

Neu: 8 Seiten mehr!

magazine

September 2014 | Nr. 525

eilektor

Professionelles Labornetzteil



Hohe Stabilität und präzises Regelverhalten

- **AVR-Prozessorboards für schnelles Prototyping** | 16-bit-ADC-Modul
- Empfänger für ELF-Wellen | **LED-Klassifizierungs-Tool** | Verbesserter Stromtransformator
- Visual Basic auf dem Raspberry Pi | **DesignSpark**
- Tipps & Tricks** ● Review: Multifunktionales Messsystem
- Peltier-Module ● **Retronik: Taschencomputer HP-71B**



Alle Elektor-Artikel der Jahre 2000 bis 2009 auf DVD!



NEU!

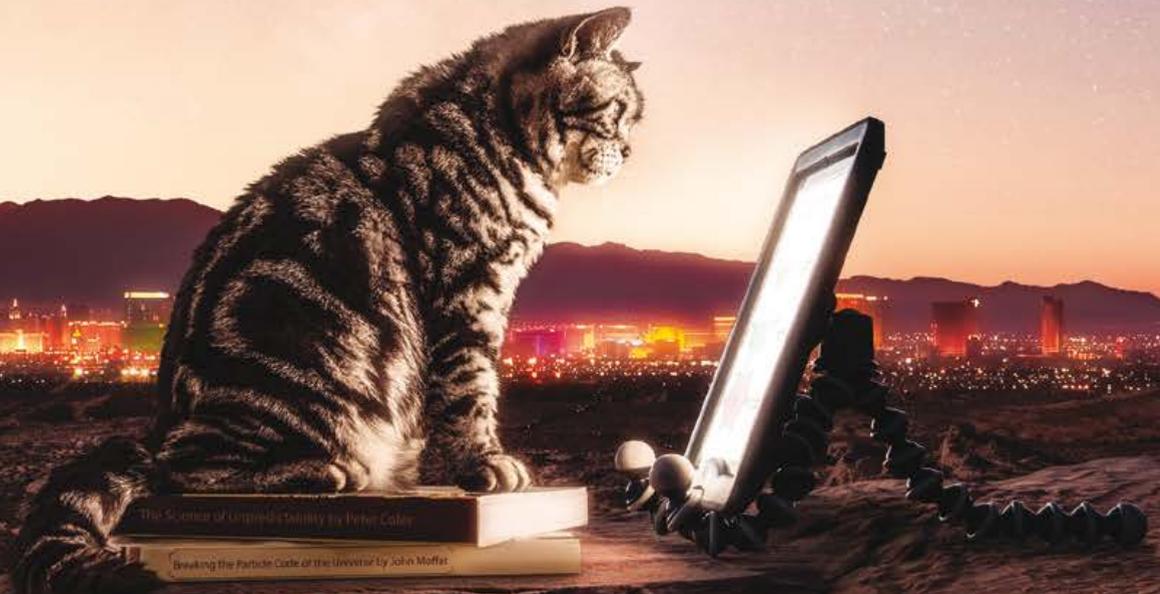
**Ein Muss
für jeden
Elektor-Leser!**

- Elektronisches Jahrgangs-Archiv, Artikel als PDF, schnelle Suchfunktion
- 110 Elektor-Hefte, über 4000 Artikel, bequeme Druckfunktion
- Ideen, Schaltungen und Projekte für Elektroniker im Beruf, in der Weiterbildung und der Freizeit

ISBN 978-3-89576-292-5
€ 89,00 • CHF 110,40

Jetzt unter www.elektor.de/2000-2009 bestellen!

EXPLAINING CES TO SOMEONE WHO HASN'T ATTENDED IS LIKE EXPLAINING PARTICLE PHYSICS TO A HOUSECAT.



In other words, you really can't. What you can do is make sure you don't miss what promises to be our most innovative and rewarding show ever. That and maybe take some pictures for the unfortunate cats left behind. Register at CESweb.org.



THE GLOBAL STAGE FOR INNOVATION

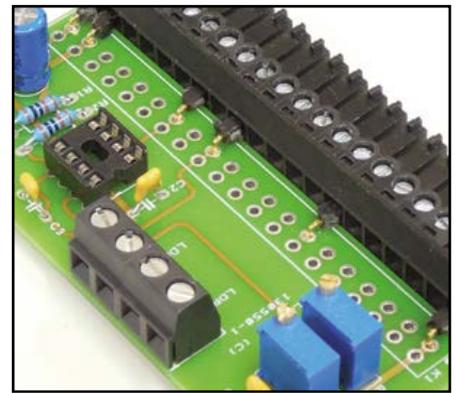
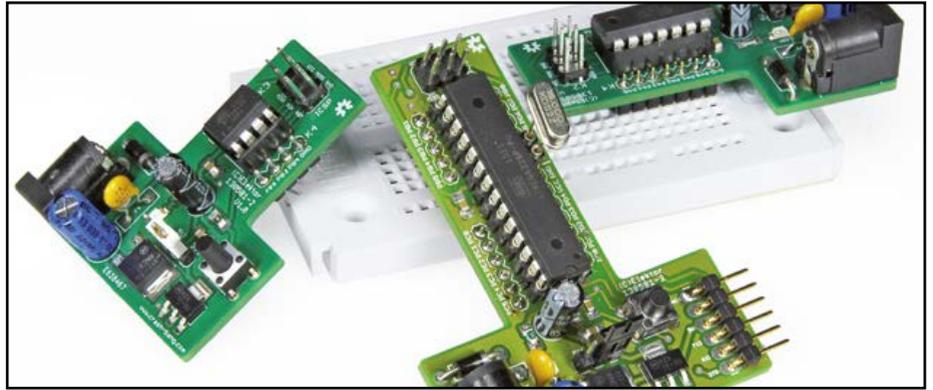


JAN. 6-9, 2015

LAS VEGAS, NEVADA

CESWEB.ORG

#CES2015



● Community

6 Impressum

8 Elektor Maker Space
@electronica 2014

10 Internet of Things –
auf dem Weg zum Protokoll

● Labs

48 DesignSpark Tipps & Tricks
Bauteilplatzierung

50 Seltsame Bauteile
Peltier-Module

52 ELPP:
Elektor Labs Preferred Parts

54 Preisgünstiger myDAQ-Anschluss

55 3D-Drucken kann nützlich sein!

56 Bauteil-Tipp: MagI³C-VDRM

Spannungsregler werden immer kleiner, besser und effizienter. Die Entwicklung ist jedoch noch lange nicht zu Ende.

● Projects

12 Professionelles Labornetzteil

Nicht nur „normale“ Netzteile gibt es wie Sand am Meer, auch Labornetzteile kann man überall in jeder Form kaufen. Wer aber Qualität will, hohe Stabilität und präzises Regelverhalten, der muss für erstaunlich wenig Ausgangsleistung verhältnismäßig tief in die Tasche greifen. Oder sich so ein Werkzeug selbst bauen...

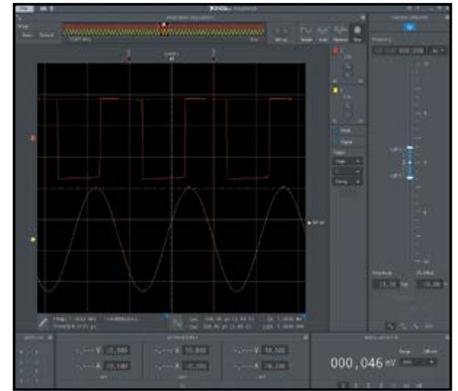
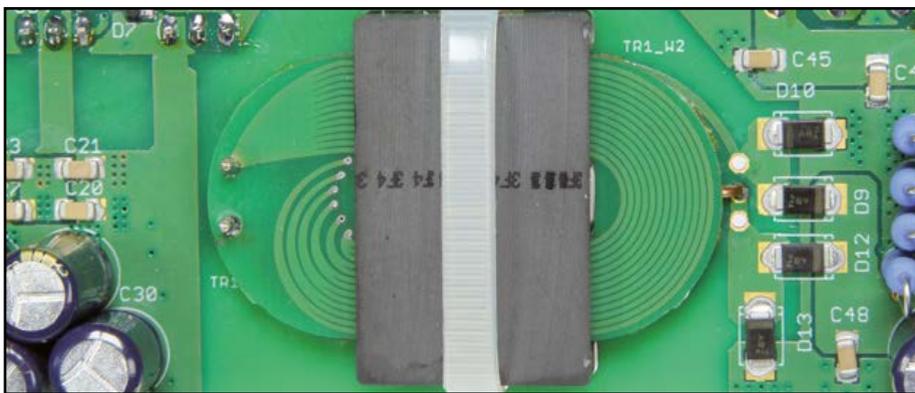
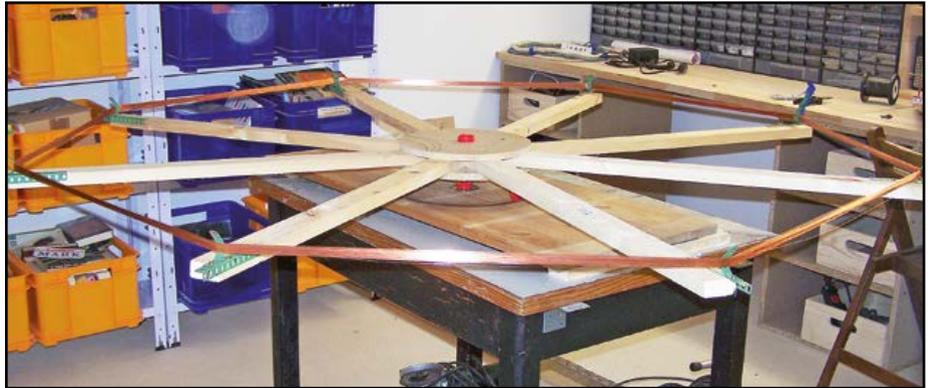
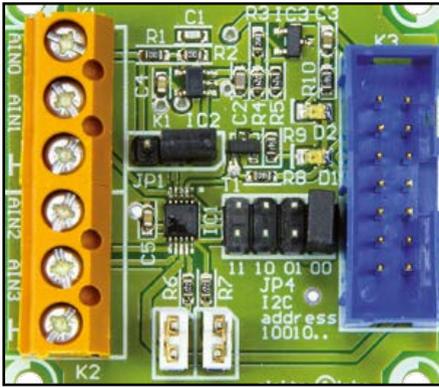
24 T-Boards 8/14/28

Die Arduino-Plattform ist ein perfekter Weg, um in die Welt der

Mikrocontroller und Embedded-Systeme einzusteigen. Irgendwann aber stehen größere Projekte an, die ein Debugging und/oder die Entwicklung eigener Boards erfordern. Dieser Schritt ist mit den T-Boards leicht gemacht: Die drei AVR-Prozessorboards enthalten bereits eine Spannungsregelung für 3,3 und 5 V und den Programmieranschluss.

34 Daten loggen mit 16 bit

Für hochauflösende und genaue Messungen bieten sich spezielle AD-Wandler-Chips an. Im Elektor-Labor haben wir eine Platine entwickelt, die einen vierkanaligen 16-bit-ADC enthält. Dank des GnuBlin/EEC-Steckverbinders lässt sich das Modul unmittelbar an das Elektor-Linux-Board, das Xmega-Webserver-Board und das neue Elektor Extension-Shield für Arduino anschließen. Eine universell verwendbare C-Bibliothek macht die Ansteuerung des AD-Wandlers einfach.



40 ELF-Empfänger

Zwischen 0 Hz und der Bahnfrequenz von 16,666 Hz sind recht interessante Signale zu empfangen. Mit dem hier gezeigten Receiver, dem in einem eigenen Artikel vorgestellten ADC-Modul und einer kostenlosen PC-Software ist es möglich, solche ELF-Signale zu empfangen und aufzuzeichnen.

60 LED-Klassifizierungs-Tool

Dieses Projekt ist ein Spin-off der einstellbaren Präzisions-Stromquelle, die wir Anfang dieses Jahres veröffentlichten. Gleichzeitig ist es ein tolles Hilfsmittel, um durch visuellen Vergleich bedrähete LEDs nach ihrer Farbe, der Helligkeit und/oder dem Abstrahlwinkel zu sortieren.

62 Mikrocontroller für Einsteiger (5)

Eines der größten Kapitel im Datenblatt des ATmega328 behandelt die drei Timer des Controllers. Ihre

Anwendungen sind so vielfältig, das hier nur ein kleiner Teil der Möglichkeiten vorgestellt werden kann. Die wesentlichen Aufgabenfelder heißen Zeitmessung, Frequenzmessung, Signalerzeugung und PWM-Ausgabe.

72 Verbesserter Stromtransformator

Ein Nachtrag zu einem Elektor-Artikel von Martin Ossmann.

74 Visual Basic auf dem Raspberry Pi

Der Raspberry Pi wird normalerweise in Python programmiert. Allerdings können Programme mit einer schönen Grafik dabei lange nicht so leicht erstellt werden wie unter Visual Basic.

76 Multifunktionales Messsystem

VirtualBench heißt ein innovatives Messsystem von National Instruments, das fünf typische Prüfplätze in einem integrierten System vereint. Bedient wird das

Messgeräte-Ensemble über einen Desktop- oder Tablet-Computer. In Europa war Elektor unter den Ersten, die ein Exemplar testen durften.

● Magazine

82 Retronik

Der Zahlenfresser HP-71B (1984)

86 Hexadoku

Sudoku für Elektroniker

90 Vorschau

Nächsten Monat in Elektor

Impressum

45. Jahrgang, Nr. 525 September 2014
Erscheinungsweise: 10 x jährlich
(inkl. Doppelhefte Januar/Februar und Juli/August)

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.

Hauptsitz des Verlags

Elektor International Media
Allee 1, NL-6141 AV Limbricht

Anzeigen:

Margriet Debeij (verantwortlich)
Tel. 02 41/88 909-13 / Fax 02 41/88 909-77
Mobil: +31 6 510 530 39
E-Mail: m.debeij@elektor.de

Julia Grotenrath

Tel. 02 41/88 909-16 / Fax 02 41/88 909-77
E-Mail: j.grotenrath@elektor.de

Es gilt die Anzeigenpreisliste Nr. 44 ab 01.01.2014

Distribution:

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

Der Herausgeber ist nicht verpflichtet, unverlangt eingesandte Manuskripte oder Geräte zurückzusenden. Auch 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.

© 2014 elektor international media b.v.
Druck: Senefelder Misset, Doetinchem (NL)
ISSN 0932-5468

Analog + Digital

Dass die Elektronik-Community in eine Analog- und eine Digital-Fraktion zerfällt, ist ein altes Klischee, das so nicht stimmen kann. Das wissen wir spätestens aus unserer Online-Leserumfrage: Eine Vielzahl von Lesern gab damals zum Beispiel an, sich gleichzeitig für Röhren und das Thema „Embedded Linux“ zu interessieren. Gerade solche vielseitig interessierten Leser dürften sich an unserem Doppelprojekt „ELF-Empfang mit Arduino-Datenlogger“ erfreuen, das Sie in dieser Ausgabe finden. Von Tipps zum Wickeln der Spule über Messungen an der Empfangsschaltung, von der Ansteuerung des 16-bit-ADCs via I2C bis zur Auswertung der gesampelten Daten per PC-Software – hier ist alles dabei. Bei der Hardware haben wir abermals Wert auf Wiederverwendbarkeit gelegt. Der verwendete 16-bit-ADC wurde auf ein einfach zu handhabendes Board gesetzt, zusammen mit dem bekannten 14-poligen EEC/Gnublin-Stecker. Per Flachkabel geht es dann weiter zu unserem neuen Elektor-Extension-Shield (das Burkhard Kainka auch in seinem Mikrocontrollerkurs verwendet). Zu diesen ersten „Gnublin-Modulen“ aus dem Elektor-Labor werden sich noch weitere gesellen. Denn Baukastensysteme liegen im Trend, da man damit besonders schnell zu funktionierenden Prototypen und Projekten kommt. Daneben bieten solche Module auch Elektronik-Anfängern einen leichten Zugang zu diesem schönen Fachgebiet beziehungsweise Hobby, das uns alle verbindet. Wichtig ist dabei auch eine gute „Versorgung“ mit Software. In den nächsten Ausgaben wollen wir weitere Impulse in diese Richtung geben.



Bis dann und viel Spaß beim Lesen des Septemberhefts!

Jens Nickel

Chefredakteur Elektor

Unser Team

Chefredakteur:

Jens Nickel (v.i.S.d.P.) (redaktion@elektor.de)

Ständige Mitarbeiter:

Dr. Thomas Scherer, Rolf Gerstendorf

Leserservice:

Ralf Schmiedel

Korrekturen:

Malte Fischer

Internationale Redaktion:

Harry Baggen, Jan Buiting, Denis Meyer

Elektor-Labor:

Thijs Beckers, Ton Giesberts, Wisse Hettinga,
Luc Lemmens, Mart Schroijen, Jan Visser,
Clemens Valens, Patrick Wielders

Grafik & Layout:

Giel Dols



Germany
Ferdinand te Walvaart
+49 241 88 909-17
f.tewalvaart@elektor.de



United Kingdom
Carlo van Nistelrooy
+44 20 7692 8344
c.vannistelrooy@elektor.com



Netherlands
Ferdinand te Walvaart
+31 46 43 89 444
f.tewalvaart@elektor.nl



France
Denis Meyer
+31 46 4389435
d.meyer@elektor.fr



USA
Carlo van Nistelrooy
+1 860-289-0800
c.vannistelrooy@elektor.com



Spain
Jaime González-Arintero
+34 6 16 99 74 86
j.glez.arintero@elektor.es



Italy
Maurizio del Corso
+39 2.66504755
m.delcorso@inware.it



Sweden
Carlo van Nistelrooy
+31 46 43 89 418
c.vannistelrooy@elektor.com



Brazil
João Martins
+31 46 4389444
j.martins@elektor.com



Portugal
João Martins
+31 46 4389444
j.martins@elektor.com



India
Sunil D. Malekar
+91 9833168815
ts@elektor.in



Russia
Nataliya Melnikova
+7 (965) 395 33 36
Elektor.Russia@gmail.com



Turkey
Zeynep Köksal
+90 532 277 48 26
zkoksal@beti.com.tr



South Africa
Johan Dijk
+31 6 1589 4245
j.dijk@elektor.com



China
Cees Baay
+86 21 6445 2811
CeesBaay@gmail.com

Unser Netzwerk



VOICE COIL



audio X PRESS



Die Elektor-Community



Unsere Partner und Sponsoren



Batronix
www.batronix.com/go/37 57



EBV Elektronik
www.ebv.com/de 59



Beta Layout
www.beta-estore.com 11



LeitOn
www.leiton.de 51



CadSoft
www.cadsoft.de 33



Reichelt
www.reichelt.de 92



CES International 2015
www.cesweb.org 3



Schaeffer AG
www.schaeffer-ag.de 51

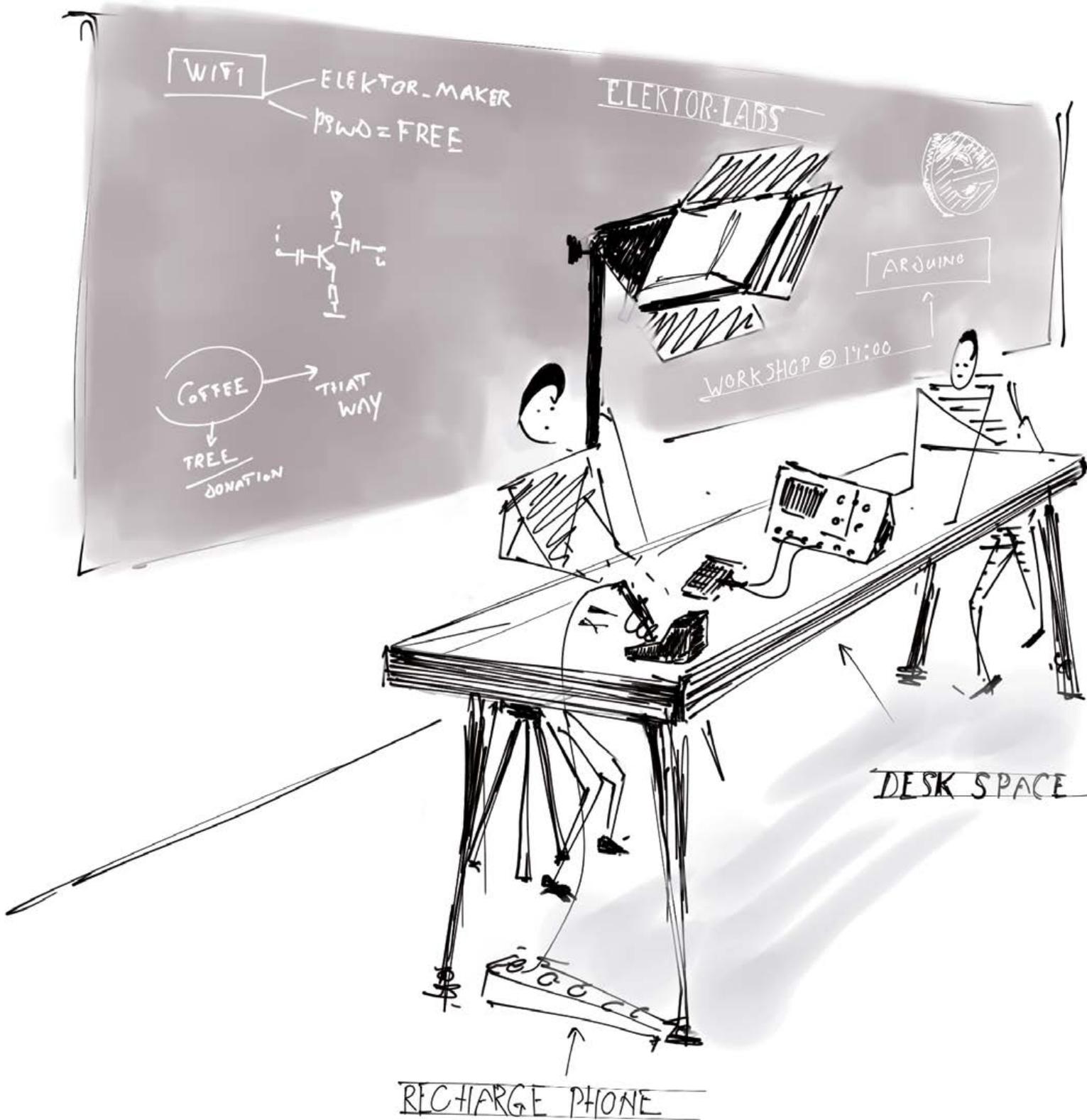


Conrad
www.conrad.de 51

Sie möchten Partner werden?

Kontaktieren Sie uns bitte unter m.debeij@elektor.de (Tel. 02 41/88 909-13).

ELEKTOR-MAKER-S



SPACE

Elektor Maker Space

@electronica 2014, München

Von **Wisse Hettinga** (Elektor-Labor)

Bestimmt ist Ihnen schon mal der Begriff „Maker“ zu Ohren gekommen und auch die damit verbundene Revolution sagt Ihnen was. Von tollen Sachen zu lesen ist eine Sache, doch so etwas auf der weltgrößten Elektronikmesse selbst zu probieren ist noch viel cooler! Man kann dort relaxen, das Smartphone aufladen, fachsimpeln, kaffeetrinken und richtig mit Elektronik in Kontakt sein. Man kann alles mitbringen, was man auspacken, (ent)löten, zeigen, messen, austesten oder sogar verarduinoen und raspberrieren möchte. Fühlen Sie sich eingeladen, Dinge ganz praktisch zu elektorisieren!

Die Entwickler und Ingenieure von Elektor stehen bereit, um Sie bei elektronischen Problemen zu unterstützen und sperrige Elektronik vor Ort zum Laufen zu bringen. Das Ganze in deutsch, englisch, niederländisch, spanisch und C++ ;-).

Für dieses spezielle Event stellt das Elektor-Labor zur Verfügung: Arbeitsplatz, Werkzeug, Messgeräte, 3D-Drucker und kostenloses WLAN. Nicht zu verachten: Mini-Workshops, Techtalk, Kaffee und jede Menge Steckdosen zum Aufladen von Handys, Tablets, Laptops und anderen Apparaten.

Navigation zum Elektor Maker Space: Halle A6, Stand 380

Lassen Sie sich von Elektor auf den neuesten Stand bringen – registrieren Sie sich für unseren Newsletter rechts unten auf www.elektor.de.

WER:

SIE UND ELEKTOR INTERNATIONAL MEDIA

WAS:

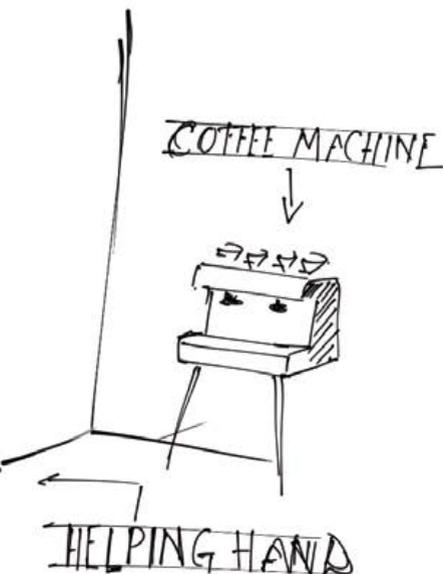
ELEKTOR MAKER SPACE

WANN:

11.-14. NOVEMBER 2014

WO:

**ELECTRONICA, MÜNCHEN,
HALLE A6 / STAND 380**



Internet of Things

Auf dem Weg zum Protokoll

Das „Internet of Things“ wird unser Leben verändern. Damit sich Geräte untereinander verstehen, braucht man aber ein einheitliches, anwendungsorientiertes Protokoll. Es gilt zum Beispiel, eine allgemeingültige Darstellung für den Messwert eines Sensors oder den Sollwert für einen Aktor zu finden.

Idealerweise sollten auch kleine, günstige Mikrocontroller mit dem Protokoll zurechtkommen. Gleichzeitig muss man zumindest einen Großteil der Anwendungsfälle abdecken, die etwa bei der Hausautomatisierung anfallen [1]. Keine leichte Nuss, und deshalb haben wir im Märzheft die Community dazu aufgerufen, uns unter der Adresse www.iot-contest.com Vorschläge einzureichen.

Mittlerweile haben uns diverse Ideen erreicht, darunter waren Weblinks auf bereits bestehende Protokolle [2][3]. Hier möchten wir noch einen weiteren Vorschlag beisteuern, der mit Absicht noch nicht bis zu Ende ausgearbeitet ist, sondern eine

Diskussionsgrundlage sein soll. Wenn Ihnen die Ideen zusagen oder auch nicht, wenn Ihnen Verbesserungsvorschläge oder neue, eigene Lösungen einfallen, sollten Sie uns das mitteilen unter www.iot-contest.com!

(140298)

[1] www.iot-contest.com/index.php?content=infos

[2] www.xpautomation.org

[3] <http://ansari-electronics.com/comlinksrv/>



Vorschlag für ein Datenformat

Zwischen den Geräten werden Nachrichten (Messages) ausgetauscht, beispielsweise als HTTP-POST-Daten (was noch diskutiert werden müsste). Jede Message ist aus einem noch zu bestimmenden Header und mehreren Blöcken zusammengesetzt. Die Art eines Nachrichtenblocks wird durch zwei Schlüsselworte bestimmt (die wir als Case und Mode bezeichnen, siehe unten).

Ein paar Beispiele:

„**Set Value**“: Setzen eines Werts auf einem Aktor

„**Current Value**“: Ein Messwert wird übermittelt

„**Query Value**“: Anfrage, einen Messwert zu übermitteln (Polling)

„**Set LimitMin**“, „**Set LimitMax**“: Grenzwerte bei einem intelligenten Sensor setzen

„**Set Interval**“: Sensor anweisen, in einem bestimmten Intervall Messwerte zu übermitteln

„**Option Min**“, „**Option Max**“: Sensor teilt den eigenen Messbereich mit

„**Option Value**“ (gefolgt von einer physikalischen Einheit und Größe): Sensor teilt mit, was er messen kann und in welcher Einheit

Hier der Aufbau der Messages im Überblick (in Klammern gesetzte Teile sind optional, die // bedeuten „oder“):
Message: Header + Block (+ Block (+ Block ...))

Block: Case + Mode + (Subnode +) (Location +) (Time +) Data

Case: „Set“ // „Current“ // „Query“ // „Option“

Mode: „Value“ // „Min“ // „Max“ // „Interval“ // „LimitMin“ // „LimitMax“

Subnode: *Integer*

Location: LocationString: *String*

Time: TimeString: *String*

Data: Value (+ Value (+ Value + ...))

Value: Number + (Accuracy +) Unit + Quantity

Number: *Float* // *Integer* // *Binary*

Accuracy: *Float* // *Integer*

Unit: UnitKey: *String*

Quantity: QuantityKey: *String*

Letzten Endes werden alle Elemente auf die Datentypen *Float*, *Integer*, *Binary* und *String* zurückgeführt. Wie die Elemente aber konkret (im ASCII-Format) dargestellt werden, ist eine Aufgabe, die noch zu lösen wäre.

Vorschläge brauchen wir unter anderem für die Darstellung des Ortes und der Zeit (LocationString, TimeString). Das Protokoll könnte sowohl noch vereinfacht als auch ausgebaut werden. Zum Beispiel fehlen noch Presets (Sollwerte, die auf einem Aktor voreingestellt werden können). Auch Alarm-Nachrichten der Sensoren („Grenzwert unterschritten“) sind noch nicht abgedeckt.

25 Jahre **Beta**

LAYOUT

create : electronics

SPARPREIS zu unserem Jubiläum!

über **50 €**
sparen! **



€ 444,00*

Jubiläums-
Reflow-Kit V3

* inkl. MwSt. und zzgl. Versandkosten

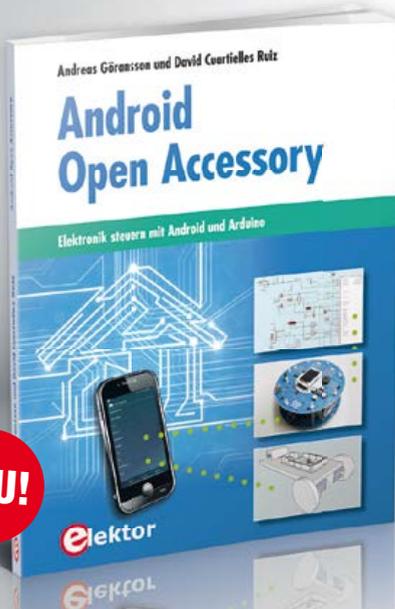
** gegenüber Kauf der Einzelkomponenten

www.beta-eSTORE.com

eSTORE[®]
Beta LAYOUT

Android Open Accessory

Elektronik steuern mit Android und Arduino



NEU!

Android Open Accessory (kurz AOA) ist ein einfaches und sicheres Protokoll zur Verbindung von Mikrocontroller-gesteuerten Geräten mit einem Android-Smartphone oder -Tablet. Dieses Buch zeigt anhand von leicht nachbaubaren Schaltungen und den dazu gehörenden Programmbeispielen, wie man AOA in Verbindung mit der Mikrocontroller-Plattform Arduino verwendet, um täglich anfallende Aufgaben im Haus zu automatisieren: Beleuchtung, Belüftung, Klimatisierung und Musik-Entertainment-Systeme – bequem und komfortabel mit dem Smartphone, wohlgemerkt!

Die Grundkenntnisse des Arduino-Frameworks voraussetzend, versorgt das visionäre Autorenduo Göransson/Cuartielles Ruiz den Leser mit den Werkzeugen (Tools), die er braucht, um nützliche und anspruchsvolle Projekte realisieren zu können.

Die Programmbeispiele aus diesem Buch stehen auf der Elektor-Website zum Gratis-Download bereit.

380 Seiten (kart.) • Format 17 x 23,5 cm
ISBN 978-3-89576-279-6

€ 42,00
CHF 52,95

Weitere Infos & Bestellung unter www.elektor.de/android-open-accessory

Professionelles Labornetzteil

Qualität gibt es nicht umsonst...



Von **Arne Hinz** und
Martin Christoph

Nicht nur „normale“ Netzteile gibt es wie Sand am Meer, auch Labornetzteile kann man überall in jeder Form kaufen. Letztere tun es für viele Zwecke. Wer aber Qualität will, hohe Stabilität und präzises Regelverhalten, der muss für erstaunlich wenig Ausgangsleistung verhältnismäßig tief in die Tasche greifen. Oder sich so ein Werkzeug selbst bauen...

Wenn man hin und wieder einmal etwas bastelt, dann ist man schon mit einem kleinen Netzteil gut bedient, das bei etwa 1 A Belastbarkeit eine Spannung liefert, die zumindest irgendwie zwischen 1 V und 12 V einstellbar ist. Damit sind

die Bedürfnisse von drei Vierteln aller kleinen Schaltungen abgedeckt, die man mal eben so zusammenlötet. So eine einstellbare aber stabilisierte kleine Spannungsquelle mit Gehäuse drumrum ist bei Versandhändlern für Elektro-

nikbedarf schon ab 50 Euro zu haben und bietet dafür meist sogar die (analoge) Anzeige von Spannung **und** Strom gleichzeitig.

Die restlichen 25 % der Fälle verlangen aber nach mehr. Sie kommen vor allem bei anspruchsvollen Amateuren vor, die sich auch komplexere Elektronik-Projekte vornehmen oder eben in professionellen Entwicklungslabors, wo gute Labornetzeile gewissermaßen die Basis der ganzen Messtechnik darstellen und daher sehr viel Geld für Profi-Equipment in die Hand genommen werden muss. Die Frage, welche Qualitäten von Netzteilen denn „allgemein“ wichtig oder relevant sind, die ist wie so häufig nicht ganz einfach zu beantworten, denn die Kriterien variieren mit der Anwendung. Sicherlich aber verdient die Flut an Labornetzteilen, die sehr viel Leistung für wenig Geld offerieren, einen kritischen Blick:

Schon ab 100 Euro kann man Netzteile aus fernöstlicher Produktion kaufen, die einen einstellbaren Spannungsbereich von (fast) 0 V bis 30 V bei einem ebenfalls einstellbaren Strom von (fast) 0 A bis etwa 3 A bieten und darüber hinaus auch noch mit Digitalanzeigen, Fernsteuerung und anderen Features ausgestattet sind. Legt man noch mal 50 Euro drauf, dann kriegt man sogar Exemplare mit nahezu 600 W Ausgangsleistung in Schaltnetzteiltechnik. Aber Geiz ist hier alles andere als geil. Man muss kein Mathegenie sein um zu wissen, dass all die schönen Daten bei diesen Preisen nicht mit wichtigen Kriterien wie gutem Wirkungsgrad, Haltbarkeit und guter Spannungsstabilisierung zusammen machbar sind. Qualität bedeutet bei Netzteilen Aufwand und Aufwand kostet. Das hier vorgestellte Netzteil schlägt zwar keine Geräte im k€-Bereich, weist aber schon ziemlich gute Daten auf.

Schaltungskonzept

Gute Labornetzteile werden erstens nicht sehr heiß und sind zweitens stabil. Ersteres macht Schaltreglertechnik erforderlich, da Verlustleistung (= Hitze im Netzteil) sowohl der Lebensdauer als auch einer geringen Drift der Ausgangsspannung abträglich ist. Letzteres bedeutet dass die Ausgangsspannung sich (fast) nicht ändert, egal ob aufgrund von Belastung oder Temperatur (statische Stabilität). Dynamisch ist eine schnelle und saubere Ausregelung wichtig, die z.B. bei Lastwechseln so kleine Überschwinger wie möglich produziert. Die weitgehende Brumfreiheit und die Abwesenheit von Störungen sind selbstverständlich. Besonders beim Einsatz von

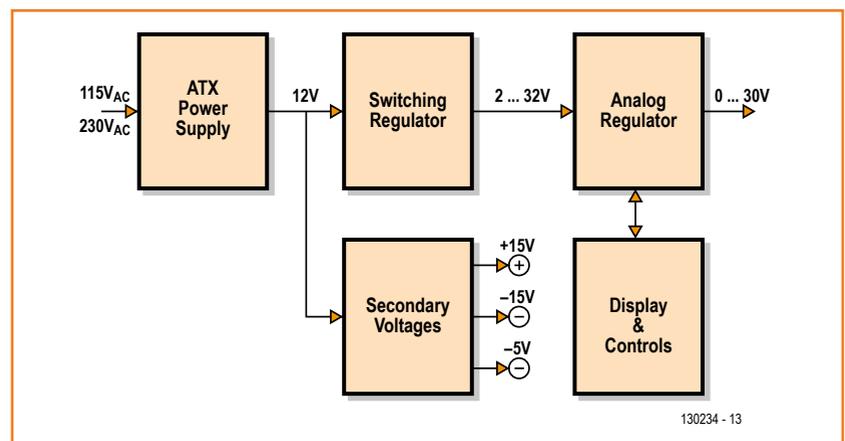
Technische Daten

- Effizientes Labornetzteil mit geschalteter Vorregelung
- Betrieb mit einem PC-Netzteil an 12 V
- Wirkungsgrad von bis zu 70 %
- Volle Potentialtrennung bei mehreren Modulen
- Ausgangsspannung 0...30 V einstellbar in 10-mV-Schritten
- Ausgangsstrom von 0...1 A einstellbar in 10-mA-Schritten
- Anzeige von Spannung und Strom mit LED-Displays
- Gute Lastausregelung
- Geringe Drift
- Schnellabschaltung per Taster

Schaltreglern sind höherfrequente Störwechsellspannungen aber gar nicht so einfach in den Griff zu bekommen. Abschließend wäre noch ein Kriterium zu nennen, dass weniger mit Qualität, sondern mit Praktikabilität zu tun hat: Wenn ein Netzteil mehrere Spannungen bietet, dann sollten diese galvanisch voneinander getrennt sein, damit man sie ohne Risiko beliebig kombinieren und zusammenschalten kann.

Wie man sieht, beißen sich die Kriterien Effektivität und Qualität (= niedrige Störpegel und hohe Regelgeschwindigkeit) gegenseitig. Die logische Auflösung des scheinbaren Widerspruchs liegt darin, dass man analoge und geschaltete Techniken kombiniert und so mit mehr Aufwand die positiven Eigenschaften beider Strategien erntet. Die Lösung ist ein klassischer Linearregler am Ausgang, mit dem die Spannung oder der Strom genau, schnell und störungsarm eingestellt werden kann. Vor dieser linearen Stufe liegt noch eine digitale, die „geschaltet“ dem analogen Teil eine Spannung zur Verfügung stellt, die immer etwas höher als die Ausgangsspannung ist, aber

Bild 1. Schon das Blockschaltbild des Labornetzteils zeigt, dass dies keine simple Schaltung ist.



dafür keine so hohe Qualität aufweist. Die lineare Ausgangsstufe wird daher auch bei niedrigen Ausgangsspannungen und hohem Strom nur eine geringe Verlustleistung aufweisen, da nicht viel Spannung an ihr abfällt. Das kostet mehr Bauteile und ist daher nicht nur teurer, sondern auch komplexer. Fertige Labornetzteile mit diesem Konzept haben daher stolze Preise.

Die Autoren haben dies berücksichtigt und ausgiebige Überlegungen angestellt. Die Entwicklung begann am Institut für Stromrichtertechnik und Elektrische Antriebe (ISEA) der RWTH Aachen; später hat Arne Hinz das Projekt im Rahmen eines Praktikums im Elektor-Labor noch optimiert. Das Resultat zeigt sich in der Blockschaltung von **Bild 1**. Um den Selbstbau einfacher zu gestalten

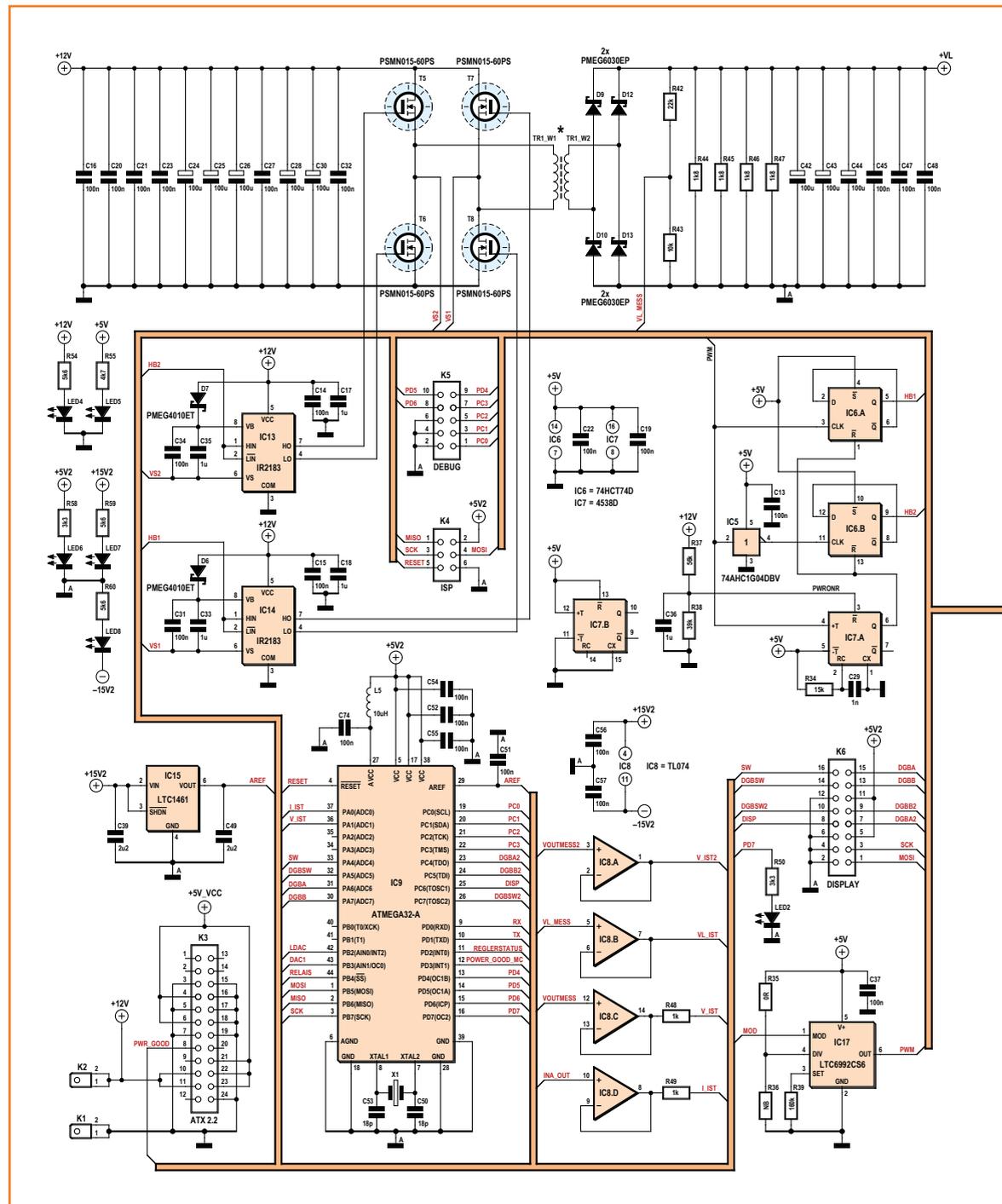
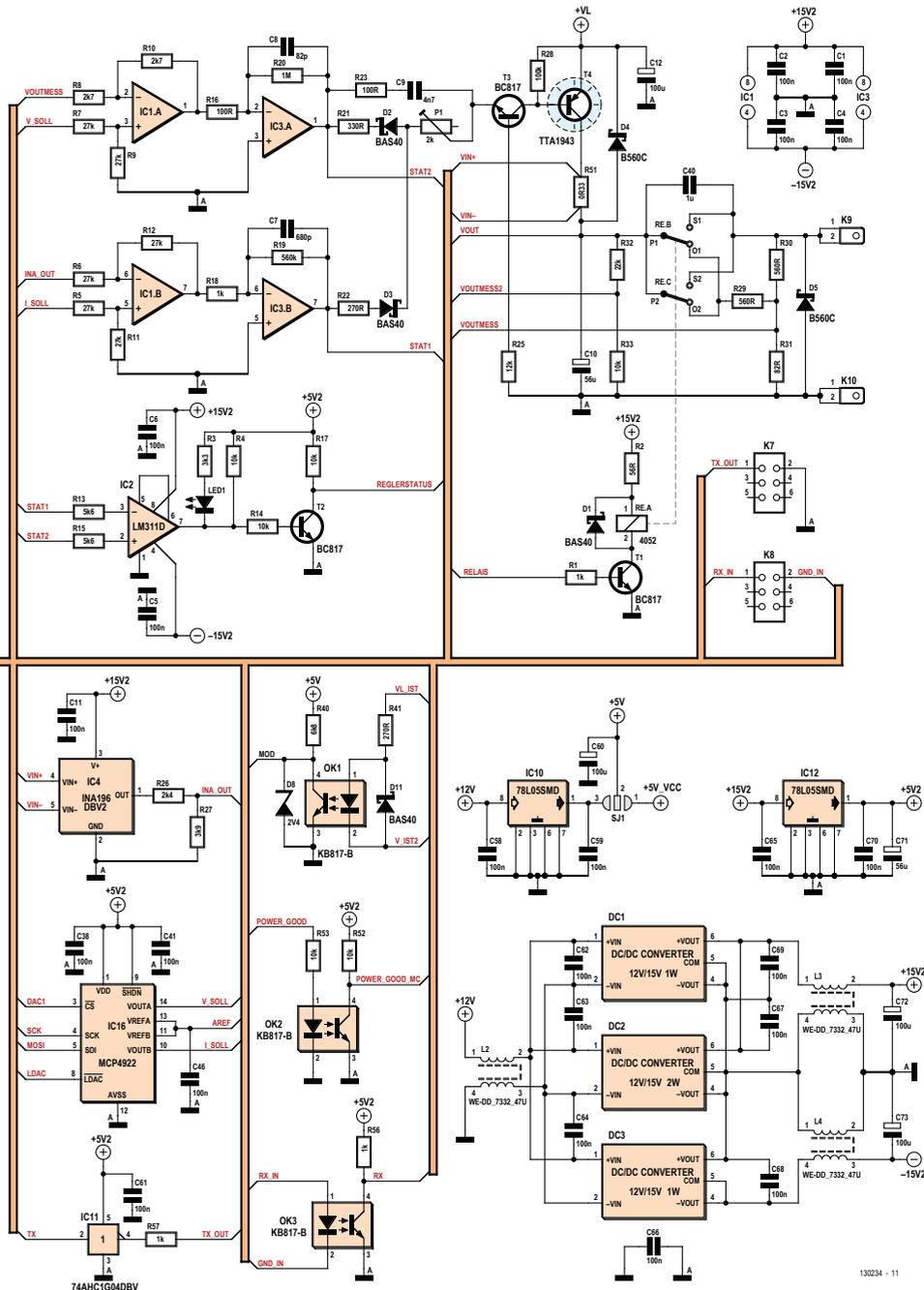


Bild 2. Die Schaltung des Labornetzteils ist ziemlich umfangreich, auch ohne Bedienelemente.

ten, wurde das vorgeschaltete Schaltnetzteil nicht für den direkten Betrieb am 230-V-Netz konzipiert. Stattdessen wird ein Niedervoltkonzept bevorzugt, das eine Eingangsspannung von 12 V benötigt. Diese 12 V lassen sich in ausreichender Qualität und Leistung bei niedrigen Kosten einem einfachen PC-Netzteil entnehmen, die es ja in vielen Leistungsstufen von unter 200 W bis zu 1 kW

gibt. Genau genommen handelt es sich also um ein dreistufiges Netzteilkonzept: In der oberen Reihe findet sich von links nach rechts zuerst ein handelsübliches ATX-Netzteil. Dieses speist den Schaltregler in der Mitte, der eine Art Vorregelung für die lineare Ausgangsstufe rechts bildet. Unten in der Mitte findet sich die Erzeugung der Betriebsspannungen von $\pm 15\text{ V}$ und $+5\text{ V}$ für



die Elektronik und rechts daneben das Modul mit Bedienelementen und der Anzeige von Spannung und Strom. Der Vorteil dieses dreistufigen modularen Konzepts ist, dass man mehrere galvanisch getrennte Spannungsquellen erhält, wenn man mehrere Kombinationen von Schaltvorstufe und Linearregler an einem einzigen ATX-Netzteil in einem Gehäuse betreibt. Die schaltende Vorregelung in der Mitte enthält nämlich dank der Verwendung eines Leistungsübertragers die galvanische Trennung. Alle Spannungen können daher beliebig kombiniert werden.

Detailschaltung

Die generellen Dinge sind gesagt: Das Labornetzteil für höhere Ansprüche besteht aus einer geschalteten Vorregelung (mit Potentialtrennung) und einer analogen Linearregelung am Ausgang. Das Ganze wird von einer leistungsfähigen 12-V-Quelle wie einem preiswerten ATX-Netzteil versorgt. Für die beiden Regelungen samt steuerndem Mikrocontroller ist eine Platine vorgesehen und für Display und Bedienelemente eine zweite. Zusammen gibt das ein kompaktes Modul, von denen man auch mehrere Exemplare in ein klassisches 19"-Gehäuse packen kann, die dann gemeinsam von einem PC-Netzteil gespeist werden.

Doch zunächst zu den Anforderungen: Qualität erschöpft sich sicher nicht in der Verwendung eines 19"-Racks oder in hoher Ausgangsleistung. Es ist die Güte der Regelung, die vor allem zählt. Damit dieses Schaltungskonzept trotz höheren Aufwands preislich im Rahmen bleibt, wurde die Ausgangsleistung so dimensioniert, dass sie für viele übliche Laboranwendungen ausreichend ist: maximal 30 V bei einem Strom von bis zu 1 A. Die 30 W an maximaler Ausgangsleistung erlauben zudem einen kleinen Trafo im Schaltregler, den man – soviel sei vorab schon verraten – ganz besonders leicht selbst herstellen kann.

12-V-Eingang

Die Schaltung in **Bild 2** ist ganz schön beeindruckend. Doch so „schlimm“ wie der voll gepackte Schaltplan aussieht, ist es auch wieder nicht. Links unten sieht man K3, einen ATX-2.2-Stromversorgungsstecker, wie er auf jedem PC-Mainboard zu finden ist. Der Hauptsteckverbinder eines ATX-Netzteils kann daher direkt eingesteckt werden. K3 ist so beschaltet, dass das PC-Netzteil startet, sobald es a) hier eingesteckt wird und b) selbst eingeschaltet ist. Der Netzschalter des PC-Netzteils ist damit auch der Netz-

schalter des gesamten Labornetzteils. Mit dessen 12-V-Schiene wird das Labornetzteil versorgt. Die beiden Flachstecker K1 und K2 dienen dazu, die 12-V-Leitung an weitere Labornetzteil-Module weiter zu reichen.

Secondary Voltages

Dieser Block von Bild 1 befindet sich in der rechten unteren Ecke von Bild 2: Hier wird die Versorgungsspannung mit ± 15 V für die Opamps etc. über die Gleichspannungskonverter DC1, DC2 und DC3 erzeugt. Für „+15V2“ sind DC1 und DC2 parallel geschaltet. Für „-15V2“ ist alleine DC3 zuständig. Diese Module sorgen gleichzeitig für eine Potentialtrennung. Da sie mit Schalttechnik arbeiten, werden ihre Störungen eingangsseitig durch die Drossel L2 und am Ausgang durch die Drosseln L3 und L4 ausgefiltert. Ohne diese Filter würden hochfrequente Störsignale bis in die Ausgangsspannung des Netzteils durchschlagen. Ganz wichtig zum Verständnis des Schaltplans ist, dass es zwei verschiedene Massesymbole gibt: Das gewöhnliche Symbol bezieht sich auf die 12-V-Eingangsseite und das Massesymbol mit dem kleinen Index „A“ auf die potentialgetrennte Ausgangsseite. Der Hochvolt-Kondensator C40 koppelt die beiden Massepotentiale HF-mäßig und beugt so Störungen vor.

Die 5-V-Versorgung ist doppelt ausgeführt: Das meiste der digitalen Elektronik samt Mikrocontroller IC9 und Display-Modul (**Bild 3**) wird mit „+5V2“ versorgt, die mit einem kleinen Linearregler (IC12) aus „+15V2“ abgeleitet sind. Getrennt davon sind auch „+5 V“ auf der Eingangsseite (bezogen auf die Eingangsfläche) erforderlich. Sie stammen entweder direkt vom 5-V-Zweig des PC-Netzteils oder aber von einem weiteren Spannungsregler (IC10) aus dessen 12-V-Zweig. Da manche PC-Netzteile ordentlich schwankende 5-V-Potentiale haben, wenn die 5-V-Schiene schwach, aber die 12-V-Schiene stark wechselnd belastet wird, ist es vorzuziehen, bei der Lötbrücke SJ1 die Pins 2 und 3 zu verbinden.

Switching Regulator

Dieser Block von Bild 1 ist über Bild 2 verstreut. Zunächst zur eigentlichen Spannungswandlung. Links oben im Schaltplan sieht man die massiv gepufferte 12-V-Eingangsseite. Die massive Parallelschaltung von fünf kleineren Elkos (C24...C26, C28 und C30) und sechs Vielschicht-Kondensatoren reduziert die Eingangsimpedanz HF-mäßig massiv und entkoppelt so das PC-Netzteil sehr

gut von den auftretenden Stromspitzen. Die gleiche Strategie wird auf der Sekundärseite verfolgt, denn solche Parallelschaltungen sind dynamisch besser als ein einzelner Kondensator mit der Gesamtkapazität.

Die vier Power-MOSFETs bilden eine Brückenschaltung, die vom „Schaltregler-IC“ LTC6992 (IC17) angesteuert wird. Zwischen diesem spannungsgesteuerten PWM-Generator und den Gates der FETs sitzen noch die beiden MOSFET-Treiber IC13 und IC14 (IRF2183), die genug Steuerstrom und eine eingebaute Totzeit bieten, damit beide FETs einer Halbbrücke nie gleichzeitig durchschalten können. Jeweils eine Schottky-Diode (D6 und D7) bildet zusammen mit Speicher-Kondensatoren (C31, C33, C34 und C35) eine Bootstrap-Schaltung, damit auch die Gates der „High-Side-FETs“ mit genügend hohen Spannungen angesteuert werden. Da IC17 nur ein PWM-Signal von 300 kHz liefert, wird ein Gegentakt mit dem Inverter IC5 erzeugt. Die beiden D-Flipflops von IC6 liefern nun abhängig vom Tastverhältnis verschobene Steuersignale mit 150 kHz. Bleibt noch die Frage, wie der Schaltregler seine

Ausgangsspannung so einstellt, dass sie immer gerade etwas über der Ausgangsspannung des analogen Reglers liegt. Hierzu wird für IC17 ein Steuersignal „MOD“ erzeugt, das vom Optokoppler OK1 stammt. Seine Sende-LED wird über R41 schlicht von der Differenz der Spannung des Schaltreglers und der Ausgangsspannung des Linearreglers angesteuert. Die Differenz muss daher immer etwa 1,5 bis 2 V sein, sonst steigen „MOD“ und damit die Ausgangsspannung des Schaltreglers an. Das Signal vom Schaltreglerausgang „VL_MESS“ stammt vom Spannungsteiler R42/R43 und das Signal „VOUTMESS2“ (Ausgang des Linearreglers) vom Spannungsteiler R32/R33. Die beiden Signale werden durch die Opamps IC8.A und IC8.B gepuffert und als „VL_IST“ bzw. „V_IST2“ an OK1 gelegt.

Analog Regulator

Die Endstufe des Linearreglers besteht aus dem PNP-Leistungstransistor T4, der von T3 in Emitter-schaltung gesteuert wird. Der Emitterstrom von T3, der auch gleichzeitig der Basisstrom von T4 ist, stammt von den Regelungen von Spannung

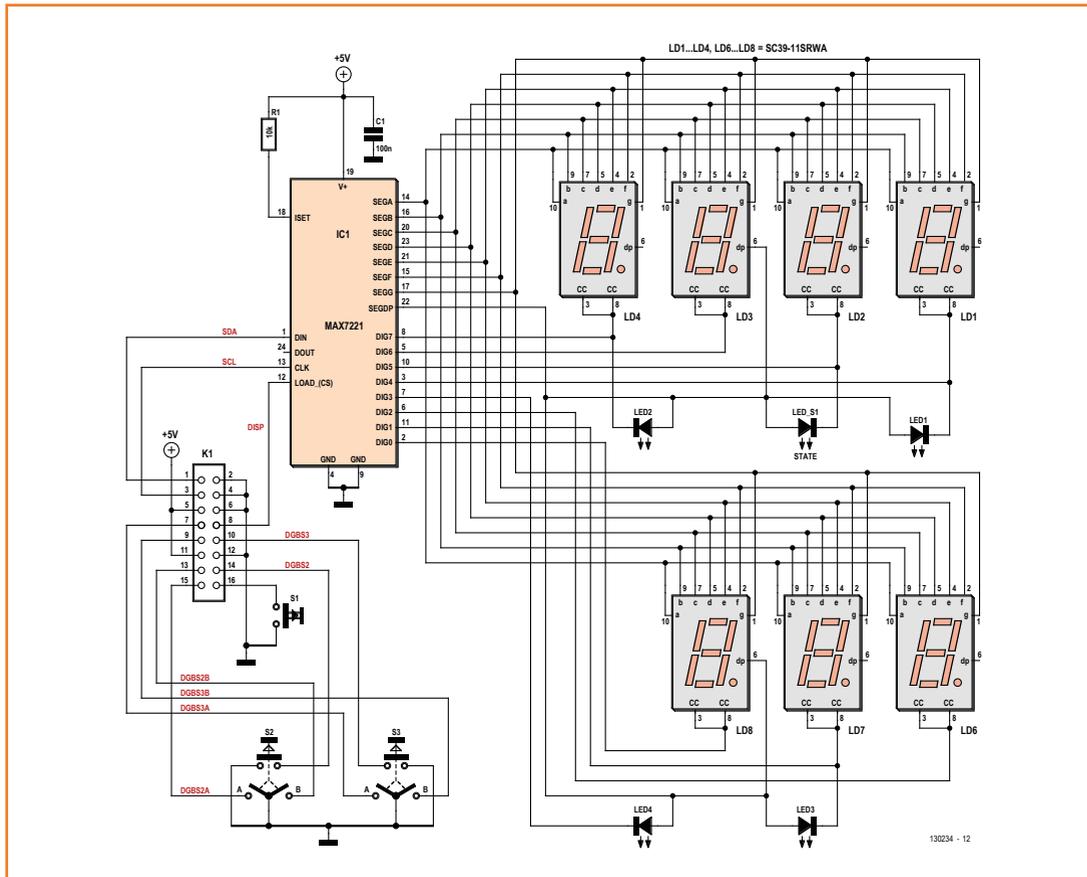


Bild 3. Die Schaltung des Bedienteils mit LED-Anzeige, LEDs und Drehencodern ist deutlich einfacher.

und Strom. Er fließt über P1, der beim Abgleich so eingestellt wird, dass T4 nur wenig in die Sättigung geraten kann, was das Regelverhalten verbessert. Die beiden Dioden D2 und D3 bilden zusammen mit R21 und R22 eine Vorrangschaltung für den Strom. Das funktioniert folgendermaßen: Solange der Ausgangsstrom unter dem eingestellten Grenzwert liegt, steuert die Spannungsregelung (IC1.A und IC3.A) den Basisstrom von T4 und damit die Ausgangsspannung, denn die Spannung am Ausgang der Stromregelung (IC1.B und IC3.B) ist „low“, weshalb D3 sperrt. Möchte der Ausgangsstrom bei entsprechend nie-

derohmiger Last aber zu hoch werden, zieht der Ausgang der Stromregelung auch den Ausgang der Spannungsregelung in Richtung positiverer Werte und die Ausgangsspannung sinkt auf den Wert, der den eingestellten Maximalstrom fließen lässt. Die Stromregelung hat Vorrang, weil R22 niederohmiger als R21 ist.

Bei der Spannungsregelung fungiert IC1.A als Differenzverstärker, der den Unterschied zwischen dem Signal „VOUTMESS“ von der Ausgangsspannung via R29, R30 und R31 mit dem Sollwert „V_SOLL“ vergleicht, der vom Doppel-DAC IC16 stammt. Ganz ähnlich liefert IC19 auch den Soll-

Stückliste Netzteil

Widerstände:

(wenn nicht anders angegeben: 0,1 W, 1 %, SMD 0603)

R1,R18,R48,R49,R56,R57 = 1 k
 R2 = 56 Ω, 5 %, 0,33 W, SMD 0805
 R3,R50,R58 = 3k3
 R4,R14,R17,R33,R43,R52,R53 = 10 k
 R5..R7,R9,R11,R12 = 27 k, 0,1%
 R8,R10 = 2k7, 0,1%
 R13,R15,R54,R59,R60 = 5k6
 R16,R23 = 100 Ω
 R19 = 560 k
 R20 = 1 M
 R21 = 330 Ω,
 R22,R41 = 270 Ω
 R25 = 12 k
 R26 = 2k4
 R27 = 3k9
 R28 = 100 k
 R29,R30 = 560 Ω, 1,5 W, SMD 2512
 R31 = 82 Ω, 0,25 W, SMD 1206
 R32,R42 = 22 k
 R34 = 15 k
 R35 = 0 Ω (Lötbrücke)
 R36 = nicht bestückt
 R37 = 56 k
 R38 = 39 k
 R39 = 160 k
 R40 = 6k8
 R44..R47 = 1k8, 1 W, 5 %, bedrahtet
 R51 = 0Ω33, 2 W, SMD 2512
 R55 = 4k7
 P1 = 2 k, Mehrgang-Trimmpoti, hoch (Farnell 9353755)

Kondensatoren:

(wenn nicht anders angegeben: 50 V, X7R, SMD 0603)
 C1..C6,C11,C13..C15,C19,C22,C31,C34,C37,C38,C41,C46,C51,C52,C54..C59,C61..C70,C74 = 100 n
 C7 = 680 p, NPO
 C8 = 82 p, NPO
 C9 = 4n7
 C10,C71 = 56 μ / 63 V, Elko, RM 3,5 mm, ø 8 mm
 C12,C24..C26,C28,C30,C42..

C44,C60,C72,C73 = 100 μ / 63 V, Elko, Low-ESR, RM 3,5 mm, ø 8 mm
 C17,C18,C33,C35,C36 = 1 μ, X5R
 C16,C20,C21,C23,C27,C32,C45,C47,C48 = 100 n, SMD 1206
 C29 = 1 n
 C39,C49 = 2μ2 / 50 V, X5R, SMD 0805
 C40 = 1 μ / 100 V, PET, RM 7,5 mm
 C50,C53 = 18 p
 C66 = 100 n / 630 V, SMD 1812

Spulen:

L2,L3,L4 = 47 μ / 0,9 A, Doppeldrossel (Farnell 1869658)
 L5 = 10 μ, SMD 0805 (Reichelt JCI 2012 10μ)

Halbleiter:

D1..D3,D11 = BAS40W Schottky, SMD SOT323 (Farnell 8734380)
 D4,D5 = B560C, Schottky, SMD SMC (Farnell 1858602)
 D6,D7 = PMEG4010ET, Schottky (Farnell 2311223)
 D9,D10,D12,D13 = PMEG6030EP, Schottky, SMD SOD128 (Farnell 1829207)
 D8 = BZV55-C2V4, Z-Diode, SMD Mini-MELF (Farnell 1097193)
 LED1, LED2, LED4..LED8 = LED, orange, SMD 0603
 OK1..OK3 = KB817-B, DMD4, Standard Optokoppler
 T1..T3 = BC817, SMD SOT23
 T4 = TTA1943, PNP, TO-3P (Farnell 1901958, alternativ 2SA1943)
 T5..T8 = PSMN015-60PS, n-Kanal-MOSFET, TO220 (Farnell 1845643)
 IC1,IC3 = TL5580, SMD SOIC8 (Farnell 1755396)
 IC2 = LM311D, SMD SOIC8 (Farnell 2293183)
 IC4 = INA196AIDBVT, SMD SOT23-5 (Farnell 1564942)
 IC5,IC11 = 74HC1G04GW, SMD SOT353 (Farnell 1085251)
 IC6 = 74HCT74D, SMD SOIC14 (Farnell 1085304)
 IC7 = 74HCT4538D, SMD SOIC16 (Farnell

1631658)

IC8 = TL074, SMD SOIC14 (Farnell 1459705)
 IC9 = ATmega32-A, SMD 44TQPF, programmiert erhältlich (EPS 130234-41)
 IC10,IC12 = 78L05, SMD SOIC8
 IC13,IC14 = IR2183SPBF, SMD SOIC8 (Farnell 1023247)
 IC15 = LT1461CCS8-4, 4,096 V, SMD SOIC8 (Farnell 1663430)
 IC16 = MCP4922-E/SL, SMD SOIC14 (Farnell 1332114)
 IC17 = LTC6992CS6-2, SMD TSOT23-6 (Farnell 1848046)
 DC1,DC3 = TMA 1215S, 12/15 V, 1 W, Traco (Farnell 1007521)
 DC2 = TMH 1215S, 12/15 V, 2 W, Traco (Farnell 1007560)

Außerdem:

X1 = Quarz, 16 Mhz, SMD HC49
 Kühlkörper für T4, Fischer SK 08, 3,2 K/W (Reichelt V 4511D)
 Kühlkörper für T5..T8, Fischer SK 125 84, 5,8 K/W (Farnell 4621335)
 K5 = 2x5-Wannenstecker
 K6 = 2x8-Wannenstecker
 K4, K7, K8 = 2x3-Wannenstecker
 K3 = 24-pol. ATX-2.2-Stecker für Platinenmontage (Farnell 2113352)
 K1,K2,K9,K10 = Flachstecker für Platinenmontage, 4,8 x 0,5 mm (Farnell 4215552)
 RE = Relais 12 V, 2 x um, (Reichelt FIN 40.52.9 12V)
 2 x Ferrit-Kerne E32/6/20-3F4 (Farnell 3056107)
 Kabelbinder zur Fixierung der Kerne
 Wärmeleitpad, keramisch, TO-3P, für T4 (RS 283-3830)
 Wärmeleitpad, keramisch, TO220, für T5..T8 (RS 177-7767)
 M3-Schrauben und Muttern für Kühlkörper und T4..T8
 19"-Gehäuse, 3 Höheneinheiten mit: Baugruppenträger 3HE, z.B. Hammond 84TE 235mm Typ F
 2 x Führungsschienen für 19"-Gehäuse
 Platine #130234-1 und -3

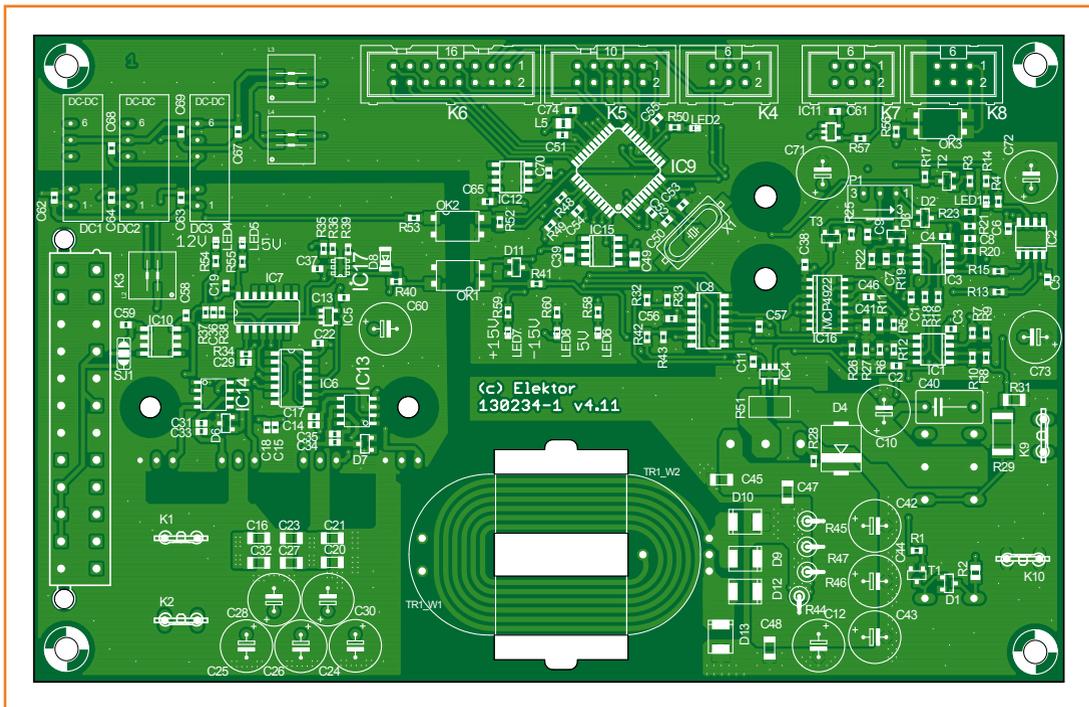


Bild 4. Bestückungsplan (85% der waren Größe) der Reglerplatine von Bild 2. Die Leistungstransistoren werden auf der Rückseite bestückt.

wert „I_SOLL“, der von IC1.B mit dem Signal „INA_OUT“ verglichen wird. Dieses Signal repräsentiert den fließenden Strom. Hierzu wird der Spannungsabfall über dem niederohmigen Shunt R51 vom darauf spezialisierten IC4 (INA196) in

eine auf Masse bezogene Spannung gewandelt. Irgendwo muss der DAC aber seine Daten her haben, damit er Sollwerte bereitstellen kann. Hier kommt endlich der Mikrocontroller IC9 ins Spiel. Er erhält Spannungs- und Stromsignale gepuf-

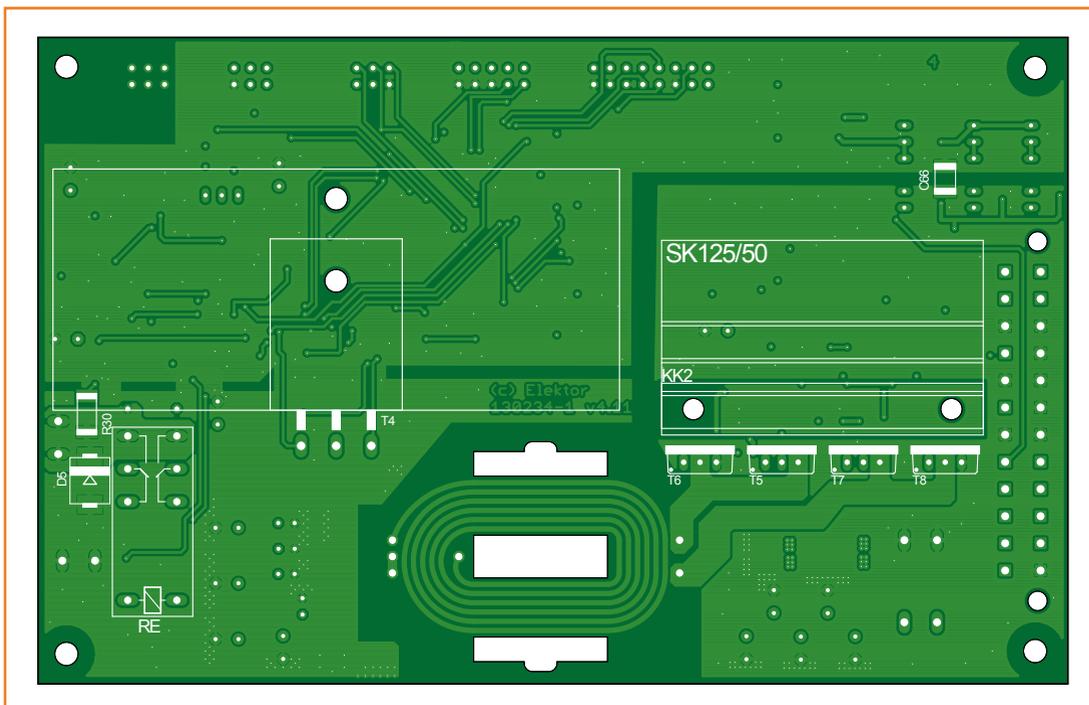


Bild 5. Die Rückseite der Reglerplatine (85% der waren Größe). Man sieht, wie die Kühlkörper mit der Platine verschraubt werden und wo die Leistungstransistoren sitzen.

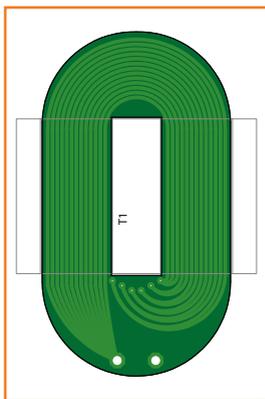


Bild 6.
So sieht die „Sekundärwicklung“ des speziellen Flachtrafos aus. Sie ist wie die Reglerplatine vierlagig ausgeführt.

fert über die beiden Opamps IC8.C und IC8.D als „V_IST“ und „I_IST“ an zwei interne Eingänge gelegt. IC9 misst die Spannungen und gibt die Werte auf dem Display aus. Daneben übergibt der Mikrocontroller dem DAC IC16 je nach Einstellung des Anwenders die passenden Soll-Daten. Oberhalb von K3 befindet sich IC15 (LT1461), ein präziser Spannungsregler, der beim angegebenen Typ eine Referenzspannung von 4,096 V mit einer Genauigkeit von nur 0,08 % bei einer Drift von nur 12 ppm/°C liefert. Man muss nicht zwingend diese Qualitätsklasse des ICs verwenden, denn die Genauigkeiten hängen stärker von der Präzision der Widerstände der Spannungsteiler ab, über welche die Spannung (R30/R31 und R32/R33) und der Strom (R26/R27 sowie R51) gemessen werden.

Display

Bild 3 zeigt die Schaltung des Bedienteils; es enthält die Anzeige nebst Treiber und LEDs sowie Bedienelemente. K1 ist mit K6 von Bild 2 verbunden. Das Treiber-IC MAX7221 wird seriell vom Mikrocontroller von Bild 2 mit Daten versorgt. Es steuert eine vierstellige Siebensegment-LED-Anzeige für die Spannung und eine dreistellige für den Strom in Multiplextechnik. Bei Spannung und Strom ist der Dezimalpunkt fix auf zwei Stellen nach dem Komma eingestellt. Da der Treiber so

etliche Dezimalpunkte nicht bedienen muss, werden stattdessen die vier LEDs LED1...LED4 sowie die in S1 eingebauten LEDs angesteuert. Die beiden rechten roten LEDs leuchten alternativ dann auf, wenn die Spannungsregelung (LED2) oder die Stromregelung aktiv ist (LED4). Die grünen LEDs links leuchten dann, wenn nach Druck auf S3 bzw. S2 gerade ein neuer Wert für Strom oder Spannung eingestellt wird. Der Taster S1 leuchtet, wenn die Ausgangsspannung anliegt.

Außerdem

Um mit S1 (Bild 3) die Ausgangsspannung ein und auch ganz schnell wieder ausschalten zu können, muss auch ein Relais vorhanden sein. In Bild 2 wird es von T1 angesteuert, welcher wiederum von einem digitalen Ausgang des Mikrocontrollers über R1 gesteuert wird. Die LEDs LED4...LED8 zeigen das Vorhandensein von Eingangs- und Hilfsspannungen an. D4 und D5 schützen die Elektronik vor den Folgen „rückwärts“ fließender Ströme durch Verpolungen, wie sie im Laboralltag z.B. bei Akkus durchaus vorkommen können. Die Regelgeschwindigkeiten werden von den RC-Gliedern R20||C8 (Spannung) und C7||R19 (Strom) beeinflusst. Der Komparator um IC2 schaut, ob eine Spannungs- oder Stromregelung vorliegt und informiert über T2 den Mikrocontroller. LED1 leuchtet, wenn die Stromregelung übernimmt.

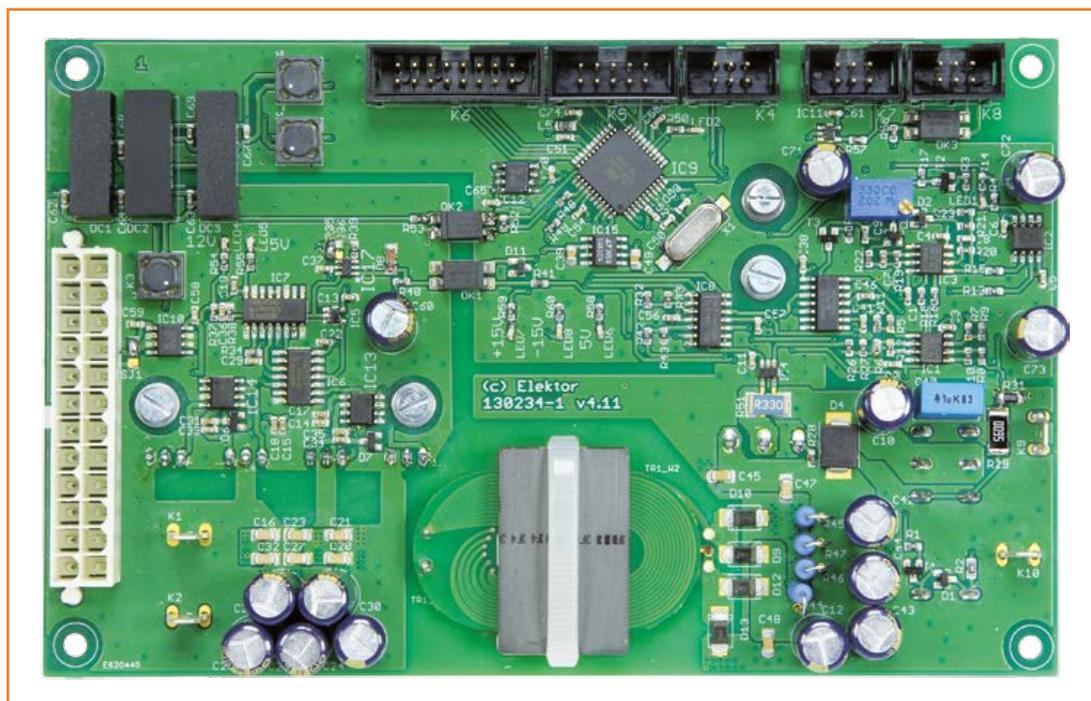


Bild 7.
Foto der fertig bestückten Reglerplatine. Man bekommt so schon einen Eindruck vom Flachtrafo.

Mit dem Monoflop IC7.A ist eine Einschaltverzögerung über den Reset-Pin von IC6 realisiert, damit der Schaltregler erst los legt, wenn sich alle Betriebsspannungen eingeschwungen haben. Mit K7 und K8 können serielle Daten über den USART des Mikrocontrollers übertragen werden. Auf diese Weise könnte man z.B. mehrere Netzteil-Module koppeln und etwa eine symmetrische Spannung realisieren. Der Eingang via K8 ist durch OK3 potentialgetrennt, sodass eine serielle Koppelung ohne galvanische Verbindung der Module möglich ist. Derzeit ist die Nutzung der Schnittstelle aber noch nicht in die Firmware integriert - man muss bei Bedarf selbst Hand anlegen. An K5 sind einige I/O-Pins des Mikrocontrollers herausgeführt, die man nach Erweiterung der Firmware für eigene Zwecke nutzen kann. Über K4 kann der Mikrocontroller mit üblichen In-System-Programmern programmiert werden. Der Trafo TR1 ist ein sogenannter Planartransformator, dessen Aufbau später beschrieben wird. Er hat ein Übersetzungsverhältnis von 2:3.

Aufbau

Zu den Schaltungen von Bild 2 und Bild 3 gibt es je eine eigene Platine. Der massive Einsatz von SMD-Bauteilen im Format 0603 ist ein Hinweis darauf, dass man nicht nur viel Lötferfahrung benötigt. Für Anfänger ist so ein komplexes Pro-

jekt sicher nichts. Zumindest sollte man sich da nicht ohne Hilfe rantrauen. Fast überflüssig daher zu erwähnen, dass man zuerst die flacheren und dann die höher aufragenden Bauteile bestückt. Bei der Reglerplatine von **Bild 4** gibt es drei Besonderheiten zu beachten. Zwei werden klar, wenn man sich die Rückseite (**Bild 5**) anschaut. T4 sowie T5...T8 werden auf der Rückseite bestückt. Dabei muss der Kühlkörper für T4 durch Befestigung mit isolierenden Abstandshülsen daran gehindert werden, einen Kurzschluss auf der Platine zu produzieren. Beim Kühlkörper für die FETs ist das einfacher, da er direkt mit kurzen M3-Schrauben auf die Platine geschraubt werden kann. Die FETs kann man anschrauben oder einfacher mit passenden Befestigungs-Clips fixieren. Alle fünf Transistoren müssen mit Wärmeleitpads vom Kühlkörper isoliert werden.

Das Ungewöhnlichste am ganzen Projekt ist sicherlich der Planartrafo. Er besteht aus zwei Ferrit-Halbschalen und das Clevere ist, dass man nichts wickeln muss. Die Primärwicklung ist nämlich schon in die Platine integriert. Für die Sekundärwicklung gibt es eine eigene kleine Platine (**Bild 6**), die über dicke Kupferdrähte mit der Basisplatine verbunden wird und auf diese mit einer Lage Kaptonband o.ä. aufgeklebt wird. Dieses darf aber nicht zwischen die Kernhälften geraten. Der „gedruckte“ Trafo nutzt zwar das Wick-

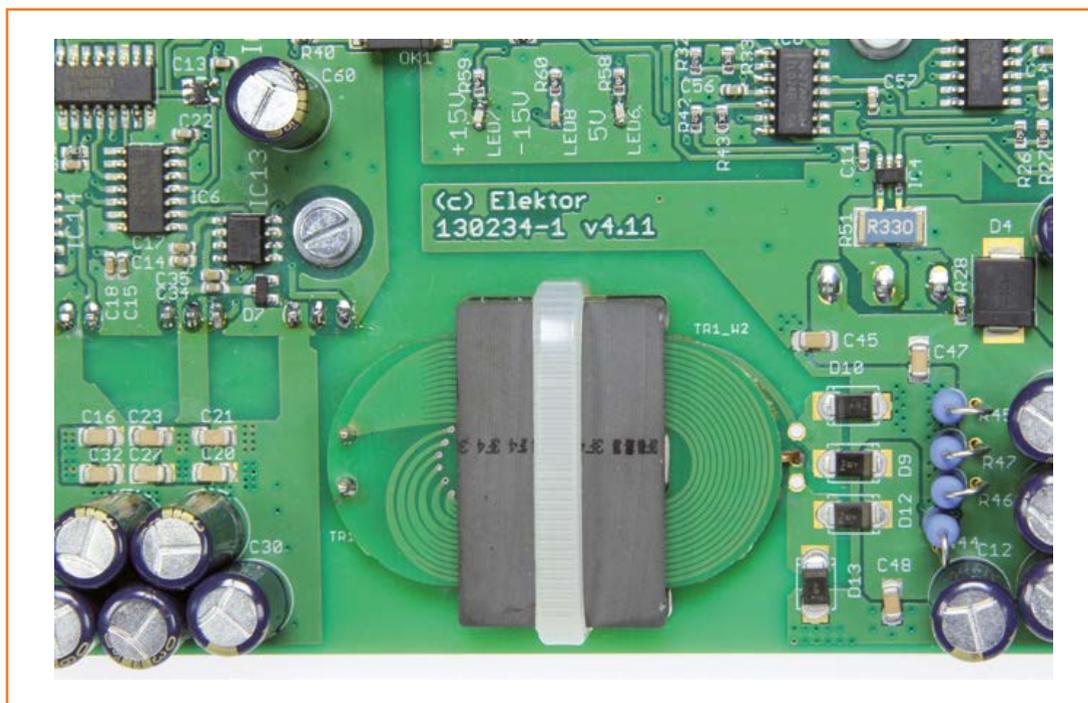


Bild 8.
Nahaufnahme des Flachtrafos. Hier kann man sehen, wie die kleine Platine mit der Sekundärwicklung „eingebaut“ wird.

lungsvolumen nicht voll aus, erspart aber eine Menge Handarbeit. Damit das überhaupt gut geht, sind beide Platinen vierlagig ausgeführt. Selbermachen der Platine ist daher nicht zu empfehlen. Die vier Lagen sorgen für vier parallele Windungen, was die negativen Wirkungen des Skin-Effekts bei hohen Frequenzen reduziert. Wenn die kleine Windungsplatine bestückt ist, werden die beiden Schalenhälften durch die Löcher in der Reglerplatine geführt und ein Kabelbinder fixiert dann die Konstruktion. Das Foto des Prototypen in **Bild 7** sowie die Nahaufnahme in **Bild 8** demonstrieren, wie das gemeint ist.

Die Platine des Bedienteils in **Bild 9** ist deutlich simpler gestrickt und auch nur „konventionell“ zweilagig. Besonderheiten gibt es nur wenige. Eine wäre, dass K1 auf der Rückseite der Platine bestückt wird und die zweite führt zur Frage, wozu die beiden runden Löcher gut sind. Vermutlich habe Sie es erraten: Es sind Aussparungen für 4-mm-Buchsen, die an der Frontplatte des Gehäuses befestigt werden. Man kann die Buchsen so mit etwas Litze und Hülsen gut mit K9 und K10 der Reglerplatine verbinden. **Bild 10** zeigt die fertige Kombination von Reglerplatine und Bedienteil.

Als Gehäuse für ein vernünftiges Labornetzgerät eignet sich ein 19“-Rack mit drei Höheneinheiten sehr gut. Da passt nicht nur das PC-Netzteil rein, sondern auch mehrere Labornetzteil-Module. Die Platinen sind explizit dafür gemacht.

Abgleich und Bedienung

Wenn die Platinen bestückt, kontrolliert und über ein Flachbandkabel miteinander verbunden sind, dann fehlt noch die Firmware auf dem Mikrocontroller. Diese wurde in C mit AVR-Studio geschrieben und ist mitsamt den Layout-Dateien der Platinen von der Elektor-Webseite zu diesem Artikel downloadbar. Zur Programmierung des Mikrocontrollers eignen sich ISP-Programmer wie z.B. Atmels AVRISP MKII und Varianten. Zur Programmierung muss der Controller natürlich „unter Strom stehen“ – die Platine muss also mit 12 V versorgt werden.

Nach der Programmierung sollte eine halbwegs vernünftige Anzeige auf den LED-Displays zu sehen sein. Wenn man einen der Drehencoder S3 oder S2 drückt, leuchtet die jeweilige grüne LED zum Zeichen, dass man die Spannung oder den Strom durch Drehen des Encoders einstellt. Ein Schritt des Encoders entspricht dabei 10 mV

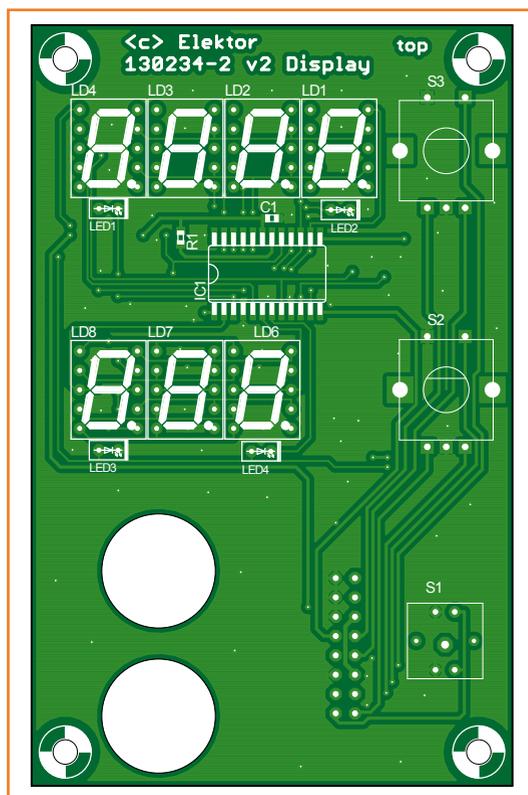


Bild 9.
Der Bestückungsplan der Platine des Bedienteils. Die Platine selbst ist konventionell zweilagig.

Stückliste Display

Widerstände:

R1 = 10 k, 0,1 W, 1 %, SMD 0603

Kondensatoren:

C1, = 100 n, 50 V, X7R, SMD 0603

Halbleiter:

LD1..LD4, LD6..LD8 = SC39-11GWA, LED-Display

rot, gem. Kath. (Farnell 2314233)

LED1,LED3 = LED grün, rechteckig, bedrahtet (Farnell 1142607)

LED2,LED4 = LED rot, rechteckig, bedrahtet (Farnell 1581150)

IC1 = MAX7221CWG+, SMD SOIC (Farnell 9725725)

Außerdem:

S1 = MULTICOMP MCSPHN3-YCA043T, Taster mit LED (Farnell 2146950)

S2,S3 = Drehencoder mit Taster (Mouser 652-PEC12R-4220F-S24)

Knöpfe für Drehencoder (Reichelt KNOPF 15/6 MATT)

Abdeckkappen für Knöpfe (Reichelt ABDECKKAPPE 15 SCHWARZ)

K1 = 2x8-Wannenstecker

2x 2x8-Buchsenleiste für Flachbandkabel

16-pol. Flachbandkabel

Platine # 130234-2

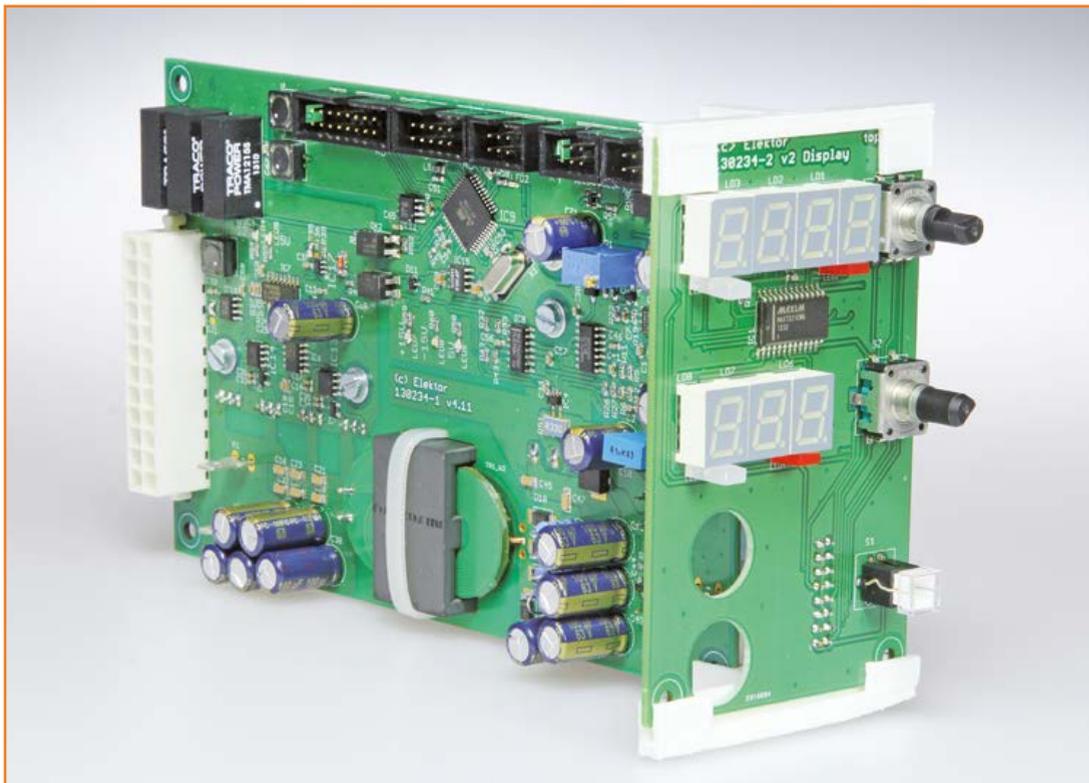


Bild 10.
Der fertige Prototyp
als Kombination von
Reglerplatine und Bedienteil.

bzw. 10 mA. Dreht man S3 während er gedrückt ist, ergeben sich 1-V-Schritte. Bei S2 entspricht ein Schritt bei gedrücktem Drehen 100 mA. Durch nochmaliges Drücken von S3 bzw. S2 verlässt man den Einstellmodus - die Werte werden übernommen und die grüne LED verlöscht. Jetzt werden wieder die aktuelle Spannung und der aktuelle Strom angezeigt.

In den Modus zur Kalibrierung gelangt man durch Betätigung von S1 und gleichzeitiges Drehen von S2. Jetzt muss man mit einem Amperemeter den Ausgang des Netzteils „belasten“, d.h. kurzschließen. Ideal ist ein Messbereich von 2 A. In diesem Modus wird die Ausgangsspannung auf 1 V eingestellt und der maximale Strom auf Maximum. Auf dem Display wird der fließende Strom und anstelle der Spannung ein Kompensationswert in % angezeigt. LED1 indiziert das Vorzeichen der Kompensation. Zunächst wird P1 etwa ein bis zwei Umdrehungen über den Punkt hinaus eingestellt, an dem der Strom nicht weiter ansteigt. Mit S3 kann man die Kompensation in Promille-Schritten ändern. Auf diese Weise wird die Stromanzeige auf den Wert des externen Amperemeters eingestellt. Verlassen wird der Modus wieder durch Betätigung von S1 plus Drehen an S2. Die Strommessung wird von Bauteiletoleran-

zen und dem minimalen Laststrom durch R29... R33 beeinträchtigt.

Einen Eindruck vom Regelverhalten des Labornetzteils gibt **Bild 11**, das einen Lastabfall von 90 % auf 10 % bei einer Spannung von 15 V zeigt.

(130234)

Weblink

[1] www.elektor-magazine.de/130234

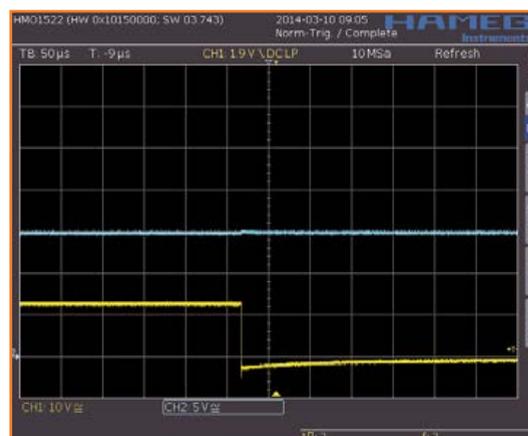


Bild 11.
Dieses Oszillogramm zeigt
kaum eine Antwort in
der Ausgangsspannung
(15 V), wenn die Last
abrupt von 90 % auf 10 %
heruntergefahren wird.

T-Boards 8/14/28

Schnelles und preisgünstiges Prototyping mit AVR-Controllern

Von **Andrew Retallack**
(Südafrika)



Die Arduino-Plattform ist ein perfekter Weg, um in die Welt der Mikrocontroller und Embedded-Systeme einzusteigen. Irgendwann aber möchten Sie wahrscheinlich auch Controller selber programmieren und debuggen. Dieser Schritt ist mit den T-Boards leicht gemacht.

Könnte man nicht fertige Arduino-Boards kaufen, würde die Gemeinschaft der Mikrocontroller-Anwender sicher nur ein Bruchteil der heutigen Community ausmachen. Irgendwann jedoch werden die Projekte ehrgeiziger und komplexer, ihre Anforderungen gehen über die Möglichkeiten der Arduino-Plattform hinaus und erfordern ein eigenes Platinendesign mit einem Mikrocontroller. Einige Projekte bedingen bestimmte Rahmenbedingungen, eine Flexibilität, die ein Arduino nicht bieten kann, zum Beispiel energieeffiziente Designs durch Betrieb mit niedrigeren Spannungen und/oder geringeren Taktraten. Für weniger komplexe Projekte ist ein kleiner AVR-Mikrocontroller von Atmel besser geeignet und kostengünstiger als ein kompletter Arduino. Was auch

immer der Grund ist, der erste Schritt auf dem Weg zum Prototypen ist es, den Mikrocontroller auf ein Steckbrett zu setzen.

Mikrocontroller auf dem Steckbrett

Auf den ersten Blick scheint es kein Problem, ein eigenes Mikrocontroller-Projekt auf einem Steckbrett zu beginnen. Es gibt viele Online-Ressourcen [1] und die dazu gehörenden Bauteile sind ziemlich preiswert. Doch wenn Sie tatsächlich mit dem Projekt beginnen, beginnen auch die Schwierigkeiten! Schon eine gewöhnliche Mikrocontroller-Basischaltung erfordert eine Reihe von Schaltdrähten, Jumpers, Kondensatoren, Widerständen und Anschlüssen. Leider ist auf einem Steckbrett immer (zu) wenig Platz, so dass

nicht nur die Anzahl der möglichen Verbindungen zum Controller beschränkt ist, sondern es auch schwierig ist, Verbindungen zu verfolgen und Fehler zu lokalisieren. Dazu kommt, dass alle Pins des Mikrocontrollers zugänglich sind, ob Sie das nun brauchen oder nicht.

Da die Anschlüsse nicht markiert sind, müssen Sie sie abzählen und abzählen und nochmals abzählen, wenn Sie sie mit einem Bauteil verbinden. Ein Fehler ist dennoch schnell passiert und der Controller „abgeschossen“. Und last not least: Ob Sie nun den Controller mit einem USB/Seriell-Konverter und Bootloader oder einem ISP-Programmer programmieren - auch dafür sind

Platz und zusätzliche Käbelchen auf dem Steckbrett erforderlich. Das Ergebnis ist ein schwer durchschaubarer Wust von Bauteilen und Leitungen, was nicht unbedingt erfreulich ist, wenn man eigentlich nur eine Mikrocontroller-Schaltung ausprobieren möchte.

Prototyping leichtgemacht: Das T-Board

Um ehrlich zu sein, das T-Board-Design ist aus dem Ärger des Autors über die Steckbrett-Experimente heraus entstanden. Das Ziel war ein Breakout-Board, das das Mikrocontroller-Prototyping entscheidend vereinfachen und beschleunigen würde. Er hat drei Versionen für die beliebtesten Atmel-AVR-Mikrocontrollerreihen, die 28-poligen ATmegas (ATmega8/48/88/168/328), die 14-poligen ATtinys (ATtiny20/24/44/84/441/841) und die 8-poligen-ATtinys (ATtiny13/25/45/85) entwickelt. Alle drei Versionen sind Steckbrett-kompatibel, können mit eigener Stromversorgung (3,3 V oder 5 V) betrieben werden und enthalten ICSP-Header für eine einfache Programmierung.

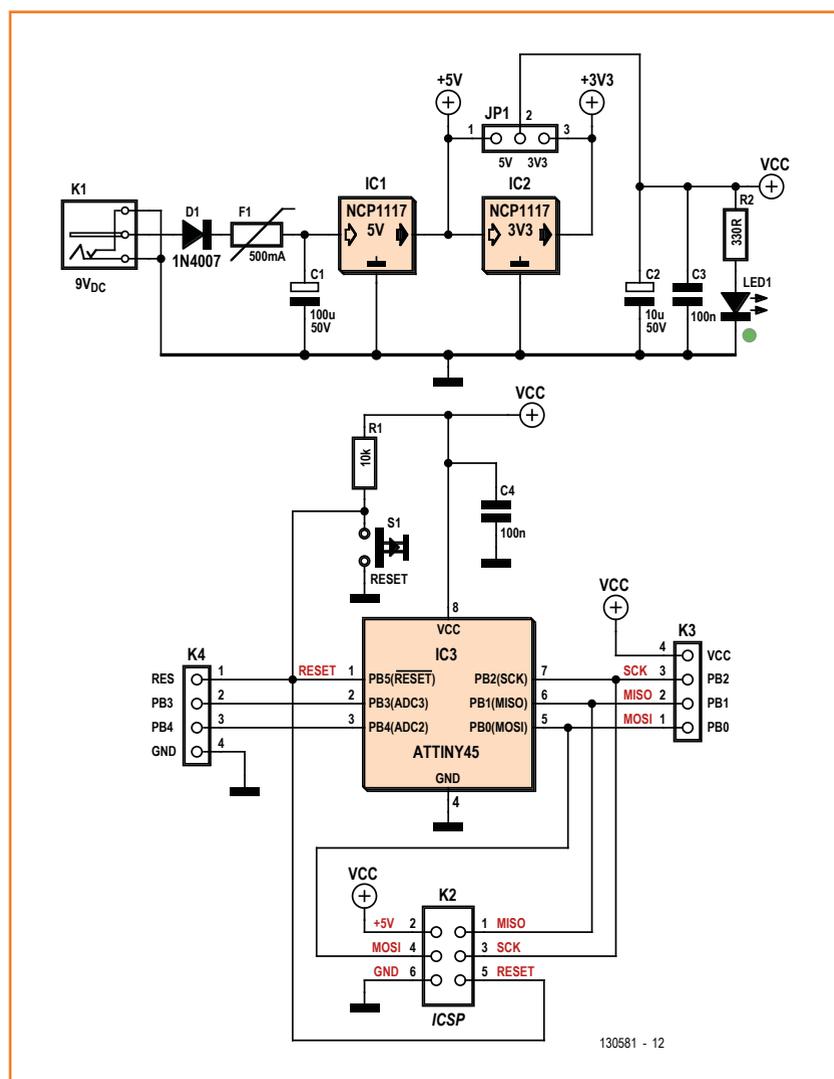
Die Elektrik

Da es sich um drei verschiedene Boards handelt, gibt es auch drei Schaltbilder: **Bild 1** zeigt das T-Board 8, **Bild 2** das T-Board 14 und **Bild 3** das T-Board 28. Abgesehen von den offensichtlichen Unterschieden der verwendeten Mikrocontroller sind die Schaltpläne weitgehend identisch. Die Stromversorgung erfolgt über die 2,1-mm-Klinenbuchse K1 (Plus in der Mitte), eine Schutzdiode (D1), die rückstellende PTC-Sicherung

F1 mit einem Auslösestrom von 0,5 A und den Puffereleko (C1). Die beiden Spannungsregler IC1 und IC2 sind Low-Dropout-Regler mit festen 5-V- (NCP1117DT50G) beziehungsweise 3,3-V- (LD1117S33TR) Ausgängen. Die maximale Eingangsspannung liegt bei 20 V und der Nennausgangsstrom bei über 1 A. Mit Jumper JP1 wählt man die Betriebsspannung VCC. Das T-Board 28 kann auch vom FTDI/Seriell-Wandler mit Strom versorgt werden, eine 5-V-Versorgung dort vorausgesetzt.

Die Mikrocontroller werden in DIL-Fassungen gesteckt und können so leicht ausgetauscht werden. Es gibt eine Reihe von Controllern mit kompatiblen Pin-Konfigurationen. Ein 100-nF-Entkopplungskondensator (C4) liegt dicht zwischen VCC und GND des Mikrocontrollers. Die sechspolige ICSP-Header-Verbindung ist Standard und

Bild 1. Schaltung des T-Board 8. Der Controller in seinem achtpoligen Gehäuse ist tatsächlich „tiny“.

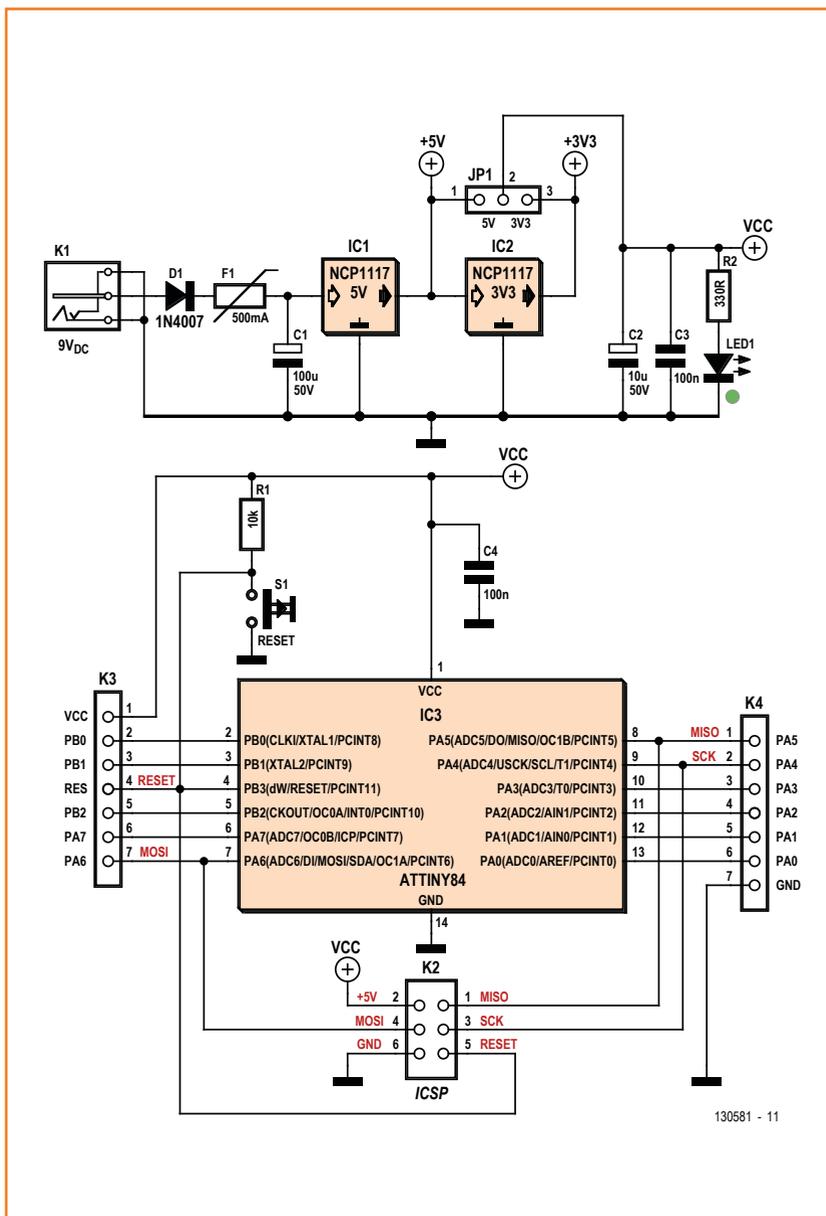


versorgt das Board während der Programmierung. Seien Sie sich bewusst, dass die Spannung dann vom ISP-Programmer bestimmt wird! Eine Reset-Taste S1, die den Reset-Pin auf Low zieht, rundet die allen drei Schaltungen gemeinsamen Elemente ab.

Drei Gleiche

Das Design der T-Boards ist relativ einfach, wie man in den **Bildern 4...6** sehen kann. Die auffällige T-Form garantiert ein Gleichgewicht von Flexibilität, Schlichtheit, Größe und Platzbedarf. Die gesamte Spannungsversorgung sowie der

Bild 2.
Schaltung des T-Board 14.
Die mittlere T-Board-Option basiert auf dem Mikrocontroller ATtiny84.



Anschluss für den USB/Seriell-Wandler sind nun vom Steckbrett entfernt, so dass der Großteil des Steckbretts zur freien Verfügung steht. Der Arbeitsbereich hätte sogar noch vergrößert werden können (je ein bis zwei Reihen neben dem Mittelkanal des Steckbretts sind verdeckt), allerdings hätte dies das Platinenlayout unnötig verkompliziert (der Mikrocontroller müsste „abgesetzt“ oder vertikal montiert werden) und/oder eine teure Multilayer-Platine erfordert. Auch mit einem SMD-Controller hätte man weitere Steckbrett-Fläche „freischaufeln“ können, allerdings dann auf die Möglichkeit verzichtet, zwischen

Wählen Sie Ihre IDE

Die Wahl einer IDE ist eine sehr persönliche Entscheidung, eine Empfehlung ist genauso schwierig wie die Wahl eines Betriebssystems Windows vs Mac vs Linux oder eine Debatte über die Vorzüge von Open Source vs kommerzielle Software. Im Endeffekt funktionieren alle IDEs, und wenn Sie ernsthaft in die Programmierung von Embedded-Systemen einsteigen, die hohe Geschwindigkeit/Speicher/Leistungsoptimierung erfordern, werden Sie zweifellos tiefer in die Materie einsteigen müssen. Hier aber ein paar Optionen für den Einstieg:

Arduino IDE

Ja, Sie können die Arduino-IDE verwenden [3], um direkt mit dem jungfräulichen Controller zu arbeiten. Sie sind auf die Controller beschränkt, die die Arduino-IDE unterstützt, aber Sie können Online-Ressourcen finden, um eine Reihe zusätzlicher Controller (wie die ATtinys) zu verwenden.

Vorteile: Arbeiten in einer vertrauten Umgebung; gute Bibliotheksunterstützung; einfach zu konfigurieren; Open Source; Windows/MacOS/Linux-Unterstützung.

Nachteile: begrenzte Flexibilität; begrenzte Controller-Unterstützung; Mangel an erweiterten Funktionen; kein Debugging.

Eclipse mit AVR-Plugin

Eclipse [4] ist eine sehr beliebte Open-Source-IDE, unterstützt viele Programmiersprachen und läuft unter Windows, MacOS und Linux. Bei der Flexibilität und der Vielzahl von Anwendungen kostet die Installation und Konfiguration einige Zeit; man benötigt auch ein Verständnis der Toolchains (Compiler, Linker, Assembler).

Vorteile: leistungsfähige und konfigurierbare IDE;

verschiedenen PDIP-Controllertypen zu wechseln. Im Endeffekt scheint das gewählte Design dann doch recht ausgewogen.

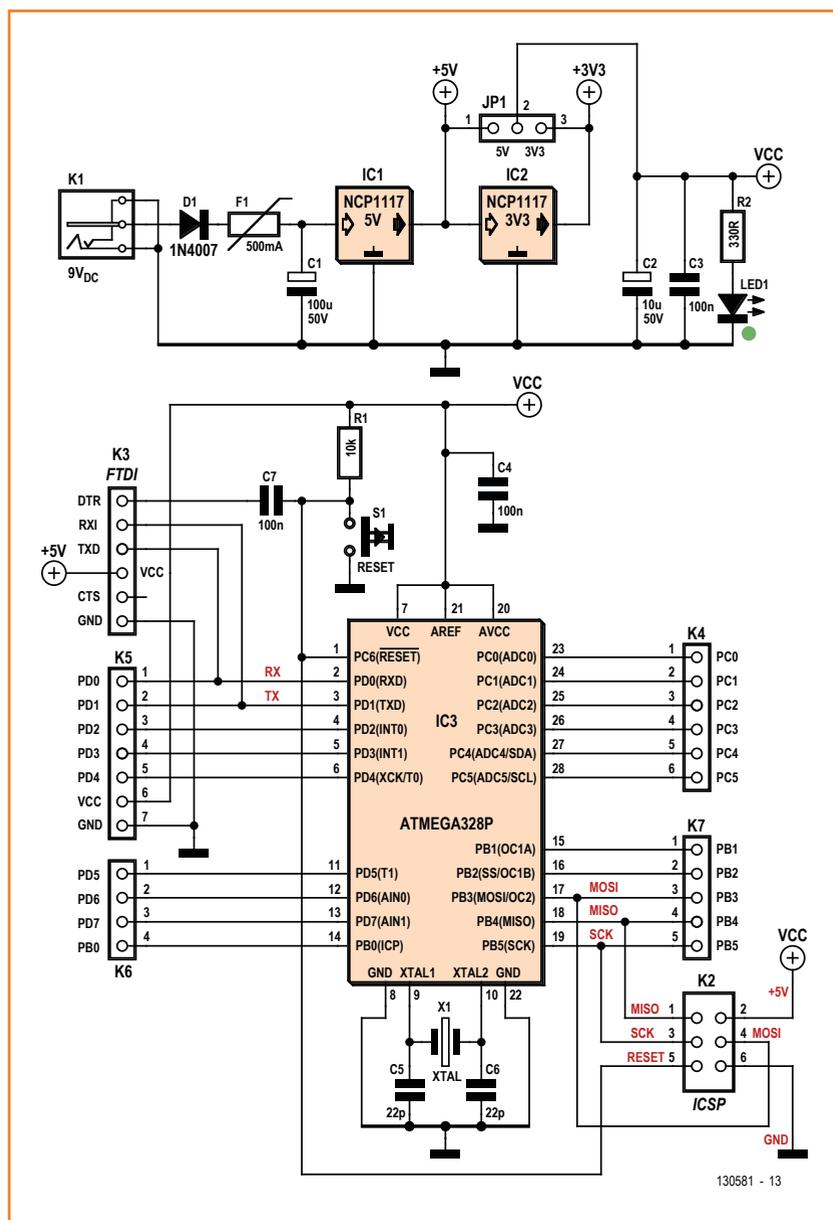
Die T-Boards sind fertig montiert zu recht angenehmen Preisen im Elektor-Shop erhältlich. Für diejenigen, die in guter alter Elektor-Tradition auf den Nachbau in Eigenleistung nicht verzichten wollen, gibt es die drei Bestückungsaufdrucke der Platinen samt Stücklisten in **Bild 7** sowie die Platinenlayouts auf der Projekt-Webseite unter [10]. Die gewählten Bauteile sind zum Teil normal bedrahtet und zum Teil SMDs (in der handhab-

baren Bauform 1206). So ist einfaches Löten mit dem HandlötKolben gut möglich; der Reflow-Ofen kann kalt bleiben.

Das T-Board 28

Das T-Board 28 mit seinem ATmega328 weist zwei Besonderheiten auf. Im Gegensatz zu ATtinys besitzen kompatible ATmega-Controller einen Hardware-UART und spezielle Tx/Rx-Pins. Deshalb verfügt auch nur das T-Board 28 über einen FTDI-Header, mit dem der Mikrocontroller auf dem Board beispielsweise über den Seriell-zu-USB-Wandler BOB [2] von Elektor seriell

Bild 3. Schaltung des T-Board 28. Das größte Board bietet den leistungsfähigsten Controller ATmega328.



Open Source; Windows/MacOS/Linux-Unterstützung; keine Code-Größenbeschränkungen; eine starke Community.
Nachteile: erfordert ein Plugin für Atmel-Controller; keine einfache Installation; unterstützt nativ kein Debugging.

Atmel Studio

Atmels eigenes Angebot [5] basiert auf der Visual Studio-Plattform. Es unterstützt natürlich alle Atmel-Controller einschließlich der ARM-Prozessoren. Wenn Sie nur mit Atmel-Controllern arbeiten möchten, ist dies ein guter Start.

Vorteile: Unterstützung für alle Atmel MCUs; erweiterte Funktionen einschließlich Debugging und Simulation; einfach zu installieren; keine Code-Größenbeschränkungen.

Nachteile: nur für Windows; manchmal Probleme mit der Visual Studio-Schnittstelle; Dokumentation könnte besser sein.

IAR Embedded Workbench [6]

Eine IDE für den professionellen Markt, mit Code-Größenbeschränkung kostenlos. Unterstützt neben Atmel AVR-Controller auch Mikrocontroller anderer großer Hersteller. Eine gute Lösung für alle, die mit verschiedenen Controllern in einem einzigen Tool arbeiten möchten.

Vorteile: Unterstützung für alle Atmel AVR-MCUs, erweiterte Funktionen einschließlich Debugging und Simulation, Profi-Tool für Controller mehrerer Hersteller.

Nachteile: kostenlose Version code-begrenzt; teuer; nur für Windows.

mit dem Terminalprogramm Ihres Computers kommunizieren kann. Man kann den Controller auch über diese Schnittstelle programmieren, vorausgesetzt, man nutzt einen Bootloader im Controller. Serielle Kommunikation könnte natürlich auch beim ATtiny in der Software implementiert werden, aber aufgrund der wenigen I/Os, die diese Controller besitzen, haben wir auf den FTDI-Verbinder bei den beiden kleinen T-Boards verzichtet.

Das T-Board 28 verfügt zusätzlich über eine 2-polige „Buchse“ für einen optionalen externen Quarz. In Übereinstimmung mit den Empfehlungen von Atmel sind 22-pF-Lastkondensato-

ren vorhanden. Aufgrund der begrenzten Anzahl von I/O-Pins und der Annahme, dass die meisten (zeitlich nicht kritischen) Anwendungen der kleinen Controller ohnehin keinen Quarz erfordern würden, wurde bei den kleinen Controllern auf die zusätzliche Präzision, die ein externer Quarz bietet, verzichtet.

Das 1-2-3 der T-Boards

Wie in der Arduino-Welt gibt es drei Schritte, um ein T-Board-Projekt zu erstellen. Doch jetzt, da wir uns der Arbeit mit einem „rohen“ Mikrocontroller verschrieben haben, gibt es ein wenig mehr Komplexität (und Flexibilität).

1. Der erste Schritt, die physikalische Gestaltung des Projekts, ist tatsächlich einfacher als mit dem Arduino. Stecken Sie das T-Board auf Ihr Steckbrett, klemmen Sie die Spannungsversorgung an und schon können Sie beginnen, Bauteile auf dem Steckbrett zu platzieren, ohne dass zusätzliche Schaltdrähte erforderlich wären.
2. Im zweiten Schritt wird der Code geschrieben, der den Controller steuert, so einfach oder komplex, wie Sie möchten. Sie können es entweder schnell mit der Arduino-Entwicklungsumgebung tun oder flexibler mit einer vollwertigen IDE wie Atmel Studio, Eclipse oder IAR Workbench (siehe **Kasten**: Wählen Sie Ihre IDE).
3. Der dritte Schritt ist es natürlich, den kompilierten Code auf den Controller zu übertragen. Beim Arduino geschieht dies gewöhnlich über den USB-Anschluss. Das T-Board bietet durch den ICSP-Anschluss die größte Flexibilität. Dies ist der einfachste Weg, um Ihr Programm auf den Controller zu flashen (siehe **Kasten**: Wählen Sie Ihren Programmer). Sie können auch den FTDI-Header auf dem T-Board 28 verwenden, aber der ISP-Programmer ist einfacher und schneller.

Bild 4.
Prototyp des T-Board 8.
Er kann sich leicht von der endgültigen Platine unterscheiden.



Bild 5.
Prototyp des T-Board 14.
Er kann sich leicht von der endgültigen Platine unterscheiden.

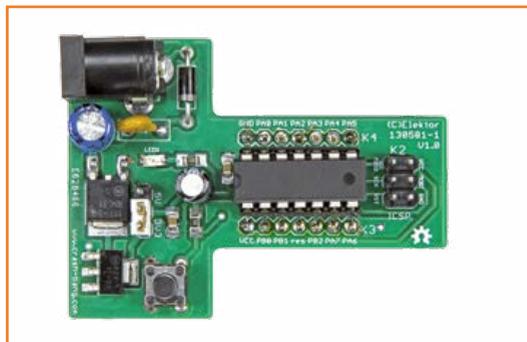
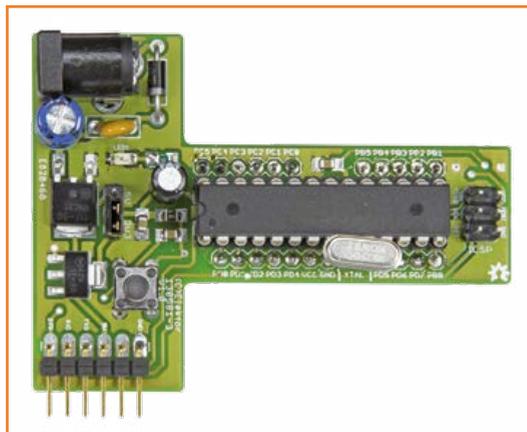


Bild 6.
Prototyp des T-Board 28.
Er kann sich leicht von der endgültigen Platine unterscheiden.



Ein kleines Projekt

Eine blinkende LED ist das „Hallo Welt“ der Embedded-Entwickler und für unsere Zwecke ein guter Anfang, um die Funktionsweise des T-Boards zu veranschaulichen. Für dieses Projekt wird das Atmel Studio benutzt, da es einfach von der Atmel-Website zu bekommen und schnell zu installieren ist. Im Prinzip funktioniert es aber mit anderen IDEs genau so. Wir folgen den drei oben erläuterten Schritten:



Die Zukunft ist unser Hemd!

Elektor.POST-Autor Jaime hat ein T-Shirt entworfen, mit dem er Arduino-Fans für die T-Boards begeistern möchte.

Ein T-Boards-T-Shirt erhalten Sie kostenlos, wenn Sie alle drei T-Boards im Elektor-Shop bestellen. Schauen Sie unter www.elektor.de und verfolgen Sie die Ankündigungen im Elektor.POST-Newsletter.



Schritt 1: Einrichtung des Steckbretts

- Stecken Sie das T-Board auf das Steckbrett;
- Stecken Sie den Spannungswahl-Jumper auf 5 V;
- Verbinden Sie mit einem Kabel den GND-Pin und die negative Versorgungsschiene des Steckbretts;
- Stecken Sie einen Widerstand zwischen einer leeren Zeile auf dem Steckbrett und (je nachdem, welches T-Board verwendet wird)
 - PB0 auf dem T-Board 28

- PA5 auf dem T-Board 14
- PB4 auf dem T-Board 8

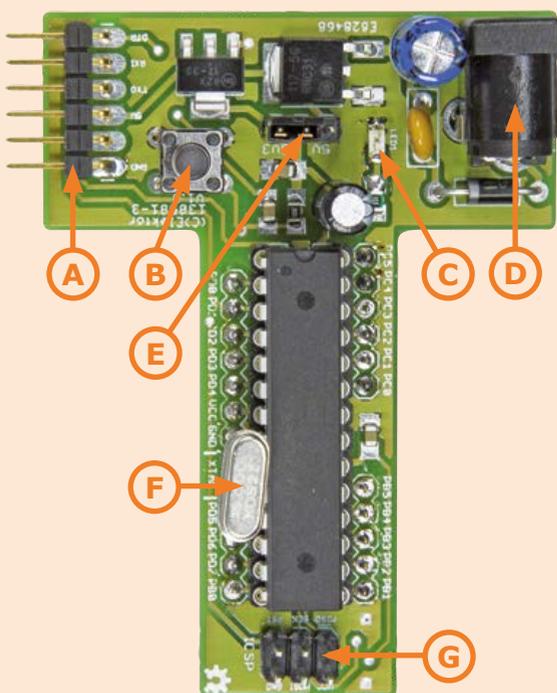
- Verbinden Sie die Anode der LED mit dem Widerstand und die Kathode mit der negativen Versorgungsschiene.

Schritt 2: Schreiben Sie das Programm

Erstellen Sie ein neues Projekt in Atmel Studio und stellen Sie sicher, dass es ein „GCC C Executable Project“ ist. Wählen Sie den richtigen Controller auf dem T-Board:

- T-Board 28: ATmega328

Physikalisches Layout und Funktion



Beispiel: T-Board 28

- A. FTDI-Anschluss:** Schließen Sie ein FTDI-Breakout-Board für die serielle Kommunikation über USB an (nur T-Board 28).
- B. Reset-Schalter**
- C. Power LED**
- D. Stromanschluss:** Standard-DC-Buchse für 2,1-mm-Pin (Plus auf Stift), max. 9 V.
- E. Jumper für Spannungsauswahl:** Ermöglicht es, den Mikrocontroller entweder mit 5 V oder mit 3,3 V zu betreiben.
- F. Quarz-Anschluss:** Gibt Ihnen die Möglichkeit, einen externen Quarz zu verwenden (nur T-Board 28).
- G. ICSP-Anschluss:** Schließen Sie einen ISP-Programmer an, um den Mikrocontroller zu programmieren.

Stückliste

T-Board 8

Nr. 130581-92 (ATTiny45), fertig bestückt erhältlich im Elektor-Shop

Widerstände:

R1 = 10 k, 250 mW, 1%
R2 = 330 Ω, 250 mW, 5%

Kondensatoren:

C1 = 100 μ, 50 V radial
C2 = 10 μ, 50 V radial
C3,C4 = 100 n, 50 V, 10%, X7R

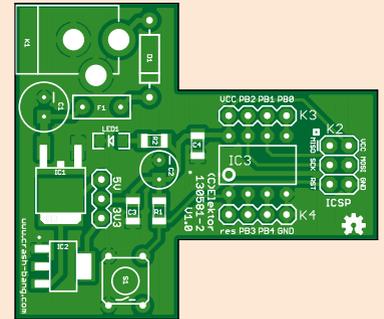
Halbleiter:

D1 = 1N4007
LED1 = SMD, grün, 20 mA
IC1 = 5 V/1A -Spannungsregler

NCP1117DT50G
IC2 = 3,3 V/1 A-Spannungsregler
NCP1117ST33T3G
IC3 = ATTiny45-20PU
(8-bit-Mikrocontroller)

Außerdem:

K1 = DC-Buchse, 2,1-mm-Pin
K2 = 2x3-polige Stiftleiste
K3,K4 = 4-polige Stiftleiste
S1 = Drucktaster, 24 V, 500 mA, 6x6 mm
F1 = Rückstellende PTC-Sicherung 500 mA
JP1 = 3-polige Stiftleiste
8-polige DIL-IC-Fassung
2-poliger Jumper, 2,54 mm



Platine 130581-2

Stückliste

T-Board 14

Nr. 130581-91 (ATTiny84), fertig bestückt erhältlich im Elektor-Shop

Widerstände:

R1 = 10 k, 250 mW, 1%
R2 = 330 Ω, 250 mW, 5%

Kondensatoren:

C1 = 100 μ, 50 V radial
C2 = 10 μ, 50 V radial
C3,C4 = 100 n, 50 V, 10%, X7R

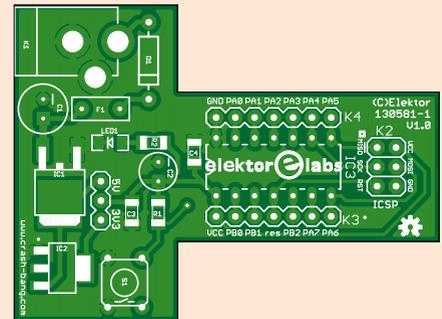
Halbleiter:

D1 = 1N4007
LED1 = SMD, grün, 20 mA
IC1 = 5 V/1A -Spannungsregler

NCP1117DT50G
IC2 = 3,3 V/1 A-Spannungsregler
NCP1117ST33T3G
IC3 = ATTiny84-20PU
(8-bit-Mikrocontroller)

Außerdem:

K1 = DC-Buchse, 2,1-mm-Pin
K2 = 2x3-polige Stiftleiste
K3,K4 = 7-polige Stiftleiste
S1 = Drucktaster, 24 V, 500 mA, 6x6 mm
F1 = Rückstellende PTC-Sicherung 500 mA
JP1 = 3-polige Stiftleiste
14-polige IC-Fassung
2-poliger Jumper, 2,54 mm



Platine 130581-1

Stückliste

T-Board 28

Nr. 130581-93 (ATmega328), fertig bestückt erhältlich im Elektor-Shop

Widerstände:

R1 = 10 k, 250 mW, 1%
R2 = 330 Ω, 250 mW, 5%

Kondensatoren:

C1 = 100 μ, 50 V radial
C2 = 10 μ, 50 V radial
C3,C4,C7 = 100 n, 50 V, 10%, X7R
C5,C6 = 22 p, 50 V, 1206

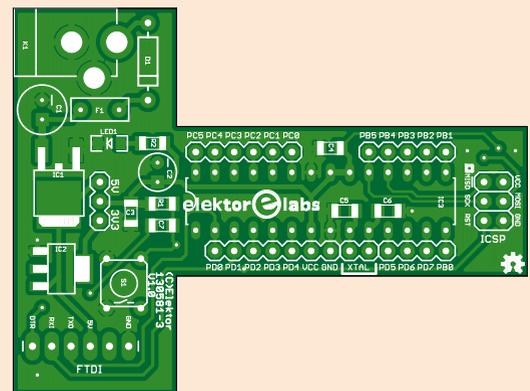
Halbleiter:

D1 = 1N4007
LED1 = SMD, grün, 20 mA
IC1 = 5 V/1A -Spannungsregler
NCP1117DT50G
IC2 = 3,3 V/1 A-Spannungsregler

NCP1117ST33T3G
IC3 = ATmega328P-PU
(8-bit-Mikrocontroller)

Außerdem:

K1 = DC-Buchse, 2,1-mm-Pin
K2 = 2x3-polige Stiftleiste
K3 = 6-polige Stiftleiste, gewinkelt
K4 = 6-polige Stiftleiste
K5 = 7-polige Stiftleiste
K6 = 4-polige Stiftleiste
K7 = 5-polige Stiftleiste
X1 = 2-polige Buchsenleiste für Quarz
S1 = Drucktaster, 24 V, 50 mA, 6x6 mm
F1 = Rückstellende PTV-Sicherung 500 mA
JP1 = 3-polige Stiftleiste
28-polige IC-Fassung 300 mil Breite
2-poliger Jumper, 2,54 mm



Platine 130581-3

Bild 7.

Bestückungsaufdrucke der drei T-Platinen. Alle drei Boards sind fertig bestückt im Elektor-Shop erhältlich (siehe [10]).

- T-Board 14: ATtiny84
- T-Board 8: ATtiny85

Dann überführen Sie den Code aus **Listing 1** ins Atmel Studio. Dieser Code ist bei weitem nicht optimal, aber einfach und so geeignet für unsere Zwecke.

Wenn Sie nicht das T-Board 28 verwenden, müssen Sie den (LED-Anschluss im) Code ein wenig anpassen (**Tabelle 1**).

Nach der Eingabe des Codes muss er kompiliert werden. Stellen Sie sicher, dass der Configuration-Manager auf „release“ eingestellt ist und drücken Sie F7, um die Kompilierung zu starten.

Schritt 3: Flashen Sie das Programm auf den T-Board-Controller

- Verbinden Sie den ISP-Programmer, den Sie verwenden möchten, mit dem T-Board und dem PC.
- Wählen Sie diesen ISP-Programmer im Menü

Tabelle 1. Änderungen am Programmcode des T-Board 28 für den Einsatz im T-Board 8 und T-Board 14.

T-Board 28	T-Board 14	T-Board 8
DDRB	DDRA	DDRB
DDB0	DDA5	DDB4
PORTB	PORTA	PORTB
PORTB0	PORTA5	PORTB4

Project und dann ...properties

- Auf dem Tab **Tool** wählen Sie den verwendeten **debugger/programmer**. Laden Sie das Programm auf das T-Board hoch: Klicken Sie auf das **debug**-Menü, dann **Start without Debugging**.
- Die LED sollte blinken.
- Wenn Sie möchten, können Sie nun das T-Board von dem ISP-Programmer trennen und es mit einer 9-V-Batterie im Stand-alone-Modus betreiben.

Listing 1. LED-Blinker mit T-Boards.

```

/*
 * T_Board_Blink.c
 *
 * Created: 24/05/2014 11:54:26
 * Author: Andrew Retallack, Crash-Bang Prototyping
 *
 */

#define F_CPU 16000000UL //We are running at 16MHz. Used to time the delay

#include <avr/io.h>
#include <util/delay.h>

int main(void)
{
    //Configure the LED port
    DDRB |= (1<<DDB0); //Set Pin PB0 as an output pin

    while(1)
    {
        PORTB |= (1<<PORTB0); //Turn the LED on, by making PB0 high
        _delay_ms(1000); //Delay 1 second
        PORTB &= ~(1<<PORTB0); //Turn the LED off, by making PB0 low
        _delay_ms(1000); //Delay 1 second
    }
}

```



Der Autor

Als Kenner der Software-Entwicklung entdeckte Andrew Retallack vor ein paar Jahren, dass es Spaß macht, auch mit der physischen Welt per Programm zu interagieren. Er ist begeistert von der Idee, seine Lernerfahrungen mit anderen Menschen zu teilen, die nicht über eine formale Elektrotechnik-Ausbildung verfügen und konzentriert sich dabei auf AVR- und MSP430-basierte Sensorik, HF- und Automatisierungsprojekte.

Immer weiter...

Die T-Board-Familie bietet also eine größere Flexibilität als ein Arduino. Von besonderem Interesse für den Autor war die Möglichkeit, einfach und relativ kostengünstig den Energiebedarf von Projekten zu optimieren.

In zukünftigen Artikeln werden Möglichkeiten diskutiert, den Stromverbrauch eines Projektes zu messen und ihn dann durch Veränderung der Eingangsspannung, den Einsatz von Low-Power-Modi und Interrupts und das Ändern des Prozessortakts zu reduzieren.

(130581)

Wählen Sie Ihren Programmer

Ein In-System-Programmer ist ein Hardware-Werkzeug, mit dem Sie kompilierten Code in den Mikrocontroller flashen (laden) können. Es gibt eine breite Palette von Internet-Anbietern, aber hier sind drei Optionen für „semiprofessionellen“ Gebrauch. Wenn Sie noch andere interessante Programmer kennen, lassen Sie es uns wissen.

Arduino als ISP-Programmer

Ein Arduino kann auch als ISP fungieren. Verbinden Sie einfach vier der Arduino-Pins mit dem ICSP-Header auf dem T-Board sowie VCC und Masse. Es gibt online ein ausführliches Tutorial [7], das Sie Schritt für Schritt durch den Prozess führt. Es gibt, wenn Sie einen Arduino besitzen, selbstredend keine zusätzlichen Kosten. Der Nachteil bei dieser Lösung ist, dass man das T-Board jedes Mal verdrahten muss, wenn Sie programmieren möchten. Sie können aber ein geeignetes Kabel mit ICSP-Header konfektionieren, um den Anschluss zu erleichtern.

USBtinyISP

Dieses großartige Low-Cost Open-Source-Projekt verwendet tatsächlich einen ATtiny als Programmer. Sie können es leicht zusammenbauen (es ist auch als Kit erhältlich). Der einzige Nachteil ist, dass es nicht direkt vom Atmel-Studio unterstützt wird, weil es die AVRdude-Software nutzt, um das Flashen des Controllers zu steuern. Es erfordert ein paar zusätzliche Schritte [8], um damit arbeiten zu können. Die Funktionalität ist etwas eingeschränkt (etwa die Einstellung der Fuses), aber dieser Weg lohnt sich dennoch der geringen Kosten wegen.

AVRISP mkII

Atmels eigener ISP, der AVRISP mkII [9], ist das teuerste der hier aufgelisteten Tools, die Kosten von etwa 40 € halten sich aber in Grenzen. Der Vorteil dieses Tools ist, dass es nativ vom Atmel-Studio unterstützt wird. Es erfordert allerdings eine zusätzliche Konfiguration, um mit anderen IDEs zu arbeiten, und scheint außerdem nicht von der IAR Embedded Workbench unterstützt zu werden.

Weblinks

- | | |
|--|--|
| [1] www.crash-bang.com/resource/breadboard-arduino/ | [6] www.iar.com/Products/IAR-Embedded-Workbench/AVR/ |
| [2] www.elektor.de/ft232r-usb-serial-bridge-bob-110553-91 | [7] www.crash-bang.com/resource/bootload-atmega328/ |
| [3] http://arduino.cc/en/Main/Software | [8] www.crash-bang.com/using-usbtiny-with-atmelstudio/ |
| [4] www.eclipse.org/ | [9] www.atmel.com/tools/avrismkii.aspx |
| [5] www.atmel.com/Microsite/atmel_studio6/ | [10] www.elektor-magazine.de/130581 |

The new EAGLE has landed!

Version 7

ab sofort erhältlich



Mehr Informationen auf www.cadsoft.de



Einsteigerseminar: MSP430-Mikrocontroller

Am 18. Oktober bietet der Elektor-Verlag in Aachen ein Einsteigerseminar zur Mikrocontroller-Serie MSP430 von Texas Instruments an. Durch die geringe Leistungsaufnahme ist die MSP430er-Serie vor allem für alle Anwendungen im Bereich Personal Healthcare und Wearable Sensors bestens geeignet. Das Seminar richtet sich an Einsteiger der Mikrocontroller-Programmierung, die bereits Programmiererfahrung beispielsweise aus dem PC-Umfeld mitbringen, aber noch keine Erfahrung in hardware-naher Programmierung haben. Durch eine Reihe von Hands-On-Übungen am Mikrocontroller wird im Laufe des Seminars schrittweise ein Mikrocontroller-basierter optischer Herzratensensor entwickelt.

Veranstaltungsort/-termin: Aachen, 18. Oktober 2014

Referenten: Marcus Köny und Thomas Schlebusch

Teilnahmegebühr: 399,00 Euro (inkl. MwSt.)

3-tägiges Seminar: Embedded Linux in Theorie und Praxis – ein Crashkurs

Ziel des Kurses ist es, Ihnen grundlegende Embedded-Linux-Konzepte sowie die Handhabung von Linux zu vermitteln. Was sind z. B. Vor- und Nachteile? Sie werden Ihren eigenen Bootloader und Kernel cross-kompilieren, diverse Programme auf einem PC erstellen/cross-kompilieren und auf einem eingebetteten System ausführen und debuggen. Eine Kombination aus Theorie und praktischen Übungen wird es Ihnen ermöglichen, das neu erworbene Wissen bei Eigenentwicklungen einzusetzen. Nach dem Kurs sind Sie wahrscheinlich noch kein Embedded-Linux-Experte, aber hoffentlich in der Lage sein, sich selbstständig zurechtzufinden.

Veranstaltungsort/-termin: Hanau, 21. bis 23. Oktober 2014

Referent: Robert Berger

Teilnahmegebühr: 1.899,00 Euro (inkl. MwSt.)

Im Preis sind Mittagessen, Seminarunterlagen, Dokumentation und Teilnahmezertifikat inbegriffen.

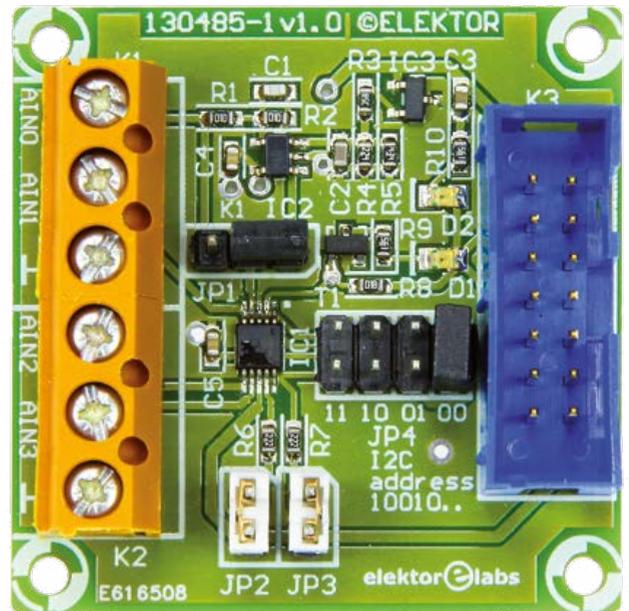
Weitere Infos & Anmeldung: www.elektor.de/events-de

Elektor-Abomitglieder erhalten 5% Rabatt auf den Seminar-Preis!

Daten loggen mit 16 bit ADC-Modul für Arduino, Linux-Board und mehr

Von **Jens Nickel**

Für hochauflösende und genaue Messungen bieten sich spezielle AD-Wandler-Chips an. Im Elektor-Labor haben wir eine Platine entwickelt, die einen vierkanaligen 16-bit-ADC enthält. Dank des Gnublin/EEC-Steckverbinders lässt sich das Modul unmittelbar an das Elektor-Linux-Board, das Xmega-Webserver-Board und das neue Elektor Extension-Shield für Arduino anschließen. Eine universell verwendbare C-Bibliothek macht die Ansteuerung des AD-Wandlers einfach.



Wieder einmal können wir von einem Fall berichten, bei dem sich unsere Projekt-Plattform Elektor.Labs [1] als Generator für eine gute Idee erwies. Unser langjähriger Autor Kurt Diedrich wollte seine Elektronik-Kenntnisse in den Bereich der Mikrocontroller-Programmierung ausdehnen und begann, mit einem Arduino Uno zu experimentieren. Wer Kurt kennt, kann sich denken, dass sich das Ganze recht schnell zu mehreren handfesten Projekten auswuchs, und ich riet ihm, seine Ergebnisse auf Elektor.Labs zu veröffentlichen. Eines der Projekte war ein Arduino-Datenlogger, bei dem der Uno Daten aufnahm und an einen PC leitete [2]. Hier kam Kurt seine Erfahrung in der PC-Programmierung und beim Entwurf von grafischen Oberflächen zugute: Als bald hatte er mit der Programmiersprache Processing ein Tool entwickelt, das die gesampelten Signale nicht nur darstellen, sondern auch spektral analysieren konnte. Das Ganze war ideal geeignet, um mit einem ebenfalls selbstgebauten Empfänger niederfrequente elektromagnetische Wellen

zu erfassen (siehe Artikel „ELF-Empfänger“ in diesem Heft).

ELF-Anwendung

Auf Elektor.Labs bekommt man einen guten Einblick in das Original-Projekt [2]: Die mit einer großen Spule aufgefangenen Signale wurden spannungsverstärkt, gefiltert und mit einem Offset von 2,5 V beaufschlagt, so dass sie sich im Bereich von 0 bis 5 V bewegten. Der interne ADC des ATmega328 (der Controller des Arduino Uno) besitzt nur eine Auflösung von 10 bit, deshalb erstand Kurt Diedrich bei Adafruit eine kleine Platine, auf die ein ADS1115 von TI aufgelötet war. Dieser ADC verfügt über vier analoge Eingänge und kann Signale mit bis zu 16 bit erfassen [3]. Von einem Mikrocontroller aus muss man den Chip per I2C ansteuern. Als cleverer Einsteiger nutzte Kurt die von Adafruit mitgelieferte Library sowie ein ebenfalls mitgeliefertes Programm-Beispiel für Arduino. Nach ein paar kleinen Abwandlungen war die Firmware

Bild 1.
16-bit-Datenerfassung mit Arduino: Das ADC-Modul wird über den Gnublin/EEC-Steckverbinder an das Elektor Extension-Shield auf einem Uno-Board angeschlossen.

des Datenloggers schon fertig, denn sie ist nicht sehr komplex. In einer Endlosschleife ruft das Hauptprogramm immer wieder die Funktion `ads.getLastConversionResults()` der Library auf, worauf der ADS1115 einen Messwert zurückgibt. Dieser wird über die serielle Schnittstelle des Arduino zum PC gesendet. Ein Delay oder gar eine Timer-Steuerung ist nicht vorgesehen, die Messfrequenz wird im Wesentlichen durch die Samplerate des ADCs bestimmt. Es stellen sich rund 112 Hz ein, was Kurt Diedrich in seiner PC-Software berücksichtigt.

Konzept

Das Ganze funktionierte recht gut, ein wenig un schön war lediglich die fliegende Verdrahtung zwischen der ADC-Platine und dem Arduino-Board. Da wir das Projekt sehr interessant fanden, dachten wir über Verbesserungen nach. Die Idee eines speziellen ADS1115-Shields für den Arduino verwarfen wir zugunsten einer noch flexibleren Lösung, nämlich eines vielseitig verwendbaren ADC-Moduls [4].

Der ADS1115-Chip sollte vom Controllerboard aus spannungsversorgt und über I2C angesteuert werden. Hier bot sich in idealer Weise der 14-polige Gnublin/EEC-Steckverbinder an. Damit konnte man unser ADC-Modul per Flachkabel an das Xmega-Webserver-Board [6] und an das im letzten Heft vorgestellte Elektor Extension-Shield [7] anschließen (**Bild 1**).

Für die vier analogen Eingänge haben wir Schraubklemmen vorgesehen. Da der Chip keine Spannungen unterhalb von 0 V messen kann, wurde zur Erfassung von Wechselspannungssignalen noch ein Offset von rund 1 V integriert, den man für den ersten Eingang zuschalten kann.

Spannungen

Der Schaltplan ist in **Bild 2** abgebildet. Der Gnublin/EEC-Steckverbinder spezifiziert eine 3,3-V-Spannungsversorgung, prinzipiell lässt

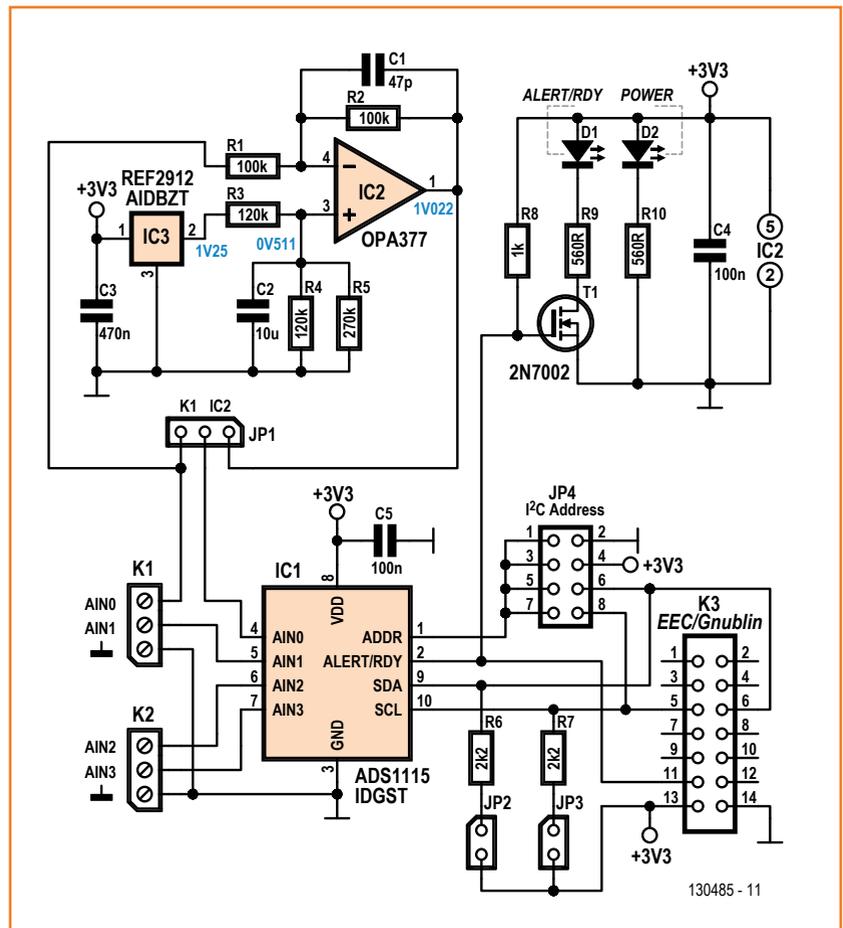
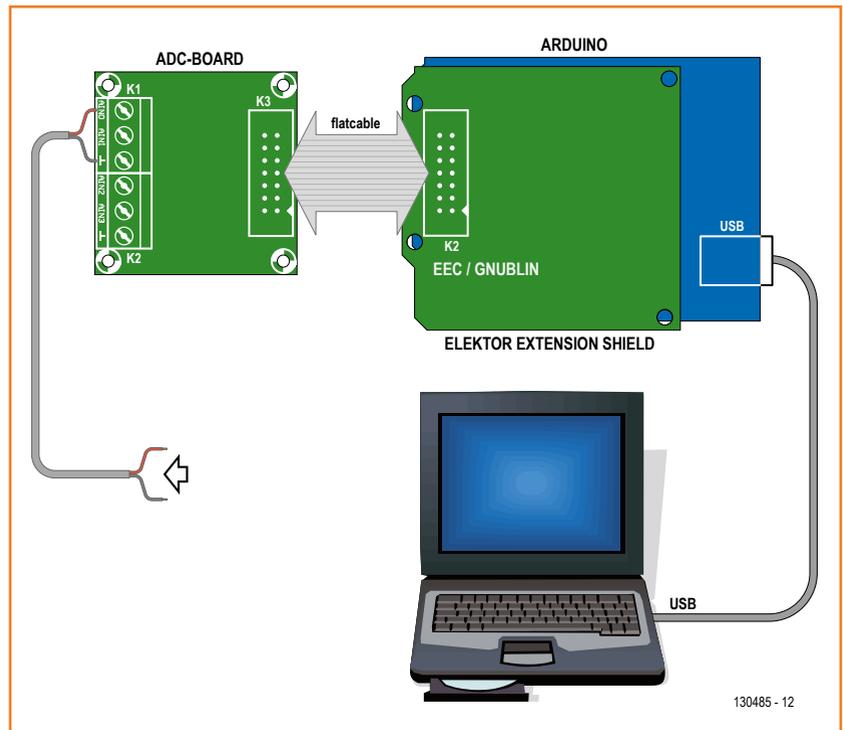
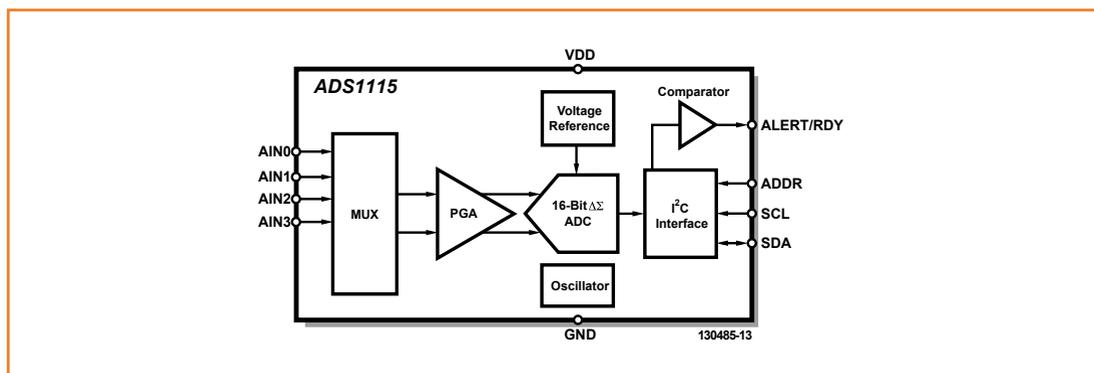


Bild 2. Schaltplan rund um den 16-bit-ADC ADS1115.

Bild 3.
Der ADS1115 enthält einige Schmauerl wie zum Beispiel den integrierten Komparator.



sich der ADC-Chip aber mit 2,0 bis 5,5 V speisen. Wie im Datenblatt [3] nachzulesen ist, kann der Baustein über I2C-Kommandos flexibel konfiguriert werden. Es lassen sich messen:

Die Spannungen von AIN0, AIN1, AIN2 und AIN3 relativ zu Masse (vier „Single-ended“-Eingänge). Die Spannung von AIN0 relativ zu AIN1 und die Spannung von AIN2 relativ zu AIN3 (zwei differentielle Eingänge).

Die Spannung von AIN0 relativ zu AIN3 und von AIN1 relativ zu AIN3 (zwei differentielle Eingänge mit gemeinsamer Basis).

Der Chip gibt immer digitale Werte von -32768 bis 32767 aus, die negativen Werte entsprechen negativen differentiellen Spannungen (negative Werte werden im Zweierkomplementformat dargestellt). Im Single-Ended-Modus ist die Spannung immer positiv, dadurch werden nur positive Werte von 0..32767 ausgegeben, was einer Auflösung von 15 bit entspricht. Mit einem Trick, den Tom Giesberts aus unserem Labor auf der .Labs-Website [8] beschreibt, lässt sich der negative Werte-Bereich und damit das 16. Bit nutzen: Man legt an AIN3 eine feste positive Spannung an, beispielsweise 1 V. Dann führt eine Spannung von 0..1 V an den Eingängen AIN0 und AIN1 zu negativen Ausgangswerten, wenn man den entsprechenden differentiellen Modus wählt.

Neben einer internen Spannungsreferenz ist im ADC auch ein programmierbarer Verstärker eingebaut, hiermit kann man den Spannungsbereich einstellen, in dem der ADC messen soll (Full Scale). In unserem Fall (3,3 V Betriebsspannung) sind die Einstellungen +/- 2,048 V, +/- 1,024 V, +/- 512 mV und +/- 256 mV sinnvoll.

Auf keinen Fall dürfen bei unserem Board an einem der Eingänge Spannungen von über 3,6 V oder von weniger als -0,3 V anliegen. Im Chip integrierte ESD-Dioden bieten einen gewissen Schutz, der Hersteller empfiehlt aber im Daten-

blatt, die Eingänge zusätzlich mit Schottky-Dioden und Widerständen in Serie zu schützen.

Über den Jumper JP1 ist ein Offset von 1,022 V zuschaltbar, der rund um eine 1,25-V-Referenzspannungsquelle IC3 und einen Standard-Opamp IC2 aufgebaut ist. Wenn man den Offset hinzuschaltet, sollte man den Spannungsbereich auf +/- 2,048 V einstellen. Eine zu messende Wechselspannung im Bereich von etwa -1 V bis +1 V führt dann zu 0..2 V am Eingang AIN0 des Chips, wodurch 15 bit ausgenutzt werden. In der eigenen Software muss man berücksichtigen, dass höhere Spannungen zu kleineren Ausgangswerten führen (Invertierung).

Eine andere Möglichkeit ist, den Offset an AIN0 für den oben beschriebenen Trick zu nutzen, die zu messende Spannung an AIN1 oder AIN3 anzulegen und den entsprechenden differentiellen Modus sowie einen Full-Scale-Bereich von +/- 1,024 V einzustellen. Man kann dann Spannungen von 0 bis 2 V mit 16 bit Auflösung messen.

Timing

Der ADC arbeitet nach dem Delta-Sigma-Prinzip, was eine hohe Auflösung und Genauigkeit verspricht. Diese Methode der Analog-Digital-Wandlung ist naturgemäß aber nicht die schnellste. Immerhin, bis zu 860 Samples/s sind drin. Die Samplerate ist konfigurierbar, voreingestellt sind 128 Samples/s. Wie viele andere ADCs auch kann der Baustein sowohl im Single-Shot- als auch im Continuous-Conversion-Mode arbeiten. Bei ersterem fordert man den Chip per Kommando zum Digitalisieren einer Spannung auf. Sobald ein gültiger Ausgangswert bereitsteht, setzt der ADC ein Flag in einem seiner Register, das man vom Mikrocontroller aus abfragen muss. Nun kann man den Wert (welcher in einem anderen Register abgelegt wird) auslesen und dem Chip bei Bedarf das nächste Kommando zum Sam-

pein geben. Beim Continuous-Conversion-Mode liefert der Chip dagegen kontinuierlich digitale Werte, ohne dass man ihn wiederholt dazu auffordern muss.

In all unseren Programmen einschließlich der ELF-Empfänger-Anwendung wird ausschließlich der Single-Shot-Modus innerhalb einer Endlosschleife benutzt. Nach dem Abholen eines neuen Ergebnis-Werts (und vor dem Absetzen des neuen Sample-Kommandos) wird dieser Wert im Mikrocontroller verarbeitet (zum Beispiel auf dem Display dargestellt oder über die serielle Schnittstelle ausgegeben). Bei der voreingestellten Sample-rate von 128 Samples/s ergeben sich Datenraten von etwa 100 bis 120 Hertz, was für die meisten Datenlogger-Anwendungen gut ausreichen dürfte.

Komparator

Der Conversion-Ready-Pin (Pin 2 von IC1) zeigt auf Wunsch ebenfalls an, wenn eine A/D-Wandlung beendet wurde. Der ADS1115 lässt sich aber auch als Komparator konfigurieren (**Bild 3**), dann dient dieser Pin als Alarm-Ausgang. Mit Hilfe zweier Register (Low Threshold und High Threshold) kann man ein hysteres-artiges Verhalten des Ausgangs konfigurieren, aber auch einen Fensterkomparator.

Der Ausgang steuert auf unserem Modul über T1 eine gelbe LED an. Zusätzlich haben wir das Signal auf einen der Pins des Gnublin/EEC-Steckers geroutet. Der Pin ist allerdings nur beim Linux- und Xmega-Board beschaltet, nicht beim Elektor Extension-Shield.

I²C

Wie schon gesagt wird der ADC über eine I²C-Schnittstelle angesprochen, wobei er dabei die Rolle des Slaves einnimmt. Mit reinen Schreib-

vorgängen werden zum Beispiel die Konfigurationsregister gefüllt oder Sample-Vorgänge angestoßen. Das Auslesen eines digitalisierten Wertes ist ein kombinierter I²C-Schreib/Lesevorgang, genauer gesagt muss man sogar zwei Mal schreiben und dann ein Mal lesen, wobei jedes Mal die Slave-Adresse des Chips mitgeschickt werden muss. Die von außen einstellbare Slave-Adresse macht es möglich, dass man mehrere ADS1115-Chips an die gleichen zwei Leitungen hängen kann (wenn man ihnen jeweils eine andere Adresse gibt). Welche Slave-Adresse der Chip in der eigenen Schaltung besitzen soll, teilt man ihm über Pin 1 mit. Insgesamt stehen vier Adressen zur Wahl, wobei man Pin 1 mit einem der Pins 3, 8, 9 oder 10 verbinden muss. Auf unserem Board haben wir alle vier Möglichkeiten mit Hilfe eines kleinen Jumperfeldes berücksichtigt.

Die beiden I²C-Signale werden über die Pins 5 und 6 des EEC/Gnublin-Steckers zum Chip geführt. Über JP2 und JP3 lassen sich I²C-Pullup-Widerstände zuschalten. Das Schöne an I²C ist, dass man unseren mit 3,3 V betriebenen Slave prima von einem 5-V-Controller aus ansteuern kann, zum Beispiel von einem Arduino Uno aus.

Software-Library

Selbst wenn man die Default-Einstellungen des ADC-Chips verwendet, kommt man um ein zeitraubendes Datenblatt-Studium nicht herum. Und auch das Bedienen der I²C-Schnittstelle des eigenen Mikrocontrollers kann mitunter eine knifflige Sache sein. Für beide Probleme bieten wir eine einfache Lösung an. Clemens Valens, Technik-Chef des Elektor-Labors, hat eine C-Bibliothek geschrieben, die einem das Heraussuchen der richtigen ADC-Register abnimmt und stattdessen High-Level-Funktionen zur Ansteuerung

Weblinks

- [1] www.elektor-labs.com
- [2] www.elektor-labs.com/project/arduino-16-bit-low-frequency-datalogger-130485-i-140035-i.13703.html
- [3] www.ti.com/lit/ds/symlink/ads1115.pdf
- [4] www.elektor-magazine.de/130485
- [5] www.elektor-magazine.de/130214
- [6] www.elektor-magazine.de/120126
- [7] www.elektor-magazine.de/140009
- [8] www.elektor-labs.com/contribution/from-the-lab-using-differential-mode.14034.html
- [9] www.elektor-magazine.de/120668



Bild 4.
Zum Test des ADC-Moduls und der Datenlogger-Anwendung kann man eine Leitung vom Arduino-Pin A3 zum Modul-Eingang AINO legen.

des Chips zur Verfügung stellt. Als Bonbon hat er eine kleine I2C-Software-Library integriert; die I2C-Leitungen werden dabei per Bit-Banging bedient. Damit sind auch Controller verwendbar, die keine Hardware-I2C-Schnittstelle mitbringen. Das Ganze ist hardwareunabhängig implementiert, lässt sich also auf allen Plattformen nutzen, bei denen ein C-Compiler zur Verfügung steht. Die Files sind zusammen mit einer kleinen Beschreibung von der Website zu diesem Artikel downloadbar [4].

Zuerst bindet man die Files `ads1x1x.c/.h` und `soft_i2c.c/.h` in das eigene Softwareprojekt ein und schreibt die entsprechenden Include-Anweisungen in den Code der Applikation. Um Anschluss an den eigenen Mikrocontroller zu bekommen, muss man folgende fünf Funktionen implementieren, was nicht schwer sein dürfte:

Weitere fünf Funktionen verbinden die ADC-Library von Clemens mit der I2C-Software-Library, diesen Code kann man aber 1:1 in eigenen Projekten übernehmen.

Als Lohn des Ganzen stehen nun Funktionen wie

```
ADS1x1x_start_conversion(&my_adc);
```

und

```
uint16 Result = ADS1x1x_read(&my_adc);
```

zur Verfügung.

`my_adc` ist eine Struktur, die beim Start des Programms zuerst definiert (`ADS1x1x_config_t my_adc;`) und dann über die Funktion `ADS1x1x_init(...)` mit den gewünschten Konfigurationsparametern gefüllt werden muss. Bei allen folgenden Befehlen wird dann ein Zeiger auf diese Konfigurationsstruktur mitgegeben. Wenn man mehrere Strukturen definiert, kann man über die Bibliothek mehrere Bausteine am selben Bus ansprechen, die sich verschieden konfigurieren lassen.

Mit folgendem Befehl definiert man zum Beispiel, dass der Chip über diejenige Slave-Adresse angesprochen wird, bei welcher der Adress-Pin mit „GND“ verbunden ist. Der Eingang AINO soll im Single-Ended-Modus abgefragt werden, wobei der Messbereich bei $-2,048\text{ V}$ bis $2,048\text{ V}$ liegt.

```
ADS1x1x_init(&my_adc, ADS1115, ADS1x1x_I2C_ADDRESS_ADDR_TO_GND, MUX_SINGLE_0, PGA_2048);
```

Der zweite Parameter zeigt an, dass wir einen ADS1115-Chip ansprechen wollen, die Library ist nämlich auch für die Schwestertypen verwendbar.

Datalogger-Software

Zur Verdeutlichung des Ganzen sollte man einmal in den Sourcecode unseres Datenlogger-Arduino-Sketches hineinschauen, welcher die oben-

```
void soft_i2c_scl_write(uint8_t value) // Zustand des mit SCL verbundenen Controller-Pins
                                        // auf High oder Low setzen
void soft_i2c_sda_write(uint8_t value) // Zustand des mit SDA verbundenen Controller-Pins
                                        // auf High oder Low setzen
uint8_t soft_i2c_sda_read(void)        // Zustand des SDA-Pins auslesen
void soft_i2c_sda_mode(uint8_t value)  // SDA-Pin als Ausgang (value=1) oder
                                        // Eingang (value=0) konfigurieren
void soft_i2c_delay(void)              // Delay von 8 µs
```

Stückliste

Widerstände (0,1 W, 0603):

R1,R2 = 100 k, 1 %
R3,R4 = 120 k, 1 %
R5 = 270 k, 1 %
R6,R7 = 2k2, 1 %
R8 = 1 k, 5 %
R9,R10 = 560 Ω, 1 %

Kondensatoren (0603):

C1 = 47 p, 50 V, 5 %, COG/NPO
C2 = 10 μ, 6V3, 20 %, X5R
C3 = 470 n, 10 V, 10 %, X5R
C4,C5 = 100 n, 16 V, 10 %, X7R

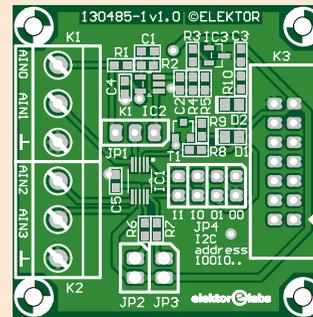
Halbleiter:

D1 = LED gelb, 0805
D2 = LED grün, 0805

T1 = 2N7002, SMD SOT23
IC1 = ADS1115IDGST
IC2 = OPA377AIDBVT
IC3 = REF2912AIDBZT

Außerdem:

K1,K2 = 3x1-Schraubklemme,
Rastermaß 5,08 mm
K3 = 2x7-Wannenstecker, Raster-
maß 2,54 mm
JP1 = 3x1-Stiftleiste, Rastermaß
2,54 mm, mit Jumper
JP2,JP3 = 2x1-Stiftleiste, Raster-
maß 2,54 mm, mit Jumper
JP4 = 2x4-Stiftleiste, Rastermaß
2,54 mm, mit Jumper
Platine 130485-1



genannten Libraries nutzt [4]. Innerhalb der nächsten Ausgaben werden übrigens noch weitere Demo-Programme und interessante Anwendungen auf Basis der Embedded Firmware Library [9] folgen.

Unser Arduino-Sketch macht genau dasselbe wie bei Kurt Diedrichs ursprünglichem Projekt: der ADC-Chip wird in einer Endlosschleife wiederholt angewiesen, die Spannung am Eingang AIN0 zu digitalisieren, im Single-Ended- und im Single-Shot-Modus. Der Messbereich beträgt -2,048 V bis 2,048 V, wobei im Single-Ended-Modus ja nur 0..2048 mV gemessen werden können und damit nur 15 bit erreichbar sind. Die Ausgangswerte (0..32767) werden zwischen den Wandlungen dezimal als ASCII-Zeichen über die serielle Schnittstelle ausgegeben, gefolgt von den Zeichen <CR> und <LF>.

Zum Test rüsten wir einen Arduino Uno mit dem Elektor Extension-Shield aus der letzten Ausgabe [7] aus und schließen das ADC-Board über ein 14-poliges Flachkabel an das Shield an. Die I2C-Adresse konfiguriert man auf „GND“, indem der betreffende Jumper direkt neben dem GnuBlin/EEC-Stecker gesteckt wird. Die Offsetkorrektur nutzen wir zuerst nicht (Jumper zeigt in Richtung Schraubklemmen).

Nun schließen wir einfach den Arduino-Pin A3 (an einer Buchsenleiste auf dem Shield zugänglich) über einen fliegenden Draht an den Eingang AIN0 an, so wie in **Bild 4** gezeigt. Die Spannung an A3 kann über das Poti auf dem Shield eingestellt

werden, somit haben wir eine einfache Testmöglichkeit. Da am Eingang des externen ADCs aber nicht mehr als 3,6 V anliegen dürfen, stellt man das Poti am Anfang auf Rechtsanschlag und dreht dann langsam nach links.

Den Sketch spielt man am besten per Bootloader auf den Arduino auf, dabei benötigt man nur ein USB-Kabel zwischen Arduino und PC. Sofort nach dem Flashen kann man sich die ausgegebenen Werte auf dem seriellen Monitor der Arduino-Entwicklungsumgebung oder einem anderen Terminalprogramm ansehen (115.200 Baud Datenrate und den richtigen COM-Port einstellen).

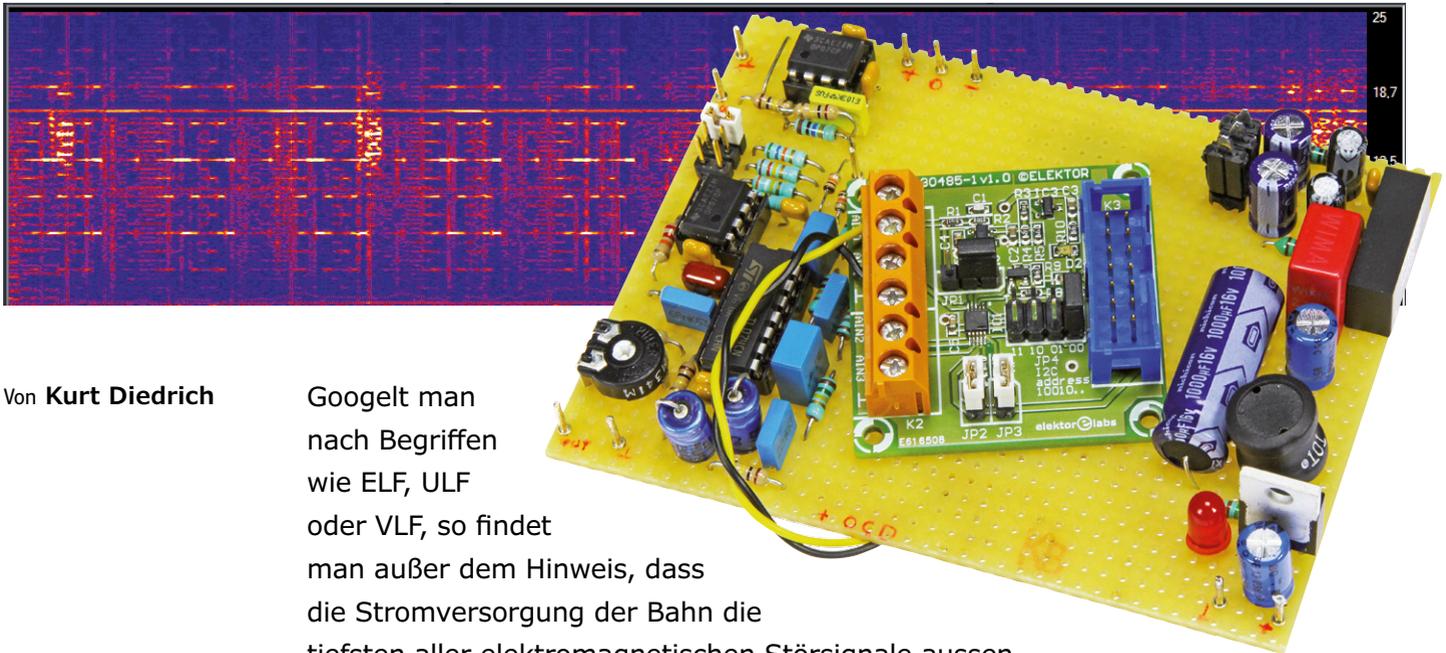
Für eine möglichst hohe Genauigkeit der Messwerte sollte man den Arduino allerdings nicht über USB, sondern über ein gutes Labornetzteil versorgen, das Gleichtaktstörungen wirksam ausfiltert.

Unheimliche Signale

Wer mag, kann nun einmal die Recorder-Software ausprobieren, die Kurt Diedrich entwickelt hat und in einem eigenen Artikel in diesem Heft beschreibt. Wenn man am Poti dreht, wird das von der Software über der Zeit aufgezeichnet. Jetzt sollte man auch einmal den Offset von 1022 mV zuschalten und schauen, wie sich die Werte verändern. Man kann nun Spannungen im Bereich von rund -1 V bis +1 V mit 15 bit Genauigkeit messen – und zum Beispiel auf ELF-Wellenjagd gehen. Am besten lesen Sie dazu gleich den Artikel „ELF-Empfänger“ in diesem Heft!

(130485)

ELF-Empfänger Mit Arduino und ADC-Modul



Von Kurt Diedrich

Googelt man nach Begriffen wie ELF, ULF oder VLF, so findet man außer dem Hinweis, dass die Stromversorgung der Bahn die tiefsten aller elektromagnetischen Störsignale aussendet, meist keine weiteren Erkenntnisse. Dass zwischen 0 Hz und der Bahnfrequenz von 16,666 Hz tatsächlich doch recht interessante Signale zu empfangen sind, scheint weitgehend unbekannt zu sein. Mit dem hier gezeigten Receiver, dem in einem eigenen Artikel vorgestellten ADC-Modul und einer kostenlosen PC-Software ist es möglich, diese Signale zu empfangen und aufzuzeichnen.

Es hat mich schon immer interessiert, was man wohl hören beziehungsweise auf einem Oszilloskop sehen würde, wenn man eine Spule von ein paar hundert Windungen an einen extrem empfindlichen Verstärker anschließt. Vor etwa zwölf Jahren beschloss ich, diesen Plan mit Hilfe moderner Elektronik endlich in die Tat umzusetzen. Meine erste zu diesem Zweck aufgebaute Schaltung unterschied sich von der hier vorgestellten Version lediglich durch ein einfacheres Filter und eine etwas altmodischere Art der AD-Wandlung. Zu meiner Überraschung war auf dem Monitor mehr als der erwartete 50-Hz-Netzbrumm zu sehen, doch leider ließen sich aus den wirren Zacken der komplexen Zeitsignale keine Rückschlüsse auf deren Zusammensetzung ziehen. Nachdem ich meine empfangenen Daten schließ-

lich einer *FFT-versus-Time*-Analyse unterzog, wurde mir sehr schnell klar, dass es sich bei dem „Gezappele“ auf dem Monitor um wiederkehrende Signale mit typischen Strukturen handelt, die nur erfassbar sind, wenn man sie über längere Zeiträume komprimiert darstellt. Sie sind bei beschleunigter Wiedergabe auch hörbar, erinnern gelegentlich an Tierlaute oder Funkfern-schreiber auf Kurzwelle und sind auf jeden Fall so interessant, dass ich mich bis heute mit ihnen beschäftige. Interessierte Leser finden weitere, detaillierte Informationen im Web-Blog vlf.it [3], einer Plattform für Amateure, die im ELF-VLF-Bereich empfangen und experimentieren. Auf diesen Seiten habe ich eine Reihe von Artikeln zum Thema in englischer Sprache veröffentlicht und zahlreiche Screenshots präsentiert.

Es ist davon auszugehen, dass unter anderem von den Versorgungstrafos in Wohngebieten extrem schwache, magnetische Wellen zwischen etwa 0,3 Hz und 25 Hz ausgestrahlt werden, die bis zu 1000 Mal schwächer als die durch den 50-Hz-Wechselstrom erzeugten Störfelder sind. Um diese Signale störungsfrei zu empfangen, muss der 50-Hz-Netzbrumm schon möglichst vor der Hauptverstärkung herausgefiltert werden, damit er den Empfänger nicht übersteuert.

Die Schaltung

Der hier gezeigte Empfänger arbeitet mit dem in einem eigenen Artikel vorgestellten ADC-Modul, einem Arduino Uno und einer – selbstverständlich kostenlosen - Recorder-Software für den PC zusammen. Die Einheit ist in der Lage ist, schwache Wechselströme beziehungsweise magnetische Wechselfelder mit bis zu unter einem Hertz zu empfangen, zu registrieren und darzustellen. Der Ausgang des Empfängers kann jedoch auch an andere Aufzeichnungsgeräte oder -Einheiten angeschlossen werden, wobei zu bedenken ist, dass Signale unter 16 Hz von PC-Soundkarten stark gedämpft werden.

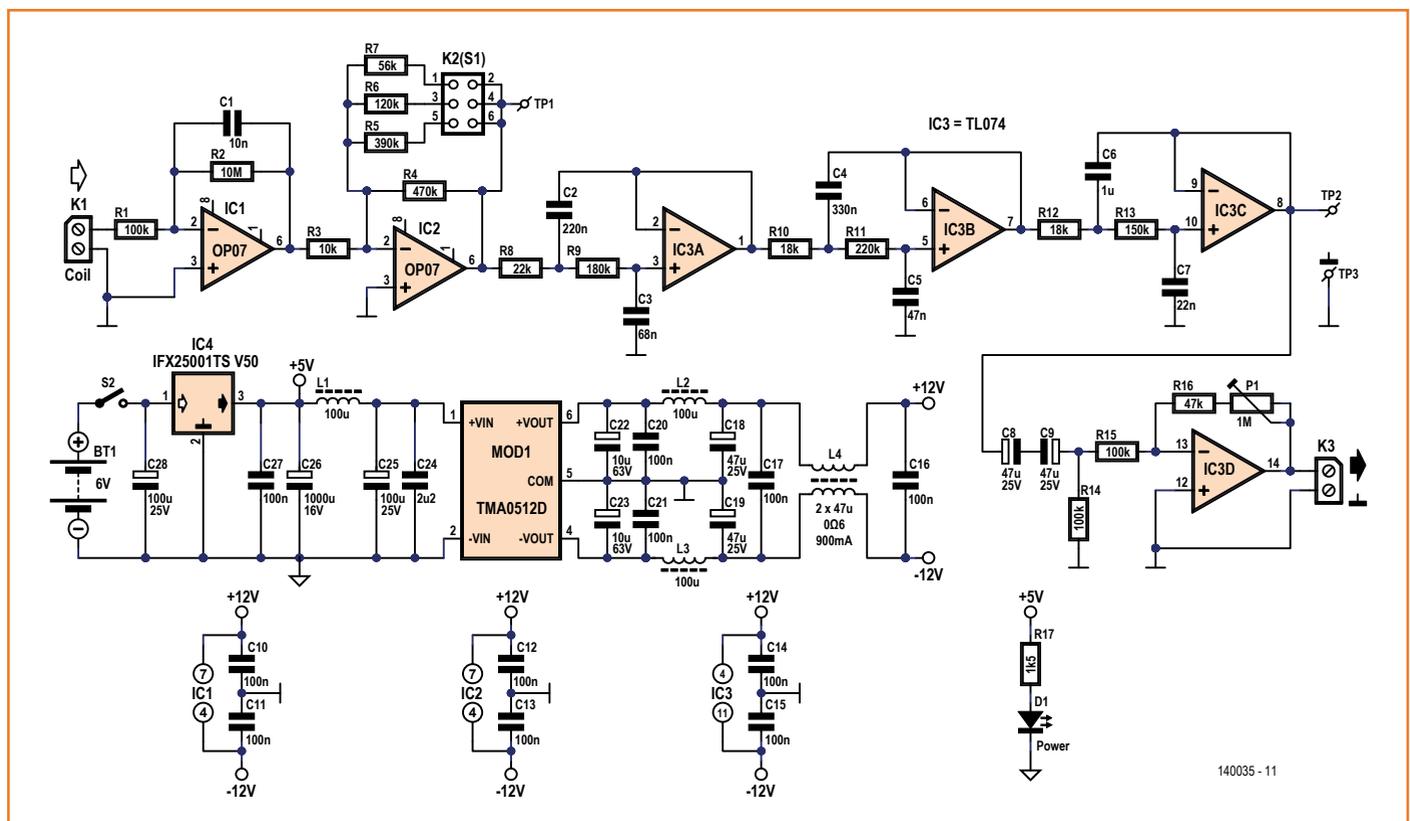
Bei der Schaltung handelt es sich um eine Kom-

bination eines hochempfindlichen Spannungsverstärkers und eines steilen (36 dB/Oktave) Sallen-Key-Tiefpassfilters mit einer Grenzfrequenz von ungefähr 21 Hz. Der Empfänger hat die Aufgabe, extrem schwache magnetische Wellen im Frequenzbereich unter 21 Hz bis (fast) 0 Hz zu verstärken und dabei den störenden Netzbrumm herauszufiltern. **Bild 1** zeigt das Schaltbild des Empfängers, das aus folgenden Funktionsgruppen besteht:

Linearisierer und Vorverstärker

Die hier interessierenden, von einer Spule kommenden und extrem schwachen Wechselspannungssignale im Mikrovoltbereich werden über eine Kombination aus Vorverstärker und Tiefpass (IC1) erst einmal so aufbereitet, dass der nachfolgende Sallen-Key-Tiefpass optimal angesteuert wird. Dieser vorgeschaltete, einfache Tiefpass (sozusagen ein Vor-Filter) ist unter anderem notwendig, um den im Verhältnis zum Nutzsignal störenden 50-Hz-Netzbrumm so weit abzuschwächen, dass er nicht zur Übersteuerung und damit zu einem Rechtecksignal zwischen den maximalen Ausgangsspannungen der Opamps führen würde. Dies könnte geschehen, wenn sich

Bild 1. Schaltbild des ELF-Empfängers (ohne Datenlogger).



die Spule in der Nähe einer Netzleitung befindet, durch die ein hoher Strom fließt. Die um IC1 aufgebaute Schaltung besitzt noch eine zweite Aufgabe: Durch die Eigenschaften der am Eingang der Schaltung liegenden Spule werden tiefere Frequenzen zusehends abgeschwächt, sodass die Amplituden der empfangenen Signale in Richtung Null Hertz immer schwächer werden. Durch die Wirkung des zu R2 parallelen Kondensators C1 wird dies weitgehend kompensiert (Linearisierer). **Bild 2** zeigt die Verstärkung am Ausgang von IC1. Bei der um IC2 aufgebauten Stufe handelt es sich

Bild 2.
Verstärkungsverlauf am Ausgang von IC1.

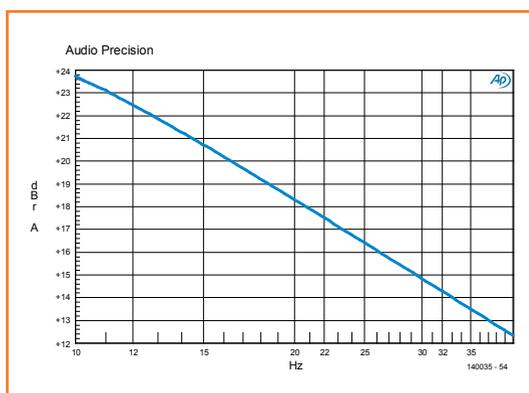


Bild 3.
Filter IC3 erreicht eine Steilheit von etwa 36 dB/Oktave!

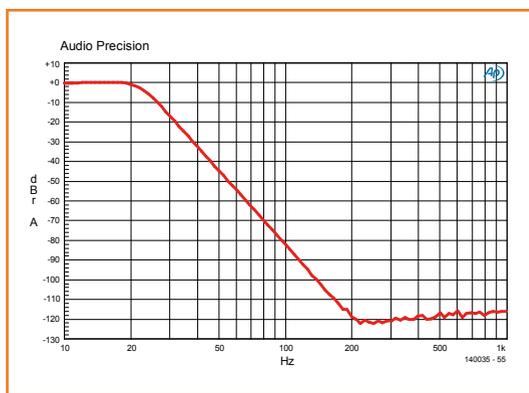
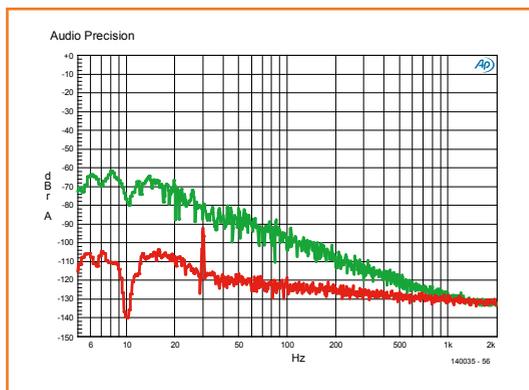


Bild 4.
Die Gleichtakt-Drossel senkt störendes Rauschen um bis zu 40 dB.



um die bekannte Standardschaltung des invertierenden Verstärkers. Der Verstärkungsfaktor ist durch die Auswahl mehrerer Gegenkopplungswiderstände umschaltbar. Dies ist unbedingt erforderlich, da sich, je nach Standort des Receivers, völlig unterschiedliche Intensitäten der empfangenen Signale ergeben können. Da R4 nicht in den Umschalter mit einbezogen ist, ergibt sich auch bei offenen Schalterstellungen immer eine definierte Gegenkopplung. Dies hat den Vorteil, dass im Moment des Umschaltens, wenn der Schaltkontakt für eine extrem kurze Zeitspanne „in der Luft hängt“ keine störenden Impulse am Receiver-Ausgang erscheinen.

Die Verstärkung des invertierenden Verstärkers ergibt sich aus dem Quotienten Gegenkopplungswiderstand geteilt durch Vorwiderstand:

$$V = R_g/R_v$$

Beim Umschalten von R7 nach R4 ergeben sich daher nacheinander (ungefähr) die Verstärkungen 5, 10, 21, 47, wobei sich der letzte Wert bei offenem Schalter/Jumperfeld einstellt.

Filter

Die restlichen vier Operationsverstärker sind in einem IC zusammengefasst, dem TL074. IC2A bis IC2C bilden zusammen ein steilflankiges Sallen-Key-Filter mit insgesamt 36 dB/Oktave. Auf Grund des hohen Pegels des 50-Hz-Signals im Verhältnis zu den erwünschten Signalen ist auch dieses Filter dringend notwendig, um Übersteuerungen zu vermeiden. Wer mehr über Sallen-Key-Filter wissen möchte, findet in der Fachliteratur und im Internet die gewünschten Hintergrundinformationen [2]. Die Eckfrequenz des Filters beträgt, um genau zu sein, 21,5 Hz, was weit genug von den störenden 50 Hz entfernt ist und sich doch noch oberhalb des erwünschten Empfangsbereiches befindet.

Die Werte der Kondensatoren und Widerstände des Filters müssen möglichst genau eingehalten werden, da ansonsten der erforderliche Übertragungsverlauf nicht mehr garantiert werden kann. In **Bild 3** ist zu sehen, wie steilflankig das resultierende Filter ist (gemessen am Ausgang von IC3C).

Hochpass und Endstufe

Bei der hohen Verstärkung (je nach Stellung von S1 bis zu etwa 50.000) können auch noch so kleine Offsetspannungen so groß werden, dass sie das Ausgangssignal um mehrere Volt in den

positiven oder negativen Bereich verschieben und somit zu einem unerwünschten Störfaktor werden. Um dies zu vermeiden, wurde zwischen den Filter-Ausgang (IC3C) und den Eingang der nachfolgenden Verstärkerstufe IC3D ein Hochpassfilter (C8+C9/R14) eingefügt, dessen Eckfrequenz weit unter dem erfassten Bereich liegt. Das Filter wirkt sich daher nicht störend auf den Frequenzgang der empfangenen Signale aus. Die Spannung am Ausgang von IC3D ist damit stets nullsymmetrisch. Die Gesamtverstärkung von IC3D kann mit P1 zwischen 0,5 und etwa 10,5 verändert werden. Dies sollte im Zusammenhang mit der umschaltbaren Vorstufe für nahezu alle Anwendungsfälle genügen.

Netzteil

Das Netzteil ist für einen Akku- oder Batteriebetrieb von 6 V vorgesehen. Mit IC4 wird die Spannung zunächst auf 5 V reduziert und stabilisiert, damit sie den Bedingungen des nachfolgenden Wandlers TMA0512D genügt. Dieser Wandler erzeugt aus der Eingangsspannung von 5 V zwei komplementäre Ausgangsspannungen von 12 V, mit denen die OpAmps versorgt werden. Um Störungen wie Rauschen zu reduzieren, ist der Einsatz aller im Schaltbild des Netzteils eingezeichneten Induktivitäten und Kapazitäten unbedingt notwendig, denn wie wohltuend sich beispielsweise der Einsatz der Gleichtakt-Drossel L4 auswirkt, zeigt **Bild 4**: Im relevanten Frequenzbereich wird das Rauschen um 30...40 dB abgesenkt!

Wer den Empfänger statt mit Akkus oder Batterien lieber mit einem Netzteil betreiben möchte, kann an den Eingang statt der Akkus auch ein 6-V-Netzteil mit der erforderlichen Leistung anschließen.

Spule und Elektroden

Um ein schwaches Magnetfeld zu detektieren, setzt man eine empfindliche Empfangsantenne ein: eine Spule mit etwa 2000 bis 4000 Windungen und möglichst großem Durchmesser. Dabei muss es sich gar nicht um solche ausgewachsenen Konstrukte wie in **Bild 5** handeln, Durchmesser von 30...50 cm reichen (zunächst) völlig aus. Die Empfindlichkeit der Spule (nicht zu verwechseln mit ihrer Induktivität!) nimmt linear mit der vom Spulenring eingeschlossenen Fläche und der Zahl der Windungen zu. Die Spule sollte die Form eines Rings besitzen und kann, wenn man den Draht von einer Rolle aus dem Fachversand abwickelt [3], mit einer selbstgebauten Wickelmaschine aus



einem alten Metallbaukasten schnell und einfach hergestellt werden.

Zur Herstellung der Spule hat sich lackisolierter Kupferdraht von 0,25 mm Durchmesser als recht gut erwiesen. Er ist noch nicht so dünn, dass er bei unvorsichtigem Hantieren gleich reißt.

Die Spule muss beim Empfang flach auf einer möglichst metallfreien Unterlage aufliegen und sollte sich möglichst weit von stark stromdurchflossenen Netzleitungen entfernt befinden. Ganz wichtig: Aufgrund der Einflüsse des Erdmagnetfeldes darf der Empfänger nur betrieben werden, wenn die Spule keinen Bewegungen oder Erschütterungen ausgesetzt ist.

Der Empfänger kann auch mit Elektroden betrieben werden, die aus circa 20 cm langen Metallstäben bestehen und in einem Abstand von etwa einem bis zwei Metern in die Erde gesteckt werden. Damit lassen sich an der Erdoberfläche vorhandene Wechselströme im vorgegebenen Frequenzbereich erfassen.

Sicherheitshinweis: Aufgrund eventueller vagabundierender Netzspannungen im Erdreich ist beim Arbeiten mit Erd-Elektroden immer ein 1:1-Mikrofon-Übertrager (Trenntrafo) am Eingang des Empfängers erforderlich, sobald der Receiver oder irgendein damit verbundenes Gerät über das Netz versorgt wird. Ich habe mit einem Exemplar der Firma *Jensen* experimentiert, das sich als hervorragend geeignet erwies (JT-11P-1). Auf der Webseite [4] dieser US-Firma werden verschie-

Bild 5.
Selbstgebaute
Spulenwickelmaschine
des Autors. Für praktische
Anwendungen genügen auch
kleinere Exemplare.

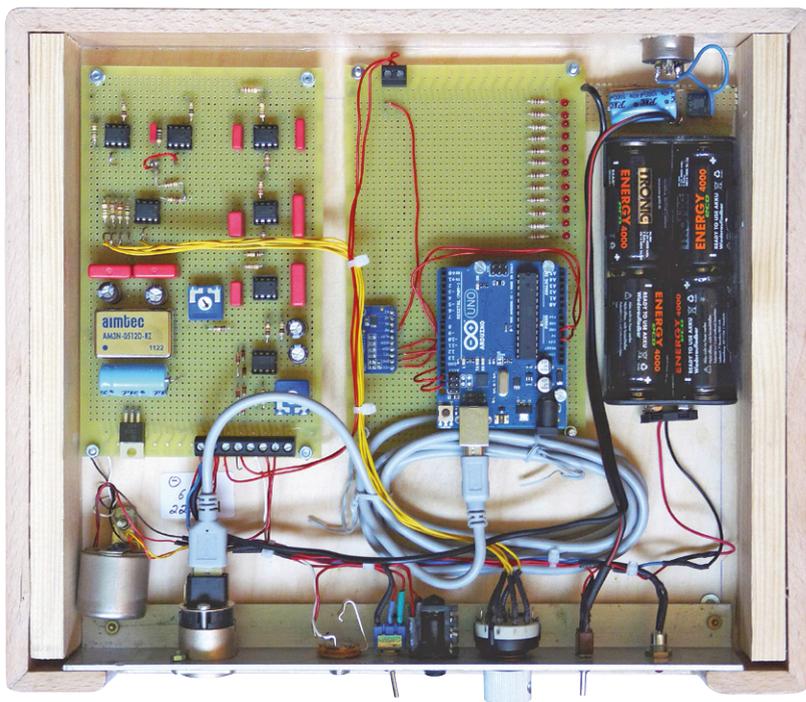


Bild 6.
Erster Prototyp des Autors.
Im Gehäuse ist auch der
Arduino-Datenlogger zu
sehen.

Bild 7.
Probeaufbau des Elektor-
Labors.

dene geeignete Exemplare vorgestellt.
Die Spieße werden über ein (nicht abgeschirmtes)
Kabel mit dem Übertrager verbunden, dessen
Sekundärseite statt der Spule an den Empfän-
gereingang angeschlossen wird.

Ableich, Anschluss und Test

Für die Schaltung des ELF-Empfängers steht kein
Platinenlayout zur Verfügung, so dass am Nach-

bau interessierte Leser hier zur Selbsthilfe grei-
fen oder die wenigen Bauteile auf einer Lochras-
ter-Platine unterbringen müssen. Beim Layouten
können Programme wie LochMaster von Abacom
oder das kostenlose Blackboard [5] behilflich sein.
Der Prototyp des Autors ist in **Bild 6** zu sehen,
in **Bild 7** der Probeaufbau des Elektor-Labors auf
einer Lochrasterplatine.

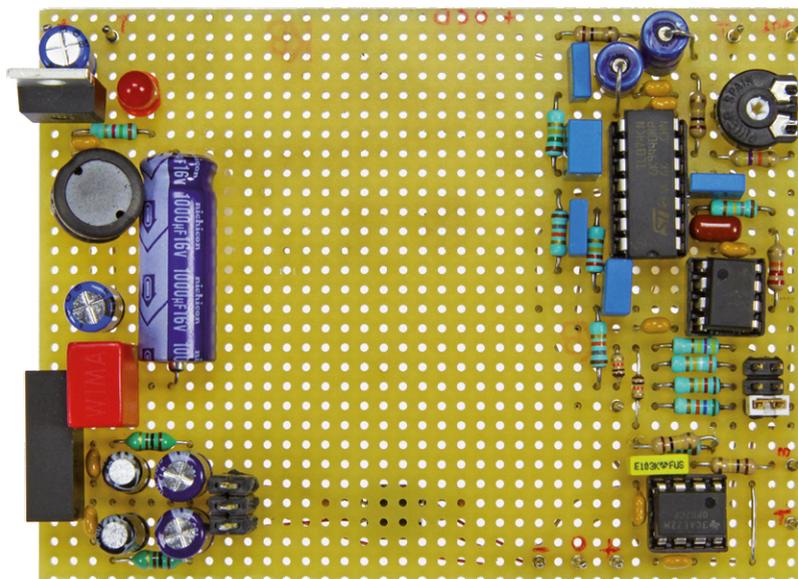
Ist alles aufgebaut und geprüft, kann der Abgleich
des Empfängers mit einem Oszilloskop erfolgen.
Schließen Sie die Spule an den Eingang an,
„bewaffnen“ Sie sich mit einem starken Magne-
ten (zum Beispiel aus einem Lautsprecher) und
untersuchen Sie die Ausgänge der Opamps der
Reihe nach. Am Ausgang von IC2 darf der (bisher
nur vorgefilterte und überall zu empfangende)
50-Hz-Brumm die 50-%-Aussteuerungsgrenze
nicht überschreiten. Reichen die zur Verfügung
stehenden Positionen an K2/S1 dazu nicht aus,
so muss der Wert von R3 erhöht werden.

Mit jedem weiteren Ausgang werden die 50-Hz-Si-
nusanteile immer schwächer. Stellen Sie nun das
Oszilloskop auf 1 V/Div ein und bewegen Sie den
Magneten von Hand in einem Abstand von 2 m
von der Spule ein bis zwei Mal pro Sekunde hin
und her. Es müssten nun deutliche Ausschläge
bis zur Clippgrenze zu beobachten sein. Auch
bei nicht bewegtem Magneten müsste nun eine
leichte Welligkeit durch die Signale aus der Um-
gebung (wenn Ihr Heim nicht gerade mitten im Wald
liegt) zu verzeichnen sein. Stellen Sie nun P1 so
ein, dass die Spitzenwerte dieser Welligkeit nicht
mehr als ± 1 V betragen und damit im optimalen
Messbereich des AD-Wandlers liegen. Am Ausgang
sollte das Signal frei von jeglichen Offsetspan-
nungen sein. Auch die 50-Hz-Sinusschwingun-
gen sollte man kaum noch erkennen können. Am
Ausgang K3 liegt nun ein reines AC-Signal für die
weitere Verarbeitung durch den AD-Wandler an.

Installation und Betrieb des Recorders

Den ELF-Empfänger schließt man an das ADC-Mo-
dul an, das in einem eigenen Artikel in diesem
Heft beschrieben ist; der ADC tastet das Signal
mit einer Auflösung von 15 bit und einer Samp-
lerate von rund 112 Hz ab. In dem Artikel ist
auch nachzulesen, wie das ADC-Modul mit einem
Arduino Uno verbunden wird, der die digitalisi-
erten Daten mit einem einfachen Programm
(Sketch) entgegennimmt und an den PC und die
Recorder-Software weiterleitet.

Die Recorder-Software wurde in der Program-
miersprache Processing verfasst, die C ähnelt.

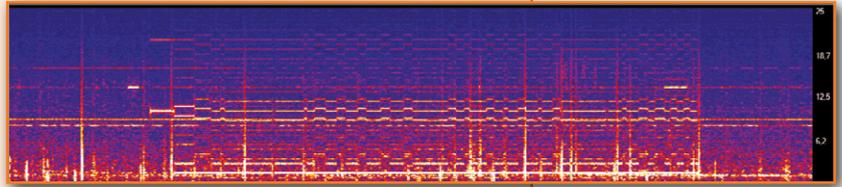


Was ist ELF?

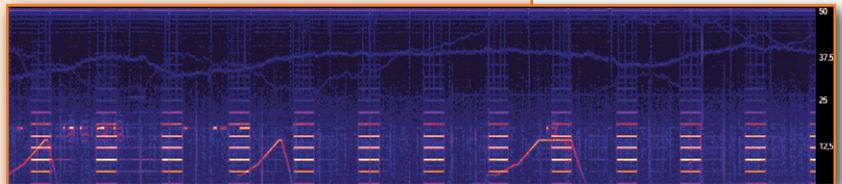
ELF-Signale: Ein geheimnisvoller und zum Teil von Mythen durchsetzter Begriff, der jedoch in Wirklichkeit nichts weiter besagt, als dass es sich hier um elektromagnetische Wellen mit extrem niedriger Frequenz (Extremely Low Frequency) von 3 Hz bis 30 Hz handelt. Da sich der kommerzielle Funk solch niedriger Frequenzen nicht bedient, ist es natürlich besonders interessant, einmal „nachzuschauen“, was sich in diesem Bereich abspielt.

Viele der mit dem hier gezeigten Empfänger nachweisbaren Signale werden offensichtlich in Form magnetischer Wellen von den lokalen Netzspannungs-Versorgungstrafos in Wohngebieten ausgestrahlt. Das dort ausgedehnt vorhandene Netz an metallischen Leitungen (Erdungen, Wasser- und Gasrohre usw.) verhält sich scheinbar wie eine große, unterirdische Antenne, die schwächste niederfrequente Wechselströme aus dem Erdreich (woher diese auch immer stammen mögen) sammelt und dem gemeinsamen Verbindungspunkt an den lokalen Trafostationen zuführt. Dort könnten (aber das ist nur eine Vermutung) diese Ströme über die mit der Erdung verbundene Petersen-Spule (zur Erdschlusskompensation) als magnetische Felder abgestrahlt werden. Neben diesen bisher als ungeklärt geltenden Signalen ist vor allem tagsüber eine zunehmende „Verschmutzung“ durch (vermutlich) kommerzielle und kommunale Anlagen wie Wechselrichter, Frequenzwandler und Schaltnetzteile zu verzeichnen. Mit etwas Geduld ist es auch möglich, die so genannte Schumann-Resonanz im Bereich um etwa 7,5 Hz sowie das 16 2/3-Hz-Signal der Bahnstromversorgung nachzuweisen, das auch hervorragend zum Test und zur Kalibrierung des Receivers dienen kann. Auch ein im Vergleich dazu recht profaner Einsatz des Empfängers als hochempfindlicher Detektor für (ausschließlich) sich bewegende Metallgegenstände ist denkbar. Dazu gehören zum Beispiel fahrende Autos, die bis zu einer Entfernung von etwa 20 m registriert werden können.

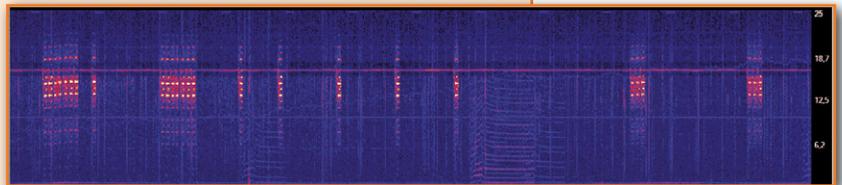
Folgende Aufnahmen zeigen einige Beispiele für die reichhaltige, vielseitige und interessante „Signal-Ernte“:



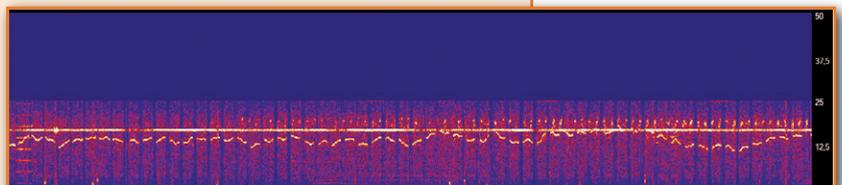
Wow! Das Signal erschien nur ein einziges Mal in einer Nacht im September 2013. Dauer etwa eine Stunde. Aufnahme mit Elektroden. Frequenz zwischen 0 und 20 Hz



Ein Rechtecksignal von 1,6 Hz, das europaweit an verschiedenen Orten zu unregelmäßigen Zeiten erscheint. Typische Kennzeichen: Phasen von Aktivität und Pausen wechseln regelmäßig.

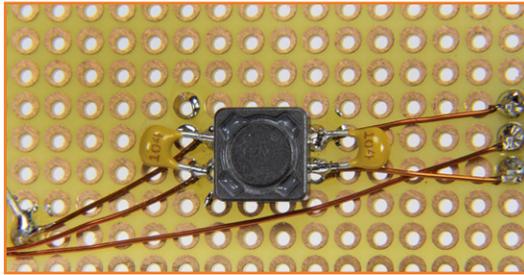


Extrem starke 16-Hz-Bursts, die sich an bestimmten Orten konzentrieren und sogar direkt als tiefes Brummen in Audioverstärkern hörbar sind.



Klingt bei schnellem Playback wie Pfeifen und erschien über Jahre hinweg täglich für mehrere Stunden am Wohnort des Autors.

Bild 8.
Spannungsversorgung
und Filter/Verstärker
sollten so weit wie möglich
auseinander liegen.
Dazwischen ist (hier auf
der Platinenunterseite) die
Induktivität L4 angebracht.



Es gibt geschweifte Klammern für Code-Blöcke; jede Anweisung muss mit einem Semikolon abgeschlossen werden. Die Programmierumgebung ist sehr einfach: Editor öffnen und Quelltext schreiben. Dann den Startknopf anklicken und los geht's.

Software-Installation

Damit ein Processing-Programm überhaupt auf Ihrem Rechner läuft, müssen Sie die dazu erforderliche Software aus dem Internet auf Ihren Rechner herunterladen.

Gehen Sie dazu auf die Processing-Webseite [7] und folgen Sie den dortigen Anweisungen. Die heruntergeladenen Daten können Sie in ein beliebiges Verzeichnis auf Ihrer Festplatte verschieben. Zu diesen Daten gehört auch eine Datei mit dem Namen *processing.exe*. Starten Sie dieses Programm, wenn Sie Ihre eigene Processing-Software schreiben möchten. Zahlreiche beeindruckende Beispielprogramme zeugen nicht nur von der Leistungsfähigkeit dieser Sprache, sondern zeigen auch, wie man sie optimal anwendet. Das in Processing geschriebene Recorder-Programm kann von der Elektor-Webseite [8] in ein beliebiges Verzeichnis heruntergeladen werden.

Wichtig: Das Processing-Programm muss sich in einem Unterverzeichnis befinden, das denselben Namen trägt wie das Programm selbst - aber ohne die Endung „.pde“. Auch alle vom Programm benötigten Ressourcen (wie wav-Dateien oder eingebundene Grafiken) müssen sich in diesem Unterverzeichnis befinden. Nach einem Doppelklick auf die Recorder-Datei (Recorder_...pde) öffnet sich automatisch das Processing-Editorfenster und der Programmcode wird aufgelistet.

In folgender Zeile muss man statt „COM3“ die vom Arduino belegte COM-Schnittstelle des PCs eintragen (siehe Geräte-Manager in Windows):
`serport = new Serial(this, "COM3", 115200);`
Dann speichert man den Programmcode mit File → Save.

Nach einem (Einfach-)Klick auf den Pfeil links oben im Editorfenster wird das Programm gestartet. Das Editorfenster mit dem Quellcode bleibt währenddessen (im Hintergrund) auf dem Bildschirm. Leider ist es mir (trotz zahlreicher Hinweise in verschiedenen Quellen) nicht gelungen, eine funktionsfähige Exe-Datei des Programms zu erzeugen. Weitere Informationen zu Processing finden Sie im Editor selbst (Help → Reference) und auf zahlreichen weiteren Internetseiten.

Bedienung

Die Bedienung des Software-Recorders im windows-ähnlichen Fenster (**Bild 9**) ist beinahe selbsterklärend. Die Messergebnisse werden auf der linken Seite in drei Fenstern dargestellt:

Zeitsignal

Ein Durchlauf entspricht fünf Sekunden. Nach dem Programmstart ist hier immer ein Signal sichtbar, auch wenn nicht aufgenommen wird - und nach dem Stopp der Aufnahme.

FFT vs. Time

Alle *x* Sekunden (*x* abhängig vom bei Downsampling eingestellten Wert) wird eine neue Linie gezeichnet - auch wenn nicht aufgenommen wird - und nach dem Stopp der Aufnahme.

Übersichtssignal

Nach jedem Durchlauf von 5 s im oberen Fenster wird die höchste Amplitude dieses Zeitabschnitts markiert.

Auf der rechten Seite des Recorders werden die Parameter für die Messung und die Darstellung eingestellt:

Recording time

Die Aufnahmelänge.

Downsampling

Zoom in Y-Richtung, sodass man tiefe Frequenzen besser sehen kann. Bezieht sich nur auf die gezeigte FFT und *nicht* auf die Aufnahme.

FFT brightness

FFT-Darstellung heller oder dunkler einstellen. Bezieht sich nur auf die gezeigte FFT und *nicht* auf die Aufnahme.

FFT scroll

Vor- und Zurückblättern der aufgezeichneten Ana-

lysedaten. Gilt nur für die während der aktuellen Aufnahmephase aufgezeichneten Daten, die sich noch im RAM befinden, da die FFT-Daten nicht auf der Festplatte gespeichert werden.

Mausposition

Koordinaten der Mausposition und Nummer des angeklickten Buttons. Sehr wichtig, wenn man selber am Programm arbeiten möchte.

Aufnahme

Linker Button

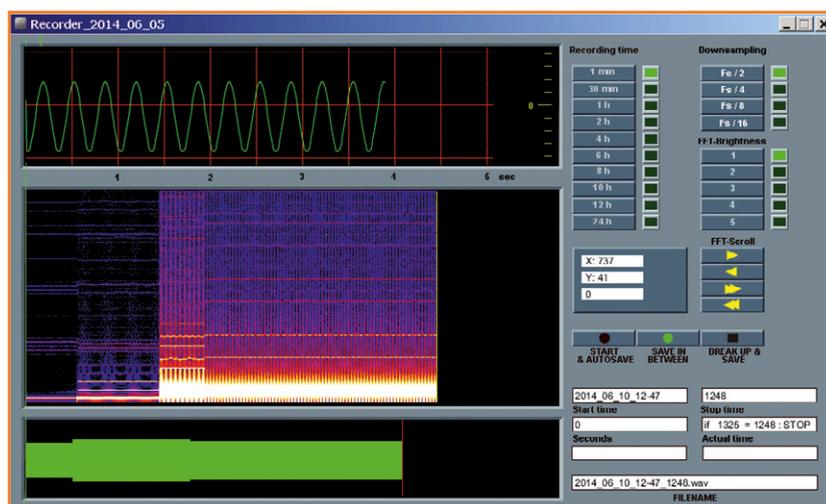
Normaler Start einer Aufnahme mit oben eingestellter Länge, automatischer Speicherung am Ende der eingestellten Zeit und regelmäßiger Zwischenspeicherung. Bei Aufnahme hellrot.

Achtung: Die automatisch gespeicherten Dateien enthalten im Dateinamen die Startzeit und die geplante Stoppzeit (zum Beispiel 18:20, wenn die Aufnahme um 12:20 gestartet wurde und eine Laufzeit von sechs Stunden eingestellt war). Wird die Aufnahme vorzeitig abgebrochen, so findet man die bisher gespeicherten Daten daher unter dem von Anfang an geplanten Dateinamen (2014_07_02_1220_1820).

Die Datei kann jedoch, je nach dem Zeitpunkt des vorzeitigen Abbruchs, ab einem bestimmten Punkt statt Daten nur noch Nullen enthalten. Bei Processing ist es nämlich nur möglich, ein komplettes Array zu speichern und nicht, wie sonst üblich, nur den mit Daten belegten Teil. Daher haben die Dateien auch bei vorzeitigem Abbruch immer die maximale Größe.

Mittlerer Button

Zum gezielten Zwischenspeichern. Für den Dateinamen gilt dasselbe wie beim automatischen Speichern. Das Drücken dieses Buttons



erlaubt es, die bis zum aktuellen Zeitpunkt aufgenommenen Daten in einem Analyseprogramm zu betrachten.

Rechter Button

Der Knopf rechts bricht die Aufnahme ab und speichert die bisher aufgenommenen Daten auf der Festplatte.

Achtung: In diesem Fall enthält der Dateiname als Stopp-Zeit die tatsächlich zum Zeitpunkt des Stopps geltende Uhrzeit. Beim Betätigen dieses Buttons wird demnach eine zusätzliche Datei erzeugt.

Datenausgabefenster am unteren, rechten Bildrand:

Nach dem Start wird anhand der eingestellten Länge automatisch auch die Zeit angegeben, zu der die Aufnahme gestoppt wird. Ganz unten steht der Dateiname, der auch beim Zwischenspeichern beibehalten wird.

(140035)

Bild 9.

Grafische Oberfläche der Recorder-Software.

Weblinks:

- [1] ELF-Web-Blog: www.vlf.it
- [2] https://en.wikipedia.org/wiki/Sallen%E2%80%93Key_topology
- [3] Spulendraht: www.sauter-shop.de/kupferlackdraht/index.html
- [4] Eingangsübertrager: www.jensen-transformers.com/In_in.html
- [5] <http://blackboard.serverpool.org/>
- [6] www.elektor-labs.com/project/arduino-16-bit-low-frequency-datalogger-130485-i-140035-i.13703.html
- [7] <http://processing.org/>
- [8] www.elektor-magazine.de/140035



DesignSpark Tipps & Tricks

Die Platzierung von Bauelementen

Von **Neil Gruending**
(Kanada)

In dieser Folge geht es darum, wie man die Tools zur Bauteil-Platzierung in DesignSpark verwendet.

In den vorausgehenden Folgen haben wir uns mit dem Zeichnen von Schaltplänen beschäftigt. Dieses Mal soll es um weitere Funktionen von DesignSpark gehen – angefangen bei der automatischen Bauteilplatzierung bis hin zu manuellen Positionierungstechniken.

Automatische Positionierung von Bauteilen

Das in den vorausgegangenen Folgen entwickelte LED-Treiber-Board eignet sich gut für das Ausprobieren des Tools zur automatischen Bauteilplatzierung. Ein Klick auf *Tools* → *Auto Place Components* → *All Components* und schon zeigt sich das Autoplace-Components-Fenster von **Bild 1**. Hier kann man das Autoplace-Tool konfigurieren, bevor man es anwendet.

Bei einem minimalen Bauteile-Abstand von 0,25 mm und einem Raster von 0,25 mm ergibt sich das Resultat von **Bild 2**. Wie man sieht, sind alle Bauteile sauber platziert und DesignSpark hat sich redlich um eine gute Organisation der Bauteile gekümmert.

Aber was, wenn man nicht alle Bauteile automatisch platzieren will? Wählt man „All Unplaced“ aus dem Menü „Auto Place Components“, nimmt DesignSpark alle noch nicht auf der Platine befindlichen Bauteile und platziert sie automatisch. Man kann auch eines oder mehrere Bauteile auswählen und „Selected Components“ wählen, um nur diese zu platzieren. Beide Befehle nutzen die letzte für „All Components“ vorgenommene Konfiguration.

Manchmal möchte man die meisten Bauteile platzieren, aber ein paar bestimmte sollen an der aktuellen Position bleiben. Als Beispiel sollen Befestigungslöcher fixiert sein, aber alle anderen Bauteile platziert werden. Eine Lösung wäre, die Befestigungslöcher durch Klick und Auswahl von „Fix Item“ zu blockieren und die Option „Don't

place fixed Components“ in der Konfiguration anzuhaken.

DesignSpark überprüft bei allen Bauteilen, ob deren Pads mit Leiterbahnen verbunden sind und überspringt diese bei der Platzierung. So kann man sicher sein, dass keine Bauteile von ihren Netzen getrennt werden. DesignSpark prüft allerdings nicht, ob die Stelle, an die es ein Bauteil setzt, frei von Leiterbahnen ist. Das kann zu Kurzschlüssen wie in **Bild 3** führen, wo Q2 fälschlicherweise mit der Leiterbahn zwischen Q1 und Q4 verbunden ist. Bei größeren Platinen benutzt man den **DRC (Design Rule Check)** und solch ein Fehler wird als „track to pad clearance error“ gelistet.

Manuelle Bauteil-Platzierung

Etwas Handarbeit gefällig? Bauteile auswählen und Herumschieben ist einfach, doch es ist nicht so einfach zu sehen, wie weit die Bauteile voneinander entfernt sind. Das Tool zur automatischen Platzierung stellt solche Berechnungen an, doch bei händischer Platzierung ist man selbst dafür verantwortlich. Die einfachste Methode hierfür ist ein 0,25-mm-Raster und das Sicherstellen ausreichend vieler Rasterpunkte für den gewünschten Abstand zwischen den Bauteilen. Wenn man z.B. zwei Bauteile so nah wie möglich ohne Überlappung platzieren möchte, bewegt man eines davon die nötige Anzahl an Rasterpunkten horizontal oder vertikal. Das funktioniert gut bei kleinen Platinen (wie der für den LED-Treiber), die Aufgabe kann aber bei unterschiedlichen Bauteilabständen oder vielen Bauteilen komplex werden. Eine Alternative ist die Verwendung von Sicherheitsabständen oder Platzierungsflächen in der Bauteil-Library. Ein so genannter „Placement Courtyard“ ist schlicht eine quadratische Fläche in einem mechanischen Layer, die den Bauteilumriss samt Bauteiltoleranzen umschließt. Eigentlich sind diese Flächen nicht für diesen Zweck gedacht, aber ich nutze sie gewöhnlich so, dass ich mit einer kleinen Zugabe auf einen Sicherheitsabstand von mindestens 0,25 mm komme. Wenn man solche Bauteile auf der Platine platziert, kann man das beliebig tun, solange sich



Bild 1.
Autoplace-Fenster.

diese Flächen nicht überlappen. Nun dazu, wie man dieses in DesignSpark realisiert.

Wenn man den Footprint-Wizard zum Zeichnen eines neuen Bauteils verwendet, wird man gefragt, ob man als letzten Schritt eine „Placement Outline“ anlegen will, was nichts anderes als solch ein „Placement Courtyard“ ist. Man muss nur angeben, wie weit weg von den Pads die Grenzen verlaufen sollen, und schon werden diese automatisch hinzugefügt. Ich benutze hierfür gerne den „Top Assembly Layer“, sodass die Grenze später auch als „Component Outline“ beim Bestückungsdruck verwendet werden kann. Man kann den Sicherheitsabstand auch manuell bei einem Bauteil wie in **Bild 4** realisieren. Doch vor einem Update der Platine sollte man zuerst die „Assembly Layer“ hinzufügen. Hierzu öffnet man das Fenster „Design Technology“ im Menü „Settings“ und klickt auf den Tab „Layer Types“. Nun auf „Add“ klicken und einen Namen vergeben. Ich verwende hier „Assembly“. Der Begriff sollte besser nichtelektrisch sein, und man sollte alle Haken in der „Include Area“ deaktivieren. Bei den Settings sollte „Placement Shapes“ aktiviert werden und dann ein Klick auf „Ok“ folgen.

Nun kann man noch per Klick auf den Layers-Tab die oberen und unteren Assembly-Layer hinzufügen. Man klicke hierzu auf „Add“ und benenne den Layer mit „Top Assembly“. Dann ändert man den Typ in „Assembly“ und wählt eine Farbe. Nach dem Klick auf „Ok“ wiederholt man das Ganze für den „Bottom Assembly Layer“ – man sollte aber nicht vergessen, dabei die Seite auf „Bottom“ umzudrehen.

Nach diesen Operationen sieht das Resultat aus wie in **Bild 5** mit einem neuem Arrangement. Alle „Placement Courtyards“ sind jetzt einfach zu sehen, wenn die Bauteile korrekt platziert wurden und diese sich nicht mit denen anderer Bauteile überlappen oder berühren.

Fazit

In dieser Folge wurden unterschiedliche Wege der Platzierung von Bauteilen beleuchtet. Ich bevorzuge die beschriebene Methode mit den Sicherheitsabständen, doch auch die automatische Platzierung eignet sich gut für den schnellen Entwurf einer Platine und zur Überprüfung, wie gut die Bauteile Platz finden. In der nächsten Folge dreht sich alles um die Routing-Fähigkeiten von DesignSpark.

(140046)

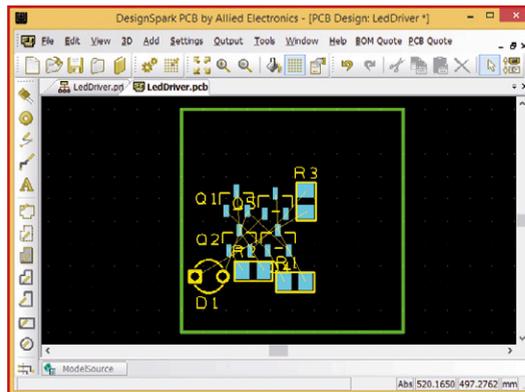


Bild 2. Beispiel der automatischen Platzierung.

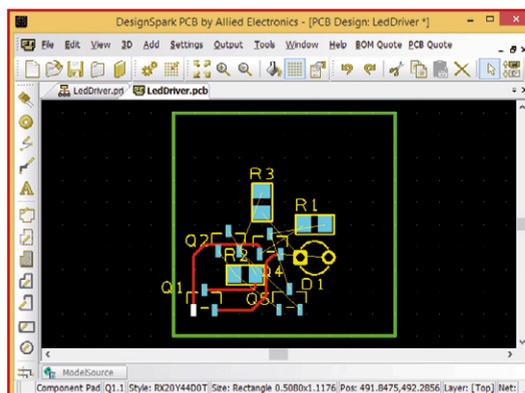


Bild 3. Platzierungsfehler.

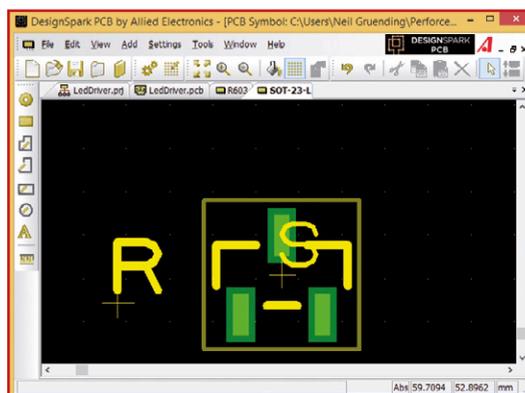


Bild 4. Sicherheitsabstand beim SOT23-Gehäuse.

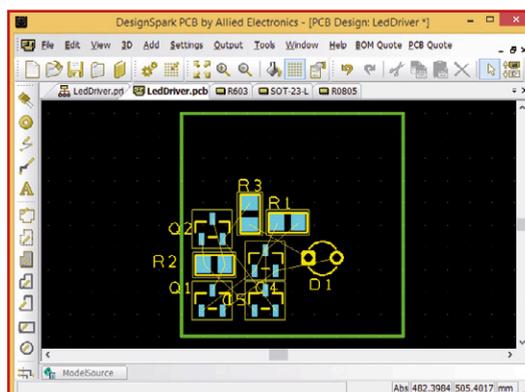


Bild 5. Platine mit „Placement Courtyards“.



Peltier-Module

Seltsame Bauteile (7)

Haben Sie sich jemals über diese kleinen, portablen Kühlplatten gewundert, die ohne bewegliche Teile auskommen? Wundern Sie sich nicht, diese Festkörperkühl-systeme nutzen Peltier-Module. Schauen wir einmal, wie die Teile funktionieren!

Von **Neil Gruending**
(Kanada)

Eine Verbindung aus verschiedenen Metallen bildet ein Thermoelement, das eine kleine, mit der Temperatur variierende Spannung erzeugt. Wenn Sie allerdings eine Spannung an ein Thermoelement anlegen, schaffen Sie damit eine Temperaturdifferenz zwischen Ober- und Unterseite des Elements. Das Phänomen ist als Peltier-Effekt bekannt. Viele kombinierte, stark dotierte p- und n-Silizium-Elemente stellen ein vielseitig nutzbares Peltier- oder Thermoelektrisches Modul dar, dessen Aufbau in **Bild 1** dargestellt ist. Peltier-Module sind natürlich nicht zu 100 % effizient, so dass ihnen Energie zugeführt werden muss, um Wärmeenergie von einer zur anderen Seite zu übertragen. Wenn die Temperaturdifferenz Null ist, muss das Peltier-Modul daher nur die Wärme ableiten, die aufgrund des Betriebs entsteht. Wenn die Temperaturdifferenz aber steigt, tritt eine Grenze auf, was die Ableitung der Wär-

memenge betrifft. In der Praxis ist bei etwa 70° C Temperaturdifferenz Schluss. **Bild 2** zeigt die typische Leistungsfunktion eines Peltierelements. Der Hauptvorteil der Peltier-Module ist ihre Zuverlässigkeit, da keine beweglichen Teile wie bei anderen (mechanischen) Kühlmethoden erforderlich sind. Sie sind in vielen Größen erhältlich und können leicht durch Variation ihrer Eingangsspannung/Strom gesteuert werden. Durch Umkehren des Stromflusses wird auch die Temperaturdifferenz „umgedreht“. Der Nachteil ist, dass Peltier-Module nur etwa 25 % so wirksam sind wie eine mechanische Kühlung, was ihren praktischen Einsatz limitiert. Nachteilig ist auch ihre begrenzte Wärmeübertragungsfähigkeit über weite Temperaturbereiche.

Peltier-Module haben eine weitere einzigartige Funktion: Sie können als thermoelektrische Generatoren verwendet werden, um Wärme unmittelbar in Strom umzuwandeln. Zugegeben nicht besonders effizient, aber sie können schon eine ordentliche Energiemenge umwandeln, je nach Konfiguration. Die große Herausforderung ist dabei der Umgang mit ihrer hohen Ausgangsimpedanz und ihrer begrenzten Wärmeleitfähigkeit. Peltier-Kühlmodule werden in einer Vielzahl von Produkten verwendet: in Kühlboxen, in klimatisierten Autositzen, in wissenschaftlichen Geräten, in Raumfahrzeugen, zur CPU-Kühlung und sogar in High-End-Digitalkameras. Peltier-Generatoren werden in der Ölindustrie, zur Notstrom-Erzeugung und zur Umwandlung von Abwärme in Strom eingesetzt. Ein wirklich schönes Beispiel ist ein mit einer kleinen Flamme betriebener Generator, der als Notfall-Energiequelle zum Aufladen batteriebetriebener Geräte gebraucht wird. Peltier-Module sind im Handel leicht zu finden, allerdings recht teuer. Schauen Sie mal im Restposten-Handel, da können Sie preisgünstige Peltiers finden, mit denen sich nach Belieben experimentieren lässt!

Bild 1.
Aufbau eines Peltier-Moduls
(Zeichner: „michbich“ unter
Creative Commons Lizenz).

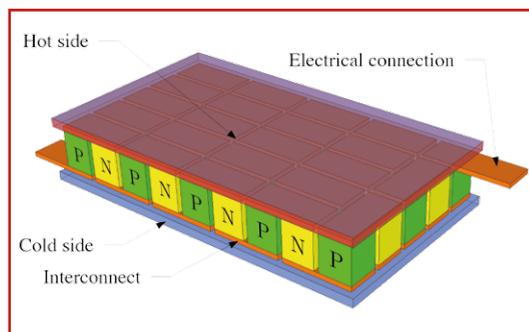
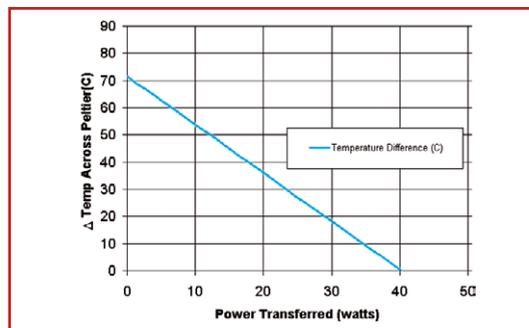


Bild 2.
Leistungskurve eines
Peltier-Elements (Quelle:
heatsinkguide.com).



(140045)

MIT FLEXIBILITÄT MEHR BEWEGEN.

FLEXIBLE LEITERPLATTEN
ONLINE BESTELLEN.

STARR
FLEX
4
bis 4 mZ sig
üblich
Lagen auf Anfrage möglich



LEITON
RECHNEN SIE MIT BESTEM SERVICE

Erfolgreich ist, wer flexibel auf neue Marktanforderungen reagiert. Gefragt sind heute kompakte, komplexe sowie sehr leichte Aufbauten, welche dynamische Biegebelastbarkeit aufweisen und dabei höchste Zuverlässigkeit der elektrischen Verbindungen bieten. Die Lösung lautet **flexible Leiterplatten von LeitOn**. Damit sparen Sie gleich dreimal: **Platzersparnis** durch optimales Anpassen der Baugruppen an die Gehäuse, **Gewichtersparnis** aufgrund sehr dünner Folien sowie **Kostensparnis** wegen der Reduktion von Steckverbindungen. Und Sie gewinnen **mehr Flexibilität** dank persönlicher Beratung am Telefon, einem kompetenten Außendienst und Angeboten auch per E-Mail in Windeseile. Sie können bei LeitOn immer mit bestem Service rechnen.

www.leiton.de

Info-Hotline +49 (0)30 701 73 49 0

Schaeffer
AG



Gratis
Frontplatten
Designer

SIE DESIGNEN – WIR FERTIGEN

Frontplatten in Profiqualität

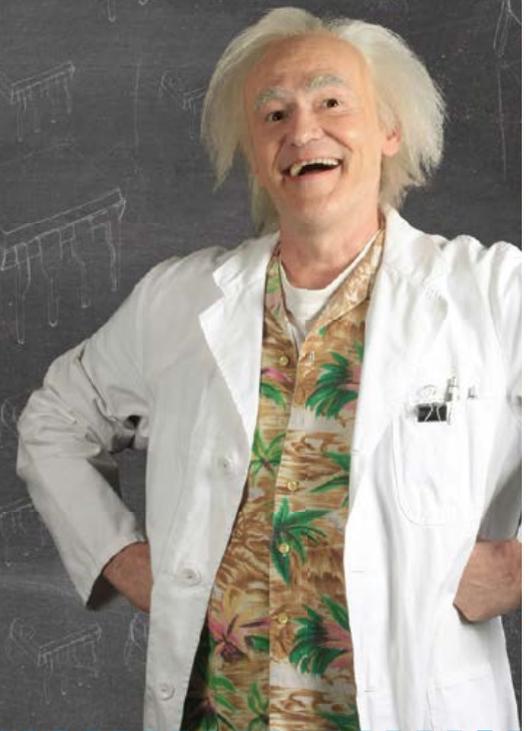
Ab einem Stück und zu einem fairen Preis!
Einfach unseren kostenlosen Frontplatten
Designer auf www.schaeffer-ag.de
herunterladen, Frontplatte entwerfen
und direkt bestellen.

www.schaeffer-ag.de

**Jetzt bekomme
ich noch mehr!**

Neu bei Conrad:
über 600.000 Artikel im Sortiment

- ✓ Deutlich erweitert im Bereich Bauelemente, Entwicklungskits, Messtechnik und Werkzeug
- ✓ Innovative Eigenmarken wie VOLTGRAFT und TOOLGRAFT
- ✓ Viele neue Marken wie Würth Elektronik, Bourns, Microchip Technology, Texas Instruments und Freescale
- ✓ Zertifizierter EPA-Bereich und ESD-Management
- ✓ Und immer: die passenden B2B Services und Leistungen



**Explore
the Future**
Hall A5.462

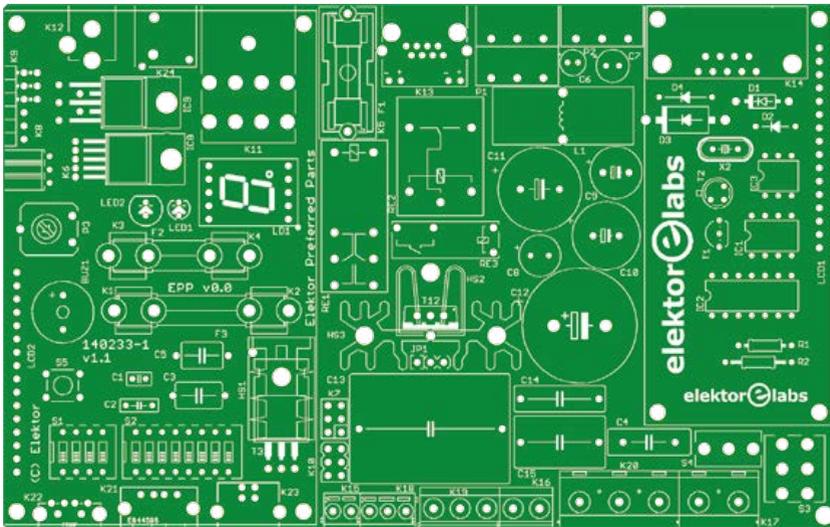
Meet us at electronica 2014
November 11-14 // electronica.conrad.de

CONRAD
Business Supplies

ELPP: Elektor Labs Preferred Parts

Von **Clemens Valens**
(Elektor.Labs)

Wenn Sie sich noch an Elektors goldene Regel „Zwei für alles“ in Form der Akronyme TUP und TUN erinnern können, dann ahnen Sie schon, worum es sich in diesem Beitrag dreht. Wenn nicht: Lesen Sie und staunen...



Kleiner Hinweis: TUP steht für „**T**ransistor **U**niversal **P**NP“ – jetzt sollten Sie auch das Kürzel TUN auflösen können ;-).

In den 1970er und 1980er Jahren war das die universelle Zauberformel für kleine Transistoren in Schaltungen, wo es nicht auf den exakten Typ ankam. Solange man keinen allzu miesen NPN-Kleinsignaltransistor einsetzte, funktionierte die Schaltung. Eine gute Wahl für einen TUN war der Typ BC547 - entsprechend BC557 für einen TUP. Seitdem hat sich viel getan. Auch wenn mein damaliges Transistorhandbuch schon tausende Typen auflistete, so gibt es heute noch viel mehr. Das gilt auch für viele andere Bauteile. Der Entwickler von heute wadet knietief in einem Meer von Opamps, Dioden, Kondensatoren, Mikrocontrollern und anderen (integrierten) Bauelementen. Wie ein Wal durchpflügt er die See und versucht, die Teile aus dem Plankton auszusieben, die für seine Schaltung passen. Die im letzten Jahrzehnt veröffentlichten Schaltungen nutzen völlig andere Bauteile als „damals“. Etliche hat man nicht unbedingt vorrätig.

Im Elektor-Labor ist das Alltag. Bei uns trudeln Schaltungen aus allen Ländern der Welt mit den unterschiedlichsten Bauteilen ein - vom 0815-Teil bis hin zu wahren Exoten; vom japanischen Chip bis zu alten Röhren aus sowjetischer Produktion. Wir schauen dann, mit welchen möglichst weltweit leicht erhältlichen Teilen wir diese ersetzen können. Was in Deutschland kaum zu bekommen ist, kann für australische Leser leicht erhältlich sein. Jedes Mal stellen sich Fragen wie: Hat das Bauteil besondere Eigenschaften oder hatte es der Entwickler gerade in der Schublade? Und wenn es ersetzt werden soll: Mit was? Auch die Entwickler im Elektor-Labor haben elektronische Lieblinge und eine persönliche Bastelkiste sowieso.

Mit den Jahren sammeln sich in diesen Kisten Tütchen mit Bauteilen, die irgendwo übrig blieben, die man aber „bestimmt nochmal brauchen kann“ ;-)...

Irgendwann entschlossen wir uns, eine Liste mit Bauteilen anzufertigen, die wann immer möglich in Elektor-Projekten verwendet werden sollen. So eine Liste bietet viele Vorteile:

- Sie vereinfacht und beschleunigt die Entwicklung, da man nicht jedes Mal wieder hunderte Teile checken und nicht so häufig Bauteil-CAD-Dateien updaten muss.
- Sie reduziert die Länge von Stücklisten und vereinfacht die Lagerhaltung.
- Höhere Stückzahlen reduzieren die Kosten. Distributoren bieten manche Bauteile nämlich nur in 10er- oder 50er-Stückzahlen an.
- Da man so die meisten Bauteile auf Lager hat, kommt man schneller durch die Prototypenphase.
- Gut getestete Bauteile-Layouts für die Platinen vermeiden Nacharbeiten.
- Man weiß, wo man die Bauteile her bekommt und kennt die Bestellnummern.

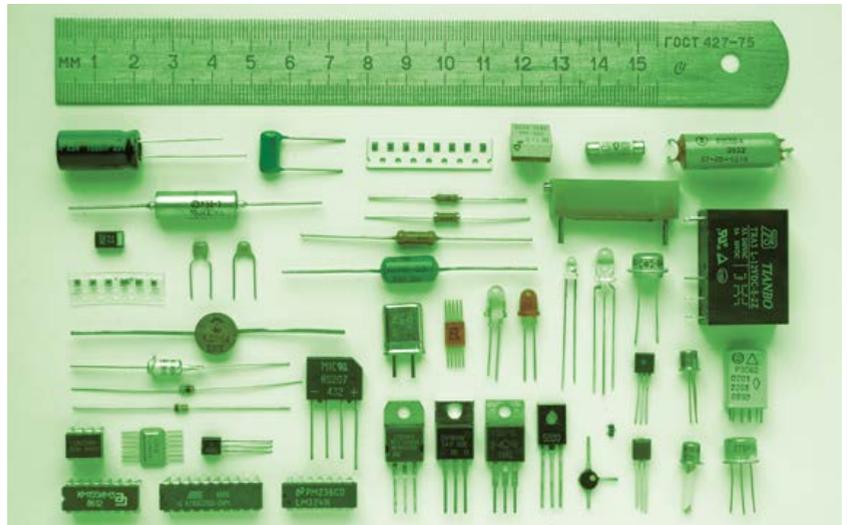
- Man kennt die genauen Spezifikationen der Bauteile. Ein Ersatz erfolgt rational und nicht Pi x Daumen.

Diese Methodik ist vielleicht nicht revolutionär und wird auch in anderen Laboren angewendet, doch so eine Liste auch passend zu den Bedürfnissen unserer Leser zu erstellen ist ein ganzes Stück aufwändiger. Am schwierigsten ist es, die Liste knapp zu halten. Jeder Mitarbeiter möchte ja unbedingt „seine“ Bauteile auf der Liste haben, aber nur „universelle“ Bauteile schaffen die Aufnahme. Außerdem müssen sie auch noch gut erhältlich sein und das möglichst global. Schließlich müssen auch noch optimierte „Footprints“ für die CAD-Library erstellt werden. Es ist sehr zeitaufwändig, die ELPP-List und -Library zu kompilieren und auf Fehler zu überprüfen. Doch hier ist sie – Tataa! – die neueste Liste mit den Elektor Labs Preferred Parts.

Im Vertrauen: Die Liste ist nicht ganz vollständig, denn sie enthält keine SMD-Bauteile. In ELPP Version 1.0 finden sich nur bedrahtete Bauelemente. SMDs werden dann in Version 2.0 aufgenommen. Die gelisteten Bauteile eignen sich für die meisten Elektor-Projekte mit Batterieversorgung bis hin zum Netzbetrieb. Die meisten diskreten Teile wie Dioden und Transistoren kommen in drei Leistungsstufen vor: niedrig, mittel und hoch (hoch meint ≤ 100 W). Mit der Ausnahme von Opamps und Spannungsreglern werden keine ICs aufgeführt. Kondensatoren sind mindestens 50-V-Typen. Widerstandswerte richten sich nach der E12-Reihe – nicht aber Kondensatoren. Nur eine einzige Spule ist aufgeführt: für einen Schaltregler. Auch Mikrocontroller sucht man vergeblich. Dafür gibt es das Drumherum in Form von Quarzen und Sockeln. Für fast alle Einträge haben wir Bestellnummern für die Distributoren Farnell und RS online herausgesucht.

Alle Bauteile sind auch in unserer ELPP-CAD-Library für Eagle aufgeführt, die man von unserer Webseite [1] downloaden kann. In Kürze wird es auch eine CAD-Library für DesignSpark geben. Alle Footprints wurden getestet und sind für das Löten von Hand optimiert. Folglich gibt es extra große Pads und standardisierte metrische Bohrungen. Auch RS- und Farnell-Bestellnummern sind eingepflegt, was das Erstellen einer Stückliste für die Bestellung massiv vereinfacht. Da auch jeder Bauteilwert enthalten ist, kann man sich den Schritt „Add Component Value“ beim Zeichnen der Schaltung sparen (außer man möchte

eine richtig ästhetische Schaltung). Vergessene Werte und inkomplette Stücklisten sind Vergangenheit, denn Sockel, Sicherungen, Jumper und anderer Kleinkram stecken ebenfalls in der Liste. Es gibt keine schwer decodierbaren Bezeichnungen mehr, denn die Library richtet sich nach den Elektor-Konventionen für Bauteilbezeichnungen. Um maximal von unserem ELPP-Ansatz zu profitieren, müssen Sie Ihre Gewohnheiten leicht anpassen. Statt irgendein Bauteil einfach in



ihr Projekt einzufügen, sollten Sie zuvor nachschauen, ob eines aus der ELPP-Library passen würde. Möglicherweise lässt sich für einen errechneten 5,6- μ F-Kondensator doch ein 4,7- oder 10- μ F-Typ einsetzen. Oft klappt das. Auch wenn eine 1.000-V-Diode in einer 3,3-V-Schaltung etwas deplatziert wirkt – schaden tut das nicht.

Denken in Standards erleichtert das Entwickeln. Die ELPP-Liste schränkt aber Entwickler nicht ein. Wenn kein passendes Bauteil in der Liste ist, dann nimmt man halt ein anderes. Die ELPP-Liste ist außerdem nicht in Stein gemeißelt. Dies ist die erste Version. Vielleicht haben wir etwas übersehen oder einen Fehler gemacht. Manchmal laufen Bauteile aus oder verbesserte Versionen verlangen nach einer Evolution der Liste. Auf jeden Fall nutzt das Elektor-Labor für Projekte ELPP-Bauteile wann immer möglich.

(140233)

[1] ELPP Eagle Library, Excel-Datei, Datenblatt: www.elektor-magazine.com/pub/Elektor%20Labs/elektor_labs_preferred_parts_elpp/

Preisgünstiger myDAQ-Anschluss

Von **Thijs Beckers**
(Elektor-Labor)

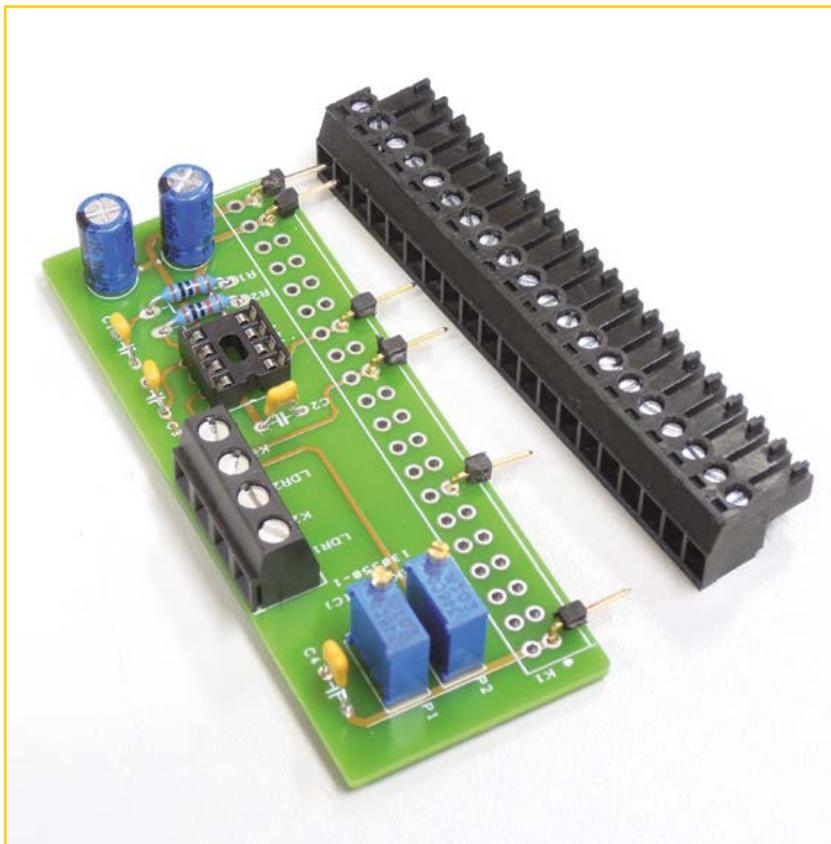
Das Optische Theremin aus der Sommerausgabe 2014 basierte auf einer myDAQ-Datenerfassungseinheit von National Instruments und der „LabVIEW“-Software. Dabei

kommt ein Platinenanschlussklemmen-Block im 3,81-mm-Anschlussraster zum Einsatz, der auf den 0,15-Zoll-Verbinder des myDAQ passt. Da dieser 20-polige Verbinder recht schwer erhältlich war und der Drucktermin des Elektor-Hefts erbarungslos näher rückte, hatten wir uns zu einer „Bastellösung“ entschlossen und zwei 10-polige Grundleisten von Phoenix Contact (Teilenummer 1862658) verwendet, wie im Artikelfoto zu sehen.



Diese Grundleisten sind recht teuer und liegen bei etwa 7 € pro Stück. Das ist nicht gerade eine wirtschaftliche Lösung. Außerdem müssen Sie die Seiten des Kunststoffgehäuses ein wenig abfeilen, damit sie nahe genug aneinander passen (siehe kleines Foto).

Gibt es da nicht eine andere Lösung? Aber sicher! Die besten Ideen entwickelt ein gutes Labor nämlich immer erst nach der Deadline für ein Projekt. Aber wir wollen sie Ihnen dennoch nicht vorenthalten.



Wie Sie auf dem Foto sehen können, haben wir auch einen Prototyp mit dem originalen myDAQ-Stecker (der große schwarze auf dem Foto) aufgebaut, der kostenlos mit jedem myDAQ geliefert wird. Er ist aber nicht zum Löten auf einer Leiterplatte geeignet. Also, wie wird er mit einer Platine verbunden? Dies kann mit einer rechtwinkligen Stiftleiste geschehen, auf die der myDAQ-Stecker geschraubt wird. Glücklicherweise sind die Stifte lang genug, um eingeschoben und durch den Anschlußblock fixiert zu werden.

Schneiden Sie einfach ein paar einzelne Stifte einer gewinkelten 0,1-Zoll-Stiftleiste (2,54 mm) ab und bringen Sie sie an den gewünschten Stellen auf der Platine an. Unser Add-on-Board für das Optische Theremin nutzt nur sechs Verbindungen zum myDAQ, aber Sie können mehr Stifte verwenden, wenn Sie einen stärkeren Halt sicherstellen wollen. Notwendig ist das aber nicht.

(140047)

3D-Drucken kann nützlich sein!

Oder wie man ein Karzinom mit Karzinogenen bekämpft...

In dieser Kolumne (Juni 2014) habe ich den Sinn und Zweck von 3D-Druckern angezweifelt, oder, um genauer zu sein, den Nutzen der erstellten Objekte. Ich habe Sie aufgefordert, mir Beispiele hausgemachter 3D-Drucke zu schicken, von denen Sie glauben, dass sie nützlich sind. Einige Leute nahmen die Herausforderung an und schickten mir Fotos von (und Links zu) den Früchten ihrer Arbeit. Dies zeigte mir zwei Dinge: Erstens, dass das Wort „sinnvoll“ nicht für alle das gleiche bedeutet und zweitens, dass die Vielfalt der Anwendungen für 3D-Drucker riesig ist. Werfen Sie einen Blick auf die Einträge auf Elektor Labs [1] und sehen Sie selbst, wie 3D-Drucker Flugmodelle aus dem Ersten Weltkrieg wiederherstellen und realistische Modelle von Alltagsgegenständen für Modelleisenbahnen generieren (oder sehr kundenspezifische Werkzeuge und Werkstücke). Nützlich sicher, aber nur begrenzt für die Allgemeinheit interessant.

Am MakerSpace56 FabLab in Vannes (in Frankreich, nicht weit von da, wo ich wohne) scheint 3D-Druck eine der Hauptbeschäftigungen zu sein. Dort stieß ich auf eine Anwendung, die neu für mich war: den 3D-Druck von Molekülen. Nicht in ihrer tatsächlichen Größe natürlich, sondern vergrößert. Ziel war es, ein zu entwickelndes Protein, das als „Falle“ für Krebszellen verwendet werden soll, zu visualisieren. Nun, das ist es, was ich unter sinnvoller Anwendung eines 3D-Druckers verstehe, etwas, das helfen kann, ein Problem vieler Menschen zu lösen. Vor ein paar Monaten habe ich an einem 3D-Druck-Wettbewerb von RS Components mitgearbeitet, wo es das Ziel war, gerade solche 3D-Objekte zu „erfinden“. Das Ergebnis dieser Veranstaltung war für mich ein wenig enttäuschend, und ich bin sicher, dass dieses Anti-Krebs-Protein dort die Konkurrenz um Längen geschlagen hätte.

Das RS-Event und viele andere 3D-Druckversuche, die ich erlebt habe, zeigten mir, dass 3D-Drucker noch einen langen Weg zu gehen haben,

bevor sie zum „normalen“ Haushaltsgerät werden. Denn jetzt ist 3D-Druck teuer, sehr langsam und oft geht es schief.

Im eingangs erwähnten Artikel habe ich versprochen, auch einen Preis für den besten 3D-Druck zu spendieren. Da das Protein nicht an diesem Mini-Wettbewerb teilnehmen konnte, entschied ich mich für François-Xavier Dufour als Sieger. Er zeigte (mir), wie er mit einem 3D-Drucker Silikon-Tastaturen erzeugt [2]. Der Nutzen dieser Anwendung ist fraglich, aber ich fand die Idee immerhin als Zwischenschritt zu einer sinnvollen Anwendung interessant. Und ich liebe Gummi-Tastaturen. Herzlichen Glückwunsch, François-Xavier!

(140048)

[1] <http://www.elektor-labs.com/node/4056>

[2] <http://www.elektor-labs.com/node/4104>

Von **Clemens Valens**
(Elektor-Labor)



Das rote 3D-Druck-Knübelchen ist das von der Krebszelle produzierte Feind-Molekül, die weißen Proteine versuchen, das rote zu fangen.



Die Silikontastatur, erstellt von François-Xavier Dufour.

Wenn Sie möchten, können Sie weiterhin Bilder von Ihren nützlichen 3D-Drucken an labs@elektor.com senden oder als Beitrag auf Elektor.Labs einstellen [1].

MagI³C-VDRM

Ein ganz besonderer Spannungsregler

Von
Viacheslav Gromov
(D)

Was assoziiert man zum Thema Spannungsregelung? Z.B. negative Aspekte wie Stabilität oder Rauschen oder diverse Schutzschaltungen zum Schutz der Regler vor Einflüssen von außen. Aber ganz sicher auch die bisherigen Fortschritte: Die Regelungen wurden kleiner, besser und effizienter. Die Entwicklung ist jedoch noch lange nicht zu Ende und besonders bezüglich Low-Power-Fähigkeiten ist noch einiges zu erwarten. Ein Beispiel dafür ist eine neue Familie moderner Abwärtswandler auf Basis des VDRM-Konzepts (**V**ariable Step **D**own **R**egulator **M**odule), die interessante Eigenschaften aufweist.

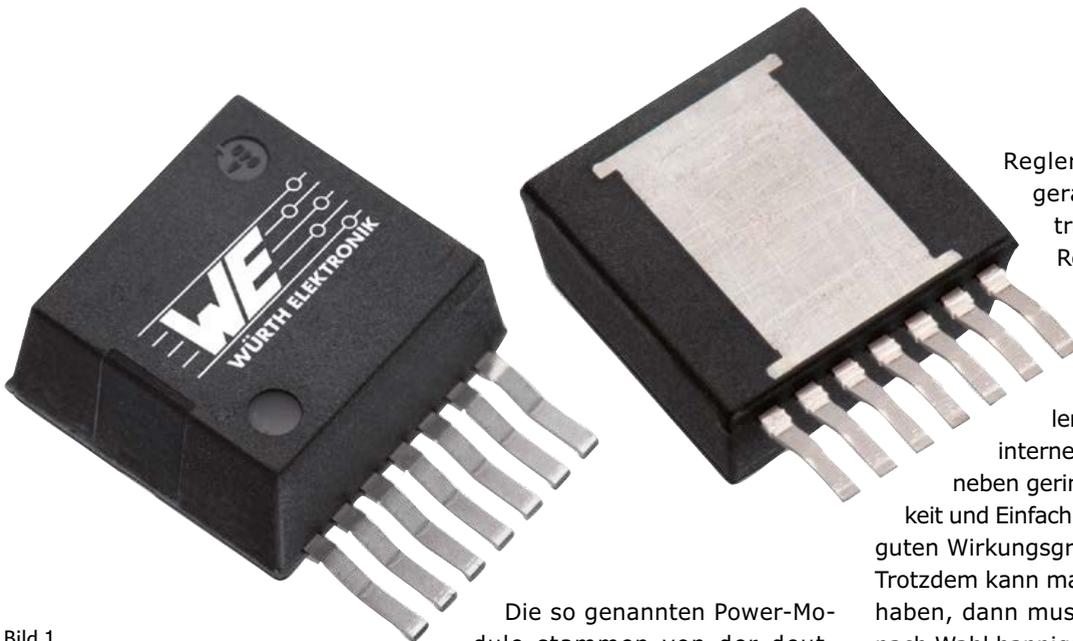


Bild 1.
Die Regler im TO263-7EP-Gehäuse.

Die so genannten Power-Module stammen von der deutschen Firma Würth Elektronik, die sich einen guten Ruf für ihre hochqualitativen passiven Bauteile erarbeitet hat. Diese neue

Regler-Familie ist allerdings nicht gerade preiswert. In diesem Fall trifft also (leider) die bekannte Redensart voll zu, dass Qualität ihren Preis hat. Es sind allerdings auch keine „Billigprodukte“, denn diese ICs und Module sind mit vielen Schutzschaltungen und sogar internen Induktivitäten bestückt, was neben geringem Platzbedarf, Zuverlässigkeit und Einfachheit der Anwendung auch einen guten Wirkungsgrad garantiert.

Trotzdem kann man schlucken: Will man Muster haben, dann muss man schon für drei Module nach Wahl happige 25 Euro auf den Tisch legen. Zusätzlich gibt es für Tests und Experimente ein hochwertiges Evaluations-Board, das ebenfalls in höheren Preisregionen angesiedelt ist.

Tabelle 1. Daten der unterschiedlichen Typen.

Typ	U _{in}	U _{out}	I _{out}	P _{out}	Schaltfrequenz	interne Induktivität	Artikel-Nr. Evaluation Board
WPMDH1200601JT	6...42 V	0,8...6 V	2 A	12 W	0,2...0,8 MHz	10 µH	178 020 601
WPMDH1102401JT	6...42 V	5...24 V	1 A	24 W	0,2...0,8 MHz	15 µH	178 012 401
WPMDM1500602JT	6...36 V	0,8...6 V	5 A	30 W	0,812 MHz	3,3 µH	178 050 601
WPMDH1152401JT	6...42 V	5...24 V	1,5 A	36 W	0,2...0,8 MHz	15 µH	178 012 402
WPMDH1302401JT	6...42 V	5...24 V	3 A	72 W	0,2...0,8 MHz	10 µH	178 032 401

UNSCHLAGBAR

beim Preis-Leistungsverhältnis.



Rigol DS1000E Oszilloskope

2 Kanäle, 50/100 MHz, 1 GSa/s Abtastrate, 1 Millionen Messpunkte Speicher, USB, LAN, einfache Messfunktionen, 3 Jahre Garantie

ab **€ 284,41**
inkl. MwSt. und Versand



Rigol DS1000Z Oszilloskope

4 Kanäle, 70/100 MHz, 1 GSa/s Abtastrate, 12 Millionen Messpunkte Speicher, USB, LAN, professionelle Mess- & Analysefunktionen, optional mit eingebautem Funktionsgenerator, 3 Jahre Garantie

ab **€ 535,50**
inkl. MwSt. und Versand

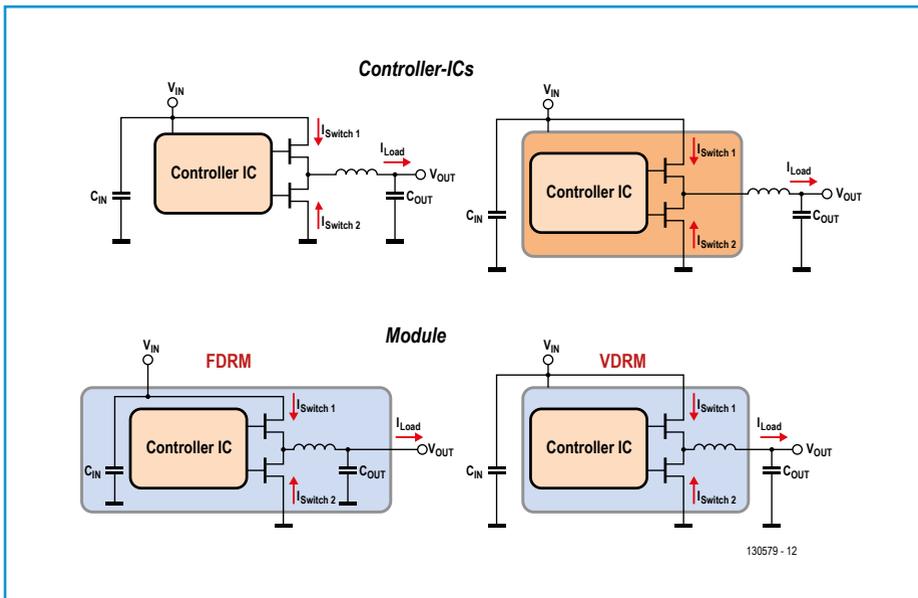


Bild 2. Unterschiede von FDRM und VDRM zu normalen Regler-ICs und -Modulen.

Die Familie

In dieser Regler-Familie gibt es Lösungen für viele Leistungs- und Spannungsbereiche (siehe **Tabelle 1**). Somit ist für fast jede Anwendung etwas dabei. Als Entscheidungshilfe für die Frage, welcher Typ für die jeweilige Anwendung optimal geeignet ist, stehen kostenlose Broschüren und Infomaterial von Würth zur Verfügung [1]. Insgesamt gibt es fünf Typen; alle im gleichen TO263-7EP-Gehäuse (**Bild 1**). Der maximale Wirkungsgrad der Familie beträgt bis zu 97 % und ist von der Leistung und anderen Faktoren abhängig. Der Unterschied dieser Familie zu der FDRM-Familie (**Fixed Step Down Regulator Module**) liegt – abgesehen von der variablen Ausgangsspannung und anderen möglichen Einstellungen der VDRMs - hauptsächlich in den integrierten Eingangs- und Ausgangskondensatoren (siehe die Übersicht in **Bild 2**). Jeder Typ verfügt über Schutzschaltungen gegen Über- und Unterspannung und ist gegen zu hohen Strom und Temperaturen über 165 °C geschützt. Diese Eigenschaften sorgen für eine unkomplizierte Anwendung.

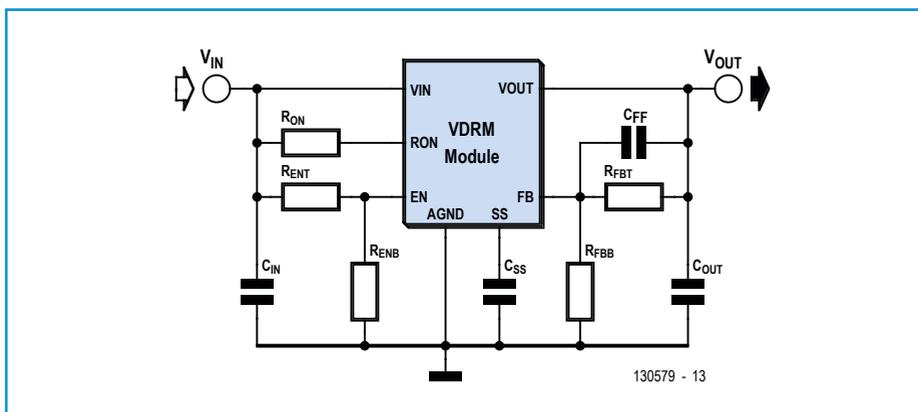


Bild 3. Beschaltung und Innenleben eines VDRM (Ausnahme: WPMDM1500602JT).

Machen Sie Ihr **LEBEN** leichter.
Führende **LABORTECHNIK**
mit **BATRONIX** Zufriedenheitsgarantie

- ✓ Rechnungskauf
100% sicher und schnell. Erst nach Erhalt der Ware zahlen.
- ✓ Große Auswahl ab Lager
- ✓ 30 Tage testen
- ✓ Geld zurück Garantie

Jetzt Angebote nutzen:
www.batronix.com/go/37

NEU

Innereien

Bei den VDRM-Typen kann man durch die Beschaltung (siehe **Bild 3**) mit einem Spannungsteiler aus R_{ENT} und R_{ENB} am EN-Eingang die Abschaltung unterhalb einer bestimmten Spannung einstellen. Diese UVLO-Funktion (**U**nder **V**oltage **L**ockout) schützt z.B. Akkus vor einer Tiefentladung. Mit dem Kondensator C_{SS} am SS-Eingang wird der Verlauf eines Softstarts eingestellt. Die Schaltfrequenz kann (mit einer Ausnahme) mit dem Widerstand R_{ON} am Pin RON im Bereich 0,2...0,8 MHz eingestellt werden.

Die Ausnahme ist der Typ WPMDM1500602JT, der eine Synchronisierung mehrerer Exemplare erlaubt. Hier ist RON durch den Pin SYNC ersetzt, in den eine Schaltfrequenz im Bereich 0,65...0,95 MHz aus einer externen Quelle eingespeist werden kann. Wenn man diesen Pin auf Masse legt, stellt sich eine Schaltfrequenz von 0,812 MHz ein. Die Ausgangsspannung wird durch den Spannungsteiler aus R_{FBT} und R_{FBB} am Feedback-Pin FB festgelegt. Zur Berechnung der diversen Bauteilewerten sei auf die Datenblätter verwiesen, die man unter [1] downloaden kann. Dort finden sich neben Formeln fertige Tabellen für verschiedene Ausgangsspannungen und vieles mehr.

Außerdem

Die Betriebstemperatur liegt im Bereich -40...125 °C. Schaltregler benötigen zur Siebung und für geringe Störpegel passend dimensionierte Kondensatoren am Ein- und Ausgang. Wie bei

anderen Reglern auch sollten sie einen möglichst kleinen ESR aufweisen. Vielschicht-Keramik Kondensatoren mit X7R- und X5R-Dielektrikum oder entsprechende Tantal-Kondensatoren eignen sich dafür gut. Wie man in Tabelle 1 sieht, sind die zulässigen Eingangsspannungen und Schaltfrequenzen bis auf die schon erwähnte Ausnahme bei allen Typen gleich. Ein Blick auf die Blockdiagramme von Bild 2 zeigt, wie die interne Induktivität und die externen Kondensatoren beschaltet sind. Eingezeichnet sind auch die integrierten MOSFET-Transistoren, die bei konventionellen Schaltregler-ICs extern beschaltet werden müssen. Im Datenblatt werden auch ein paar Worte zum Platinen-Layout gesagt, die man tunlichst ernst nehmen sollte, um möglichst geringe elektromagnetische Interferenzen zu erzielen. Wichtig ist natürlich, das Masse-Pad des ICs auch gut mit Masse verbunden ist und die Feedback-Leitung möglichst kurz gehalten wird. Wenn man lediglich die üblichen Ausgangsspannungen benötigt, eignen sich auch die FDRM-Typen mit fester Ausgangsspannung und kleinerem Gehäuse, die zudem den Ein- und Ausgangskondensator schon integriert haben.

Wer Serien plant oder die Eigenschaften dieser kompakten Schaltregler-Module genau untersuchen will, dem seien die verschiedenen Evaluation-Boards (**Bild 4**) empfohlen.

(130579)

Weblink

[1] <http://katalog.we-online.de/de/pm/MagIC-VDRM>

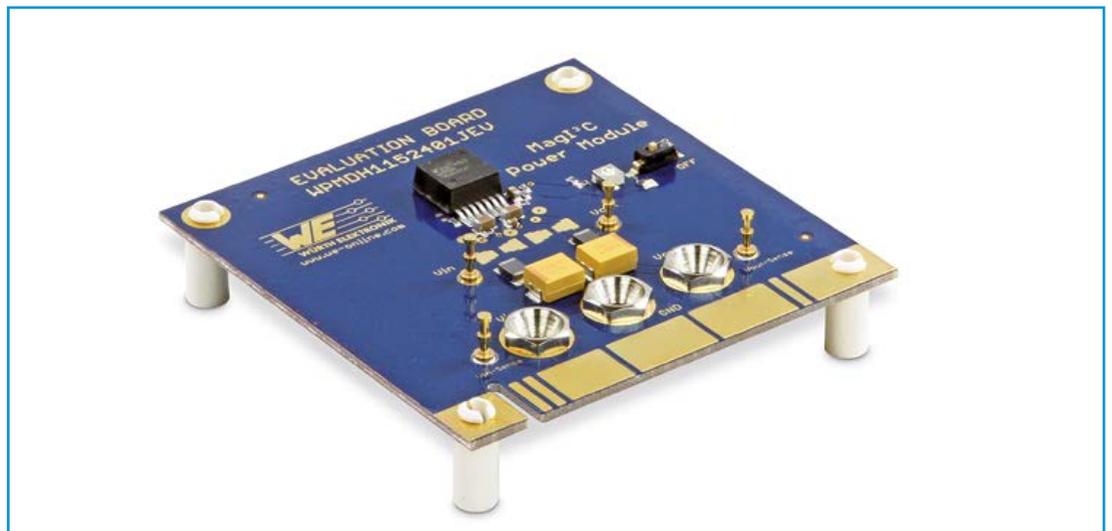


Bild 4.
Ein Evaluation-Board.



'The Quintessence' of Cybersecurity

EBV Elektronik präsentiert die vierzehnte Ausgabe des Wissensmagazins 'The Quintessence'

Auch als
Tablet-App
erhältlich
TQ by EBV



Inhalte von 'The Quintessence' of Cybersecurity sind unter anderem:

- Im Gespräch mit Marc Elsberg:
Autor des Technik-Thrillers „Blackout“
- Auf dem Radar der Hacker: Angriffsziel M2M-Kommunikation
- Innovativer als die Produktpiraten: Immer mehr Schäden durch Plagiate
- Experten-Gespräch: Cybersecurity als Herausforderung und Chance

Interessiert?

Bestellen Sie sich Ihr persönliches Exemplar von 'The Quintessence' gedruckt oder als umweltfreundliche eBrochure unter ebv.com/tq.

Weitere Produkt-Highlights und Market-Updates finden Sie im EBV Newsletter – einfach registrieren unter ebv.com/newsletter.

Distribution is today. Tomorrow is EBV!

www.ebv.com/de

 **EBV Elektronik**
| An Avnet Company |

LED-Klassifizierungs-Tool

Von **Henry Morizot**
(Frankreich)

Dieses Projekt ist ein Spin-off der einstellbaren Präzisions-Stromquelle, die wir Anfang dieses Jahres veröffentlichten [1]. Gleichzeitig ist es ein tolles Hilfsmittel, um durch visuellen Vergleich bedrahtete LEDs nach ihrer Farbe, der Helligkeit und/oder dem Abstrahlwinkel zu sortieren.

Obwohl sie alle identisch aussehen, können LEDs ganz verschieden sein, nämlich wenn sie leuchten und vor allem, wenn sie etwa in einem LED-Balken nahe beieinander montiert sind. In den „frühen LED-Tagen“ waren Farb- und Helligkeitsunterschiede auch beim gleichen Hersteller, ja selbst bei gleicher Type und Produktionscharge an der Tagesordnung.

Zwar haben sich die Produktionsprozesse gebessert, aber jeder Elektroniker hat doch ein mehr oder weniger großes Kästchen im Labor, angefüllt mit LEDs unbekannter Typs. Solche „Fremdkörper“ kann man nebeneinander in das hier vorgestellte Accessoire stecken und als nächstes die Präzisionsstromquelle auf verschiedene Werte einstellen. In der „Lichtleiste“ lassen sich dann selbst kleine Unterschiede in Farbe, Helligkeit und Abstrahlwinkel mit Sicherheit entdecken.

Die maximale Klemmenspannung der einstellbaren Präzisionsstromquelle liegt bei etwa 20 V, womit bis zu zehn „normale“ LEDs in Serie betrieben werden können. Die genaue Anzahl hängt von der Art und der Farbe ab, eine rote LED benötigt typischerweise eine Vorwärtsspannung von etwa 1,8 V, während es bei blauen LEDs schon mal 3 V sein können.

Stellen Sie die Klemmenspannung nun auf das Maximum ein und den Strom auf den Bereich 1 mA .. 20 mA.

Beginnen Sie mit der Referenz-LED in der äußersten linken Buchsenposition, mit der Kathode (kurzer Anschluß) links, und stecken Sie die anderen LEDs in gleicher Weise in die benachbarten Positionen. Ein Jumper muss auf die beiden Stifte in der Position gesteckt werden, an der sich die am weitesten rechts liegende LED befindet. Alle

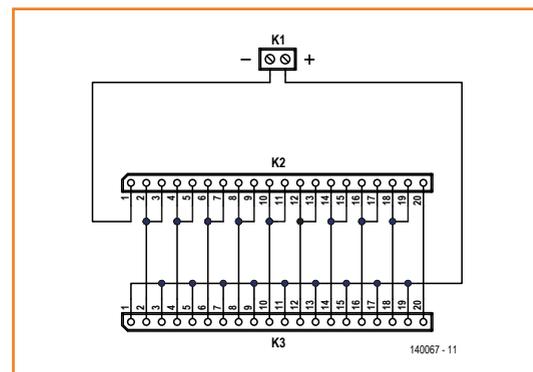
LEDs sollten dann leuchten. Die LC-Anzeige der Stromquelle zeigt die Vorwärtsspannung aller LEDs der Reihe an. Sie können auch die Vorwärtsspannung einer LED bestimmen, indem sie eine einzelne oder auch mehrere LEDs durch Jumper ausschalten.

Wenn der Clamping-Indikator auf der Frontplatte des Geräts aufleuchtet, überschreitet die Vorwärtsspannung der LED-Kette die Möglichkeiten der Stromquelle (Sie haben doch den Knopf ganz nach rechts gedreht - oder?).

(140067)

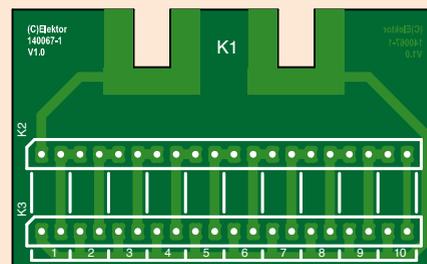
Weblink

[1] [Einstellbare Präzisionsstromquelle: Elektor April 2014, www.elektor-magazine.de/130287](http://www.elektor-magazine.de/130287)



Stückliste

- K1 = Löt pads und Ausschnitte in der Platine
- K2 = 20-polige Buchsenleiste, RM 0,1"
- K3 = 20-polige Steckerleiste, RM 0,1"
- JP = Jumper
- Platine 140067-1



Lesen Sie die neue Elektor ein Jahr lang in der ultimativen GOLD-Mitgliedschaft und profitieren Sie von allen Premium-Vorteilen!



Die Elektor-GOLD-Jahresmitgliedschaft bietet Ihnen folgende Leistungen/Vorteile:

- Sie erhalten **10 Elektor-Hefte** (8 Einzelhefte + 2 Doppelausgaben Januar/Februar und Juli/August) pünktlich und zuverlässig frei Haus.
- **Extra:** Jedes Heft steht Ihnen außerdem als PDF zum sofortigen Download unter www.elektor-magazine.de (für PC/Notebook) oder via App (für Tablet) bereit.
- **Neu & Exklusiv:** Sie erhalten alle 2 Wochen per E-Mail ein neues Extra-Schaltungsprojekt (frisch aus dem Elektor-Labor).
- **Neu & Exklusiv:** Wir gewähren Ihnen bei jeder Online-Bestellung 10% Rabatt auf alle unsere Webshop-Produkte – dauerhaft!
- **Neu & Exklusiv:** Der Online-Zugang zum neuen Community-Bereich www.elektor-labs.com bietet Ihnen zusätzliche Bauprojekte und Schaltungsideen.
- **Extra:** Die neue Elektor-Jahrgangs-DVD (Wert: 27,50 €) ist bereits im Mitgliedsbeitrag inbegriffen. Diese DVD schicken wir Ihnen sofort nach Erscheinen automatisch zu.
- **Extra:** Top-Wunschprämie (im Wert von 30 €) gibts als Dankeschön GRATIS obendrauf!

UMWELTSCHONEND – GÜNSTIG – GREEN

Möchten Sie Elektor lieber im elektronischen Format beziehen? Dann ist die neue GREEN-Mitgliedschaft ideal für Sie! Die GREEN-Mitgliedschaft bietet (abgesehen von den 10 Printausgaben) alle Leistungen und Vorteile der GOLD-Mitgliedschaft.



Jetzt Mitglied werden unter www.elektor.de/mitglied!

Mikrocontroller für Einsteiger (5)

Timer-Anwendungen



Von **Burkhard Kainka**

Flipflops, Teiler und Zähler sind wichtige Elemente der digitalen Elektronik und somit auch von Mikrocontrollern. Und damit lässt sich eine Menge anfangen. Diese Folge zeigt typische Anwendungen.

Eines der größten Kapitel im Datenblatt des ATmega328 behandelt die drei Timer des Controllers. Ihre Anwendungen sind so vielfältig, dass hier nur ein kleiner Teil der Möglichkeiten vorgestellt werden kann. Die wesentlichen Aufgabenfelder heißen Zeitmessung, Frequenzmessung, Signalerzeugung und PWM-Ausgabe.

Mikrosekunden messen

Zeiten messen ist immer wieder eine wichtige Aufgabe in der Elektronik. Eine typische Fragestellung: Wie lange dauert eigentlich so eine LCD-Ausgabe? Geht es hier um Millisekunden oder um Mikrosekunden? Keine Ahnung, das müsste mal gemessen werden. Und einen passenden Zeitmesser hat der Mega328 ja an Bord, den Timer1 mit einer Auflösung von 16 bit. So ein Timer ist

nichts anderes als ein Zähler, der regelmäßige Ereignisse zählt. In diesem Fall kann er bis 65535 (16 bit) zählen, danach springt er wieder auf 0 (Überlauf). Wenn man ihn so einstellen könnte, dass er genau jede Mikrosekunde hochzählt, könnte man bis 65535 μ s messen. Zwischen dem Quarzoszillator und dem Zählereingang gibt es einen umschaltbaren Vorteiler (ähnlich wie beim AD-Wandler), der das Taktsignal herunterteilt. Bei einem Quarztakt von 16 MHz müsste dieser auf 16 eingestellt sein, damit 1 MHz am Zählereingang ankommen. Einfach mal versuchen: `Config Timer1 = Timer , Prescale = 16`

Das geht aber leider nicht. Bascom bringt eine Fehlermeldung: "Prescale value must be 1,8,64,256 or 1024" Das Datenblatt des Mega328 bestätigt diese Aus-

sage. Na gut, dann eben Vorteiler 8, Eingangstakt 2 MHz, Auflösung 0,5 μ s (vgl. **Bild 1**). Die Zeitmessung reicht dann bis 32767,5 μ s. Das 16-bit-Register Timer1 kann jederzeit ausgelesen oder mit einem neuen Wert belegt werden. Damit ist die Sache einfach. Vor der LCD-Ausgabe setzt man den Zähler auf Null (Timer1 = 0). Nach der LED-Ausgabe liest man den Zähler aus (D = Timer1) und hat dann das Ergebnis der Zeitmessung in der Einheit 0,5 μ s. Oder man teilt das Ergebnis noch einmal durch 2 und bekommt dann glatte Mikrosekunden. **Listing 1** zeigt das fertige Programm.

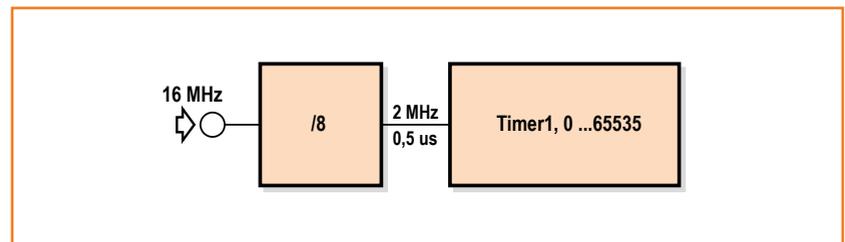
Das Ergebnis wird hier (wie auch in den folgenden Programmen) auf dem LCD des in der letzten Folge vorgestellten Arduino-Shields angezeigt. Gleichzeitig wird der Wert mit dem Print-Befehl an das Terminal geschickt. Das Programm funktioniert auch, wenn kein LCD angeschlossen ist, denn Bascom fragt nicht nach, ob die Daten vom Display wirklich angenommen wurden. Die erste Ausgabe lautet „Timer 1 = 0 us“, aber das liegt natürlich daran, dass die Variable D vor der ersten Messung noch leer ist. Die zweite Ansage sieht schon anders aus und liefert ein Ergebnis für die LCD-Aktivitäten der ersten Anzeige: „Timer 1 = 17105 us“. Jetzt mussten vier Zeichen mehr angezeigt werden, das wird wohl etwas länger gedauert haben. Und tatsächlich, die Zeitmessung stellt sich auf 21933 μ s ein. Und da bleibt sie dann auch stehen. Jede neue Messung liefert wieder das gleiche Ergebnis mit einer Abweichung von maximal einer Mikrosekunde.

So genau kann man also sagen, wie lange ein komplexer Vorgang wie die LCD-Ausgabe dauert. Ein Linux- oder Windows-Programmierer würde glauben er sei schon im Himmel, denn bei solch großen Systemen ist es mit der Echtzeitfähigkeit nicht weit her. Da laufen ständig irgendwelche Sachen im Hintergrund ab und man kann nie ganz genau sagen, wie lange etwas dauert. Aber in Bascom ist das anders. Es läuft genau das, was man hingeschrieben hat, nichts anderes. Man ist also sehr nahe an der Hardware, und manche Dinge wie zum Beispiel eine einfache Portausgabe sind in weniger als einer Mikrosekunde erledigt. Die LCD-Ausgabe dauert mit rund einer Millisekunde pro Zeichen nur deshalb etwas länger, weil Bascom Rücksicht auf den LCD-Controller nimmt und ihm genügend Zeit lässt. Wenn Sie wollen können Sie die LCD-Ausgabe mal mit einem Oszilloskop am E-Pin des LCD (Portpin PD3,

Arduino-Pin 3) belauschen, da kommt ebenfalls ungefähr 1 ms zum Vorschein.

Messung von Periodendauern

Mit einer kleinen Änderung kann das Programm auch Impulslängen oder -Perioden (**Listing 2**) messen. Hier wird der Eingang PC0 (Arduino A0, vgl. **Bild 2**) verwendet. Eine Periode der Netzspannung sollte im Idealfall 20 ms lang sein (in den USA 16,667 ms). Gemessen wird die Zeitspanne zwischen zwei positiven Flanken des Eingangssignals. Um eine Flanke eindeutig zu erkennen braucht man zwei Abfrageschleifen. Zuerst wartet man bis der Eingangspegel Null ist, dann wartet man bis er auf Eins wechselt. Genau zu der Zeit gab es eine positive Flanke. Nun wird



Listing 1. Messung der LCD-Ausgabezeit.

```

|-----|
'UNO_Timer1.BAS  Timer1 0.5 us
|-----|
...

Dim D As Word
Config Timer1 = Timer , Prescale = 8
'Clock 2 MHz

Do
  Print "Timer1 = ";
  Print D;
  Print " us"
  Timer1 = 0
  Locate 1 , 1
  Lcd "Timer1 ="
  Locate 2 , 1
  Lcd D
  Lcd " us"
  D = Timer1
  D = D / 2
  Waitms 1000
Loop

```

Bild 1.
Zeitmessung mit dem
Timer1.

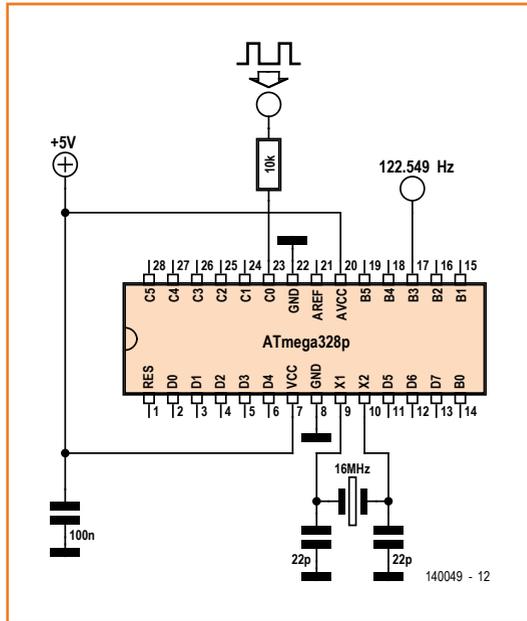


Bild 2.
Messung von
Periodendauern.

der Zeitgeber auf Null gesetzt. Dann wartet man bis zur nächsten Flanke, um schließlich den Zeitgeber auszulesen.

Nun kann man wie gehabt den Finger an den Eingang legen, um 50-Hz-Signale einzufangen. Und tatsächlich, es werden rund 20.000 μ s angezeigt. Tatsächlich haben wir bei einer realen Messung im Durchschnitt etwa 20.030 μ s gemessen. Die Generatoren aller europäischen Kraftwerke laufen ja phasensynchron. Manchmal laufen sie

offensichtlich alle etwas zu langsam. Zu viele Kraftwerke abgeschaltet, die Sonne scheint nicht und der Wind ist gerade flau. Also lieber mal ein paar Lampen ausschalten und nachmessen ob es besser wird...

Damit man gleich auch noch etwas richtig Zuverlässiges zu messen hat, bringt das Programm seine eigene Taktquelle mit. Dafür hat man ja die PWM-Ausgänge. Nur leider gibt es da eine Kollision. PWM1A und PWM1B können nicht eingesetzt werden, weil der Timer1 ja gerade mit der Zeitmessung beauftragt ist. Da gibt es dann noch den Timer0 und den Timer2, und jeder von ihnen kann ebenfalls zwei PWM-Ausgänge bedienen. Allerdings liegen die meisten dieser zusätzlichen PWM-Ausgänge am Port D und würden damit die LCD-Ausgabe stören. Eine Ausnahme bildet der Ausgang PWM2A am Portpin PB3 (Arduino-Pin 11). Und hier soll ein Signal ausgegeben werden. Da es sich um einen 8-bit-Timer handelt und der Vorteiler auf 256 eingestellt wurde, beträgt die Ausgangsfrequenz $16.000.000 \text{ Hz} / 256 / 255 / 2 = 122,549 \text{ Hz}$. Die Periodendauer ist dann 8,16 ms. Eine Verbindung zwischen PB3 und PC0 erlaubt die Messung des eigenen Signals. Das Ergebnis lautet 8160 μ s. Passt!

Rechteckgenerator 125 Hz bis 4 MHz

Oft braucht man einen Signalgenerator mit einstellbarer Frequenz. Mal soll es 1 MHz sein um einen Frequenzzähler zu überprüfen, mal 440 Hz,

Listing 2. Periodendauer in Mikrosekunden messen.

```

'-----
'UNO_Timer2.BAS Timer1, 0.5 us
'-----
...

Dim D As Word

Config Timer1 = Timer , Prescale = 8 'Clock 2 MHz
Config Timer2 = Pwm , Prescale = 256 ,
    Compare A Pwm = Clear Up
Ddrb = 255
Pwm2a = 128

Do
    Do
        Loop Until Pinc.0 = 0
    Do
        Loop Until Pinc.0 = 1

Timer1 = 0
Do
Loop Until Pinc.0 = 0
Do
Loop Until Pinc.0 = 1
D = Timer1
D = D / 2
Locate 1 , 1
Lcd "Timer1 ="
Locate 2 , 1
Lcd D
Lcd " us "
Print "Timer1 = ";
Print D;
Print " us"
Waitms 1000
Loop
    
```

Listing 3. Timer1 als Rechteckgenerator.

```

'-----
'UNO_Timer3.BAS B1 Fout 250 Hz...4 MHz
'-----
...

Dim D As Long
Dim F As Long
Dim N As Byte

Config Timer1 = Pwm , Prescale = 1 , Pwm = 10 ,
    Compare A Pwm = Clear Up

Tccr1a = &B10000010 'Phase-correct PWM, Top=ICR1
Tccr1b = &B00010001 'Prescaler=1

D = 8000
Do
    N = Ischarwaiting()
    If N = 1 Then
        Input F
        D = 8000000 / F
        If D > 64000 Then D = 64000
        If D < 2 Then D = 2
    End If

    If S1 = 0 Then
        D = D + 1
        If D > 100 Then D = D + 1
        If D > 1000 Then D = D + 100
        If D > 10000 Then D = D + 1000
        If D > 64000 Then D = 64000
    End If

    If S2 = 0 Then
        If D > 2 Then D = D - 1
        If D > 100 Then D = D - 10
        If D > 1000 Then D = D - 100
        If D > 10000 Then D = D - 1000
        If D > 64000 Then D = 64000
    End If

    Locate 1 , 1
    F = 16000000 / D
    F = F / 2
    Lcd F
    Lcd " Hz      "
    Icr1 = D
    Ocr1a = D / 2
    Waitms 50
Loop

```

um eine Geige zu stimmen. Den ganzen Bereich von 125 Hz bis 4 MHz erfasst das Programm in **Listing 3**. Timer1 wird hier als Frequenzteiler verwendet, das Signal entsteht am PWM1A-Pin (B1, Arduino-Pin 9, vgl. **Bild 3**). Die Anwendung stellt ein Beispiel dafür dar, wie man in Bascom etwas machen kann, was eigentlich nicht vorgesehen war. Es gibt nämlich keine vorgefertigte Initialisierung für diesen Zweck. Man muss einige Register direkt programmieren. Wie man das macht, habe ich beim DCF-Simulator von Roger Leifert [1] abgeschaut, und er hat das Verfahren vom SDR-Kurs von Martin Ossmann [2] übernommen, wo es in C programmiert wurde. Kurz gesagt ist es eine Abwandlung der PWM-Ausgabe, aber mit frei einstellbarer Frequenz und jeweils ungefähr 50 % Tastverhältnis. Das Register LCR1 wird mit dem gewünschten Teilerfaktor geladen. Lädt man hier den Wert 100 hinein, dann ergibt sich die Frequenz $16.000.000 \text{ Hz} / 100 / 2 = 80.000 \text{ Hz}$. Damit dann ein symmetrisches Rechtecksignal herauskommt, muss man das Register OCR1A entsprechend mit dem Wert 50 beladen. Das Programm kann (gleichzeitig) auf zwei Arten bedient werden, über die serielle Schnittstelle und

ein Terminalprogramm oder über die Taster. Die Taster vergrößern oder verkleinern direkt den Teilerfaktor. Was dann am Ende dabei herauskommt, wird ausgerechnet und angezeigt. Wieder kann man durch kurzes Antippen den Wert in kleinen Schritten verändern oder durch einen langen

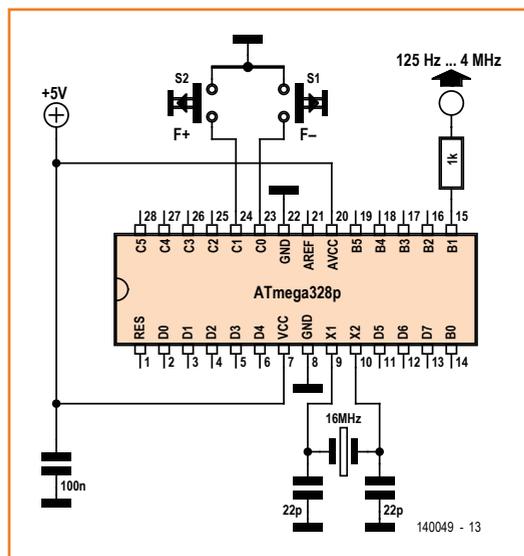


Bild 3. Einstellbarer Rechteckgenerator.

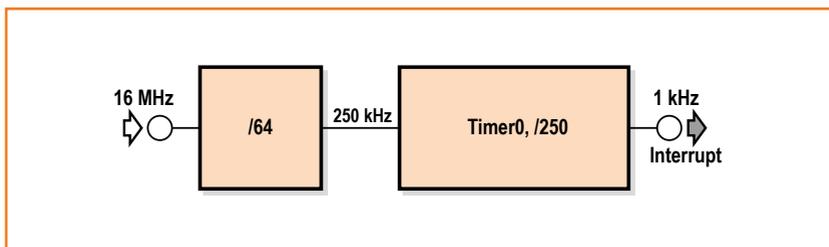


Bild 4.
1 kHz aus 16 MHz.

Tastendruck quasi kontinuierliche Änderungen erzeugen. Damit die ganze Sache bei mehr als 65000 möglichen Werten nicht zu lange dauert, wurden je nach Bereich unterschiedliche Schritt-

Listing 4. Genauere Sekunden im Interrupt.

```

'-----
'UNO_Timer4.BAS
'Timer0-Interrupt, Seconds
'-----
...

Dim Ticks As Word
Dim Seconds As Word
Dim Seconds_old As Word

Config Timer0 = Timer , Prescale = 64
On OvF0 Tim0_isr
Enable Timer0
Enable Interrupts

Do
  If Seconds <> Seconds_old Then
    Print Seconds
    Seconds_old = Seconds
    Locate 1 , 1
    Lcd Seconds
  End If
Loop

Tim0_isr:
  '1000 µs
  Timer0 = 6
  Ticks = Ticks + 1
  If Ticks = 1000 Then
    Ticks = 0
    Seconds = Seconds + 1
  End If
Return

End
  
```

weiten vorgesehen. Das wiederum führt dann bei manchen Wunschfrequenzen dazu, dass man sie nicht genau, sondern nur ungefähr erreicht. Die Frequenz wird in Hz angezeigt; man kommt daher nicht mehr mit einer Word-Variablen aus. Verwendet wird Long, also ein Typ mit 32 bit, sowohl für die Frequenz F als auch für den Teilerfaktor D. Nur so funktioniert auch die Umrechnung.

Dass das Programm sowohl auf die Tasten als auch auf serielle Eingaben reagieren kann ist nicht ganz selbstverständlich. Schreibt man einfach Input F, dann wartet das Programm an dieser Stelle, bis tatsächlich eine Eingabe kommt. Dann wird das nichts mit den Tasten. Man muss also erstmal vorsichtig nachfragen, ob überhaupt etwas über die serielle Schnittstelle reinkommt. Die Funktion IsCharWaiting() liefert den Wert 1, wenn bereits mindestens ein Zeichen angekommen ist, ansonsten 0. Wenn hier aber ein Zeichen angekommen ist, dann kommen auch mehr. Dann kann man getrost den neuen Wert F abholen. Aus ihm wird dann der Teilerfaktor berechnet, was in vielen Fällen zum genauen Ergebnis führt. 440 Hz gehen jedenfalls ohne nennenswerte Abweichung. Bei 549 Hz bekommt man dagegen nur noch eine ähnliche Frequenz. Es ergibt sich nämlich ein ganzzahliger Teilerfaktor von 14, und daraus werden dann 571,428 Hz. Die kleinsten Teilerfaktoren liefern die höchsten Frequenzen, und zwar 4 MHz, 2,666666 MHz, 2 MHz, 1,6 MHz, 1,333333 MHz und 1 MHz. Brauchbar? Dann bauen Sie es auf!

Die steilflankigen Rechtecksignale am Ausgang PB1 enthalten übrigens Oberwellen bis weit in den UKW-Bereich. Da muss man schon aufpassen, dass der Signalgenerator nicht versehentlich zu einem Störsender wird. Man hält z.B. das Messkabel des Oszilloskops an den Ausgang, hat aber vergessen, auch die Masseleitung auf kurzem Wege anzuschließen. Masse liegt ja aber über den PC und den USB-Anschluss schon an. Nun hat man eine große Schleife gebildet, die als wirksame UKW-Antenne arbeitet. Da muss man sich nicht wundern, wenn plötzlich der FM-Radioempfang versagt. Und der der Nachbarn auch, was erheblichen Ärger einbringen kann. Um die EMV-Richtlinien einzuhalten muss man entweder abgeschirmte Kabel verwenden oder wenigstens einen Widerstand nahe am Ausgang einfügen, der dann die Oberwellenabstrahlung stark dämpft. Ein Widerstand von 1 kΩ bildet zusammen mit einer angenommenen Kapazität des Kabels von 30 pF einen Tiefpass mit einer Grenzfrequenz

von 5 MHz. Die Flanken werden soweit abgeflacht, dass UKW-Störungen effektiv um etwa 20 dB gedämpft werden.

Timer-Interrupt

In der letzten Folge gab es bereits ein Beispielprogramm, das einen Sekudentakt erzeugte. Es wurde einfach „Waitms 1000“ geschrieben, fertig war die Sekunde. Nur kann das leider nicht ganz exakt sein, denn es kommen noch Programmlaufzeiten für die Endlosschleife und für die Ausgabe hinzu. Aus diesem Dilemma hilft ein Timer-Interrupt, also eine vom Timer gesteuerte Programmunterbrechung. Und das funktioniert so: Ein Hardware-Timer zählt völlig unbeeinträchtigt vom übrigen Geschehen im Mikrocontroller vor sich hin. Immer wenn er den höchsten Zählerstand erreicht hat und auf Null überspringt (Timer-Überlauf), wird automatisch das laufende Programm unterbrochen und ein spezielles Unterprogramm (Interrupt-Service-Routine = ISR) aufgerufen. Hier kann man dann Dinge tun, die in einem genauen Zeittakt ablaufen sollen. Wie lange das jeweils dauert ist relativ egal, denn der Hardware-Zähler zählt unterdessen stur weiter. Man muss nur rechtzeitig vor dem nächsten Überlauf dieses Interrupt-Unterprogramm wieder verlassen.

Hier soll ein präziser Takt von einer Sekunde erzeugt werden. Für diese Aufgabe ist der Timer1 zu schade. Es reicht schon der Timer0 mit einer Auflösung von 8 bit. Der Timer soll alle 1000 μ s einen Überlauf erzeugen und damit einen Interrupt auslösen. Die Interrupt-Service-Routine wird also jede Millisekunde angesprungen; da es sich um einen Interrupt von Timer0 handelt, haben wir sie `Tim0_isr`: genannt. Der Doppelpunkt macht die Bezeichnung im Programmcode zu einem Label, das während des Programmlaufs angesprungen werden kann. „`On OvF0 Tim0_isr`“ sagt dem Controller, dass er das Label anspringen soll, wenn ein Überlauf (`Ovf` = Overflow) am Timer0 stattfindet. Die Interrupt-Routine muss mit `Return` beendet werden. Erst danach kann ein weiterer Interrupt ausgeführt werden.

Das Beispielprogramm (**Listing 4**) initialisiert den Timer0 mit einem Vorteiler von 64, er erhält also einen Takt von 250 kHz (**Bild 4**). Ohne besondere Maßnahmen würde auf jeweils 256 Taktimpulse ein Überlauf kommen, weil die Zählerbreite 8 bit beträgt. Damit nach jeweils genau 250 Takten der nächste Überlauf kommt, wird der Zähler ganz am Anfang der Interrupt-Routine auf den

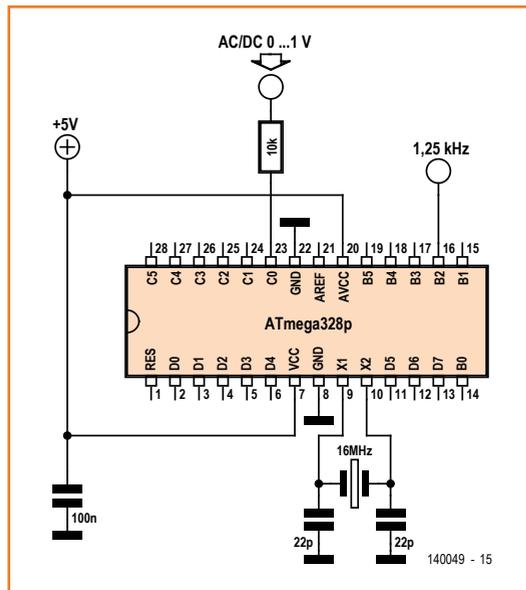


Bild 5.
AC/DC-Spannungsmessung.

Zählerstand 6 vorgestellt. So erzeugt man einen genauen Millisekunden-Takt. Eine Word-Variable `Ticks` wird bei jedem Timer-Überlauf um 1 hochgezählt. Bei einem Stand von 1000 wird jeweils eine Sekundenvariable „`Seconds`“ erhöht. Sowohl `Seconds` als auch die Millisekunden-Ticks können im Hauptprogramm ausgewertet werden. In diesem Fall zeigt das Programm den aktuellen Stand der Zeit in Sekunden seit dem Start an.

Damit die Interrupt-Routine tatsächlich aufgerufen wird, muss der entsprechende Interrupt (Überlauf-Interrupt von Timer0) noch „freigegeben“ werden, dies macht man mit der Zeile „`Enable Timer0`“). Außerdem muss man dem Controller noch generell erlauben, Interrupts zu behandeln („`Enable Interrupts`“). Am Terminal und im LCD erscheinen dann die aufsteigenden Sekunden. Umgekehrt kann man mit „`Disable Interrupts`“ alle Interrupts sperren.

Gemittelte Analogwerte

Analoge Messwerte werden oft durch überlagerte 50-Hz-Störsignale verfälscht. Dagegen hilft eine Mittelwertbildung (**Listing 5**). Wenn es gelingt, die Messwerte ganz genau über eine Periode von 20 ms oder ein Vielfaches davon zu mitteln, dann ergibt sich eine Nullstelle bei 50 Hz, d.h. Signale mit 50 Hz werden sehr stark gedämpft.

Auch in diesem Fall sorgt wieder ein Timer-Interrupt für einen genauen Zeitablauf. Der Mittelwert soll über 500 Einzelmessungen an ADC0 (**Bild 5**) gebildet werden, die über 400 ms (20 volle

Listing 5. Mittelung im Timer-Interrupt.

```

'-----
' UNO_Timer5.BAS  Timer1-Interrupt, ADC0 average
'-----
...

Dim Ticks As Word
Dim Ad As Word
Dim Ad0 As Long
Dim Ad0_mean As Long

Config Adc = Single , Prescaler = 32 , Reference = Internal
Config Portb.2 = Output

Config Timer2 = Timer , Prescale = 64
On OvF2 Tim2_isr
Enable Timer2
Enable Interrupts

Do
  Disable Interrupts
  Ad0_mean = Ad0_mean * 2443      'AC
  'Ad0_mean = Ad0_mean * 1100    'DC
  Ad0_mean = Ad0_mean / 1023
  Ad0_mean = Ad0_mean / 500
  Print Ad0_mean
  Locate 1 , 1
  Lcd Ad0_mean
  Lcd " mV      "
  Enable Interrupts
  Waitms 1000
Loop

Tim2_isr:
  '800 µs
  Timer2 = 56
  Portb.2 = 1
  Ticks = Ticks + 1
  Ad = Getadc(0)
  Ad0 = Ad0 + Ad
  If Ticks >= 500 Then
    Ticks = 0
    Ad0_mean = Ad0
    Ad0 = 0
  End If
  Portb.2 = 0
Return

End

```

50-Hz-Perioden) verteilt sind. Daraus ergibt sich ein Messintervall von 800 µs. Hier wird Timer2 (ebenfalls mit einem Vorteiler von 64) verwendet. Bei jedem Überlauf wird das Timer2-Register mit 56 geladen, sodass weitere 200 Takte später der nächste Überlauf erfolgt.

800 µs sind reichlich Zeit, um eine oder auch mehrere Messungen durchzuführen und die Messwerte aufzusummieren. Die Interruptaufrufe und damit die Einzelmessungen werden in Ticks gezählt. Nach jeweils 500 Messungen wird die Summe AD0 in der Variablen AD0_mean nach außen gereicht. Sie wird dann im Hauptprogramm ausgewertet und das Ergebnis wird an das Terminal gesendet.

Es ist gute Praxis, so einer Interruptroutine auch einmal mit einem Oszilloskop auf die Finger zu schauen. Sind es denn wirklich 800 µs bzw. 1,25 kHz? Und wie viel der gesamten Rechenzeit wird hier verbraucht? Der Trick ist ganz einfach. Am Anfang der Interruptroutine setzt man irgendeinen Port hoch und am Ende wieder runter. Hier ist es der Port PB2, an dem die LED 2 liegt. Das Oszilloskop zeigt tatsächlich Impulse im Abstand von 800 µs und mit einer Länge von ca. 60 µs. Alles im grünen Bereich. Aber in anderen Fällen ist es schon mal vorgekommen, dass jemand zu viel in die Interruptroutine gepackt hat und sich dann wunderte, dass das Hauptprogramm irgendwie klemmt. Wenn man mit Timer-Interrupts arbeitet, ist das Timing der Vorgänge im Hauptprogramm nicht mehr so leicht zu durchschauen. Eine Faustregel besagt, dass nicht mehr als 50 % der gesamten Zeit mit Interrupts verbraucht werden sollte.

Die Mittelung führt zu einer so guten Unterdrückung aller 50-Hz-Anteile, dass sich die Methode sogar zur Wechsellspannungsmessung mittels Einweggleichrichtung eignet. Der AD-Wandler ist selbst schon ein Einweggleichrichter, weil er nur positive Spannungen misst. Alles unter Null wird Null. Legt man eine Wechsellspannung über einen Schutzwiderstand von 10 kΩ direkt an den Messeingang, dann sieht der AD-Wandler nur die positiven Halbwellen. Und daraus wird der Mittelwert gebildet. Das Ergebnis ist daher bei einer Rechteckform im Schnitt nur halb so groß wie ein Gleichspannungssignal mit gleichem (Effektiv-) Wert. Dazu kommt für sinusförmige Spannungen noch ein Faktor $\text{Pi}/2 = 1,571$, d.h. ein reiner Mittelwert ist nur 90,03 % eines echten Effektivwerts. All das berücksichtigt die Umrechnung in die Anzeige in Millivolt. Für reine DC-Messungen

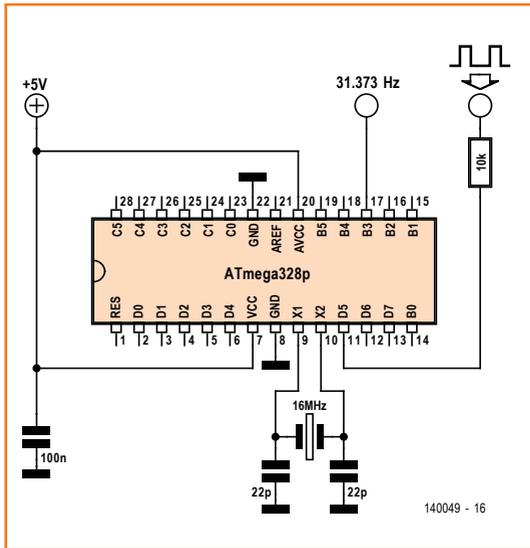


Bild 6. Frequenzmesser mit Test-Ausgang.

führt die Referenzspannung von 1100 mV zum Faktor 1100. Für Wechselspannungen steht hier stattdessen ein Faktor von 2443. Das Messgerät zeigt Wechselspannungen bis zu einem Spitzenwert von 1,1 V korrekt an. Und dabei kommt es nicht einmal auf die genaue Frequenz an. Alles zwischen 50 Hz und 50 kHz kann gemessen werden. So bekommt man ein richtiges Breitband-Millivoltmeter für Audiozwecke und anderes.

Frequenzmessung

Bei allen bisherigen Beispielen bekam der Timer seine Taktimpulse vom Prozessor (entweder wurde der Prozessortakt oder ein Bruchteil davon benutzt). Die Signale können aber auch von außen kommen. Dann arbeitet der Timer als Impulszähler (Counter). Der Timer1 eines mit 16 MHz getakteten ATmega328 kann externe Impulse mit einer Frequenz bis etwa 4 MHz zuverlässig zählen. Weil der zugehörige Eingang am Portpin PD5 liegt (**Bild 6**), muss man diesmal leider auf die Verwendung des LCDs verzichten. Einen Ausweg bietet die Verwendung eines externen LCDs, das über eine serielle Schnittstelle angesteuert wird. Hier werden die Ergebnisse zuerst einfach an ein PC-Terminal gesandt. Um aus dem Timer1 einen echten Frequenzmesser (**Listing 6**) zu machen, muss auch die Zeit genau erfasst werden. Deshalb werden hier zwei Timer und zwei Interrupts verwendet. Damit Timer1 auch Frequenzen über 65535 Hz messen

Listing 6. Frequenzmessung bis 4 MHz.

```

'-----
'UNO_Timer6.BAS  Frequency 0...4 MHz
'-----
...

Dim Lowword As Word
Dim Highword As Word
Dim Ticks As Word
Dim Freq As Long

Config Timer0 = Timer , Prescale = 64
On Ov0 Tim0_isr
Enable Timer0

Config Timer1 = Counter , Edge = Falling , Prescale = 1
On Ov1 Tim1_isr
Enable Timer1

Config Timer2 = Pwm , Prescale = 1 , Compare A Pwm = Clear
Up
Pwm2a = 128          'B3: 31373 Hz
Enable Interrupts

Do
  Print Freq;
  Print " Hz          ";
  Print Chr(13);
  Waitms 1000
Loop

Tim0_isr:
  '1000 µs
  Timer0 = 6
  Ticks = Ticks + 1
  If Ticks = 1 Then
    Timer1 = 0
    Highword = 0
  End If
  If Ticks = 1001 Then
    Lowword = Timer1
    Freq = Highword * 65536
    Freq = Freq + Lowword
    Ticks = 0
  End If
Return

Tim1_isr:
  Highword = Highword + 1
Return
End

```

Listing 7. Textausgaben auf dem LCD.

```

'-----
'UNO_Display.BAS  COM input B0
'-----
...

Dim Text1 As String * 16
Dim Text2 As String * 16

Open "comb.0:9600,8,n,1" For Input As
#2
'Software COM input at B0

Do
  'Input Text1
  Input #2 , Text1
  Locate 1 , 1
  Lcd Text2
  Text2 = Text1 + "      "
  Locate 2 , 1
  Lcd Text2
Loop
End

```

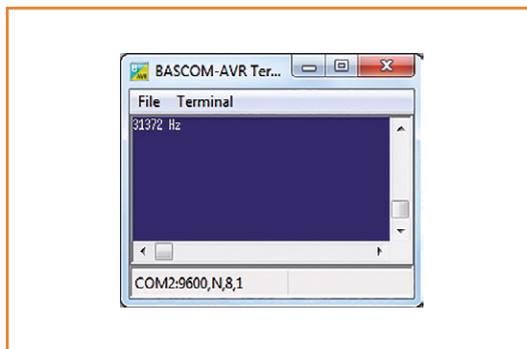


Bild 7.
Anzeige der Frequenz im Terminal.

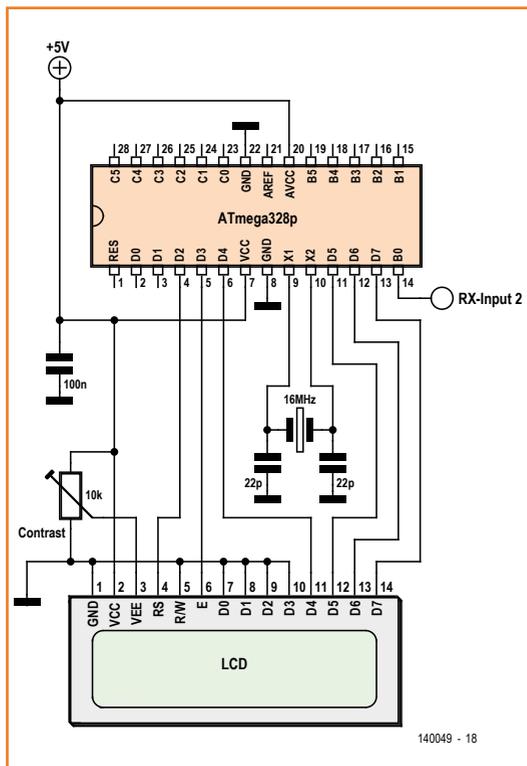


Bild 8.
Das LCD-Terminal.

kann, wird in Tim1_isr bei jedem Überlauf die Variable Highword erhöht.

Timer0 sorgt für die exakte Zeitmessung und eine genaue Torzeit von einer Sekunde. Zur Zeit Ticks = 1 wird der Timer1 zurückgesetzt und die Messung damit gestartet. Genau 1000 ms später wird der aktuelle Zählerstand von Timer1 in Lowword ausgelesen und zusammen mit den Überträgen in Highword zur Frequenz verrechnet. Das Hauptprogramm kann dann das Ergebnis Freq in Hertz ausgeben.

Lässt man Timer1 als Timer mit einem Takt von 16 MHz laufen (Config Timer1 = Timer , Prescale = 1) wird tatsächlich die Frequenz 16000000 Hz angezeigt. Als Counter schafft der Timer1 allerdings nur etwas mehr als ein Viertel dieser Frequenz, weil der Eingangszustand des Pins nur mit einer zwar hohen, aber begrenzten Rate abgefragt wird. Versucht man also z.B. eine Frequenz um 6 MHz zu messen, wird etwa jeder zweite Impuls überlesen, d.h. es werden ca. 3 MHz angezeigt. Bis etwas über 4 MHz ist der Zähler dagegen sehr genau.

Bei der Print-Ausgabe sieht man eine Besonderheit. Normalerweise schließt Bascom jede Ausgabe mit Chr(13) und Chr(10) ab. Nachfolgend soll die Ausgabe aber von einem anderen Bascom-Programm wieder empfangen werden. Und in Empfangs-Richtung wird nur ein Chr(13) erwartet. Deshalb wird hier beim Senden das Zeilenende unterdrückt und ein Chr(13) selbst angefügt. Um sich das Ergebnis anzeigen zu lassen, kann man auch das Bascom-eigene Terminal verwenden. Hier sieht man dann immer neue Messwerte in derselben Zeile. Das fehlende Chr(10) bewirkt also, dass keine neue Zeile angefangen wird (**Bild 7**).

Externes Display

Die Sache mit dem fehlenden Display lässt sich ganz einfach lösen, wenn man zwei Arduinos einsetzt. Einer dient als Anzeigeeinheit mit aufgesetztem Elektor-Shield, der andere arbeitet als Frequenzzähler und sendet seine Daten seriell an die Anzeigeeinheit. **Listing 7** zeigt ein einfaches Display mit rollender Anzeige. Man sieht immer die beiden letzten Zeilen.

Das Programm zeigt zwei Möglichkeiten für den seriellen Empfang. Input Text1 (auskommentiert) verwendet den normalen seriellen RX-Input an D0. Der Eingang ist über 1 kΩ mit dem USB-Wandler auf dem Uno-Board verbunden. Die von dort kommenden Signale muss man niederohmig überschreiben. Das gelingt, wenn man eine direkte Verbindung vom TX-Pin des sendenden Uno (der mit der Frequenzmessung) zum RX-Eingang der Display-Einheit herstellt. Der Nachteil ist, dass damit die Kommunikation beim nächsten Programm-Upload (per Bootloa-

der) gestört wird. Wenn man diese Tatsache vergisst, gibt es bei der nächsten Programmänderung erstmal Kopfschmerzen.

Eine zweite Empfangsmöglichkeit erhalten wir über „Input #2, Text1“. Bascom kann nämlich eine Software-Schnittstelle mit den passenden Parametern an einem beliebigen Portpin bilden. Hier wurde PBO (Arduino-Pin 9, **Bild 8**) gewählt. Die Texteingabe funktioniert genauso schön, und nun klappt es auch mit dem Upload.

(140049)

Weblinks

- [1] Roger Leifert, „DCF-Testsender“, Elektor 06/2014, www.elektor-magazine.de/130571
- [2] Martin Ossmann, „SDR mit AVR (2)“, Elektor 04/2012, www.elektor-magazine.de/100181

Anzeige



1-tägiges Altium Elektor Seminar:

„Tu Gutes und schreib darüber – wie dokumentiere ich meine Leiterplattenentwicklung richtig“

Altium

Ein nicht unerheblicher Teil der Arbeit eines Entwicklers wird von der Dokumentation dessen, was er macht, aufgefressen. Muss das sein? Kann der Anteil dieser in der Regel anspruchsvollen, notwendigen aber auch nicht gerade kreativen Tätigkeit eventuell durch einfache Maßnahmen reduziert werden?

In diesem Seminar sollen gemeinsam Richtlinien und Vorlagen für eine gute Dokumentation unter Ausnutzung der Möglichkeiten eines State-of-the-Art EDA-Werkzeugs erarbeitet werden. Dabei werden sowohl der Schaltplan- als auch der Leiterplattenbereich berücksichtigt. Die erarbeiteten Ergebnisse sind prinzipiell zwar unabhängig von der gewählten Entwicklungsplattform, die Umsetzung hingegen wird zugeschnitten auf die Möglichkeiten von Altium Designer erfolgen.

Veranstaltungsort/-termin: Köln, 14. Oktober 2014

Referent: Dipl. Ing. Christian Keller

Teilnahmegebühr: 149,00 Euro (inkl. MwSt.) für Elektor-Mitglieder (50% Rabatt!)

299,00 Euro (inkl. MwSt.) für Nicht-Mitglieder

Im Preis sind Mittagessen, Seminarunterlagen sowie Dokumentation und Aushändigung des Teilnahmezertifikats inbegriffen.

Weitere Infos & Anmeldung: www.elektor.de/events-de

Elektor-Abomitglieder erhalten 50% Rabatt auf den Seminar-Preis!

Elektor-Verlag GmbH | Süsterfeldstr. 25 | 52072 Aachen
Tel. +49 241 88 909-16

www.elektor.de
j.grotenrath@elektor.de

Verbesserter Stromtransformator

Von
J.T. van Es
(NL)

Dieser Nachtrag bezieht sich auf zwei Veröffentlichungen von Martin Oßmann, die im März und April 2014 in Elektor erschienen. Dort wurde eine niedrige untere Grenzfrequenz, abhängig von L/R, unter anderem durch die Wahl des Materials für den Trafokern angestrebt. Der Weg ist sicher nicht falsch, doch es geht, wie nachstehend gezeigt, auch einfacher.

Bild 1.
Basisschaltung.

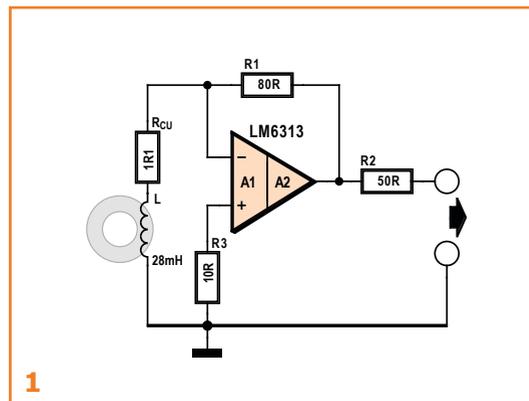
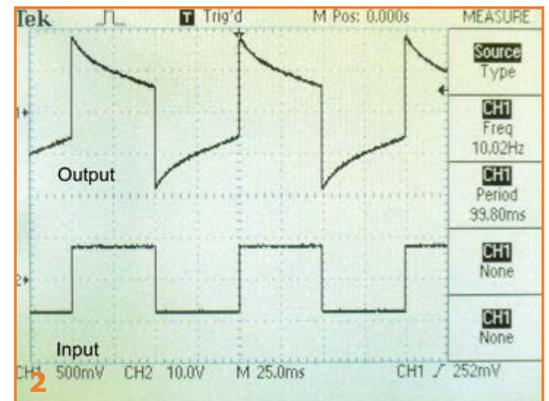


Bild 2.
Ausgangssignal (10 Hz) ohne
Mitkopplung



In **Bild 1** ist die Schaltung eines Strommessers dargestellt, den der Autor im Jahr 2005 entwarf. Als Stromtransformator dient die Empfänger-Induktivität einer Benzin-Einspritzung des Kfz-Zulieferers Tubantor aus dem Jahr 1983. Auf einem Ferrit-Ringkern ist eine aus 80 Windungen bestehende Wicklung aufgebracht. Die Bandbreite des Systems beträgt 7 Hz...5,5 MHz bei -3 dB. Ein Rechtecksignal mit der Frequenz 10 Hz wird, wie **Bild 2** zeigt, erheblich verformt. Widerstand R3

wurde hinzugefügt, um den Verstärker-Offset zu kompensieren.

Die Mitkopplung des Verstärkers über R4 und R5, dargestellt in **Bild 3**, verlagert den mit R5 einstellbaren Eingangswiderstand in den negativen Bereich, so dass der Kupferwiderstand R_{Cu} kompensiert wird. Der Schleifenwiderstand sinkt, und das Verhältnis L/R steigt. In **Bild 4** sind die Auswirkungen auf das Signal unübersehbar.

Wenn die Mitkopplung noch etwas gesteigert

Bild 3.
Mitkopplung über R4/R5.

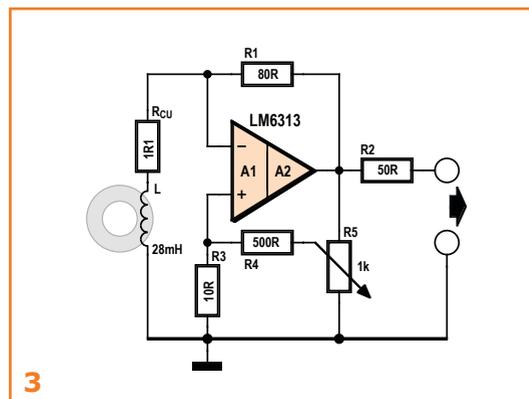
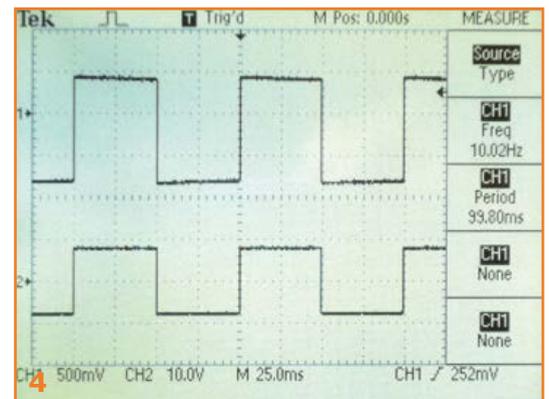


Bild 4.
Ausgangssignal (10 Hz) mit
Mitkopplung.



wird, passiert auch ein 1-Hz-Signal fast unbeschadet das System, siehe **Bild 5**. Dann sind jedoch Einbußen an der Gleichstrom-Stabilität zu verzeichnen. Eine Lösung des Problems zeigt **Bild 6**, sie besteht aus dem Regelkreis mit einem zusätzlichen Opamp sowie R7 und C1. Der Opamp muss FET-Eingänge besitzen, so dass die Werte von R7 und C1 hoch sein können. Für C1 ist nur ein Folienkondensator mit niedrigem Leckwiderstand geeignet.

Da bei dieser Schaltungstechnik die Sekundärwicklung nahezu perfekt kurzgeschlossen ist, liegt praktisch keine Spannung an. Dies gilt abgesehen vom Spannungsabfall am Kupferwiderstand auch für die primäre Seite. Die Induktion in den Kern, die bei Stromtransformatoren prinzipiell niedrig ist, erreicht ihr absolutes Minimum. Bei der beschriebenen Konstellation kann der Wert für L niedrig gewählt werden, was für das hochfrequente Verhalten günstig ist. Das Messen von Gleichströmen ist jedoch nicht möglich, dies lässt sich nur mit einem Hall-Sensor realisieren. Dazu muss der Kern mit einem Spalt versehen werden, der Spalt würde jedoch den μ -Faktor und folglich die Induktivität L ungünstig beeinflussen. Der Autor berichtet, dass er bereits vor etwa 30 Jahren ein ähnliches System mit einem Vitrovac-Ringkern entworfen hat. Der Innendurchmesser lag bei etwa 8 cm und die Bandbreite, die das System erreichte, bei 23 MHz...20 MHz!

(140266)gd

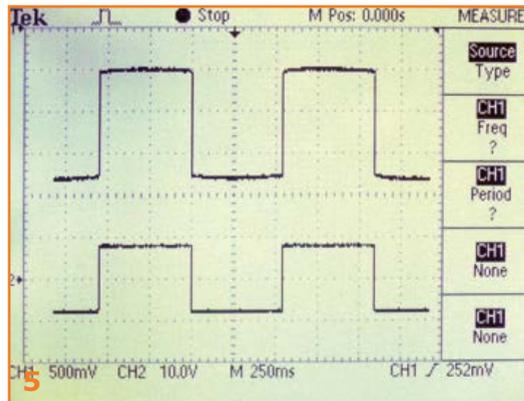


Bild 5. Ausgangssignal (1 Hz) bei gesteigerter Mitkopplung.

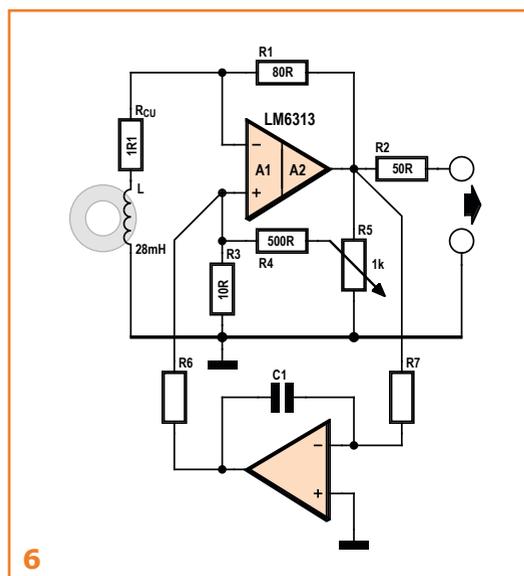
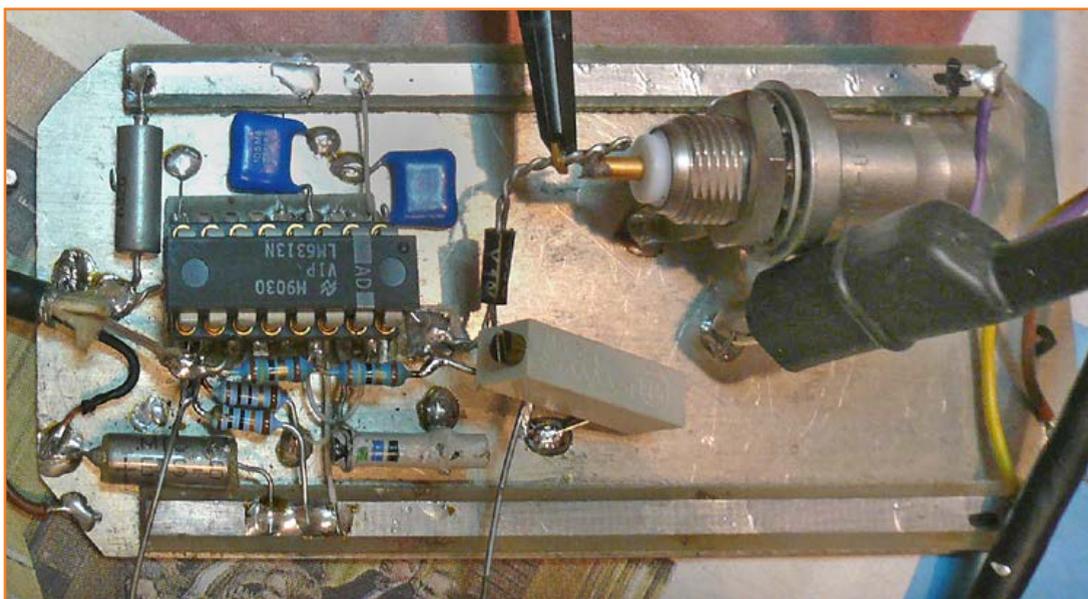


Bild 6. Verbesserte DC-Stabilität durch zusätzlichen Opamp und R7/C1.



Verstärkeraufbau des Autors.

Visual Basic auf dem Raspberry Pi



Von **Bert van Dam**
(Niederlande)

Der Raspberry Pi wird normalerweise in Python programmiert, einer leistungsfähigen und einfach zu verwendenden Sprache. Allerdings können Programme mit einer schönen Grafik lange nicht so leicht erstellt werden wie unter Visual Basic.

Natürlich lassen sich auch mit Python grafische Templates wie in [1] verwenden, es ist aber viel einfacher, sie auf einem Windows-PC zum Beispiel mit Visual Basic zu entwickeln und sie dann auf dem Raspberry Pi einzusetzen. Auf diese Weise profitieren Sie von der Rechenleistung des PCs und den umfangreichen grafischen Möglichkeiten von Visual Basic!

Visual Basic verwendet .NET, eine Art Framework mit einer großen Sammlung von Bibliotheken, die Programmierern zur Verfügung steht. Programme werden in einen Zwischencode für .NET kompiliert. Im Englischen nennt man dies Common Language Interface. Auf diese Weise laufen Programme, die für einen Computertyp entwickelt wurden, auch auf anderen, solange nur .NET installiert ist. NET wurde von Microsoft entwickelt und ist deshalb vor allem für Windows-Computer gedacht, aber es gibt auch Versionen für Linux, von denen „Mono“ die gebräuchlichste ist.

In diesem Artikel installieren wir die Linux-Version von .NET auf einem Raspberry Pi und probieren eine Visual Basic-Demo aus, die auf einem PC (Windows 7, 64-bit) entwickelt wurde.

Was ist zu tun?

Wenn Sie nicht über Visual Basic auf Ihrem PC verfügen, laden Sie Visual Basic Express 2010 von der Microsoft-Website [2] herunter. Es gibt auch eine neuere Version (2013), aber da die Linux-Version von .NET der Entwicklung immer ein wenig hinterher hinkt, sollte man die erprobte 2010er-Variante von Visual Basic einsetzen.

Erstellen Sie ein Testprogramm in Microsoft Visual Basic. Im Download für diesen Artikel [3] finden Sie den Quellcode für ein einfaches Programm mit einer Schaltfläche namens „monotest“. Wenn Sie das Programm starten, erscheint ein Fenster mit einem Button. Klicken Sie auf diese Schaltfläche, so wird der Text „Hello World“ angezeigt (**Bild 1**

und 2). Wer (noch) nicht über Visual Basic auf dem PC verfügt, kann die kompilierte Version (monotest.exe) des Programms verwenden.

Stellen Sie sicher, dass Ihr Raspberry Pi mit dem Internet verbunden ist und installieren Sie „mono“ mit den folgenden Befehlen (dies wird einige Zeit in Anspruch nehmen; antworten Sie auf alle Fragen während der Installation mit „Y“ = yes):

```
sudo apt-get update
sudo apt-get install mono-vbnc
```

Kopieren Sie die Datei monotest.exe vom PC auf den Raspberry Pi. Das geht ganz einfach mit dem Programm WinSCP, wie im Buch „Raspberry Pi“ beschrieben. Dieses Programm ist Teil des Downloads für diesen Artikel und muss nicht installiert werden. Starten Sie das Programm WinSCP.exe und geben Sie unter „Session“ bei „Host name“ die IP-Adresse des Raspberry Pi ein. Vergewissern Sie sich, dass Port 22 ausgewählt ist. Wenn Sie zum ersten Mal WinSCP mit dieser IP-Adresse verbinden, erscheint eine Warnung, die Sie akzeptieren beziehungsweise ignorieren. Dann gehen Sie zu „Commander“. Sie werden nun Dateilisten des PCs (links) und des Raspberry Pi (rechts) sehen. Suchen Sie die ausführbare Datei auf der linken Seite und ziehen Sie den Link zum RPi.

Führen Sie das Programm auf dem Raspberry Pi aus mit dem Befehl `mono monotest.exe`.

Jetzt können Sie nach Herzenslust mit dieser schönen Technik experimentieren. Weil Sie das Programm auf dem PC entwickeln und kompilieren, sind allerdings spezifische Raspberry-Pi-Komponenten (wie die GPIO-Pins) nicht einfach zu erreichen. Die Methode ist aber besonders geeignet, um auf einfache Weise schöne grafische Programme (zum Beispiel Spiele) zu realisieren.

Software

Das Demoprogramm monotest, der dazu gehörende Visual-Basic-2010-Quellcode sowie das WinSCP-Programm stehen zum Download auf der Elektor-Website zur Verfügung [3]. Diese Software wurde mit der SD-Karte getestet, die dem Raspberry-Pi-Buch beiliegt, funktioniert aber auch mit dem normalen Raspbian Wheezy.

(140263)

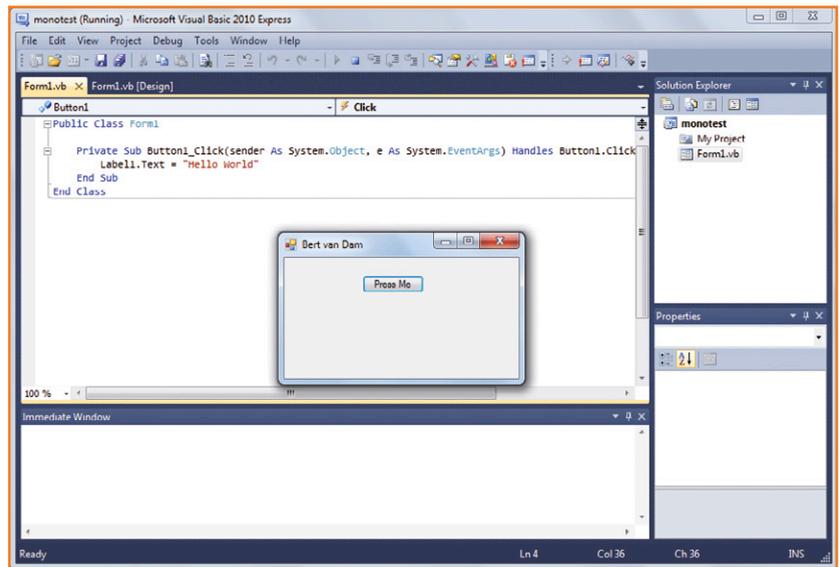


Bild 1. Das Programm Monotest, geöffnet in Visual Basic auf einem Windows PC.

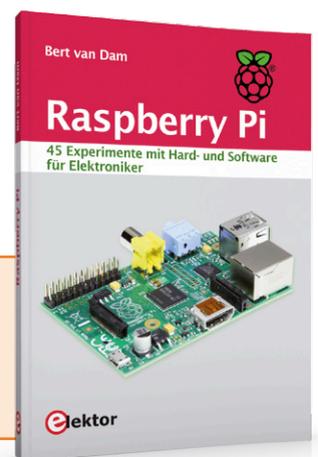


Bild 2. Links das Programm auf dem Windows-PC, rechts auf dem Raspberry Pi.

Weblinks

- [1] www.elektor.de/rpi-buch
- [2] www.visualstudio.com/en-us/downloads#d-2010-express
- [3] www.elektor-magazine.de/140263

Weitere interessante Raspberry-Pi-Projekte finden Sie im Elektor-Buch „Raspberry Pi - 45 Experimente mit Hard- und Software für Elektroniker“ von Bert van Dam [1].



Multifunktionales Messsystem

Fünf Geräte, die Elektroniker brauchen

Von **Harry Baggen**
(Redaktion NL) und
Luc Lemmens
(Elektor-Labor)

VirtualBench heißt ein innovatives Messsystem von National Instruments, das fünf typische Prüfplatzgeräte in einem integrierten System vereint. Bedient wird das Messgeräte-Ensemble über einen Desktop- oder Tablet-Computer. In Europa war Elektor unter den Ersten, die ein Exemplar testen durften.



Bild 1.
Klein, aber oho:
VirtualBench von National
Instruments wird diesem
Statement voll gerecht.

Den Arbeitsplatz des Elektrikers prägen Geräte, die er ständig braucht: Ein Multimeter, ein Oszilloskop und ein Labornetzgerät sind unabdingbar. Der Trend geht zum multifunktionalen System, zum Beispiel zum Allround-Oszilloskop, das neben seiner angestammten Aufgabe auch Dienste als Logic-Analyser, Funktionsgenerator oder Multimeter leistet. Doch auch diese Geräte folgen dem konventionellen Standard, sie arbeiten „stand-alone“, mit eigenem Bildschirm und eigenen Bedienelementen.

National Instruments (NI), dessen Name hinter LabVIEW sowie einer weitläufigen Reihe professioneller Messgeräte steht, ist nicht grundlos als Hersteller von „Mainstream“-Messgeräten bekannt. Mit seiner Produktlinie VirtualBench geht NI nun alternative Wege, die einer neuen Philosophie folgen. Diese Philosophie unterscheidet sich grundlegend von den Produkten der Konkurrenz. VirtualBench nutzt vollständig die Funktionalität des vorhandenen PCs, Laptops oder Tablets für die Bedienung des Systems und die Ausgabe der Messgrößen. Die Idee der Auslagerung von Funktionen ist an sich nicht neu, doch sie wurde noch

nie so konsequent umgesetzt. NI hat in einem kompakten, fast unscheinbaren Gerät die Funktionalitäten eines Zweikanal-Oszilloskops, eines 32-bit-Logic-Analysers, eines dreifachen Labornetzgeräts, eines Funktionsgenerators und eines 5½-Digit-Multimeters auf einen einheitlichen Nenner gebracht. Sämtliche Funktionen werden über eine gemeinsame Oberfläche auf dem PC bedient.

Als wir von National Instruments das Angebot bekamen, ein erstes auf dem europäischen Markt erhältliches Exemplar auf Herz und Nieren zu testen, konnten wir natürlich nicht widerstehen. Wir waren gespannt: Hält VirtualBench im rauen Laboralltag wirklich das, was NI verspricht?

Zierlich, aber grundsolide

Die Abmessungen des VirtualBench sind zwar auf der Website von NI vermerkt, doch als wir das zierliche Gerät zum ersten Mal in Augenschein nahmen, waren wir überrascht. Wie ein Behälter für die Frühstücksbrote, mehr Volumen hat dieser Vieleskönner nicht. Verglichen mit dem Gerätepark, den VirtualBench funktional ersetzt, wirkt der Arbeitsplatz jetzt einladend aufgeräumt.

Zum Lieferumfang gehören sämtliche Leitungen und Verbindungen: Zwei Oszilloskop-Kabel, zwei Multimeter-Messkabel, ein 40-poliges Verbindungsstück mit Messkabeln für den Logic-Analyser, ein arretierbares USB-Kabel sowie das Netzkabel. Sogar ein kleiner Schraubendreher ist dabei.

Das Gehäuse wirkt solide, robust und unempfindlich. Die Frontplatte sowie Boden und Deckel sind aus alufarbenen Metallplatten gefertigt, während die übrigen Teile aus blauem Kunststoff bestehen. Wie **Bild 1** zeigt, sind alle wichtigen Anschlüsse und der Netzschalter auf der Frontplatte vereint. Die Kontaktleisten des digitalen I/O-Ports und der Stromversorgung lassen sich verschrauben, so dass die Verbindungen mechanisch gesichert werden können. Auf der Gehäuserückseite befinden sich der Netzanschluss, die USB-Buchse und eine WLAN-Antenne. Die Kühlung der Elektronik übernimmt ein kaum hörbarer Lüfter.

Das Gerät ist nur zusammen mit einem Computer unter Windows oder einem Tablet betriebsfähig, als Tablet kann vorläufig nur ein iPad genutzt werden. Der Datenaustausch mit dem PC oder Laptop findet über USB, den integrierten WLAN-Host oder ein vorhandenes WLAN statt. Mit dem Tablet kommuniziert das Gerät immer drahtlos.

Alles in einem Fenster

Das Installieren von Software auf dem PC oder Laptop ist nicht erforderlich. Nach dem Ein-



VirtualBench ist auch zusammen mit einem iPad von Apple einsetzbar.

schalten von VirtualBench erkennt Windows das über USB angeschlossene Gerät (ähnlich wie bei einer externen Festplatte oder einem DVD-Laufwerk), anschließend startet die im Flash-Speicher von VirtualBench stehende Software. Beim ersten Start installiert Windows diverse USB- und HID-Treiber, möglicherweise benötigt die Windows-Firewall für den Zugriff auf das Programm die Zustimmung des Benutzers. Bei nachfolgenden Sitzungen entfällt diese Prozedur, es geht dann etwas schneller. Das iPad dem PC eine Nasenlänge voraus. Wenn die Software einmal über den iTunes-Store installiert ist,

Eigenschaften

Oszilloskop (2 Kanäle)

- Eingangsbandbreite: 100 MHz
- Sample-Rate: 1 Gigasample/s (500 Megasample/s bei 2 Kanälen)
- Puffergröße: 1 Megasample/Kanal

Logic-Analyser

- Anzahl der Kanäle: 34
- Max. Eingangsfrequenz: 100 MHz
- Eingangsspannung: 0...5 V

Funktionsgenerator

- Frequenzbereich: 20 MHz (Sinus), 5 MHz (Rechteck)
- Signalformen: Sinus, Rechteck, Dreieck, DC

Digitales Multimeter

- Display: 5½ Digit

- Grundgenauigkeit: 0,015 % (V_{DC})
- Messfunktionen: V_{DC} , V_{AC} , I_{DC} , I_{AC} , Widerstand, Diode, Durchgangstester
- Max. Spannung/Strom: 300 V/10 A

Labornetzgerät

- Anzahl der Ausgänge: 3
- Kanal 1: 0...6 V / 1 A
- Kanal 2: 0...+25 V / 0,5 A
- Kanal 3: 0...-25 V / 0,5 A

Digitale I/O-Schnittstelle

- Anzahl der Kanäle: 8
- Schaltbar als Eingänge oder Ausgänge
- Eingänge 3,3 V/5 V kompatibel, Ausgänge 3,3 V

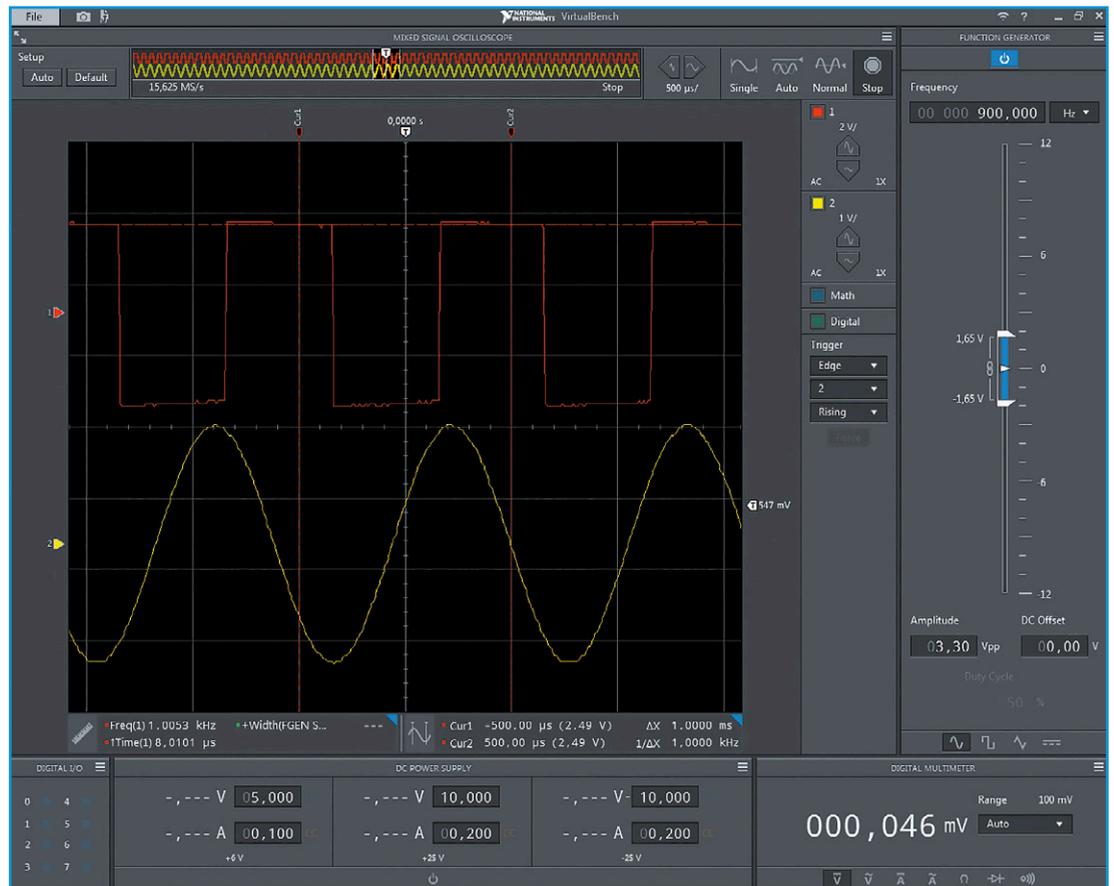


Bild 2.
Die Anwenderoberfläche auf dem Bildschirm eines Laptops (Windows-Version).

muss sich das Tablet nur noch in das WLAN des VirtualBench einloggen.

Aus **Bild 2** ist ersichtlich, wie sich VirtualBench unter Windows auf dem Bildschirm präsentiert. Sämtliche Mess- und Einstellfunktionen werden über das gleiche Fenster bedient. Das Oszilloskop und der Logic-Analyser mit ihren Bedienelementen belegen den größten Teil der Fläche. Die Fensterelemente der anderen Geräte ranken sich um diesen Teil des Fensters herum. Der Funktionsgenerator mit zwei Schiebern für die Signalamplitude und den DC-Offset ist rechts angeordnet, darunter befindet sich das Multimeter. Das schmale Feld links unten gehört zum digitalen I/O-Port, das größere Feld rechts daneben ist für die drei Kanäle des Netzgeräts zuständig. Die Handhabung ist unkompliziert, sie bedarf kaum einer Erklärung. Fast alle Funktionen sind mit der Maus bedienbar. Spannungen, Frequenzen oder Amplituden können als Zahlenwerte über die Tastatur oder durch Scrollen mit der Maus eingegeben werden. Wenn ein Wert außerhalb des zulässigen Bereichs liegt, wird ein Warnhinweis

ausgegeben, der Wert wird nicht akzeptiert. Bei den meisten Feldern erscheint nach Klicken auf die Ecke rechts oben ein Menü, in dem weitere Einstellungen vorgenommen werden können, oder es werden zusätzliche Informationen angezeigt. Im Hauptmenü ist eine Screenshot-Funktion integriert, die zum Dokumentieren der Messdaten dienen kann. Die Messdaten können auch im CSV-Format gespeichert werden.

Die Version für den iPad hat nahezu gleiche Funktionalität, doch hier wird VirtualBench natürlich über den Touchscreen bedient. Manchmal ist die Bedienung mit der Maus bequemer, manchmal hat der Touchscreen Vorteile.

VirtualBench: Fünf in Eins

Im Verlauf unserer Testreihe wurde zwar auch die Bedienung mit dem iPad erprobt, doch wir hatten den Eindruck, dass diese App noch nicht ganz ausgereift ist. Deshalb haben wir uns entschlossen, dem PC unter Windows den Vorzug zu geben. Sicher sind hier auch persönliche Präferenzen im Spiel, doch schon der größere Bildschirm

hat nicht von der Hand zu weisende Vorteile. Die folgenden Betrachtungen beziehen sich auf die Windows-Version der Software.

Wir haben mit VirtualBench schnell Bekanntschaft geschlossen. Die auf der Vorderseite versammelten Anschlüsse sind selbst erklärend, sie entsprechen den konventionellen Laborgeräten. Die Abstände einiger Buchsen sind etwas knapp bemessen, wir hätten uns etwas mehr Platz zwischen den BNC-Buchsen sowie den BNC-Buchsen und den Logic-Analyser-Anschlüssen gewünscht. Doch das liegt sicher auch daran, dass wir es anders gewohnt sind. Die Schraubklemmanschlüsse des integrierten Netzgeräts

halten wir zwar für praktikabel, doch nach einiger Zeit wünschten wir uns die Bananenbuchsen zurück, wie sie an anderen Geräten Standard sind. Falls die Schraubklemmanschlüsse oder die Anschlüsse der digitalen I/O-Schnittstelle nach intensivem Langzeiteinsatz verschlissen sind, liefert NI Ersatzteile.

Die Bedienung über den Bildschirm bedarf nur einer kurzen, ersten Orientierung, alle Bildschirmelemente sind plausibel und selbst erklärend. Das mitgelieferte Handbuch mussten wir niemals um Rat fragen – was bei erfahrenen Praktikern wie uns nicht anders zu erwarten ist.

Oszilloskop/Logic-Analyser

Auf dem Bildschirm belegt dieser Teil die größte Fläche, was sicher angebracht ist, denn die Signale sollen möglichst groß und deutlich dargestellt werden. Das Feld lässt sich zur Vollbildarstellung zoomen. Die Schaltfläche *Auto-Setup* hilft, gemessene Signale schnell optimal darzustellen. Das Oszilloskop ist mit allen Einstellungen und Triggerarten ausgestattet, die üblicherweise bei konventionellen Oszilloskopen vorhanden sind. Ferner wird auf dem Bildschirm der Inhalt des Messpuffers dargestellt. Näher zu betrachtende Abschnitte können mit der Maus selektiert und sofort sichtbar gemacht werden. Unter der Oszilloskop-Bildfläche befinden sich zwei Leisten, mit denen ein Signalteil abgrenzbar ist. Der gewählte Abschnitt kann diversen Operationen unterschiedlicher Art (mehr als 20!) unterzogen werden. Hinter der Schaltfläche *Math* verbergen sich mathematische Operationen; beim gegenwärtigen Stand der Software sind das zunächst nur Addition, Subtraktion, Multiplikation und eine Standard-FFT, die vergleichsweise wenige Einstellungen bietet. Über die Schaltfläche *Digital* werden die digitalen Kanäle sowie die gewünschte Triggerung gewählt.

Während der Arbeiten mit dem Oszilloskop hatten wir den Eindruck, dass die Reaktion auf Signaländerungen

recht träge verlief. Das lag daran, dass die Signalaufbereitung standardmäßig auf *32 Averaged* eingestellt ist. Diese Einstellung haben wir zunächst auf *2 Averaged* und anschließend auf *Sample* rückgesetzt. Nun hatten wir das Gefühl, dass wir Signaländerungen in Echtzeit folgen. Alle Einstellungsänderungen müssen gespeichert werden, anderenfalls startet VirtualBench mit den Standard-Einstellungen. In der grafischen Darstellung haben wir die Angabe der Maßstäbe vermisst, die für die horizontale und vertikale Achse gelten. Die Werte sind nur vergleichsweise winzig bei den virtuellen Bereichsdrehknöpfen vermerkt. Beim Einsatz als Logic-Analyser lassen sich die Signale nur effizient verfolgen, wenn ihre Anzahl gering ist. Sind viele der maximal 32 Kanäle belegt, entsteht auf dem Bildschirm ein verwirrendes, unübersichtliches Bild. Für den Vielkanal-Betrieb ist ein professioneller, ausgewachsener Logic-Analyser die bessere Alternative, er bietet darüber hinaus eine Fülle weiterer Möglichkeiten. Das Problem der Unübersichtlichkeit tritt bei vielen Geräten auf, die wahlweise als Oszilloskop und Logic-Analyser arbeiten. Hinzu kommt, dass es Geduld erfordert, alle Messleitungen des 40-poligen Steckverbinders (leider geliefert ohne Klemmen) mit dem zu messenden System zu verbinden.

Funktionsgenerator

Der im VirtualBench integrierte Funktionsgenerator (14 bit, 125 MS/s) erzeugt alle Standard-Signalformen in Frequenzbereichen, die für die meisten Anwendungen mehr als weit bemessen sind. Der Ausgang kann auch einstellbare Gleichspannungen liefern, sie können bis ± 6 V an 50Ω oder ± 12 V an $10 \text{ k}\Omega$ betragen. Die periodischen Signale (Sinus, Rechteck, Dreieck) lassen sich mit der

Maus über weite Bereiche einstellen. Prinzipiell sind bei diesem Funktionsgenerator auch beliebige, vom Standard abweichende Schwingungsformen programmierbar. Diese Option ist jedoch (noch?) nicht in der Software implementiert. Nutzer können aber ein LabVIEW VI erstellen, das eigene Signalformen generiert.

Multimeter

Das Multimeter hat uns ebenfalls überzeugt, es arbeitet mit hoher Genauigkeit, die Betriebsarten entsprechen einem herkömmlichen Multimeter. Wir hätten uns lediglich gewünscht, dass das Multimeter-Feld gezoomt werden kann. Dieser Wunsch betrifft übrigens auch andere Felder auf

dem Bildschirm, leider sind sie unverrückbar, ihr Ort und ihre Größe sind fest vorgegeben. Eine Ausnahme macht das Feld des Oszilloskops und Logic-Analysers, es lässt sich auf Vollbildarstellung zoomen.

Labornetzgerät

Auch wenn das Labornetzgerät auf dem Bildschirm nur eine bescheidene Fläche einnimmt, kann es sich mit herkömmlichen Netzgeräten messen. Die Ausgangsströme reichen für die meisten Zwecke vollkommen aus. Die Spannungen lassen sich präzise einstellen, für jeden der drei Kanäle ist eine Strombegrenzung möglich. Auf dem Bildschirm werden die tatsächlich an den Ausgangsklemmen

liegenden Spannungswerte sowie die Werte der fließenden Ströme ausgegeben. Diese Eigenschaften sind willkommen, sie fehlen bei den meisten konventionellen Labornetzgeräten. Ein kleiner Wermutstropfen sind die Anschlussklemmen, sie waren, wie schon oben erwähnt, im täglichen Laborbetrieb eher unhandlich.

Digitale I/O-Schnittstelle

Zum Testen dieser Funktion stand uns leider kein geeignetes Testobjekt zur Verfügung. Mit Sicherheit existieren trotzdem

viele Anwendungen, insbesondere wenn VirtualBench zusammen mit LabVIEW-Programmen zum Einsatz kommt.

Unser Fazit

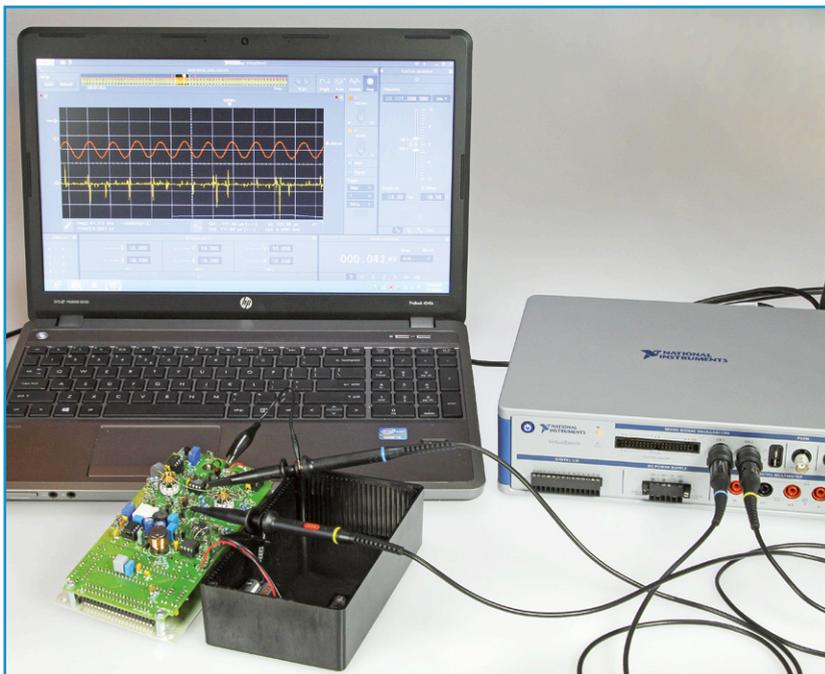
Viele Details mussten unerwähnt bleiben, denn sie würden den Rahmen dieses Beitrags sprengen. Trotzdem hoffen wir, dass es uns gelungen ist, einen ersten Eindruck zu vermitteln. Vielleicht

ist das Bild entstanden, dass wir bewusst nach Unzulänglichkeiten gesucht haben, doch davon sind wir weit entfernt. Mehrere Elektor-Entwickler würden gern einen VirtualBench an ihrem Arbeitsplatz sehen, am liebsten auch noch ein zweites Exemplar in der privaten Werkstatt zu Hause. VirtualBench ist ein praxisnahes Messgeräte-Ensemble, das den meisten Aufgaben mit Bravour gewachsen ist. Lediglich die Software halten wir noch für entwicklungs- und erweiterungsfähig. Was wir uns wünschen, könnte in nachfolgenden Versionen realisiert sein. Einem völlig neuen Produkt sei zugestanden, dass die Software im ersten Anlauf mit Unvollkommenheiten behaftet ist. Das Konzept dieses Produkts hat uns uneingeschränkt überzeugt. VirtualBench ist für den alltäglichen Einsatz in Entwicklungslabors, Service-Betrieben oder Lehrinstituten eine zukunftsweisende Lösung. Auch der Heim-Elektroniker ist, wenn es das Budget erlaubt, mit VirtualBench gut beraten. Der Preis schlägt zwar mit 1690 € plus Nebenkosten zu Buche, doch dieses Geld ist das Produkt wert.

(140252)gd

Weblink

www.ni.com/virtualbench/



Elektor 3D-Drucker

Drucken und Modellieren in der dritten Dimension

Elektor bietet in Kooperation mit dem 3D-Drucksystem-Spezialisten German RepRap diesen 3D-Drucker (mit blauer Speziallackierung) in der Sonderausführung mit exklusivem Lieferumfang an.

Der neue Elektor 3D-Drucker ermöglicht bessere Druckergebnisse und einen wesentlich schnelleren Aufbau als gängige Drucker-Bausätze. Für die im Vergleich wesentlich kürzere Aufbauzeit sorgen Steckanschlüsse bei den Schrittmotoren, bestückte Platinen und ein fester, stabiler Edelstahlrahmen.

Technische Daten:

- Abmessungen (B x H x T): 500 x 460 x 460 mm
- Druckvolumen (X x Y x Z): 230 x 230 x 125 mm
- Geschwindigkeit: 3 mm Material bis 100 mm/Sekunde, 1,75 mm Material bis ca. 180 mm/Sekunde, Leerlauf bis 350 mm/Sekunde
- Betriebsspannung: 230/115 V Wechselspannung
- Material: ABS / PLA / PS / PVA / Laywood (Holzdraht) / Laybrick



Lieferumfang:

- Kunststoffteile (Polyamid-Kunststoff, besonders schlagfest und belastbar)
- Edelstahlrahmen (in blau) mit Nivelierfüßen, Schrauben, Gewindestangen, Wellen, Linearkugelbuchsen, Kugel- und Gleitlager
- T 2.5 Antriebsriemen und gefräste Riemenscheiben
- 5x NEMA17 Schrittmotor mit 0,52 Nm Drehmoment
- Ramps v1.4 Elektronik inkl. Kabel und Netzteil
- Modulares Hot-End 3 mm mit 0,5 mm Düse (optional auch mit 0,4 / 0,3 mm Düse)
- Beheiztes Druckbett 12 V
- 1x PLA Plastik 750 g (rot 3 mm)
- 1x PLA Plastik 750 g (gelb-grün 3 mm)
- 1x PLA Plastik 750 g (blau 3 mm)
- Software-CD inkl. Druckcode für ein Druckobjekt

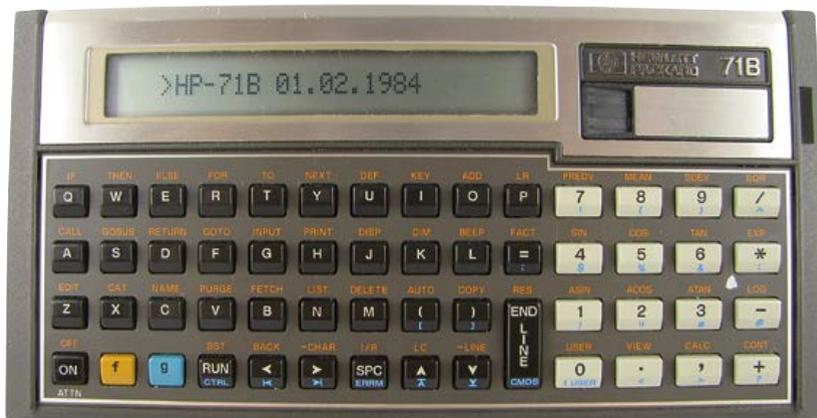
Preis: 1.199,00 Euro
(inkl. MwSt.)

Die Lieferung innerhalb Deutschlands erfolgt frei Haus.

Weitere Infos & Bestellung unter www.elektor.de/3d-drucker

Der Zahlenfresser HP-71B (1984)

Ein wahrhaftes Universalwerkzeug



Am 1.2.1984 stellte Hewlett-Packard den Taschen-Computer HP-71B zum stolzen Preis von 525 \$ vor. Das erregte einiges Aufsehen. In einem nur 19 x 9,7 x 2,5 cm messenden flachen Gehäuse steckte nicht nur ein kompletter, in BASIC programmierbarer Computer, sondern auch ein technisch-wissenschaftlicher Taschenrechner für den mobilen professionellen Einsatz.

Von Dipl.-Inf.
Karl-Ludwig Butte
(D)

Die damaligen 525 \$ hätten heute eine Kaufkraft von etwa 1190 \$ [1]. Und was bekam man für so viel Geld? Beim HP-71B handelt es sich um die erste Maschine, bei der die brandneue Saturn-CPU eingesetzt wurde. Diese erfolgreiche CPU wurde dann später auch noch in anderen Taschenrechnern wie dem HP-48 verbaut. Diese 4-bit(!)-CMOS-CPU war mit 640 kHz getaktet und wurde mit 5 V versorgt [2]. Ein einzeliliges LCD mit 22 Zeichen und einer QWERTY-Tastatur mit numerischem Tastenfeld sowie „viel“ ROM (64 KB) und RAM (17,5 KB) plus sechs Erweiterungsports komplettierten das System.

Software und Kontroversen

Genau genommen war dies nach dem HP-75C/D schon der zweite Versuch von HP, im von Casio und Sharp dominierten Markt kleiner BASIC-Computer Fuß zu fassen. An Sharps erfolgreichen PC-1500 wird sich sicherlich noch der eine oder andere Leser erinnern. Eine interessante Randnotiz ist, dass der HP-75 von den Desktop-Maschinen des Typs HP-85 abgeleitet wurde, während beim HP-71B die in Corvallis (Oregon) beheimatete Taschenrechnerabteilung verantwortlich zeichnete.

Abgesehen von der proprietären Saturn-CPU war das Besondere des HP-71 seiner Software und seinen Erweiterungsmöglichkeiten geschuldet. Es gab die beiden Modi „Taschenrechner“ und „BASIC-Computer“, die aber durch dynamisches Speicher-Management verknüpft waren. Man konnte z.B. auf die vom Taschenrechner belegten Variablen auch unter BASIC zugreifen. Der Taschenrechnerteil brachte schockierende Neuerungen für die an UPN (umgekehrte polnische Notation) gewöhnten Anwender älterer HP-Taschenrechner. Beim HP-71B gab es nämlich Klammern und ein Gleichheitszeichen. Die Eingabe folgte also den gleichen Konventionen wie bei konkurrierenden Produkten von Texas Instruments und nannte sich AOS (Algebraic Operating System). Diese Neuerung lieferte Stoff für epische Debatten über Vor- und Nachteile gegenüber UPN. Aber HP wäre nicht HP, wenn dieses AOS nicht etwas Besonderes gewesen wäre: Bei den endlosen Diskussionen über die „richtige“ Eingabe-Methode zeigte sich immer wieder, dass der Vorteil der einen Methode der Nachteil der anderen Methode war. Bei AOS werden Formeln so eingegeben, wie sie auch in Büchern gedruckt bzw. von Hand geschrieben und in der Schule

gelehrt werden. Allerdings sieht man keine Zwischenergebnisse und muss daher Endergebnisse ausgiebig auf Plausibilität überprüfen. Bei UPN hingegen sieht man sofort nach jedem Schritt ein Zwischenergebnis. Fehler fallen also viel schneller auf. Dafür zahlt man den Preis einer an der Maschine orientierten Bedienung. HP schaffte aber eine Symbiose beider Welten, indem eine Formel zwar via AOS eingegeben wurde, dieses AOS aber bei jeder möglichen Gelegenheit ein Zwischenergebnis anzeigte [3]. Im Beispiel von **Tabelle 1** sieht man, wie das funktioniert. Man könnte nun glauben, dass durch diese geniale Fusion beider Techniken die Debatte UPN/AOS ein Ende gefunden hätte, aber die Tatsache, dass per UPN in der Regel weniger Tastendrucke benötigt wurde, hielt die Diskussion weiter am Leben, denn Speicher war damals knapp und teuer.

Eine weitere Neuerung des HP-71B war nicht so offensichtlich: Es war nämlich schon die damals noch nicht formal verabschiedete Norm IEEE 754-1985 zur Darstellung von Gleitkommazahlen und zur Durchführung mathematischer Operationen implementiert. Gegenüber der früher verwendeten BCD-Darstellung für Zahlen konnten mit Hilfe dieser Norm nun auch „unendlich“ und „Keine Zahl“ (NaN = Not a Number) dargestellt werden [4].

Mit über 240 unterschiedlichen Befehlen war der eingebaute BASIC-Interpreter sehr mächtig. Wem das noch nicht reichte, der konnte über sog. Language-Extension-Dateien (LEX-Files) den Sprachumfang selbst erweitern. Die dynamische Speicherverwaltung sorgte dafür, dass Variablen vor ihrer Verwendung nicht erst explizit deklariert werden mussten. Arrays änderten während des Programmlaufs abhängig von ihrer Belegung ihren Speicherbedarf. Dank Labels waren jetzt Sprünge möglich, die unabhängig von Zeilennummern waren, was sonst bei Code-Änderungen für viel Frust bei der Fehlersuche führen konnte. Eine Echtzeituhr und drei Timer machten den BASIC-Interpreter fit für Steuerungs-, Regelungs- und Messaufgaben. Auf dieses Thema wird bei den Erweiterungsmöglichkeiten noch einmal detaillierter eingegangen. Ein besonderes Feature war die Möglichkeit rekursiver Unterprogrammaufrufe, welche die Programmierung von Sortieralgorithmen etc. stark vereinfachten.

Programme und Daten ließen sich mit einem Passwort schützen. Wurde ein Passwort vergeben, dann wurde dessen Eingabe vom Rechner bei

EST[®] 2004

Retronik ist eine monatliche Rubrik, die antiker Elektronik und legendären Elektor-Schaltungen ihre Referenz erweist. Beiträge, Vorschläge und Anfragen telegrafieren Sie bitte an Jan Buiting (editor@elektor.com).

jedem Einschalten verlangt. Bei falscher Eingabe wurde der Rechner sofort ausgeschaltet. Die einzige Möglichkeit zum Reset des Passwortschutzes war, die Batterien zu entfernen. Doch war dann auch der Speicher gelöscht und es gab schlicht nichts mehr zu schützen [5].

Erweiterungsmöglichkeiten

Sechs Erweiterungsports sorgten für fast unbegrenzte Möglichkeiten. Vier dieser Ports befanden sich vorne an der Front und dienten zur Aufnahme von RAM- und ROM-Modulen (**Bild 1**). Zwei weitere Ports waren von unten zugänglich und für die Aufnahme eines Magnetkartenlesers und eines HP-Interface-Loop-Moduls vorgesehen.

Ein optionaler Magnetkartenleser stellte wohl die wichtigste Erweiterung dar. Er ermöglichte die dauerhafte Speicherung von Programmen und



1

Tabelle 1. Berechne: 2 + (3 × 5)

Eingabeverfahren	Tastendrucke	Anzeige
Reines AOS	2+(3*5)=	17
UPN	2 Enter	2
	3 Enter	3
	5 *	15
	+	17
HP 71B	2+(3*5)	2+(15)
	2+(3*5)=	17

Daten. War er vorhanden, konnte man einen 10'' (25,4 cm) langen Magnetstreifen von Hand (!) durch den oben rechts sichtbaren Kartenleser



ziehen, um Daten und Programme zu speichern oder wieder einzulesen (**Bilder 2 und 3**). Auf einem solchen Magnetstreifen konnten stolze 1,3 KB abgelegt werden.

Die ultimative Erweiterungsmöglichkeit aber war das Erweiterungsmodul HP-IL (HP-Interface-Loop). In **Bild 4** ist das Modul teilweise herausgezogen. Bei HP-IL handelt es sich um ein Netzwerk in Ring-Topologie, an das eine Vielzahl an peripheren Geräten angeschlossen werden kann. Die Anzahl möglicher Netzteilnehmer war fast nur durch den Geldbeutel beschränkt. Dies gilt insbesondere deshalb, weil damals HP gerne spöttisch mit „High Price“ übersetzt wurde. Unter den möglichen Geräten waren Drucker, Plotter, ein Kassettenlaufwerk, ein Floppy-Disk-Laufwerk, ein Video-Interface, eine RS232- und eine IEEE-488-Schnittstelle sowie ein Multimeter - alle mit HP-IL-Anschluss und vom Computer gesteuert. Dabei wurden alle Geräte außer dem Video-Interface mit Batterien bzw. Akkus betrieben, so dass sich die damit erstellten Systeme auch für den mobilen Einsatz eigneten. Während der Taschenrechner HP-41 lediglich mit einem HP-IL-Modul umgehen konnte, war der HP-71B in der Lage, sogar gleichzeitig zwei IL-Module in unterschiedlicher Konfiguration zu betreiben. Ein Modul konnte entweder als Master oder Slave fungieren. Ein HP-71B konnte also durchaus ein IL-Netz als Master betreiben und gleichzeitig als Slave an einem anderen Netz teilnehmen. Mit dem HP-3468-Multimeter und dem IEEE-488-Interface konnte ein HP-xx-B als Datenlogger und Auswertungszentrum betrieben werden.

Zusammen mit den ausgezeichneten Application-Packs in ROM-Modulen konnte man mit einem HP-71B eine besonders umfangreiche und dazu mobile Computing-Lösung für technisch-wissenschaftliche Anwendungen der 80er Jahre realisieren, die ihresgleichen suchte.

(140062)

Weblinks

- [1] Dollar-Kaufkraft: www.minneapolisfed.org/index.cfm?&TC=1
- [2] Eigenschaften des HP-71B: www.finseth.com/hpdata/hp71b.php
- [3] <http://h71028.www7.hp.com/enterprise/us/en/solutions/calculators-hp71b-math-machine.html>
- [4] IEEE 754: http://en.wikipedia.org/wiki/IEEE_floating_point
- [5] Hewlett-Packard Journal, Juli 1984: www.hpl.hp.com/hpjournal/pdfs/IssuePDFs/1984-07.pdf
- [6] Wikipedia zum HP-71B: <http://en.wikipedia.org/wiki/HP-71B>
- [7] HP-Rechner-Museum: www.hpmuseum.org/hp71.htm

Professionelle Hard- & Software zum Sonderpreis!

Exklusiv für Studenten!

Als neuer Vertriebspartner von National Instruments bietet Elektor ab sofort die Produkte der NI-Plattform für Ausbildung und Lehre für Studenten und schulische Einrichtungen an. Diese edukative Plattform vereint Hardware, Software und Unterrichtsmaterial, um Schülern und Studenten ein attraktives und inspirierendes Lernumfeld zu ermöglichen.

LabVIEW

Mit der Systemdesignsoftware *LabVIEW* können Studenten praxisorientiert anhand von Projekten und Systemen in einer einzigen Umgebung lernen und sich so Fähigkeiten und Verfahrensweisen aneignen, die im späteren Berufsleben unschätzbar sind.



Circuit Design Suite

Die *Circuit Design Suite* umfasst *Multisim* und *Ultiboard* und ist eine vollständige Plattform für Entwurf, Simulation und Validierung von Schaltplänen sowie den Leiterplattenentwurf. Die Suite verfügt über Funktionen, die speziell auf die Anforderungen von Studenten zugeschnitten sind, die bei der Entwicklung von elektronischen Konzepten hilfreich sind.



myDAQ

Bei *myDAQ* handelt es sich um ein kostengünstiges Datenerfassungsgerät, das überall und jederzeit Messungen und Analysen physikalischer Signale ermöglicht. *myDAQ* ist kompakt und portabel, sodass Studenten auch außerhalb des Labors und unter Einsatz branchenüblicher Werkzeuge und Methoden praktische Erfahrungen sammeln können.



Studentenversionen:

- Der Preis von *LabVIEW* und *Circuit Design Suite* beträgt jeweils nur 23,95 Euro.
- Das *myDAQ Education-Kit* bestehend aus *myDAQ* + 3 Software-DVDs (*LabVIEW*, *Circuit Design Suite* und *DIAdem*) kostet nur 174,95 Euro.

Jetzt bestellen unter www.elektor.de/ni-plattform!

Hexadoku Sudoku für Elektroniker

Die Tage werden wieder kürzer und der Sommer neigt sich dem Ende zu. Da kommt ein neues Hexadoku gerade recht, das zum gemütlichen Rätseln einlädt.

Geben Sie ihr Bestes, denn mit etwas Glück können Sie einen von fünf Buchgutscheinen gewinnen. Hierzu müssen Sie uns nur die richtigen Zahlen in den grauen Kästchen zuschicken.

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 einen von fünf Buchgutscheinen im Wert von 50 Euro gewinnen!

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 30. September 2014!

Die Gewinner des Hexadokus aus der Juni-Ausgabe stehen fest!

Die richtige Lösung ist: **F6B04**.

Einen Elektor-Buchgutschein über je 50 € haben gewonnen:

Walter Ranson, Jos van Goethem, Sam Abadani, Amir Omahic und Emil Cugini.

Herzlichen Glückwunsch!

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

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

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

Learn to program

Spielerisch programmieren lernen – ohne Tastatur

NEU!

Learn to program ist ein einfaches Lernspiel für Kinder und Schüler, das im Stil der Anfangsjahre des Computerzeitalters das Programmieren mit nur vier Tasten erlaubt. Auf dem Spielbrett befinden sich alle nötigen Ein-/Ausgabeeinheiten wie etwa eine LED-Ampel, ein Helligkeitssensor und ein Piezopiepser. Damit lassen sich verschiedene Programme entwickeln, wie zum Beispiel eine einfache Eieruhr, ein Wecker (der morgens wie ein Hahn „Kikeriki“ ruft) oder auch einfache Reaktionsspiele. Die Batterieversorgung macht die Programmierung netzunabhängig.



Für viele Elektroniker ist es selbstverständlich, Programme zu schreiben, einfache Regelungen zu implementieren oder Steuerungen zu realisieren. Nur ist einem dabei oft nicht klar, was für ein langer Weg es war, um soweit zu kommen. Mit diesem edukativen Spiel kann man das Mysterium des Programmierens Kindern anschaulich erklären.

Eigenschaften des Boards:

- 16 verschiedene Befehle • 8 Eingabetaster • Tonausgabe (Piezopiepser) • LED-Ampel • 4 weitere LEDs • Lichtsensor

Inhalt der Lernbox:

- Bestückte Mikrocontrollerplatine (Spielbrett)
- 73-seitiges A6-Handbuch

€ 44,95
CHF 56,95

Weitere Infos & Bestellung unter www.elektor.de/learn-to-program

Mikrocontroller verstehen und anwenden

Schnell und einfach mit Arduino und Elektor-Shield



NEU!

Mit diesem Buch erweitert der Leser seine Mikrocontroller-Kenntnisse auf Grund eigener Erfahrungen und Erfolgserlebnisse und wird dazu noch ganz nebenbei in die Welt des Arduino und seiner Entwicklungsumgebung eingeführt. Am Ende dieses vergnüglichen und fast spielerischen Lehrgangs stellen Begriffe wie I/O, Speicherplatz, Interrupts, Kommunikationsstandards, A/D-Konverter (und vieles mehr) keine Geheimnisse mehr dar und der Leser ist in der Lage, auch andere Mikrocontroller zu programmieren. Mit anderen Worten: ein erstes Mikrocontroller-Buch mit Happy End.

Dieses Buch ist für Sie geeignet, wenn Sie Anfänger auf dem Gebiet der Mikrocontroller sind, als Arduino-User bzw. -Enthusiast Ihre Kenntnisse vertiefen möchten, Elektronik studieren oder als Lehrer inspiriert werden möchten. Alle gezeigten Programme können kostenlos von der Elektor-Website heruntergeladen werden.

350 Seiten (kart.) • Format 14 x 21 cm
ISBN 978-3-89576-296-3

€ 42,00
CHF 52,95

Weitere Infos & Bestellung unter www.elektor.de/mcu-verstehen-anwenden



Mit Visual Basic in die analoge Welt

1 VB-Express und die Hardware

Visual Basic zählt nach wie vor zu den sehr weit verbreiteten Programmiersprachen. Seine Beliebtheit resultiert gerade für den Einsteiger aus der schnellen Erlernbarkeit und der einfachen Lesbarkeit des Programmcodes. Dieses Buch ist für Einsteiger in die Programmiersprache Visual Basic.NET gedacht, auch und gerade unter Berücksichtigung der Bedürfnisse von Elektronikern.

**287 Seiten (kart.) • ISBN 978-3-89576-270-3
€ 36,80 • CHF 45,95**

Grundlagen für Visual Basic-Einsteiger und VB6-Umsteiger

2 VB Express 2010

Dieses Buch unterstützt den Anwender bei den ersten Schritten mit Visual Basic, in dem es sich auf die Werkzeuge der Toolbox und deren Eigenschaften konzentriert, die zum Schreiben praktisch verwertbarer Programme notwendig sind. Zu jedem Thema findet

der Leser ausführlich kommentierte Beispielprogramme, die er selbst ausprobieren kann und die sich auf das Mindeste beschränken, was zum Starten der Software notwendig ist.

**284 Seiten (kart.) • ISBN 978-3-89576-269-7
€ 34,80 • CHF 43,95**

45 Experimente mit Hard- und Software für Elektroniker

3 Raspberry Pi

Dieses Buch beschreibt 45 spannende und interessante Projekte mit Raspberry Pi, wie zum Beispiel ein Wechselblinklicht, eine Motorregelung, Erzeugen und Verarbeiten analoger Signale, ein digitales Thermometer, ein Lichtmesser. Aber auch kompliziertere Projekte wie eine Motor-Geschwindigkeitsregelung, ein Webserver mit CGI (Common Gateway Interface) und Client-Server-Programme werden vorgestellt. Sie können dieses Buch als Projektbuch verwenden und die Projekte nachbauen, um sie dann in der Praxis einzusetzen.

**271 Seiten (kart.) • ISBN 978-3-89576-273-4
€ 39,80 • CHF 49,95**

Arrays und serielle Daten

4 LabVIEW 2

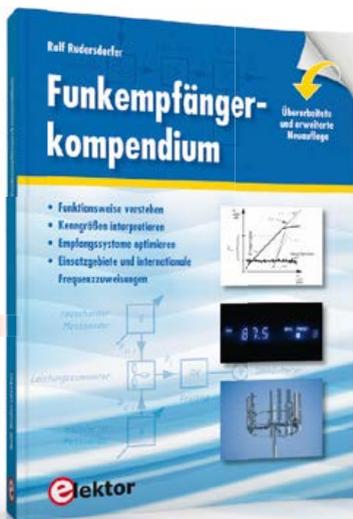
Der zweite Band der LabVIEW-Lehrbuchreihe beschäftigt sich u.a. mit Arrays, Cluster und den seriellen VISA-Funktionen. Als Erstes werden vier neue zusammengesetzte Datentypen (Enum, Ring, Array, Cluster) vorgestellt und deren Verwendung wird anhand zahlreicher praktischer Beispiele und Übungen erläutert. Danach wird es praktisch: Ein 8051er-Mikrocontrollersystem dient dabei als Datenquelle und -senke für verschiedene LabVIEW-VIs.

**248 Seiten (kart.) • ISBN 978-3-89576-274-1
€ 34,80 • CHF 43,95**

Der professionelle Ratgeber für Funkempfängertechnik

5 Funkempfängerkompodium

Wollten Sie schon immer wissen, wie sich die klassische Funkempfängertechnik fortentwickelt hat? Wie funktionieren professionelle Funkempfänger heute und was können sie leisten? Welche Empfangssysteme und Techniken stehen heute zur Verfügung? Möchten Sie

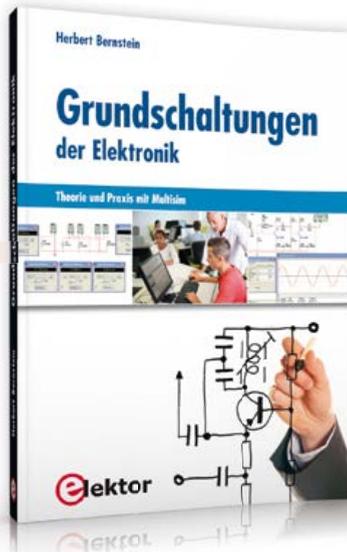


5

auch ausgefallene Anwendungen von Empfängern kennenlernen und wissen, wie ein Software Defined Radio (Digitalempfänger) nun wirklich funktioniert und was der letzte Stand der entsprechenden Technik kann? In diesem Buch findet man die Antworten!
397 Seiten (geb.) • ISBN 978-3-89576-276-5
€ 49,00 • CHF 61,95

Theorie und Praxis mit Multisim **6 Grundsaltungen der Elektronik**

Dieses Buch ist ein Nachschlagewerk über Elektronik mit praxisorientierten Fakten und ausführlichen Erklärungen. Der Autor hat selbst für komplexe Vorgänge oder Formeln praktische kurze Erklärungen und Näherungsrechnungen entwickelt, ohne die Darstellungen zu simplifizieren. Als Ausgangspunkt wurde das Simulationsprogramm Multisim gewählt, das zahlreiche Bauelemente und umfangreiche Messinstrumente zur Verfügung stellt.
360 Seiten (kart.) • ISBN 978-3-89576-286-4
€ 44,00 • CHF 54,95



6

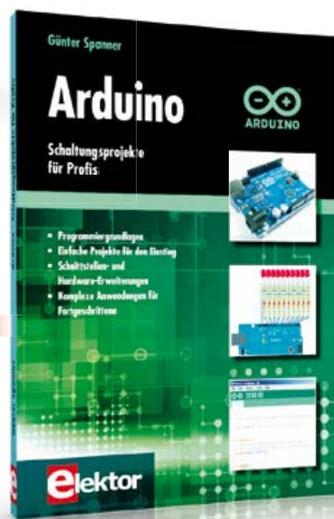
Schaltungsprojekte für Profis

7 Arduino

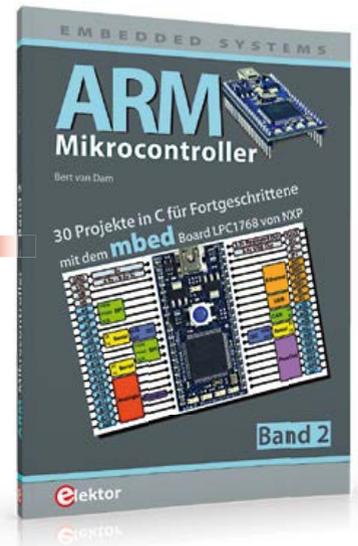
Für den großen Erfolg der Arduino-Plattform lassen sich zwei Ursachen finden. Zum einen wird durch das fertige Board der Einstieg in die Hardware enorm erleichtert; der zweite Erfolgsfaktor ist die kostenlos verfügbare Programmieroberfläche. Unterstützt wird der Arduino-Anwender durch eine Fülle von Software-Bibliotheken. Die täglich wachsende Flut von Libraries stellt den Einsteiger vor erste Probleme. Nach einfachen Einführungsbeispielen ist der weitere Weg nicht mehr klar erkennbar, weil oft detaillierte Projektbeschreibungen fehlen. Hier setzt dieses Buch an. Systematisch werden Projekte vorgestellt, die in verschiedene Themengebiete einführen. Dabei wird neben den erforderlichen theoretischen Grundlagen stets größter Wert auf eine praxisorientierte Ausrichtung gelegt.
270 Seiten (kart.) • ISBN 978-3-89576-257-4
€ 39,80 • CHF 49,95

30 Projekte in C für Fortgeschrittene **8 ARM-Mikrocontroller 2**

Die im Buch beschriebenen Projekte mit dem mbed-Board sind für Einsteiger in C und ARM-Mikrocon-



7



8

troller ausgelegt. Der mbed NXP LPC1768 nutzt Cloud-Technologie, ein revolutionäres Konzept in der Software-Entwicklung. Es bedeutet, dass man keinerlei Software auf seinem PC installieren muss, um den mbed zu programmieren. Das Einzige, was Sie brauchen, ist ein Webbrowser mit Internetzugang und einen freien USB-Anschluss an Ihrem PC.
243 Seiten (kart.) • ISBN 978-3-89576-271-0
€ 39,80 • CHF 49,95

Weitere Informationen zu unseren Produkten sowie das gesamte Verlagsortiment finden Sie auf der Ektektor-Website:

www.elektor.de

Ektektor-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

•Nächsten Monat in Elektor



Nostalgie mit Nixie

Nixie-Röhren sind Legende und Gegenwart zugleich, sie haben schon manchen Elektroniker in ihren Bann gezogen. Auch Elektor konnte der Magie nicht widerstehen und widmete diesen nostalgischen Ziffernanzeigen bereits mehrere Projekte. In der nächsten Ausgabe stellen wir eine Digitaluhr mit Nixies vor, die von einem GPS-Modul synchronisiert wird. Eine Synthese aus Klassik und Moderne!



USB-Hub mit RS232/RS485

Der technische Fortschritt ließ die herkömmliche serielle Schnittstelle des Computers aussterben, doch bisher ist sie nicht verzichtbar. Viele bewährte periphere Geräte, die durchaus noch auf dem Stand der Zeit sind, nutzen diesen Weg der Kommunikation. Unser USB-Hub ist universell ausgelegt: Drei USB-Ports werden von zwei voll duplex arbeitenden RS232-Schnittstellen und zwei RS422/485-Interfaces unterstützt.



Temperatur-Sensor-Board

Oft müssen Daten, die von Temperatursensoren aufgenommen werden, über weite Distanzen transportiert werden. Weil der RS485-Bus ein zuverlässiger Übertragungsweg ist, haben wir ein Sensor-Board mit einem Mikrocontroller ATtiny und einem RS485-Treiber entworfen. Das Board stellt Anschlüsse für vier Sensoren bereit. Die Firmware basiert auf dem ElektorBus-Protokoll, ferner ist ein Demo-Programm für den PC verfügbar.

Änderungen vorbehalten. Elektor Oktober 2014 erscheint am 24. September 2014. Verkaufsstellen findet man unter www.pressekaufen.de.

Rund um die Uhr und
sieben Tage die Woche

Projekte, Projekte, Projekte:
www.elektor-labs.com
Machen Sie mit!

A screenshot of the Elektor Labs website. The header features the 'elektor labs' logo and the tagline 'Sharing Electronics Projects'. Below the header is a navigation menu with 'Home', 'Proposals', 'In Progress', and 'Finished'. A Facebook social media widget is present, along with a 'Featured Video' section. The main content area is divided into three columns: 'Proposals', 'In Progress', and 'Finished', each with sub-sections for 'Active' and 'Popular' projects. A 'Create a Project' button is visible at the bottom right.

eC-reflow-mate

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 "eC-reflow-mate" ist ein kompromisslos konstruierter, universeller SMD-Lötofen – unverzichtbar für alle, die SMD-bestückte Platinen in Einzelanfertigungen oder Kleinserien entwickeln und produzieren.



Technische Daten:

- Betriebsspannung: 230 V/50 Hz
- Leistung: 3000 Watt
- Gewicht: 29 kg
- Abmessungen (B x H x T): 62,0 x 24,5 x 52,0 cm
- Heizung: Infrarot-Strahler kombiniert mit Umluft
- Bedienung: Funktionstaster und LCD am Ofen oder via USB mit PC-Programm
- 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)
- Mehrsprachige Menüführung

Besonderheiten:

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

Preis:
2.995,00 Euro
(zzgl. MwSt. und Versand)

Weitere Infos & Bestellung unter www.elektor.de/ec-reflow-mate



http://rch.lt/REDP



Bequem bestellen!

Einfach QR-Code scannen und alle Artikel, die sie zum Red Pitaya-Start benötigen, liegen im Warenkorb!

Micro SDHC-Card für den Red Pitaya

- Lesen: 24 MB/s
- Spezifikation: Class 6



SAMS MB-MS08DA 8 GB **10,95**

Modulartastkopf für den Red Pitaya

TESTEC

- Teilung: x 1 / x 10
- Bandbreite: 15/150 MHz
- Anstiegszeit: 24/2,3 ns



TESTEC LF 312 **16,95**

Micro-USB-Steckernetzteil für den Red Pitaya

- Eingangsspannung: 100-240 V, 50-60 Hz
- Ausgang: 5V DC, 1,2 A

gobay®



NT MICRO USB 1,2 1,40 m **3,95**

Cat. 5e-Netzwerkkabel für den Red Pitaya

- RJ45-Stecker
- Farbe: schwarz



PATCHKABEL 1 SW 1 m **0,80**

HF-Adapter für den Red Pitaya

SMA-Stecker auf BNC-Buchse



K 512 **1,15**



RED Pitaya

Open-Source-Messlabor, 50 MHz

CPU/FPGA:
 Xilinx Zynq 7010 SoC,
 Dual ARM® Cortex™-A9

USB-Port

16 digitale Ein- und Ausgangsports

2 HF-Ausgänge,
 50 MHz, 125 MS/s

4 analoge Hilfeingangskanäle
 4 analoge Hilfsausgangskanäle

2 HF-Eingänge,
 50 MHz, 125 MS/s

RED PITAYA
446,25

Features:

- Kanäle: 2
- Bandbreite: 50 MHz
- Abtastrate: 125 MS/s
- Horizontale Zeitbasis: 4 ns
- Vertikale Auflösung: 14 Bit
- DDR3 RAM 512 MB
- Stromversorgung: micro-USB
- Maße: 107 x 21 x 60 mm

Mit dem RedPitaya verwandeln Sie Ihr Smartphone, Tablet oder Ihren PC in ein umfangreiches Messinstrument!



Der Red Pitaya:

Beim Red Pitaya handelt es sich um ein Open-Source-Messgerät im Kreditkartenformat, welches Sie individuell auf Basis Ihrer Anforderungen konfigurieren können. Hierdurch können Sie eine Vielzahl preisintensiver Messgeräte wie Oszilloskope, Funktionsgeneratoren, Spektrum-Analysatoren usw. mit nur einem Gerät ersetzen.

Das Gerät wird in ein Netzwerk integriert, basiert auf einem GNU/Linux-Betriebssystem und kann mit einer Vielzahl von Software-Schnittstellen individuell programmiert werden. Die Bedienungsoberfläche für sämtliche Messvorgänge wird durch einen Webdienst mittels HTML-Seiten realisiert. Hierdurch wird die Steuerung der Messplattform mittels Smartphones, Tablet-PCs oder Computern ermöglicht.



Jetzt bestellen: www.reichelt.de

Bestell-Hotline: +49 (0)4422 955-333

