letzten Schritt die UUIDs der im Weiteren benötigten Funktionen eines Dienstes, welche die BLE-Spezifikation als GATT-Characteristics bezeichnet. Im Falle der Spielzeugcontroller sind vor allem die Funktionen zum Zugriff auf die Ein- und Ausgänge zum Anschluss von Motoren und Sen-

soren interessant. Sobald die Geräteadresse und Adressen der Funktionen bekannt sind, lassen sich die Ein- und Ausgänge ansteuern.

Die genaue Bedeutung der GATT-Funktionen und deren UUIDs stehen in ihren jeweiligen Beschreibungen auf

bluetooth.com (siehe ct.de/w7z7). Hersteller- und gerätespezifische Funktionen werden aber in der Regel nicht durch standardisierte Dienste und Funktionen abgedeckt, sodass man Unterlagen der Hersteller braucht oder eigene Forschungen betreiben muss.

WICHTIGSTE HERSTELLERSPEZIFISCHE FUNKTIONEN FÜR LEGO WEDO-2.0-HUB

Funktion	Schreiben von Ausgangs-Kommandos
Service-UUID	00004f0e-1212-efde-1523-785feabcd123
Funktions-UUID	00001565-1212-efde-1523-785feabcd123
Beispiel-Byte-Sequenzen	
Motor setzen	CH 01 01 SD (CH=1/2, SD = Richtung -100...+100)
Ton ausgeben	05 02 04 FL FH DL DH (FH/FL= Frequenz, DH/DL = Dauer)
LED-Farbindex setzen	06 04 01 CI (CI=Farbe 01-09, 10 = aus)
Funktion	Schreiben von Sensor-Konfigurationen
Service-UUID	00004f0e-1212-efde-1523-785feabcd123
Funktions-UUID	00001563-1212-efde-1523-785feabcd123
Beispiel-Byte-Sequenzen	
Neigungssensor-Modus setzen	01 02 CH 22 TM 01 00 00 00 02 01 (CH = 1/2, TM 00 = Winkel, 01 = Neigung)
Bewegungssensor-Modus setzen	01 02 CH 23 MM 01 00 00 00 02 01 (CH = 1/2, MM 00 = Distanz, 01 = Zähler)
Funktion	Lesen von Sensor-Werten
Service-UUID	00004f0e-1212-efde-1523-785feabcd123
Charakteristik-UUID	00001560-1212-efde-1523-785feabcd123
Beispiel-Byte-Sequenzen	
Neigungssensor an Eingang 2	03 02 00 00 a0 40 (Byte 2-5: 32-Bit-Float-Wert)
Funktion	Lesen von Anschlussereignissen
Service-UUID	00001523-1212-efde-1523-785feabcd123
Charakteristik-UUID	00001527-1212-efde-1523-785feabcd123
Beispiel-Byte-Sequenzen	
Neigungssensor (Typ 22) an Anschluss 1 angesteckt	01 01 00 22 00 00 00 10 00 00 00 10
Motor (Typ 01) an Anschluss 2 angesteckt	02 01 01 01 01 00 00 00 01 00 00 00
Gerät von Anschluss 2 entfernt	02 00