

**Twittern
mit E-blocks!**

www.elektor.de

Elektor



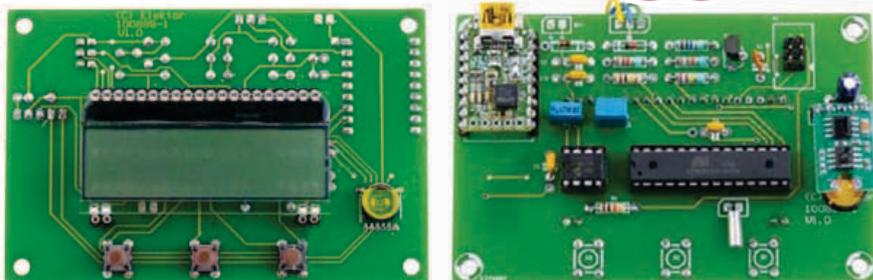
**Wiederverwendbar,
praktisch, schnell einzusetzen,
für alle Betriebssysteme**

FT232R USB/Seriell-Konverter

+ Digitaler Kompass mit ATM18

+ ARM Cortex-M3 User-Interface

USB-Langzeit- Wetter-Datenlogger



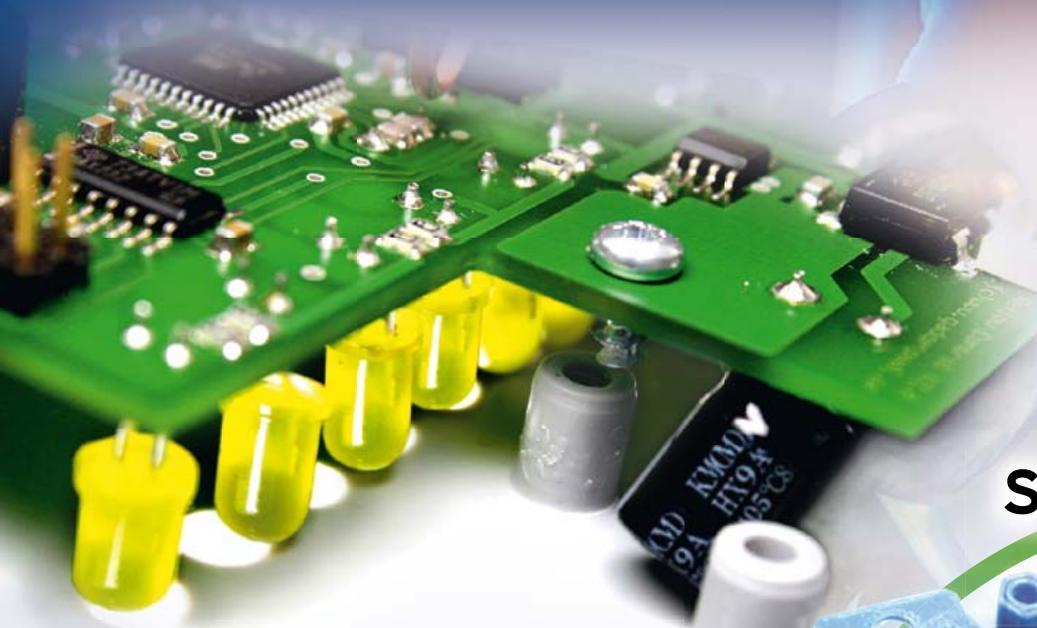
**Mit I²C-Sensoren für Luftdruck,
Luftfeuchte und Temperatur**



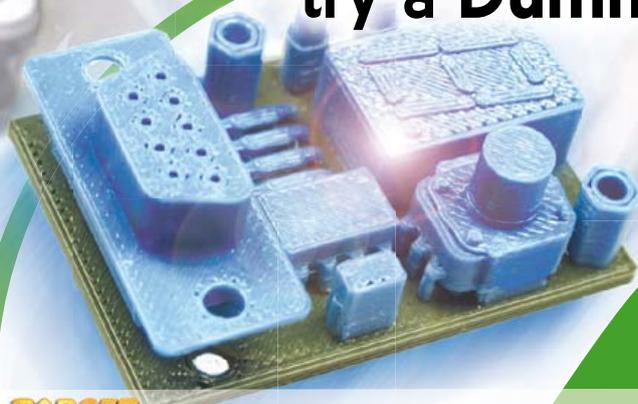


FITS-OR-NOT

Die Kollisionsprüfung zum Anfassen!



Save money
try a **Dummy!**



PCB	174,80 EUR
Bauteile	17,00 EUR
Bestückung	138,04 EUR
Preis	329,84 EUR*
Lieferzeit	11 AT

PASST NICHT!

TARGET 3001
Erstellt mit TARGET 3001! V15



29,75 €*
3 AT

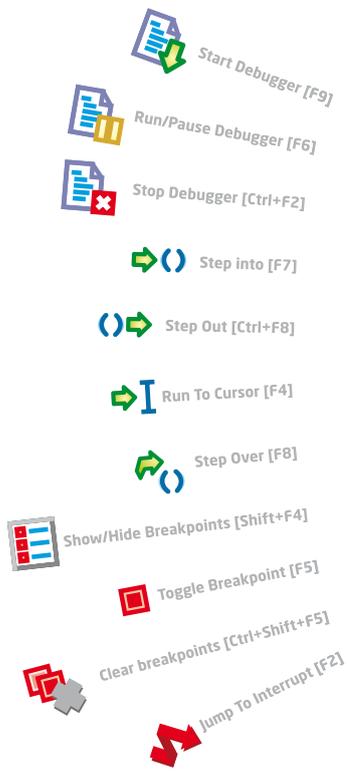
* Preisbeispiel für 1 dm² in 3 AT plus Bestückung 14x THT
zzgl. Versandkosten UPS Standard (GER) € 6,90

* Preisbeispiel für 8 cm² zzgl. Versandkosten UPS Standard (GER) € 6,90

www.pcb-pool.com/fitsornot

www.beta-layout.com

Beta
LAYOUT



mikroProg™ \$99

programmer und In-Circuit Debugger

Ein Programmierer, der alle beherrscht.



Ein Programmierer für 574 Controller
Alle PICs abgedeckt

Ein einziger Programmierer für 574 verschiedene Controller: PIC10, PIC12, PIC16, PIC18, dsPIC30/33, PIC24 und PIC32 - wir alles abgedeckt. Sie brauchen nie mehr einen anderen Programmierer!



Unterstützt durch In-Circuit Debugger
Schneller mikroICD™ Debugger

Debuggen Sie ihren Code und überwachen Sie die Ausführung des Codes auf dem Chip selbst. mikroICD™-Technologie bringt Ihnen bevorzugten mikroE Compiler die Leistung des Hardware-Debuggings.



Eine einzige Firmware sorgt für alles.
Unterstützung neuer Chips ist einfach

Das Hinzufügen von neuen Chips ist einfach. Laden Sie einfach die neueste mikroProg Suite Software herunter, und neue Controller werden verfügbar – unter Verwendung der gleichen Hardware und der gleichen Firmware!



Unterstützung in 9 mikroE-Compilern
Compiler-Kompatibilität

mikroProg™ wird mit mikroC-, mikroBASIC- und mikroPASCAL-Compilern für PIC, dsPIC und PIC32 unterstützt.

Auf ins Labor!

Es ist schon ein wenig her, aber die „alten Hasen“ bei Elektor erinnern sich noch: Unser Labor und die Redaktion gingen früher fließend ineinander über. Wie selbstverständlich entwickelten, bestückten und testeten die Redakteure „ihre“ Platinen, und sobald etwas zufriedenstellend funktionierte, lieferten sie dazu einen Artikel ab. Dass der „Schreiber“ mit allen Interna des Projektes vertraut war (und manchmal deswegen auch mal eine unruhige Nacht verbrachte) merkte man den Artikeln an: Hier schrieben Praktiker für Praktiker.

Inzwischen haben sich die Zeiten geändert: Die Jungs im Labor auf der einen und wir Redakteure auf der anderen Seite sind mit vielerlei speziellen Aufgaben betraut. Doch ist es interessant, dass alle Elektor-Redakteure, die damals noch nicht dabei waren, inzwischen gegenüber den älteren Kollegen in Sachen Kreativität aufholen. Von Clemens Valens, Chefredakteur der französischen Ausgabe, stammt unter anderem das leistungsfähige Zepter-Board. Zu diesem Heft hat Clemens, der sehr gern mit Controllern von NXP arbeitet, mit einem ARM-Cortex-M3-Board beigetragen, das man als flexibles User-Interface in eigene Projekte integrieren kann. Und ich kann Ihnen versichern: Der in Frankreich lebende Niederländer hat noch eine lange Reihe weiterer interessanter Ideen in der Pipeline! Thijs Beckers, der jüngste in unserer Runde, wird von nun an sowieso mit einem Bein in unserem Labor stehen: Der ausgewiesene Audio-Fan und praktizierende Musiker entwickelt dabei nicht nur eigene Projekte (wie zum Beispiel das Konferenz-Mikrofon für Meetings aus unserem Maiheft), sondern betreut auch die Rubrik „Labcenter“ in der Mitte unseres Heftes.

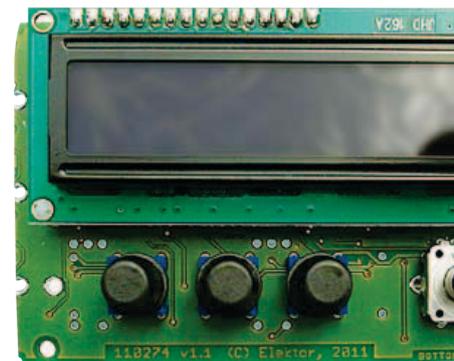
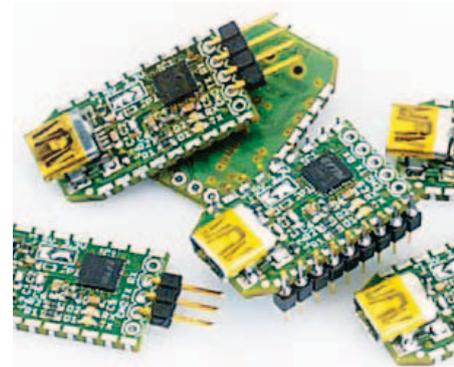
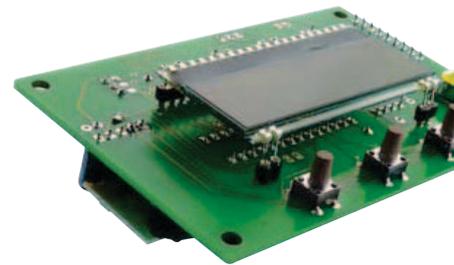
Und schließlich durfte auch ich selbst kreativ tätig werden: Im Zuge der Entwicklung des ElektorBusses häuften sich Utensilien wie Multimeter, Netzgerät und Oszi auf meinem Arbeitsplatz an, ich machte eine Art Crashkurs in Schaltungsentwicklung, Embedded-Programmierung und Projekt-Management und wurde schließlich sogar konzentriert lötlend in unserem Labor angetroffen (natürlich wurde all das von den Kollegen schmunzelnd registriert). Das Ergebnis meiner Bemühungen können Sie in diesem und den nächsten Heften weiterverfolgen – und ich verspreche schon jetzt, dass die Bus-Serie nicht langweilig werden wird...

Nicht nur die alten Hasen, sondern auch der Elektor-Nachwuchs arbeitet also daran, dass unser Magazin eine ganz besondere Elektronikzeitschrift ist und bleibt...

Jens Nickel

elektor

- 6 Impressum**
Who is who bei Elektor
- 8 Mailbox**
Briefe, E-Mails und Ideen
- 10 News**
Neuheiten und Nachrichten
- 14 USB-Wetterlogger mit Langzeitspeicher**
Sechs bis acht Wochen Aufzeichnungsdauer lassen sich schon mit drei Mignonzellen erreichen.
- 20 I²C-Sensoren für Druck, Temperatur und Luftfeuchte**
Mit den digitalisierten Messwerten erhält der angeschlossene Mikrocontroller auch gleich Kalibrierungsdaten über den I²C-Bus geliefert, sodass kein Abgleich erforderlich ist.
- 24 USB-FT232R Breakout-Board**
Mini-Platine zum einfachen Einbau in eigene Elektronik!
- 28 Vielseitiges HMI-Modul mit ARM Cortex-M3**
Ein Mensch-Maschine-Interface mit dem leistungsstarken ARM Cortex-M3-Controller LPC1343.
- 36 Preiswertes FPGA-Board**
Mikrocontroller dominieren bei der Lösung digitaler Aufgabenstellungen. Für einen richtigen Hardware-Fan sind FPGAs aber häufig die interessantere Möglichkeit, digitale Schaltungen zu ersetzen.
- 40 Hier kommt der Bus (7)**
Diesmal stellen wir ein einfaches Protokoll vor, mit dem sich bis zu vier Soll- oder Ist-Werte gleichzeitig übermitteln lassen. Es eignet sich nicht nur für die Hausautomatisierung, sondern auch für viele andere Aufgaben aus dem Bereich Messen, Steuern und Regeln.
- 45 Labcenter**
Geisterhaftes LED-Blinken
Alibaba



INHALT

42. Jahrgang
September 2011
Nr. 489

14 USB-Wetterlogger mit Langzeitspeicher



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

24 SB-FT232R Breakout-Board



Klein ist er, dieser USB-nach-TTL-Wandler, nicht viel größer als der angegossene Stecker eines USB-Kabels! Seine nützlichen Dienste kann die Mini-Platine unter Windows, Linux und anderen Betriebssystemen entfalten. Mit dem praktischen Konverterboard lassen sich eigene Schaltungen einfach um einen USB-Anschluss erweitern und leicht USB/RS232- oder RS485-Wandler realisieren. Doch damit sind die Grenzen noch lange nicht erreicht.

28 Vielseitiges User-Interface mit ARM Cortex-M3



Dreiviertel aller Applikationen mit Mikrocontrollern haben eine recht ähnliche Konfiguration: Die Hauptzutaten sind der Controller, ein LC-Display sowie einige Drucktaster. Mit unserem „Human-Machine-Interface“ schlagen wir einen alternativen, flexibleren Weg ein, der den Einsatz unterschiedlicher LCDs und eine variable Anzahl Taster erlaubt. Der neueste ARM Cortex-M3-Controller LPC1343 macht das Board sehr leistungsstark, trotzdem ist der Umgang damit erstaunlich einfach.

54 2,4-GHz-Fernsteuerung für Modellflugzeuge



Schon seit einigen Jahren gibt es Funkfernsteuerungen für Flugmodelle, die im 2,4-GHz-Band operieren. In diesem Frequenzbereich ist die Störsicherheit groß, was neue Möglichkeiten für die Telemetrie eröffnet. Das hier vorgestellte Projekt erlaubt sogar die nachträgliche Aufrüstung von älteren 35/40/41-MHz-Systemen auf zeitgemäße 2,4 GHz. Außerdem kann es leicht nach eigenen Vorstellungen modifiziert werden.

Leckere Pizzas
Wetterprobleme
Kleine Fallgruben

50 ATM18-Kompass

Das Kompassmodul CMPS03 verwendet zwei Sensoren, um das Magnetfeld der Erde zu messen; die Berechnung der Orientierung erfolgt durch einen integrierten Mikrocontroller. Auf den Winkel-Wert kann man über I²C zugreifen, was wir mit unserem ATM18 gleich ausprobiert haben.

54 2,4-GHz-Fernsteuerung für Modellflugzeuge

Neue Möglichkeiten für die Telemetrie zeigt dieses Open-source-Projekt auf.

62 Audio-DSP-Kurs (3): Das Board

Das DSP-Board ist sofort „stand alone“ einzusetzen und genügt mit 24 bit/192 kHz auch sehr hohen Qualitätsansprüchen!

70 Leserprojekt: Kompaktes Blinkrücklicht

Stroboskop-Licht für Radfahrer: Die Schaltung ist einfach gehalten, so dass man mit Blinkfrequenz und LEDs nach Herzenslust experimentieren kann.

72 Twittern mit E-Blocks

Wir zeigen, wie man eine WLAN-Karte in ein Mikrocontroller-System einbindet, um eine eigene Website zu hosten. Sogar Twitter-Meldungen lassen sich absetzen!

75 Hexadoku

Sudoku für Elektroniker

76 Retronik

Die Chaos-Maschine

80 Elektor-Shop

Bücher, CDs, DVDs, Bausätze & Module

84 Vorschau

Nächsten Monat in Elektor

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



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

IMPRESSUM

42. Jahrgang, Nr. 489 September 2010
Erscheinungsweise: 11 x jährlich (inkl. Doppelheft Juli/August)

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

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

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

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

Internationale Chefredaktion **Wisse Hettinga**

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

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

Redaktionssekretariat **Hedwig Hennekens**

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

Grafische Gestaltung und Layout
Giel Dols, Mart Schroyen

3-tägiges Fachseminar C-Programmierung für Mikrocontroller



Von den Autoren/Entwicklern des Elektor- μ C-Fernlehrgangs



Die Beschäftigung mit Mikrocontrollern ist sehr reizvoll und vielseitig, teilt sie sich doch in zwei große Gebiete der Elektrotechnik auf: zum einen ist da die Hardware-Seite, die sich mit der Konzeption und der Entwicklung kompletter Systeme auseinandersetzt. Auf der anderen Seite ist die Hardware aber nicht funktionsfähig ohne die zugehörige Betriebssoftware für den Controller. Beide Entwicklungsgebiete sind also aufs Engste und unzertrennbar miteinander verzahnt.

In diesem 3-tägigen Seminar lernen Sie nun verschiedene Facetten der Softwareerstellung für Mikrocontroller kennen.

Wir stützen uns dabei auf ein modernes Mikrocontroller-System mit dem leistungsfähigen AT89C51CC03 (8051er-Familie) der Firma Atmel ab und erläutern Schritt für Schritt, von Anfang an, die Softwareentwicklung in der international gebräuchlichen Programmiersprache C. Eine einfach zu handhabende integrierte Entwicklungsumgebung (IDE) zur Erstellung und zum Austesten von C-Programmen bildet die Software-Basis des Seminars.

Praxisgerechte Beispiele, die von den Teilnehmern während des Seminars programmiert und ausgetestet werden, vertiefen sofort das Erlernte und bieten vielfältige Möglichkeiten für eigene Weiterentwicklungen. Da C weitgehend genormt ist, lässt sich ein Transfer unserer Ergebnisse auch sehr einfach auf andere Mikrocontroller-Familien und andere C-Entwicklungsumgebungen durchführen.

Die Referenten:

Prof. Dr.-Ing. Bernd vom Berg und Dipl.-Ing. Peter Groppe von der Technischen Fachhochschule (TFH) Georg Agricola zu Bochum sind seit mehr als 15 Jahren auf dem Gebiet der „Mikrocontroller-Technik für Lehre und Ausbildung“ tätig. In mehr als 10 praxisnahen Lehr-, Lern- und Arbeitsbüchern und zahlreichen Applikationsveröffentlichungen haben sie den „Anfängern“ die Grundlagen und Anwendungen der Mikrocontroller-Technik näher gebracht. Sie sind ebenfalls die Autoren des sehr erfolgreichen Elektor-Mikrocontroller-Fernlehrgangs.

Veranstaltungsort/-termin:

Hanau 13.09. bis 15.09.2011

Teilnehmergebühr (zzgl. MwSt.):

1240,00 Euro

Elektor-Abonnenten erhalten 5% Rabatt!



Im Preis sind sämtliche Dokumentation (Handouts/Seminarunterlagen), Aushändigung des Teilnahmezertifikats, Imbiss mit Getränken und Mittagessen inbegriffen.

Weitere Infos & Anmeldung unter

www.elektor.de/c-prog

Geschäftsführer/Herausgeber: Paul Snackers

Marketing/Vertrieb (Leitung): Carlo van Nistelrooy

Anzeigen (verantwortlich): Irmgard Ditzgens
ID Medienservice

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

E-Mail: service@id-medienservice.de

Es gilt die Anzeigenpreisliste Nr. 40 ab 01.01.2010

Vertriebsgesellschaft: IPS Pressevertrieb GmbH
Postfach 12 11, 53334 Meckenheim

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

E-Mail: elektor@ips-pressevertrieb.de
Internet: www.ips-pressevertrieb.de

Vertrieb Österreich

Pressegroßvertrieb Salzburg/Anif - Niederalm 300
Tel. +43/62 46/37 21-0

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

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

© 2011 elektor international media b.v.

Druck: Senefelder Misset, Doetinchem (NL)



ISSN 0932-5468

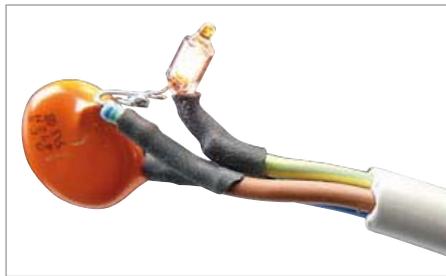
Zurück zur Erde

Labcenter-Beitrag in Elektor 06/2011, S. 45 (110396)
 Zu dem Kurzbeitrag über das seltsame Verhalten einer Schutzleiter-Testschaltung erhielten wir eine Reihe von Zuschriften mit Hinweisen zur Lösung des Rätsels. Hier zwei davon, die ganz richtig lagen:

Könnte es sein, dass Ihr belgischer Leser in einer Gegend wohnt, wo die Netzspannung 220 V ist und nicht wie heute üblich 220/380 V beträgt? (Ich nenne bewusst die alten Spannungswerte). Früher waren die Drehstrom-Ortsnetze nur 220 V. Das bedeutet, dass der Netztrafo in Dreieck geschaltet war. Es war somit kein Sternpunkt vorhanden und zwischen zwei Außenleitern lagen 220 V, die Phasen hingen „in der Luft“. So wie bei Ihnen im Labornetz durch den Trenntrafo. Heute sind die Ortsnetz-Trafos grundsätzlich in Stern geschaltet. In den 1950er- und 1960er-Jahren wurden diese alten Netze (meist kleine Dörfer) dann umgerüstet. Auf älteren Drehstrom-Motoren findet sich die Volt-Angabe 220/380. In so einem alten Netz ist der Motor somit in Dreieck zu schalten.

Karl-Heinz Ziener

Ich denke, die Lösung der Frage ist ganz einfach. Normalerweise wird ein TN-3x380-V-Netz verwendet, und zwar in Sternschaltung. Das bedeutet: Zwischen den Leitern liegen 380 V, zwischen den Leitern (egal ob L1, L2 oder L3) und dem Nullleiter sind es 220 V (nach alter Definition, siehe auch <http://de.wikipedia.org/wiki/Niederspannungsnetz>). Es gibt aber in einigen Gegenden auch 3x220-V-Netze, diese sind dann in Dreieckschaltung ausgeführt und es gibt keinen zentralen Nullleiter (IT-Netze). Wahrscheinlich ist bei dem Leser aus Belgien ein solches Netz im Einsatz. Ich weiß, dass in Berlin (ehemals Ostberlin) noch an vielen Stellen dieses Netz vorherrschte und auch, dass in Berlin-Mahlsdorf eine Gegend existiert, in der ein



solches Netz noch genutzt wird. Da dies ja vom Energieversorger bereitgestellt wird, hat man als Abnehmer keinen Einfluss darauf, aber ein Erkennungsmerkmal für diese Netze sind zwei Sicherungen pro Stromkreis.

Michael Westphal

Tatsächlich scheint das alte 3x220-V-Netz (mit heute 3x230 V) in Belgien noch relativ verbreitet zu sein. Wie bei unserem Labornetz mit Trenntrafo werden an jede Steckdose zwei Phasen angeschlossen, zwischen denen dann 230 V liegen. Da beide Leitungen Spannung führen, ist jede Leitung einzeln zu sichern (zwei Sicherungen pro Kreis). Dass der dann im Halbleiterheft veröffentlichte Schutzleiter-Tester noch aus einem anderen Grund eigenartig ist, zeigt die folgende Lesermail:

Zweifelhafter Schuko-Tester

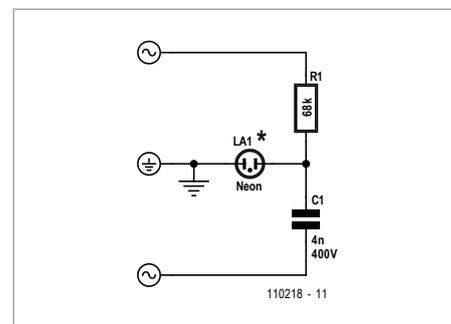
Schutzkontakt-Prüfer, Elektor 7-8/2011, S. 83 (110218)

Die vorgeschlagene Schaltung eignet sich leider NICHT zum Erkennen eines nicht angeschlossenen Schutzkontakts. Der Widerstand und der Kondensator stellen zusammen einen Spannungsteiler von ca. 10:1 dar. Wenn man die Schaltung an ein ordentlich geschütztes Netz anschließt, passiert das, was im Elektor-Juniheft im Labcenter-Beitrag „Zurück zur Erde“ beschreiben wurde: In der einen Richtung leuchtet die Glühlampe und in der anderen nicht. Mal sieht die Glühlampe ca. 225 V und leuchtet, mal sieht die Glühlampe 25 V und leuchtet nicht. Wenn der Schutzkontakt kurz vor dem „Prüfer“ defekt ist, dann geht in der Tat die Glühlampe nicht an. Aber es kann leider auch passieren, dass der Schutzkontakt irgendwo weiter entfernt nicht ordentlich Kontakt hat. Dann stellt die Leitung bis zur Steckdose einen kapazitiven Spannungs-

teiler zum Schutzleiter dar. Im Falle einer „verdrihten“ Leitung (z.B. Mantelleitung) sogar einen 1:1 Spannungsteiler. Somit liegt das Glühlämpchen zwischen 125 V und 25 V - also an 100 V. Das reicht, um die Glühlampe leuchten zu lassen. Ihr solltet die Leser darauf hinweisen, dass diese Schaltung nicht als Prüfer geeignet ist.

Walter Mayer

Sie weisen zu Recht darauf hin, dass der „Prüfer“ nur prüft, ob ein Schutzleiter an der Steckdose angeschlossen ist. Ob dieser Schutzleiter auch einwandfrei geerdet ist, lässt sich mit dieser einfachen Schaltung nicht prüfen.



BASCOM und/oder C

Bei vielen AVR Projekten wird die Software in BASCOM AVR geschrieben. Die freie Version ist aber auf 4 KB Code limitiert. Da in AVRSTUDIO5 ein freier „C Compiler“ integriert ist und auch in AVRSTUDIO4 die freie Software Winavr installiert werden kann, wäre es schön, wenn für zukünftige Projekte die Software auch in „C“ verfügbar wäre. Eventuell in Basic und in „C“, wie bei einem früheren Elektor-Projekt bereits geschehen.

Johann Wenzels

Sie haben Recht, am besten wäre beides, wobei wir den Code ja meistens von externen Entwicklern zur Verfügung gestellt bekommen, die eben nur in einer der Welten zuhause sind. Und für viele BASCOM-Programmierer spielen eben die Codelimitierung und gewisse Limitierungen bei der Strukturierung der Programme eine untergeordnete Rolle, entscheidend ist es, schnell zum Ziel zu kommen.

Für uns wäre der Aufwand für eine Neu-Implementierung in C samt Dokumentation, Kommentaren und Test dann natürlich sehr hoch. Bei unserem ElektorBus-Projekt wird aber nun auch in C programmiert und AVRStudio5 verwendet.

Jens Nickel

MailBox

In dieser Rubrik veröffentlichen wir Kritik, Meinungen, Anregungen, Wünsche oder Fragen unserer Leser. Die Redaktion trifft die Auswahl und behält sich Kürzungen vor. Bitte geben Sie immer an, auf welchen Artikel und welche Ausgabe (Monat/Jahr) sich Ihr Schreiben oder Ihre Mail bezieht.

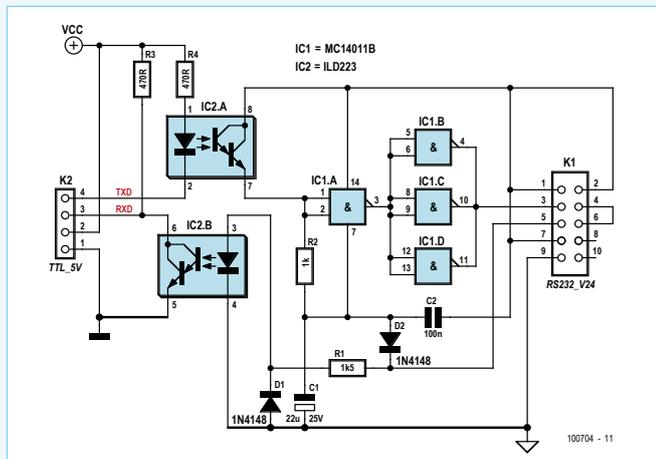
Sie erreichen uns per E-Mail redaktion@elektor.de, per Fax (02 41/88 909-77) oder unter der Anschrift: Redaktion Elektor Susterfeldstr. 25 - 52072 Aachen

Updates und Ergänzungen

RS232-Pegelwandler mit Isolation

Elektor 7-8/2011, S. 38 (100704)

Es wurde versehentlich eine noch nicht korrigierte Version des Schaltplans abgedruckt (u.a. Masseanschlüsse verwechselt, so dass keine Isolation gegeben ist). Der korrigierte Schaltplan ist hier abgebildet und auch im Artikel-PDF enthalten, das man unter www.elektor.de/100704 kostenlos herunterladen kann.



Einfacher Rechteck-Generator & Tester

Elektor 7-8/2011, S. 67 (100763)

Der Schaltplan ist wie folgt zu korrigieren:
Die +VEE-Leitung (z.B. Pluspol C10) ist mit der „Plus“-Leitung des Schaltungsteils mit IC1 (z.B. mit der Kathode D2) zu verbinden.

I²C-Bedieneinheit

Elektor 7-8/2011, S. 72 (110079)

In beiden Schaltplänen sind die Signale SDA und SCL bei den Anschlusspunkten verwechselt worden. Die Pinbelegung am IC ist jeweils korrekt.

Tochterblitz für Unterwasserkamera

Elektor 7-8/2011, S. 80 (100584)

Im Schaltbild ist die BPW34 (T1) als Fototransistor dargestellt. Die BPW34 ist aber eine Fotodiode, die in der Schaltung mit der Kathode an +3V3 liegt.

Ab 0 mA einstellbare Konstantstromquelle

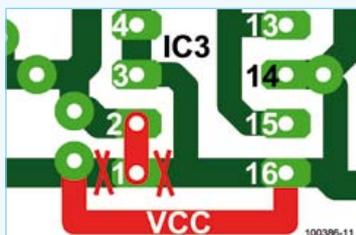
Elektor 7-8/2011, S. 105, (100935)

Für T1 (PNP-Transistor) wurde im Schaltplan versehentlich ein BC547B (NPN) angegeben. Es muss aber ein PNP-Transistor sein, und wegen der höheren Belastbarkeit ist ein BC327-40 einem BS557B vorzuziehen.

VGA-Grafikkarte

Elektor 5/2011, S. 56 (Platine 100147-1)

Einige Platinen booten nicht immer richtig. Das Problem ist, dass der dsPIC nicht immer einen Low-Pegel am Anschluss SO erkennt. Das Problem wird gelöst, wenn man Pin 1 (SI) von IC3 an GND legt (statt an VCC). Diese Änderung lässt sich leicht durchführen: Entweder Pin 1 aus der Fassung ziehen und an Pin 2 von IC3 löten oder die Korrektur entsprechend der hier gezeigten Abbildung ausführen.



Event-Kalender

Workshops • Seminare • Weiterbildungen

➔ Top-Fachleute aus der Branche referieren über ein faszinierendes Thema!



C-Programmierung für Mikrocontroller

Hanau 13.09. bis 15.09.2011

www.elektor.de/c-prog

Workshop zum µC-Fernlehrgang 2

Hanau 17.09.2011

www.elektor.de/tfh-workshop

PIC-Mikrocontroller-Programmierung in C

Karlsruhe 20.09. + 21.09.2011

www.elektor.de/pic-prog

CAN/AVR für Einsteiger

Hamburg 21.09.2011

Berlin 28.09.2011

www.elektor.de/can-workshop

NEU!

Grafische AVR-Programmierung mit Flowcode

Hamburg 22.09.2011

Berlin 27.09.2011

Dortmund 10.11.2011

www.elektor.de/avr-prog

AVR-Mikrocontroller (für Einsteiger)

Hanau 24.09.2011

Hamburg 05.11.2011

Hannover 10.12.2011

www.elektor.de/avr-workshop

FPGA – Der Logikbaukasten für Jedermann

Hannover 29.09.2011

Hanau 05.10.2011

www.elektor.de/fpga-seminar

Eagle PCB und Design

Hannover 29.09.2011

Hanau 06.10.2011

www.elektor.de/eagle-seminar

LabVIEW meets µC

München 07.10. + 08.10.2011

www.elektor.de/lv-seminar

Echtzeitbetriebssysteme in Theorie und Praxis

Hamburg 02.11. bis 04.11.2011

München 06.12. bis 08.12.2011

www.elektor.de/ezb-systeme

events

Änderungen vorbehalten.

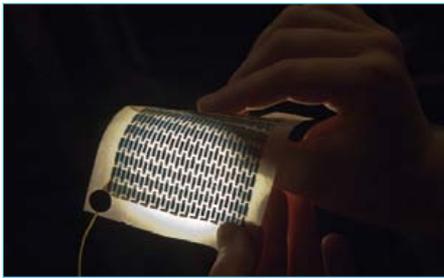
Weitere Infos unter

www.elektor.de/events

Von Phil Knurhahn

Solarzellen aus dem Drucker

Hört sich etwas seltsam an, ist aber – im Labor – schon Realität: Am Institute of Advanced Materials des Massachusetts Institute of Technology (MIT) wurden Solarzellen entwickelt, die man auf einem Stück Papier ausdrucken kann, es lassen sich aber auch Kunststoff und sogar Textilien nutzen. Der Druckprozess arbeitet mit Dampf bei ungefähr 120°C und nicht mit Pulvertoner oder Flüssigkeiten. Allerdings läuft das Verfahren nicht in einem einzigen Druckvorgang ab, sondern man braucht insge-



samt fünf Layer, die übereinander gedruckt werden müssen – und das Ganze auch unter Vakuum. Die Prototypen waren auf PET-Plastik aufgebracht (Bild: Patrick Gillooly / MIT), sie ließen sich 1000 Mal hin- und her falten. Wegen der Leichtigkeit des Substrats könnten solche Solarzellen gegenüber den heute verwendeten ziemlich rekordverdächtige spezifische Leistungen (in Watt/kg Solarzelle) erreichen.

<http://web.mit.edu/newsoffice/2011/printable-solar-cells-0711.html>

Wieder mal ein Flugauto

Diese Variante eines „Flugautos“ hat gerade in den USA ihre Flugzulassung bekommen (die Straßenzulassung war schon länger erteilt). Das „Transition“ genannte Objekt wird vom Unternehmen Terrafugia hergestellt, hinter dem sich MIT-Absolventen verbergen. Die Flugtests wurden in Plattsburgh (am Lake Champlain, in Norden des Bundesstaats New York) absolviert (Bilder: Terrafugia). Zum Fahren auf der Straße werden die Flügel per Knopfdruck einfach hochgeklappt (was rund eine Minute dauert). Auf der Straße kommt das „Fahrzeug“ auf



etwa 105 km/h, als „Flugzeug“ in der Luft auf maximal 185 km/h. Die Länge der Startbahn muss mindestens 850 m betragen. Bei einer mittleren Geschwindigkeit von 170 km/h liegt der Treibstoffverbrauch bei rund 19 Liter/h. Das „Transition“ ist rund 2 m hoch, 2,3 m breit (mit zwei Sitzen nebeneinander) und 6 m lang. Die Spannweite über die ausgefahrenen Flügel beträgt 8 m, das Leergewicht wird mit 440 kg, die Zuladung mit 210 kg angegeben. Bleibt noch der Preis: Mit rund 180.000 € ist das Flugauto viel preiswerter als ein Lamborghini, fällt aber trotzdem genauso auf. Erste Auslieferungen sind für Ende 2012 geplant.

www.terraflugia.com

100 Mal schneller als Flash

Weltweit sind die Forscher auf der Suche nach neuen Speichertechnologien. Unter den zahlreichen Optionen scheint sich jetzt eine durchzusetzen, für die die Forscher von IBM in Rüschlikon bei Zürich entscheidende Vorteile erarbeiten konnten: Nichtflüchtigkeit, gleich vier Speicherbits pro Zelle (und damit hohe Speicherdichte) sowie sehr gute Beständigkeit gegen Alterungsprozesse. Es sind die „Phase-Change-Memories“ (PCM), die in den nächsten fünf Jahren das Rennen machen könnten. Zwar sind auch andere schon existierende Speicher „nicht-flüchtig“ – aber die von den IBM-Forschern entwickelten Prototypen sind 100 Mal schneller als Flash-Speicher und schaffen leicht 10 Millionen Schreib- und Lesezyklen. Zum Vergleich: Flash-Speicher bringen es auf etwa 30.000 Zyklen. Erreicht wurden diese

Werte durch Verbesserungen im Schreib- und Leseprozess. Die Möglichkeit, vier Bits in einer einzigen Speicherzelle abzulegen, ist dabei einer der großen Fortschritte. PCMs nutzen die Widerstandsveränderung eines Speichermaterials, das zwischen zwei Steuerelektroden angeordnet ist. Diese Speicherschicht ist ein Gemisch aus verschiedenen Elementen, das bei der Ansteuerung von außen seinen Zustand verändert – von kristallin (mit niedrigem elektrischen Widerstand) zu amorph (mit hohem elektrischem Widerstand) oder zurück. Die sehr dünne Speicherschicht wird hierzu durch elektrische Impulse kurzzeitig aufgeheizt, wobei die Länge der Pulse und die Stromstärke bestimmen, in welche Richtung der Phasenwechsel stattfindet. Zur Speicherung von vier Bits werden verschiedene Widerstands-„Potentiale“ genutzt. Die Latenzzeit – das ist die Zeitspanne, die zwischen Steuerimpuls und Reaktion des Materials liegt – konnte auf 10 Mikrosekunden heruntergedrückt werden – das ist um zwei Größenordnungen besser als bei existierenden Flash-Speichern.

www.zurich.ibm.com/sto/memory

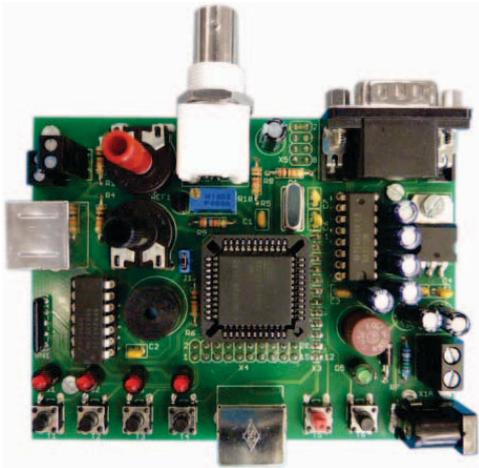
Weniger als 20 Euro ...

... soll der Raspberry Pi Computer kosten, der gerade mal so groß ist wie ein USB-Stick. Natürlich braucht man noch ein Fernsehgerät als Display, eine Maus und eine Tastatur zum Betrieb – sonst aber lässt der Winzling für den angedachten Zweck (ein preiswerter Schulcomputer für alle Schüler) wenig zu wünschen übrig. Als Prozessor dient ein ARM11 mit 700 MHz Taktfrequenz, der auch im iPhone 3G steckt. Als Arbeitsspeicher kommt SDRAM mit 128 MB zum Einsatz. Maus und Tastatur werden über USB angeschlossen. Als Speichermedium für die persönlichen Dateien sollen SD/MMC/SDIO-Speicherkarten verwendet werden, die man seitlich in den Mini-Computer schieben kann. Die Kosten für den PC können deshalb so niedrig gehalten werden, weil mit der Linux-Distribution Ubuntu ein frei erhältliches Betriebssystem verwendet wird. Die Bildschirmoberfläche ähnelt entfernt der von Windows und erlaubt alle gängigen Arbeiten, vor allem auch das Surfen im Internet. Das Charity-Unternehmen Raspberry Foundation (in England) sucht nun noch nach frei erhältlicher Unterrichtssoftware.

www.raspberrypi.org

Neuer Mikrocontroller-Kurs für Einsteiger

An Mikrocontrollern kommt man in unserer heutigen technisierten Welt nicht mehr vorbei: Angefangen bei elektronischen Spielzeugen und elektrischen Zahnbürsten, über Haushalts- und Unterhaltungsgeräte bis hin zu Autos, Produktionsmaschinen, Flugzeugen und Raumfähren: Überall wacht, steuert und regelt eine verborgene elektronische Intelligenz.



Ob Lehrer/Ausbilder, Student, Schüler oder Hobbyist: Ein Einstieg in diese faszinierende technische Welt ist nicht nur interessant, sondern macht auch Spaß und kann darüber hinaus auch beruflich lohnend sein. Alle nötigen „Werkzeuge“ findet man im neuen „First Step“-Paket, das bei Elektor erhältlich ist. Das fertig bestückte und getestete „First Step“-Board und drei exakt darauf abgestimmte Arbeitshefte (plus Software-CD) machen die ersten Experimente mit einem Mikrocontroller zum Kinderspiel.

Der Kurs setzt beim Teilnehmer keine besonderen Vorkenntnisse voraus. Nach dem Motto „Learning by Doing“ sind die drei Arbeitshefte sowohl für das Selbststudium als auch als Grundlage für einen fort-

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

schrittlichen Mikrocontroller-Schulunterricht geeignet.

Die Programmierung des gesamten Systems erfolgt in der Programmiersprache C. Der Programm-Download vom Entwicklungs-PC zum Mikrocontroller erfolgt dabei direkt über die serielle COM-Schnittstelle; spezielle Programmieradapter oder Zusatzschaltungen sind nicht erforderlich. Das „First Step“-Paket ist jetzt zum Einführungspreis erhältlich, Support gibt es in einem eigenen Forum (www.elektor.de/fs-forum).

www.elektor.de/first-step

Neue Xmega-Controller

Atmel hat eine neue Xmega-Generation vorgestellt. Die zur Xmega A-Serie pin- und funktionskompatiblen Bausteine wurden um Full Speed USB (Device) und einige technische Details ergänzt, außerdem wurde der Analogteil verbessert. Der „Multi-Package Mode“ ermöglicht hohe USB-Übertragungsraten ohne nennenswerte CPU-Last. Die USB-Schnittstelle mit 31 Endpoints kommt ohne externe Komponenten - sogar ohne Quarz - aus und unterstützt die Transfer-Modi Control, Interrupt, Isochronous und Bulk.

Features: Bis zu zwei ADCs mit 12 bit Auflösung und 2 MSamples/s Abtastrate ab 1,6 Volt Vcc; 12-bit/1-MSamples/s-DAC mit zwei Sample & Hold-Stufen; 8-Kanal-Event-System zur CPU-unabhängigen, zeitlich determinierten Steuerung der Peripherie; 4-Kanal-DMA; „picoPower“-Stromspartechnologie mit typisch 0,6 µA Stromaufnahme

im Sleep Mode bei laufendem RTC und vollem Datenerhalt; 32 MHz Harvard RISC CPU; AES- und DES-Crypto-Module; bis zu 32 PWM-Ausgänge, acht USARTs, vier TWI/I²C- und vier SPI-Schnittstellen; CRC-Generator; Seriennummer; extra Bootloaderbereich; EEPROM; On-Chip-Debugger und vieles mehr!

In die neuen Xmegas sind zahlreiche Kunden-Anregungen eingeflossen:

- Rail-to-Rail-Betrieb des ADCs nun ohne externe Komponenten,
- Auf CRC-16 (RC-CCITT) und CRC-32 (IEEE 802.3) erweitertes CRC-Modul, auch für Kommunikationsdaten,
- Konstantstromquelle,
- High-Resolution Extension für PWM-Signale mit 4 ns Auflösung,
- Alternative Pins für PWM, SPI, USART und 32,768-kHz-Quarz, dadurch größere Flexibilität bei Funktionsauswahl und Layoutgestaltung,
- RTC, Peripheral Clock und Event System Signale auf Pins routbar,
- 16-bit-Timer/Counter in je zwei 8-bit-Timer/Counter teilbar, dadurch mehr PWM-Kanäle, etwa für die Helligkeitssteuerung von LEDs.

Die Bezeichnungen der Bausteine wurden um ein „U“ ergänzt, aus dem alten ATxmega128A1-AU wird also der neue ATxmega128A1U-AU. AppNotes mit Tipps zum Umstieg findet man auf der Hersteller-Webseite. Das neue AVR Studio 5 stellt USB-Software-Bibliotheken und Treiber zur Verfügung (für Maus, Keyboard, Massenspeicher und mehr).

(Andreas Riedenaier, Ineltek Mitte GmbH, www.ineltek.com)

www.atmel.com/xmega

Anzeige



**PCBs
Muuuch Cheaper...**

16,94 EURO*

5 pcs, 100 mm x 100 mm
*per piece, incl. VAT (21%)
+ shipping costs e. g. Germany 10,71 EURO

No-frills policy



www.jackaltac.com

eC-reflow-mate

Neuer professioneller SMT-Reflow-Ofen für perfekt gelötete Platinen

Nach dem Erfolg des im Oktober 2008 vorgestellten SMT-Reflow-Ofens, präsentieren wir jetzt ein in Zusammenarbeit mit EuroCircuits entstandenes und komplett neu entwickeltes Nachfolgemodell. Mit höherer Präzision, mehr Platz für die Platinen der Möglichkeit der Steuerung mit dem PC. Der neue Ofen erfüllt in jeder Hinsicht die Anforderungen, die der tägliche Einsatz im professionellen Elektronik-Labor stellt.

Wichtige Eigenschaften

Betriebsspannung: 230 V/50 Hz	Temperaturbereich: 25...300 °C (300 °C Spitzenwert und 260 °C beim Löten)
Leistung: 3500 W	Effektive Platinengröße: max. 350 · 250 mm
Gewicht: ca. 29 kg	Temperatursensoren: 2 feste interne Sensoren, 1 externer Sensor (im Lieferumfang)
Abmessungen: 620 · 245 · 520 mm (B · H · T)	Besonderheiten:
Heizung: Infrarot-Strahler kombiniert mit Umluft	<ul style="list-style-type: none"> • Infrarot-Strahler für homogene Temperaturverteilung • Schublade fährt nach Lötprozess-Ende selbsttätig aus • Fenster in der Gerätefront zur ständigen Sichtkontrolle
Bedienung: Funktionstaster und LC-Display am Ofen oder über PC-Programm und USB-Verbindung	
Menüsprachen: Deutsch, Englisch, Französisch, Italienisch, Niederländisch, Ungarisch	

Der **eC-reflow-mate** ist die perfekte Lösung, wenn es um das Löten von Platinen für Prototypen und Kleinserien mit SMD-Komponenten geht. Das neue Modell 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.

Die fünf voreingestellten Temperatur-Verlaufsprofile lassen sich schnell an individuelle Erfordernisse anpassen. Das zum Lieferumfang gehörende PC-Programm stellt den Temperaturverlauf auf dem PC-Bildschirm grafisch dar. Mit diesem Programm können auch vorhandene Profile modifiziert und neue Profile entworfen werden.

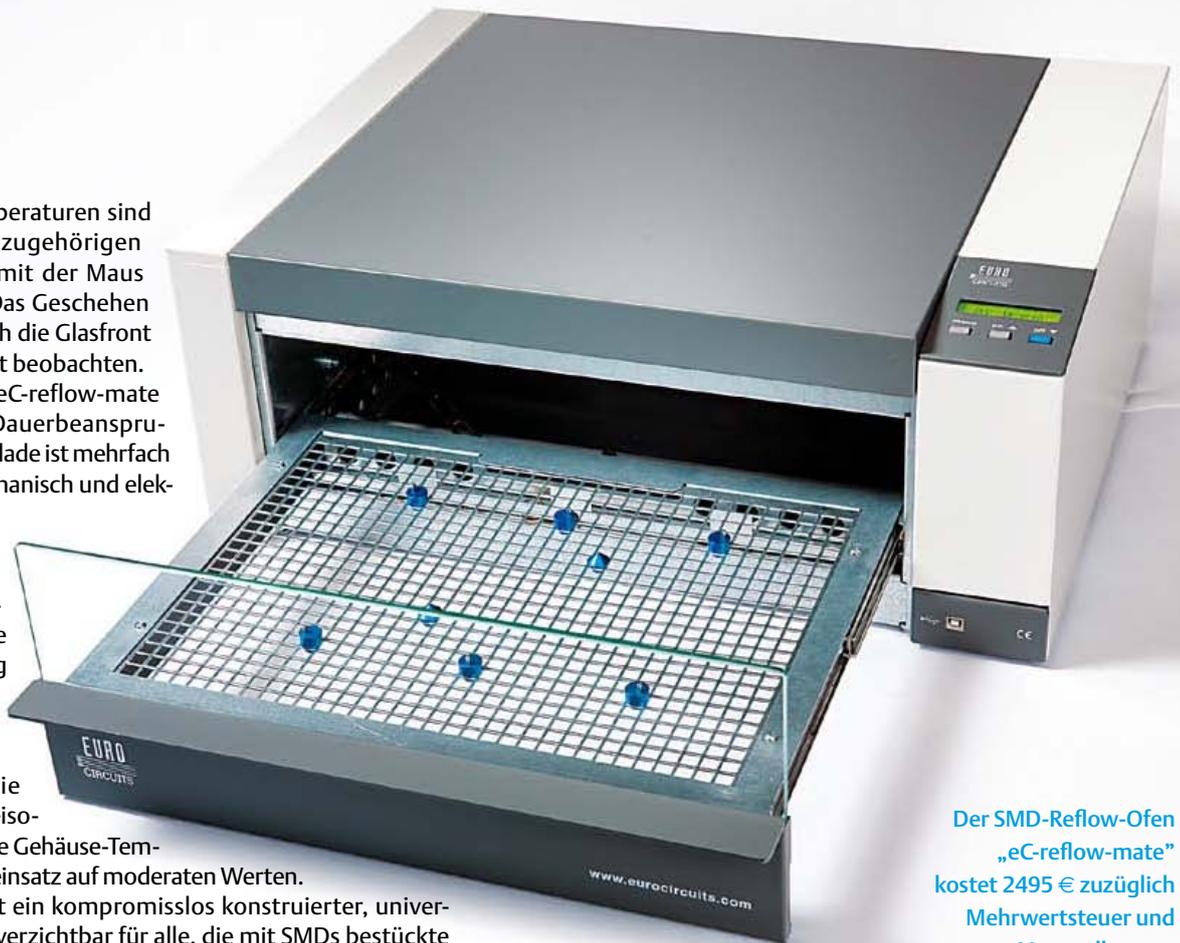
PC-Steuerung

Das zum SMT-Reflow-Lötofen „eC-reflow-mate“ gehörende PC-Programm heißt „eC-reflow-pilot“, es läuft unter den Windows-Versionen XP, Vista und 7. Mit diesem Programm ist über eine USB-Verbindung die uneingeschränkte und lückenlose Fernsteuerung des Ofens möglich. Auf dem Bildschirm werden parallel das gewählte oder eingestellte Temperaturprofil sowie der Temperaturverlauf im Ofen dargestellt. Das Programm aktualisiert kontinuierlich die dargestellte Temperaturverlaufskurve und die Anzeige der Sensor-Temperaturen.



Die Temperaturprofile sind modifizierbar, indem Kurvenpunkte mit der Maus in Richtung der Temperaturachse (vertikal) oder Zeitachse (horizontal) verschoben werden. Durch Mausklick lassen sich geänderte oder neue Temperaturprofile speichern, gespeicherte Profile werden auf gleiche Weise geöffnet. Es ist auch möglich, den Ofen über die Programmoberfläche ein- oder auszuschalten. Für „eC-reflow-pilot“ sind zurzeit sechs Programmsprachen verfügbar, weitere Sprachen sind in Vorbereitung.

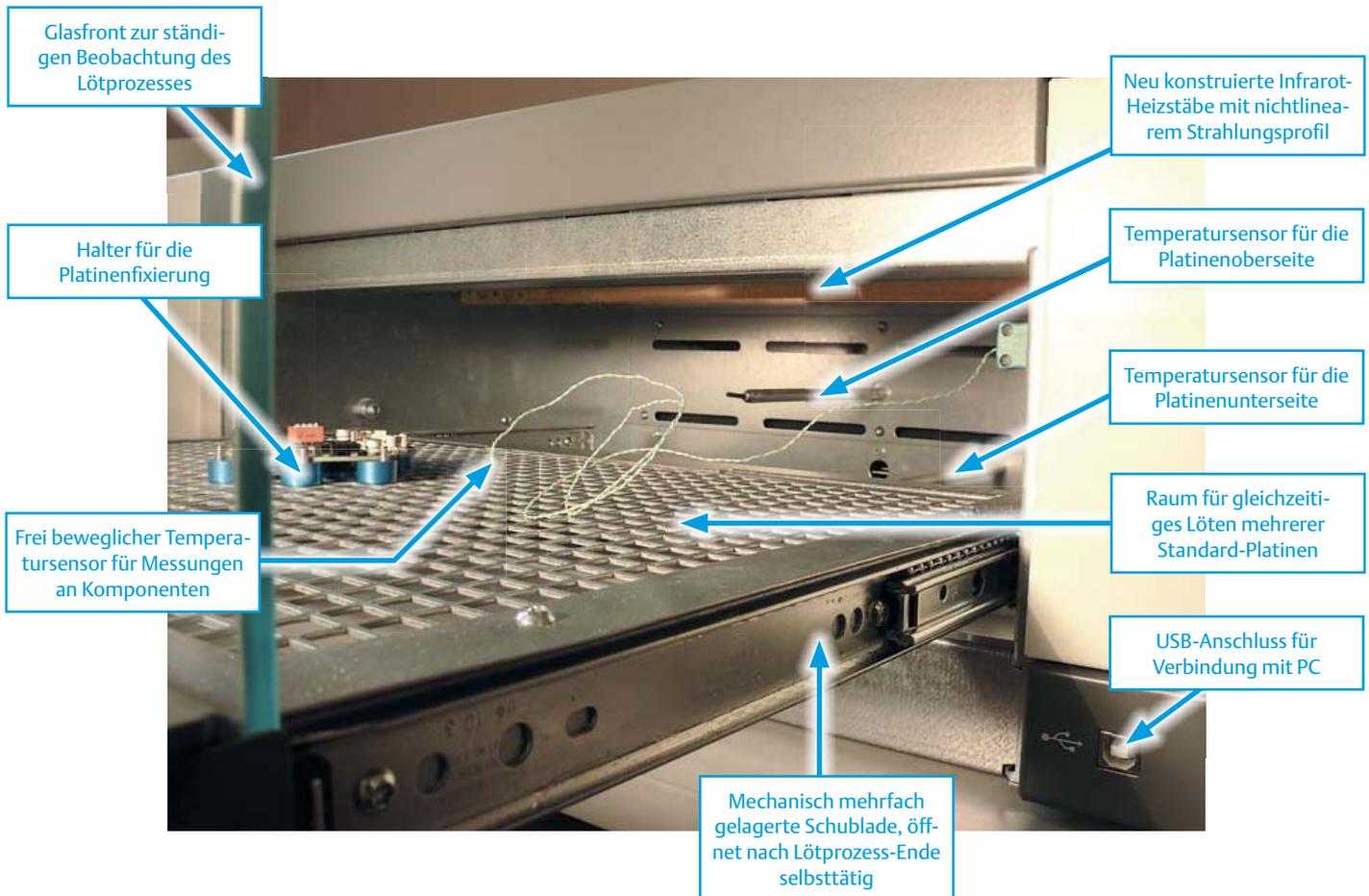
Schaltzeiten und Temperaturen sind änderbar, indem die zugehörigen Punkte in der Grafik mit der Maus verschoben werden. Das Geschehen im Ofen lässt sich durch die Glasfront der Schublade jederzeit beobachten. Die Konstruktion des eC-reflow-mate hält auch intensiver Dauerbeanspruchung stand. Die Schublade ist mehrfach gelagert, sie kann mechanisch und elektrisch bewegt werden. Wenn das Ende eines Lötprozesses erreicht ist, fährt die Schublade selbsttätig aus. Die interne Luftumwälzung sorgt für eine homogene Temperaturverteilung im gesamten Innenraum. Durch die hochwirksame Wärmeisolierung bleibt die äußere Gehäuse-Temperatur auch im Dauereinsatz auf moderaten Werten. Der eC-reflow-mate ist ein kompromisslos konstruierter, universeller SMT-Lötofen, unverzichtbar für alle, die mit SMDs bestückte Platinen in Einzelanfertigungen oder Kleinserien entwickeln und produzieren.



Der SMD-Reflow-Ofen „eC-reflow-mate“ kostet 2495 € zuzüglich Mehrwertsteuer und Versandkosten.

Auf www.elektor.de/reflow-mate finden Sie alle Informationen zur Bestellung und Lieferung.

(100447)gd

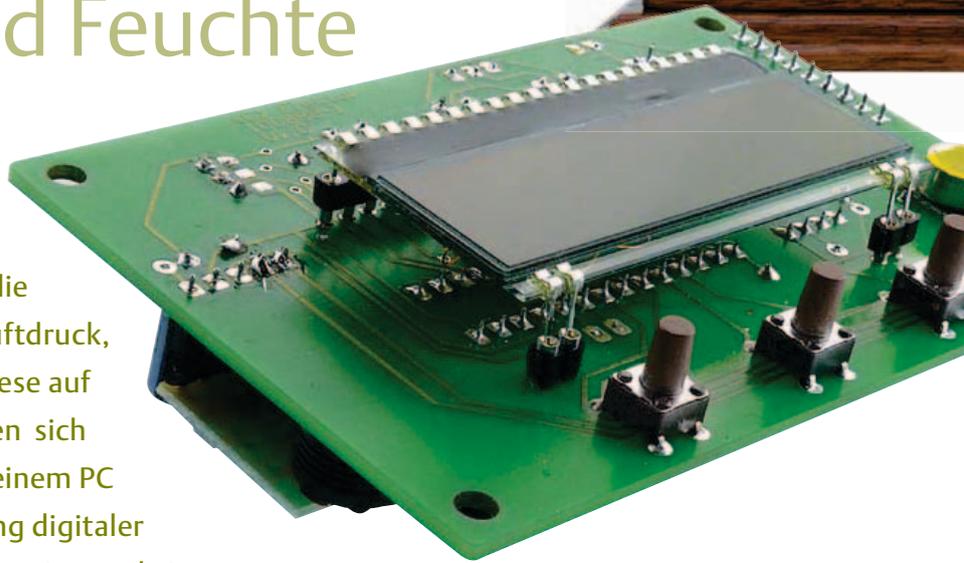


USB-Wetterlogger mit Langzeitspeicher

Mit I²C-Sensoren für Luftdruck, Temperatur und Feuchte

Von Wilfried Wätzig (D)
wr-waetzig@t-online.de

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



Der Autor hat diesen Datenlogger als Messsystem zur Langzeit-Aufzeichnung von Luftdruck, Temperatur und Luftfeuchtigkeit ausgelegt. Diese Werte werden in regelmäßigen Zeitabständen erfasst und in einem seriellen EEPROM gespeichert, so dass sie auch ohne Betriebsspannung erhalten bleiben. Über ein Seriell-USB-Modul erfolgt das Auslesen in den auswertenden PC. Die an anderer Stelle in dieser Ausgabe [1] ausführlicher beschriebenen Messmodule (Druck- und Temperatur-Modul und Luftfeuchtigkeits-Modul) mit I²C-Schnittstelle werden bereits kalibriert geliefert und vereinfachen den Aufbau und die Auswertung beträchtlich. Dank des eingebauten LC-Displays und der Batterie- oder Akku-Stromversorgung

lässt sich der Logger völlig autonom einstellen und betreiben. Die stromsparende Einstellung des verwendeten ATmega88 reduziert die mittlere Stromaufnahme auf unter 2 mA.

Die in C programmierte und mit AVR-GCC compilierte Firmware steht mit Sourcecode kostenlos zur Verfügung, so dass man die Software bei Bedarf auch an andere I²C-Sensoren anpassen kann.

Hardware

Durch den Einsatz von Funktionsmodulen wird die Schaltung in **Bild 1** einfach und übersichtlich. Die zentrale Steuerung erfolgt mit dem ATmega88 (IC2) [2], der mit den verschiedenen Modulen verbunden ist.

Zwei davon sind wie üblich an Portleitungen angeschlossen:

- Die Anzeige DOGM162 (LCD1) [3] mit 2 Zeilen zu je 16 Zeichen (ohne Hintergrundbeleuchtung). Das LCD-Modul wird im 4-Bit-Datenmodus über den PORTB betrieben und wurde gewählt, weil es sich auch mit 3,3 Volt betreiben lässt.
- Mod1 ist ein neues Seriell-nach-USB-Konvertermodul (BOB-FT232R) [4], das auch an anderer Stelle in dieser Ausgabe beschrieben wird.

Die anderen drei Module (eigentlich zwei Module und ein IC) sind mit dem Mikro-

Elektor Produkte & Service

- Platine: #100888-1
- Programmierter Controller: #100888-41
- Seriell/USB-Modul BOB-FT232R-V2.20: #110553-91

- Luftfeuchtesensor HH10D: #100888-71
 - Luftdrucksensor HP03S: #100888-72
 - Projekt-Software: #100888-11 (Gratis-Download)
- Erhältlich unter www.elektor.de/100888



Eigenschaften

- Genaue Messung von Luftdruck, Temperatur und Feuchte ohne Abgleich
- Speicherung von bis zu sechs Messungen (Luftdruck, Temperatur und Luftfeuchtigkeit) pro Stunde
- Nichtflüchtiger Speicher für 8191 Mess-Datensätze
- Bis zu acht Wochen Betriebsdauer mit drei Mignonzellen
- Kalibrierte Messmodule mit Digitalausgang für Luftdruck, Temperatur und Luftfeuchtigkeit
- Auslesen der Messdaten über USB-Schnittstelle



controller über einen I²C-Bus verbunden, der von Atmel aber als Two Wire Interface (TWI) bezeichnet wird. Dieser weit verbreitete Bus erlaubt die Kontrolle von bis zu 128 angeschlossenen Busteilneh-

mern („Slaves“) über zwei Leitungen. Der Atmega88 ist hier mit seinem integrierten TWI-Controller [5] der „Master“ über folgende I²C-Slaves:

- Das Druck-Messmodul HP03S von Hope enthält einen piezoresistiven Drucksensor mit integriertem 15-bit-A/D-Wandler (ADC), Steuerlogik und I²C-Interface. Der Sensor liefert eine vom Luftdruck und eine von der Temperatur abhängige Spannung. Diese Analogwerte werden

vom ADC nacheinander konvertiert und über die I²C-Schnittstelle ausgegeben. Auf dem Sensorchip werden vom Hersteller bei der Kalibrierung sieben sensorspezifische Koeffizienten von zwei Byte Länge und vier Sensor-Parameter mit einem Byte Länge in einem EEPROM gespeichert, die ebenfalls an den Mikrocontroller gesandt werden. Zur Taktung des ADCs wird ein 32-kHz-Signal mit 3 V Amplitude benötigt. Da der 32-kHz-Oszillator des Atmega88 im Sparbetrieb

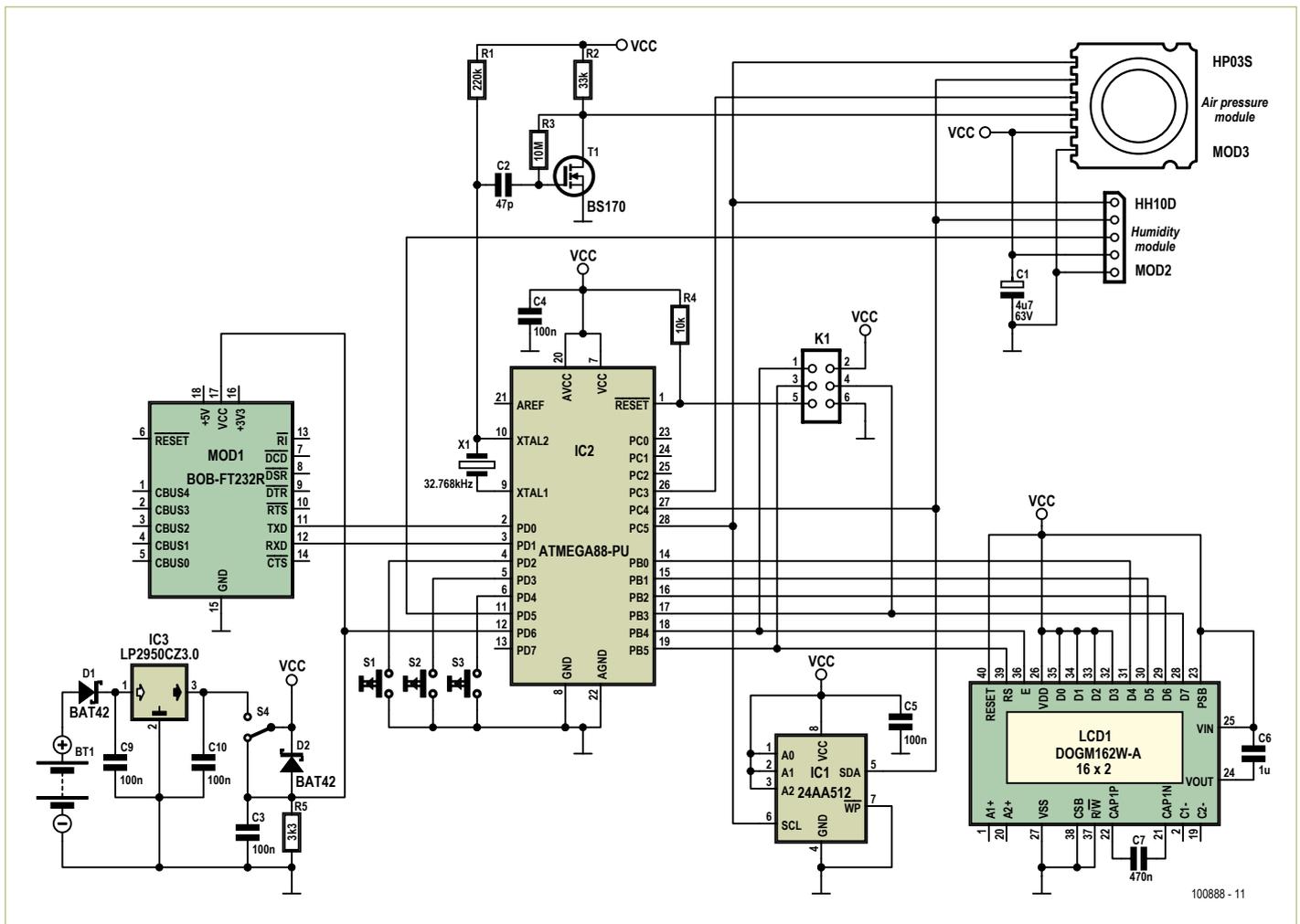


Bild 1. Das Schaltbild des Loggers: ATmega88 mit I²C-EEPROM und vier Modulen (USB, LCD, Druck/Temperatur und Feuchte).

Stückliste

Widerstände:

R1 = 220 k
R2 = 33 k
R3 = 10 M
R4 = 10 k
R5 = 1k3

Kondensatoren:

C1 = 4 μ 7/63 V stehend
C2 = 47 p
C3,C4,C5,C9,C10 = 100 n
C6 = 1 μ MKT 5 mm RM
C7 = 470 n

Halbleiter:

D1,D2 = BAT42
T1 = BS170
IC1 = 24AA512
IC2 = ATMEGA88-20PU (programmiert: EPS 100888-41)
IC3 = LP2950-3.3 oder -3.0

Außerdem:

S1,S2,S3 = 6-mm-Taster für Platinenmontage
S4 = 1-poliger Umschalter
X1 = 32,768-kHz-Quarz
LCD1 = DOGM162W-A (Electronic Assembly)
Mod1 = BOB-FT232R-V2.20 (EPS 110553-91)
Mod2 = Luftfeuchtesensor HH10D (Hope RF, EPS 100888-71)
Mod3 = Luftdrucksensor HP03S (Hope RF, EPS 100888-72)
K1 = 2x3-polige Stiftleiste (optional für ISP-Interface)
Buchsenleiste 1x20-polig für LCD1
Buchsenleisten 2x9-polig für MOD1
Buchsenleiste 1x5-polig für MOD2
IC-Sockel für IC1 (8-polig) und IC2 (28-polig)
Platine EPS 100888-1

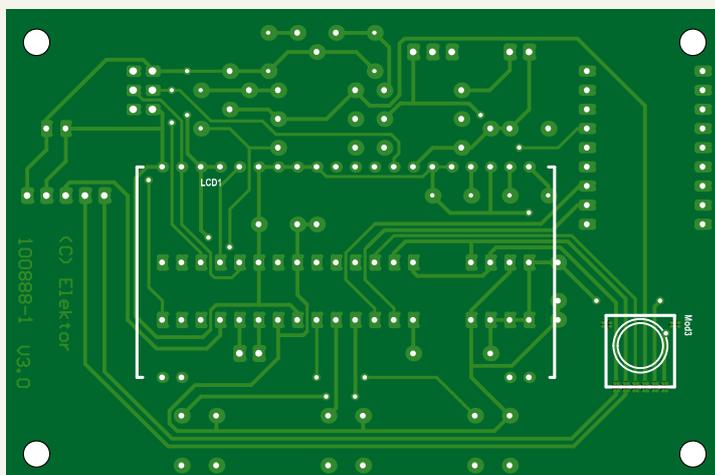
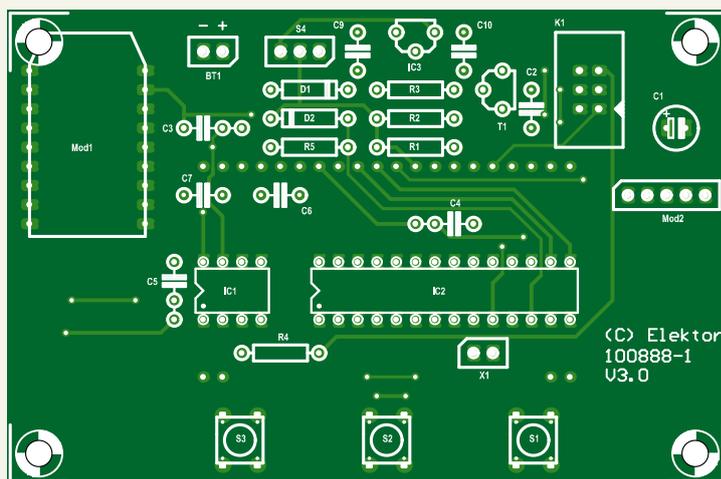


Bild 2. Die doppelseitige Platine ist einfach zu bestücken.

nur mit einer Amplitude von ca. 0,5 V läuft, muss das Signal mit einem BS170 (T1) verstärkt werden.

- Das Luftfeuchte-Messmodul HH10D stammt vom gleichen Hersteller (Hope RF), der Elektor-Lesern wahrscheinlich schon durch seine Funkmodule [6] bekannt ist. Ein feuchtigkeitsempfindlicher Kondensator als Sensorelement bestimmt die Frequenz eines Oszillators mit dem Timer-IC ICM7555. Die gemessene Frequenz im Bereich von 6...7 kHz wird über zwei Kalibrierungs-Parameter, die in einem seriellen I2C-EEPROM auf dem Modul gespeichert sind, vom ATmega88 in die relative Luftfeuchtigkeit umgerechnet. Zur Frequenzmessung werden die Ausgangs-Impulse des Moduls mit dem 16-bit Zähler TIMER1 im Atmega88 für eine Sekunde lang gezählt.
- Das serielle I2C-EEPROM 24AA512P (IC1) [7] mit einer Kapazität von 64 KB speichert die gemessenen Daten (Stundenmessung, Uhrzeit, Luftfeuchtigkeit, Temperatur und Luftdruck). Es kann maximal 8191 Mess-Datensätze aufnehmen.

Für die Bedienung sind drei Taster (S1...S3) vorgesehen. Mit S4 lässt sich die Betriebsspannung des Geräts vom Batterie/Akku-Betrieb auf USB-Stromversorgung umschalten. K1 erlaubt den Anschluss eines AVR-ISP-kompatiblen Programmers.

Software

Die Firmware für den Mikrocontroller ist in „C“ geschrieben und mit AVR-GCC (WinAVR 20080610) 4.3.0 übersetzt. Die Funktionen der einzelnen Module werden in eigenen Unterprogrammen ausgeführt.

Das Hauptprogramm `weather_station.c` ruft bei Bedarf die Unterprogramme `lcd_driver.c` (Ansteuerung des Displays), `usart_driver.c` (serielle Schnittstelle mit USART) und `TWI_driver.c` (I2C-Steuerung) auf, welche die gerätespezifischen Aufgaben ausführen.

Der TIMER2 wird durch den 32,768-kHz-Quarz angesteuert und erzeugt jede Sekunde einen Interrupt. In der Interrupt-Routine `TIMER2_COMPA_vect` wird die Uhrzeit (Stunde-Minute-Sekunde) erhöht

und die Event-Variable `flag1sec` gesetzt. Bei Beginn einer neuen Minute wird geprüft, ob x/n Stunde (z.B. $x/5$ Stunde entspricht alle 12 Minuten) erreicht wurde. Wenn ja, wird die Event-Variable `flagmstor` gesetzt. In der Endlos-Schleife im Hauptprogramm werden diese Variablen abgefragt. Ist `flag1sec` gesetzt, so wird in der 59. Sekunde vom `SLEEP_MODE_POWER_SAVE` in den `SLEEP_MODE_IDLE` gewechselt (Wieder-Einschalten der `IO_CLK`), sodass mit `TIMER1` die Frequenz für den Feuchtigkeits-Sensor gemessen werden kann.

Ist die Variable `flagmstor` gesetzt, dann werden aus dem Luftdruck-Modul die Daten ausgelesen, die Korrektur-Rechnungen ausgeführt und die Ergebniswerte im `I2C_EEPROM` abgespeichert. Danach wartet die CPU wieder im `SLEEP_MODE_POWER_SAVE` auf einen erneuten Interrupt von `TIMER2`.

Aufbau

Die doppelseitige Platine in **Bild 2** ist – der Schaltung entsprechend – sehr übersichtlich ausgefallen. Bis auf den Drucksensor sind alle Bauteile mit Anschlussdrähten versehen und normal zu bestücken. Am besten beginnt man mit dem Drucksensor, der als noch gut von Hand lötbare SMD-Bauteil auf der Platinenunterseite aufgelötet wird. Ebenfalls auf der Platinenunterseite bestückt werden die drei Taster `S1` bis `S3` und die Buchsenleisten für die Montage des LCD-Moduls. Für das Aufstecken des Luftfeuchtmoduls und des Seriell-USB-Moduls sind auf der Bestückungsseite der Platine Buchsenleisten einzulöten. Für die beiden ICs verwendet man am besten Fasungen. Die erste bestückte Elektor-Musterplatine ist in **Bild 3** und **Bild 4** zu sehen (Bild 2 ist die endgültige Version).

Die Einstellung der Fuses für die Programmierung des Controllers über den ISP-Anschluss (`K1`) ist im Textkasten angegeben. Das Programmieren kann man sich ersparen und die Stiftleisten für `K1` weglassen, wenn man den bereits programmierten Controller verwendet, der ebenso wie die Sensormodule und das Seriell-USB-Modul im Elektor-Shop erhältlich ist. Nach sorgfältiger Kontrolle der Bestückung kann man den Logger zum ersten Mal ein-

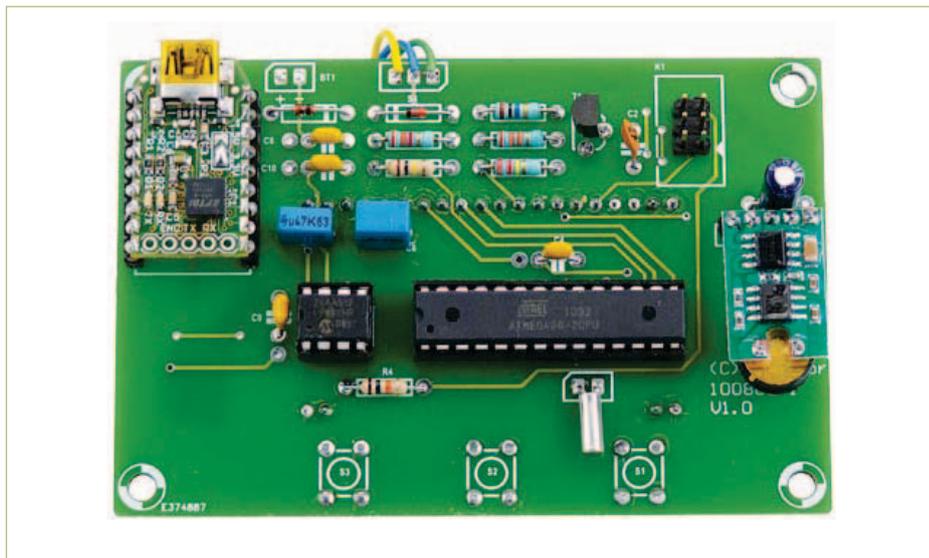


Bild 3. Die Bestückungsseite der Elektor-Musterplatine.

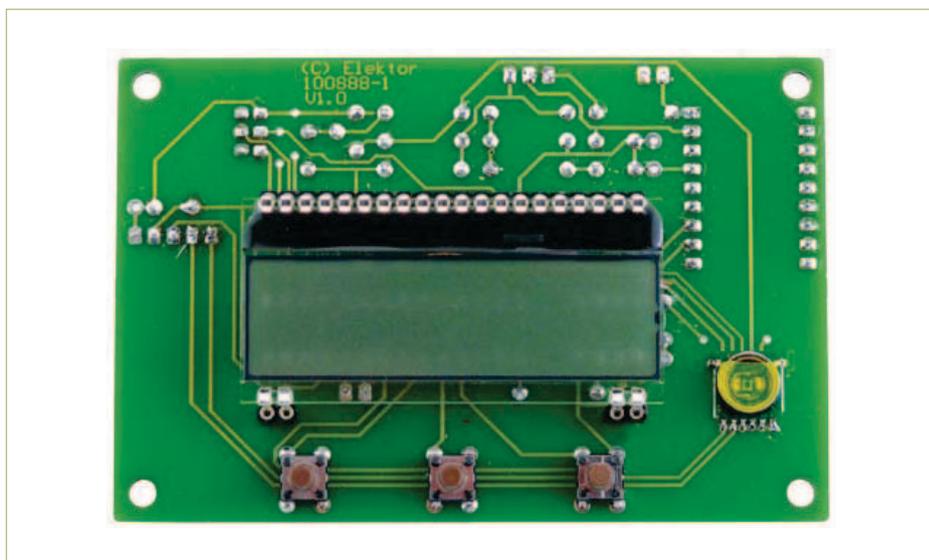


Bild 4. Das LCD-Modul und der Drucksensor befinden sich auf der Platinenunterseite.

schalten - mit `S4` in der richtigen Stellung – je nachdem, ob die Versorgung über USB oder extern (über eine Spannung an `BT1`) erfolgt. Abzugleichen gibt es dabei nichts.

Bedienung

Über die drei Taster (`S1`, `S2`, `S3`) lassen sich die Uhrzeit und weitere Parameter einstellen. **Tabelle 1** gibt eine Übersicht über die

Tabelle 1. Einstellungen und Funktionen (Taster `S1...S3`)

Funktionsauswahl mit <code>S1</code>	Funktion <code>S2</code>	Funktion <code>S3</code>
0: normale Anzeige	zeigt Messwert Druck	zeigt Messwert Feuchtigkeit
1: Stellen der Uhr	erhöht Stunden	erhöht Minuten
2: Setzen M/N	erhöht M: 0..6	löscht N
3: UART-Kontrolle	weiter	Exit
4: Anzeige Messwerte	weiter	Exit

M = Messungen pro Stunde; N = Anzahl der Messungen

Tabelle 2. Interaktive Kommandos zur Datenabfrage	
Befehl	Funktion
h =help	druckt die verfügbaren Kommandos, z.B. # h=help/a=show-p/p#=print#/m#=set-fm#/c=clear/x=exit
a =show-p	druckt die Anzahl der Messungen
p# =print#	p0 druckt die Messwerte mit Bezeichnung, z.B. 123 12:30:00 T= 25.6 degC H=43% P= 987.6 hPa p1 druckt nur die Zahlen, z.B. 123 12:30:00 256 43 9876 p2 druckt die Zeitachse in Stunden und die anderen Werte ohne Bezeichnung im Eingabeformat für GNUplot, z.B. 68.50 256 43 9876
m# =set-fm#	setzt die Anzahl der Messungen pro Stunde #=0..6 m0 keine Messung m6 Messung 6 mal pro Stunde, d.h. alle 10 Minuten
c =clear	löscht die Anzahl der Messungen N
x =exit	beendet die serielle Verbindung

Tabelle 3. Stromverbrauch des Gesamtsystems @ 3,3 V		
CPU-Frequenz	SLEEP-MODE	Stromverbrauch
8 MHz	keiner	5,8 mA
1 MHz	keiner	2,4 mA
1 MHz	IDLE	2,0 mA
1 MHz	POWER_SAVE	1,5 mA

Tabelle 3. Einstellung der Fuses für den ATmega88			
Fuses:	EXT.	0xF9	8 Mhz internal oscillator : 8 => CPU-CLOCK = 1 MHz
	HIGH	0xDF	CKDIV8 enabled, brown-out disabled
	LOW	0x62	65 ms startup

Funktionen der Taster.
Mit dem Taster S1 wählt man die Funktionen 0 bis 4 aus. Mit jedem Tastendruck von S1 wird um eine Position (= Zeile in der Tabelle) weitergeschaltet. In der gleichen Zeile der Tabelle ist die jeweilige Funktion der Taster S2 und S3 angegeben.
Für die Übertragung der Daten zu einem

PC oder Laptop ist ein USB-Kabel anzuschließen. Die Treibereinstellung auf dem PC wird im Artikel über das Seriell-nach-USB-Konvertermodul (BOB-FT232R) an anderer Stelle in dieser Elektor-Ausgabe beschrieben.
Mit dem Port PD6 wird vom Mikrocontroller abgefragt, ob die 3,3 V vom BOB-FT232R

vorhanden sind.
Für die Datenabfrage mit dem PC über die virtuelle serielle Schnittstelle sind die in **Tabelle 2** angegebenen interaktiven Befehle (z.B. mit Hyperterminal oder Hterm) vorgesehen.

Stromversorgung

Der Wetterlogger arbeitet mit einer Betriebsspannung von nominal 3,3 V, die beim USB-Betrieb vom Seriell-nach-USB-Konvertermodul BOB-FT232R an PIN 17 (VCC) bereitgestellt werden, wenn auf der Platine des Konvertermoduls bei JP1 die richtige Lötbrücke gesetzt ist (siehe [4]). Beim autonomen Betrieb erfolgt die Versorgung durch eine an BT1 angeschlossene Spannung (3,5 bis 30 V) und einen LDO-Spannungsregler (IC3), wobei die Diode D1 als Verpolschutz dient.

Um eine möglichst lange autonome Betriebsdauer zu erzielen, sind Maßnahmen zum Stromsparen sehr wichtig. Der Mikroprozessor ATmega88 ist der größte Verbraucher im Gesamtsystem, deshalb konzentrieren sich diese Maßnahmen auf ihn. Bei 3,3 V wurde der in **Tabelle 3** angegebene Stromverbrauch des Gesamtsystems gemessen. Bei Verwendung von drei Alkali- oder NiMH-AA-Zellen (4,5 bzw. 3,6 V, Kapazität 2...3 Ah) kann der Wetterlogger demnach mindestens 1000 Stunden lang Daten aufnehmen. Anstelle der drei AA-Zellen kann man auch eine einzelne Lilon- oder LiPo-Zelle (nominal 3,6 bzw. 3,7 V) verwenden.
Als weitere Sparmaßnahme bietet es sich an, nur eine Messung pro Minute statt pro Sekunde durchzuführen. In dieser Zeit wer-

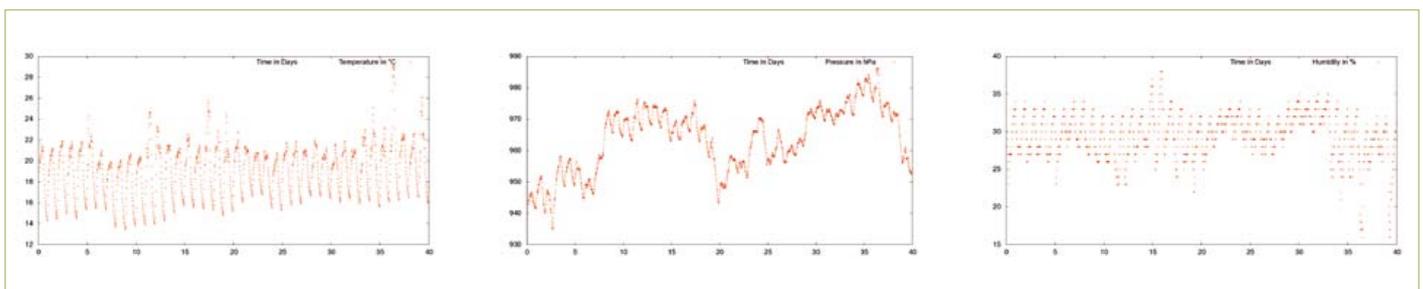


Bild 5. Der Verlauf von Temperatur, Luftfeuchte und Luftdruck lässt sich zum Beispiel mit dem Programm GNUplot grafisch darstellen.

den nur die Uhrzeit, die Anzahl der Messungen und die abgelaufenen Stunden auf dem Display angezeigt. Dieser Zustand wird erreicht, wenn 100 Sekunden lang kein Taster gedrückt wird.

Bei einer Messung nimmt das System für die Dauer von ca. 120 ms einen erhöhten Strom von ca. 2,5 mA auf, in den Pausen zwischen den Messungen hingegen nur für ca. 10 ms (nur mit dem Sekunden-Interrupt).

Bei jeder 59. Sekunde wird vom POWER_SAVE-Modus in den IDLE Modus umgeschaltet. Damit wird die IO-Clock des Mikrocontrollers (notwendig zum Zählen der Impulse des Feuchtigkeits-Sensors) wieder eingeschaltet, so dass in der 60. Sekunde der korrekte Wert ermittelt werden kann. Diese Stromsparmaßnahmen sind nur wirksam, wenn der ATmega88 mit 1 MHz CPU-CLOCK betrieben wird. Dafür sind bei der Programmierung des Mikrocontrollers die im Textkasten angegebenen Fuses zu setzen.

Messwertdarstellung

Mit dem kostenlosen Programm GNUplot lassen sich die ausgelesenen Messwerte auf dem PC grafisch darstellen. **Bild 5** zeigt Beispiele für die erzeugten Bilddateien, die den Verlauf von Temperatur, Luftfeuchte und Luftdruck wiedergeben.

Wie es zu solchen Grafiken kommt, zeigt das folgende Beispiel mit allen Details für die Übertragung der Daten und die Darstellung der Messwerte mit den Programmen Hterm/Hyperterm und GNUplot (die angegebenen Dateinamen beziehen sich auf die Beispiel-Messung):

1. Einstellungen in HTerm auf dem PC für das Auslesen der Daten über USB:

- COM3 9600Bd 8N1
- newline at: CR
- send on enter: CR
- storage Format: RAW

2. Wetterstation in den USB-Modus bringen:

- Wetterstation mit USB-Kabel an Rechner anschließen
- mit „Function“-Knopf S1 durchtasten bis Meldung „UART-control>USB“ erscheint,
- S2 drücken für „continue“; Meldung „USB-UART active“

Luftdruck und Messwerte

Der Logger misst und speichert den tatsächlichen momentanen Luftdruck, der nicht nur vom Wetter, sondern auch von der Ortshöhe und von weiteren Faktoren wie Temperatur und Luftfeuchte abhängt. Die von den Wetterdiensten angegebenen Luftdruckwerte sind aber auf Meeresspiegel (0 m Höhe) umgerechnete („reduzierte“) Werte. Bei der Umrechnung werden für Temperatur, Feuchte etc. die Werte der „Internationalen Standardatmosphäre“ zu Grunde gelegt. So ist beispielsweise die Temperatur mit 15 °C und der Standard-Luftdruck mit 1013,25 hPa definiert, jeweils für Meereshöhe. Die Druckwerte des Loggers kann man mit der „Internationalen Höhenformel“ wie folgt auf Meeresebene umrechnen:

$$p(h) = \frac{p(0)}{(1 - 22,558 \cdot 10^{-6} \cdot h)^{5,255}}$$

worin:

$p(h)$ = in der Höhe h gemessener Luftdruck in Hektopascal

h = Höhe in Meter

$p(0)$ = auf 0 m Höhe (Meeresspiegel) reduzierter Luftdruck

Es geht aber auch einfacher: Näherungsweise kann man in unteren Höhen bis etwa 2000 m mit einer linearen Druckabnahme von 1 hPa pro 8 m Höhe rechnen. Ein in 400 m Ortshöhe gemessener Luftdruck von zum Beispiel 970 hPa ergibt somit auf Meeresebene umgerechnet einen Luftdruck von ca. 1020 hPa.

3. In HTerm das Kommando „p2“ eingeben

- Es startet die Datenausgabe im Format: Stunden Temperatur*10 Feuchtigkeit Druck*10
- Beenden der Übertragung mit „x“
- Speichern der Daten als „pltdata0.txt“.

4. Plotten der Daten mit wGNUplot

- open file „plotfile0.plt“, dieser enthält die Kommandos für wGNUplot zur Erzeugung von drei Bild-Dateien:
- ptemp0.png (Temperatur-Verlauf)
- ppres0.png (Druck-Verlauf)
- phumid0.png (Feuchtigkeits-Verlauf)

Diese drei Bild-Dateien können dann je nach Bedarf mit einem Grafik- oder einem Text-Programm (WinWord, OpenOffice) weiterverarbeitet werden. Eine zusätzliche bebilderte Anleitung ist im Software-Download [8] enthalten.

(100888)

Weblinks:

- [1] www.elektor.de/110376
- [2] www.atmel.com/dyn/resources/prod_documents/doc2545.pdf
- [3] www.lcd-module.de/pdf/doma/dog-m.pdf
- [4] www.elektor.de/110553
- [5] www.atmel.com/dyn/resources/prod_documents/doc1981.pdf
- [6] www.elektor.de/071125
- [7] <http://ww1.microchip.com/downloads/en/devicedoc/21754h.pdf>
- [8] www.elektor.de/100888

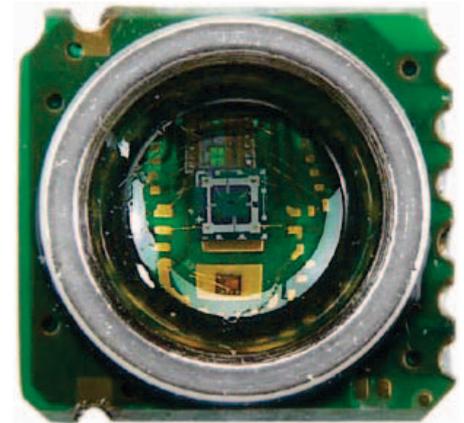
Der Autor

Während seines Physikstudiums kam Wilfried Wätzig mit den damals verfügbaren Rechenanlagen Zuse Z22 und Electrologica X1 und mit der Programmiersprache ALGOL in Kontakt. Später entwickelte er im universitären Umfeld Interface-Schaltungen und Datenerfassungsprogramme für Prozessrechner. Zuletzt arbeitete er als Systemadministrator in einem Hochschulrechenzentrum und verfolgt jetzt im Ruhestand mit Interesse die Hardware- und Software-Entwicklungen im EDV-Sektor.

I²C-Sensoren

Für Druck, Temperatur und Luftfeuchte

Die meisten Sensoren sind analog, und bevor man ihre Signale digitalisieren kann, muss man sie meist noch verstärken, kompensieren und kalibrieren. Bei den hier vorgestellten Sensormodulen ist das alles schon integriert, und mit den digitalisierten Messwerten erhält der angeschlossene Mikrocontroller auch Kalibrierungsdaten über den I²C-Bus geliefert, so dass kein Abgleich erforderlich ist.



Die in digitale Sensoren bereits integrierte Signalaufbereitung spart nicht nur Entwicklungsaufwand, sondern auch Platz und Kosten. Darüber hinaus wirkt sich die unmittelbare elektrische und thermische Kopplung zwischen Sensorelement und Signalverarbeitung auch positiv auf die Eigenschaften aus. Bei den hier vorgestellten beiden Sensormodulen von Hope RF [1] handelt es sich außerdem um kostengünstige Messmodule mit sehr geringem Strombedarf.

Luftdruck- und Temperaturmessmodul HP03S

Das Modul HP03S enthält einen piezoresistiven Drucksensor, der elektrisch gesehen aus vier Widerstandselementen in Brückenschaltung besteht. Wie das Blockschaltbild (Bild 1) zeigt, ist dieser klassische Drucksensor direkt mit einem Signalverarbeitungschip verbunden, das im Wesentlichen einen Eingangsumschalter, einen A/D-Wandler und ein I²C-Interface enthält. Teil des Interfaces ist ein 24C02-kompatibles EEPROM, das die vom Hersteller abgespeicherten Kalibrierungsdaten des Sensors enthält. Für die Druckmessung wird die Brückenspannung des Sensors mit dem ADC digitalisiert, während bei der Temperaturmessung die Temperaturabhängigkeit des Brückengesamtwiderstands ausgewertet wird, der dann Teil einer Messbrücke mit R1...R4 ist.

Als A/D-Wandler dient ein 16-bit-Delta-Sigma-ADC, wobei die tatsächliche Auflösung 14 bit beträgt. Weitere technische Daten sind im Kasten „Eigenschaften“ zusammengefasst. Die **Genauigkeit** der absoluten Druckmes-

sung wird im Datenblatt für den barometrischen Bereich (750-1100 hPa) wie folgt angegeben:

- ±1,5 hPa (HP03SA im Temperaturbereich 0...+50 °C)

Eigenschaften Druck/Temperaturmodul HP03S

- Druck-Messbereich: 300...1100 hPa
- Temperatur-Messbereich: -40...+85 °C
- Betriebsspannung: 2,2...3,6 V (typ. 3 V)
- Stromaufnahme: 1 µA (Standby), 500 µA (bei Messung)
- Betriebstemperatur: -40 bis + 85 °C
- I²C Schnittstelle: - SCL maximal 500 kHz
- MCLK 30 bis 35 kHz (typ. 32,768 kHz)

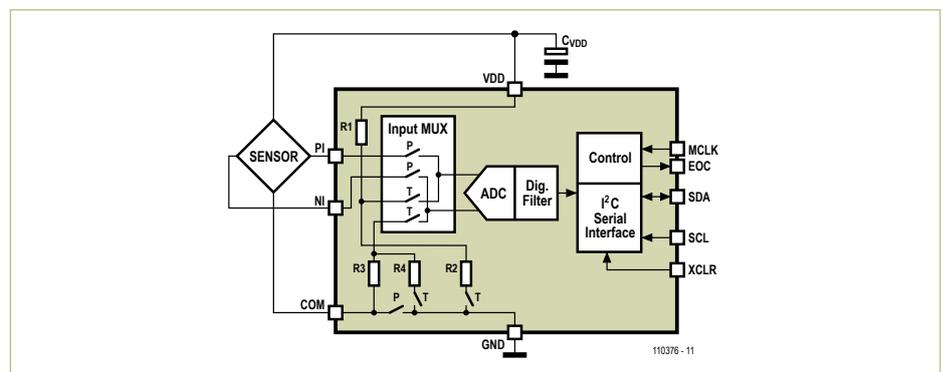


Bild 1. Blockschaltbild des Druck- und Temperatur-Sensormoduls HP03S.

Elektor Produkte & Service

- Luftfeuchtesensor HH10D: #100888-71
- Luftdrucksensor HP03SA: #100888-72
- Beispiel-Software: #100888-11 (Gratis-Download)
Erhältlich unter www.elektor.de/100888 und [2]

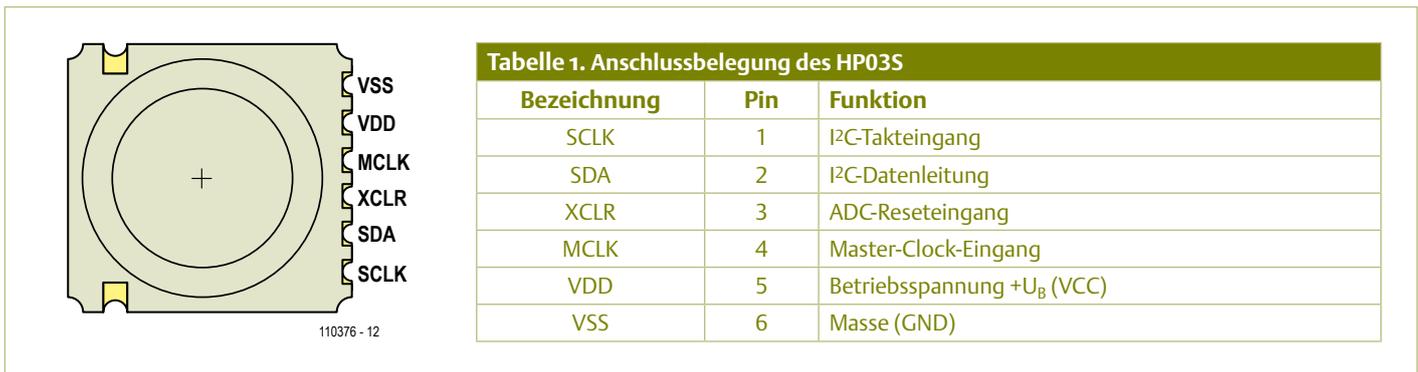


Bild 2. Anschlussbelegung des HP03S in der SMD-Ausführung.

- ±3,0 hPa (HP03SA im Temperaturbereich -20...+60 °C)
- ±3,0 hPa (HP03SB im Temperaturbereich 0...50 °C)
- ±5,0 hPa (HP03SB im Temperaturbereich -20...+60 °C)

Für beide Sensor-Versionen (A und B) gilt:

- Langzeitstabilität (12 Monate) typ. 2 hPa
- Spannungsabhängigkeit (2,4 bis 3,6 V) ±1,5 hPa.
- Genauigkeit der Temperaturmessung (0 bis +50 °C) ±1,0 °C
- Genauigkeit der Temperaturmessung (-20 bis +60 °C) ±2,0 °C

Die Anschlussbelegung ist in **Bild 2** zu sehen und korrespondiert mit der Signalbezeichnung in **Tabelle 1**.

Etwas Beachtung verdient dabei der Anschluss XCLR, der den ADC zurücksetzt. Der XCLR-Pin wird **nur** zur A/D-Wandlung und zum Auslesen von Druck- und Temperaturwerten auf High gesetzt. Ansonsten muss er immer Low sein - im Ruhezustand ebenso wie beim Auslesen des EEPROMs.

Die Qualität des 32-kHz-Taktsignals MCLK beeinflusst die Stromaufnahme des Druckmoduls. Neben Pegel (minimal 2,2 V) und Flankensteilheit ist auch ein Duty cycle im Bereich von 40-60 % (typisch 50 %) gefordert. Der ADC liefert über die I²C-Schnittstelle unkompenzierte Messwerte, und zwar den Messwert „D1“ für den Druck und den Messwert „D2“ für die Temperatur. Da der Drucksensor stark temperaturabhängig ist, muss der Druckmesswert temperaturkompensiert werden. Dafür sind in dem seriellen EEPROM sieben Koeffizienten und vier Sensor-Parameter gespeichert, die vom Hersteller in einem Kalibrierungslauf ermittelt wurden:

- C1 Sensitivity coefficient
- C2 Offset coefficient
- C3 Temperature coefficient of

- sensitivity
- C4 Temperature coefficient of offset
- C5 Reference Temperature
- C6 Temperature coefficient of Temperature
- C7 Offset fine tuning
- A, B, C, D Sensorparameter

Mit den vom ADC gelieferten Messwerten D1 für den Druck und D2 für die Temperatur werden die folgenden Rechenoperationen für die Temperatur T, den Offset OFF, die Sensitivity SENS und den endgültigen Druckwert P ausgeführt. Die Arithmetik ist an einen Mikroprozessor angepasst und enthält keine Fließkomma-Operationen:

$$dUT = D2 - C5$$

$$T = 250 + dUT * C6 / 2^{16}$$

*Temperatur in Grad Celsius * 10*

$$OFF = (C2 + (C4 - 2048) * dUT / 2^{14}) * 4$$

$$SENS = C1 + C3 * dUT / 2^{10}$$

$$X = SENS * (D1 - 7168) / 2^{14} - OFF$$

$$P = X * 10 / 2^5 + C7$$

*Druck in hPa * 10*

Ein praktisches Anwendungsbeispiel für die Verwendung des Sensors an der I²C-Schnittstelle eines Mikrocontrollers stellt das Projekt „USB-Wetterlogger mit Langzeitspeicher“ [3] in dieser Elektor-Ausgabe dar. Der Sourcecode der in „C“ programmierten Firmware steht unter [3] zum kostenlosen Download zur Verfügung.

Anhand des kommentierten Quellcodes kann man das Auslesen der Koeffizienten und Parameter aus dem EEPROM und der Messwerte des ADCs sowie die Berechnung der

kompenzierten Werte gut nachvollziehen. Alternativ gibt es auch beim Hersteller eine Programmieranleitung mit einem Anwendungsbeispiel [4], das für einen 8051-Mikrocontroller in „C“ programmiert ist. EEPROM und ADC haben unterschiedliche Chip-Adressen: 0xA1 für das Lesen des EEPROMs und 0xEE für den ADC. Für das Bus-Timing kann man sich am Datenblatt des 24C02-EEPROMs [5] orientieren. Beim Auslesen des ADCs darf man nicht vergessen, XCLR auf High zu setzen. Das Modul gibt alle Daten im Hex-Format aus. Nach dem Einschalten (Power-up) sollte der zuerst gelesene Wert ignoriert und erst der zweite Wert verwendet werden. Im Datenblatt [6] des HP03S findet man ein ausführliches Beispiel für die Druck- und Temperaturberechnung unter Berücksichtigung der Sensor-Koeffizienten und Parameter. Bei einer Verwendung als Höhenmesser wird für eine höhere Auflösung empfohlen, den Druckwert mit $P = X * 100 / 2^5 + C7 * 10$ zu berechnen. Im Datenblatt wird auch ausdrücklich darauf hingewiesen, dass dieser Sensor nicht für sicherheitsrelevante Anwendungen verwendet werden darf, insbesondere nicht für solche, bei denen ein Versagen des Sensors Leben und Gesundheit von Menschen gefährden könnte.

Feuchte-Sensor HH10D

Das Feuchte-Messmodul HH10D von Hope RF verwendet einen kapazitiven Feuchtigkeitssensor, der im Schaltplan (**Bild 3**) als variabler Kondensator die Frequenz eines Oszillators mit dem CMOS-Timer ICM7555 bestimmt. Das Ausgangssignal dieses „C/f-Konverters“ steht am Anschluss FOUT des Moduls (siehe Anschlussbelegung in **Bild 4**) zur Verfügung.

Das I²C-Interface dient beim HH10D ausschließlich dem Auslesen eines 24C02-

Eigenschaften Luftfeuchte-Messmodul HH10D

- Relative-Feuchte-Messbereich: 1...99 %
- Genauigkeit: ± 3 %
- Auflösung: 0,3 % bis 0,05 % (typ. 0,08 %)
- Reproduzierbarkeit: $\pm 0,3$ %
- Reaktionszeit: 8 Sekunden
- Hysterese: 1 %
- Langzeitstabilität: $\pm 0,5$ %
- Betriebsspannung: 2,7...3,3 V (typ. 3 V)
- Stromaufnahme: 120 bis 180 μA (typ. 150 μA)
- Betriebstemperatur: -10 bis + 60 °C
- Ausgangsfrequenz (F_{OUT}): 5 bis 10 kHz (typ. 6,5 kHz)
- I²C Schnittstelle: entspricht M24Co2BN

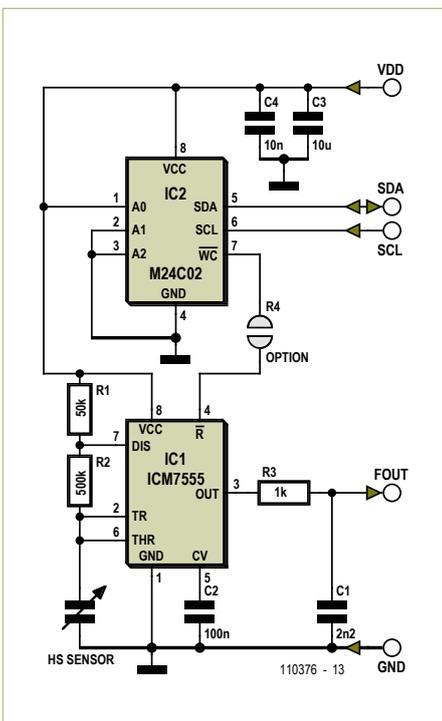


Bild 3. Schaltplan des Feuchte-Messmoduls HH10D.



Bild 4. Anschlussbelegung des HH10D, der als Aufsteckmodul mit Stiftleiste ausgeführt ist.

EEPROMs, in dem wiederum Kalibrierungswerte des Sensors gespeichert sind. Jeder Sensor wird vom Hersteller zweimal in zwei verschiedenen Klimakammern mit genau definierter Feuchte kalibriert. Der kapazitive Sensor hat den Vorteil, dass er sehr schnell auf eine Änderung der Luftfeuchte reagiert. Das Messmodul zeichnet sich außerdem durch eine hohe Auflösung und dank der gespeicherten Kalibrierungswerte auch durch eine hohe Genauigkeit bei sehr geringer Stromaufnahme aus. Die wichtigsten Eigenschaften

und Daten sind im Textkasten zum HH10D zusammengefasst.

Zur Auswertung muss ein mit SCL, SDA und FOUT verbundener Mikrocontroller zuerst die im seriellen EEPROM gespeicherten zwei Kalibrierungskoeffizienten (jeweils zwei Bytes) auslesen. Diese sind:

- SENS Sensitivity coefficient (EEPROM-Adresse 10)
- OFF Frequency Offset (EEPROM-Adresse 12)

Die Adresse des EEPROMs selbst ist auf 01 festgelegt.

Nun kann die relative Feuchtigkeit RH in % aus dem mit einem Timer/Counter ausgezählten Oszillatorsignal FREQ relativ einfach berechnet werden:

$$RH = (OFF - FREQ) * SENS / 2^{12}$$

Auch hier findet man in dem Projekt „USB-Wetterlogger mit Langzeitspeicher“ mit dem Sourcecode unter [3] ein praktisches Anwendungsbeispiel. Das Datenblatt des Feuchtesensors kann man unter [7] herunterladen.

Weitere I²C-Sensoren

Mittlerweile gibt es eine größere Auswahl an Sensoren mit I²C-Schnittstelle. Wie auch das Beispiel der zuvor beschriebenen Sensoren zeigt, sind die Ausführungen aber sehr unterschiedlich, und am gründlichen Studium des Datenblatts führt oft kein Weg vorbei.

Von Hope RF gibt es mit dem HDPM01 ein weiteres interessantes I²C-Sensormodul, das neben einem HM03-Drucksensor auch einen zweiachsigen Kompass-Sensor [8] enthält. Ein anderer I²C-Kompasssensor (CMPS03) wird in dem ATM18-Projekt [9] in dieser Ausgabe verwendet. Bei den Temperatur- und Feuchtesensoren mit I²C-Bus ist natürlich Sensirion [10] zu nennen. Die größte Auswahl findet man bei Temperatursensoren mit I²C-Schnittstelle, zum Beispiel von NXP [11] und von Maxim [12]. Zu den bekannten I²C-Temperatursensoren zählen auch der LM76 [13] von National Semiconductor, der DS1621/31 [14] von Dallas und der TMP100 von TI.

(110376)

Weblinks:

- [1] www.hoperf.com
- [2] www.elektor.de/110376
- [3] www.elektor.de/100888
- [4] http://www.hoperf.com/upload/sensor/HP03_code.pdf
- [5] ww1.microchip.com/downloads/en/DeviceDoc/21202J.pdf
- [6] www.hoperf.com/upload/sensor/HP03S.pdf
- [7] www.hoperf.com/upload/sensor/HH10D.pdf
- [8] www.hoperf.com/sensor/app/HDPM01.htm
- [9] www.elektor.de/110389
- [10] www.sensirion.com/en/01_humidity_sensors/00_humidity_sensors.htm
- [11] <http://ics.nxp.com/products/i2cthermal>
- [12] www.maxim-ic.com/products/thermal-management
- [13] www.elektor.de/990083
- [14] www.elektor.de/990072

BESSER GLEICH ONLINE KALKULIEREN.

STARRE- UND FLEXIBLE LEITERPLATTEN.



LEITON
RECHNEN SIE MIT BESTEM SERVICE

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

www.leiton.de

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

→ firma.embedded-projects.net

DAS HARDWARE FOR YOUR PROJECTS-PORTAL



In unserem Online-Shop finden Sie eine große Auswahl verschiedenster Mikrocontrollerboards, Programmer, Debugger u.v.m.

→ shop.embedded-projects.net

Unser Büro in Augsburg besteht aus leidenschaftlichen Entwicklern. Sprechen Sie uns an, wir finden eine Lösung für Ihr Problem.

→ projekte.embedded-projects.net



Speziell für Studenten und Hochschulen, bieten wir diese Ausbildungsinitiative an. Mikrocontrollerboards für den Kleinen Geldbeutel.

→ student.embedded-projects.net



Holzbachstraße 4, D-86152 Augsburg
Tel +49 (0) 821 279599-0
Fax +49 (0) 821 279599-20
info@embedded-projects.net



embedded projects GmbH
HARDWARE FOR PROJECTS

Wir sind Hersteller von physikalischen / elektronischen Meßgeräten und suchen einen erfahrenen

Elektroniker (m/w)

Aufgabengebiet: **Softwareentwicklung** für Mikrocontroller und PC (C++, AVR, etc.), **Schaltungsentwicklung (A/D)**, Unterstützung unseres Prüffeldes

Wir erwarten: Kreativität, Englischkenntnisse, Bereitschaft zu kurzen Auslandsreisen.

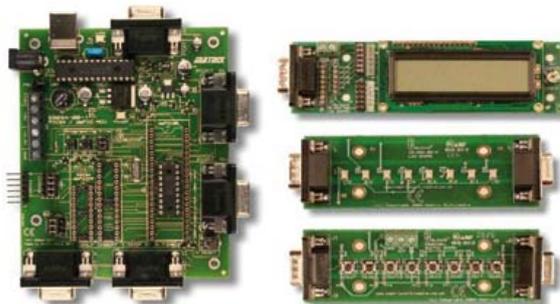
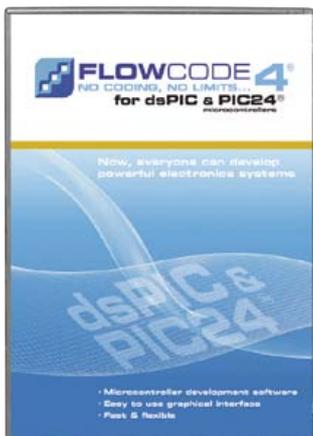
Interessante, abwechslungsreiche, ausbaufähige Tätigkeit, attraktives Gehalt.

NAGY Messsysteme GmbH, 71126 Gäufelden.

Bewerbung bitte per E-Mail an: bw22@nagy-instruments.de

dsPIC/PIC24-Bundle

Kostengünstige Hardware/Software-Lösung für die schnelle Projektentwicklung



Dieses neue Vorteilspaket ist ideal für alle, die Systeme mit dem leistungsfähigen 16-bit-Controller von Microchip entwickeln wollen. Das aus Hard- und Software bestehende Paket wird komplett mit einem dsPIC30F2011 geliefert und ist voll kompatibel mit der gesamten Palette an E-blocks-Boards und -Zubehör. Zu jedem einzelnen Bestandteil stehen Datenblätter separat zur Verfügung.

Lieferumfang:

- Flowcode 4 für dsPIC/PIC24 (Professional-Version)
- dsPIC/PIC24-Multiprogrammer mit ICD-Funktionalität (bei Verwendung mit Flowcode 4)
- LCD-Board
- LED-Board
- Switch-Board
- Steckernetzteil
- USB-Kabel

Bundle-Preis:
Nur 371,90 Euro

13% Ersparnis
gegenüber Einzelkauf!

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

USB-FT232R Breakout-Board

Von Antoine Authier (Elektor-Labor)

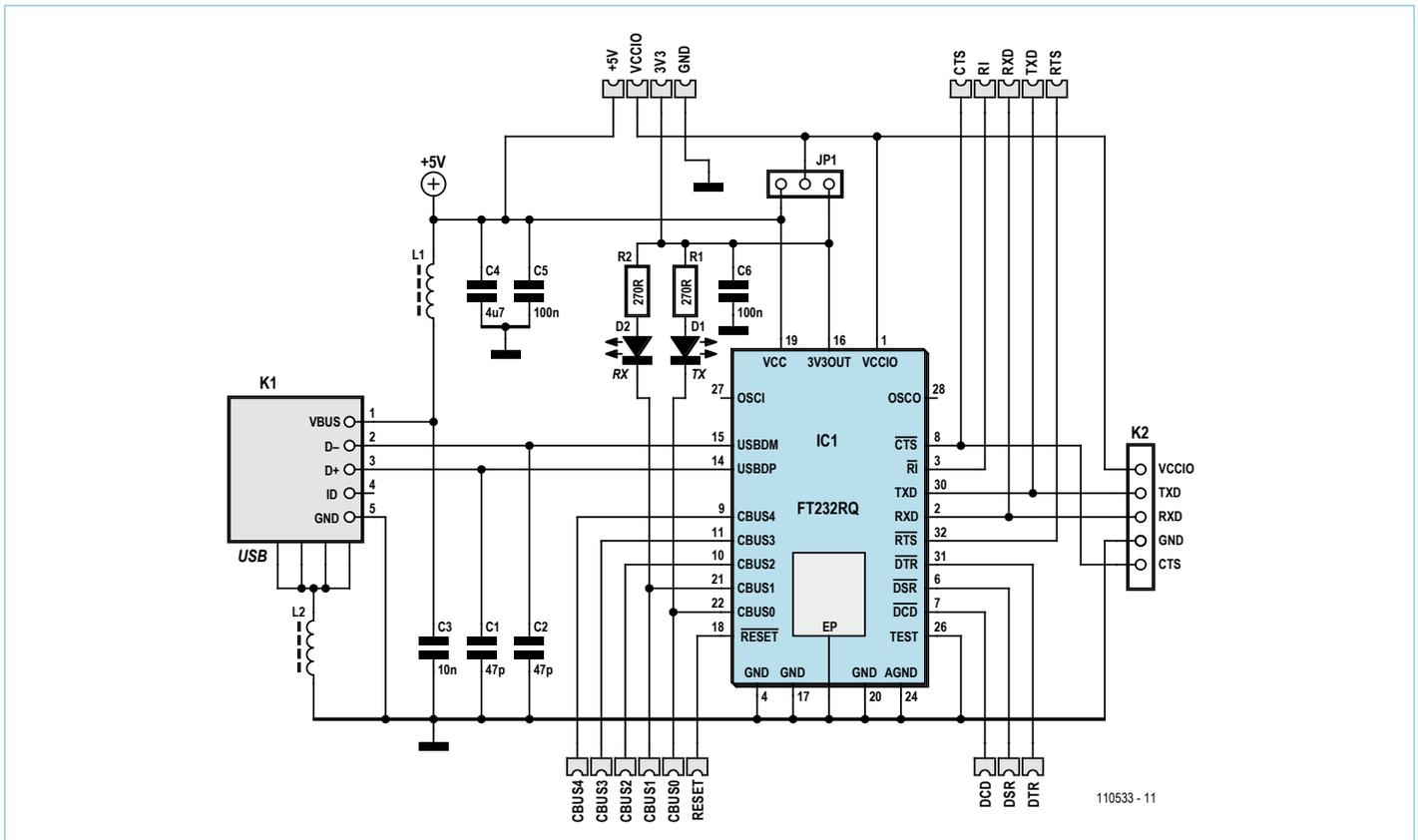
Klein ist er, dieser USB-nach-TTL-Wandler, nicht viel größer als der angegossene Stecker eines USB-Kabels! Seine nützlichen Dienste kann er unter Windows, Linux oder anderen Betriebssystemen entfalten. Aus wenigen Komponenten ist er auf einer Mini-Platine aufgebaut. Der Kleine ist überall dort praktisch, wo TTL erwünscht, aber nur USB vorhanden ist.

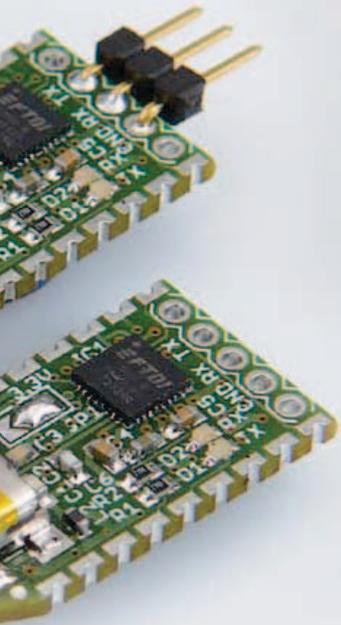
USB-nach-TTL-Wandler, die mit dem bekannten Chip FT232R von FTDI arbeiten, gibt es viele auf dem Markt. Ich suchte, doch ich fand keinen, der mich überzeugen konnte: Zu groß, zu teuer, oder einfach nur schlecht gemacht. Aus diesem Grund habe ich meinen eigenen Mini-Wandler als *Breakout-Board* (BOB) entworfen. „Breakout“ bedeutet so viel wie „Durchbruch durch

Grenzen“, und das trifft den Nagel auf den Kopf. Ein Breakout-Board verschafft Zugang zu Signalen eines komplexen Systems, die normalerweise nicht zugänglich sind. Im übertragenen Sinn werden äußere Barrieren des Systems durchbrochen.

In diesem Fall ist das komplexe System der schon fast als Industrie-Standard geltende

FT232R, der sich – in Kunststoff eingebettet und vergossen – in den USB-nach-TTL-Kabeln von FTDI verbirgt (siehe [1] und [2]). Meine Schaltung basiert auf den Datenblättern von FTDI [5], alle von FTDI vorgeschlagenen Applikationen wie beispielsweise RS232- und RS485-Wandler lassen sich mit dieser Schaltung realisieren. Doch damit sind die Grenzen noch nicht erreicht.





Ich denke an JTAG und BitBang, vielleicht komme ich darauf in einem späteren Beitrag zurück.

Werfen wir einen Blick auf die Schaltung: Die Ferritperlen L1 und L2 dämpfen Störungen, die der Spannung 5 V des USB-Bus überlagert sind. Die Kondensatoren C3 und C6 dienen der Entkopplung. Zwei LEDs signalisieren die Datenübertragung in die eine oder andere Richtung. Der Datenfluss wird aus Blickrichtung des an USB-Buchse K1 angeschlossenen Host betrachtet. Wenn Daten über K2 zum FT232R gelangen, werden diese Daten empfangen, LED RX leuchtet auf (*RX=Receive*). Verläuft der Datenfluss in Gegenrichtung, werden Daten gesendet, hierbei leuchtet LED TX auf (*TX=Transmit*). Sicherungen sind in dieser Schaltung nicht vorgesehen. Die einzige Option wäre eine so genannte *Polyfuse* (rücksetzbare Polymer-Sicherungen), doch leider stellen sie nur eine halbe Lösung dar. Die Erfahrung hat gezeigt, dass diese Sicherungen manchmal einen nicht geringen Spannungsabfall verursachen. Die 5-V-Spannung, die der USB-Port des Host liefert, kann über JP1 an die mit K2 verbundene externe Schaltung weiter gereicht werden. Falls nötig, ist diese Leitung gegen Überlast und Kurzschluss auf andere Weise zu sichern.

Die vom USB-Port kommende 5-V-Spannung ist auch an der Platinenaußenseite verfügbar. Sofern der USB-Port die Spezifikation 2.0 erfüllt, ist die Spannung bis 500 mA belastbar, bei USB 1.x beträgt der maximale Strom nur 150 mA. Der FT232R stellt seinerseits die Spannung 3,3 V bereit.

Eagle-Bibliothek

Mit dieser Bibliothek können Sie das USB-FT232R-Breakout-Board in Ihre Projekte einbauen. Gehen Sie wie folgt vor:



1. Installation

- Legen Sie einen Ordner mit dem Namen *library/Elektor* unter dem Ordner an, in dem Eagle installiert ist (wir nennen ihn nachfolgend *\$EAGLEDIR*).
- Entpacken Sie das heruntergeladene Archiv in diesen Ordner.
- In Eagle fügen Sie im Menü *options/directories* den Eintrag *:\$EAGLEDIR/library/ELEKTOR* hinzu. Statt des Doppelpunkts (:) am Anfang können Sie falls nötig ein Semikolon (;) ausprobieren.

2. Freie Wahl

Für den schnellen Start stehen acht Gehäuse und sechs Symbole zur Wahl:

- BOB-FT232R-MIN - Minimalistische Version mit nur drei Anschlüssen: GND, TX, RX
- BOB-FT232R-TINY - Kleinste Version mit I/O-Betriebsspannung: GND, TX, RX, VCCIO
- BOB-FT232R-CABLE - Funktional identisch mit USB-TTL-Kabel von FTDI, sechs Anschlüsse
- BOB-FT232R-WIDE - Vollständiges Pinout des Breakout-Board, sieben Anschlüsse
- BOB-FT232R-EDGE - Kontakteleisten an den Platinenseiten, für Piggyback-Montage
- BOB-FT232R-FULL - Vollversion

3. Alles startklar

In Eagle klicken Sie das Symbol Ihrer Applikation an. Wählen Sie falls gewünscht ein Gehäusesymbol aus und klicken Sie auf *Add*, um das Symbol Ihrem Projekt hinzuzufügen. Das Gehäuse erscheint im Konstruktionsfenster von Eagle, es muss nur noch eingefügt werden.

Software

1. Microsoft Windows

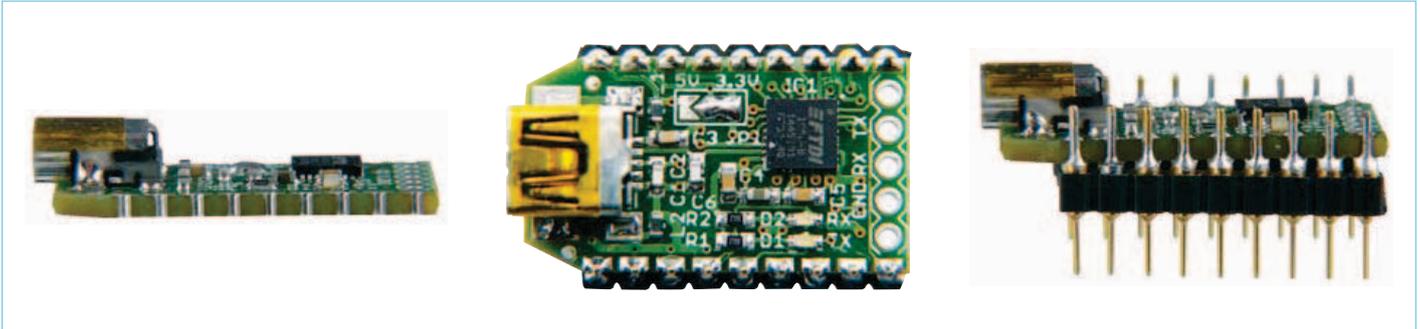
Ein von FTDI bereitgestellter Treiber bindet das USB-FT232R-Breakout-Board als gewöhnlichen COM-Port in das Betriebssystem ein. Der Treiber, der von www.ftdichip.com/Drivers/VCP.htm heruntergeladen werden kann, ist nur für die Einbindung als COM-Port notwendig. Wenn Sie *TeraTerm* oder *HTerm* als Terminal-Emulation einsetzen (empfohlen!), haben Sie über die D2XX-Treiber unmittelbaren Zugang zu den Funktionen des FT232R.

2. Linux (Kernel Version 2.6.31 und höher)

In den Linux-Distributionen mit Kernelversionen 2.6.31 und höher ist der neueste Treiber für die FT232R-Port-Emulation bereits implementiert (Modul *ftdi_sio*). Möglicherweise trifft dies auch auf ältere Kernel-Versionen zu. Das USB-FT232R-Breakout-Board ist über *peripheral/dev/ttyUSBx* zugänglich. Für Linux werden die Terminal-Emulationen *GTKTerm* oder *HTerm* empfohlen.

3. MAC OS X

Wie unter Windows können Sie auch auf Ihrem Mac einen COM-Port-Emulator installieren, er wird von FTDI ebenfalls auf www.ftdichip.com/Drivers/VCP.htm bereit gestellt. Nach der Installation ist das USB-FT232R-Breakout-Board unter */dev/tty.usbserialx* in das Betriebssystem eingebunden. Mit dem Programm *Screen*, das zu Mac OS X gehört, können Sie ein Terminal emulieren.



Hier ist unbedingt zu beachten, dass höchstens 50 mA entnommen werden dürfen. Anderenfalls kann der FT232R irreparabel beschädigt werden.

„Jumper“ JP1 besteht aus drei benachbarten Lötunkten. Wenn an K2 die Betriebsspannung 5 V oder 3,3 V liegen soll, müssen zwei Lötunkte durch einen Tropfen Lötzinn miteinander verbunden werden. Welche Lötunkte dies sind, ist auf der Platine vermerkt. Für die Spannung 5 V befinden sich die Lötunkte auf der Seite der USB-Buchse, für 3,3 V auf der Gegenseite. Wichtig ist, dass die Lötbrücke nur zwei Lötunkte verbinden darf. Wenn alle drei Lötunkte miteinander in Kontakt sind, hat dies ebenso wie das Verbinden der äußeren Lötunkte fatale Folgen. Die Lötbrücke muss gesetzt werden, *bevor* die Schaltung in Betrieb genommen wird.

Eine 3-polige Stiftleiste, Raster 2,54 mm, stellt auf der Gegenseite der USB-Buchse die Verbindungen nach Masse (GND), RX und TX her. Damit entsteht ein universeller und gleichzeitig unkomplizierter USB-nach-UART-Konverter, der für viele Zwecke einsatzfähig ist. Über Stiftleisten an den Seitenkanten ist der Zugang zu diversen internen Signalen sowie zur Betriebsspannung des FT232R möglich. Die Bezeichnungen der äquivalenten Signale eines FTDI-Kabels [1][2][3] gehen aus dem Aufdruck am Platinenrand hervor. Der Abstand zwischen den Kontaktreihen an den Platinenseiten beträgt 15,24 mm, er stimmt exakt mit dem 0,6-Zoll-Pinabstand einer (weiten) DIP-Fassung überein. Falls die Konstruktion besonders flach ausfallen soll, können an den Seitenkanten Stiftleisten mit abgewinkelten Kontakten montiert werden. Ferner bietet sich an, die Platine über die Kontaktstifte auf die Platine des umgebenden Systems aufzusetzen, so dass eine so genannte *Piggyback*-Konstruktion entsteht.

Auf der zu diesem Beitrag gehörenden Projektseite [4] finden Sie eine Eagle-Bibliothek, mit der Sie das USB-FT232R-Breakout-Board schnell und unkompliziert in Ihre

Applikation implementieren können. Dort haben wir auch ein Datenblatt mit detaillierter technischer Beschreibung eingestellt. Falls Sie Ihre Applikation debuggen müssen, dürfte dieses Dokument unverzichtbar sein. Auch der geübte und erfahrene Elektroniker, der nicht nur über gute Augen, sondern auch über die passende Werkzeugausrüstung verfügt (insbesondere über einen Heißluft-LötKolben!), wird das USB-FT232R-Breakout-Board nicht „blind“ und im Handumdrehen aufbauen können.

Wenn Sie in diesem Metier Newcomer sind oder es Ihnen einfach nur an Zeit fehlt, können Sie es bequemer haben: Der Elektor-Shop hat das fertig bestückte Board einschließlich diverser Kontaktleisten in sein Programm aufgenommen, es ist getestet und sofort einsatzbereit. Über die Modalitäten einer Bestellung gibt ebenfalls die Projektseite [4] Auskunft.

(110553)gd

Stückliste

Widerstände:

R1,R2 = 270 Ω (0603)

Kondensatoren:

C1 = 10 n (0603)

C2,C3 = 47 p (0603)

C4 = 4,7 μ/6,3 V (0603)

C5, C6 = 100 n (0603)

Induktivitäten:

L1,L2 = Ferritperle > 30 Ω @ 100 MHz / 1,5 A (0603)

Halbleiter:

IC1 = FT232RQ

D1 (TX) = LED grün (0603)

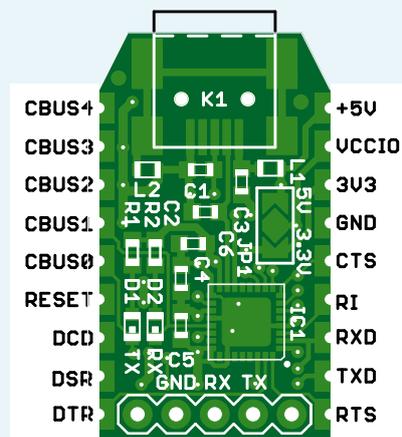
D2 (RX) = LED rot (0603)

Außerdem:

K1 = USB-Buchse Mini-B

Aufgebautes und getestetes Modul:

EPS 110553-91



200 % der wahren Größe

Weblinks

- [1] [USB-TTL Interface-Kabel](#), Juni 2008, und zugehörige Kabel in 5- und 3,3-V-Ausführung (Elektor-Shop 080213-71 und 080213-72): www.elektor.de/080213
- [2] [USB/RS-232 Adapter](#), Juli/August 2008 (Elektor-Shop 080470): www.elektor.de/080470
- [3] [Erweiterung für serielles USB-TTL Kabel](#), Juli/August 2010 (Elektor-Shop 100007): www.elektor.de/100007
- [4] [USB-FT232R Breakout-Board](#): www.elektor.de/110553
- [5] [Datenblatt des FT232R](#): www.ftdichip.com/Products/ICs/FT232R.htm

Elektor OSPV



**Wheelie reloaded:
Das Open-Source-Projekt auf 2 Rädern**

Mit unserem beliebten ElektorWheelie (Elektor 06/2009) sind inzwischen viele Menschen unterwegs, die Technik des balancierenden Fortbewegungsmittels fasziniert Jung und Alt. Hinter dem neuen Open Source Personal Vehicle (kurz „OSPV“) stehen die gleichen Ideen – Unterschiede gibt es jedoch in der Umsetzung. Der leichter gebaute, zusammenlegbare „OSPV“ wurde für den Indoor-Betrieb entwickelt, Konstruktion und Software sind wiederum Open Source. Auch mit diesem innovativen Einachser ist maximaler Fahrspaß garantiert!

elektor

NEU!



Technische Daten:

- 2 x 200-W-DC-Getriebemotoren
- 2 x 12-V-Bleigel-Akkus mit 9 Ah
- 2 x PU-Räder mit 14 cm Durchmesser
- Zahnriemenantrieb
- Höchstgeschwindigkeit: 15 km/h
- Reichweite: ca. 8 km
- Gewicht: 25 kg
- Maße (H x B x T): 120 x 47 x 47 cm
- Belastung: 90 kg
- Bodenfreiheit Trittlflächen: 2 cm
- Höhe Trittlflächen: 5,6 cm
- Breite zwischen Trittlflächen: 29,5 cm
- Ladezeit: 2,5 Stunden

Der „Elektor OSPV“-Komplettbausatz umfasst zwei DC-Motoren (je 200 W), zwei 12-V-Bleigel-AGM-Akkus, zwei PU-Räder mit Riemenscheibe und HTD-Zahnriemen, Gehäusesatz inkl. aller Schrauben, bestückte und getestete Controller-Platine mit aufgesteckter Sensor-Platine + Ladegerät.

Art-Nr.: 110320-91

Preis: 1095,00 € (inkl. MwSt., zzgl. Porto/Versand)

Weitere Infos und Bestellung unter
www.elektor.de/ospv

www.elektor.de

1, 2, Leiterplatte!

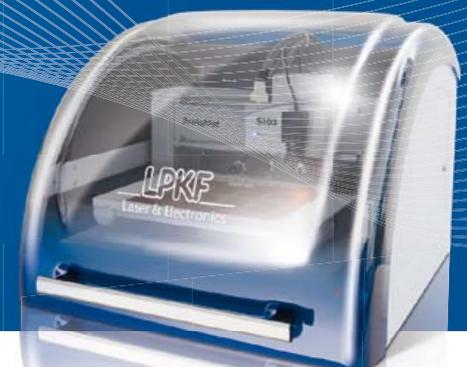


Von der Idee über das strukturierte Board zur fertigen Leiterplatte!

Inhouse-Leiterplatten-Prototyping:

Von der Idee bis zur bestückten Leiterplatte. Die Komplettlösung für Ihr Leiterplatten-Prototyping! www.lpkf.de/prototyping

LPKF Laser & Electronics AG Tel. +49 (0) 5131-7095-0



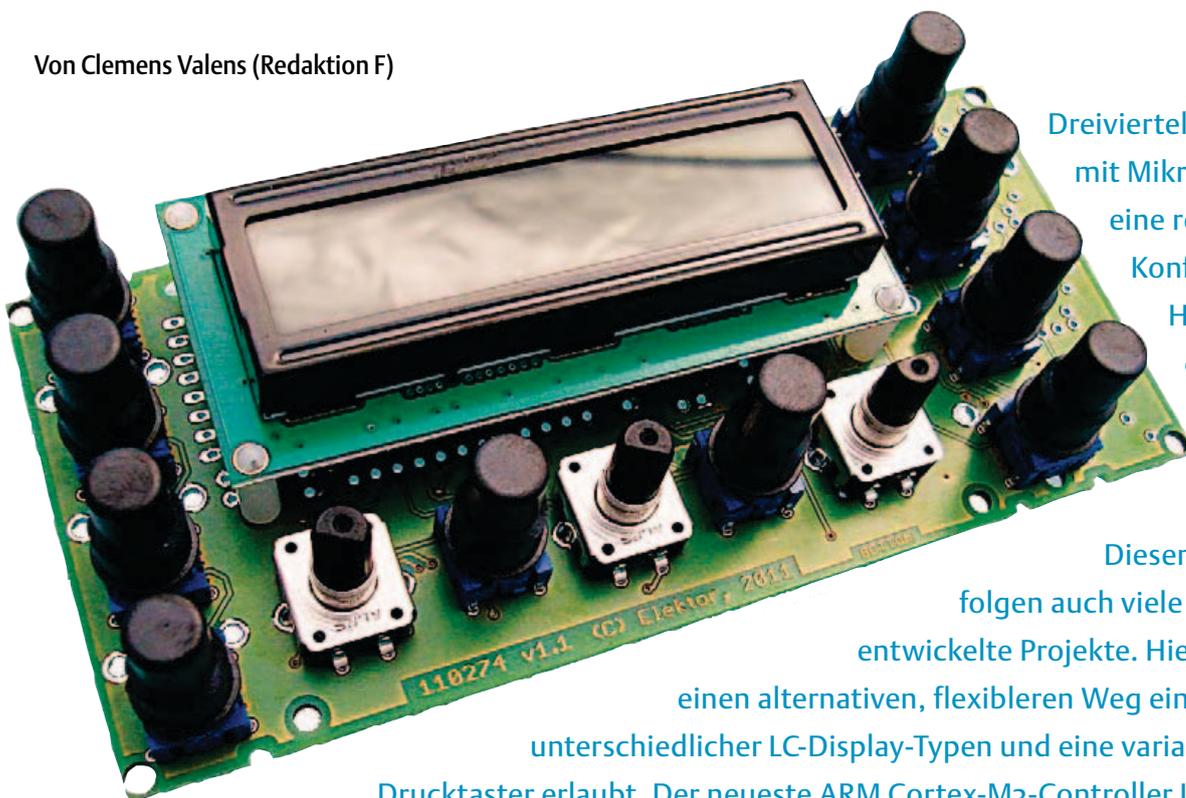
LPKF

Laser & Electronics

J²B: Vielseitiges HMI-Modul mit ARM Cortex-M₃

Das Rad müssen Sie nicht neu erfinden!

Von Clemens Valens (Redaktion F)



Dreiviertel aller Applikationen mit Mikrocontrollern haben eine recht ähnliche Konfiguration: Die Hauptzutaten sind ein Mikrocontroller, ein LC-Display sowie einige Drucktaster.

Diesem Schema folgen auch viele von Elektor entwickelte Projekte. Hier schlagen wir einen alternativen, flexibleren Weg ein, der den Einsatz unterschiedlicher LC-Display-Typen und eine variable Anzahl Drucktaster erlaubt. Der neueste ARM Cortex-M₃-Controller LPC1343 verhilft diesem Board zu hoher Leistungsstärke, der Umgang mit dem Board ist erstaunlich einfach.

Die LC-Displays vieler kleiner Mikrocontroller-Projekte stellen entweder 2 · 16 Zeichen oder 4 · 20 Zeichen dar. Nur selten sind mehr als vier Drucktaster vorhanden, zunehmend kommen auch Drehimpulsgeber zum Einsatz. Bei den kleinen Displays (2 · 16) sind die Taster in der Regel darunter angeordnet, während bei den großen Displays (4 · 20) die Montage auf der rechten oder linken Seite üblich ist. Der Ort der Bedienelemente hängt vom Projekt und auch vom Anwender ab. Rechtshänder bevorzugen die rechte Seite, während Linkshänder mit links vom Display angeordneten Tastern besser zurechtkommen. Flexible Lösungen müssen solchen Unterschieden Rechnung tragen und die freie Wahl der Gestaltung zulassen.

Die Idee, eine möglichst universell einsetzbare HMI zu entwickeln (HMI = *Human-Machine-Interface*, Mensch-Maschine-Schnittstelle), entstand bereits vor Jahren, als die I/O-Ports mehrerer entwickelter Systeme für die Steuerung einer HMI nicht ausreichten und eine Port-Erweiterung unumgänglich war. Inzwischen steht mit den 32-bit-Mikrocontrollern ARM Cortex-M0 und -M3 von NXP eine ebenso elegante wie kostengünstige Lösung zur Verfügung. Besonders anwenderfreundlich ist der Typ LPC1343, denn er lässt sich erstaunlich einfach programmieren. Ein Programmiergerät wird nicht benötigt, und auch ein USB-nach-RS232-Wandler ist entbehrlich. Dieser Mikrocontroller verhält sich wie ein USB-Stick, das Programm wird über den

USB-Port in den Mikrocontroller kopiert. Allerdings ist diese Methode nur unter Windows anwendbar, unter Linux oder Mac OS sind eine serielle Verbindung oder ein spezielles Programmiergerät erforderlich. Für den Datenaustausch mit der Applikation stehen mit diesem Mikrocontroller unter anderem die Schnittstellen USB, I²C, SPI sowie ein UART zur Verfügung. Da die Display-Steuerung und die Abfrage der Bedienelemente bei vielen Applikationen rund vier Fünftel des Programms ausmachen, liegt eine weitere Überlegung nahe: Der Mikrocontroller dieser HMI kann auch das Ausführen der Applikation übernehmen. Damit ist gleichzeitig die Leistungsstärke eines 32-bit-Mikrocontrollers nutzbar. Oder anders betrachtet wird nicht zur Applikation

eine Port-Erweiterung hinzugefügt, sondern die Port-Erweiterung wird durch die Applikation ergänzt.

Da der Mikrocontroller LPC1343 nur im SMD-Gehäuse gefertigt wird, haben sinnvollerweise auch die übrigen Bauelemente SMD-Bauform (bis auf Taster und Steckverbinder). Auf dem mit Steckverbindern ausgestatteten Board können weitere Funktionen untergebracht werden, zum Beispiel ein LiPo-Akku mit zugehörigem Lader. Mit einer eigenen Stromversorgung ist das Board auch für den mobilen Einsatz gerüstet.

Eigenschaften

- Steuerung für LC-Displays 2 · 16, 4 · 16 oder 4 · 20 Zeichen mit 14- oder 16-poliger Steckleiste, Hintergrundbeleuchtung von der Applikation schaltbar
- Tastatur-Matrix 5 · 6 für maximal 12 Taster, 9 Drehimpulsgeber mit eingebauten Tastern (entsprechend 27 Tastern!) oder gemischte Konfiguration
- Akustischer Signalgeber
- LED
- Stromversorgung über USB, extern 5 V, Batterie (0,9...4,5 V) oder LiPo-Akku
- 32-bit-Mikrocontroller LPC1343 mit 48 Anschlüssen, 32 KB Flash-Speicher, 8 KB RAM und viel Peripherie, zum Beispiel USB, I2C, SPI, PWM, UART sowie diverse Counter
- Kompatibel mit den kostenlosen Entwicklungsumgebungen LPCXpresso [2][3][4] und CooCox [7]
- Kompatibel mit den Programmern/Debuggern LPC-Link und CooCox
- Erweiterungsmöglichkeiten: Fast alle Mikrocontroller-Anschlüsse sind über Steckverbinder oder Löt-Pads zugänglich
- Teilbares Board: Nicht benötigte Teile können abgetrennt werden, absetzbares Mini-Tasterfeld mit maximal 4 Tastern oder 3 Drehimpulsgebern
- Abmessungen passend für Gehäuse Bopla 26160000
- Hardware und Software sind Open Source

Arbeitsweise

Mikrocontroller

Das zentrale Bauelement des Systems ist der Mikrocontroller LPC1343 von NXP. Für das

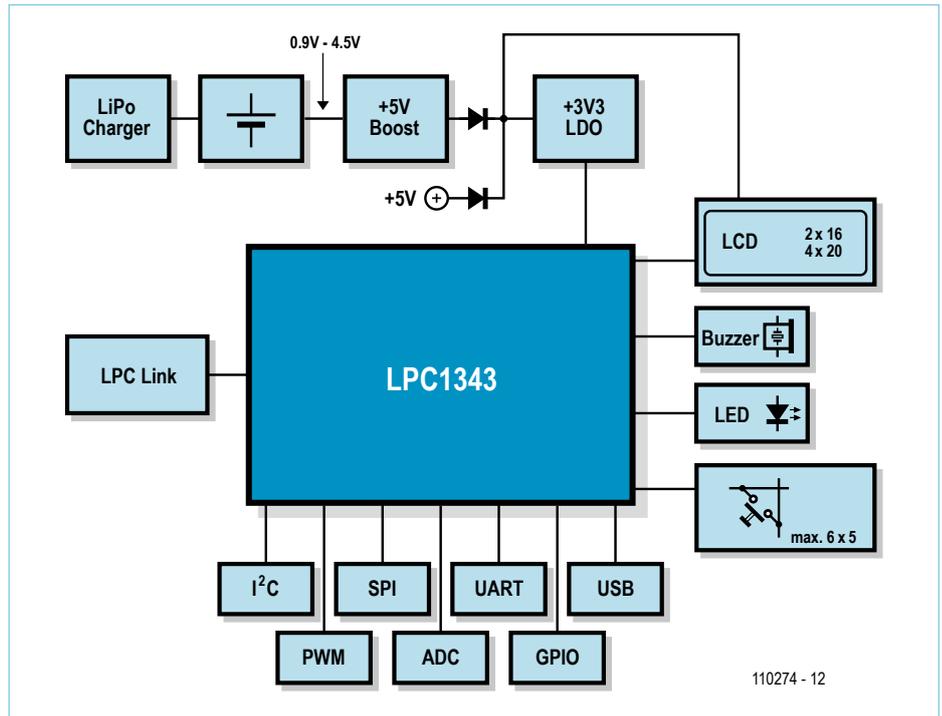


Bild 1. Blockschematische Darstellung des J2B-Boards.

Taktsignal stehen drei Optionen zur Wahl: Entweder wird das Taktsignal vom internen RC-Oszillator erzeugt (IRC), ein externer Oszillator liefert das Taktsignal oder das Taktsignal wird von einem fremden System bezogen. Ohne äußere Beschaltung arbeitet der Mikrocontroller mit dem internen Taktsignal. Für Applikationen, die ein hochpräzises Taktsignal voraussetzen, kann der Mikrocontroller mit einem Quarz beschaltet werden. Das Datenblatt des LP1343 empfiehlt für die Bürde-Kapazitäten C14 und C15 abhängig von den Quarz-Eigenschaften Werte zwischen 18 pF und 39 pF. Mit dem in der Schaltung angegebenen Wert 18 pF schwingen die meisten Quarze stabil.

Nach dem Reset hängt der Startmodus des Mikrocontrollers von den logischen Signalen an PIO0.1 und PIO0.3 ab. Wenn PIO0.1 niedrig ist, wird der Bootlader im ISP-Modus gestartet, anderenfalls startet die Applikation. Der LPC1343 hat zwei ISP-Modi: Programmierung als USB-Stick (PIO0.3 hoch) oder über den seriellen Port (PIO0.3 niedrig). Der ISP-Modus wird mit den Widerständen R3 und R13 eingestellt. Im Prinzip erfüllt bereits die USB-Spannungserkennung die Aufgabe von R3. Andererseits verhindert R3, dass ein undefiniertes Signal auftritt, falls der Anschluss des USB-Kabels fehlerhaft ist. Wenn der Mikrocontroller über den seriellen Port programmiert wer-



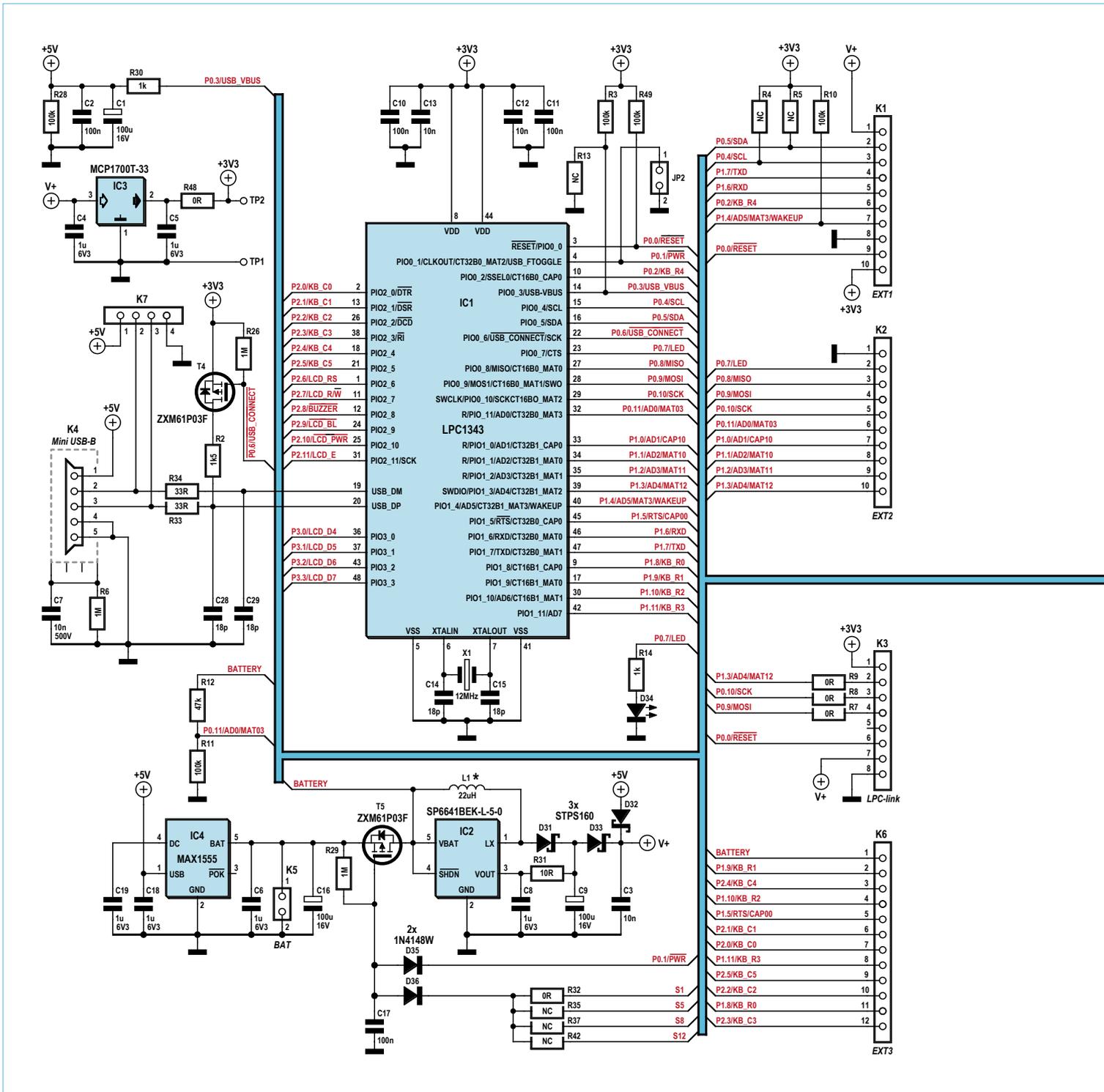


Bild 2. Schaltung des Boards. Stückliste, Eagle-Dateien und Software stehen auf [1] zum Download bereit.

den soll, wird R13 anstelle von R3 montiert. In den integrierten USB-Treiber hat sich leider ein Bug eingeschlichen, der zur Folge hat, dass bei der ersten Programmiersitzung etwa 30 Sekunden vergehen, bis Windows den Mikrocontroller als Stick erkennt. Danach verläuft die Erkennung normal, vorausgesetzt der Mikrocontroller bleibt unter

Spannung. Der Mikrocontroller kann auch über einen seriellen Debug-Port (*Serial Wire Debug, SWD*), ähnlich einem seriellen JTAG) programmiert werden. Zu diesem Zweck ist Steckverbinder K3 vorhanden, seine Kontaktbelegung ist kompatibel mit dem LPC-Link der frei erhältlichen Entwicklungsumgebung LPCXpresso. Die mit PIO0.7 verbun-

dene LED (D34) ist ebenfalls kompatibel, so dass das Testprogramm „LPCXpresso1343_blinky“, das Bestandteil der Entwicklungsumgebung ist, unverändert laufen kann.

Tasterfeld

Das Tasterfeld besteht aus 12 Tastern, jeweils vier Taster sind rechts und links vom

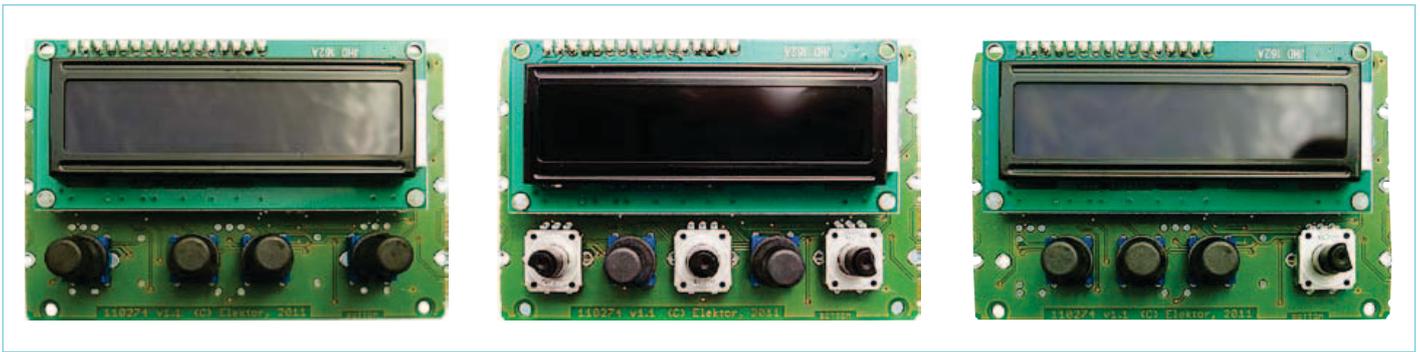


Bild 3. Einige Varianten mit der Display-Größe 2x16. Die Taster können beliebig angeordnet werden.

Displays beträgt üblicherweise 5 V. Wenn das System an einer externen 5-V-Spannungsquelle betrieben wird (beispielsweise am USB-Bus), ist der Anschluss unproblematisch. Für den Betrieb an einer mobilen Stromquelle (Batterie oder Akku) ist ein 5-V-Boost-Spannungsregler vorhanden. Über T3 wird die Betriebsspannung des Displays unterbrochen, so dass das Display rückgesetzt oder Energie gespart werden kann. Zur Energieeinsparung trägt auch T1 bei, der die Display-Hintergrundbeleuchtung schaltet. Strombegrenzungswiderstand R1 hat wegen der auftretenden Verlustleistung die SMD-Bauform 1206. Der Wert von R1 ist nicht kritisch, er bestimmt die Hintergrundhelligkeit.

Akustischer Signalgeber

Ein Standard-Beeper, Durchmesser 12 mm und Rastermaß 6,5 mm, kann gesteuert über T2 Töne oder Alarmsignale wiedergeben. Widerstand R36 verhindert, dass störende Geräusche auftreten, solange Leitung PIO2.8 während der Mikrocontroller-Programmierung als Eingang dient. Der Beeper kommt ohne Widerstand R41 aus, der Widerstand ist nur notwendig, wenn der Beeper durch eine LED oder ein Relais ersetzt wird. Sowohl R1 als auch R41 haben die SMD-Bauform 1206.

Stromversorgung

Die Hauptversorgungsspannung (V+) beträgt 5 V, sie wird nur für das Display benötigt, während der Mikrocontroller und die übrige Schaltung mit der Spannung 3,3 V arbeiten. Spannungsregler IC3 stellt diese Spannung aus der Spannung 5 V bereit. Mit Nullohm-Widerstand R48 lässt sich die 3,3-V-Leitung unterbrechen, so dass die Spannung ohne Gefahr für den Mikrocontroller getestet werden kann.

Die Spannung 5 V kann eine von drei Quellen liefern: Ein externes Netzteil, verbunden mit K7, der USB-Port über USB-Buchse K4 oder der Boost-Spannungsregler IC2. Der Boost-Spannungsregler setzt Eingangsspannungen im Bereich 0,9...4,5 V auf 5 V hinauf, so dass ein Akku, mehrere 1,5-V-Batterien oder auch nur eine 1,5-V-Batterie,

Bauelemente der Stromversorgung weggelassen werden. In diesem Fall sind nur die Komponenten D32, C3, C4, C5, IC3 und R48 zu montieren. Die Option, dass die Applikation das System ein- oder ausschaltet, ist dann nicht verfügbar.

USB und Erweiterungen

Am Schluss dieser Schaltungsbetrachtung stehen die Verbindungen nach außen: K4 ist die auf der Platine montierte Mini-USB-Buchse, während der Entwicklungsphase wird der Mikrocontroller im ISP-Modus über diese Verbindung als USB-Stick programmiert. Für die spätere Applikation befindet sich USB-Buchse K4 wahrscheinlich an falscher Stelle. Deshalb kann an Steckverbinder K7 eine zweite USB-Buchse angeschlossen werden, die verbunden über ein Kabel an beliebiger Stelle montiert werden kann. Der Mikrocontroller signalisiert dem PC über Transistor T4, dass ein USB-Gerät angeschlossen ist.

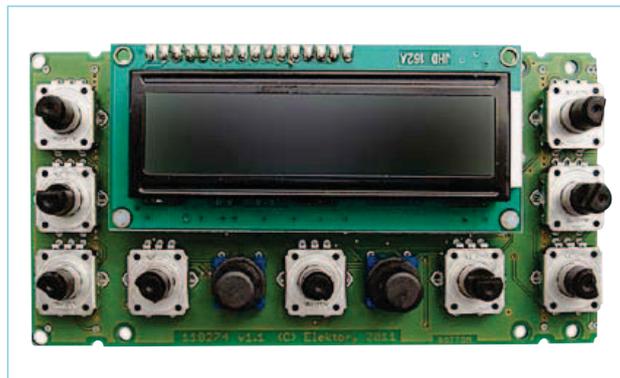


Bild 4. Mehr geht nicht: Neun Drehimpulsgeber und zwei Taster entsprechen 29 Schaltkontakten!

angeschlossen an K5, das System versorgen können. Der Mikrocontroller hat einen niedrigen Leistungsbedarf (60 mW bei 72 MHz Taktfrequenz), so dass eine einzelne 1,5-V-Monozelle für lange Zeit ausreicht. Da der Mikrocontroller an Betriebsspannungen im Bereich 2,0...3,6 V arbeitet, ist ein Spannungsregler 3,3 V nicht zwingend notwendig. Stattdessen sind auch eine 3-V-Lithium-Batterie oder ein 3-V-Lithium-Akku verwendbar.

Wenn ein LiPo-Akku (Spannung < 4,5 V) die Stromversorgung übernimmt, der beispielsweise aus einem Mobiltelefon stammt, wird der LiPo-Akku über Akkulader IC4 aus dem USB-Port (K4) oder dem externen Netzteil (K7) geladen.

Bei ausschließlicher Stromversorgung aus einem externen Netzteil können diverse

Über K1, K2 und K6 sind fast alle I/O-Leitungen des Mikrocontrollers zugänglich. Auch an K8, der die Verbindung zum Display herstellt, liegen diverse Mikrocontroller-Signale. Die Signale der I/O-Leitungen PIO0.6, PIO2.8, PIO2.9 und PIO2.10 werden von Transistoren gepuffert, in diesem Fall liegen die I/O-Leitungen an Löt pads.

Die Kommunikationsleitungen wie I²C, SPI und UART wurden an die Steckverbinder K1 und K2 gelegt. Dort sind auch die I/O-Leitungen zu finden, die vom LC-Display und von der Tastermatrix nicht benutzt werden. K1 ist mit dem USB/TTL-Wandlerkabel von FTDI kompatibel (Elektor-Shop 080213 [8]). I/O-Leitung PIO1.4 ist mit einem Pullup-Widerstand (R10) beschaltet, weil diese Leitung im „Deep Power-Down Modus“ des Mikrocontrollers eine spezielle Funktion



Bild 5. Viele Möglichkeiten, auch mit einem 4x20-Zeichen-Display.

hat. An die I²C-Leitungen können die Pullup-Widerstände R4 und R5 gelegt werden. Steckverbinder K6 fasst die meisten Signale zusammen, die das Display und das Tasterfeld steuern. Bestimmte Signale liegen gleichzeitig an K8 und K9, sie sind für den Anschluss eines abgesetzten und an anderer Stelle montierten Mini-Tasterfelds bestimmt.

Montage

SMDs, die wegen ihrer Abmessungen Lötprobleme bereiten könnten, sind nicht vorhanden. Die Montage des Mikrocontrollers erfordert etwas Geduld, hier lassen sich Kurzschlüsse durch überschüssiges Lötzinn mit Entlötlitze entfernen.

SMD-Induktivitäten sind nicht immer leicht beschaffbar. Deshalb wurde eine Kombination aus Typen von Coilcraft, von Coiltronics und Drahtanschluss-Typen Raster 3,5 mm gewählt. Coilcraft [9] bietet einen Musterexemplar-Service an, Coiltronics-Produkte sind unter anderem bei Farnell erhältlich. Die Induktivitäten anderer Hersteller können natürlich ebenfalls passend sein.

Der akustische Signalgeber, die LED, Steckverbinder K8 und die Taster haben ihren Platz auf der Lötseite der Platine. Die verwendeten Taster bestehen aus einem Körper, auf den Kappen in verschiedenen Formen, Farben und Größen aufgesetzt werden können. Die freie Auswahl vereinfacht das Anpassen an die Applikation oder auch an individuelle Vorlieben. Die Platine kann bei Bedarf entlang der montierten Taster und des Displays mit einer Säge durchtrennt werden. Ungeteilt passt die Platine perfekt in das Gehäuse 26160000 von Bopla (siehe Foto am Artikelanfang). Ein Display der Größe 2 · 16 Zeichen und vier darunter angeordnete Taster nehmen zusammen

mit der Platine weniger Platz ein als ein einzelnes Display mit 4 · 20 Zeichen. Weil sich das Tasterfeld abtrennen lässt, ist die Version mit einem 4 · 20-Zeichen-Display nicht wesentlich größer als das Display selbst.

Inbetriebnahme

Das beschriebene HMI-Modul (HMI =

in Elektor 7-8/2011 [5]. Die Zahl „1114“ tauschen Sie überall im Beitrag gegen „1343“ aus. Verbinden Sie den LPC-link mit Steckverbinder K3 des HMI-Moduls. Vorausgesetzt die LED (D34) ist vorhanden, müssen Sie nur den Anweisungen folgen, um die LED blinken zu lassen.

Falls Sie kein LPCxpresso-Board haben, können Sie trotzdem mit der IDE arbeiten. Ohne das Board nutzen Sie den ISP-Modus des Mikrocontrollers als USB-Stick. Dazu müssen Sie die IDE lediglich so konfigurieren, dass sie ausführbare Dateien in geeignetem Format generiert. Klicken Sie im Menü auf *Project* und anschließend auf *Properties*. Klicken Sie auf das „+“ für *C/C++ Build* und wählen Sie *Settings*, danach klicken Sie auf den Tab *Build Steps*. Fügen Sie in die Gruppe *Post-build steps* bei *Command* folgendes ein:

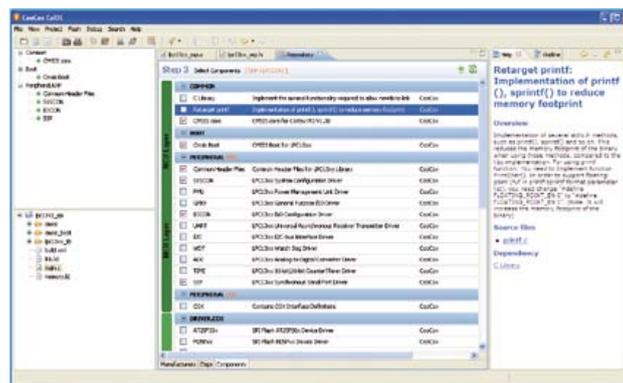


Bild 6. CoCoX CoIDE [7], eine starke, kostenlose IDE für ARM Cortex-M0 und -M3 Mikrocontroller.

Human-Machine-Interface, Mensch-Maschine-Schnittstelle) ist kompatibel mit der IDE „LPCxpresso“ (IDE = *Integrated Development Environment*, Integrierte Entwicklungsumgebung) von NXP, Embedded Artists und Code Red [2][3][4]. Zu dieser kostenlosen IDE, die für den LPC1343 und andere Mikrocontroller entwickelt wurde, gehören diverse Bibliotheken und viele Beispiele. Die IDE wird ergänzt durch LPCxpresso-Boards, auf denen sich außer einem gestaltbaren Mikrocontroller-System ein Programmer/Debugger namens „LPC-link“ befindet. Mit dem LPCxpresso-Board, das mit dem LPC1343 arbeitet, ist das hier beschriebene HMI-Modul voll kompatibel. Wenn Sie ein LPCxpresso-Board besitzen, trennen Sie den LPC-link-Teil ab und lesen Sie den Beitrag „LPCxpresso für Anfänger“

```
arm-none-eabi-size
${BuildArtifactFileName};
arm-none-eabi-
objcopy -O ihex
${BuildArtifactFileName}
${BuildArtifactFileNameBaseName}.
hex; arm-none-eabi-
objcopy -O binary
${BuildArtifactFileName}
firmware.bin; checksum
firmware.bin;
```

Einige dieser Kommandos sind bereits vorhanden, dort muss das Zeichen „#“ entfernt werden. Vorstehende Kommandos haben zur Folge, dass die generierte Datei den Namen **firmware.bin** erhält und die Checksumme angehängt wird. Der Mikrocontroller erkennt daran, dass es sich um eine ausführbare Datei handelt. Schließen Sie die Eingabe ab, indem Sie auf „Ok“ klicken.

Nach dieser Modifikation generiert die IDE



Quelle: Fritzgerald, Jakketo und Cromheecke, Taco Zip 3 – Heavy Kunst, 1992, mit freundlicher Erlaubnis von Luc Cromheecke

ausführbare Dateien, die Sie wie bei einem USB-Stick in den Mikrocontroller kopieren können (dort ist immer nur eine Datei „firmware.bin“ vorhanden). Ein Bug im USB-Treiber des LPC1343 ist schuld daran, dass der „USB-Stick“ vom Betriebssystem manchmal erst nach etwa 30 Sekunden erkannt wird.

Außer der IDE „LPCXpresso“ sind weitere kostenlose Optionen verfügbar. Die Website von microBuilder [6] beschreibt, wie eine Programmierumgebung für den LPC1343 auf der Basis von Yagarto eingerichtet werden kann. Auf dieser Website steht auch eine Bibliothek zum Download bereit, die beim Schnellstart mit dem Mikrocontroller Hilfestellung leistet. Wir haben diese Bibliothek für die Entwicklung unseres Testprogramms [1] genutzt.

In gewissem Sinn verwandt mit LPCXpresso ist CoCoX CoIDE [7] (Bild 6). Zu dieser IDE, bei der ebenso wie bei LPCXpresso Eclipse im Hintergrund steht, gehören zahlreiche Treiber für die Mikrocontroller-Peripherie sowie

einiges mehr, beispielsweise das RTOS CoOS. Erhältlich ist auch ein Programmer und Debugger, der auf dem LPC1343 aufsetzt. Natürlich können Sie ihn käuflich

Weblinks

- [1] www.elektor.de/110274
- [2] <http://ics.nxp.com/lpcxpresso/>
- [3] www.embeddedartists.com/products/lpcxpresso/
- [4] <http://lpcxpresso.code-red-tech.com/LPCXpresso/Home>
- [5] www.elektor.de/110448
- [6] www.microbuilder.eu/projects/LPC1343ReferenceDesign/LPC1343Toolchain.aspx
- [7] www.coocox.org/
- [8] www.elektor.de/080213
- [9] www.coilcraft.com
- [10] <http://elektorembdedded.blogspot.com>

erwerben (beachten Sie die Versandkosten!), doch da es sich um ein Open-Source Hardware-Projekt handelt, können Sie ihn

auch selbst bauen. Mit der Option „ISP über USB-Stick“ des Mikrocontrollers wird hier ein „Henne-oder-Ei“-Problem umgangen. Denken Sie bitte daran, dass weder LPCXpresso noch CoIDE automatisch die Checksummen ausführbarer Dateien berechnen.

8 Bit statt 32 Bit

Wir sind sicher, dass viele Elektor-Leser den Einstieg in die Welt der 32-bit-Mikrocontroller eher skeptisch betrachten, obwohl diese Mikrocontroller inzwischen leichter handhabbar und sogar preiswerter als 8-bit-Mikrocontroller sind. Für diese Leser werden wir in einer nächsten Elektor-Ausgabe ein vergleichbares, jedoch vereinfachtes HMI-Modul beschreiben, das mit einem 8-bit AVR-Mikrocontroller arbeitet. Das 8-bit-Modul, aufgebaut mit Standard-Komponenten ohne SMDs, wird

zu Arduino, Mikroelektronika und BASCOM-AVR kompatibel sein.

(110274)gd

Änderungen der microBuilder-Bibliothek

Wie schon im Hauptteil erwähnt, wurde für das Testprogramm die Open-Source LPC1343-Bibliothek von microBuilder [6] genutzt. Das Testprogramm, ebenfalls Open-Source, ist auf [1] verfügbar. Es war notwendig, die Bibliothek von microBuilder zu ändern, da ihre Eigenschaften nachbesserungsbedürftig erschienen. Die originale Bibliothek ist hier nicht mehr einsatzfähig. Dies sind die wichtigsten Änderungen:

- gpioSetValue: Zukünftig muss die Version *bit masked* von GPIOonDATA verwendet werden, um Konflikte zu vermeiden, falls diese Funktion von einer Interruptroutine aufgerufen wird.
- UART: Der Puffer *uartRxBuffer* wurde durch einen allgemeinen Puffer mit dem Namen *uartBuffer* ersetzt. Hinzugefügt wurde die interruptgesteuerte Datenübertragung.
- SPI: Hier wurden verschiedene Änderungen vorgenommen.
- cmd.c: Es fehlte ein Algorithmus, der Overflows des Zeichenpuffers verhindert.

Von den Machern von Elektor!

elektor **SPECIAL PROJECT**

RÖHREN 7

High-End und Musik

11007
(D) 16,90 €
(A) 18,50 €
CHF 28,90
(L) 18,50 €
(B) 18,50 €

- 100 Jahre Triode:**
Herzlichen Glückwunsch
- Filter:**
LCR-Siebketten verstehen
- FET-ter Sound:**
JFET simuliert Triode



Röhrensaltungen in der Praxis:

- Eintakt-A-Endstufe mit der Triode 833
- Phonovorstufe in Röhrentechnik
- Mehrkanalverstärker für Surroundanlagen
- High-End-Audio
- Digital-Analog-Converter

Jetzt neu am Kiosk!

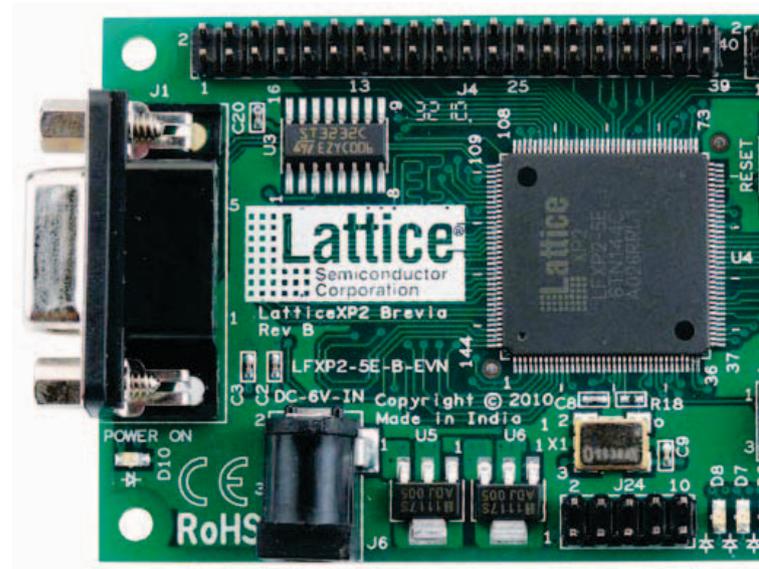


Oder frei Haus unter www.elektor.de bestellen!

Moderner Logikbaukasten

Preiswertes FPGA-Board für den Einstieg in die Welt der programmierbaren Logik

Längst ist die Sammlung von TTL- und CMOS-Bausteinen verstaubt und das Steckbrett hat ausgedient. Mikrocontroller dominieren bei der Lösung digitaler Aufgabenstellungen. Für einen richtigen Hardware-Fan sind FPGAs aber häufig die interessantere und bessere Möglichkeit, digitale Schaltungen zu ersetzen.



Von Peter Schulz (D)

Wer erinnert sich nicht mehr gerne daran: Ein Magazin voll mit TTL-Bausteinen, ein Steckbrett und viele Kabel luden zum Experimentieren ein. Mit jedem Bit war man ver-

traut, jedes kleinste Detail einer Schaltung konnte beobachtet und verstanden werden. Aus dem professionellen Bereich ist diese Vorgehensweise längst verschwunden. ICs beinhalten heute komplette Systeme, die man kaum noch bis ins Detail verstehen kann. Der ambitionierte Elektroniker ist

längst zu den Mikrocontrollern abgewandert. Die Entwicklungsumgebungen sind sehr preiswert oder sogar kostenlos erhältlich. Nur Umdenken muss man: Boolesche Algebra und Karnough-Tafeln haben ausgedient. Jedes Problem ist so umzustellen, dass man es sequentiell lösen kann. Wer

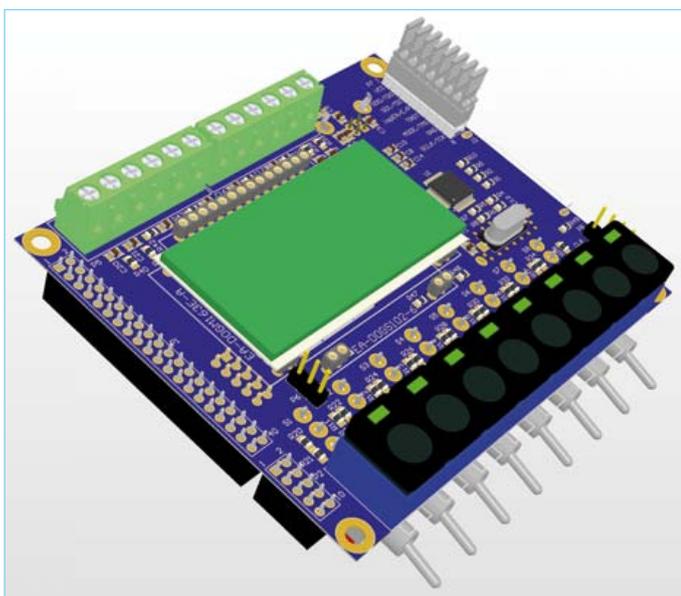


Bild 2. Für den Lehrbetrieb entwickeltes Erweiterungsboard.

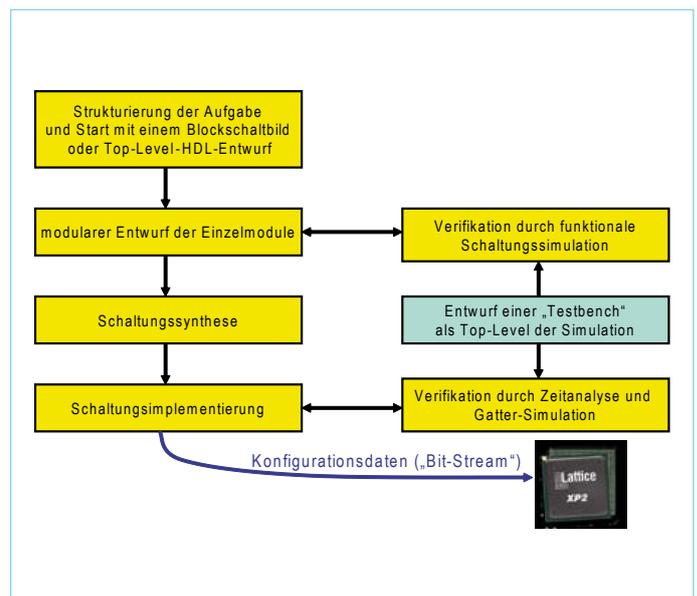


Bild 3. Der Entwicklungsablauf beim Schaltungsentwurf mit FPGAs.

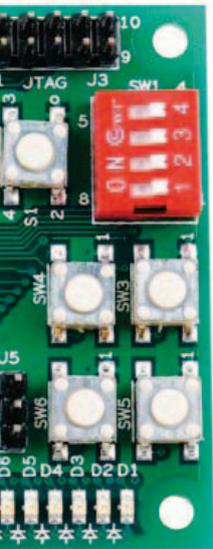


Bild 1. Das Brevia-Board: Kompakt und erweiterungsfreudig.

sich damit nie so richtig anfreunden konnte, dem bietet die moderne Elektronik mit den FPGAs eine interessante und leistungsfähige Alternative.

Auswahl

Wie auch die Hersteller von Mikrocontrollern, so beglücken uns inzwischen

Firmen wie Altera [1] oder Xilinx [2] mit kostenlos zu nutzender Entwicklungssoftware für ihre FPGA-Bausteine. Lediglich die Beschaffung der Hardware in Form der so genannten Starterkits oder Evaluation-Boards verursacht einige Kosten. Die Preise hierfür liegen aber seit einigen Jahren schon in Bereichen, die auch für Schüler und Studenten erschwinglich sind. Von der Preispolitik der FPGA-Hersteller profitieren Bildungseinrichtungen und kleine Unternehmen gleichermaßen. Die kostenlose Abgabe der Entwicklungssoftware ist dabei ein Marketinginstrument, das den Absatz der Bausteine ankurbeln soll.

Budgetgründe dürften somit nicht gegen die Beschäftigung mit FPGAs sprechen. Aber wie sieht es mit der Beherrschbarkeit der Entwicklungswerkzeuge aus? Hier hilft zum Beispiel der Elektor-Verlag mit seiner neu gestarteten Seminarreihe, die in diesem Jahr mit dem Grundlagenseminar „FPGA - Der Logikbaukasten für Jedermann“ [4] angelaufen ist. In diesem Seminar wird das „LatticeXP2 Brevia Development Kit“ [5] verwendet.

Ein Argument für die Auswahl des Brevia Development Kits ist der niedrige Preis von derzeit 49 US-\$ (zzgl. Versandkosten) im Onlineshop von Lattice [6]. Bei diesem Preis darf man von der Ausstattung keine Wunder erwarten. Aber gerade die geringe Komplexität macht die Hardware für einen Anfängerkurs besonders geeignet. Das Board ist auch für Mikrocontroller-Fans interessant, dafür sorgen externes RAM, exter-

Ausstattung

LatticeXP2 Brevia Evaluation Board

- FPGA LatticeXP2-5E 6TN144C (5000 LUT)
- 2-Mbit-SPI-Flash-Speicher
- 128 kB SRAM
- RS232 Schnittstelle mit 9-poligem DSUB-Steckverbinder
- Erweiterungsstecker 40-polig

- Erweiterungsstecker 10-polig
- JTAG Programmieranschluss 10-polig
- 4 frei belegbare Taster
- 1 Reset-Taster
- 1 frei belegbarer 4-fach DIP-Schalter
- 8 frei belegbare LEDs (SMD)

ner Flash-Speicher sowie ein serieller Port (RS232).

Auf dem Foto des Boards (**Bild 1**) klar zu erkennen ist das FPGA im TQFP144-Gehäuse. Es handelt sich um den Typ LFXP2-5E. Mit 5000 Look-Up-Tabellen (LUTs) ist dieser Baustein der kleinste Vertreter der XP2-Familie [6]. Bei dieser Familie handelt es sich um so genannte SRAM-basierte

FPGAs. Die Ausstattung mit Bedien- und Anzeigeelementen scheint auf den ersten Blick etwas bescheiden zu sein. Der Anwender kann vier Tasten, vier DIP-Schalter und acht SMD-Leuchtdioden frei verwenden, was für einen Grundkurs auf jeden Fall ausreichend ist. Über gebräuchliche Pfostenverbinder kann das Board leicht in eigene Projekte integriert werden. An der Fachhochschule Dortmund wurde ein Erwei-

FPGA-Grundbegriffe

FPGAs (Field Programmable Gate Arrays)

Bausteine der programmierbaren Logik, die auf einer matrixförmigen Anordnung von kleinen Logikelementen basieren, die über Verbindungsressourcen flexibel miteinander verknüpfbar sind.

LUT (Look-Up-Table)

Mögliche Realisierungsform für den kombinatorischen Anteil einer FPGA-Logikzelle. Kann jede beliebige boolesche Funktion der angeschlossenen logischen Eingänge realisieren. Dazu wird die komplette Wahrheitstabelle dieser Funktion in einem kleinen RAM abgelegt. Die logischen Eingänge dienen dabei als Adresse. Der ausgelesene Datenwert entspricht dem Ausgabewert der booleschen Funktion. Trotz der Realisierung als RAM verhält sich die programmierte LUT wie ein ROM. In neueren FPGAs kann die LUT auch direkt als RAM oder als Schieberegister genutzt werden. Die Lattice-XP2-Familie beinhaltet LUTs mit vier Eingängen.

SRAM-basiertes FPGA

Hierbei befinden sich die Konfigurationsdaten (LUT-Inhalte und Verbindungseinstellungen) in flüchtigem SRAM-Speicher. Dieser muss nach dem Einschalten des Bausteins erst aus einem nichtflüchtigen Spei-

cher geladen werden. Das ist üblicherweise ein externes PROM oder, wie bei der Lattice-XP2-Familie, ein interner Flash-Speicher.

JTAG-Programmierschnittstelle

Ursprünglich für den Leiterplattentest entwickelte serielle Schnittstelle. Bei FPGAs dient sie zum Download der Konfigurationsdaten in den internen Speicher.

Soft-Core

Eine Funktionseinheit höherer Komplexität, z.B. ein Mikroprozessor, die zusammen mit der Anwenderlogik implementiert wird und somit auch LUTs belegt.

Hard-Core

Eine Funktionseinheit höherer Komplexität (z.B. ein Dual-Port-RAM) die im FPGA-Baustein fest eingebaut ist und somit keine LUTs belegt.

VHDL (Very High Speed Integrated Circuit Hardware Description Language)

Eine Sprache, mit der sich Hardware-Strukturen und -Funktionen formulieren lassen. Sie ist textbasiert und kann nicht direkt ausgeführt werden. VHDL-Beschreibungen lassen sich für die Simulation compilieren. Aus einer Untermenge der Sprache lassen sich digitale Schaltungen synthetisieren.

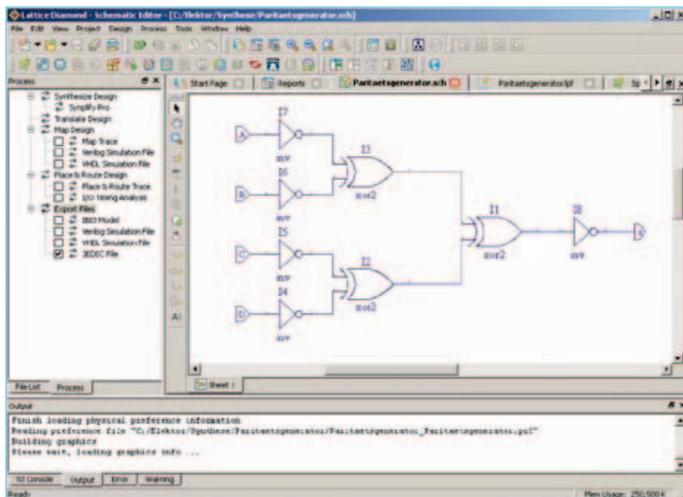


Bild 4. Der einfache Einstieg mit der Schaltpläneingabe.

Der Autor

Dr. Peter Schulz ist seit 2004 Professor für Messtechnik und Mikroprozessortechnik an der Fachhochschule Dortmund. Seit 1991 hat er sich in Forschung und Industrie intensiv mit der Anwendung von FPGAs beschäftigt. Diese Technik spielt auch heute bei seinen Lehr- und Forschungsaktivitäten eine bedeutende Rolle. Prof. Schulz ist Sprecher des Instituts für Mikroelektronik und Eingebettete Systeme (IMES).

terungsboard (Bild 2) entwickelt, das mit robusteren Bedienelementen auch den Dauereinsatz im Lehrbetrieb ermöglicht.

Zum Lieferumfang gehören noch ein international einsetzbares Steckernetzteil, ein Verbindungskabel für die serielle Schnittstelle und ein Downloadkabel für die Programmierung des FPGAs über den JTAG-Anschluss [8]. Um das Downloadkabel einsetzen zu können, benötigt man noch einen PC mit „echtem“ Parallelport. Ist diese Voraussetzung nicht erfüllt, muss man ein passendes USB-Kabel separat erwerben ([3] oder [7]).

Einstieg

Im Auslieferungszustand ist bereits eine Applikation einprogrammiert, mit der man das Board in Betrieb nehmen kann [9]. Dazu braucht man nur das Steckernetzteil anzuschließen und mit dem seriellen Kabel die

Verbindung zum COM-Port eines PCs herzustellen. Jetzt muss man noch ein Terminalprogramm starten, die Verbindungsparameter einstellen - und schon kann man über einfache Tastaturkommandos die Komponenten des Boards testen.

Das hört sich aber eher wie die Vorgehensweise bei Mikrocontrollern an. Haben wir es hier nicht mit einem FPGA zu tun? Einfache Antwort: Lattice demonstriert mit der Vorkonfiguration eines so genannten Soft-Cores in FPGAs auch sein eigenes Mikrocontrollersystem implementieren kann. Dieser Soft-Core (Mico8) [10] führt ein in Assembler oder C geschriebenes Testprogramm aus.

Für die „richtige“ Arbeit mit dem FPGA-Board benötigt man die Entwicklungssoftware „DIAMOND“ von Lattice. Diese kann man bei Lattice downloaden [3] und nach Registrierung eine freie Lizenz anfordern. Diese freie Lizenz ist auf bestimmte FPGA-Familien beschränkt, funktioniert aber mit dem Brevia-Board einwandfrei.

Wer noch nicht mit Hardwarebeschrei-

bungssprachen (VHDL oder Verilog) vertraut ist, der kann seine FPGA-Schaltung auch mittels eines Schaltplanelitors entwerfen. Die dazugehörige Bibliothek enthält unter anderem logische Grundfunktionen (Gatter), Flipflops und FPGA-spezifische Elemente.

Die Schaltplanmethode kann aber nur der Einstieg sein. Für komplexere Projekte ist es erforderlich, sich mit der Verwendung von Hardwarebeschreibungssprachen (HDL) für das Entwerfen von Schaltungen vertraut zu machen. Dazu gehört der Einsatz des Simulationsprogramms „Active-HDL Lattice Web Edition“ von Aldec [3]. Auch dafür stellt Lattice eine kostenfreie Lizenz zur Verfügung.

Fazit

Preiswerte FPGA-Boards kommen mit durchaus professionellen Entwicklungswerkzeugen einher. Auf den Umgang damit konnte hier nicht näher eingegangen werden. Wer sich auf den Lernprozess einlässt, kann sehr interessante Projekte realisieren. Dabei helfen auch einige der im Literaturverzeichnis angegebenen Bücher [11, 12, 13]. Wer ganz legal fremdes geistiges Eigentum (IP = intellectual property) einsetzen möchte, der sei auf die Website von „Opencores“ [14] verwiesen.

(110348)

Links und Literatur:

- [1] www.xilinx.com
- [2] www.altera.com
- [3] www.latticesemi.com
- [4] Peter Sauer, Peter Schulz: „FPGA - Der Logikbaukasten für Jedermann“, Handout zum gleichnamigen Elektor-Seminar 2011
- [5] www.latticesemi.com/documents/XP2-Brevia%20User%20Guide.pdf
- [6] www.latticesemi.com/products/developmenthardware/developmentkits/

- xp2breviadevelopmentkit.cfm
- [6] Lattice Semiconductor Corporation: „LatticeXP2 Family Data Sheet“, August 2008
- [7] www.hardware-design.de
- [8] IEEE Standard Test Access Port and Boundary-Scan Architecture
- [9] Lattice Semiconductor Corporation: „LatticeXP2 Brevia Development Kit QuickSTART“, Juni 2010
- [10] Lattice Semiconductor Corporation: „LatticeMico8 Microcontroller User’s Guide“, November 2010

- [11] Peter Sauer: „Hardware-Design mit FPGA“, Elektor-Verlag, Aachen, 2010
- [12] Peter J. Ashenden: „Digital Design – An Embedded Systems Approach Using VHDL“, Morgan Kaufmann Publishers, 2008
- [13] Jürgen Reichardt, Bernd Schwarz: „VHDL-Synthese – Entwurf digitaler Schaltungen und Systeme“, 5. Auflage, Oldenbourg Verlag München, 2009
- [14] www.opencores.com

FIRST STEP

NEU!

Erste Schritte mit dem Mikrocontroller

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

Bestandteile des „First Step“-Pakets:



→ 3 Arbeitshefte

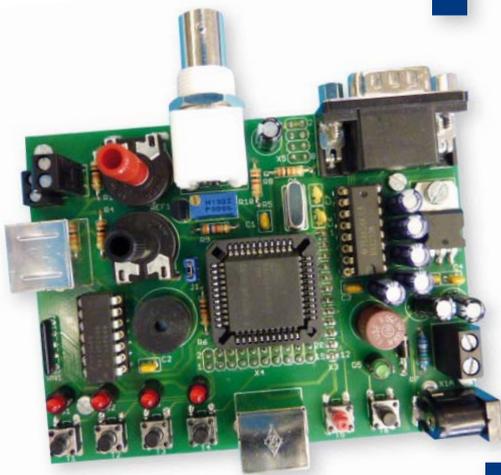
(inkl. passendes DIN A4-Ringbuch)

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



→ 1 „First Step“-Mikrocontroller-Board

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



→ 1 CD-ROM mit Zusatzinfos

- Datenblätter
- Systemdokumentation
- Entwicklungsumgebung
- Beispielprogramme

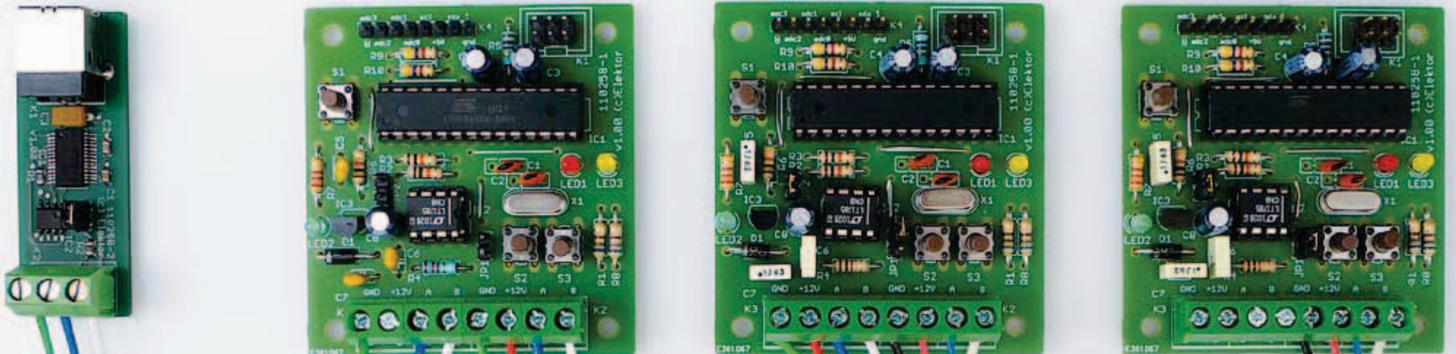
Das „First Step“-Paket kann jetzt zum Einführungspreis von nur **179,00 €** (statt ~~199,00 €~~) bestellt werden.

Dieses Angebot ist gültig bis 31.08.2011!

Weitere Infos und Bestellung unter
www.elektor.de/fs

Hier kommt der Bus (7)

Ein einfaches Anwendungs-Protokoll



Nach einer kleinen Sommerpause fährt unser Bus nun weiter: In dieser Ausgabe stellen wir ein einfaches Protokoll vor, mit dem sich bis zu vier Soll- oder Ist-Werte gleichzeitig übermitteln lassen. Geeignet ist das Ganze nicht nur für die Hausautomatisierung, sondern auch für andere Aufgaben aus dem Bereich Messen, Steuern und Regeln. Außerdem arbeiten wir erstmals auch mit dem AVR Studio und C.

Von Jens Nickel

In den bisherigen Folgen der Serie hatten wir ein einfaches Rahmenprotokoll vorgestellt, mit dem sich Nutzdaten von einem Sender zu einem adressierbaren Empfänger verschicken lassen. Zur Erinnerung schau man sich **Bild 1** an. Eine Nachricht unseres *ElektorMessageProtocols* besteht grundsätzlich aus 16 Bytes, wobei das Byte 0 zu Synchronisationszwecken immer gleich $AA_{hex} = 170$ ist. Wenn die Bits 7 und 6 des nachfolgenden *ModeBytes* gleich 0 sind, werden die Bytes 2 bis 5 zur Adressierung verwendet. Da die Bytes E und F optional noch einen Prüfwert aufnehmen, bleiben acht Bytes für Nutzdaten.

In der letzten Folge haben wir außerdem ein einfaches System zur Regelung des Busverkehrs genutzt (*HybridMode*). Knoten, die

regelmäßig etwas mitzuteilen haben (wie zum Beispiel Sensoren) werden von einem *Scheduler* reihum angesprochen. Dazwischen liegen sogenannte *FreeBusPhases*, in denen sich Knoten auch ohne spezielle Aufforderung äußern dürfen. Da hier Kollisionen auftreten können, müssen die *NonScheduledMessages* (Nachrichten, die in der Freie-Bus-Phase übermittelt werden) vom Empfänger bestätigt werden. Hierzu sendet er eine sogenannte *AcknowledgeMessage* zum Sender zurück.

Die *NonScheduledMessages* werden vor allem dann nötig, wenn ein Knoten aufgrund eines plötzlichen Ereignisses etwas zu sagen hat, der sonst recht schweigsam ist und aus Effizienzgründen daher nicht regelmäßig abgefragt wird. Ein Beispiel aus dem Domotikbereich könnte ein Lichtschalter sein. Doch auch ein Sensor, der eine

Grenzwertüber- oder -unterschreitung meldet, fällt in diese Kategorie; man denke zum Beispiel an einen Wasserstandsmelder.

Unterknoten

Der *HybridMode* erlaubt es prinzipiell aber auch, dass ein Knoten regelmäßig abgefragt wird **und** ereignisgesteuert in der *FreeBus-Phase* sendet. Man denke an einen Temperatursensor, der periodisch aktuelle Temperaturwerte übermittelt, aber auch einen Grenzwert überwacht. Diese Möglichkeit hatte ich im letzten Teil nicht ausdrücklich erwähnt; stattdessen wurde in der Demo-Software strikt nach „*PolledNodes*“ (oder besser gesagt „*ScheduledNodes*“) und „*FreeBusNodes*“ unterschieden [1]. Prompt machte mich ein mitdenkender Leser darauf aufmerksam, dass sich beides ja nicht ausschließen würde. Francis Stevenson schlug darüber hinaus vor, dass solch ein Sensor in jedem Fall

Elektor Produkte & Service

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

- Gratis Software-Download (Firmware in BASCOM und C plus PC-Software)

Alle Produkte und Downloads sind über die Website zu diesem Artikel erhältlich: www.elektor.de/110382

zuerst die Grenzwertüberschreitung melden sollte, egal ob nach einer periodischen Aufforderung oder in der *FreeBusPhase*. Ein guter Gedanke, der bei besonders eiligen Messages wirklich nützlich sein kann.

Per Mail diskutierten Francis und ich auch die Möglichkeit, auf ein- und demselben physikalischen Knoten verschiedene *Devices* („Geräte“) ansprechen zu können. Ein Knoten-Board ist dann gleichzeitig unter mehreren Adressen erreichbar, was Hardwarekosten spart. Die Prozessorfirmware muss dafür sorgen, dass die Anfragen an die verschiedenen Untereinheiten weiterverteilt werden. Im Grunde haben wir das schon in der Demo-Software des letzten Teils realisiert [1], denn der PC diente gleichzeitig als Scheduler (Adresse 0) und Master (Adresse 10).

Mehr Kanäle

Wenn aber nur ein paar einfache Sensoren und/oder Aktoren an einem gemeinsamen Knoten hängen, ist die Aufsplittung in *Devices* mit eigener Adresse gar nicht nötig. Es wäre sogar ineffizient, wenn alle Sensoren eine separate Message mit ihrer eigenen Senderadresse zum Master schicken würden, um ihre Werte mitzuteilen. Wir bedienen uns hier besser dem Konzept der Kanäle oder *Channels* (DMX lässt grüßen ☺). Da wir pro Message acht Nutz-Bytes zur Verfügung haben, ist es ohne weiteres möglich, vier Temperaturwerte gleichzeitig zu übermitteln, die wir auf jeweils zwei Bytes aufteilen. Das deckt sich übrigens sehr schön mit der Hardware unseres Experimentalknotens, wo vier ADC-Eingänge mit jeweils einem Pin der Stiftleiste K4 verbunden sind.

Welche Bytes innerhalb der Nutzdaten zu welchem Channel beziehungsweise Sensor gehören, ergibt sich dann einfach durch deren Position, das heißt es wird grundsätzlich immer der Wert von *Channel0*, dann von *Channel1* und so weiter gesendet (siehe **Bild 2**). Mit Hilfe dieses Konzepts können natürlich auch vier Aktoren über eine einzige Message mit Werten angesteuert werden, vorausgesetzt, die Werte sind mit Hilfe von zwei Bytes zu codieren.

In der Demo-Software des letzten Teils haben wir zwei Bytes genutzt, um einen über den Controller-ADC eingelesenen

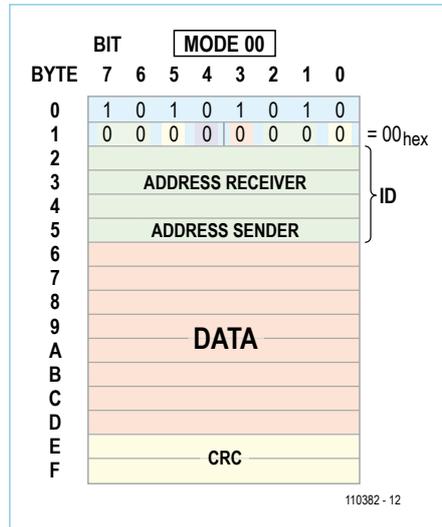


Bild 1. Das *ElektorMessageProtocol* definiert nur den Rahmen für die Nutzdaten [8].

10-bit-Wert zu übermitteln. Wir hatten dabei die unteren sieben Bits in eines der Nutz-Bytes und die oberen drei Bits in ein anderes Nutz-Byte gepackt. Dies hatte den Vorteil, dass ein Wert AA_{hex} innerhalb der Nutzdaten generell ausgeschlossen war, anderenfalls wäre unsere einfache Synchronisation durcheinandergekommen. Denselben „Trick“ verwenden wir nun auch für unsere Channel-Werte, und dabei sind wir schon mittendrin in unserem Anwendungs-Protokoll!

ElektorApplicationProtocol

So ein Anwendungsprotokoll ist nötig, damit sich die Knoten (Sensoren, Aktoren) auf dem Bus auch verstehen, und eine spätere Erweiterung durch neue Hardware einfach ist. Damit wir nicht in einigen Monaten schon wieder ein neues Protokoll erfinden müssen, sollte unser *ElektorApplicationProtocol* noch einigermaßen einfach, aber auch vielseitig sein und mindestens folgende Bedingungen erfüllen:

- Übermitteln von 10-bit-Werten plus Vorzeichen, etwa von einem Sensor zur Datenauswertung oder umgekehrt von

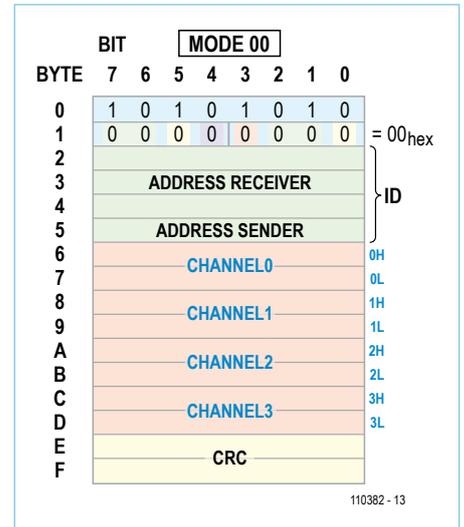
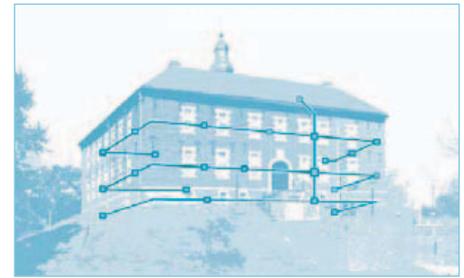


Bild 2. Unser Anwendungsprotokoll teilt die acht Nutz-Bytes in vier Byte-Paare auf. Hintereinander werden jeweils die Werte von vier *Channels* (Sensoren/Aktoren auf dem Knoten) übertragen.

- einer Steuerung zu einem Aktor.
- Das Ganze wahlweise mit 20 bit plus Vorzeichen, wofür wir noch einen 4-Byte-pro-Channel-Modus benötigen.
- Einstellung einer Einheit und eines Skalenfaktors für intelligente Sensorknoten.
- Einstellung eines Mess-Intervalls für Sensorknoten.
- Einstellung von mehreren Grenzwerten.
- Übermittlung von Grenzwert-Alarmen.
- Konfiguration und Aufruf von Presets (Voreinstellungen) bei Aktoren.
- Unterscheidung einer *AcknowledgeMessage*, welche die empfangenen Werte zur Kontrolle zum Sender zurücktransportiert, von der Original-Nachricht. (So eine Unterscheidung hatten wir schon in der Software des letzten Teils implementiert.)

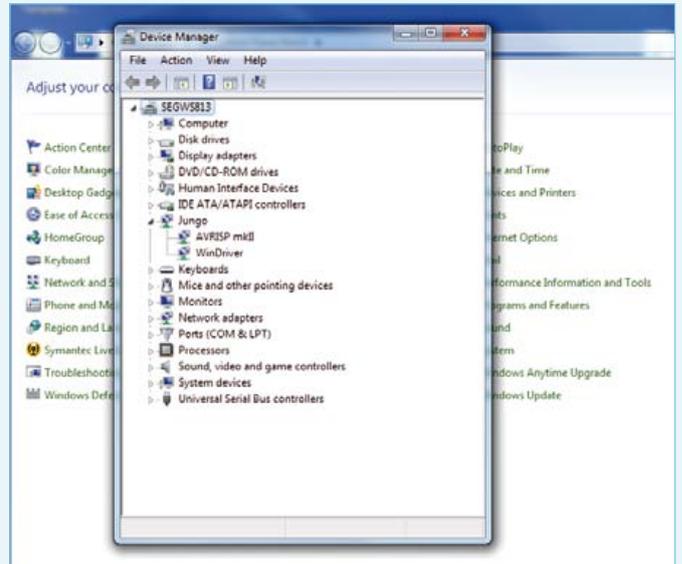
Das Protokoll soll übrigens weder auf die gemeinsame Verwendung mit dem *HybridMode*, noch auf Domotik-Anwendungen beschränkt sein. Es wird sich durchaus auch zur Fern-Abfrage eines Messgeräts oder eine andere Punkt-zu-Punkt-Verbindung eignen (bei der es ja gar nicht zu Kollisionen kommen sollte).

Parallel mit AVR Studio und BASCOM

Nach Abschluss unserer Sommerausgabe hatte ich etwas Luft, mein Versprechen, dass C-Code folgen würde, eigenhändig in die Tat umzusetzen. Die neueste Version 5 der Entwicklungsumgebung AVR Studio bringt gleich einen integrierten C-Compiler (AVRGCC) und eine mächtige, komfortable Benutzeroberfläche mit, sie ist nach Registrierung kostenlos bei Atmel downzuladen [3].

Meine erste Aufgabe bestand darin, aus AVR Studio Kontakt zum AVRISP-mkII-Programmer aufzunehmen. Obwohl bei der Installation der Entwicklungsumgebung auch schon der passende Treiber installiert worden war, gelang dies zunächst nicht. Der Programmer war nämlich noch mit dem LibUSB-Treiber verknüpft, den ich zur Entwicklung mit BASCOM installiert hatte. Erst als ich den LibUSB-Treiber kurzerhand deinstallierte, funktionierte das Programmieren über AVR Studio. Und zwar ohne weitere Mühe, das simple Anstecken des Programmers an einen USB-Anschluss genügte. Im Device Manager von Windows 7 muss das Ganze so wie im Screenshot aussehen. Natürlich wollte ich weiter parallel mit BASCOM arbeiten, was auch funktioniert; der LibUSB-Treiber muss dann als sogenannter Filtertreiber installiert werden. Eine Anleitung hierzu findet sich unter [4].

Beim Anlegen eines neuen Projektes in AVR Studio wird man nach dem Zielprozessor gefragt. Erfreulicherweise ist sonst nicht allzu viel zu konfigurieren. Ein Druck auf F7 (oder „Build Solution“ im Menü) erzeugt aus dem Sourcecode und den referenzierten Bibliotheken das entsprechende Hex-File. So wird man davor verschont, eigenhän-



dig ein Make-File anlegen zu müssen, das den Build-Prozess steuert. Zum Flashen drückt man auf den Button mit dem kleinen Blitz. Im aufgehenden Programmer-Fenster ist zuerst oben der Knopf „Apply“ zu drücken (Tool=AVRISP mkII, Device=ATmega88, Interface=ISP). Unter dem Menüpunkt „Memories“ findet man den Knopf „Program“. In der Combobox darüber ist vorher der Pfad für das Hex-File einzutragen.

Steuerung mit 10 bit

Damit aus diesem Artikel kein Datenblatt wird, wollen wir die obengenannten Funktionen in mehreren Schritten einführen. Wir beginnen mit der Übermittlung eines 10-bit-Wertes, den wir wie oben besprochen auf die zwei Bytes von *Channel0* aufteilen (genannt *OH* und *OL* für High und Low). Wie in **Bild 3** gezeigt, bleiben die Bits *OH.7* und *OL.7* (also die beiden höchstwertigen Bits unserer zwei Nutz-Bytes) immer 0, so dass ein Wert AA_{hex} ausgeschlossen ist. Rein rechnerisch verschwenden wir damit 50 % der möglichen Werte, doch werden wir die übrigen erlaubten Werte später noch für Spezialfunktionen nutzen.

Bit *OH.3* ist für das Vorzeichen reserviert (1 = -, 0 = +), die Datenbits *D9* bis *D0* werden wie oben beschrieben aufgeteilt. Damit bleiben noch 3 Bits übrig, die wir folgendermaßen verwenden:

Bit *OH.6* bestimmt, ob zwei Bytes (=1) oder vier Bytes (=0) pro Channel verwendet werden (den *4ByteMode* brauchen wir später für genauere Werte und einige der Spezialfunktionen).

Bit *OH.5* sagt uns, ob es sich um einen Soll- oder einen Ist-Wert handelt (1 = Soll bzw. *SetValue*; 0 = Ist bzw. *CurrentValue*).

BIT		7	6	5	4	3	2	1	0	
0	1 (2/4)	SET/ CURRENT	ACK/ ORIGINAL	-	+	D9	D8	D7		xH
0		D6	D5	D4	D3	D2	D1	D0		xL

110382 - 14

Bild 3. Die beiden Bytes eines *Channels* im Detail. Es lassen sich hier Werte mit bis zu 10 bit Auflösung plus Vorzeichen übertragen.

Bit *OH.4* verwenden wir schließlich zur Kennzeichnung einer *AcknowledgeMessage* (= 1).

Als Beispiel soll uns eine Jalousie dienen, die von einem Domotik-Master aus auf 30 % Verschattung gestellt werden soll (hier reichen 10 bit Genauigkeit natürlich locker aus).

Der Domotik-Master und die Jalousie-Steuerung müssen vorher ausgehandelt haben, wie die 30 % Verschattung in Zahlen ausgedrückt werden; zur Skalierung in einer der späteren Folgen mehr. Nehmen wir an, Prozentwerte würden direkt als Zahl codiert und die Jalousie wäre über den *Channel0* erreichbar. An Position 6 und 7 unseres *Elek-*

torMessageProtocols würden also folgende zwei Bytes vom Master gesendet:
0-1-1-0-0-0-0-0 0-0-0-1-1-1-1-0
($00011110_{bin}=30_{dez}$)

Die Jalousie antwortet mit einer Bestätigung, bei der das *AcknowledgeBit* gesetzt ist:
0-1-1-1-0-0-0-0 0-0-0-1-1-1-1-0

Eine intelligente Jalousie sollte natürlich auch den momentanen Ist-Wert der Verschattung messen und rückmelden können. Eine gute Idee wäre, einen Ist-Wert von 30%-Verschattung zurückzumelden, wenn dieser nach dem Verfahren der Jalousie erreicht worden ist:

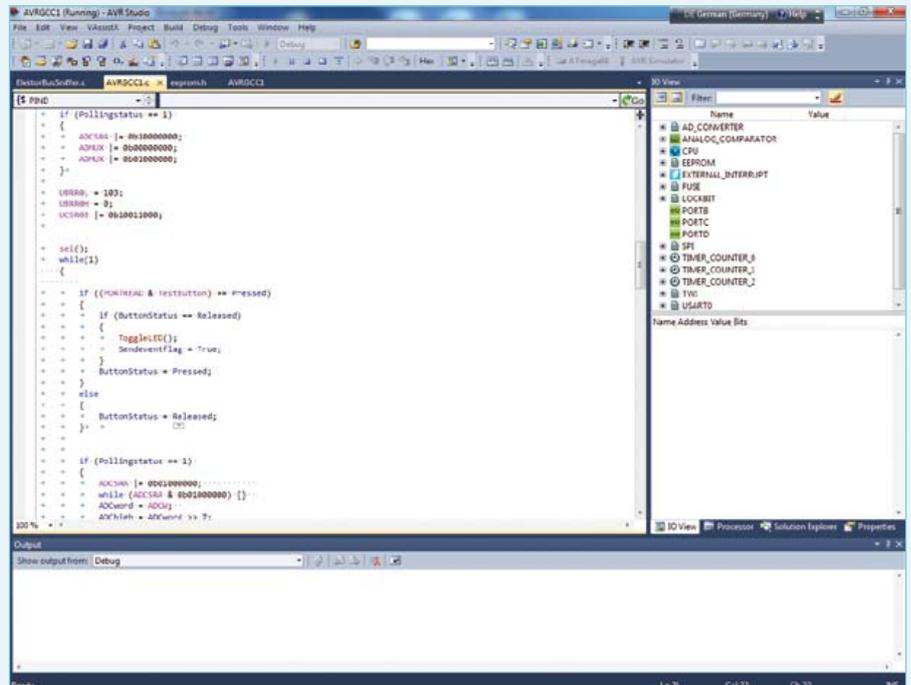
0-1-0-0-0-0-0-0 0-0-0-1-1-1-1-0

Erste C-Versuche

Bei meinen ersten Embedded-C-Gehversuchen erwies sich das Internet als enorm hilfreich (Beispiel: [5]). Zuerst ließ ich die LEDs blinken, las dann einmal Werte über den ADC ein und gab schließlich ein paar Bytes über den Controller-UART auf den Bus, die ich mir mit dem Terminalprogramm Terminal.exe [6] auf dem PC ansah. Man hüte sich dennoch vor simplem Copy & Paste: Im Internet ist diverser C-Code für AVR-Controller zu finden, den man nicht einfach übernehmen darf, ohne ins Datenblatt des ATmega88 zu schauen [7]. So kann zum Beispiel die Bezeichnung der Controller-Register bei unterschiedlichen Typen voneinander abweichen. Zum Beispiel muss man statt UDR (Register, das die zu sendenden und die empfangenen Bytes aufnimmt) beim ATmega88 UDR0 schreiben. Ein böser Stolperstein ist die Bezeichnung des Interruptvektors, mit dessen Angabe die Interruptroutine für ein empfangenes Zeichen im Code eingeleitet wird. Die im Internet an vielen Stellen zu findende Angabe von „ISR(USART_RX_vect) { ... }“ führt nämlich dazu, dass sich ein ATmega88 einfach innerhalb der Routine aufhängt. Es kostete mich ein schönes Stündchen, bis ich herausfand, dass es bei diesem Controller „ISR(USART_RX_vect) { ... }“ heißen muss.

Wer von BASCOM beziehungsweise BASIC wechselt, sollte auch besonders auf die folgenden Fehler aufpassen, bei denen der Compiler unter Umständen nicht einmal meckert. Das wäre einmal das doppelte Gleichheitszeichen bei einem Vergleich (nur bei der Zuweisung ist nur ein Gleichheitszeichen zu schreiben). Unbedingt muss man auch auf superkorrekte Groß- und Kleinschreibung achten, bei den selbst vergebenen Variablennamen, aber auch bei den Schlüsselwörtern wie „if“. Es muss außerdem unbedingt hinter jeden Funktionsaufruf ein Klammerpaar, also zum Beispiel „ToggleLED();“. Schließlich kann auch ein vergessener Strichpunkt zu völlig undurchsichtigen Fehlermeldungen führen. Man sollte also immer konzentriert noch einmal die Syntax checken, bevor man den Compiler anwirft. Als nächste Aufgabe nahm ich mir die Übertragung der Demo-Firmware von BASCOM in C vor. Der entsprechende C-Code findet sich ebenfalls auf der Projektwebsite [2]. Damit man vergleichen kann, habe ich die Struktur so weit wie möglich beibehalten (sicher gäbe es hier noch eine Menge zu optimieren; C-Kenner dürfen übrigens gerne verbesserte Versionen einsenden!).

Bei einem Vergleich sieht man sehr schön, dass man in C prinzipiell näher an der Hardware programmieren muss. BASCOM nimmt einem mit Befehlen wie „Start ADC“, „Enable Urcx“ und „Printbin ...“ hier manches ab. Dass man die Register des Controllers kennenlernen muss, ist aber nicht unbedingt ein Nachteil. Und der etwas höhere Aufwand kann mehr als ausgeglichen werden, wenn man wieder verwendbare, maßgeschneiderte Subroutinen schreibt oder nutzt. Während der weiteren Entwicklung unserer Firmware sollen solche



Unterprogramme in Form einer kleinen Bibliothek entstehen.

Zum Verständnis des Codes für C-Einsteiger noch dieses: die Ports und die anderen Register (zum Beispiel für den ADC und den UART) muss man in C immer mit einem ganzen Byte befüllen. Wenn man nur eines der Bits setzen will, muss man dafür sorgen, dass die anderen Bits erhalten bleiben.

Dies erreicht man durch Oder-Verknüpfung:

```
PORT = PORT | Bitmaske;
```

oder kürzer

```
PORT |= Bitmaske;
```

Um ein Bit zu resettet, muss man mit der inversen Bitmaske und-verknüpfen:

```
PORT &= ~Bitmaske;
```

Für unsere LEDs, Buttons und die übrigen Portpins habe ich über Befehle wie

```
#define TestLED 0b00010000
```

entsprechende Bitmasken (und Portnamen) definiert, so dass ich im Code

```
PORT |= TestLED;
```

schreiben konnte, um die TestLED leuchten zu lassen. In vielen Listings im Internet findet man aber auch die Schreibweise

```
#define TestLEDbitposition 4
```

...

```
PORT |= (1 << TestLEDbitposition);
```

wobei der Ausdruck `1 << TestLEDbitposition` durch Linksverschieben eines Bits 1 die obengenannte Bitmaske generiert.



Schließlich bestätigt der Master noch, dass er den Wert empfangen hat:

0-1-0-1-0-0-0-0 0-0-0-1-1-1-1-0

Vermutlich würde man diese Kommunikation nur abgespeckt implementieren, aber das Beispiel demonstriert recht schön den Einsatz des *Set-* und *AcknowledgeBits*.

Demo-Software

Auch zu diesem Teil bieten wir Beispiel-Software für den PC und die ATmega88-Controller an, die auf den Experimental-Knoten zum Einsatz kommen. Es handelt sich dabei um eine angepasste Version der Demo-Software aus dem letzten Teil, bei der alle drei Knoten den Status ihrer Test-LED und einer der Knoten den vom ADC0 gemessenen Wert zur PC-Software übermitteln. Die Software ist wie immer als Zip-File von der Elektor-Website downloadbar [2], die Demo-Software aus dem letzten Teil erhält man zum Vergleich unter [1].

Zur Übermittlung des ADC-Wertes (nur Knoten 2) wird *Channel0* verwendet, den LED-Status übermitteln alle Knoten über *Channel1*. Ein Byte innerhalb des EEPROMS (Adresse 006) bestimmt, ob der Knoten einen ADC-Wert sendet oder nicht (die entsprechende Variable im Code heißt *Device-mode*). Dieses Byte muss bei Knoten 2 auf

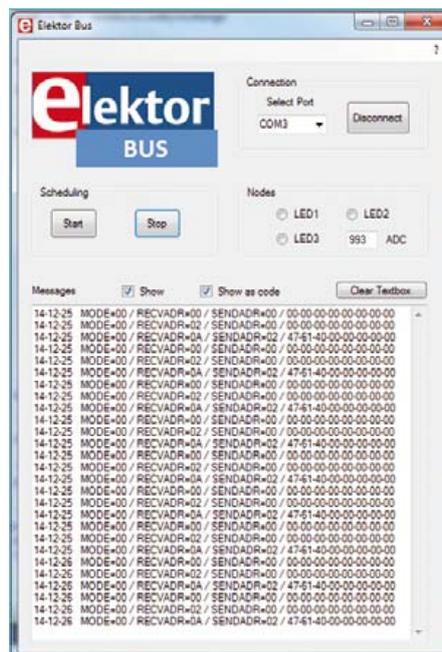


Bild 4. Screenshot der PC-Software, die den Test-LED-Status von drei Knoten und den ADC0-Wert von Knoten 2 anzeigt.

01 gesetzt werden, was man in BASCOM manuell erledigen kann. Die Variable *Polingstatus* wurde im Code außerdem in „Scheduled“ umbenannt, was besser wiedergibt, ob es sich um einen *ScheduledNode*

oder *FreeBusNode* handelt (schließt sich in dieser Softwareversion wiederum aus). Dazu gibt es nun erstmals auch Firmware in C, die mit dem neuen AVR Studio 5 erstellt wurde. Die Textkästen verraten hier mehr. Auch für BASCOM-User lohnt es sich in jedem Fall, einmal in den C-Code hineinzusehen.

Weiter geht es im Oktoberheft!

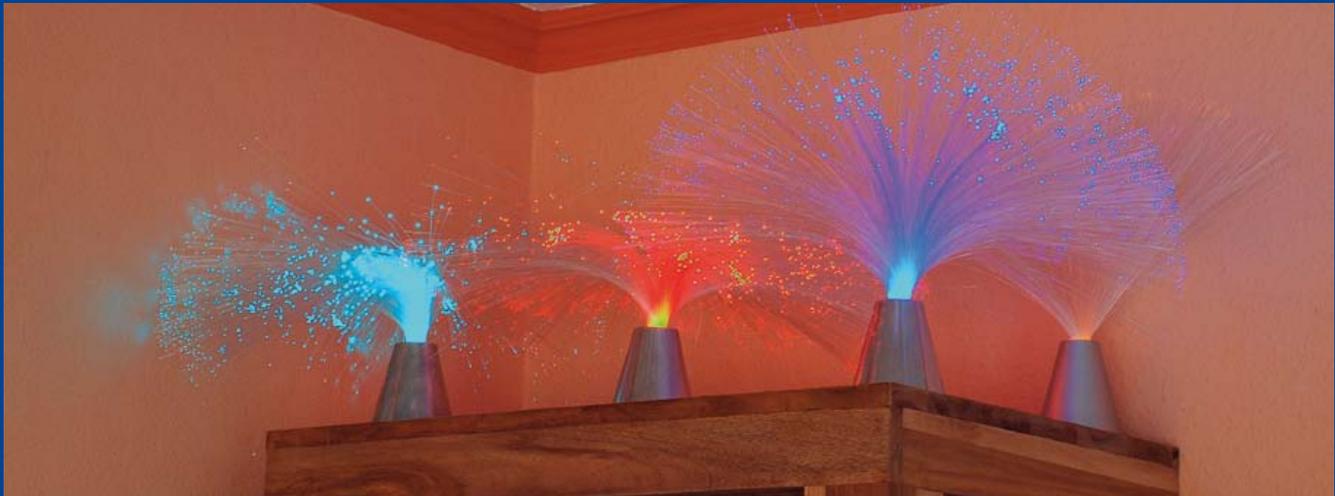
(110382)

Entwickeln Sie mit! Hinweise und Ideen sind willkommen unter redaktion@elektor.de!

Weblinks

- [1] www.elektor.de/110258
- [2] www.elektor.de/110382
- [3] www.atmel.com/dyn/products/tools_card.asp?tool_id=17212
- [4] <http://avrhelp.mcselec.com/index.html>
- [5] www.mikrocontroller.net/articles/AVR-GCC-Tutorial
- [6] www.elektronik-labor.de/RS232/2010Terminal.html
- [7] www.atmel.com/dyn/resources/prod_documents/doc2545.pdf
- [8] www.elektor.de/110012

Geisterhaftes LED-Blinken



Von Dr. Thomas Scherer (D)

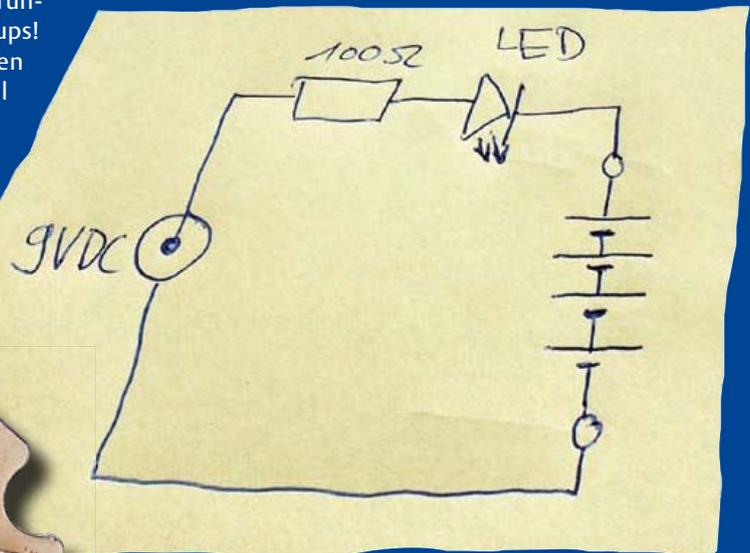
Etwa ein Jahr ist es her, da blinkte die Ladekontroll-LED meines elektrischen Korkenziehers. Bevor Sie fragen: Ja, ein richtiger Elektroniker braucht so ein Gadget! ;-)

„Nanu!“ dachte ich, „Da ist doch mehr Elektronik in der Ladestation, als ich vermutet hätte.“ Doch das Blinken war etwas unregelmäßig und die LED-Farbe vom satten Grün ins Grün-gelbe verändert. Das machte mich neugierig und schwups! war die Ladestation auseinander genommen. Drinnen fand sich: Nein, kein IC, kein Transistor und nicht einmal ein Kondensator oder eine Spule. Außer der LED war da nur ein einziger Widerstand und sonst nichts. Wie in der nebenstehenden Schaltung zu sehen ist, floss der Ladestrom für die vier NiCd-Akkus des Korkenziehers vom 9-V-Steckernetzteil direkt über die LED und wurde nur durch einen Vorwiderstand begrenzt. Ich baute die LED aus. Auch an meinem Labornetzteil blinkte die LED fröhlich weiter. Die Blinkfrequenz steigerte sich mit dem Strom.

Bei etwa 70 mA schließlich verabschiedete sich die LED. Ich war zwar etwas erstaunt, ersetzte aber einfach die defekte LED und erhöhte den Vorwiderstand auf 470 Ω , denn ein Dauerladestrom von 5 mA müsste ja ausreichen, um die Akkus stets geladen zu halten.

Ich hätte die Angelegenheit längst vergessen, wenn nicht eines Abends meine Frau „Thomas, schau mal!“ gerufen hätte. Und siehe da, der Geist der selbstblinkenden LED hatte sich reinkar-

niert. Jetzt spukte er in einer roten Power-LED, die sich in einer der Leuchten auf unserem Wohnzimmerschrank befand. Sie blinkte unregelmäßig vor sich hin und nach weiteren 15 Minuten verschied sie ohne weiteres Zutun. Auch hier bestand die gesamte Schaltung lediglich aus LED plus Vorwiderstand. Das untenstehende Bild der toten Power-LED verrät nicht viel. Sie



sieht aber ziemlich mitgenommen aus. Mein Fehler war nämlich, dass ich damals bei der Montage die Wärmeleitpaste vergessen hatte.

Damit ist aber noch nicht erklärt, warum eine LED am verfrühten Ende ihres Lebens ganz von alleine zu blinken beginnt, bevor sie ihren Geist aushaucht. Also, liebe Elektor-Leser: Schicken Sie Ihre Erklärung an „redaktion@elektor.de“. Die besten Erklärungen des Phänomens werden selbst verständlich in Elektor veröffentlicht. Ich bin gespannt...

(110459)

Alibaba

Von Thijs Beckers (Redaktion NL)

Haben auch Sie das Gefühl, dass unsere Welt immer mehr schrumpft? Zu Zeiten eines Jules Verne war es noch unvorstellbar, die Welt in achtzig Tagen zu umrunden. Heute ist eine Weltreise in achtzig Stunden kaum rekordverdächtig.

Ja, die guten alten Zeiten. Als Kind ging ich mit meiner Familie auf „große Reise“, um 260 km entfernt lebende Verwandte zu besuchen. Für mich war eine solche Reise ein Abenteuer. Heute fliegen wir geschäftlich mal eben nach London, Paris oder Amsterdam, und ein ferner Kontinent ist unser selbstverständliches Urlaubsziel.

Aus fernen Regionen dieser Welt kommen heute die meisten Elektronik-Komponenten, sie werden längst in so genannten Billiglohnländern produziert. Bis sie bei uns verfügbar sind, haben sie eine weite Reise hinter sich. Die Globalisierung von Produktion und Handel wirkt sich auch auf die Welt des Elektronik-Entwicklers aus. Er muss die Komponenten, die er braucht, weltweit recherchieren.

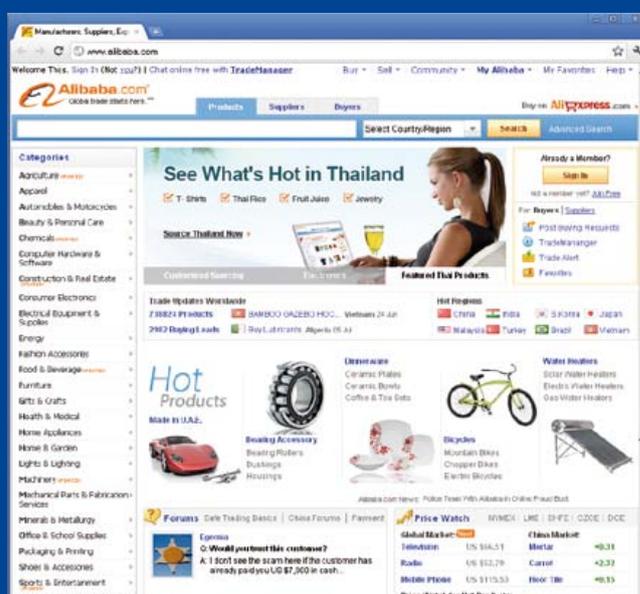
Bei meinen Recherchen im World Wide Web entdeckte ich die Seite „alibaba.com“. Mit den Märchen aus „Tausendundeine Nacht“ hat diese Seite nichts zu tun, sie ist ein Portal für weltumspannenden und allumfassenden Handel. Gehandelt wird mit allem und noch viel mehr, Automobilen und Reis, Chemikalien und Golfschläger sind nur einige willkürlich herausgegriffene Handelsgüter. Selbstredend gehören Elektronik-Komponenten aus dem fernen China auch dazu.

Alibaba knüpft Kontakte zu Handelspartnern in Echtzeit. Der potentielle Kunde kann mit qualifizierten Mitarbeitern des Herstellers chatten, die fernen Standorte mögen am anderen Ende der Welt liegen. Fragen zu Produkten, Preisen und Liefermodalitäten werden meistens sofort beantwortet. Bei solchen Direktkontakten in andere Regionen des Globus müssen natürlich Zeitunterschiede und nationale Feiertage bedacht werden.

Über Alibaba habe ich mit einem chinesischen Unternehmen eine Absprache zur Lieferung eines speziellen Bauteils für ein Elektor-Projekt getroffen. Weil das Projekt gegenwärtig erst in der frühen Entwicklungsphase steht, möchte ich an dieser Stelle noch nichts verraten. Vorausgesetzt, dass das chinesische Bauteil im Projekt seine Bewährungsprobe besteht, ist durch Alibabas Dienste für die weltweite Beschaffbarkeit gesorgt. Mich hat Alibaba bis jetzt nicht enttäuscht.

Natürlich schrumpft die Welt nicht wirklich, doch durch die moderne Kommunikationstechnik ist sie in unserem Denken und Handeln zusammengerückt. Für Elektronik-Entwickler sind spannende Zeiten angebrochen.

(110549)gd



Leckere Pizzas

Von Thijs Beckers und
Jan Visser (Redaktion NL / Elektor-Labor)

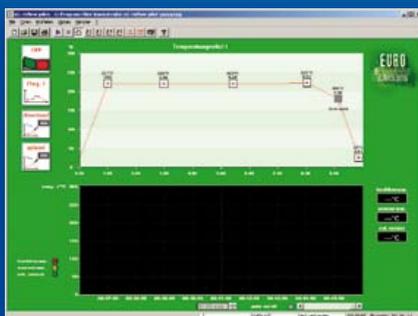
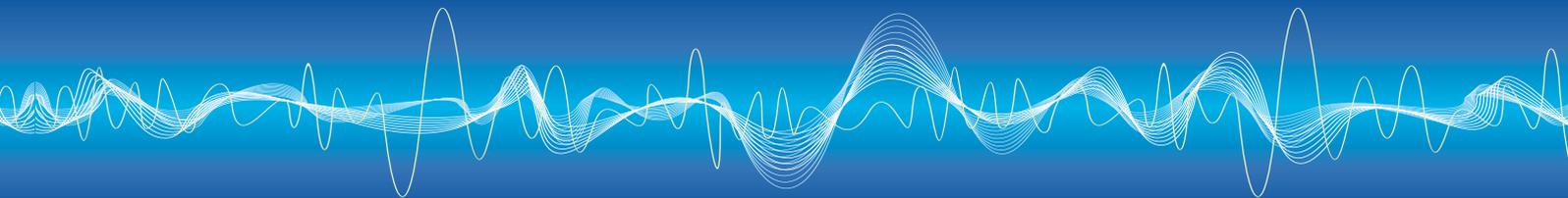
Nein, wir haben nicht spontan unsere Interessen verlagert und sind nicht auf Backrezepte klassischer italienischer Spezialitäten umgestiegen. Hier geht es auch nicht um einen Urlaubsreisebericht, der die Qualitäten der italienischen Gastronomie würdigt. Es geht um den kürzlich angeschafften High-Tech-SMD-Lötofen des Elektor-Labors.

An anderer Stelle konnten Sie bereits Näheres über die Eigenschaften unseres neuen Backwunders erfahren. Dort ist jedoch nichts über die Probeläufe zu lesen, in denen die High-Tech-Errungenschaft von unserem Kollegen Jan Visser auf Herz und

Nieren getestet wurde. Eine Zweckentfremdung der besonderen Art, bei der auch das leibliche Wohl nicht zu kurz kam, war das Aufbacken tiefgekühlter Pizzas. Kollege Jan Visser brachte sie aus dem nahe gelegenen Supermarkt mit.

In ebenso intensiven wie umfangreichen Probe- und Testläufen fand Jan empirisch den für diese Pizzas optimalen Backprozess-Verlauf heraus. Nun konnten die belegten runden Teigscheiben nicht mehr verbrennen, steinharte Ränder, verdorrte Champignon-Fragmente und angekohlte Käse-Streusel gehörten der Vergangenheit an. Während dieser Testreihen breitete sich dort ein Appetit anregender Duft aus, wo es sonst nur nach heißem Lötzinn und verdampftem Flussmittel roch.

„Damit die Pizza möglichst rasch und gleichmäßig aufgeba-



cken wird," sagte Jan, „muss der Ofen vorgeheizt werden. Dazu ist Zeit, wenn wir die Pizza dem Tiefkühlschrank entnehmen und auspacken. Der Ofen durchläuft eine sogenannte ‚kurze Temperaturkurve‘. Die progressiv gewundenen Heizspiralen, die von den Ofen-Konstrukteuren strategisch optimal platziert wurden, sorgen für eine homogene Temperaturverteilung über Fläche und Raum. Den Rost mit der Pizza müssen wir auf halber Höhe einschieben, damit die von oben und unten strahlende Wärme gleiche Effizienz hat. Die Temperatursensoren, die an der Decke und im Boden des Innenraums eingelassen sind, dürfen miteinander keinen ‚Sichtkontakt‘ haben. Darauf ist insbesondere zu achten, wenn wir im Ofen die beliebten Mini-Pizzas aufbacken.

Da die Backzeit acht Minuten bei 220 °C beträgt, programmieren wir den Ofen so, dass diese Temperatur nach einer Minute erreicht ist und anschließend acht Minuten konstant gehalten wird. Mit dem beweglichen Temperatursensor können wir die Temperatur des Pizza-Bodenrands überwachen. Das trägt dazu bei, dass der Boden nach dem Backen weder weiche noch harte, sondern knusprige Konsistenz hat.

Beim Einkauf müssen wir darauf achten, dass die Pizza flach und die Backzeit kurz ist. Backzeiten von mehr als acht Minu-

ten machen natürlich andere Ofen-Programmierungen notwendig, und eine Pizza, die höher als 2,5 cm ist, kann mit dem oberen Temperatursensor in Berührung kommen, was das perfekte Ergebnis in Frage stellt.

Der vordere Temperaturkurven-Anstieg muss nicht extrem steil sein, die tiefgekühlte Pizza braucht sowieso etwas Zeit, bis sie der Kurve folgen kann. Die zum Ofen gehörenden Platinen-Halterungen haben beim Pizza-Backen keine Funktion. Ein Stück Backpapier, das unter die Pizza gelegt wird, kann tropfenden heißen Käse auffangen.

Nach neun erwartungsvollen Minuten ist die Pizza gebacken, wie von magischer Hand öffnet sich die Ofentür. Die nun folgende Abkühlphase, in die der Ofen beim Platinenlöten eintritt, ist für die Pizza-Bäckerei natürlich nicht zweckdienlich. Die Pizza muss sofort aus dem Ofen genommen werden, damit sie heiß und knusprig an die hungrigen Kollegen verteilt werden kann.“ Vielleicht sind Sie auf den (Pizza-)Geschmack gekommen?

Die Pizzaback-Temperatur-Verlaufskurve des Kollegen Jan Visser können Sie unter www.elektor.de/110537 von unserer Website herunterladen. Wir wünschen guten Appetit!

(110537)gd

Wetterprobleme

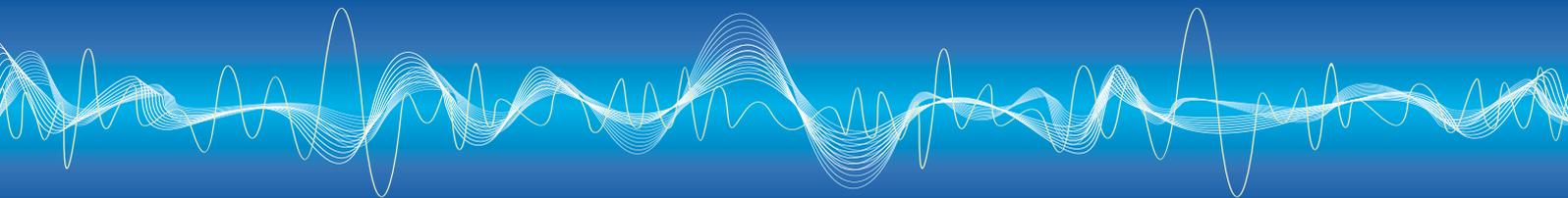
Von Luc Lemmens & Thijs Beckers (Elektor-Labor)

Beim Test der Prototypen-Platine des USB-Wetterloggers stießen wir auf Eigenartiges: Bei der relativen Luftfeuchtigkeit wird eine vom Sensor gelieferte Frequenz ausgewertet, um die relative Luftfeuchte zu berechnen und anzuzeigen. Unser Labormuster erzeugte aber in feuchten Umgebungen ein Signal für mehr als 150% relative Luftfeuchte! Auch wenn es im Keller der Wasserburg, in dem sich unser Labor befindet, schon recht feucht ist, sind 150% relative Luftfeuchtigkeit sowohl praktisch als auch theoretisch natürlich nicht möglich. Besonders befremdlich war, dass der Prototyp anfangs recht gut arbeitete (es zeigte sich kein Wert über 100%). Es musste also etwas sehr Spezielles der Fall sein.

Den Musteraufbau des Autors hatten wir leider schon zurückgeschickt. Ein Vergleich wäre interessant gewesen, denn dieses schon vor längerer Zeit gebaute Gerät zeigte eher zu niedrige Werte. Der Autor wollte dies nochmals prüfen. Noch ein zweites



Muster aufzubauen oder das Autorenmuster nochmals anzufordern war angesichts der Terminlage eigentlich keine Option mehr, und der Fehler müsste mit systematischer Fehlersuche auch so zu finden sein...



Also wurden die in der Software durchgeführten Berechnungen ein weiteres Mal gecheckt und die Platine zum wiederholten Mal auf mögliche Fehler bei der Bestückung untersucht. Doch all die Mühe brachte nichts. Das Einzige, was Luc noch einfiel, waren die Werte zur Kalibrierung, die im EEPROM des Sensors abgelegt waren. Das aber ist als Fehlerquelle sehr unwahrscheinlich, da ein zweiter angeschlossener Sensor genau den gleichen Fehler zeigte. Also doch eher ein systematischer Fehler, wie eine fal-

sche Adresse beim Auslesen der Korrekturwerte oder Ähnliches. Als diese Elektor-Ausgabe in Druck ging, war das Problem noch nicht gelöst. Doch wir vertrauen auf unsere Erfahrung mit solchen Phänomenen und sind uns ziemlich sicher, dass alles hundertprozentig funktioniert, wenn sie diesen Artikel lesen und die Software von der Elektor-Webseite downloaden können.

(110383)

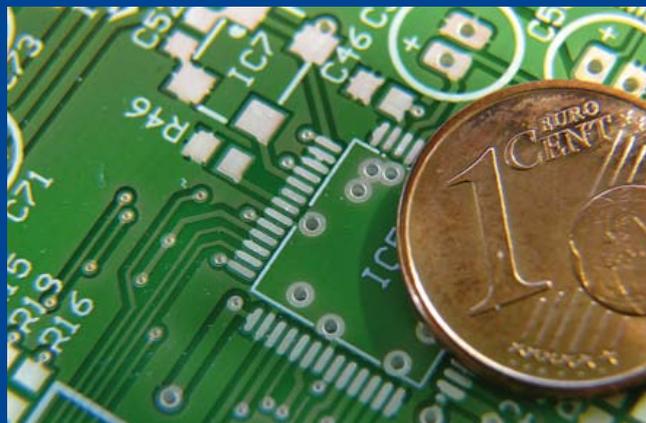
Kleine Fallgruben

Von Thijs Beckers und Ton Giesberts (Elektor-Labor)

Vermutlich hat auch die aktuelle Fortsetzung unserer Artikelreihe zum Thema DSP Ihr Interesse gefunden. Wir haben darin alles Wichtige beschrieben - was natürlich bei weitem nicht alles ist, was man beschreiben könnte. Für vieles, was uns dabei beschäftigt hat, fehlt in den Artikeln der Platz. Einiges wollen wir Ihnen trotzdem nicht vorenthalten.

Als wir vor einer Reihe von Jahren den Klasse-D-Verstärker „Clarity“ entwickelten, konnten wir auf nur wenige Erfahrungen im Layouten SMD-bestückter Platinen zurückgreifen. Da ein Klasse-D-Verstärker mit hohen Strömen und hohen Schaltfrequenzen arbeitet, erschien uns eine möglichst kompakte Gestaltung der Platine geboten. Um den Platz frei liegender Durchkontaktierungen (VIAs) einzusparen, legen wir sie dort, wo es möglich war, mit den Pads passiver Bauelemente zusammen. Vor Anlauf der Platinen-Produktion wies uns unser Zulieferer darauf hin, dass VIAs in Pads Ursache von Lötproblemen sein können. Es bestehe die Gefahr, dass sich die Komponenten während des Lötprozesses aufrichten. Dieses Phänomen werde auch „Tombstoning“ genannt, und schuld sei die Kapillarwirkung des Kontaktlochs. Die seinerzeit produzierten Platinen blieben glücklicherweise von dem gefürchteten Effekt verschont, doch seitdem sind wir diesem Risiko durch Umdenken bei der Platinen-Gestaltung aus dem Weg gegangen.

Außer dem „Tombstoning“ mussten bei der DSP-Platine noch andere Fallstricke und Gruben umgangen werden. Der LED-Treiber TLC5926 ist ein Chip im HTSSOP-Gehäuse, der über einen „Exposed Thermal Pad“ an der Gehäuse-Unterseite Wärme abführt. Da dieser Pad gleichzeitig der allgemeine Masse-Anschluss des Schaltkreises ist, wird seine Unterseite an die Massefläche der Platine gelötet. Hier besteht die Gefahr, dass der Chip buchstäblich weg gleitet, sobald das Lötmedium während des Lötprozesses flüssig geworden ist. Diese Gefahr ist umso größer, je mehr Lötpaste an der Berührfläche aufgetragen wird. Andererseits darf mit der Lötpaste nicht so sparsam umgegangen werden, dass sich der Wärmeübergang zwischen Chip und Platinenfläche verschlechtert. Bewährt hat sich das Auftragen der Lötpaste mit Hilfe einer Schablone (Englisch: stencil). Eine dünne Folie mit Perforationen an den Orten, an



denen Lötpaste aufzutragen ist, leistet gute Dienste. Nach Füllen der Perforationen mit Lötpaste wird überschüssiges Volumen einfach mit einem Rakel abgestreift.

Manchmal haben Durchkontaktierungen noch andere Tücken: Auch Kupferflächen, die wie beim TLC5926 unserer DSP-Platine der Wärmeableitung dienen, werden häufig mit VIAs versehen. Diese Maßnahme verbessert die Wärmeabfuhr, denn über die Durchkontaktierungen wird die Wärme auch auf die andere Platinnenseite geleitet. Hier besteht die Gefahr, dass beim Lötten ein Teil des Lötmittels durch die VIA zur anderen Platinenseite abfließt. Im schlimmsten Fall bleibt zu wenig übrig, um die Gehäuse-Lötfläche sicher mit dem Pad auf der Platine zu verbinden.

Das Abfließen lässt sich verhindern, indem man solche VIAs durch die Löstopfmaske mit Lötstopplack umgibt. Möglichst kleine Durchmesser der Durchkontaktierung haben auf das Abfließen des Lötmittels ebenfalls eine hemmende Wirkung. Apropos Durchmesser der Durchkontaktierungen: Unser Platinen-Zulieferer gibt vor, dass die Bohrungen für die VIAs nicht kleiner als 0,25 mm sein dürfen. Der kupferne Rand muss mindestens 0,15 mm breit sein, so dass der Gesamtdurchmesser einer VIA etwas größer als 0,5 mm ist. Verglichen mit den Bohrungen klassischer „Through-Hole“-Bauelemente haben die Durchkontaktierungen wirklich winzige Abmessungen, das Foto führt den Unterschied anschaulich vor Augen...

(110551)gd

Abonnieren Sie jetzt die führende US-amerikanische Fachzeitschrift für Mikrocontroller-Anwendungen und Embedded Systems!



Wählen Sie unter

www.elektor.de/cc-abo

Ihr gewünschtes
Abonnement aus!

12 Original-Ausgaben jährlich für nur

Digital: 38 US-Dollar

Print: 63 US-Dollar

Digital + Print: 90 US-Dollar

**CIRCUIT
CELLAR®**
THE MAGAZINE FOR COMPUTER APPLICATIONS

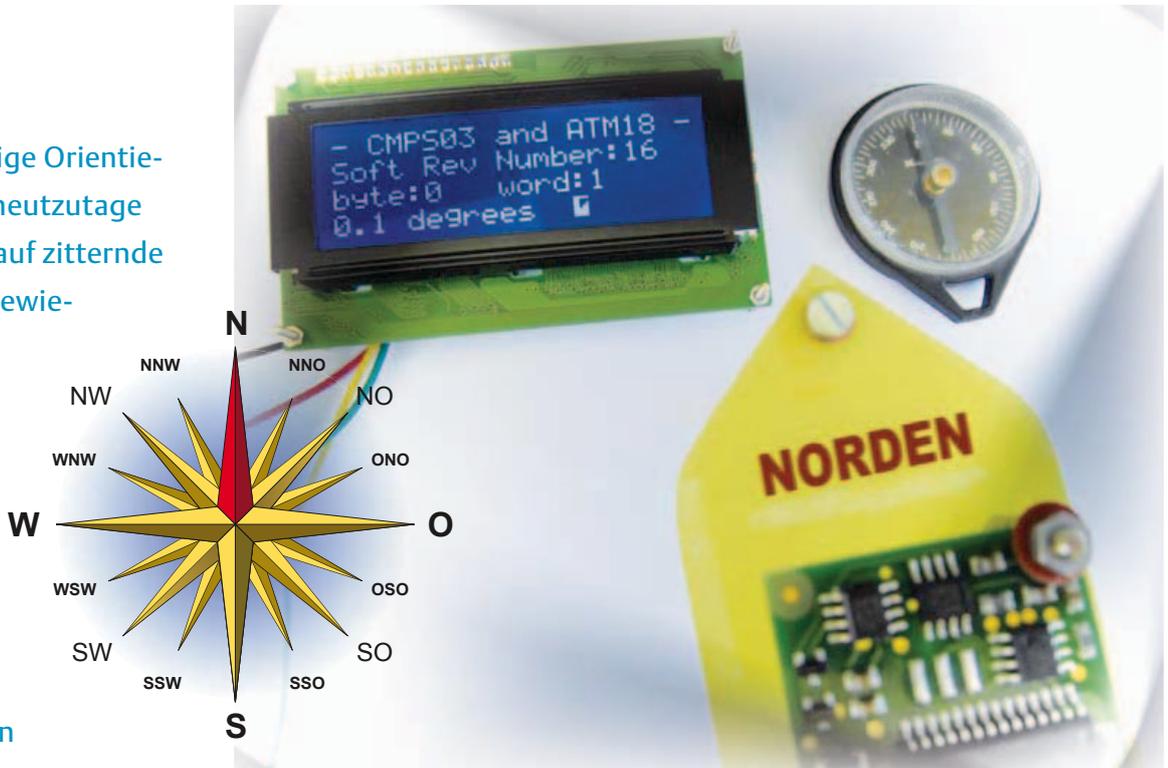
ATM18-Kompass

Immer auf Kurs

Von Grégory Ester (F)

Wer eine zuverlässige Orientierung benötigt, ist heutzutage nicht mehr länger auf zitternde Magnetnadeln angewiesen. Mit Hilfe von Magnetfeldsensoren lässt sich ein Kompass auch durch eine elektronische Schaltung realisieren. Dafür benötigt man natürlich einen Mikrocontroller. Womit

wir wieder einmal eine interessante Anwendung des ATM18-Mikrocontrollerboards vorstellen können.



Das Kompassmodul CMP503 [2] misst den Winkel zwischen zwei Linien: Der Linie zwischen Nord- und Südpol des Erd-Magnetfelds und der Linie (Zeigerichtung) des Kompassmoduls (**Bild 1**). Es verwendet dabei zwei Sensoren, um das Magnetfeld der Erde zu messen. Die Erfassung der von diesen Sensoren gelieferten Daten und die Berechnung des Winkels erfolgen durch einen auf der Platine integrierten Mikrocontroller. Eine Bus-Verbindung ist ebenfalls implementiert, man kann also direkt auf den berechneten Wert dieses Winkels über I²C zugreifen, sei es als Byte (0 bis 255) oder als 16-bit-Wort (0 bis 3599). Im letzteren Fall liefert eine einfache Division durch 10 den direkt lesbaren Winkelwert von 0 bis 359,9 Grad (**Bild 2**).

Der ATM18 [3] übernimmt die Kommunikation mit dem CMP503-Modul. Ein Zweidraht-LCD [4] zeigt die empfangenen Informationen an.

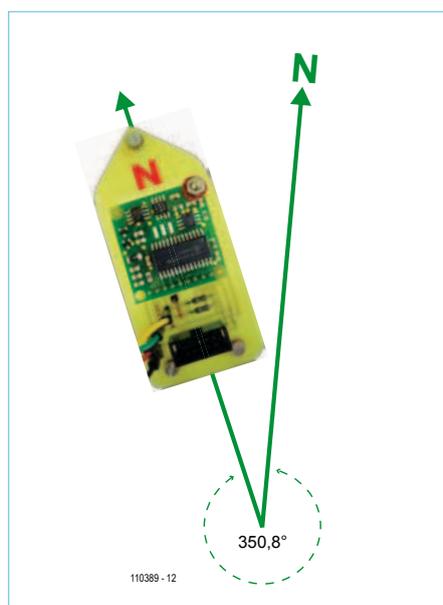


Bild 1. Eine richtungweisende Anwendung...

Das Kompass-Modul CMP503

Das Modul verwendet zwei magnetoresistive Sensoren KMZ51 von Philips, die empfindlich genug sind, um das Erdmagnetfeld zu messen. Magnetoresistive Sensoren sind im Prinzip nichts anderes als Widerstände, deren Wert durch ein Magnetfeld verändert wird, und zwar in Abhängigkeit von der Stärke und der Richtung des Magnetfelds. Bei konstanter Stärke des Magnetfelds ist die Änderung primär von der Richtung abhängig. Dreht man den Sensor im Magnetfeld, erhält man allerdings in jedem Halbkreis der Drehung den gleichen Verlauf.

Für einen Kompass benötigt man daher zwei Sensoren, die im rechten Winkel zueinander angeordnet sind. So ist es möglich, die Veränderungen in Nord-Süd- und in Ost-West-Richtung aufzunehmen. Jeder Sensor enthält eine Wheatstone-Brücke (**Bild 3**), um den Wert des sich ändernden Widerstandes präzise zu messen. Die Änderung des

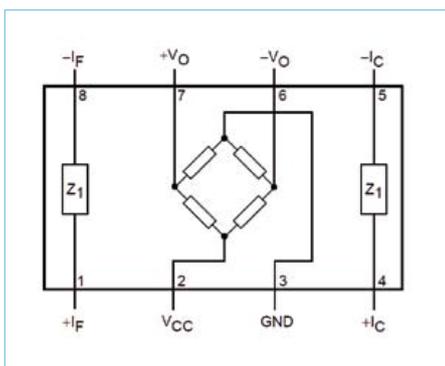


Bild 3. Innenschaltung des KMZ51.

Widerstandes führt zu einer Änderung der Brückenspannung, die von der Schaltung des CMPS03 mit einem Opamp (LMC6032) verstärkt und über den A/D-Wandler des Mikrocontrollers (ANO und AN1 des PICs) gemessen wird. Schließlich wird die magnetische Ausrichtung berechnet und ein Mikrocontroller (PIC18F2321) gibt das Resultat über einen I²C-kompatiblen Bus aus – man muss es nur noch auslesen.

Für eine korrekte Messung muss das Modul waagrecht zur Erdoberfläche ausgerichtet sein.

Das Modul verfügt über eine integrierte Kompensationsschaltung, um Störungen durch das Wechselstromnetz zu reduzieren. Diese Schaltung lässt sich über Pin 7 des Moduls an die jeweilige Netzfrequenz anpassen. Bei einem 50-Hz-Netz legt man Pin 7 an Masse und bei einem 60 Hz-Netz an +5 V. Im Lieferzustand ist das Modul auf eine Inklination von 67° kalibriert (am Herstellungsort in England). In Deutschland liegt die Inklination zwischen 63° im Süden und 70° im Norden, so dass man die Kalibration des Herstellers beibehalten kann. Sofern Sie den Kompass viel weiter nördlich oder südlich verwenden, müssen Sie nochmals kalibrieren. Dieser Schritt muss nur ein einziges Mal durchgeführt werden, denn die Parameter werden im EEPROM gespeichert. Die technischen Unterlagen des Moduls auf der Website des Herstellers [5] vertiefen dieses Thema.

Richtungweisend

Zur Realisierung des Projekts sind zuerst die einzelnen Baugruppen wie in **Bild 4** gezeigt zu verdrahten. Als nächstes laden Sie das Programm [1] in den ATM18-Controller, legen die Versorgungsspannung an, halten das Kompassmodul waagrecht und suchen den Norden. Das geht ganz einfach: Drehen Sie sich um sich selbst – mit der Schaltung in



Bild 2. Das Display zeigt das Messergebnis und Systeminformationen.

der Hand. In dem Moment, in das Kompassmodul genau nach Norden zeigt, ertönt ein akustisches Signal, das Cmps03_bearing_byte sowie das Cmps03_bearing_word nehmen den Wert Null an. Diese beiden Funktionen aus **Listing 1** greifen auf die 8-

und 16-bit-Ergebnisse zu, sie wurden nach Maßgabe des vom Hersteller vorgegebenen Timings geschrieben (**Bild 5**).

Falls man nun zum Beispiel die beiden vom CMPS03-Modul berechneten Ergebnisbytes auslesen möchte, schickt die Funktion

Listing 1. Auslesen des Registerinhalts

```
Function Cmps03_soft_revision() As Byte
    I2cstart
    I2cwbyte Cmps03_addr_write
    I2cwbyte 0
    I2crepstart
    I2cwbyte Cmps03_addr_read
    I2crbyte Cmps03_soft_revision , Nack
    I2cstop
End Function

Function Cmps03_bearing_byte() As Byte
    I2cstart
    I2cwbyte Cmps03_addr_write
    I2cwbyte 1
    I2crepstart
    I2cwbyte Cmps03_addr_read
    I2crbyte Cmps03_bearing_byte , Nack
    I2cstop
End Function

Function Cmps03_bearing_word() As Word
    Local Hi_byte As Byte
    Local Lo_byte As Byte
    I2cstart
    I2cwbyte Cmps03_addr_write
    I2cwbyte 2
    I2crepstart
    I2cwbyte Cmps03_addr_read
    I2crbyte Hi_byte , Ack
    I2crbyte Lo_byte , Nack
    I2cstop
    Cmps03_bearing_word = 256 * Hi_byte
    Cmps03_bearing_word = Cmps03_bearing_word + Lo_byte
End Function
```

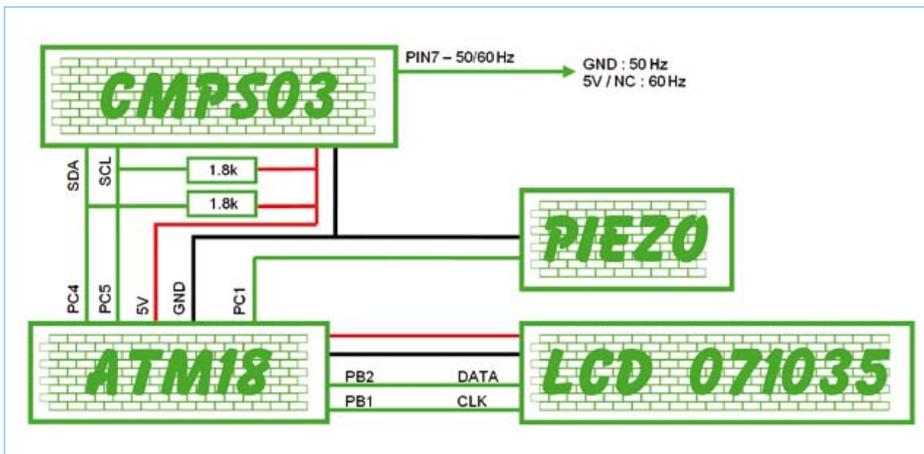


Bild 4. Der Verdrahtungsplan des Projekts.

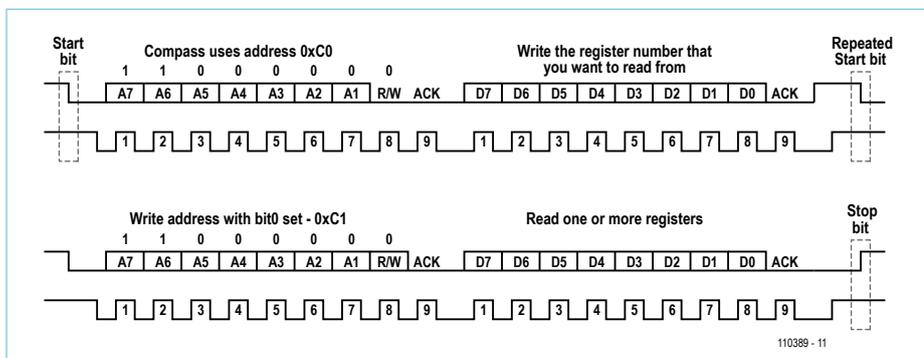


Bild 5. Impulsdigramm der Kommunikation mit dem CMPS03-Modul.

cmps03_bearing_word zunächst ein Startbit, gefolgt von der Schreibadresse (\$C0) des Moduls und der Adresse (Tabelle 1) des Registers, welches man auslesen möchte (hier 2); danach wird ein weiteres Startbit erzeugt, diesmal gefolgt von der Leseadresse (\$C1) des Moduls. Daraufhin wird das höchstwertige Byte gelesen und der Empfang bestätigt (ACK), schließlich liest der Befehl I2crbyte Lo_byte, Nack das niederwertige Byte ein und legt es in der Variablen Lo_Byte ab. Wie stets beim letzten gelesenen Wert wird kein ACK gesandt.

Graphische Darstellung

Um die Navigation zu vereinfachen, sind die 360° in acht Segmente aufgeteilt, und die

Himmelsrichtungen werden graphisch dargestellt. Bild 6 und Bild 7 zeigen, wie man nach Nordnordost beziehungsweise Süd-südwest weist.

Die Flüssigkristallanzeige ist mit einem HD44780-Controller ausgestattet, welcher die Zeichenerzeugung und die Steuerung des LCDs übernimmt. Für den Anwender vereinfacht das den Programmcode. Es genügt, dem Controller die Befehle, die anzuzeigenden Zeichen und die Angabe, wo sie dargestellt werden sollen, zu senden. Die Sonderzeichen aus Bild 6 und 7 gibt es im CGROM des LCD-Controllers nicht. Man muss sie deswegen selbst erzeugen und im CGRAM (Character Generator RAM) ablegen.

Tabelle 1. Die von der Firmware ausgewerteten Register des CMPS03	
Register	Funktion
0	Version des Programms im PIC18F2321 des CMPS03
1	Winkel als Byte : 0 bis 255 für einen ganzen Kreis
2, 3	Winkel als Word (zwei Bytes) : 0 bis 3599 für einen ganzen Kreis. Dies entspricht Werten von 0 bis 359,9 Grad in Klarschrift.

In diesem Speicher beginnt der Bereich für benutzerdefinierte Zeichen bei Adresse \$40 (mit Auto-Increment) und reicht über acht Byte pro Zeichen. Acht eigene Zeichen können auf diese Weise gespeichert werden. Um sie anzuzeigen, sendet man dem LCD die ASCII-Codes von 0 bis 7.

Ein alphanumerisches Zeichen im 5x7-Raster besteht aus sieben Zeilen zu je fünf Punkten. Es gibt eine achte Zeile, in der üblicherweise der Cursor angezeigt wird. Diese wird nicht verändert.

BASCOM-AVR enthält ein Tool namens LCD-Designer (Bild 6 und 7), welches beim Entwurf selbstdefinierter Zeichen hilft. Da das Kompass-Projekt das Zweidraht-LCD verwendet, wird die von diesem Tool automatisch generierte Kommandozeile Deflcdchar [0],31,24,25,27,31,31,31,32 auskommentiert, sie zeigt uns aber dennoch, welche acht Bytewerte diesem Zeichen entsprechen.

Um zum Beispiel das in Bild 6 abgebildete Zeichen zu erstellen, muss man acht Bytes senden, wobei jedes für eine Reihe von Punkten steht. Die drei höchstwertigen Bits werden ignoriert. Ein Pixel ist sichtbar, wenn das entsprechende Bit auf 1 gesetzt ist. Listing 2 zeigt einen Auszug aus der Prozedur, die frei erstellte Zeichen im CGRAM ablegt.

Listing 3 ist ein Teil derjenigen Prozedur, die den Cursor auf die Position (X_lcd, Y_lcd) setzt, an der das Sonderzeichen angezeigt werden soll. Man ruft diese Prozedur wie folgt auf: Call Pointing(„NNE“ , 14 , 4).

Es ist ein ganz natürliches physikalisches Phänomen, das diese Anwendung erst ermöglicht hat: Das Erdmagnetfeld, in dem wir uns ständig aufhalten, und das uns nicht nur vor tödlicher Teilchenstrahlung bewahrt, sondern auch vor Orientierungslosigkeit. Der vorgestellte Sensor könnte zum Beispiel auch in ein Roboterfahrzeug eingebaut werden. In jedem Fall sollte man das Modul sorgfältig von allen elektromagnetischen Störquellen fernhalten, um die korrekte Funktion der Kompass Elektronik nicht zu stören.

(110389)

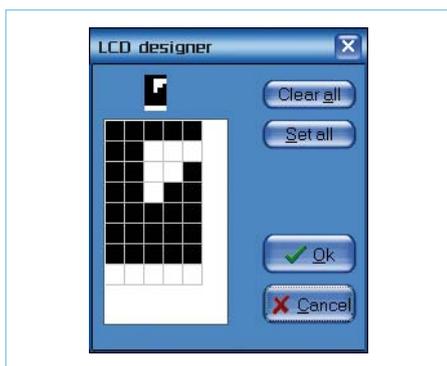


Bild 6. Mit dem BASCOM-Tool LCD-Designer...

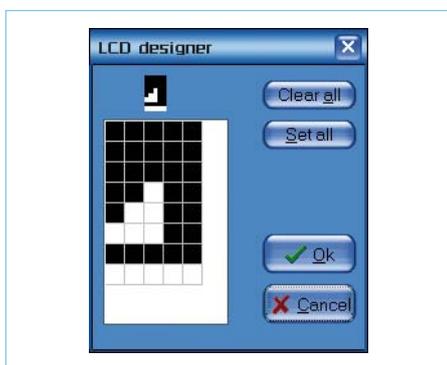


Bild 7. ...kann man eigene Zeichen zusammenstellen.

Weblinks

- [1] www.elektor.de/110389
- [2] www.lipoly.de oder www.robotikhardware.de
- [3] www.elektor.de/atm18
- [4] www.elektor.de/071148
- [5] www.robot-electronics.co.uk/htm/cmps3tech.htm

Listing 2. Aufbau selbstdefinierter Zeichen

```

`ADRESSES CG-RAM A PROGRAMMER
`CHAR1 : $40 à $47
`CHAR2 : $48 à $4F
`CHAR3 : $50 à $57
`...
Sub Lcd_custom_char()
  Rs = 0
  Waitms 20
  Lcd_write_byte &H40      `Basisadresse des CGRAM
  Rs = 1                  `Senden von Daten
  Waitms 20
  `-----CHAR1_NNE
  `Deflcdchar [0],31,24,25,27,31,31,31,32
  Lcd_write_byte 31      `$40
  Lcd_write_byte 24      `$41
  Lcd_write_byte 25      `.....
  Lcd_write_byte 27
  Lcd_write_byte 31
  Lcd_write_byte 31
  Lcd_write_byte 31
  Lcd_write_byte 32
  `-----CHAR2_ENE
  Lcd_write_byte 31      `$48
  Lcd_write_byte 30      `$49
  Lcd_write_byte 28      `.....
  Lcd_write_byte 24
  Lcd_write_byte 31
  Lcd_write_byte 31
  Lcd_write_byte 31
  Lcd_write_byte 32
  `-----CHAR3_ESE
  Lcd_write_byte 31      `$50
  ..
  ..
  ..
End Sub

```

Listing 3. Anzeigen selbstdefinierter Zeichen

```

Sub Pointing(byval Direction As String , Byval X_lcd As Byte , Byval Y_lcd As Byte)
  Lcd_pos X_lcd , Y_lcd
  Select Case Direction
    Case « NNE »
      Rs = 1      ` Senden von Daten
      Waitms 20
      Lcd_write_byte &H00
    Case « ENE »
      Rs = 1      ` Senden von Daten
      Waitms 20
      Lcd_write_byte &H01
  ..
  ..
End Select
End Sub

```

2,4-GHz-Fernsteuerung für Modellflugzeuge

ZigBee-Technik im Modellbau eingesetzt

Von Michel Kuenemann (F)

Schon seit einigen Jahren gibt es Funkfernsteuerungen für Flugmodelle, die im 2,4-GHz-Band operieren. In diesem Frequenzbereich ist die Störsicherheit groß. Er bietet vor allem neue Möglichkeiten für die Telemetrie, die bei den schmalbandigeren Sendern mit 35, 40 oder 41 MHz (je nach Land) nicht denkbar waren. Das hier vorgestellte Projekt erlaubt sogar

die nachträgliche Aufrüstung von älteren 35/40/41-MHz-Systemen auf zeitgemäße 2,4 GHz. Außerdem kann es leicht nach eigenen Vorstellungen modifiziert werden, da es ein offenes Projekt ist.



Bild 1 zeigt die Blockschaltung des Systems. Sender für Funkfernsteuerungen haben normalerweise eine Steckverbindung, an dem ein standardisiertes PPM-Signal (Pulse Position Modulation) anliegt, das die Informationen der Kanäle enthält, die durch den Sender übertragen werden. Diese Steckverbindung dient auch häufig zum Training neuer Piloten oder zur Verbindung mit einem PC mit Flugsimulator-Software. In unserem Fall wird das Signal zu einem extra Gehäuse durchgeschleift, in dem sich ein 2,4-GHz-Sender befindet. Außer dem Sender befinden sich auch noch ein Akku und ein LCD im Gehäuse, sodass es sich dabei um ein recht autonomes Subsystem handelt. Durch diese Anordnung kann der Pilot mit seiner gewohnten Fernsteuerung problemlos ältere Modelle im 35/40/41-MHz-Band fliegen und trotzdem schnell und einfach auf 2,4 GHz umschalten.

Der in das Modellflugzeug eingebaute Empfänger besteht aus einer Platine, die mit einem oder zwei Empfänger-Akkus und den Servos verbunden ist. Ein wesentlicher Vorteil der 2,4-GHz-Technik ist, dass die Antenne sehr kurz ist. Damit entfallen die Probleme, die es bei Flugmodellen häufig mit den (zu) langen Drahtantennen bei den niedrigen Frequenzen gibt.

Bei diesem Projekt handelt es sich um ein offenes, experimentelles System, das Kommunikations-Schnittstellen wie eine serielle Verbindung (UART), einen CAN-Bus und einen I²C-Bus mitbringt.

Technik

Die Entwicklung und der Bau eines gut funktionierenden Transceiver-Moduls für 2,4 GHz dürfte nicht wenige Elektroniker schlicht überfordern. Glücklicherweise gibt es mittlerweile viele Fertigmodule für dieses Band, wodurch die 2,4-GHz-Technik

einfacher zu realisieren ist, als es den Anschein hat. Der Autor entschied sich für das Modul MRF24J40MB [2] von Microchip. Dieses Standard-Modul ist kompakt, hat genug Leistung und ist außerdem preiswert. Diese Eigenschaften machen es zum idealen Kandidaten für dieses Projekt. Ein mit 24 MHz getakteter Mikrocontroller vom Typ PIC18LF2685 sorgt für die problemlose Ansteuerung verschiedenster Peripherie. Um die praktische Anwendung noch weiter zu vereinfachen, kann eine einzige Platine dank der Software sowohl als Sender als auch als Empfänger konfiguriert werden. Ein gesteckter Jumper macht die Einheit zum Sender – ohne diesen wird sie zum Empfänger.

Schaltung

Die Platine enthält alle Betriebsspannungen, die für den Betrieb als Sender oder Empfänger notwendig sind (Bild 2).

Anmerkungen und Warnungen

Das System wurde vom Autor mehrere Monate mit Erfolg in verschiedenen Modellflugzeugen getestet, die sowohl mit Verbrennungsmotor als auch mit Elektromotor ausgestattet waren. Die Reichweite im freien Feld war größer als 1 km. Es zeigten sich keine Interferenzen mit anderen Fernsteuerungen, die mit 41 MHz oder 2,4 GHz arbeiteten. Dennoch handelt es sich um ein Projekt mit experimentellem Charakter, bei dem jeder Anwender selbst die Verantwortung für Gebrauch und Einsatzart hat.

Das HF-Modul MRF24J40MB von Microchip ist offiziell für Europa (ETSI), die Vereinigten Staaten von Amerika (FCC) und Canada (IC)

geeignet. Die Ausgangsleistung beträgt 100 mW. Die Kanäle 20 bis 16, bei denen in Frankreich nur 10 mW zugelassen sind, werden nicht verwendet.

Zur Stromversorgung werden LiPo-Akkus empfohlen. Diese können prinzipiell explodieren oder auch Brände verursachen, wenn zu große Kräfte einwirken oder zu hohe Ströme fließen. Wenn man keine LiPos einsetzen möchte, kann man den Sender auch gut mit drei NiMH-Zellen in Serie (Größe R6) versorgen. Für den Empfänger eignen sich Akku-Packs mit vier oder fünf Zellen, die für Modellbauzwecke konfektioniert sind.

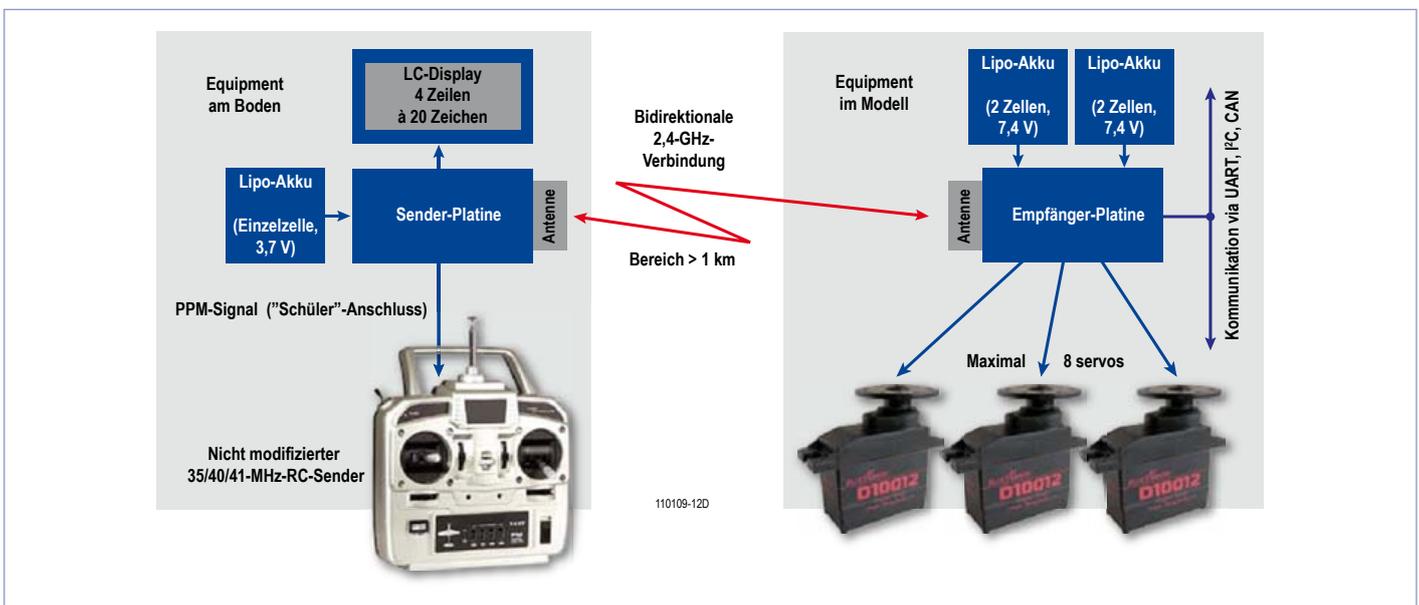


Bild 1. Aufbau des Systems.

Der Sender wird von einem extra LiPo-Akku (Lithium-Polymer) versorgt. Hierfür ist auch schon eine Ladeschaltung integriert, die den Akku mit einer Spannung von maximal 4,2 V und einem auf 200 mA begrenzten Strom lädt. Die konstante Ladespannung wird vom Spannungsregler LM317 (U4) erzeugt, wobei Transistor T3 den maximalen Strom über den Shunt R20 auf etwa 200 mA

begrenzt. Ein Akku mit einer Kapazität von 1000 mAh kann so in rund fünf Stunden geladen werden. Der Schalter für den Sender verbindet die Ladeschaltung mit dem Akku, wenn der Sender ausgeschaltet ist. An die Ladebuchse CN10 kann ein beliebiges Steckernetzteil angeschlossen werden, das minimal 9 V bei 0,2 A liefert. Übrigens reicht ein voller Akku mit 1 Ah für gut 15

Stunden Sendebetriebs.

Die Stromversorgung des Empfängers ist dank zweier Akkus redundant ausgeführt. Auch wenn während des Flugs ein Akku versagt, steht noch ein zweiter zur Verfügung und der Empfang wird nicht unterbrochen. Die „Umschaltung“ zwischen den beiden Akkus geschieht der Einfachheit und Zuverlässigkeit halber mit zwei Dio-

Technische Eigenschaften

- Senden mit acht proportionalen Kanälen
- PPM-Modulation, kompatibel mit Sendern für 35/40/41 MHz
- Bestehende Fernsteuerung muss nicht ausgewechselt werden
- Empfänger mit doppelter Versorgung und lineare Regler für ein oder zwei LiPo-Akkus
- Fernmessung der Spannung der Empfänger-Akkus
- Empfänger kompatibel mit BEC-Versorgung
- Fernmessung des Empfänger-Stromverbrauchs und Anzeige der verbrauchten Energie in mAh
- Fernmessung der Empfänger-Signalstärke
- LCD im Sender für Parameter-Überwachung
- Akustischer Alarm beim Sender
- Kommunikation via UART, CAN, I²C
- ZigBee-Technologie
- Reichweite im freien Feld > 1 km (vom Autor getestet)
- Reaktionsverzögerung 20 ms



den. Das gleiche Redundanzprinzip wird übrigens auch in anderen fehlertoleranten Systemen wie zum Beispiel in der Luftfahrt eingesetzt. Eine hohe Zuverlässigkeit wird auch dadurch erreicht, dass während des Fluges die Spannung der Akkus überwacht wird. Ohne eine solche Kontrolle würde sich ein Defekt erst dann zeigen, wenn auch der zweite Akku ausfällt – mit den entsprechenden Folgen. Es wird niemand zum Einsatz von zwei Empfänger-Akkus gezwungen, aber angesichts der Kosten eines Flugmodells ist diese Lösung durchaus empfehlenswert.

Die Spannungen der beiden Akkus werden über zwei Spannungsteiler erfasst: R21/R22 ist für Akku A zuständig und R29/R31 für Akku B. Wie man im Schaltbild sehen kann, wird R21/R22 auch dazu verwendet, die Spannung des Sender-Akkus zu messen. Der lineare Spannungsregler U6_1 (MC33375) liefert die für den Betrieb des Mikrocontrollers und des 2,4-GHz-Moduls notwendige Spannung von 3,3 V. Die integrierte Ladungspumpe U5 (ICL7660) erzeugt aus den +3,3 V die negative Hilfsspannung für das LCD. Der lineare Spannungsregler U6 (LT1764A) liefert eine mit 3 A belastbare Spannung von 5...6 V zur

Versorgung der Servos. Zusätzlich wird der Strom überwacht: Der mit U7 (LTC6106) verstärkte Spannungsabfall am Shunt R23 (10 mΩ) ist ein Maß für den Servo-Strom. Der über F1 mit 3 A abgesicherte Anschluss V_{CAN} dient zur Versorgung eventueller Erweiterungen.

In Modellen mit elektrischem Antrieb wird die Energie für den Betrieb von Empfänger und Servos normalerweise über eine Schaltung namens BEC (Battery Eliminator Circuit) durch den elektronischen Motorregler geliefert. Diese Funktion übernimmt hier Diode D4 (BAT54J), da so aus der Betriebsspannung der Servos der Regler U6_1 gespeist wird.

An CN14 kann der Schalter für den Empfänger des Modellflugzeugs angeschlossen werden. Dieser Schalter funktioniert umgekehrt wie ein normaler Schalter: Ist er offen, wird der Empfänger versorgt – ist er geschlossen, ist der Empfänger abgeschaltet. So wird erreicht, dass Kontaktprobleme nicht zu einem Abschalten des Empfängers führen. Diese Art der Schalterfunktion ist deutlich zuverlässiger und sorgt so dafür, dass das Modell nicht unvermittelt und schwer nachvollziehbar den Funkkontakt verliert.

Ein Mikrocontroller vom Typ PIC18LF2685 verfügt über 96 KB an Flash-Speicher, 3 KB an RAM und ein integriertes CAN-Bus-Interface in einem 28-Pin-Gehäuse. Der Prozessortakt von 24 MHz wird von einem Quarz erzeugt. Die hohe und genaue Taktfrequenz sorgt einerseits für Arbeitstempo und andererseits für präzises Timing der Servo-Ansteuerung. Die Ports RA0 bis RA3 sind als analoge Eingänge konfiguriert und dienen zum Messen von Akkuspannungen, V_{CAN} und Servo-Strom.

Das IC U1 (MAX3054) ist ein fehlertoleranter CAN-Transceiver mit einer maximalen Datenübertragungsrate von 250 kbit/s. Dieses Bauteil wird ebenso wie die zugehörige Steckverbindung CN2 natürlich nur dann bestückt, wenn man tatsächlich einen CAN-Bus nutzen will. An CN2 liegt auch gleich die Betriebsspannung V_{CAN} für Erweiterungskarten an, die im Daisy-Chain-Prinzip als Kette angeschlossen werden können.

Über CN3 und CN4 lassen sich eigene Erweiterungsschaltungen anschließen. An CN5 wird der schon erwähnte Jumper zum Betrieb als Sender gesteckt. Ist er nicht gesteckt, arbeitet die Platine als Empfänger. Die Ports RC1 und RC2 bilden den I²C-Bus, der im Sendemodus das LCD und im Empfängermodus die Servos steuert. Diese beiden Pins liegen zusammen mit der 3,3-V-Betriebsspannung an den beiden Steckverbindern CN6 und J1. Dabei ist J1 kompatibel mit dem portablen Mini-Terminal, das als Teil des Projekts „Motoren-Prüfstand“ in Elektor April 2009 [3] veröffentlicht wurde.

Über die Pins RC6 und RC7 wird ein serielles Interface (UART) mit 3,3-V-Pegel realisiert. Die Signale liegen am Anschluss CN7.

Bei U8 (MCP23008) handelt es sich um ein 8 bit breites Port-Erweiterungs-IC für I²C. Im Empfängermodus können damit acht Servos angesteuert und mit den Steckverbindern ST1 bis ST8 angeschlossen werden. Im

Offenes Projekt

Wenn man kein Faible für den Modellbau hat, kann man die Elektronik auch sehr gut für andere Zwecke im Haus oder in der Robotik einsetzen, die auf ZigBee basieren oder einen CAN- oder I²C-Bus aufweisen. Der Programmspeicher des Mikrocontrollers bietet noch ge-

nug Platz für komplexe Algorithmen. Kreative Modellbauer können das Projekt auch als Basis für einen Autopiloten verwenden oder für exotische Flugzeuge wie Multicopter.

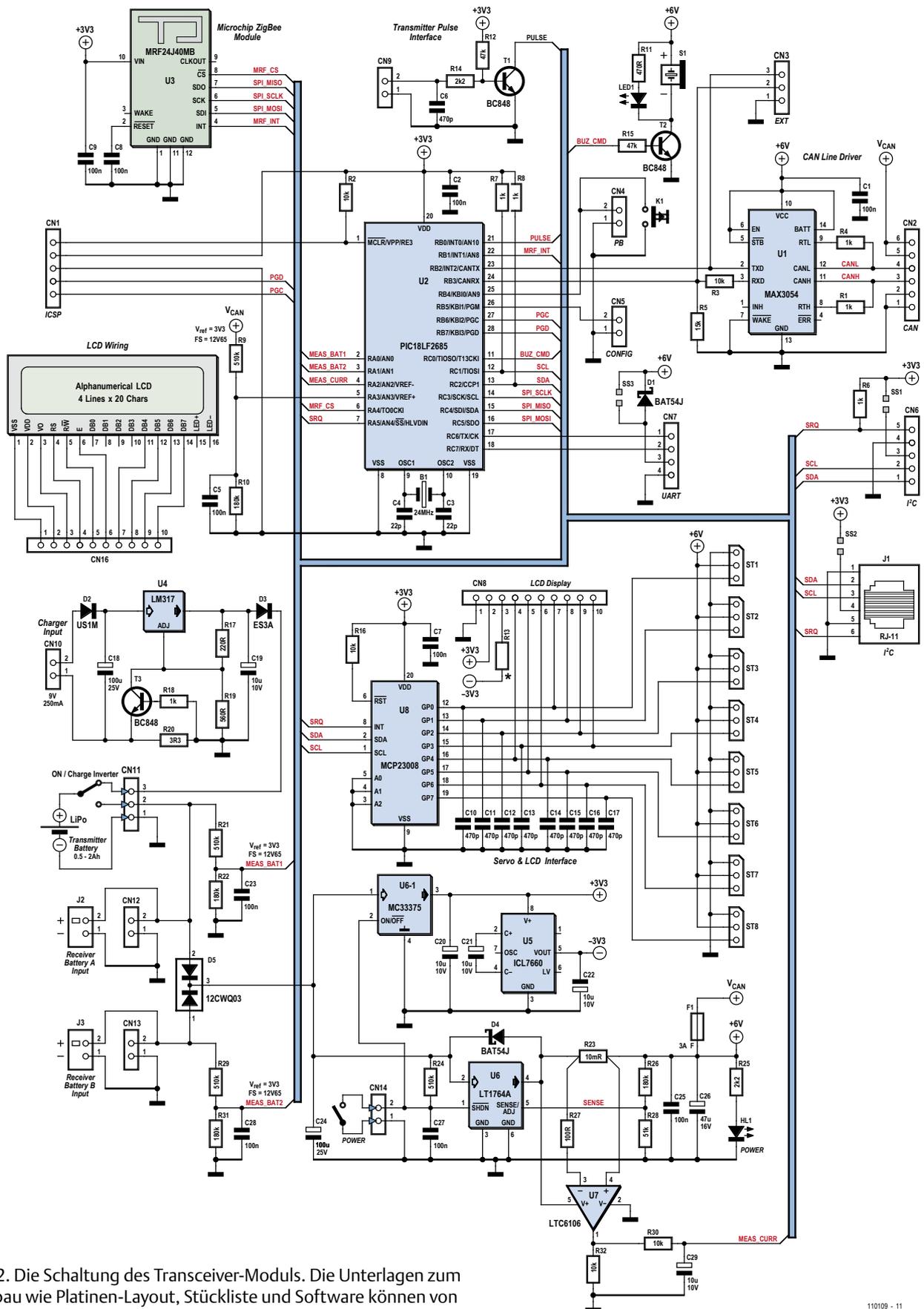


Bild 2. Die Schaltung des Transceiver-Moduls. Die Unterlagen zum Aufbau wie Platinen-Layout, Stückliste und Software können von [1] kostenlos heruntergeladen werden.

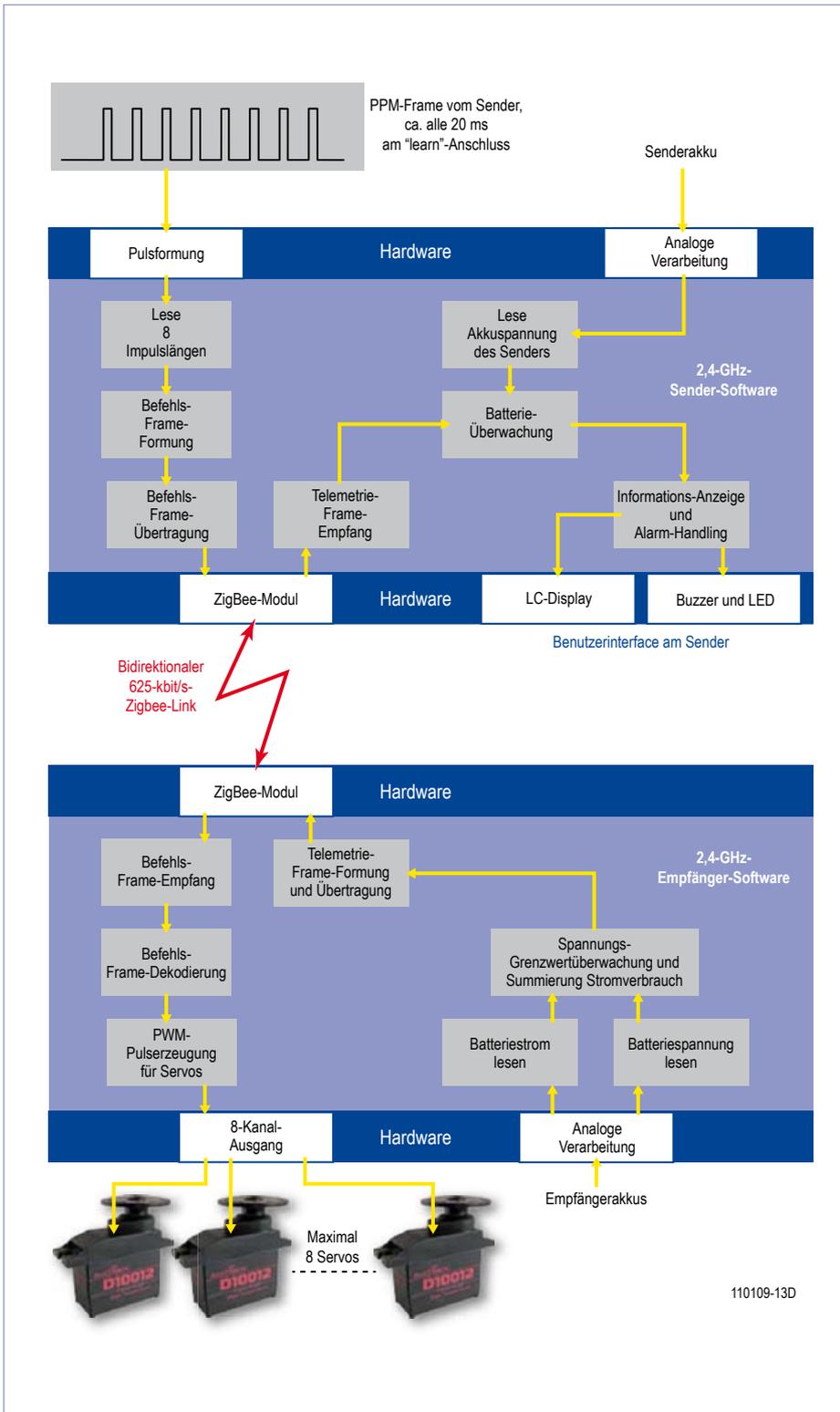


Bild 3. Übersicht über die verschiedenen Software-Teile.

Sendermodus dient das IC als Interface für den Anschluss eines Standard-LCDs mit vier Zeilen zu zwanzig Zeichen.

U3 (MRF24J40MB) ist als HF-Modul für die drahtlose Übertragung der Steuer- und Telemetriesignale zuständig. Das Modul ist mit einem klassischen SPI-Anschluss ausgestattet und wird mit einer Spannung von 3,3 V versorgt. Die Ausgangsleistung von 100 mW entspricht den üblichen gesetzlichen Anforderungen.

Der durch T2 angesteuerte Summer (Buzzer) soll im Sendermodus den Piloten darauf aufmerksam machen, dass sich einer der gemessenen Werte dem eingestellten Grenzwert nähert. Im Empfängermodus kann der Buzzer durch LED1 ersetzt werden. Transistor T1 bildet das Interface für das über CN9 empfangene PPM-Signal.

Die Software

Die Software für den Mikrocontroller wurde komplett in C geschrieben. Wie bei solchen Projekten üblich, ist der Quell-Code kostenlos über die Projekt-Webseite von Elektor [1] verfügbar. Bei dieser Software handelt es sich übrigens um echtzeitfähigen Code. Es findet nämlich eine strikte Synchronisation mit den vom Sender empfangenen PPM-Frames statt, die mit einer Frequenz von 50 Hz eintreffen. Diese direkte Reaktionsfähigkeit ist wichtig für ein gutes Gefühl des fernsteuernden Piloten.

In der Praxis ergibt sich bei diesem System eine Verzögerung von nur 4 ms zwischen dem Ende eines PPM-Frames und dem Anfang eines Kommandos beim ersten Servo des gesteuerten Flugzeugmodells.

Bild 3 bildet die wichtigsten Teile des für Sender und Empfänger notwendigen Programms ab.

Wenn es näher interessiert: Der Autor setzt hier einen von ihm entwickelten Multitasking-Kernel ein. Diese Grundlage ist entscheidend für eine hohe Reaktionsgeschwindigkeit, die für eine solche Anwendung sehr wichtig ist. Der Kernel und die zugehörigen Dienste sind in der Datei „lisezmoi.txt“ (= Liesmich.txt) beschrieben, die sich ebenfalls im Software-Archiv befindet.

Montage und Test

Beim Lötten an den beiden ICs MCP23008 und LTC6106 muss sehr vorsichtig vorge-

Tabelle 1: Übersicht über gemessene Größen und Werte.

Display-Zeile	Zu messende Größe	Messbereich	Messauflösung	Einheit	Alarmschwelle
1	Tx Akku	0...12,65	10 mV	Volt	< 3,8 V
1	Tx RSSI	0...99	1	%	Nicht verwendet
2	Rx AkkuA	0...12,65	10 mV	Volt	< 7,6 V
2	Rx AkkuB	0...12,65	10 mV	Volt	< 7,6 V
2	Rx RSSI	0...99	1	%	Nicht verwendet
3	Rx Strom	0...3300	3 mA	mA	Nicht verwendet
3	Rx Verbrauch	0...9999	1 mAh	mAh	Nicht verwendet
4	Spannung für Antrieb (künftige Erweiterung)	0...25,00	10 mV	V	Wird noch festgelegt
4	Strom für Antrieb (künftige Erweiterung)	0 – 200	1	A	Nicht verwendet
4	Verbrauch Antrieb (künftige Erweiterung)	0 – 9999	1	mAh	Nicht verwendet

gangen werden, da deren Pins ein Rastermaß von nur 0,635 mm aufweisen. Das LCD-Modul des Senders wird mit zehn Leitungen mit CN8 verbunden. Wenn der Mikrocontroller nicht schon vorab programmiert ist, kann das auch nach der Bestückung über das ISP-Interface CN1 mit einem Programmierer von Microchip oder einem kompatiblen Exemplar (ICD2, PICKIT oder ähnlich) erfolgen.

Widerstand R13 ist für die Kontrasteinstellung des Displays zuständig. Sein Wert hängt vom konkret eingesetzten LCD-Typ ab.

Nach Bestückung und visueller Kontrolle der Lötstellen kann man Jumper CN5 stecken und eine Betriebsspannung von 4,2 V an die Pins 1 und 2 von CN11 legen. Nun sollte die LED HL1 kurz aufleuchten und sich ein Stromverbrauch von etwa 60 mA einstellen. Der Summer gibt zwei kurze Pieptöne von sich, anschließend erscheint eine Begrüßung auf dem LCD. In der ersten Zeile des Displays sollte nun die Betriebsspannung korrekt angezeigt werden. Ist das der Fall, kann die Spannung langsam auf 3,8 V abgesenkt werden. Nach ungefähr zehn Sekunden sollte der Summer 20 Töne hören lassen und in der vierten Zeile des Displays die Meldung „No PPM signal“ erscheinen. Wenn nun der Trainingsanschluss der Fernsteuerung mit C9 verbunden und die Fernsteuer-



Bild 4. So sieht das Display bei einer Versorgung des Senders mit 4,2 V und des Empfängers mit zwei 8,4-V-Akkus aus.

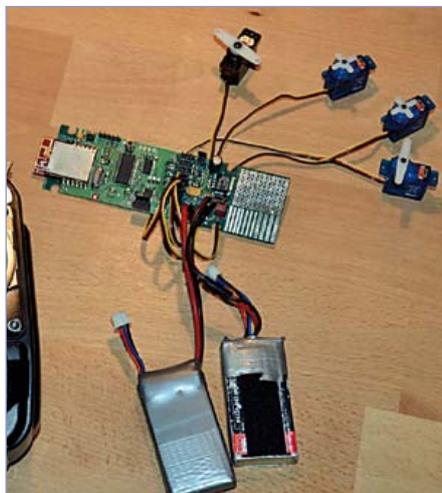


Bild 5. Der Prototyp des Autors, hier verbunden mit vier Servos und zwei LiPo-Akkus.

ung eingeschaltet wird, sollte die Fehlermeldung verschwinden.

Jetzt muss noch der Akkuladungs-Schaltungsteil überprüft werden. Hierzu legt man ohne angeschlossenen Akku eine Spannung von 9 V an CN10 an. Am besten verwendet man dafür ein Netzteil, bei dem man den Ausgangsstrom sicherheitshalber auf 1 A begrenzen kann. Jetzt misst man die Spannung an Pin 3 von CN11, die keinesfalls mehr als 4,2 V betragen darf. Legt man nun ein Amperemeter zwischen die Pins 3 und 1 von CN11, sollte der fließende Strom unter 250 mA bleiben.

An diesem Punkt ist schon ein großer und relevanter Teil der Schaltung getestet. Das 2,4-GHz-Modul wird in den folgenden Schritten überprüft.

Zunächst bestückt und überprüft man auf die gleiche Weise eine zweite Platine, bei welcher der Summer durch LED1 ersetzt ist. Die Bauteile für den Akkulader brauchen nicht bestückt zu werden. Zunächst wird die zweite Platine im Sendermodus getestet. Klappt alles, wird der Jumper CN5 abgezogen und das LCD entfernt. Nun legt man eine Betriebsspannung von 8,4 V zunächst an CN12 und dann an CN13. Nach dem Anlegen dieser Spannung sollte LD1 deutlich flackern (mit etwa 20 Hz). Die Fla-



Bild 6. Der eingebaute Empfänger.

ckerfrequenz entspricht dem Tempo der eintreffenden Frames. Das Display des Senders muss nun so wie in **Bild 4** aussehen. **Tabelle 1** gibt einen Überblick über die

gemessenen Größen und Werte. Die hier noch nicht verwendete vierte Zeile des Displays ist für Telemetriedaten (die vom Antriebsakku eines Elektroflugmo-



Bild 7. Die Christen Eagle des Autors mit eingebautem 2,4-GHz-Empfänger.

dells stammen) vorgesehen. Diese Funktion soll in einem künftigen Artikel vorgestellt werden.

Wenn der Empfänger nur aus einem Akku gespeist wird, kann man die Anzeige des zweiten Akkus ignorieren. Man sollte noch überprüfen, ob der Empfänger auch ausgeschaltet wird, wenn Anschluss C14 kurzgeschlossen ist. Zusätzlich sollte man noch kontrollieren, ob bei beiden Akkuanschlüssen ein Alarm ausgelöst wird, wenn die Spannung unter 7,6 V fällt. Die Messwerte für Spannung und Strom des Empfängers sollten korrekt sein. Schließt man ein paar Servos an den Empfänger an (**Bild 5**) und bewegt den Steuerknüppel, sollten die entsprechenden Servos reagieren.

Nun müssen beim Sender noch das Display und der Akku in einem nicht abschirmenden Kunststoffgehäuse untergebracht werden, das irgendwie am Sender befestigt wird. Wichtig ist ein haptisch wie mechanisch guter Schalter und eine Ladebuchse. Wenn man keine kleine Antenne anbringen will, dann kann man das Modul MRF24J40MB auch durch den Typ MRF24J40MC ersetzen. Bevor man das System in ein Modellflugzeug einbaut (**Bild 6 & 7**), sollte man nochmals alle Lötstellen und die Spannungen überprüfen. Beim geringsten Verdacht empfiehlt sich eine gründliche Untersuchung, damit mögliche Fehler vor dem Einbau behoben werden – es geht schließlich um Ihr wertvolles Modell.

(110109)

Weblinks

- [1] www.elektor.de/110109
- [2] 2,4 GHz Modul von Microchip: www.microchip.com/wwwproducts/Devices.aspx?dDocName=en027752
- [3] www.elektor.de/080253
- [4] ZigBee-Netzwerkanalyzer – ZENA: www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en520682
- [5] Website des Autors: <http://breakinbench.free.fr>

 <p>Entwicklung industrietauglicher Software und Hardware sowie Elektronik 03303/212166 oder www.jasys.de</p>	<p>Bausätze zu ELEKTOR 1986 bis heute! Teilesätze, Platinen, programmierte Controller sowie Cds zu fast allen Elektor-Projekten vom Spezialist. Alle Elektor-Artikel zum Verlagspreis. Ihr zuverlässiger Partner für aktive und passive elektronische Bauteile und Komponenten:</p>	<p>Auch Ihr Unternehmen ist eine Anzeige wert!</p>	<p>www.anttronic.de ab 1 Stck. ANTTRONIC Leiterplatten zu TOP-Preisen!!</p>
<p>Alles Spule! Wir liefern und fertigen: Drähte, HF-Litzen, Ferrit- und Eisenpulverkerne, Spulenkörper, Isoliermaterial, Klebebänder, Tränklacke, Übertrager, RFID-Spulen, Sensor- und Aktorspulen, Prototypen, Kleinserien, Serien, Ersatzteile und vieles mehr. MM Menting Mikroelektrik www.spulen.com</p>	<p>Geist Electronic-Versand GmbH Tel.: 07720/36673 Fax: 07720/36905 Mail: info@geist-electronic.de Shop: www.geist-electronic.de</p>	<p>VTS – ELEKTRONIK GbR Preiswerter Leiterplatten Service info@vts-elektronik.de www.vts-elektronik.de</p>	<p>SCOPES und mehr HAMEG® Instruments A Rohde & Schwarz Company</p>
<ul style="list-style-type: none"> • Konfigurierbare digitale & analoge Schaltaktoren für die Hausautomation • Seriell ansteuerbar über Modbus von PC, SPS oder µController • 4,3" TFT Touch-Display mit PC-Extender  <p>www.elconeq.de Tel. 02832-9784 301 Elconeq TECHNOLOGIES</p> <p>Hard- & Softwareentwicklung</p> <ul style="list-style-type: none"> • µController-Module (8051-komp.) z.B. 64kFlash, 2xCAN, 2xUART, I²C, RTC, 32K-FRAM, ID • PC-Erweiterungen digital/analog 	<p>Laehn-Versand.de schlanke Preise - fettes Angebot Schnellversand ohne Mindestumsatz Bauteile - Ersatzteile - Zubehör Fernbedienungen - Zeilentrafos HDTV DVB-T Sat Audio/Video Überwachungstechnik und vieles mehr. www.Laehn-Versand.de</p>	<p>Baugruppenbestückung vom Prototypen bis zur Serie FS-ELECTRONIC.de</p>	<p>GESUCHE</p>
<p>Ausgabe: Elektor November 2011</p> <p>Anzeigenschluss: 20. 09.2011</p> <p>Erscheinungstermin: 19.10.2011</p>	<p>Die Buchung einer Anzeige beinhaltet einen kostenlosen Eintrag auf der Website von Elektor, inklusive eines Links zu Ihrer Seite.</p> <p>Reservieren Sie jetzt Ihre Jahresbuchung!</p>		

www.elektor.de

Audio-DSP-Kurs

Teil 3: das DSP-Board

Alexander Potchinkov (D)

Nun wird das DSP-Board vorgestellt, auf dem nicht nur die in den Folgeartikeln vorgestellten Anwendungen laufen werden, sondern auch die ersten eigenen –und dann hoffentlich viele weitere- Schritte in der digitalen Audiosignalverarbeitung gegangen werden können. Das DSP-Board ist sofort ‚stand alone‘ anwendbar. Auch wenn es als ‚Lern-Board‘ ideal ist, genügt es doch mit seiner 24-Bit Signalverarbeitung bei Abtastfrequenzen von bis zu 192 kHz und den leistungsfähigen Schnittstellen auch sehr hohen Qualitätsansprüchen, die sowohl an die Störsignalabstände der Signale als auch an die Rechenleistung des DSP gestellt werden können.

Das DSP-Board dient der Verarbeitung von Audiosignalen mit einem DSP. Die Signale können entweder analoge oder digitale Signale sein, die auch zusammen genutzt werden können. **Bild 1** zeigt ein Blockbild der Schaltung. Wegen der getroffenen Auswahl an Bauteilen werden trotz des recht beträchtlichen Leistungsumfangs nur 13 ICs und eine Leiterplatte von 97 mm auf 66 mm benötigt. Für die Verarbeitung der analogen Signale stehen zweikanalige ADC und DAC zur Verfügung, die 24-Bit Wortbreite und Abtastfrequenzen von bis zu 192 kHz bieten. Diese Umsetzer wurden nach Preis, möglichst geringer notwendiger Peripherie und Verfügbarkeit ausgewählt. Sie sind für den Hardware-Modus geeignet und müssen nicht für unterschiedliche Abtastfrequenzen unterschiedlich konfiguriert werden. **Bild 2** zeigt die möglichen Audiosignalfade auf unserem DSP-Board. Der DSP ist der Audiomaster und empfängt eingangsseitig die in das I²S-Format gewandelten Analog- und Digitalsignale. Ausgangsseitig werden Signale zur gleichzeitigen Wandlung in Analog- und Digitalaudiosignale zur Verfügung gestellt. Im DSP können wir uns einen dreistufigen Quellenumschalter vorstellen. In der Stellung 1 wird das Analog-Audio, in der Stellung 2 das Digital-Audio und in der

Stellung 3 vom DSP selbst erzeugte Signale auf den Audioausgang geschaltet, wenn wir nur Audiopfade und nicht auch noch Audiosignalverarbeitung betrachten wollen.

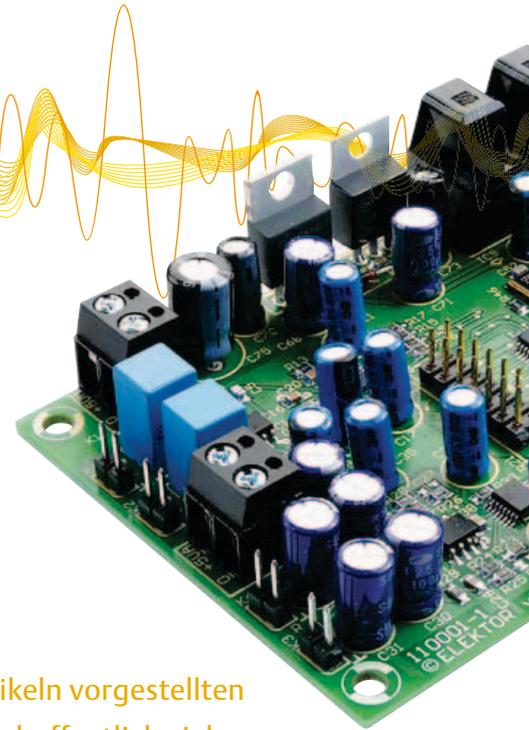
Ohne Verwendung des DSP ist weder Signalverarbeitung noch Signalübertragung möglich.

Digitalsignale sind wahlweise über optische oder elektrische Schnittstellen ein- und auszugeben. Eingangsseitig ist der Betrieb entweder nur optisch oder nur elektrisch möglich, ausgangsseitig lassen sich der optische und der elektrische Ausgang gleichzeitig nutzen. Eingangsseitig dient ein (asynchroner) Abtastratenumsetzer SRC (Sample Rate Converter) dazu, Digitalsignale in einem sehr weiten Bereich von Abtastfrequenzen auf Digitalsignale mit der Abtastfrequenz umzusetzen, mit der die digitale Signalverarbeitung ausgeführt wird. Für diese Artikelserie wird die professionelle Abtastfrequenz 48 kHz gewählt, die genügend Bandbreite und dennoch eine hohe Rechenleistung ermöglicht. Diese digitale Signalverarbeitung wird mit einem DSP56374 von Freescale vorgenommen, der speziell für die Audiosignalverarbeitung zugeschnitten und universell programmierbar ist. Wenn mit

dem DSP eine Signalverarbeitung bei einer Abtastfrequenz von beispielsweise 96 kHz gewünscht ist, müssen lediglich in zwei Schnittstellen-Konfigurationsregistern je ein Bit geändert werden.

Mit dem DSP-Board sind zahlreiche Anwendungen möglich. Man kann einen CD-Player direkt anschließen und mit dem LED-Board dieser Artikelserie einen Aussteuerungsmesser aufbauen, man kann ein Digital-Mikrophon und einen Verstärker mit Lautsprecher anschließen und eine Unterdrückung des Rückkopplungspfeifens realisieren oder die Klirrkoeffizienten eines Analogsignals berechnen und die Rechenergebnisse auf ein Display schreiben. Trotz der vielseitigen Anwendungsmöglichkeiten des DSP-Boards werden nur recht wenige Bauelemente eingesetzt, was wegen der mittlerweile sehr hoch entwickelten Digital-Audio-IC-Technik möglich ist.

Der Blick auf das Schaltbild zeigt vier Signalverarbeitungsblöcke, die Analogsignal-Ein- und -Ausgabe, die Digitalsignal-Ein- und -Ausgabe, den DSP und die Peripherie des DSP. Diese Blöcke werden wir im folgenden vorstellen, damit der Leser erfährt, was alles vorhanden ist und was mit dem DSP-Board verwirklicht werden kann.





Kommunikation auf dem DSP-Board

Das Blockbild in **Bild 3** gibt Aufschluss über die Kommunikationsstrukturen auf unserem DSP-Board. Auf den ersten Blick überrascht das Bild mit seiner Fülle an Details angesichts des kleinen Boards, aber es lässt auch die Flexibilität und die vielen Verwendungsmöglichkeiten erkennen. Die Kommunikation erfolgt über zwei Busse, Audio über den I²S-Audiobus und Steuer- sowie sonstige Daten über den SPI-Bus. Der Audiobus hat fünf Teilnehmer, den DSP als Master und die anderen vier Teilnehmer als Slaves. Die Taktleitungen sind schwarz markiert und liegen in zwei Gruppen vor, die aber im DSP korrespondierend miteinander verbunden werden können. Wir haben dies mit geklammerten Bezeichnungen und gestrichelt gezeichneten Verbindungsleitungen symbolisch dargestellt. Wir haben die Trennung in Taktsignale auf dem Board und Taktsignale am I²S-Port K6 deswegen vorgenommen, damit der Port auch als GPIO-Port genutzt werden kann, wenn kein Bedarf an einem Audio-Port vorliegt. Bei Nutzung des Ports als I²S-Port können DSP-intern die Taktleitungen HCKR mit HCKT, FSR mit FST und SCKR mit SCKT verbunden werden. Die sechs oberen Audiobusleitungen sind die Audiodatenleitungen, von denen drei intern auf dem Board genutzt werden und drei auf den Port K6 geführt werden. Unterhalb des markierten I²S-Netzes sind die Verbindungen zu den Analog- und Digitalaudioschnittstellen des Boards dargestellt.

Der zweite Bus ist der SPI-Bus mit vier Bus Teilnehmern. Auch hier ist der DSP der Master, der den Bittakt für den Schieberegisterbetrieb zur Verfügung stellt. Die drei Slaves sind das SEEPROM als wiederbeschreibbarer nichtflüchtiger Speicher mit geringer Zugriffsgeschwindigkeit, der SRC und der SPI-Port K7, über den ein beliebiger externer SPI-Slave angeschlossen werden kann. In unserem DSP-Kurs werden wir an diesem Port eine LED-Balkenanzeige mit Daten versorgen. In einem SPI-System müssen die

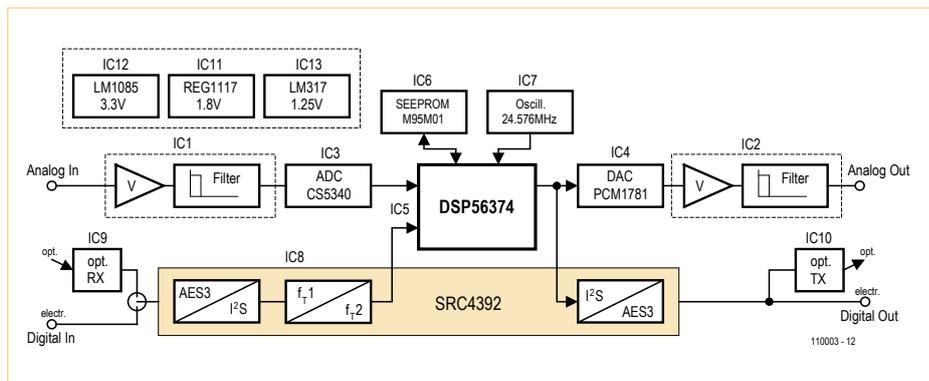


Bild 1. Blockbild der Schaltung. Die Schnittstellen sind zweikanalig ausgeführt.

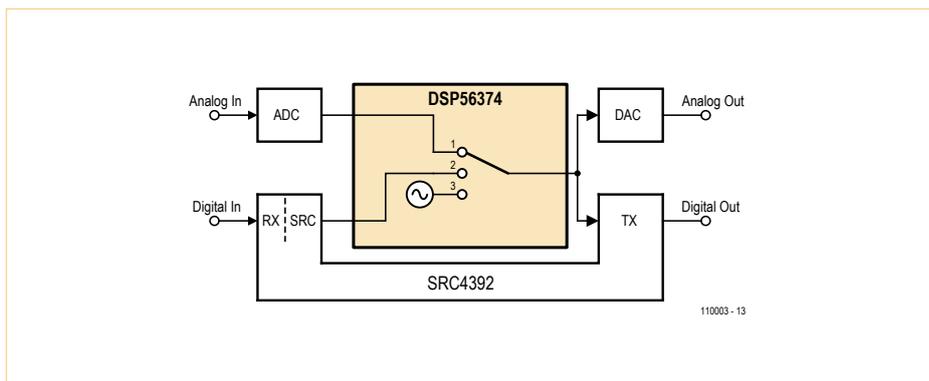


Bild 2. Audiosignalfade.

Slave-Teilnehmer mit Chip-Select-Signalen aktiviert bzw. deaktiviert werden, damit nicht zwei Slaves gleichzeitig die MISO-Leitung beschreiben. Diese Signale werden vom DSP erzeugt. Weitere Leitungen dienen der Bedienung des SRC. Mit einer Leitung kann der SRC in den Reset-Zustand gesetzt werden, was z.B. vor seiner Programmierung erfolgen muss und mit der anderen Leitung signalisiert der SRC dem DSP das Vorhandensein eines Digital-Audio-Signals an seinem Digital-Audio-Empfänger.

Analogsignal Ein-und-Ausgabe, IC1 bis IC4

Die zweikanalige Analogsignaleingangsschnittstelle mit den Stiftleisten K1 und K2 ist rund um die beiden Operationsverstärker

IC1.A und IC1.B sowie den ADC IC3 vom Typ CS5340 aufgebaut (**Bild 4**). Der ADC wird analogseitig mit einer 5V-Betriebsspannung versorgt. Die beiden Operationsverstärker mit Einheitsverstärkung fügen dem mit den Kondensatoren C1 und C2 AC-gekoppelten Audiosignal einen DC-Offset von der halben Betriebsspannung hinzu und sorgen für eine niederohmige Ansteuerung des ADC und des analogen Antialiasingteufilters für den überabtastenden ADC. Der ADC arbeitet fest eingestellt als Audio-Slave, d.h. der DSP sorgt für die notwendigen Audiotakte, die Masterclock, die Bitclock und die LR-Clock (Left-Right-Clock, entspricht der Abtastfrequenz). Je nach gewünschter Abtastfrequenz wird der ADC in einem der drei Modi, Single-, Double- und Quad-

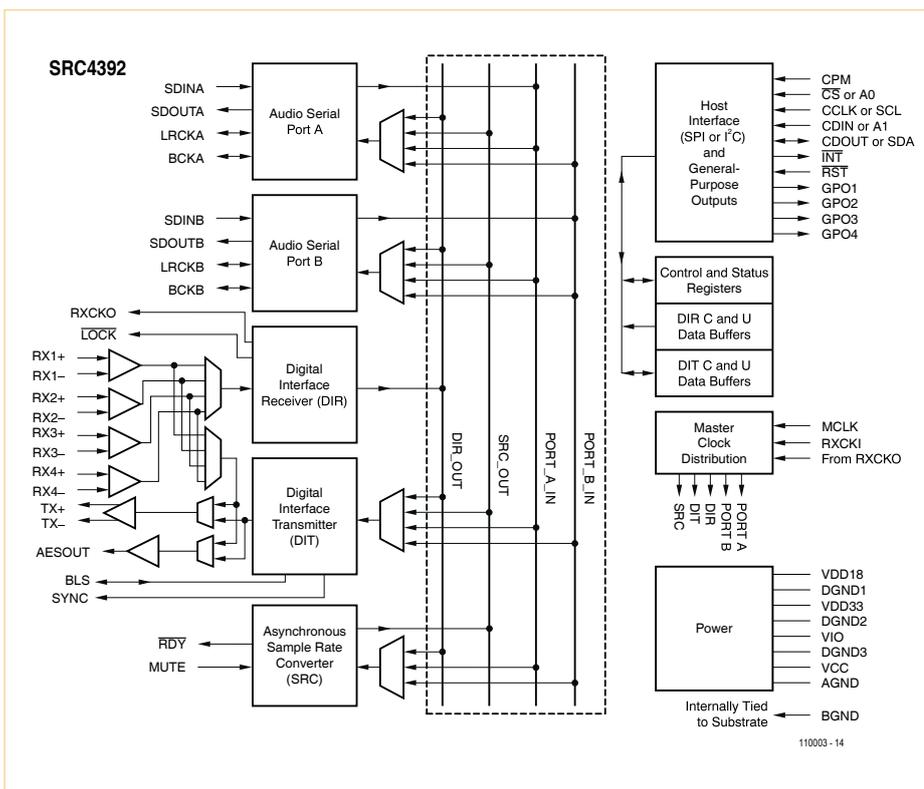


Bild 3. Kommunikation auf dem DSP-Board.

Speed-Mode, betrieben, für die jeweils unterschiedliche Verhältnisse von Master zu Left-Right-Clock im Bereich von 64 bis zu 512 vorgegeben werden können. Der ADC ist in der Lage, das vom DSP vorgegebene Taktverhältnis, das mit dem jeweiligen DSP-Programm festgelegt wird, selbst zu erkennen. Laut Datenblatt werden 101 dB Dynamik und -94 dB THD+N erreicht, was auch für anspruchsvolle Anwendungen genügt. Der Spitze-zu-Spitzen-Wert bei sinusförmigen Signalen liegt zwischen dem 0,53- und 0,59-fachen der Betriebsspannung mit einem Mittelwert entsprechend von 2,8 V. Auf der Digital-Seite, die mit der DSP-Peripherie-Betriebsspannung von 3,3 V betrieben wird, ist der I²S-Audiomodus mit dem Pull-Up-Widerstand R14 am Pin 4 eingestellt. Der I²S-Audiomodus wird für alle DSP-Board-internen Audiosignale verwendet. R13 und C20 sorgen für den Einschalt-Reset.

Die zweikanalige Analogsignalaußgangsschnittstelle mit den Stiftleisten K3 und K4 ist rund um den DAC IC4 vom Typ PCM1781 und die beiden Operationsverstärker IC2.A und IC2.B aufgebaut. Der DAC bietet bei 24 Bit Audiowortbreite 106 dB Dynamik und einen THD+N von typisch 0,002 %, was ungefähr -94 dB entspricht. Die mög-

lichen Abtastfrequenzen liegen im Bereich von 5 kHz bis zu 200 kHz. Der DAC ist auf I²S eingestellt und wird als Audio-Slave vom DSP bedient. Die vom DSP erzeugten Audiotakte sind für alle digitalen Audioschnittstellen gleich, d.h. für ADC, DAC und SRC. Die vier Einstellpins 1 bis 4 sind eingestellt auf I²S, De-Emphasis off und Mute off. Auch der DAC verfügt über eine Automatik, die das vom DSP vorgegebene Verhältnis von Master- zu LR-Clock erkennt und so einen Betrieb ohne Programmierung erlaubt. Mit den beiden Operationsverstärkern werden die beiden Rekonstruktionsfilter (Tiefpassfilter, mit denen das überabgetastete Digitalsignal in ein Analogsignal umgeformt wird) als Potenz- oder Butterworthfilter zweiter Ordnung mit einer DC-Verstärkung von A₀ = 1 aufgebaut. Mit den angegebenen Bauteilwerten liegt die Grenzfrequenz bei ca. 30 kHz. Wir haben diese Grenzfrequenz absichtlich nicht sehr hoch gewählt, da solche DAC im wesentlichen Hochpass-Rauschen als Störsignal erzeugen und man daher die Bandbreite so klein wie möglich wählen sollte.

Eine Änderung der Filtereigenschaften ist nicht schwierig. Wenn die Butterworth-Charakteristik beibehalten werden soll, beträgt

die Filtergüte $Q=1/\sqrt{2}=0,7071$. Man gibt die DC-Verstärkung A₀, den Kapazitätswert des Kondensators C26 für das Filter des linken bzw. C27 für das Filter des rechten Kanals und die Grenzfrequenz f₀ bzw. $\omega_0 = 2\pi f_0$ vor. Die weiteren Bauelemente werden für den linken Kanal zu $C28 = C26/(4 \cdot Q^2 \cdot (1+A_0))$, $R24 = 2 \cdot Q/(\omega_0 \cdot C26)$, $R25 = (1+A_0) \cdot R24$ und $R23 = (1+A_0) \cdot R24/A_0$ berechnet. Die Formeln vereinfachen sich für Q = 0,7071 und A₀ = 1 zu $C28 = C26/4$, $R24 = 1,4142/(\omega_0 \cdot C26)$, $R23 = R25 = 2 \cdot R24$. Entsprechend sind die Bauelemente für das Filter des rechten Kanals zu berechnen. Der DC-Offset für die Filter wird vom DAC am Pin 13 gewonnen.

Besonderes Augenmerk wurde auf den Schutz der Analogausgänge gelegt, falls diese an einen Mikrophoneingang mit einer Phantomspannung von bis zu 48 V angeschlossen werden. Um Gleichspannungen in dieser Höhe in beiden Richtungen vertragen zu können, sind die Ausgangselkos ausreichend spannungsfest und bipolar verschaltet. Die Doppel-Schottky-Dioden D1 und D2 schützen die Operationsverstärker für diesen Betriebsfall bei einem möglichen Kabelkurzschluss, denn in diesem Fall würden die auf ca. 48 V aufgeladenen Elkos ihre Energie an die Operationsverstärker abgeben. Die Ausgangsspannung beträgt bei Vollaussteuerung des DAC mit einem Sinussignal ca. 3,9 V_{pp}.

Digitalsignal Ein-und-Ausgabe, IC8 bis IC10

Die Audioübertragung innerhalb des DSP-Boards erfolgt im Format I²S, einem synchronen Drei-Leitungs-Format mit LR-Clock, Bit-Clock und den Audiodaten. Nach außen muss ein anderes Format, Digital-Audio, benutzt werden, das so codiert ist, dass ein Betrieb z.B. über eine optische Verbindung mit einem einzigen Lichtwellenleiter möglich ist. Hierzu werden zwei Konverter benötigt, die das I²S-Signal in Digital Audio und umgekehrt umsetzen. Diese beiden Konverter, im folgenden als Receiver RX und Transmitter TX bezeichnet, sind im IC8 vom Type SRC4392 untergebracht.

Bevor die Funktion und Bedienung von IC8 besprochen wird, werden kurz die Digital-Audio-Schnittstellen des DSP-Boards

beschrieben. Auf dem Board selbst sind ein optischer Empfänger IC9 und ein optischer Sender IC10 vorhanden, die an den RX und TX im IC8 anzuschließen sind. Elektrische Schnittstellen sind ebenfalls vorhanden. Mit einem Jumper wird an der Stiftleiste JP1 der optische oder der elektrische Eingang gewählt. Zum Betrieb der optischen und der Koaxialschnittstelle muss an der Stiftleiste K9 ein Jumper auf die Positionen 1 und 2 gesteckt werden (Minus-Anschluss mit Masse verbinden), damit der Eingang RX1-des SRC gegen Masse über C54 terminiert wird. Wenn ein professionelles Signal eingespeist werden soll, ist dieser Jumper zu entfernen. An der Stiftleiste K9 lässt sich entweder ein asymmetrisches Signal mit einem Abschlusswiderstand von 75 Ω (R54) einspeisen oder aber ein symmetrisches professionelles Signal, wozu man normgerecht einen Standardimpulsübertrager und zwei 18-Ω-Widerstände direkt an der hierfür benötigten XLR-Buchse anschließen sollte. Den Impulsübertrager kann man sich mit einem kleinen Ferritringkern und wenigen Zentimetern Kupferlackdraht selbst anfertigen. An der Stiftleiste K10 steht ein differentieller RS422-Ausgang zur Verfügung, der entweder mit einem Impulsübertrager und einem 110-Ω-Widerstand als professioneller AES-3-Ausgang oder mit einem Widerstand und einem 10-nF-Kondensator als asymmetrischer S/PDIF-Ausgang verwendet werden kann. Darüber hinaus sind an der Stiftleiste K6 weitere I²S-Schnittstellen angeschlossen, die vom DSP bedient werden und den Anschluss üblicher ICs mit I²S-Schnittstelle erlauben.

Der Baustein IC8 ist ein sehr leistungsfähiger Schnittstellenbaustein und wird vom Hersteller als *Two-Channel, Asynchronous Sample Rate Converter with Integrated Digital Audio Interface Receiver and Transmitter* bezeichnet. Der Blick in das Blockbild in Bild 5 lässt einige Funktionsblöcke und vier Audio-Busse erkennen. Auf der linken Seite sind die Blöcke für das Digital-Audio-Signal abgebildet. Die beiden Audio-Serial-Ports A und B, von denen auf dem DSP-Board nur Port A genutzt wird, stellen die Anbindung an den DSP her. Der Port wird im I²S-Modus betrieben und ermöglicht dem DSP die Audio-Ein- und -Ausgabe. Wir haben zwei

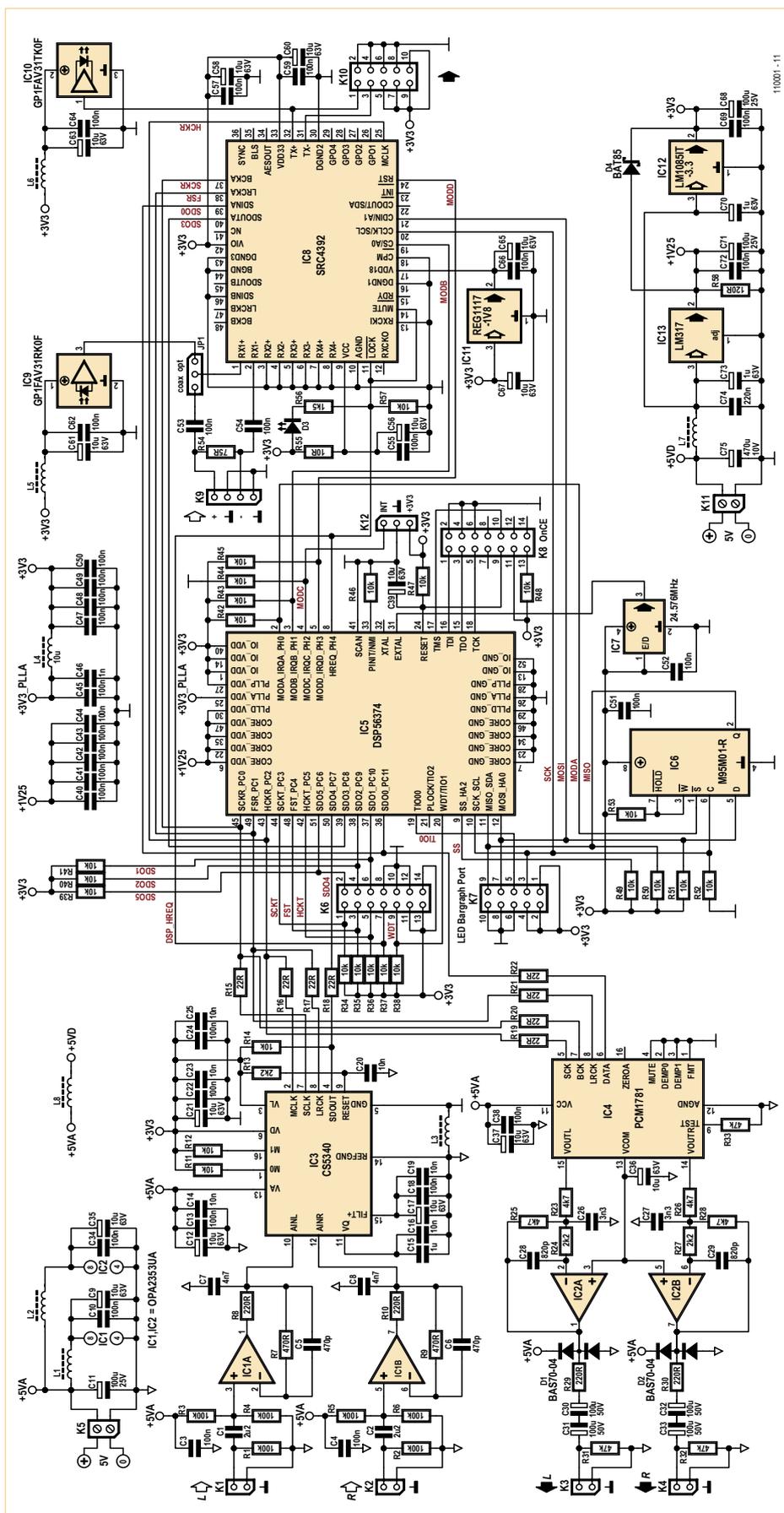


Bild 4. Die komplette Schaltung des DSP-Boards.

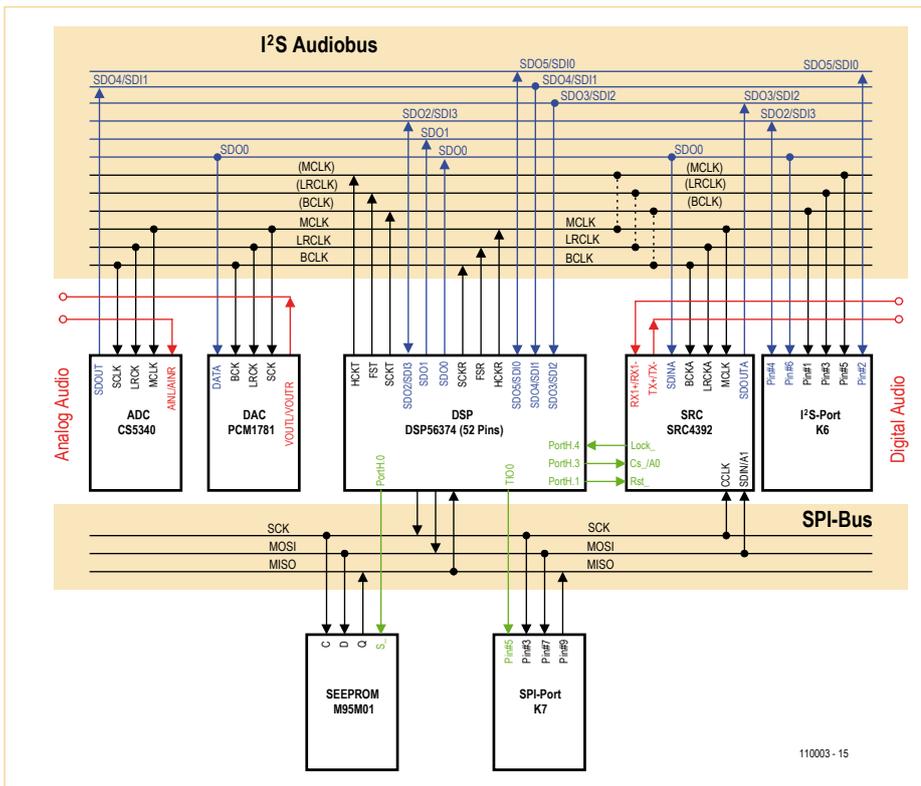


Bild 5. Blockbild des SRCs.

Signalpfade für die Ein- und Ausgänge von Digital-Audiosignalen vorgesehen: Differentieller Digitaleingang RX1+ und RX1- auf den Digital-Interface-Receiver DIR, von dort über den Bus DIR-OUT zum asynchronen Sample-Rate-Konverter SRC und schließlich über den Bus SRC_OUT zum seriellen PortA, Datenausgang SDOUTA, an den der DSP angeschlossen ist.

Das in Port A eingespeiste Audiosignal wird über den Bus PORT_A_IN zum IC8 eigenen Digital-Audio-Transmitter DIT transportiert und dort in Digital Audio gewandelt. Der DIR setzt Digital Audio auf ein synchrones internes Audiosignal um, das vom SRC gelesen werden kann, und synchronisiert die Eingangsschnittstelle des SRC. Die Ausgangsschnittstelle des SRC wird vom DSP synchronisiert, der die Audiotakte über den PortA an IC8 abgibt. Ein (asynchroner) SRC interpoliert das Audiosignal. Man kann sich das so vorstellen, dass aus dem eintreffenden Audiosignal ein quasikontinuierliches Signal, das wie ein Analogsignal anzusehen ist, erzeugt wird und anschließend, mit einer anderen Abtastfrequenz abgetastet, das Ausgangssignal bildet. Da ein Analogsignal nicht benötigt wird, kann der Interpolationsvorgang rein digital ablaufen. Der SRC kann Audiosignale mit Abtastfrequen-

zen zwischen 20 kHz und 216 kHz verarbeiten und diese beispielsweise auf die professionellen Abtastfrequenzen 48 kHz oder 96 kHz umsetzen, mit denen dann der DSP die Audiosignalverarbeitung vornimmt. Über den PortA wird der DSP mit Audiodaten versorgt und speist sie auch in diesen Port über den Pin SDINA ein.

Wenn man den Funktionsumfang und die Audioqualität von 24 bit bzw. 140 dB betrachtet, dann sieht man, dass der gewählte SRC zum oberen Leistungsbe- reich vergleichbarer Bausteine gehört. Und vielleicht ahnt man bereits, dass auch noch ein Nachteil erwähnt werden sollte. Ein so komplexer Baustein hat keinen Hardware-Modus, d.h. er muss programmiert werden. Die Programmierung erfolgt mit dem Schreiben von 52 Bytes in die Kontrollregisterbank, wofür eine SPI-Kommunikation gewählt wurde (siehe Tabelle).

Die übrigen Register werden mit Nullen geschrieben. Vor Beginn mit dem Register \$01 sind zwei Bytes \$01 zu schreiben, so dass insgesamt 54 Bytes übertragen werden. Die Byte-Folge ist in der Datei `src4392.tab` abgelegt. Ein wichtiges Signal von IC8 ist das Lock-

Signal an Pin 11, mit dem ein gültiges Digital- Audio-Signal am Empfänger angezeigt wird. Dieses Signal nutzen wir in den DSP- Programmen, um von Digital- auf Analogeingang und umgekehrt umzuschalten. Liegt ein gültiges Digitalsignal vor, wird das SRC-Signal verarbeitet, andernfalls das ADC-Signal. Das gültige Digitalsignal wird mit der LED D3 angezeigt.

Das IC8 hat eine eigene 1,8-V-Spannungs- versorgung, die mit einem Low-Dropout- Regler (IC11) aus der 3,3-V-Versorgungs- spannung gewonnen wird.

DSP mit Taktoszillator, IC5 und IC7

Der Audio-DSP, ein DSP56374 von Freescale, ist ein hochintegrierter Baustein, der nur wenig Peripherie benötigt. Der Takt des DSP wird aus einem 24,576-MHz-Quarzoszillator IC7 abgeleitet und mit dem DSP-internen PLL-basierten Taktfrequenzvervielfacher auf das sechsfache, also 147,456 MHz hochgetaktet. So stehen bei einer Abtastfrequenz von 48 kHz dem Anwender 3072 Instruktionen in einem Abtastzeitintervall für die Signalverarbeitung zur Verfügung. Aus diesem Prozessortakt werden durch DSP-interne Teilung die Audiotakte Master-, Bit- und LR-Clock gewonnen, die von ADC, DAC und SRC genutzt werden. Der DSP arbeitet als Audiomaster. Es lassen sich so u.a. die wichtigen Abtastfrequenzen 48 kHz, 96 kHz und auch 192 kHz einstellen, wobei die zuletzt genannte Abtastfrequenz technisch wenig sinnvoll ist und wohl eher den Werbestrategen als Argument für ihre ADC und DAC dient, die nach wie vor nicht eine 24-bit-Audioqualität aufweisen und vor allem mit eindrucksvoll großen (und unnötigen) zulässigen Abtastfrequenzen prunken.

Im Schaltsymbol für den DSP im Bild 4 erkennt man gut die hinsichtlich ihrer jeweiligen Funktion zusammengefassten Anschlussgruppen. Oben und unten sind die Anschlüsse für Versorgungsspannungen und Masse. Der DSP verwendet für die Peripherieschaltungen die Versorgungsspannung 3,3 V und für den Prozessorkern die geringe Spannung von 1,25 V, womit auch bei hohen Taktfrequenzen eine niedrige Leistungsaufnahme ermöglicht wird. Auf der linken Seite ist mit einer Gruppe von 12 Anschlüssen die Audioschnittstelle

zusammengefasst. Die eine Hälfte der Anschlüsse sind die Datenleitungen von 6 I²S-Schnittstellen, die als Ein- und Ausgänge dienen, wobei maximal 4 der 6 Leitungen Eingänge sein können. Die andere Hälfte stellt die Audiotakte mit zwei Gruppen zu je drei Taktanschlüssen dar. Ein- und Ausgänge können getrennt getaktet werden, womit man z.B. zwei unterschiedliche Abtastfrequenzen benutzen kann. Wir machen davon keinen Gebrauch und nutzen die den Eingängen zugeordneten Anschlüsse für die Ein- wie auch für die Ausgänge. Es entsprechen so SCKR, FSR und HCKR den I²S-Takten Bit-, LR- und Masterclock. Mit der DSP-Software ist durch geeignete Programmierung der Peripherieregister die Takterzeugung nach Wunsch einzustellen. Auf dem DSP-Board wird der I²S-Anschluss SDO0 an Pin 36 genutzt, um den DAC und den SRC.TX anzusteuern, damit es im SRC in Digital Audio umgesetzt wird. Mit dem als Eingang konfigurierten Datenanschluss SDO4 wird der ADC an den DSP angeschlossen. Der Anschluss SDO3 wird für das Ausgangssignal des SRC.RX in IC8 genutzt. Dem DSP stehen somit beide Board-Eingangssignale, analog wie auch digital, zur Verfügung. Die drei Taktleitungen werden von IC3, IC4 und IC8 genutzt, so laufen ADC, DAC und SRC synchron. Die Anschlüsse TIO0 und WDT/TIO1 der nächsten Anschlussgruppe werden auf Stiftleisten geführt und können bei Bedarf für das DSP-Timersystem, den Watchdog-Timer oder aber als GPIO genutzt werden.

Tabelle. Programmieren der Funktionen des SRC.			
Adresse	Inhalt	Block	Funktion
\$01	\$37	Alle	Aktivierung von PortA, TX, RX und SRC
\$03	\$31	PortA	Format I2S, Source=SRC, Slave, Mute off
\$07	\$60	TX	Source=PortA, ClockRatio=512, ClockSource=MCLK
\$09	\$03	TX	BufferSource=SPI
\$0D	\$08	RX	Input RX1, ReferenceClock=MCLK
\$0E	\$10	RX	Clock free run after loss of clock
\$0F	\$22	RX	Reference Clock=24,576MHz
\$1B	\$07	GPO1	Receiver non valid data active high
\$2D	\$42	SRC	Source=DIR, RefClock=MCLK, Mute off, Tracking on
\$2E	\$03	SRC	64 samples group delay, true decimation, deemphasis off

Die Anschlussgruppe SS_HA2 bis MOSI_HA0 stellt eine synchrone serielle Schnittstelle zur Verfügung, die wir als bidirektionale SPI-Schnittstelle nutzen. Es werden mehrere Peripheriebausteine bedient. Zum einen wird der Schnittstellenbaustein IC8 programmiert, wofür mit dem DSP-Anschluss MODB das IC8 selektiert wird. Zum anderen ist das serielle EEPROM am SPI angeschlossen und kann gelesen und geschrieben werden. Das Selektieren erfolgt mit dem DSP-Anschluss MODA. Schließlich wird in einem der Projekte dieser Artikelserie auch ein LED-Board über das SPI geschrieben, mit dem zwei 40-LED-Balkenanzeigen aufgebaut werden. An der SPI-Stiftleiste K7 lassen sich selbstverständlich auch andere SPI-Peripherieeinheiten anschließen, wie z.B. ein Mikrocontroller mit Eingabeelementen und Display, mit dem die DSP-Programme unterschiedliche Einstellungen erhalten können und diese mit dem Display angezeigt werden.

In der nächsten Gruppe, oben auf der rechten Seite, sind die Anschlüsse MODA_

IRQA bis MODD_IRQD zusammengefasst. Nach dem DSP-Reset liest der DSP diese Anschlüsse und nutzt die eingestellten Pegel zur Wahl des Boot-Modus. Mit den Pull-up-Widerständen R42, R43 und R45 und dem Pull-down-Widerstand R44 wird das Booten aus dem board-eigenen SEEPROM via SPI voreingestellt. Nach dem Boot-Vorgang können die Anschlüsse für Hardware-Interrupts oder als GPIO genutzt werden. Wir verwenden drei der vier Anschlüsse als GPIO. Der Anschluss HREQ liegt am Lock-Ausgang von IC8, an dem ein gültiges Audiosignal am Digital-Audio-Eingang erkannt werden kann. Mit dem Anschluss MODD kann IC8 in den Reset-Zustand gebracht werden, was vor der Programmierung von IC8 erfolgen muss. Der DSP-Anschluss MODC_IRQC wird auf die Pfostenleiste K12 geführt, womit die Möglichkeit zur Nutzung eines Hardware-Interrupts geschaffen wurde. Im Falle einer Nutzung ist aber ein Konflikt mit dem Booten über das SEEPROM im DSP-Bootmodus 11 zu vermeiden, denn hierfür muss der Pegel an diesem Anschluss für kurz Zeit nach dem

Anzeige



EURO
CIRCUITS

Europas Leiterplatten-Referenz für Prototypen & Kleinserien

www.eurocircuits.de

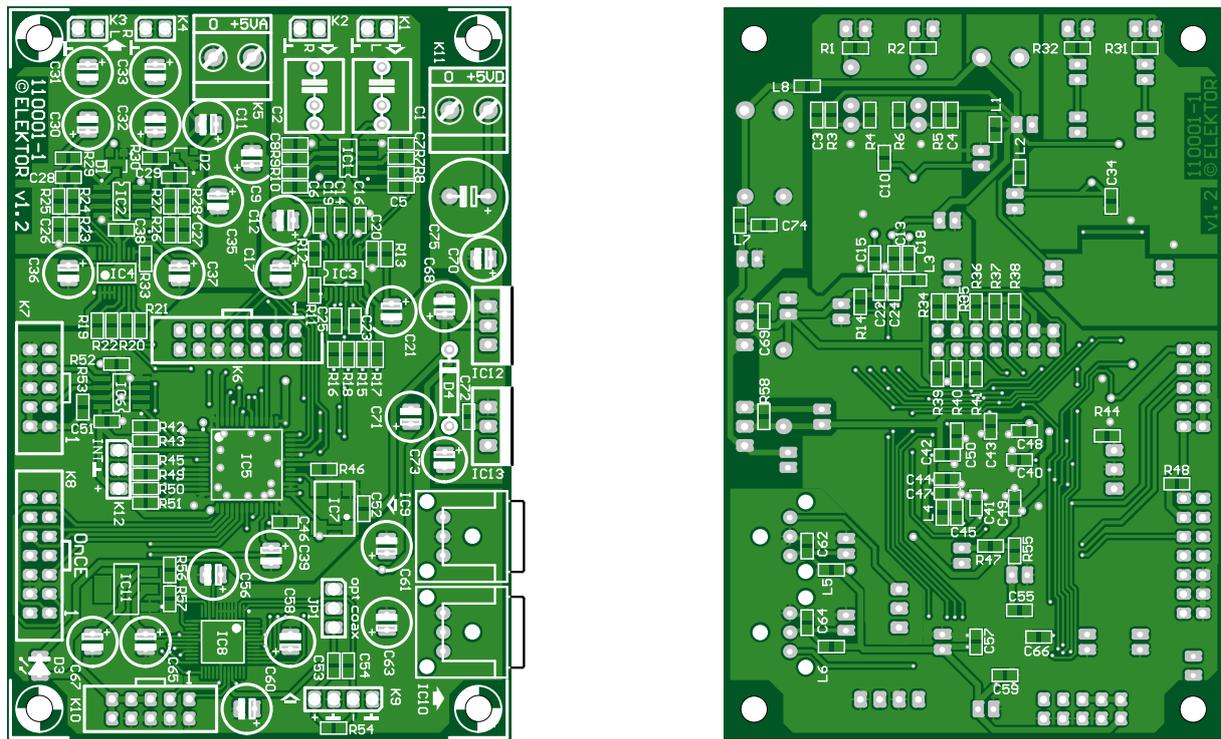


Bild 6. Bestückungsplan für beide Seiten der Platine des DSP-Boards (Stückliste auf www.elektor.de/110003).

Reset Low sein, was auf dem DSP-Board mit dem Widerstand R44 erreicht wird.

Die Anschlüsse an den Pins 31, 32 und 33 sind für die Signalverarbeitung nicht wichtig. Hier wird der Taktgenerator angeschlossen und festgelegt, ob nach dem Reset die DSP-PLL in der Taktvervielfacherschaltung aktiv ist.

Die letzte Anschlussgruppe mit den Pins 15 bis 18 ist das Debuggerinterface, das auf Stiftleiste K8 geführt wird. Hier wird die Kommunikation mit der auf einem PC laufenden Debug-Software hergestellt, mit der sich Programme laden lassen und alle DSP-Register und der DSP-eigene flüchtige Speicher gelesen und geschrieben werden können.

DSP-Peripherie, IC6 und IC7

Die DSP-Peripherie ist mit wenigen Worten kurz zu beschreiben, denn sie besteht nur aus zwei ICs, dem bereits erwähnten Taktoszillator IC7 und dem seriellen SPI-1-Mbit-EEPROM IC6 vom Type M95M01. Der DSP verfügt über dreimal 6K-Worte RAM, was $3 \cdot 6.144 \cdot 24 = 442.368$ bit entspricht und weniger als die Hälfte des Speichervermögens des SEEPROMs ausmacht. Nur in

wenigen Anwendungen wird es nötig sein, alles DSP-RAM beim Booten aus dem SEEPROM schreiben zu müssen. Darüber hinaus kann das SEEPROM während des Betriebs gelesen und auch geschrieben werden. Dies erfolgt, verglichen mit dem DSP-Takt, recht langsam. Man kann aber ohne weiteres spezielle Einstellungen oder Ähnliches, die über das SPI von einem User-Interface mit Mikrocontroller an den DSP übermittelt werden, speichern und lesen. Ein wenig Aufwand verlangt die Programmierung des SEEPROMs bei Verwendung des nützlichen Autoincrement-Modus der Adressierung schon, da das SEEPROM in Seiten organisiert ist und man beim Lesen und Schreiben großer Datenmengen die Seitenaufteilung berücksichtigen muss.

Spannungsversorgung, IC11 bis IC13

Zuletzt betrachten wir noch in aller Kürze die DSP-Board-Spannungsversorgung. Das Board hat zwei 5-V-Anschlüsse, getrennt nach analog an Schraubklemme K5 und digital an Schraubklemme K11. Beide Anschlüsse sind mit der Induktivität L8 zusammengefasst. Bei dem von Elektor gelieferten Board braucht man nur den

Anschluss K11 mit einem externen 5V-Netzteil zu verbinden. Möglicherweise lassen sich mit getrennten Versorgungen bessere Signal-Rauschabstände der Analogsignale erzielen, was aber vor allem von der Qualität der Spannungsversorgungen abhängt. Die darüber hinaus benötigten drei Digitalversorgungsspannungen von 3,3 V für die Digitalschaltungen, 1,25 V für den DSP-Kern und 1,8 V für IC8 werden mit Hilfe der drei Linearregler IC11, IC12 und IC13 erzeugt.

Das DSP-Board

Wie schon erwähnt, wird das DSP-Board von Elektor komplett bestückt und getestet geliefert. Im nächsten Heft beschreiben wir erste Schritte zur Verwendung des Boards und eine Anzahl von Testroutinen. Außerdem stellen wir die benötigte PC-Software und deren Anwendung vor.

(110003)

elektor

Elektor Print

Gewohnter Lesespaß auf Papier



Elektor Digital

Neuer Lesespaß auf PC, Notebook
oder Tablet




Elektor PLUS

Ultimativer Lesespaß zu Hause oder unterwegs

**Lesen Sie Elektor im vorteilhaften
PLUS-Abonnement!**

Jetzt abonnieren oder upgraden: www.elektor.de/abo

Kompaktes Blinkrücklicht

Fällt auf bei Nacht...

Von Peter Lehmann (USA)

Dieses kleine Projekt soll Radfahrer, die unvorsichtigerweise im Dunkeln ohne Licht unterwegs sind, zumindest etwas schützen. Ein Stroboskop-Licht signalisiert nämlich den schnelleren motorisierten Verkehrsteilnehmern, dass sie vorsichtig sein sollen.



Diese Schaltung eignet sich für viele Anwendungen. Es gibt keine Spezialbauteile. Die Schaltung ist so einfach gehalten, dass man mit Blinkfrequenz und LEDs nach Herzlust experimentieren kann.

Prinzip

In der Schaltung von **Bild 1** sieht man vier gleiche ICs: den Spannungs-Konverter ICL7660 von Maxim. Das IC ist auch als MAX1044 erhältlich und kann – laut Datenblatt [1] mit sehr hohem Wirkungsgrad von 98 % - mit Hilfe von Kondensatoren positive Spannungen invertieren, verdoppeln, teilen oder vervielfachen.

In der Schaltung operiert IC1 als Spannungsinverter und sorgt so dafür, dass die nachfolgende Elektronik mit fast der doppelten Batteriespannung betrieben werden kann. Die nächste Stufe mit IC2 simuliert einen doppelpoligen Umschalter und legt die

Versorgungsspannung in 0,5-s-Intervallen abwechselnd an IC3 oder IC4. Ist IC3 unter Strom, wird C3 im 50-ms-Rhythmus geladen und über die LEDs D1 und D3 wieder entladen. Wenn IC4 an die Reihe kommt, passiert das Gleiche mit C4, D2 und D4. Resultat ist ein Wechselblinker, der mit hoher Blitzfrequenz für hohe Aufmerksamkeit sorgt.

Zwei hintereinander geschaltete Alkali-AAA-Zellen (Micro) versorgen die Schaltung mit einer Spannung von 3 V. Wenn sich die Zellen im Laufe der Zeit entladen, werden die LEDs erst unterhalb von 2,4 V merklich dunkler. Glücklicherweise benötigt die Elektronik nur etwa 19 mA Strom, sodass es selbst bei den kleinen AAA-Zellen gut 50 h dauert, bis sie auf 2,4 V entladen sind. Bei AA-Zellen (Mignon) wäre noch weit mehr als das Doppelte möglich.

Platine, Gehäuse und Rad

Der Autor hat passend zur Schaltung eine kleine Platine entwickelt. Das zugehörige Platinen-Layout und der Bestückungsplan sind in **Bild 2** zu sehen. Beides kann als Datei von der Elektor-Projekt-Webseite [2] zu diesem Artikel kostenlos herunter geladen werden.

Da konsequent auf SMDs verzichtet und stattdessen nur bedrahtete Bauteile verwendet wurden, ist die Bestückung der Platine sehr einfach. Man muss lediglich darauf achten, dass man die ICs nicht um 180° verdreht einlötet und die Elkos nicht verpolt.

Das für dieses Projekt vorgeschlagene Gehäuse ist das G203 von Velleman, ein dichtes Polycarbonat-Klarsichtgehäuse mit den Abmessungen 115 × 65 × 40 mm. In **Bild 2** sind an den Ecken der Platine Recht-

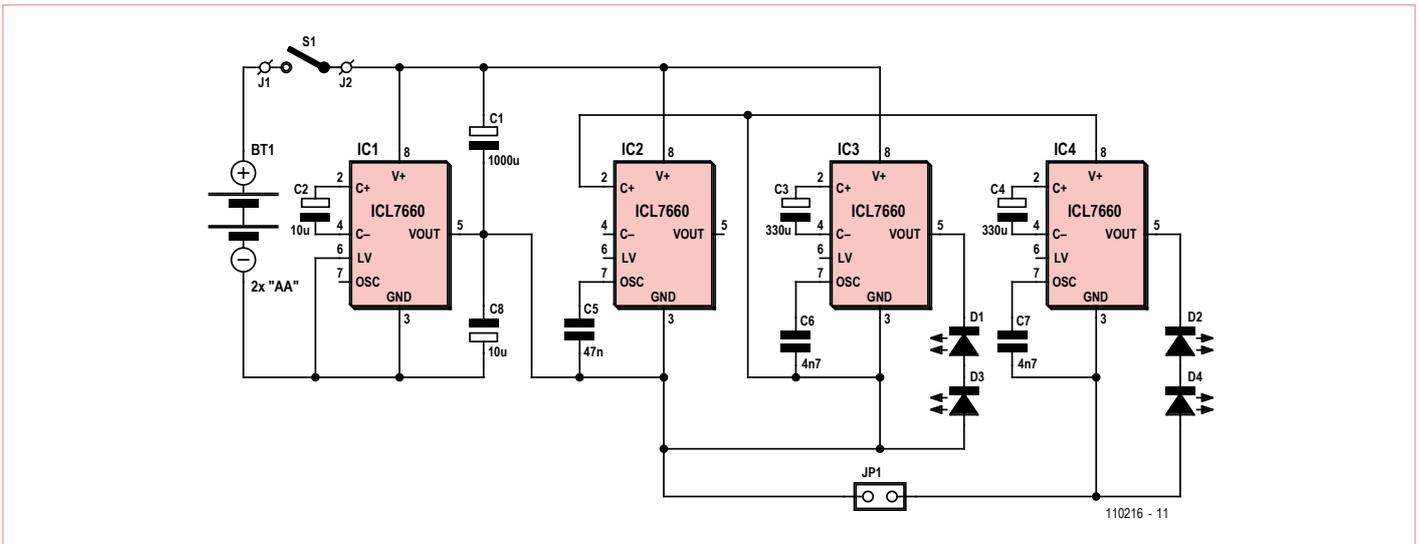


Bild 1. Die Schaltung des Blitz-Rücklichts besteht im Wesentlichen aus vier gleichen Ladungspumpen-ICs vom Typ ICL7660, die jeweils für 20 mA gut sind.

ecke zu sehen. Diese Flächen sollten entfernt werden, damit die Platine exakt in das vorgeschlagene Gehäuse passt. Die vier runden Kreisflächen indizieren die Positionen der Befestigungslöcher.

Die vier roten LEDs D1...D4 haben aus zwei Gründen einen Durchmesser von 10 mm: Erstens sind 10-mm-Typen besser zu sehen als solche mit nur 5 mm Durchmesser, die üblicherweise in kommerziellen Rückleuchten verwendet werden. Zweitens würden 5-mm-LEDs im Velleman-Gehäuse etwas verloren aussehen.

An der Seite des Gehäuses in der Nähe von IC2 und Kondensator C1 wurde ein wasserdichter Druckschalter angebracht.

Das Foto am Artikelanfang zeigt außerdem eine Möglichkeit, wie man das Blitz-Rücklicht am Fahrrad unter dem Sattel anbringen könnte. Ein Blechwinkel wurde mit einer passenden Inbus-Schraube an der Sattelstütze mit angeschraubt und das Rücklicht selbst dann mit Schrauben und selbstsichernden Muttern befestigt. Das Gehäuse hat hierzu schon zwei passende Löcher.

(110216)

Achtung! Für die Fahrradbeleuchtung existieren in einigen Ländern Vorschriften, nach denen das Fahren mit selbstgebaute Beleuchtung oder aber mit nur blinkenden Rücklichtern nicht zulässig sein kann.

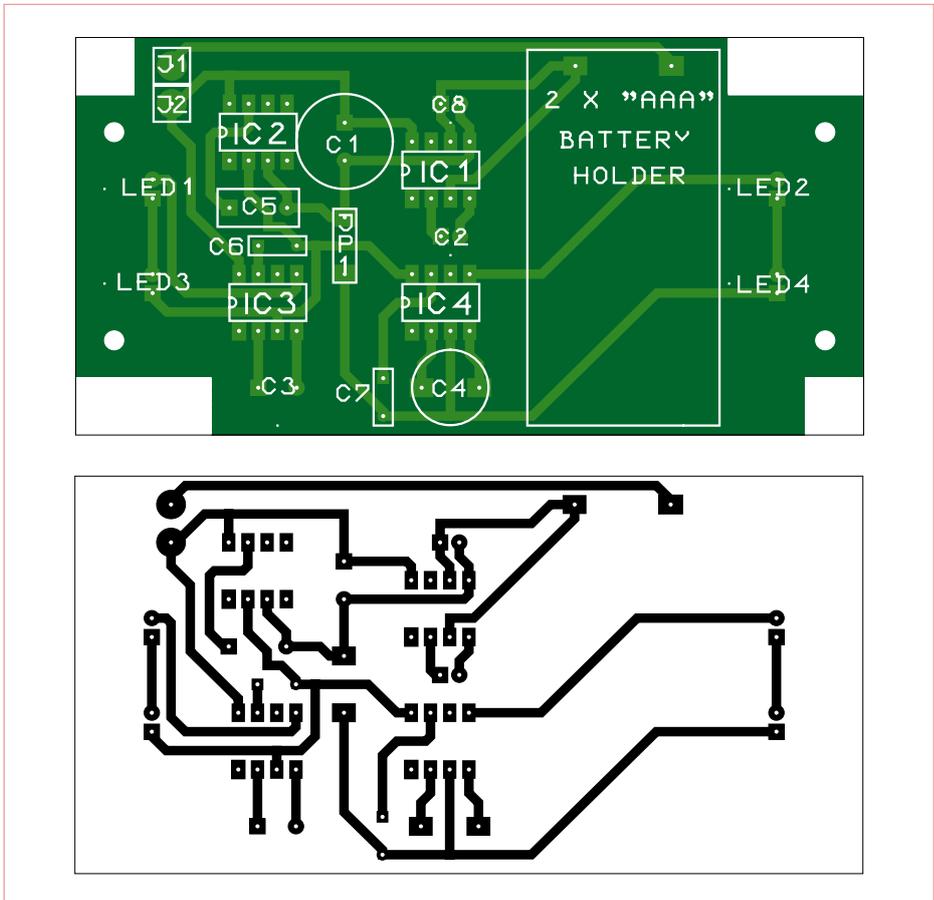


Bild 2. Bestückungsplan und Kupferseite (von oben gesehen) der vom Autor entwickelten Platine.

Weblinks und Referenzen

- [1] www.maxim-ic.com/datasheet/index.mvp/id/1017/t/al
- [2] www.elektor.de/110216

Twittern mit E-Blocks

Funknetze mit Mikrocontrollern nutzen

Von Ben Rowland (UK)

In diesem Projekt wird demonstriert, wie man eine WLAN-Karte in ein Mikrocontroller-System einbindet, um damit eine passende Webseite zu hosten – und wie man über diese sogar Twitter-Meldungen absetzen kann!



Mein örtlicher Segel-Club hat mehr als 1.000 Mitglieder. Ein Problem besteht darin, die Mitglieder über geeignete Segelbedingungen zu informieren. Es geht dabei nicht so sehr um die Windverhältnisse – dafür gibt es auch andere Informationsquellen. Viel wichtiger ist, dass ausgebildetes Rettungspersonal zur Verfügung steht, wenn sich viele Segler auf unserem See tummeln. Zur Lösung dieses Problems wurde eine Webseite vorgeschlagen, auf der die Anwesenheit der Rettungskraft in Echtzeit zu sehen sein sollte - zusammen mit lokalen Wettermeldungen und anderen fürs Segeln hilfreichen Infos. Auf dieser Webseite sollte außerdem das soziale Netzwerk Twitter [1] verlinkt sein. Auf diese Weise kann ein Mitglied andere wissen lassen, ob es in den Club kommt und ob die Bedingungen zum Segeln gut sind. Weitere Vorschläge betrafen die Integration einer Webcam, die natürlich per Servo von den Besuchern der Webseite fernsteuerbar sein sollte.

Eingesetzte Hardware

Um das Projekt zusammenzustellen und zunächst einmal auf dem Labortisch zum

Laufen zu bringen, verwendete ich die in **Bild 1** gezeigte Zusammenstellung von

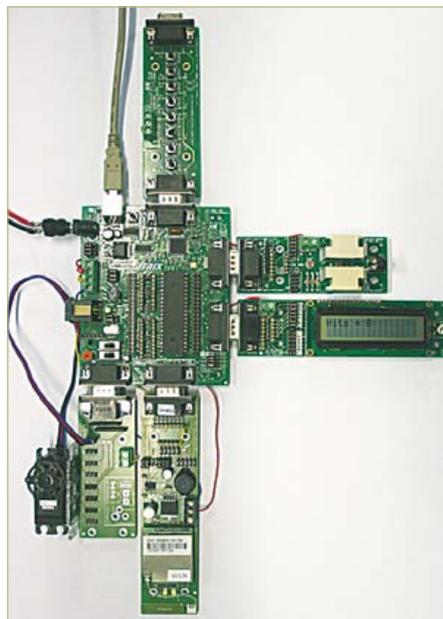


Bild 1. Der aus E-blocks-Boards zusammengestellte Prototyp.

E-blocks-Modulen. Die Kombination besteht aus einem mit PIC18F4455 bestückten Multiprogrammer vom Typ EB006, einem Sensor-Board (EB003), einem Schalter-Board (EB007), einem LCD-Board (EB005), einem Servo-Interface-Board (EB059) und einem WLAN-Board (EB069). Beim Prototypen wurde die Temperaturerfassung derart gelöst, dass ein Temperatursensor in einem Gehäuse aus rostfreiem Stahl an das Sensor-Board angeschlossen wurde. Zur Helligkeitsmessung diente der auf dem Sensor-Board befindliche LDR und die Erfassung der Windgeschwindigkeit wurde mittels eines ebenfalls auf dem Board befindlichen Potis simuliert. Als Rettungsdienst-Anwesenheitsdetektor wurde SW0 des Schalterboards genutzt. So einen Schalter könnte man beispielsweise in das von der Rettungskraft in Bereitschaft benutzte Sitzmöbel integrieren, um damit automatisch geeignete Twitter-Meldungen – ganz ohne aktives menschliches Zutun – zu generieren. Das WLAN-Board von Matrix Multimedia ist ein zentraler Teil des Systems: Damit wird der Zugang zum WLAN des Segelclubs realisiert und von daher ist es praktisch, dass das

Elektor Produkte & Service

- E-blocks-Multiprogrammer (EB006)
(Hinweis: PIC18F4455 nicht enthalten)
- E-blocks-Sensor-Board (EB003)
- E-blocks-Schalter-Board (EB007)

- E-blocks-LCD-Board (EB005)
 - E-blocks-Servo-Interface-Board (EB059)
 - E-blocks-WLAN-Board (EB069)
 - Flowcode für dsPIC/PIC24: #TEDSSI4
 - Flowcode-Software: 110388-11.zip (siehe [2])
- Siehe auch: www.elektor.de/e-blocks

Modul voll von Flowcode 4 unterstützt wird. Mit diesem Modul kann man einem bestehenden Netzwerk als Client beitreten oder selbst ein WLAN errichten. Im Host-Mode ist es schwierig, eine Internetverbindung zu realisieren, weswegen das Modul im Client-Mode betrieben wird.

WLAN-Konfiguration

Da es darum geht, mit einem E-blocks-System „ins Internet“ zu kommen, muss zunächst wie in **Bild 2** eine Verbindung mit einem bestehenden WLAN hergestellt werden. Über das WLAN-Board kann dann ein Server angeschlossen sein, der selbst Webseiten auf Anfrage aus dem Netz ausliefert, oder es

gen. Die zugehörigen Routinen finden sich in **Bild 4**. Anschließend wird das Modul mittels des Connect_To_SSID-Komponenten-Makros mit einem WLAN-Router verbunden. Das Makro benötigt zwei Parameter: den Netzwerknamen (SSID) und das Passwort. Bei einem ungesicherten WLAN kann man einen leeren String als Passwort übergeben. **Bild 5** zeigt den zugehörigen Code.

Router-Konfiguration

Nach dem Start sollte man von anderen Rechnern im lokalen Netzwerk aus eigentlich schon auf die Webseite des Systems zugreifen können. Hierzu muss man allerdings die lokale IP-Adresse der WLAN-Karte kennen, die in der DHCP-Client-Liste

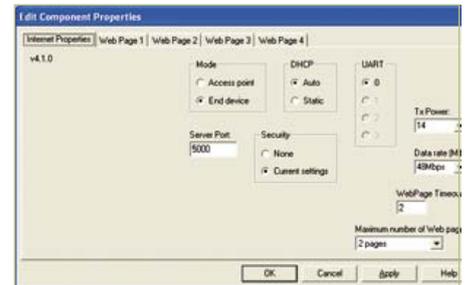


Bild 3. WLAN-Eigenschaften in Flowcode.

192.168.0.4 und der Server-Port 5000, dann ergibt sich folgender URL:

<http://192.168.0.4:5000/>

Internetverbindung

Nach der Konfiguration des WLAN-Moduls muss dem Router noch beigebracht werden, wie er Anfragen aus dem Internet an das WLAN-Modul weiterleitet. Damit kann dann von überall her auf unser System zugegriffen werden. Eine gute Unterstützung für die Konfiguration spezieller Router bietet

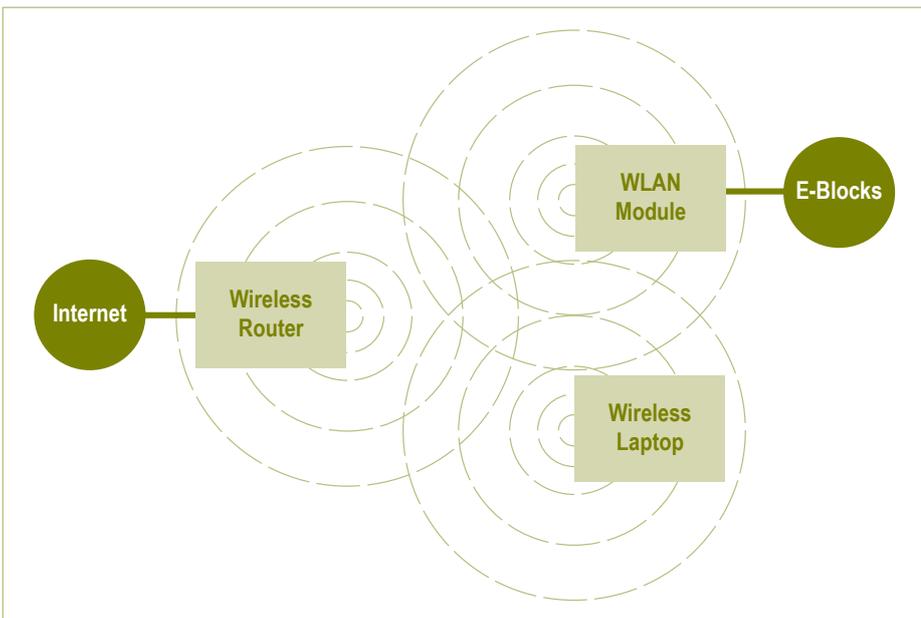


Bild 2. So wird das WLAN konfiguriert.

kann darüber als Client eine Verbindung mit einem anderen Server aufgebaut werden. Fürs Erste sollte die WLAN-Komponente daher als „end device“ konfiguriert werden. Für die meisten Anwendungen sollte eine Konfiguration in Flowcode wie in **Bild 3** passend sein. Für Internet-Zugriffe via WLAN muss das WLAN-Modul initialisiert werden. Hierzu werden nach einem Reset die in Flowcode gesetzten Eigenschaften ans Modul übertra-

des Routers aufgeführt sein muss. Der Aufruf dieser Adresse mit einem Browser sollte direkt zur Konfigurationswebseite des WLAN-Moduls führen, das üblichen Konfigurationsseiten von Routern ähnelt. Hier können die mit Flowcode gemachten Einstellungen überprüft werden. Um auf andere vom System gehostete Webseiten zugreifen zu können, muss neben der IP-Adresse auch der jeweilige Port angegeben werden. Ist die WLAN-IP-Adresse

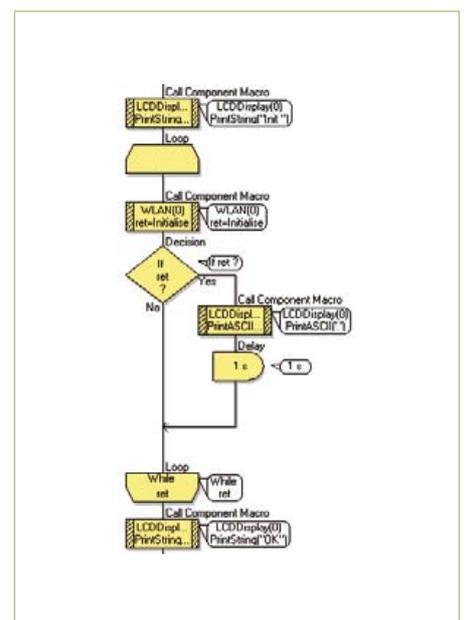


Bild 4. WLAN-Initialisierung in Flowcode.

die Webseite www.portforward.com, die Schritt für Schritt die einzelnen Einstellungen erläutert. Wer gar nicht weiter kommt, kann sich dort sogar gegen Bezahlung individuell helfen lassen. Doch normalerweise sollte ein Elektroniker mit dem Manual seines Routers klar kommen.

Um Zugriff aus dem weltweiten Internet auf unser System zu erlangen, muss man die IP-Adresse verwenden, die dem verwendeten Router vom Provider zugeteilt wird. Da es sich in der Regel um eine dynamische Adressvergabe handelt, kann man über viele Router mit Hilfe von Diensten wie www.no-ip.com oder www.dyndns.org eine pseudostatische Adresse hinbekommen. Auch das WLAN-Modul selbst unterstützt diese Art von DDNS (Dynamic Domain Name System), so dass man von außerhalb keine IP-Adressen eingeben muss, sondern einen lesbaren Domain-Namen nutzen kann. Man muss nur Dienst, Benutzernamen und Passwort eintragen und schon wird die dynamische Adresse immer frisch aktualisiert.

Webseiten in Flowcode

Als Webseiten-Inhalt kann sowohl HTML- als auch JavaScript-Code direkt in der WLAN-Komponente von Flowcode eingegeben werden. Ein Beispiel davon zeigt **Bild 6**. Die verwendeten Variablen wie „temperature“ und „wind speed“ sind direkt mit Programm-Variablen in Flowcode verknüpft. Ausgehende Variablen werden durch ein Komponenten-Makro verarbeitet und direkt in HTML mit Hilfe eines Prozentzeichens (%) gefolgt von einem Index integriert. Ein Beispiel:

```
temperature = %0
```

Eingehende Variablen werden durch ihren Index in Kombination mit der URL angesprochen – ganz ähnlich wie Variablen in der Script-Sprache PHP übergeben werden. Ein Beispiel:

```
index.htm?0=255&1=39
```

Seitenanfragen werden in Flowcode durch das regelmäßig aufgerufene Komponenten-Makro „Check_For_Page_Requests“ behandelt. Resultat ist ein vielseitig einsetzbares Mikrocontroller-System mit Internet-



Bild 7. Wettermeldungen des Segel-Clubs.



Bild 8. Steuerung der Webcam des Segel-Clubs

Anschluss, das Variablen zum und vom System transferieren kann. Beispiele so erstellter Webseiten sind in den **Bildern 7 und 8**

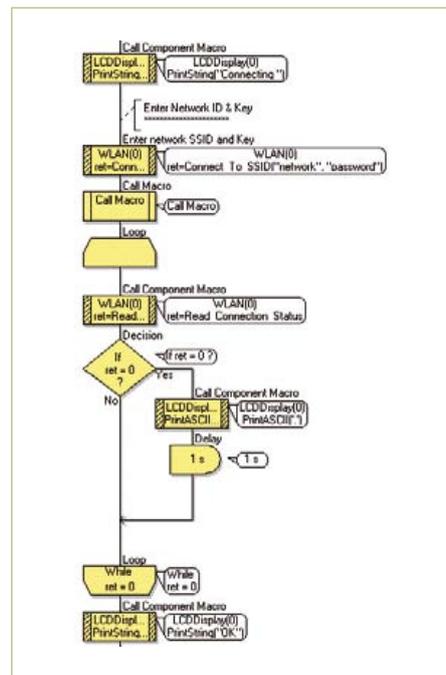


Bild 5. Verbindung zu einem existierenden WLAN.

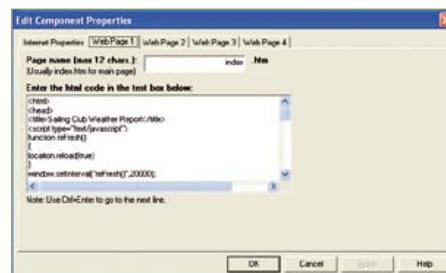


Bild 6. HTML-Seite per WLAN mit Flowcode.

zu sehen. Die Hauptseite zeigt Wetterinformationen und enthält einen Link zu einer Unterseite, mit der Besucher die Webcam steuern können. Auf der Hauptseite ist außerdem Twitter verlinkt, womit Mitglieder Anwesenheitsmeldungen an alle senden können, die diesen Feed abonniert haben.

Twitter-Links erzeugen

Die letzte Aufgabe war die Integration von Twitter als schnelles Informationssystem für alle Feed-Abonnenten. Hierzu wurde ein Twitter-Button in die Webseite integriert und dieser mit den Daten der Sensoren verknüpft. Wenn nun ein Webseiten-Besucher auf diese Schaltfläche klickt, kann er an seine Follower eine Meldung mit den aktuellen Segelbedingungen schicken. Meine Versuche, das System automatische Twitter-Meldungen über die Anwesenheit des Rettungsdienstes generieren zu lassen, zeigten sich als nicht zuverlässig genug und wurden daher vorerst nicht integriert.

Fazit

Das System arbeitet ausgezeichnet – bei mir auf dem Labortisch. Der nächste Schritt ist nun, das System unter realen Bedingungen mit einem richtigen Anemometer zu testen...

Die zugehörige Flowcode-Software ist wie immer auf der Elektor-Artikelwebseite [2] kostenlos verfügbar.

(110388)

Weblinks & Literatur

- [1] www.twitter.com
- [2] www.elektor.de/110388
- www.portforward.com
- www.no-ip.com; www.dyndns.org

Hexadoku

Sudoku für Elektroniker

Vom Urlaub zurück? Dann warten Sie sicher schon sehnsüchtig auf ein neues Hexadoku?! Und hier ist es - nach dem ganz besonderen Rätsel der Sommerausgabe wieder in gewohnter Form. Wie immer gilt: Wer uns die richtigen Ziffern zuschickt, kann einen von vier Gutscheinen gewinnen!

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

durch die dickeren schwarzen Linien) **genau einmal** vorkommen. Einige Zahlen sind bereits eingetragen, was die Ausgangssituation des Rätsels bestimmt.

Wer das Rätsel löst - sprich die Zahlen in den grauen Kästchen herausfindet - kann wie jeden Monat einen Hauptpreis oder einen von drei Trostpreisen gewinnen!

Mitmachen und gewinnen!

Unter allen internationalen Einsendern mit der richtigen Lösung verlosen wir

- einen **ELEKTOR-Gutschein** im Wert von 100 € und
- drei **ELEKTOR-Gutscheine** im Wert von je 50 €.

Einsenden

Schicken Sie die Lösung (die Zahlen in den grauen Kästchen) per E-Mail, Fax oder Post an:

Elektor – Redaktion Süsterfeldstr. 25 52072 Aachen
 Fax: 0241 / 88 909-77 E-Mail: hexadoku@elektor.de

Als Betreff bitte nur die Ziffern der Lösung angeben!
 Einsendeschluss ist der 30. September 2011!

Die Gewinner des Hexadokus Digests aus dem Juniheft stehen fest!

Die richtige Lösung ist: B18AD.

Der Elektor-Gutschein über 100 € geht an: Mads Thorup aus Kopenhagen (Dänemark).

Einen Elektor-Gutschein über je 50 € haben gewonnen: Wolfgang Kallauch, Tommy Vanhullebusch und Timon Zijngje.

Herzlichen Glückwunsch!

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

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

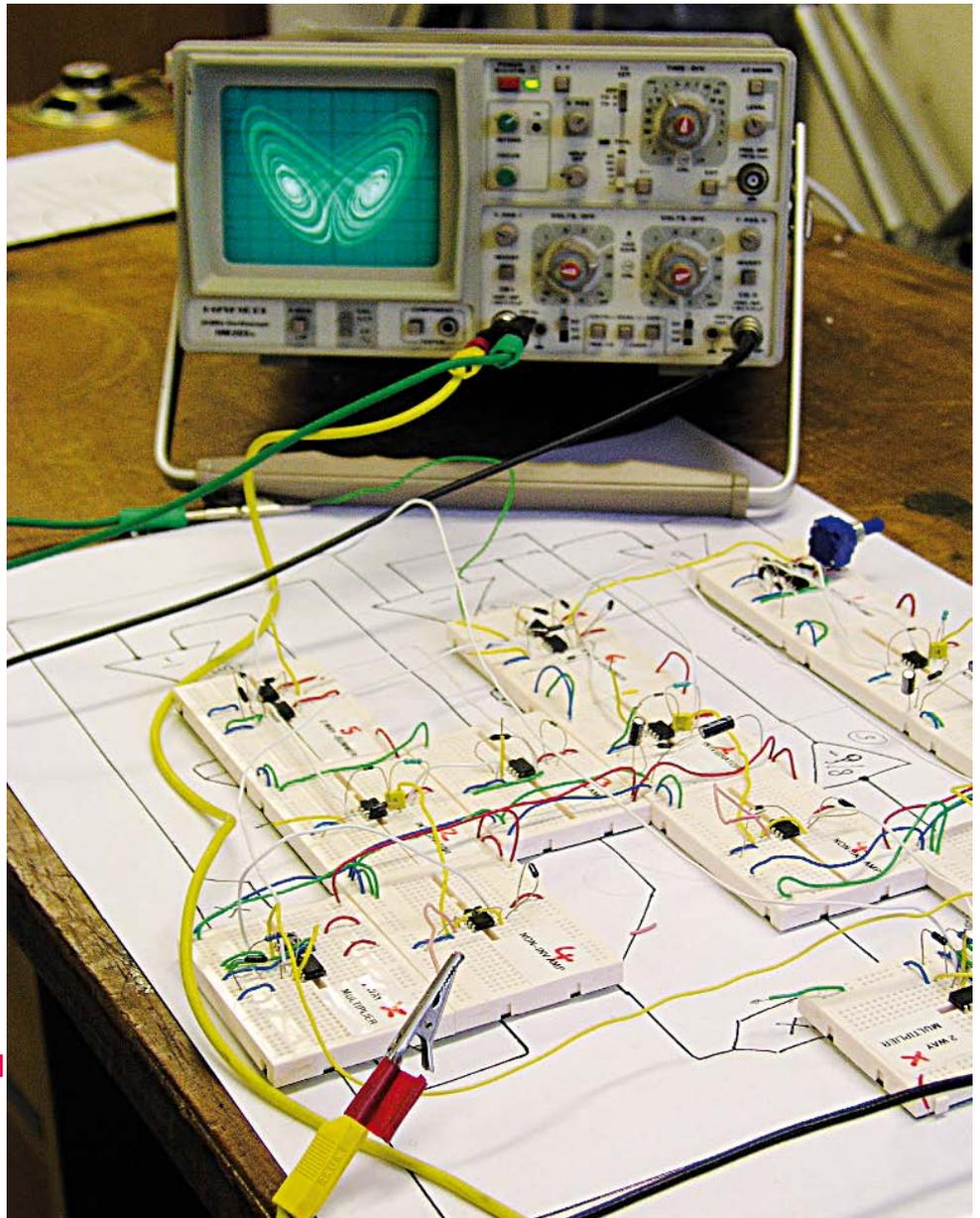
Der Rechtsweg ist ausgeschlossen. Mitarbeiter der in der Unternehmensgruppe Elektor International Media B.V. zusammengeschlossenen Verlage und deren Angehörige sind von der Teilnahme ausgeschlossen.

Die Chaos-Maschine

Analoge Computer wiederentdecken (1)

Von Maarten H. P. Ambaum und
R. Giles Harrison (Department of
Meteorology, University of Reading, UK)

Analoge Computer bieten eher tatsächliche als virtuelle Darstellungen von Modellsystemen. Es sind leistungsfähige und eindrucksvolle Rechner, die sich relativ einfach und preiswert bauen lassen. Dieser zweiteilige Retronik-Artikel hilft beim Selbstbau (und beim Verständnis) eines eigenen analogen Computers, der Lorenz-Schmetterlinge simulieren kann, die zum Symbol für die Chaos-Theorie wurden. Zunächst etwas Geschichte und Hintergrund...



Für manche Zeitgenossen ist es erstaunlich, dass bis Mitte der 1960er Jahre so etwas wie Echtzeit-Berechnungen mit digitalen Computern kaum möglich waren. Für diese Zwecke wurden damals analoge Computer eingesetzt, da sie schneller und zuverlässiger waren. Analoge Computer sind Maschinen, die dazu gebaut wurden, sich wie das zu berechnende System zu verhalten. Ein berühmtes Beispiel ist der MONIAC [1] (Bild 1) von William (Bill) Phillips aus den 1950ern, der in einem System aus Leitungen und Behältern aus transparentem Kunststoff (Acrylglas) mit gefärbtem

Wasser den Geldfluss in einer Volkswirtschaft abbilden sollte. Tatsächlich aber wurden für die meisten Zwecke elektronische Schaltungen bevorzugt. Der Terminus „analog“ bezieht sich hierbei auf das Verhalten des Computers, das dem zu simulierenden System analog sein sollte. Im Gegensatz dazu bezieht sich „digital“ auf die Methode, das Verhalten eines Systems durch eine Kette von Zahlen darzustellen, die mit einem Algorithmus transformiert werden. Anders als früher bezeichnet der Begriff „analog“ heute alles, was nicht „digital“ ist.

Moderne Analog-Computer

In den 1960er-Jahren zeichnete sich ab, dass die digitalen Computer über kurz oder lang den analogen Computern den Rang ablauen würden. Durch Fortschritte bei der Chip-Herstellung wurden digitale Computer immer zuverlässiger und so stieg deren Verfügbarkeit für Menschen und Organisationen. Dank der verbesserten Chip-Technologie kann man heute auch sehr einfach sehr genaue Analog-Computer realisieren.

Als Teil eines Projekts wurde in unserer Abteilung beschlossen, die Genauigkeit moderner Analog-Elektronik im Rahmen einer Ausstellung mit dem Lorenz-Modell zu eruiieren, das quasi als Synonym für die Unberechenbarkeit des Wetters steht (siehe hierzu das Buch „Chaos: Making a new science“ von James Gleick - eine tolle Einführung in die Chaos-Theorie und ihre Geschichte).

Der Bau eines analogen Computers war für uns eine sehr spannende und lehrreiche Erfahrung. Zunächst geht es um die weniger bekannten, aber bemerkenswerten Eigenschaften analoger Computer und im zweiten Teil dann um die Realisierung und die Simulation des Lorenz-Modells mit dieser „Chaos-Maschine“.

Schmetterlinge, Poltergeister, Mathematik

Die Lorenz-Gleichungen wurden 1963 vom Meteorologen Edward N. Lorenz zur Behandlung der Konvektion erwärmter Luft entwickelt [2]. Genau genommen handelt es sich um ein System aus drei Gleichungen, die das Verhalten der drei Variablen X, Y und Z in der Zeit beschreiben:

$$dX/dt = \sigma(Y-X)$$

$$dY/dt = \rho X - Y - XZ$$

$$dZ/dt = XY - \beta Z$$

Was diese Gleichungen mit heißer Luft zu tun haben, ist nicht so offensichtlich. Außerdem funktionieren sie nicht besonders gut.

Was Lorenz aber entdeckte war, dass sich sein Modell bei geringfügigen Veränderungen der drei änderbaren Parameter σ , ρ und β in unvorhersehbarer Weise – also chaotisch - verhält. Das war neu und unerwartet und bahnte den Weg zu einer wissenschaftlichen Revolution. Wenn man die Variablen X, Y, und Z als Werte eines Punktes im dreidimensionalen Raum interpretiert, dann ergibt das die berühmten Lorenzschen Schmetterlinge: eine fraktales Muster in 3D (siehe **Bild 2**).

Unsere Chaos-Maschine kann zwei der drei Variablen X, Y, und Z an ein Oszilloskop im XY-Modus ausgeben und so eine elektronische Variante eines zweidimensionalen Schmetterlings sichtbar machen. Man kann diese Signale auch einem Audio-Verstärker zuführen und sich auf diese Weise anhören, wie Chaos klingt. Letzteres ist sehr beeindruckend, denn die Chaos-Maschine kreischt und quietscht auf besonders bizarre Weise wie ein elektronischer Poltergeist.

Wie funktionieren Analog-Computer?

Ein elektronischer Analog-Computer löst Gleichungen mit Werten, die Spannungen in Schaltungen entsprechen. Man kann Module, die

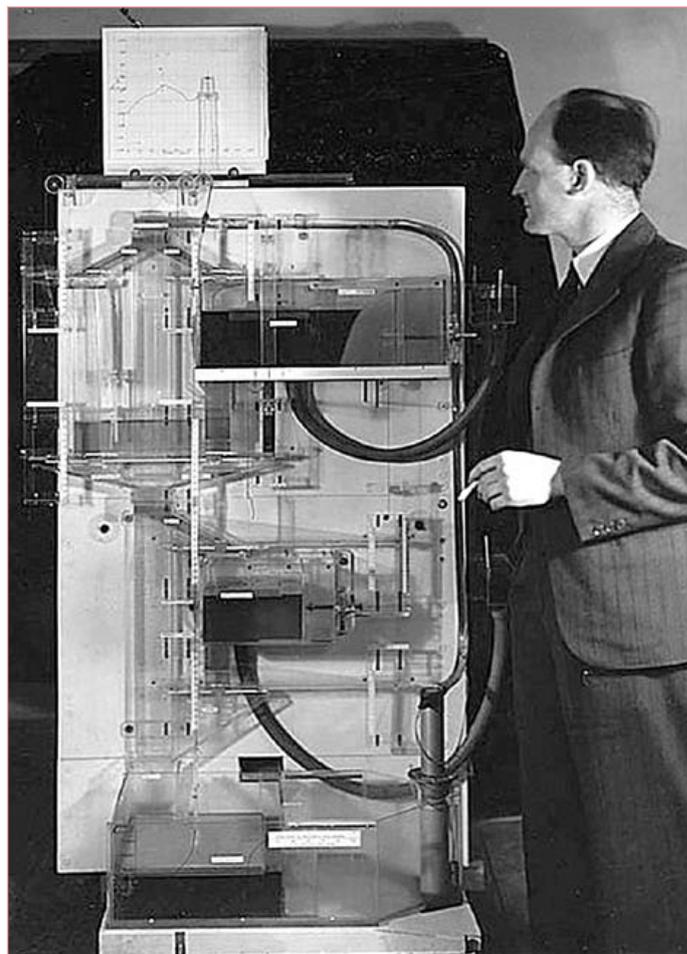


Bild 1. Professor A.W.H (Bill) Phillips wurde als Wirtschaftswissenschaftler an der LSE (London School of Economics) durch die nach ihm benannte „Phillips-Kurve“ (Zusammenhang zwischen Inflation und Arbeitslosigkeit) bekannt – und er entwickelte den MONIAC, einen hydraulischen Analog-Computer, in dem gefärbtes Wasser den Geldfluss in einer Volkswirtschaft abbildet.

Bild: Wikimedia Commons

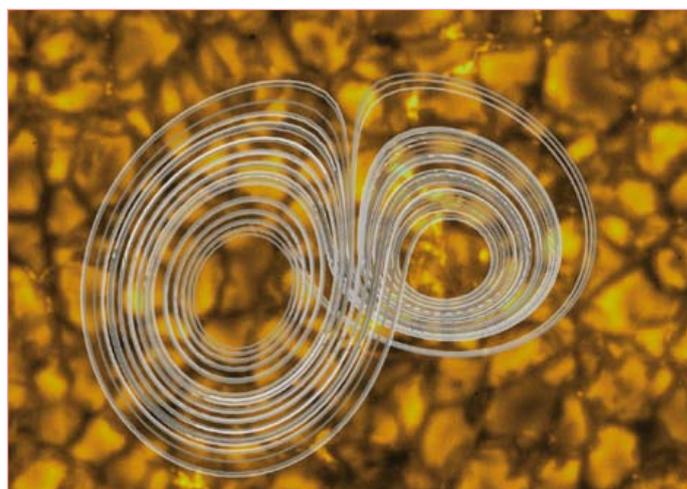


Bild 2. Der Lorenz-Schmetterling. Im Hintergrund befindet sich eine Abbildung solarer Konvektion, die zu den Lorenz-Gleichungen inspirierte.

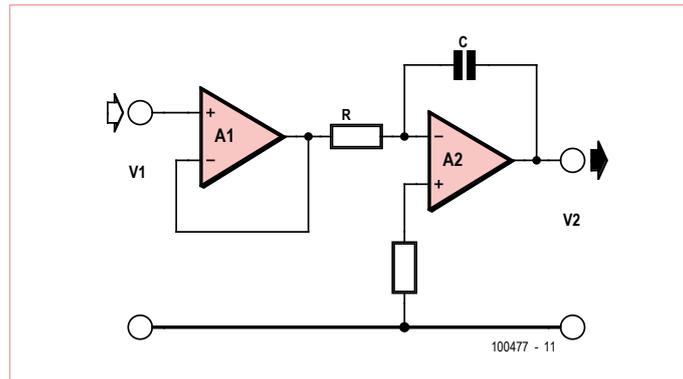


Bild 3. Basisschaltung des Integrationsmoduls.

spezielle arithmetische Operationen durchführen, miteinander verbinden. Beispielsweise hat ein Subtraktionsmodul zwei Eingänge und einen Ausgang, wobei die Spannung am Ausgang die

Differenz der Eingangsspannungen darstellt. Praktisch entspricht das einem gewöhnlichen Differenzverstärker mit einfacher Verstärkung.

Die Topologie eines Analog-Computers hat Ähnlichkeiten zum Gehirn: Die Axone entsprechen hier den Drahtverbindungen der Module, die den Zellkörpern der Neuronen entsprechen. Die Eingänge kann man sich als Dendriten denken. Das ist ein großer Unterschied zu digitalen Computern, in welchen Variablen im Speicher abgelegt und gelegentlich zur Verarbeitung in die Register eines Prozessors kopiert sowie Änderungen anschließend wieder in den Speicher kopiert werden.

Ein Digital-Computer verändert Variablenwerte nur dann, wenn die CPU das vorsieht. Die Änderungen gehen sukzessive vor sich. Beim analogen Computer hingegen bleiben die Werte immer konsistent: Bei der Funktion $a+b=c$ gilt dies immer für alle Werte der Variablen. Es gibt kein durch einen Takt vorgegebenes Arbeitstempo. Berechnungen werden fast sofort vollzogen. Beim digitalen Computer sind die Daten erst gültig, nachdem der Prozessor die Addition vollzogen hat und auch nur, solange die Werte für a und b sich noch nicht wieder verändert haben.

Zeitliche Integration ist für einen analogen Computer eine sehr „natürliche“ Angelegenheit. Auch hier bleiben Eingangs- und Ausgangsspannung immer konsistent: Der Ausgang entspricht immer dem Integral der Eingangsspannung. Es gibt da keine Taktschritte wie bei einer numerischen Integration zu berücksichtigen. Man muss sich auch nicht mit Dingen wie numerischer Instabilität oder mit Engpässen bezüglich Rechenleistung oder Speicherplatz herumschlagen. Die Basisschaltung eines Integrationsmoduls ist in **Bild 3** zu sehen. Praktisch funktioniert die elektronische Integration, indem die Eingangsspannung mit einem Opamp in einen Strom gewandelt wird, der dann einen Kondensator lädt. Die Spannung am Kondensator entspricht dann dem zeitlichen Integral der Eingangsspannung. Die elektronische Schaltung aus zwei Opamps plus Beiwerk kann sich ändernde Eingangsspannungen integrieren. Die Eingangsspannung $V_1(t)$ steuert A1 und dessen Ausgang dann den Integrator aus R, C und A2. Der Betrag der Ausgangsspannung $V_2(t)$ entspricht dem Integral der Eingangsspannung, skaliert mit $1/RC$. A1 ist ein Puffer mit einfacher Verstärkung zur Abtrennung der Signalquelle, dessen niedriger Ausgangswiderstand relativ niedrige Werte für R erlaubt. Der zusätzliche Widerstand bei A2 dient lediglich Kompensationszwecken. Auch die Stromversorgung wurde der Übersichtlichkeit halber ignoriert.

Mit Opamps lassen sich natürlich auch andere arithmetische Operationen realisieren. Die Subtraktion zweier Spannungen erledigt sehr einfach ein Differenzverstärker. Eine Multiplikation und ähnliche

Funktionen sind allerdings nicht ganz so einfach herzustellen. Analog-Computer brauchen keinen Speicher. Diese Eigenschaft macht sie äquivalent zu dem zu simulierenden System. Ein schwingendes Pendel hat ja auch keinen Zugriff auf vorherige Zustände. Falls Speicherung oder exaktes Messen

erforderlich ist, kann man immer noch einen A/D-Wandler hinter einen Analog-Computer schalten und kommt so zu einem hybriden Computer. Puristen können zur Vermeidung digitaler Technik auch analoge Bandgeräte verwenden oder aber Kurvenschreiber, womit sich Ergebnisse von Simulationsläufen ebenfalls speichern lassen. Auf diese Weise umgeht man auch Probleme wie das Aliasing, das beim Digitalisieren immer auftritt.

Analog-Computer sind nicht ganz trivial zu programmieren, denn die Programmierung ist ja schon im (Auf-)Bau des Computers enthalten. In Punkto Flexibilität können sie digitalen Computern bei weitem nicht das Wasser reichen. Während digitale Computer einfach Zugriff auf Speicher haben, entspricht bei einem analogen Rechner jeder Variablen eine Strippe, die vom Ausgang eines Moduls zum Eingang eines anderen Moduls führt. Auch wenn ein digitaler Computer aus sehr komplexer Hardware besteht, kann eine einzige Konfiguration doch für sehr viele verschiedene Aufgaben eingesetzt werden. Digitale Computer sind im Grunde nichts anderes als universelle Turing-Maschinen, weshalb man damit viele Probleme lösen kann. Ein analoger Computer hingegen ist exakt für ein Problem gestrickt beziehungsweise verdrahtet.

Ein weiterer fundamentaler Unterschied besteht darin, dass ein digitaler Computer mit einer angenäherten virtuellen Repräsentation eines Systems operiert, während sein analoges Gegenstück letztlich nichts anderes als eine direkte elektronische Simulation eines Systems darstellt. Wenn man ein Pendel mit einem analogen Computer simulieren will, dann baut man eine Elektronik, die exakt solche Schwingungen wie das Pendel produziert. Der Computer ist dann die Übertragung eines Pendels in Elektronik. Genau das macht den Charme analoger Computer aus. Wenn man an das Lorenz-System unserer Chaos-Maschine denkt, dann gibt es da nicht einmal eine physikalische Entsprechung, denn es wurde ja als Gleichungssystem definiert. Ein analoger Computer ist dann also die einzige Möglichkeit, ein physikalisch operierendes Modell eines mathematischen Systems zu realisieren.

Tempo

Wer zum ersten Mal mit analogen Computern zu tun hat, stellt sich vermutlich die Frage, wie schnell sie denn im Vergleich mit einem modernen digitalen Computer sind. So ein Vergleich ist allerdings nicht so einfach. Bei digitalen Computern ist die Arbeitsgeschwindigkeit durch die Taktfrequenz und die Lade/Speicherzeiten für Variablen determiniert. Eine Operation liegt typischerweise im Bereich

einer Nanosekunde (10^{-9}). Bei analogen Computern hängt die Geschwindigkeit hauptsächlich von den Eigenschaften der Opamps ab, da sie die geschwindigkeitslimitierenden Bauteile sind. Wichtig ist hier, wie schnell sich die Ausgangsspannung eines Opamps ändern kann. Ein Maß dafür ist die so genannte Slew-Rate. Auch im analogen Bereich liegen die Änderungsgeschwindigkeiten im Bereich weniger Nanosekunden und spielen somit für die meisten Belange keine limitierende Rolle. Doch analoge Computer rechnen nicht, sie simulieren etwas. Nach der Rechengeschwindigkeit zu fragen ist also ähnlich unpassend, wie bei einem Pendel darüber zu sinnieren, wie lange es zur Berechnung seiner Schwingungen benötigt.

Trotzdem kann man Geschwindigkeitsvergleiche machen. Zum Beispiel wird die Geschwindigkeit eines analogen Integrators von den beiden Bauteilen seines RC-Glieds abhängen. Letztlich wird ein digitaler Computer von der Taktfrequenz bestimmt, während ein analoger durch die Eigenschaften der verwendeten Bauteile determiniert ist, weshalb man andere Geschwindigkeiten durch Auswahl anderer Bauteile erreicht. Nach oben hin werden analoge Computer durch die Slew-Rate ihrer Opamps und die Dämpfung hoher Frequenzen durch die in der Praxis auftretenden Streukapazitäten begrenzt.

Für komplexere Funktionen gibt es glücklicherweise spezielle analoge Chips, die preiswert und dennoch mit hoher Genauigkeit und Temperaturstabilität bei hoher Geschwindigkeit logarithmieren, multiplizieren, dividieren oder Quadratwurzeln ziehen.

Opamps

Der Fortschritt bei elektronischen Bauteilen bringt für viele wissenschaftliche Anwendungen Vorteile. Insbesondere vereinfacht er die Realisierung analoger Computer. So ist das zentrale Bauelement in Form eines Opamps quasi standardisiert. Mit Opamps kann man sehr viele unterschiedliche mathematische Operationen erzeugen (Integration, Addition, Multiplikation, Skalierung). Opamps wurden schon in den 1940ern in militärischen Anwendungen eingesetzt - vor allem bei den Waffensystemen von Flugzeugen, auch wenn im Vietnam-Krieg noch eine Art mechanischer Analog-Computer verwendet wurde. Die Bezeichnung „operational amplifier“ (oder Opamp) tauchte 1947 auf. Der erste kommerzielle Opamp vom Typ K2-W wurde von Philbrick 1953 gefertigt. Er bestand aus zwei Dual-Tetrodenröhren (siehe **Bild 4** sowie [3] und [4]). Transistorisierte



Bild 4. Der Philbrick K2-W gilt als erster kommerzieller Opamp der Welt.

Opamps folgten in den 1960ern und der erste integrierte Opamp 1965 (laut dem „Op Amp Applications Handbook“ von Jung, das einen guten Überblick über Geschichte und Anwendung von Opamps bietet).

Der Vergleich zwischen historischen und modernen Opamps veranschaulicht die Fortschritte: Während das Modell K2-W eine Drift von ± 5 mV pro Tag aufwies, kommt der Typ OP97, der in unserer Chaos-Maschine eingesetzt wird, auf eine thermische Drift von nur $0,6 \mu\text{V}/^\circ\text{C}$ (es gibt auch „Zero-Drift“-Opamps mit weniger als $0,1 \mu\text{V}/^\circ\text{C}$). Auch der Energiebedarf ist sehr unterschiedlich: Ein K2-W benötigt ± 300 V bei 10 mA zum Betrieb sowie zusätzlich eine Heizung mit 6,3 V bei 0,6 A. Ein OP97 hingegen nimmt an ± 15 V nur 0,6 mA auf. Frühe analoge Computer kamen daher auf ein beträchtliches Volumen, benötigten viel Energie und erzeugten reichlich Abwärme.

IC-basierte Analog-Computer von heute sind recht kompakt und die Drift stellt kaum noch ein Problem dar. Außerdem sind die nötigen Opamps recht preiswert, wodurch man auch recht komplexe Funktionen mit vertretbarem

Aufwand realisieren kann. Der Einsatz zusätzlicher Opamps ist ja fast kein Kostenfaktor mehr.

Ein realer analoger Computer besteht aus einer Ansammlung unabhängiger Module, die zur Lösung eines Problems zusammengestellt wurden, die aber für andere Probleme in anderen Anordnungen wiederverwendet werden können. Das modulare Konzept vereinfacht die „Programmierung“ analoger Computer beträchtlich – man muss dazu nicht einmal die enthaltene Elektronik verstehen.

Im zweiten Teil dieses Retronik-Artikels geht es dann konkret um die Elemente, mit denen man die Chaos-Maschine bauen kann.

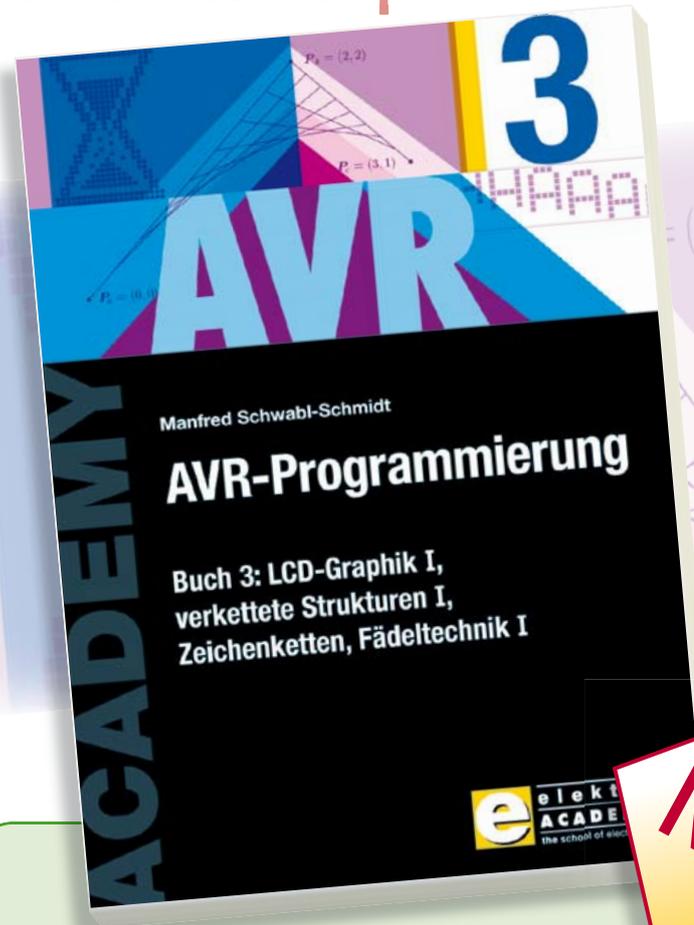
(100477)

Weblinks und Referenzen

- [1] http://en.wikipedia.org/wiki/MONIAC_Computer
- [2] <http://mathworld.wolfram.com/LorenzAttractor.html>
- [3] www.philbrickarchive.org
- [4] Philbrick K2-W, die Mutter aller Opamps, Elektor Oktober 2009.

Starke Stücke

Die ganze Welt der Elektronik in einem Shop!



NEU!

LCD-Graphik I, verkettete Strukturen I, Zeichenketten, Fädelsprachen I

AVR-Programmierung 3

Dieser dritte Band der Buchreihe zur Assembler-Programmierung von AVR-Mikrocontrollern richtet sich nicht nur an Einsteiger. Auch die C-Programmierer von AVR-Prozessoren profitieren von der Erläuterung der Besonderheiten, die es bei der Assembler-Programmierung zu beachten gilt.

Die Programmierung eines LCD-Graphikmoduls mit allen Unterprogrammen für das Zeichnen von Linien, Polylinien, Kreisen und Bézier-Kurven ist Gegenstand des ersten Abschnitts. Die Darstellung von Hieroglyphen und Text kommt dabei selbstverständlich auch nicht zu kurz, ebenso das Clipping wird ausführlich besprochen.

Nach der Erläuterung der statischen Datenstrukturen in Buch 2 folgt in diesem Buch der Einstieg in die dynamischen Strukturen. Er beginnt nach einer allgemeinen Einführung mit der einfachsten Struktur, der verketteten Liste. Der letzte Teil führt in den Selbstbau von Fädelsprachen ein, die ein äußerst personalisiertes und projektorientiertes Programmieren erlauben.

319 Seiten (kart.) • ISBN 978-3-89576-231-4 • € 46,00 • CHF 62,60



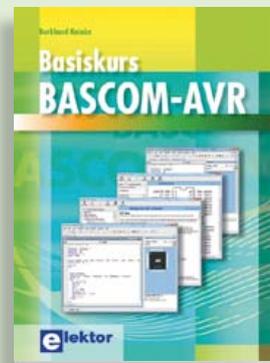
„Die Spannung steigt!“

Stromversorgungen in der Praxis

Die Elektronik bestimmt unser tägliches Leben mehr denn je – Tendenz steigend. Dabei benötigen alle elektronischen Geräte und Systeme eine gut funktionierende Stromversorgung mit spezieller Anpassung an die Betriebsbedingungen. Dieses neue Buch beschreibt die entsprechenden Möglichkeiten vom Transformator bis zum passenden Kühlkörper. Behandelt werden die wichtigsten Merkmale, Einsatzmöglichkeiten und das Betriebsverhalten von vielen unterschiedlichen Stromversorgungsgeräten.

366 Seiten (kart.) • ISBN 978-3-89576-239-0

€ 46,00 • CHF 62,60



Von Top-Autor und Entwickler Kainka

Basiskurs BASCOM-AVR

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

223 Seiten (kart.) • inkl. Software-CD

ISBN 978-3-89576-238-3 • € 39,80 • CHF 54,20



MIFARE und kontaktlose Smartcards angewandt

RFID

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

464 Seiten (kart.) • ISBN 978-3-89576-219-2
€ 56,00 • CHF 76,20



Visual Studio

C# 2010 Programmierung und PC-Anbindung

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

349 Seiten (kart.) • ISBN 978-3-89576-244-4
€ 44,00 • CHF 59,90



Callbacks, Fließkommafunktionen und BCD-Arithmetik

Systemprogrammierung II für AVR-Mikrocontroller

Mit jeder neuen Controllergeneration nimmt die Anzahl der bereitgestellten Interrupts zu. Eine Möglichkeit Interrupts zu organisieren, um dieser Zunahme Herr zu werden, nämlich der Einsatz von Callbacks und Callback Chains, wird im ersten Teil des Buches vorgestellt. Es folgt die Ergänzung und Erweiterung der Fließkommaarithmetik aus dem ersten Buch der Reihe „Systemprogrammierung“ mit den wichtigsten mathematischen Funktionen.

319 Seiten (kart.) • ISBN 978-3-89576-243-7
€ 42,00 • CHF 57,20



Mikrocontroller selber programmieren

Das MSP430 Mikrocontroller Buch

Moderne Mikrocontroller werden immer leistungsfähiger und können vielfältige Aufgaben übernehmen, für die vor wenigen Jahren noch ein kompletter Computer nötig gewesen wäre. Dieses Buch eröffnet einen schrittweisen Einstieg in die Welt der Mikrocontrollerprogrammierung und führt mit ausführlichen Anwendungsbeispielen in die Fähigkeiten dieser außergewöhnlichen Prozessorfamilie ein.

296 Seiten (kart.) • ISBN 978-3-89576-236-9
€ 42,00 • CHF 57,20



Grundlagen – Reparatur – Modifikationen

Gitarren-Röhrenverstärker

Obwohl subjektive Qualitätskriterien für „guten Sound“ rational kaum fassbar sind, sind die technischen Zusammenhänge eindeutig. Dieses Buch deckt diese Zusammenhänge in Theorie und Praxis auf. Es richtet sich an den technisch ambitionierten Musiker ebenso wie an denjenigen, der sich mit dieser außerordentlich interessanten Verstärkertechnik näher befassen möchte. Neben unverzichtbaren Funktionsbeschreibungen findet der Leser Anleitungen zur Fehlersuche und Instandsetzungstechnik.

253 Seiten (kart.) • ISBN 978-3-89576-241-3
€ 42,00 • CHF 57,20

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

www.elektor.de

Elektor-Verlag GmbH

Süsterfeldstr. 25

52072 Aachen

Tel. +49 (0)241 88 909-0

Fax +49 (0)241 88 909-77

E-Mail: bestellung@elektor.de

elektor

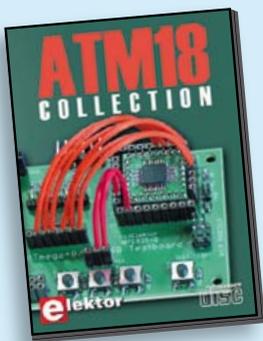


Kompletter Elektor-Jahrgang 2010 auf DVD

Elektor-DVD 2010

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

ISBN 978-90-5381-267-9 • € 27,50 • CHF 37,40



Mikrocontroller-Programmierung leicht gemacht

ATM18-Collection

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

ISBN 978-0-905705-92-7 • € 29,50 • CHF 40,20



NEU!

USB/RS485-Konverter

(Elektor Juni 2011)

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

Bestückte und getestete Platine

Art.-Nr. 110258-91 • € 24,95 • CHF 34,00



Pico C

(Elektor April 2011)

HF-Entwickler müssen häufig parasitäre Kapazitäten im Picofarad-Bereich aufspüren. Leider werden auch bessere Digitalmultimeter ziemlich unpräzise, wenn es um das Messen solcher kleiner Kapazitäten geht. Gewöhnlich liegt der kleinste Messbereich bei 2.000 pF! Mit dem „Pico C“ klappt das Messen besser, er zeigt die Picofarads sogar mit Nachkommastelle an!

Bausatz mit allen Bauteilen inkl. 'Elektor Project Case', Platine, progr. Controller und LCD

Art.-Nr. 100823-71 • € 82,50 • CHF 112,20



OBD-2-Wireless

(Elektor April 2011)

Die preiswerteste Lösung für die Autodiagnose ist ein OBD-2-Interface zum Anschluss an einen Notebook/PC mit Diagnosesoftware. Allerdings ist eine solche Kabellösung nicht wirklich praktisch. Ein autonomer OBD-Tester ist hingegen wesentlich teurer und nicht so vielseitig wie eine PC-gestützte Diagnose. Eine interessante Alternative ist dieses kabellose OBD-Interface mit Funkverbindung zum PC (via ZigBee oder Bluetooth).

(ZigBee-)Bausatz mit allen Bauteilen inkl. Gehäuse + ZigBee-USB-Stick

Art.-Nr. 100872-71 • € 124,95 • CHF 170,00

(Bluetooth-)Bausatz mit allen Bauteilen inkl. Gehäuse

Art.-Nr. 100872-72 • € 124,95 • CHF 170,00



SatFinder

(Elektor März 2011)

Wer regelmäßig eine Satellitenschüssel neu ausrichten muss (zum Beispiel, wenn ein Camper oder ein Boot damit ausgestattet ist), steht immer vor dem Problem, die jeweilige Satellitenposition zu ermitteln. Der GPS-basierte SatFinder verfügt über eine Datenbank mit den beliebtesten TV-Satelliten und berechnet mit Hilfe der GPS-Standortdaten alles, was man zum Ausrichten der Antenne auf den Satelliten braucht.

Bausatz inkl. progr. Controller, Display und Platine (europäische Version)

Art.-Nr. 100699-71 • € 79,95 • CHF 108,80

September 2011 (Nr. 489) €

+++ Das Lieferprogramm zu dieser Ausgabe finden Sie auf www.elektor.de +++

Juli/August 2011 (Nr. 487/488)

Akkutester

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

Timer für 2-4-6 Stunden

110219-41 Progr. Controller PIC12F675 DIL8 9,95

Morseuhr

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

Elex-Experimentierplatine

ELEX-1 Experimentier-Platine Elex-1 5,50

ELEX-2 Experimentier-Platine Elex-2 9,95

ELEX-4 Experimentier-Platine Elex-4 17,95

Baustellenampel für den Modellbau

110203-41 Progr. Controller ATtiny13 DIP8..... 9,95

Berührungsloses Thermometer

100707-1 Platine 22,95

100707-41 Progr. Controller PIC16F876A DIL28 15,00

R8C/13 spricht CAN / Mehr Portleitungen für den R8C/13

050179-91 R8C/13-Starterset (aufgebautes Board + Software-CD) .. 13,95

Schweißlicht für die Modellbahn

110085-41 Progr. Controller PIC10F200-I/P DIP8..... 9,95

Jogging-Timer

110160-41 Progr. Controller ATtiny44-20PU DIP14 9,95

Tochterblitz für Unterwasserkamera

100584-41 Progr. Controller PIC12F675i/p DIL8 9,95

RGB-Solarlampe

100581-41 Progr. Controller ATtiny13 DIL8 9,95

WAV-Türgong

110080-41 Progr. Controller ATmega328p DIL28 12,95

Juni 2011 (Nr. 486)

Hier kommt der Bus (6)

110258-1 Platine Experimental-Knoten 5,95

110258-1C3 ... 3 x Platine Experimental-Knoten..... 12,95

110258-91 USB/RS485-Konverter (bestückt und getestet) 24,95

Positionsbestimmung mit ATM18

071035-91 SMD-bestückte Platine mit allen Bauteilen
(ATM18-Controllermodul) 12,50

071035-92 SMD-bestückte Platine mit allen Bauteilen
(ATM18-Testboard)..... 37,50

071035-93 SMD-bestücktes LCD-Board mit allen Bauteilen
und Stiftleisten inkl. 4x20-LCD 29,95

E-blocks: RC5 in Flowcode

EB007 Switch-Board 17,85

EB058 Grafisches LCD-Display..... 81,85

EB060 RC5-Infrarot-Board 37,20

EB064 dsPIC/PIC24-Multiprogrammer 119,00

Mai 2011 (Nr. 485)

Fernbedienung per Handy

071035-72 Bausatz mit Relais-Platine
inkl. allen Bauteilen und Relais 49,95

071035-91 SMD-bestückte Platine mit allen Bauteilen
(ATM18-Controllermodul) 12,50

071035-92 SMD-bestückte Platine mit allen Bauteilen
(ATM18-Testboard)..... 37,50

071035-93 SMD-bestücktes LCD-Board mit allen Bauteilen
und Stiftleisten inkl. 4x20-LCD 29,95

071035-95 SMD-bestückte Porterweiterungs-Platine 16,95

Elektor Proton Robot

110263-71 Komplettbausatz
(Rumpf + Kopf + Audio + Greifer + PIC-Controller)..... 1249,00

110263-72 Komplettbausatz
(Rumpf + Kopf + Audio + Greifer + AVR-Controller) 1249,00

110263-91 Komplett aufgebauter und getesteter Roboter
(mit PIC-Controller) 1699,00

110263-92 Komplett aufgebauter und getesteter Roboter
(mit AVR-Controller) 1699,00

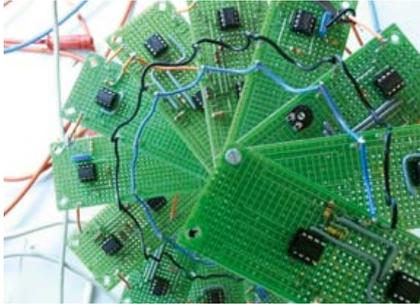
Bestseller

Bücher	1		Stromversorgungen in der Praxis ISBN 978-3-89576-239-0 € 46,00 CHF 62,60	
	2		Basiskurs BASCOM-AVR ISBN 978-3-89576-238-3 € 39,80 CHF 54,20	
	3		Embedded Linux in der Mikrocontrollertechnik ISBN 978-3-89576-208-6 € 44,00 CHF 59,90	
	4		RFID ISBN 978-3-89576-219-2 € 56,00 CHF 76,20	
	5		C# 2010 Programmierung und PC-Anbindung ISBN 978-3-89576-244-4 € 44,00 CHF 59,90	
Bausätze & Module	CD- & DVD-ROMs	1		ECD 6 ISBN 978-90-5381-258-7 € 29,50 CHF 40,20
		2		Elektor-DVD 2010 ISBN 978-90-5381-267-9 € 27,50 CHF 37,40
		3		The Audio Collection 3 ISBN 978-90-5381-263-1 € 21,50 CHF 29,30
		4		ATM18-Collection ISBN 978-0-905705-92-7 € 29,50 CHF 40,20
		5		Wireless-Toolbox ISBN 978-90-5381-268-6 € 32,50 CHF 44,20
1		USB/RS485-Konverter Art.-Nr. 110258-91 € 24,95 CHF 34,00		
2		Pico C Art.-Nr. 100823-71 € 82,50 ... CHF 112,20		
3		Netzer Art.-Nr. 100552-91 € 59,95 CHF 81,60		
4		Bluetooth für OBD-2-Analyser NG Art.-Nr. 090918-71 € 29,95 CHF 40,80		
5		OBD-2-Wireless (Bluetooth) Art.-Nr. 100872-72 € 124,95 ... CHF 170,00		

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



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



Retro-Elektronik: Chaos Generator (2)

Im Anschluss an die Retronik-Rubrik in dieser Ausgabe, die sich mit Analogrechnern und einem Chaos-Generator beschäftigt, werden wir im nächsten Heft über die Ergebnisse der praktischen Umsetzung berichten. Der Chaos-Generator wurde mit Operationsverstärkern auf Universal-Platinen aufgebaut und funktioniert wunderbar – wir haben noch nie so seltsame Signale auf einem Oszilloskop gesehen. Die XY-Darstellung lässt alles Mögliche erkennen - von Seepferdchen über Schmetterlinge bis zu modernen Management-Modellen. Auch die akustische Wiedergabe der Chaos-Signale bietet Überraschungen – und viel Raum für Interpretationen...



Platino

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



Der Traum vom elektrischen Fliegen...

...ist längst kein Traum mehr. Das erste für den Luftverkehr zugelassene Flugzeug mit Elektroantrieb ist schon seit 2004 in Produktion, und der Airbus-Mutterkonzern EADS präsentierte in diesem Jahr das vollelektrische Antriebssystemkonzept „VoltAir“ für künftige Verkehrsflugzeuge.

Wir berichten über den aktuellen Stand in diesem noch wenig beachteten Bereich der Elektromobilität und über den für den 25. September 2011 geplanten Ökoflugzeug-Wettbewerb Green Flight Challenge, der von der NASA mit einem Preisgeld von 1,65 Millionen Dollar unterstützt wird.

Elektor Oktober erscheint am 21. September 2011.

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

Änderungen vorbehalten!



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

Bestellannahme und Bestellservice: Peter Custers
E-Mail: bestellung@elektor.de
Tel. +49 241 88 909-66

Geschäftszeiten

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

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

Einzelheft

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

Jahresabonnement-Standard

Deutschland	€ 77,50
Österreich, Belgien, Luxemburg	€ 84,50
Schweiz	€ 97,50
Andere Länder	€ 102,50

Jahresabonnement-PLUS

Deutschland	€ 95,00
Österreich, Belgien, Luxemburg	€ 102,00
Schweiz	€ 115,00
Andere Länder	€ 120,00

Probeabonnement

Alle Länder (zzgl. Porto)	€ 14,90
---------------------------	---------

Studentenabo-Standard

Deutschland	€ 62,00
Österreich	€ 67,60
Schweiz	€ 78,00

Studentenabo-PLUS

Deutschland	€ 79,50
Österreich	€ 85,10
Schweiz	€ 95,50

Upgrade zum Abo-PLUS

Alle Länder	€ 17,50
-------------	---------

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

Änderungen und Irrtümer vorbehalten.

Bankverbindungen

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

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

Elektor-PCB-Service

Die Adresse für Platinen, Prototypen und Multilayer

Möchten Sie Ihre selbst entworfene Platine schnell und zuverlässig geliefert bekommen? In Kleinserie und dabei zu einem unschlagbar günstigen Preis? Bestellen Sie jetzt Ihre individuelle Platine beim Elektor-PCB-Service!



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

www.elektorpcbservice.de

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

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

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

Hier ist meine Anschrift:

Firma _____

Vorname _____

Name _____

Straße, Nr. _____

PLZ, Ort _____

Land **DE**

Kunden-Nr. _____

E-Mail _____

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

Innerhalb
Deutschlands
kein Porto
nötig!

Hier ist meine Anschrift:

Firma _____

Vorname _____

Name _____

Straße, Nr. _____

PLZ, Ort _____

Land **DE**

Kunden-Nr. _____

E-Mail _____

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

Hier ist meine Anschrift:

Firma _____

Vorname _____

Name _____

Straße, Nr. _____

PLZ, Ort _____

Land **DE**

Kunden-Nr. _____

E-Mail _____

Antwort

Elektor-Verlag GmbH
Süsterfeldstr. 25
52072 Aachen

Innerhalb
Deutschlands
kein Porto
nötig!



Elektor Electronic Toolbox

Erhältlich im App Store

Umfangreiche Elektroniker-App von Elektronikern für Elektroniker

Die neue „Elektor Electronic Toolbox“-App ist ganz auf die Belange von Elektronikern zugeschnitten. 29 Einzelprogramme/Anwendungen können über eine übersichtliche Oberfläche ausgewählt werden.

Sehr hilfreich im Entwickleralltag sind die Datenbanken für die Bauteilgruppen Bipolar-Transistoren, FETs, Triacs, Thyristoren, Dioden und ICs. Ein Bauteil kann anhand der Typenbezeichnung kinderleicht ausgewählt werden – eine Internetverbindung ist nicht notwendig. Insgesamt sind über 45.000 Bauteile in den Datenbanken verzeichnet. Hinzu kommt eine Spezialdatenbank, in der die Belegung einer Vielzahl von Steckverbindern aus den Bereichen Audio & Video, Computertechnik und Telefon nachgeschlagen werden kann. Nützlich sind auch die interaktiven Bauteilwert-Kalkulatoren.

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

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



INSERENTENVERZEICHNIS SEPTEMBER 2011

Beta Layout	www.pcb-pool.com/fitsornot	2
Embedded Projects	www.embedded-projects.net	23
Eurocircuits	www.eurocircuits.de	67
Jackaltac	www.jackaltac.com	11
LPKF	www.lpkf.de/prototyping	27
LeitOn	www.leiton.de	23
Markt		61
MikroElektronika	www.mikroe.com	3
NAGY Messsysteme	www.nagy-syssysteme.de	23
Reichelt	www.reichelt.de	88

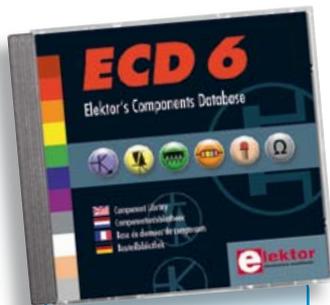
ECD 6

Elektor's Components Database



Neue Edition mit noch mehr Bauteilen

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



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

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

Weitere Infos & Bestellung unter

www.elektor.de/shop



Elektor-Newsletter E-weekly jetzt gratis abonnieren!

Jeden Freitagmorgen erscheint E-weekly, der kostenlose Newsletter von Elektor. Unsere E-weekly-Redakteure halten Sie mit neuesten und interessanten Meldungen, Tipps & Trends aus der Welt der Elektronik auf dem Laufenden. Außerdem werden Sie schnell und umfassend über aktuelle Elektor-Projekte (Nachlesen & Updates) sowie über das umfangreiche Elektor-Sortiment und spezielle Angebote als Erster informiert.

Daneben erhalten E-weekly-Abonnenten exklusiv vollen Zugang zu allen Newsberichten und zu unserem Forum, das von kompetenten Elektronik-Experten moderiert wird.



Klicken Sie jetzt auf **www.elektor.de/newsletter!**



1-36VDC
0,05-3 A

NSP 3630
99,95

Labor-Kompaktnetzgerät

- 4-stelliges, hintergrundbeleuchtetes LC-Display
- hochwertiger Drehimpulsgeber
- präzise Einstellung und Anzeige von Spannung und Strom mit einer Genauigkeit: $\pm 0,5\%$ bei 5 V sowie $\pm 0,5\%$ bei 1 A

nur 7 cm

Programmierbares Labornetzgerät

- Fernsteuer-Anschluss für Spannung und Strom, Lastausgang
- Schutz vor Kurzschluss, Überlast, Überspannung und Übertemperatur
- zwei LED-Anzeigen
- drei, frei programmierbare Presets
- thermisch gesteuerter Lüfter
- Restwelligkeit: 5/50 mV

1-30VDC
0-15 A

HCS 3302
145,00



Markenqualität - Top-Service - günstige Preise!



Katalog kostenlos!
 Tagesaktuelle Preise:
www.reichelt.de

Industrie-Schaltnetzteile

Leistung: 60 W

Eingangsspannung: 85 – 264 VAC, 120 – 370 VDC
 Einschaltstoßstrom: 60 A
 Wirkungsgrad: > 73%
 Maße: 97 x 38 x 159 mm



Bestellnummer:	Ausgang	Last (max.)	URW
SNT MW60-05	5V	12,0 A	120 mV
SNT MW60-12	12V	5,0 A	120 mV
SNT MW60-15	15V	4,0 A	150 mV
SNT MW60-24	24V	2,5 A	150 mV
SNT MW60-DA	5/12V	4,0 / 3,0 A	75 / 150 mV
SNT MW60-DB	5/24V	3,0 / 1,8 A	75 / 150 mV
SNT MW60-TA	5/12/5V	5 / 2,5 / 0,5 A	100 mV



Hutschienen-Schaltnetzteile

Serie MDR-60 W

Eingangsspannung: 85 – 264 VAC
 120 – 370 VDC
 Einschaltstoßstrom: 60 A (230 VAC)
 Wirkungsgrad: 78 – 87 %
 Maße: 40 x 90 x 100 mm



Bestellnummer:	Ausgang	Last	URW
SNT MW-MDR60-05	5V max.	10,0 A	80 mV
SNT MW-MDR60-12	12V max.	5,0 A	120 mV
SNT MW-MDR60-24	24V max.	2,5 A	150 mV
SNT MW-MDR60-48	48V max.	1,2 A	200 mV



Leistung: 100 W

Eingangsspannung: 100 - 120 / 200 - 240 VAC
 248 - 370 VDC
 Einschaltstoßstrom: 60 A
 Wirkungsgrad: > 76%
 Maße: 97,8 x 37,8 x 197 mm



Bestellnummer:	Ausgang	Last (max.)	URW
SNT MW100-05	5V	20,0 A	100 mV
SNT MW100-7,5	7,5V	13,5 A	125 mV
SNT MW100-12	12V	8,5 A	125 mV
SNT MW100-15	15V	6,7 A	125 mV
SNT MW100-24	24V	4,5 A	150 mV
SNT MW100-48	48V	2,2 A	150 mV



Leistung: 480 W

PFC-Schaltkreis nach EN 61000-3-2
 Eingangsspannung: 180 – 264 VAC
 250 – 370 V DC
 Einschaltstoßstrom: 36 A (230 VAC)
 Wirkungsgrad: 89 %
 Maße: 227 x 125 x 100 mm,
 2,4 kg



Bestellnummer:	Ausgang	Last	URW
SNT MW-DRP480-24	24V max.	20 A	120 mV
SNT MW-DRP480-48	48V max.	10 A	120 mV



Unterbrechungsfreie Stromversorgungen

APC Back-UPS ES

Steckdosenleiste mit 4 Überspannungsschutz- und 4 Netzausfallschutz-Steckdosen.
 Ausgangsleistung 240 bzw. 330 Watt (400 bzw. 550 VA).

Eine wartungsfreie, versiegelte Bleibatterie mit suspendiertem Elektrolyt, auslaufsicher, sorgt für eine übliche Betriebsdauer von 18 bzw. 14 Minuten bei halber Last (5 bzw. 3,4 Minuten bei voller Last)



APC BKES400	64.20	Leistung	400 VA
APC BKES550	71.95	550 VA	
APC BKES700	81.95	700 VA	



APC Back-UPS Pro

- Stromsparfunktionen (Master controlled Outlets)
- Line-Interactive mit autom. Spannungsregelung – AVR
- multifunktionale LCD-Status- und Steuerungskonsole zur Abfrage und Anzeige von Betriebsarten, Systemparametern und Alarmen in Form von Texten und Übersichtsschaltbildern
- Überspannungsschutz für Telefon, Fax, Modem, DSL
- autom. Abschaltung der Stromversorgung ungenutzter Geräte zur Senkung des Stromverbrauchs



APC BKRS550 LCD	116.95	Leistung	330 Watt 550 VA
APC BKRS900 LCD	164.95	540 Watt	900 VA
APC BKRS1200 LCD	249.95	720 Watt	1200 VA
APC BKRS1500 LCD	297.50	865 Watt	1500 VA

