

e lektor

www.ektor.de

Android-Tablet steuert Zeitraffer-Aufnahmen

Unknackbares Codeschloss
IR mit 128-bit-AES-Verschlüsselung

USB-Datenlogger
Flash-Stick am Mikrocontroller

SMD-LED-Kerze
mit Sensor zum Ausblasen!

Digitaler Audio-Generator
DSP-Board in der Praxis

Personal Download for I © Elektor



4 198630 307401

Neue 8-bit-Mikrocontroller mit integrierter konfigurierbarer Logik in 6- bis 20-poligen Gehäusen

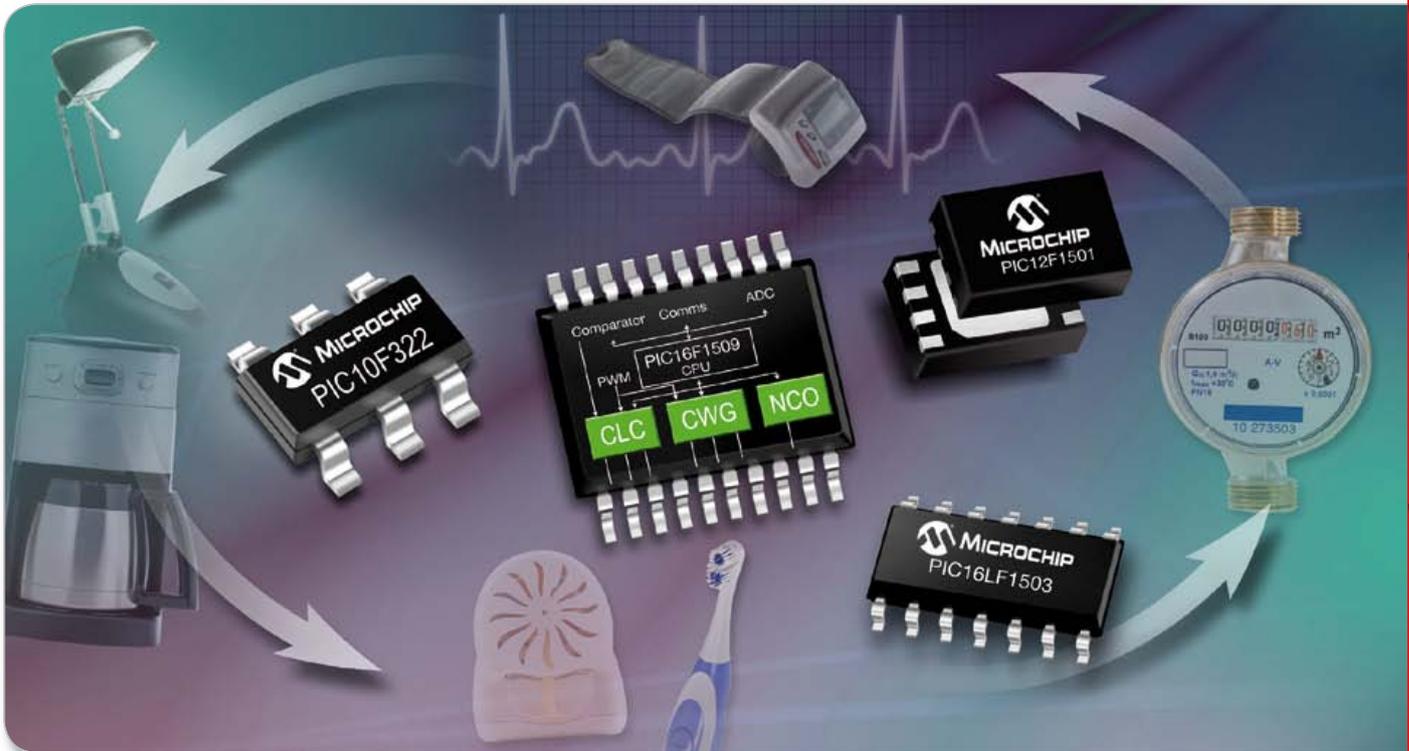
Microcontrollers

Digital Signal
Controllers

Analog

Memory

RF & Wireless



Mit den neuen 8-bit-Mikrocontrollern PIC10F32X, PIC12F150X und PIC16F150X von Microchip können Sie zusätzliche Funktionalitäten in Ihre Anwendungen aufnehmen, die Größe reduzieren, Energie sparen und Kosten senken. Sie sind insbesondere für preisgünstige Anwendungen oder Einwegprodukte geeignet. Onboard befinden sich konfigurierbare Logikzellen (CLCs), ein komplementärer Funktionsgenerator (CWG) und ein numerisch gesteuerter Oszillator (NCO).

Die kombinatorische und sequentielle Logik lässt sich über die konfigurierbaren Logikzellen (CLCs) per Software steuern. Dies hat den Vorteil, dass Funktionalitäten hinzugefügt, externe Komponenten eliminiert und Codeplatz eingespart werden können. Der komplementäre Funktionsgenerator (CWG) hilft bei der Verbesserung der Schalteffizienz zwischen den verschiedenen Peripherien, während der numerisch gesteuerte Oszillator (NCO) eine lineare Frequenzeinstellung und höhere Auflösung der Anwendung ermöglicht, wie zum Beispiel in Tongeneratoren und Vorschaltgeräten.

Zusätzlich zur Einführung dieser neuen Peripherien bieten der PIC10F/LF32X und der PIC12/16F/LF150X MCUs einen internen 16-MHz-Oszillator, einen ADC, bis zu vier PWMs sowie ein integriertes Temperaturmessmodul zur preisgünstigen Temperaturmessung. Das alles ist in kompakten 6- bis 20-poligen Gehäusen untergebracht.

ENTWICKLUNGSWERKZEUGE FÜR DEN SCHNELLSTART



PICDEM™ Lab Entwicklungs-Kit
- DM163045



PIC16F193X F1 Evaluationsplattform
- DM164130-1



PICkit™ Demoplattine für geringe
Anschlusszahlen - DM164120-1

Freie CLC-Konfigurationswerkzeuge:
www.microchip.com/get/euctool

Erfahren Sie mehr über PIC® MCUs mit der Peripherie der nächsten Generation
und geringer Anschlusszahl: www.microchip.com/get/eunew8bit

microchip
DIRECT
www.microchipdirect.com

www.microchip.com

MICROCHIP

Prozessoren werden immer leistungsfähiger und stromsparender, dazu kommen technische Fortschritte bei Akkus und Displays – dies macht es möglich, dass heute jedermann einen kleinen Computer in der Hosentasche herumtragen kann, der zum Surfen, für Multimedia und vielerlei andere Anwendungen geeignet ist. Diese Rechner, welche die Bezeichnung „Mikrocomputer“ erstmals wirklich verdienen, sind überdies satt ausgestattet: WLAN, Bluetooth, Touchscreen, Kamera, GPS, Beschleunigungssensor, reichlich RAM und vieles mehr findet man fast bei allen Geräten. Das sind Qualitäten, die auch in der Embedded-Entwicklung begehrt sind. Bei Anwendungen zum Messen, Steuern und Regeln muss viel Zeit und Energie dafür aufgewendet werden, ein passendes Bedieninterface zu entwickeln, sowohl auf der Hardware- als auch auf der Softwareseite. Weitere Herausforderungen sind das Logging von Daten und ein Netzzugang. Oft ist es da am praktischsten, einfach einen PC zu nehmen. Und es soll noch immer Kuhställe geben, in denen ein C64 seinen Dienst verrichtet!

Spätestens jetzt drängt sich die Frage auf, ob man nicht auch ein Smartphone oder Tablet als Steuerzentrale benutzen könnte. An den Anschaffungskosten soll es nicht mehr scheitern: Die Geräte sind inzwischen zu Preisen von ab 100 Euro zu haben; ein Mikrocontrollerboard, das eine ähnliche Ausstattung aufweist, kostet leicht das Dreifache.

Nun steht man aber vor zwei Problemen: Wie schließe ich so ein Smartphone an die eigene Elektronik an? Und wie kann ich die Software – zum Beispiel für das quelloffene Android-Betriebssystem - möglichst einfach programmieren? Beide Fragen werden von Elektor in den kommenden Heften ausführlich beantwortet. Den Anfang macht ein Artikel zu Android-Tablets nebst einer Beispielanwendung in diesem Heft. In der nächsten Ausgabe werden wir dann eine Platine vorstellen, die den Anschluss eigener Elektronik an Android-Smartphones sehr einfach macht.

Das klassische Mikrocontrollerboard hat natürlich weiterhin seine gewichtigen Vorteile. Ich sehe die kleinen Computer hier nicht als Konkurrenz – vielmehr könnte nun wieder zusammenwachsen, was zusammengehört: In Zukunft werden sich noch mehr Elektroniker mit Computern beschäftigen, doch genauso wird so mancher Computerfreak Lust auf Elektronik bekommen!

Jens Nickel

6 Impressum

Who is who bei Elektor

8 News

Neuheiten, Nachrichten und Termine

12 Der PCB Prototyper in der Praxis

Der PCB Prototyper ist eine hochpräzise Fräsmaschine - wir haben uns bei einem überzeugten Anwender umgeschaut.

14 Android als Entwicklungsumgebung

Tablet-Computer sind eine wertvolle Ergänzung eigener Elektronik – ein Überblick.

18 Zeitraffer mit Android-Tablet

Ein Android-Tablet, eine Digitalkamera und ein Servo spielen hier zusammen.

22 USB-Stick am Mikrocontroller

Ein PIC24F von Microchip sorgt für die nötige USB-Host-Funktionalität.

26 Elektronische LED-Kerze

Ein beheizter NTC detektiert die Luftbewegung beim Ausblasen der Kerze.

32 Unknackbares Codeschloss

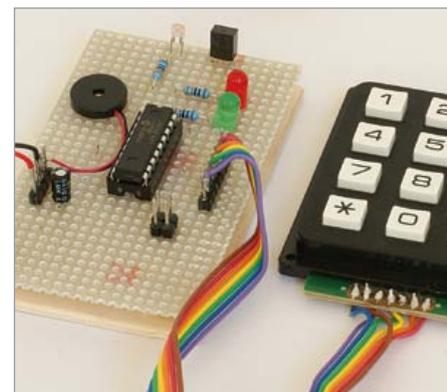
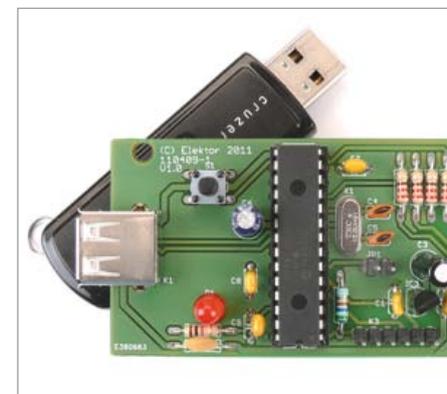
Der Hardwareaufwand ist minimal: Die Schaltung kann einfach auf einem Stück Lochrasterplatine aufgebaut werden.

36 Hier kommt der Bus (10): Messen mit 22 bit

Jetzt wird's hochauflösend: An unseren Busknoten schließen wir einen externen ADC an, der Werte mit 22 bit sampeln kann. Dank der SPI-Schnittstelle ist das Einlesen der Daten in den Mikrocontroller recht einfach.

43 Labcenter

LED-Exorzismus
Stinkender Bus
Itsy Bitsy Spider...
Pins perfekt auf Maß
Vorsicht, Baustelle!



14 Android-Tablets als Steuerung

Die im Preis drastisch gesunkenen Tablet-Computer bringen einen hochauflösenden Touch-Screen sowie reichlich Speicher und Rechenleistung mit. Für Anwendungen wie das Messen, Steuern und Regeln sind die Geräte geradezu prädestiniert!

Neben einem Überblicks-Artikel findet man in diesem Heft auch ein Beispiel-Projekt. Ein Android-Tablet lässt eine Digitalkamera in festgelegten Intervallen auslösen, so dass ein Zeitraffer-Effekt entsteht. Die vorgestellte Steuerung eignet sich für die verschiedensten Kameras, da der Auslöser mit Hilfe eines Servos betätigt wird.

22 USB-Stick am Mikrocontroller

Mit einem USB-Stick hat man einen bewährten und billigen nichtflüchtigen Speicher, mit dem sich Daten in idealer Weise aufzeichnen, transportieren und transferieren lassen. Für den Anschluss eines USB-Sticks an eigene Elektronik benötigt man aber einen Host-Controller, außerdem muss die Firmware Daten in Files ablegen können. Beide Aufgaben wurden in diesem Projekt elegant gelöst.

26 Elektronische LED-Kerze

Kerzenimitate mit einer LED als Kunstflamme gibt es überall zu kaufen. Doch hier geht es um eine Variante, die dem Vorbild viel näher kommt: Diese elektronische Kerze flackert nicht nur realistisch - man kann sie auch ausblasen! Bei der LED-Kerze kommt der Typ PIC16F1827 von Microchip zum Einsatz. Sein 4 KB fassender Programmspeicher reicht für die Animation des Lichts und die anderen Funktionen locker aus.

32 Unknackbares Codeschloss

Haben Sie sich schon einmal gefragt, ob die Funkverriegelung für Ihr Auto wirklich sicher ist? Dieses Projekt demonstriert, wie man mit Hilfe eines Standard-Mikrocontrollers sichere und praktisch unknackbare Sender und Empfänger für eine IR-Fernbedienung realisieren kann. Dabei wird die Zweirichtungskommunikation per 128-bit-AES-Algorithmus verschlüsselt.

48 Kontaktlose BasicCard

Die einzige in Basic programmierbare Chipkarte ist schon mehr als 10 Jahre auf dem Markt. Als eine der bekanntesten Karten mit offenem Betriebssystem ist sie nun durch eine RFID-Version noch interessanter geworden.

53 Vom Oszilloskop zum Reflektometer

Mit einem Reflektometer kann man nicht nur Fehler in Kabeln orten, sondern sogar ermitteln, in welchem Abstand vom Kabelende der Fehler gesucht werden muss!

56 Audio-DSP-Kurs (6): Digitaler Audiogenerator

Ein Signalgenerator für Audio sollte zum einen äußerst klirrarmer Sinusschwingungen und zum anderen Tief- und Bandpassrauschsignale erzeugen können. Ein DSP ist für solche Aufgaben besonders geeignet!

64 Xcos: Modellierer und Simulator

Scilab ist ein Open-Source-Programm zum Lösen mathematisch-wissenschaftlicher Problemstellungen. Dazu gehört Xcos, ein Modellierer und Simulator für wichtige Aufgabenbereiche des Ingenieurs.

70 Robusta: Studenten bauen einen Satelliten

Der Picosatellit trägt einen Versuchsaufbau zur Alterung elektronischer Bauteile.

76 Retronik

RCA Cosmac Development System IV von 1978

78 Hexadoku

Sudoku für Elektroniker

80 Elektor-Shop

Bücher, CDs, DVDs, Bausätze & Module

84 Vorschau

Nächsten Monat in Elektor

Eine multimediale und interaktive Plattform für jeden Elektroniker - das bietet Elektor International Media. Ob Anfänger oder Fortgeschrittener, ob Student oder Professor, ob engagierter Profi oder leidenschaftlicher Hobbyist: Hier finden Sie wertvolle Informationen, Inspiration für die eigenen Entwicklungen, Unterstützung bei der Ausbildung und nicht zuletzt eine gute Portion Unterhaltung. Gedruckt und im Web. Analog und digital. In Theorie und Praxis.

**ANALOG • DIGITAL
EMBEDDED • MIKROCONTROLLER
AUDIO • MESSTECHNIK**

IMPRESSUM

42. Jahrgang, Nr. 492 Dezember 2010
Erscheinungsweise: 11 x jährlich (inkl. Doppelheft Juli/August)

Elektor möchte Menschen anregen, sich die Elektronik zu Eigen zu machen – durch die Präsentation von Projekten und das Aufzeigen von Entwicklungen in der Elektronik und technischen Informatik.

Elektor erscheint auch in Englisch, Französisch, Niederländisch, Spanisch und weiteren Sprachen.
ELEKTOR ist in über 50 Ländern erhältlich.

Verlag
Elektor-Verlag GmbH - Süsterfeldstraße 25, 52072 Aachen
Tel. 02 41/88 909-0 - Fax 02 41/88 909-77

Technische Fragen bitten wir per E-Mail an redaktion@elektor.de zu richten.

Internationale Chefredaktion **Wisse Hettinga**

Redaktion Elektor Deutschland
Ernst Krempelsauer (Chefredakteur, v.i.S.d.P.)
Jens Nickel
(E-Mail: redaktion@elektor.de)

Internationale Redaktion
Harry Baggen, Thijs Beckers, Jan Buiting, Eduardo Corral, Clemens Valens

Redaktionssekretariat **Hedwig Hennekens**

Labor/Technische Redaktion
Christian Vossen (Ltg.), Thijs Beckers, Ton Giesberts, Luc Lemmens, Raymond Vermeulen, Jan Visser

Grafische Gestaltung und Layout
Giel Dols, Mart Schroyen

eC-reflow-mate

NEU!

➔ Professioneller SMD-Reflow-Ofen für perfekt gelötete Platinen

Der „eC-reflow-mate“ ist die perfekte Lösung, wenn es um das Löten von Platinen für Prototypen und Kleinserien mit SMD-Komponenten geht. Er hat einen großzügig bemessenen Innenraum, so dass mehrere Standard-Platinen gleichzeitig gelötet werden können. Zwei fest eingebaute Temperatursensoren und die neu konstruierten, nicht linearen Infrarot-Heizstäbe sorgen dafür, dass die Wärme im Innenraum homogen verteilt wird. Mit einem dritten, beweglichen Sensor kann die Temperatur der Platinenoberfläche oder die einer ausgewählten Komponente in die Steuerung einbezogen werden. Das Geschehen im Ofen lässt sich durch die Glasfront der Schublade jederzeit beobachten. Die Konstruktion des „eC-reflow-mate“ hält auch intensiver Dauerbeanspruchung stand.

Der neue „eC-reflow-mate“ ist ein kompromisslos konstruierter, universeller SMD-Lötfen – unverzichtbar für alle, die mit SMDs bestückte Platinen in Einzelanfertigungen oder Kleinserien entwickeln und produzieren.



Technische Daten:

- Betriebsspannung: 230 V/50 Hz
- Leistung: 3500 W
- Gewicht: ca. 29 kg
- Abmessungen: 620 x 245 x 520 mm (B x H x T)
- Heizung: Infrarot-Strahler kombiniert mit Umluft
- Bedienung: Funktionstaster und LC-Display am Ofen oder über PC-Programm und USB-Verbindung
- Temperaturbereich: 25...300 °C (300 °C Spitzenwert und 260 °C beim Löten)
- Effektive Platinengröße: max. 350 x 250 mm
- Temperatursensoren: 2 feste interne Sensoren, 1 externer Sensor (im Lieferumfang)
- Menüsprachen: Deutsch, Englisch, Französisch, Italienisch, Niederländisch, Ungarisch

Besonderheiten:

- Infrarot-Strahler für homogene Temperaturverteilung
- Schublade fährt nach Lötprozess-Ende selbsttätig aus
- Fenster in der Gerätefront zur ständigen Sichtkontrolle

Art.-Nr.: 100447-91

Preis: 2495,00 € (zzgl. MwSt. und Versand)

Weitere Infos & Bestellung unter

www.elektor.de/reflow-mate

Geschäftsführer/Herausgeber: Don Akkermans

Marketing/Vertrieb (Leitung): Carlo van Nistelrooy

Anzeigen (verantwortlich): Irmgard Ditzgens
ID Medienservice

Tel. 05 11/61 65 95-0 - Fax 05 11/61 65 95-55

E-Mail: service@id-medienservice.de

Es gilt die Anzeigenpreisliste Nr. 40 ab 01.01.2010

Vertriebsgesellschaft: IPS Pressevertrieb GmbH

Postfach 12 11, 53334 Meckenheim

Tel. 0 22 25/88 01-0 - Fax 0 22 25/88 01-199

E-Mail: elektor@ips-pressevertrieb.de

Internet: www.ips-pressevertrieb.de

Vertrieb Österreich

Pressegroßvertrieb Salzburg/Anif - Niederalm 300

Tel. +43/62 46/37 21-0

Der Herausgeber ist nicht verpflichtet, unverlangt eingesandte Manuskripte oder Geräte zurückzusenden. Auch wird für diese Gegenstände keine Haftung übernommen. Nimmt der Herausgeber einen Beitrag zur Veröffentlichung an, so erwirbt er gleichzeitig das Nachdruckrecht für alle ausländischen Ausgaben inklusive Lizenzen. Die in dieser Zeitschrift veröffentlichten Beiträge, insbesondere alle Aufsätze und Artikel sowie alle Entwürfe, Pläne, Zeichnungen einschließlich Platinen sind urheberrechtlich geschützt. Ihre auch teilweise Vervielfältigung und Verbreitung ist grundsätzlich nur mit vorheriger schriftlicher Zustimmung des Herausgebers gestattet. Die veröffentlichten Schaltungen können unter Patent-

oder Gebrauchsmusterschutz stehen. Herstellen, Feilhalten, Inverkehrbringen und gewerblicher Gebrauch der Beiträge sind nur mit Zustimmung des Verlages und ggf. des Schutzrechtsinhabers zulässig. Nur der private Gebrauch ist frei. Bei den benutzten Warenbezeichnungen kann es sich um geschützte Warenzeichen handeln, die nur mit Zustimmung ihrer Inhaber warenzeichengemäß benutzt werden dürfen. Die geltenden gesetzlichen Bestimmungen hinsichtlich Bau, Erwerb und Betrieb von Sende- und Empfangseinrichtungen und der elektrischen Sicherheit sind unbedingt zu beachten. Eine Haftung des Herausgebers für die Richtigkeit und Brauchbarkeit der veröffentlichten Schaltungen und sonstigen Anordnungen sowie für die Richtigkeit des technischen Inhalts der veröffentlichten Aufsätze und sonstigen Beiträge ist ausgeschlossen.

© 2011 elektor international media b.v.

Druck: Senefelder Misset, Doetinchem (NL)



ISSN 0932-5468

Green Flight Challenge: Europäische Elektroflugzeuge konkurrenzlos

Mit insgesamt 1,65 Millionen Dollar an Preisgeld war der von der NASA und von Google unterstützte Green Flight Challenge der amerikanischen Comparative Aircraft Flight Efficiency Foundation (CAFE) der am höchsten dotierte Flugwettbewerb aller Zeiten. Von den Ende 2010 zum Ökoflug-Wettbewerb zugelassenen 13 Teams waren bei der Austragung im kalifornischen Santa Rosa Ende September letztlich nur noch drei Teams mit qualifizierten Flugzeugen dabei.

Den ersten Preis in Höhe von 1,35 Millionen Dollar hat das elektrische Doppelrumpf-Flugzeug Taurus G4 von Pipistrel gewonnen, das speziell für den Wettbewerb aus zwei 2-sitzigen Taurus-Motorseglern dieses slowenischen Herstellers zusammengebaut worden war.

Der mit 120.000 Dollar dotierte zweite Preis und der Lindbergh-Preis für das leiseste Flugzeug in Höhe von 10.000 Dollar ging an das Team eGenius der Universität Stuttgart, das mit einem ebenfalls elektrisch angetriebenen, aber zweisitzigen Flugzeug angetreten war.

Um zu gewinnen, mussten die teilnehmenden Flugzeuge 200 Meilen (322 km) in weniger als zwei Stunden fliegen und dabei mit weniger als einer Gallone Benzin (= 33,7 kWh Energieäquivalent) pro Sitzplatz auskommen. Auf 100 km durfte ein Zweisitzer somit maximal 2,36 l (= 21 kWh) und ein Viersitzer maximal 4,72 l (= 42 kWh) verbrauchen – bei einer Durchschnittsgeschwindigkeit von mindestens 161 km/h.

Durch die Wertung des Verbrauchs pro Sitzplatz war das viersitzige Experimentalflugzeug von Pipistrel gegenüber dem zweisitzigen eGenius prinzipiell im Vorteil. Trotzdem mussten sich die Stuttgarter mit weniger als 7% Verbrauchsunterschied pro Sitzplatz nur knapp geschlagen geben, wozu auch ein Problem mit dem Navigationssystem beigetragen haben könnte. Beim absoluten Verbrauch (kWh/100 km) waren sie dem Taurus G4 ohnehin um 85% überlegen.

Beide Gewinner haben den geforderten Effizienz-Wert von 200 Sitzplatz-Meilen pro Gallone (pMPG) bei weitem übertrafen: Pipistrel kam auf 403,5 und e-Genius auf 375,7 pMPG. Umgerechnet in Liter, kWh und km wurden folgende Verbrauchswerte erzielt:

	Liter (kWh)/100 km pro Sitz	Liter (kWh)/100 km absolut
Pipistrel	0,58 (5,2)	2,32 (20,7)
e-genius	0,63 (5,6)	1,26 (11,2)

In der Geschwindigkeitswertung wurden Durchschnittsgeschwindigkeiten von 183 km/h (Pipistrel) und 173 km/h (e-genius) erzielt. Dabei sollte die 200-Meilen-Strecke möglichst schnell geflogen werden, wobei

aber nach der Landung eine für mindestens 30 Flugminuten ausreichende Energiereserve im Tank beziehungsweise im Akku nachzuweisen war. Beide Flugzeuge waren mit Lithium-Ionen-Akkus und bürstenlosen Elektromotoren unterwegs. Pipistrel mit 145-kW-Antrieb und 100-kWh-Akku, e-genius mit vergleichsweise bescheidenen 60 kW und 56 kWh. Auch das dritte Flugzeug in der Wertung stammte aus Europa. Der tschechische Phoenix-Motorsegler hatte aber mit seinem konventionellen Rotax-Benzinmotor keine Chance gegen die beiden Elektroflieger und konnte die CAFE-Forderungen ebensowenig erfüllen wie der außer Konkurrenz mitfliegende EcoEagle, ein modifizierter Stemme-S10-Motorsegler aus Berlin mit einem von Flight Design aus

Stuttgart-Echterdingen entwickelten Hybridantrieb (100-PS-Rotax-Benzinmotor kombiniert mit 30-kW-Elektromotor). Beide kamen beim Verbrauch noch nicht einmal auf die Hälfte der geforderten 200 Sitzplatz-Meilen pro Gallone.

Wäre die Praxistauglichkeit stärker bewertet worden, hätte ohne Zweifel eGenius den Green Flight Challenge gewinnen müssen. Der mit den Bauvorschriften CS22 für EASA-zugelassene Touring-Motorsegler konforme eGenius ist in Handling und Performance bereits heute einem modernen Reise-Motorsegler mit 80-PS-Benzinmotor ebenbürtig, aber viel leiser und emissionsfrei - und das bei 85% weniger Energieverbrauch!

www.elektor.de/
elektronik-news/GFC

Bildquellen:

Pipistrel und Universität Stuttgart



Nachrichten aus Forschung und Technik, interessante Produkt-Neuheiten und vieles mehr findet man aktuell unter www.elektor.de!

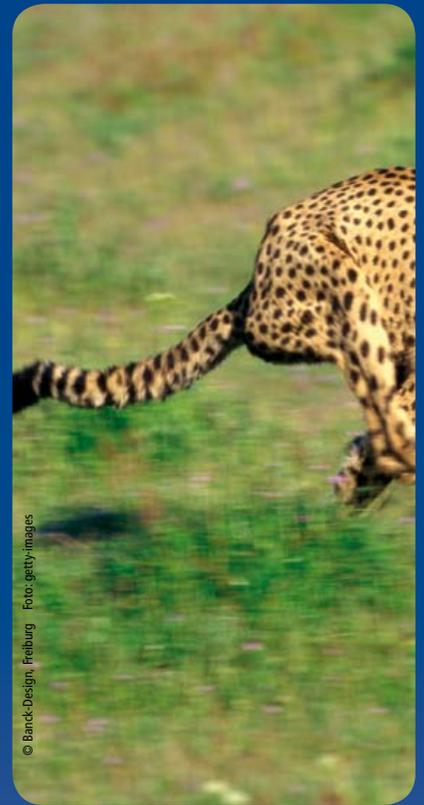
Ideale Diode

Linear Technology hat eine Reihe interessanter ICs für Stromversorgungen im Programm, unter denen der Typ LTC4358 dank seines 5-A-MOSFETs heraussticht. Ein diesen MOSFET ansteuernder interner Controller macht dieses IC zum besonders verlustarmen Ersatz für Schottky-Dioden in Anwendungen für Ströme bis zu 5 A und Spannungen bis 26,5 V. Der Chip verhält sich dank internem MOSFET mit nur 20 m Ω Kanalwiderstand wie eine Diode mit besonders niedriger Durchlassspannung. Dadurch ist die Verlustleistung selbst gegenüber der Verwendung einer Schottky-Diode deutlich reduziert. Man erzielt daher einen höheren Wirkungsgrad und spart Platz, weil kein Kühlkörper benötigt wird. Der LTC4358 steuert den Durchlassspannungsabfall des MOSFETs so, dass beim Umschalten von einem Pfad auf einen anderen keine Oszillationen auftreten. Eine Pull-Down-Schaltung minimiert Rückstromtransienten, falls eine Stromversorgung

ausfällt oder einen Kurzschluss aufweist. Der Baustein kann wie eine Diode mit drei Anschlüssen für allgemeine Anwendungen eingesetzt werden. Typische Einsatzgebiete sind z.B. der Verpolungsschutz für Elektronik-Baugruppen oder die Oder-Verknüpfung von Stromversorgungen in Anwendungen, die erhöhte Systemzuverlässigkeit erfordern. Das IC eignet sich auch für Anwendungen, bei denen mehrere redundante Stromversorgungen parallelgeschaltet sind und sich die Last teilen.

In N+1-redundanten Stromversorgungen wird somit der Ausfall eines Netzteils gepuffert. Eine solche Oder-Verknüpfung ermöglicht es darüber hinaus, Stromversorgungen während des laufenden Systembetriebs auszutauschen. Außerdem schützt eine solche Anordnung gegenüber Kurzschlüssen in einer der Stromversorgungen. Bei einem Ausfall oder Kurzschluss einer Stromversorgung gewährleistet das IC ein Abschalten innerhalb von 500 ns.

www.linear.com/product/LTC4358



© Barock-Design, Freiburg Foto: getty-images

WOW! Schon da!

**punktgenau
plangenau
preisgenau.**

**Der schnelle
Leiterplatten-
Spezialist
mit über
25 Jahren Erfahrung!**

**BECKER
MÜLLER** 

www.becker-mueller.de
Mit Online-Kalkulator!

Neue Version von EAGLE angekündigt



Farnell hat die neue Version 6.0 des bekannten Platinen-Layout-Editors EAGLE (Firma CadSoft) angekündigt. Bei der neuen Version wurde besonderer Nachdruck auf Benutzerfreundlichkeit und Flexibilität gelegt. Unter anderen Dingen wird nun auf eine

XML-Datenbank im Textformat gesetzt, wodurch Änderungen in Entwürfen schnell und einfach durchgeführt werden können. Ab sofort sind alle neuen Käufer der aktuellen Eagle-Version zum kostenlosen Bezug des Updates auf die Version 6.0 berechtigt.

EAGLE 6.0 wartet unter anderem mit folgenden Funktionen auf:

- XML-Datenbank; Informationen können somit unabhängig von Eagle gelesen und verändert werden.
- Optimierte Ablage; Schaltungs- und Platinen-Teile können per Copy & Paste übertragen werden, was besonders hilfreich bei Schaltungen mit identischen Abschnitten ist.
- Verbessertes Routing; Undo/Redo-Funktionen sowie Wandlung zwischen Zoll und metrischen Einheiten mit verbesserter interner Auflösung.
- Überarbeiteter Bibliotheks-Editor mit der Möglichkeit zur Erstellung interner Durchkontaktierungen und verbesserten Beschreibungsmöglichkeiten.

<http://de.farnell.com> www.cadsoft.de/

Von Phil Knurhahn

Hüpfender Mikroroboter

Hier stellen wir einen Mikroroboter vor, der nur vier Millimeter klein ist, aber zwanzig Mal so hoch hüpfen kann (Bild: UMD Micro-robotics). Der vom Army Research Laboratory gemeinsam mit der University of Maryland auf der International Conference on Intelligent Robots and Systems 2011 vorgestellte Prototyp hat alles an Bord: Eine Energiequelle, ein integriertes Steuerungssy-



stem und diverse Lichtsensoren. Um solche Sprünge zu machen, greift der Roboter auf eine chemische Reaktion zurück, bei der ein Gas gebildet wird – eine Art kontrollierter Explosion. Den Sprung schafft er derzeit nur ein einziges Mal. Die weiteren Arbeiten zielen darauf ab, mehrere Sprünge auszuführen. In einer modifizierten Variante soll dann auch die Sprunghöhe ausgebaut werden. Ziel ist es, etwa 65 m zu erreichen – das 16.000-fache der Roboter-Länge!

www.enme.umd.edu/news/news_story.php?id=6030

Erste Prozessoren für das „Cognitive Computing“

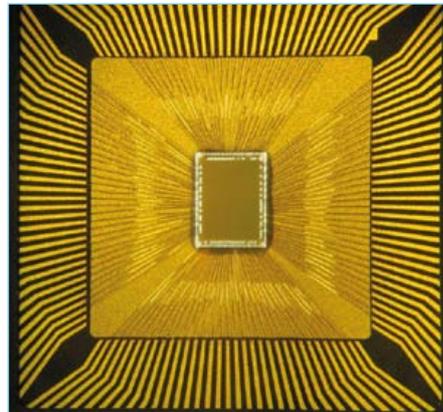
Wer erinnert sich schon an die ersten Mikroprozessoren? Ende der 60er Jahre entstand bei Intel die richtungsweisende Idee, statt anwendungsorientierter Prozessoren einen Universalprozessor zu entwickeln, der dann per Software auf seine Anwendung getrimmt wird. So entstand der 4004-Prozessor, der etwa so leistungsfähig war wie die berühmte ENIAC (Electronic Numerical Integrator and Computer) von 1946, aber etwa zehn Millionen Mal kleiner (Bild: Intel). Seitdem basieren alle Prozessoren auf einer Grundarchitektur, die der Princeton-Professor John von Neumann 1947 schuf: Ein Computer besteht aus Rechen-, Steuer-, Eingabe- und Ausgabeeinheit sowie einem Arbeitsspeicher. Er arbeitet mit definierten Programmen und weiteren Instruktionen, um die gestellten Aufgaben Schritt für Schritt durchzuführen.

Forscher des IBM Research Center in Almaden, Kalifornien, haben jüngst Prototypen einer neuen Klasse von Computerchips vorgestellt. Sie ahmen die Fähigkeiten des menschlichen Gehirns in Sachen Wahrnehmung, Erkennung und Reaktion in Grundzügen nach. Dieser Ansatz wird „Cognitive Computing“ genannt. Damit könnte man lernende Computer realisieren, die um ein Vielfaches effizienter und kleiner sind als heutige Systeme.

Die vorgestellten neurosynaptischen Computerchips (Bild: IBM) enthalten übliche Silizium-Schaltkreise sowie Algorithmen, deren Aufbau aus der Neurobiologie stammt. Diese Algorithmen machen ähnliche Abläufe möglich, wie sie zwischen Neuronen und Synapsen im Gehirn auftreten. Ein „neurosynaptischer Kern“ enthält einen integrierten Speicher (analog zu den Synapsen des Gehirns), die Recheneinheit (entsprechend den Neuronen) und einen Kommunikationsblock (der den Axons entspricht).

Zwei erste Prototypen wurden gefertigt und befinden sich derzeit in der Testphase. Die Prozessorkerne wurden in 45-nm-SOI-CMOS-Technologie hergestellt und enthalten 256 Neuronen. Einer der Testchips enthält 262.144 programmierbare Synapsen, der andere 65.536 lernende Synapsen. Die IBM-Forscher zeigten damit Anwendungen wie Navigation, maschinelles Sehen, Mustererkennung oder assoziative Speicherung. Solche Systeme verlassen die klassische Von-Neumann-Architektur, vielmehr verschmilzt die Grenze zwischen Hard- und Software. Dies ermöglicht den Computern, durch Erfahrung selbst zu lernen, Korrelationen zu finden und Hypothesen zu bilden. Auf diese Weise ahmen sie letztlich das menschliche Gehirn nach.

Die Chips wurden im Rahmen der mehrjährigen Forschungsinitiative SyNAPSE entwickelt (Systems of Neuromorphic Adaptive



Plastic Scalable Electronics). Dabei konnten Erkenntnisse aus der Nano- und Neurowissenschaft eingebracht werden. Auch Elemente des Supercomputings wurden verwendet.

IBM und eine Reihe US-amerikanischer Universitäten haben nun für die zweite Phase des SyNAPSE-Projektes Unterstützung in Höhe von 21 Millionen Dollar von der amerikanischen Defense Advanced Research Projects Agency (DARPA) erhalten. Ziel von SyNAPSE ist es, ein Computersystem zu entwickeln, das nicht nur verschiedene sensorische Eingangsdaten gleichzeitig analysiert, sondern sich auch auf Basis seiner Interaktion mit der Umwelt dynamisch rekonfiguriert. Wei-

ter will man dabei ein System mit geringem Energieverbrauch und Volumen realisieren, ähnlich dem menschlichen Gehirn. Dabei stellen sich die IBM-Forscher ein System mit 10 Milliarden Neuronen und 100 Billionen Synapsen vor, das weniger als 1 Kilowatt an elektrischer Energie verbraucht und ein Volumen von weniger als 2 Litern hat.

Der Projektleiter bei IBM in Almaden, Dr. Dharmendra Modha (Bild: IBM) sieht künftige Anwendungen in der Informatik unter einem neuen Aspekt: Zunehmend wird nach Funktionalitäten verlangt, die mit der heutigen Architektur nicht mehr effizient umgesetzt werden können. Die nun vorliegenden Prototypen sieht er als (weiteren) wichtigen Schritt in der Computerentwicklung. Künftige Prozessoren sollen selbständig Informationen erarbeiten, ganz ohne externe Programmierung. Als Beispiel nennt Modha intelligente Verkehrsleitsysteme, die selbst sehen können, Abgase und andere Gerüche analysieren und Geräusche auswerten – und dann selbständig eine Entscheidung treffen, wo man eventuell Straßenabschnitte über Verkehrsampeln sofort sperren sollte (z.B. nach einem Unfall oder Brand).

www.ibm.com/smarterplanet/us/en/business_analytics/article/cognitive_computing.html

Geschwindigkeitsrekord für Elektroautos

Studenten der Brigham Young University im US-Bundesstaat Utah haben mit ihrem Elektro-Auto „Electric Blue“ einen neuen Geschwindigkeitsrekord in der E1-Klasse aufgestellt (Elektrofahrzeuge mit einem Gewicht von weniger als 1.100 Pfund = 498,95 kg). Das Fahrzeug erreichte auf den beiden vorgeschriebenen Messfahrten eine Durchschnittsgeschwindigkeit von 155,8 Meilen pro Stunde (250,74 km/h). Die Rekordfahrten wurden auf dem bekannten Salzsee von Bonneville in Utah durchgeführt (Foto: Brigham Young University).

Die besondere Leistung der studentischen Entwickler war, dass sie das Gewicht des

Fahrzeugs trotz der relativ schweren Lithium-Eisenphosphat-Akkus unter dem Grenzwert der E1-Klasse halten konnten. Zu dem Ergebnis beigetragen haben die Verwendung von Kohlefaser-Verbundwerkstoff und die erzielte aerodynamische Güte,

nicht zuletzt dank der vollständig in die Karosserie integrierten Räder.

Da die Bodenfreiheit nur etwa 2,5 cm (1 Zoll) beträgt und der Wendekreis sehr groß ist, lassen sich Hochgeschwindigkeitsfahrten nur auf einer sehr großen ebenen Fläche wie auf dem Salzsee von Bonneville durchführen. Ein erster Rekordversuch im Jahr 2010 war gescheitert, als sich der Rennwagen bei der zweiten Messfahrt überschlug.

<http://news.byu.edu/archive11-oct-byuelectriccarrecord.aspx>



Updates und Ergänzungen

Die Chaos-Maschine

Elektor 10/2011, S. 74 (110546)

Im Schaltplan des Differenzverstärkers in Bild 2a muss ein 60-k-Gegenkopplungswiderstand zwischen dem Ausgang (Pin 1) und dem Eingang (Pin 2) des Opamps A2.A geschaltet werden. In Bild 2f sind bei IC A8.B die Pin-Nummern 5 und 6 vertauscht. Pin 5 ist der nichtinvertierende Eingang und Pin 6 der invertierende Eingang (die Bezeichnungen „+“ und „-“ der Eingänge sind korrekt angegeben).

USB-Wetterlogger mit Langzeitspeicher

Elektor 09/2011, S. 14 (Platine 100888-1)

In die Stückliste im Heft hat sich ein Tippfehler eingeschlichen. Es muss R5 = 3k3 heißen (wie im Schaltplan angegeben) und nicht 1k3. Die Schaltung funktioniert aber auch mit 1k3 für R5. Die Firmware V3 wurde am 18.10.2011 im Download-Zip-File 100888-1.zip auf der Seite www.elektor.de/100888 durch die Version V4 ersetzt. Der Grund: Bei einer Störung der USB-Verbindung bleibt der Wetterlogger mit der Firmware V3 in einer Schleife stehen und kann nur durch einen Reset neu gestartet werden.

Dieser Fehler wurde in der Firmware V4 beseitigt; mit SWITCH_3 wird die USB-Kommunikation sowohl beim Warten auf ein Kommando vom PC als auch bei der Ausgabe der Messdaten beendet.

Rekord: 61 % Wirkungsgrad bei roter LED

Im Entwicklungslabor von OSRAM wurde mit einem elektro-optischen Wirkungsgrad von 61 % ein neuer Effizienzrekord bei einer roten Hochleistungs-LED erzielt. Der nur 1 mm² große Chip leuchtet bei einer Wellenlänge von 609 nm und schafft bei 40 mA den Rekordwert von 201 lm/W. Bei 350 mA sind es immer noch 168 lm/W, womit auch bei höheren Leistungen mehr als die Hälfte der elektrischen Energie in Licht umgewandelt wird.

LED-Farbmischsysteme aus farbigen LEDs sind im Kommen, gerade wenn es um Allgemeinbeleuchtung geht. Mit ihnen lassen sich alle möglichen Weißvarianten realisieren, von warmweißer über neutralweiße bis hin zu tageslichtähnlicher Lichtfarbe. Dabei wird die Gesamtleistung des Systems von der schlechtesten Einzelkomponente limitiert. OSRAMs neue rote LED verspricht eine weitere Verbesserung der

Lichtqualität mit weniger Energie – speziell für warmweißes Licht. Davon profitieren werden jedoch nicht nur Farbmischkonzepte, sondern auch andere Anwendungen mit roten LEDs – vor allem im Projektions- und Automotive-Bereich.

Höhere Effizienz führt außerdem auch zu weniger Chips für die gleiche Helligkeit, was flexiblere Möglichkeiten für das Design ergibt. Die Lichtquellen können bei großer Helligkeit immer kleiner ausgelegt werden. Die Forschungsergebnisse bei der LED-Entwicklung von OSRAM Opto Semiconductors lassen sich angeblich sogar auf andere Wellenlängen der verwendeten InGaAlP-Chiptechnologie übertragen, so dass mit einem breiten Effizienzschub bei diesen Lichtfarben zu rechnen ist – sogar bei einer Wellenlänge von 660 nm, wie sie beispielsweise in der Pflanzenbeleuchtung nötig ist. Die Ergebnisse dieses Entwicklungsprojekts über den ganzen Wellenlängenbereich werden voraussichtlich in etwa einem Jahr in die Produktion überführt.

www.osram.de/osram_de

Anzeige



**PCBs
Muuuuch Cheaper...**

16.94 EURO*

5 pcbs, 100 mm x 100 mm
*per piece, incl. VAT (21%)
+ shipping costs e. g. Germany 10.71 EURO

No-frills policy



www.jackaltac.com

Der PCB Prototyper in der Praxis

Ideal für Prototypen und Kleinserien

Der PCB Prototyper, den Elektor vor genau einem Jahr vorstellte, fand inzwischen seinen Weg in viele Labors, Werkstätten und Betriebe. Mit dieser genialen, hoch präzisen Fräsmaschine können ohne weitreichende Vorkenntnisse in kürzester Zeit einseitige oder doppelseitige Platinen angefertigt werden. Wir haben uns bei einem überzeugten Anwender umgeschaut.

Von Harry Baggen (Redaktion NL)

Nach der Vorstellung des PCB Prototyper im Dezember 2010 erreichten uns schon bald die ersten Bestellungen. Bei unserem Vertragspartner Colinbus war die Produktion bereits angelaufen, so dass die ersten Maschinen zügig ausgeliefert werden konnten. Die Rückmeldungen der neuen Besitzer ließen nicht lange auf sich warten: Die Maschine bekam von allen Seiten viel Lob und Anerkennung. Hoch bewertet wurde auch das PreisLeistungsverhältnis, viele Käufer hatten nicht erwartet, ein so präzises und anwenderfreundliches Produkt zum Sparpreis von 3500 € (+ MwSt.) zu bekommen. Vergleichbare Platinen-Fräsmaschinen sind regelmäßig wesentlich teurer.

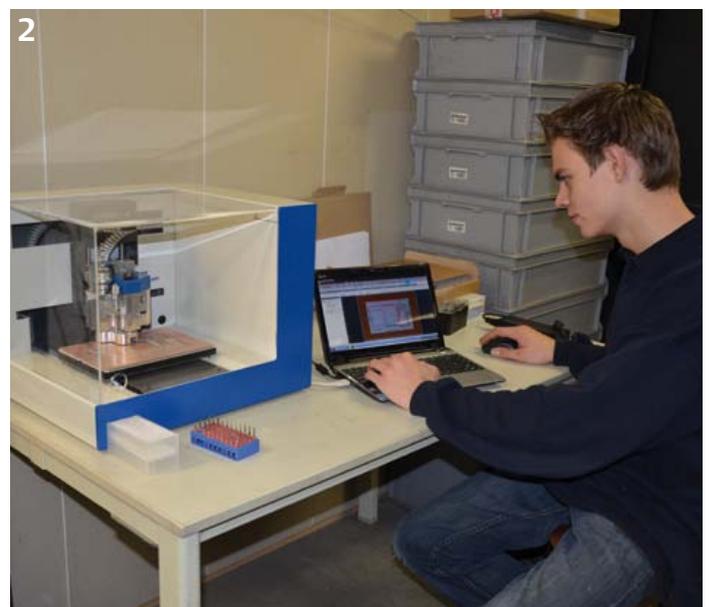
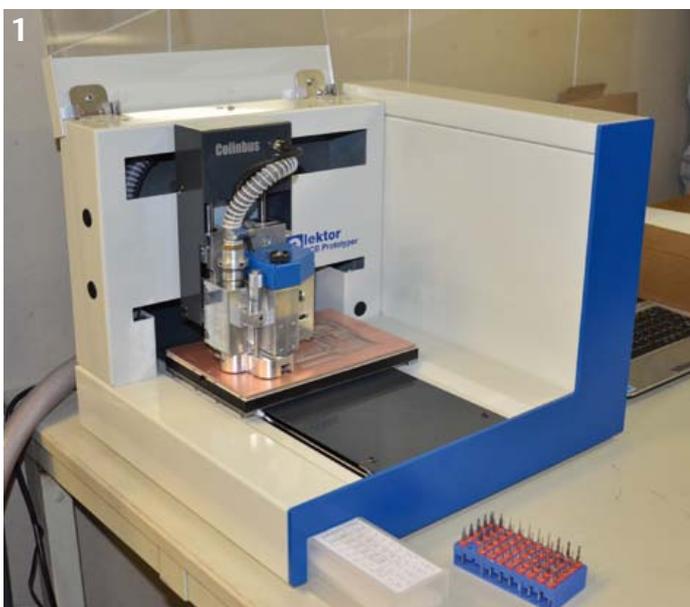
Avasto ist der Name eines mittelständigen Unternehmens, das sich mit Konstruktionen im Metallbau, mit der Automatisierung von Fertigungsprozessen und mit dem Bauen und Warten von Wellness-Anlagen, bei-

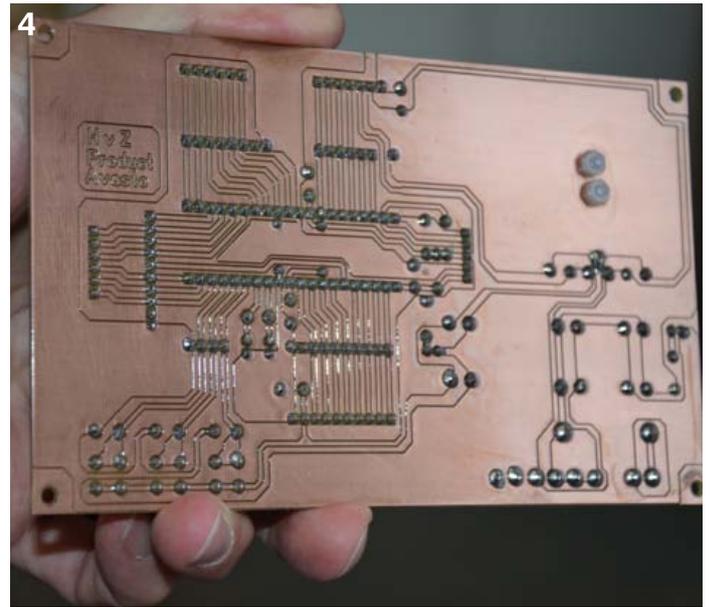
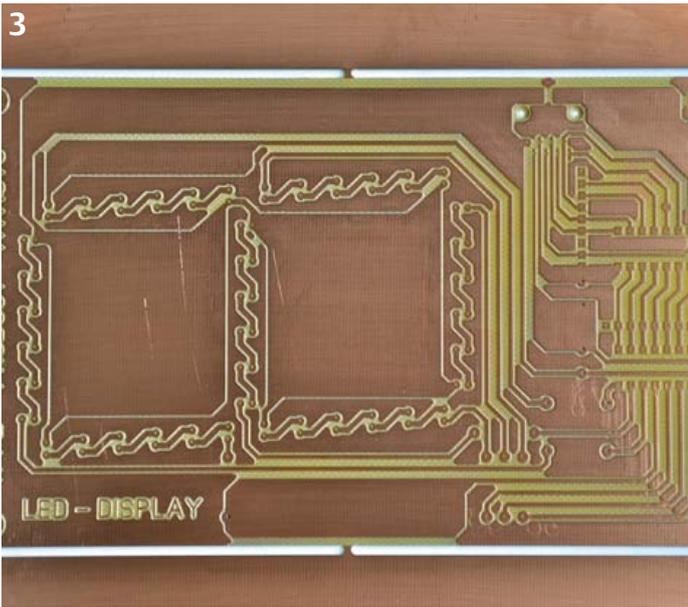
spielsweise Saunen und Schwimmbädern beschäftigt. Elektronische Steuerungen, oft aufgebaut mit speicherprogrammierbarer Logik (PLCs), sind immer häufiger dabei. Das jüngste Produkt ist ein Sicherheitssystem für Wasserrutschen in Schwimmbädern, das Kollisionen im Badebetrieb vorbeugt. Der wartende Badegast erhält ein Signal, wenn sein Vorgänger die Rutsche verlässt und die Rutsche frei ist. Zu Beginn der Rutschpartie kann der Badegast eine Stoppuhr starten, angekommen im Schwimmbecken kann er die Stoppuhr anhalten. Die letzten von rutschbegeisterten Schwimmern erreichten Zeiten sind für das Publikum auf einer Anzeigetafel sichtbar. Das System erhöht nicht nur die Sicherheit im Badebetrieb, es bringt auch ein Wettkampf-Element unter die Schwimmer-Gemeinde.

Avasto hat dieses Sicherheitssystem vollständig im eigenen Haus entwickelt, es erhielt ein offizielles Label für geprüfte Sicherheit. Inzwischen wurden bereits mehrere Einheiten ausgeliefert und installiert.

Miteigentümer Swen van Vrouwerff ist Ingenieur aus Leidenschaft, er ist mit allen Vorgängen in seinem Betrieb vertraut, egal ob es um den Bereich der Mechanik oder Elektronik geht. Anders als in früheren Jahren entwickelt Avasto heute die meisten elektronischen Baugruppen in eigener Regie. Als der PCB Prototyper in Elektor vorgestellt wurde, war sich Swen van Vrouwerff sicher, dass dies die richtige Maschine ist, um mit wenig Zeitaufwand Platinen für Prototypen und Kleinserien zu fertigen. Zwar bieten auch spezialisierte Dienstleister das Anfertigen gefräster oder geätzter Platinen an, doch der Aufwand an Zeit und Geld ist deutlich höher. Swen van Vrouwerff hat scharf kalkuliert, er ist inzwischen überzeugt, dass sich die Investition in den PCB Prototyper lohnt hat.

Avasto setzt den PCB Prototyper zurzeit fast ausschließlich zum Fertigen von Platinen ein, die Bestandteile der erwähnten Sicherheitssysteme für Wasserrutschen sind. Die Elektronik ist im Gehäuse der





Anzeige untergebracht, sie besteht aus insgesamt neun Platinen. Der PCB Prototyper (1) wird bei der Platinen-Fertigung von einem Netbook gesteuert, das neben der Maschine steht (2). Mit der Fertigung der Wasserrutschen-Sicherheitssysteme wurden zwei junge Mitarbeiter betraut, auch sie sprachen sich anerkennend über den PCB Prototyper aus. Insbesondere wurden die unkomplizierte Bedienung und die hohe Qualität der Platinen gelobt. Die Mitarbeiter berichteten, dass das Vertrautwerden mit der Maschine nach dem Aufstellen einschließlich der ersten Probeläufe nur einen Arbeitstag in Anspruch genommen hatte. Das Fertigen einer Platine im Europa-Format dauert ungefähr eine halbe Stunde. Da

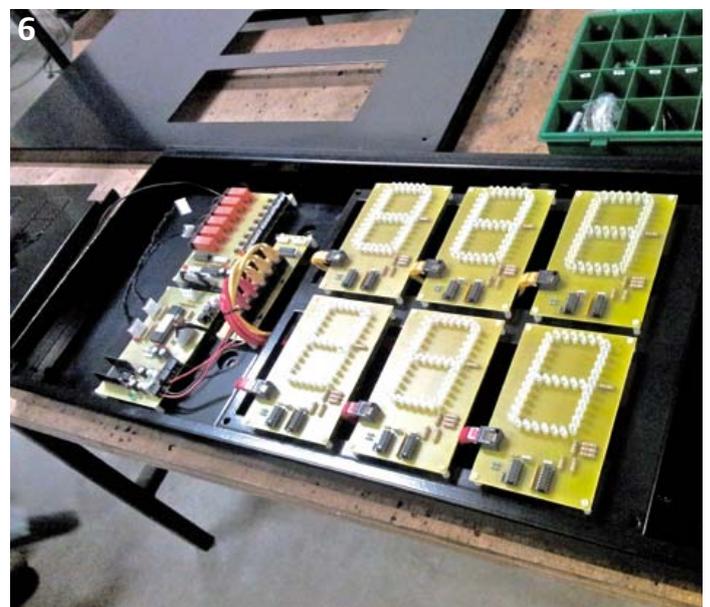
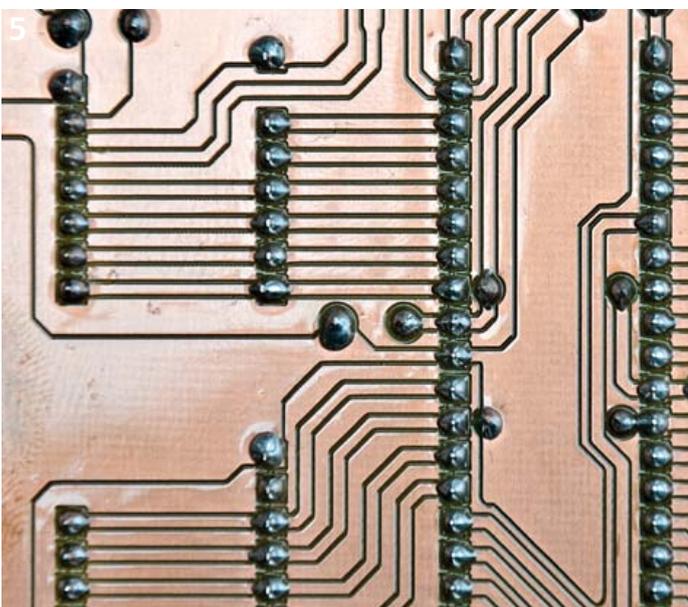
der PCB Prototyper zum Auswechseln von Bohrern oder Fräseinsätzen anhält, muss er nicht beaufsichtigt werden. Der Mitarbeiter kann die Zwischenzeit für andere Tätigkeiten nutzen. Das nächste Foto (3) zeigt eine gefräste Platine, hier für ein Display-Segment. Nach dem Fräsen werden die Kupferseiten mit dicken Lagen Schutzlack überzogen. Der Schutzlack hält die in Schwimmbädern unvermeidlichen Chlordämpfe vom Kupfer fern (4). Den Qualitätsstandard der gefrästen Platinen führt noch einmal das nächste Foto (5) vor Augen, während das letzte Foto (6) den Innenaufbau des Schwimmbad-Sicherheitssystems mit montierten Platinen zeigt. Die Schwimmbad-Sicherheitstechnik ist nur

ein Beispiel für den Einsatz des PC Prototyper bei Avasto. Gegenwärtig werden dort nur einseitige Platinen für bedrahtete Bauelemente gefertigt. Der Umstieg auf SMDs steht bevor, auch dafür ist der PC Prototyper gut gerüstet. Wir haben für die nahe Zukunft verschiedene Erweiterungen für den PC Prototyper in Planung, sie werden die Vielseitigkeit dieser Maschine weiter steigern.

(110694)gd

Weblinks

- [www.elektor.de/projekte/pcb-prototyper-\(100619\).1594426.lynkx](http://www.elektor.de/projekte/pcb-prototyper-(100619).1594426.lynkx)
- www.avasto.nl



Android als Entwicklungsumgebung

Preisgünstige Tablets in der Embedded Elektronik



Von Elbert Jan van Veldhuizen (NL)

Die im Preis drastisch gesunkenen Tablet-Computer sind hoch entwickelte elektronische Geräte, die zu mehr taugen als nur zu ihrem eigentlichen Bestimmungszweck. Haben Sie schon einmal an den Einsatz von Tablet-Computern in der Embedded-Elektronik gedacht?

Tablets mit dem Betriebssystem Android, die unter 100 € kosten, sind heute keine Seltenheit mehr. Die Tablets sind nicht nur mit viel Elektronik vollgestopft, sie sehen darüber hinaus auch noch attraktiv aus. Viele der Funktionalitäten, die Tablets bieten, lassen sich oft nur mit hohem Aufwand in eigene Entwicklungen implementieren. Schon das ist Grund genug, den Einsatz von Tablets in der Embedded-Elektronik zu erwägen.

Hardware

Auch Tablets unter 100 € sind zeitgemäß gestylte Produkte, die sich nicht verstecken müssen. Die Idee liegt nahe, sie an den Fronten eigener elektronischer Geräte zu montieren und zu professionell gestalteten Bedienfeldern umzufunktionieren. Tablets sind mit beleuchteten Displays ausgestattet, die gleichzeitig Touchpad-Funktion haben. Eine solche Außenfront bietet sich zum Darstellen grafischer und alphanumerischer Inhalte an. Die Touchpad-Funktion kann genutzt werden, um virtuelle Bedienelemente zu realisieren, die auch kontextbezogen sein können. Natürlich sind auch Menüstrukturen und Eingabefelder realisierbar, die Möglichkeiten sind vielfältig. Für Multimedia-Anwendungen sind Tablets gut ausgestattet, auf ihren Displays lassen sich Aufnahmen ferner IP-Kameras ebenso wie Bedienhilfen in

Form von Videos darstellen. Die Displays der unteren Preiskategorie haben Diagonalen von 18 cm bis 25 cm (7" bis 10"), die Auflösungen liegen zwischen 800 · 480 und 1024 · 600 Pixel. Die sensorische Oberfläche ist aus Kostengründen meistens vom resistiven Typ, sie kann gleichzeitig nur einen Berührungspunkt erfassen. Beim Einsatz in Embedded-Systemen ist dies aber nur selten ein Handicap.

Vielleicht ist jetzt der Eindruck entstanden, dass Tablets lediglich so genannte intelligente Terminals ersetzen können. Das stimmt natürlich nicht, ganz im Gegenteil. Sogar in den preiswertesten Tablets steckt mindestens ein ARM11-Mikrocontroller mit über 600 MIPS Rechenleistung. Verglichen mit Mikrocontrollern, die in aktuellen Entwicklungen eingesetzt werden, liegt die Rechenleistung zehn bis hundert Mal höher. Für Anwendungen, die an die Rechenleistung hohe Anforderungen stellen, sind diese Tablets gut gerüstet. Prozesse in Signalfiltern wie beispielsweise Fourier-Transformationen, die üblicherweise von DSPs abgearbeitet werden, sind für Tablet-Mikrocontroller ein leichtes Spiel. Allerdings ist Android kein Echtzeit-Betriebssystem, so dass Android-Tablets für Echtzeit-Anwendungen nur bedingt einsatzfähig sind. Unter Android sind Echtzeit-Anwendungen nur mit zeitlichen Toleranzen im Sekunden-Bereich möglich

(Near Real Time).

In der Juni-Ausgabe dieses Jahres hatte Elektor beschrieben, wie Apps für Android entwickelt werden können [1]. Android ist multitaskingfähig, so dass mehrere Apps mit unterschiedlichen Funktionen parallel laufen können.

Eine weitere wichtige Funktionalität jedes Tablets ist seine Anbindung an das Internet über eine integrierte WLAN-Schnittstelle. Zu den Standard-Internet-Apps für Android gehören ein Webserver, ein FTP-Server sowie diverse Email-Programme. Auch diese Apps sind parallel im Hintergrund lauffähig. Das Internet macht den Datenaustausch über globale Entfernungen möglich. Systeme, die am anderen Ende der Welt installiert sind, können fernüberwacht und ferngesteuert werden. Tablets der gehobenen Preisklassen verfügen über 3G-Mobilfunk-Schnittstellen, hier ist die Applikation nicht mehr an die lokalen Reichweiten der WLAN-Router oder Access-Points gebunden. Die 3G-Anbindung macht den netzweiten, ortsunabhängigen Einsatz möglich.

In Tablets sind immer nicht flüchtige Speicher (Flash) verbaut, meistens lassen sie sich durch externe Speicherkarten erweitern. Die Kapazitäten von Micro-SD-Karten haben inzwischen die 32-GB-Marke überschritten. Ein typischer Einsatz eines solchen hoch kapazitiven Speichers ist das



Sammeln von Daten mit einer Datenlogger-App. Die Kapazität 32 GB reicht aus, um ein Jahr lang einen Datenstrom zu speichern, der aus 1000 Werten in der Sekunde besteht. Auch Multimedia-Anwendungen machen gern von hohen Speicherkapazitäten Gebrauch. In den meisten Tablets sind Lautsprecher sowie Mikrofone und Webcams integriert und auch externe Anschlüsse vorhanden. Ferner stellen Tablets über ihre USB-Anschlüsse die stabilisierte Spannung 5 V zur Verfügung. Auch diese Beigabe kann oft nützlich sein.

Zu einer anderen Gerätegruppe gehören solche Smartphones, die ebenfalls mit dem Betriebssystem Android arbeiten, sie werden schon zu Preisen unter 150 € angeboten. Verglichen mit den Tablets sind die Displays zwar bedeutend kleiner, doch die 3G-Mobilfunkanbindung ist immer implementiert. Wenn diese Eigenschaft wichtiger als die Display-Größe ist, können Android-Smartphones eine interessante Alternative sein. Ferner hat jedes Smartphone eine Bluetooth-Schnittstelle, die zur drahtlosen Kommunikation mit der nahen Umwelt dient. Das Projekt „Amarino“ [2] nutzt Bluetooth, um Arduino-Boards und Android-Smartphones miteinander zu koppeln.

USB

Mindestens ein USB-Port gehört zur Ausstattung jedes Tablets. Das Anbinden externer Elektronik über diesen Weg ist theoretisch unkompliziert, doch in der Praxis lauern einige Fallstricke. Zum Teil liegt dies daran, dass Android-Tablets auf der Grundlage von Android-Smartphones entwickelt wurden. Die USB-Ports der Smartphones verhalten sich wie Slaves, damit sie beispielsweise mit PCs kommunizieren können. Auch die USB-Ports von Android-Tablets haben dieses Verhalten, was bedeutet, dass die angeschlossene Elektronik die Rolle des Master übernehmen muss. Die dazu notwendigen USB-Controller sind komplexer aufgebaut als ihre Gegenstücke. Viele Tablets können ihren USB-Port in den Master-Modus umschalten, was ein spezielles USB-Kabel notwendig macht,



Bild 1. USB-Stecker für den Betrieb des Tablets im Master-Modus.

oder sie besitzen einen zweiten USB-Port, der als Master arbeitet. Der zweite USB-Port ist zum Anschließen von Tastaturen, Mäusen oder Speicher-Sticks vorgesehen, und auch externe Elektronik mit Slave-Verhalten kann hier angeschlossen werden. Microchip bietet Software für die Kommunikation mit Android-Systemen an, die als Slave konfiguriert sind. Ferner hat Microchip spezielle Entwickelboards im Programm [3]. Im nächsten Heft wird auch Elektor eine Platine vorstellen, welche auf einem Vinculum-Chip von FTDI basiert, der als USB-Master fungiert. Die Anbindung eigener Elektronik an Android wird damit sehr einfach!

In Android gibt es aber noch keine standardisierte Schnittstelle für USB-Anwendungen. In den zurzeit aktuellen Versionen Android 2.3.4 und 3.1 stecken zwar mehr Möglichkeiten, doch sie sind nicht ohne Weiteres nutzbar. Zum Einbinden geeigneter Steuerprogramme muss eine angepasste Android-Version installiert werden. Dafür wird Software benötigt, das so genannte „ROM“, sowie der „Root-Access“. Für preisgünstige Tablets mit Standard-Android-Installation existieren Apps, die den „Root-Access“ realisieren können. Das Erstellen eines „ROM“ ist jedoch eine Aufgabe für Experten, hierbei müssen notwendigerweise alle Steuerprogramme vorhanden sein. Für ein einzelnes Projekt bedeutet dies einen unverhältnismäßig hohen Aufwand an Arbeit und Zeit.

Eine Lösung dieses Problems ist in Sicht, denn Google, die Schmiede von Android, arbeitet mit Hochdruck an der Version 4, Codename „Ice Cream Sandwich“. In diese Version fließen die Version 2 (für Mobiltelefone und Low-Cost-Tablets) und die Version 3 (für Tablets höherer Preisklassen) ein. Zurzeit schafft Google

die Voraussetzungen dafür, dass möglichst viele Systeme mit Android gekoppelt werden können. Dazu wurde die Software-Class „accessory“ eingeführt [4], deren Funktionalität zwar schon in den Versionen 2.3.4 und 3.1 implementiert war, bisher fehlte jedoch die Unterstützung im „ROM“. Durch diese Erweiterung würde sich der USB-Port als Master verhalten, so dass externe Systeme mit einfachen USB-Controllern auskommen. Zukünftig wird vermutlich das „ROM“ die Unterstützung übernehmen, so dass Upgrades nicht mehr notwendig sind.

Die Version 4 des Betriebssystems Android soll Ende 2011 erscheinen. Es ist abzusehen, dass die System-Anforderungen die Eigenschaften heutiger Low-Cost-Tablets übersteigen. Bevor Low-Cost-Tablets mit dieser Android-Version verfügbar sind, wird noch einige Zeit vergehen.

Architektur

Wenn ein Tablet in ein Projekt integriert wird, ist zu entscheiden, an welchem Ort (Tablet oder externe Elektronik) die „Intelligenz“ angesiedelt werden soll. Das Tablet hat den Vorteil der übersichtlichen Programmierumgebung mit einer API, die unmittelbaren Zugriff auf die Tablet-Systemkomponenten erlaubt. Weiter ist vorteilhaft, dass das Tablet über hohe Rechenleistung und viel Speicher verfügt. Doch wie schon erwähnt, sind Echtzeit-Anwendungen unter Android nicht möglich. Nach Verlagern der „Intelligenz“ in die externe Hardware besteht diese Einschränkung nicht mehr, hier gibt es keine Barriere für Echtzeit- oder High-Speed-Anwendungen. In der Praxis wird der Entwickler zu mehr oder weniger hybriden Lösungen neigen. Die Funktionalitäten, die nahen Bezug zur Tablet-Peripherie haben, werden dort implementiert, während die



Bild 2. Die API des Betriebssystems Android ist umfassend dokumentiert.

externe Elektronik das Steuern der Ein- und Ausgabekanäle übernimmt.

Applikationen

Die nachfolgend beispielhaft genannten Applikationen sind eine keinesfalls repräsentative Auswahl aus der Fülle der Möglichkeiten:

- **Datenlogger:** Einstellungen werden am Tablet vorgenommen, Daten können grafisch aufbereitet auf dem Display oder einer Webseite dargestellt werden. Die Messdaten sind über das Internet mit FTP abrufbar. Zur externen Hardware müssen ein ADC und ein USB-Port gehö-



Bild 3. Einsatz eines Tablet-Computers in der Domotik.

ren. Auf der Software-Seite werden drei Apps benötigt: Webserver und FTP-Server (verfügbar als Standard-Apps) sowie eine individuell zu schreibende App, die Daten erfasst und speichert, die Ausgabe für das Display aufbereitet und die Einstellungen der virtuellen Bedienelemente umsetzt.

- **Telemetrie:** Aktuelle Daten von Fahrzeugen, beispielsweise bei Motorsportveranstaltungen, werden in Echtzeit über eine 3G-Mobilfunkverbindung übertragen und auf dem Tablet-Display dargestellt. In den Fahrzeugen müssen OBD2-nach-USB-Umsetzer installiert sein, die Tablet-Software muss die empfangenen Daten aufbereiten, speichern und darstellen. Für das sporadische Übertragen einzelner Werte genügt eine Standard-FTP-App, während Echtzeit- oder Quasi-Echtzeit-Betrieb das Schreiben einer individuellen App notwendig macht. Diese App kann sich an Standard-Apps orientieren, die für die serielle Datenkommunikation bestimmt sind.
- **Domotik:** Eine Heizungsanlage mit Wärmekollektoren auf dem Hausdach wird nach den Prognosen von Internet-Wetterdiensten über eine Webseite und ein Tablet ferngesteuert. Die zu erwartenden Außentemperaturen und die prognostizierte Sonnenscheindauer sind zwei relevante Größen. Die Informationen kann eine individuell zu schreibende App beispielsweise aus RSS-Feeds extrahieren.
- **Digitales Oszilloskop:** Einstellungen werden über virtuelle Bedienelemente auf dem Tablet-Display vorgenommen, das gleichzeitig zur grafischen Ausgabe der Messergebnisse dient. Auch hier ist die bereits beim Datenlogger genannte Hardware-Konfiguration notwendig. Wegen der nötigen Signalgeschwindigkeiten muss die externe Elektronik den größten Teil der „Intelligenz“ übernehmen.
- **Lichtshow-Steuerung:** Der Tablet-PC, auf dem eine Webserver-App läuft, dient als fernsteuerndes Bedienpult. Auch hier hat die externe Elektronik wegen der hohen Anzahl der (gemultiplexten) Kanäle und der gewünschten Reaktionsgeschwindigkeiten einen relativ hohen Umfang.
- **Versorgungseinrichtungen:** Es ist keine Utopie, dass beispielsweise Getränkeautomaten E-Mails an einen zentralen In-House-Service schicken, wenn ihr

Vorrat zur Neige geht oder ein Störfall auftritt. Während externe Elektronik die Verbindungen zum Gerät herstellt, kann das Tablet die Daten speichern und weiterleiten.

- **Vor-Ort-Service:** Tablets bieten ideale Voraussetzungen, um als Konfigurationswerkzeuge, zur Störungsdiagnose oder für Wartungsarbeiten vor Ort eingesetzt zu werden. In der einfachen Variante kommuniziert das Tablet, auf dem eine objektbezogene App läuft, über USB unmittelbar mit dem Zielsystem. Nur wenig aufwendiger ist eine Verbindung über einen USB-nach-JTAG-Wandler. Nicht zuletzt wegen der niedrigen Kosten könnten Service-Tablets bald zum selbstverständlichen Zubehör jeder Einrichtung oder Anlage gehören.
- **Prototyping:** Da das Schreiben einer App nur relativ wenig Zeit in Anspruch nimmt, können Tablets das Entwickeln und Testen von Prototypen beschleunigen. Schon die leistungsstarken Standard-Funktionen und die Vielzahl der für das Betriebssystem Android verfügbaren Apps machen die Realisierung umfangreicher Funktionalität innerhalb kurzer Zeit möglich.

Fazit

Android-Tablets der unteren Preisklassen sind als effiziente, schnell implementierbare und gleichzeitig kostensenkende Systemkomponenten einsetzbar. Für den Entwickler sind die weit reichenden Funktionalitäten leicht zugänglich, das äußere Design hat professionellen Charakter. Zurzeit ist das Anbinden über USB noch eine Sache von Spezial-Hardware, doch das wird sich ändern, wenn im kommenden Jahr die Android-Version 4 verfügbar ist.

(110667)gd

Weblinks

- [1] www.elektor.de/110265
- [2] www.amarino-toolkit.net
- [3] www.microchip.com/android
- [4] <http://developer.android.com/guide/topics/usb/accessory.html>

HUEY JANN-High-Power-Module für industrielle Applikationen



High-Power - LED-Modul, 10 Watt

- auf ausreichend Wärmeabfuhr achten!
- Abstrahlwinkel: 120°
- Maße: 20 x 20 x 3,15 mm



13,65

	Farbtemperatur	UF / IF (V / A)	Lumen
LED H10WG CWS	8000 K, cool-weiß	14 / 700	820
LED H10WG PWS	6000 K, pur-weiß	14 / 700	820
LED H10WG NWS	4000 K, natur-weiß	14 / 700	796
LED H10WG WWS	3000 K, warm-weiß	14 / 700	728

13,95
15,55
16,70

High-Power-LED-Modul, 20 Watt:

	Farbtemperatur	UF / IF (V / A)	Lumen
LED H20WG CWS	8000 K, cool-weiß	14 / 1400	1500
LED H20WG PWS	6000 K, pur-weiß	14 / 1400	1500
LED H20WG NWS	6000 K, natur-weiß	14 / 1400	1500
LED H20WG WWS	3300 K, warm-weiß	14 / 1400	1348

25,95

25,95
25,20
30,65

Markenqualität - Top-Service - günstige Preise!

Katalog kostenlos!
 Tagesaktuelle Preise:
www.reichelt.de

DOG LCD-Module

Stellen Sie sich Ihr Wunsch-Display zusammen!

- Anzeige und Beleuchtung werden separat bestellt, somit ergeben sich vielfältige Kombinationsmöglichkeiten
- Display und Beleuchtung werden einfach zusammengesteckt
- Maße: 55 x 27 mm, superflach: 2,0 mm (5,8 mm, inkl. Beleuchtung)
- Controller ST7036 für 4-/8-Bit, 4-Draht SPI



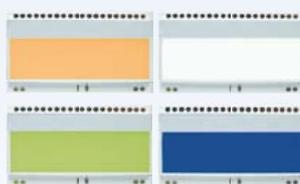
EA DOGM162B-A	Schriftfarbe	Hintergrund	empfohlene Beleuchtung
EA DOGM162E-A	wie Beleuchtung	blau	weiß oder gelbgrün
EA DOGM162L-A	schwarz	wie Beleuchtung	gelbgrün
EA DOGM162S-A	wie Beleuchtung	schwarz	nicht beleuchtbar
EA DOGM162W-A	schwarz	wie Beleuchtung	weiß
			weiß oder blau

11,85
11,85
11,85
11,85
11,85

LCD-Hintergrundbeleuchtung für DOGM

- Maße: 55 x 31 mm

EA LED55X31-A	Farbe	3,15
EA LED55X31-B	amber	8,70
EA LED55X31-G	blau	3,15
EA LED55X31-W	gelbgrün	10,35
	weiß	



Beleuchtete Folientastaturen mit Klebebeschichtung



Universell einsetzbare Folientastatur für verschiedene Anwendungsbereiche.

- Rastermaß: 2,54 mm
- Temperatur: max. -30 bis +55 °C
- bedruckt

FTAC 3559 B	Matrix	Maße (HxB)	12,95
FTAC 3560 B	1 x 4	95 x 38 mm	14,25
FTAC 3561 B	3 x 4	95 x 76 mm	14,50
	4 x 4	95 x 95 mm	



High-Power 5 W LED-Emitter/-Modul

- Auf ausreichend Wärmeabfuhr achten!
- Emitter-Maße: Ø 7,25 x 5,4 mm

Emitter	Farbe	UF / IF (V / mA)	Lumen	Winkel
LED H5WEF BL	465 nm, blau	3,8 / 1400	96	120
LED H5WEF PWS	6000 K, pur-weiß	3,8 / 1400	262	120
LED H5WEF WWS	3300 K, warm-weiß	3,8 / 1400	232	120

5,15
5,70
5,95

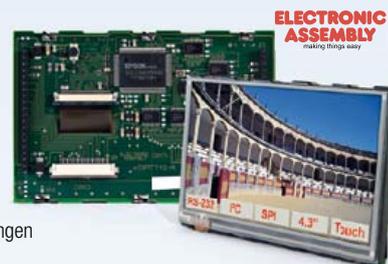
- Modul-Maße: Ø 20 x 19 x 7,1 mm

LED-Modul	Farbe	UF / IF (V / mA)	Lumen	Winkel
LED H5WMF BL	465 nm, blau	3,8 / 1400	96	120
LED H5WMF PWS	6000 K, pur-weiß	3,8 / 1400	262	120
LED H5WMF WWS	3300 K, warm-weiß	3,8 / 1400	232	120

5,65
6,20
6,50

TFT-Grafikdisplay mit Touch-Funktion

Diese intelligenten Displays sind durch das integrierte Micro-Controllersystem samt Touchpanel sowie die integrierten Schriften und Grafikfunktionen sofort betriebsbereit. Über die farbige Darstellung lassen sich wichtige Prozessparameter oder Grenzwertüberschreitungen hervorheben.



- 10,9-cm-TFT-Display (4,3") EA eDPTFT43-ATP
- USB-Programmierboard EA 97777-USB
- CD mit Treibern, Editor, Compiler

Starter-Kit
 sofort lauffähig
 embedded
 intelligent

EA START TFT43 **239,95**

Die neuen OLED-Displays - Displays der Superlative

- voller Kontrast von 2000:1 bis hinunter zu -40 °C
- 5-V- und 3,3-V-Betrieb, ohne Anpassung
- extrem schnell mit 10µs Response Time
- keine Einschränkung des Blickwinkels

	Displayformat	Bildgröße	Farbe
EA W082-XLG	2 x 8	58 x 32 mm	gelb
EA W162-X3LG	2 x 16	80 x 36 mm	gelb
EA W162-X3LW	2 x 16	80 x 36 mm	weiß
EA W162-XBLG	2 x 16	122 x 44 mm	gelb
EA W202-XLG	2 x 20	116 x 37 mm	gelb
EA W204-XLG	4 x 20	98 x 60 mm	gelb

21,95
29,95
30,95
52,95
31,95
47,95



Zeitraffer mit Android-Tablet

Servo steuert Auslöser

Von Elbert Jan von Veldhuizen (NL)

Mit Hilfe eines Android-Tablets und etwas Hardware kann man eine Steuerung für eine Kamera realisieren, um Einzelbildreihen mit einstellbaren Intervallen aufzunehmen. Außerdem ist es so auch möglich, die Einstellungen für die Kamera via Web-Browser von jedem PC aus per WLAN an das Tablet zu übermitteln.

Dieser Beitrag beschreibt ein Verfahren, das eine Kamera zu festgelegten Zeiten Bilder schießen lässt. Anschließend kann man diese Bilder zu einem Film kombinieren, bei dem die Geschehnisse im Zeitraffer-tempo ablaufen, so dass man Stunden oder gar Tage in Sekunden wiedergeben kann. In einer früheren Elektor-Ausgabe wurde etwas Ähnliches schon für eine Kamera mit Eingang für einen externen Auslöser beschrieben. Die hier vorgestellte Lösung ist hingegen universeller, da der Auslöser mechanisch mit Hilfe eines Servos betätigt wird. Eine weitere Besonderheit ist, dass hier ein preiswertes Android-Tablet für die Bedienung über das Internet mit passender GUI genutzt wird. Außerdem

erlaubt dieser Artikel einen Einblick in die Programmierung von Android-Tablets für Embedded-Anwendungen.

Entwurf und Bedienung

Dank der mechanischen Auslösung der Kamera mit Hilfe eines Servos eignet sich die Steuerung für praktisch jede Kamera - von der einfachsten Kompaktknipse bis hin zur professionellen Vollformatkamera. Ob digital oder analog ist dabei ebenfalls egal. Der Autor hat zuerst versucht, die Anschlüsse des Auslösers bei einer Kompaktkamera nach außen zu führen, doch die zunehmende Miniaturisierung macht so ein Vorhaben so gut wie unmöglich, will man

dabei

nicht die Kamera zerstören.

Wie in **Bild 1** zu sehen, zieht der Servo an einem Hebel, der über einen Stift den Auslöser der Kamera betätigt. Es sind natürlich auch andere mechanische Lösungen denkbar.

Wie in einem anderen Artikel über Android-Tablets in dieser Elektor-Ausgabe erläutert, ist es zurzeit nicht so einfach, den USB-



Bild 1. Prototyp mit zugehöriger Mechanik.

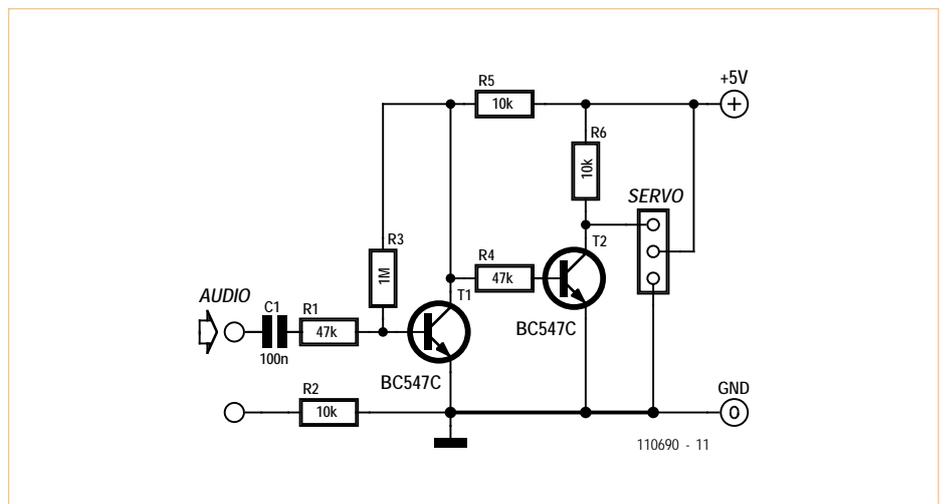


Bild 2. Mit dieser Schaltung wird das Audio-Signal auf 5-V-TTL-Pegel verstärkt. Zur Stromversorgung reicht ein kleines 5-V-Steckernetzteil.

Anschluss des Tablets zu verwenden (eine Lösung mit eigener Platine stellen wir aber im nächsten Heft vor). Aus diesem Grund wird hier der Audio-Ausgang zur Ansteuerung des Servos zweckentfremdet. Die

Position eines Servos wird nämlich durch Impulse bestimmt, die zwischen 1 und 2 ms breit sind. Bei manchen Servos liegen die Zeiten zwischen 0,5 und 1,75 ms. Die Pulsfrequenz beträgt 50 Hz. Der Audio-Controller eines Tablets ist direkt zur Erzeugung solcher Impulse in der Lage. Die Amplitude der

Impulse am Audio-Ausgang beträgt allerdings nur etwa 1 V. Man muss sie also erst noch auf TTL-Niveau bringen. Diese Aufgabe erledigt die Schaltung in **Bild 2**. T1 und T2 verstärken das Signal. Zwecks Erzeugung sauberer Pegel werden Gleichspannungsreste mit C1 abgetrennt. Die Bauteilwerte sind nicht kritisch.

Da an der Software nichts angepasst werden muss, wird „timeLapse.apk“ (im „bin“-Verzeichnis der ZIP-Datei) zur Installation via USB oder Speicher-Karte ins Tablet übertragen. Der Transport via E-Mail funktioniert nämlich nicht. Zur eigentlichen Installation muss man das mit dem Tablet gelieferte Installationsprogramm verwenden. Unter „settings/application settings/unknown sources“ muss dann „allow installation of non-market applications“ eingestellt werden. Die Bedienung kann über zwei Wege erfolgen: mit GUI ausgestattete Software per Tablet (siehe **Bild 3**) und extern via Web-Interface (siehe **Bild 4**). Die Oberfläche der Software enthält zwei Eingabefelder für die Anzahl der Fotos und der Intervallzeit (in Sekunden, Minimum = 3 s). Mit den drei Schieberegler kann man drei Servo-Positionen einstellen. Position 1 entspricht der Ruhstellung der Kamera. Position 2 drückt den Auslöser halb für die Scharfstellung (Autofokus). Position 3 macht dann das Bild. Vor der tatsächlichen Inbetriebnahme

sollte man diese Slider auf Werte zwischen 0,5 und 2 ms einstellen, da einige Servos beschädigt werden können, wenn man sie länger mit ungeeigneten Impulsdauern ansteuert.

Die Schaltfläche „phase“ dient zur Umschaltung von positiven auf negative Impulse. Abhängig von der Tablet-Hardware kann es nämlich notwendig sein, den Impuls zu invertieren. Dann gibt es noch den Start-Knopf, mit dem man die Zeitraffer-Sequenz starten und stoppen kann.

Auch über das Internet kann die Kamera gesteuert werden. Die Software auf dem Tablet lauscht auf Port 8090 auf Anweisungen. Mit dem Browser eines Computers muss man nun die IP-Adresse des Tablets samt Port eingeben. Je nach heimischem LAN könnte diese Adresse so lauten: „http://192.168.1.101:8090/“. Dann erscheint der aktuelle Status der Applikation im Browser-Fenster. Auf dieser Seite können dann auch neue Werte eingegeben und die Zeitraffer-Sequenz gestartet und gestoppt werden. Das links oben erscheinende Logo wird aus der Datei „logo.jpg“ vom Pfad „/sdcard/webserver_TL“ des Tablets eingelesen und kann vom Anwender nach Belieben verändert werden. Dieses Verzeichnis wird automatisch nach dem ersten Start der Applikation generiert.

App-Programmierung

In der Juni-Ausgabe von Elektor wurden die Grundlagen der Programmierung für Android beschrieben. Der Autor hat dort ausgiebig von Eclipse [1] mit ADT (Android Development Toolkit) [2] Gebrauch gemacht.

In Eclipse kann man eine GUI grafisch generieren (siehe **Bild 5**). Die Knöpfe und Eingabefelder etc. können aus einem Menü auf den virtuellen Tablet-Bildschirm gezogen werden. Als Ergebnis erhält man die Datei „main.xml“, die die Beschreibung aller Elemente inklusive eindeutiger IDs enthält.

Die eigentliche Software steckt in der durch Eclipse generierten Datei „TimeLapseActivity.java“ unter der Klasse „TimeLapseActivity“. Diese Klasse enthält die Interaktion mit den Bedienelementen. Um auf ein Element zuzugreifen, muss das Objekt mit einer zuvor erzeugten ID gekoppelt werden.

Ein Beispiel:

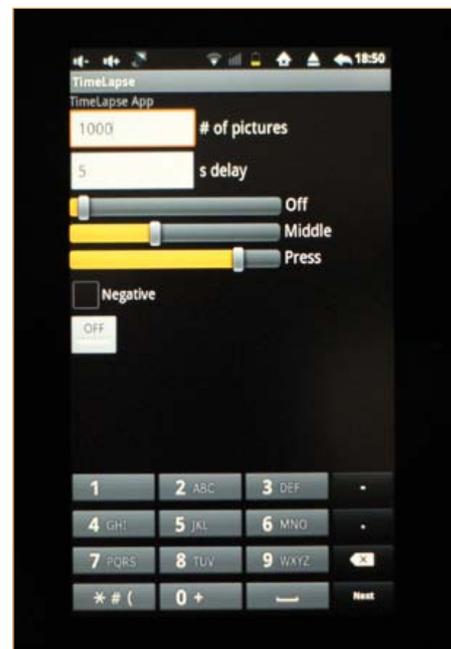


Bild 3. Die GUI der Kamera-Steuerung auf einem Tablet.

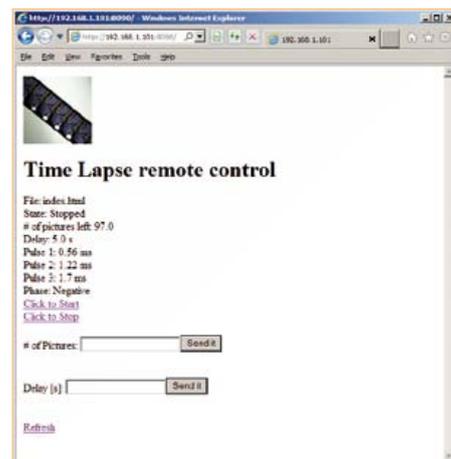


Bild 4. So sieht das Web-Interface in einem Browser aus.

```
ToggleButton mtoggleButton1 =
(ToggleButton) findViewById(R.
id.toggleButton1);
```

Knöpfe gibt es in zwei Ausführungen: aktiv und passiv. Ein passiver Knopf kann via Objektfunktionen ausgelesen und sein Wert



Bild 5. Der grafische Entwurf in Eclipse.

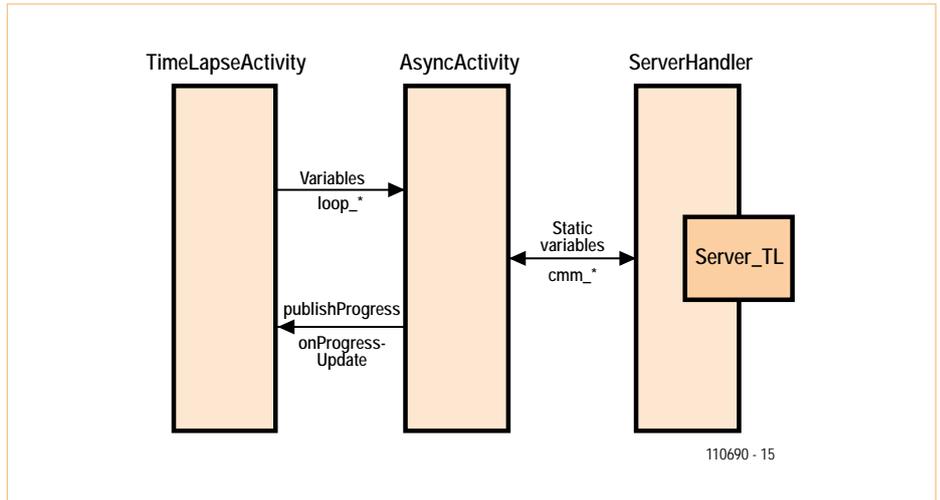


Bild 6. Die Kommunikation zwischen den Klassen.

kann gesetzt werden. Ein aktiver Knopf reagiert auf Betätigung und verändert seinen Status, indem die Klasse „mtoggleButton1.setOnClickListener(this)“ aufgerufen wird. In dieser App ist lediglich der Start-Button aktiv.

Der Zeitraffer-Teil der App besteht aus einer sehr lange laufenden Prozedur. Dies sollte nicht in der Haupt-Klasse „TimeLapseActivity“ geschehen. Diese Klasse ist nämlich für den Betrieb der Oberfläche verantwortlich. Solange diese Klasse beschäftigt ist, kann die GUI nicht aktualisiert werden. In diesem Fall würde die App aus Benutzersicht einfrieren. Android schließt solche Klassen, wenn sie nicht innerhalb von 5 s auf ein Event reagieren, da Android dann davon ausgeht, dass die App abgestürzt ist.

In Android gibt es allerdings auch die Klasse „AsyncTask“. Diese läuft in einem getrennten Thread und kann permanent laufen. Dieser Thread darf allerdings nur einmal gestartet werden. Darum wird dieser Thread nicht mit jeder Zeitraffer-Sequenz neu gestartet, sondern läuft stattdessen kontinuierlich selbst dann, wenn gerade keine Sequenz ausgeführt wird.

Aus „AsyncTask“ heraus darf nicht direkt auf GUI-Elemente zugegriffen werden, sonst erfolgt ein Absturz. Hierzu gibt es deshalb die Funktionen „publishProgress“ und „onProgressUpdate“. „publishPro-

gress“ ruft aus „AsyncTask“ heraus „onProgressUpdate“ auf. Nur diese letzte Funktion darf nämlich Elemente der GUI verändern. Da alle Elemente mit dieser einen Funktion behandelt werden müssen, wird in „onProgressUpdate“ in Abhängigkeit von einem mitgegebenen Parameter eine bestimmte Aktion ausgeführt (siehe den Code in „onProgressUpdate“ und die Variablen „pp_*“).

Die Kommunikation zum Start und Stopp der Sequenzen geschieht über die Variablen „loop_state“, „loop_count“ und „loop_delay“. **Bild 6** zeigt die Kommunikation zwischen den verschiedenen Klassen. Die Form der Impulse für den Servo wird in ein Array geschrieben und mit Hilfe der Funktion „audioTrack“ ausgegeben.

Webserver

Die Steuerung kann auch über WLAN erfolgen. Mit Hilfe einer Webseite können Status und Einstellungen inspiziert und die App gesteuert werden. Um dies generell zu ermöglichen, gibt es drei denkbare Wege: Man kann einfach einen App-Webserver verwenden, von denen es etliche kostenlos im Internet gibt. Damit hätte man Zugriff auf Dateien des Dateisystems. Diese einfache Lösung hätte zur Folge, dass immer wieder eine neue HTML-Datei abgelegt werden

müsste, was auf Dauer für die Zellen des Flash-Speichers nicht gerade optimal ist. Am anderen Ende dieses Spektrums sind umfangreiche Pakete wie „i-jetty“ [3] zu verorten. Hiermit kann man einen vielseitigen Webserver implementieren. Dieser Umfang und die daraus folgende Komplexität sind allerdings nicht wirklich notwendig. Der Autor hat sich deshalb dafür entschieden, den Code der Open-Source-Lösung „android-webserver“ [4] in seine Software aufzunehmen und entsprechend anzupassen. Der größte Änderungsbedarf entsteht dadurch, dass dieser Webserver zwar immer noch Dateien des Dateisystems ausliest (für „logo.jpg“ oder CSS-Dateien im Verzeichnis „websrvr_TL“ durchaus notwendig), aber auch eine Website dynamisch generieren kann, hierzu wird eine änderbare Dateibezeichnung (hier „index.html“) verwendet. Da der Webserver in einem anderen Thread läuft und deshalb keinen direkten Zugang zum Hauptprogramm „TimeLapseActivity“ hat, werden Einstellungen und Status über Static-Variablen übertragen (siehe die Variablen „cmm_*“). Eine Static-Variable kann man als fixe Stelle im Speicher betrachten. Solche Variablen sind für alle Objekte einer Klasse gleich (im Gegensatz zu „normalen“ Variablen, die für jedes Objekt einer Klasse neu generiert werden). Sie können daher auch von anderen Klassen gelesen und geschrieben werden. Static-Variablen wer-

den deshalb auch mit dem Klassen- und nicht mit dem Objektname angesprochen. Die Steuerung der App geschieht durch die Analyse von Links. Wie bei HTML-Formularen (im GET-Mode) werden die Variablen nach einem „?“ von der URL (URI) platziert. In diesem Fall wird mit „index.html?act=start“ eine Zeitraffer-Sequenz starten. Die Anzahl zu machender Bilder sowie die Intervallzeit wird zum Beispiel mit „index.html?del=5“ eingestellt. Dies wird automatisch mit dem HTML-FORM-Befehl erzeugt. Auch die Kommunikation dieser Einstellungen und Befehle geschieht mit Hilfe von Static-Variablen. In der Schleife von „AsyncTask“ werden diese Variablen untersucht. Eine andere Option wäre ein extra Handler für diese Kommunikation gewesen. Dabei hätte es zwei Alternativen gegeben: ein allgemeiner Handler oder einer, der bei der Erstellung des Objekts generiert wird. Dies war aber in diesem Projekt nicht erforderlich.

Übrige Funktionen

Nachfolgend einige allgemeine Anmerkungen zum Programmieren für Android: Es wäre nicht gerade ergonomisch, müsste man Programmeinstellungen wie Impulsbreite, Phase, Bilderzahl und Intervall bei jedem Start der App neu eingeben. Um das zu vermeiden, verfügt Android über spezielle Funktionen, die der Registry von Windows ähneln. Diese Funktionen sind eine Untermenge der Klasse „SharedPreferences“. Beim Programmstart werden hierüber die entsprechenden Parameter ein-

gelesen. Beim Beenden des Programms (bei „onStop“) werden diese Werte wieder gesichert. Diese Einstellungen bleiben sogar bei Installation einer neuen Version des Programms erhalten.

In aktuellen Android-Versionen ist kein Programm-Manager mehr vorhanden. Daher ist nicht sichtbar, welche Apps gerade im Hintergrund laufen. Man kann so schnell den Überblick verlieren. Und so kann es dann passieren, dass man eine App zum zweiten Mal startet. Dabei wird dann tatsächlich eine zweite Instanz der App parallel zur ersten laufen. Bei der hier beschriebenen App wäre das besonders ärgerlich, da zwei auf dem gleichen Port lauschende Webserver sich unvorhersehbar verhalten. Deshalb gibt es drei Gegenmaßnahmen: Zuerst wird das Icon in den Status-Bar gesetzt, wodurch die gestoppte App leicht wieder gestartet werden kann. Hierzu gibt es zwei Funktionen in der Klasse „NotificationManager“. Weiter wird in der Datei „TimeLapsemanifest.xml“ angegeben, dass von dieser App keine zweite Instanz laufen soll (android:launchMode = „singleInstance“). Außerdem sollte das Tablet keinen Bildschirmschoner laufen lassen, denn dies würde die Zeiten ungenauer machen (android:keepScreenOn = „true“). Es gibt Hinweise darauf, dass diese Problematik in der kommenden Android Version 4 weniger störend ausfallen wird.

In Android muss bei der Installation der App durch den Anwender die Zustimmung zu bestimmten Rechten der App gegeben werden, damit sie die Möglichkeit hat, mit

bestimmten Hard- und Software-Teilen zu kommunizieren. Bei diesem Programm ist der Zugriff auf das WLAN-Interface und die Speicherkarte (auf der die Dateien für den Web-Server liegen) sowie die Möglichkeit zur Blockade des Bildschirmschoners erforderlich. Diese Rechte müssen in der Datei „TimeLapsemanifest.xml“ als so genannte „uses-permissions“ enthalten sein. Übrigens taucht an dieser Stelle häufiger der Tippfehler „user-permission“ auf, was ziemlich viel Kopfzerbrechen verursachen kann.

Zum Schluss kann die App noch mit selbstgemachten Icons aufgehübscht werden. Hierzu muss man die Bilder in den Auflösungen 72x72, 48x48 und 36x36 Pixel im „res“-Verzeichnis des Projekts in „drawable-hdpi“, „drawable-mdpi“ und „drawable-ldpi“ im .png-Format ablegen.

Alle Dateien sind wie immer kostenlos per Download von der Elektor-Webseite zum Projekt [5] erhältlich.

(110690)

Weblinks

- [1] <http://eclipse.org/>
- [2] <http://developer.android.com/sdk/eclipse-adt.html>
- [3] <http://code.google.com/p/i-jetty/>
- [4] <http://code.google.com/p/android-webserver/>
- [5] www.elektor.de/110690

—Anzeige



Europas Leiterplatten-Referenz für Prototypen & Kleinserien

www.eurocircuits.de

USB-Stick am Mikrocontroller

Logger für serielle Daten

Von Thomas Fischl (D) thomas@fischl.de

Mit diesem Projekt erhält man die Möglichkeit, einen handelsüblichen USB-Stick direkt an ein Mikrocontroller-System anzuschließen. Damit hat man einen bewährten und billigen nichtflüchtigen Speicher, mit dem sich die Daten vom Mikrocontroller in idealer Weise aufzeichnen, transportieren und transferieren lassen. Für den Anschluss eines USB-Sticks benötigt man aber einen Host-Controller und für das Aufzeichnen der Daten Loggingfunktionen. Beide Aufgaben wurden mit einem PIC24FJ64GB002 von Microchip elegant gelöst.



Eigenschaften:

- USB 2.0 kompatibel
- Datenrate bis 12 Mbit/s
- Verbindung zum Host: +5 V, GND, μ C-Tx
- UART über Datei auf Stick konfigurierbar
- Open-Source-Firmware
- Versorgung: +5 V, 50 mA bis 80 mA

Selbst die kleinsten Mikrocontroller besitzen serielle Schnittstellen (UART). Oft wird diese Schnittstelle zum Austausch von Konfigurationsparametern oder zur Auswertung des aktuellen Betriebszustandes verwendet. Informationen lassen sich so mit Hilfe eines Steuerrechners austauschen. Nicht selten kommt es vor, dass zum Test einer Schaltung Daten über längere Zeit hinweg aufge-

zeichnet werden müssen, um das Verhalten der Baugruppe auch über einen längeren Zeitraum verfolgen zu können. Einen PC für diese Aufgabe zu verwenden, ist auf Grund örtlicher Gegebenheiten oft nicht möglich und außerdem wegen des hohen Stromverbrauches auch Energieverschwendung.

Gegen die Verwendung des vorhandenen Mikrocontrollers spricht der zusätzliche Aufwand zur Integration der Loggingfunktionen und das Speichern großer Datenmengen, zumal RAM und EEPROM für Langzeitaufzeichnungen nur bedingt geeignet sind.

Die energiesparende und universelle Lösung ist der hier vorgestellte USB-Stick-Logger: Er zeichnet alle Daten, die vom vorhandenen Mikrocontroller-System über die serielle Schnittstelle gesendet werden, in einer Datei auf dem angeschlossenen USB-Stick auf.

Hardware

Der Schaltung (Bild 1) besteht im Wesentlichen aus dem Mikrocontroller PIC24FJ64GB002 von Microchip. Dieser Mikrocontroller besitzt eine USB-2.0-OTG-Funktionalität. OTG bedeutet „On-The-Go“ und ist eine Ergänzung der ursprünglichen USB-2.0-Spezifikation, die eine eingeschränkte USB-Host-Funktion und das Wechseln zwischen der Rolle als USB-Host und USB-Device beinhaltet. Letzteres ermöglicht auch die Kommunikation zwischen zwei Geräten, die beide über USB-OTG verfügen. In unserem Fall wird nur die USB-Host-Funktionalität gebraucht, um die Kommunikation mit einem USB-Stick über die normale USB-A-Buchse herzustellen, wie man sie sonst nur an einem PC findet. Der Kern des Mikrocontrollers arbeitet mit 3,3 V. Diese werden über den Spannungsregler IC2 zur Verfügung gestellt. Die serielle Schnittstelle ist auf

Elektor Produkte & Service

- Platine 110409-1
- Programmierter PIC-Controller 110409-41
- Kostenloser Layout-PDF-Download unter [1]
- Kostenloser Firmware-Download (Datei 110409-11 unter [1])

5,5-V-tolerante Eingänge geführt und mit 220-Ω-Serienwiderständen geschützt. Die 5-V-Betriebsspannung der Schaltung dient auch als Busspannung (VBUS) zur Versorgung des USB-Sticks. Zum Schutz vor zu hoher Belastung wird sie über eine rückstellbare Sicherung an die USB-Buchse gelegt. LED D1 und Taster S1 sind direkt an den Mikrocontroller angeschlossen. Die LED zeigt einen Datentransfer an. Mit S1 wird die Datenaufzeichnung beendet. Zum Programmieren des Mikrocontrollers in der Schaltung ist mit K3 eine Programmierschnittstelle vorgesehen, die mit bekannten Programmieradaptern wie PICKit 3 oder ICD2/ICD3 von Microchip kompatibel ist. Der Jumper JP1 wurde in Hinblick auf eine mögliche Weiterentwicklung der Firmware vorgesehen. In der aktuellen Version wird er noch nicht verwendet.

Der Aufbau der Schaltung gestaltet sich mit Hilfe der Platine (Bild 2) sehr einfach. Die Bestückung ist vollständig SMD-frei, und alle Bauteile befinden sich auf der Platinenoberseite. Für den Mikrocontroller kann man problemlos eine Fassung verwenden. Bild 3 zeigt die bestückte Elektor-Musterplatine.

Software

Microchip stellt mit den „Microchip Application Libraries“ Funktionen zur USB-Ansteuerung zur Verfügung. Unter anderem gibt es Unterstützung für die Gerätekategorie „Mass Storage Device“, zu der ein USB-Stick gehört. Das bei Speichersticks am häufigsten eingesetzte Dateisystem FAT wird ebenfalls unterstützt. Die Firmware wurde in MPLAB, der Entwicklungsumgebung von Microchip, mit Hilfe des C30 Compilers erstellt. Alle Programme und erforderlichen Bibliotheken sind kostenlos verfügbar, ebenso kann die Firmware von der Elektor-Projektseite [1] kostenlos heruntergeladen werden. Die Firmware wird über ein Programmiergerät wie zum Beispiel dem PICKit 3 in den PIC übertragen. Alternativ kann man auch einen bereits programmierten Controller einsetzen, der ebenfalls auf der Projektseite [1] angeboten wird. Der Datenfluss zwischen dem UART und dem Dateisystem wurde mit zwei Ping-Pong-Puffern realisiert. Über die serielle Schnittstelle empfangene Zeichen werden in einem der beiden Puffer abgelegt. Ist

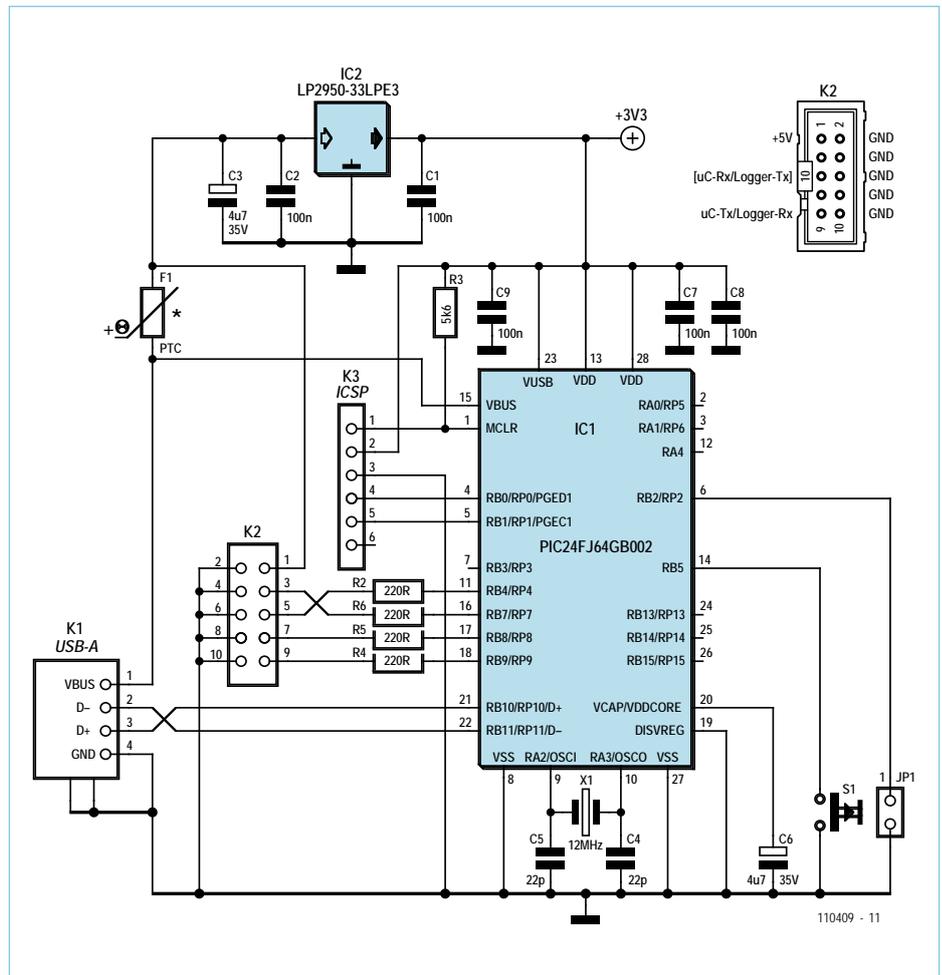


Bild 1. Die Schaltung des USB-Stick-Loggers für serielle Daten besteht im Wesentlichen aus einem Mikrocontroller mit USB-Host-Funktionalität.

dieser vollständig gefüllt, wird er der Programmlogik zum Speichern auf den USB-Stick übergeben. Nachdem die Informationen aus dem Zwischenspeicher verarbeitet sind, wird dieser geleert und wieder an die Empfangseinheit „zurückgespielt“.

Anschlüsse

Die Verbindung zwischen dem USB-Stick-Logger und dem die Daten liefernden Mikrocontroller-System erfolgt über die serielle Schnittstelle (UART). Die Logik-Signalepegel können dabei im Bereich zwischen 3 V und maximal 5,5 V liegen. Soll der Logger an einem RS232-Interface betrieben werden, ist ein entsprechender Pegelwand-

ler (RS232/TTL) erforderlich.

Zur Versorgung des Loggers sind +5 V nötig, die bei Mikrocontrollersystemen jedoch meist bereits vorhanden sind. Der Strombedarf hängt etwas vom verwendeten USB-Stick ab und liegt im Bereich von 50 mA bis 80 mA.

Alle Verbindungen zwischen USB-Logger und Mikrocontroller-System erfolgen über den Steckverbinder K2. In der derzeitigen Logging-Funktion beschränken sich die Verbindungen auf drei Anschlüsse von K2: +5 V (Pin 1), µC-Tx/Logger-Rx (Pin 9) und Masse (Pin 10). Die Pins 2, 4, 6 und 8 liegen ebenfalls an Masse. Die zweite Signalleitung des

Stückliste

Widerstände:

R1 = 1 k
R2, R4..R6 = 220 Ω
R3 = 5k6

Kondensatoren:

C1, C2, C7..C9 = 100 n
C3, C6 = 4µ7/35 V stehend
C4, C5 = 22 p

Halbleiter:

D1 = Low-current-LED, 3 mm
IC1 = PIC24FJ64GB002-I/SP (programmiert: 110409-41)

IC2 = LP2950-33LPE3

Außerdem:

F1 = rückstellbare Sicherung, 250 mA Haltestrom, 500 mA Auslösestrom (Littlefuse 72R025XPR)
X1 = 12-MHz-Quarz
K1 = USB-Buchse Typ A für Platinenmontage
K2 = 2x5-polige Stiftleiste, gewinkelt
K3 = 6-polige Stiftleiste
Platine 110409-1

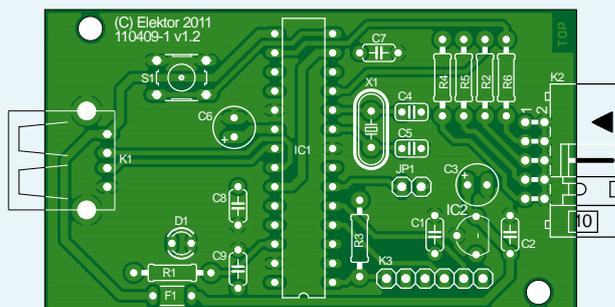


Bild 2. Die Bestückung der vollkommen SMD-freien Platine.

seriellen Interfaces (µC-Rx/Logger-Tx) wird nicht verwendet, ist aber an Pin 5 herausgeführt und steht somit für künftige Firmware-Erweiterungen zur Verfügung, ebenso wie die Port-Pins 16 und 17 des Mikrocont-

rollers, die über 220-Ω-Widerstände an Pin 3 und Pin7 von K2 liegen.

Eine Erweiterungsmöglichkeit wäre zum Beispiel das Lesen vom USB-Stick.

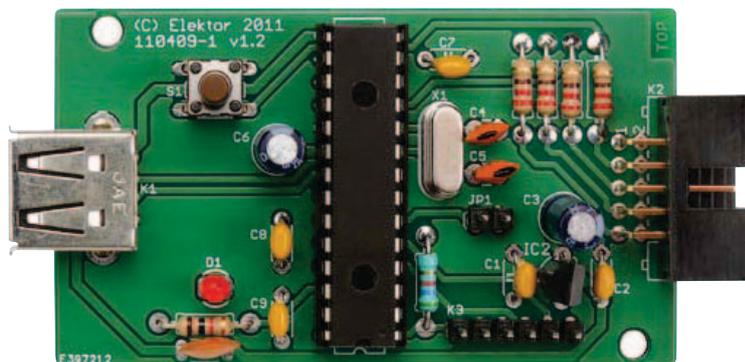


Bild 3. Die bestückte Elektor-Musterplatine.

Konfiguration

Die Parameter der seriellen Verbindung werden über eine einfache Textdatei auf dem USB-Stick abgelegt: „config.txt“. Wird ein USB-Stick erkannt (das kann beim Anlegen der Versorgungsspannung oder später beim Einstecken eines USB-Sticks sein), wird diese Datei ausgelesen und die serielle Schnittstelle entsprechend konfiguriert. Ohne diese Konfigurationsdatei werden Standardwerte gesetzt: 9600 Baud, 1 Startbit, 1 Stoppbit, keine Parität.

Betrieb

Ist der Logger mit Spannung versorgt und mit einem USB-Stick verbunden, befindet er sich im Aufzeichnungsmodus. Ein kurzes Aufleuchten der LED signalisiert das Empfangen von Zeichen über die serielle Schnittstelle. Die empfangenen Daten werden 1:1 in der Datei „logging.txt“ abgelegt. Bevor der USB-Stick abgezogen wird, sollte der Taster betätigt werden. Dann werden alle noch im Puffer befindlichen Zeichen geschrieben und die Logging-Datei sauber geschlossen. Die aufgezeichneten Daten können auf jedem PC gelesen werden – einfach den Stick anstecken und „logging.txt“ mit einem beliebigen Texteditor öffnen.

Anregungen

Neben der reinen Logging-Funktion wären mit der vorgestellten Schaltung auch Firmwareerweiterungen denkbar, die nicht nur das Beschreiben des USB-Sticks erlauben, sondern auch das Auslesen von Daten. Eine andere Möglichkeit wäre der Umbau zum Stand-alone-Logger, der Zustände an den analogen und digitalen Eingängen automatisch in vorgegebenen Intervallen auf dem Speicher-Stick ablegt. An K2 sind insgesamt vier Datenleitungen des PIC-Mikrocontrollers herausgeführt, die über eine interne Matrix mit verschiedenen Peripherie-Signalen belegt werden können, um zum Beispiel eine SPI-Schnittstelle oder einen weiteren UART zu realisieren.

(110409)

Weblink:

[1] www.elektor.de/110409

Max-Planck-Institut für Quantenoptik



Das **Max-Planck-Institut für Quantenoptik**, ein international ausgerichtetes Laserforschungsinstitut mit Sitz in Garching bei München, sucht für die Abteilung Quanten- und Vielteilchensysteme (Leitung: Prof. Dr. I. Bloch) zum nächstmöglichen Termin

eine Elektronikerin / einen Elektroniker (für Geräte und Systeme)

Die Abteilung forscht auf dem aktuellen Gebiet der experimentellen Quanten-, Laser- und Ionenfallenphysik. Die Experimente beruhen auf einem präzisen Zusammenspiel verschiedener modernster Technologien unter Einsatz eines breiten Spektrums an Elektronik (analoge und digitale Regelungen, Radiofrequenz- und Mikrowellentechnik, Präzisions- und Leistungselektronik, kryogene Elektronik sowie Hochspannungsquellen).

Aufgabenbereich: Sie unterstützen uns bei der Entwicklung und Klein(st)serienfertigung von Geräten auf konventioneller und SMD Basis, einschließlich der notwendigen mechanischen Arbeiten für den Gehäusezusammenbau etc. – als Bestandteil der hochkomplexen physikalischen Versuchsaufbauten. Im Einzelnen bedeutet dies, dass Sie die Entwickler bei der Realisierung von Aufträgen sowie bei der messtechnischen Prüfung der gefertigten Komponenten unterstützen. Dies beinhaltet auch die Unterstützung bei der Messtechnikprogrammierung bzw. Ablaufoptimierung. Sie wirken bei der Analyse und Optimierung von Fertigungsprozessen mit. In Ihren Zuständigkeitsbereich fällt auch die Vervollständigung der Gerätedokumentationen bzw. der Fertigungsunterlagen. Ein weiteres Aufgabengebiet stellt die Herstellung von Muster und Unikate einschließlich der Konzipierung von einfachen Schaltungen mit Layout und Gerätedesign dar. Dazu wählen Sie mechanische, elektrische und elektronische Komponenten aus, passen sie an und montieren sie zu Systemen. Sie sind verantwortlich für die Vergabe und Koordination von Aufträgen zur Beschaffung von Bauteilen etc. für die Realisierung von internen und externen Kundenaufträgen. Darüber hinaus wirken Sie bei der Bestandserhaltung des umfangreichen Klein- und Kleinstteilelagers der Abteilung mit. Das Warten von Komponenten und Geräten bzw. die Instandsetzung fällt ebenso in Ihre Verantwortung.

Anforderungen: Wir erwarten eine abgeschlossene Ausbildung zum/zur Elektroniker/in für Geräte und Systeme und setzen (elektro-)technisches Verständnis voraus. Freude und Interesse an der Mikrocontroller-Programmierung und Leiterplattenentwurf sollten Sie mitbringen. Erfahrungen mit Analogschaltungen und RF Technik sind erwünscht. Wenn Sie handwerkliches Geschick sowie die Fähigkeit zum eigenständigen Arbeiten mitbringen und Freude an der Zusammenarbeit mit Studierenden und Wissenschaftlern haben, dann freuen wir uns auf Ihre Bewerbung. Englischkenntnisse wären von Vorteil. Neue Herausforderungen packen Sie an, um Ihr breites Fachwissen anzuwenden und Neues zu lernen.

Wir bieten einen modern ausgestatteten Arbeitsplatz, ein angenehmes Betriebsklima sowie eine intensive Einarbeitung und unterstützen Sie ggf. bei der Wohnungssuche. Jährliche Fortbildungsmöglichkeiten werden im Arbeitsplatzumfeld geboten. Die Vergütung erfolgt entsprechend Befähigung und fachlicher Eignung gemäß Tarifvertrag für den öffentlichen Dienst (TVöD). Die Sozialleistungen entsprechen den Regelungen des öffentlichen Dienstes (Bundesdienst).

Schwerbehinderte werden bei entsprechender Eignung bevorzugt berücksichtigt.

Das Max-Planck-Institut für Quantenoptik ist bestrebt, den Anteil der Frauen im wissenschaftlichen Bereich zu erhöhen und bittet daher Frauen um Bewerbungen.

Ihre Bewerbung mit den üblichen Bewerbungsunterlagen richten Sie bitte unter Angabe Ihrer Telefonnummer an das

**Max-Planck-Institut für Quantenoptik
Betriebsleitung
Hans-Kopfermann-Str. 1, 85748 Garching b. München**

Gerne können Sie Ihre Bewerbung auch per E-Mail an uns richten: Nina.Kilgus@mpq.mpg.de

THE NEW PICOSCOPE 2205 MSO MIXED SIGNAL OSCILLOSCOPE

GREAT VALUE, PORTABLE, HIGH END
FEATURES AS STANDARD AND EASY TO USE



Think Logically...

For your chance to **WIN** a PicoScope 2205 MSO visit
www.picoscopemso.com/3116 and enter the code ELK3

Channels	2 Analog, 16 Digital
Resolution	8 bit
Bandwidth	Analog 25 MHz,
Digital frequency	Digital 100MHz combined
Sampling rate	200MS/s
Trigger modes	Edge, Window, Pulse width, Window pulse width, Dropout, Window dropout, Interval, Runt pulse, Digital, Logic
Price	£349

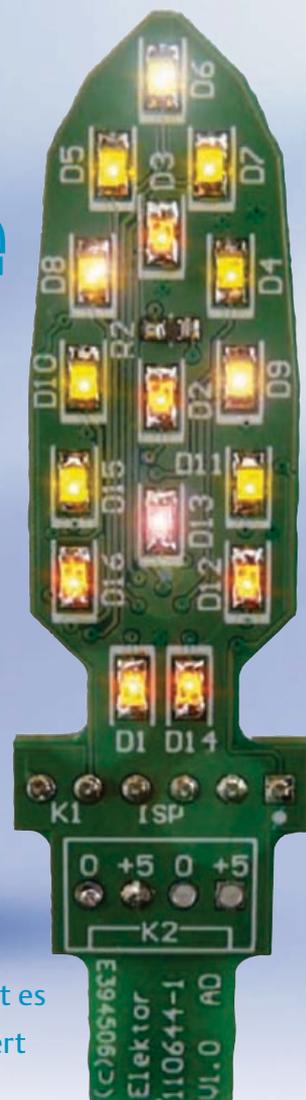
pico
Technology



www.picoscopemso.com

Elektronische LED-Kerze

Man kann sie ausblasen!



Kerzenimitate mit einer LED als Kunstflamme gibt es überall zu kaufen. Doch hier geht es um eine Variante, die dem Vorbild viel näher kommt: Diese elektronische Kerze flackert nicht nur realistisch - man kann sie auch ausblasen!

Von Antoine Deschamps (F)

Technische Daten

- Realitätsnahe Flammensimulation
- Ausblasfähig
- Anzünden durch Berührung
- PIC16F1827 programmiert in C
- Freier Source-Code
- Luftzug-Detektor mit NTC
- Schwierigkeitsgrad: mittel

Kerzenlicht ist von Natur aus ungleichmäßig und von daher muss man schon die eine oder andere Maßnahme ergreifen, um eine Kerze samt Flamme einigermaßen realistisch zu imitieren. Also braucht man Lösungen für das Anzünden, für animiertes Leuchten mit Flackereffekt und nicht zuletzt für die Ausblasfähigkeit.

Vor allem das animierte Leuchten schreit quasi nach einem Mikrocontroller. In diesem Projekt kommt der Typ PIC16F1827 von Microchip zum Einsatz. Sein 4 KB fassender Programmspeicher reicht für die Ani-

mation des Lichts und die anderen Funktionen locker aus.

Der verwendete Mikrocontroller verfügt zudem über einfach anzuwendende Funktionen zur Ansteuerung von Sensortasten (denglisch: Touchpad). Dafür wurde die Rückseite der Platine mit einer entsprechenden Kupferfläche versehen, um das so generierte Touchpad zum „Anzünden“ der Kerze nutzen zu können. Der Punkt auf dem i ist dann die Möglichkeit, die Kerze ausblasen zu können. Hierzu wurde einfach ein NTC verwendet – das funktioniert erstaunlich gut!

Blasdetektor

Bei dem als Luftbewegungsdetektor verwendeten NTC R2 (siehe **Bild 1**) handelt es sich um ein Exemplar im SMD-0603-Gehäuse mit einem Nennwiderstand von 220 Ω bei 25 °C, einem so genannten Beta-Koeffizienten von 3.540 K und einer Verlustleistung von maximal 180 mW. Ein großer Beta-Koeffizient korrespondiert mit einem großen Temperaturkoeffizienten, der Widerstand reagiert also sensibel auf Temperaturunterschiede.

Die Anordnung funktioniert wie folgt: Wenn man annimmt, dass der Luftstrom,

der beim Blasen entsteht, in etwa Raumtemperatur hat, dann sollte man auf den ersten Blick annehmen, dass sich damit keine große Widerstandsänderung des NTCs erreichen lässt. Dass die Sache doch funktioniert – und zwar gut und zuverlässig – liegt daran, dass der NTC schwach geheizt wird. Dadurch führt ein Luftzug zu einer Temperatursenkung mit der Folge einer Widerstandsänderung, die man elektronisch problemlos detektieren kann. Diese Widerstandsänderung wird mit einem der analogen Eingänge des Mikrocontrollers gemessen.

Der NTC bildet zusammen mit dem Vorwiderstand R1 (100 Ω) einen niederohmigen Spannungsteiler. Direkt nach dem Einschalten fließt bei einer Raumtemperatur von 25 °C der Strom:

$$I_{25} = 5 \text{ V} / (100 \Omega + 220 \Omega) = 16 \text{ mA}$$

Die Spannung über dem NTC ist dann:

$$U_{25} = 220 \Omega \times I_{25} = 3,5 \text{ V}$$

Daraus ergibt sich dann die Heizleistung am NTC von:

$$P_{25} = U_{25} \times I_{25} = 3,5 \text{ V} \times 16 \text{ mA} = 56 \text{ mW.}$$

Das ist noch weit genug von der maximalen Verlustleistung von 180 mW entfernt, reicht aber aus, um die Temperatur des NTCs gegenüber der Umgebungsluft zu erhöhen. Dabei wird der NTC niederohmiger und es entsteht in der Folge sogar noch etwas mehr Leistung (theoretisches Maximum ist 62,5 mW bei $R_2 = 100 \Omega$). Nach wenigen Sekunden stellt sich ein thermoelektrisches Gleichgewicht ein und die Temperatur des NTCs bleibt stabil, wie man am Oszillogramm in **Bild 2** sehen kann. Die Software ist so gestrickt, dass zunächst 20 s gewartet wird, bis sich die Temperatur stabilisiert hat. Erst dann werden Messungen vorgenommen. Es ist gar nicht so einfach, die Änderung der Spannung bei einem Ausblasversuch zu erfassen. Nach einigem Probieren zeigte sich, dass eine positive Veränderung um 25 mV als Schwelle zur Blaserkennung ausreicht.

Aus Software-Perspektive ist es sinnvoll, die Abtastrate des A/D-Konverters an das Zeitverhalten der Signaländerungen anzupassen. Nach eingangsseitiger Filterung werden dazu Mittelwerte aus 16 Messungen gebildet. Die Schwelle von 25 mV bezieht sich auf den Unterschied zwischen zwei Mittelwerten. Als Ergebnis muss man dem NTC schon recht nahe kommen, um die Kerze erfolgreich ausblasen zu können. Bei niedrigeren Schwellen wäre der Blasdetektor allerdings empfindlicher für Zugluft. Mann muss es mit den Präzisionsanforderungen an dieser Stelle nicht übertreiben. Es geht ja nur darum, eine Abweichung der Temperatur von einem Mittelwert zu detektieren. Man braucht also keine externen Spannungsreferenzen, und selbst die interne Referenz des Controllers ist nicht nötig. Im Gegenteil: Die Betriebsspannung als Referenz hat sogar den Vorteil, dass sich ihre Drift nicht auf die Messung auswirkt.

Touchpad

Das Touchpad zum Anzünden der Kerze ist direkt mit dem Analogeingang 0 des Mikrocontrollers IC1 verbunden (siehe **Bild 1**). Das Datenblatt zum PIC16F1827 beschreibt den Anschluss kapazitiver Sensoren recht erschöpfend, und daher ist die Program-

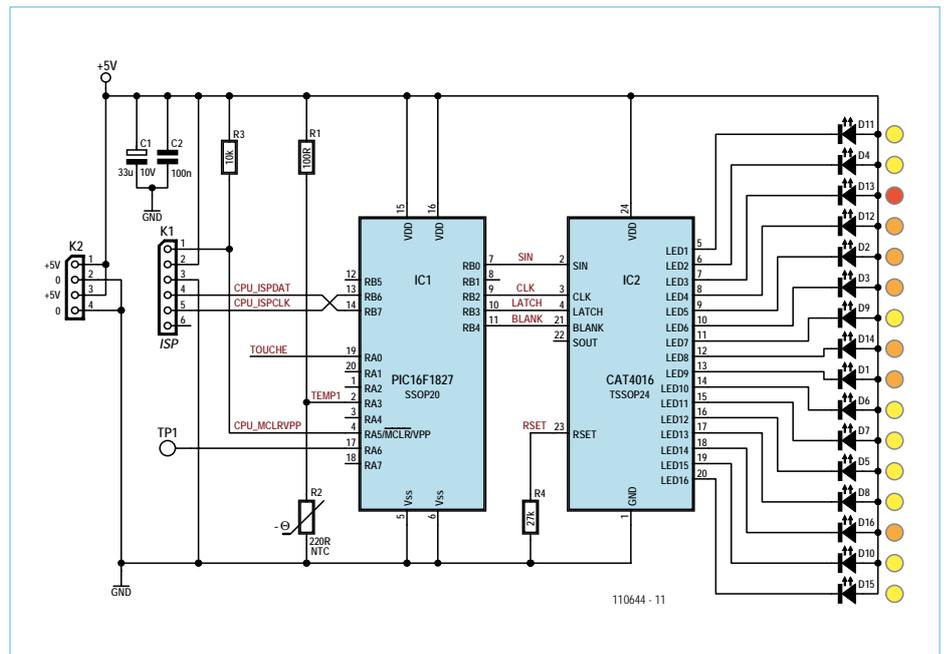


Bild 1. Schaltung der LED-Kerze. Dank kleiner SMD-Gehäuse passt die Elektronik komplett auf die Rückseite der vorderseitig mit LEDs bestückten Platine. Die 5 V eines externen Netzteils werden vom Elko C1 nochmals gesiebt. Der Mikrocontroller IC1 kommt mit minimalster Beschaltung aus. Da Präzision hier keine Rolle spielt, genügt der interne Taktgenerator.

mierung nicht allzu schwierig. Nur ein relevantes Detail war unklar: Man kann nicht wissen, auf welcher Frequenz der mit dem Touchpad aufgebaute Oszillator schwingt. Die Frequenzvariationen entstehen durch Kapazitätsänderungen, die sich bei Annäherung eines Fingers an die Kupferflächen der Platine ergeben. Es bleibt also nichts anderes übrig, als die Frequenz zu messen, um Referenzwerte zu erhalten. Aus diesem Grund ist der Oszillator-Ausgang mit dem frei laufenden Timer 0 verbunden. Als Testpunkt zur Messung der Frequenz wurde das MSB des Zähler-Registers TMR0 von Timer 0 an einen Ausgang gelegt. Eine Messung mit dem Oszilloskop ergab, dass die Ruhefrequenz von etwa 130 Hz auf bis zu 90 Hz absinkt, wenn ein Finger das Touchpad berührt. Da es sich hier um das MSB eines 8-bit-Registers handelt, ist die reale Frequenz um den Faktor 256 höher und beträgt daher am Eingang des Timers rund 32 kHz. Da die Software mit 1 kHz getaktet wird, muss man nur jede Millisekunde den Zählerstand abfragen. TMR0 hat bei nicht berührtem Touchpad einen Wert von 32, der bei Berührung auf 23 sinkt. Nach jedem Auslesen wird TMR0 natürlich wieder auf „0“ gesetzt, damit ein neuer Zählzyklus beginnt. Der gewählte Schwellwert von 26 ist schon recht niedrig. Man muss also den Finger schon deutlich auf die Platine

legen, damit er erkannt wird. Höhere Werte machen die Schaltung empfindlicher. Noch eine Anmerkung: Wenn man solche Projekte entwickelt, sollte man von vorne herein einige Testpunkte mit einplanen, denn ein Oszilloskop ist eine nützliche Hilfe beim Programmieren...

Die Flamme

Diese elektronische Kerze ist nicht zur Beleuchtung, sondern zur dekorativen Illumination gedacht. Aus diesem Grund werden durchweg Low-current-LEDs mit einem maximalen Dauerstrom von 2 mA betrieben. Die-

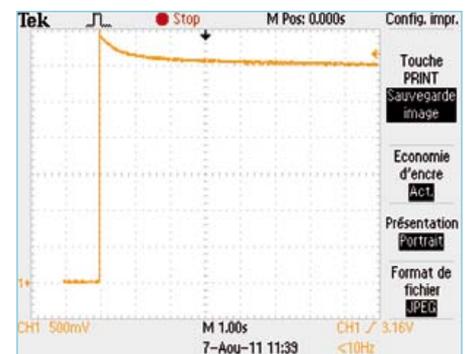


Bild 2. Die Temperatur des Ausblasdetektors stabilisiert sich innerhalb von wenigen Sekunden nach dem Einschalten.

Stückliste

Widerstände (SMD 0603):

- R1 = 100 Ω
- R2 = NTC, 220 Ω, Epcos
B5731 1V2221J60 (Farnell 129-9912
oder RS-Components 706-2702)
- R3 = 10 k
- R4 = 27 k

Kondensatoren:

- C1 = 33 μ/10 V Elko, C-Gehäuse,
z.B. Panasonic EEE1AA330SR
- C2 = 100 n, SMD 0603

Halbleiter:

- D1..D3, D12, D14, D16 = LED, orange,
low current, SMD 0805, z.B. King-
bright KPHCM-2012SECK
- D4..D11, D15 = LED, gelb, low cur-
rent, SMD 0805, z.B. Kingbright
KP-2012SYCK
- D13 = LED, rot, low current, SMD 0805,
z.B. Kingbright KPHCM-2012SURCK

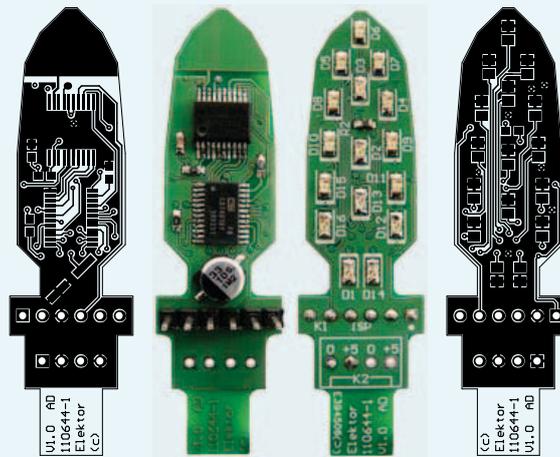


Bild 3. Die doppelseitige Platine kann trotz der vielen SMDs noch von Hand bestückt werden.

IC1 = PIC16F1827-I/SS (SSOP20), Mi-
crochip (programmiert: 110644-41)
IC2 = CAT4016Y-T2 (TSSOP24),
ON Semiconductor

Außerdem:

K1 = 6-pol. Pfostenstecker, RM 2,54 mm
K2 = 4-pol. Pfostenstecker, RM 2,54 mm
Platine: 110644-1

Software, Platine etc.

www.elektor.de/110644

ser Leistungsbereich wird gut von SMD-LEDs abgedeckt, die zudem den Vorteil haben, nur wenig Platinenfläche zu benötigen.

Zur Ansteuerung der insgesamt 16 LEDs wird mit dem CAT4016 ein spezielles IC von ON Semiconductor verwendet, das Konstantstromquellen für maximal 16 LEDs enthält. Der LED-Strom lässt sich durch einen einzigen Widerstand (R4) für alle LEDs gemeinsam einstellen. Dabei entspricht der LED-Strom dem 50-fachen Wert des Stroms, der durch den am Pin Rset angeschlossenen Widerstand fließt. Bei einer Spannung von 1,2 V an Rset ergibt sich mit R4 = 27 kΩ ein Strom von 44 μA, was in der Spitze 2,2 mA durch die LEDs entspricht. Bei Verwendung anderer LEDs kann man mit dem Wert von R4 den Strom entsprechend anpassen.

IC2 wird seriell angesteuert. Dazu sind vier Signale vom Controller erforderlich:

- CLK: Taktsignal
- SIN: Datensignal „serial in“
- LATCH: Speichersignal
- BLANK: Signal zum gleichzeitigen Austasten aller 16 LEDs

Das Datenblatt zum CAT4016 beschreibt die Signalfolgen zur individuellen Ansteuerung der 16 LEDs. Die Daten werden an SIN mit dem Takt CLK eingelesen und per LATCH-Signal in einem Schieberegister gespeichert, über das sie auch an einem seriellen Ausgang (SOUT) zur Verfügung stehen. Letzteres wird hier zwar nicht ausgenutzt, aber auf diese Weise könnte man mehrere solcher ICs in einer Kette verschalten, wobei die vier Signale vom Mikrocontroller nur einmal benötigt werden.

Ein CAT4016 kann die angeschlossenen LEDs

nur ein- oder ausschalten. Um die Animation etwas realistischer zu gestalten, wäre eine individuelle Helligkeitssteuerung wünschenswert. Dieses Feature bleibt somit die Aufgabe von Software und Mikrocontroller. Da der Takt des Mikrocontrollers 16 MHz beträgt und pro Befehl vier Taktzyklen benötigt werden, kommt man auf einen effektiven Befehlstakt von 4 MHz. Timer 1 wird so eingestellt, dass er jede Millisekunde einen Interrupt generiert. Dieser Takt wird häufig als System-Tick bezeichnet. Bei jedem Tick wird der Inhalt des Schieberegisters von IC2 aktualisiert. Beim Prototypen benötigte der Transfer der Befehle für 16 LEDs insgesamt 160 μs, was einer CPU-Auslastung von 16 % entspricht.

Dank dieser hohen Tick-Rate kann das menschliche Auge kein Flackern der LEDs wahrnehmen. Es wäre sogar gefahrlos möglich, die Tick-Rate deutlich zu reduzieren, falls noch etwas mehr freie Rechenleistung für andere Aufgaben gebraucht wird. Mit der hier beschriebenen Anordnung wurde also eine Art Software-PWM realisiert. Dabei erwiesen sich die fünf Helligkeitsabstufungen 0 %, 25 %, 50 %, 75 % und 100 % als ausreichend, um eine flackernde Kerze zu simulieren. Trotz der relativ großen Helligkeitssprünge klappt die Simulation gut genug. Eine feinere Abstufung bedeutet auch einen größeren Codeumfang und somit mehr Speicherplatzbedarf für den Code.

Platine

Bei der nur 17 mm × 60 mm messenden Platine beanspruchen die LEDs mit 17 mm × 239 mm den größten Teil der Fläche einer Platinenseite (siehe Bild 3). Der Rest der

Platine beherbergt auf der Rückseite den Mikrocontroller, die LED-Ansteuerung, den Programmieranschluss und die Anschlüsse für die Stromversorgung.

Auf der Oberseite der Platine befinden sich die SMD-LEDs, die in ihrem 0805-Gehäuse mit nur 2 × 1,25 mm ziemlich winzig ausfallen. Auch der kleine Temperatursensor ist logischerweise auf dieser Seite bestückt, sonst könnte man die Kerze ja kaum ausblasen.

Auf der „Lötseite“ kann man sehen, dass die Spitze der Platine aus der Kupferfläche für das Touchpad besteht. Ein Antippen oben reicht also zum Anzünden. Außerdem findet sich hier noch der Testpunkt TP1 oberhalb des Mikrocontrollers.

Damit alle anderen Bauteile auf der Rückseite Platz finden, wurden besonders kleine IC-Gehäuse vorgesehen: Der Mikrocontroller steckt in einem SSOP20-Gehäuse und der LED-Treiber in einem TSSOP2-Gehäuse, beide im Rastermaß 0,65 mm. Die kleinen Leiterbahnabstände sind eine Herausforderung, die mit der richtigen Ausrüstung gut zu bewältigen ist. Dazu zählen: eine beleuchtete Lupe, ein temperaturgeregelter LötKolben mit feiner Spitze, Löt Draht mit 0,5 mm Durchmesser (AWG24) und nicht zu vergessen: Flussmittel in einer Spritze! Letzteres hilft beim Nachlöten ungemein. ;-) Die klassische Bestückungstechnik für SMD-ICs ist: Zuerst etwas Lötzinn (oder Lötpaste) auf ein Pad für einen Pin an einer Ecke aufbringen, dann den Pin anlöten, das IC ausrichten und einen Pin an der gegenüber liegenden Ecke anlöten. Nun kann man den

Code-Auszug, der zeigt, wie Timer 1 für die 1-kHz-System-Ticks präpariert wird:

```

T1CON = 0b01100101 ; // Timer 1 configuration (Tick Interrupt Source)
// 01      : Timer1 source is Fosc = 16 MHz
// 10      : Prescaler = 4 -> Ftimer = 4 MHz
// 0       : Timer1 oscillator circuit disabled
// 1       : Do not synchronize external clock input
// 0       : unimplemented
// 1       : TMR1ON=1 -> Timer1 enabled

T1GCON = 0b00000000 ; // Timer 1 gate control
// 0       : TMR1GE=0 -> Timer 1 counts regardless of Gate functions

TMR1H = 0xF0 ; // 0xF05F is the complement of 0x0FA0=4000
TMR1L = 0x5F ; // Counter startup value for 1ms interrupt period (1000 Hz)
PIE1 = 0b00000001 ; // Interrupt sources
// 1       : TMR1IE=1 -> Timer1 IT source enabled

```

Ein Auszug, der die Initialisierung der Funktionen für das Touchpad zeigt:

```

OPTION_REG = 0b01101000;
// 0       : WPUEN#=0 -> Weak pull-ups are enabled by individual WPUx
// 1       : Interrupt on rising edge of INT (not used)
// 1       : TMR0CS=1 -> Timer 0 counts on TOCKI
// 0       : TMR0SE=0 : Timer 0 counts of rising edge of TOCKI
// 1       : PSA=1 : prescaler is not assigned to Timer0
// 000: Prescaler value, 1:2

// Capacitive sensing configuration
CPSCON0 = 0b10000101 ;
// 1       : Capacitive module is enabled
// 000     : unimplemented
// 01      : oscillator in low-range
// 0       : current direction status (read-only)
// 1       : TOXCS=1 -> Timer0 input is Capative Oscillator output

CPSCON1 = 0b00000000 ; // Source is CPS0/RA0

```

anderen Pins mit Lot und Flussmittel zu Leibe rücken. Ungewollte Lötzinnbrücken zwischen den Pins treten bei aller Sorgfalt immer wieder auf, lassen sich aber mit dem Lötkolben und etwas Entlötlitze gut entfernen. Die Platine ist doppelseitig und mit Durchkontaktierungen versehen. Die Stromversorgung wird mit großen Kupferflächen realisiert: Die Fläche für +5 V befindet sich vorne bei den LEDs, da deren Anoden an +5 V liegen. Auf der Platinenrückseite befindet sich die Massefläche.

Der Anschluss für die In-circuit-Programmierung K1 besteht aus einer Reihe von Löchern für einen einreihigen sechspoligen Pinheader, dessen Belegung dem ICD3-Programmier-Interface von Microchip entspricht (siehe **Tabelle 1**).

Der Anschluss für die Stromversorgung K2 ist vierpolig. Je zwei Pads sind für +5 V (1 und 3) und Masse (2 und 4) zuständig.

Tabelle 1. Pin-Belegung des ICSP-Pfostensteckers K1. Pin 1 ist der mit dem quadratischen Pad.

Pin	Signal
1	MCLR/Vpp
2	+5 V
3	Masse
4	ISPDAT
5	ISPCLK
6	nicht verwendet (PGM-Signal)

Wie bei K1 wird auch hier Pin 1 durch ein quadratisches Pad angezeigt. Die doppelten Anschlüsse für die Versorgungsspannung sind nützlich, wenn man mehrere elektronische Kerzen für einen besonders eindrucksvollen Effekt kombiniert. Man kann sie dann alle in einer Kette miteinander verbinden.

Software

Die in C codierte Software belegt 84,5 % des verfügbaren Programmspeichers im Mikrocontroller – in anderen Zahlen: 3.460 der 4.096 verfügbaren Speicherplätze sind belegt. Etwa 75 % des Codes besteht aus der Definition der Flackermuster. Falls der Platz für weitere Software-Funktionen knapp werden sollte, lässt sich hier noch etwas optimieren. Der vollständige Source- und Hex-Code ist wie immer kostenlos von der Elektor-Webseite zu diesem Artikel per Download verfügbar. Dort findet man auch das Platinen-Layout als PDF-Datei und Bestellmöglichkeiten für die Platine und für den programmierten Controller.

Wir wünschen Ihnen viel Freude mit der elektronischen Kerze und schöne Weihnachtstage!

(110644)

Von den Machern von Elektor!

elektor

SPECIAL PROJECT

Audio 5

5/2012
(D) 16,90 €
(A) 18,50 €
CHF 28.90
(B) 18,50 €
(L) 18,50 €

High-End-Verstärker und Lautsprecherboxen

**Sieben Verstärker
für den Selbstbau, u. a.:**

- 1-W-Gitarrenverstärker
für den Hausgebrauch
- Röhrenvorstufe
mit aktiver Frequenzweiche
- Symmetrischer Vorverstärker
- Single-Ended KR300 B

**Sieben Lautsprecher
für den Selbstbau, u. a.:**

- Bildlautsprecher für Flat-TV
- D'Appolito-Box Silverado
- HATT Mikro Monitor
- Morel Elite 420

**Jetzt neu
am Kiosk!**



Oder frei Haus unter www.elektor.de bestellen!

Elektor Academy Web-Seminare - in Kooperation mit *element14*

Elektronik-Fans aufgepasst: *Elektor Academy* und die Community-Plattform *element14* präsentieren mehrere (englischsprachige) Webinare zu besonders interessanten Elektor-Projekten. Die Teilnahme ist komplett kostenlos! Sie müssen sich nur unter der Adresse www.elektor.de/webinar registrieren.

Webinar-Programm:

E-blocks, Twitter and the Sailing Club

Datum: 17. November 2011

Beginn: 16:00 Uhr (MEZ)

Präsentiert von: Ben Rowland und John Dobson (Matrix Multimedia)

E-blocks sind kleine Schaltungs-Boards für typische Aufgaben der Elektronik- und Embedded-Entwicklung. In diesem Webinar demonstrieren Ben Rowland und John Dobson die Rapid-Prototyping-Fähigkeiten der E-blocks – am Beispiel einer Applikation, die Twitter-Meldungen an die Mitglieder eines Segelclubs absetzt.

Let's Build a Chaos Generator

Datum: 15. Dezember 2011

Beginn: 16:00 Uhr (MEZ)

Präsentiert von: Maarten Ambaum und R. Giles Harrison (Reading University)

Dieses Webinar ist ein Making-Of des interessanten Chaos-Generator-Projekts, das im September und Oktober 2011 in Elektor vorgestellt wurde. Seien Sie in Wort und Bild dabei, wenn wir die OpAmps zum Glühen bringen!

Here comes The Elektor Bus!

Datum: 19. Januar 2012

Beginn: 16:00 Uhr (MEZ)

Präsentiert von: Jens Nickel (Elektor-Redakteur)

Zum ElektorBus-Projekt haben Dutzende Leser mit Ideen, Hinweisen und eigenen Entwicklungen beigetragen. Elektor-Redakteur Jens Nickel erzählt uns, wie alles anfang – doch er kann auch mit vielen Hintergrund-Details zu diesem interessanten Projekt aufwarten, das immer wieder knifflige Herausforderungen bereithält.

Webinar-Archiv:

Platino – an ultra-versatile platform for AVR microcontroller circuits

Präsentiert von: Clemens Valens (Elektor)

In vielen Elektronik-Projekten kommt die Platine oft nur an zweiter Stelle. Natürlich ist es einfacher, eine Schaltung auf einer Platine aufzubauen. Eine Schaltung lässt sich aber auch ohne Platine realisieren – eine Platine jedoch nicht ohne Schaltung. Dass das nicht immer so sein muss, beweist das Platino-Projekt. Dabei dreht sich alles um die Platine, und die Schaltung ist eigentlich nicht so wichtig.

Webinar verpasst? Aufzeichnung abrufbar unter www.element14.com!

Jetzt
GRATIS
anmelden!



element14

www.element14.com

Die Teilnehmerzahl ist begrenzt! **JETZT ANMELDEN** unter www.elektor.de/webinar!

Unknackbares Codeschloss

Verschlüsselt mit 128-bit-AES

Von Elbert Jan von Veldhuizen (NL)

Haben Sie sich schon einmal gefragt, ob die Funkverriegelung für Ihr Auto wirklich sicher ist? Dieses Projekt demonstriert, wie man mit Hilfe eines Standard-Mikrocontrollers sichere und praktisch unknackbare Sender und Empfänger für eine IR-Fernbedienung realisieren kann. Dabei wird die Zweiwege-IR-Kommunikation per 128-bit-AES-Algorithmus verschlüsselt.

Funksender zur Ent- und Verriegelung von Autos sind mittlerweile Standard und längst kein Luxus mehr. Diese Funkfernbedienungen senden einen Code. Sobald der Empfänger (das Schloss) den Code erkennt, werden die Türen geöffnet (oder abgeschlossen). Ein Nachteil dieses verbreiteten Systems ist, dass ein böswilliger Zeitgenosse mit einem Empfänger den Code aufzeichnen und so das Schloss mit einfachen Mitteln knacken kann. Der Bösewicht kann den gespeicherten Code nämlich jederzeit reproduzieren, und dann ist das Auto nicht mehr geschützt. Ein wesentlich sichereres Verfahren stellt das „Challenge Handshake Authentication Protocol“ dar, das auch bei einigen Formen des Internet-Bankings zum Einsatz kommt. In diesem Fall schickt das Schloss einen Code und die Fernbedienung muss diesen empfangen, eine Berechnung damit durchführen und das Resultat zurücksenden. Da es hier nicht auf den Code, sondern auf die Berechnung ankommt, kann der Code permanent gewechselt werden, so dass das Abhören des Codes oder der Antwort zwecklos ist.

Beim nächsten Durchgang wird das Schloss also einen anderen Code senden. Solange ein Mensch mit böser Absicht diese Berechnung nicht kennt (und diese nicht zu simpel ist), ist das Schloss gesichert.

Der Dreh- und Angelpunkt dieser Methode ist also eine gute Berechnung. Verschlüsselungstechniken sind für diesen Zweck gut geeignet, denn auch dabei werden mit Hilfe eines Schlüssels Daten in andere Daten transformiert. Die Fernbedienung muss also verschlüsseln können.

Algorithmen

Die nächste Frage ist dann, was einen guten Krypto- oder Verschlüsselungs-Algorithmus ausmacht. Bei einer Verschlüsselung gibt es die zu verschlüsselnden Daten (in), den Schlüssel (key) und die verschlüsselten Daten (out).

- Der Krypto-Algorithmus entspricht der Funktion $out = f(in, key)$. Es gibt auch eine Funktion $in = f_{inv}(out, key)$, aber eben keine Funktion $key = f_{key}(in, out)$.
- Zu jedem Wert „in“ gehört ein spezifischer Wert „out“. Oft ist es auch möglich, dass mehr als ein Wert für „in“ zum gleichen Wert „out“ führt.
- Ähnliches gilt für den Schlüssel: Verschiedene Schlüssel sollten aus einem Wert „in“ verschiedene Werte für „out“ generieren.

Der erste Punkt sorgt dafür, dass selbst wenn sowohl „in“ als auch „out“ abgehört wird, noch lange kein Schlüssel mit einer Funktion „ f_{key} “ ableitbar ist. Die einzige gangbare Methode zur Auffindung des Schlüssels ist eine so genannte „brute force attack“, bei der schlicht alle möglichen Schlüssel für die Funktion „ f “ durchprobiert werden. Der Schlüssel muss so lang sein, dass stures Ausprobieren aller Kombinationen schlicht zu lange dauert. Leistungsfähige Computer können heute Schlüssel mit 64 bit Länge innerhalb eines Tages knacken. Bei 128 bit sollte diese Methode länger dauern, als das Weltall bislang existiert. Das dürfte ausreichen. ;-)

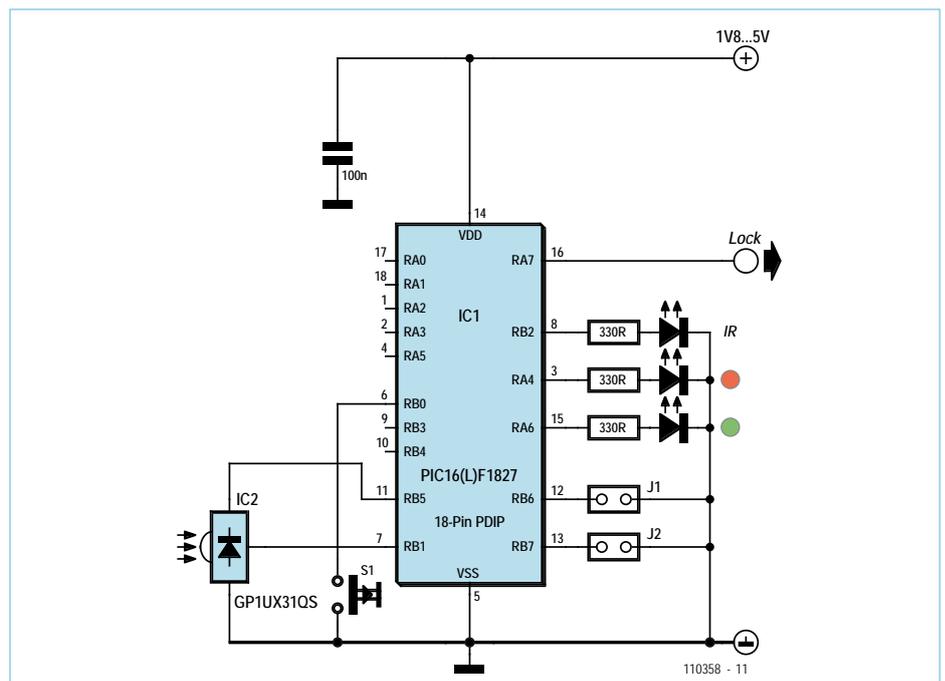


Bild 1. Schaltung des Schlosses...



Als Verschlüsselungs-Algorithmus wurde AES gewählt. Damit werden beispielsweise auch WLAN-Router gegen Lauschattacken und unbefugte Mitbenutzung gesichert. Bis heute wurde dieser Algorithmus noch nicht geknackt. Es wurde also bisher noch keine Funktion „f_{key}“ gefunden, die es erlauben würde, mit weniger Kombinationen den richtigen Schlüssel zu finden. Die Ressourcen eines Mikrocontrollers werden durch AES allerdings sehr strapaziert. Die Berechnungen alleine erfordern schon 240 Byte an RAM. Der Code dafür ist etwa 1.500 Befehle lang und seine Ausführung erfordert rund 30.000 Befehlszyklen. Außerdem werden viele Tabellen und Arrays benötigt.

Der noch recht neue Controller PIC16F1827 von Microchip ist diesen Anforderungen gewachsen. Er verfügt über einen Code-Speicher für 4096 Worte und 386 Byte RAM. Außerdem kann er mit bis zu 32 MHz arbeiten und verfügt über einen verbesserten Befehlssatz, der den Umgang mit Arrays sehr vereinfacht. Das RAM ist in Blöcken zu je 80 Byte organisiert, kann mit den neuen Befehlen jedoch auch als zusammenhängender Block gesehen werden (linear mode), was das Handling von Tabellen vereinfacht.

Die Schaltung

Außer einem Mikrocontroller braucht es für eine funktionierende Sender- und Empfän-

gerschaltung mit AES nur wenige andere Bauteile. **Bild 1** zeigt die Schaltung des Schloßes und **Bild 2** die Schaltung der Fernbedienung. Die Kommunikation läuft der Einfachheit halber über IR-LEDs, ganz wie bei den Fernbedienungen für die Unterhaltungselektronik. Für das Senden ist LED D1 zuständig, und der IR-Empfänger ist im IR-Modul IC2 enthalten, das mit 36 kHz arbeitet. Die Schalter für Bedienung und Konfiguration werden einfach direkt an die I/O-Ports des Mikrocontrollers angeschlossen. Die integrierten Pull-up-Widerstände des Controllers machen extra Widerstände überflüssig. Man kann an die Fernbedienung auch eine Matrix-Tastatur für den PIN-Code anschließen.

Die Fernbedienung kann entweder aus zwei AAA-Batterien oder einer Lithium-Knopfzelle versorgt werden. Das Schloß versorgt man besser über ein Netzteil oder eine beliebige Festspannung zwischen 1,8 und 5 V. Die LF-Version des Controllers verträgt allerdings maximal nur 3,3 V.

Funktionen

Der Ablauf startet, wenn die Fernbedienung den Code A6h sendet. Nach Empfang des Codes schickt das Schloß dann eine Zufallszahl mit 128 bit Länge. Eine Zufallszahl ist besser als eine voraussehbare Zahl, da sonst das Knacken des Schloßes durch Abhören ein Stück leichter wäre. Der Algorithmus zur Verschlüsselung ist gleichzeitig auch ein sehr guter Zufallsgenerator (siehe Kasten). Der Startwert kommt aus einem Zähler. Der Algorithmus macht daraus eine Zufallszahl und nutzt dazu einen extra Schlüssel. Der Zählerstand wird außerdem auch noch im Flash-Speicher abgelegt, damit nach einem Stromausfall immer noch echte Zufallszahlen generiert werden. Da der Flash-Speicher nur für 100.000 Schreiboperationen spezifi-

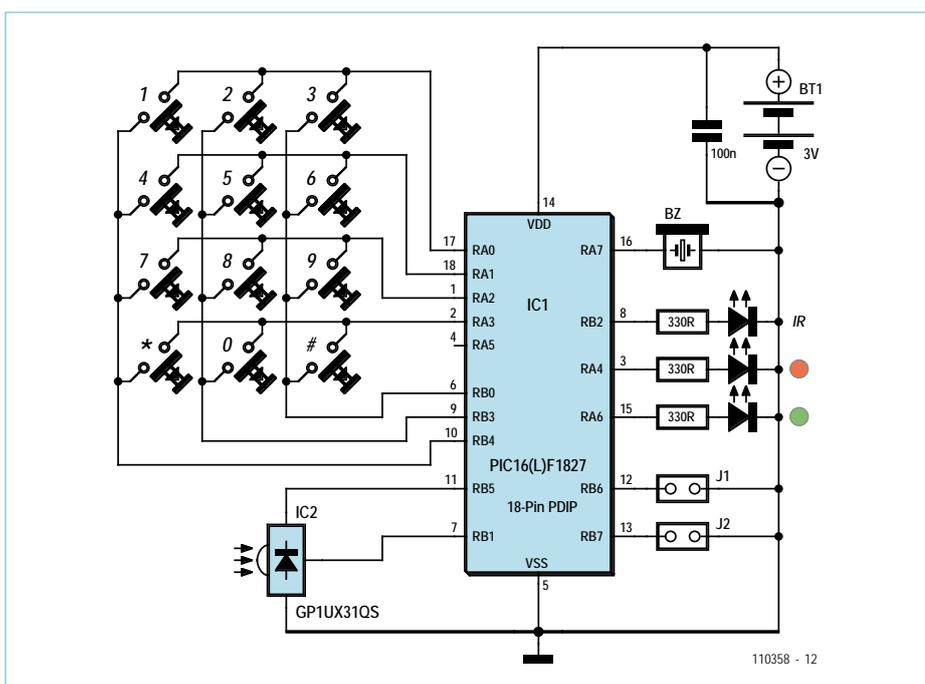


Bild 2. ...und die Schaltung der Fernbedienung.

Verschlüsselung als Zufallszahlen-Generator

Als Zufallszahlen-Generator wird normalerweise ein Schieberegister mit Rückkopplung eingesetzt. Der davon erzeugte Bitstrom kommt statistisch gesehen echtem Zufall sehr nahe. Leider ist die erzeugte Abfolge aber nicht wirklich zufällig, sondern berechenbar. Wenn der Algorithmus bekannt ist (zum Beispiel durch Analyse eines Datensatzes), dann können der Status des Schieberegisters ermittelt und die Zahlen vorausgesagt werden.

Doch auch ein guter Krypto-Algorithmus entspricht statistisch dem Zufall. Sogar das Verhältnis von „0“ und „1“ im Bitstrom beträgt 50 %. Bei unbekanntem Schlüssel ist zudem der Bitstrom nicht voraussehbar. Erst nach 2^{131} bit wird der Strom wiederholt (oder der Schlüssel wurde berechnet). Selbst bei einer Datenrate von 1 Gb/s wird es das Abermilliardenfache der bisherigen Existenzdauer des Weltalls brauchen, bis der Bitstrom einmal durch ist.

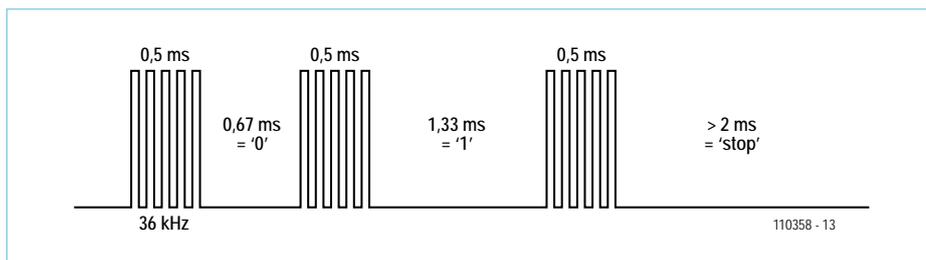


Bild 3. Für die IR-Übertragung der Daten eignet sich eine Variante des Sharp-Protokolls.

ziert ist, wird dieser Wert lediglich in einem von 65.536 Mal und zudem an wechselnden Orten gesichert. Im extrem unwahrscheinlichen Fall von Speicherdefekten (nach 13.000.000 Stromunterbrechungen oder 900.000.000.000-fachem Gebrauch) tritt eine Notfunktion in Kraft, die verhindert, dass man vor einem verschlossenen Schloss steht. Dann muss 16 Mal eine Taste der Fernbedienung betätigt werden und aus dem Zeitverhalten dieser Tastendrucke wird dann die neue Zufallszahl abgeleitet.

IR-Kommunikation

Die Fernbedienung empfängt die 128-bit-Zufallszahl. Standard IR-Module erlauben bei solch langen Übertragungen aber nur einen maximalen Duty-Cycle von 30%. Der verbreitete Manchester-Code (im RC5-Protokoll zum Beispiel) hat aber einen Duty-Cycle von 50 %. Aus diesem Grund wird das Sharp-Protokoll verwendet. Eine „1“ oder eine „0“ wird hier als Pause zwischen zwei Impulsen codiert. Eine Pause von 0,67 ms entspricht einer „0“ und eine Pause von 1,33 ms einer „1“. Der Impuls selbst hat eine Breite von 0,5 ms. Das Ende der Übertragung wird durch eine Pause von über 2 ms symbolisiert (siehe **Bild 3**). Das Timing ist locker und der Algorithmus selbstsynchronisierend, sodass keine hohen Anforderungen an die Genauigkeit des Oszillators gestellt werden. Dank des Stopp-Bits kann mit diesem Protokoll eine beliebige lange

Übertragung gemacht werden, was für die nötigen 128 bit ideal ist.

Sowohl Fernbedienung als auch Schloss führen anschließend eine Verschlüsselung (mit identischem Schlüssel) dieser Zufallszahl durch. Die Fernbedienung schickt ihr Resultat dann an das Schloss, das diese Zahl mit ihrer eigenen vergleicht. Stimmen die Zahlen überein, öffnet das Schloss. In Abhängigkeit von Jumper 1 schickt das Schloss eine Nachricht an die Fernbedienung, die darüber informiert, ob das Schloss offen ist (0xAB) oder die Sache nicht geklappt hat (0xB5). Die Rückmeldung des Resultats macht es theoretisch möglich, so ein Schloss automatisiert zu knacken. Allerdings ist der Zeitbedarf zu hoch und ein Knacken in der Praxis daher kaum möglich. Wer dieses minimale Risiko ausschließen will, der steckt Jumper 1 und es wird kein Resultat verschickt.

Jumper 1 der Fernbedienung hat eine ähnliche Funktion. Die Fernbedienung erzeugt nämlich eine optoakustische Rückmeldung, wenn der Code nicht passt oder das Schloss nicht reagiert. Mit gestecktem Jumper wird dies abgeschaltet.

Schlüssel-Genese

Jumper 2 aktiviert die Programmierfunktion des Schlüssels. Hierzu muss das Schloss aus und wieder eingeschaltet werden. Wenn man dann 32 Mal S1 betätigt, werden zwei

Schlüssel erzeugt. Die rote LED leuchtet, wenn dies erfolgt ist. Auch hier wird das Zeitverhalten beim Drücken von S1 erfasst und daraus die Basis eines zufälligen Schlüssels erzeugt. Diese Zahlen werden dann im EEPROM abgelegt. Anschließend muss man gleich die Fernbedienung programmieren. Hierzu muss auch auf der Fernbedienung Jumper 2 gesteckt werden. Die Fernbedienung sendet dann (eventuell nach Betätigung der „#“-Taste oder der PIN-Eingabe) den Code „0xAD“. Das Schloss schickt dann zweimal den Schlüssel. Die Fernbedienung überprüft dann, ob die beiden Zahlen identisch sind und schreibt im Erfolgsfall diesen Schlüssel ins eigene EEPROM. Dabei leuchtet dann die grüne LED und es ertönt ein Quittungston. Dies kann jederzeit wiederholt werden. Durch Abziehen des Jumpers und erneutes Aus- und Einschalten versetzt man das Schloss dann in den normalen Betrieb, und die Fernbedienung kann ab sofort verwendet werden.

Aus Sicherheitsgründen kann der Schlüssel nur im Schloss generiert und von diesem versendet werden. Es kann daher auf keinen Fall später noch heimlich eine zweite Fernbedienung mit dem gleichen Schlüssel versehen werden, denn es werden ja in jedem Fall neue Schlüssel generiert. Versucht man dies, wird die alte Fernbedienung gesperrt und die Sache fliegt auf.

Anschließend ist sowohl beim Schloss als auch bei der Fernbedienung der Datenschutz des EEPROMs aktiviert. Der generierte Schlüssel kann also auch nicht mit elektronischem Know-how ausgelesen werden. Auch bei der Erzeugung bleibt der Schlüssel unbekannt, da der Anwender ja gänzlich uninformatiert lediglich ein paar Tasten drückt. Der erzeugte Schlüssel bleibt deshalb versteckt im EEPROM abgelegt. Eine Gefahr besteht noch, dass das System direkt, quasi jungfräulich und ohne Erstellung eines Schlüssels eingesetzt wird. In beiden EEPROMs stecken nämlich ab Fabrik durch die Bank die Werte „0xFF“. Von daher ist der Schlüssel von Fernbedienung und Schloss gleich und die Sache funktioniert. Der Anwender könnte also meinen, dass er keinen Schlüssel erstellen müsste. Dann aber muss der böse Schlossknacker

es lediglich mit der Byte-Folge FF...FF probieren und schon ist er am Ziel. Um dies zu verhindern, inkrementiert die Fernbedienung den aus dem EEPROM ausgelesenen Schlüssel. Damit sind die Schlüssel von Schloss und Fernbedienung nicht mehr gleich und der Anwender muss zwingend neue Schlüssel generieren. Um die Inkrementierung zu kompensieren, wird der Schlüssel vor der Ablage im EEPROM entsprechend dekrementiert.

PIN-Code

An der Fernbedienung ist der Anschluss einer Tastatur zur PIN-Eingabe vorgesehen. Standardmäßig ist der PIN-Code ausgeschaltet. Wenn kein PIN-Code gewünscht ist, muss lediglich ein Taster zwischen RA3 und RBO angeschlossen werden (entspricht der Taste „#“ der Tastatur). Der PIN-Code kann durch Stecken von Jumper 2 eingestellt werden, wenn man anschließend die Taste „*“ betätigt (oder den aktuellen PIN eingibt) und danach zweimal hintereinander den neuen PIN eingibt. Um den PIN-Code zu deaktivieren muss ein neuer PIN-Code programmiert werden, wobei als PIN „####“ oder „*****“ eingegeben werden muss. Bei eingestelltem PIN-Code muss dieser beim Einschalten der Fernbedienung eingegeben werden. Wird drei Mal der falsche Code eingegeben, bleibt die Fernbedienung bis zur Erzeugung eines neuen Schlüssels blockiert, wozu man die Fernbedienung dann neu mit dem Schloss synchronisieren muss.

An/Aus

Vielleicht ist schon aufgefallen, dass die Fernbedienung über keinen Ein/Aus-Schalter verfügt. Das „Problem“ mit neueren Controllern ist nämlich, dass deren Stromverbrauch so gering ist, dass der stromlos gemachte Controller nicht schnell abschaltet, sondern aus dem Entstörkondensator noch eine Weile versorgt wird. Dem wurde begegnet, indem die Fernbedienung nach 5 s ohne Aktion in den Sleep-Mode geht und dann so gut wie keinen Strom zieht (sehr viel weniger als es der Selbstentladung von Batterien entspricht). Die „#“-Taste sorgt für einen Interrupt und weckt den Controller wieder auf.

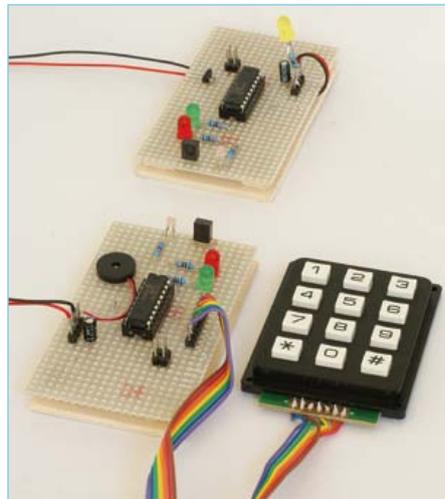


Bild 4. Beide Schaltungen können leicht auf einem Stück Lochrasterplatine aufgebaut werden.

Software

Der Autor hat Open-Source-Code für AES in der Sprache C++ (die Quelle steht im Code) nach Assembler umgesetzt, da der C-Compiler nicht optimal für diese Aufgabe war. Der Code für Schloss und Fernbedienung befindet sich in einer Datei, da weite Teile gleich sind. Am Anfang der Datei sorgt dann ein „#define remote“ oder „#define home-station“ für den passenden Hex-Code. Der Code kann natürlich verändert werden. IR-LED und -Sensor sind am seriellen Port (TX/RX) angeschlossen. An diesem Port könnte auch ein Mobilfunk-Modem angeschlossen werden. In diesem Fall könnte das Schloss durch Versenden einer SMS geschaltet werden.

Bedienung

Verglichen mit der Programmierung ist die Bedienung einfach: Ein Druck auf die „#“-Taste der Fernbedienung schaltet diese ein. Während die LEDs schnell blinken, muss der PIN-Code eingegeben werden. Nach drei Fehleingaben blinkt die rote LED dauerhaft, und die Fernbedienung benötigt einen neu generierten Schlüssel.

Es gibt drei mögliche Resultate:

- Hoher Ton & grüne LED – tiefer Ton & rote LED: Schloss reagiert nicht (ist

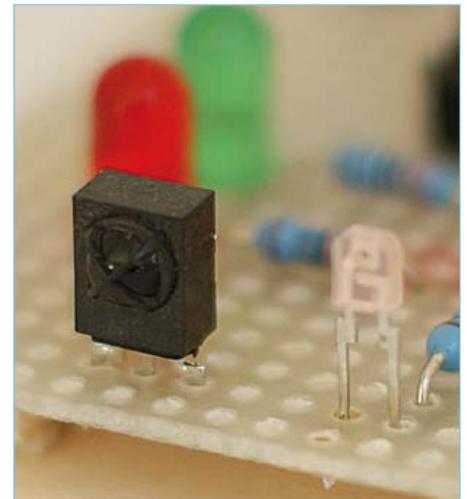


Bild 5. IR-LED und IR-Empfänger stehen einander so gegenüber, dass sie einander „sehen“ können.

- möglicherweise zu weit entfernt etc.).
- Hoher Ton & grüne LED – hoher Ton & grüne LED – tiefer Ton & rote LED: Falscher Schlüssel, Schloss öffnet nicht (möglicherweise ist die Fernbedienung auch blockiert).
- Hoher Ton & grüne LED – hoher Ton & grüne LED – hoher Ton & grüne LED: Schloss öffnet.
- Wenn Jumper 2 gesteckt ist, wird das Resultat in den letzten Fällen „hoher Ton & grüne LED – hoher Ton & grüne LED“ sein. Die einzige Rückmeldung ist dann das Schloss selbst, das öffnet - oder auch nicht.
- Innerhalb von 5 s kann nach Eingabe von „#“ eine neue Aktion durchgeführt werden, ohne dass man erneut den PIN eintasten muss.
- Nach 5 s ohne Aktion schaltet die Fernbedienung ab.

(110358)

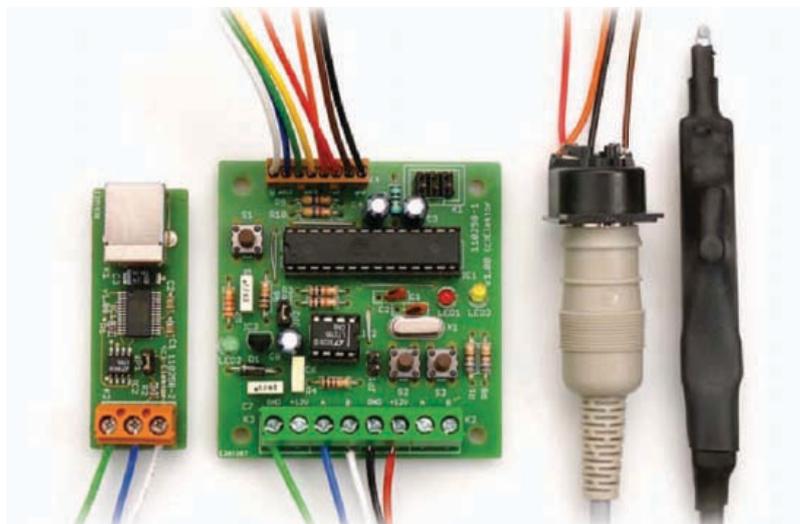
P.S. Falls jemand es schaffen sollte, den Schlüssel zu extrahieren, so möge er sich unter redaktion@elektor.de melden...

Hier kommt der Bus (10)

Messwerte erfassen mit 22 bit

Von Jens Nickel

Jetzt wird's hochauflösend: An unseren Busknoten schließen wir einen externen ADC an, der Werte mit 22 bit sampeln kann. Dank der SPI-Schnittstelle ist das Einlesen der Daten in den Mikrocontroller recht einfach. Auch die Anzeige auf dem PC ist schnell erstellt; wir müssen nur die HTML-Seite aus dem letzten Teil der Serie etwas anpassen.



Das Schöne an der Elektronik ist, dass man sich leicht von anderen Projekten inspirieren lassen kann. In diesem Fall war es das Leserprojekt „Gradwanderung“ aus dem letzten Heft [1]. Um Temperaturänderungen hochge-

nau zu messen, hatte unser Leser Dr. Dietmar Schröder eine Messschaltung rund um einen externen ADC (MCP3551 von Microchip) aufgebaut, der Spannungswerte mit 22 bit Auflösung in Daten wandeln kann. **Bild 1** zeigt die kleine Schaltung

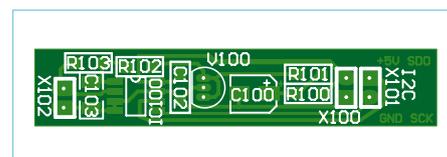


Bild 2. Platine von Dietmar Schröder. X100 und X101 sind mit K4 des Experimental-Knotens zu verbinden.

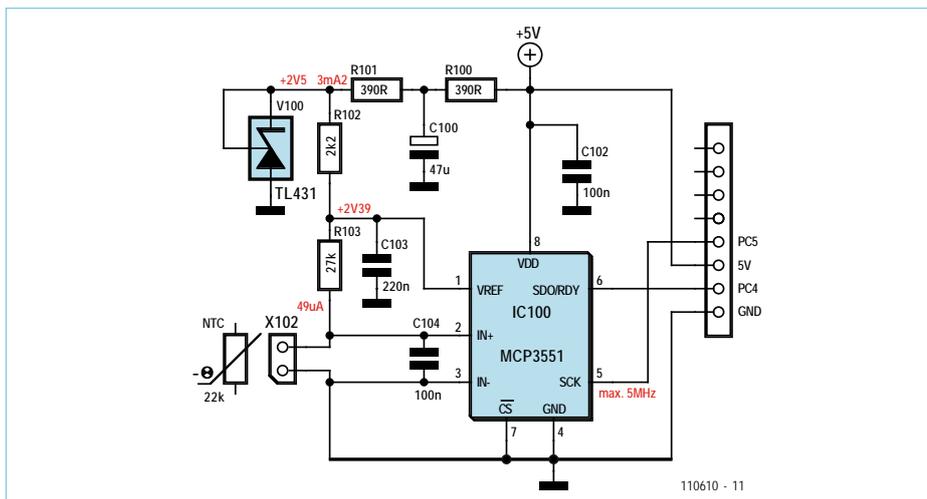


Bild 1. Der Schaltplan des hochauflösenden Temperatursensors, mit einem Anschluss für unseren Experimental-Knoten.



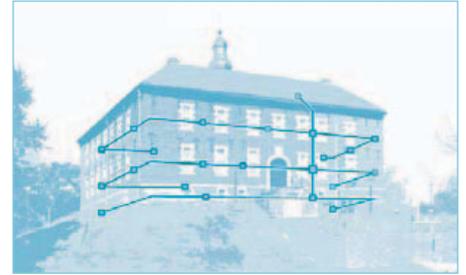
Bild 3. Anzeige der ADC- und Temperaturwerte im *ElektorBusBrowser*.

Elektor Produkte & Service

- Experimental-Knoten (Platine 110258-1 oder 3er-Set Platinen 110258-1C3)
- USB/RS485-Konverter (fertig aufgebaut und getestet 110258-91)

- Gratis Software-Download (Controller-Firmware plus PC-Software)

Alle Produkte und Downloads sind über die Website zu diesem Artikel erhältlich: www.elektor.de/110610



mit diesem Delta-Sigma-Wandler, dem NTC und einigen weiteren Bauteilen.

In **Bild 2** ist eine passende Platine des Autors zu sehen, die unter [1] heruntergeladen werden kann. Der A/D-Konverter stellt an seinem digitalen Ausgang – einem SPI-Interface - etwa 14 Mal pro Sekunde einen Messwert bereit. Wie geschaffen war diese Schaltung zum Anschluss an einen Bus-Experimentalknoten. Über zwei Pins unseres achtpoligen Steckverbinders [2] können wir die Messwerte auslesen, über zwei weitere Pins versorgen wir die Sensorschaltung. Und da beim „Temperaturänderungsmessgerät“ ebenfalls ein ATmega zur Aufbereitung der Daten genutzt wurde, lag auch schon eine gute Code-Grundlage vor (ebenfalls downloadbar unter [1]).

Gebangte Bits

Die SPI-Schnittstelle wird dabei per „Bit-Banging“ bedient, also manuell per Software und nicht über das Hardware-SPI-Interface des Controllers. Die zuständige C-Funktion des Original-Codes konnte ich rasch in BASCOM umsetzen (**Listing 1**). In `Readexternaladc()` wird zuerst gewartet, bis der ADC den SDO-Pin auf Low legt. Dies ist das Zeichen, dass ein neuer Messwert abgerufen werden kann. Dann zieht der Controller die Leitung SCK abwechselnd auf High und Low. Nach jeder steigenden Flanke liegt an der SDO-Leitung ein Bit des digitalisierten Messwertes an, wobei höherwertige Bits zuerst übertragen werden. Die Long-Variable `Dat` dient zur Aufnahme des Ergebnisses. Falls SDO high ist, wird ein Bit=1 ganz rechts zum Wert addiert, alle Bits in `Dat` werden daraufhin um eine Stelle nach links geschoben. Im Datenblatt [3] ist nachzulesen, warum man 24 Bits einlesen muss und zum Schluss zur Sicherheit noch ein Takt extra folgen sollte. Wichtig für uns noch: Der ADC wird in der Schaltung im „Continuous Conversion Mode“ betrieben. Wir müssen nicht mehr tun, als die Spannung anzulegen, damit es mit den Messungen losgeht.

Wie im Artikel des Originalprojektes beschrieben ist, sind durch das unvermeidliche Rauschen nur etwa 19 Bits signifikant (was dennoch eine stolze Auflösung von etwa 3/10000 Grad ermöglicht). Wir können im Ergebnis also problemlos die zwei niedrigwertigen Bits durch Rechts-Verschieben verfallen lassen. Es bleiben 20 Bits, wobei das erste Bit im Temperaturbereich über -35°C immer 0 ist. Das passt perfekt zu unserem *ApplicationProtocol*, wo Ganzzahlen bis 19 bit (plus Vorzeichen) übertragen werden können [4]. In dem von [5] downloadbaren BASCOM-Code ist leicht nachzuvollziehen, wie die 19 Bits vor dem Übertragen auf drei Daten-Bytes verteilt werden. Ein erster Test fiel nicht schwer, da uns der Autor seine Sensorplatine noch eine Weile überlassen hatte. Ich brauchte jetzt nur noch ein User-Interface für den PC. Einfach ließ sich dieses mit dem im letzten Heft vorgestellten Konzept verwirklichen. Es basiert auf HTML-Seiten, die in einem *ElektronBusBrowser* dargestellt werden. Die HTML-Seite „Index.htm“ aus dem letzten Teil der Serie musste ich nur dahingehend anpassen, dass statt einem „VALUE2“ nun ein „VALUE4“ empfangen und in der Textbox dargestellt wurde.

User-Interface

Einer der Busknoten hatte sich damit in ein hochauflösendes Messgerät verwandelt, schon die Annäherung der Hand ließ die ADC-

Listing 1: BASCOM-Code zum Auslesen des ADC-Wertes

```
Function Readexternaladc() As Long

    Dat = 0
    Sck = 1
    Notimeout = 100

    While Sdo = 1 And Notimeout > 0
        Notimeout = Notimeout - 1
        Waitms 1
    Wend

    For Ia = 0 To 23
        Sck = 0
        Waitus 80

        Shift Dat , Left , 1

        Sck = 1
        If Sdo = 1 Then
            Dat = Dat + 1
        End If
        Waitus 80

    Next

    Sck = 0
    Waitus 80
    Sck = 1
    Waitus 80

    If Notimeout = 0 Then
        Readexternaladc = 0
    Else
        Readexternaladc = Dat
    End If

End Function
```

Werte purzeln. Natürlich wollte ich nun auch die Temperaturwerte in Grad Celsius angezeigt bekommen. In der Firmware für den Sensorknoten implementierte ich eine entsprechende Berechnungsroutine, wobei ich die NTC-Kennlinie (in 5°-Schritten) aus dem Code des „Temperaturänderungsmessgeräts“ übernommen und entsprechend angepasst habe.

Meine Demo-Firmware kommt übrigens ohne Fließkommazahlen und Divisionen aus. Als Ergebnis erhält man wahlweise Werte in 1/10000 oder 1/1000 Grad, die ebenfalls als 4-Byte-Ganzzahl übertragen werden können (Anmerkung: Die absolute Genauigkeit der Temperaturwerte kann ohne Kalibrierung leider nicht besonders hoch sein).

Listing 2: Script in der HTML-Seite

```

var SetFlag = false;
var QuantityToSet = 0;
var ScaleToSet = 0;
var DisplayScale = 0;

function ProcessPart(part)
{
    if ((part.Sender == 2) && (part.Parttype == PARTTYPE_VALUE4))
    {
        if (part.Channel == 0) {TextboxSetvalueScaled(,ADC', part.Numvalue, DisplayScale);};
    }

    if (SetFlag==true)
    {
        if ((part.Sender == 2) && (part.Parttype == PARTTYPE_SCALE) && (part.Channel == 0))
        {
            if (QuantityToSet==TEMPERATURE) {TextSetvalue(,unit', '°C');};
            if (QuantityToSet==RAWVALUE) {TextSetvalue(,unit', 'ADC-Value');};
            DisplayScale = ScaleToSet;
            SetFlag = false;
        }

        var parts = InitParts();
        parts = SetScale(parts, 10, 2, 0, 0, QuantityToSet, 0, ScaleToSet);
        SendParts(parts, true);
    }
}

function SetSensorScaleIndirect(quantity, scale)
{
    SetFlag = true;
    QuantityToSet = quantity;
    ScaleToSet = scale;
}

```

Die Umschaltung (ADC-, 1/1000 oder 1/10000-Grad-Werte) erfolgt natürlich über Buttons in der HTML-Benutzeroberfläche (siehe **Bild 3**). Damit können wir noch mal schön demonstrieren, wie die physikalische Größe, die Einheit und die Skalierung auf einem Sensor gesetzt werden kann. Die zu sendenden Kommando-Bytes lauten in Dezimaldarstellung 40, 193, 33, -4 für eine Anzeige in 1/10000 Grad (genutzt wird Channel 0 des Knotens).

Dank der Javascript-Bibliothek JSBus müssen wir die Bytes nicht von Hand ausrechnen, sondern können einfach mit folgenden Script innerhalb unserer HTML-Seite einen entsprechenden *Part* generieren und versenden:

```

var parts = InitParts();
parts = SetScale(parts, 10, 2,
0, 0, TEMPERATURE, 0, -4);
SendParts(parts, true);

```

Zuverlässige Übertragung

Die obigen Codezeilen ließ ich zuerst direkt aufrufen, wenn der entsprechende Button im HTML-Formular gedrückt wurde, genauso wie das im letzten Teil der Bus-Serie beschrieben ist [6]. Bei Tests fiel mir aber auf, dass die Klicks nicht immer zum gewünschten Erfolg führten. Die *Out-Command*-Textbox im ElektorBusBrowser zeigte aber, dass die Daten-Bytes der Message korrekt berechnet worden waren. Es handelt sich also vermutlich um ein Problem der Knoten-Firmware. Meine

Vermutung ist, dass die Message vom Sensor mitunter nicht korrekt empfangen werden kann, wenn dieser mit dem Auslesen des externen ADCs beschäftigt ist. Interessanterweise wird die Übermittlung des Kommandos zuverlässiger, wenn `intFreeBusTime` im Quellcode des ElektorBusBrowsers heruntersgesetzt wird (strenggenommen brauchen wir bei dieser Anwendung natürlich gar keine *Free-BusPhase* mehr, daher soll später noch eine Möglichkeit implementiert werden, diese ebenfalls von Javascript aus steuern zu können). Da wir noch über keinen Bus-Monitor und eine Möglichkeit zum Firmware-Debugging verfügen, konnte ich dem Fehler leider nicht weiter nachgehen. Doch bietet dieser eine gute Gelegenheit,

noch einmal zu demonstrieren, wie man solche Probleme umschiffen kann.

Listing 2 zeigt eine Lösung. Die Buttons zum Umstellen der Einheit rufen eine Javascript-Routine namens `SetSensorScaleIndirect(...)` auf, diese versendet aber nicht direkt den entsprechenden Kommando-Part, sondern setzt nur ein Flag. So lange dieses Flag gesetzt ist, wiederholt der Master das Versenden des Kommandos. Die dazu nötigen Codezeilen können wir in der Routine `ProcessPart(part)` unterbringen, die ja periodisch von der Library `JSBus` aufgerufen wird, wenn ein Messwert ankommt.

Das Flag wird gelöscht, wenn eine Bestätigung des Knotens empfangen wird (die `BASCOM`-Firmware packt die nötigen vier Bytes einfach noch zusätzlich zum Messwert in die regelmäßig ausgesandte Message). Erst dann wird auch die Anzeige der Einheit auf der HTML-Seite geändert. Zur Dezimalkomma-Darstellung des Wertes in der Textbox wird die Funktion `TextboxSetvalueScaled(...)` verwendet (in der neuen Version von `JSBus` implementiert [5]). Im downloadbaren HTML-File kommt schließlich noch etwas `CSS` [7] zum Einsatz, damit die Benutzeroberfläche etwas hübscher aussieht.

Kleine „Optimierung“

Bei weiteren Tests stellte sich heraus, dass die Anzeige mitunter nach einigen Minuten einfror. Mit einem eilig aus dem Labor geholten Oszi konnte ich feststellen, dass aber noch Bits über den Bus flitzten; und zwar so lange, bis ich den Scheduler anhielt. Daraufhin ließ ich mir alle Bus-Bytes in einer separaten Textbox anzeigen – und siehe da, es wurden kontinuierlich nur noch *FreeBus-Messages* des Schedulers versandt. Was da schief lief, war nicht ganz leicht zu ergründen, denn insgesamt laufen beim Scheduling drei Prozesse nebeneinander ab. Erstens die Scheduler-Schleife selbst, welche die Knoten hintereinander aufruft. Zweitens die Routine `ShowMessage`, die parallel abgearbeitet wird, wenn eine 16-Byte-Message ankommt; falls diese Nachricht vom aufgerufenen Knoten stammt, kann der nächste Knoten an die Reihe kommen. Und drittens mischt auch noch ein Timer mit; denn wenn sich der Busteilnehmer nicht innerhalb einer

gewissen Zeit meldet, wird ebenfalls zum nächsten Knoten gewechselt. Schließlich kam ich darauf, dass ich einen groben Fehler gemacht hatte. Um anzuzeigen, dass der nächste Knoten an die Reihe kommen soll, hatte ich allen Routinen gestattet, direkt die globale Variable `intPolledNodesCursor` zu ändern, die auf den nächsten Knoten innerhalb der Scheduling-Liste zeigt. Falls diese Variable aber von außen geändert wurde, während die Scheduler-Schleife gerade bei der Arbeit war, konnte es zu schwierigen zu debuggenden Seiteneffekten kommen. Ich änderte den Code daher so ab, dass nun ein Flag namens `boolNextNode` verwendet wird, um zum nächsten Knoten weiterzuschalten. Dieses Flag wird nur zu Beginn der Scheduler-Schleife abgefragt und hat ansonsten keinen Einfluss auf die Code-Ausführung. Jetzt lief es stabil! Glücklicherweise waren auch die Files für den vorangegangenen Teil der Serie noch nicht auf unsere Website hochgeladen worden. Im Download unter [5] findet man also schon den verbesserten `ElektorBusBrowser`, als `VB.NET`-Quellcode und `.exe`-Datei.

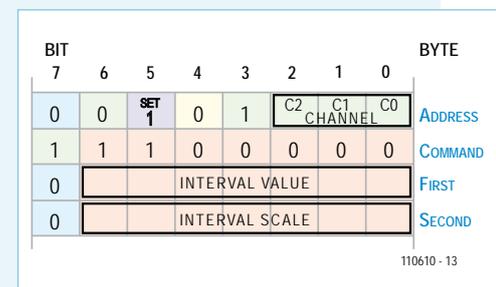
Noch ein paar Ideen

Gerne hätte ich noch weitere Features verwirklicht, etwa die Anzeige der Temperaturänderung, samt einer Filterung mit einstellbaren Parametern. Damit hätten wir einen vollwertigen Ersatz für die Prozessorplatine, das Display und die Einstellpotis von Dietmar Schröders Original-Projekt schaffen können. Leider war meine Entwicklungszeit diesmal recht begrenzt - und wurde noch weiter geschmälert durch ein kleines Unglück mit meinem `USB/RS485-Konverter` (siehe `Labcenter` in diesem Heft). Interessierte Leser dürften das Projekt aber ohne große Hindernisse fertig stellen können - mit etwas `C-Knowhow`, dem Quellcode des „Temperaturänderungsmessgeräts“ und den bisher vorgestellten `Bus-Tools`.

In Gedanken hatte ich außerdem, den `ElektorBusBrowser` so zu erweitern, dass er auch ohne Scheduling Nachrichten von Knoten empfangen konnte. Im Prinzip unterhalten sich bei dieser Anwendung ja nur noch zwei Teilnehmer miteinander. `RS485` und die beschriebenen Protokolle würden sich jedenfalls auch sehr gut zur Fernsteuerung

Mess-Intervall setzen

Die Fernsteuerung eines einzigen Messgeräts oder Sensors kommt im Prinzip ohne Scheduling aus. Man könnte dem Gerät mitteilen, wie oft ein neuer Messwert übermittelt werden soll, etwa für `Datenlogger-Anwendungen`.



Die Grafik zeigt das entsprechende 4-Byte-Kommando des `ApplicationProtocols`. Der Intervall-Zahlenwert (`Interval Value`) wird über sieben Bits codiert. Bei der Intervall-Skala bedeuten:

Hex	Dez	Intervall
04	4	µs
05	5	1/100.000 s
06	6	1/10.000 s
07	7	ms
08	8	1/100 s
09	9	1/10 s
0A	10	1 s
0B	11	10 s
0C	12	100 s
10	16	1 min
11	17	10 min
12	18	100 min
18	24	1 h
19	25	10 h
20	32	1 d
21	33	10 d
22	34	100 d
28	40	1 Monat
30	48	1 y
31	49	10 y

Checksumme und Zuverlässigkeit

Bei der Entwicklung eines Bussystems kann man sich gar nicht genug Gedanken darüber machen, wie man unter allen Umständen eine zuverlässige Übermittlung der Nachrichten gewährleistet. Im letzten Teil der Serie hatten wir ja schon zwei Acknowledge-Mechanismen vorgestellt. Einmal auf Message-Ebene – hauptsächlich gedacht, um Kollisionen festzustellen, die während der unsicheren *FreeBusPhase* auftreten können. Und einmal auf der Ebene des ApplicationProtocols, gedacht für sonstige Störungen bei der Übermittlung oder Verarbeitung (auch bei den kollisionsfrei abgefragten Knoten, so wie in dieser Anwendung, siehe Text). In beiden Fällen schickt der Empfänger die empfangenen Bytes einfach zum Sender zurück, ein kleines Flag-Bit dient dabei jeweils zur Unterscheidung von der Originalnachricht. Diese Methodik schien mir einigermaßen sicher zu sein, weswegen ich bisher auch auf die Implementierung einer CRC oder Checksumme verzichtet hatte.

Glücklicherweise verfügen wir aber mittlerweile über viele mittendekende Bus-Mitstreiter. Elektor-Leser Werner Koch gab zu bedenken, dass dieser Mechanismus nicht ausreicht. Zwar kann man so – falls Messages verloren gehen – den Sender zu einer Wiederholung veranlassen. Doch ist man nicht dagegen gefeit, dass eine Störung auch einmal eine Message „generieren“ kann, die man gar nicht senden wollte. Zwar würde zum Beispiel ein Aktor eine entsprechende *AcknowledgeMessage* zum Master zurücksenden; damit könnte dieser erkennen, dass etwas schief gelaufen ist. Doch hat der Aktor dann bereits das Relais geschaltet; resultierende Unglücke mag sich ein jeder selbst ausmalen.

Man kann dieses Problem lösen, indem der Aktor abermals auf eine Bestätigung der Steuerung wartet, bevor das Relais geschaltet wird (3-way-handshake). Eine andere Möglichkeit bietet das Übertragen redundanter Daten – nur wenn die Checksumme zu den übrigen Bits passt, schaltet der Aktor das Relais. Damit macht man sich zunutze, dass das zufällige Entstehen einer solchen gültigen Nachricht viel unwahrscheinlicher ist als bei einer nicht gesicherten Message.

Nun entbrannte eine Diskussion, ob wir eine 16-bit-CRC oder eine einfache Checksumme benutzen wollten. Mein Vorschlag, auch in den hierfür vorgesehenen letzten beiden Bytes der Message den Wert AA_{hex} zu verbieten, wurde von einigen Teilnehmern unserer

Mailing-Liste vehement abgelehnt; doch war der Vorteil unbestreitbar, dass wir dann unsere sehr einfache Synchronisation beibehalten konnten. Die andere Seite favorisierte aber eine 16-bit-CRC und eine fortgeschrittene Synchronisation. Unter anderem könnte man gerade die CRC dazu verwenden, um festzustellen, dass eine Nachricht an dieser Stelle endete.

Schließlich machte ich den salomonischen Vorschlag, beide Möglichkeiten zuzulassen. Zur Unterscheidung verwenden wir ein Bit unseres Modebytes:

Bit	1	0
7	keine ID-Bytes, Daten ab Byte 2	ID-Bytes ab Byte 2
6	Byte 2 und 3 sind ID-Bytes	Byte 2 bis 5 sind ID-Bytes
5	keine CRC/Checksumme	Byte E und F sind 16-bit-CRC/Checksumme
4	Fortgeschrittene Synchronisation	Kein AA _{hex} ab Byte 2
3	letztes ID-Byte ist Fragmentnummer	Alle ID-Bytes für Adressierung
2	höchste sechs Adressbits für Bus-Segment	keine Segment-Adresse
1	AcknowledgeMessage	Original-Nachricht
0	AcknowledgeMessage erwartet	nicht erwartet

Mindestens eine der Gültigkeitsprüfungen soll in der C-Bibliothek für AVR-Controller umgesetzt werden, die gerade am Entstehen ist.

Etwas später hatte ich noch die Idee, die Redundanz innerhalb der Datenbytes zu verwirklichen: So könnte man ein wichtiges, zwei oder vier Byte umfassendes Kommando auch mehrmals hintereinander innerhalb derselben Message übermitteln, zum Beispiel zwei oder sogar vier Mal. Bei nicht-periodischen Störungen wäre die Wahrscheinlichkeit, dass eine solche Nachricht aus dem Nichts entsteht, nahezu Null.

eines Messgeräts eignen. Da der Sensor dann nicht mehr regelmäßig durch einen Scheduler abgefragt wird, könnte man ihn anweisen, von sich aus Messwerte in einem bestimmten Intervall zu übermitteln. Der Kasten zeigt, wie ein solches Intervall-Kommando mit Hilfe des ApplicationProtocol codiert werden kann.

In der Pipeline haben wir aber noch weit mehr: Im nächsten Heft soll zuerst einmal die versprochene Anbindung an ein Android-Smartphone erfolgen, und man darf auf Weiteres gespannt sein!

(110610)

Entwickeln Sie mit! Hinweise, Ideen und eigene Anwendungen sind willkommen unter [redaktion@elektor.de!](mailto:redaktion@elektor.de)

Weblinks:

- [1] www.elektor.de/110151
- [2] www.elektor.de/110258
- [3] <http://ww1.microchip.com/downloads/en/DeviceDoc/21950e.pdf>
- [4] www.elektor.de/110428
- [5] www.elektor.de/110610
- [6] www.elektor.de/110517
- [7] http://de.wikipedia.org/wiki/Cascading_Style_Sheets

Oft kopiert – doch nie erreicht:

	PCB-POOL [®] Beta LAYOUT	Basista	Euro-circuits	Leiton	WEdirekt	multi-cb
 Leiterplatten online kalkulieren	✓	✓	✓	✓	✓	✓
 FREE STENCIL	✓	—	—	—	—	—
 Bestückung online	✓	—	—	—	—	—
 Kostenlose Layoutsoftware	✓	—	—	—	—	—
 Bewertungs-Rabattsystem	✓	—	—	—	—	—
 Akzeptierte Layoutformate	16	6	1	3	5	3
 Kollisionsprüfung zum Anfassen	✓	—	—	—	—	—
 Auftragsverfolgung mit Ansprechpartner	✓	—	—	—	—	—
 Watch"ur"PCB	✓	—	—	—	—	—
 Pünktlich oder kostenlos	✓	—	—	✓	—	—
 8h-Eilservice	✓	✓	—	—	—	—
 Online Daten-Restore Service	✓	—	—	—	—	—

Hi Michar,
 wenn du einfach nur
 billig suchst probier mal:

 www.Jackaltac.com

Das Original seit 1994!

www.pcb-pool.com

• **Subscribe** to *audioXpress* magazine!

Do your **electronics speak** to you? Are the words **"audio"**, **"vacuum tubes"**, and **"speaker technology"** music to your ears?

Then you should be **reading *audioXpress*!**

Recently acquired by The Elektor Group, *audioXpress* has been providing engineers with incredible audio insight, inspiration and design ideas for over a decade. If you're an audio enthusiast who enjoys speaker building and amp design, or if you're interested in learning about tubes, driver testing, and vintage audio, then *audioXpress* is the magazine for you!

What will you find in *audioXpress*?

- In-depth interviews with audio industry luminaries
- Recurring columns by top experts on speaker building, driver testing, and amp construction
- Accessible engineering articles presenting inventive, real-world audio electronics applications and projects
- Thorough and honest reviews about products that will bring your audio experiences to new levels

Choose from print delivery, digital, or a combination of both for maximum accessibility.

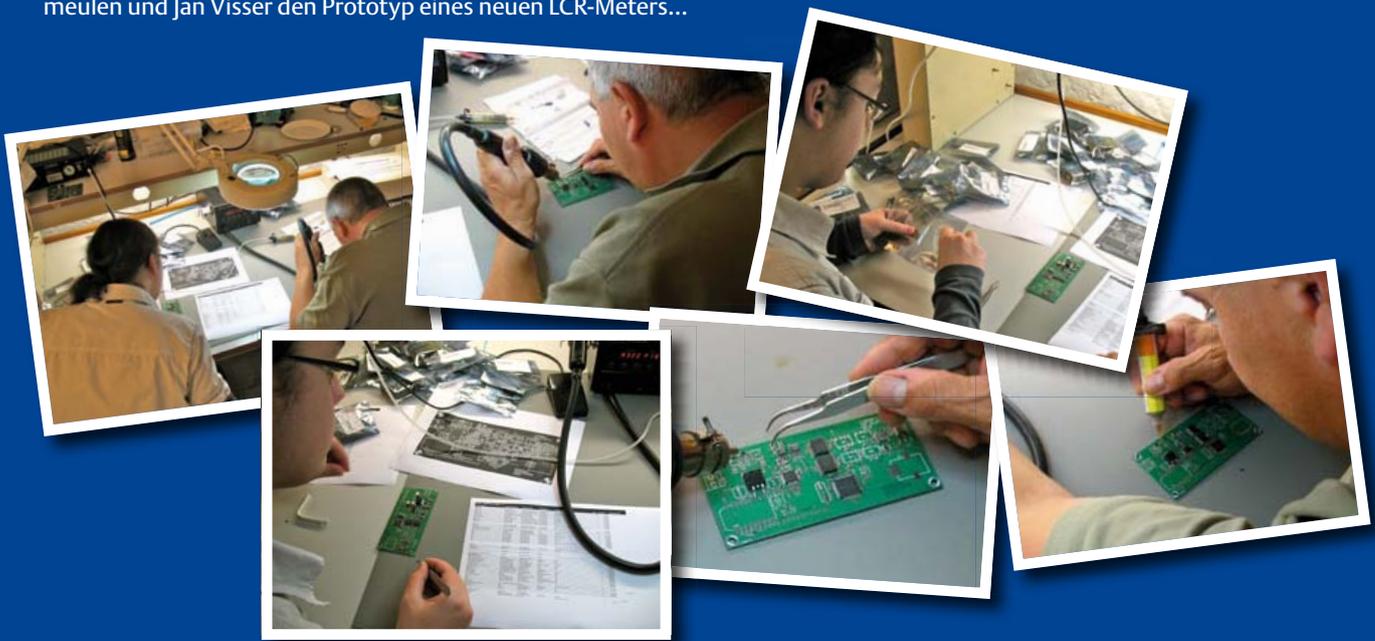
Subscribe to *audioXpress* at www.cc-webshop.com today!

audioXpress

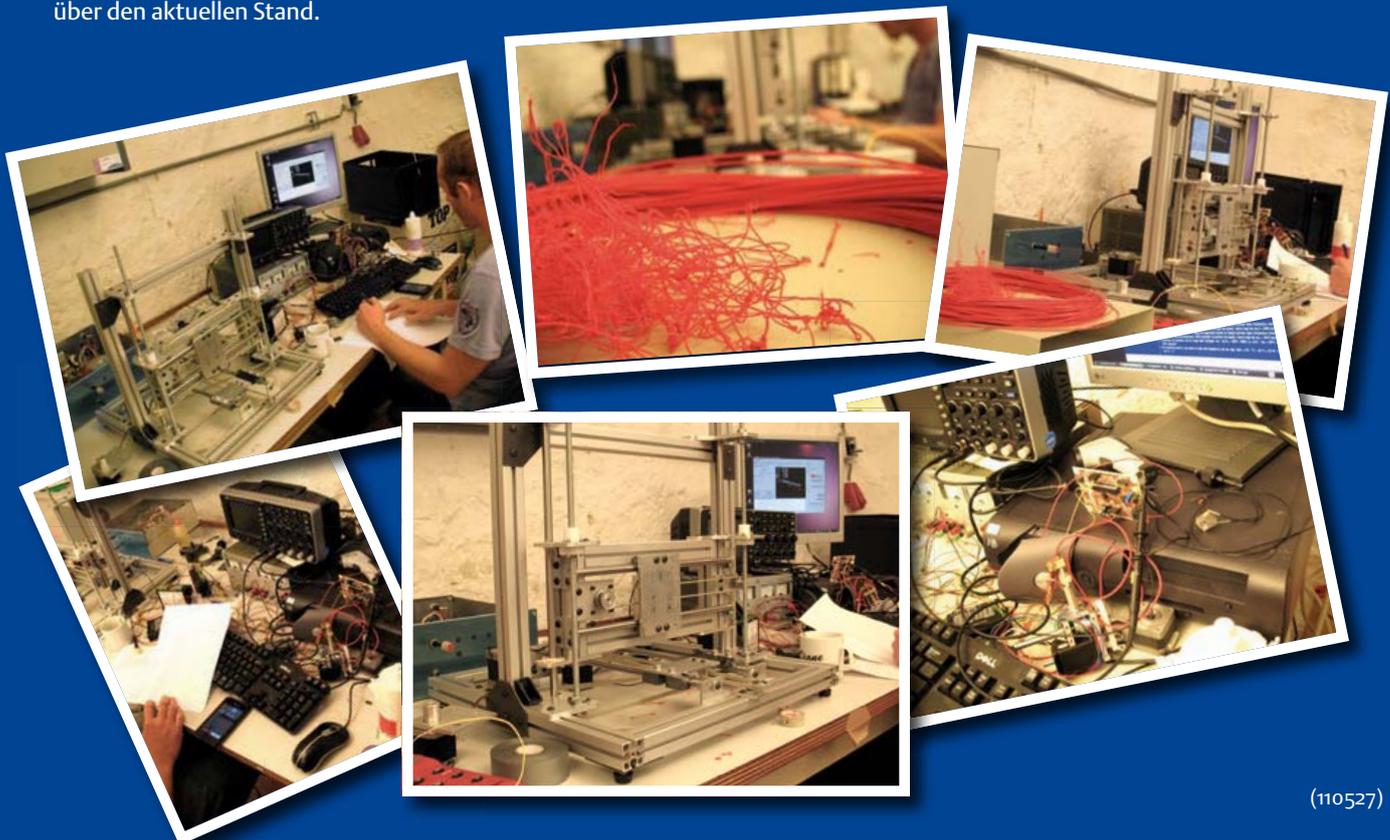


Vorsicht, Baustelle!

Meistens verschanzen sich die Mitarbeiter des Elektor-Labors wie in einer Festung, bei der die Brücken hoch gezogen sind. Nur am alljährlichen „Tag des Offenen Denkmals“ stehen dem Publikum die Tore weit offen. Schließlich gehört das Schloss, in dem Elektor residiert, zu den geschützten Baudenkmalern. Trotzdem dringt hin und wieder etwas von dem nach außen, was hinter geschlossenen Labortüren geschieht. Dies sind einige Blicke durchs Schlüsselloch auf zwei zukünftige Projekte. Zuerst bauen Raymond Vermeulen und Jan Visser den Prototyp eines neuen LCR-Meters...



Die zweite Fotostrecke zeigt den hoch motivierten Einsatz unseres Laborleiters Christian Vossen, der einen 3D-Drucker entwickelt. Bis zur Veröffentlichung wird noch etwas Zeit vergehen, die Webseite www.techthefuture.com/3D-printer informiert bis dahin über den aktuellen Stand.



(110527)

LABCENTER

LED-Exorzismus

Geisterhaftes LED-Blinken (2)

Von Dr. Thomas Scherer (D) & Elektor-Lesern

In der September-Ausgabe wurde mein Staunen über autonom blinkende Nicht-Blink-LEDs nebst Aufruf zur Dingfestmachung des in den LEDs spukenden Blink-Geists veröffentlicht. Kaum war das Heft gedruckt und ausgeliefert, kamen auch schon die ersten interessanten/klugen/amüsanten Reaktionen von Elektor-Lesern. Doch zuvor noch eine nüchterne Beschreibung des Sachverhalts:

Eine Ladekontroll-LED in einem elektrischen Korkenzieher sowie eine 1-W-Power-LED in einer Deko-Beleuchtung fingen erst an zu blinken und hauchten dann ihren Geist aus. Da die Schaltung nur aus einer LED mit einem strombegrenzenden Vorwiderstand bestand, kann man sich als Elektroniker schon verwundert die Augen reiben. Die Redaktion war daher mit einem Appell an die geballte Kompetenz der Elektor-Leserschaft zur Aufklärung des Rätsels einverstanden. Und so erklären sich Elektor-Leser das Phänomen:

Wolfgang Bredow aus Lilienthal schreibt:

„Das erinnerte mich sofort an ein Experiment, das ich in den späten 1970er-Jahren durchgeführt hatte. Damals nahm ich gelegentlich Kennlinien von allen möglichen Bauteilen mit einem Kennlinienschreiber auf. Aus einer Laune heraus wurde der vorsätzliche Mord an einer LED aufgezeichnet. Den Schrieb habe ich sogar aufgehoben (siehe Bild 1).

Wie man sieht, wurde das Opfer (eine grüne LED) über ihren Arbeitspunkt hinausgetrieben. Ab etwa 7 V und 500 mA veränderte sich zunächst die Farbe in dunkelrot. Dann fing die LED an zu blinken (aha!!!), was den Kennlinienschreiber zu irrsinnigen Bewegungen veranlasste. Die gestrichelten Linien waren die Folge einer zu trägen Mechanik, die dem ständigen Ein- und Ausschalten des LED-Stromes nicht folgen konnte.“

Herr Bredow schwört noch, dass dies der einzige Mord aus Neugier an elektronischen Bauteilen gewesen sei...

Dr.-Ing. Ulrich Pilz aus Hohen Neuendorf:

„Das geisterhafte LED-Blinken habe ich auch selbst schon kurz vor dem Ausfall einer LED beobachtet. Meine Erklärung dafür ist, dass sich durch das wechselnde Ausdehnen und Schrumpfen der LED-Struktur beim Ein- und Ausschalten (thermische mechanische Spannungen im Material) die Kontaktierung des LED-Chips mit den Anschlüssen lockert. Es kommt dann zu – zunächst noch reversiblen – Unterbrechungen des Stromkreises. Irgendwann ist dann der Stromkreis endgültig unterbrochen und die LED (nicht der LED-Chip) ist defekt.“

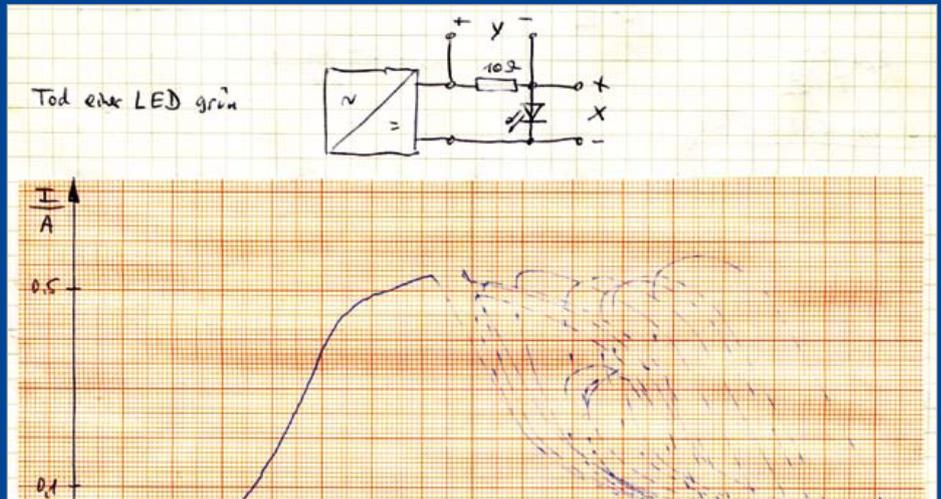


Bild 1. Originaler Kennlinienschrieb einer sterbenden LED aus den 1970ern.

Herr Pilz war nicht der einzige, der in der Kontaktstelle zwischen Bond-Draht und Chip eine wahrscheinliche Ursache für das Blinken sah. Einen Leser inspirierte die Geschichte sogar zu einer LED-wieder-heile-mach-Schaltung...

Karl-Heinz Ziener aus Lauenstein:

„Ich vermute mal, in den LEDs gibt es einen Haarriss. Womöglich im Die oder der Bonddraht bricht durch die Wärmeausdehnung. Das Blinken kommt meiner Ansicht nach deshalb zustande, weil bei kalter LED die Bruchstelle verbunden ist. Die LED leuchtet und erwärmt sich, durch die Wärme dehnt sich das Material aus, am Haarriss entsteht ein Spalt und der Stromfluss wird unterbrochen. Kein Strom, keine Wärme – das Material schrumpft und die Bruchstelle verbindet sich wieder. Das Spiel beginnt erneut.

Einen (nicht ernst gemeinten) Reparaturvorschlag zeigt Bild 2: Parallel zur LED wird ein Kondensator geschaltet. Geht die LED aus, lädt sich der Kondensator bis zur Batteriespannung auf. Will die LED, weil im Innern erkaltet, wieder leuchten, dann geht die ganze gespeicherte Energie des Kondensators in die Bruchstelle und verschweißte diese. Reparatur gelungen.“

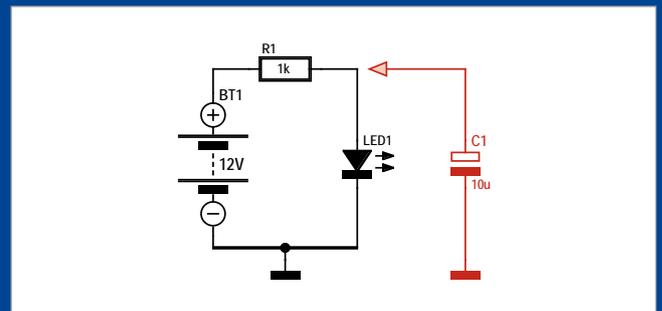


Bild 2. Zur Selbstheilung: Die Ladung im Kondensator soll den Bond-Draht wieder mit dem LED-Chip verschweißen.

Das hat natürlich nur dann Aussicht auf Erfolg, wenn die LED noch blinkt und ergo noch nicht ganz tot ist. ;-)

Gwyn Evans mailt per Smartphone aus München:

„Offensichtlich ist der Grund für diese Erscheinungen die Tatsache, dass die 9 V DC wohl keine echte Gleichspannung waren, sondern eine gleichgerichtete Wechselspannung, weshalb die Beobachtung zwingend auf eng lokalisierte Zusammenbrüche des Raum-Zeit-Kontinuums im 50-Hz-Rhythmus zurückzuführen ist.“

Ich hatte es schon fast befürchtet. Weiter: „Demnach müsste es möglich sein, die Lebensdauer einer LED durch den Betrieb mit 1-kHz-Impulsen entscheidend zu verlängern.“

Ob ich mal die Energieversorger fragen soll, ob man die Generatorfrequenz von Windrädern nicht direkt ins Netz einspeisen könnte?

Hubert Maiwald aus Neutraubling:

„Die meisten LEDs haben einen Bond-Kontakt auf der Oberseite und werden mit Silberleitkleber auf den Leadframe geklebt, der dabei die andere elektrische Verbindung herstellt.

Dieser Silberleitkleber kann sich durch Fehlbehandlungen, wie zu heißes Löten, Überhitzen, schlechte Kühlanbindung etc. lösen. Zunächst bilden sich dann zwischen dem LED-Die und der Klebestelle so genannte Lunker.

Dies verschlimmert sich dadurch, dass die thermische Anbindung zwischen Die und Leadframe immer schlechter wird und so die Verlustleistung immer schlechter abgeführt werden kann. Dadurch gast der Kleber immer mehr aus und bildet immer mehr Lunker. Die Sache verschlechtert sich weiter. Mit steigender Temperatur verschieben sich die Wellenlängen in den langwelligen Bereich.

Schließlich ist der Kontakt kaum noch vorhanden und löst sich dann durch die unterschiedlichen Ausdehnungen der verschiedenen Materialien ganz, weil die Vergussmasse im Bereich der LED viel heißer wird als die Unterseite der LED. Ohne Stromfluss kühlt alles wieder ab und der Die wird wieder in die ursprüngliche Lage zurückgedrückt. Dadurch wird wieder ein elektrischer Kontakt hergestellt und der LED-Kristall erhitzt sich...

Die thermische Zeitkonstante dieser Oszillation im Bereich 0,1...10 s ist vom Strom und der Verlustleistung abhängig. Die Funktion dieses Blinkers ist also ähnlich wie beim alten mechanischen Bimetall-Relais. Das Phänomen ist übrigens nicht sonderlich wahrscheinlich, denn es setzt voraus, dass durch die Überhitzung die Sperrschicht der LEDs nicht vor der Klebeverbindung geschädigt wird und auch der Bonddraht nicht vorher geöffnet oder beschädigt wird. Meistens geht nur die Klebeverbindung auf und bleibt einfach irreversibel offen.“

Klingt logisch, oder?

Nun, liebe Elektor-Leser, was halten Sie für die plausibelste Erklärung? Dass es etwas Thermisches ist, kann man wohl mit einiger Sicherheit annehmen. In der Elektronik gibt es immer wieder Rätsel – und gerade dort, wo man sie nicht vermuten würde...

(110668)

Weblink

Ein Tipp von Herrn Hyatt zum Thema LED-Fehler:

[www.emsnow.com/cnt/files/White Papers/DFRLEDFailures.pdf](http://www.emsnow.com/cnt/files/White%20Papers/DFRLEDFailures.pdf)

Pins perfekt auf Maß

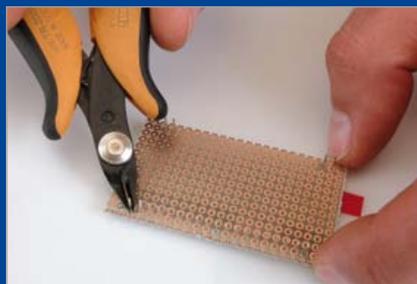
Von Thijs Beckers (Redaktion NL)

Unter „Praktisch, dieses Display“ hatten wir im Labcenter vom Oktober dieses Jahres beschrieben, wie ein Stück Lochraster-Platine mechanisch empfindliche LC-Displays vor Beschädigung bei der Demontage schützen kann. Als Beispiel für bruchgefährdete Typen hatten wir die DOGM-Displays von Electronic Assemblies genannt. Diese LC-Displays stehen gewissermaßen auf Stelzen, sie haben recht lange Anschlusspins. Die Pins müssen so lang sein, wenn das Display von der Rückseite beleuchtet werden soll. Die Hintergrundbeleuchtung braucht zwischen Display und tragender Platine etwas Raum.

Falls keine Hintergrundbeleuchtung vorgesehen ist, können

die Pins gekürzt werden. Kürzere Pins verringern den Raumbedarf und erhöhen die Stabilität des Aufbaus. Soll das Display wechselbar in einer Fassung stecken, muss das Kürzen vor dem Einsetzen geschehen. Die verkürzten Pins müssen identische Längen haben. Auch hier leistet die Lochraster-Platine einen nützlichen Dienst: Die Anschlusspins werden so durch die Löcher gesteckt, dass das Display auf der Lochraster-Platine aufliegt. Jetzt lassen sich die Pins auf der Platinenrückseite mit dem Seitenschneider bequem auf gleiche Längen trimmen. Wenn der Seitenschneider an der Platinenrückseite entlang geführt wird, haben die Pins die perfekte Länge für die Fassung.

(110664)gd



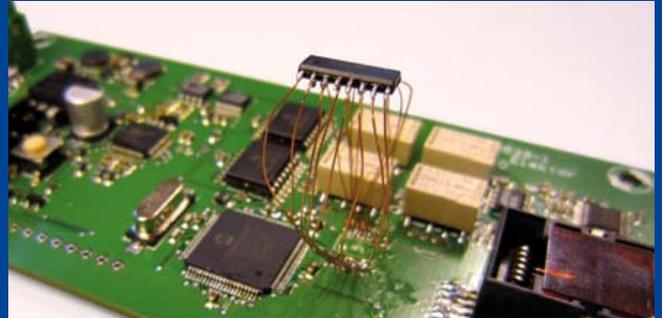
Itsy Bitsy Spider...

Von Raymond Vermeulen (Elektor-Labor)

Nach Wochen des Tüftelns und Feilens an meinem Projekt ist es so weit: Ich gebe die Platine in Auftrag und stelle die Bauelemente zusammen. Doch dann der Schreck: Ein IC und sein „Fußabdruck“ auf der Platine passen nicht zusammen! Es war der Relais-Treiber DS2003, der seit 2009 nicht mehr im TSSOP-Gehäuse produziert wird, nur die SOIC-Version kommt noch aus den Chip-Automaten. Gut zu wissen, aber leider zu spät.

Eine korrigierte Platine habe ich nicht bestellt, dann wären wieder zwei oder mehr Wochen verstrichen. Auch habe ich die bereits teilbestückte, fehlerhafte Platine nicht entsorgt (schade um Material und Zeitaufwand), sondern mich zur Improvisation entschlossen. Das Ergebnis ist, wie das Foto zeigt, zwar „quick“, aber auch „dirty“. Doch da dieses IC nur ein Relais schaltet, kam es auf kürzeste Signalwege nicht an.

Zuerst habe ich an jedes IC-Bein ein Stück Kupferlackdraht gelötet. Dann habe ich die Drähte auf gleiche Längen getrimmt und die Enden umgebogen. Aus zwei Gründen sind die Drähte vergleichsweise lang: Zum einen muss die Lötkehlspitze zwischen ihnen hindurch passen, und zum anderen darf die Wärmeleitung des Kupferlackdrahts nicht dazu führen, dass sich ein Drahtende löst, während am anderen Drahtende gelötet wird.



Für Menschen mit ruhiger Hand sind solche Feinarbeiten schnell erledigt, andere mit weniger ruhiger Hand, zu denen auch ich gehöre, brauchen gelegentlich etwas mehr Zeit. Es ist ratsam, das IC vor dem Lötten zu fixieren, damit es nicht wegrutscht, und die Drahtenden vom Isolierlack zu befreien, dann lassen sie sich leicht lötten.

Die ganze Aufregung und Mühe hätte ich mir natürlich sparen können, wenn ich mich vorher über die Bauform des DS2003 informiert hätte...

(110692)gd

Stinkender Bus

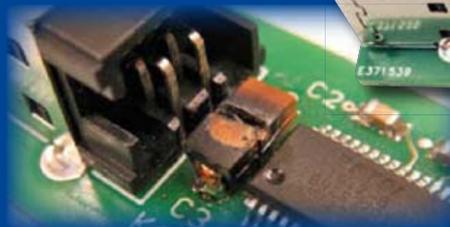
Von Thijs Beckers (Redaktion NL)

Bei der Bearbeitung eines Artikels über den Einsatz von preiswerten Tablets im Zusammenhang mit Embedded-Elektronik (an anderer Stelle in dieser Ausgabe) stach mir der typische „Strom“-Geruch verschmorter Bauteile in die Nase. Meiner Nase folgend - in meiner Eigenschaft

als Betriebsanwiter bin ich natürlich auch verpflichtet, Brandgerüchen nachzugehen - stieß ich ein Zimmer weiter auf meinen deutschen Kollegen Jens Nickel, der gerade mit dem Elektor-Bus-Projekt beschäftigt war. Es stellte sich heraus, dass ein Tantal-Elko des an das Elektor-Bus-System angeschlossenen USB/RS485-Konverters [1] seinen Geist ausgehaucht hatte. Jens war mit dem defekten Board schon im Labor gewesen, doch es roch noch immer nach „abgeraucht“...

Natürlich stellt sich bei einem solchen Schaden sofort die Frage nach der Ursache und den möglichen Folgen. Es könnte sich ja möglicherweise um einen Serienfehler handeln, der auch alle schon ausgelieferten Platinen betrifft. Handelt es sich um einen Bauteilfehler oder war der Elko falsch gepolt eingelötet worden? Müssen wir eine Rückrufaktion starten? War es unser Fehler oder ist der Platinen-Bestücker verantwortlich? Wie viel wird das kosten?

Eine Überprüfung der noch im Lager vorhandenen Exemplare sollte diese Fragen eigentlich schnell klären. Da aber unser Aus-



lieferungslager einige Kilometer entfernt in einem Industriegebiet liegt und dort schon Feierabend war, mussten wir uns damit bis zum nächsten Morgen gedulden.

Nach Kontrolle der Platinen aus dem Lager große Erleichterung: Alles richtig bestückt, also kein Serienfehler. Der Vergleich mit den Serienexemplaren zeigte auch, dass das Board mit dem defekten Elko nicht aus unserem Lagerbestand stammen konnte. Es musste sich um ein Vorserien-Exemplar handeln, das im Elektor-Labor bestückt worden war (siehe Foto). Dies ließ sich auch an Hand der verwendeten Bauteile zweifelsfrei ermitteln. Ungeklärt bleibt die Frage, wie so etwas passieren konnte, wo doch jeder weiß, dass bei SMD-Tantal-Elkos der Pluspol und nicht wie bei normalen Elkos der Minuspol markiert ist...

Wie dem auch sei, der Elko war der verkehrten Polarität jedenfalls auf Dauer nicht gewachsen und hat ordentlich geknallt und geraucht. Mehr ist zum Glück nicht passiert - der USB-Port des PCs, an den der USB/RS485-Konverter angeschlossen war, hat den vom Elko verursachten Kurzschluss unbeschadet überstanden. Erstaunlich ist allerdings, wie lange der Konverter mit dem verpolten Elko ohne Probleme funktioniert hat.

(110693)

MIAC – ein PIC für alle Fälle

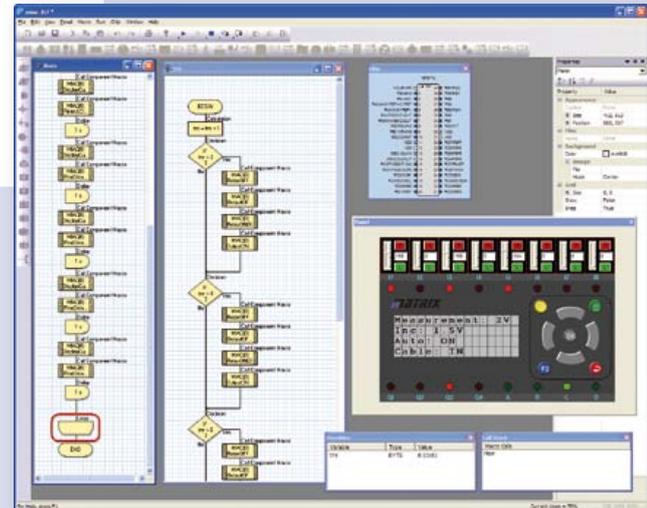


Was ist ein MIAC?

MIAC ist eine Steuerung in Industriequalität, die sich für eine breite Palette elektronischer Systeme eignet. Neben dem großen Einsatzgebiet bei industriellen Steuerungen und in der Automation eignet sich MIAC auch perfekt für die private Nutzung als Steuerung in PIC-Mikrocontroller-Projekten, die etwas Pepp vertragen könnten.

Ausstattung/Anschlüsse

- | | |
|------------------------------------|--|
| 1. Hutschienenmontage | 10. USB-Transfer-LED |
| 2. LCD (4x16 Zeichen) | 11. Tastenbedienfeld |
| 3. Power-LED | 12. M3-Montagelöcher |
| 4. Status-LEDs für Eingänge | 13. Status-LEDs für Motoren |
| 5. Stromversorgungsbuchse (2,1 mm) | 14. Schraubklemmen für Motoren |
| 6. Schraubklemmen für Eingänge | 15. Hutschienen-Arretierung unten |
| 7. Hutschienen-Arretierung oben | 16. Schraubklemmen für Relais-Ausgänge |
| 8. Reset-/Start-Taste | 17. Status-LEDs für Relais-Ausgänge |
| 9. USB-Buchse | |



Flowcode – die grafische Programmiersprache (gehört zum Lieferumfang von MIAC)



MIAC ermöglicht mit dem optional erhältlichen FlowKit-Modul (Art.-Nr. HP299) volles In-Circuit-Debugging in Flowcode

Vorteile

- Flexibel und erweiterbar
- Mit Flowcode, C oder Assembler leicht programmierbar
- Mechanisch und elektrisch sehr robust

Features

- Programmierbar via USB
- Basierend auf PIC18F4455
- Programmiersoftware Flowcode (im Wert von 180 €) im Lieferumfang inbegriffen
- Kompatibel mit C-Compilern von Drittherstellern
- 8 digitale oder analoge Eingänge
- 4 Relais-Ausgänge mit je 10 A Belastbarkeit
- 4 gesteuerte Motor-Ausgänge (500 mA)
- LCD mit 4 Zeilen à 16 Zeichen
- 9 Steuertasten im Bedienfeld

Realisieren Sie jetzt Ihr PIC-Projekt mit dem vorteilhaften MIAC-Bundle – bestehend aus einem MIAC-Modul sowie der prämierten Programmiersoftware „Flowcode 3 für PIC“ (Professional-Version)!



Bundle-Preis:
Nur 178,50 Euro

Jetzt bestellen unter
www.elektor.de/miac-bundle

Kontaktlose BasicCard

Chipkarten sehr diskret einsetzen

Von Patrick Gueulle (F)

Die einzige in Basic programmierbare Chipkarte ist schon mehr als 10 Jahre auf dem Markt und wird seither kontinuierlich weiterentwickelt. Als eine der bekanntesten Karten mit offenem Betriebssystem ist sie neuerdings durch eine RFID-Version noch interessanter geworden. Ein großer Vorteil ist auch in Verbindung mit der kontaktlosen Chipkarte die Verfügbarkeit von sehr leistungsfähigen (und kostenlosen!) Entwicklungswerkzeugen, die dazu einlädt, sich mit dieser faszinierenden Technologie zu beschäftigen.



Seit dem Erscheinen der ersten BasicCard „Compact“ im Jahr 1998 hat sich die Philosophie hinter diesem Produkt nicht geändert: Entwickeln (aber auch interessierten Amateuren) ein Mittel an die Hand zu geben, mit dem sie unabhängig von der auf große Stückzahlen ausgerichteten Chipkartenindustrie eigene Anwendungen realisieren können. Dabei herausgekommen ist eine ganze Familie von asynchronen Karten in Flash-Technologie, deren Mitglieder auch in Kleinmengen und Einzelstücken verfügbar sind. Ausgestattet mit einer (wieder-)programmierbaren „virtuellen Maschine“ (in manchen Versionen auch „multiapplikationsfähig“), unterstützen sie eine Hochsprache, die einfacher als Java, aber genauso leistungsfähig ist: ZCBasic (**Bild 1**).

Ein komplettes Entwicklungssystem (Compiler, Simulator, Double-Debugger und Handbuch mit etwa 250 Seiten) steht zum kostenlosen Download unter [1] zur Verfügung.

Mit nur wenigen Quellcode-Zeilen ist es möglich, eine BasicCard kompatibel zu fast jedem beliebigen Terminal zu machen, das kann zum Beispiel auch ein Mobiltelefon sein. Das bisherige Know-how kommt auch der kontaktlosen Version ZC7.5 RFID zu Gute. Der Übergang wird auch dadurch vereinfacht, dass es mit der ZC7.5-Combi eine Karte mit zwei Interfaces gibt: Eines über Kontakte (*transport layer protocol* T=0 ou T=1) und ein kontaktloses (ISO 14443 type A T=CL). Als RFID-Leser für das kontaktlose Inter-

face kann man zum Beispiel einen ACR122 oder Omnikey 5321 verwenden.

Karten-Applikation

Es genügen etwa zwanzig Programmzeilen (RFIDspy.BAS [2]), um die Flexibilität einer Karte wie der ZC7.5-Combi auszuloten. Das Programm belegt nur einen winzigen Teil (weniger als 400 Byte P-Code) im 32 KByte-EEPROM-Bereich des sehr leistungsfähigen Chips. Dieses kurze Programm ist eine T=CL-Variante eines *loggers*, den wir bereits für die erste BasicCard mit Kontakten vorgestellt hatten (mit T=0, für ZC4.1), erschienen im Mai 2002 [3]. Damit lassen sich die Befehle speichern und später wiedergeben, die ein Lesegerät zum Aufbau eines Dialogs mit einer ihm präsentierten Karte verwendet. Fallabhängig kann dieser „Spion“ sehr frühzeitig abgelehnt werden - oder auch mehr oder weniger lange mit einer echten Karte verwechselt werden, die für dieses Terminal vorgesehen ist.

Die zugrundeliegende Idee ist natürlich, diesen Basis-Code Stück für Stück zu erweitern. Dann kann der eine oder andere dem Lesegerät bekannte Befehl, auf den eine präzise Antwort erwartet wird, immer besser emuliert werden. So spielt man ein wenig Katz und Maus, um die Sicherheitsmechanismen aufzudecken, die der originale Aufbau verwendet und um deren Stärken und Schwächen abzuschätzen. Kurz, wir haben hier ein sehr komfortables Experimentier-Werkzeug.

Betrachten wir nun die ersten drei Zeilen des Quelltextes, welche nur einleitende Deklarationen sind, und achten besonders auf die Direktiven vom Typ `#Pragma`, wovon die beiden ersten für den *Contactless*-Modus spezifisch sind. Dazu muss man wissen, dass die meisten *contactless*-Objekte (Karten oder Tags) eine eindeutige Nummer (UID) von einigen Byte Länge enthalten, die bei der Herstellung in einem ROM abgelegt wird. Diese wird nur selten zur Kloning-Abwehr eingesetzt, dient aber vor allem für Anti-Kollisions-Prozesse, die der *Contactless*-Leser nutzt, um auch dann nur mit einer bestimmten Karte zu kommunizieren, wenn sich mehrere RFID-Karten im Lesebereich befinden.

Diese Abläufe sind relativ komplex, werden aber zum Glück von Karten und Leser selbst abgearbeitet, ohne dass der Programmierer hier eingreifen müsste (es sei denn, er wollte die Algorithmen selbst schreiben). Immerhin kann er die Anzahl der Bytes festlegen, aus denen die UID besteht, um sich den tatsächlich eingesetzten Karten anzupassen.

Die ZC7.5 erlaubt drei klassische Varianten: *Single* (vier Bytes, wie Mifare Classic), *Double* (sieben Bytes, wie Mifare Ultralight) und schließlich *Triple* (zehn Bytes). Es ist auch möglich, die auf dem Chip fest verdrahtete UID durch eine Zufallszahl (Random) zu ersetzen, um jede Rückverfolgung des Kartenbesitzers zu verhindern. In unserem Beispiel wird eine Gruppe von vier zufälligen Bytes ausgesandt, sobald der Leser das *Polling* startet:

```
#Pragma UID(Random,Single)
```

Sobald die Kommunikation mit der Karte hergestellt ist, wählt der Leser sie aus und wartet auf ihre Antwort (ATS oder *Answer To Select*), vergleichbar mit ATR (*Answer To Reset*) einer Karte mit Kontakten.

```
#Pragma ATS(TA1=0,FWI=7,TC1=0,HB="EMVA")
```

Dieser zweite Befehl dient dazu, die Default-Kommunikationsparameter ganz oder teilweise zu verändern, um so die Kompatibilität mit dem vorhandenen Leser zu optimieren. Im Beispiel wurden sie nach den *EMV Contactless Specifications* gewählt. Diese sind öffentlich einsehbar [4]. Diese Spezifikationen sichern die Kompatibilität zwischen Chipkarten und Terminals für den elektronischen Zahlungsverkehr, die ein besonderes Logo tragen, wenn sie *Contactless*-Karten akzeptieren (**Bild 2**).

Ein ähnliches Prinzip erlaubt es, die Parameter der Direktive ATR zu ermitteln, die die Karte ausgibt, wenn sie in einen kontaktbehafteten Leser gesteckt wird:

```
#Pragma ATR(Direct,T=1,HB="RFIDspy")
```

Diese Anweisung legt Protokoll T=1 fest und wählt als Übertragungstechnik *direct convention* aus, wir hätten genauso gut T=0 und/oder *direct convention* wählen können. Schauen wir nun, was

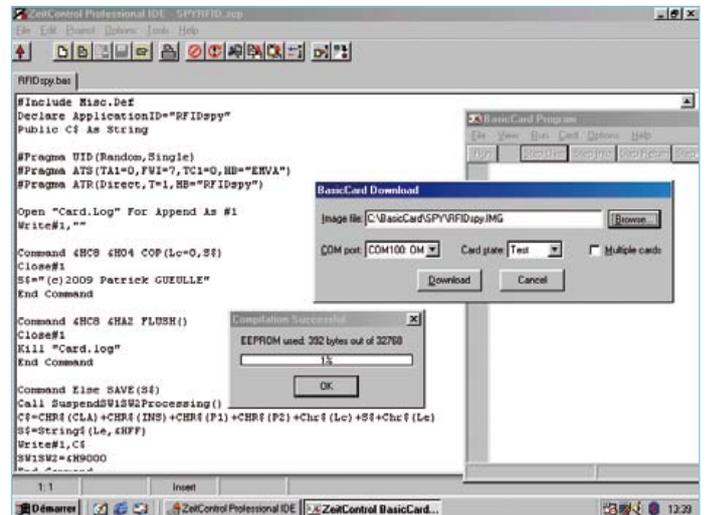


Bild 1. Die Entwicklungsumgebung für die BasicCard von ZeitControl.



Bild 2. Das Logo EMV *contactless* findet sich auf einer wachsenden Zahl von POS-Terminals.

in der Karte vor sich geht, sobald sie vom Lesegerät aktiviert wurde. Die BasicCard verfügt über ein internes Dateisystem, vergleichbar dem von MS-DOS. Eine Datei mit Namen CARD.LOG ist einfach zu öffnen:

```
Open „Card.Log“ For Append As #1
```

Um sie von einer beliebigen externen Umgebung aus löschen zu können, ist es zweckmäßig, einen gesonderten Befehl zu schreiben, hier FLUSH genannt.

```
Command &HC8 &HA2 FLUSH()
Close
Kill „Card.log“
End Command
```

Es genügt also, der Karte den Befehl `C8 A2 00 00 00` zu senden, wenn man die Datei löschen möchte. Aber das eigentliche Herzstück des Programms besteht aus den folgenden sieben Zeilen:

```

Command Else SAVE(S$)
Call SuspendSW1SW2Processing( )
C$=CHR$(CLA)+CHR$(INS)+CHR$(P1)+CHR$(P2)+Chr$(Lc)+Chr$(Le)+S$
S$=String$(Le,&HFF)
Write#1,C$
SW1SW2=&H9000
End Command

```

Das ist alles, was man braucht, um jeden nicht definierten Befehl (daher die Benennung ELSE), der von der Karte empfangen wird, in die Datei CARD.LOG zu schreiben. Dies geschieht zusammen mit den Parametern CLA, INS, P1, P2, Lc, Le und eventuell vom Terminal empfangenen Daten („Eingangs“-Befehle).

Bei „Ausgangs“-Befehlen liefert die Karte per default eine Anzahl Bytes FFh entsprechend dem Wert von Le (Länge der erwarteten Daten). Man könnte selbstverständlich auch eine andere Antwort herbeiführen, indem man den Inhalt von S\$ frei wählt, auch können die Statusbytes SW1 und SW2 von 90 00 verschieden sein, je nach gewünschtem Effekt auf das Lesegerät.

Terminal-Applikation

Jetzt muss nur noch der Inhalt der Datei CARD.LOG zu weiteren Analysen ausgelesen werden.

Da die Datei normalerweise geöffnet bleibt, um mehrere aufeinanderfolgende Sitzungen aufnehmen zu können, ist es die erste Aufgabe des Programms RFIDutil.BAS, einen Befehl C8 04 00 00 00 zu senden, der die Datei wie im folgenden Programmteil beschrieben schließt:

```

Command &HC8 &H04 COP(Lc=0,S$)
Close#1
S$="(c)2009 Patrick GUEULLE"
End Command

```

Dies benötigt nur zwei Zeilen im Quelltext „Terminal“:

```

Declare Command &HC8 &H04 COP(S$,Le=&H17)
Call COP(S$)

```

Den Inhalt der Datei auszulesen ist ebenfalls einfach. Es genügt, ein Präfix „@:“ zu setzen. Die Datei wird geöffnet und kann über das Terminal-Programm ausgelesen werden. Das Betriebssystem erzeugt alle notwendigen Befehle automatisch:

```
Open"@:card.log" For Input As #1
```

Danach gibt man für jeden Befehl, für den die Datei eine Antwort enthält, folgendes ein:

```
Input#1, Z$
```

Der ganze Rest des Codes dient nur dazu, die gelieferten Daten in

Text umzuwandeln, auf der Festplatte zu speichern und auf dem Bildschirm anzuzeigen.

Praxis

Die Entscheidung, ob man mit der integrierten Entwicklungsumgebung (sehr gut geeignet zum Verwalten von Projekten) oder mit dem Compiler ZCMBasic im Kommandozeilenformat arbeitet, ist eine reine Formalie (vorausgesetzt, man hat vorher das Handbuch studiert). Es wird eine Datei RFIDutil.EXE erzeugt, die direkt unter Windows (im Kommandozeilenmodus) lauffähig ist. Außerdem entsteht eine Datei RFIDspy.IMG beziehungsweise RFIDspy.DBG, welche in den Speicher der Karte geladen wird. Sobald die Datei dort gespeichert ist, braucht man die Karte nur noch in den Lesebereich des Terminals zu bringen, das man untersuchen möchte.

Der Autor hat die Karte an dem POS-Terminal einer Supermarktkasse in Frankreich getestet. Dort kann man bei Beträgen unter 20 € bereits kontaktlos mit EMV-Karten wie *payPass* von Mastercard oder *PayWave* von Visa bezahlen. Wenn man die BasicCard in die Nähe des POS-Terminals bringt, bevor der richtige Zahlvorgang mit einer kontaktbehafteten Bankkarte begonnen wird, findet man später in der Datei CARD.LOG eine Reihe von Select-Befehlen finden, die der folgenden ähnelt:

```

00 A4 04 00 0E 32 50 41 59 2E 53 59 53 2E 44 44
46 30 31
00 A4 04 00 07 A0 00 00 00 04 30 60
00 A4 04 00 07 A0 00 00 00 04 10 10
00 A4 04 00 07 A0 00 00 00 04 99 99
00 A4 04 00 07 A0 00 00 00 03 20 10
00 A4 04 00 07 A0 00 00 00 03 10 10
00 A4 04 00 07 A0 00 00 00 43 10 10
00 A4 04 00 07 A0 00 00 00 42 10 10

```

00 steht für die Klasse ISO (CLA) des Befehls, A4 für den Opcode (INS). 04 00 sind die Parameter P1P2, getrennt vom *application identifier* (AID) durch ein Byte mit der Längenangabe (Lc). Es ist festzuhalten, dass die Null am Ende jeder Zeile weggelassen wurde, weil sie nicht zu den für die Analyse interessanten Daten gehört, sondern lediglich anzeigt, dass der Befehl keine Antwort erwartet (Le = 0).

Die erste Zeile stellt einen Versuch der Selektion dar, genannt PPSE (*Proximity Payment System Environment*) mit dem Identifier (übertragen nach ASCII) 2PAY.SYS.DDF01. Dies ist durchaus vergleichbar mit dem PSE der EMV-Karten mit Kontakten, wobei der Identifier hier 1PAY.SYS.DDF01 lautet [5]. Da unsere Karte mit ungültigen Daten antwortet, nimmt das Terminal an, dass das PPSE, welches üblicherweise eine Liste mit von der Karte unterstützten Applikationen liefert, nicht verfügbar ist. Es versucht also im Blindflug alle seinerseits unterstützten Applikationen zu selektieren, um so empirisch zu ermitteln, welche von der Karte erkannt werden.

In den beiden folgenden Zeilen zeigt sich, dass das Terminal versucht, zwei MasterCard-Applikationen auszuwählen, mit der Priorität Maestro (A0 00 00 00 04 30 60), einer Karte für Online-Transaktionen.

Was enthält das Kit ?



Die BasicCard ist eine Erfindung von ZeitControl, einem deutschen Mittelständler, der sich vom Anbieter von Zeiterfassungssystemen zum Chipkartenspezialisten entwickelt hat. Die erste BasicCard wurde schon 1996 vorgestellt.

Im Vergleich zu früheren BasicCard-Kits schließt die Version *Dual Interface* zusätzlich eine RFID-Experimentierplatine ein (USB-Anschluss, Summer, LED-Anzeigen und Print-Antenne runden den TagTracer 14443 ab), welche die Einstellung der einzelnen Applikationen sehr erleichtert. Dieses Entwicklungs-Kit ist nach Meinung des Autors wirklich „anders als die anderen“ und bietet weit mehr Möglichkeiten als alles, was er an Vergleichbarem auf dem Markt gefunden hat.

- Omnikey 5321 USB – Dual Interface PC/SC Smartcard Reader/Writer
- Taschenkartenleser (balance reader)
- Kontaktloser USB Reader/Writer ISO 14443 als PCB Dev Board
- Software Development Kit (SDK) für Windows
- Dokumentation auf CD-ROM
- Technisches Handbuch (Papier, 250 Seiten)
- 4 x BasicCard ZC7.5 Combi (32 KB EEPROM)

Weitere Informationen: www.basiccard.com

Noch bevor die Applikationen Visa Electron (A0 00 00 00 03 20 10) und Visa credit/debit (A0 00 00 00 03 10 10) aufgerufen werden, wird eine mysteriöse Applikation A0 00 00 00 04 99 99 aufgerufen. Dabei könnte es sich um eine Applikation für eine Kundenkarte handeln. Offensichtlich fragt das *Contactless*-Lesegerät nicht nur nach Bankkarten, sondern liest auch gleich noch eine Kundenkarte aus (wenn vorhanden). Erst zum Schluss wird nach der (frankreichspezifischen Kreditkarte) Carte Bleue gefragt (A0 00 00 00 42 10 10).

Umgekehrt versucht ein Automat, der nur kontaktbehaftete Karten annimmt (öffentliche Telefone, automatische Zapfsäulen etc.) die Anwendungen eher in dieser Reihenfolge zu selektieren:

```
00 A4 04 00 07 A0 00 00 00 42 10 10
00 A4 04 00 07 A0 00 00 00 42 20 10
00 A4 04 00 0E 31 50 41 59 2E 53 59 53 2E 44 44
46 30 31
00 A4 04 00 07 A0 00 00 00 03 10 10
00 A4 04 00 07 A0 00 00 00 03 20 10
00 A4 04 00 07 A0 00 00 00 04 10 10
00 A4 04 00 07 A0 00 00 00 04 30 60
```

Hier werden die französischen Bankkarten zuerst aufgerufen, noch bevor PSE gelesen wird. Und erst danach werden die internationalen Applikationen durchprobiert. In dem einen wie dem anderen Fall haben diese Strategien das Ziel, diesen Prozess maximal zu beschleunigen, und das ist im *contactless*-Modus der kritische Punkt. Manchmal lässt sich ein Versuch feststellen, die Applikation Moneo

zu selektieren (00 A4 04 00 06 A0 00 00 00 69 00), dieses elektronische Portemonnaie gibt es in Versionen mit und ohne Kontakte. Identifier, die länger als ein Dutzend Bytes sind, verraten, dass eine Karte im Co-Branding mehrerer kommerzieller Anbieter erwartet wird.

Der Autor hat noch keine Erfahrungen mit kontaktlosen Bankkarten in anderen Ländern gesammelt. In Deutschland wird es auch noch einige Zeit bis zur Einführung solcher Karten dauern. Einer dpa-Meldung vom Juni 2011 war zu entnehmen, dass die deutschen Sparkassen mit der Ausgabe kontaktloser EC-Karten noch Ende 2011 beginnen werden. In den nächsten vier Jahren wollen sie dann alle 45 Millionen EC-Karten ihrer Kunden austauschen und so die Voraussetzung für das kontaktlose Bezahlen per Funkchip an der Ladenkasse schaffen. Wie schon seit Jahren in Frankreich soll es dann auch in Deutschland möglich sein, bis zu einem Betrag von 20 Euro kontaktlos (und ohne PIN-Eingabe!) zu bezahlen.

(090378)

Weblinks

- [1] www.basiccard.com
- [2] www.elektor.de/090378
- [3] www.elektor.de/010138
- [4] www.emvco.com

FIRST STEP

NEU!

Erste Schritte mit dem Mikrocontroller

Sie interessieren sich als Auszubildender, Schüler, Student – oder einfach nur so – für Mikrocontroller-Technik? Mit dem neuen „First Step“-Paket haben Sie den Schlüssel und alle nötigen Werkzeuge für diese faszinierende Welt in der Hand! Das fertig bestückte und getestete „First Step“-Board und drei exakt darauf abgestimmte Arbeitshefte (plus Software-CD) machen die ersten Experimente mit einem Mikrocontroller zum Kinderspiel.

Bestandteile des „First Step“-Pakets:

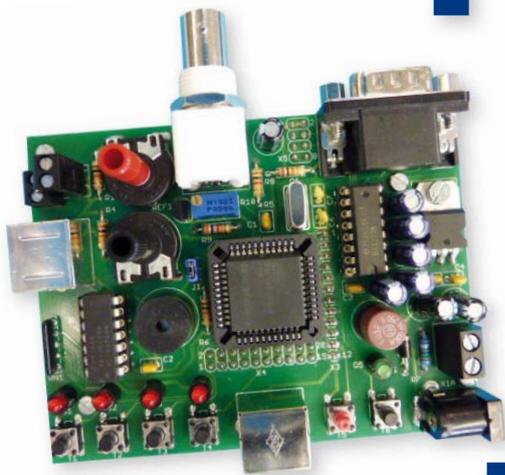
→ 3 Arbeitshefte

(inkl. passendes DIN A4-Ringbuch)

- Beschreibung der Hardware
- Beschreibung der integrierten Entwicklungsumgebung IDE
- Einführung in die Programmiersprache „C“
- Zahlensysteme, Arithmetik, Variable
- Logische Operationen
- Digitale I/O-Ports
- A/D-Wandler, Timer/Counter



elektor
ACADEMY
the school of electronics



→ 1 „First Step“-Mikrocontroller-Board

- 8051er-Mikrocontroller: AT89C51CC03
- 2,5-V-Referenzspannungsgeber für A/D-Wandler: LT1009
- TTL/RS-232-Pegelwandler MAX232
- Treiber für LEDs und Piezo-Summer: 74HC04
- 4 Taster (Eingabe von binären Signalen)
- 4 LEDs (Ausgabe von binären Signalen)
- Piezo-Summer (Ausgabe von akustischen Signalen)
- BNC-Buchse (Ein-/Ausgabe von externen binären Signalen)
- 2 Potentiometer (Eingabe von analogen Signalen)
- 2 Mini-DIN-Buchsen und eine Doppelstock-Schraubklemme
- Karten-Format: 98 x 75 mm
- Spannungsversorgung: 9 V DC, max. 100 mA, Verpolungsschutzdiode und Miniatorsicherung

→ 1 CD-ROM mit Zusatzinfos

- Datenblätter
- Systemdokumentation
- Entwicklungsumgebung
- Beispielprogramme

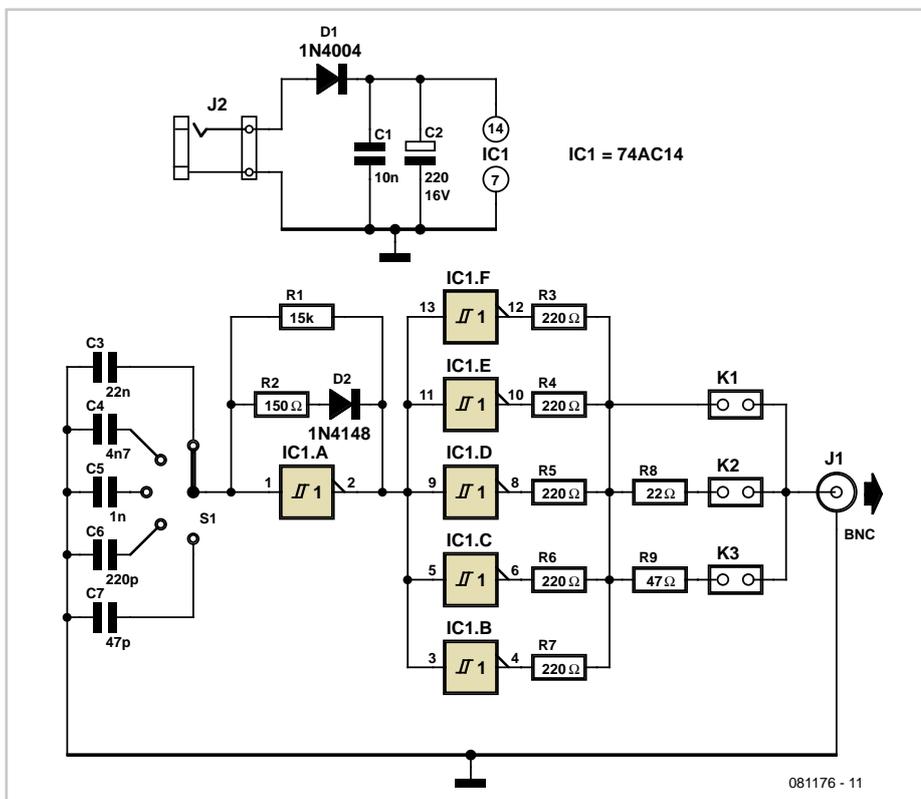
Das gesamte „First Step“-Paket kostet nur 199,00 Euro.

Weitere Infos und Bestellung unter
www.elektor.de/first-step

Vom Oszilloskop zum Reflektometer

Von Christian Tavernier (F)

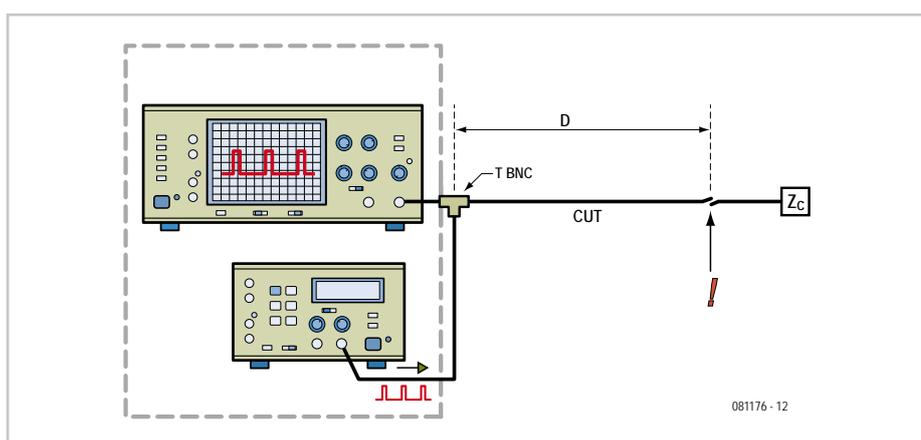
Falls Sie einem so genannten Reflektometer noch nicht begegnet sind, werden Sie vielleicht staunen, was damit möglich ist. Mit einem Reflektometer können Sie Fehler in Kabeln auf Distanz orten. Das Reflektometer gibt darüber Auskunft, in welchem Abstand vom Kabelende der Fehler gesucht werden muss.



Ein Signalgenerator, der mit einem Oszilloskop und dem zu testenden Kabel verbunden ist (CUT, Cable Under Test), sendet impulsförmige Signale mit sehr steilen Flanken. Wenn das Kabel nicht defekt ist, hat es über die gesamte Länge gleiche Impedanz. Ist das Kabel außerdem mit der korrekten Impedanz abgeschlossen, werden die Impulse nicht reflektiert. Das Oszilloskop stellt lediglich das gesendete Signal dar. Ein Leiterkurzschluss oder Leiterbruch hat zur Folge, dass das gesendete Signal reflektiert wird. Auf dem Oszilloskop erscheinen in diesem Fall das gesendete Signal und das reflektierte Signal. Durch Messen der Zeit zwischen dem gesendeten und dem reflektierten Impuls kann die Entfernung der Fehlerstelle vom Ort der Signaleinspeisung mit guter Genauigkeit bestimmt werden.

Reflektometer haben für die Praxis hohen Gebrauchswert, da defekte Kabel nicht mehr blindlings in langen Teilstücken oder sogar auf voller Länge ausgetauscht werden müssen. Falls das Kabel in der Erde liegt, muss nur an der georteten Stelle gegraben werden. Auch bei der Fehlersuche in Computer-Netzwerken oder beim Testen von Antennenkabeln leisten Reflektometer gute Dienste.

Normalerweise übersteigt der Preis eines Reflektometers die finanziellen Mittel, die in einem kleinen (und/oder privaten) Labor zur Verfügung stehen. Ein Reflektometer, das „stand alone“ einsetzbar ist, vereint einen Präzisions-Impulsgenerator, ein Oszilloskop gleich hoher Qualität sowie eine Elektronik für die Rechenarbeit im gleichen Gehäuse. Vorausgesetzt, dass Sie bereits ein Oszilloskop und einen Taschenrechner Ihr Eigen nennen, schlagen wir Ihnen vor,



Stückliste

Widerstände:

- R1 = 15 k
- R2 = 150 Ω
- R3...R7 = 220 Ω
- R8 = 22 Ω
- R9 = 47 Ω

Kondensatoren:

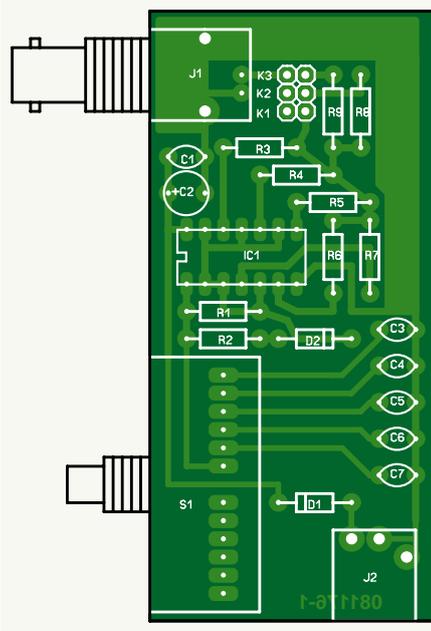
- C1 = 10 n
- C2 = 220 μ/15 V
- C3 = 22 n
- C4 = 4,7 n
- C5 = 1 n
- C6 = 220 p
- C7 = 47 p

Halbleiter:

- D1 = 1N4004
- D2 = 1N4148
- IC1 = 74AC14 (z.B. Farnell 1014022)

Außerdem:

- J1 = BNC-Buchse
- J2 = Buchse für Stromversorgung
- K1...K3 = Stiftleiste 2-polig mit Jumper
- S1 = Mehrfachschalter 1x 5 Kontakte, Lorlin PT6422/BMH (z.B. Farnell 1123675)



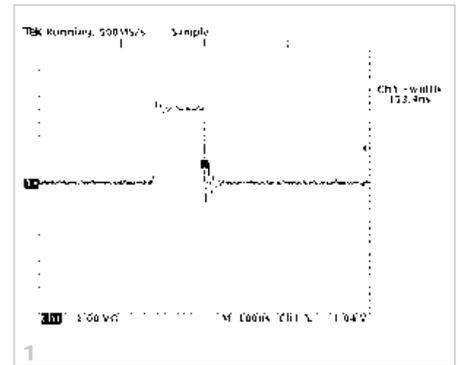
diese Geräte für etwa 20 € zu einem Reflektometer zu erweitern. Trotz dieser wirklich niedrigen Investition müssen Sie sich nicht mit mäßigen oder unzuverlässigen Resulta-

ten begnügen. Die nebenstehenden Oszillogramme beweisen, dass Sie mit der genannten Kombination zu Ergebnissen kommen, die sich mit professionellen Reflektometern messen können.

Die Erweiterung zum Reflektometer arbeitet mit nur einem simplen IC, einem sechsfachen Inverter mit Schmitt-Trigger-Eingängen in AC-Technologie (AC steht für „Advanced CMOS“). IC1.A ist als astabiler Oszillator mit sehr kleinem Duty-Cycle geschaltet. Diode D2 trägt dazu bei, dass sehr schmale Impulse erzeugt werden, wobei die Folgefrequenz vergleichsweise niedrig ist. Die Impulsbreite lässt sich mit Mehrfachschalter S1 auf unterschiedliche feste Werte einstellen. Das Messen kurzer Kabel ist nur mit sehr kurzen Impulsen möglich, denn anderenfalls trifft ein reflektierter Impuls bereits ein, bevor der zu sendende Impuls vollständig gesendet ist. In diesem Fall wäre das Oszillogramm nicht brauchbar. Beim Messen langer Kabel reicht die Energie sehr kurzer Impulse nicht aus, um eindeutige reflektierte Signale zu erzeugen. Hier müssen breitere Impulse gesendet werden, deren Energieinhalt größer ist.

Damit die Impedanz des Impulsgenerators im Bereich der charakteristischen Impedanz des zu messenden Kabels liegt, sind die fünf anderen in IC1 enthaltenen Inverter parallel geschaltet. Die resultierende Ausgangsimpedanz lässt sich durch Stecken der Jumper K1, K2 oder K3 anpassen. Mit K1 beträgt die Ausgangsimpedanz 50 Ω, mit K2 liegt sie bei 75 Ω, und mit K3 ist sie ungefähr gleich 100 Ω. Dies sind die typischen Impedanzen der meisten gebräuchlichen Kabel. Die Betriebsspannung der Schaltung beträgt +6 V. Am Arbeitsplatz kann ein Labornetzteil diese Spannung liefern, im mobilen Einsatz sind vier 1,5-V-Batterien die passende Energiequelle. Diode D1 schützt die Schaltung gegen Verpolen der Betriebsspannung.

Für den Schaltungsaufbau empfehlen wir Ihnen die von uns entworfene Platine [1], sie gewährleistet zuverlässige Arbeitsweise und präzise Messergebnisse. Die Verbindungswege auf der Platine beschränken sich auf das Notwendige, denn Mehrfach-



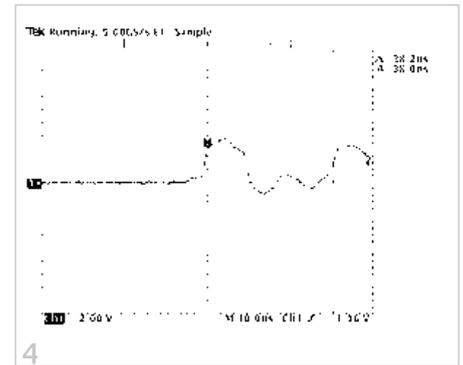
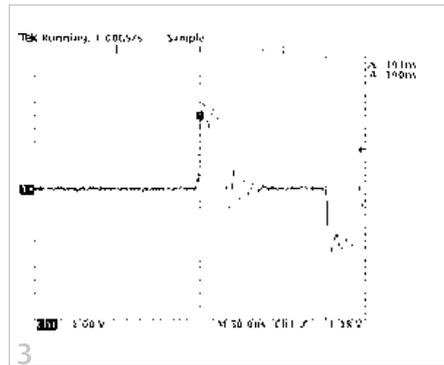
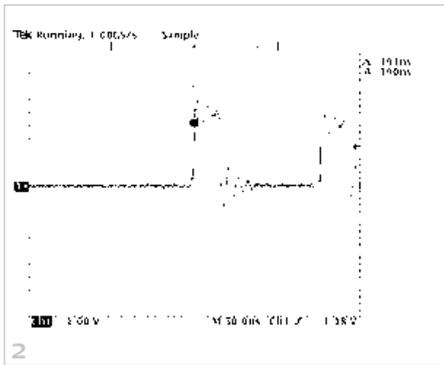
schalter S1 und Ausgangsbuchse J1 werden unmittelbar auf die Platine gelötet. Schalter S1 ist eine vertikale, für Platinenmontage konstruierte Ausführung des Herstellers Lorlin, die Typenbezeichnung lautet PT6422/BMH.

Das Ausgangssignal wird zwar über eine BNC-Buchse herausgeführt, doch es können nicht nur Coax-Kabel gemessen werden. Für andere Kabelarten werden lediglich passende BNC-Steckeradapter benötigt. Das Messen mit dem Reflektometer gestaltet sich unkompliziert, gehen Sie einfach wie folgt vor:

Stecken Sie abhängig von der Impedanz des zu messenden Kabels einen Jumper auf K1 (50 Ω), K2 (75 Ω) oder K3 (100 Ω). Verbinden Sie die Schaltung wie im ersten Bild dargestellt über einen T-Adapter mit dem Oszilloskop und dem Messobjekt. Wählen Sie die Impulsbreite mit S1 abhängig von der Kabellänge aus, und stellen Sie die Zeitbasis und Triggerung des Oszilloskops so ein, dass der gesendete Impuls sichtbar ist.

Bei intaktem Kabel und korrekter Abschlussimpedanz stellt das Oszilloskop, wie Oszillogramm 1 zeigt, nur den gesendeten Impuls dar. Ein gebrochenes oder nicht abgeschlossenes Kabel hat einen reflektierten Impuls mit gleicher Polarität wie der gesendete Impuls zur Folge, siehe Oszillogramm 2. Falls die Kabelleiter kurz geschlossen sind, erscheint auf dem Oszilloskop ein Bild ähnlich Oszillogramm 3: Ein reflektierter Impuls, dessen Polarität im Vergleich zum gesendeten Impuls vertauscht ist.

In den Fehlerfällen müssen Sie auf dem Oszilloskop die Zeit zwischen den Vorderflanken des gesendeten und des zugehörigen reflektierten Impulses ablesen (hier 191 ns), um fest zu stellen, wo sich die Fehlerstelle befindet. Sie können davon ausgehen, dass sich Signale im Kabel mit der



Geschwindigkeit 200 m/μs fort pflanzen. Jetzt genügt eine einfache Rechenoperation, um die Distanz vom Kabelanfang bis zum Fehlerort zu bestimmen. Wenn D die Distanz in Metern, V die Signalgeschwindigkeit in m/μs und T die Zeit zwischen zwei Vorderflanken in μs ist, dann gilt: $D = (V \cdot T) / 2$. Im Beispiel befindet sich der Fehler in 19,1 m Abstand zum Kabelanfang, denn die gemessene Zeit beträgt 191 ns. Bei nicht eindeutigem Fehler oder bei zu kurzer Impulsbreite kann die Darstellung

auf dem Oszilloskop wie in Oszillogramm 4 gestört sein. Das Messen der Zeit zwischen zwei ansteigenden Flanken ist trotzdem möglich, im dargestellten Beispiel beträgt der Abstand zur Fehlerstelle 3,8 m.

Zum Schluss noch ein Hinweis: Wenn Sie möglichst genau messen wollen, müssen Sie die Berechnung mit der exakten Signalfortpflanzungsgeschwindigkeit durchführen, die für das gemessene Kabel gilt. Auskunft über diese Größe geben in der Regel

die technischen Daten, die der Kabelhersteller für den Kabeltyp angibt.

(081176)gd

Weblink

[1] www.elektor.de/081176 (Projektseite mit Download des Platinen-Layouts)

—Anzeige

Schaeffer AG

FRONTPLATTEN & GEHÄUSE

Kostengünstige Einzelstücke und Kleinserien

Individuelle Frontplatten können mit dem Frontplatten Designer mühelos gestaltet werden. Der Frontplatten Designer wird kostenlos im Internet oder auf CD zur Verfügung gestellt.

- Automatische Preisberechnung
- Lieferung innerhalb von 5-8 Tagen
- 24-Stunden-Service bei Bedarf

Preisbeispiel: 34,93 € zzgl. USt./Versand

Schaeffer AG
Nahmitzer Damm 32
D-12277 Berlin
Tel +49 (0)30 8 05 86 95-0
Fax +49 (0)30 8 05 86 95-33
Web info@schaeffer-ag.de
www.schaeffer-ag.de

Schrittmotorsteuercarte SMC-TR-1000 *Neuheit*

für PC- Positioniersysteme, Bohr-, Gravier- und Fräsanlagen!

Steuerung von 3 Achsen **nur € 89,-***

- Phasenströme einstellbar
- Eingänge für Takt- u. Richtungssignale
- Mikroschrittbetrieb (1/1, 1/2, 1/4, 1/8, 1/16)
- Auch von SPS oder Taktgenerator ansteuerbar
- Inklusive Treibersoftware für Windows
- Automatische Ruhestromreduzierung
- Temperaturüberwachung
- 3 Eingänge für Referenzschalter
- 1 Eingang für Not-Aus-Schalter
- Nur 1 Spannungsvers. (24VDC) notwendig
- Hohe Geschwindigkeit durch optimierte Stromregelung
- Kompakte Bauweise
- Auf Sonderwünsche kann eingegangen werden

Besuchen Sie unsere Homepage

emis GmbH

Zur Drehscheibe 4
92637 Weiden
Tel. (0961) 32040, Fax (0961) 31494
www.emisgmbh.de + MWST

BESSER GLEICH ONLINE KALKULIEREN.

STARRE- UND FLEXIBLE LEITERPLATTEN.

ab 12h
Leiterplatten-Express! Pünktlich oder gratis!

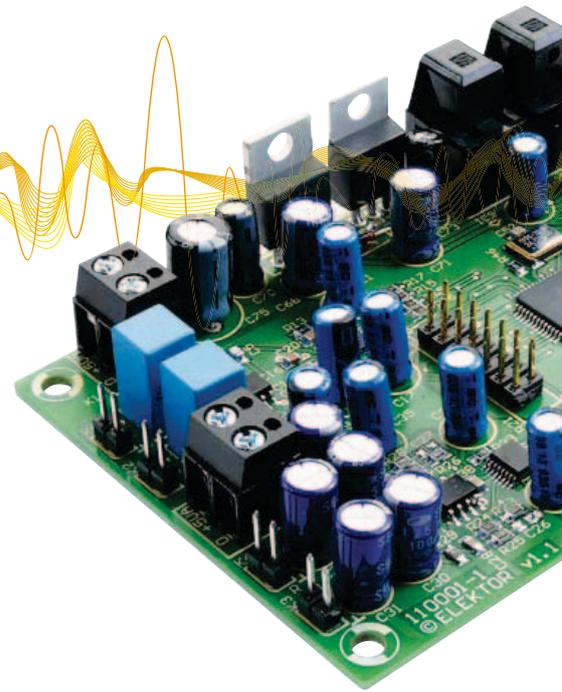
LEITON
RECHNEN SIE MIT BESTEM SERVICE

Schluss mit umständlichen Rechenoperationen! Bei uns kalkulieren Sie auch Ihre exotischsten Leiterplatten jederzeit schnell und einfach online. Doch nicht genug: Bei LeitOn gilt die Online-Kalkulation auch für Serien und flexible Leiterplatten! Ebenso einmalig ist der LeitOn Leiterplatten-Expressdienst mit Top-Garantie: Platinen sind gratis bei überschrittenem Liefertermin! Neugierig? Unsere persönliche Telefonberatung und unser kompetenter Außendienst helfen Ihnen gerne weiter. Denn Sie wissen: Bei LeitOn rechnen Sie immer mit bestem Service.

www.leiton.de Info-Hotline +49 (0)30 701 73 49 0

Audio-DSP-Kurs

Teil 6: Digitaler Audiogenerator



Ein Signalgenerator für Audio als Laborgerät sollte zum einen äußerst klirrarmer Sinusschwingungen mit einstellbaren Frequenzen und Amplituden zur Messung von Frequenzgängen und Klirrfaktoren an Audiogeräten und zum anderen Tief- und Bandpassrauschsignale für elektroakustische Messungen an Schallwandlern erzeugen können. Ein DSP ist für solche Aufgaben in besonderer Weise geeignet. Das DSP-Board dieser Artikelserie wird hier als zentrale Komponente der Signalerzeugung eines Laborgenerators verwendet und ermöglicht das Erzeugen digitaler und analoger Signale in einer hohen Qualität.

Von Alexander Potchinkov (D)

Ein Audiosignalgenerator gehört zur Grundausstattung an Testgeräten, auch in einem kleinen Audio- und Elektroakustiklabor. Besonders vielseitig verwendbar ist der Audiosignalgenerator dann, wenn er nicht nur analoge, sondern auch digitale Signale zur Verfügung stellt und zweikanalig ausgeführt ist. Dann können wir Analoggeräte und Digitalgeräte sogar gleichzeitig mit Testsignalen speisen. Im Labor und an Anlagen an ihren Einsatzorten ist einiges zu messen, so wie die Frequenzgänge von selbstentwickelten Lautsprecherboxen und von Lautsprecheranlagen zur Verwendung bei Veranstaltungen. Auch müssen wir lineare und nichtlineare Verzerrungen von Audiogeräten messen.

Der in diesem Beitrag beschriebene Audiosignalgenerator stellt hierfür die Test- oder Messsignale zur Verfügung, deren Eigenschaften die Aussagekraft einer Messung nicht erheblich verfälschen dürfen. Wenn die Bandbreite nach oben hin nicht zu groß sein muss (für Audiomessungen reicht in den meisten Fällen der Hörfrequenzbereich von 20 Hz bis 20 kHz aus), kann der Generator mit dem DSP realisiert werden.

So erreichen wir unschlagbar niedrige Kosten bei gleichzeitig hervorragender Qualität der Signale, womit bei Sinussignalen Klirrfaktor und Rauschcharakter und bei Rauschsignalen die Eigenschaften der Rauschformungsfilter gemeint sind. Mit Analoggeneratoren sind diese Eigenschaften, jedenfalls unter vergleichbaren Bedingungen, nicht erreichbar. Für einen Laborgenerator ist das DSP-Board noch mit einer Benutzerschnittstelle zur Eingabe der Generatoreinstellparameter und bei Bedarf mit einer einstellbaren analogen Ausgangsstufe zu ergänzen. In diesem Beitrag stellen wir die Signalverarbeitung und das hierfür dienende DSP-Programm vor und geben am Schluss Hinweise darauf, wie eine Benutzerschnittstelle ausgeführt werden kann.

Signalverarbeitung des digitalen Audiosignalgenerators

Das Blockbild der Signalverarbeitung für einen der beiden gleichwertigen Kanäle, wir bezeichnen sie als *linken* und *rechten* Kanal, zeigt in **Bild 1** zwei Generatoren. Der eine ist ein Sinusgenerator mit einstellbarer Frequenz und der andere ist ein Rauschgenerator für weißes Rauschen. Im Weiteren sehen wir eine Filterbank mit 42 Rauschformungsfiltern und zwei Vielstufenschaltern zur Auswahl des Filters sowie einen Signalquellen-

umschalter und einen Signalabschwächer. Je nach gewählter Schalterstellung und gewähltem Filter können wir Sinusschwingungen oder gefiltertes Rauschen erzeugen. Da beide Kanäle unabhängig voneinander sind, lassen sich Sinusschwingungen und (gefiltertes) Rauschen gleichzeitig auch jeweils einkanalig erzeugen.

Die Signalverarbeitung in **Bild 1** benötigt fünf positivwertige Parameter, das Phaseninkrement $d\phi$ zur Einstellung der Frequenz der Sinusschwingung, den Index F_1 des Rauschformungsfilters (wir können diesen Index als Schalterstellung ansehen), die Schaltposition S_1 des Signalquellenumschalters, den Abschwächungsfaktor α und einen Rauschsignalverstärkungsfaktor Gain, der in Stufen zu 6 dB (Verdopplung der Ausgangsspannung) geschaltet werden kann. Mit der Nutzung beider Parameter α und Gain lassen sich beliebige Verstärkungswerte auch zwischen den 6-dB-Stufen einstellen. Wegen des zweikanaligen Aufbaus werden für das DSP-Programm später zwei Sätze zu je fünf Parametern benötigt.

Im Folgenden werden wir nacheinander die drei Blöcke der digitalen Signalverarbeitung

- Sinusgenerator,
- Rauschgenerator und
- Digitalfilter zur Rauschformung beschreiben.



Sinusgenerator

Zur Erzeugung einer digitalen Sinusschwingung in der Form

$$x(n) = a \sin(2\pi n f_s / f_T), n = 0, 1, \dots$$

mit der Abtastfrequenz f_T , der Schwingungsfrequenz $f_s < f_T/2$, einem verschwindenden Nullphasenwinkel und der Amplitude a verwendet man zumeist zwei Techniken der Schwingungserzeugung. Zum einen werden die aus der Analogtechnik vertrauten Oszillatoren mit Rückkopplung wie der *Wien-Brücken-Oszillator* verwendet, die digital als *rekursive Schwinger zweiter Ordnung* bezeichnet werden und bei hohen Anforderungen an die Klirrarkeit des Signals analog wie auch digital mit einer Amplitudenregelung arbeiten müssen. Zum anderen verwendet man einen *Phasenakkumulator* mit einem *nichtlinearen Funktionsformer*, den Elektor-Leser mit der DDS-Technik (Direct Digital Synthesis, Elektor Oktober 2003, HF-Messender) kennengelernt haben. Beim Digitalsinusgenerator erzeugt der Phasenakkumulator ein Sägezahnsignal mit der Periodendauer der zu erzeugenden Sinusschwingung. Der nichtlineare Funktionsformer verformt die Sägezahnsschwingung zu der gewünschten Sinusschwingung. Eine solche Technik hat man auch mit Diodenfunktionsnetzwerken und einem dreieckförmigem Phasenakkumulatorsignal analog realisiert, was aber wegen des notwendigen Abgleichs und der Temperaturabhängigkeit der Bauteile zu wenig befriedigenden Ergebnissen führte. Für den nichtlinearen Funktionsformer im digitalen Sinusgenerator benutzt man häufig eine so genannte *Look-Up-Table*, in der eine abgetastete Sinusschwingung mit beispielsweise 1024 Abtastwerten abgelegt ist. In den Fällen, in denen der Phasenakkumulatorwert zwischen den der Tabellierung entsprechenden Abtastzeitpunkten liegt, interpoliert man linear zwischen zwei Tabellenwerten. Ein Nachteil dieser Technik ist es, dass die Signalverzerrungen von der Frequenz abhängig sind und größer werden, wenn viele Werte durch Interpolation gewonnen werden müssen. Eine Alternative hierzu ist die *Polynomapproximation*, die zwar mehr Rechenaufwand mit dem Auswerten eines Polynoms benötigt, auf der anderen Seite aber die Look-Up-Table

spart und die geringstmöglichen Verzerrungen garantiert, die außerdem auch noch frequenzunabhängig sind. Wir haben diese Technik gewählt, da die spektrale Qualität der Sinusschwingung das primäre Entwurfsziel war. Wir meinen, dass sich der Aufwand einer digitalen Signalverarbeitung zur Erzeugung von Sinusschwingungen nur

$f_T = 48 \text{ kHz}$ eine Sinusschwingung mit der Frequenz $f_s = 3 \text{ kHz}$ erzeugt werden, so ist das Phaseninkrement zu $d\phi = 2\pi \cdot 3 / 48 = 0,125$ zu wählen. Die im DSP56374 benutzte Zweierkomplementarithmetik, bei der wir uns die Zahlen auf einem Ring angeordnet vorstellen können, liefert uns die gewünschte Sägezahnsschwingung ohne weiteres Zutun,

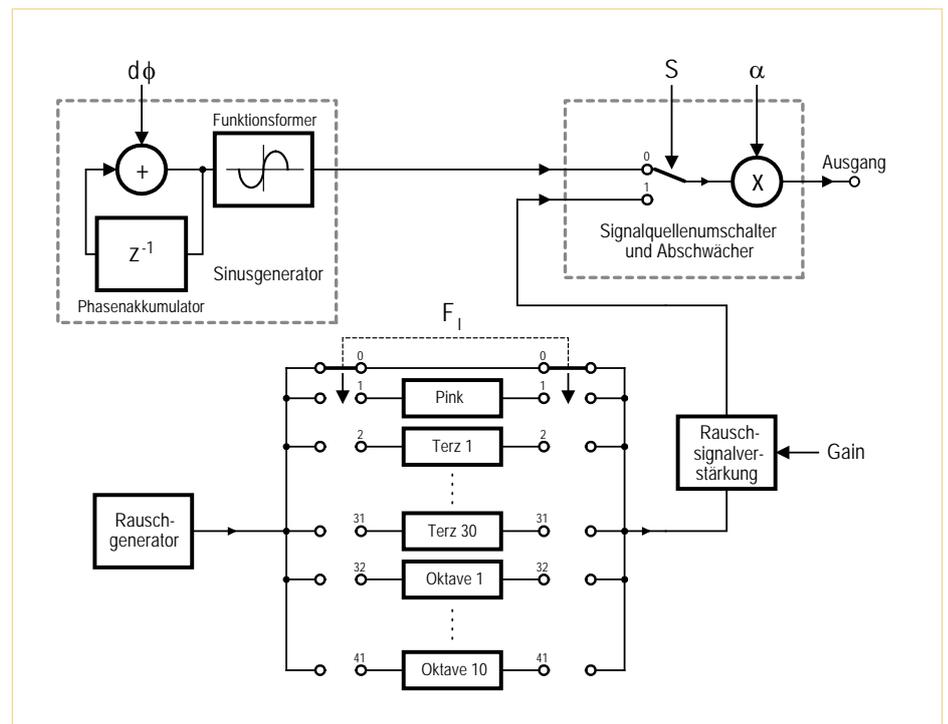


Bild 1. Blockbild der Signalverarbeitung.

lohnt, wenn diese eine höhere Klirrarkeit aufweisen als sie mit bezahlbarer Analogsignalverarbeitung erreichbar ist. Der bereits erwähnte höhere Rechenaufwand ist in Anbetracht der Rechenleistung des DSPs so gering, dass er als Argument für eine einfachere Schwingungserzeugung nicht angeführt werden kann. Für den interessierten Leser sei erwähnt, dass unsere Polynomkoeffizienten durch eine *Tschebyscheffapproximation* gewonnen wurden, und dass wir ein Polynom vom Grad 11 verwenden und es mit dem *Horner-Schema* auswerten. Der Phasenakkumulator startet bei Null, und mit jedem Abtastzeitpunkt wird sein Wert um das Phaseninkrement $d\phi = 2\pi f_s / f_T$ erhöht. Soll bei einer Abtastfrequenz von

was mit der Ringanordnung erklärt werden kann. Der Phasenakkumulator wird in doppelter Genauigkeit mit 48 bit gerechnet, was der DSP56374 mit den 48-bit-Long-Moves unterstützt. Die Frequenzstabilität unserer Sinusschwingungen wird vom Quarzoszillator des DSP-Boards bestimmt und ist für Audio-Zwecke mehr als ausreichend hoch.

Rauschgenerator

Auch für einen Rauschgenerator gibt es mehrere Möglichkeiten der Realisierung. Wir haben uns für die einfachste entschieden, nämlich für die Verwendung eines rückgekoppelten Schieberegisters. „Einfach“ bedeutet aber in diesem Fall sicher

Tabelle 1: Indices der Filter für die Filterschalter

Filterindex	Filter
0	Weißes Rauschen (keine Spektralformung)
1	Rosa-Rausch-Filter
2	Terzfilter, Mittenfrequenz $f_m = 25$ Hz
...	...
31	Terzfilter, Mittenfrequenz $f_m = 20$ kHz
32	Oktavfilter, Mittenfrequenz $f_m = 31,5$ Hz
...	...
41	Oktavfilter, Mittenfrequenz $f_m = 16$ kHz

Tabelle 2: Verstärkungsmaße von Terz- und Oktavfilter für die Mittenfrequenz $f_m = 1$ kHz

Verstärkungsmaß	Terzfilter		Oktavfilter	
	links	rechts	links	rechts
-3 dB	895 Hz	1117 Hz	718 Hz	1393 Hz
-20 dB	790 Hz	1266 Hz	509 Hz	1958 Hz
-40 dB	611 Hz	1636 Hz	291 Hz	3385 Hz
-60 dB	385 Hz	2581 Hz	145 Hz	6507 Hz

Tabelle 3: Programmparameter und ihre Wertebereiche

Parameter	Wertebereich*	Datentyp	Wortlänge	Position
SL, SR	[0,1]	Integer	24	Rechtsbündig
FiL, FiR	[0,1,...,41]	Integer	24	Rechtsbündig
DphiL, DphiR	(0,1)	Fractional	48	Linksbündig
AlphaL, AlphaR	(0,1)	Fractional	24	Linksbündig
GainL, GainR	[0,1,...,6]	Integer	24	Rechtsbündig

* Die eckigen Klammern zeigen an, dass die Intervallrandwerte eingeschlossen sind, wohingegen die runden Klammern anzeigen, dass die Intervallrandwerte nicht eingeschlossen sind.

Tabelle 4: Defaultparameterwerte

Programmparameter des linken Kanals		Programmparameter des rechten Kanals	
Parameternamen	Defaultwert	Parameternamen	Defaultwert
DphiL	0,041666667	DphiR	0,041666667
FiL	18	FiR	18
SL	0	SR	1
AlphaL	0,5	AlphaR	0,5
GainL	0	GainR	1

Tabelle 5 Programmdateien des digitalen Audiogenerators

AudioGen.asm	Das Hauptprogramm
Kocz_SinCoef.tab	Sinus-Polynomkoeffizienten
ElektorFilter.tab	Koeffizienten der Digitalfilter
src4392.tab	Bytefolge zur Programmierung des SRC
ivt.asm	Einträge in die Interruptvektortabelle, Audio-Interrupts
esai4r2t.asm	Audio ISR, 4 Eingangskanäle, 2 Ausgangskanäle
mioequ.asm	Nützliches, Klarnamen für die DSP-IO-Adressen

nicht „von geringem Wert“. Diese Technik der rückgekoppelten Schieberegister wird schon seit langem für den (sehr einfachen) Aufbau von Analograuschgeneratoren mit Hilfe einiger Logik-ICs verwendet. Wenn sich der Leser den DSP-Code ansieht, wird er erkennen, dass die Softwarerealisierung fast noch einfacher ist. Das so erzeugte periodische Rauschsignal ist ein Pseudoruschsignal, da in einer Periode jeder erlaubte Zustand des Schieberegisters genau einmal eingenommen wird. Ein möglicher Zustand (alle Register zeigen den Wert Null) ist verboten, da dieser Zustand nicht verlassen werden kann. Um ein Zahlenbeispiel zu geben, gehen wir von einem 4-Flipflop-Schieberegister aus, das 16 mögliche Zustände kennt, von denen $15 = 2^4 - 1$ erlaubte Zustände sind (entsprechend den Zahlen 1 bis 15). Wir verwenden zwei Schieberegistergeneratoren mit jeweils 24 Flipflops, was zu einer Periodendauer von ungefähr sechs Minuten bei einer Abtastfrequenz von 48 kHz führt. Das ist für alle Audioanwendungen mehr als ausreichend. Beide Generatoren werden mit unterschiedlichen Rückkopplungsanzapfungen betrieben, was dazu führt, dass beide Generatoren für die Praxis ausreichend unkorrelierte, das heißt, statistisch unabhängige Signale liefern. Somit lassen sich auch echte Zweikanalmessungen durchführen. Auch mit diesem Detail unterstreichen wir den hohen Anspruch an die Qualität der Signalverarbeitung in diesem Projekt.

Filter

Die Filter dienen der spektralen Rauschformung und werden in einer Filterbank mit 42 digitalen Filtern zusammengefasst, deren Indizes in der **Tabelle 1** eingetragen sind:

- Ein Dummy-Filter für das weiße Rauschen.
- Ein Pink-Noise-Filter, mit dem aus dem weißen Rauschen durch Tiefpassfilterung rosa Rauschen geformt wird. Die Lautsprecherboxenspezialisten unter den Lesern wissen, dass dieses rosa Rauschen nicht nur die gegen Überlastung besonders empfindlichen Hochtöner vor dem Hitzetod schützt, sondern beim Messen der Lautsprecherboxen mit dem Terzbandanalysator die Terzpegel idealerweise alle gleich groß sind.

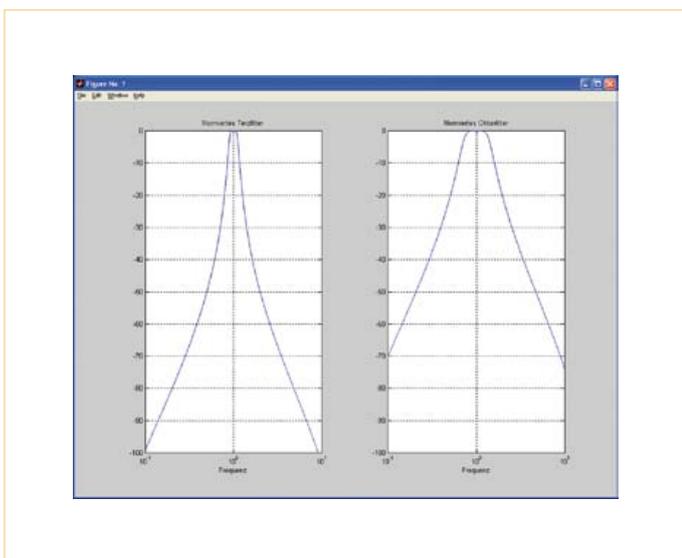


Bild 2. Betragsfrequenzgänge eines Terz- und eines Oktavfilters in Dezibel für normierte Frequenzen.

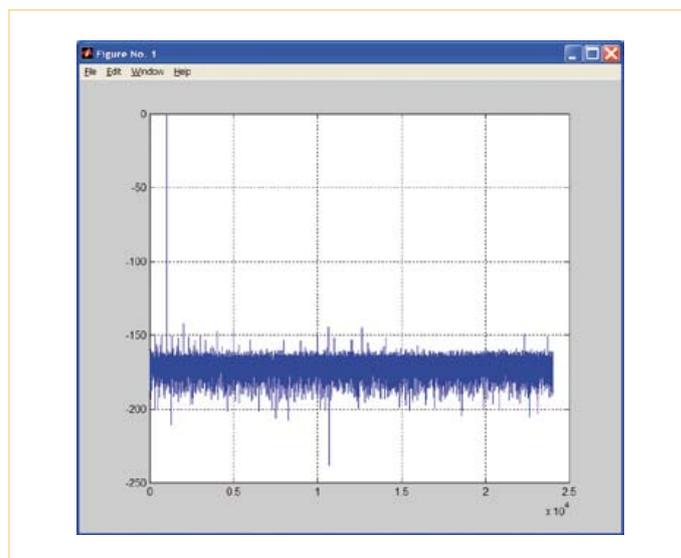


Bild 3. Spektrum einer digital erzeugten 1-kHz-Sinusschwingung.

- 30 Terzbandfilter zur Erzeugung von Terzbandrauschen mit den Normmittenfrequenzen 25, 31,5, 40, 50, 63, 80, 100, 125, 160, 200, 250, 315, 400, 500, 630, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000, 6300, 8000, 10000, 12500, 16000 und 20000, alle Angaben in Hertz nach DIN EN 61260.
- 10 Oktavbandfilter zur Erzeugung von Oktavbandrauschen mit den Normmittenfrequenzen 31,5, 63, 125, 250, 500, 1000, 2000, 4000, 8000 und 16000, alle Angaben in Hertz nach DIN EN 61260.

Beispielhaft zeigt **Bild 2** die Betragsfrequenzgänge eines Terz- und eines Oktavfilters mit normierter Frequenz. Wenn wir die Filtermittenfrequenzen auf 1 kHz in die akustische Mitte legen, können wir die in der **Tabelle 2** angegebenen Filterbetragsfrequenzgangswerte, ausgedrückt durch die Verstärkungsmaße, ablesen. Die -3-dB-Grenzfrequenzen wurden in der Tabelle farblich markiert.

Da es sich um digitale Filter handelt und wir eine nicht sehr hohe Abtastfrequenz von 48 kHz benutzen, weichen die Filter rechts von der Mittenfrequenz von der analogen Vorgabe bei hohen Frequenzen ab. Der Grund dafür ist, dass wir bei der halben Abtastfrequenz, sprich 24 kHz, unendlich große Dämpfung haben. Man nennt dies in der Lehre der Signalverarbeitung *Nullstelle des Frequenzgangs*. So geht auf der einen Seite die Symmetrie zwischen linkem und rechtem Ast verloren, auf der anderen Seite aber verbessert sich die Selektivität

im rechten Ast, so dass dieser Effekt nicht von Nachteil ist.

Alle Filter sind sog. *rekursive* oder *IIR*-Filter vom Grad 6, die man, so wie man es auch bei Analogfiltern praktiziert, aus jeweils drei hintereinander angeordneten Teilfiltern vom Grad 2 aufbaut. Ein Filter vom Grad 6 benötigt in der gewählten Struktur einen Zustandsspeicher mit 8 Speicherzellen. Dieser Zustandsspeicher entspricht den Kondensatoren und/oder den Induktivitäten der Analogfilter. Im Weiteren benötigt ein Filter 15 Filterkoeffizienten, mit denen seine Eigenschaften festgelegt werden. Wir haben die $41 \cdot 15 = 615$ Filterkoeffizienten in einer DSP-Tabelle abgelegt. Aus programmiertechnischen Gründen gibt es ein 42stes Filter, das keine Funktion hat und für das weiße Rauschen, genauer gesagt sein Durchreichen, verwendet wird. Dieses Filter hat den Filterindex 0.

Wenn man ein Filter wechseln möchte, muss lediglich der Koeffizientenzeiger auf die entsprechenden Filterkoeffizienten zeigen. Das kann man leicht ausrechnen. Ausgehend von der Filterkoeffizientenbasisadresse A_B (Beginn der Tabelle) und dem Filterindex F_I wird eine Zeigeradresse $A = A_B + 15 \cdot F_I$ berechnet, die den Beginn eines Filterkoeffizientenfelds anzeigt. Hier erkennen wir eine der ganz wichtigen Stärken der digitalen Signalverarbeitung. Man bedenke nur, welchen Aufwand und welche Kosten eine vergleichbare Analogfilterbank verursachen würde. Der Autor hat in seinem Labor eine betagte Terz- und Oktavfilterbank von Brüel&Kjaer stehen, die gut 15 kg

schwer ist. Wir hätten ohne weiteres auch hunderte von Filtern im DSP-Speicher unterbringen können, die in einfachster Weise durch „Umbiegen von Zeigern“ gewechselt werden können.

Für den interessierten Leser erwähnen wir, dass die 40 Bandpassfilter mit ihrer Spezifikation den Class-0-Anforderungen der Norm DIN EN 61260 entsprechen. Das Pink-Noise-Filter erreicht im Bereich von 10 Hz bis 20 kHz eine Pegel-Abweichung von weniger als 0,1 dB. Die Bandpassfilter lassen sich ohne weiteres auch für Sperrdämpfungen größer als 100 dB nutzen, was mit Analogfiltern bei vertretbarem Aufwand kaum möglich ist.

DSP-Programm

Das DSP-Programm mit der Signalverarbeitung wird als ein Block in die Audioloop eingesetzt. Auch wenn wir keine Audiosignale in den Generator einlesen, ändern wir die Kurs-Audioloop an dieser Stelle nicht und lesen dennoch pro forma Audiosignale ein, da die Synchronisation innerhalb der Audioloop sich auf die empfangenen Daten bezieht und wir für alle drei Projekte ein einheitliches Rahmenprogramm verwenden wollen. Das DSP-Programm verwendet zehn Programmparameter als lokale Parameter. Die **Tabelle 3** listet die Parameter, ihre Wertebereiche und weitere Eigenschaften auf. Die Defaultwerte sind so gewählt, dass der linke Kanal eine Sinusschwingung mit der Frequenz 1 kHz und der rechte Kanal ein Terzbandrauschen mit der Mittenfrequenz 1 kHz erzeugt. In beiden Fällen halbiert der Abschwächer die Signalwerte oder

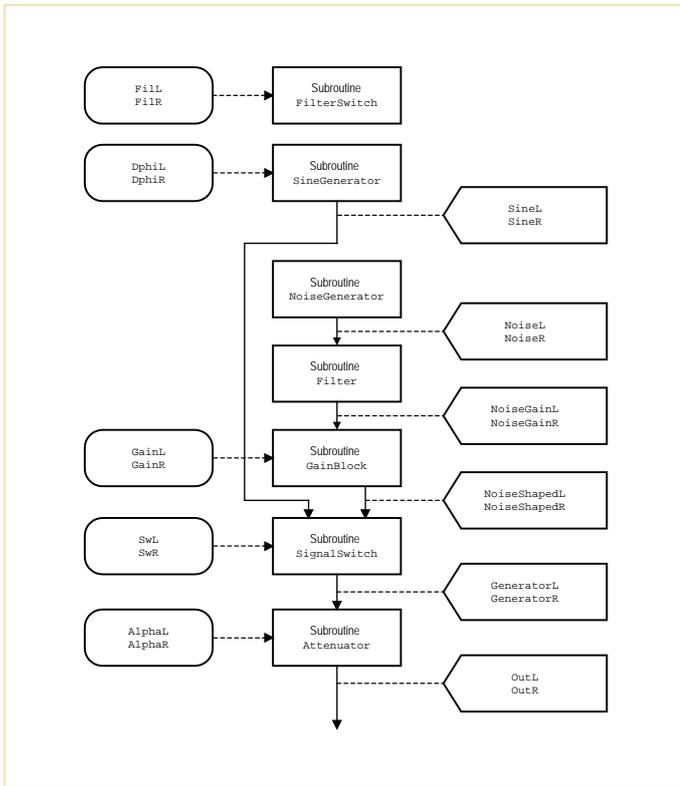


Bild 4. Unterprogramme und Signale in der Audioloop.

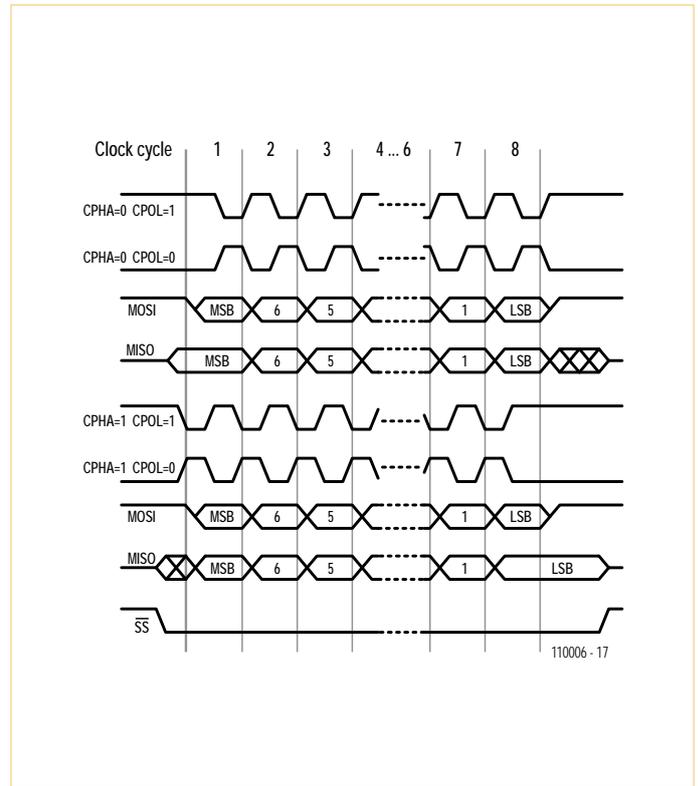


Bild 5. Timing der SPI-Schnittstelle, Bild aus Wikipedia.

senkt den Ausgangspegel um 6 dB. In der Rauschsignalverstärkerstufe ist für beide Kanäle ein Verstärkungsmaß von 0 dB voreingestellt. Die Defaultparameterwerte sind in der Tabelle 4 abgelegt. Die für das DSP-Programm benötigten Dateien werden in der Tabelle 5 aufgeführt.

Unterprogramme und Signale in der Audioloop

Zwei Unterprogramme sind vor dem Eintreten in die Audioloop auszuführen. Das Unterprogramm `InitStateVars` löscht die Zustandsspeicher der Filter und die beiden Phasenakkumulatoren. Die Schieberegisterzustände werden auf einen erlaubten Wert gesetzt. Mit dem Unterprogramm `SetDefaultParams` werden die Parameter entsprechend den Angaben in der Tabelle 4 gesetzt.

Innerhalb der Audioloop sind sieben Unterprogramme auszuführen, von denen sechs die Audiosignalverarbeitung erbringen. Das Unterprogramm `FilterSwitch` liest die beiden Filterindexparameter `FiL` und `FiR` und berechnet daraus die effektiven Start-Adressen des korrespondierenden Filterkoeffizientenfelds mit der Länge 15. Dieser Vorgang hat nichts mit Signalverarbeitung zu tun, sondern es handelt sich um eine Parameterübersetzung. Sie findet in der

Audioloop statt, da es möglich sein muss, Parameter im laufenden Betrieb zu ändern. Da die effektiven Adressen berechnet werden müssen und die anderen Parameter ohne weitere Berechnung verwendet werden können, muss diese Berechnung innerhalb der Audioloop stattfinden.

Das Unterprogramm `SineGenerator` enthält die beiden voneinander unabhängigen Sinusgeneratoren für den linken und für den rechten Kanal. Die beiden Phaseninkrements-Parameter `DphiL` und `DphiR` legen die Schwingungsfrequenz fest. Die eigentliche Berechnung der Sinusfunktion mit einem Näherungspolynom ist im Makro `sine` abgelegt. Die Zustandsvariablen des Unterprogramms sind die beiden Phasenakkumulatorregister `l:PhaseAccuL` und `l:PhaseAccuR`, in denen die aktuellen Phasenwerte gespeichert werden. Im Weiteren werden die sechs Näherungspolynomkoeffizienten benötigt, die im Y-RAM ab der Basisadresse `SinCoef` abgelegt sind. Die beiden Sinussignale lassen sich bei `y:SineL` und `y:SineR` beobachten.

Das Unterprogramm `NoiseGenerator` enthält die beiden für die Praxis statistisch voneinander unabhängigen Rauschsignalgeneratoren für den linken und für den rechten Kanal. Es wird ebenfalls ein Makro

zur Rauschsignalerzeugung verwendet. Das Makro hat zwei Argumente, einen Speicherplatz für das Schieberegister und eine 24-Bit-Konstante zur Festlegung der Rückkopplungen. Die Zustandsvariablen sind die beiden 24-bit-Schieberegister `y:NoiseL` und `y:NoiseR`, die gleichzeitig auch die beobachtbaren Ausgangssignale der beiden Rauschsignalgeneratoren sind.

Das Unterprogramm `Filter` enthält die beiden Rauschformungsfilter. Es sind *rekursive* oder *IIR*-Filter der Ordnung 6. Zur Anwendung kommt ein Makro `iir2mac` aus der Programmibliothek von Freescale, ehemals Motorola. Das Makro kann für IIR-Filter beliebiger Ordnung verwendet werden und ist daher ein Standardprogramm. Das Unterprogramm benötigt keine Parameter, da die Basisadressen beider Koeffizientenfelder der Länge 15 bereits mit dem Unterprogramm `FilterSwitch` berechnet wurden. Es werden zwei Zustandsräume mit jeweils acht Einträgen benötigt, deren Basisadressen `FstateL` und `FstateR` sind. Die beiden gefilterten Rauschsignale können an `y:NoiseGainL` und `y:NoiseGainR` beobachtet werden.

Das Unterprogramm `GainBlock` enthält die Verstärkungsstufe für die gefilterten Rauschsignale. Die Signalverstärkung ist nicht gegen Übersteuerung geschützt und

Terzfilter und Rauschsignalverstärkung

Mit unserem Generator erzeugen wir Terz- und Oktavbandrauschen, das wir aus gefiltertem weißem Rauschen erzeugen. Wenn wir für das weiße Rauschen bei einer Bandbreite von 0 bis 24 kHz einen Bezugspegel von $L_R=0\text{dB}$ annehmen, ist der Pegel für ein gefiltertes Rauschen niedriger, da ja nur ein Teil der Rauschleistung ein Filter passiert. Das wollen wir im Folgenden etwas genauer betrachten. Die Bandbreiten B (die Abstände zwischen den unteren und den oberen Grenzfrequenzen) für ideale Filter betragen in Abhängigkeit der Filtermittenfrequenz f_m

$$\text{Terzfilter } B = (2^{1/6} - 2^{-1/6}) * f_m = 0,2316 * f_m$$

$$\text{Oktavfilter } B = (2^{1/2} - 2^{-1/2}) * f_m = 0,7071 * f_m.$$

Mit diesen Bandbreiten können wir die jeweiligen Leistungs-Pegel in Dezibel berechnen:

$$\begin{aligned} \text{Terzfilterpegel } L_{\text{Terz},f_m} &= L_R + 10 * \log_{10}(f_m) + 10 * \log_{10}(0,2316/24000) \\ &= L_R + 10 * \log_{10}(f_m) - 50,1547 \end{aligned}$$

$$\begin{aligned} \text{Oktavfilterpegel } L_{\text{Oktav},f_m} &= L_R + 10 * \log_{10}(f_m) + \\ &10 * \log_{10}(0,7071/24000) = L_R + 10 * \log_{10}(f_m) - 45,3073. \end{aligned}$$

Wir fassen die Ergebnisse in der folgenden Tabelle zusammen:

Rauschpegel des gefilterten Rauschens		
Mittenfrequenz f_m	Terzfilterpegel L_{Terz,f_m}	Oktavfilterpegel L_{Oktav,f_m}
25	-36,18	
31,5	-35,17	-30,32
40	-34,13	
50	-33,17	
63	-32,16	-27,31
80	-31,12	
100	-30,15	
125	-29,19	-24,34
160	-28,11	
200	-27,14	
250	-26,18	-21,33
315	-25,17	
400	-24,13	
500	-23,17	-18,32
630	-22,16	
800	-21,12	
1000	-20,15	-15,31
1250	-19,19	
1600	-18,11	
2000	-17,14	-12,3
2500	-16,18	
3150	-15,17	
4000	-14,13	-9,29
5000	-13,17	
6300	-12,16	
8000	-11,12	-6,28
10000	-10,15	
12500	-9,19	
16000	-8,11	-3,27
20000	-7,14	

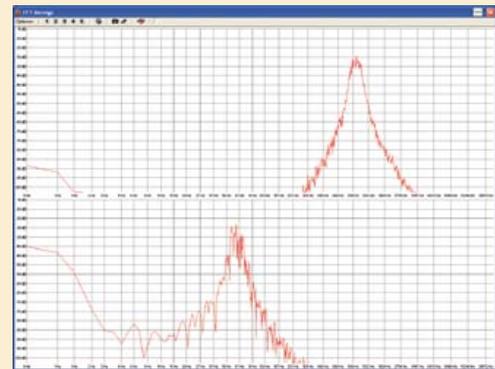


Bild A. GainL = 2, GainR = 3, keine Übersteuerung

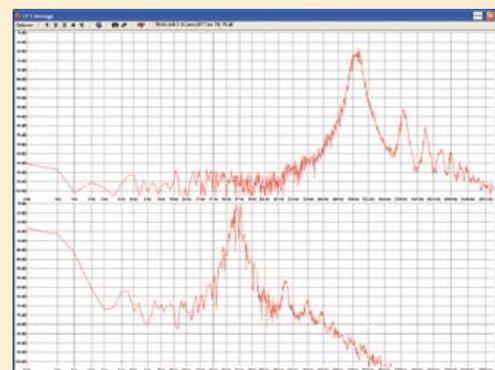


Bild B. Übersteuerung, nichtlineare Verzerrungen durch Signalbegrenzung, GainL = 3, GainR = 5.

Es ist leicht nachzurechnen, dass die Terzfilterpegel um 1 dB pro steigende Terz und die Oktavfilterpegel um 3 dB pro steigende Oktave ansteigen. Wenn wir, vor allem für die niedrigen Mittenfrequenzen, höhere Pegel benötigen, müssen wir die Signale verstärken, was mit den Parametern GainL und GainR in 6-dB-Schritten vorgenommen werden kann. Wenn wir aber den Pegel durch Verstärkung genau ausgleichen, dann wird der Generator übersteuert, was zu einer Verfälschung des Signalspektrums führt, denn wir müssen für die Wahl der Verstärkung den Spitzenwert des Signals und nicht seinen Pegel berücksichtigen. Ein brauchbarer Richtwert sind die tabellierten Pegel aber schon. Dennoch sollten wir im konkreten Fall das Spektrum des verstärkten Signals überprüfen. Die **Bilder A und B** zeigen den Effekt der Übersteuerung von Terzbandrauschen. Mit unserem Generator erzeugen wir zweikanaliges Terzbandrauschen, links mit der Mittenfrequenz $f_m = 1\text{ kHz}$ und rechts mit der Mittenfrequenz $f_m = 80\text{ Hz}$. Für das Bild A, erzeugt mit Wavelab, haben wir die Verstärkungen links zu 12 dB und rechts zu 18 dB gewählt. Für das Bild B wurden die Verstärkungen links zu 18 dB und rechts zu 30 dB gewählt. In diesem Bild kommen zu den Terzrauschspektren noch gut zu erkennende Verzerrungsanteile hinzu, die ihre Ursachen in der Signalbegrenzung aufgrund von Übersteuerung haben. Wenn man eine hohe Verstärkung nutzen möchte, sollte man mit einem Spektralmonitor überprüfen, ob Übersteuerung vorliegt oder nicht.

Jetzt erhältlich!

Sie erhalten das DSP-Board inklusive Programmieradapter jetzt zu einem speziellen Preis, damit Sie selbst praktische Erfahrungen mit diesem Kurs sammeln können.

Siehe www.elektor.de/110001-92.

sollte daher auch nur dann verwendet werden, wenn das gefilterte Rauschsignal nur über wenig Aussteuerung verfügt. Genaue Angaben hierzu findet der Leser in einem separaten Abschnitt. Die Programmausführung ist sehr einfach. Ein Verstärkungsmaß von 6 dB, (Verdoppelung der Abtastwerte) wird mit einem Linksshift ausgeführt. Da der DSP über einen Barrel- oder Multibitshifter verfügt, kann er auch mehrstelliges Linksshift für ein Verstärkungsmaß als Vielfaches von 6 dB in einem einzigen Schritt ausführen. Die beiden gefilterten und gegebenenfalls verstärkten Rauschsignale können an `y:NoiseShapedL` und `y:NoiseShapedR` beobachtet werden. Das Unterprogramm `SignalSwitch` enthält die beiden Signalquellenumschalter, mit denen zwischen Sinus- und Rauschsignalen umgeschaltet wird. Die Schalter werden mit den beiden Parametern `SwL` und `SwR` betätigt. Zustandsvariablen und Koeffizienten benötigt das Unterprogramm nicht. Die durch die beiden Schalter gewählten Signale können unter `y:GeneratorL` und `y:GeneratorR` beobachtet werden. Das letzte Unterprogramm in der Audio-loop ist das Unterprogramm `Attenuator`. Hier werden die beiden Signalabschwächer direkt vor der Audioausgabe gerechnet. Die Abschwächfaktoren werden mit den beiden Parametern `AlphaL` und `AlphaR` festgelegt. Zustandsvariablen und Koeffizienten benötigt das Unterprogramm ebenfalls nicht. Die abgeschwächten Signale können an `y:OutL` und `y:OutR` beobachtet werden und werden an die Audioausgabe übergeben.

Parametereinstellungen

Um bei unserem Audiogenerator im Betrieb die Signale zu ändern, müssen wir zweimal fünf Parameter für die beiden Kanäle ändern. Im Moment müsste dies so erfolgen, dass im DSP-Programm im Unterprogramm `SetDefaultParams` die gewünschten Parameterwerte eingetragen werden, das Programm neu assembliert und via Debugger in den DSP geladen wird. Viel eleganter wäre es, eine Eingabeeinheit mit Display zu verwenden, um die Parameter seriell via SPI in den DSP zu schreiben. Es ist auch möglich, den DSP selbst beispielsweise an ein Tastenfeld oder ein Digi-

talpotentiometer und ein übliches Text-LCD anzuschließen, aber dies wird durch das zeitliche Nebeneinander von Reaktion auf Benutzereingaben und Audiosignalverarbeitung erschwert. Man sollte beide Vorgänge daher getrennt halten und zum Beispiel einen Mikrocontroller für die Eingabeeinheit verwenden. Wir wollen uns hier nicht auf einen bestimmten Controller festlegen, da es eine große Anzahl an geeigneten μC gibt. Diese werden entweder ebenfalls in ihrer Assemblersprache oder (komfortab-

der Abfragen kann man ein Vielfaches des Audiotakts verwenden. Wir haben in denjenigen Testprogrammen aus dem vierten Artikel, mit denen der DSP Sinustestsignale erzeugt, bereits einen Audiotaktzähler für 192 Werte verwendet. Das nachstehende DSP-Code-Segment zeigt eine bidirektionale DSP-SPI-Kommunikation mit Pollen der Schnittstellenflags. Man kann wenigstens prinzipiell ersehen, was alles notwendig ist, auch wenn im konkreten Fall Einstellungen anzupassen sind.

```
bclr    #HEN,x:HCSR        ; SHI disable, SPI-Reset
movep  #$002048,x:HCKR    ; Cpol=0, Cpha=0, narrow spike filter,
                                ; f=Fosc / 2 / 8 / 10 = 0.9216MHz
movep  #$000040,x:HCSR    ; 8-Bit, Master-Betrieb, FIFO off
bset   #HEN,x:HCSR        ; SHI enable

...

move   #Buffer,r0
do     #N,RW_MuC
    jclr #HTDE,x:HCSR,*    ; transmit register empty?
    movep x:(r0),x:HTX
    jclr #HRNE,x:HCSR,*    ; receive register full?
    movep x:HRX,y:(r0)+
RW_MuC
```

ler) in einer Hochsprache programmiert. Der Autor verwendet für solche Zwecke μC -Veteranen vom Typ 68HC11, die er den Lesern nicht mehr empfehlen möchte, auch wenn sie mit ihrer Prozessorarchitektur besonders gut zu unserem DSP passen. Die SPI-Schnittstelle im DSP, zu **betreiben als Master**, ist mit den beiden Peripherie-Registern HCSR und HCKR zu konfigurieren. Das heißt, man stellt Taktrate, Takt polarität und -phase sowie die Wortlänge (zumeist 8 bit) ein. **Bild 5** zeigt die Timing-Varianten. Dann programmiert man ein Parameter-Set-Unterprogramm, das beispielsweise als ISR von einem IRQ aufgerufen wird, den der μC auslöst, wofür auf unserem DSP-Board der IRQC des DSPs genutzt werden kann. Der μC ist als **SPI-Slave** zu betreiben. Eine Alternative ohne Verwendung von externen Interrupts (nicht sehr elegant, dafür aber einfach) ist es, den DSP periodisch die Parameter beim μC abfragen zu lassen und im Falle einer Änderung zu übernehmen. Für das Timing immer wiederkehren-

Das Programmsegment schreibt und liest zwei Buffer mit der Länge N, einen Sendebuffer im X-RAM und einen Empfangsbuffer im Y-RAM, die beide denselben Adressbereich einnehmen. Mit den ersten vier Programmzeilen wird die Schnittstelle in den Reset-Zustand versetzt, konfiguriert und freigegeben.

Der Rest des Programmsegments enthält Lesen und Schreiben der beiden Buffer. An dieser Stelle lässt sich noch eine kleine Programmierübung vorstellen, mit der man eigene erste Programmierschritte gehen kann. Um den Sinusgenerator zu betreiben, verwenden wir das Phaseninkrement $d\phi = 2f_s/f_T$ als linksbündigen Fraktionalwert. Es wäre entschieden eleganter, die Signalfrequenz f_s als rechtsbündigen Integer-Parameter zu verwenden und den DSP das benötigte Phaseninkrement selbst berechnen zu lassen. Diese Parameterwahl käme auch der Anbindung des μC als Eingabeeinheit entgegen.

(110006)



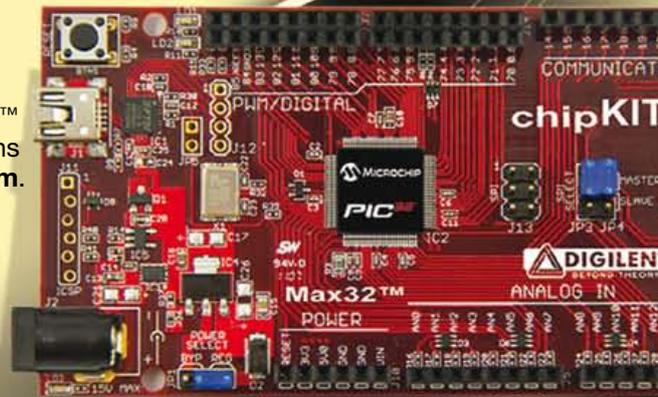
DESIGNSPARK

Turn a **hot**
idea into a
cool
solution.

DesignSpark chipKIT™ Challenge

DesignSpark is the perfect environment for engineers to work together, challenge each other to think outside the box and engage with fellow designers from around the world. Participants of the DesignSpark chipKIT™ Challenge are encouraged to post photos/videos, summaries, and questions about their projects, code, and PCB designs at www.chipkitchallenge.com. Register today and you could be selected for a chipKIT™ Community Choice Award!

Prizes include a \$100 voucher for RS Components/Allied Electronics and a free digital subscription to *Circuit Cellar* and *Elektor* magazines. Community Choice Awards will include a winner of the most creative project proposal, selected by the DesignSpark team, and most popular project, selected by the DesignSpark community.*



Visit www.chipkitchallenge.com

for complete rules and details.

* Participation in the Community Choice Awards does not increase your chances of winning the Grand Prize with your Final Project(s) submission. See website for more information.

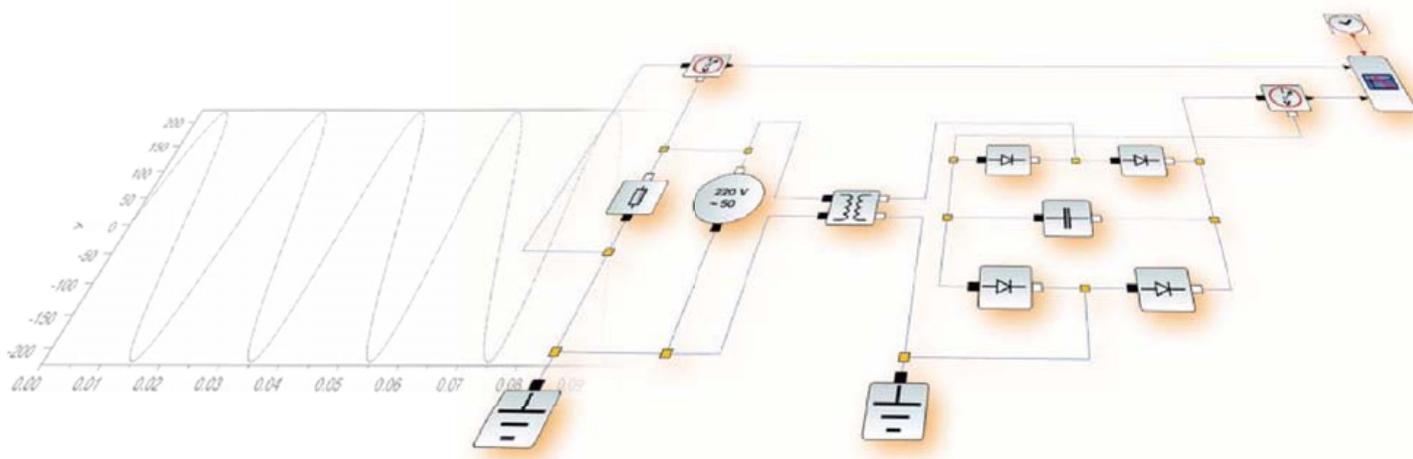
IN ASSOCIATION WITH:



Xcos: Modellierer und Simulator

Open-Source modellieren und simulieren

Im letzten Monat hatten wir uns mit Scilab befasst, ein Open-Source-Programm zum Lösen mathematisch-wissenschaftlicher Problemstellungen wie Modellrechnungen und Simulationen. Zu Scilab gehört Xcos, ein Modellierer und Simulator für dynamische Hybridsysteme.



Von Vincent Couvert, Clément David, Bruno Jofret und Julie Paul (Frankreich)

Auf einen kurzen Nenner gebracht: Xcos [1] ist für Scilab das, was Simulink [2] für Matlab ist. Xcos wurde vollständig in die Scilab-Umgebung eingebunden, es wird über das Menü *Applications* aufgerufen. Der Umgang mit Xcos kann als intuitiv und ergonomisch bezeichnet werden. Im Xcos-Editor sind Funktionsblöcke selektierbar, die in Paletten organisiert sind. Funktionsblöcke werden durch

Ziehen und Ablegen mit der Maus auf dem Arbeitsblatt platziert, zum Koppeln von Funktionsblöcken genügen wenige Mausklicks. So entsteht ein Modell, dessen Verhalten simuliert werden kann.

Schon die Standard-Installation von Xcos bringt zahlreiche Paletten und Funktionsblöcke für wichtige Aufgabenbereiche des Ingenieurs mit: Elektrotechnik, Signalverarbeitung, hydraulische Systeme, kontinuierliche und diskrete mathematische Verfahren. Die schnellste Methode, sich mit Xcos vertraut zu machen, ist das Betrachten eines bereits existierenden Modells. Über das Hilfemenü („?“) und *Scilab*

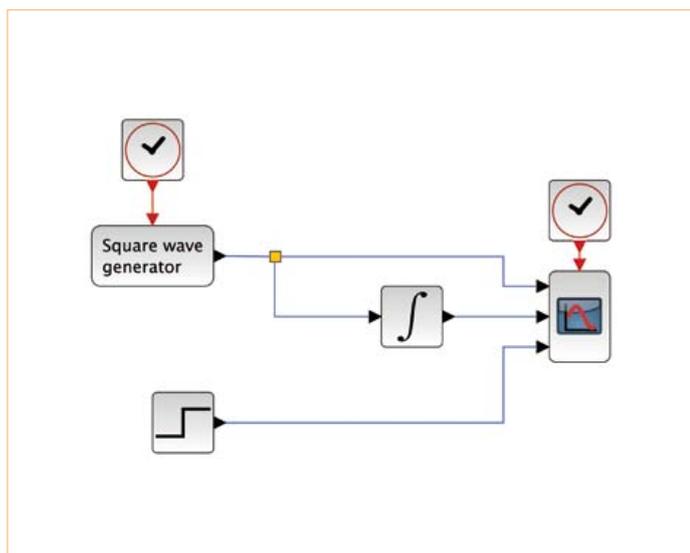


Bild 1: Ein einfaches Beispiel.

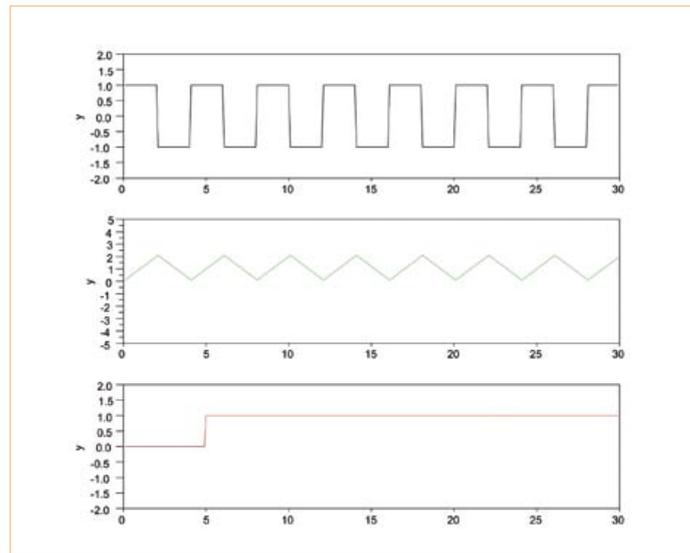


Bild 2: Ergebnis der Simulation.

für dynamische Hybridsysteme

Demonstrations gelangt der Anwender zu *Demos* und *Xcos*. Hier stehen viele Beispiele mit unterschiedlichen Schwierigkeitsgraden zur Wahl, von simpel bis hoch komplex, vom RC-Netzwerk bis zur Simulation der Radaufhängung an einem Kraftfahrzeug.

Innerhalb der gewählten Palette kann der Anwender auch eigene Funktionsblöcke definieren. Unterstützt werden die Sprachen Fortran, C und Scilab, was Matlab ähnlich ist, sowie Modelica. Mit Modelica ist im Prinzip jedes physische System auf der Basis seiner charakteristischen Gleichungen und Übergangsfunktionen beschreibbar. Zu den Stärken von Xcos gehört das Lösen mess- und regelungstechnischer Aufgaben in der Elektrotechnik und im Maschinenbau.

In Xcos kann der Anwender vieles den aktuellen Gegebenheiten und Erfordernissen anpassen. Durch individuelles Organisieren von Paletten und Funktionen entstehen neue externe Module, die mit Applikationen starten können.

Getreu dem Open-Source-Prinzip, dass alles Wissen teilbar ist, lassen sich Blöcke und Paletten an andere Anwender verschicken. Ein Portal für externe Module ist ATOMS, auf diese Module kann der Anwender über die Scilab-Konsole zugreifen.

Beispiel 1: Erste Schritte mit Xcos

Am Anfang einer Simulation mit Xcos steht immer ein Simulationsschema. Jeder Block repräsentiert eine Funktion mit einer Anweisung, die während der Simulation ausgeführt wird. Von der Struktur des Schemas hängt die Abfolge bei der Ausführung ab. Die ausführbaren Blöcke werden in der Grafik durch Linien miteinander verbunden. Ein Linientyp symbolisiert Datenflüsse, während der andere Linientyp Wege beschreibt, über die Aktionen und Ereignisse ausgelöst werden.

Funktionsblöcke treten in Aktion, sobald sie „aktiviert“ werden, ent-

weder als Folge von Ereignissen (rote Ports und Linien) oder beim Eintreffen bestimmter Informationen (über blaue Linien). Zu jedem Block gehören Parameter, die für sein Verhalten maßgeblich sind. Ein einfaches Beispiel ist die Simulation einer Schrittfunktion (Heaviside-Funktion), bei der nur die Zeit und die Schrittamplitude die Eigenschaften des Blocks bestimmen.

In dem Beispiel, das **Bild 1** zeigt, sind drei Signale vorhanden: Ein Rechtecksignal, das gleiche Signal, das durch einen Integrator läuft, sowie das Signal einer Schrittfunktion. Die Eigenschaften aller Blöcke müssen konform zum Gesamtsystem sein, jeder Block muss sich konsistent zu den übrigen Blöcken verhalten. In Xcos können die Blockparameter nach Doppelklicks auf die Blöcke betrachtet und modifiziert werden. Das Ergebnis der Simulation in **Bild 2**, das vom Oszilloskop-Block als Funktion der Zeit dargestellt wird, überrascht kaum. Wie bei einem realen Funktionsgenerator entsteht aus dem Rechtecksignal durch Integration ein Dreiecksignal. Ferner fällt auf, dass die Flankensteilheit auch in der Simulation nicht unendlich ist. Oszilloskop-Blöcke in Xcos arbeiten ebenso wie reale digitale Oszilloskope mit einstellbaren Sample-Frequenzen. Nach Einstellen der Periodenzeit auf 1.0 in der Simulation werden die nicht idealen Flankensteilheiten, wie **Bild 3** zeigt, recht deutlich sichtbar. Diese Erscheinung ist die Folge eines Undersamplings. Ferner ist ersichtlich, dass der Integrator seine Funktion trotzdem erfüllt. Ursächlich für das nicht ideale Verhalten sind folglich die Einstellungen, nicht der Simulator!

Xcos war ursprünglich für Anwendungen in der Automatisierung gedacht, durch die Palette seiner Möglichkeiten sind jedoch die Einsatzbereiche weit gestreut. Das beweist das nächste Beispiel, die Modulation und Demodulation eines Sinussignals. In **Bild 4** ist das Simulationsschema des Gesamtsystems dargestellt, während **Bild 5** nur den Modulator zeigt. Es handelt sich hier um die mathemati-

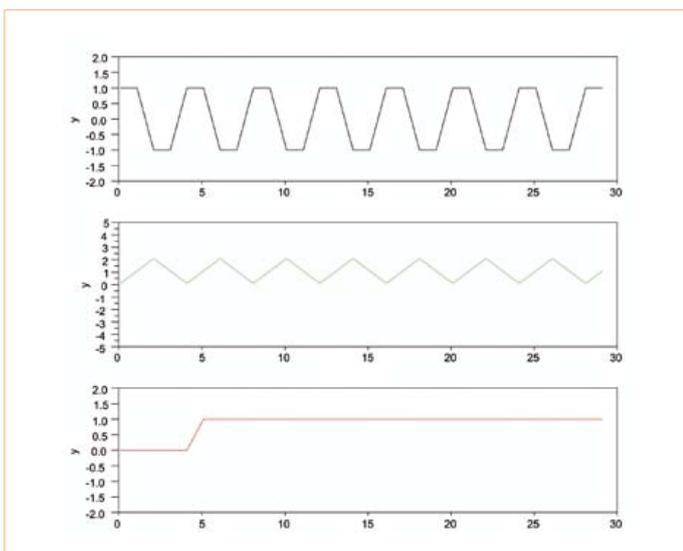


Bild 3: Das Problem ist die Abtastrate.

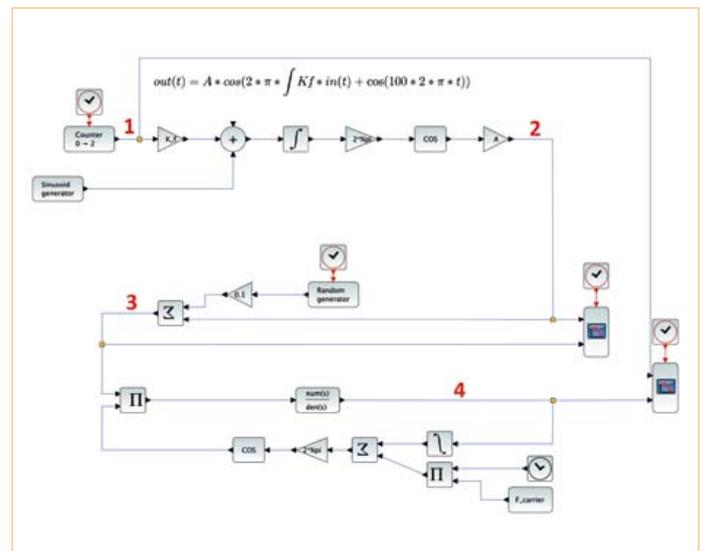


Bild 4: Simulationsschema einer FSK-Übertragung.

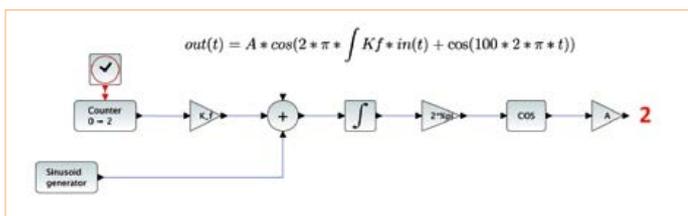


Bild 5: Modulation des Signals.

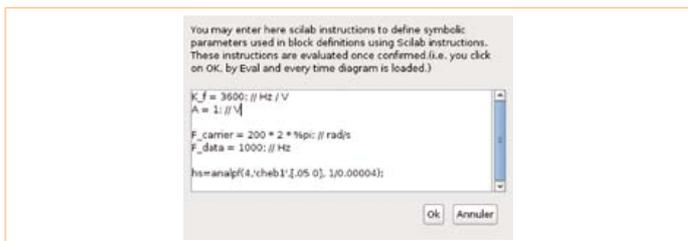


Bild 6: Funktionsblöcke sind auch auf diesem Weg definierbar.

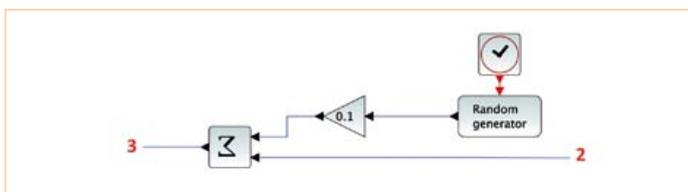


Bild 7: Modell eines Übertragungskanals.

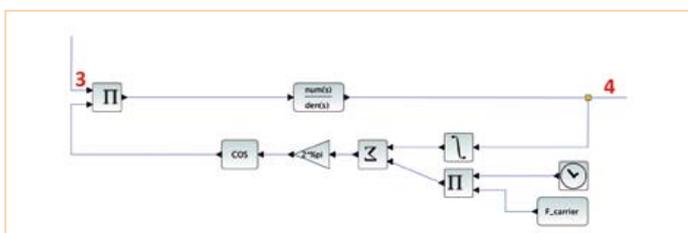


Bild 8: Demodulation des frequenzmodulierten Signals.

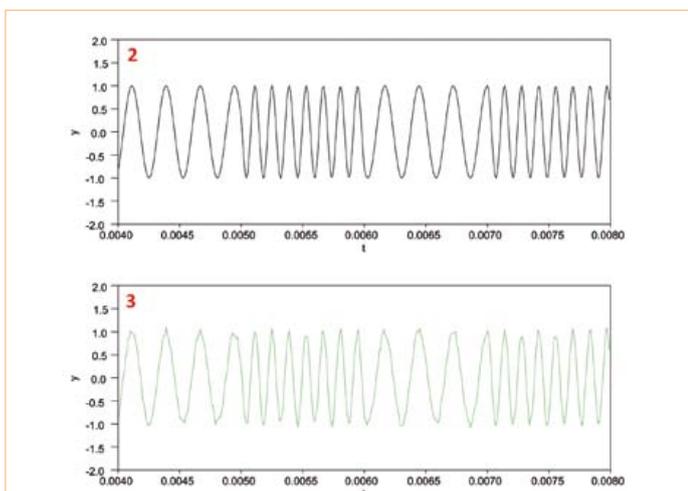


Bild 9: Moduliertes Signal (2), überlagert durch Rauschen (3).

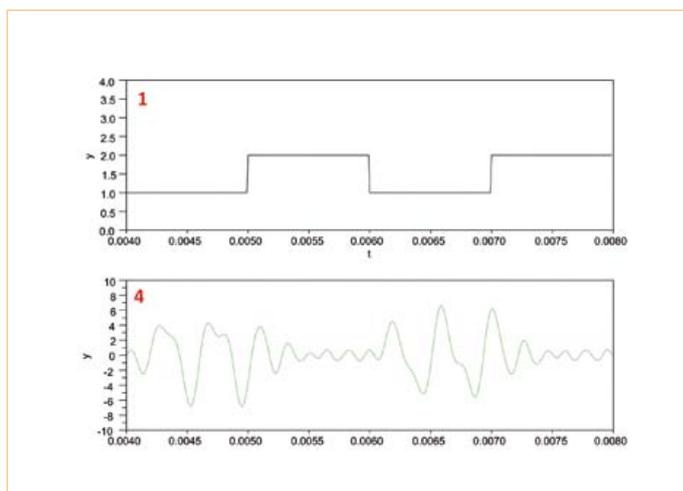


Bild 10: Signale am Eingang (1) und Ausgang (4).

sche Beschreibung eines FM-Signals in Form von Funktionsblöcken. Das zu übertragende Signal wird vom Block *Counter* generiert, die Frequenz hängt vom daran gekoppelten Taktgeber (Uhr-Symbol) ab. Der mathematische Ausdruck ist eine Annotation, die mit der LaTeX-Engine von Scilab geschrieben wurde, sie dient nur zur Verdeutlichung. Anzumerken ist noch, dass alle Blöcke zur Kategorie der kontinuierlichen Funktionen gehören, simuliert wird hier ein analoger FM-Modulator.

Die Parameter der Funktionsblöcke können entweder in Scilab oder über das Kontextmenü von Xcos definiert werden (Bild 6). Beispielsweise ist der Wert von A die Amplitude des Ausgangssignals, hier eingestellt auf 1. Der Scilab-Programmcode, in dem die Anweisungen geschrieben werden, gibt Zugriff auf alle vordefinierten Scilab-Funktionen, zum Beispiel steht *anapf* für einen *Analog Low Pass Filter*. Um näherungsweise das zu modellieren, was dem Signal auf dem Übertragungskanal widerfährt, wird ihm in Bild 7 von einem Zufalls-generator über einen Abschwächer (*gain=0.1*) weißes Rauschen zugefügt. Natürlich ist dies nur eine grobe Vereinfachung der realen Verhältnisse. Das vollständige Modell eines Übertragungskanals würde aus einer Vielzahl von Xcos-Blöcken bestehen, das Signal selbst hat ebenfalls Einfluss auf das Modell.

Anschließend soll das FM-Signal von einer PLL (*phase locked loop*) demoduliert werden. Auch hier ist das Simulationsschema (Bild 8) eine direkte Umsetzung der mathematischen Beschreibung. Die folgenden Hinweise sollen dem besseren Verständnis dienen:

- Der Taktgeber generiert hier Werte, keine Ereignisse. Es soll simuliert werden, wie sich das Signal (Y-Achse) abhängig von der Zeit (X-Achse) verhält. Das ist nur möglich, wenn die Werte als Funktion der Zeit aufgetragen werden.
- In diesem Beispiel besteht die Schleife aus analogen Blöcken kontinuierlicher Signale. Eine diskrete Variante könnte natürlich ebenfalls simuliert werden. Abgetastete Signale lassen sich mit einem Modulo-2p-Zähler und definierter Frequenz modellieren.

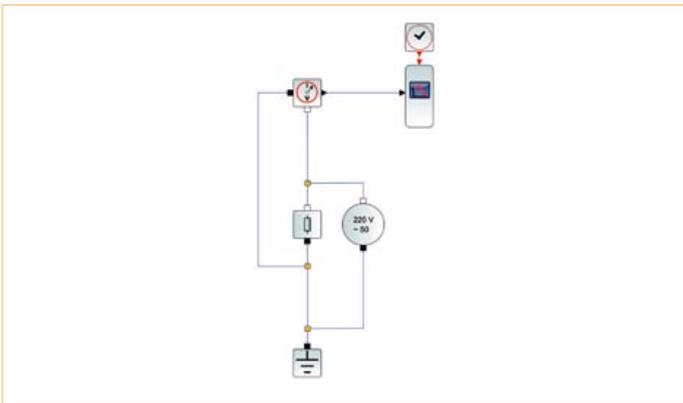


Bild 11: Simulationsschema einer Spannungsquelle.

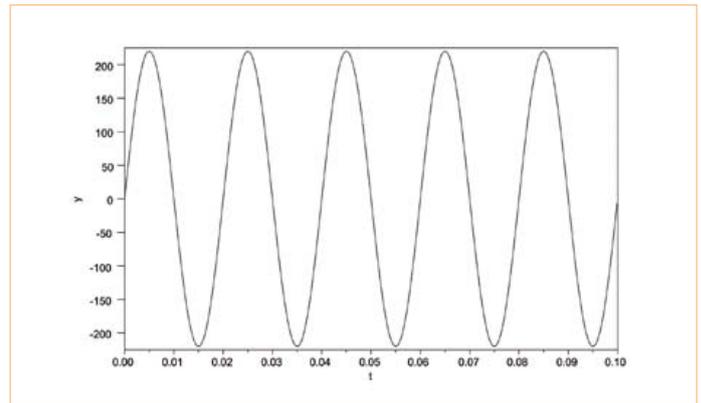


Bild 12: Die Spannung auf dem Oszilloskop-Modul.

Die Blöcke der diskreten Funktionen sind in der Palette für diskrete Systeme zu finden.

- Die Parameter des Schleifenfilters gelten ausschließlich für den jeweiligen Block. Das Implementieren der Übertragungsfunktion beschränkt sich auf das Eingeben eines mathematischen Polynoms.
- Die **Bilder 9** und **10** geben die Signale an den diversen Messpunkten des Simulationsschemas in Bild 4 wieder. Nachdem die ersten Schritte in Xcos getan sind, kann sich der Anwender an eigene Aufgabenstellungen heranwagen. Hier sind drei Vorschläge, die an vorstehende Beispiele anknüpfen:
- Hinzufügen einer Hysterese,
- Erstellen eines einfachen digitalen oder analogen Filters...
- oder eines erweiterten Amplituden-Demodulators.

Beispiel 2: Simulieren eines Spannungswandlers 220 V ~ / 12 V = mit Modelica

Zum Spannungswandler gehören eine Wechselspannungsquelle, ein Transformator, ein Brückengleichrichter und ein Kondensator, aus 220 V ~ sollen hier 12 V = entstehen. Es geht folglich um ein einfaches Netzteil, für das Beispiel wurde 220 V als typischer Wert für den unteren Bereich der Netzspannung ($230\text{ V} \pm 10\%$) angenommen. In der Sprache *Modelica* wird die Simulationsfunktion mit dem *Modelica-Solver* erstellt. Die Simulation beruht im Wesentlichen auf dem Bestimmen von Werten aus der Simulationsfunktion, was *Modelica* übernimmt. Mit Xcos kann das Ergebnis im Zeitbereich betrachtet werden. Die Grafik zeigt nebeneinander die kontinuierliche und diskrete Darstellung.

Verglichen mit dem ersten Beispiel haben Funktionsblöcke in *Modelica* eine abweichende Semantik. In *Modelica* existieren keine Aktivierungsverknüpfungen. Beim Spannungswandler können jedoch alle auftretenden Variablen als mathematische Ausdrücke geschrieben werden. Die Variablen sind als Signale auf Datenleitungen interpretierbar.

Die Simulation der unten stehenden Schemen setzt einen C-Compiler voraus. Die erstellte Funktion wird von Scilab während der Simulation geladen.

Der erste Funktionsblock ist der Block der Spannungsquelle, in diesem Beispiel liefert sie die Wechselspannung 220 V, 50 Hz (**Bild 11**). Die Spannung wird vom parallel liegenden Voltmeter gemessen. Wie in der Realität muss auch in der Simulation ein Massepunkt vorhanden sein. Ein Oszilloskop-Block stellt das Ergebnis dar (**Bild 12**). Die Spannungsquelle wird mit einem Transformator verbunden (**Bild 13**), der im Modell ideal und verlustfrei arbeitet. Der Transfor-

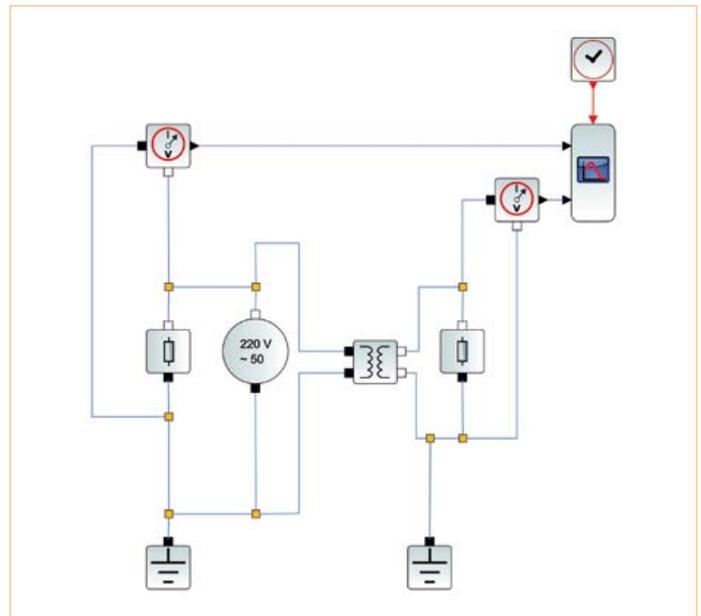


Bild 13: Ein idealer Transformator an der Spannungsquelle.

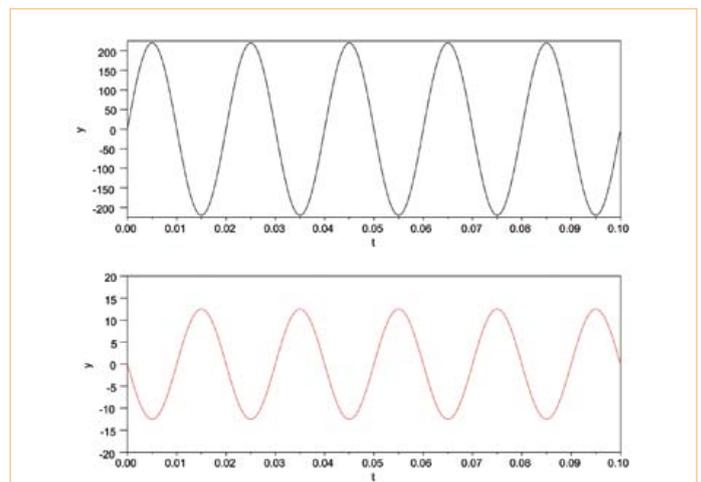


Bild 14: Die Primär- und Sekundärspannung des Trafos.

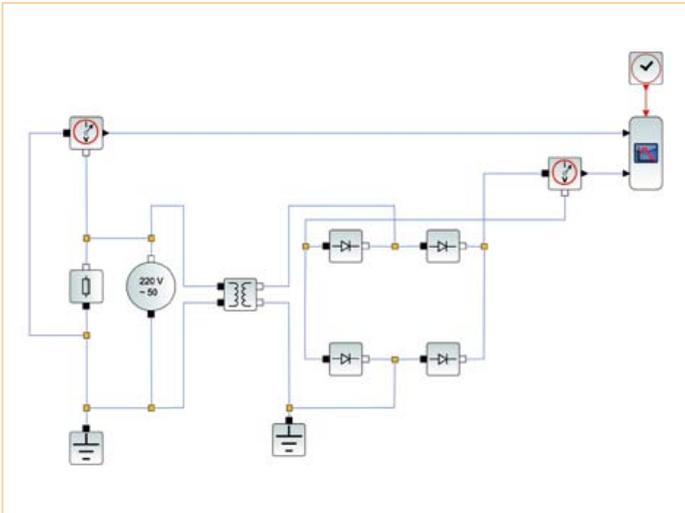


Bild 15: Ein Brückengleichrichter wird angeschlossen.

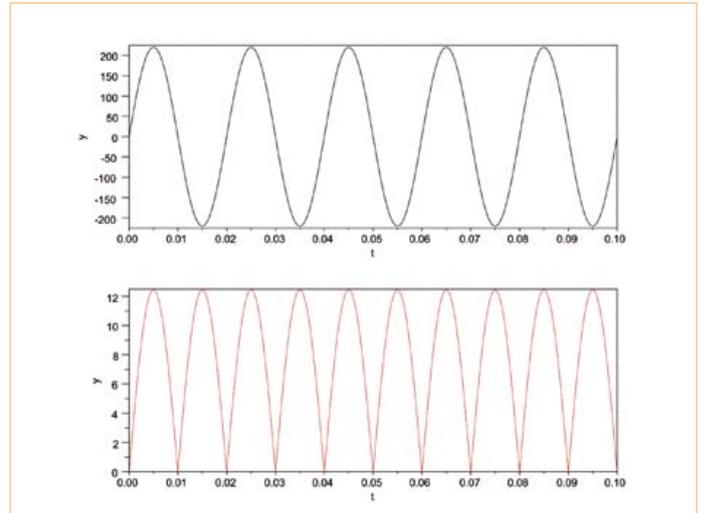


Bild 16: Primärspannung und pulsierende Gleichspannung.

mator-Ausgang liefert die Wechselspannung 12 V (unten in Bild 14), sie muss anschließend gleichgerichtet werden. Dazu dient ein Brückengleichrichter (Bild 15), das Ergebnis ist aus Bild 16 ersichtlich. Die pulsierende Gleichspannung wird von einem Kondensator geglättet, so dass die Gleichspannung 12 V entsteht. In Bild 17 und 18 sind das Gesamtsystem sowie die Spannungen am Eingang und Ausgang dargestellt.

Mit den Funktionsblöcken können nach dieser Methode auch andere elektronische Systeme modelliert werden, beispielsweise Funktionsgeneratoren, Gleichspannungswandler oder Schaltnetzteile. Die Modelle sind als Komponenten in anderen Simulationsschemen verwendbar, die beispielsweise mechanische oder thermische Systeme abbilden. Unabhängig von der Art der Systeme liefern Simulationen wertvolle Aufschlüsse über das Systemverhalten. Der Bau von Prototypen wird meistens überflüssig.

Zum Schluss

Das hohe wissenschaftliche Niveau von Scilab und die beachtlichen Stärken von Xcos sind in dieser Bündelung ein Werkzeug, das in vie-

len Disziplinen der Forschung und Entwicklung seinen Platz hat. Das Modellieren und Simulieren von Systemen spart nicht nur Zeit und Geld, es wirkt sich auch schonend auf die Materialressourcen aus. Wenn ein entwickeltes System in der Simulation die vorgegebenen Ziele erreicht, wird sich das äquivalente reale System mit hoher Wahrscheinlichkeit in identischer Weise verhalten.

Scilab und Xcos sind keine kommerziellen Produkte, sie sind frei und kostenlos erhältlich. Auch das hat dazu beigetragen, dass Scilab mit Xcos in Entwicklungszentren der Industrie ebenso wie im universitären Bereich Verwendung findet.

(110492)gd

Weblinks

- [1] Scilab : www.scilab.org
- [2] Simulink: www.mathworks.com/simulink
- [3] Beispiele: www.elektor.de/110492

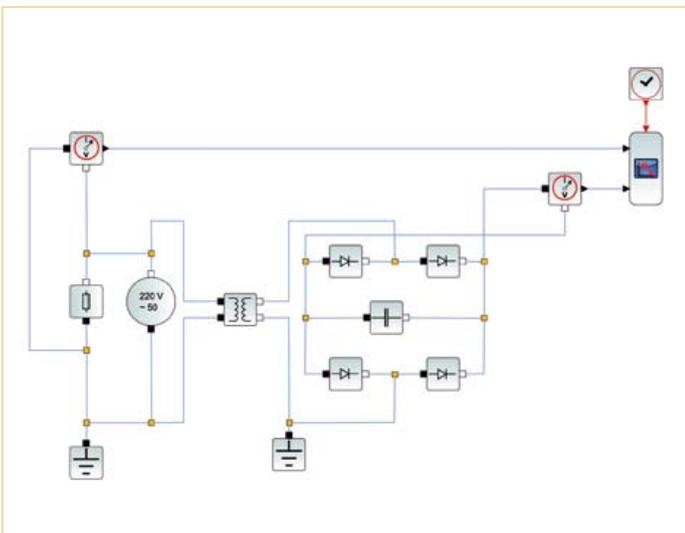


Bild 17: Ein Kondensator glättet die pulsierende Gleichspannung.

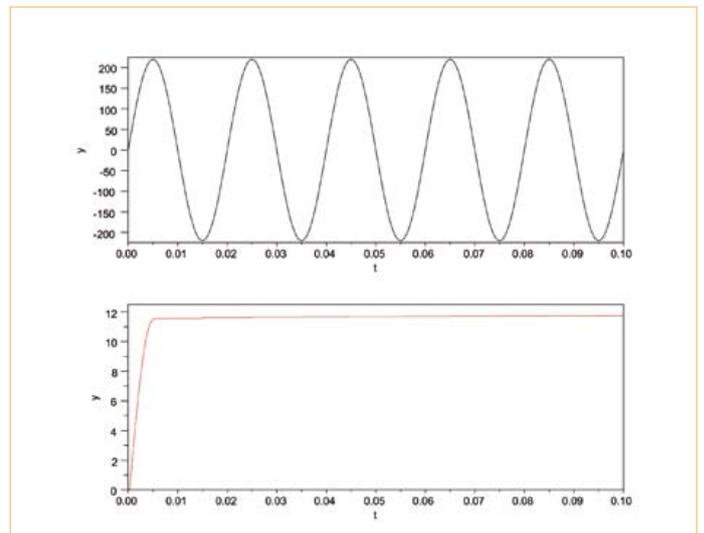


Bild 18: Geglättete Gleichspannung mit Einschaltknick.

„Elektor ist Wissen für Profis und für die, die es werden wollen. Der perfekte Einstieg in mein Studium!“

– Christian, 19 Jahre, Abiturient –



30%
Preis-
Vorteil

Elektor ist Wissen für Profis.

Sichern Sie sich jetzt Ihren Wissensvorsprung durch ein Abo!

Ihre Vorteile im Abo:

- Preisvorteil: Schüler und Studenten sparen **30%** gegenüber dem Einzelkauf
- Abonnenten erhalten exklusiv auf spezielle Elektor-Produkte bis zu **40% Rabatt**
- Keine Ausgabe verpassen: Elektor kommt pünktlich, bequem und zuverlässig frei Haus
- Immer up-to-date: Lesen Sie jede Elektor-Ausgabe vor allen anderen



www.elektor.de/studentenabo • Tel. 02 41/88 909-0

Personal Download for I © Elektor

elektor

Robusta: Studenten bauen einen Satelliten

Picosatelliten bringen die Ausbildung voran

Von Frederic Giamarchi (F)

Frankreich ist bekannt für seinen bedeutenden Beitrag zum europäischen Weltraumprogramm und für seine leistungsfähige Weltraumindustrie. Weniger bekannt ist es für die Ausbildung auf diesem Gebiet. Deshalb hat das Raumforschungszentrum CNES (Centre national d'études spatiales) Expresso ins Leben gerufen. Im Rahmen dieses Programms wurden 2006 die ersten Projekte an Hochschulen gestartet. Eines davon ist „Robusta“ der Universität Montpellier 2, ein Picosatellit vom Typ *CubeSat*.



Der Satellit Robusta (*Radiation On Bipolar University Satellite Test Application, Bild 1*) [1] soll Anfang 2012 beim Erstflug der neuen Europarakete Vega in einen elliptischen Orbit von 340 km bis 450 km mit einer Bahnneigung von 71° gebracht werden. Seine Nutzlast besteht aus einem wissenschaftlichen Versuchsaufbau zur Messung der Alterung (Degradation) elektronischer Bauteile. Während des

ganzen Fluges sendet er die Messergebnisse der Testobjekte und verschiedene Zustandsgrößen an die Bodenstation, die sich auf dem Campus der Universität Montpellier befindet. Da Robusta verschiedenen Strahlungsquellen, dem Sonnenwind, den in den Strahlungsgürteln eingefangenen kosmischen Teilchen und der kosmischen Strahlung ausgesetzt ist, wird er sich der Erde langsam annähern und schließlich nach zwei Jahren beim Wiedereintritt in die Erdatmosphäre verglühen.

Robusta

Dieser Satellit hat eine echte wissenschaftliche Aufgabe: er misst die durch ionisierende Strahlungen verursachte Alterung von inte-

CubeSat

Satelliten vom Typ „CubeSat“ stammen aus einem Ausbildungsprogramm, das 2000 von der polytechnischen Universität Kalifornien (CalPoly) gestartet wurde [2], um Studenten konkrete Erfahrungen und Kenntnisse im Bereich der Weltraumforschung und -industrie vermitteln zu können.

Ein CubeSat ist ein würfelförmiger Satellit mit 10 cm Seitenlänge, 1 kg Maximalgewicht und 1 W Höchstleistung. Er besteht aus einer Nutzlast,

genannt, die dem mitgeführten Experiment entspricht, und einer Plattform, die verschiedene Elektroniksysteme zur Versuchssteuerung, zur Kommunikation mit der Bodenstation und zum Energiemanagement enthält. Das Ganze stellt das sehr kleine Äquivalent zu einem größeren konventionellen Satelliten dar, das denselben Bedingungen unterliegt wie ein großer Satellit, was thermische Schocks, extreme Vibrationen beim Start, die Strahlung und das Vakuum im All anbelangt.

grierten Schaltkreisen mit bipolaren Transistoren. Als Testobjekte wurden der Komparator LM139 und der Verstärker LM124 ausgewählt, welche oft in Satelliten verwendet werden. Deren Alterung wird anhand von Strom-, Spannungs-, Temperatur-, und Strahlungsdosis-Messungen quantifiziert (**Bild 2**). Diese Dosis korrespondiert mit der absorbierten Strahlungsenergie pro Masseinheit. Die Ergebnisse werden in der Bodenstation mit den Daten verglichen, die eine Testmethode liefert, die von Forschern des Uni-Labors IES (Institut d'Électronique du Sud, Montpellier) ausgearbeitet wurde [3][4].

Die Dauer der Mission ist auf zwei Jahre festgelegt. Die Daten werden zumindest alle zwölf Stunden gemessen. Danach werden sie mit einem im Amateurfunk verwendeten Protokoll auf Amateurfunk-Frequenzen zur Bodenstation in Montpellier übermittelt. Die Übertragung findet als *broadcast* statt, das heißt kontinuierlich jede Minute, unabhängig davon, ob der Satellit für die Bodenstation sichtbar ist oder nicht.

Wesentlich für den Erfolg der Mission ist das Energiemanagement. Die Versorgung erfolgt durch Lo-Ion-Akkus von Saft, die durch spezielle weltraumtaugliche Dreischicht-Solarzellen mit 27 % Wirkungsgrad aufgeladen werden.

Interne Struktur

Der mechanische Aufbau, die Dimensionierung und Anordnung der elektronischen Baugruppen (Boards) und das Launch-System mit einem p-pod (**Bild 3**) wurden von den Fachgruppen GMP (Génie Mécanique et Productique) und GEII (Génie Électrique et Informatique Industrielle) der IUT in Nîmes übernommen. Das Subsystem der Spannungsversorgung mit dem Management der Akkus und Solarzellen wurde ebenfalls von GEII beigesteuert. Das Controllerboard-Subsystem, das die verschiedenen anderen Boards steuert und die Messwerte speichert, die Software für den Mikrocontroller und Testempfänger-Elemente wurden von der Fachgruppe Polytech'Montpellier entwickelt. Das Subsystem für das eigentliche Experiment, welches die Testobjekte trägt, das Dosimeter und die Temperatursensoren wurden von den EEA-Diplom- und Master-Studenten der wissenschaftlichen Fakultät entworfen. Das Subsystem für die Funkübertragung und die Bodenstation wurden von Studenten des Hochfrequenz-Kurses erarbeitet.

Die in diesem Projekt verwendeten handelsüblichen Bauteile und Materialien sind bis auf wenige Ausnahmen wie Akku und Solarzellen nicht strahlungsgehärtet. Stattdessen wurde ein besonders strenges Qualitätssicherungsverfahren eingesetzt, um die mit der Strahlenbelastung verbundenen Risiken möglichst zu reduzieren. Man schätzt, dass das Projekt ein voller Erfolg ist, wenn es mehr als ein Jahr Betriebszeit erreicht.

Mechanischer Aufbau

Die mechanische Basisstruktur wurde aus dem Vollen gefräst, um ein nur ein einziges Teil zu erhalten (**Bild 4**). Die verschiedenen Bauteile des Satelliten wie Solarzellen, Platinen, Schrauben, Steckverbinder, Verdrahtung und so weiter wurden mit dem Voranschreiten des Projekts entworfen und dimensioniert. Eine ständige Zusammenarbeit der einzelnen Teams an den verschiedenen Instituten war

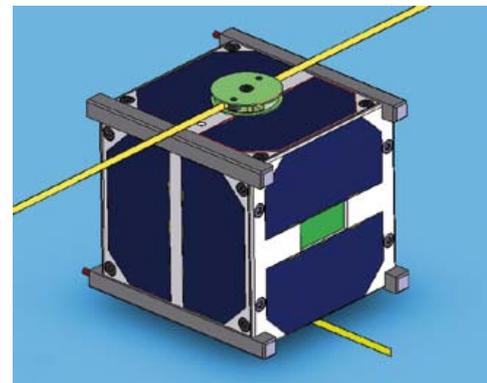


Bild 1. CAD-Modell des Robusta-Satelliten. (Quelle: RobustaCom)

notwendig, um die Bauteile an den Entwicklungsfortschritt anzupassen und auf den neuesten Stand zu bringen. Die Struktur wurde aus Aluminium 6061 mit konstanter Dichte unter Weltraumbedingungen gefertigt.

Leistungselektronik

Das Power-Board sorgt für das Laden der Batterie und die Verteilung der benötigten Spannungen auf die verschiedenen Subsys-

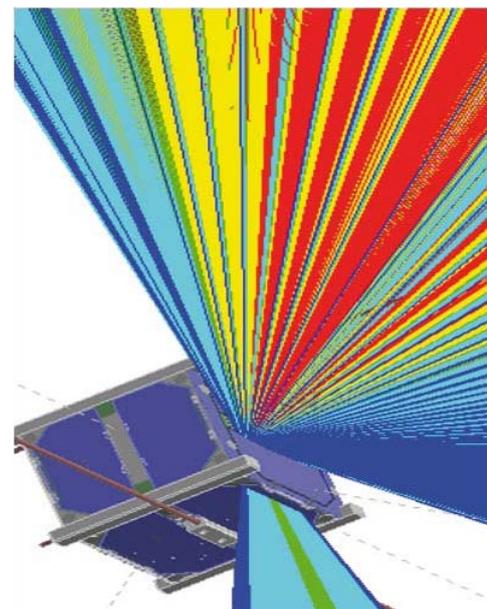


Bild 2. Beispieldarstellung einer Berechnung der von einem Robusta-Baustein aufgenommenen Dosis durch das Programm FASTRAD. (Quelle: RobustaCom)

LESERSCHALTUNG

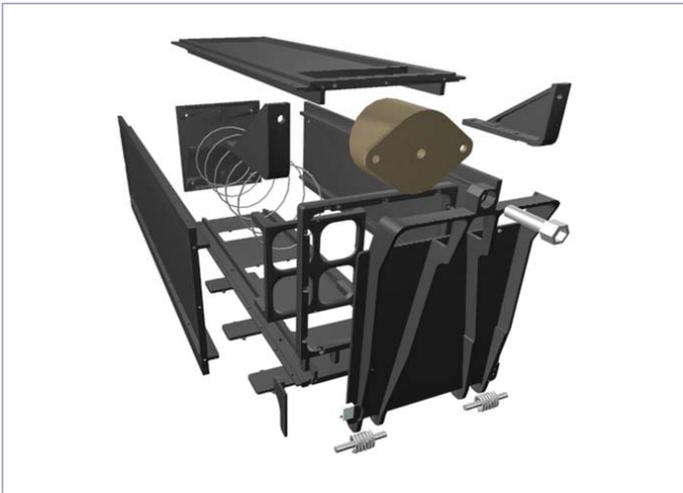


Bild 3. 3D-Explosionsmodell eines p-pod. (Quelle: CubeSat Project)

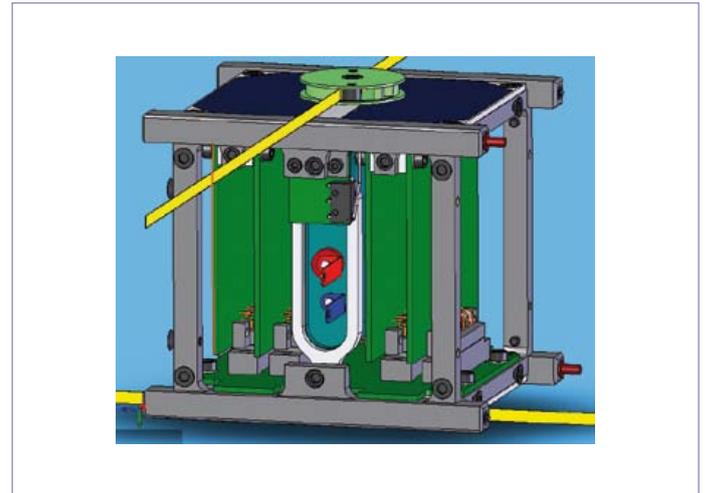


Bild 4. 3D-Darstellung des Satelliten. (Quelle: RobustaCom)

teme. Hier findet sich auch das Auslösesystem für das Entfalten der Antennen im Orbit. Die Dimensionierung der Bauteile bietet eine beträchtliche Reserve über die erforderliche Leistung hinaus. Das Ladesystem berücksichtigt Änderungen der Solarzellenspannung, die durch Temperaturschwankungen und Alterung verursacht werden. Es werden drei Spannungen bereitgestellt: 8 V für die Sendeverstärker, 6 V für die Logikbausteine und -5 V für die Testobjekte. Die mit Solarzellen bestückten sechs Seiten des Cubesat-Würfels werden abhängig von der Eigenbewegung des Satelliten in zufälliger Weise von der Sonne angestrahlt.

Durch individuelle Messung von Strom und Spannung der sechs Solarmodule lässt sich sowohl das Laden des Akkus überwachen als auch die Eigenbewegung des Satelliten messen. Für die Kommunikation zwischen Leistungselektronik und Controllerboard wurde von den Studenten der I²C-Bus gewählt.

Versuchsplatine

Die Schaltung der Versuchsplatine musste sorgfältig geplant und getestet werden. Jeder zu testende Baustein (LM124 und LM139) enthält acht Elemente, für die Spannung, Strom, Temperatur und

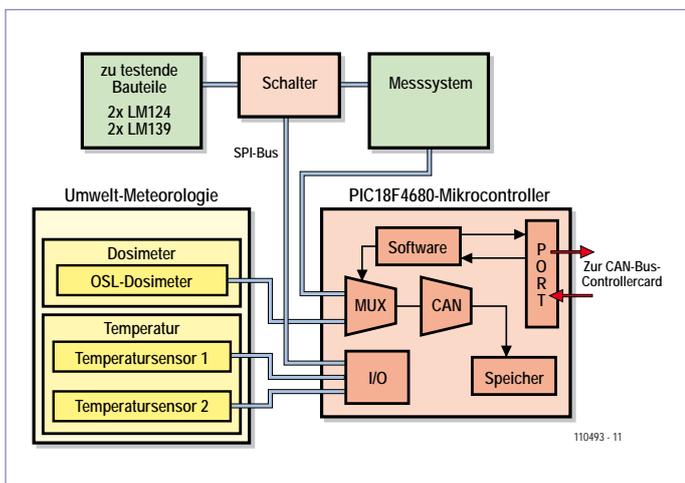


Bild 5. Blockschaltbild der Versuchsplatine. (Quelle: RobustaCom)

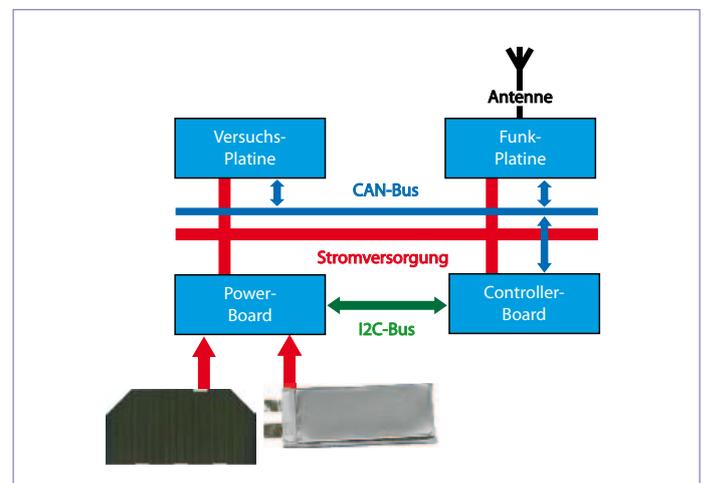


Bild 6. Verbindung der vier Platinen mittels I²C-Interface und CAN-Bus.

Strahlung zu erfassen sind. Es musste also eine Architektur auf der Basis von mikrocontrollergesteuerten Analogschaltern gefunden werden, um die Messungen an den verschiedenen Anschlüssen der Bauteile zu multiplexen (Bild 5). Die Studenten mussten einen Bus wählen, der die große Zahl der Schalter-Adressen verwalten kann und haben so I²C und SPI gut kennengelernt.

Mit Ausnahme des Power-Boards verfügen alle Boards über einen PIC18F4680, ein CAN-Interface und ein *Anti-latchup*-System (zum Schutz des Mikrocontrollers vor Kurzschlüssen durch ionisierende Teilchen).

Controller-Board

Dies ist das Hirn des Satelliten, das die Aufgaben der anderen Platinen koordiniert. Das Controller-Board steuert den Dialog mit den anderen Karten und ist verantwortlich für das Energiemanagement. Sie unterbindet zum Beispiel eine Verbindung mit der Bodenstation, solange ein Versuch abläuft, da beides gleichzeitig zu viel Leistung erfordert.

Erst im Verlauf der Inter-Subsystem-Treffen wurde beschlossen, einen CAN-Bus für die Kommunikation zwischen den einzelnen Sub-

100 % Studenten

Der Robusta-Satellit und die zugehörige Bodenstation wurden zur Gänze von Studenten entworfen und realisiert, die von verschiedenen Einrichtungen der Universität Montpellier 2 stammen: IUT Nîmes, Ingenieurakademie Polytech'Montpellier, Aufbau- und Masterstudiengänge EEA (Elektronik, Elektrotechnik, Automatisierung) der Faculté des Sciences. Dieses Projekt in Zusammenarbeit mit dem CNES wird durch weitere große Unternehmen der Elektronikindustrie unterstützt.

dien hinzufügen. Der Satellit soll von Studenten erdacht, gebaut und getestet werden, angeleitet von den im jeweiligen Fachgebiet erfahrenen Dozenten und stets unter Kontrolle des CNES. Die Studenten sind auch für das Projektmanagement zuständig.

Dieses Projekt ist ein Experiment mit industrieller Ausrichtung, das in Bezug auf Dauer, Kosten und den technologischen Level für Studenten zugänglich bleibt. Robusta bietet als Gesamtsystem Absolventen von Kurz- bis Doktoratsstudiengängen die Möglichkeit, fortschrittli-



Bei diesem Projekt kann man verschiedene große Linien erkennen: Systemstudien und das verbundene Projektmanagement, der mechanische Aufbau, begleitende Tests und die Subsysteme aus den verschiedenen Gebieten der EEA: Energiemanagement, Nutzlast, Controllerboard, Funkkommunikation, Bodenstation. Man muss noch etwas Kommunikation mit den und Verbreitung durch die Me-

che Prototypen zu entwickeln, ihren Kommunikationssinn zu schärfen und darüber hinaus die ganze Welt der Raumfahrt zu entdecken. Sie müssen sich vom Missionsentwurf bis hin zur Auswertung der Messdaten einbringen, über alle Phasen der Konzeption, der Bauteilbeschaffung, des Prototypenbaus und der Tests hinweg.

systemen einzusetzen (**Bild 6**). Andererseits konnten die Studenten beim Entwurf des Prototypen ihr eigenes Datenaustauschprotokoll entwickeln. Da die Zahl der zu versendenden Nachrichten hoch ist, wurde ein Petri-Netz verwendet, um Blockaden und den Verlust von Nachrichten zu vermeiden und um alle Auflagen zu berücksichtigen.

Funk-Board

Für das Subsystem zur Funkkommunikation haben die Studenten der Auswahl der für die Applikation verwendeten Frequenzen große Aufmerksamkeit gewidmet. Nachdem mehrere Sende-Empfangs-Modelle untersucht wurden, haben sie sich für ein System mit zwei getrennten Frequenzen entschieden: 435,325 MHz für das Senden zur Bodenstation (Downlink) und 145,95 MHz für den Empfang der Steuersignale (Uplink). Die Wahl der Komponenten und insbesondere der Verstärker geschah in direkter Abstimmung mit den für das Subsystem Bodenstation zuständigen Studenten, um das Link-Budget einzuhalten. Neben der Beschäftigung mit den spezifischen Problemen der Funkübertragung galt es auch, sich mit der Signal-

verarbeitung auseinanderzusetzen und geeignete Modulations- und Demodulationsverfahren auszuwählen. Auch die Simulation wurde nicht vergessen, besonders für die Antennen des Satelliten, die mittels einer professionellen Hochfrequenz-Software (CST Microwave Studio) vollständig simuliert wurden.

Bodenstation

Die Bodenstation ist ein integraler und unverzichtbarer Teil für den Erfolg einer Raummission. Sobald sich der Satellit im Orbit befindet, ist sie das einzig mögliche Kommunikationsmittel. Sie erlaubt es, alle Daten der Experimente und die Flugparameter (Telemetrie) zu empfangen und die Funkbefehle zur Änderung des Versuchsablaufs oder des Satellitenverhaltens zu senden (zum Beispiel Energiemanagement durch Aktivieren/Deaktivieren bestimmter Bauteile). Die Robusta-Bodenstation (**Bild 7**) verwendet Hardware aus dem Amateurfunk-Bereich. Das zentrale Element ist ein Transceiver, der die Modulation/Demodulation von AFSK-Signalen im 430-MHz-Band für die Telemetrie und im 144-MHz-Band für die Fernsteu-

Expresso

Das zum CNES gehörende Raumfahrtzentrum Toulouse (Centre Spatial de Toulouse, CST) ermöglicht es Studenten, konkrete Erfahrungen im Bereich orbitaler Systeme zu erwerben.

Hier bietet sich auch die Gelegenheit, neue Technologien zu testen und wissenschaftliche Versuche für die Raumfahrtgemeinschaft

mit geringen Kosten zu realisieren. Um das Projekt zu unterstützen, bietet das CNES finanzielle Mittel an und stellt einen Projekt-Koordinator sowie Experten des CST für thermische Analysen, Solarzellen, Vibrationstest, Nutzungsrechte für Frequenzen und so weiter zur Verfügung.



Bild 7. Die Robusta-Bodenstation.
(Quelle: RobustaCom)



Bild 8. Die Trägerrakete Vega.
(Quelle: ESA)

erbefehle ermöglicht. Zwei Motorantennen stellen die Verbindung zum Satelliten her. Ein PC steuert neben den Antennenmotoren den Versand der Funkbefehle und den Empfang der Telemetriedaten. Alle Programme für die Bodenstation sind entweder selbst entwickelt oder Open Source, nicht zuletzt das Betriebssystem Ubuntu für den PC, der die Bodenstation steuert. Dies erlaubt die Anpassung der Software an spezielle Bedürfnisse in kurzer Zeit und ohne besondere Einschränkungen.

Trägerrakete Vega

Nach einem Bewerbungsauftrag wurde der Robusta-Satellit zusammen mit acht anderen ausgewählt, um beim Testflug der Vega (**Bild 8**) ins All transportiert zu werden. Das Vega-Trägersystem wurde entwickelt, um kleinere Satelliten von 300 bis 2000 kg in polare und niedrige Erdumlaufbahnen zu bringen. Der Robusta-Satellit wird beim Erstflug dieser Trägerrakete dabei sein, der Anfang 2012

von der Basis Kourou abheben wird.

Insgesamt werden neun CubeSats zusammen mit der Hauptnutzlast, dem Wissenschaftssatellit LARES und einem edukativen Minisatelliten namens ALMASat abgesetzt.

(110493)



Vibrationstest des CubSats. (Quelle: CNES)

Weblinks und Literatur

- [1] Das Robusta-Projekt: www.ies.univ-montp2.fr/robusta/
- [2] CubeSat der California Polytechnic State University: <http://polysat.calpoly.edu/>
- [3] J. Boch, "Estimation of Low Dose Rate Degradation on Bipolar Linear Integrated Circuits Using Switching Experiments", IEEE Trans. Nuclear Science, vol. 52, p.2626-2621, decembre. 2005.
- [4] J. R. Vaile, F. Ravotti, P. Garcia, M. Glaser, S. Matias, K. Idri, J. Boch, E. Lorgevre, P. J. McNulty, F. Saigne, L. Dusseau, „Online dosimetry based on optically stimulated luminescence materials“ IEEE Trans. on Nuclear Science, vol. 52, Issue 6, Decembre 2005 pp. 2578 – 2582.

Das Weltraumabenteuer geht weiter mit Ristretto und Friends

Robusta, Espresso, Friends und Ristretto haben nichts mit Kaffee zu tun, sondern stehen hier für die Namen der studentischen Projekte zur Eroberung des Weltraums. Von kleinen zu immer größeren Satelliten: das bietet die Universität Montpellier 2 in Zusammenarbeit mit dem CNES, der ESA und der Universität N.E. Bauman in Moskau.

Um sich diesen neuen Herausforderungen zu stellen, wird die Univer-

sität Montpellier 2 demnächst ein universitäres Raumfahrtzentrum namens SOLARIUM einrichten (Systèmes Orbitaux Liés aux Activités de Recherche Interdisciplinaires de l'Université Montpellier 2). Dieses wird durch die Van-Allen-Stiftung unterstützt, die erste in Frankreich, die sich für kleine Satelliten einsetzt.

Kontakt: Tel. 0511/61 65 95-0 – Fax 0511/61 65 95-55 – E-Mail: service@id-medienservice.de

 <p>Entwicklung industrietauglicher Software und Hardware sowie Elektronik 03303/212166 oder www.jasys.de</p>	<p>Bausätze zu ELEKTOR 1986 bis heute! Teilesätze, Platinen, programmierte Controller sowie Cds zu fast allen Elektor-Projekten vom Spezialist. Alle Elektor-Artikel zum Verlagspreis. Ihr zuverlässiger Partner für aktive und passive elektronische Bauteile und Komponenten:</p>	<p>Auch Ihr Unternehmen ist eine Anzeige wert!</p>	<p>www.anttronic.de ab 1 Stck. ANTTRONIC Leiterplatten zu TOP-Preisen!!</p>
<p>Alles Spule! Wir liefern und fertigen: Drähte, HF-Litzen, Ferrit- und Eisenpulverkerne, Spulenkörper, Isoliermaterial, Klebebänder, Tränklacke, Übertrager, RFID-Spulen, Sensor- und Aktorspulen, Prototypen, Kleinserien, Serien, Ersatzteile und vieles mehr. MM Menting Mikroelektrik Spulen für Elektronik www.spulen.com</p>	<p>Geist Electronic-Versand GmbH Tel.: 07720/36673 Fax: 07720/36905 Mail: info@geist-electronic.de Shop: www.geist-electronic.de</p>	<p>VTS – ELEKTRONIK GbR Preiswerter Leiterplatten Service info@vts-elektronik.de www.vts-elektronik.de</p>	<p>Ein Projekt für Leute, die den 8bit-Heimcomputern nachtrauern www.bomerezprojekt.de</p>
<p>• Konfigurierbare digitale & analoge Schaltaktoren für die Hausautomation • Seriell ansteuerbar über Modbus von PC, SPS oder µController • 4,3" TFT Touch-Display mit I²C-Extender Elconeq TECHNOLOGIES www.elconeq.de Tel. 02832-9784 301</p>	<p>Laehn-Versand.de schlanke Preise - fettes Angebot Schnellversand ohne Mindestumsatz Bauteile - Ersatzteile - Zubehör Fernbedienungen - Zeilentrafos HDTV DVB-T Sat Audio/Video Überwachungstechnik und vieles mehr. www.Laehn-Versand.de</p>	<p>LOETRONIC Embedded MP3 Module www.loetronic.com</p>	<p>Baugruppenbestückung vom Prototypen bis zur Serie FS-ELECTRONIC.de</p>
<p>Ausgabe: Elektor Februar 2011</p> <p>Anzeigenschluss: 19. 12. 2011</p> <p>Erscheinungstermin: 18. 01. 2012</p>	<p>HEXWAX LTD www.hexwax.com Treiberunabhängige USB-ICs von einem der Weltmarktführer • USB-UART/SPI/I²C-Konverter • TEAleaf-USB Authentifizierungs-Dongles • expandIO-USB I/O-USB-Expander • USB-FileSys Flash-Drive mit SPI-Interface • USB-DAQ Flash-basierter Datenlogger</p>	<p>LCD DESIGN LCDs und mehr www.lcd-store.de www.LC-Design.de www.crystalfontz.de</p>	
	<p>SCOPES und mehr HAMEG® Instruments A Rohde & Schwarz Company MESSTECHNIK zum fairen Preis</p>	<p>Die Buchung einer Anzeige beinhaltet einen kostenlosen Eintrag auf der Website von Elektor, inklusive eines Links zu Ihrer Seite.</p> <p>Reservieren Sie jetzt Ihre Jahresbuchung!</p>	
<p>NienTech SCHNITTSTELLENWANDLER von WLAN LAN USB nach RS485 RS422 RS232 TTY über virtuellen COM-Port ansprechbar www.NienTech.de</p>			

www.elektor.de

RCA Cosmac Development System

hELLO wORLD aus der Steinzeit



Von Jan Buiting (Elektor UK & US)

Mein erster Kontakt mit Mikroprozessoren datiert auf Anfang 1980, als alle möglichen „Heim“-Computer auf der Basis Z80, 6502 und 8085 um die Gunst der Käufer buhlten. Ich habe Elektronik, die schon vom Nichtstun heiß wird, noch nie leiden können. Von daher ignorierte ich diese stromdurstigen NMOS-CPU's samt Fan-Clubs und zugehörigen Mailboxen und wandte mich mit dem CDP1802 einem weniger populären Zahlenfresser zu. Dieser Chip stammt - wie die wohlbekannte Logik-Serie CD4000 - von RCA. Die stromsparende CMOS-CPU 1802 und ihre „Cosmac“-Familie peripherer IC's wurde in den USA recht bekannt, nachdem 1976 in der Zeitschrift „Popular Electronics“ ein nettes Selbstbau-System unter der Bezeichnung „Cosmac ELF“ veröffentlicht wurde. Das war etwa die Zeit, als Jobs & Woz in ihrer Garage in Palo Alto mit dem 6502 herumbosselten. Die verbesserte europäische Variante „Cosmos“ erschien vier Jahre später und wurde dann „mein System“ (siehe „CDP1802 – Computer im Weltraum“ in Elektor Oktober 2006). 1980 studierte ich noch. Wer als Student ein „RCA Cosmac Data Book“ in die Hände bekam, war König. Mir gelang es, von Vekano (ehemaliger niederländischer RCA-Distributor) ein Exemplar „dauerhaft auszuleihen“. Vermutlich um das Buch dicker und die Marketingabteilung von RCA glücklich zu machen, drehten sich die letzten 40 Seiten um Software-Tools von RCA, Hardware, Programmer, Compiler (BASIC und PL/M) sowie Cosmac-Entwicklungs-Systeme. Mein selbst gebautes Cosmos-System war ganz ordentlich (48 KByte RAM etc.). Es war mit einem Strombedarf von unter 200 mA an 5 V ziemlich „grün“ und mit 3,58 MHz auch schnell. Ich träumte damals anhand der Daten und der wenigen Fotos von RCAs Spitzenprodukt: das monumentale Cosmac Development System IV CDP18S008. Kein Preis war zu finden – man hätte ihn telefonisch erfragen müssen.

Je nach Hobby und Interessen kann es leicht Jahrzehnte dauern, bis man seine Jugendträume verwirklichen kann.

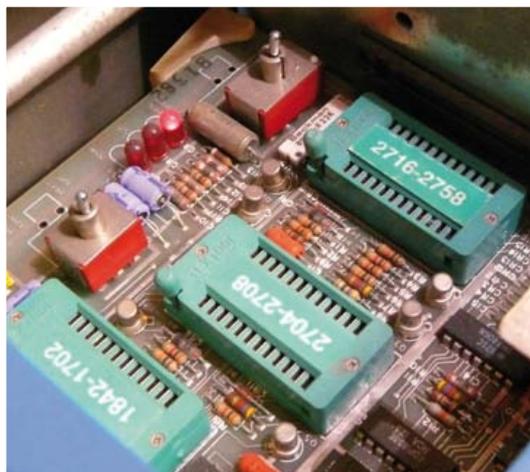
Ich habe mein Cosmos-CDP1802-System samt aller Karten und Peripherie immer in Ehren gehalten. Als ich aber vor einigen Jah-

ren durch die ELF-Seiten bei Yahoo surfte, traf mich fast der Schlag: In einem Posting bot ein Landsmann der überwiegend amerikanischen Leserschaft ein echtes Cosmac-IV-System an. Das Angebot wurde aufgrund der hohen Transportkosten in die USA (von wo es 1981 kam) dankend abgelehnt. Um die Sache kurz zu machen: Nach Zahlung eines eher symbolischen Preises konnte ich das komplette System in mein Auto laden. Auch vieles von der Hardware, die damit einmal entwickelt wurde, war beigelegt. Außerdem noch die komplette Dokumentation und die Software auf 8“-Disketten. Das war das erste Mal, dass ich die Scheinwerfer-Höhenverstellung meines Autos justieren musste, um den Gegenverkehr nicht zu blenden. Mir war bis dato nicht bewusst, dass der Gegenstand meiner früheren Träume so groß und schwer war. Doch jetzt war er meins!

Die hellblau- und cremefarbige Cosmac-IV-Einheit besteht aus einem dummen Bildschirm-Terminal, das mit einem eingebauten CDP1802-System kommuniziert. Es wiegt „nur“ 17 kg. Erstaunlicherweise bestand das Terminal aus einem CDP1802-Video-System, das zwischen Tastatur und 12“-Bildröhre steckte. Das tatsächliche Entwicklungssystem bestand aus einer Backplane, auf der CDP18Sxx-„Mikromodule“ wie die CPU (CDP1802), ROM, RAM, I/O, FDISK etc. steckten. Die mit dem System zu entwickelnde Software wird bis zur Reife auf den Karten ausgebrütet (schreiben, testen und debuggen), um dann im Hex-Format als Firmware in (E)PROMs wie 2708 oder 2716 gebrannt zu werden. Der (E)PROM-Programmer kommt unter einer Klappe zum Vorschein. Die Abdeckung öffnet und schließt sich durch Drücken. Das fanden viele lustig. Auf der Rückseite des Terminals gibt es Anschlüsse für Netzspannung, Disk, Drucker, „CRT EIA“, „SYS“, „EIA“, „MOPS EIA“, Reserve 1 und Reserve 2.

Das Terminal-Bildschirm ist mit seiner nostalgischen grünen Bildröhre durchaus gut zu lesen, auch wenn er angesichts der „Auflösung“ von 24 Zeilen zu je 80 Zeichen im Vergleich zu modernen LCDs eher winzig erscheint. Das Keyboard mit 73-Hall-Tasten von Sperry klingt auch bei hohem Schreibtempo angenehm. Mit dem „Full-Screen“-Editor (FSE) wird man beim Schreiben von Code (in Assembler natürlich) kaum den Luxus von Maus, Menüs, Icons oder

IV (CDP18S008) von 1978



die Ablenkung durch das Internet vermissen.

Auch wenn man, fasziniert vom blauweißen Cosmac-IV-Terminal, kaum abwarten kann, damit Code zu schreiben und diesen in ROMs zu brennen, sollte man kurz innehalten. Es gibt noch zwei weitere Komponenten:

Zunächst wäre da das Doppel-Diskettenlaufwerk für 8"-Floppy-Disks mit sagenhaften 256 KB (= 0,000256 GB). Der Hersteller ist Pertec (Chatsworth & Irvine, CA) und die Abdeckung wurde in „RCA blue“ lackiert, laut Beipackzettel. Diesem war außerdem zu entnehmen, dass das Modell 3712 gut 34 kg wiegt. Das gleiche Laufwerk gab es unter anderem in abweichender Lackierung für Altair-Systeme (CPU = 8080). Die Floppy-Einheit produziert ordentlich Lärm, da der rückseitige Ventilator und die Laufwerksmotoren permanent laufen (vermutlich zwecks kurzer Zugriffszeit). Bei jeder Selektion oder Deselektion einer Disk gab es ein lautes Klacken. Wenn die Schreib/Leseköpfe über die Disketten rasen, kommt noch ein extra Ton dazu. Die Elektronik besteht aus einem Logik-Board vom Ausmaß zweier iMacs, das mit rund 150 ICs bestückt ist (überwiegend 5-V-TTL-ICs). Das Laufwerk ist zwar laut, heiß und langsam – funktioniert aber nach 30 Jahren immer noch einwandfrei. Sogar alle vom Vorbesitzer beigefügten Disketten konnten noch problemlos gelesen werden. Ich überlege mir, ob ich nicht einen Konvertierungsdienst 8"-Diskette nach USB-Stick anbieten soll. ;-)

Eine dritte Komponente nennt sich „Micromonitor“ (CDP18S030), wobei „micro“ für Mikrocontroller und nicht für die Größe steht. Das Alu-Gehäuse ist wieder in RCA-blauer Farbe gehalten und enthält nicht viel mehr als LEDs, Schalter und ZIF-Sockel. Die Idee ist, damit die CPU der späteren Anwendung mit Hilfe eines langen 40-poligen Flachbandkabels anzuschließen und dann im echten Einzelschrittbetrieb zu schauen, was sich auf den CPU-Pins so tut. Damit konnte man also CDP1802-Anwendungen debuggen, und zwar realistisch in-System, in Echtzeit und auf dem Zielsystem! Das „wie“ lässt sich hier nicht gut demonstrieren, denn dazu müsste

man Maschinen-Code analysieren und editieren können. Eine spätere Variante (Version CDP18S030A) verfügte über eine abtrennbare Display/Keyboard-Einheit, die einem Taschenrechner aus den 1970er Jahren ähnelte.

Die einzigen in Serie auf Basis des CDP1802 produzierten Geräte, die noch existieren, sind amerikanische Verkehrsampeln, die gelegentlich bei eBay auftauchen. Außerdem gibt es noch ein UHF-Autotelefon von Nokia aus dem Jahre 1996 (!), das gerne von Funkamateuren modifiziert wurde. Benähe hätte ich es vergessen: Da war noch ein Heim-Computer mit der Bezeichnung COMX35.

Auch heute gibt es eine kleine verschworene Gemeinschaft, die gerne mit der Cosmac-CPU CDP1802 arbeitet. Ihre Aktivitäten findet man leicht im Internet. Auch ich nutze mein Cosmac-IV-System gelegentlich, um damit die Software zur Klimatisierung und Bewässerung meines selbstgebautes Gewächshauses zu warten, das über eine auf einem CDP1802 basierende Steuerung verfügt. Ich habe meine Freude an PL/M, CDOS, UT5, MOPS, BASIC1 und ASM8. Das riesige Diskettenlaufwerk und der „Micromonitor“ sind allerdings in Rente, da ich das Laufwerk mit zwei statischen RAMs emuliere, die von Batterie versorgt werden. Ich bin also Zeitzeuge von Fortschritt und Miniaturisierung bei Mikrocontrollern während der letzten 30 Jahre. Heute würde die Intelligenz eines 70 kg schweren CDP18S008-Systems locker in ein einziges Spartan-FPGA von Xilinx passen.

Im Internet habe ich einen Preis von 1976 gefunden: Irgendwo wurden damals 70.000 US-\$ für ein voll ausgebautes CDP18S008-System (wie das hier beschriebene) verlangt. Ich kann das leider nicht überprüfen.

Wenn ein Leser der Retronik-Rubrik noch original Cosmac-CDP18Sxx-Disketten von RCA mit Software-Tools oder höheren Programmiersprachen haben sollte: Bitte lassen Sie es mich wissen. Das gilt auch für das CDP18S021-Microterminal mit UT5.

(110528)

Retronik ist eine monatliche Rubrik, die antiker Elektronik und legendärer ELEKTOR-Schaltungen ihre Referenz erweist. Beiträge, Vorschläge und Anfragen schicken Sie bitte an: editor@elektor.com

Hexadoku

Sudoku für Elektroniker

Sie mögen Rätsel? Eine große Anzahl Elektor-Leser gehört in jedem Fall zu den Fans, denn wir bekommen jeden Monat eine Menge Lösungen eingeschickt. Machen Sie doch auch mit – und entdecken Sie, wie faszinierend unser ganz besonderes Sudoku sein kann. Wer uns die (richtigen) Ziffern in den grauen Kästchen zuschickt, kann einen von vier schönen Gutscheinen gewinnen!

Die Regeln dieses Rätsels sind ganz einfach zu verstehen: Bei einem Hexadoku werden die Hexadezimalzahlen 0 bis F verwendet, was für Elektroniker und Programmierer ja durchaus passend ist. Füllen Sie das Diagramm mit seinen 16 x 16 Kästchen so aus, dass **alle** Hexadezimalzahlen von 0 bis F (also 0 bis 9 und A bis F) in jeder Reihe, jeder Spalte und in jedem Fach mit 4 x 4 Kästchen (markiert

durch die dickeren schwarzen Linien) **genau einmal** vorkommen. Einige Zahlen sind bereits eingetragen, was die Ausgangssituation des Rätsels bestimmt. Wer das Rätsel löst - sprich die Zahlen in den grauen Kästchen herausfindet - kann wie jeden Monat einen Hauptpreis oder einen von drei Trostpreisen gewinnen!

Mitmachen und gewinnen!

Unter allen internationalen Einsendern mit der richtigen Lösung verlosen wir einen **ELEKTOR-Gutschein** im Wert von 100 € und drei **ELEKTOR-Gutscheine** im Wert von je 50 €.

Einsenden

Schicken Sie die Lösung (die Zahlen in den grauen Kästchen) per E-Mail, Fax oder Post an:
 Elektor – Redaktion Süsterfeldstr. 25 52072 Aachen
 Fax: 0241 / 88 909-77 E-Mail: hexadoku@elektor.de
Als Betreff bitte nur die Ziffern der Lösung angeben!
 Einsendeschluss ist der 31. Dezember 2011!

Die Gewinner des Hexadokus aus dem Oktoberheft stehen fest!

Die richtige Lösung ist: D0837.
 Der Elektor-Gutschein über 100 € geht an: Olavi Parkka aus Finnland.
 Einen Elektor-Gutschein über je 50 € haben gewonnen: Robert Amandine, Susanne Müller-Furrer und Thierry Notot.
 Herzlichen Glückwunsch!

1	D			7		C	A			9	5				
				5		B	1	8		D		0			4
			9			F				C				6	
B				E	A	D	4	F	2						7
E	8				B		F								0 1
				1				9	6	4		5			C
1			2	3				7	B						D
3		C	6		4			D	8					7	B
9		1				8	6			5		2	4		F
	0					1	7					6	B		A
4			5		E	0	3					B			
7	A							4		1					9 3
	9					4	8	5	E	7	C				0
		8				7				0			F		
2			4		1		0	6	3		D				
	C		7				E	B		9				D	4

2	A	0	7	B	C	F	3	9	D	6	4	8	E	1	5
B	8	3	9	A	5	4	D	7	0	E	1	6	2	F	C
5	1	C	D	E	9	0	6	8	F	2	3	7	4	A	B
4	6	E	F	8	7	1	2	A	5	B	C	D	0	9	3
C	2	1	8	6	D	3	B	0	7	4	E	9	A	5	F
E	D	F	A	7	8	2	5	B	6	3	9	1	C	0	4
0	7	5	3	4	A	9	F	C	8	1	2	B	6	D	E
9	4	6	B	1	E	C	0	D	A	5	F	2	3	7	8
1	5	8	C	9	4	6	A	E	B	F	0	3	7	2	D
6	B	2	0	C	F	D	8	1	3	7	A	E	5	4	9
D	3	7	E	2	0	5	1	4	9	C	8	A	F	B	6
A	F	9	4	3	B	7	E	5	2	D	6	C	1	8	0
F	9	B	6	5	1	E	C	2	4	A	D	0	8	3	7
3	C	A	5	D	2	8	7	F	E	0	B	4	9	6	1
7	E	4	2	0	6	B	9	3	1	8	5	F	D	C	A
8	0	D	1	F	3	A	4	6	C	9	7	5	B	E	2

Der Rechtsweg ist ausgeschlossen. Mitarbeiter der in der Unternehmensgruppe Elektor International Media B.V. zusammengeschlossenen Verlage und deren Angehörige sind von der Teilnahme ausgeschlossen.

Starke Stücke

Die ganze Welt der Elektronik
in einem Shop!



Bücher

Kreative Lösungen aus allen Bereichen der Elektronik

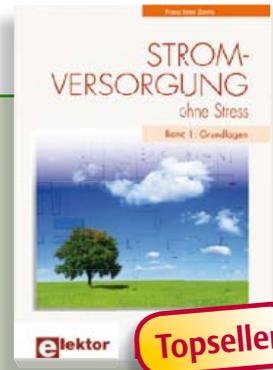
311 Schaltungen

Das mittlerweile zwölfte Buch aus Elektors erfolgreicher „Dreihunderter“-Schaltungsreihe bietet ein weiteres Mal neue Konzepte sowie einen (fast) unerschöpflichen Fundus zu allen Bereichen der Elektronik: Audio & Video, Spiel & Hobby, Haus & Hof, Prozessor & Controller, Messen & Testen, PC & Peripherie, Stromversorgung & Ladetechnik sowie zu Themen, die sich nicht katalogisieren lassen.

„311 Schaltungen“ enthält viele komplette Problemlösungen, zumindest aber die Idee hierzu. Nicht zuletzt sind die 311 Schaltungen der Anstoß zu ganz neuen Überlegungen.

Dieses Buch ist ideal für alle, die kreativ mit der Elektronik umgehen; sei es im Beruf oder als Hobby.

544 Seiten (kart.) • ISBN 978-3-89576-255-0 • € 36,80 • CHF 45,70



Band 1: Grundlagen

Stromversorgung ohne Stress

Die Funktion aller elektronischen Schaltungen und Geräte steht und fällt mit der Stromversorgung. Schon deshalb muss man dieser Baugruppe besondere Aufmerksamkeit widmen. Dieses Buch beinhaltet Grundlagen und Schaltungen der Stromversorgungstechnik für elektronische Geräte aus der Praxis. Dem aktuellen Trend folgend hat der Autor der mobilen Stromversorgungstechnik und der Schaltnetzteiltechnik besondere Aufmerksamkeit gewidmet.

294 Seiten (kart.) • ISBN 978-3-89576-248-2
€ 38,00 • CHF 47,20



Der 3. Band der neuen Buchreihe

Embedded Electronics 3

Die Bände Reihe wenden sich an jene, die sich – als Auszubildende, Studierende oder Berufseinsteiger – von A bis Z in die professionelle Schaltungs- und Systementwicklung einarbeiten wollen. Sie bieten, was Praktiker und Lernende brauchen: Eine Auffrischung und Vertiefung der Grundlagen, Anregungen zu eigenen Gedanken und Zugänge zu Einzelheiten, Querverbindungen und Spitzfindigkeiten.

412 Seiten (kart.) • ISBN 978-3-89576-185-0
€ 49,00 • CHF 60,80



„Die Spannung steigt!“
**Stromversorgungen
in der Praxis**

Die Elektronik bestimmt unser tägliches Leben mehr denn je – Tendenz steigend. Dabei benötigen alle elektronischen Geräte und Systeme eine gut funktionierende Stromversorgung mit spezieller Anpassung an die Betriebsbedingungen. Dieses neue Buch beschreibt die entsprechenden Möglichkeiten vom Transformator bis zum passenden Kühlkörper. Behandelt werden die wichtigsten Merkmale, Einsatzmöglichkeiten und das Betriebsverhalten von vielen unterschiedlichen Stromversorgungsgeräten.
366 Seiten (kart.) • ISBN 978-3-89576-239-0
€ 46,00 • CHF 57,10



Von Top-Autor und Entwickler Kainka
Basiskurs BASCOM-AVR

BASCOM und AVR-Controller sind ein starkes Team! Was immer man entwickeln möchte, meist hat ein ATmega schon das Wichtigste an Board: Ports, Timer, AD-Wandler, PWM-Ausgänge und serielle Schnittstelle, RAM, Flash-ROM und EEPROM, alles ist reichlich vorhanden. Und BASCOM macht die Anwendung zu einem Kinderspiel. Auch komplexe Peripherie wie LCD, RC5 und I²C lassen sich mit wenigen Befehlen nutzen.

223 Seiten (kart.) • inkl. Software-CD
ISBN 978-3-89576-238-3 • € 39,80 • CHF 49,40



Topseller

LCD-Graphik I, verkettete Strukturen I,
Zeichenketten, Fädeltchnik I
AVR-Programmierung 3

Dieser dritte Band der Buchreihe zur Assembler-Programmierung von AVR-Mikrocontrollern richtet sich nicht nur an Einsteiger. Auch die C-Programmierer von AVR-Prozessoren profitieren von der Erläuterung der Besonderheiten, die es bei der Assembler-Programmierung zu beachten gilt. Nach der Erläuterung der statischen Datenstrukturen in Buch 2 folgt in diesem Buch der Einstieg in die dynamischen Strukturen. Er beginnt nach einer allgemeinen Einführung mit der einfachsten Struktur, der verketteten Liste. Der letzte Teil führt in den Selbstbau von Fädelsprachen ein, die ein äußerst personalisiertes und projektorientiertes Programmieren erlauben.
319 Seiten (kart.) • ISBN 978-3-89576-231-4
€ 46,00 • CHF 57,10

Weitere Informationen
zu unseren Produkten
sowie das gesamte
Verlagssortiment finden Sie
auf der Elektor-Website:

www.elektor.de

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen
Tel. +49 (0)241 88 909-0
Fax +49 (0)241 88 909-77
E-Mail: bestellung@elektor.de



**MIFARE und kontaktlose
Smartcards angewandt**
RFID

MIFARE ist die weltweit meistgenutzte RFID-Technologie. Dieses neue Buch bietet einen praxisorientierten und umfassenden Einstieg in diese Technologie. Die einleitenden Kapitel behandeln u. a. die physikalischen Grundlagen, die relevanten Normen, das RFID-Antennendesign, die Sicherheitsaspekte und die Kryptografie. Das vollständige Hardware- und Software-Design eines Readers ist ausführlich beschrieben.

464 Seiten (kart.) • ISBN 978-3-89576-219-2
€ 56,00 • CHF 69,50



Visual Studio
**C# 2010 Programmierung
und PC-Anbindung**

Ziel dieses Buches ist, auf einfache Weise zu zeigen, wie mit der populären Hochsprache C# ein PC programmiert werden kann. Am Anfang beschreibt das Buch Datentypen und Programmsteuerungen, die dann um fortschrittliche Konzepte wie die objektorientierte Programmierung, Threads, die Internetkommunikation und Datenbanken erweitert werden. Alle verwendeten Code-Beispiele können kostenlos von der Elektor-Webseite heruntergeladen werden.

349 Seiten (kart.) • ISBN 978-3-89576-244-4
€ 44,00 • CHF 54,60

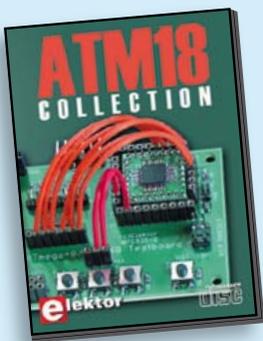


Kompletter Elektor-Jahrgang 2010 auf DVD

Elektor-DVD 2010

Die neue Elektor-Jahrgangs-DVD enthält alle Artikel des Jahrgangs 2010. Sie verfügt über eine sehr übersichtlich gestaltete Benutzeroberfläche. Mit der Elektor-DVD 2010 können Sie: Platinenlayouts in perfekter Qualität drucken; diese Layouts mit einem Zeichenprogramm verändern; die Schnellsuchfunktion benutzen, mit der Sie in den einzelnen Artikeln oder im ganzen Jahrgang nach Wörtern, Bauteilen oder Titeln suchen können; Schaltbilder, Platinenlayouts, Illustrationen, Fotos und Texte exportieren.

ISBN 978-90-5381-267-9 • € 27,50 • CHF 34,10



Mikrocontroller-Programmierung leicht gemacht

ATM18-Collection

Diese CD-ROM enthält Artikel der populären Elektor-Serie „CC2-AVR-Projekt“ mit mehr als 25 Projekten mit dem ATM18-Board inkl. der benötigten Software und Platinenlayouts sowie weiteren Zusatzinformationen. Des Weiteren umfasst die CD auch den kompletten 6-teiligen Elektor-BASCOM-AVR-Kurs.

ISBN 978-0-905705-92-7 • € 29,50 • CHF 36,60



Verbesserter Strahlungsmesser

(Elektor November 2011)

Für die Messung radioaktiver Strahlung braucht man nicht viel mehr als eine PIN-Fotodiode und den passenden Sensorverstärker. Bei diesem Projekt handelt es sich um einen optimierten Vorverstärker mit einem Mikrocontroller-Zähler. Der Controller übernimmt auch gleich die Zeitmessung und zeigt die Impulsrate in „counts per minute“ an. Das Gerät kann mit unterschiedlichen Sensoren für Gamma- und Alphastrahlung verwendet werden. Es eignet sich gut für Langzeitmessungen und für Untersuchungen an schwach strahlenden Proben.

Bausatz mit allen Bauteilen inkl. Platine, Display und progr. Controller

Art.-Nr. 110538-71 • € 39,95 • CHF 49,60



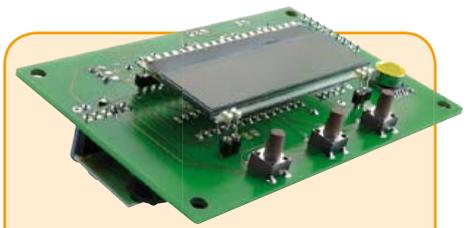
USB-FT232R Breakout-Board

(Elektor September 2011)

Dieser USB-nach-TTL-Wandler ist nicht viel größer als der angelegene Stecker eines USB-Kabels. Seine nützlichen Dienste kann die Mini-Platine unter Windows, Linux und anderen Betriebssystemen entfalten. Mit dem praktischen Konverterboard lassen sich eigene Schaltungen einfach um einen USB-Anschluss erweitern und leicht USB/RS232- oder RS485-Wandler realisieren. Die Platine ist überall dort praktisch, wo TTL erwünscht, aber nur USB vorhanden ist.

Bestückte und getestete Platine

Art.-Nr. 110553-91 • € 15,00 • CHF 18,60



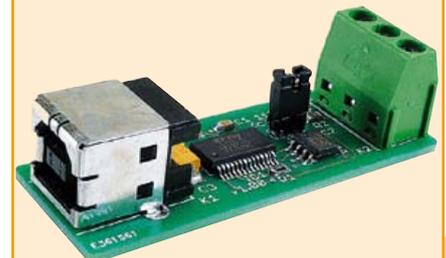
USB-Wetterlogger mit Langzeitspeicher

(Elektor September 2011)

Dieser autonome Datenlogger zeichnet die von I²C-Sensoren gelieferten Daten für Luftdruck, Temperatur und Feuchte auf und zeigt sie auf einem LC-Display an. Die Ergebnisse lassen sich über USB auslesen und mit GNUplot auf einem PC grafisch darstellen. Dank der digitalen Sensormodule ist der Hardwareaufwand gering und ein Abgleich nicht erforderlich. Die Betriebsdauer mit drei Mignonzellen beträgt sechs bis acht Wochen.

Kit bestehend aus Platine, prog. Controller, Feuchte- und Luftdrucksensor

Art.-Nr. 100888-73 • € 34,95 • CHF 43,40



USB/RS485-Konverter

(Elektor Juni 2011)

Der USB/RS485-Konverter stellt die Verbindung zwischen dem USB-Port am PC oder Laptop und RS485-Signalen her. Der Konverter ist speziell für den 2-Draht-RS485-Bus konzipiert, der Anschluss an den Bus erfolgt unkompliziert durch eine dreipolige Klemmleiste. Dank der Verwendung des bekannten FT232L-Schnittstellenwandlers von FTDI stehen virtuelle COM-Port-Treiber nicht nur für alle gängigen Windows-Versionen, sondern auch für Windows CE, Windows Mobile, Linux und Mac OS X zur Verfügung.

Bestückte und getestete Platine

Art.-Nr. 110258-91 • € 24,95 • CHF 31,00

Dezember 2011 (Nr. 492) €
 +++ Das Lieferprogramm zu dieser Ausgabe finden Sie auf www.elektor.de +++

November 2011 (Nr. 491)

Verbesserter Strahlungsmesser

110538-41	Progr. Controller ATmega88PA-PU	10,50
110538-71	Bausatz mit allen Bauteilen inkl. Platine, Display und progr. Controller	39,95

Hier kommt der Bus (9)

110258-1	Platine (Experimental-Knoten)	5,95
110258-1C3 ...	3 x Platine (Experimental-Knoten)	12,95
110258-91	USB/RS485-Konverter (bestückt und getestet)	24,95

Fledermaus-Sonar

110550-1	Platine	9,95
----------------	---------------	------

OnCE/JTAG-Interface

110534-91	Programmer-Platine (bestückt und getestet)	39,95
-----------------	--	-------

Oktober 2011 (Nr. 490)

AVR-Platine Platino

100892-1	Platine	12,95
----------------	---------------	-------

Hier kommt der Bus (8)

110258-1	Platine (Experimental-Knoten)	5,95
110258-1C3 ...	3 x Platine (Experimental-Knoten)	12,95
110258-91	USB/RS485-Konverter (bestückt und getestet)	24,95

Audio-DSP-Kurs

110001-91	DSP-Board (bestückt und getestet)	129,95
110001-92	Kit bestehend aus DSP-Board (110001-91) und Programmer (110534-91 / Veröffentlichung im November-Heft / erhältlich ab KW42)	149,95

September 2011 (Nr. 489)

USB-Wetterlogger mit Langzeitspeicher

100888-1	Platine	17,95
100888-41	Programmierter Controller ATmega88-20PU	9,95
100888-71	HH10D Feuchtesensor	7,95
100888-72	HP03SA Luftdrucksensor	6,45
100888-73	Kit bestehend aus Platine, prog. Controller, Feuchte- und Luftdrucksensor	34,95

PC-Sensoren

100888-71	HH10D Feuchtesensor	7,95
100888-72	HP03SA Luftdrucksensor	6,45

USB-FT232R Breakout-Board

110553-91	Bestückte und getestete Platine	15,00
-----------------	---------------------------------------	-------

J2B: Vielseitiges HMI-Modul mit ARM Cortex-M3

050176-74	Gehäuse Bopla Unimas 160	9,95
110274-71	Kit bestehend aus Platine mit LPC1343-Controller, Quarz, Spannungswandler, LCD- und USB-Interface (bereits bestückt), LED und Stiftleisten	39,95
110274-72	LCD, 4 x 20 Zeichen (HD44780-kompatibel)	9,95

Hier kommt der Bus (7)

110258-1	Platine (Experimental-Knoten)	5,95
110258-1C3 ...	3 x Platine (Experimental-Knoten)	12,95
110258-91	Bestückte und getestete Platine	24,95

Twittern mit E-blocks

EB003	E-blocks Sensor-Interface	26,80
EB005	E-blocks LCD-Board	29,75
EB006	E-blocks PIC-Multiprogrammer	89,25
EB007	E-blocks Switch-Board	17,85
EB059	E-blocks Servo-Board	17,85
EB069	E-blocks WLAN-Board	164,95
TESS14	Flowcode 4 für dsPIC (Professional-Version)	221,65

Juli/August 2011 (Nr. 487/488)

Akkutester

110154-41	Progr. Controller PIC16F873A	13,95
-----------------	------------------------------------	-------

Timer für 2-4-6 Stunden

110219-41	Progr. Controller PIC12F675 DIL8	9,95
-----------------	--	------

Morseuhr

110170-41	Progr. Controller ATtiny4520-PU DIP8	9,95
-----------------	--	------

Bestseller

Bücher	1	↑	311 Schaltungen ISBN 978-3-89576-255-0 € 36,80 CHF 45,70
	2	↻	AVR-Programmierung 3 ISBN 978-3-89576-231-4 € 46,00 CHF 57,10
	3	↻	Embedded Electronics 3 ISBN 978-3-89576-185-0 € 49,00 CHF 60,80
	4	↻	Stromversorgungen in der Praxis ISBN 978-3-89576-239-0 € 46,00 CHF 57,10
	5	↻	C# 2010 Programmierung und PC-Anbindung ISBN 978-3-89576-244-4 € 44,00 CHF 54,60
CD- & DVD-ROMs	1	↻	ECD 6 ISBN 978-90-5381-258-7 € 29,50 CHF 36,60
	2	↻	Wireless-Toolbox ISBN 978-90-5381-268-6 € 32,50 CHF 40,30
	3	↻	ATM18-Collection ISBN 978-0-905705-92-7 € 29,50 CHF 36,60
	4	↻	The Audio Collection 3 ISBN 978-90-5381-263-1 € 21,50 CHF 26,70
	5	↻	Elektor-DVD 2010 ISBN 978-90-5381-267-9 € 27,50 CHF 34,10
Bausätze & Module	1	↑	Verbesserter Strahlungsmesser Art.-Nr. 110538-71 € 39,95 CHF 49,60
	2	↻	USB-FT232R Breakout-Board Art.-Nr. 110553-91 € 15,00 CHF 18,60
	3	↻	USB/RS485-Konverter Art.-Nr. 110258-91 € 24,95 CHF 31,00
	4	↻	USB-Wetterlogger mit Langzeitspeicher Art.-Nr. 100888-73 € 34,95 CHF 43,40
	5	↻	Pico C Art.-Nr. 100823-71 € 82,50 CHF 102,30

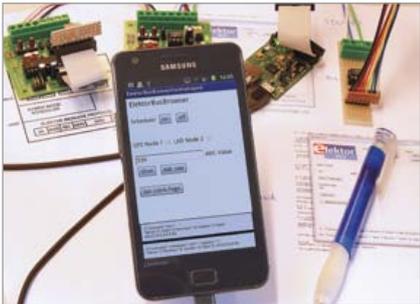
Bestellen Sie jetzt einfach und bequem online unter www.elektor.de/shop oder mit der portofreien Bestellkarte am Heftende!

elektor
 Elektor-Verlag GmbH
 Süsterfeldstr. 25, 52072 Aachen
 Tel. +49 (0)241 88 909-0
 Fax +49 (0)241 88 909-77
 E-Mail: bestellung@elektor.de



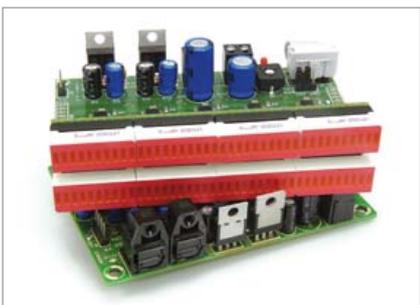
Netzlupe

Die Frequenz der Netzspannung beträgt nominell 50 Hz. An der Abweichung der Netzfrequenz von diesem Sollwert kann man erkennen, wie gut (oder wie schlecht) die Energieverteilung im Netz funktioniert. Bei einem Überangebot auf der Einspeisungsseite steigt die Frequenz, bei einem Unterangebot sinkt sie ab. Normalerweise liegen die Schwankungen im Bereich unter 0,2 Hz. Die Netzlupe misst die Frequenz genau in diesem Bereich von $50 \text{ Hz} \pm 0,4 \%$ und ist damit ein Indikator für die Versorgungsqualität in Ihrem lokalen Netz.



Andropod

Android-Geräte eignen sich sehr gut zur Verwendung in Verbindung mit selbst entwickelten Mikrocontroller-Systemen. Für wenig Geld erhält man ein Display, Rechenleistung, Schnittstellen und Sensoren. Das Problem besteht darin, die Ressourcen des Android-Geräts für die eigene Anwendung nutzbar zu machen. Genau dafür wurde Andropod entwickelt – ein Android-USB-Interface mit Host-Funktionalität.



Digitale Aussteuerungsanzeige

In Teil 7 unseres DSP-Kurses wird das DSP-Board mit einer LED-Aussteuerungsanzeige kombiniert. Das ermöglicht die genaue Pegelanzeige von zwei digitalen Audiosignalen. Die verwendeten LED-Treiber von Texas Instruments steuern 2×40 LEDs. Für noch mehr LEDs lassen sich mehrere Platinen in Reihe schalten.

Elektor Januar 2012 erscheint am 14. Dezember 2011.

Elektor gibt es im Bahnhofsbuchhandel, Elektronik-Fachhandel, an ausgewählten Kiosken und garantiert beim Presse-Fachhändler. Sie können Elektor auch direkt bei www.elektor.de bestellen.

Änderungen vorbehalten!



Abo-Service:
E-Mail: abo@elektor.de

Bestellannahme und Bestellservice:
E-Mail: bestellung@elektor.de

Geschäftszeiten
Montag – Donnerstag von 08:30 bis 17:00 Uhr
Freitag von 08:30 bis 12:30 Uhr
Tel. +49 241 88 909-0
Fax +49 241 88 909-77

Unser Kundenservice berät Sie bei allen Fragen zu Bestellungen, Lieferterminen und Abonnements. Änderungen, Reklamationen oder besondere Wünsche (wie z. B. Geschenkabonnement) richten Sie ebenfalls an den Kundenservice. Vergessen Sie bitte nicht, Ihre Kundennummer anzugeben – falls vorhanden.

Einzelheft
Deutschland € 7,40
Österreich, Belgien, Luxemburg € 8,20
Schweiz CHF 14,30

Jahresabonnement-Standard
Deutschland € 79,50
Österreich, Belgien, Luxemburg € 86,50
Schweiz € 99,50
Andere Länder € 104,50

Jahresabonnement-PLUS
Deutschland € 97,00
Österreich, Belgien, Luxemburg € 104,00
Schweiz € 117,00
Andere Länder € 122,00

Probeabonnement
Alle Länder € 14,90

Studentenabo-Standard
Deutschland € 62,00
Österreich € 67,60
Schweiz € 78,00

Studentenabo-PLUS
Deutschland € 79,50
Österreich € 85,10
Schweiz € 95,50

Upgrade zum Abo-PLUS
Alle Länder € 17,50

Jahres- und Studentenabonnements (11 Hefte) dauern immer 1 Jahr und verlängern sich automatisch um weitere 12 Monate, wenn nicht spätestens 2 Monate vor Ablauf schriftlich gekündigt wird.

Änderungen und Irrtümer vorbehalten.

Bankverbindungen
Commerzbank Aachen
Konto 1 201 102 (BLZ 390 400 13)
IBAN: DE89 3904 0013 0120 1102 00
BIC: COBADEFFXXX

Postgiro Köln
Konto 229 744-507 (BLZ 370 100 50)
IBAN: DE17 3701 0050 0229 7445 07
BIC: PBNKDEFF

Elektor-PCB-Service

Die Adresse für Platinen, Prototypen und Multilayer

Möchten Sie Ihre selbst entworfene Platine schnell und zuverlässig geliefert bekommen? In Kleinserie und dabei zu einem unschlagbar günstigen Preis?



Bestellen Sie jetzt Ihre individuelle Platine beim Elektor-PCB-Service!



Überzeugen Sie sich selbst vom Elektor-Leiterplatten-Service – jetzt unter

www.elektorpcbservice.de!

Elektor-PCB-Service ist der Leiterplatten-Service von Elektor! Über die Website www.elektorpcbservice.de können Sie Ihren Entwurf als professionelle Platine herstellen lassen. Der Elektor-PCB-Service ist die richtige Adresse für Prototypen von neu entwickelten Platinen und für die Produktion modifizierter Elektor-Platinen.

Brauchen Sie kurzfristig einige Muster (Protos) oder eine Kleinserie (Batch), bestehend aus 5 bis 50 Exemplaren? Der Elektor-PCB-Service bietet jetzt beides zu einem günstigen Preis. Sie müssen uns nur über unsere Website Ihr Platinenlayout zusenden.

- Höchste Präzision und Industrie-Qualität zum günstigen Preis
- Kein Mindestbestellwert
- Keine Film- oder Einrichtungskosten
- Keine versteckten Kosten
- Online-Preisrechner
- Versand innerhalb von 5 Werktagen

Hier ist meine Anschrift:

Firma _____
Vorname _____
Name _____
Straße, Nr. _____
PLZ, Ort _____
Land _____
Kunden-Nr. **DE** _____
E-Mail _____

Innerhalb
Deutschlands
kein Porto
nötig!

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

Hier ist meine Anschrift:

Firma _____
Vorname _____
Name _____
Straße, Nr. _____
PLZ, Ort _____
Land _____
Kunden-Nr. **DE** _____
E-Mail _____

Innerhalb
Deutschlands
kein Porto
nötig!

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

Hier ist meine Anschrift:

Firma _____
Vorname _____
Name _____
Straße, Nr. _____
PLZ, Ort _____
Land _____
Kunden-Nr. **DE** _____
E-Mail _____

Innerhalb
Deutschlands
kein Porto
nötig!

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

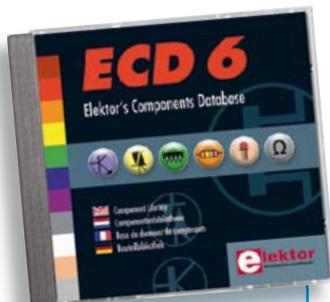
ECD 6

Elektor's Components Database

NEU!

➔ Neue Edition mit noch mehr Bauteilen

Diese neue Edition umfasst acht Datenbanken für ICs, Germanium- und Silizium-Transistoren, FETs, Thyristoren, Triacs, Dioden und Optokoppler. Weitere elf Anwendungen zur Berechnung von Vorwiderständen bei LEDs, Spannungsteiler, Ohmsches Gesetz sowie Farb-codeschlüssel für Widerstände und Induktivitäten etc. runden das Paket ab.



ISBN 978-90-5381-258-7
€ 29,50 • CHF 40,20

So verfügen Sie über die Daten von über 7.800 ICs; knapp 35.700 Transistoren, FETs, Thyristoren und Triacs; etwa 25.000 Dioden und über 1.800 Optokopplern. Alle genannten Datenbank-Anwendungen sind interaktiv, d. h. Sie können Bauteile hinzufügen, ändern oder ergänzen.

Weitere Infos & Bestellung unter
www.elektor.de/shop



Elektor Electronic Toolbox

Erhältlich im
App Store

Umfangreiche Elektroniker-App von Elektronikern für Elektroniker

Die neue „Elektor Electronic Toolbox“-App ist ganz auf die Belange von Elektronikern zugeschnitten. 29 Einzelprogramme/Anwendungen können über eine übersichtliche Oberfläche ausgewählt werden. Sehr hilfreich im Entwickleralltag sind die Datenbanken für die Bauteilgruppen Bipolar-Transistoren, FETs, Triacs, Thyristoren, Dioden und ICs. Ein Bauteil kann anhand der Typenbezeichnung kinderleicht ausgewählt werden – eine Internetverbindung ist nicht notwendig. Insgesamt sind über 45.000 Bauteile in den Datenbanken verzeichnet. Hinzu kommt eine Spezialdatenbank, in der die Belegung einer Vielzahl von Steckverbindern aus den Bereichen Audio & Video, Computertechnik und Telefon nachgeschlagen werden kann. Nützlich sind auch die interaktiven Bauteilwert-Kalkulatoren.

Tools wie eine virtuelle Widerstandsuhr, ein Umrechner zwischen Maßeinheiten, eine Schaltsymboldatenbank und vieles mehr runden die Elektor-App ab.

Die neue „Elektor Electronic Toolbox“ (geeignet für iPhone, iPod und iPad) kann zum Preis von nur 4,99 Euro heruntergeladen werden.

INSERENTENVERZEICHNIS DEZEMBER 2011

AudioXpress	www.cc-webshop.com	42
Becker & Müller	www.becker-mueller.de	9
Beta Layout	www.pcb-pool.com	41
DesignSpark chipKIT™ Challenge	www.chipkitchallenge.com	63
Emis	www.emisgmbh.de	55
Eurocircuits	www.eurocircuits.de	21
Jackaltac	www.jackaltac.com	11
LeitOn	www.leiton.de	55
Max-Planck-Institut	www.mpg.mpg.de	25
Markt		75
Microchip	www.microchip.com	2
MikroElektronika	www.easypic7.com	3
PICO Technology	www.picoscopemso.com/3116	25
Reichelt	www.reichelt.de	17
Schaeffer AG	www.schaeffer-ag.de	55
Volkswagen AG	www.volkswagen-nutzfahrzeuge.de/gewerbekunden	88

Beilagenhinweis:

Pollin Electronic www.pollin.de

Event-Kalender

Workshops • Seminare • Webinare • Weiterbildungen

➔ Top-Fachleute aus der Branche referieren über ein faszinierendes Thema!



Echtzeitbetriebssysteme in Theorie und Praxis

München 06.12. bis 08.12.2011
www.elektor.de/ezb-systeme

FPGA – Der Logikbaukasten für Jedermann

Villingen-Schwenningen 09.12.2011
www.elektor.de/fpga-seminar

CAN und AVR

Stuttgart 01.03.2012
München 22.03.2012
Frankfurt 26.04.2012
Zürich (CH) 07.06.2012
www.elektor.de/can-workshop

Gratis-Webinare

Twitter mit E-blocks 17.11.2011
Let's Build a Chaos Generator 15.12.2011
Here comes The Elektor Bus! 19.01.2012
www.elektor.de/webinar



Änderungen vorbehalten.

Weitere Infos unter
www.elektor.de/events



Für alle Elektrikergebäudetechnikerinstallateure. Und alle anderen. Die „Handwerksfamilie“ von Volkswagen Nutzfahrzeuge.

Mit der „Handwerksfamilie“ bieten wir Ihnen das passende Fahrzeug für jeden Bedarf. Und wer kennt den besser als Sie selbst? Entscheiden Sie sich für unsere Modelle und deren Produktlinien aus der „Handwerksfamilie“: den Caddy®, den Amarok DoubleCab, den Transporter oder den neuen Crafter. Zu den Aktionsausstattungen der einzelnen Modelle können Sie weitere Sonderausstattungen individuell auswählen und sich so Ihr persönliches Fahrzeug zusammenstellen. Für alle, die nicht nur beim Kauf, sondern auch beim Verbrauch sparen möchten, gibt es viele weitere Modelle mit der zukunftsweisen BlueMotion Technology. Auf zum Familientreffen – und profitieren Sie als Innungsmitglied zusätzlich von attraktiven Sonderkonditionen** aus dem Branchenabkommen mit dem Zentralverband der Deutschen Elektro- und Informationstechnischen Handwerke. Kontaktieren Sie uns und erfahren Sie mehr.



Nutzfahrzeuge

*Je nach Modell. Gegenüber der unverbindlichen Preisempfehlung für ein vergleichbar ausgestattetes Serienmodell zuzüglich Sonderkonditionen für Innungsmitglieder. Hier am Beispiel des Crafter Kastenwagens (ausgewählte Modellvarianten). Die festgelegten Aktionsausstattungen sind bei allen Modellen notwendig, um den vollen Preisvorteil zu erhalten. **Das Angebot „Handwerksfamilie“ gilt ausschließlich für gewerbliche Einzelabnehmer, die hauptberuflich tätig sind. Sonderkonditionen gelten für ausgewählte Modelle aus der „Handwerksfamilie“ für Branchen, mit denen ein Abkommen besteht. Detaillierte Informationen zur Aktion erhalten Sie bei Ihrem teilnehmenden Volkswagen Nutzfahrzeuge Partner. Die Aktion ist befristet vom 18.07. – 31.12.2011. Der Produktname Caddy® ist eine eingetragene Marke von Caddie S. A. und wird von Volkswagen Nutzfahrzeuge mit freundlicher Genehmigung von Caddie S. A. benutzt. Abbildung zeigt Sonderausstattung gegen Mehrpreis.