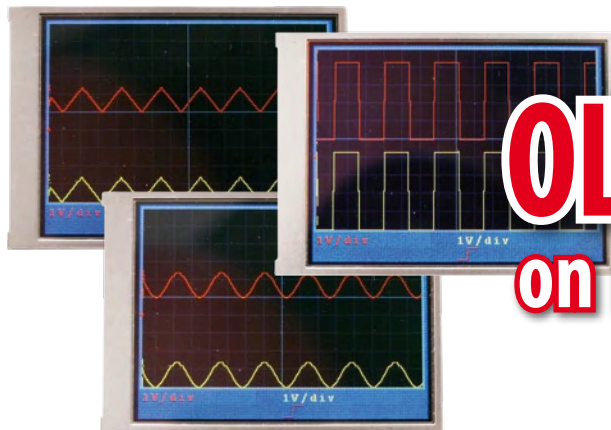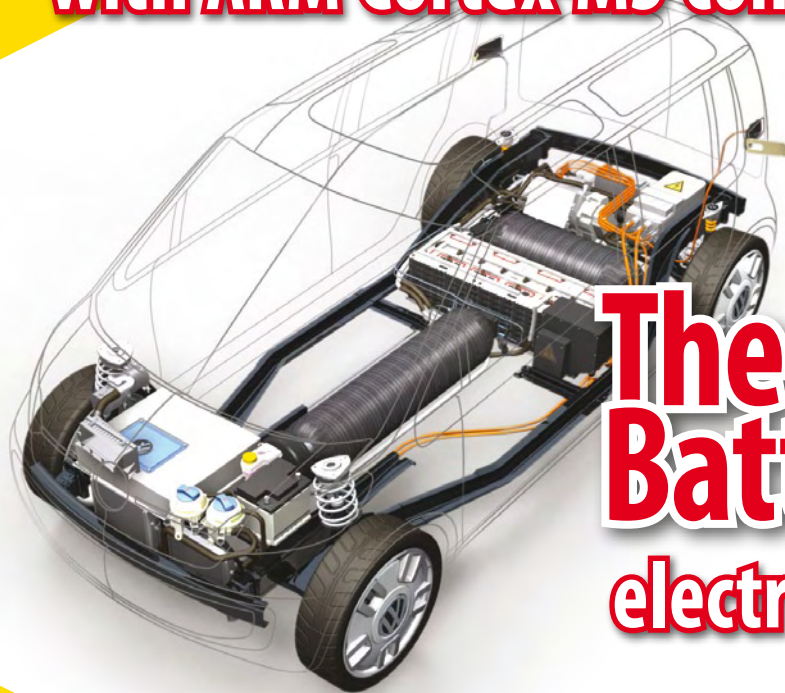# elektor
electronics & microcontrollers

OLED 'scope
on R32C experimenter's board

OBD2 Analyzer
with ARM Cortex-M3 controller

UP DXM1 ESC
DOWN Open Handheld v1.0.2 OK

powered by DXM

OBD2 open handheld

The Road to Battery Power
electric vehicles come of age

Power Charger for ElektorWheelie
ATM18 Mini Chess Computer
GPS Datalogger

09

0 74470 24965 8

Electromobile

This month's focus is on automotive and I believe the subject is covered from a number of interesting angles. The main project, for example, is our new pocket-size OBD2 Analyzer with an extensive range of features. Fully DIY to Elektor standards, the instrument should be compatible with all modern cars having OBD connectivity. The built-in analyzer will faithfully read and display engine data and any error codes you or the mechanic will want to clear by matching repair work on the vehicle. A graphic display ensures a clear readout with plain-language descriptions.

Those of you keen to use Google Earth, a regular USB memory stick and your PC to trace the actual route you have followed by car, bike or on foot will find the GPS Datalogger on page 30 an awesome project, if only because it's modular and the software is ever so easy to adapt to personal requirements. RU ready boots (or wheels)… start logging!

The article on technological developments in electric cars concentrates on energy sources available (and desirable) to power these vehicles. Which sources are likely to be used by future generations and what are all these car and battery manufacturers doing in their labs and on their test tracks? Our ElektorWheelie self-balancing electro scooter is covered in two articles, one on the mechanical assembly of the kit parts and your careful first attempts to ride the vehicle, and another on a symmetrical battery charger developed as a premium alternative to the charger normally supplied with the kit.

Of course, it's not all vehicles and batteries in this issue. With all 'wheels' safely parked or connected to a charger, why not play a game of chess against our ATM18 Mini Chess Computer, or explore how an R32C microcontroller drives an OLED display.

Finally, I'm happy to announce a new four-page section in Elektor called *E-Labs Inside*. These pages are firmly positioned in the center of the magazine, just like the real Elektor Lab is at the center of our publications and many of our products. *E-Labs Inside* reports on what our lab staff have run into during the month — you can expect to read about techie problems and workarounds, equipment to drool on, gizmos, burnt PCBs, techno-gossip, tips and tools of the trade. Happy reading!

Jan Buiting
Editor

Development of electric vehicles continues apace and the enormous potential is clear. The key to the success of electric vehicles lies, contrary to early expectations, in lithium-ion cells rather than in fuel cells. This article has an overview of current and future technologies.
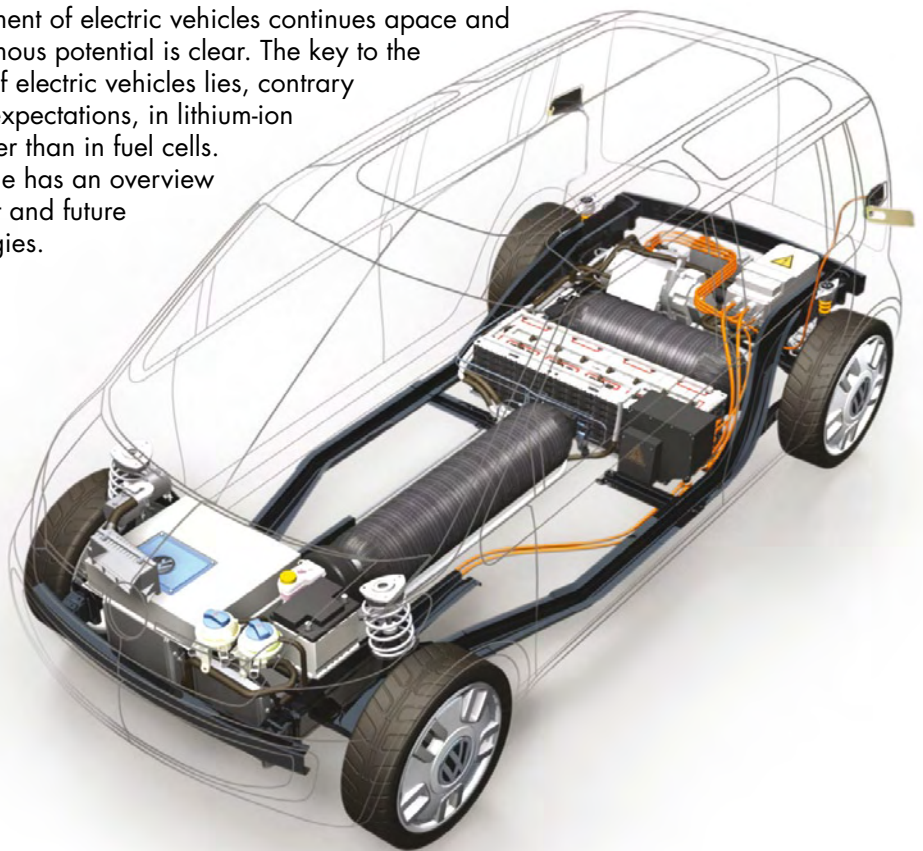
The compact OBD2 Analyzer in the June 2007 issue was an enormous success. Now, it's enhanced with a graphical display, Cortex M3 processor and an Open Source user interface, the next generation (NG) of Elektor's standalone analyzer sets new standards for a DIY OBD2 project.
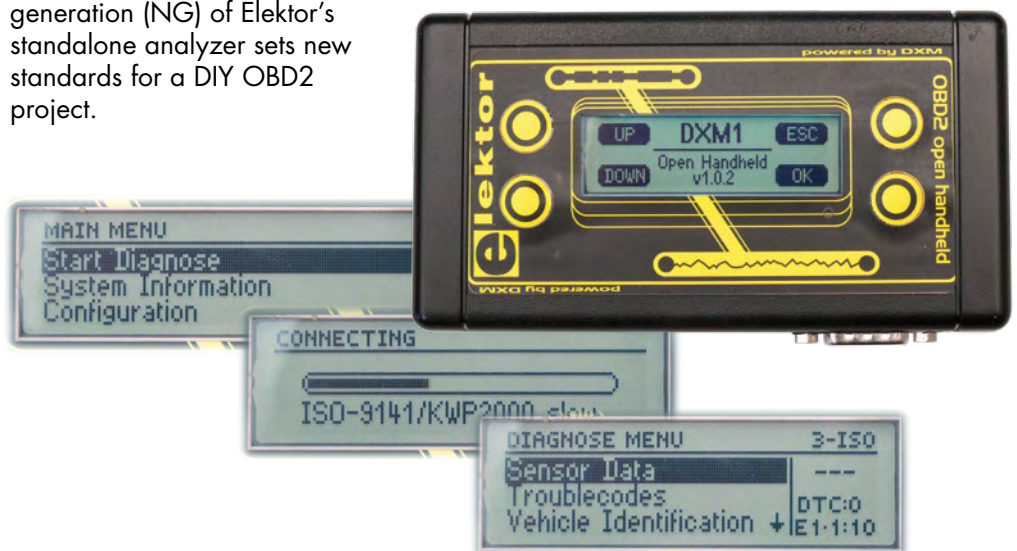
CONTENTS

30 GPS Datalogger

This project allows you to log the path of a bike or car trip and load the data into GoogleEarth™ using a common USB memory stick. The hardware is based on the famous Parallax Basic Stamp and open for further development.
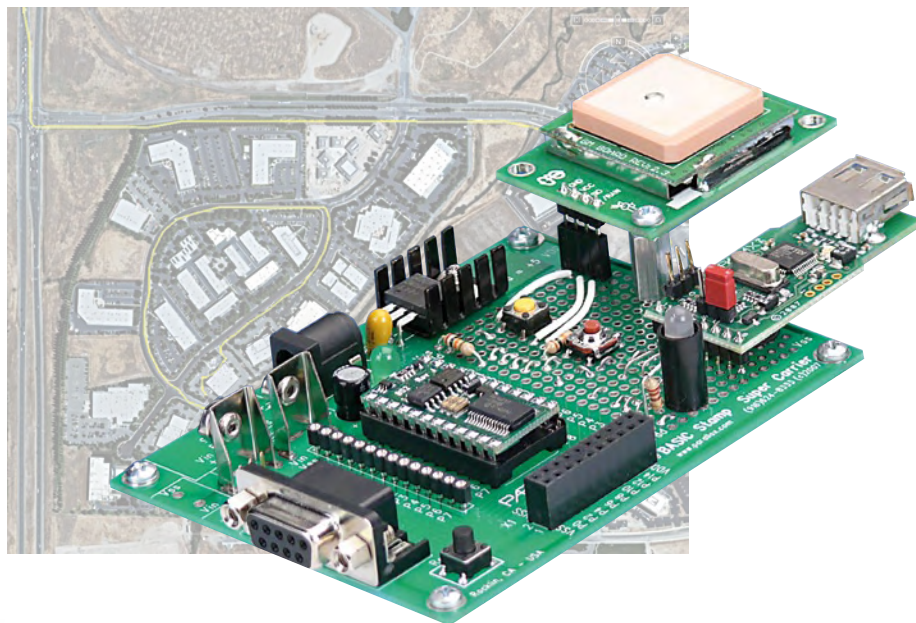
48 Battery Monitor

This circuit can be used in any application where batteries are charged and discharged. The circuit uses an LPC2103 microcontroller connected to a 22-bit A/D converter to measure charge and discharge currents, battery voltage, charge status (or available capacity) and the instantaneous power being supplied to or drawn from the battery.

elektor international media

Elektor International Media provides a multimedia and interactive platform for everyone interested in electronics.
From professionals passionate about their work to enthusiasts with professional ambitions.
From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment.
Analog and digital; practical and theoretical; software and hardware.

English
German
Dutch
French
Chinese

Italian
Spanish
Swedish
Finnish

Visit China with Elektor

➡ **combine electronics and culture**

events

The next Elektor study trip to China is planned for November 7–16 , 2009. If you would like to join us, please request further details on the Elektor website.

During this 10-day trip we will visit the China Electronics Fair in Shanghai, a professional industrial electronics fair with an area of no less than 60,000 m². We will also pay at least one visit to the well-known 'electronics high street' in Shanghai. As the name suggests, this street is entirely dedicated to electronics shops, each vying to be the largest. In addition, a variety of interesting company visits are on the itinerary, including a tour of the production department. We are also organizing a business conference where you can learn all the do's and don'ts of doing business in China. Naturally, there's also time for culture. We will visit the Great Wall, The Temple of Heaven, the Forbidden City, Tiananmen Square and the Shanghai TV tower. On the closing day of the Elektor study trip the participants will go on a cycling tour through Beijing!

REGISTER NOW!
Places are strictly limited

More detailed information at www.elektor-usa.com/china

Invasion of the Trochotrons

In response to my call for information on 'trochotron' tubes in Mailbox, April 2009, I received the following email from Elektor reader Nick de Smith.

Hi Jan — Regarding the note in Elektor April 2009, I have several examples of an Haydu 6700 (NIB), Burroughs BD-301 & BX-1000 (NIB with original data sheets). I'm perfectly happy to lend you a bunch of these — if you want to light them up, please have a go — they are not very expensive, just a bit rare — they are so esoteric that collectors haven't yet forced the price too high...

I'll have my office ship some later this week — I'm just pleased they will be used...

A selection of the trochotrons kindly sent by Rick is pictured here: a VS10B from TEL (red plastic encapsulated device), a BD-301 from Burroughs (light metal encapsulated device) and a BX 1000 'Beam-X' from Burroughs (bare tube). The VS10B was noticed for its very strong magnitc field.

A further response was received from Roly Roper, a long standing Elektor reader from Melbourne, Australia:

Dear Jan — the name 'Trochotron' derives from the trochoidal (spiral) path the electrons follow in crossed magnetic and electrostatic fields, and is confusingly applied to at least three different devices; some forms of mass spectrometer, non-displaying decade counters, and one displaying counter. Trochotron counters are hard vacuum cathode-ray Beam Switching Tubes which were initially developed for telephone service c. 1950 where they seem to have been used as ten-way analog switches, possibly as (de)multiplexers, but soon

evolved into the fastest counting devices available. The speed of gas tubes is limited to kilocounts/s by the long de-ionisation time, while vacuum tubes are mainly limited by inter-electrode capacitance allowing Mc/s operation. Their main application as counters seems to have been in neuclionics, so while cross-licensed to other manufacturers most were something of a 'special' only produced in limited numbers.

With a single exception these tubes came encased in a strong ring magnet [1] and require special handling and mounting [2]. Because of their magnetic fragility it is unlikely that any of this type are still functional.

The unique exception is the E1T which is fully electrostatic (thus has no magnet and is as robust as any valve), and displays as well as counts. Developed at Philips Netherlands it featured in their PW4032 Counter. [3] Ronald Dekker's excellent website [4] has extensive information on the history, development, operation, and application of the E1T. Contrary to some opinion, Ronald shows these are not particularly difficult to drive, and his 40-year-old samples still work perfectly. It is worth noting his efforts to preserve these rare historic items in working order, such as using a reduced voltage constant-current heater supply.

As it happens I do still have the four GS10C decatrons (dekatrons?) with sockets, and four E1T trochotrons I bought from Proops over 40 years ago with the vague teenage dream of building an 8-digit counter/scaler.

Instead I took only 25 years to finish my TTL+Numitron [4] counter that still serves on the bench.

1. www.tubecollector.org/vs10g.htm
2. www.ekco-radar.co.uk/hubbard/nucleonics.php
3. www.cs.ubc.ca/~hilpert/e/edte/PhilipsPW4032/index.html
4. www.dos4ever.com/trochotron/TROCH.html

V & I Calibrator challenged

Hi Elektor — I'm impressed with the simple voltage/current calibrator in the May 2009 issue, and plan to build one. A note of caution though: combining multiple high precision components (in this case each with 0.1% precision) inevitably results in a net precision which is significantly degraded. The extent of the lost precision depends on the statistical distribution of the errors, but as a simple example, combining 4 components (e.g. the voltage
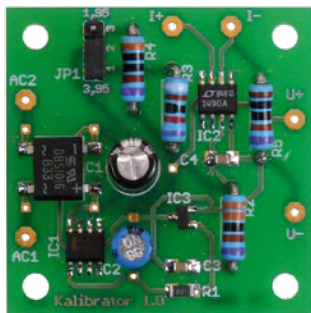
reference, R2, R3 and R5) each of which is 0.1% away from its nominal value could (if unlucky) result in a 0.4% error; more generally an error of

approximately 0.2% could be expected.

Another point is that the nominal values for the 1.9 V resistor divider actually deliver a voltage which is about 1.7 mV too high (i.e. almost 0.1%). But all in all a very nice little circuit!

Neil Conway (UK)

The author of the article, Dr Thomas Scherer, responds:

Although accuracy is a nice subject for really long discussions, it is also within the

realms of calculation. It is, after all, not that simple. I will give it a try. Theoretically, tolerances add up in worst case conditions. For the voltage, the effect is most pronounced at 1.9 V. The 'worst case' then translates into R2 being R2+ 0.1% or 3.9039 kΩ, and at the same time the sum of R4A and R4B –0.1% off the mark, i.e. at 3.37662 kΩ instead of 2.7 kΩ + 0.68 kΩ = 3.38 kΩ. Now we stir in a –0.1% error on part of the LM4050 supplying 4.091904 V, causing a reference voltage of

4.091904 V / 7.28052 kΩ
× 3.37662 kΩ
= 1.89777203 V

resulting in an error of

1.9 V / 1.89777203 V – 1
= **+0.1171263%**

Conversely, with the LM4050 deviating +0.1% = 4.100096 V, R2 at –0.1% = 3.8961 kΩ and (R4A + R4B) at 0.1% = 3.3838 kΩ we get

4.100096 V / 7.280523904 kΩ × 3.3838 kΩ
= 1.901575537 V

instead of the desired 1.900000 V; an error of

1.9 V / 1.901575537 V = –**0.08285432%**.

A far cry from the dreaded 0.4% and proving how dull theory can become when it is oversimplified = incorrectly used.

For the sake of completeness the worst case offset voltage of the op amp should be calculated in (max. 1.2 mV). The negative error then becomes 1.89657203 V (–0.14586781%) and the positive error, 1.902775537 V (+0.1807456%).
Only with the current measurement (I), the worst case error is cumulative and would amount to total positive and negative errors of +0.343% and –0.308% respectively.
All this is based on a worst case scenario. In reality different values can be expected, the tolerance on the components follows the Gaussian (bell) curve and an overall tolerance of 0.035% is safe to assume.
Now we also need to consider that statistical errors do not add linearly. As a rule of thumb, two sources E_a and E_b with the same error produce a cumulative error E as in

$$E_a + E_b = 1.4\ E$$

Realistically the error at 1.9 V should be of the order of

±0.06% (estimated, not calculated), because the typical offset voltage of the op amps is 0.22 mV, equating to an error of 0.013%.
For the I (current) calibrator, we should anticipate an error of 0.08%, which for all practical purposes becomes the typical error. Extremes may occur, naturally, but still within the bounds described above.

Memory effect in Hi-Fi audio transistor amplifiers

Dear Sir — the Lavardin Technologies IT Hi-Fi amplifier was introduced in 1997 and has achieved worldwide recognition as an amplifier with incomparable performance. Lavardin claim the performance was achieved by the removal of memory effect from their design.
I have checked with five of the major manufacturers of high performance audio transistors and none of them know anything about memory effects in silicon.
I have attached two papers from the Audio Engineering Society (AES) describing memory effect and its measurement.
The Lavardin products have the part numbers removed from the transistors and ICs, and other parts of the circuits are potted. Do Elektor engineers have any knowledge of memory effect? Would Elektor like to develop a test arrangement for the measurements of memory effect?
Dr Louis D Thomas (USA)

Neither of our in-house audio experts Harry Baggen and Ton Giesberts was aware of Lavardin's claim or the phenomenon it is reportedly based on.
Considering the publication date of the articles and the fact that no widespread discussion appears to have been triggered, at least not in JAES, we are lead to believe that it's a marketing ruse and audio enthusiasts can not be bothered. None the less we are inviting interested readers to respond with their viewpoints.

Robohockey

Dear Editor — I write regarding an article published in Elektor's April 2009 edition called 'OkISEC: Multiplayer Online Robohockey'.
The credits printed are somehow incomplete because there where six persons involved in the project, the ones listed in the article plus Nuno Paiva, Daniel Bastos and myself.
In the paper submitted to a robotics conference the project was originally named HorolSEC but we changed the name after a professor's suggestion. At http://horoisec.07x.net, if you follow the link to the team ('Equipa' in Portuguese) everyone who worked on the project is duly listed.
Ricardo Faria (Portugal)

Thanks for that Ricardo, credit's due where credit's due.

Corrections & Updates

Data Logger "deluxe"
March 2008, p. 26–31, ref. 070745
The firmware for this project has been upgraded to version 2.00. The new version has a number of bug fixes and adds some new functions, including a programmable recording interval (1 to 9999 seconds or 1 to 9999 minutes), and support of the CSV file format. A summary of modifications and enhancements is contained in the download found on the project webpage at www.elektor-usa.com/070745.

Bathroom Fan Controller
July & August 2009, p. 97, ref. 090078
The value of R5 is erroneously shown as 1 kΩ and should be changed to 100 kΩ for the circuit to operate properly.
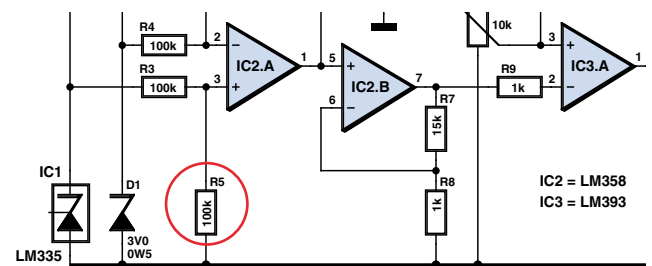
Unofficial Propeller Expo West event

June 27th and 28th 2009 marked the first West Coast Unofficial Propeller Expo (UPE) event. Staged at Parallax Inc. in Rocklin, California, this event was full of exciting projects, presentations, and the highly innovative, good-natured users the Propeller chip naturally attracts.

The "unofficial" expo is the brain child of Jeff Ledger, a Propeller customer and active Parallax forum member, known as OBC (Oldbitcollector) on the Parallax Propeller Forum. In 2008, Ledger came up with the Propeller expo as a way for Propeller fans and like minds to get together to show off their latest projects. "When I started this, I figured we might find 30 people to meet for an afternoon to eat pizza and discuss Propeller projects," said

Ledger. Fast forward to a year later, and the expos are now wildly popular. Over 150 people from around the world attended the West Coast Expo at Parallax, still jokingly referred to as "unofficial."

The event attracted people from 10-year-olds to 60+, whose experience ranged from hobbyist to Ph.D. The presentations were awesome examples of the flexibility of the Propeller chip. Everything from autopilot flight controls and microcomputer systems to robotics and sailing was presented. The level of excitement around each project and presentation was intoxicating.

In a not-too-distant room, attendees were greeted with free Propeller boards from SchmartBoard (www.schmartboard.com) and encour-

aged to sit down and solder up their own surface-mount Propeller chip using this convenient system. A live webinar with Chip Gracey (President, Parallax Inc.), Jeff Martin (Software Engineer, Parallax Inc), and Hanno Sander (President, MyDanceBot.com), included both live and remote participants and was aimed at answering many questions covering a wide range of Propeller topics.

Announcements on future UPE events, videos, and more pictures of Propeller Expo West may be found at the website below.

The next expo will be hosted in Norwalk, Ohio, on Saturday August 22nd, 2009. As on the West event, every attendee will receive a free copy of Elektor USA magazine.

www.Parallax.com/event

(090569-I)

Pololu 3pi Robot

The popular 3pi robot from Pololu Robotics and Electronics is a small, high-performance, autonomous robot designed to excel in line-following and maze-solving competitions. Powered by four AAA batteries (not included) and featuring unique power system that operates the motors at a regulated voltage of 9.25 V, the 3pi is capable of speeds up to 100 cm/second while making precise turns and spins that are not affected by the battery voltage when the batteries run low. The robot comes fully assembled with two micro-metal

gearmotors, five reflectance sensors, a 2-line by 8-character LCD, a buzzer, three user pushbuttons and other features, all connected to a user-programmable AVR microcontroller. The 3pi is approximately 3.7 inches (9.5 cm) in diameter and weighs 2.9 oz (83 g) without batteries.

The 3pi is based on an Atmel ATmega328P microcontroller running at 20 MHz with 32 KB of flash program memory and 2 KB of data memory. The ATmega328P microcontroller

makes the 3pi compatible with the popular Arduino development platform. Free C and C++ development tools are available, and an extensive set of libraries make it easy to interface with all of the integrated hardware. Sample programs are available to show how

to use the various 3pi components, as well as how to perform more complex behaviors such as line following and maze solving.

The unit price is USD 99.95.

www.pololu.com

(090569-II)

Popular antenna now available for MICS band

Antenna Factor's popular Splatch embeddable antenna is now available for the Medical Implant Communication Service (MICS) frequency band. The antenna is centered at 403 MHz and designed to operate of the range of 402–405 MHz, which makes it well suited for MICS as well as a wide variety of general RF applications.

By utilizing a proprietary grounded-line technique, the Splatch antenna achieves excellent performance in a compact surface-mount pack-

age. Unlike many compact antennas, the Splatch exhibits good stability in proximity to objects and persons.

The antenna is RoHS compliant and designed for manual or automated assembly. It measures 1.1 x 0.5 x 0.062 inch (L×W×H) and has a 50-ohm characteristic impedance with a typical VSWR of less than 1.9. An evaluation kit is available.

www.antennafactor.com

(090569-III)

402-408 MHz Splatch
MICS BAND EMBEDDABLE ANTENNA

Datakey Electronics releases new immersion-rated rugged data carrier system

Datakey Electronics, Inc. has released the Bar Series, the latest addition to its line of rugged, portable memory products. The new Bar Series data carrier system, consisting of a memory token and mating receptacle, comes with an IP-67 immersion rating and is designed to meet the requirements of the most demanding military and industrial applications.

The Bar Receptacle allows original equipment manufacturers (OEMs) to add a portable data carrier to their product without sacrificing large amounts of panel space or interior volume. The Bar system's unique surface-mount form factor minimizes the amount of interior space required, as only the receptacle's pins extend inside the OEM device. Also, the Bar receptacle has a small footprint of only 0.82 in² (5.3 cm²). The receptacle is secured using two screws and an adhesive gasket, which also contributes to the Bar's IP-67 rating and provides a tamper-evident installation.

Designed for use in the most challenging environments, the Bar Series meets MIL-STD-810F specifications, including shock, vibration, immersion, salt-fog, blowing dust, blowing sand, freezing rain, and more. The Bar has an operating temperature range of –40°C to

+85°C (–40°C to +100°C storage) and ESD protection to greater than 15 kV. In addition, the Bar token and receptacle are both designed to minimize electromagnetic emissions, which is important in many military applications.

The Bar features intuitive slide-in/slide-out operation with an open design for easy in-field cleaning, an important feature for tactical devices. The Bar token and receptacle have physical features to insure that the token can only be inserted in the proper orientation, to enable installation by feel
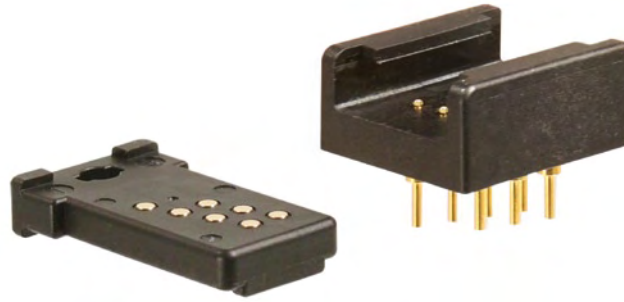
in the dark. The Bar receptacle has a retention pin that provides tactile feedback when the token is fully inserted and helps keep the token firmly in place during operation. The Bar memory token and the Bar receptacle both use solid overmolded construction, making them extremely rugged and durable. The Bar token uses industry-standard serial EEPROM with SPI, I²C, and Microwire serial interface options and memory sizes from 1 Kbit to 256 Kbit. The Bar token also has a writeable label and is small enough to be carried in a pocket or placed on a keychain or lanyard.

While designed to meet the requirements of tactical military applications such as crypto ignition keys (CIKs), the Bar can also be used in other harsh environment applications, such as machine access control, calibration and parameter uploads, feature enable/disable, and data transfer.

Unlike consumer memory solutions such as SD cards and USB flash drives, the Bar Series memory tokens and receptacles meet a host of MIL-STD-810F environmental specifications, have a high mating cycle life (25,000 cycles for the receptacle and 15,000 cycles for the token), are sold only to OEMs, and have long-term product availability.

www.datakeyelectronics.com

(090569-IV)

Total Phase announces Data Center™ 3.0 software

Total Phase has announced the release of the all-new Data Center™ 3.0 software, the only protocol analysis software in the industry with true real-time performance and cross-platform support for Windows, Linux, and Mac OS X. The Data Center software is free bus monitoring software that can capture and display USB, I²C, and SPI bus data in real time with the Beagle line of hardware protocol analyzers. Version 3.0 of the Data Center software has been re-engineered with a new extensible architecture and features LiveDisplay™, LiveFilter™, and LiveSearch™ technology that allow developers to easily see, filter, and search bus data as it is being captured in real time. This major upgrade includes a new and improved graphical interface with an innovative command line tool, tree view of connected devices on

a bus, local language support, and much more.

With LiveDisplay technology, developers gain instant access to serial bus data. While many solutions claim to have real-time analysis capability, the LiveDisplay technology shows the actual bus data as it is happening in true real time, breaking from the traditional capture/download process model of protocol analysis.

To help manage large streams of incoming data, the LiveFilter feature allows developers to instantly filter live USB, I²C, and SPI traffic

against any number of parameters. Additionally, the LiveSearch functionality offers a powerful way to quickly locate data of interest in a live capture.

Beagle Protocol Analyzers are the hardware complement to the Data Center software. These analyzers can non-intrusively capture and display USB, I²C, and SPI serial bus data. Total Phase offers three different bus monitors: the Beagle USB 480 Protocol Analyzer, which is a High-Speed USB 2.0 monitor; the Beagle USB 12 Protocol Analyzer, which is a Full-Speed USB monitor; and the Beagle I²C/SPI Protocol Analyzer, which can capture and display I²C and SPI data.

The Data Center 3.0 software is available on the Total Phase website as a free download for Windows, Linux, and Mac OS X. Beagle protocol analyzers are available for purchase online.

www.totalphase.com

(090569-V)
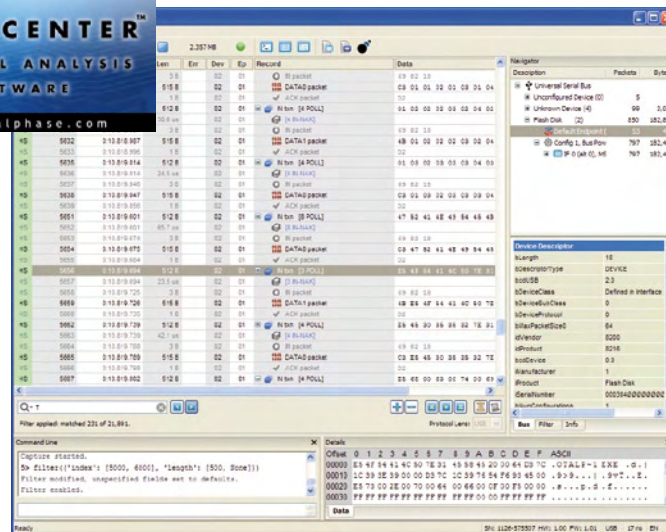
Project Whirlwind comes home

The Project Whirlwind Computer collection, comprising pioneering digital computing research conducted at MIT in the 1940s and 1950s, has been transferred back to MIT from the MITRE Corporation, and its contents are being opened to the public for the first time.

The research project, which included the creation of "Whirlwind I" – the first digital computer at MIT and the fastest of its time – began at the Institute and was subsequently moved to Lincoln Lab and then later to MITRE. Materials in the collection were previously unavailable to researchers, but with its transfer to MIT, the collection is now accessible to the public. Key documents from the Whirlwind collection have also been digitized and will soon be made available online.

Whirlwind I was completed in 1951, the same year that Project Whirlwind was detached from the Servomechanisms Lab to become the MIT Digital Computer Laboratory. Unclassified research projects using the computer were managed by the Digital Computer Lab staff on the MIT campus, where Whirlwind I occupied 3,300 square feet in the two-story Barta Building (N42).

As a precursor to modern-day computers, Whirlwind's legacy can be seen in today's software and hardware. Parallel digital processing combined with random-access magnetic-core memory enabled the initial launch of commercial computers, and interactive visual computer displays. Its groundbreaking design also laid the foundation for simulation and real-time technology. In operation until 1959, Whirlwind formed the basis for the U.S. Air Force's Semi-Automatic Ground Environment (SAGE) air defense system.

The project materials have been transferred to the MIT Institute Archives and Special Collections. Documents on the project will soon be available through the MIT Libraries website.

For additional information, see the article published in the June 3, 2009 issue of MIT Tech Talk (PDF download: web.mit.edu/newsoffice/2009/tech-talk53-27.pdf).

libraries.mit.edu/archives/whirlwind

(090569-VI)

Antero launches standard development kit for ARM7

The Antero Single Board Computer (SBC) family is the newest series of products from Vesta Technology. The Antero Development Kit is optimized for embedded control applications and allows the SBC to be very configurable, making it the Swiss Army knife of SBCs. Based on 32-bit ARM7 technology, the SBC combines high performance with low power consumption and a small form factor. Antero features a rich I/O set that includes CAN bus, USB, I2C, SPI, and several serial, PWM, quadrature and digital interfaces. Antero is constructed in the Industry Pack form factor. The IP module version is compatible with Industry Pack carrier boards.

The Antero SBC with its full development environment is being offered now at a special introductory price.

The SBC comes with software tools, sample application programs, and an RTOS. Antero is ideal for machine and industrial control, equipment monitoring, process control, instrumentation, and many other applications.

www.vestatech.com

(090569-XII)

Ultra-thin RFID modules target paper bags

Murata has increased the applicability of its tiny RFID modules by significantly reducing the thickness, making its Magicstrap range suitable for paper label inlays for mainstream retail applications.

Using its multi-layer ceramic technology expertise, the Japanese group has managed to embed all the necessary RF circuitry, including antenna filters, matching circuitry and ESD protection, in the LTCC substrate .
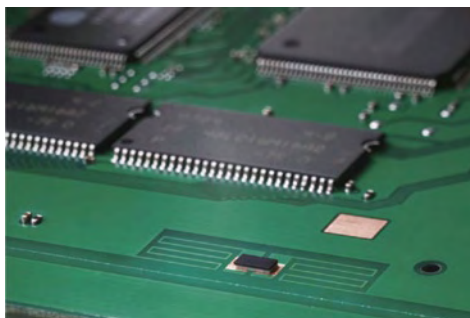
The resulting second-generation modules measure 1.6 x 1.0 x 0.25 mm, representing a reduction in module thickness by more than half and an 89% reduction in volume. The modules can be mounted using ordinary adhesive on to almost any conductive surface to act as the antenna, with relaxed placement accuracy in the millimeter range.

The relatively low placement accuracy requirements allow RFID inlays for paper tags to be manufactured without sophisticated component placement machinery, leading to substantial cost savings for manufacturers. The module's embedded wideband antenna matching circuit allows the part to be read at a distance of up to 5 meters in the 860–960 MHz UHF band. According to Murata, this makes the modules suitable for worldwide use (in Europe, the USA and Japan), thus reducing cost and inventory.

www.murata-northamerica.com

(090569-XI)

USB in the
FAST LANE

FT2232H/ FT4232H
USB 2.0 Hi-Speed Interface ICs and Evaluation Modules

- FT2232H (Dual 2.0 Hi-Speed USB to Multipurpose UART/ FIFO IC) has 4k bytes Tx and Rx data buffers per interface

- FT4232H (Quad 2.0 Hi-Speed USB to Multipurpose UART/MPSSE IC) has 2k bytes Tx and Rx buffers

- Multi-Protocol Synchronous Serial Engines (MPSSE), capable of speeds up to 30Mbits/s, provide flexible interface configurations

- Entire USB protocol on a chip with integrated LDO regulator and PLL

- Extended temperature range (-40ºC to +85ºC)
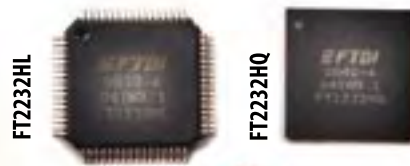
FT2232HL

FT2232HQ

FT2232H Mini Module

USB-COM232-Plus2

Visit us at the ESC Boston 2009 at Stand 621.

Learn today. Design tomorrow.
ESC
Boston • September 21 – 24, 2009

NEW at FTDI

**FTDI announce the availability of a range of USB 2.0 to single/dual-port RS232/RS422/RS485 modules.
These modules provide an ideal solution to communicate with legacy peripherals, which use these standard serial interfaces, via a USB 2.0 connection.**

FTDI Chip

Future Technology Devices International Ltd.

7235 NW Evergreen Parkway, Suite 600, Hillsboro, OR 97124-5803, USA
Tel: +1 (503) 547-0988
Fax: +1 (503) 547-0987
E-Mail (Sales): us.sales@ftdichip.com

Now available to order at www.ftdichip.com

High-definition video interface features JPEG2000 compression

Curtiss-Wright Controls Embedded Computing has announced the XMC-280 interface board for high-definition video with JPEG2000 compression. The XMC-280 provides an open standards based solution for capturing, compressing, decompressing and displaying two channels of video with resolution up to 1080p or 1920 x 1200.

Using the XMC-280, multiple channels of high-definition video can be distributed over standard gigabit Ethernet networks with excellent quality and minimal latency. Details are retained at 50:1 compression, which enables the transmission of eight 1080p60 channels over a single Gigabit Ethernet link. The XMC-280 has two video inputs and two DVI-D digital outputs. Captured video is compressed using a FPGA-based JPEG2000 algorithm or transmitted via the PCIe interface uncompressed. It also provides two stereo channels (or four mono channels) of audio input/output (16-bit 48 kHz, WAV and PCM encoding).

Designed for demanding military video applications, the XMC-280 facilitates the design of systems for recording or distributing video, such as situational awareness applications. JPEG2000 offers better quality higher and compression ratios than MPEG in real-time embedded systems, and each frame is compressed individually to eliminate dependence on preceding or subsequent frames. Frame-by-frame compression also reduces latency and increases immunity to transmission errors.

The XMC-280 is supported on Intel x86 and Power PC hosts under Windows, Linux and VxWorks, and as a standard XMC-format module it can be used in a variety of system types, such as PCs and embedded rack-mount displays.

**www.cwcembedded.com/
products/0/189/669.html**

(090569-VII)

Authentication IC provides battery protection

Atmel Corporation has announced the AT88SA100S, an ultra low-cost cryptographic battery authentication IC for mobile phones, cameras, portable power tools, and other battery-powered devices.

The AT88SA100S CrytpoAuthentication™ IC is a unique battery authentication IC that uses a SHA-256 cryptographic engine and 256-bit key that cannot be cracked using brute-force methods. It can be used to protect mobile phones, portable power tools, cameras, and other microcontroller-based products against counterfeit battery packs. The AT88SA100S ensures that replacement batteries meet the product manufacturer's standards by providing secure, reliable authentication that prevents operation and/or charging with a counterfeit battery.
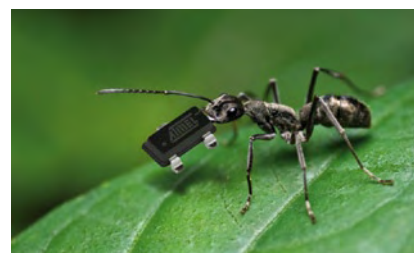
Counterfeit batteries are typically less expensive than OEM batteries because they often lack protective mechanisms to prevent short circuits, overheating, leakage, ignition, rupture, and other malfunctions. They are also likely to lose their charge sooner and wear out more quickly than genuine battery packs. It is estimated that 75% of all replacement batteries sold are counterfeits and clones.

The AT88SA100S has 256 bits of SRAM for key storage, a guaranteed unique 48-bit serial number stored permanently in the chip, and 88 one-time, user-programmable fuses that can be used to store battery parameters or status information. The 256-bit key is stored in on-chip SRAM in the battery manufacturer's facility that is powered directly by the battery pack. Physical attacks to recover the key are very difficult to carry out because removing the CryptoAuthentication chip from the battery pack erases the SRAM memory, rendering the chip useless.

Battery authentication uses a challenge–response protocol between the microcontroller in the portable end-product (host) and the CrytpoAuthentication IC in the battery (client).

To speed system design, Atmel can provide complete ARM® and AVR® compatible source code libraries that implement all necessary cryptographic modules for host-side authentication capability. The AT88SA100S requires only one GPIO pin on the host processor and only three wires on the connector to the battery, plus a standard bypass capacitor.

**www.atmel.
com/products/cryptoauthentication**

(090569-VIII)

USB interface links GPIB instruments to laptop PCs

ADLINK Technology has announced the USB-3488A, a cost-effective IEEE-488 GPIB to USB interface controller. The USB-3488A is fully IEEE 488.1 and IEEE 488.2 compatible and provides the benefits of plug-and-play USB connection for simple, direct connection of GPIB instruments to a laptop computer.

The USB-3488A was developed using ADLINK's intellectual property in an FPGA device that incorporates the GPIB controller and provides reliable GPIB bus control capability while supporting a transfer rate of greater than 1.2 MB/s. The USB-3488A is also fully compatible with NI-488.2 software, and it supports industry-standard VISA I/O library software so that existing GPIB applications can be used without modification.

The USB-3488A enables simple control of thousands of stand-alone GPIB instruments for a wide variety of measurement applications. The USB-3488A is both compact (3.2 × 2.4 × 1.1 inches) and lightweight (180 g) for easy portability. A built-in two-meter USB cable supplies power and eliminates the need for external GPIB cables. The USB-3488A has passed vibration testing at 1 G while in operation and 2.5 G while not operating.
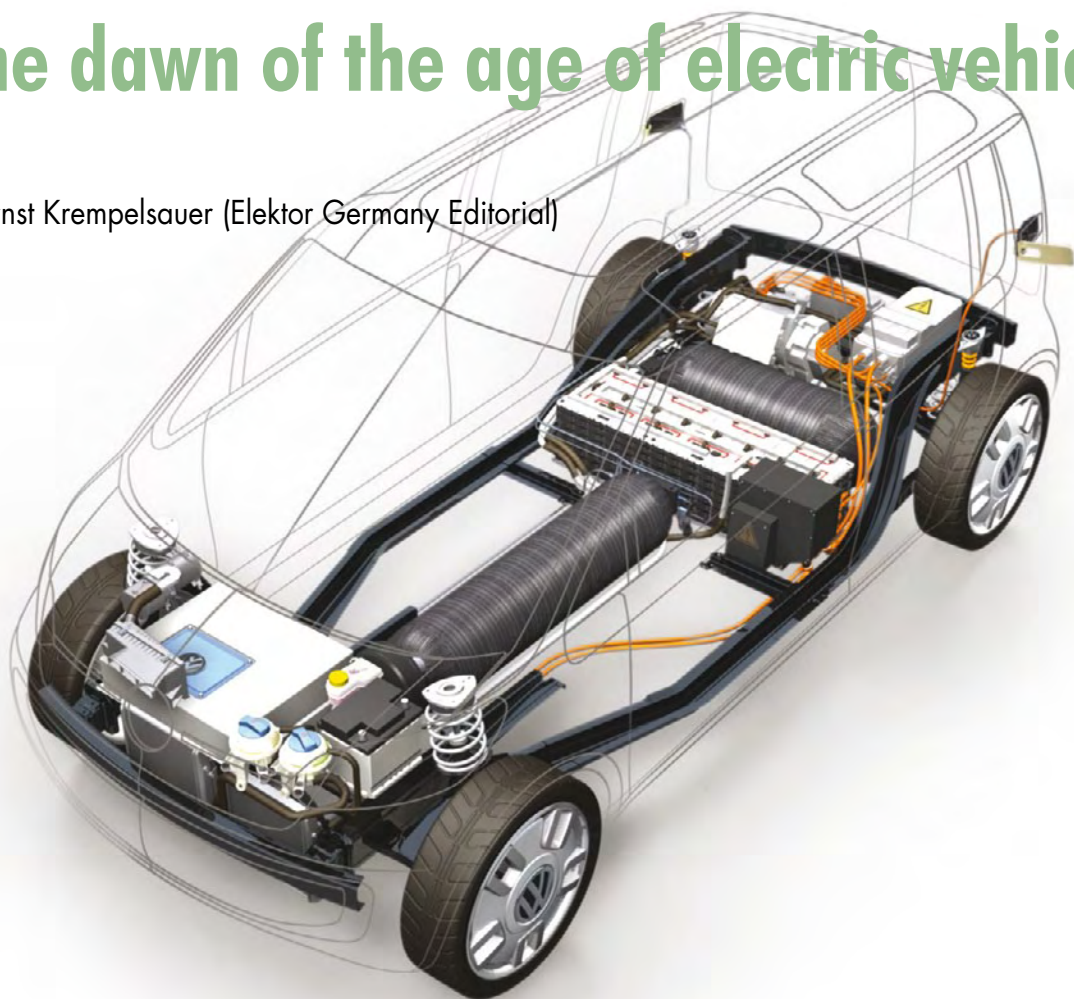
http://www.adlinktech.com/GPIB

(090569-V)

The Road to Battery Power

The dawn of the age of electric vehicles

By Ernst Krempelsauer (Elektor Germany Editorial)

At the beginning of the twentieth century the internal combustion engine started to take over from electrically-powered vehicles. Now, a hundred years later, electric vehicles are slowly but surely making a come-back. Development continues apace and the enormous potential is clear. The key to the success of electric vehicles lies, contrary to early expectations, in lithium-ion cells rather than in fuel cells.

It was an electric vehicle, driven by Belgian Camille Jenatzy, that was first to break the magical 60 mph (about 100 km/h) barrier in 1899. His torpedo-shaped car (**Figure 1**) was accelerated to 105.88 km/h by 200 V lead-acid batteries powering two 25 kW motors. Although not lacking in power, the vehicle fell rather short when it came to endurance. No less an individual than Ferdinand Porsche, working at the Lohner company in Vienna, was inspired by this to try to overcome the disadvantages of electric drive by combining it with a petrol engine, thereby inventing the first

hybrid vehicle. The design was also the first to feature an electric hub motor and all-wheel drive (**Figure 2**).
From then on the onward march of the motor car, first with internal combustion petrol engines and then with diesel engines, seemed unstoppable. The most significant factor in their rise was the ready availability of cheap crude oil from which fuel could be made. A century later, conditions have changed for the internal combustion engine. Oil is becoming scarcer and more expensive, and there is pressure to reduce CO_2 emissions significantly. In today's hybrid

vehicles the benefits now work in the opposite direction: the electric drive serves to help overcome the disadvantages of the internal combustion engine. And the prospects for these vehicles improve as batteries improve; ultimately the internal combustion engine may disappear altogether.

This article presents an overview of current developments and the prospects for battery technologies in future electric vehicles.

Paving the way for hybrid vehicles

Hybrid electric vehicles (HEVs) have been in mass production since the introduction of the Toyota Prius I in 1997. They have shown that the electric drive train is suitable for mass production and reliable. With the manufacture of some two million HEVs in Japan, and simultaneous rapid progress in the development of lithium ion batteries, it is inevitable that companies all over the world are racing to produce the electric vehicles of the future. The next development is the so-called plug-in hybrid electric vehicle (PHEV), with a socket to allow its battery to be charged, and an increased range in purely electric operation. The first PHEV, the F3DM from BYD in China, is already being made in small quantities and being sold to governmental organizations. BYD ('Build Your Dream') is the world's biggest producer of mobile phone batteries and also makes vehicles, most recently in a joint venture with Volkswagen.

The Chevrolet Volt (and the Opel sister model, the Ampera) are expected to be the first mass-market PHEVs, available at the end of 2010. Current reports indicate that Toyota and others will join the market in 2012. In the next three years the mass production of pure electric vehicles (EVs) is also likely to begin. Current models, manufactured on a relatively small scale, include the Tesla Roadster, made in California, and the Norwegian TH!NK. The i-MiEV is expected to go into production this year: this compact car, announced as the first mass-production pure electric vehicle, is the result of collaboration between Mitsubishi in Japan and PSA Peugeot Citroën in France. The PSA group is still the biggest maker of EVs, having sold more than ten thousand units, but the situation is changing rapidly. Besides its PHEV line, Toyota has also announced an EV version of the new iQ for 2012, and in the same year Nissan expects to sell one hundred thousand units of its EV model, to be unveiled at the Frankfurt Motor Show in September 2009 or perhaps even earlier.

One left over

Until recently hybrid vehicles used exclusively NiMH batteries: manufacturers committed to the technology for three to five years and built capacity to suit. However, NiMH is a mature technology and its development seems to have reached its limits; for modern PHEVs and EVs it does not offer a compelling proposition in terms of cost and energy density.

From the point of view of cost the stalwart lead-acid batteries are hard to beat. Recent technological developments have made them suitable for use in hybrid vehicles. With the explosive rate of development in lithium ion batteries, however, lead-acid technology seems to be being left behind. There is little that can be done to improve energy density and, in the world of electric vehicles, its use is confined to low-cost scooters. Supercapacitors, although unmatched in terms of cycle life, store too little energy to be useful, and industry will likely not wait for the capacitors announced by Eestor, with an enormous claimed energy density, to materialize: devices suitable for use in electric vehicles are

Figure 1.
Back to the future: 112 years ago the electric vehicle was the leader of the pack. In 1899 Camille Jenatzy reached a top speed of 105.8 km/h (66.1 mph) in a vehicle powered by lead-acid batteries delivering 50 kW.
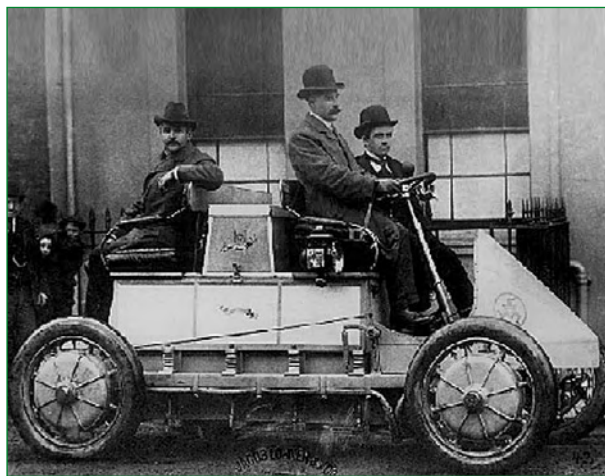

Figure 2.
The 1900 Lohner-Porsche was a hybrid car: a petrol engine drove a DC generator to produce power for the electric hub motors.

at least three years from mass production.

The current best-selling battery for electric vehicles is manufactured in great secrecy: the ZEBRA battery, which uses a NaNiCl chemistry, is relatively low-cost (around $ 700 per kWh) but has the disadvantage of an operating temperature of around 300 °C and thermal losses of 100 W, making it not ideally suited to private vehicles. A significant advantage, however, is that it is highly insensitive to ambient temperature. The high-temperature units are available as an option on the TH!NK city electric car; the alternatives available are A123 cells with a Lithium Iron phosphate ($LiFePO_4$) chemistry, and the EnerDel $LiMn_2O_4$ spinel cells. For future electric vehicle development it would appear that lithium ion cells offer the best prospects, and that ZEBRA cells will occupy only a small market niche. Indeed, lithium
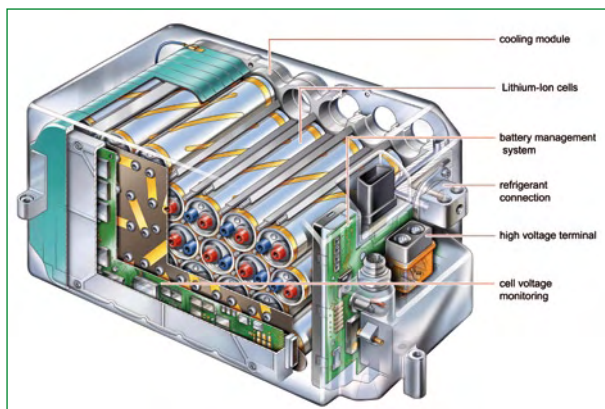

Figure 3.
The Mercedes S400 Hybrid is the first mass-production hybrid car to use a lithium ion battery. The battery is completely sealed and temperature-controlled. Picture: Daimler Benz AG.

Carbon footprint and power consumption

Although electric vehicles are themselves zero-emission, the generation of power may of course produce CO_2. The average kWh of electricity generated in the UK produces about 540 g of CO_2; in countries such as Norway (which has a lot of hydroelectric generation) and France (which has a lot of nuclear generation) the figure is rather lower. However, even at 540 g per kWh an efficient electric car like the i-MiEV has an advantage: a journey of 60 miles (100 km) might use around 10 kWh of electricity, making the total carbon emission 100 g CO_2 per mile (60 g per km), less than any petrol-engined car.

The carbon footprint can of course be further reduced by generating a greater fraction of our power from zero-emission sources. An interesting possibility here is the use of biomass. If we translate the energy available per hectare per year (in central Europe) into distance we find that using a biomass-to-liquid (BTL) process and powering a car's engine directly from biofuel gives about 60,000 kilometres per hectare per year (about 15,000 miles per acre per year), whereas using photovoltaic panels to produce electricity for an electric car gives 1,000,000 kilometres per hectare per year (about a 250,000 miles per acre per year).

The extra electricity demand created by electric cars is less than often supposed. If all private cars were electrically powered, the electricity companies would only sell about 15 % to 20 % more power; and the oil companies would only be supplying diesel for trucks.

Image: Tesla Motors, Inc

The present and the near future

Tesla Roadster
Manufactured since 2008 in small quantities. Total energy storage is 53 kWh in 6,831 notebook-style lithium ion cells (375 V overall voltage); maximum range 220 miles (360 km); maximum power 225 kW. In May 2009 the Daimler Group (which includes Mercedes) acquired a 10 % stake in Tesla.

Image: MITSUBISHI MOTORS Germany GmbH

Mitsubishi i-MiEV
In mass production from the end of 2009. This electric version of a small car popular in Japan has an under-floor battery made from prismatic lithium ion cells by Yuasa. Energy stored is 16 kWh with a nominal battery voltage of 330 V, giving a 100 mile (160 km) range. The battery can be charged from an AC power outlet in seven hours, or fast charging is possible at specially-equipped stations.

Image: IFCAR

Chevrolet Volt
In mass production from the end of 2010. General Motors are pinning their hopes on this first mass production plug-in hybrid. A petrol engine is used to recharge the battery from time to time via a 53 kW generator, extending the range of the vehicle. When fully charged via its socket, the 16 kWh lithium ion battery is good for a range of 40 miles; with the petrol engine this is increased to 400 miles (600 km). Originally A123 were to supply the cells, but for mass production the cells will be made by LG Chem of Korea. The petrol engine is a 1.4 litre four-cylinder model made at the Opel plant in Vienna, and Opel has considerable involvement in the design of the Volt. The Opel (European GM) version will be called the Ampera and is expected to appear in 2011.

ion cells are already found in the first mass-produced European hybrid car: the Mercedes S400 Hybrid has an electric drive rated at 15 kW and is classed as a 'mild hybrid'. The lithium ion battery is from Johnson Controls/Saft and has a capacity of 6.5 Ah at 120 V, for a total energy stored of 0.78 kWh (**Figure 3**). An interesting aspect of the design is that the car's air conditioning system is used to help stabilize the temperature of the battery.

Chemistry set

The common features of all lithium ion cells are that, in the charged state, one electrode contains lithium, and that charge is transported across the cell by lithium ions. A significant advantage of lithium-based cells is their high terminal voltage and good capacity-to-weight ratio.

A wide range of recipes is available for the electrode material itself, each having its characteristic terminal voltage and other properties (see **Figure 4**). The other factors affecting characteristics and cost are the choice of electrolyte and of separator. Liquid electrolytes based on organic solvents and conductive lithium salts can be used, as can solid or gel film electrolytes (as in lithium-polymer, or 'LiPo' cells). LiPo cells have a particularly high energy density and are available in a range of shapes; however, they do not perform well at low temperatures.

The choice of separator is an important factor in both the manufacturing cost and the safety of the cells. An interesting recent development is a ceramic separator called Separion® by Evonik (formerly Degussa), already in use in cells by German battery maker Li-Tec. The separator can withstand high temperatures, and can thus help to prevent an internal short-circuit in the cell. This is a particularly important aspect as fully-charged lithium ion cells, in contrast to NiMH cells, cannot continue to be accept current without damage, including risk of explosion. The other side of this coin is the very high efficiency of the cell (90 % to 95 %). Safer types are the lithium iron phosphate ($LiFePO_4$) cells from A123 and GAIA, at the cost of around 10 % less energy density and terminal voltage.

Power and life

Figure 5 shows a comparison of specific power and energy for the storage technologies described above using various types of lithium ion cells. The gap between high-power lithium ion cells and high-energy cells is striking. **Table 1** shows an overview of the typical characteristics of these two product types. High power output is important for hybrid vehicles, which need to draw or store large amounts of energy over short time periods. Figures of 200 A (20C) are typical for a 10 Ah cell. For very high power applications such as hybrid Formula 1 racing cars, cells with specific power as high as 6 kW/kg are available.

In pure electric vehicles the battery is relatively larger than it is in hybrid vehicles. A charge or discharge current of
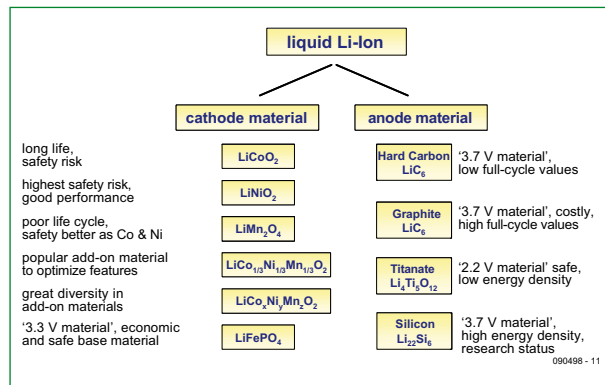

Figure 4.
There is a wide range of recipes for the electrode materials used in lithium ion cells, all with different characteristic properties.
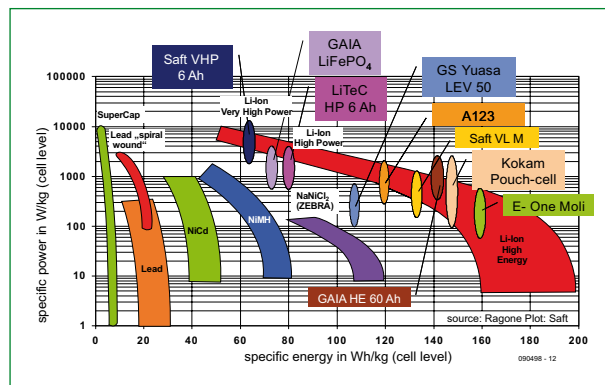Source: ISEA, RWTH Aachen University

Figure 5. Power density and energy density of commercial lithium ion cells compared to other energy storage technologies.
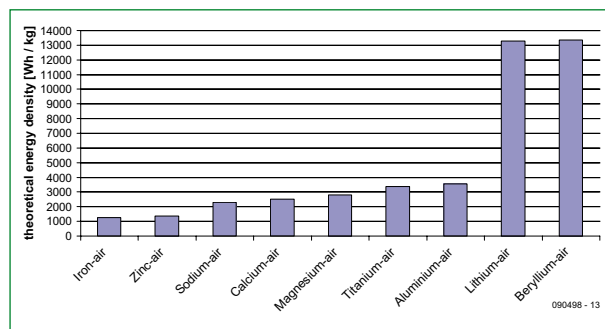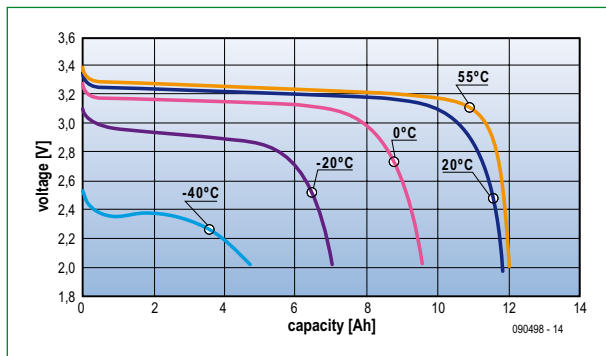Source: ISEA, RWTH Aachen University

Figure 6.
Theoretical energy density of metal-air cells. Rechargeable systems of this type remain a long way off.
Source: ISEA, RWTH Aachen University

around 3C is a reasonable value to give acceptable acceleration and braking. The problem here is chiefly one of energy density: a battery might have an energy density of around 190 Wh/kg, some six times smaller than that of petrol (1154 Wh/kg). Notwithstanding the regular announcements of magical new nanomaterials promising five- to ten-fold increases in energy density, Dirk Uwe Sauer, professor at the Institute for Power Electronics and Electrical Drives, RWTH Aachen University, believes that energy densities of up to 300 Wh/kg should be possible using 5 V cath-
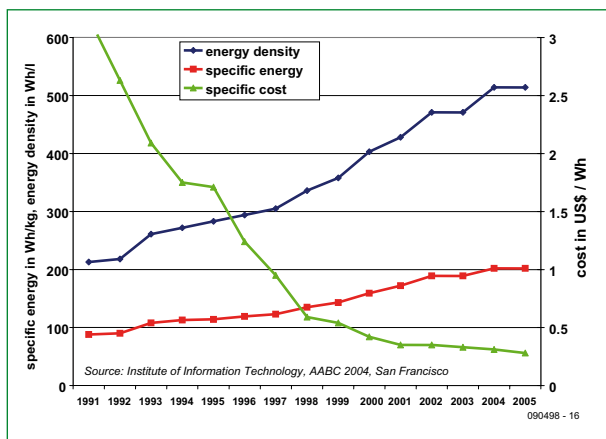
Table 1. Ratings of high-energy and high-power lithium ion cells		
	High Energy	**High Power**
Power density (25 °C)	200 to 400 W/kg	2000 to 4000 W/kg
Energy density	120 to 160 Wh/kg	70 to 100 Wh/kg
Efficiency	approximately 95 %	approximately 90 %
Self-discharge	< 5 % per month (at 25 °C)	< 5 % per month (at 25 °C)
Cycle life	up to 5000 complete cycles	106 cycles (at 3.3 % DOD)

Figure 7.
At very low temperatures the power output of a lithium ion cell falls markedly.
Source: ISEA, RWTH Aachen University
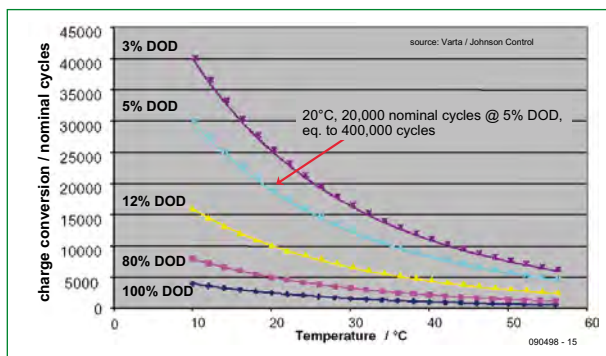
Figure 8.
Cell life as a function of discharge depth and temperature for NiMH cells. The characteristics of lithium ion and lead-acid cells are similar.
Source: Varta / Johnson Control

Figure 9.
Evolution of energy density (ratio of energy to volume or weight) and cost of lithium ion laptop batteries (standard cell size 18650).
Source: Institute of Information Technology, AABC 2004, San Francisco

a power supply not just to charge the battery, but also to keep it warm.

The life of the battery depends chiefly on three factors: temperature, depth of discharge (DOD) and age. As **Figure 8** shows, deep cycles cause such wear and tear on the cell that the degradation with temperature is negligible, and with shallow cycles (DOD of a few percent) the situation is reversed. What the illustration does not show is that the cells deteriorate gradually over time so that a little-used cell will, after a few years, become completely 'kaputt', as Professor Sauer puts it.

The operating life requirements for HEV batteries are from 8 years to 12 years, and spend most of their life between 40 % and 60 % charged. In pure EVs the DOD is typically around 80 % (charge level varying from 20 % to 100 %).

Costs and resources

The special high-energy batteries used in EVs are not yet made in large quantities and so prices remain correspondingly high at around $ 2,000 per kWh. By way of comparison, Kokam has LiPo batteries available in quantity at under $ 825 per kWh; lithium ion high-energy batteries are available from China at around £/€ 250 per kWh (they are used in electric bicycles, of which some twenty million are made every year); and lithium ion laptop batteries have fallen in cost by a factor of five over the ten years from 1995 to 2005 (see **Figure 9**), and now cost around $ 325 per kWh. Experts at RWTH Aachen University believe that with mass production high-power batteries could fall in price to $ 825 per kWh and high-energy batteries to $ 400 per kWh. Estimates from Japan of $ 250 per kWh are somewhat more optimistic.

Opinions also differ on the question of long-term availability of supplies of the raw material lithium. Lithium is more abundant in the earth's crust than either lead or tin, but most of the economically exploitable reserves are concentrated in a small number of countries, chiefly in South America. This brings a number of new risks, and a spike in demand could quickly lead to shortages and price hikes. The difficulty can be ameliorated by careful planning of production capacity and early construction of recycling facilities for lithium batteries.

Small is beautiful

Although there has been much popular interest in pure EVs with plenty of power and range, such as the Tesla Roadster, Professor Sauer thinks a better approach is to move towards vehicles with smaller batteries. He sees the idea of PHEVs equipped with rather small batteries as the best way of replacing petrol with electricity as quickly as possible There are many arguments in favour of this 'small is beautiful' position.

- The larger the battery, the more expensive the vehicle, and so fewer vehicles will be sold. This is not the way to move quickly towards the widespread use of electric vehicles.
- In industrialized countries the average distance traveled by a car is around 25 miles (40 km) per day. A battery providing a 120 mile (200 km) range (about 30 kWh) will remain 80 % unused, but this unused fraction of the battery must still be paid for and its weight must be carried around by the vehicle. A PHEV with a 30 mile (50 km) range on battery power (5 kWh to 10 kWh) might spend two-thirds of its time in electric mode. Coupled with the increased efficiency of the hybrid drive this leads to a reduction of 70 % in petrol usage.

ode materials such as $LiCoPO_4$ or $LiNiPO_4$, or silicon-based anode materials such as $LiSi_5$. The only theoretically feasible option for energy densities above 1000 Wh/kg is the metal-air cell (**Figure 6**), although a rechargeable cell of this type looks rather unlikely to appear within the next ten years. Of course, it is always possible to revert to the old plan of exchanging one's metal-air battery at the filling station, with the old batteries being regenerated on an industrial scale.

So, to return to the present and lithium ion cells: an important factor in their operation is temperature. Excessively high temperatures can be dangerous, leading to thermal runaway; operation at somewhat elevated temperatures reduces a cell's life, and operation at low temperatures reduces the power. The loss is noticeable at 0 °C and at –30 °C the output power is reduced to less than 10 % of that at room temperature (**Figure 7**). Some kind of temperature management system, such as that used in the Mercedes S400 Hybrid as we mentioned above, is a good idea. An EV for use in Alaska needs to be plugged into

- The only EVs with a significant chance of a large market are small city cars, especially in developing markets such as India and China. In 2006 in China more electric vehicles were sold than any other type, mostly scooters and bicycles. There would be a natural 'upgrade path' to an electric Tata Nano, and the Chinese government has announced funding for the development of electric vehicles to the tune of around a billion dollars.

As Professor Sauer says: 'prices will fall as a result of mass production, not as a result of waiting'.

The long view

Taking into consideration the effects of CO_2 emissions on the climate, it seems that if we want to maintain the level of personal transport (or, as in China and India, increase it) the only practical approach in the medium term is to switch to electric vehicles. The two-car household of the next decade will likely own a medium-sized PHEV for family trips and a small EV with a range of 30 to 50 miles (50 to 90 km). Further into the future, experts see the possibility of using a fuel cell as a substitute for the petrol engine in plug-in hybrid vehicles, so that private cars run on two-thirds electricity and one-third hydrogen rather than on petrol or diesel.

A widespread take-up of electric vehicles is expected in countries such as India and China. A study by management consultants McKinsey predicts the market for electric cars in China to be worth well over a billion dollars in 2030.
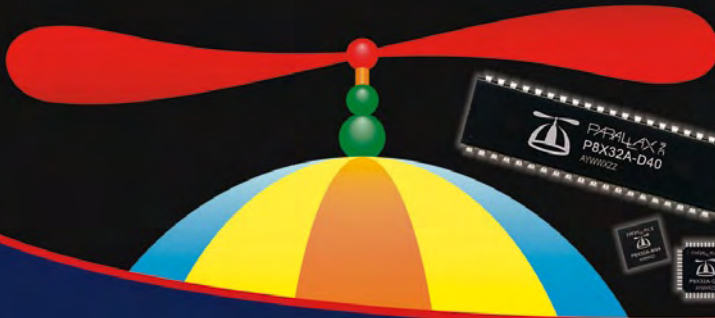
(090498)

OBD Analyzer NG

Next-generation handheld with graphical display, ARM Cortex M3 controller and Open Source user interface

By Folker Stange and Erwin Reuss (Germany)
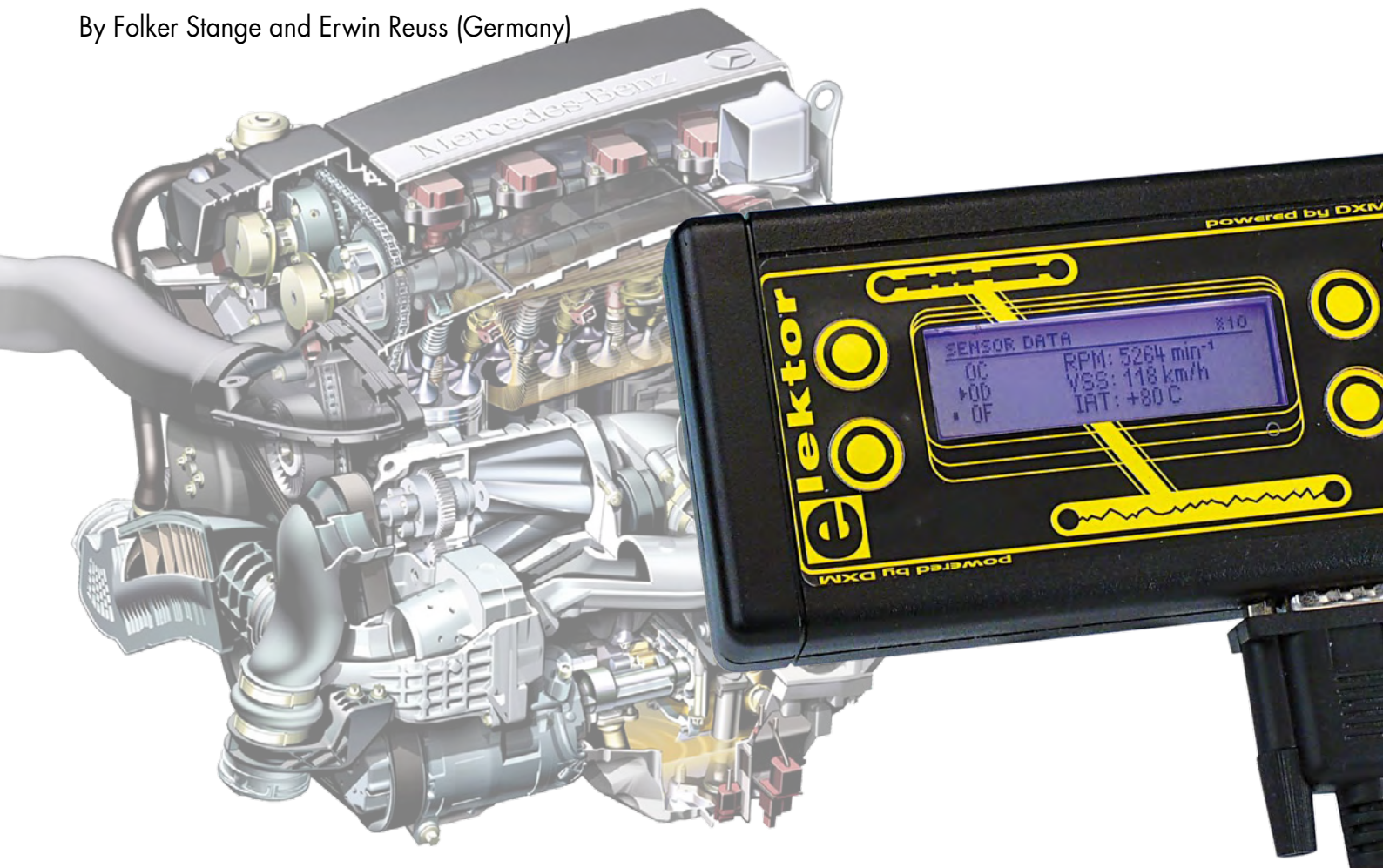
The compact OBD2 Analyzer in the June 2007 issue was an enormous success — not surprising for an affordable handheld onboard diagnostics device with automatic protocol recognition and error codes explained in plain language. Now enhanced with a graphical display, Cortex M3 processor and an Open Source user interface, the next generation of Elektor's standalone analyzer sets new standards for a DIY OBD2 project.

The key advantage of the OBD2 Analyzer NG is that it's self-contained and can plug into any OBD diagnostic port. It is handheld and lightweight, requires no batteries and works without a notebook computer, making it far more practical than a PC adapter. A glance at the comprehensive **Features** panel indicates that this new gizmo has plenty more to offer. The source codes for the firmware of the controller are clearly identified, providing readers keen on programming with complete control over the configuration of the GUI, the way that process

results are displayed and all manner of other functions. The hardware offers plenty of scope for expansion too, with the PCB already laid out for adding extra options such as a flash memory data recorder, a USB interface and a real-time clock (provision), in case you wish to implement an additional data logging function.

DXM

These days OBD diagnostic chips are based increasingly on CAN-capable processor platforms and can exchange data in various ways via protocol scan. The diagnostic chip 'pings' protocols sequentially and links up with the engine management system once the correct protocol has been recognized. The 8-bit PIC or AVR controller used as the hardware platform does, however, require rather a lot of additional hardware to meet the demands of all the multiple protocols. This makes the time ripe for a more efficient solution using a modern controller. This was the motivation for developing the Diamex DXM module [1], which is used in our OBD2 Analyzer NG.

The DXM module comprises an ARM Cortex M3 controller and the necessary surface-mount device (SMD) circuitry on a small printed circuit board (PCB). This module (**Figure 1**) reduces external circuitry requirements to a minimum and offers a unified yet versatile hardware basis for developing compact OBD2 projects. Designed as a universal OBD2 diagnostics and control unit with its own dedicated firmware, it can be connected direct to a vehicle's diagnostics port. A high-speed (up to 250,000 Baud) serial interface provides an external connection. At the heart of the module is the 32-bit ARM Cortex M3 CPU with a clock frequency of 72 MHz, 64 KB of flash memory and 20 KB of RAM. The environment comprises a K line protection IC, a CAN driver, MOS transistors for the PWM-based protocols, two signal lines and a dedicated power supply with over-voltage protection (**Figure 2**). The available I/O ports offer many possibilities over a modest number of lines, such as connection to a standard LCD display (SPI interface), which can even use dedicated text strings.

Firmware updates can be applied via the serial interface, so that standard-

Features

Hardware Handheld Analyser:
- Full graphics display 132 x 32 pixels
- RGB backlighting
- Convenient four press button control
- Power supply taken from onboard diagnostics port (12 V car battery)
- Uses standard OBD cable
- Convenient size (5 in. wide x 2.7 in. tall x 1 in. deep)
- Weight approx. 110 g

Hardware and software DX module (DXM-PCB):
- Hardware for onboard OBD2 control
- Firmware for onboard OBD2 control
- 3V3 tx-rx level
- Cortex M3 CPU (32-bit controller)
- 72 MHz internal clock rate
- Onboard 3V3 power supply for external device – max. 55mA
- Jumperless
- Bootloader
- LED indicators for onboard Connect and Data Stream
- Analog battery voltage measurement
- AT control set
- Supports all currently implemented protocols: PWM, VPWM, ISO9141-2, ISO14230-4 (KWP2000), ISO15765-4 (CAN, 11/29 Bit , 250/500 kBaud)
- Firmware update via ISP interface
- Rapid OBD connection

Open Source firmware functions:
- Graphical user control interface display
- Selection of vehicle data, PID list, error code list, VIN, MIL status
- Selection menu for active transmission control system (for vehicles with multiple transmission options, e.g. automatic gearbox)
- Saved error store (freeze-frames) for previous faults
- Expandable memory bank for sample error codes
- Erasable error store
- Live display of sensor data
- Acoustic signals
- Selection menu for scan mode (automatic or manual)
- Menu text in English
- Choice of direction (rotatable through 180 degrees)
- Controllable RGB backlight brightness
- Battery voltage measurement

Expansion options:
- USB port for data transfer or supporting use with PC
- Real-time clock (RTC) for data recording (e.g. time and date stamping)
- Adequate checklist flash memory (1, 2 or 4 MByte) for data logging functions

Open Source:
- Open Source firmware for the controller
- ISP interfaces accessible for AT90CAN128 and AT90USB162
- Other firmware can be substituted if required
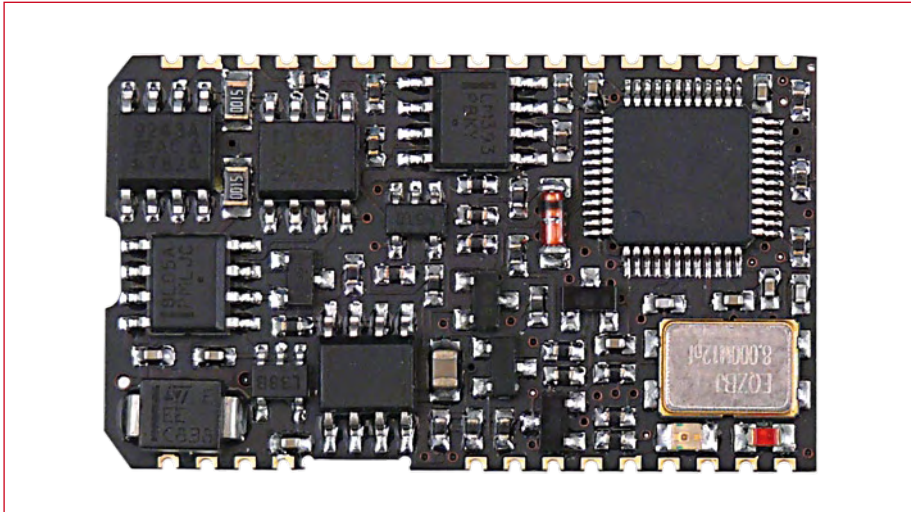- Demo firmware for 'Speedometer with warning functions'
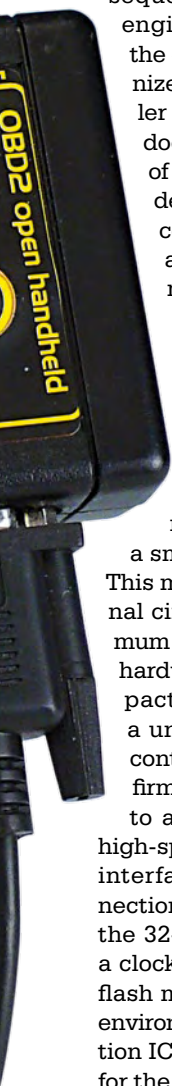
Figure 1. The DXM module equipped with ARM Cortex M3 processor used in the new OBD analyser.

ization alterations or new functions can be made without difficulty at a later stage.

The module is built on a four-level multi-layer PCB (20 mm × 34 mm) using surface-mount technology. The pitch of the connector pins is 1.9 mm, arranged in two rows of 13 and 17 pins. These pins carry 12 V and ground, the OBD2 lines, the serial lines RX, TX, RTS and CTS, the SPI interface, the 8 MHz clock frequency, the 3.3 V supply voltage together with some control wires. An 18 V suppressor diode protects against voltage surges from the vehicle.

In parallel with the two status LEDs it is possible to connect additional outboard low-current LEDs using a series resistor of 220 Ω to the cathodes and 3.3 V to the anodes. The red LED comes

Is your car equipped for diagnostics?

Whether and to which extent OBD2 diagnostics will work depends on your car. Generally speaking, you're OK if you run a 2000-model (or newer) petrol car or a diesel built since 2003, although there are plenty of exceptions. You will find several checklists online, such as the one compiled by Florian Schäffer [9].

OBD2 can talk to a whole range of test devices, but only if they use controllers and components designed to be compatible with OBD2. The engine management system and elements of the exhaust system are crucial to this. In many new vehicles switching on the ignition activates the engine management system and also the automatic transmission system. On vehicles equipped with multiple management systems the OBD2 handheld described here may well be able to select

the particular active management system to be analyzed.

OBD2 does not support safety-critical components, in-car convenience (comfort and entertainment) electronics or maintenance reminders. As a result, you cannot use the diagnostics interface to reset airbag indications or ABS lights. Commands for this kind of equipment simply have not been standardized and each car manufacturer invents its own procedures. Maintenance intervals are also model-specific and cannot be unified. Convenience features are a key differentiator between cars so there is no commonality among the control and command sets either.

Note that maintenance interval warnings are reset in the workshop (the handbook will give full details). Setting up convenience features generally calls for special tools. Safety features are enabled over the remaining pins of the diagnostics port in ways that differ according to the make and model.
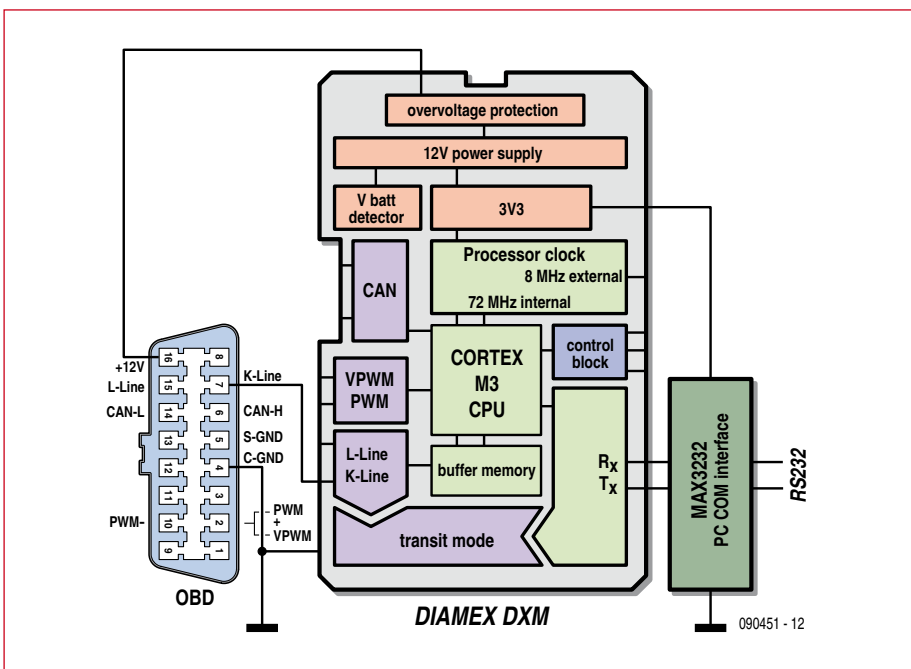

Figure 2. Block diagram of a simple OBD2 adapter with the DXM module.

on when a connection has been established with the engine management system and the green LED flashes during data transfer.

Since the beginning of 2000 most European vehicles have used the protocol ISO9141-2 or KWP2000, using just one communication wire for data transfer. This makes it very simple to create a diagnostics adapter: just connect the K line and the car's onboard supply (+12 V and ground) to the DXM and make the connection to the PC or Notebook via a MAX3232 level converter (**Figure 3**) to start the diagnostics process. A simple terminal program [2] is perfectly adequate for this. Connecting up to the other wires or lines makes other protocols readable. A special OBD software package such as 'moDIAG express' [3] rounds this off. The comprehensive command set of the DXM is largely backwards-compatible with older diagnostics chips like AGV
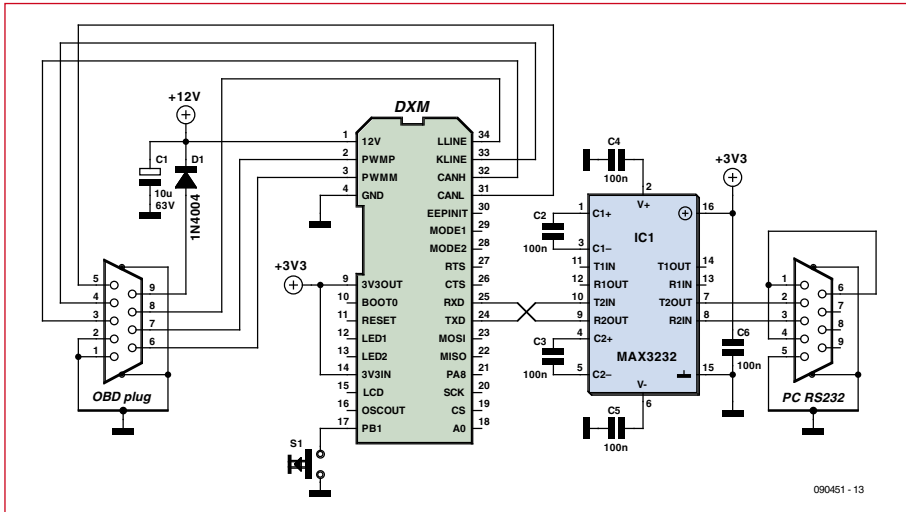
Figure 3. Basic circuit for an OBD2 interface with the DXM module.

OBD2 (OBD-II) explained

OBD, as you must have guessed, stands for On Board Diagnostics and in this particular context relates to a vehicle's self-diagnostic and reporting capability. OBD2 (also written as OBDII) is an enhancement to the original specification, improving both capability and standardisation.

OBD systems provide vehicle owners and repair technicians access to state of health information for various vehicle sub-systems. Although basic OBD systems appeared in the 1970s, it is only in recent times that protocols and connectors have become standardised and made mandatory. Within the UK OBD became mandatory for all new UK car designs in 2000, existing cars in 2001 and diesels in 2004 (although most manufacturers implemented it before these dates, since many cars were also sold in the US, where the state of California began to require emission control systems on 1966 model cars). Further information at http://en.wikipedia.org/wiki/On-board_diagnostics, http://en.wikipedia.org/wiki/Obd2#OBD-II and http://www.obd-codes.com.

and ELM but will not tolerate slow processing, since processing the many AT commands involved is very time- and processor-intensive. A comprehensive data sheet giving the AT commands and the Eagle Library 'footprint' for the DXM can be downloaded gratis from [6]. This website also has information on the 'transit' mode, which enables proprietary PC software to talk to the vehicle's onboard electronics via the K line without involving an OBD2 connection. An 'extended' version of the DXM module additionally offers a CAN low-level mode, which in conjunction with specialist software ('Can-Hacker' [2]) can be used fairly easily to analyze
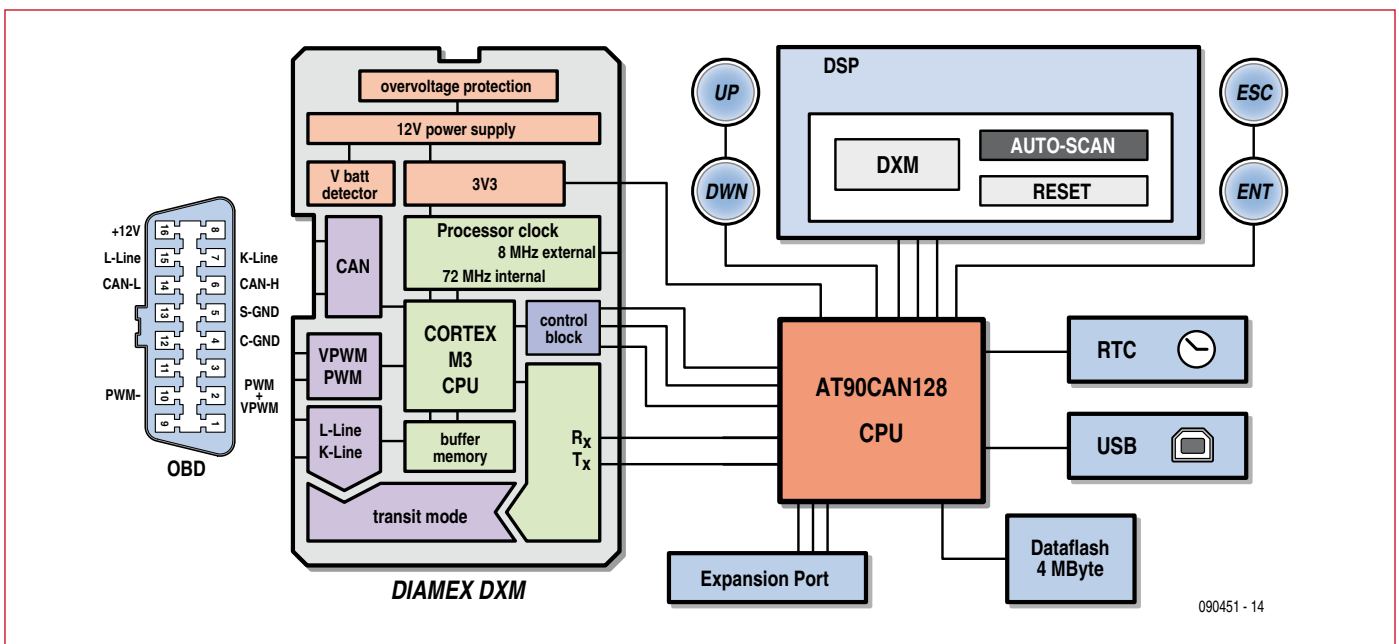

Figure 4. Block diagram of the new handheld OBD2 analyser. Flash memory, RTC and USB connector are optional extras.

CAN data traffic.

Handheld analyzer puts DXM through its paces

Simply amazing results are achieved when you put an AVR or PIC controller in command of the DXM board. Consequently the new OBD2 analyzer (**Figure 4**) employs an AVR controller plus command software written in C as an Open Source project. That's the reason why the somewhat cryptic slogan 'handheld open' appears on the front panel of the case.

As already mentioned in the introduction, the power supply is taken from the car's battery, which is connected to the OBD2 diagnostics port at Pin 16 (12 V) and 4/5 (ground). It's easy then to keep the device in the glove compartment or in the tool kit, ready for use whenever it is needed. The hardware is competitively priced and very effective. Equipped with a full graphic display and four touch switches inside a compact and robust ABS case, it is highly intuitive and ergonomic in use. The most important components on the circuit diagram (**Figure 5**) are the microcontroller AT90CAN128 used for controlling operations, the 132×32 pixel graphic display, an 8 MHz crystal, four touch switches for operations, a micro loudspeaker, an ISP programming port, the PWM control for the backlight LEDs, a switching regulator and a low voltage-drop regulator for the power supply, plus of course the DXM module itself. The switching regulator IC2 provides 5 V for the backlight LEDs. The low drop regulator LD1117 supplies 3.3 V for the AVR controller, its peripherals and the DXM board.

The diagram also shows, in the section marked off, the optional components for expansion with an additional data logging function (see panel **Expansion Components**).

Construction…

All components are fitted to the printed circuit board (**Figures 6 and 7**). Since the circuitry cannot be realized without using surface mount devices, these are already fitted on the PCB. Not included, however, are the SMD

components required for the optional expansion (the 'Expansion Components'), which are not shown in the

component list either.

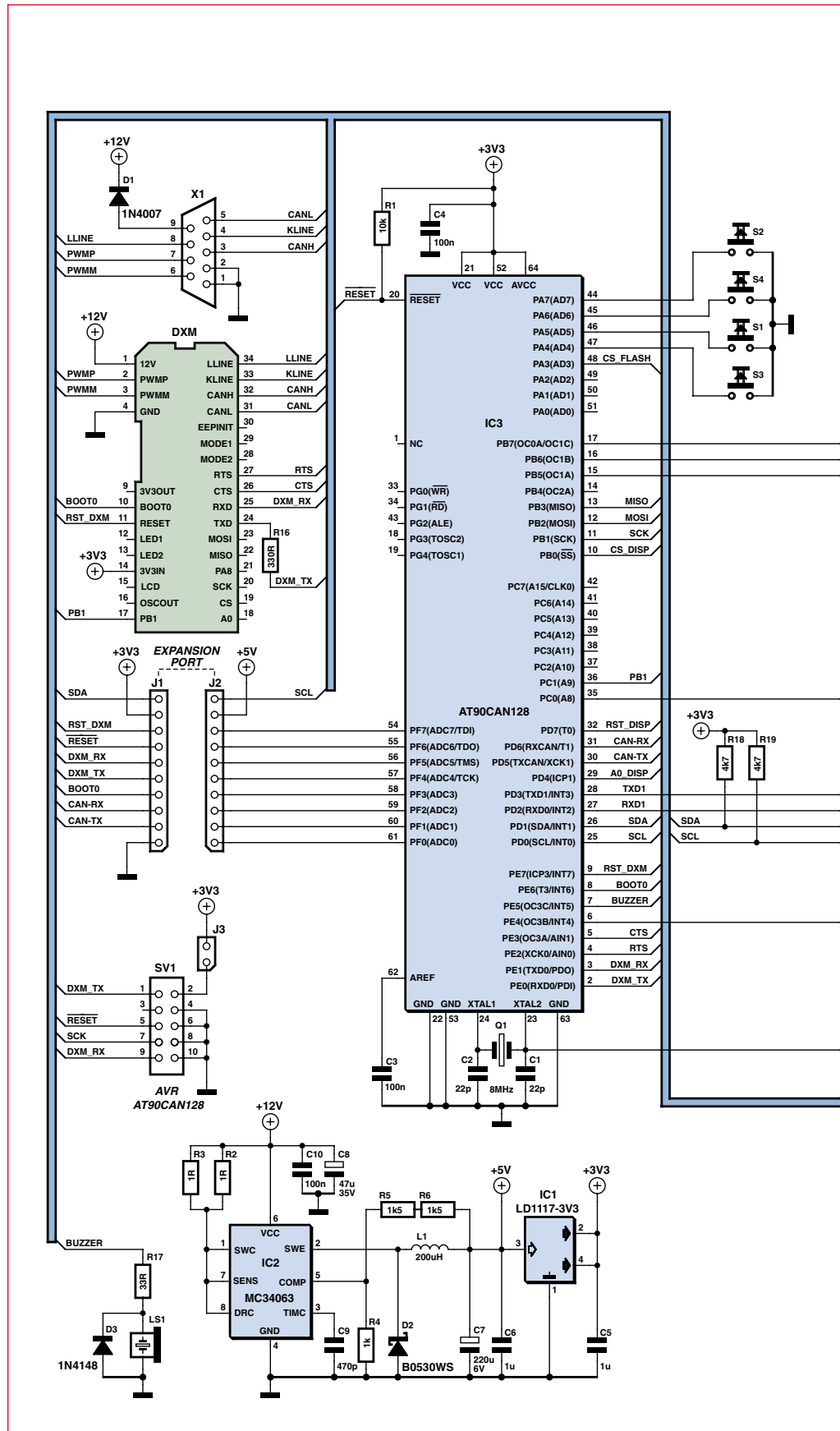The switching regulator IC2 is functional once you have fitted the 220-

Figure 5. The DMX module lies at the heart of the handheld analyzer's circuit.

μH choke (L1). L1 needs to be angled round by 90 degrees to face the center of the PCB. The next placement task is to solder the DXM module precisely into position on the main board. Begin by soldering just one pin. Make sure the module lines up properly, so don't solder everything solidly straightaway. Just a brief and sensitive touch of the soldering iron with the minimum of solder is best. When all the unsoldered pins line up exactly with their holes, then the other pins can be soldered firmly and the solder on the first pin reflowed to relieve any tension.

Before mounting the graphic display we need to install the backlight. This is not difficult: just separate the sub-board already fitted with SMD LEDs (LED1, LED2, LED3) from the main board by breaking it away gently and carefully. Then plug it into the recess provided. The LEDs should face the center of the board on the display side. The four pads should be soldered together (see **Figure 8**). After this fit the touch switches flush against the PCB, so that a clean pressure point can be felt.

Now we need to embed the satin-finish acrylic sheet diffuser into the laser-cut packing material supplied (the edges should be bent up first and let into the slots in the diffuser) and fixed at the four corners with narrow (3 mm wide) pieces of adhesive tape (Sellotape or similar). The diffuser is positioned on the PCB and the display placed above, after first removing the protective film.

The display is the centerpiece of the unit, so solder only one pin to begin and check all round that it fits properly. Once soldered into place, it is very difficult to adjust afterwards and is easily damaged.

The Sub-D connector is slid onto the board sideways and aligned accurately (**Figure 9**). Before you do this, make sure the stand-off sleeves have been screwed tight onto the connector. Last in line is the micro loudspeaker, the opening of which should face in the direction of the backlights.

Assembling the case is very simple by comparison. Four aluminium plungers are set above the touch switches and the PCB is affixed with the screws supplied. Finally the case lid is placed on top and screwed together.

…and commissioning

To begin we can apply a voltage of 12 V at the Sub-D connector (Pin 9 = 12 V; Pin 1 or 2 ground). The interface boots up and draws around 100 mA of current. Current requirement varies according to the brightness setting of the backlight, as this draws the most juice. A certain amount of warmth is completely normal with high-brightness LEDs. Shift-
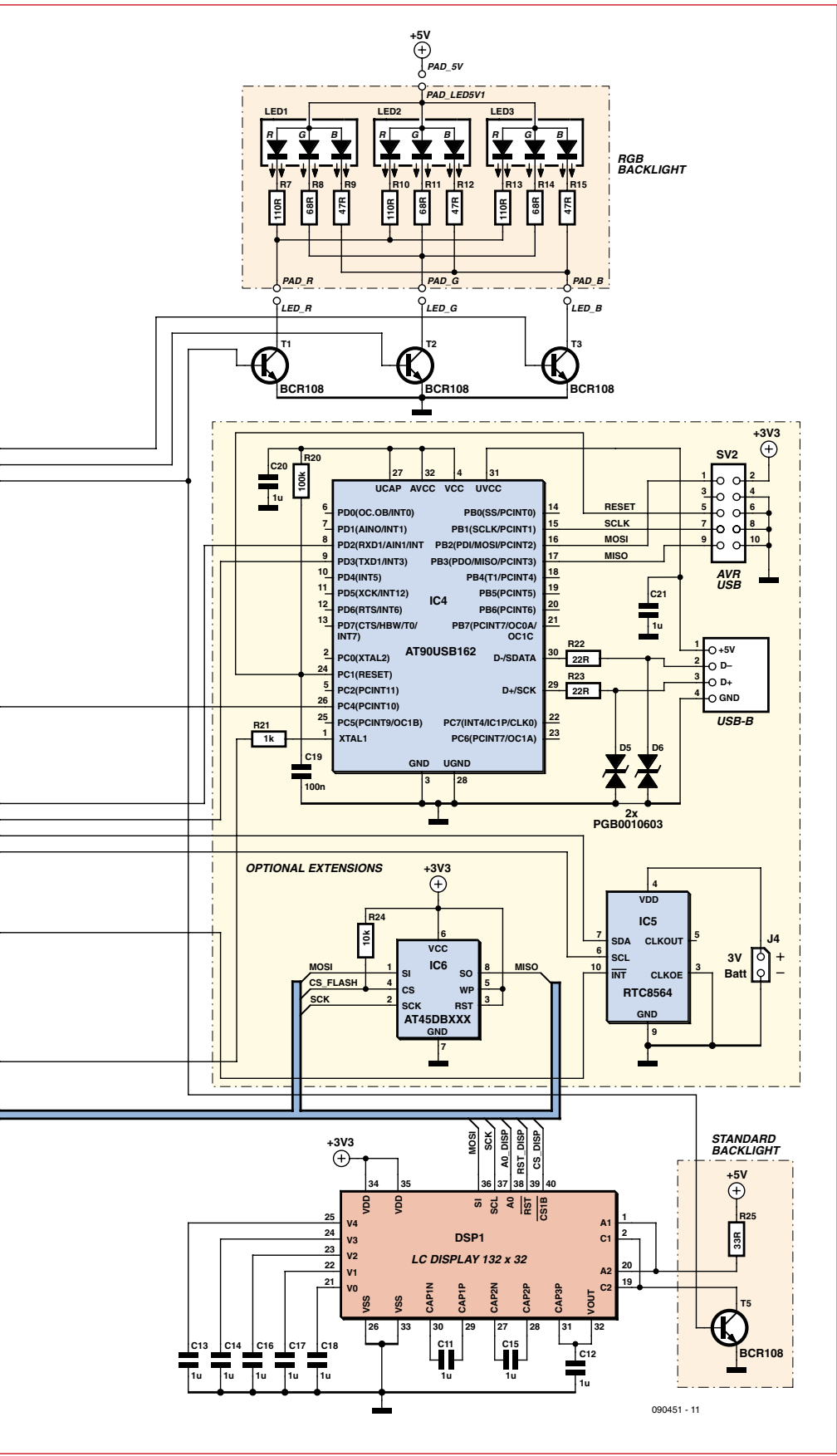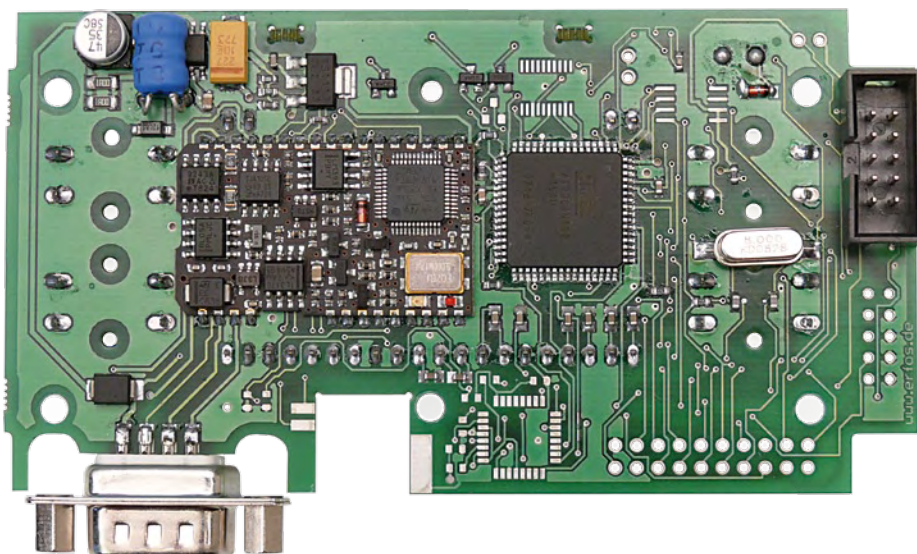
Figure 6. Component side of the prototype board.

ing the default values alters the current consumption measured.

Now we can start the in-vehicle test. The most important requirement is an OBD2-capable vehicle with the 16-pin connector (see panel **Is your car equipped for diagnostics?**).

To avoid any problems start the car first and only then connect the OBD2 Analyzer to the vehicle's diagnostics port. The background to this is the timeout of five seconds embedded in the ISO and KWP protocols, which prevents connection with the engine management system if you get the sequence wrong. It is best to begin with an automatic scan. Please pay careful attention to the advice in the panel **Safety Tips**.

Operation of the Open Source firmware (available free) has been structured to be entirely intuitive. Consequently, since the display can be turned through 180 degrees, there is little

COMPONENT LIST

SMD components (prefitted on board):
C1,C2 = 22pF
C3,C4 = 100nF
C5,C6,C11–C18 = 1µF
R1 = 10kΩ
R2,R3 = 1Ω
R4 = 1kΩ
R5,R6 = 1.5kΩ
R7,R10,R13 = 110Ω
R8,R11,R14 = 68Ω
R9,R12,R15 = 47Ω
R16 = 330Ω
R17 = 33Ω
R18,R19 = 4.7kΩ
Q1 = 8MHz quartz crystal
D1 = 1N4007

D2 = B0530WS
D3 = 1N4148
T1,T2,T3 = BCR108
IC1 = LD1117 3V3
IC2 = MC34063
IC3 = AT90CAN128
LED1,LED2,LED3 = RGB LED, Kingbright
  type KAA-3528SURKVGAPBA

Components to be fitted separately:
DSP1 = LCD 132 x 32
DXM = OBD module
J1,J2 = 10-way SIL pinheader
J3 = 2-way pinheader
L1 = 220µH (choke coil)
LS1 = miniature loudspeaker
S1–S4 = pushbutton
SV1 = 10-way boxheader
X1 = 9-pin sub-D plug with solder buckets

Fastenings and accessories:
4 x pushbutton plungers
4 x case screws
5 x PCB screws
Diffuser
Cardboard packaging
Case with front panel
Sub-D connector fixings
Standard OBD2 cable

Sources of supply:
Component set # 090451-71 available from the Elektor Shop contains all necessary components and the PCB with SMD components pre-fitted together with the case (with custom front overlay), fixings and the standard OBD2 cable. See Elektor Shop advertisement at the back of this issue and www.elektor-usa.com/090451.

Figure 7. On the other side of the board are just four touch switches, the display LCD and the miniature loudspeaker.

point in labeling the touch switches. The buttons to the left of the (visible) display are 'UP' and 'DOWN', at upper right 'CANCEL' (ESC) and below this 'INPUT' (OK). This makes navigation nice and simple. From a software perspective the buttons can of course be programmed differently and used in other ways. With a view to worldwide usability the menu text of the software starts in English. All details like this are accessible in the source code, so it is easy to alter the user language.

Wide open for further development

The disclosed firmware [4] of the AVR controller is a good basis for automotive diagnostics. There is already free 'speedometer firmware' available, which transforms the analyzer into an accurate speed measuring device with presettable
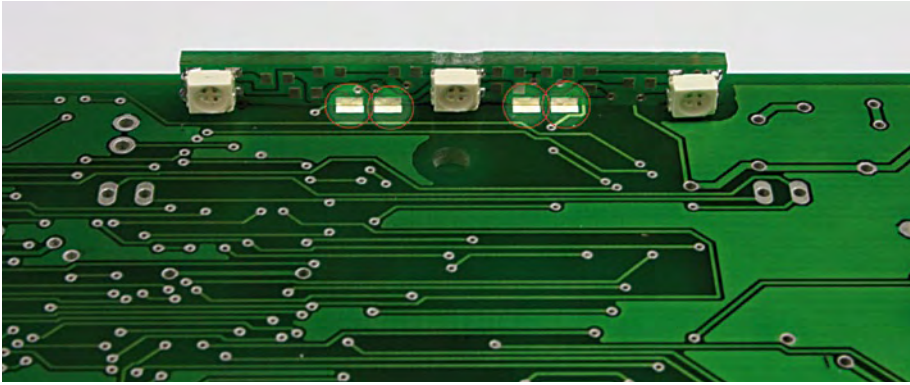
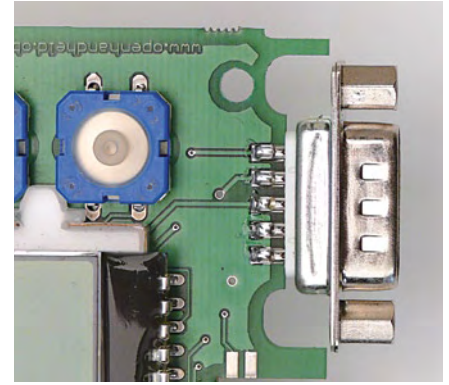Figure 8. This is how the LEDs for the display backlight are fitted.

Figure 9. The Sub-D connector needs to be soldered on both sides of the PCB.

threshold values that actuate acoustic and optical alarms. This sample software demonstrates that the analyzer hardware is not restricted to diagnostics. As the firmware is Open Source, users are free to alter and expand this themselves, also make enhancements available to other users. A commercial approach is also feasible, perhaps for LPG gas conversion or additional convenience features, such as speed-dependent central locking or audible warning of speed. Other ideas include a stopwatch function, GPS navigation, GPRS redirection, event-driven relay functions or even a dynamometer, an acceleration meter and a small onboard computer. The expansion port could be used to scan additional sensors (e.g. temperature). Average fuel consumption is another popular subject, with an interesting approach to be found at [5]. A CO_2 monitor and an 'eco-meter' would make very topical projects to support greener motoring.

The programming connections of the AVR are available via an Atmel-conformant 10-pin ISP interface. You need to set jumper J5 if the programming adapter is expecting a supply voltage on Pin 2. The authors used a low-cost programmer [6] for this.

A data bank of error codes is already included in the firmware but this too can be altered, for instance to expand the information already entered. The only limitation is the amount of flash memory available in the controller. Further tips can be found among the comments included in the source code.

New firmware can be burnt in rapidly using an ISP programmer and a compatible GUI using **AVR Dude** [6], for which the original program is available as hex code.

Many of the development ideas have already appeared on the DXM Platform [1] and in a forum [7]. This and the Elektor forum [8] are great places for swapping ideas and suggestions, also finding construction help and application tips.

(090451-I)

[1] www.dxm.obd-diag.net
(in German only, use Google to translate)

[2] www.er-forum.de/obd-diag-dl
(starts up in German but will display in English if you register [click on Login and select Register] and set Sprache to English)

[3] www.modiag.de
(in German only, use Google to translate)

[4] www.elektor.com/090451

[5] www.lightner.net/lightner/bruce/Lightner-183.pdf

[6] www.stange-distribution.de
(English version available)

[7] www.forum.obd-diag.net (German), www.obdii.com/forums/ubbthreads.php and www.obd-codes.com/forums/ (English)

[8] www.elektor.com/forum

[9] http://carlist.blafusel.de
(English version available)

Expansion components

You have the option of equipping the board with a number of additional components ready for future developments:

• **RTC – Real-time clock**
RTC8564 with I2C Interface. A 3-V lithium battery is connected at J3. Using the type provided with solder tags, solder it with two wires to J3 (check polarity) and sleeve the battery with heatshrink tubing. With the battery protected like this, you can fix it to the back of the case with duct tape. The RTC8564 is connected to the controller via the Interrupt line in order to trigger time-controlled events (time-stamp facility).

• **Dataflash AT45DBXXX**
Choice of 1.2 or 4 MB memory capacity. A USB connection to an AT90USB162 processor would make sense if you are contemplating using professional software [3] to diagnose or transfer the stored data. A B-type USB connector needs to be retrofitted in this case, fitted at right angles. Mechanical stability is assured by soldering the screen to the circuit board. The casing is not deep enough for fitting it the normal way round.

• **Expansion port**
20-way expansion port for additional features with all connections from Port F, the I2C lines SDA and SCL, together with the CAN Bus connections from the host controller.

For fixing the optional SMD components the authors recommend: 1. Tinning the pads with 0.5 mm solder (Sn60Pb40) and putting the part into place approximately. 2. Using a gas soldering iron fitted with a hot air jet, heat the pins and align the component into the correct position with a watchmaker's screwdriver. Practise this technique first on scrap PCBs, such as a defective CD-ROM drive.

The use of these optional components assumes that suitable software has been optimized for them (the current Open Source firmware does not yet have any expansion capabilities implemented).

GPS Datalogger
Basic Stamp linked to Google Earth

By Chris Savage (USA)

There are plenty of projects out there that deal with GPS and microcontrollers, many of which make use of the data for such things as navigation. The Robo-Magellan competition is one such application. But what if you wanted to visualize the path your robot took through such a course? Better yet, what if you wanted to log the path of a bike or car trip? Here's how.

Sure, you could export the data for processing into some other application that does this, but you could also make use of a very popular application called Google™ Earth.

Plotting a course

Google Earth is a virtual globe program that can show you, at a continent, state, city or even street level, various locations on the planet, based on satellite images. You can follow streets as well as see popular locations. Provided your area's been photographed recently by one of those low-orbiting birds, using Google Earth, you should be able to see houses, including gardens in the back yard in your neighborhood.

One interesting thing about Google Earth is that it's available free and supports Keyhole Markup Language (KML) which is an XML-based language for expressing geographic annotation, maps and even 3D objects. One feature of KML is the ability to plot a path using GPS coordinates. So if you

have a source of coordinates, such as a GPS module, you can create a KML file with a little help from a BASIC Stamp® microcontroller module and a Parallax Memory Stick Datalogger.

The author took a short drive around the vicinity of the Parallax home base in Rocklin, CA. A screenshot of the Google Earth plot of the journey is shown in **Figure 1**. The KML file of this trip can be downloaded with the source code and is called LOGDATA. KML. You can load it into Google Earth for a detailed look at the trip or you can open it using Notepad to see the structure of the KML data. Note that due to the margin of error in accuracy of GPS as well as differences in the Google Earth terrain map, at some points it looks like the car was driven off the road or in the oncoming lane. Rest assured it was driven safely!
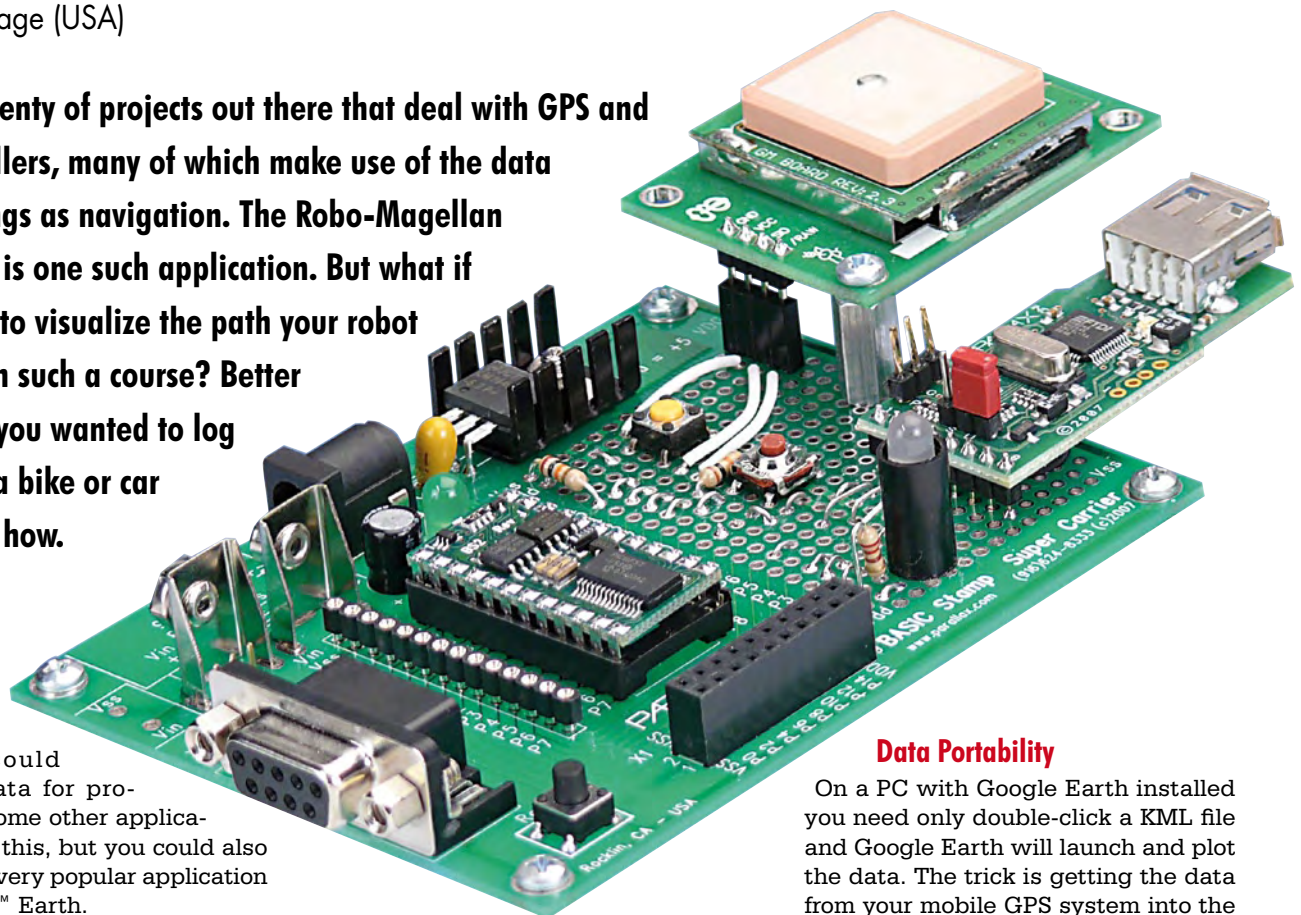
Data Portability

On a PC with Google Earth installed you need only double-click a KML file and Google Earth will launch and plot the data. The trick is getting the data from your mobile GPS system into the PC in the correct format. Parallax manufactures a Memory Stick Datalogger, which is essentially a USB host bridge using the Vinculum chip from FTDI. The Memory Stick Datalogger allows you to use a portable storage device such as a USB thumb drive to store the GPS data. Since the FAT (file allocation table) system happily is supported, you can save the file in the native format and it will be directly readable by the PC and Google Earth. Since KML is an XML-based language there are a lot of tags similar to those used in HTML. For now we'll simplify things by saying that the important parts of the file we are creating are pretty much always going to be the same.

Decoding KML

Google has an extensive specification for KML at [1] so no need to go into all the

details here. The language is very powerful and warrants a look if you're interested in creating files for Google Earth. For our purposes we will only want to create paths. In order to do this we will need three pieces of information:

1. First we will need what's best called the 'header information'. This is essentially all the information Google Earth will need to know, such as the version of the language, source URL, line types and colors, mode, etc.
2. Next, we need the coordinates themselves. Oddly Google Earth expects the data as Longitude, Latitude, then Altitude. Conversely, [2] uses Latitude then Longitude. In any event this will be the part of the data we will be supplying as we go.
3. Finally the file will need the closing tags, called the 'footer data' here. The trick now is to get a BASIC Stamp to put all this data onto a USB thumb drive so it can be read by a PC.

BASIC Approach to KML

For an understanding of the discussion below, you will find it useful to refer to the 'GPS Datalogger v. 1.0' code listing contained in archive file # 081079-11.zip at [3].

The program is very straightforward and includes many pieces of code from other programs the author wrote for the various hardware used.

The BASIC Stamp module has 2 K of EEPROM available for program and data to be stored. The original concept of doing this by hard-coding the data into SEROUT statements was put to the test. However, this left no room for making the program do anything other than writing the data straight from the GPS Module, so it was decided to create a data table in EEPROM of all the header data used to create the KML file. In order to do this, quotation marks had to be coded into ASCII values since quotes are used to enclose text. To trim a few more bytes, instead of using a CRLF (carriage return / line feed; remember?) for each line a single null (zero) byte was used, being replaced by the program whenever encountered. There was also a need to change the Altitude mode easily. After trying several methods it was ultimately decided to hard code the text section of that one command based on a mode constant being 0 or 1.

Figure 1. Screenshot from Google Earth of a car trip around Parallax Inc. in the Rocklin CA industrial area, as recorded with the GPS logger.

Since this creates a break in the header block the value 255 is used to separate blocks of KML code, see **Listing 1**.

Building the GPS Datalogger

Figure 2 shows the schematic for the GPS Datalogger. For the sake of completeness the drawing includes the parts on the Parallax Super Carrier Board used for building the prototypes. Use the Component List to do your shopping. **Figure 3** shows the constellation of boards that make up the logger. In **Figure 4** you can see the 'user I/O' parts (switches, resistors,

Practical use

The Parallax Memory Stick Datalogger and Parallax GPS Modules are used, so the first thing that happens is the program initialises and establishes communication with the Datalogger. A bi-colour LED is used to indicate status and blinks green while the Datalogger is initialized. If no USB drive is connected the program will wait until it is connected before moving on. Once the Datalogger is initialized and the drive is identified the program will write out the KML header data. This will take several seconds since it is being read byte-by-byte from EEPROM and written out to the drive. While this is happening the LED will be fast-blinking red.

Once the data is written the program will attempt to get the satellite signal status from the GPS module to see if the signal is valid. During this process, which may take up to 2 minutes to complete, the LED will slow-blink red. Once the satellite signal is valid the LED will turn solid green to indicate that

the system is ready to start logging data. There are two pushbuttons on the GPS Datalogger Board. Pressing the yellow one (SW2 on port line P15) until the LED turns red will start the logging process. The GPS Datalogger writes data to the USB drive at a fixed rate of about one sample every three seconds. If the GPS signal is lost during logging, the last coordinates are written at the same rate. This makes it possible to determine how long the data was accumulated, even during times when the signal is not valid.

At any time once the USB drive has been initialized you can hold down the red button (SW3 on port line P12) until the LED starts blinking red. This will write out the KML footer data and close the file. When it is safe to remove the USB drive the LED will blink red/green alternately. It is not advisable to remove it at any point before this, since corruption of the data or even the file system is possible.
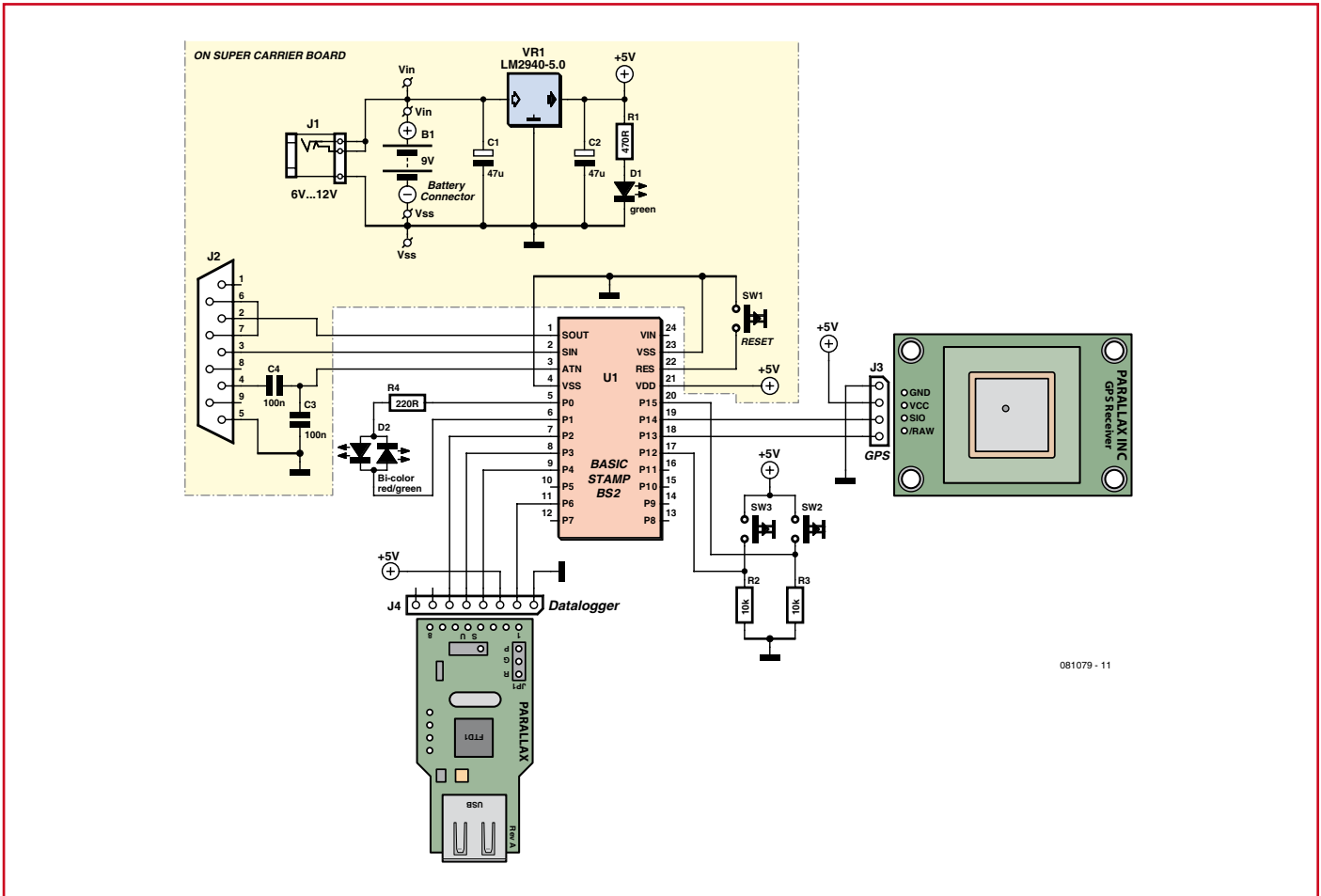
Figure 2. GPS Datalogger schematic. The yellow area indicates the Super Carrier board.

COMPONENT LIST
Parallax part numbers in brackets

Super Carrier Board (#27130) (optional,
  see text)
BASIC Stamp 2 Module (#BS2-IC)
GPS Module (#28146)
Memorystick Datalogger (#27937)

2x 10kΩ resistor
220Ω resistor
2x tactile switch (#400-00002)
Bi-Color LED (#350-00005)

Optional components
1 rubber foot (included with Super Carrier
  board)
1 12mm (0.5") M3 threaded PCB standoff

 (US: F-F, 4-40)
2 M3 screw (US: 4-40, 1/4") (#700-00028)
4-pin SIP Socket (#450-00401)
LED standoff (#350-90000)

bicolor LED) soldered in place on the prototyping area of the Super Carrier. Also note that a mounting hole was pre-drilled for the GPS Module and a 4-pin SIP socket was installed. While not required, this is recommended. You will also need a 12 mm (0.5") threaded standoff between the board and the GPS Module.

A rubber foot from the Super Carrier Board was cut in half and used as a pad to hold up one end of the Memory Stick Datalogger so that there is extra support when inserting a USB drive into it.

At this point it is a good idea to run the test programs included in the source code package. The programs allow you test individual subsystems of a project

while building it to simplify debugging if there is an issue. There are programs for testing the GPS, Datalogger, LED and buttons.

The 'GPSTest.bs2' file is a modified version of the GPS Demo program. The 'DataloggerTest.bs2' file is a modified version of the Datalogger Demo program and the 'ButtonLEDTest.bs2' program was quickly written for the purposes of testing those components. Downloading the 'ButtonLEDTest.bs2' program will blink the bi-color LED red 5 times, green 5 times, then alternate red/green 5 times. After that it will report whenever either of the buttons is pressed. You should run these programs to make sure all the hardware is functioning before moving on.

A hungry client

Once everything is assembled you can download the file 'GPS Datalogger V1.0.bs2' and start logging data. The Super Carrier Board powers everything from its own 5 volt regulator, so for testing a cable which plugs into a cigarette lighter or auxiliary connector in a car is conveniently used to power the board. Note that the cigarette lighter socket normally does not supply power when the ignition is switched off.

Placing the board up in the centre of the dashboard nearest the window will yield the best results and you will rarely lose signal. Our advice is if you're going to test it like we did, get some Velcro® or temporarily tape it in

place. **All regulations in force in your country governing minimum unobstructed windscreen area should be observed.**

The whole system consumes about 200 mA of current depending on the USB drive used. This causes the voltage regulator on the Super Carrier Board to become quite warm. Be careful of grabbing the board near the regulator. Another note about this is due to current consumption **this unit cannot be powered from a regular 9 volt battery**. You will need a high capacity power source, or if used in a car or on a motorbike you can use the cigarette lighter or auxiliary connector and you should be okay. For a bicycle a generator may even be possible.

Final Thoughts

The code as supplied for the project (downloadable at [3]) is relatively large and uses all but six bytes of the EEPROM space. The main reason is the number of very verbose DEBUG statements, which are not needed and can be taken out or commented out.

Many people like to see some feedback when testing things for the first time and the author was inclined to provide that although DEBUG statements (or anything with added text) use a lot of program memory.

Note that by default the data logged ignores elevation (altitude) data. If you wish to enable this you can change the mode constant toward the beginning

Listing 1. GPS datalogger code (extract)

```
Write_Header:
  SEROUT TX\CTS, Baud, [$07, $20, "logdata.kml", CR] ' Delete File
  GOSUB Get_Data                              ' Purge Receive Buffer

  SEROUT TX\CTS, Baud, [$09, $20, "logdata.kml", CR] ' Create File
  GOSUB Get_Data                              ' Purge Receive Buffer

  counter = 0                                 ' Reset Byte Counter
  DO
    TOGGLE LEDA                               ' Blink Red LED
    READ Header + counter, ioByte             ' Get Next Byte From EEPROM
    IF ioByte = 0 THEN                        ' Replace 0 Bytes w/CRLF
      SEROUT TX\CTS, Baud, [$08, $20, $00, $00, $00, $02, $0D,
                                  $0D, $0A, $0D]
    ELSEIF ioByte = 255 THEN                  ' End Of Data Block
      EXIT
    ELSE
      SEROUT TX\CTS, Baud, [$08, $20, $00, $00, $00, $01, $0D,
  ioByte, $0D]
    ENDIF
    counter = counter + 1                     ' Increment Pointer
    GOSUB Get_Data                            ' Purge Receive Buffer
  LOOP
```

of the program from 0 to 1. Be warned that GPS modules don't always match the elevation maps used by Google Earth and this can sometimes cause your path to disappear under the terrain, making it invisible. This will cause breaks in the path. It was also noted sometimes it looked like the vehicle was several meters above the ground (must be a flying car). You can also change the mode in the KML file directly by using notepad. If you change the word 'clampToGround' to 'absolute' and reload it into Google Earth it will now show altitude data. This data is shown as a line extruded from the virtual ground.

More buttons may be added to this project, adding flexibility to the hardware for those (we know) are going to do more with this. One possible application would be tracking a child driver when borrowing the family car. The code and hardware could be slightly modified to start logging when the ignition is on.

Currently time/date stamping is not supported, but in reviewing the KML specifications it seems it is available as an option to use that data and it is readily available from the GPS Module. Perhaps we will see what some of you come up with. You can post your findings and improvements to the discussion forums accessible via www. elektor-usa.com. Drive carefully!

(081079-I)

Internet Links

[1] http://earth.google.com

[2] http://maps.google.com
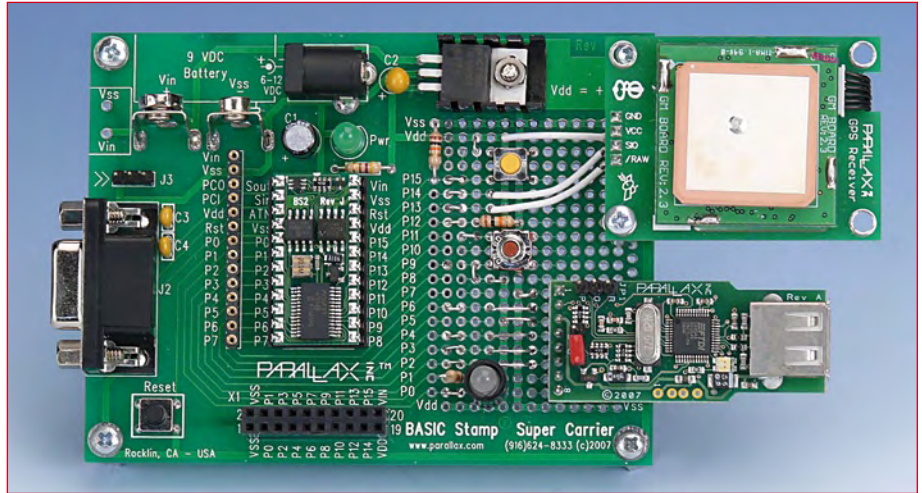
[3] www.elektor-usa.com/081079

Figure 3. The prototype less USB stick normally plugged on to the data logger.
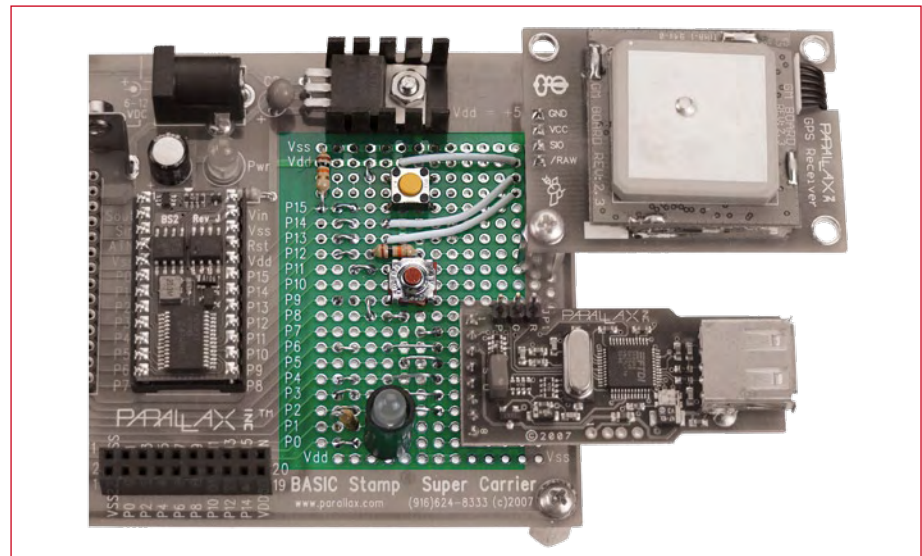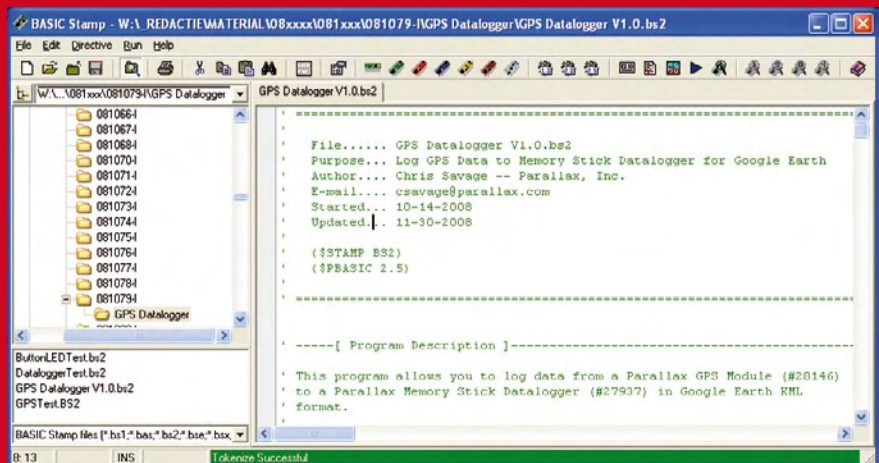

Figure 4. A closer look at how the 'bare bones' user interface is built on the Super Carrier board. For your convenience everything except the prototyping area has been greyed out. Note the PCB standoff used to secure the GPS module to the board.

BASIC Stamp programming

Programming the BASIC Stamp (BS) plug-in module is no Rocklin Science and based on a free tool called Basic Stamp Windows Editor, which is also available in Mac and Linux flavors. The latest version may be found at http://www. parallax.com/tabid/477/Default.aspx.

Basically, all you have to do is connect the BS dev board to the PC using a standard RS232 cable, run the BS Windows Editor program, load the file you want to transfer to the BS (it has the .bs2 extension), run a quick syntax check on it, and if all is beyond reproach (message: 'Tokenize Successful'), download it to the BS.

R32C Application Board

Ready-to-use 32-bit microcontroller and OLED display

By Marc Oliver Reinschmidt (Germany) and Jens Nickel (Elektor Germany Editorial)

We recently announced that we would bring you an R32C starter kit project, including not only its own power supply, I2C and SD card interfaces, but also an OLED display panel. Here we deliver on our promise: see for yourself with our example oscilloscope project!

Most Elektor readers will surely remember our R8C project [1]. The super low cost kit included the processor board and a toolchain by download, and the series continued with an application board in March 2006 [2]. This motherboard, onto which the processor board could readily be mounted, greatly simplified the development of a wide range of applications. In May of that year the series continued with a simple oscilloscope application [3]. As well as being a slightly unusual project, it clearly demonstrated the great potential of the microcontroller.

What could be more appropriate than an update of that project as part of our series on the R32C? This 32-bit device offers considerably more processing power than its forebear, while still remaining largely code-compatible with it. New is a floating-point unit, which makes dealing with measured quantities much simpler and faster. And finally we have the OLED panel, whose operation we discussed in the May 2009 issue of Elektor [4], which does a splendid job of displaying our oscilloscope traces.

Power supply

Looking at the circuit diagram (**Figure 1**), anyone familiar with microcontrollers will quickly recognize the main features of our R32C application board. Let us start with the power supply: voltage regulator IC3 produces the 3.3 V operating voltage for both the microcontroller and for the OLED panel supply circuit.
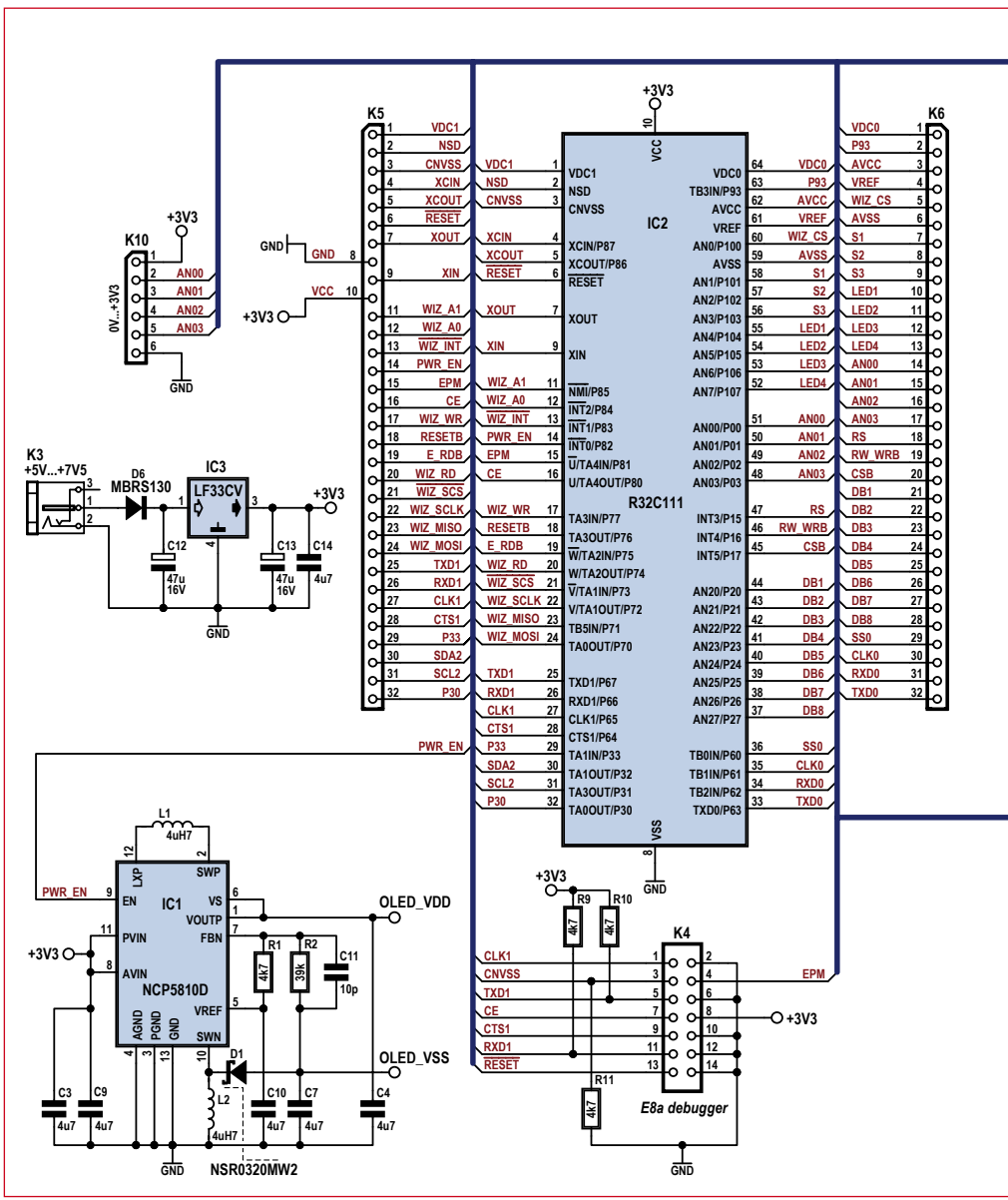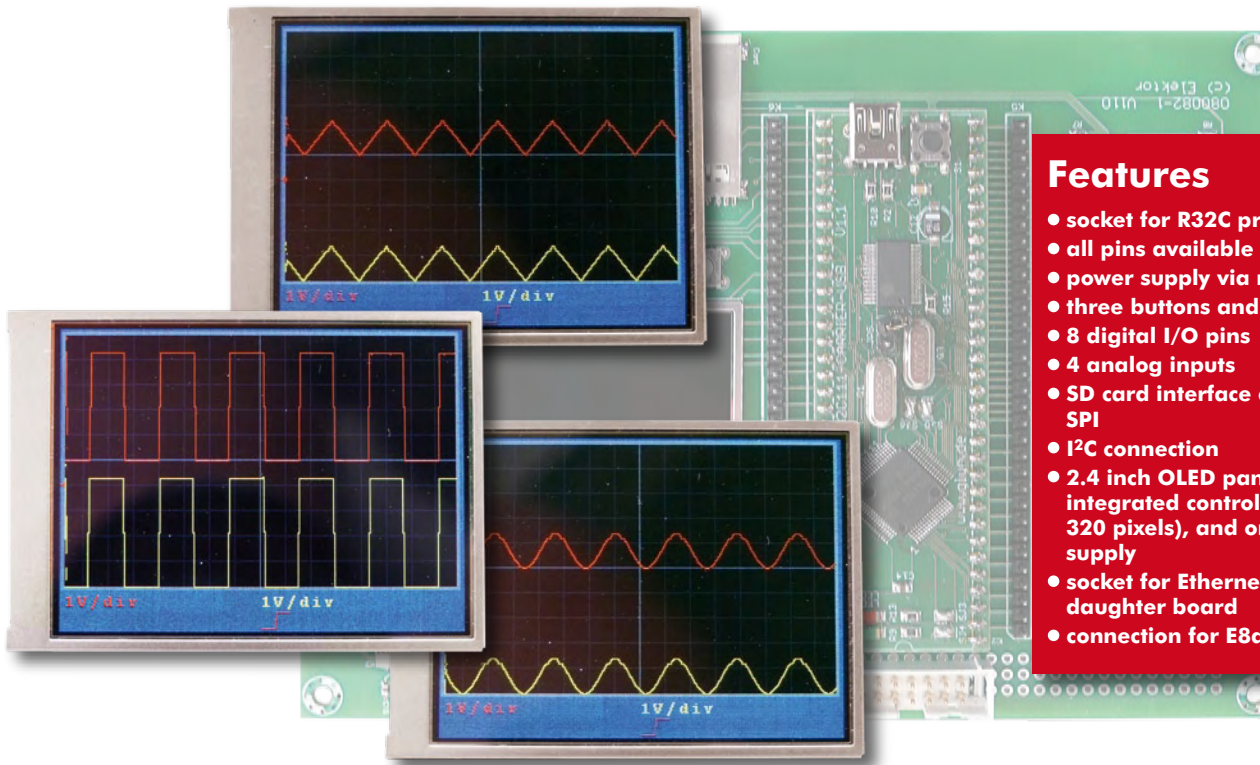
Features

- socket for R32C processor board
- all pins available on headers
- power supply via mains adaptor
- three buttons and four LEDs
- 8 digital I/O pins
- 4 analog inputs
- SD card interface controlled over SPI
- I²C connection
- 2.4 inch OLED panel with integrated controller (240 by 320 pixels), and on-board power supply
- socket for Ethernet and TCP/IP daughter board
- connection for E8a debugger
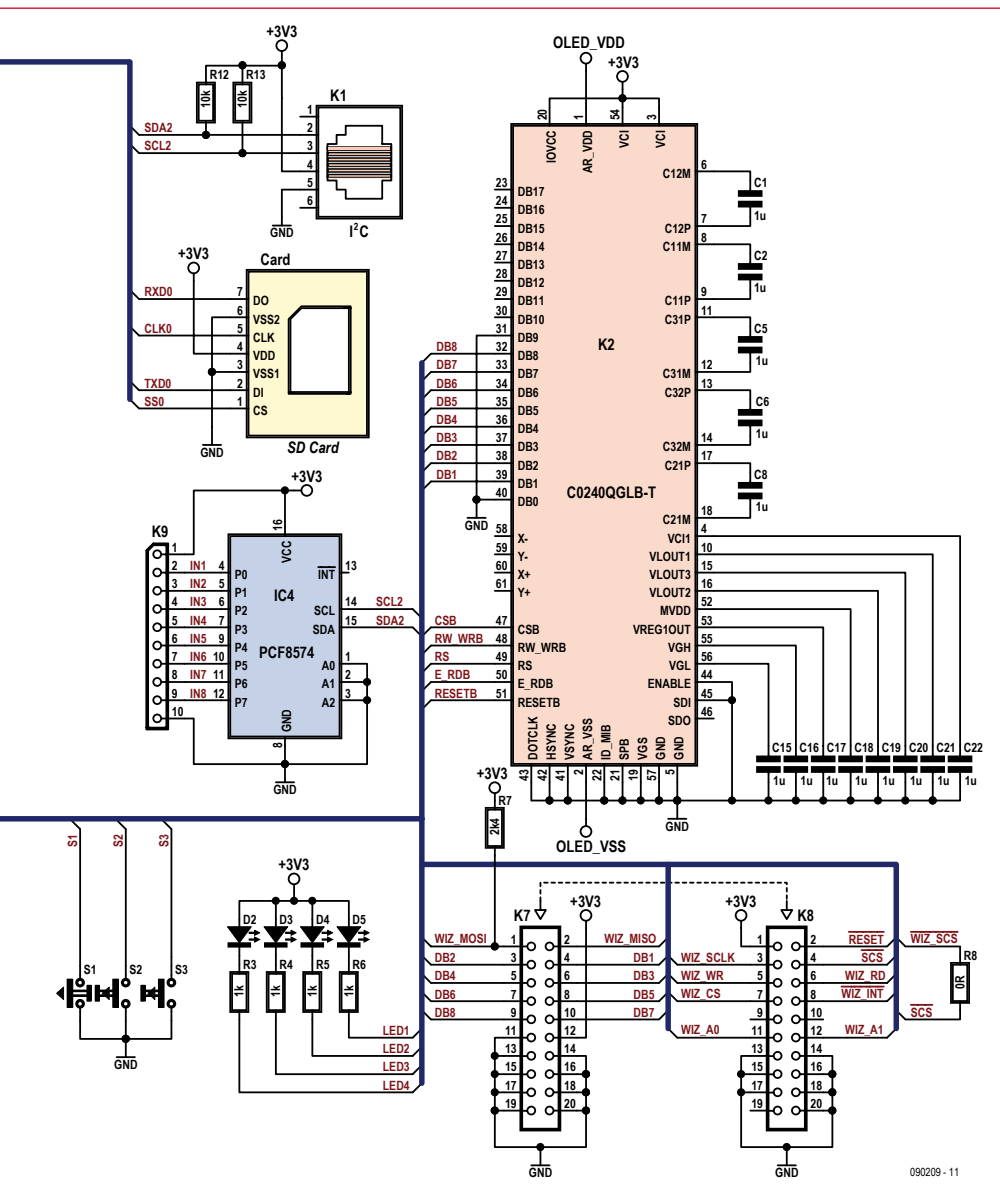
Power is made available on the various headers on the board that the R32C uses to communicate with the outside world. These can provide power to I²C slave devices or any more sophisticated interface circuits that you might design. Circuits can be constructed on the prototyping area provided on the board.

We will look at the interfaces in more detail later, but for now we just note that **under no circumstances should the R32C be powered over the USB socket on the processor board when it is mounted on the application board!** This would lead to 5 V rather than 3.3 V appearing on all the power supply pins, likely causing irreversible damage to components.

In order to isolate the board from the 5 V supply present on the USB connector when it is used to program the microcontroller, it is **essential** to **remove jumper SJ7**, a solder bridge located on the underside of the processor board (see **Figure 2**). Before using the application board, therefore, use a soldering iron and a little desoldering braid to break the connection to prevent the risk of the high voltage permanently damaging the board.

Figure 1. Circuit diagram of the application board. The R32C processor is mounted on a daughter board, although for clarity we have omitted these connectors and shown the processor as directly connected to the peripherals.
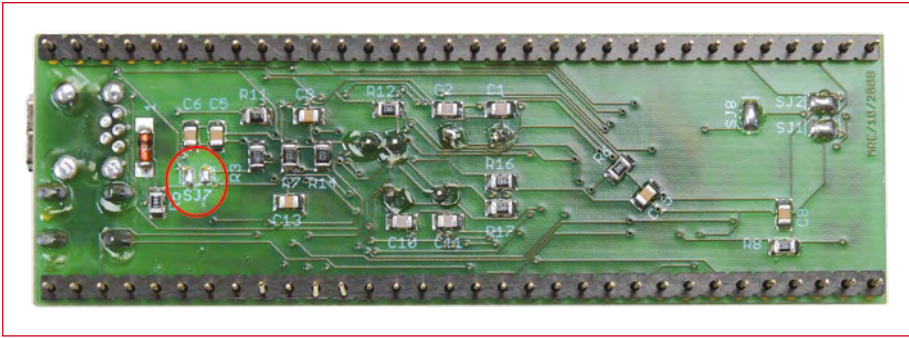
Figure 2. It is essential to open this solder bridge
before fitting the processor board to the application board.

Interfaces

No development board is complete without pushbuttons and LEDs, of course; the R32C application board is no exception, sporting three pushbuttons and four LEDs. Each of these devices is connected to an I/O port pin on the microcontroller, making them ideal for use in simple programming experiments. In more advanced applications they can be used as part of the user interface or for status indication. Eight digital I/O pins are made available on header K9. It seems a pity to use so many of the microcontroller's own I/O pins for a simple port like this when they are better employed driving the OLED display panel and for other purposes we will describe later; and so we have instead employed an I^{2}C port

expander device. The port expander is connected to USART3 of the R32C, one of whose modes of operation is as an I^{2}C interface. Being implemented in hardware the interface is fast and its configuration is flexible. To simplify programming, C functions are available for download from the web pages accompanying this article [5] that allow simple writing and reading of the I/O pins.

These days there are many devices available that are controlled over an I^{2}C port. We have therefore made the I^{2}C bus signals available on RJ11 connector K1, which offers enormous scope for further expansion.

Next door is the SD card slot. Writing to and reading from an SD memory card is done over the microcontroller's hardware SPI interface. Further details can

be found on the Internet [6].

Header K10 allows up to four analog signals (from 0 V to 3.3 V) to be read by the microcontroller.

OLED and more

By far the most important feature of the application board is the 2.4 inch color graphic OLED display. In the May 2009 issue of Elektor we showed how to draw simple geometric figures on the display and how to show images loaded from a file. The small, freely-downloadable C graphics library [4] provides these functions and also allows text to be displayed on the panel. The May 2009 article also described how the OLED is driven. What we did not show in that article was how to power the OLED panel. The controller in the device requires a 3.3 V supply, and the display itself needs a symmetrical 4 V supply. We cannot generate a negative voltage using a linear regulator, and so we have selected a device from ON Semiconductor, specifically designed for powering OLED panels. The details of how to use the device are covered in its datasheet [7].

Connector K4 allows the connection of the E8a debugger, which we briefly described in an article in April 2009 [8]. This relatively inexpensive hardware tool makes debugging a more pleasant experience than using the KD100 software-only debugging solution that forms part of the starter kit and which was also described in the April 2009 article.

One final neat feature: headers K7 and K8 will accept the WIZ812MJ Ethernet module. This elegant little daughter board, made by WIZnet [9], provides a network interface and even implements the higher network protocol layers. This 'network card for microcontrollers' includes the special-purpose W5100 IC, an RJ45 socket, the necessary transformer, and status LEDs. It provides the R32C with the ability to send e-mails or act as a web server. FTP, UDP and many other protocols are supported. We do not have the space to give a detailed description of the module and its software here, but we hope to devote a whole article to the device towards the end of this year.
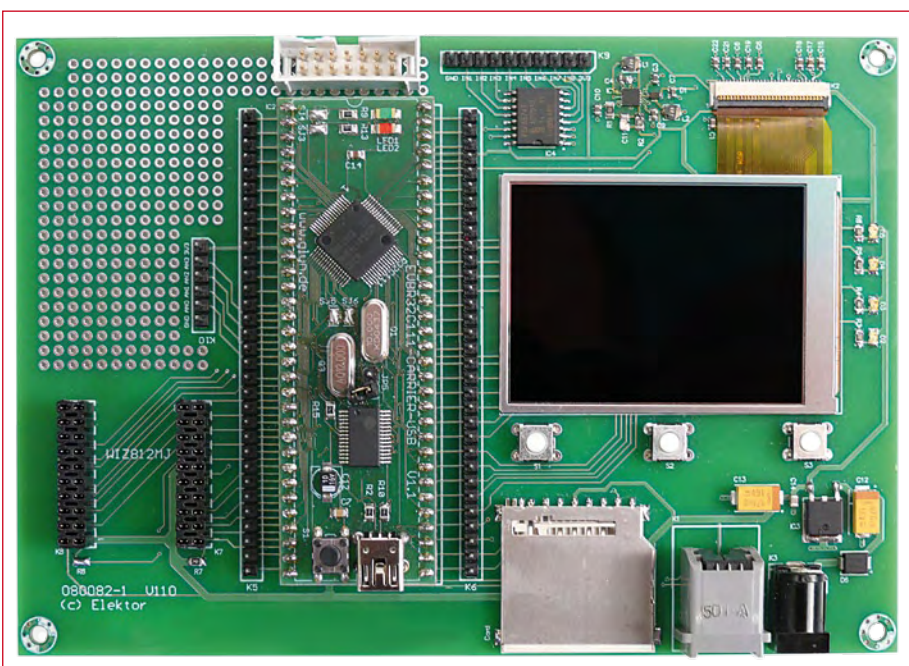
Oscilloscope

Now we come to the main course: a two-channel oscilloscope example

Figure 3. The board sports pushbuttons, LEDs, an I2C interface, an OLED panel,
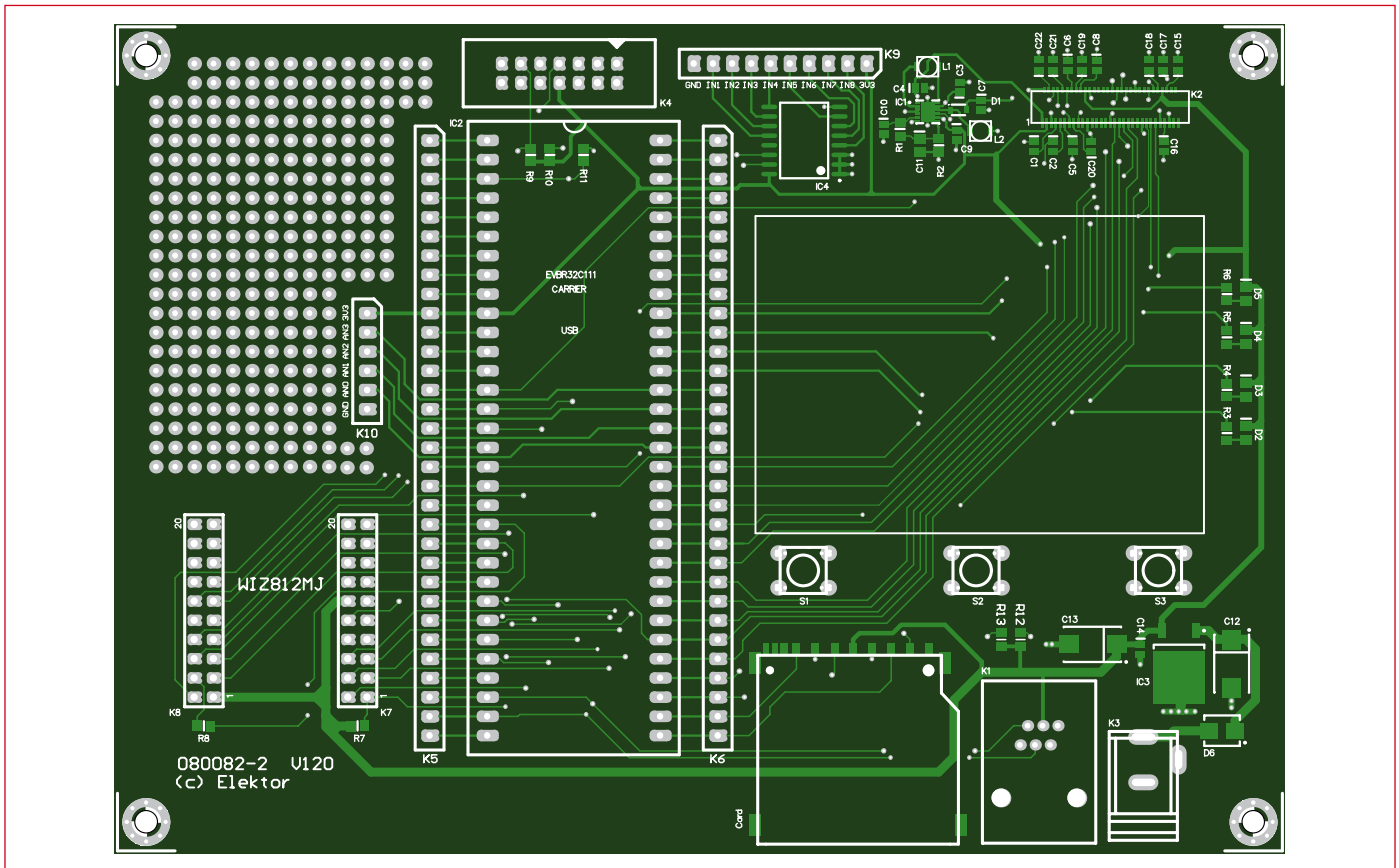an SD card interface and a socket for an Ethernet module.

Figure 4. There is plenty of space on the board for further expansion.

application to show off the power of the R32C. The software is available for free download from the web pages accompanying this article [5].

The software only emulates a few of the basic functions of an oscilloscope, and can of course be extended and modified as the fancy takes you. The analog-to-digital (A/D) converter in the R32C offers a conversion time of 2.06 μs and thus a maximum sam-

ple rate of 480 kHz. According to the Nyquist-Shannon sampling theorem, the available bandwidth is equal to half the sample rate, in this case 240 kHz. Although this may not seem very impressive, it is worth remembering that we are using the microcontroller's built-in converter rather than a special-purpose A/D converter IC. If higher sample rates are required, an external device can be connected, using the SPI or a similar port. When

using an external device it is also easier to suppress noise by providing the converter with a power supply free of interference from the circuitry inside the microcontroller.

The program

When the program starts it carries out all the necessary initializations, in particular of the R32C's clock generator and of the display. The function library

COMPONENT LIST

Resistors
(all SMD0805)
R8 = 0Ω
R3,R4,R5,R6 = 470Ω
R7 = 2.4kΩ
R1,R2,R9,R10,R11 = 4.7kΩ

Capacitors
C11 = 10pF
C1,C2,C5,C6,C8,C15–C22 = 1µF 25V (SMD0603)
C3,C4,C7,C9,C10 = 4.7µF 10V (SMD0603)
C14 = 100nF (SMD0603)
C12,C13 = 47µF 16V tantalum (SMD7343)

Inductors
L1,L2 = 4.7µH (VLF3010AT-4R7MR70)

Semiconductors
D1 = NSR0320MW2
D2,D3,D4,D5 = LED, red (SMD0805)
D6 = MBRS130
IC1 = NCP5810D
IC2 = socket: 2 pcs 32-pin SIL socket strip (Fischer SL1 series), R32C processor board comprised in R32C Starter Kit # 080928-91, see below
IC3 = LF33CV
IC4 = PCF8574T

Miscellaneous
K1 = RJ11 socket
K2 = OLED connector (Hirose

FH23-61S-0.3SHW(05))
K3 = DC adaptor socket, PCB mount
K4 = 14-way boxheader
K7,K8 = 20-way DIL socket strip (Fischer BL2 series)
K5,K6 = 32-way SIL pinheader (Fischer SL1 series)
K9 = 10-way SIL pinheader (Fischer SL1 series)
K10 = 6-way SIL pinheader (Fischer SL1 series)
K11 = SD card socket (Multicomp SDCMF-10915W0T0)
S1,S2,S3 = pushbutton
OLED, CMEL type C0240QGLA-T

R32C starter kit, Elektor Shop # 080928-91, see [5],[8]
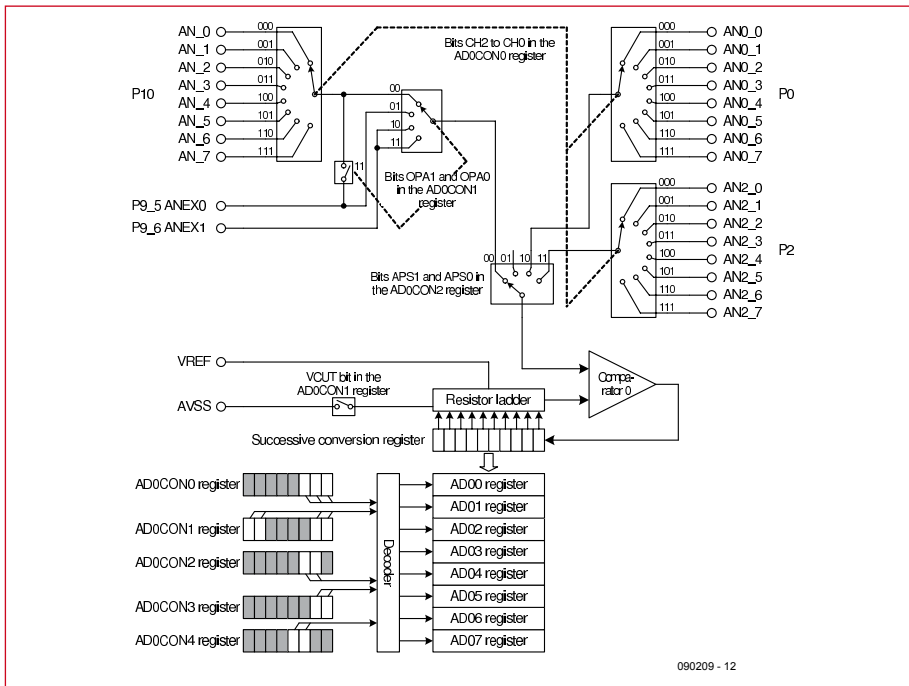PCB # 080082-2, see [5]

Figure 5. After an A/D conversion the results are found in registers AD00 and AD01.

So that the data can be processed quickly, the values in these registers are transferred to separate RAM buffers after each conversion cycle using DMA (direct memory access). The DMA process can copy data from one memory area to another without intervention from the processor. Indeed, the only effect on the processor is that the internal bus is occupied briefly while the transfer takes place. The transfer is triggered by the end-of-conversion signal from the A/D converter. The next conversion begins immediately. The DMA process continues to run, completely automatically, until all the samples required for the trace have been obtained and stored. The DMA engine requires the following parameters to be set:

● source address (where the data come from, in this case the A/D converter);

● destination address (where the data go to, in this case a RAM buffer);

● trigger source (what initiates a transfer, in this case the A/D converter);

● transfer count (how many data items are to be transferred).

Listing 1 shows how the relevant registers are initialized.

Image construction

Now that we have all our data sitting in RAM, we are in a position to construct the display image. First we must erase any previous trace on the screen. Iterating over the entire image would take rather a long time, so instead we make the optimization of only clearing to background color those pixels that were set in the previous image. We then set the appropriate pixels for the new trace. This process is very quick as only 320 pixels (the width of the display) need to be addressed in each of the two passes; a full re-write of the display screen would address $320 \times 240 = 76800$ pixels.

In fact, because of the display graticule and other background symbols, it is not enough to simply reset the pixels of the old trace to the background color. Instead, the required background pixel colour is loaded from a table located in the R32C's flash memory.

In order to maintain a steady trace on the display, it is necessary to imple-

hwsetup.c includes an easy-to-use clock set-up function: the PLL parameters and division ratios are configured in the accompanying header file. All the OLED display driver functions, as described in the May 2009 issue [4], are also included in this library.

For the oscilloscope demonstration software pins AN00 and AN01 (A/D port 0) are used. These inputs are made available on connector K10. The

A/D converter supports a range of different modes, including a single-shot mode and repeat modes. Since in a two-channel oscilloscope two sets of readings need to be taken, we configure the converter so that the two inputs are sampled alternately (the relevant registers are shown in **Figure 5**). The results end up in separate registers: the R32C has eight buffer registers per A/D port. In this case our readings will be found in registers AD00 and AD01.

Listing 1: Initialisation

```
void measurement_init(void)
{
    // Init A/D-Converter
    ad0con0 = 0xB0; // conversion stopped | timer trigger | single sweep
    ad0con2 = 0x25; // Trigger Timer B2 | AN0_0 Group | with sample & hold
    ad0con1 = 0x38; // VRef connected | 10bit resolution | sweep AN0 & AN1
    ad0con3 = 0x01; // DMAC ENABLED!!!!
    ad0con4 = 0x00; //

    // Init Timer B2
    tb0mr = 0x40;        // Timer Mode | f8
    tb2 = 0x0030;        // Timer Value 0x30

    // Init DMAC
    dmd0   = 0x00;       // Disable all functions
    dm0sl  = 0x18;       // AD0-interrupt
    dm0sl2 = 0x00;
    dsa0   = &ad00;      // source is AD00
    dm0ic  = 0x02;       // interrupt priority 02 (DMA)
    ad0ic  = 0x01;       // interrupt priority 01 (ADC)
    tb2ic  = 0x01;       // interrupt priority 01 (Timer B1)
}
```

Listing 2: Rendering the oscilloscope trace

```c
/*********************************************************************
 * DRAWING                        *
 *********************************************************************
 * Parameter:                                                       *
 * - channel: describes the channel which should be drawn           *
 * - data:    reference to the data buffer                          *
 * - offset:  offset for the measured values                        *
 * - yoffset: position of base line:                                *
 *            channel 0: 0                                          *
 *            channel 1: -100                                       *
 *********************************************************************/
void oszi_draw_data(unsigned char channel, unsigned int *data, unsigned int offset, signed char yoffset)
{
    unsigned int pos, ypos, ypos_ref, ypos_prev, ypos_draw, data_pos;
    // set offset for data buffer
    // the values of both channels are alternating
    data_pos = offset * 2 + channel;
    // calculate the first value
    ypos_prev = 105-yoffset-((unsigned int)(data[data_pos])%1024)*100/1023;
    // calculate all 309 values and draw the connection lines
    for(pos=1; pos<=310; pos++)
    {
        // calculate the new position for data buffer
        data_pos = (pos+offset) * 2 + channel;
        // calculate new Y-position
        ypos = 105-yoffset-((unsigned int)(data[data_pos])%1024)*100/1023;
        // calculate the average of old and new Y-position
        ypos_ref = (ypos_prev + ypos) / 2;
        // draw falling or horizontal connection lines
        if(ypos_prev >= ypos)
        {
// draw line starting from first X-value of the old Y-position to the reference (half height)
            for(ypos_draw=ypos_prev;ypos_draw>=ypos_ref; ypos_draw--)
                setPixel(ypos_draw, 315-(pos-1), 0x3F, (channel==0)?0x00:0x3F, 0x00);
// draw line starting from the second X-value of the reference to the new Y-position.
            for(ypos_draw=ypos_ref;ypos_draw>=ypos; ypos_draw--)
                setPixel(ypos_draw, 315-pos, 0x3F, (channel==0)?0x00:0x3F, 0x00);
        }
        // draw rising connection lines
        else
        {
// draw line starting from first X-value of the old Y-position to the reference (half height)
            for(ypos_draw=ypos_prev;ypos_draw<=ypos_ref; ypos_draw++)
                setPixel(ypos_draw, 315-(pos-1), 0x3F, (channel==0)?0x00:0x3F, 0x00);
 // draw line starting from the second X-value of the reference to the new Y-position.
            for(ypos_draw=ypos_ref;ypos_draw<=ypos; ypos_draw++)
                setPixel(ypos_draw, 315-pos, 0x3F, (channel==0)?0x00:0x3F, 0x00);
        }
        // store Y-position as the old Y-position
        ypos_prev = ypos;
    }
}
```

ment some kind of trigger function. The trigger condition in the demonstration software is based on the rate of change of the input signal. The display is only refreshed when the trigger condition is satisfied. The relevant C function has a loop that checks the incoming readings until it finds a point where the trigger condition is satisfied. The trace is plotted from this point on (see **Listing 2**). For this approach to work, it is of course necessary to have enough input data available. Rather than fetching just 320 samples, which would be

enough to fill the width of the screen, we therefore fetch twice that number. This guarantees that if a suitable trigger point is found there will be enough data to render a complete trace.

As you can see in our main photograph, the final result looks rather impressive.

(090209-I)

Internet Links

[1] www.elektor.com/r8c

[2] www.elektor.com/050179-3

[3] www.elektor.com/050179-5

[4] www.elektor-usa.com/081029

[5] www.elektor-usa.com/090209

[6] www.captain.at/electronic-atmega-mmc.php

[7] www.onsemi.com/pub_link/Collateral/ NCP5810-D.PDF

[8] www.elektor-usa.com/080928

[9] www.wiznet.co.kr/en

A dirt cheap cell phone

By Jens Nickel

Just when you thought mobile phones couldn't get much smaller Elektor Labs spotted one the size of a credit card and priced just under 17 euros or about 25 dollars. No contract, no SIM card lock or restrictions! The Simvalley RX-80 Pico comes with a 1.2" LCD, SMS messaging, telephone directory and even a back-lit keypad. Intrigued, we took a closer look!

It's not just the price but also the size and weight of the 'Simvalley RX-80 Pico' from Pearl Diffusion that is astonishing. At 50 x 80 x 10 mm and tipping the scales at just 44 g this phone can be easily

On the PCB we can see the SIM card holder, USB connector for battery charging, the battery connections, loudspeaker and microphone. At this point Antoine looked genuinely surprised 'Wow, that's so tiny'. Despite its diminutive size the microphone provides surprisingly good speech quality.

Keypad backlighting is essential at night and in this phone a few LEDs provide the necessary glow. We took the opportunity to power up using a three volt supply (see picture above right).

Further circuitry is tucked away behind a plate which provides RF shielding. The hot-air gun wasn't much use here so we set to work with a Dremel tool.

The largest IC is a Flash memory from the Taiwanese company Eon
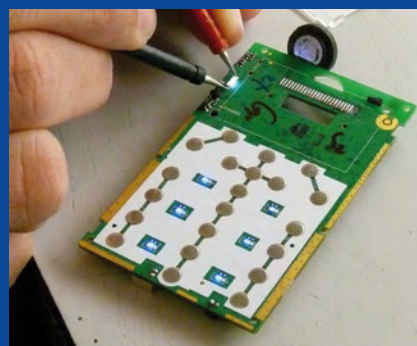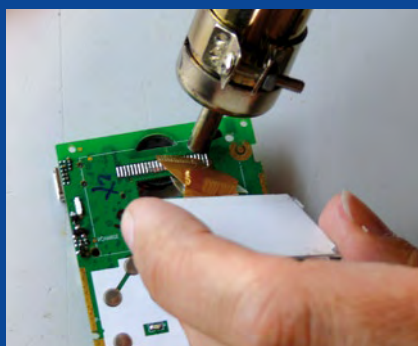
stashed in a wallet or purse. Heiko Loy, the press spokesman for Pearl, told us that the 17 Euro mobile phone is mass produced in China.

Antoine Authier the head of our laboratory here at Elektor took a closer look at the phone. Armed with a plastic tool designed to open iPods he quickly separated the two halves of the case. The PCB is retained by clips in the lower half of the case (screws would be way too expensive!).

A short length of adhesive tape holds the display to the PCB, this is quickly removed as Antoine wields a hot-air gun. The 1.5 mm thick display is connected via 14 pins to the PCB. We searched for any identification marks on this component but sadly were out of luck this time.

Silicon Solution [2]. It is most likely used to store SMS texts, phone directory, caller information and multiple versions of the menu text in every language option.

The second chip carries the inscription SKY77518-21 identifying it as a dual-band cellphone front end module from Skyworks Solutions [3]. The third chip (inscribed 7880 1.3G FC GO914) is probably some variant of the old 7880 cellphone baseband processor marketed by Infineon. As always we value your feedback, if you have more information on this device don't hesitate to get in touch!
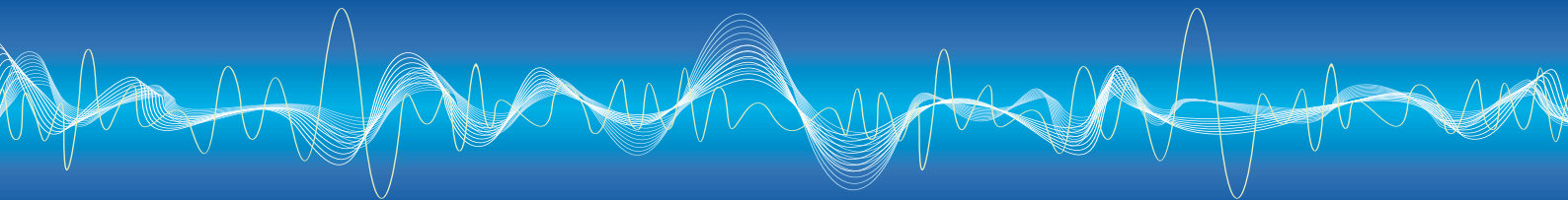
[1] www.pearl.fr
[2] www.eonssi.com
[3] www.skyworksinc.com

(090502-I)

E-LABS INSIDE
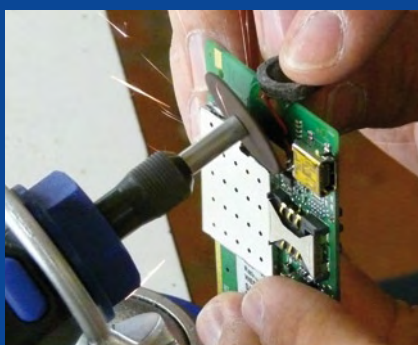
Double-sided soldering in reflow oven!

All a matter of accuracy – but it's child's play !

By Antoine Authier (Elektor Labs)

Many readers are eager to find out just what our reflow oven is capable of have often asked me if they could solder components on both sides of their boards. My initial answer was certainly yes — at least in theory; so I started looking around for a project to demonstrate in practice that double-sided soldering is possible.

I settled on the USB-I²C interface published in February 2009, as a PCB hadn't yet been designed for this project, and the prototype we had certainly wouldn't win any prizes for elegance. To my way of thinking, a dongle ought to be compact and have a neat appearance; it contains a handful of passive components and a few ICs.

While I was about it, I thought I'd add an electronic serial number function using a DS2401, along with ESD protection (yet another Maxim chip). These devices are all available in SMD packages ideal for a double-sided board, and so meeting my criteria. Of the three through-hole components, I kept only the RJ11 connector, particularly to give me an excuse for mentioning the order for soldering through-hole components with respect to the surface-mount devices.

Get together the component layout, component list, and the components themselves.

You need to solder the 'underneath' SMD components first. Start
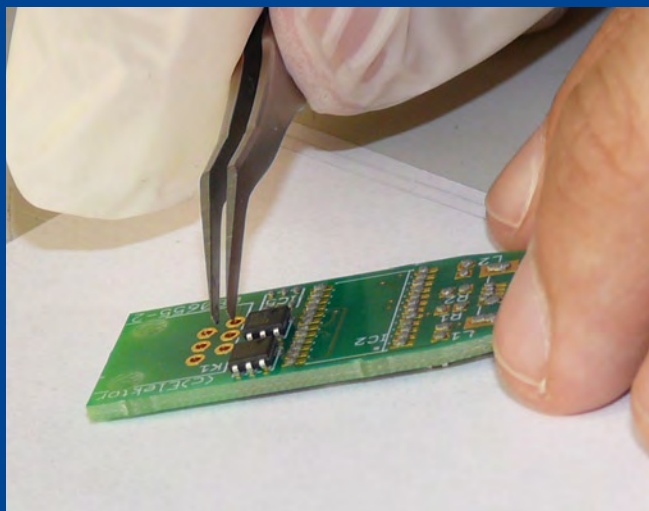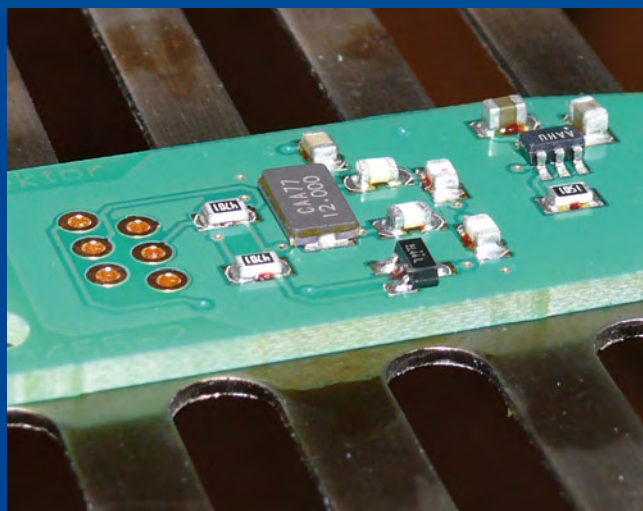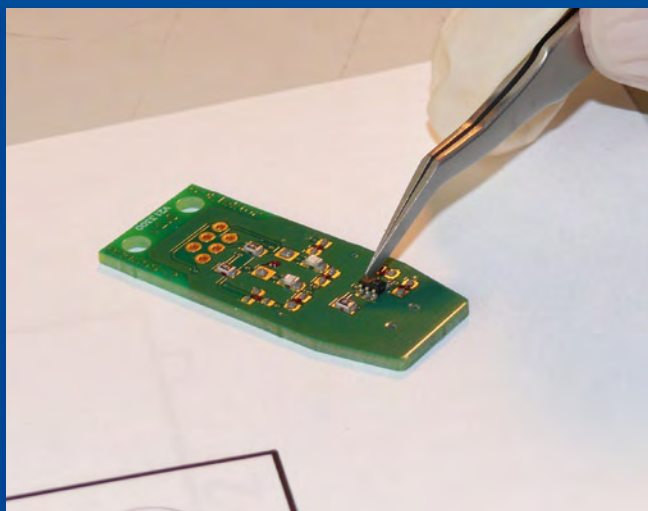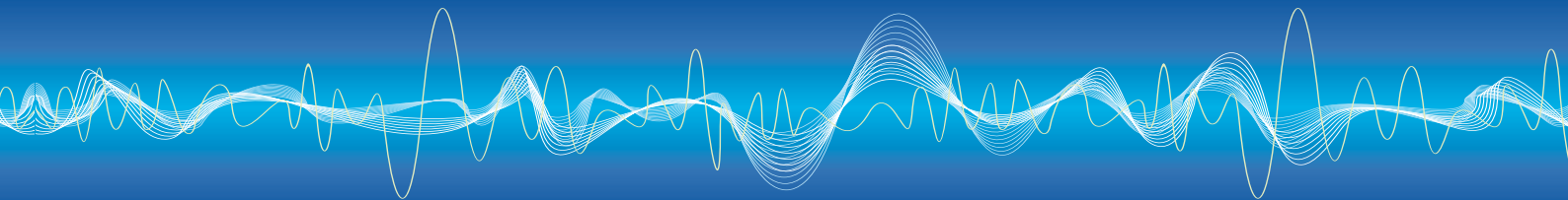
by applying soldering paste to the solder pads, then a small drop of SMD adhesive between the pads, preferably in the center of the location for the component being soldered. When heated, the epoxy-based SMD adhesive polymerizes and holds the component — it's like magic! Now place the components in their positions and solder in the oven using your favorite/usual reflow program. Tip: before starting the soldering process, check one last time that the components haven't shifted while being moved to the oven.

The soldering method is simple

Leave to cool down, and get ready for the next step: apply the soldering paste, this time to the 'top' pads, then place the SMD components. Here, there's no need for adhesive. Solder in the oven using your favorite program. The underneath components won't fall off, as they are glued.

Once the board has cooled down, all that's left to do is solder the through-hole components using a conventional soldering iron.

The notions of 'underneath' and 'top' are arbitrary — you can of course start by soldering the top components and finish with the underneath ones; the important thing is to glue the components

for the first soldering operation, so they are held on the board while the second side is being soldered.

To decide which side to glue, as I see it there are three main criteria. The first one is strategic: glue preferably passive components, to avoid as far as possible soldering the more valuable ICs twice.

The second criterion is a practical one: you should glue as few large components as possible, since in the event of a mistake, the smaller components are the easiest to unglue.

The last criterion is an economic one: it's better to use the adhesive on the side with fewest components.

It's also important to take these considerations into account when designing the board, so as to distribute the components sensibly between the two sides.

So we can deduce that on the glued side, there should be as few components as possible, and preferably all passive ones.

One last small detail: to unglue a component, all you have to do is hold it firmly using tweezers, heat it up (preferably with a hot-air gun) and push it gently away from its position in order to break the blob of adhesive.

Now let's get cooking!
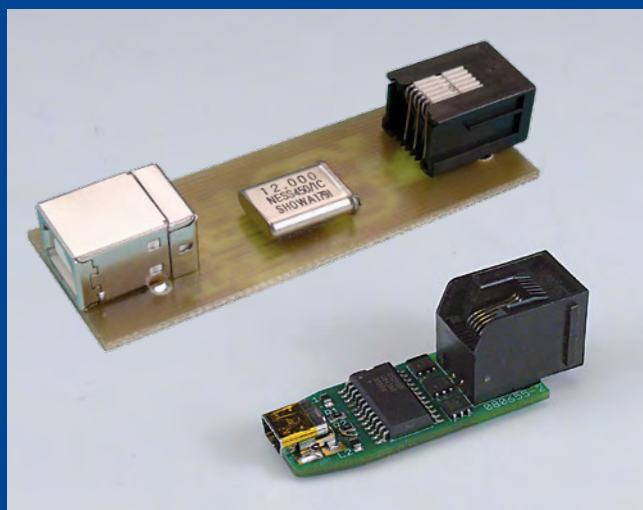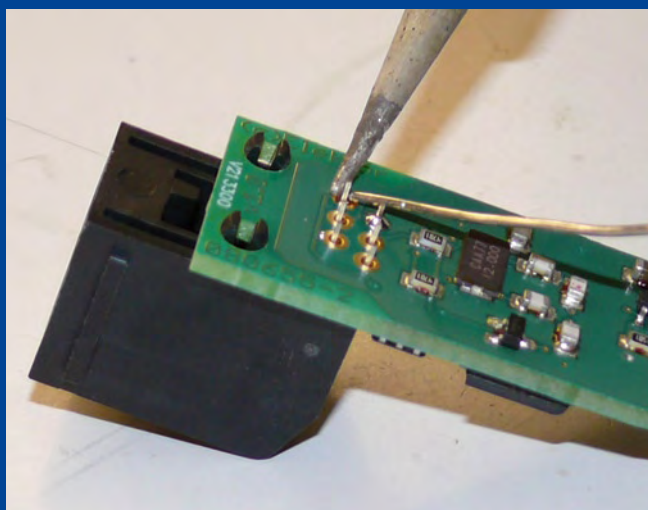
"I'll take it"

It's not every day that our lab engineers Luc and Chris have an opportunity to work with an oscilloscope that costs around €12,000. Thanks to the generosity of the scope manufacturer, Yokagawa, we had a DLM 2054 available on loan in the lab for a few months. The only question was: who would get to use it?

Luc: A scope is indispensable in the lab; we use them every day for all sorts of measurements. The DLM 2054 is remarkably easy to use. The menu is intuitive, which is just right for me.

Chris: The amount of memory in the scope is important. This instrument comes standard with 1.8 GB of memory, so it can hold a lot of signal history.

Luc: And it has a sampling rate of 2.5 gigasamples per second, with a bandwidth of 500 MHz and four channels.

Chris: Of course, you have to consider what you want to use it for. With I2C circuits, the special trigger modes are a real help.

Chris: And it has a built-in printer, which is handy when you need to check something quickly.

Luc: If you can put down the 15,000 dollars, I can certainly use the scope.

Chris: Maybe we should just say that we need more time to try it out. But I'm the one who wants it!

Why do you want it?
Look at the specs: that's why.

Frequency bandwidth:	500 MHz
Analog input channels:	4
A/D resolution:	8 bit (25 LSB/div)
Real time sampling mode :	interleave off : 1.25 GS/s
	interleave on: 2.5 GS/s
Maximum record length:	Repeat / Single / Single Interleave
	12.5 M / 62.5 M / 125 Mpoints
Maximum sampling rate:	1.25 GS/s
Trigger modes:	Auto, Auto level, Normal, single, N-single
Trigger types (excerpt):	Edge, State, Pulse width, State width, I2C, SPI, UART, CAN
Display:	8.4 inch TFT LCD
Built-in printer	
USB peripheral connection terminal	
USB PC connection terminal	
Ethernet	
External dimensions:	226 (W) x 293 (H) x 193 (D) mm
Weight: approx.	4.2 kg

(090572)

Website of the DLM2000 series (starting price around $ 4,000) from Yokogawa: www.dlm2000.net

Chris Vossen studied Electronics at Heerlen Polytechnic and afterwards worked in a small company for a number of years where he designed data logging equipment. He has worked as a designer here at the Elektor lab since 2005. He specialises in larger microcontroller projects for example the recent R32C and the ATM18 project series.

Luc Lemmens studied Electrical Engineering at Eindhoven Technical University and has been a fixture here at Elektor. for the last 19 years where he works as a development engineer and technical editor. Luc's expertise is in measurement techniques and microcontroller circuits. He is also responsible for selecting smaller projects and for example the choice of designs for our special Summer Circuits edition.

Battery Monitor
Everything under control

By Steffen Graf (Germany)

This circuit was originally designed for monitoring the charge status of the batteries in a solar-powered water feature. However, it can be used in any application where batteries are charged and discharged. The circuit uses an LPC2103 microcontroller connected to a 22-bit analog-to-digital (A/D) converter to measure charge and discharge currents, battery voltage, charge status (or available capacity) and the instantaneous power being supplied to or drawn from the battery.

There are various ways of providing rechargeable battery packs with an indication of their charge status. The most widespread uses a simple voltage measurement; however, NiMH and NiCd cells have a rather flat voltage discharge curve and so the method is not particularly accurate in this case. A better approach is also to measure the current into or out of the battery. This allows calculation of the flows of power and charge, and hence (if the initial state is known) also the available battery capacity. The module described

here offers all these features and can be used to monitor batteries or more generally to meter voltages, currents and power in any DC supply system. The author uses the module to monitor a small solar-powered installation.

The unit can work over a wide voltage range, from 6 V to 42 V, and so can be used in typical lead-acid battery applications at 6 V, 12 V, 24 V or 36 V. The current range is also very wide; despite this, resolution is high, with measurements accurate to around 1 mA. A 50 mΩ SMD shunt resistor, rated at 2 W, makes for a compact construction and allows currents of up to 6 A to be measured.

Summer Circuits (July/August) 2008 issue of Elektor. This remarkable IC (see text box) is a 22-bit delta-sigma converter with differential inputs and an SPI port, available in a reasonably easy-to-solder SMD package.

An important aspect of this project is the generation of the various supply voltages. The circuit requires 5 V for the LCD, 3.3 V for the A/D converter and for the microcontroller, 1.8 V for the microcontroller core and 1.2 V as a reference for the A/D converter. All these voltages must be generated

These devices require minimal external circuitry. The input side of IC6 (the MCP3550) is simply connected across the shunt resistor to measure the battery current: the shunt is connected directly between the input and output terminals on the printed circuit board. The board is inserted in the line to the battery using these terminals, and hence can monitor current flows into and out of the battery. Terminal K1 also provides the power supply for the circuit via 5 V switching regulator IC1, and, via the voltage divider formed by R3, P1 and R8, provides the volt-

Features

- Display of current (in mA or A), voltage (mV), capacity (Ah) and power (mW or W)
- Suitable for monitoring all types of rechargeable battery
- Suitable for battery voltages from 6 V to 42 V
- Maximum measured current: ±6 A
- High precision: voltage resolution 76 mV, current resolution 12 μA (internal), 1 mA (displayed)
- Low power consumption due to use of high-efficiency step-down converters: <200 mW (backlight off), <300 mW (backlight on)
- Module takes into account its own power consumption in calculating current and capacity
- Display update rate 1 Hz
- Over- and under-voltage fault LED
- Backlit 2-by-16-character LCD
- Compact construction
- Serial interface for outputting measured values
- Firmware available for free download from Elektor website

Overview

The microcontroller used is the LPC2103, a 32-bit ARM7-based RISC device from NXP. It differs in only minor details from the LPC2106 device used in the ARMee development board described in the April 2005 issue of Elektor and in the article 'Automatic Running-in Bench' published in April 2009. Both are based on the 32-bit ARM7TDMI core, which we described in an article in the March 2005 issue. The LPC2103's internal A/D converter has a resolution of only 10 bits: this is enough for our voltage measurement, but not enough for the current measurements we want to make. Furthermore, since we want to be able to measure current flow in either direction (charge and discharge) we need an A/D converter with a differential input. We therefore use the internal A/D converter only for voltage measurement, and employ an external converter for current measurement. We selected the Microchip MCP3550-50, which was briefly described on page 87 of the

efficiently from a highly variable input supply (between 6 V and 42 V). That demands a switching converter to provide a 5 V regulated supply directly from the input. A second switching converter drops the 5 V supply to 3.3 V, and the remaining voltages (1.8 V and 1.2 V) are generated from the 3.3 V supply using linear regulators. Even at high input voltages the power dissipation of the circuit is low, removing the need for a bulky heatsink.

Circuit and construction

The complete circuit is shown in **Figure 1**. Out of the total of six ICs, four are voltage regulators. The switching regulators (IC1 and IC2) can be identified by the inductors connected to them, while the linear regulators (IC3 and IC4) just have decoupling and smoothing capacitors. The meter circuit itself comprises just the LPC2103 (IC5) and the A/D converter (IC6).

age measurement input. The divider reduces the voltage to a level suitable for input to the A/D converter built in to the LPC2013, which can only accept voltages from 0 V to 3.3 V. The potential divider can be set up in two different ways:

1. use 0.1 % tolerance components for R3 and R8, in which case P1 can be dispensed with;

2. use normal (1%) resistors and then calibrate the measured voltage by adjusting P1. R8 is then omitted.

If the input voltage is over 40 V then adjustment of P1 will be necessary in any case for best accuracy; and if the input voltage is over 42 V adjustments to the software are also needed.

Jumper JP2 allows the unit's 5 V supply to be taken from the programming interface. This allows the microcon-
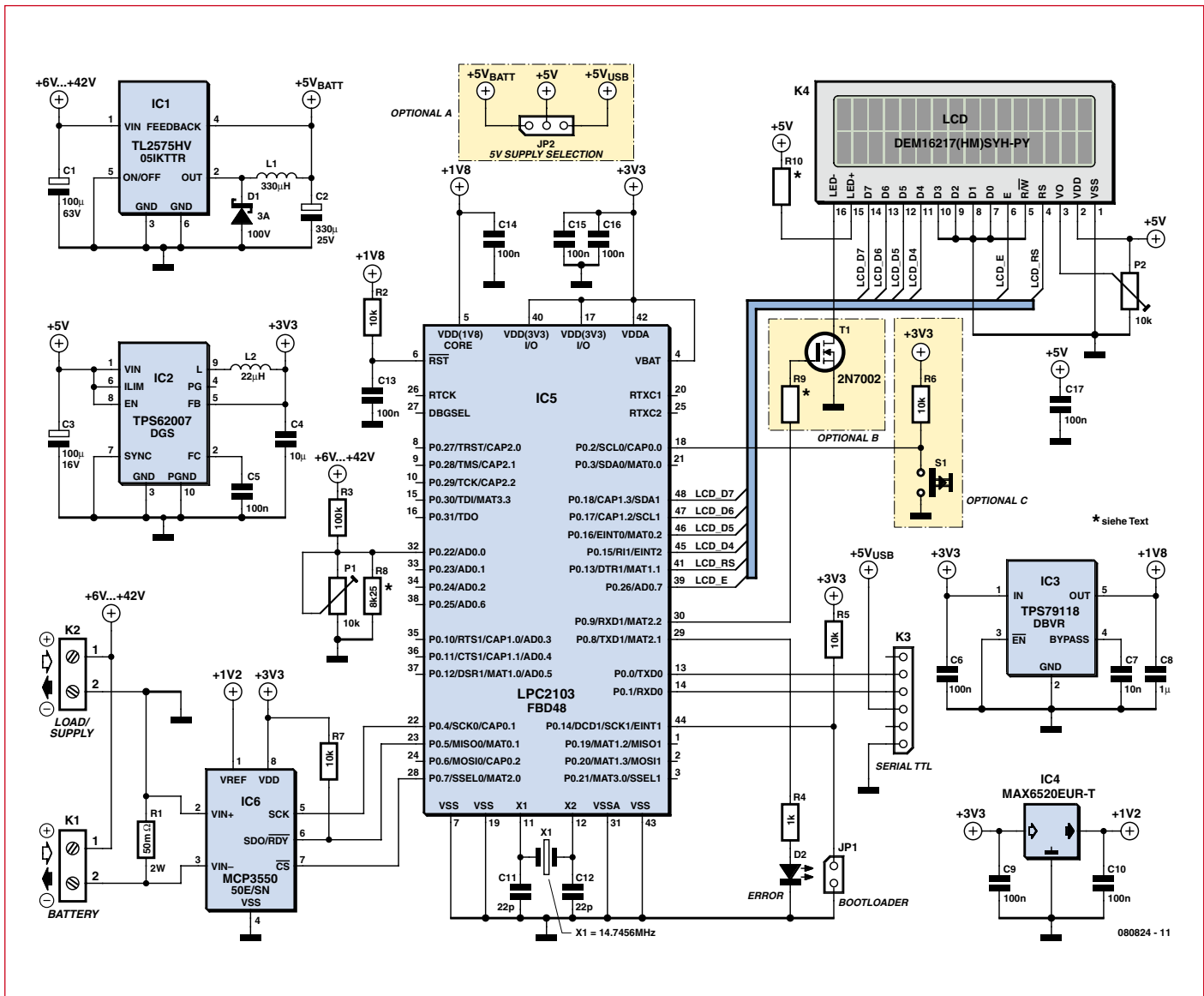
Figure 1. Circuit diagram of the battery monitor. Out of the total of six ICs, four are voltage regulators.
The meter circuit proper consists of a 32-bit ARM7 microcontroller (IC5) and a 22-bit A/D converter (IC6).

troller to be programmed without a battery being connected. If a battery is connected while power is being supplied over the programming interface, IC1 will only draw a few milliamps from it: the rest of the circuit will still take its power from the programming interface. Don't forget to change the jumper back to battery power after programming the microcontroller. Alternatively, if the ability to power the unit over the programming interface is not needed, JP1 can be replaced by a wire link joining +5 V$_{BATT}$ and +5 V.
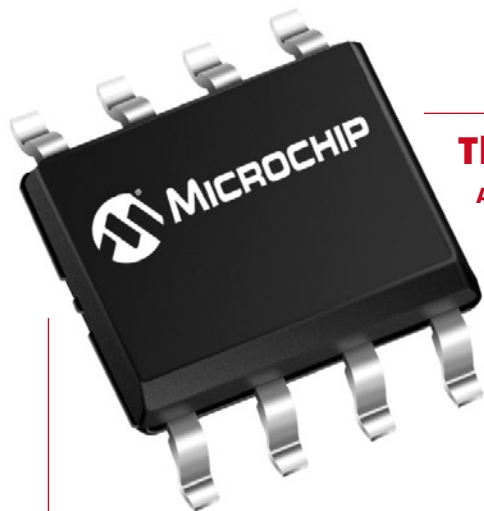
The LCD panel used has a backlight. In the interests of reducing power consumption the light can be turned on and off. Button S1, connected to port pin P0.2 on the microcontroller, controls

a routine in software that causes a PWM signal to appear on pin P0.9. This in turn drives MOSFET T1 to provide a dimmable backlight. If the ability to turn the backlight off is not needed, R10 can be replaced by a 220 Ω resistor (instead of 0 Ω) and the MOSFET replaced by a wire link joining source and drain. R9, as well as R6 and S1, can be dispensed with. If the backlight is not required, all the related components (S1, R6, R9, R10 and T1) can be dispensed with.

In order to make the module no larger than the LCD, the printed circuit board (**Figure 2**) is almost entirely populated with SMD components. This demands a certain amount of deftness with the soldering iron, especially when it

comes to the LPC2103 in its LQFP48 package. Despite the 0.5 mm lead pitch, it is lining the device up with the pads on the board that is the tricky part of assembly, not the soldering itself. The easiest way to solder the device is to melt solder over all of the leads on one side at a time (not worrying about short circuits) and then remove excess solder using braid. It is helpful to use a little flux. It is best to mount the microcontroller first so that the other components do not get in the way of the action. Then solder all the other ICs and finally the other SMD components. Finally mount the leaded components. **Figure 3** shows our prototype board, which differs in minor details from the final layout shown in Figure 2.

Figure 2. The printed circuit board is mostly populated with SMD components and is the same size as the LCD panel.

COMPONENT LIST

Resistors:
(SMD0805, 0.125W, 1% unless otherwise indicated)
R1 = 0Ω05 2W 1% (SMD shunt resistor, 5mm x 0.3mm, e.g. Vishay/Dale)
R2,R5,R6,R7 = 10kΩ
R3 = 100kΩ (see text)
R4 = 1kΩ
R8 = 8kΩ25 (see text)
R9,R10 = 0Ω (or 1Ω, R10 also 220Ω, see text)
P1 = 10kΩ multiturn preset, 19mm (see text)
P2 = 100kΩ preset, horizontal mounting, 10mm

Capacitors
C1 = 100μF 63V 20%, aluminium, SMD (G) or axial wired
C2 = 330μF 25V 20%, aluminium, SMD (F) or axial wired
C3 = 100μF 16V 10%, tantalum SMD (7343-31)
C4 = 10μF 16V 10%, X5R, multilayer SMD 1210

C5,C6,C9,C10,C13–C17 = 100nF 50V 10 %, XR7, multilayer SMD 0805
C7 = 10nF 50V, 10%, XR7, multilayer SMD 0805
C8 = 1μF 16V 10 %, XR7, multilayer SMD 0805
C11,C12 = 22pF 50V 5%, multilayer SMD 0805

Inductors
L1 = 330μH, 590mA, 20%, SMD 10mm x 10mm (e.g. EPCOS)
L2 = 22μH, 925 mA, 30%, 5mm x 35mm (e.g. WE-TPC, Würth)

Semiconductors
D1 = 3A Schottky diode, 100V, 620mV, SMC case, e.g. 30BQ100TRPBF (Vishay, IRF)
D2 = LED, red, low current (2mA)
T1 = 2N7002 (N-channel MOSFET, 60V 115mA, Fairchild, SOT-323)
IC1 = TL2575HV-05 (5V 1A step-down switch-mode regulator, TI, TO263)
IC2 = TPS62007 (step-down switch-mode regulator, TI, MSOP10)
IC3 = TPS79118 (LDO regulator, 1.8V 100

mA, TI, SOT-23-5)
IC4 = MAX6520 (1.2V voltage reference, Maxim, SOT23-39
IC5 = LPC2103FBD48 (70MHz ARM7, 16KB Flash, 8KB RAM, NXP, LQFP48)
IC6 = MCP3550-50 (22-bit delta-sigma ADC, Microchip, SOIC08)

Miscellaneous
X1 = 14.7456MHz quartz crystal, 18pF load capacitance, 30ppm
S1 = SPNO pushbutton, PCB mount (e.g. Omron B3W-1000)
K1,K2 = PCB screw terminals, 16A, pitch 5mm
K3 = SIL 6-way right-angled pinheader
K4 = SIL 16-way socket strip, for LCD mounting
JP1 = 2-way pinheader for jumper
JP2 = 3-way pinheader for jumper
LC display, 2 lines of 16 characters, with backlight; see text (e.g. Elektor Shop # 030451-72)
2 jumpers, pitch 2.54 mm
4 standoffs M2.5, l = 15mm (0.6")
PCB no. 080824-1 from the Elektor Shop

Software

The ARM core used in the LPC2103 offers plenty of processing power to extend the capabilities of the unit. The firmware is written in C and source code is available for free download from the Elektor web pages for the project [1]. The program is easy to understand and modify. Only the start-up code has to be in assembler (for which thanks to Alexander Graf, who wrote that code). The free GCC C compiler is used.

The microcontroller runs in an infinite loop, and once per second calls a function which performs the measurement process and displays the results. To obtain the one second period, Timer

0 is configured so that its counter is incremented every millisecond. The infinite loops tests to see whether the counter has reached 1000 (as 1000 * 1 ms = 1 s).

The calculation routine first reads a value from the A/D converter to obtain a current reading. This is relatively straightforward as there is no need to send any data to the MCP3550: it is simply a matter of taking its chip select signal /CS low. It is then necessary to wait while the conversion takes place. The MCP3550 signals that the conversion is complete by taking the data output line low. We need to switch the port to the correct mode in order to detect this event.
Once the conversion is complete the

data bits can be read out. There is a total of three eight-bit data blocks, which are logically ORed into the appropriate positions in a variable. The twenty-second bit gives the polarity of the voltage measured at the device's input: if the bit is a '1' the value is negative and the collected value must be suitably modified: the two's complement is taken by calculating a bitwise (ones') complement (ad_current = ~ad_current;) and then incrementing by one. Since we only want the least significant 22 bits of the result we mask off the remainder using a logical AND operation: ad_current &= 0x3fffff;.

Now we can compute the actual current flowing. To avoid floating-point

The MCP3550

A simple and accurate delta-sigma A/D converter

The MCP3550 is a rather slow, but thrifty and accurate analog-to-digital converter. Its current consumption is just 110 µA! The MCP3550-50 was selected for this project because it has a differential input and a notch filter for suppressing 50 Hz AC power hum.

080824 - 14
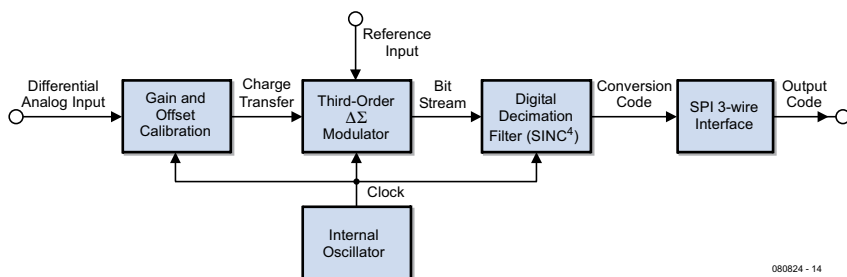
Its innards, as shown in the block diagram, appear simple. At the input is a 'gain and offset calibration' circuit. This compensates for offset and slope errors in the rest of the circuit. The actual A/D conversion is performed, with the help of the reference voltage, by the 'third order delta-sigma modulator', a one-bit converter that produces at its output a stream of bits at a high sample rate. The digital decimation filter then reduces the sample rate of this stream while simultaneously increasing its word length (and hence precision) to 22 bits. The digital circuitry is clocked from an internal oscillator, operating at 102.4 kHz in the case of the MCP3550-50.

080824 - 15
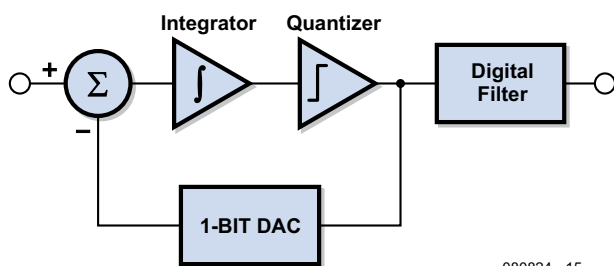
Even simpler is the block diagram of a first-order delta-sigma modulator. The input signal forms one input to a voltage subtractor. The other input to the subtractor comes from a 1-bit D/A converter fed from the output of the modulator, forming a negative feedback loop. The output of the subtractor can be thought of as an error signal, being the difference between the output of the modulator and its input. The error is integrated and quantised to form the output of the modulator. This loop operates at a much higher frequency than the quoted overall sample rate of the A/D converter and produces a high-frequency bitstream.

The MCP3550 uses a third-order delta-sigma modulator, which is an extension of the idea described above. Instead of one subtractor and integrator stage there are three connected in series. There is a quantizer after the final stage which feeds back to all three stages.

The bitstream produced is digitally filtered to produce a usable signal. In the MCP3550 this is a fourth-order modified sinc filter. A sinc filter (that is, a filter whose impulse response takes the form of a sinc function) has an ideal low-pass characteristic. It is here that the 50 Hz notch filtering also takes place. The output of the filter is a high-resolution representation of the input voltage, which can be read out over the SPI port.

operations we work with currents in nA and voltages in µV. The resulting error in the measured result is negligible, deviating from the exact value by less than the tolerance of the reference voltage or voltage divider.

All the displayed information is also output over the serial port in the following format:

Voltage: 12599mV
Current: –0050mA
Power: –0629mW
Capacity: –0035mAh

The output data can be fed to a data logger for subsequent analysis. If two of the units are used it is possible to make simple and accurate measurements of the efficiency of switching power supplies.

Programming

The battery monitor has a serial interface which uses TTL voltage levels rather than RS-232 voltage levels. Despite its 3.3 V supply, the ports of the LPC2103 are compatible with logic powered from 5 V.

The serial interface is used for uploading firmware to the device as well as for outputting readings. To program the device over USB the USB-to-TTL serial cable described in the June 2008 issue of Elektor [2] can be used: this is available from the Elektor shop (order code 080213-91). The pinout of the cable is compatible with that of K3 on the printed circuit board. Of course, you can equally well read the pinout of the connector from Figure 1 and use your own serial adaptor cable, as long as it uses 5 V logic levels.

When the circuit has been assembled the firmware must be flashed into the microcontroller. Close jumper JP1 and then apply power to the circuit. If the circuit is to be powered over the USB-to-TTL cable, JP2 must be set to position 2–3 (linking +5 V and +5 V$_{USB}$). At this point only the first row should appear on the display: if it is not visible, adjust the contrast using P2. The firmware can be downloaded using the free Flash Magic software [3]. **Figure 4** shows how this program is configured: only the COM port to be used and the path need to be set.

If you wish to avoid the fuss of programming the microcontroller yourself, it is available ready-programmed from the Elektor shop (order code 080824-41).

Start-up

When programming has successfully completed, remove JP1 and check that JP2 is correctly set (+5 V linked to +5 V$_{BATT}$). Connect a battery (rechargeable or not), or a bench power supply with an output voltage between 6 V and 40 V, to K1. If a voltage source was already connected to terminals K1, it will be necessary to remove it briefly to reset the circuit. The LCD should now show a brief message and then display the measured values. If nothing appears, check the contrast setting (P2) again. If the welcome message does not disappear from the display, it is likely that there is a fault in the connection to IC6. If everything is working as expected the LCD should show the current consumption of the meter module and the

Figure 3. Our populated prototype.

Figure 4. Settings required for the Flash Magic software to upload the firmware into the microcontroller's flash memory.

supply voltage on the top row. The second row shows the charge that has been transferred (in Ah) and the power draw (in mW or in W): see **Figure 5**. The current draw is strongly dependent on the supply voltage; at 12 V around 15 mA should flow.

If P1 has been fitted it should now be adjusted so that the displayed voltage agrees with the actual voltage at K1 (as measured by your trusty multimeter). The Battery Monitor is now ready for use.

(080824-I)

Internet Links

[1] http://www.elektor-usa.com/080824

[2] http://www.elektor-usa.com/080213

[3] http://www.flashmagictool.com/download.html

Datasheets

TL2575HV-05:
www.ti.com/lit/gpn/tl2575hv-05

LPC2103:
www.nxp.com/acrobat/datasheets/LPC2101_02_03_3.pdf

MCP3550:
http://ww1.microchip.com/downloads/en/DeviceDoc/21950D.pdf

TPS62007:
www.ti.com/lit/gpn/tps62007

TPS79118:
www.ti.com/lit/gpn/tps79118

MAX6520:
http://datasheets.maxim-ic.com/en/ds/MAX6520.pdf

DEM16217:
www.display-elektronik.de/DEM16217SYH-LY.PDF

About the author

Steffen Graf is 20 years old and is in his second term studying Electronics and Information Technology at the Technical University of Darmstadt, Germany. Electronics has been one of his main hobbies for many years. In 2008 he entered a battery capacity monitor design for a German regional youth science competition, winning second prize in the 'Working World' category. He also won a special prize for a practical and useful invention.
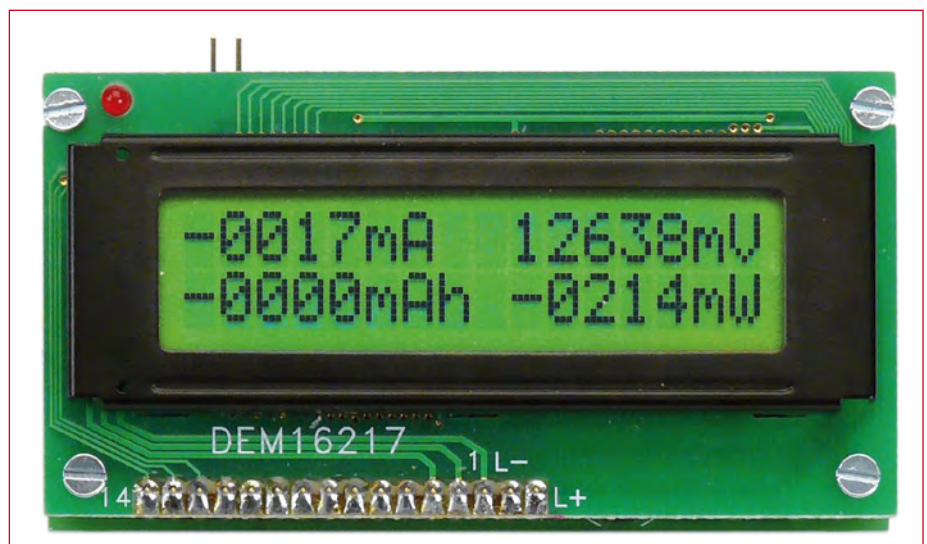
Figure 5. The LCD panel in normal operation.
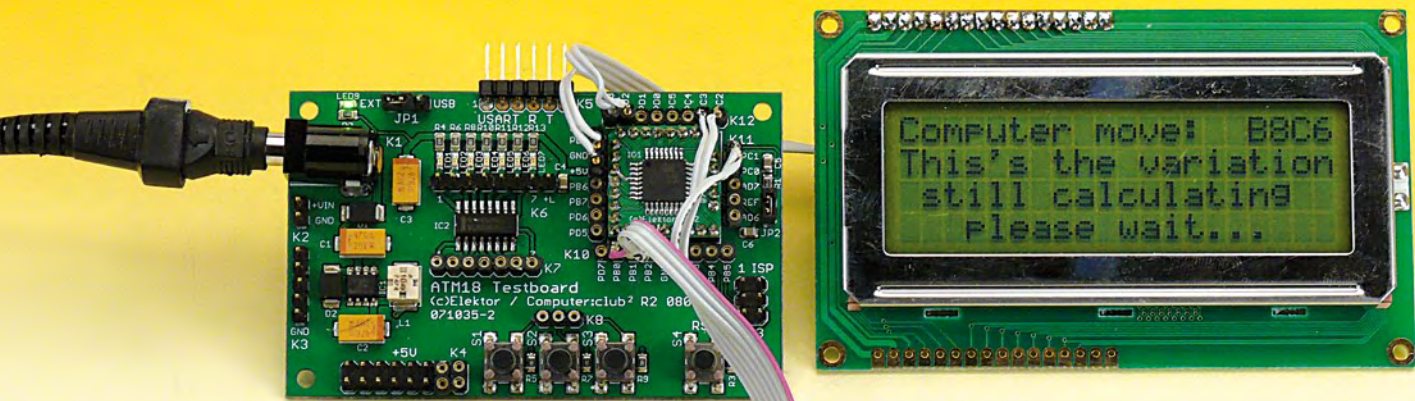
ATM18 Mini Chess Computer

Elektor AVR board rubs shoulders with kings, queens, bishops...

By Wolfgang Rudolph with Andre Adrian (Germany)

The Elektor ATM18 system can be used to make a surprisingly simple and effective chess computer. The only additional hardware required is a few low-cost pushbuttons. The software for the project is written in C, and it was far from an easy task to fit the program in the 8 kB of program memory offered by the ATmega88.

Chess holds a special position in the world of board games. Its origins can be traced back to a game called 'Chaturanga', played in India. The game spread to Arab lands via Persia, and was then brought to Spain and the rest of Europe by the Moors. By the fifteenth century the game was enjoying widespread popularity in Europe. The name 'chess' is thought to derive from the Persian word 'shah', which can be translated approximately as 'king'. Following a major revision of the rules, the first chess tournaments began. The earliest such tournament took

place in 1851 in London, and the first chess world championship was fought in 1886 between Wilhelm Steinitz and Johannes Zuckertort.

Playing perfect chess is well beyond the capability of even the fastest computers. Even the number of distinct positions is not known exactly, although it has been estimated at around 2.28×10^{46}, and the number of possible games lasting 40 moves by each player is approximately 10^{120}. For comparison, scientists estimate the number of atoms in the billions of galaxies in the universe at only around 10^{78}.

From the 'mechanical Turk' to the chess computer

As early as 1769 the Hungarian court official and skilled mechanical engineer Wolfgang von Kempelen demonstrated a chess-playing automaton, to the astonishment of his contemporaries. Unfortunately it was an elaborate hoax: the 'robot' sitting at the chess table, dressed as a Turk (**Figure 1**), was operated via a complex system of levers and linkages to make moves on the board, all under the control of a man hidden inside the table.

It was not until long after Kempelen's death that the secret of his 'mechanical Turk' became generally known. The first chess machine actually capable of calculating and executing moves was a mechanical device constructed at the beginning of the twentieth century by a Spanish engineer named Leonardo Torres y Quevedo, who had also developed a transatlantic airship and a cable car over the Niagara falls. In 1912 his 'El Ajedrecista' automaton was capable of winning the endgame of white king and rook versus black king. The algorithm the machine used was not optimal, and could take up to 63 moves to deliver checkmate to the black king. An improved version was constructed in 1922 under Leonardo's direction by his son Gonzalo. It was first demonstrated at the Paris Cybernetic Congress in 1951. This 'chess computer' is of a purely electromechanical design and is still in working order. It can be seen in the Torres Quevedo museum at the Polytechnic University of Madrid.

Konrad Zuse [1], a pioneer of the modern computer, joked to his friends in 1938 that within fifty years a computer would be able to beat the world champion at chess. It was a bold prediction at the time, but in retrospect surprisingly prescient. He wrote his first chess program in the language 'Plankalkül' ('plan calculus') [2], which he had developed between 1941 and 1945. The language was not, however, supported by the computers of the time; it was not until 2000 that the first Plankalkül compiler was written. Chess programs suddenly gained enormous popularity with the advent of the first microprocessors and the introduction of home computers. The first was 'Microchess', by Peter Jennings, written for the MOS Technology 6502 processor. Even more successful was 'Sargon' by Dan and Kathe Spracklen. Sargon originally ran on Z80-based systems and was later ported to the Apple II. The Apple version sported 1-bit digital speech synthesis using the built-in loudspeaker. Since then, hundreds of chess programs, ranging from the excellent to the awful, have been released.

More information about the history of chess programming and chess computers can be found at [3], [4] and [5].
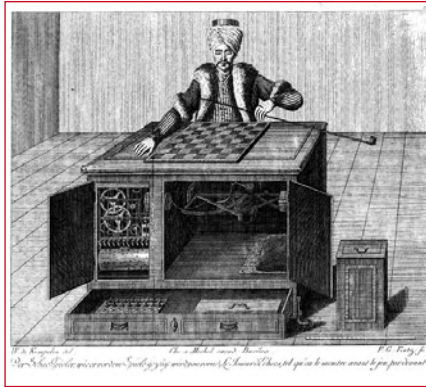
Figure 1. The celebrated 'mechanical Turk' hoax chess automaton of 1769.

Figure 2. Konrad Zuse, shown here with a replica of his Z1 computer, was a pioneer of modern computing and even wrote a chess program. (Photograph courtesy of Prof. Horst Zuse)
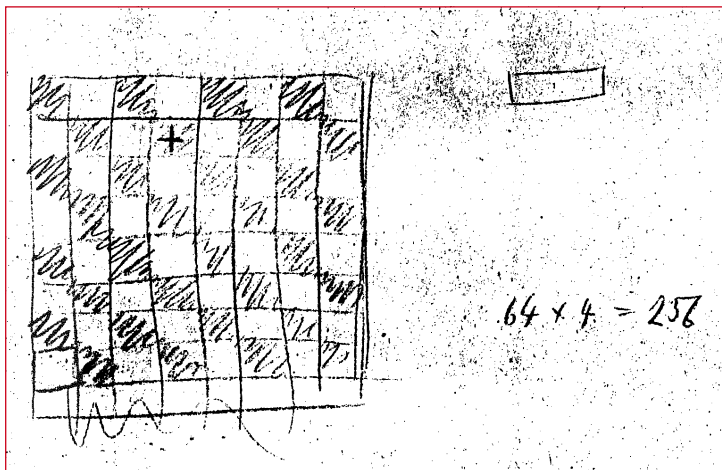
Figure 3. Sketch by Konrad Zuse dating from 1941, thought to show the memory requirement calculation for an internal board representation (64 squares x 4 bits per square = 256 bits total). (Source: Konrad Zuse internet archive at http://www.zib.de/zuse/English_Version/index.html)

Chess on the ATM18

The chess program Micro-Max [6] was written by Harm-Geert Müller [7]. As Müller himself writes, it was his goal to write the shortest chess program in the world in the C programming language. His target was 1024 bytes, but even omitting the castling and *en passant* rules the smallest he could achieve was 1200 characters. He therefore increased his target source code size to 2 kB and included almost all of the official FIDE (Fédération Internationale des Échecs, or World Chess Federation) rules of chess. Validation of user input is also included. Initially the only missing feature was 'under-promotion' (promoting a pawn to a piece other than a queen on reaching the eighth rank), which he felt unnecessary; however, the feature was subsequently built in to a later version of the program.

The program consists of fewer than 150 lines of C source code totalling under 2000 ASCII characters. Andre Adrian has ported Micro-Max version 4.8 to an Atmel AVR 8-bit microcontroller using the WinAVR GCC compiler. The result is a compact chess computer whose LED display and operation are reminiscent of the celebrated 'Mephisto' dedicated chess computer. This 'AVR-Max' project will be described in a future issue of Elektor. From this project sprang the idea of adapting AVR-Max to run on the Elektor ATM18 system. The project looked rather tricky at first. The LED version uses a timer interrupt routine to drive the display and scan the keyboard, which does not fit well with the two-wire LCD interface used on the ATM18. Despite a number of early problems, eventually all the bugs were tracked down and ironed out. The software was modified to display additional text on the two-wire LCD module. This makes the ATM18 version of the mini chess computer rather more communicative with its adversary than its AVR-Max cousin can manage with its four-digit seven-segment display. As usual the software is available for free download from the Elektor web site [8]. The keyboard scanning code present in the LED version of the software has been completely carried over to the ATM18 version,
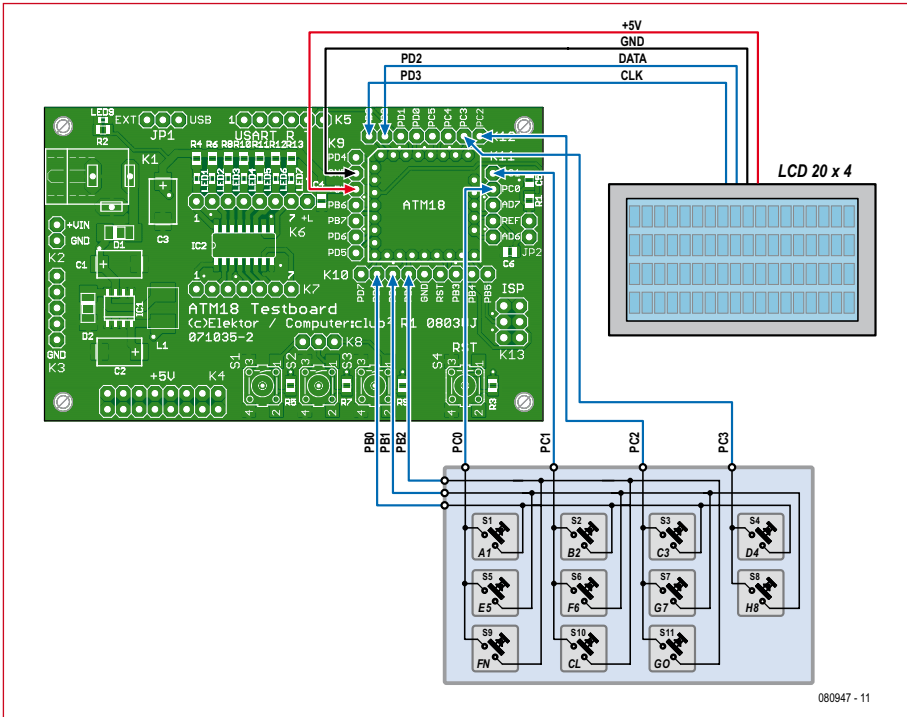
Figure 4. Wiring diagram for the CC2 mini chess computer.
The two-wire display and 11 pushbuttons are connected to the ATM18 test board.

and so the only addition necessary to the hardware is to connect a matrix of pushbuttons to the ATM18 board.

Keyboard input

The rather grandly-titled 'keyboard' consists of a total of eleven pushbuttons, eight (labeled 'A1' to 'H8') which are used to enter moves, and three function buttons. Seven wires suffice to connect the buttons to the ATM18 board, using a total of just seven port pins. The circuit diagram of the keyboard is shown in **Figure 4**. It should be clear from the figure how the buttons can be arranged on a piece of perforated prototyping board and wired up. In the figure the eight buttons used to enter moves are shown

in two rows of four, with a separate row comprising the three function buttons (marked 'FN', 'CL' and 'GO'). Other arrangements for the keyboard are of course possible: as an example, the photograph of our prototype shows three rows of three buttons and a final row of two.

The 'FN' button is used to access special functions. The 'CL' button is used to correct mistakes made after a move has been entered, but before the 'GO' button has been pressed. The 'GO' button has an extra function besides its use as an 'Enter' key, which will be described in the section below on operation.

The keyboard is wired as a matrix. It is driven using open-collector outputs on the microcontroller, which means that if more than one button is pressed simultaneously the outputs of the processor will not be damaged by the short-circuit.

It is advisable to use very flexible cable for the seven-wire connection between the ATM18 test board and the keyboard, and it is a good idea to mount the three printed circuit boards (test board, keyboard and LCD) on a baseboard or similar fixing to ensure that the connecting cables do not suffer undue strain.

The AVR-Micro-Max chess program

The program uses the 'negamax' search algorithm to calculate its next move. It is implemented in Micro-Max as a recursive function. The negamax function calls itself until a termination condition (such as its reaching a maximum allowed search depth) is satisfied. Recursive functions require stack-based storage for local variables and for saving values in the CPU's registers. However, it is always possible to rewrite a recursive function in iterative form: instead of having the function call itself, we use a loop around the function. There is then no longer any need to preserve the CPU's registers. For simple functions, such as calculating factorials or elements of the Fibonacci sequence, the conversion to iterative form is trivial. The negamax algorithm can also be modified in this way, and Andre Adrian has implemented such a version. His implementation requires just 34 bytes of storage per half-move (a move by one player, also called a 'ply'). Storing the chessboard in '0x88' format requires 129 bytes of RAM. With 1 kB of RAM available, we can therefore search to a maximum depth of

---

The Elo rating system

The 'Elo system' is a way of grading the strength of chess players on a numerical scale. It was proposed by Arpad Elo and adopted by the United States Chess Federation in 1960. In 1970 it was also adopted by FIDE and became an internationally accepted standard. It was renamed the 'FIDE rating system', although the term 'Elo rating' is still in common use. Other national rating systems are also used alongside the FIDE system.

A beginner will typically have an Elo rating of below 1000. Amateur players' ratings normally lie between 1200 and 2000, while experts and would-be masters have ratings of between 2000 and 2200. Masters will have ratings of from 2200 to 2500; beyond that lie the grandmaster and so-called 'super-grandmaster' (above 2700) ranks.

There are also Elo ratings lists for chess computers: well-known examples include the SSDF list (http://ssdf.bosjo.net) and the Wiki Elo list (http://www.schach-computer.info/wiki/index.php/Wiki-Elo-Liste_En).

20 half-moves. The iterative version requires no more ROM space than the recursive version: stack accesses, relative to the stack frame pointer, take a similar amount of code on the ATmega microcontroller to the array accesses that replace them.

The AVR-Micro-Max program can also be run in the AVR Studio simulator (the free version is adequate for this purpose). If you feel so inclined you can modify the program to suit your own preferences and test the results immediately in the simulator. The GCC compiler is of course available free of charge and free of arbitrary limitations.

Our aim was that the program should play at a strength of between Elo 1200 and Elo 1399 (see text box) under tournament time controls (120 minutes for the first 40 moves). This corresponds to amateur class D, an average hobby player. The program has nine levels of play. Level 1 is a 'blitz' mode, taking about 7 seconds per move; level 5 is suitable for quick games at around 30 seconds per move, and level 8 is designed for tournaments, taking 3 minutes per move. On power-up the computer defaults to level 3.

Keen ('coffeehouse') amateurs typically play at strengths of between Elo 1400 and Elo 1700, and it is rare for non-club players to have ratings above Elo 1800. Although our chess computer is not capable of playing at such elevated levels, it can still play a challenging game.

Operation

When the software has been flashed into the microcontroller module and the keyboard, display and power supply have been connected, everything is ready to go. It is easiest to play with a real chessboard to hand on which to make the moves as the game progresses.

When power is applied the text shown in **Figure 5** will appear on the display. You can now enter a move, such as d2d4 (P-Q4), by pressing the following sequence of buttons:

D4, B2, D4, D4

The LCD now shows:

Your move: D2D4
Enter coordinates or GO to continue

If your move was legal, the computer will now calculate its move. If this will take some time, a 'please wait' message appears. The computer's move will then appear on the display along with an invitation for you to enter

Figure 5. When power is applied the computer invites you to enter your first move.

your reply. If your move was not valid, a suitable message appears on the display.

If one of the players is checkmated, the message 'Checkmate!' appears on the display.

The 'CL' button is used to correct entry errors before 'GO' is pressed.

On power-up the computer takes the black pieces; if 'GO' is pressed at this point, the computer will take the white pieces and play the first move.

The 'GO' button has a further use. If it is pressed without entering a move, the computer will calculate a move for you. This can be used to exchange sides during a game: if you are on the point of losing to the computer, simply press 'GO' without entering a move and the computer will take your side for the rest of the game. If you repeatedly press 'GO' without entering moves the computer will play itself with you as a spectator.

Pressing the 'FN' button leads to a menu with three options.

1. Pressing 'A1' starts a new game.

2. Pressing 'B2' followed by another button selects a new level: press 'A1' to 'H8' for levels 1 to 8 and 'FN' for level 9.

3. Press 'C3' to enable or disable display of the principal variation (PV).

When PV display is enabled the computer will show the move that is currently best in its opinion. Every time the depth-first search returns to the root of the game tree the computer updates its idea of the 'best move found so far'. It then checks to see if it has used all of the time allotted to its move. If it has, it plays this as its next move; if there is still time left, it increases its search depth by one half-move and repeats the search. This process is called 'iterative deepening'.

Check and mate

You should find it quite a challenge to beat the computer on its higher levels. A tip: you can play the first few moves at a lower level and then increase the level when the opening is over. The computer does not have an openings book. It is possible to force the computer into a given opening by entering white's and black's moves manually, pressing the 'GO' button just once after each move. One popular opening begins 1. e2e4 e7e5 2. g1f3 b8c6 (1. P-K4 P-K4 2. N-KB3 N-QB3).

If the ATM18 system is to be used permanently as a chess computer it is a good idea to fit it in an enclosure, the details of which we will leave to the imagination of our readers. Good luck!

(080947)

[1] www.zuse.de/Konrad_Zuse/index.html

[2] www.zib.de/zuse/English_Version/index.html

[3] www.andreadrian.de/schach (in German)

[4] www.computerhistory.org/chess/

[5] www.geocities.com/SiliconValley/Lab/7378/comphis.htm

[6] http://home.hccnet.nl/h.g.muller/max-src2.html

[7] http://chessprogramming.wikispaces.com/Harm-Geert+M%C3%BCller

[8] www.elektor-usa.com/080947

Getting Started with

Part 3: Miscellaneous interrupts and determining

By Bert Korthof MA (The Netherlands)

In this third and final article of the short Embedded C course, based on the MSP430 experimenter's board, we get to grips with some of the components on the board that haven't been used so far, such as the 7-segment display and the piezo buzzer.

Amongst other things, the MSP430 board contains a segment display, a 4094 shift register used to drive this display, a piezo buzzer, a watch crystal with a frequency of 32,768 Hz (connected to the internal TimerA of the MSP430) and connectors for adding extra hardware, such as an I²C interface or an analog input signal for the 16-bit A/D-converter. We'll be using these parts in this final installment of the course. We'll also take a look at the software, and investigate the uses of polling and interrupts.

Polling and interrupts

We begin with polling and interrupts. Below is an example C program that uses polling.

```
void main(void)
{
   while(1)
   {
      ………..;
      if (!(P1IN & BIT5)) a=1;
               // push-button S1, active low
      delay(65000);
      set_7segment(3,1)
      ……..;
   } // endless loop
}
```

When we flash this program on the board and run it in step-mode, then pressing S1 will only set 'a' to 1 if the processor happens to execute the IF instruction at that time. If the delay() function was called during the brief time that the push button was pressed then 'a' won't be set to 1. In this case we have to keep the button pressed until the IF instruction is executed. In time-critical processes such as time keeping or an alarm function this could cause a lot of problems. This method of determining the status of push button S1 is called **polling**; the status of S1 is determined every time the IF instruction is executed.

Inside the processor chip is some special hardware that can interrupt the workings of the processor. This hardware makes the processor stop after it finishes the instruction it is currently processing, then remembers where it is (e.g. it stores the value of the program counter on the stack) and then calls a special **Interrupt Service Routine** (ISR), see **Figure 1**. Once the ISR has finished the processor jumps back to the exact point in the main program where it was interrupted. The main program then carries on as normal,

except that a few milliseconds delay occurred.
Imagine that we've configured a port pin to initiate an interrupt when its input is a certain state. This means that the program will be interrupted (for example during statement x) when the interrupt hardware detects the pre-programmed state on this pin. The interrupt service routine will then be called after which the processing will resume where it left off in the main program.
There are several ways in which the hardware can generate an interrupt:

- by a change of state on a port pin;
- by the watchdog timer;
- when the counter of TimerA is 'full';
- when the compare register of TimerA is equal to the value stored in CCR0;
- by setting a 'flag' in software, for example to be able to react immediately to a fault condition.

When an ISR is active and a 'flag' is set at that instant for another ISR, then the first ISR is completed before the new ISR is dealt with. This interrupt is therefore not lost but just delayed in its execution. For this reason it is important to keep ISR routines as short as possible, since they are often time-critical and this way several different ISRs can be executed one after another. When more than one interrupt flag is active, the ISRs are dealt with in order of priority (which is fixed in the hardware). For example, a CCR0 interrupt takes precedence over a port pin interrupt.

For port pins that have push buttons connected to them there are two possibilities (see **Figure 2**). The push buttons (S1, S2 en S3) on the MSP430 board are connected via pull-up resistors to the positive supply rail (V_{cc}) and go to P1.5, P1.6 and P1.7. In the non-active state of push button S2 the signal at P1.6 is high and this becomes low when S2 is pressed (falling edge). We can configure the registers for the port interrupts such that an interrupt is **automatically** generated when a falling edge is detected. Each port pin has its own setting for interrupts, which can be configured via the P1IE register (enable interrupts for this port pin); P1IES is the interrupt selection register that is used to choose between a rising or falling edge. Each port pin interrupt also has its own interrupt flag (P1IFG); in other words, when this is at '1' it means that its interrupt is in progress, unless another interrupt is active. As an example, a few register values are shown in **Figure 3**.

Embedded C

your reaction speed
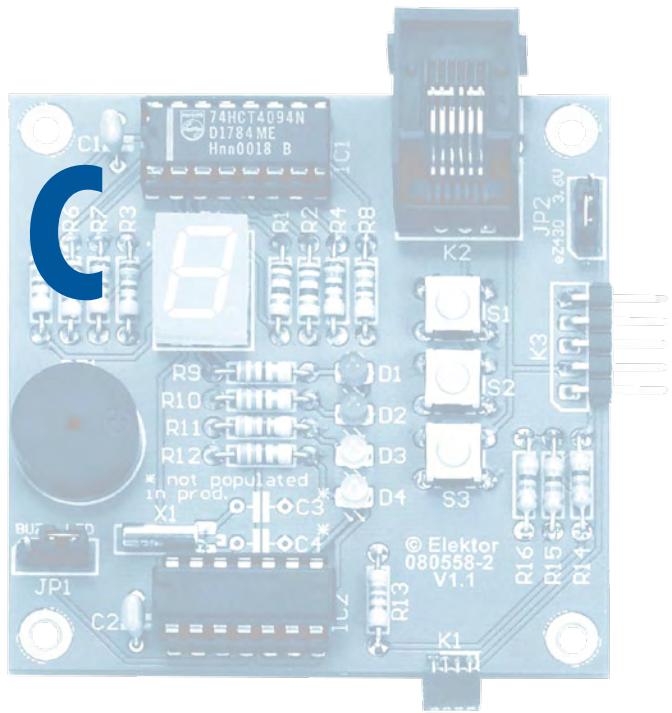
Interrupts can therefore only occur when the signal changes in the right direction and the relevant port pin has its interrupts enabled. In the example shown, this only applies to the input signal at P1.2: this pin has interrupts enabled for a falling edge, so the corresponding flag has been set in P1IFG.

The ISR is defined outside the program main(), just as with normal functions:

```
void main(void)
{
    ................ .
    P1IE |= BIT5;   // signal from push button S1
                    // can generate interrupts
    P1IES |= BIT5;  // active on falling edge
    _EINT();        // enable interrupts

    the rest of the program follows (no more
    statements for interrupts are needed, since
    the processor automatically deals with
    these)
    .........................
}
#pragma vector=PORT1_VECTOR
        // interrupt for port P1.5 via S1
__interrupt void Poort1actief(void)
                // ISR function
{
    P1OUT |= (BIT11 + BIT2);
                    // turn on LEDs 1 and 2
    P1IFG &= ~BIT5; // reset interrupt flag
}
```

For this interrupt we have to reset the flag ourselves at the end of the ISR, otherwise the processor would immediately jump back to this ISR as soon as it had completed!

Next we'll take a look at the important TimerA interrupt. Since the board contains an accurate watch crystal, we can automatically interrupt the main program at fixed intervals to run an ISR, for example to keep the time or to generate precise PWM (Pulse Width Modulation) signals.

The watch crystal generates an accurate frequency of 32,768.000 Hz. The accuracy depends somewhat on the parasitic capacitance of the layout of the tracks on the board and can be adjusted by turning on internal capacitors of the processor. For more details refer to the application note SLAA322 by Texas Instruments. The crystal is connected to the internal 16-bit TimerA; see the block diagram
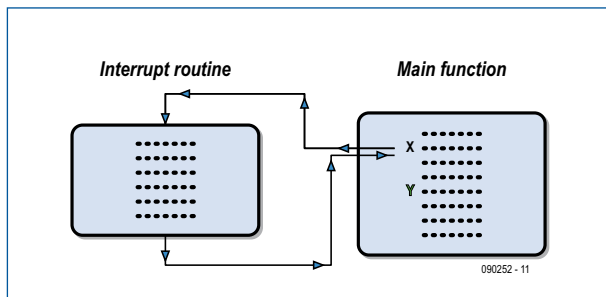
Figure 1.
After completing an interrupt routine the program jumps back to the exact location where the main program was interrupted.
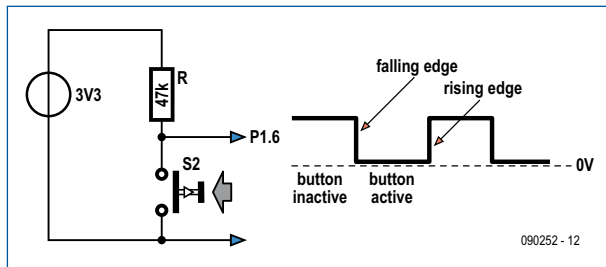
Figure 2.
The push buttons are connected to the supply via pull-up resistors and produce a logic Low level when they are pressed.

P1IE | = BITx; //signal on port pin x interrupt-activated
P1IES | = BITx; //active on falling edge
P1IES & = ~BITx; //active on rising edge

Figure 3.
Several examples for setting the interrupt registers.
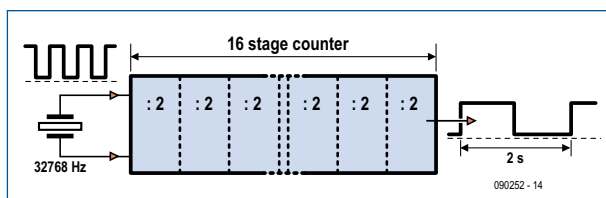
Figure 4.
The watch crystal is connected to the internal 16-bit TimerA. After division the timer has a period of 2 seconds.
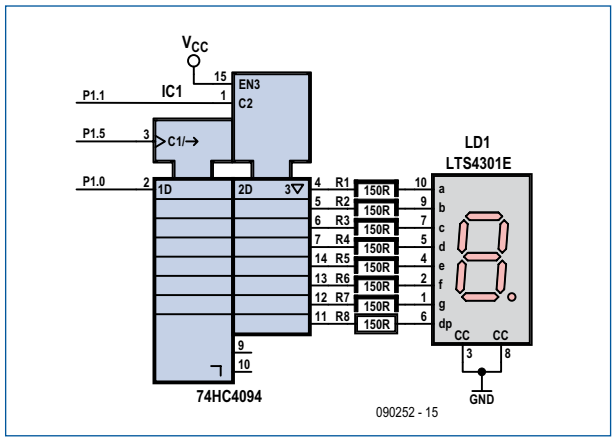
Figure 5.
The 7-segment display is driven by a 74HC4094 shift register.
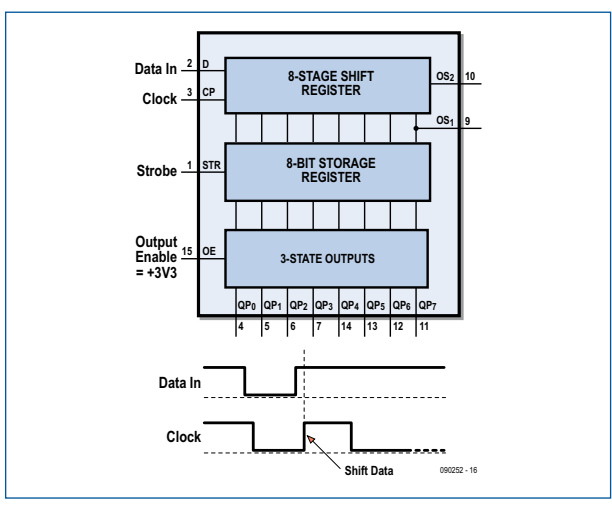
Figure 6.
The 8-bit 4094 shift register contains a shift register, a storage register and output buffers.

This can be used by the **TimerA overflow interrupt** in the software in order to increment a counter and hence keep track of the minutes, hours and even days and months.

One day has $24 \times 60 \times 60 = 86{,}400$ seconds. If we require the board to be accurate within 1 second per day it follows that the maximum deviation of the frequency may be $(1/86400) \times 10^6 = 12$ ppm (parts per million). The crystal frequency should therefore be this accurate as well, hence the facility to fine-tune the crystal frequency via the software.

There is the facility to generate even longer periods by adding two extra stages (ID0 and ID1, divide by 2,4 or 8) to the input of the timer with the following code:

```
TACTL = TASSEL0 + TACLR;
               // TimerA clock is ACLK + clear
TACTL |= MC1;   // TimerA in Continous UP mode
TACTL |= TAIE;  // enable overflow interrupt
TACTL |= ID0;   // + ID1;
     // TimerA overflow after 4 [2,8 or 16] sec.
CCTL0 |= CCIE;  // enable CCR0 interrupt
_EINT();        // enable all active interrupts
```

TACTL is the 16-bit TimerA control register that is used to configure the mode in which TimerA functions. In this case we use the UP mode.

Since we're always dividing by two, we end up with the exact binary value of the number of pulses counted. When we reset the counter (TACTL | = TACLR;) it will count from 0 to 65536 in two seconds. Since the binary value of the number of pulses counted is available we can also generate an interrupt for any arbitrary value. To do this you should store the required value in the CCR0 register. If CCR0=32768, then there will be a timer interrupt after exactly one second. This is called the **TimerA compare interrupt**, where the value of TimerA is compared with the value in the CCR0 register.

We have written an example program that tests this type of timer interrupt (see the file *TimerInterruptDemo.c* in the software package for this article, available from the Elektor website, file # 090252-11.zip). When you flash this file to the board and run it, you'll see LED D1 flash every two seconds and every minute you'll hear a beep. If you start the program exactly on the hour you could wait one day and compare the time with that of a DCF clock to find out

in **Figure 4**.
The counter repeatedly divides the input signal by two. After 16 stages we end up with a frequency

$$f = 32768 / 2^{16} = 0.5 \text{ Hz}$$

which means there will be a rising edge every 2 seconds.

Figure 7.
Part of the code from the program demodisplay.c.

```
for(i=1;i<=128;i*=2)        // i=1->BIT0;i=2->BIT1; etc. loop for all 8 display segments
    {
    for(j=7;j>=0;j--)       // generate 8 data bits, starting with MSB
        {
        if (i & (BIT0 << j))  // e.g. j=7: BIT0 --> BIT7 (make i =6: display =1, etc.)
            {
            P1OUT |=BIT0;     // set DataIn high
            }
        else
            {
            P1OUT &= ~BIT0;   // set DataIn low
            }
        P1OUT |= BIT5;        // set CLK
        delay(0);             // a very small delay
        P1OUT &= ~BIT5;       // reset CLK
        delay(0);
        }
    P1OUT |= BIT1;            // set strobe high
    delay(0);
    P1OUT &= ~(BIT0|BIT1| BIT5); // reset all outputs
    delay(1);                // about 1 s to display 1 segment
    }
```

how many seconds the crystal is out by per day (and then possibly use that information to make a correction to the program).

At this stage I'm happy to give some general advice to enthusiasts and students who design their own hardware and software: when creating new software for new hardware it is important to only add new functionality to your code when you can base it on code that is known to work properly. It is a pity when you end up programming for days, but you can't progress because of various mistakes in the code and you don't know what exactly is happening and what is going wrong. For example, you could start by checking that the crystal on the board with its thin connecting wires functions properly (crystals are sensitive to shock and sudden impact, such as being dropped onto the floor). You should therefore first load the *TimerInterruptDemo.c* file in order to test the hardware!

7-segment display

Before we look at the program *Game2Elektor_080558.c*, which tests our reaction speed, we first look in detail at the hardware and software for the 7-segment display.

In **Figure 5** you can see the part of the circuit diagram of the MSP430 board that deals with the display. In this circuit you can see that three lines go to the 74HC4094 and 7+1 lines go to the display. This method was necessary because our MSP430 processor has only eight I/O lines and other peripheral components also needed to be connected, such as push buttons and LEDs. This solution has consequences for the software: we now have to send the eight bits of data for driving the segments in a serial manner instead of parallel, which requires extra code and processor time. Again we see how the hardware and software influence each other: adding a bit more software means that we can reduce the amount of hardware required!

The display (LTS4301E) is a common cathode (CC) type, which means there is a common ground for all the LEDs in the display. We drive these LEDs via current limiting resistors with a value of 150 Ω. The current through the LED is then about

$$(V_{cc} - V_{LED}) / 150 = (3.3 - 1.9) / 150 \approx 10 \text{ mA.}$$

The 4094 8-bit shift register has a data input, a clock input and a strobe input (see **Figure 6**). After eight clock pulses the 8-bit data will have been shifted completely into the chip and can then be saved in the storage register using the strobe signal. The level at the data input is read during the rising edge of the clock signal, as can be seen in the waveforms in Figure 6. Pin 15 of the IC is connected to the supply voltage, which means that the output buffer is always activated and can source current to the LED segments. Output QP7 (pin 11) drives the LED for the decimal point (dp).

Before we start developing our own code to drive the display we have to test the hardware associated with the display. We do this by flashing and running the file *demodisplay.c*. Part of the code for this program is reproduced in **Figure 7**.

The outer FOR loop determines what number is sent serially. This is the variable 'i', which has a value of 1, 2, 4, 8

etc. (these values are referred to as BIT0, BIT1, etc.). Of the 8 bits only one will be high at a time. When all bits have been shifted inside it follows that only one segment will light up. All segments will then light up one after another, including the decimal point.

Variable 'i' is sent serially to the shift register. The MSB (Most Significant Bit) is sent first. The inner FOR loop starts with j=7, which becomes BIT7 due to the instruction BIT0 << j. This instruction makes the lowest bit shift 7 places to the left, thereby becoming BIT7 (in reality BIT0 is multiplied by 2^7). Next the instruction (i & (BIT0 << j)) checks if this bit is high or low. For the first time round this loop this becomes: if(1 & BIT7). This is not true so the data line to the shift register is made low via the ELSE instruction (see the arrows in Figure 7).

In this way the inner loop sends all eight bits when i=1 to the shift register and clocked in, as we momentarily make the clock signal high and then low again. When all eight bits have been sent to the shift register the contents of this register can be copied to the storage register using the strobe signal.

The outer loop makes each segment light up in turn. Run the file *demodisplay* to check that it works!

Test your reactions

We'll now take a closer look at the code in the file *Game2Elektor_080558.c*. This program generates a random number, which is used to generate a beep a random time after S1 is pressed. This can happen as soon as 0.25 s later or at most after 2 s. Following the beep you should press S3 as quickly as possible. TimerA is connected to the 32 kHz crystal and is reset at the time the beep is generated (TACTL |= TACLR;). The relevant code from the above file is shown here:

```
TACTL |= TACLR;       // TimerA clear(TAR=0),
                      // start measuring
while(!ready);        // wait for S3
t=TAR;                // read contents of TimerA
c=t/reshigh;          // resolution of 0.05 s
if(c>15) { c=-1;dp=1;}
else dp=0;
P1OUT &=~BIT4;
delay(65000);
```

```
P1OUT |= BIT4;
set_7segment(c,dp); // clear 7-segment display
```

The timer starts counting from 0. The **TAR register** of TimerA will then have a value of 0 and will count upwards. The program hasn't gone further than the while(!ready); loop. This loop consists of just a single statement because there is a semi-colon after while(). As long as ready=0 this statement will keep executing. At the instant when S3 is pressed the port pin interrupt will be called, and inside this ISR the variable ready is set to 1. The while() loop is then terminated and the value of the counter is stored in the variable t. This value is therefore proportional to the amount of time passed since the beep sounded. An extra divider (x1/2) is added after the 32,768 Hz crystal using the instruction TACTL |= ID0. This means it will now take 4 seconds before the timer fills up. We can now calculate the conversion factor required to show the time in tenths of a second. Four seconds correspond to 65536, so 0.1 s corresponds to a counter value of 65536/40 = 1638. We called this constant reslow using the define statement: #define reslow (1638). For people with a very fast reaction time a '1' will appear on the display, which corresponds to a reaction time of less than 0.2 seconds. We have also added a higher resolution mode using the variable reshigh ($1/2$ * 1638 = 819) where we can measure in units of 0.05 s.

In this program we made use of the source code from the second instalment, but for easy reference we have combined all code within one file. In this instance we haven't used a library file, which is often useful when we have a number of well-documented functions already written. For clarity we have shown the three interrupts used in **Figure 8**.

What is noticeable is that we have to reset the interrupt flag ourselves in the software, except for the TimerA0 compare interrupt (where the hardware automatically resets the flag when the interrupt finishes).

It is very useful to go through a program in step-mode and gain an understanding of all the instructions used, which makes it easier to think of your own interesting programs and then create them.

As is often the case, you learn the most when you try things out yourself!

(090252-I)

```
#pragma vector=TIMERA1_VECTOR    // timer-overflow interrupt, value of
__interrupt void Timer (void)    // TAR = 65535; every 4 seconds
    {
     P1OUT &= ~BIT3;                 // turn LED D3 on
     delay(3000);
     P1OUT |= BIT3;                  // turn LED D3 off
     TACTL &= ~TAIFG ;               // reset interrupt overflow flag
    }

#pragma vector=TIMERA0_VECTOR     // counter TimerA equals CCR0 value
__interrupt void timerA_Compare_IRQ(void)
{
    start =1;
}

#pragma vector=PORT1_VECTOR
__interrupt void buttonactive(void)
{
    ready=1;
    P1IE = 0;                        // turn off port interrupts
    P1IFG = 0;                       // reset interruptPort flag
}
```

Figure 8.
The three interrupts used in the program Game2Elektor_080558.c.

ElektorWheelie
Construction and driving

In the first instalment about the self-balancing single-axle electric 'vehicle' in the Summer Circuits issue we explained the operation of the electronics. There is, of course, also some mechanical construction required before the ElektorWheelie is ready to be used.

The ElektorWheelie is delivered as a kit containing all the necessary parts. The kit contains all electronic bits, the batteries, the steel housing, the wheels, the motors, the control lever and an economy battery charger. The printed circuit board comes completely assembled and ready mounted in the housing, just like the motors. In this article, with the help of a few photographs, we describe what remains to be done by you.

DIY tinkering

1. The main things you have to do yourself is mounting the control lever and attaching the wheels.
2. We start with the control lever. For this you need the brackets/bearing mounts and four bolts with matching washers and nuts.
3. The mounting brackets/bearing mounts are attached with the bolts to the middle of the base plate (between the motors). The housing already has the mounting holes for these. Tighten the bolts firmly so that they cannot come loose due to vibrations.
4. Now you need to attach the coupling between the control lever and the potentiometer. Start by adjusting the potentiometer to roughly the centre position. Make sure there is sufficient free movement clockwise as well as anti-clockwise because the pot is used to determine the position of the control lever.
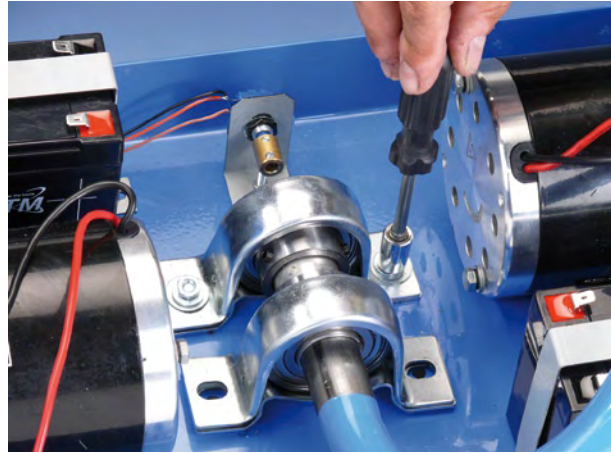
5. Now turn to the cabling. The wiring diagram shows the electrical connections effectively made. Note that some cables may be a bit short for the next step and should be connected afterwards.

6. You can now slide the cover at an angle over the control lever and fasten it at the sides with four screws. After this you attach the wheels and the two remaining parts of the control lever. Make sure the large wheel nuts are fastened securely!

As already mentioned, the kit is supplied with a simple charger. A complete battery charging cycle takes about 16 hours. An improved charger based on an earlier Elektor design is described elsewhere in this issue. This circuit ensures that the batteries can be charged independently so no imbalance can occur in respect of individual capacity. Before the first trip it is a good idea to test whether the ElektorWheelie functions properly. To do this, place the Wheelie on a crate or box with the wheels off the ground. Turn the Wheelie on and check whether it responds correctly to tilting forwards and backwards. Also check the operation of the control lever.

The first trip

To allow you to drive the ElektorWheelie, you first need to insert the double pin of the *Safety Switch*. The electronics can now be turned on with the *Main Switch*. Hold the Wheelie upright, in the position in which you will drive it (base plate as horizontal as possible). The electronics will calibrate itself in this position and will attempt to maintain this position when you are driving the Wheelie.
Now place your right foot on the *Foot Switch* built into the

base plate. This switch lets the electronics know that you have stepped onto the Wheelie. You can now test the Wheelie by leaning forwards or backwards a little and check whether the Wheelie moves with it. The foot switch is part of the safety for the Wheelie, a little more about that later.
Now put your other foot on the ElektorWheelie platform. It is important that you stand fully upright and not bend your knees! It also helps if you look straight ahead, not down.
You accelerate by moving your weight towards your toes. Don't push against the control lever, that doesn't work well. To slow down you move your weight towards your heels. The electronics in the vehicle ensures that you will remain balanced and that you can remain standing on the balancing two-wheeler as demonstrated in the ElektorWheelie video on YouTube.
Steering is done with the control lever. Push it to the left slightly and the left motor will turn a little slower, so that the Wheelie will make a left turn. A small deflection to the right and the Wheelie will veer to the right.
When the rider is no longer standing on the base plate, the current to the motors is interrupted after a short delay (hysteresis), so that the Wheelie will not carry on and decide for itself where to go. As an additional safety feature the ElektorWheelie is fitted with an emergency stop (*Safety Switch*). This works as follows: A pin is attached with a cord to a strap around your arm. Should you fall unexpectedly then the pin will be pulled out by the cord, causing the motors and electronics to be switched off *instantly*.

Tips

Make the first trip with the assistance of someone else! All
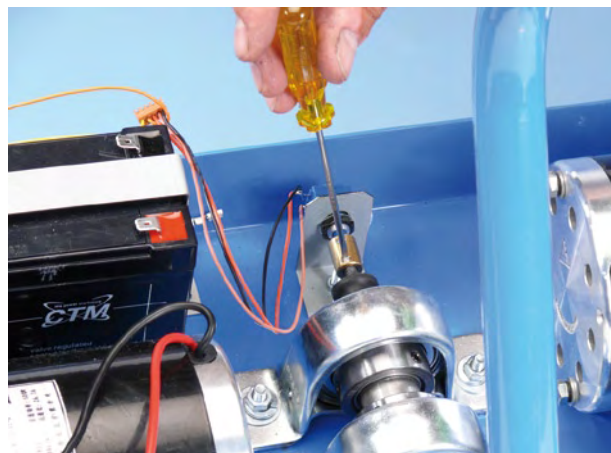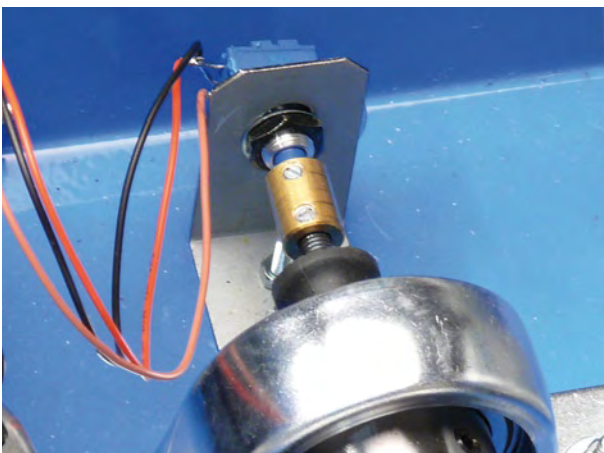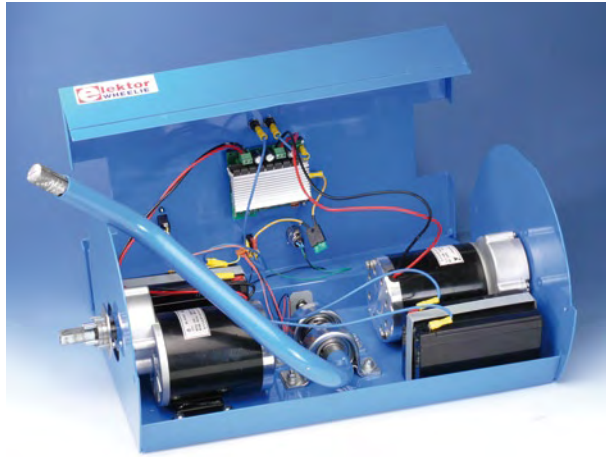
sorts of things can go wrong. Helmet, knee and elbow protection are certainly recommended initially.

The software controlling the ElektorWheelie vehicle is 'open'. That means that you can adapt and change the firmware to your heart's content, but at your own risk. However, the supplied software has passed thorough testing and has had extensive development and is therefore very suitable without any further changes.

You are also invited to visit the ElektorWheelie forum accessible via www.elektor-usa.com. There you can discuss and

gather ideas about the software and the hardware and, of course, share your experiences with other Elektor readers.

(090249-I)

Notice

This article provides a brief overview of the assembly of a pre-production sample. Each ElektorWheelie kit supplied to customers comes with an extensive construction manual. A pdf file of the manual will be available as a free download from www.elektor.com/wheelie in due course. Photos, videos and impressions may be found in the ElektorWheelie blog at http://ewheelie.blogspot.com.
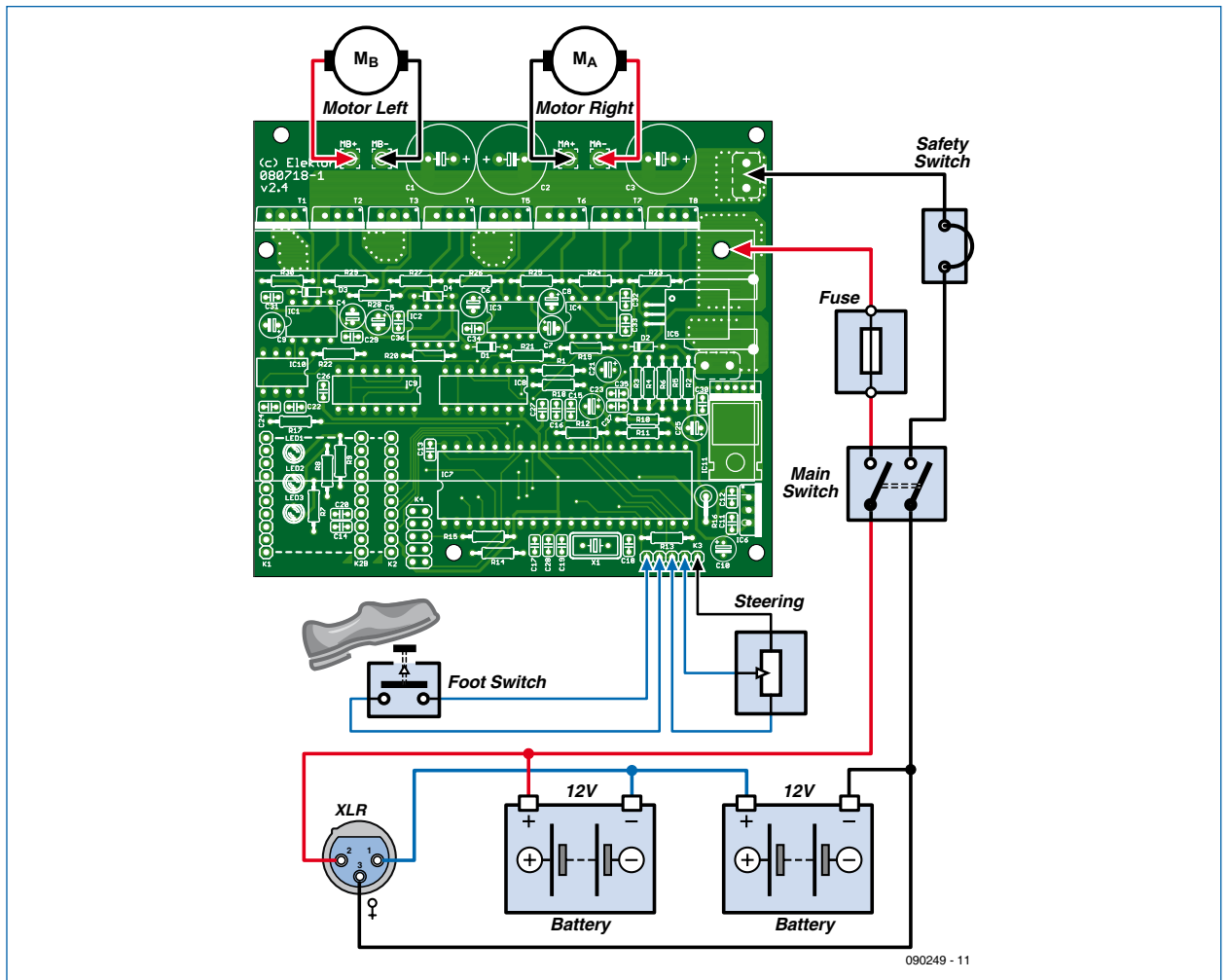
The wiring diagram shows how all the parts are to be connected together. Pay careful attention to the polarity of the batteries and the connections to the motors (do not swap left and right).

090249 - 11

GIGANTIC!

www.elektor-usa.com/subs · Phone 1-888-924-9465

elektor
electronics worldwide

Or use the subscription order form near the end of the magazine.

Power Charger for ElektorWheelie

Dual battery charger for lead acid gel batteries

Specifications

- Balanced operation; charges 2 batteries simultaneously
- Charging current: 0.8 A
- Estimated charge time for ElektorWheelie batteries: 8 to 10 hours (without deep discharge cycling)
- Input voltage: 120 VAC or 230 VAC (using the recommended transformer)
- Output voltage can be adapted for 6 V batteries

The self-balancing single-axle DIY electrovehicle uses sealed lead acid (SLA) batteries for power. You can of course use the supplied charger to charge them, but it can also be done differently (better). The charger described here is not only suitable for charging the batteries in the ElektorWheelie, but can also be used as a general charger for 12 or 24 volt batteries.

When charging several lead acid batteries at the same time, you should take into account any possible small differences between them. Two 12 V batteries are used in the ElektorWheelie. The charger described here can be used to fully charge these batteries simultaneously, but independently from each other (this is in contrast to the charger included in the kit). The batteries can also be charged faster with this charger.

The charger is based on a previously published design in Elektor, which is still perfectly suitable for the job. For this application it has been implemented twice, so that two batteries can be charged at the same time.

The design

As can be seen from **Figure 1**, the same circuit has been used twice, each built around an L200 voltage regulator IC.

The circuit is actually a two-stage charger for sealed lead acid (SLA) gel batteries, and includes temperature compensation. The latter is formed by the five diodes connected in series to the Ground connection of the L200 (D4–D8 and D15–D19). The voltage drop across these diodes causes the output voltage to rise by about 3.5 V, but this has nothing to do with their actual function. They've been added solely to compensate for the temperature coefficient of
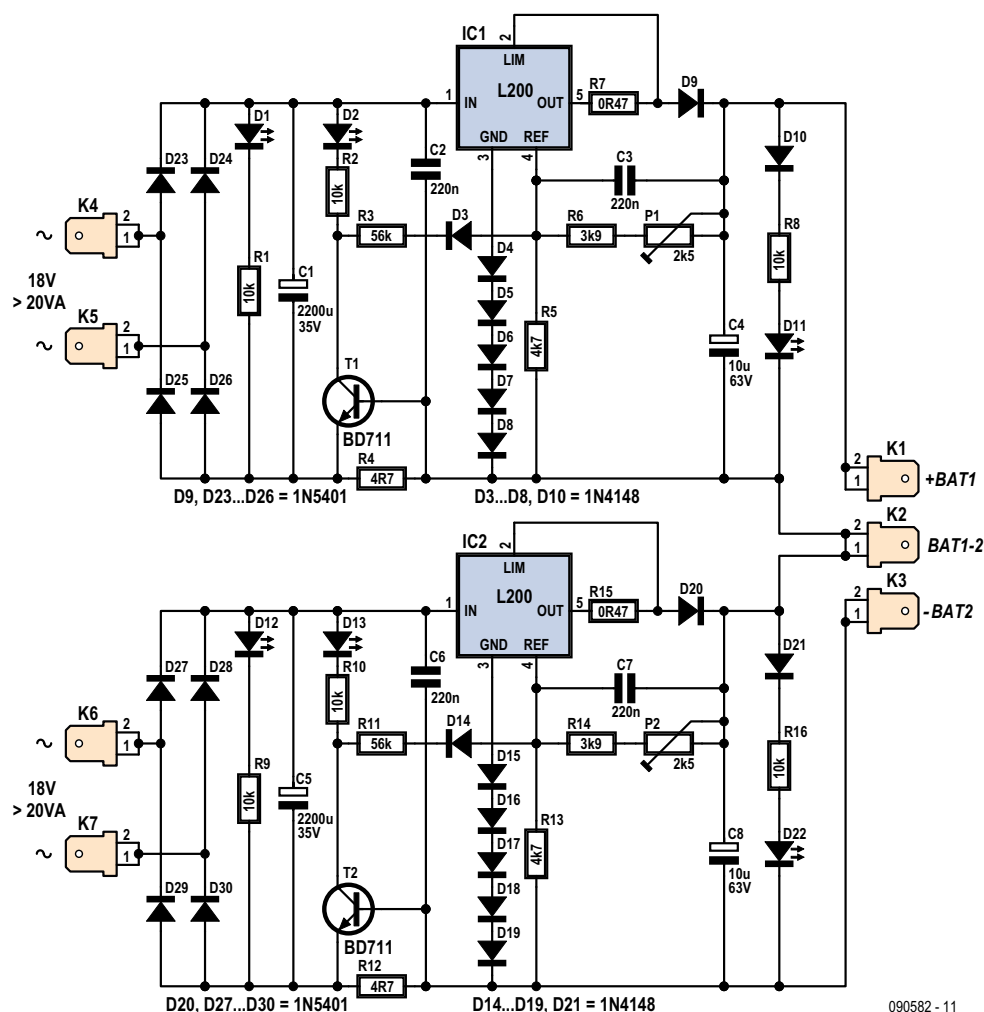
Figure 1. The dual nature of the design can be clearly seen in the circuit diagram for the charger.

the average gel battery (+3.85 mV/°C). The diodes cause the output voltage to vary by a similar amount in the opposite direction. The 'safe' operating range with the inclusion of these components is -20 °C to +40 °C. An additional advantage of these diodes is that they protect the voltage regulators against reversed polarity at the output.

The circuit operates as a two-stage charger and can either charge the battery normally or trickle-charge it. The fact that the charging current reduces when the battery becomes fully charged comes in handy. The L200 has been configured as a constant current source in the usual way, although an extra circuit has been added: Normally the output voltage of the regulator is determined by the relationship between the resistors in the voltage divider connected to pin 4 (R6+P1/R5, with reference to the top charger), which in this case results in an output voltage of about 13.8 V (considering only the 12 V ver-

sion). However, when a battery is connected to K1 (+) and K2 (–) and sufficient current flows through R4, transistor T1 starts to conduct and the R3/D3 branch is connected in parallel to R5. This changes the proportions of the potential divider such that the output voltage increases to about 14.4 V.

When a discharged battery is connected the initial output voltage is therefore 14.4 V, or 2.4 V per cell. Series resistor R7 ensures that the resulting charging current is limited to about 0.1 C (assuming that a battery with C = 9 Ah is used). R7 should therefore be adjusted for the maximum charging current of the battery concerned. It's possible to adjust the values of R6 and R3 such that the output voltage is halved, making the charger suitable for use with 6 V batteries. The appropriate values for these components may be found in the **table**.

The more a battery is charged up, the higher its cell voltage will be. As

a consequence, the charging current will gradually drop. However, as long as a sufficient current flows, the output voltage will remain at a steady 14.4 V (or 7.2 V for a 6 V version). The current through R4 will not rise above about 200 mA, even with a large charging current. The rest of the output current flows via the base-emitter junction of T1. The BD711 can cope with base currents up to 5 A without any problem. Once the charging current drops to a value of about 0.01 C it's safe to assume that the battery is fully charged. At that point the voltage drop across R4 will be too small for T1 to conduct, and R3 no longer has an effect on the potential divider connected to pin 4. The charger now switches over to an output voltage of 13.8 V (or 6.9 V for a 6 V version). This safe voltage results in a trickle-charge that keeps the battery in a good condition.

Since gel batteries can't cope well with a large voltage ripple, a fairly
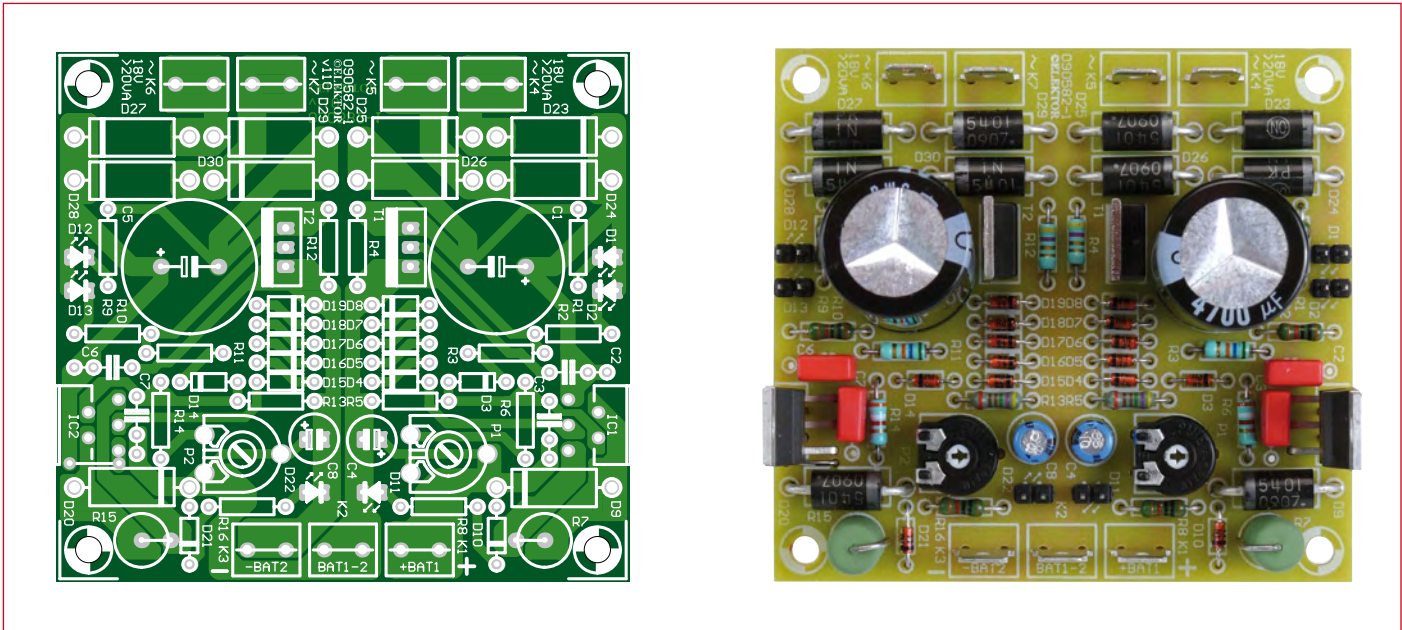
Figure 2. To keep the home construction of the charger as simple as possible we've used only trusty old through-hole components.

large smoothing capacitor (C1) has been added to the input. The smaller electrolytic capacitor at the output (C4) has the function of keeping the L200 stable under all conditions.

LED D1 indicates that an input voltage is present. D11 does the same for the output, but also lights up when a battery is connected. LED D2 is on while the charging voltage is 2.4 V per cell and turns off when the charger switches to trickle-charging at 2.3 V per cell. Diode D9 protects the L200 against reverse polarity and prevents the connected battery from discharging via IC1.

Input voltage

The minimum input voltage required by the circuit is determined by the maxi-mum output voltage, the voltage drop across D9 and R7, and the maximum dropout voltage of the voltage regulator. In the case of 12 V batteries this means that a minimum of 18 V is required and for 6 V batteries you need a minimum of 11 V. When selecting a transformer, do take account the ripple across C1 and the voltage drop across the bridge rec-tifier (D23–D26). The ripple across the smoothing capacitor at 1 A is about 4.5 V (as a rule of thumb: $I / (2fC)$) and the volt-age drop across the bridge rectifier is 1.2 V. This means that a transformer is required with a minimum peak voltage of 23.7 V (12 V version). This results in a choice of an 18 V transformer. It should be rated for at least 40 VA (for an output current of 1 A).

In the parts list we've chosen a 50 VA

type, but any comparable type can be used as well. This particular type has dual primary windings. The two pri-mary windings have to be connected in series for operation at 230 VAC. The grey and purple leads have to be joined, with the connection thoroughly isolated. For use with 115-120 VAC (60 Hz) mains voltage, the two wind-ings have to be connected in parallel (connect blue to purple and grey to brown).

A 35 V type is required for C1 because the voltage can rise above 25 V when no battery is connected.

Construction

A compact PCB has been designed for the charger. The component layout is

COMPONENT LIST

Resistors
R1,R2,R8,R9,R10,R16 = 10kΩ
R3,R11 = 56kΩ
R4,R12 = 4.7Ω
R5,R13 = 4.7kΩ
R6,R14 = 3.9kΩ
R7,R15 = 0.47Ω
P1,P2 = 2.5kΩ (2.2kΩ)

Capacitors
C1,C5 = 2200µF 35V radial, diam. max. 16 mm, lead pitch 7.5mm (0.3"), e.g. Nich-icon UPS1V222MHD (Farnell # 8813027)
C2,C3,C6,C7 = 220nF, lead pitch 5mm (0.2") or 7.5mm (0.3")
C4,C8 = 10µF 63V radial, diam. max.

6.3mm (0.25")

Semiconductors
D1,D11,D12,D22 = low current LED 3 mm, green
D2,D13 = low current LED 3 mm, red
D3-D8,D10,D14-D19,D21 = 1N4148
D9,D20,D23-D30 = 1N5401
T1,T2 = BD711
IC1,IC2 = L200CV

Miscellaneous
K1-K7 = spade plug (male), 2-way, PCB mount
PCB, order # 090582-1, see [1]

Not on board
Mains transformator 2x18 V sec, 2x115 V pri-mary. 50 VA, diam. 81 mm (e.g. # Farnell

9530380)
IEC switched /fused AC power inlet, Bul-gin type BVA01/Z0000/01 (e.g. Farnell # 3136152)
Fuse 315 mA slow (630 mA slow for 120 VAC)
XLR plug (male) for cable, Neutrik type NC-3MXX-BAG (e.g. Farnell # 9991808)
Cable, 1m, 3-way, 1.5 mm² , outside diam. 6.7 mm, Lapp type 1119303 (e.g. # Farnell 1491485)
Strain relief, Lapp Kabel type 53111600 (bijv. Farnell 1286049)
Locknut M12 for strain relief, Lapp Kabel type 53119000 (e.g. Farnell # 1178909)
Enclosure, Boss Enclosures type BIM 2006/16/GY (e.g. Farnell # 1171686)
Aluminium sheet, thickness 3mm or 4mm, size 178x178mm
12 spade terminals 6.3x0.8 mm (female)

shown in **Figure 2**. The PCB artwork (copper track layout) can be downloaded from the Elektor website [1] as usual. The construction should be fairly straightforward. Start with the smallest components, such as the resistors and finish with the largest. Remember to put in the wire link next to D20.

The prototype was mounted in a U-shaped tray made from 2 mm thick aluminium. The heat conduction of this wasn't really sufficient, so for the final version we recommend a thickness of 3 or 4 mm. This tray functions as a heatsink for IC1 and IC2. The ICs should be electrically isolated from the tray, so use a mica sheet and a plastic sleeve. We used two ceramic isolators (1.5 mm thick), because this was more convenient to use after the aluminium was bent. The tray and the 50 VA transformer are then mounted in a case

Component values			
component	6 V battery	12 V battery	notes
R6,R14	wire link	3.9k	
R3,R11	6.8k	56k	
R7,R15	0.45 / I_{max}	0.45 / I_{max}	
R4,R12	0.5(V) : 0.01(C)	0.5(V): 0.01 (C)	Choose nearest E12 value
C = nominal capacity of the battery (in Ah) I_{max} = maximum (initial) charge current			

A suitable hole has to be made for the AC power input (including a properly rated fuse and switch). In the parts list is a type that comes with spade terminals. This makes it easier if the circuit has to be repaired, but takes up a bit more space than a separate switch, fuseholder and an AC power cord with a strain relief. The charging cable has been fed into the case via a strain

Alignment

At room temperature (20 to 25 °C), using P1, adjust the output voltage of the circuit to 13.8 V (6.9 V for the 6 V version). Next, check that the output voltage increases to 14.4 V (7.2 V for the 6 V version) when you momentarily short the collector/emitter junction of T1. Everything is fine if the measurement is within 0.1 V of the required
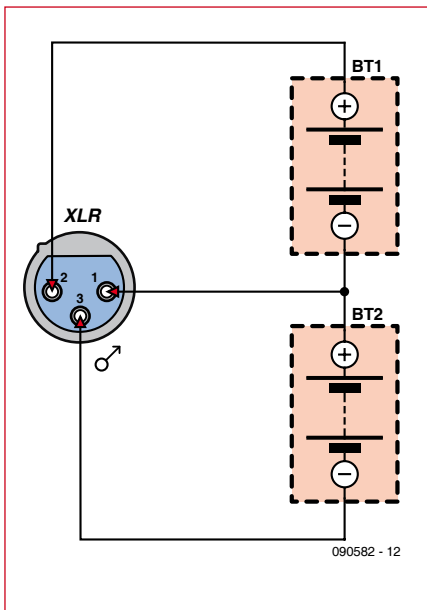
Figure 3. This is how the charging cable should be connected for use with the ElektorWheelie.

measuring 60 × 190 × 110 mm. The tray is mounted using four bolts with four nuts used as spacers. This prevents the warm part of the tray (where the voltage regulators are mounted) from making contact with the plastic case. The transformer is mounted on a separate sheet of aluminium. For this the same four bolts are used. In this way the tray can be larger (i.e. the 'U' a bit taller). The large head of the transformer bolt no longer has to be between the tray and the case, but is now between the extra sheet and the U shape.

relief and connected to the board using spade terminals. When the charger is used with the ElektorWheelie, the XLR plug should be connected as shown in **Figure 3**. The transformer connections are also made with spade terminals. The LEDs are mounted in the lid. The cables are long enough to allow the lid to be put down next to the case. On the board we mounted 2-way headers instead of the LEDs. Suitable header sockets were connected to the LED cables, which makes it easier to work on the circuit.

Don't forget to add sufficient holes to the case to assist ventilation.

value. If the voltage is too high you should increase the value of R3. Conversely, if the voltage is too low, R3 should be decreased.

When using the charger, always carry out these steps in the following order:
- Check the polarity of the battery.
- Connect the battery to the charger.
- Check that D11 is lit.
- Apply AC power.

(090582-I)

Internet Link

[1] www.elektor-usa.com/090582

Quantum die

By Markus Bindhammer (Germany)

To paraphrase Albert Einstein, 'God does not play dice': the famous physicist was not happy to accept that the world is ultimately governed by chance. A radioactive nucleus does not decay at precise, regular intervals; instead, there is a certain probability that it will decay at some point in the next hour or year. These days the nucleus is viewed as being in a superposition of states, 'decayed' and 'not decayed', and the uncertainty as to its state is only resolved when it is observed. This idea is so alien to everyday human experience that, even now that the consumer market starts to see devices that rely on the principles of quantum mechanics for their operation, it is easy to sympathise with Einstein's point of view.

The circuit here demonstrates that God does indeed play dice! Rather than using a single radioactive nucleus we use a small quantity of radioactive material, for example in the form of a sample of pitchblende, an ore of uranium. It is possible to purchase small specimens of pitchblende from suppliers to collectors of rocks and minerals. Care is needed in handling the material (see below).

We can use a Geiger counter to count the number of decay events that occur in a given period of time. This count is perfectly random and therefore ideal for driving electronic dice. In our circuit we use a ZP 1320 Geiger-Müller tube, which is available for purchase from various Internet sites. This tube is designed for a supply voltage of between 540 V and 650 V; our circuit can deliver a voltage from around 300 V to around 1000 V, which also allows other tube types to be used instead. The tube is connected to the circuit using a screened cable and must be mounted in a robust and electrically insulated enclosure.

To generate the high voltage we use a 230 V-to-6 V transformer wired 'back to front'. A MOSFET driven from IC2 via R2, pulls one side of the 6 V winding to ground. The resulting voltage spikes on the primary side are multiplied up using a diode-capacitor ladder to produce over 1000 V DC. R3 to R6 and P1 form a potential divider that couples the high voltage to the base of T2. If the high voltage exceeds the level set by P1, T2 turns on and blocks T1. The high voltage then falls until the feedback voltage is once again below the transistor's threshold.

The high voltage is taken to the tube's anode via protection resistor R7. The tube is filled with a noble gas; if ionizing radiation is incident on the tube, electrons are knocked out of their shells in the noble gas atoms. These free electrons accelerate towards the anode, ionizing further atoms as they go. The avalanche of ionization creates a brief current flow between anode

060309 - 11

Caution: It is essential to be careful when handling radioactive materials such as pitchblende. Keep a distance, wash your hands after touching the sample, and do not eat, drink or smoke while handling it. Do not under any circumstances cut or work the mineral sample.
Caution: High voltages are present in this circuit, which is recommended only for experienced constructors. The Geiger-Müller tube and the circuit must be mounted in a robust and electrically insulated enclosure.

and cathode. This current results in a voltage pulse across bleeder resistor R8, triggering IC1. The pulses are output by IC1 on pins 7 and 3. T3 turns on briefly with each pulse, causing the piezo sounder to emit a loud click.

The pulses that are output on pin 3 are passed via transistor T4. This is turned on by the output of IC5, which is configured as a monostable multivibrator. When button S1 is pressed the output of IC5 goes high and T4 conducts. This gates pulses through to the 7490 counter. After a preset time interval the output of IC5 returns low and T4 blocks. The time period over which the radioactive decay pulses are measured is determined by C14, R19 and P2:

$$t \approx 1.1 \times C14 \times (R19 + P2)$$

where t is in units of seconds, C14 in µF, and R19 and P2 in MΩ.

The 7490 counter is wired so that it is reset to 9 (1001) rather than to 0 (0000). It therefore counts from 1001 up to 0100. IC3 contains six inverters and is used to drive the seven LEDs that form the display of the die. The inputs of the two unused inverters are tied to the positive supply to prevent unpredictable behavior.

The quantum die is perhaps not ideal for ludo-type games if your fellow players are accustomed to blaming their defeats on the dice being loaded!

(060309-I)

Working for Morons

By Gerard Fonte (USA)

It is unfortunate but true that some time during an engineer's career he will find himself working for a moron. This can be an extremely difficult and frustrating situation. And in today's job market, it's more important than ever to excel at work. Note that I use the male gender for brevity and simplicity. Most engineers are male but I recognize that there are many women engineers, too. So, please excuse my chauvinism.

Dilbert's pointy-haired boss is an extreme example. This is a manager with no real feel for any of the technical aspects of your work. Nevertheless he is required to direct and evaluate you. Note the term 'required'. That's his job. It's also useful to remember that he is graded by his ability to direct you and the rest of the technical people in the department. After all, he's the decider. Therefore, it's important to know how he makes his technical decisions. If you can influence his decision making, you can have a better professional relationship and better enjoy your job as well.

There are generally three ways for non-technical managers to make technical decisions: 1) a command from a higher-up, 2) discussion with engineers, 3) guess.

A keyboard for morons.

Simon Sez

Unfortunately, the 'voice from above' is probably the most difficult situation for the working engineer. There is little opportunity to express any suggestions and there is little chance of anyone listening. Some VIP has made a choice and now that has to be realized. Worse, your boss doesn't have the means to evaluate the idea. He assumes that the request is sensible. If it's not, you're the scapegoat when it fails.

In the late 1970's I was told to design a system that would identify an aircraft's position, by video camera, while flying 50 feet over the runway (for use in certifying commercial airport instrument accuracy). The idea was to use the runway markings and determine mathematically the exact height, heading and attitude of the plane. We had a computer running at a few megahertz and maybe 65 K of memory, no real operating system, no optical recognition software and the image sensor was 512 × 512 pixels. We needed 1-inch accuracy. I knew from the start that this was an impossible task.

So, I quietly did an error analysis. It showed that the best accuracy possible was eight to twelve inches, or about an order of magnitude off. I gave this to my boss and asked him how to proceed. Naturally, he simply passed my analysis up the ladder—he certainly couldn't make such a decision.

I refused to let the matter rest because if I didn't, I was going to have to sign off on a design that I knew couldn't meet the specs. After a few more meetings and analyses, they had someone else run the numbers. That showed a worst-case error of 3 feet (although about half could be calibrated out). I provided an alternate design that was accurate to 1/10th of an inch and much simpler and cheaper but didn't use a camera. I was told

that if it didn't use a camera, the design wasn't acceptable. So, I again asked how we could possibly meet the specifications. My boss, now annoyed and frustrated with me, answered quite candidly. *"We'll cheat. Everyone else does."* I left that company within a few weeks.

This was obviously an extreme situation. But, it is not unusual to be asked to compromise your values or ethics because your boss doesn't know any better. Don't take it personally. Don't get angry. If you feel you must act, calmly explain your position. If it's important or a safety issue, try to get something in writing to protect yourself. That's probably the most you can do.

Hearing Voices

This is your best situation. If your boss makes decisions after discussing them then you need to be one of those he listens to. Find out who he talks to and why. Getting in may be as easy as asking to be included. If you can write well (clearly and at the proper technical level), providing him with pertinent information on a regular basis may be a way to catch his ear. Always remember that he is probably very aware that he does not have the engineering experience or knowledge that you do. He may be intimidated by intelligent and aggressive people (or in his view, a smart-ass). Always be respectful and never talk down to him. He will only respect you and your ideas if you clearly demonstrate that you respect him.

Magic 8-Ball Management Style

If your boss's decisions appear completely random, he may be guessing at the right answer. Usually this is an indication of low self-esteem. He is unwilling to admit his ignorance for fear of appearing weak. Such people generally avoid anything they perceive as a risk. They are terrified of failure. So, any new idea or procedure may be rejected because it's too risky. It's usually pretty clear if this is the case.

Alternatively, there may be factors that you are not aware of. It is possible that he is making a proper choice—perhaps for financial or marketing considerations. He may have more information than you have. He may not be the moron, you might be. Don't be so conceited to think you know everything. Your boss sees things differently. A good decision from your point of view may be a poor one from his.

With either scenario, it's important to communicate with your boss. Ask him why he made his decision. In either case you will learn something. If he is timid, carefully try to show that you can help him succeed. Don't push. This will take considerable time. Point out that technical progress in his department will make him look good to his boss. Conversely, you might learn something about marketing, finance or business. Perhaps another department just developed a special widget that only works with a particular design—the one you don't like. Just because your boss acts like a moron doesn't mean that you have to act like one too.

(090644-I)

Hexadoku

Puzzle with an electronics touch

Hopefully you've recovered from the awesome Hexamourai puzzle featured in the July/August 2009 issue. This month it's back to normal with a regular Hexadoku to contend with. As always, all correct solutions we receive in our international mailboxes enter a prize draw for an E-blocks Starter Kit Professional and three Elektor Shop vouchers.

The instructions for this puzzle are straightforward.

In the diagram composed of 16 × 16 boxes, enter numbers such that **all** hexadecimal numbers 0 through F (that's 0-9 and A-F) occur once only in each row, once in each column and in each of the 4x4 boxes (marked by the thicker black lines). A

number of clues are given in the puzzle and these determine the start situation.
All correct entries received for each month's puzzle go into a draw for a main prize and three lesser prizes. All you need to do is send us the numbers in the grey boxes. The puzzle is also available as a **free download** from the Elektor USA website.

SOLVE HEXADOKU AND WIN!

Correct solutions received from the entire Elektor readership automatically enter a prize draw for an

E-blocks Starter Kit Professional

worth **$425**

and three **Elektor SHOP Vouchers** worth $55.00 each.

We believe these prizes should encourage
all our readers to participate!

The competition is not open to employees of Elektor International Media, its business partners and/or associated publishing houses.

PARTICIPATE!

Please send your solution (the numbers in the grey boxes) by email to:
hexadoku@elektor-usa.com
Subject: hexadoku 09-2009.
Include with your solution: **full name and street address.**
Alternatively, by fax or post to: **Elektor Hexadoku**
PO Box 876 - Peterborough NH 03458-0876 - USA
Fax: 603-924-9467
The closing date is **1 October 2009.**

PRIZE WINNERS

The solution of the June 2009 Hexadoku is: **579BD**.
The **E-blocks Starter Kit Professional** goes to:
Michael Andersen (Denmark).

An Elektor SHOP voucher worth $55.00 goes to:
Silvia Rotondi (Italy); Haije Talsma (The Netherlands); Roland Louis (France).
Congratulations everybody!

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

(c) PZZL.com

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

Leak coaxial trough-line VHF FM stereo tuner (1962)

By Martin Cooke (United Kingdom)

Harold Leak was something of a pioneer in the design of high-fidelity sound reproduction equipment. From the 1950s to the end of the 60s his company produced a whole range of hi-fi equipment. The unit in this month's Retronics spotlight is a 'trough-line' FM tuner in original condition which still works 47 years after it left the assembly line in 1962.

The front panel is a thick clear plastic material with a maroon and gold design silk-screened onto the reverse. This has proved to be quite durable, later trough-lines were fitted into an aluminum case and it is more difficult to find a pristine example. Two knobs on the front panel control volume and tuning while slide switches select Automatic Frequency Control (AFC) and RF sensitivity. Just to the left of the tuning scale is a 'magic eye' tuning indicator showing two green luminous bars which converge as the signal strength increases. Once tuned the AFC can be used to turn the eye off and keep the set on station. At the rear are connections for the FM aerial, a phono socket for the multiplex signal output, AC power voltage selector and a length of screened cable carrying the output audio.

Once the lid is lifted on this tuner it does not take long to spot the odd man out. Amongst the conventional array of tubes and a triple-ganged tuning capacitor there is a length of 1.3 inch diameter pipe which would look more at home in a plumbers tool bag. This is the coaxial trough-line which gives the tuner its name. At the time this set was designed nation-wide FM broadcasting was in its infancy and the original Leak trough-line was amongst the first FM tuners on the market. Receiver designs were quite rudimentary, it was normal for the local oscillator (LO) to comprise of an oscillator with a LC tuned circuit where the C was one gang of the tuning
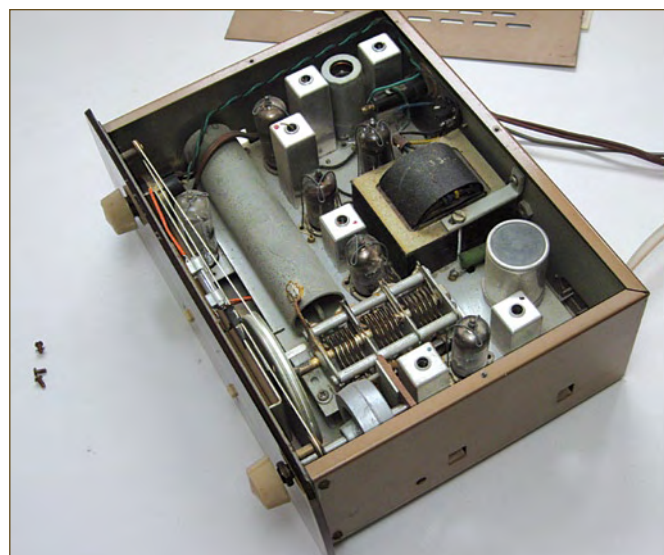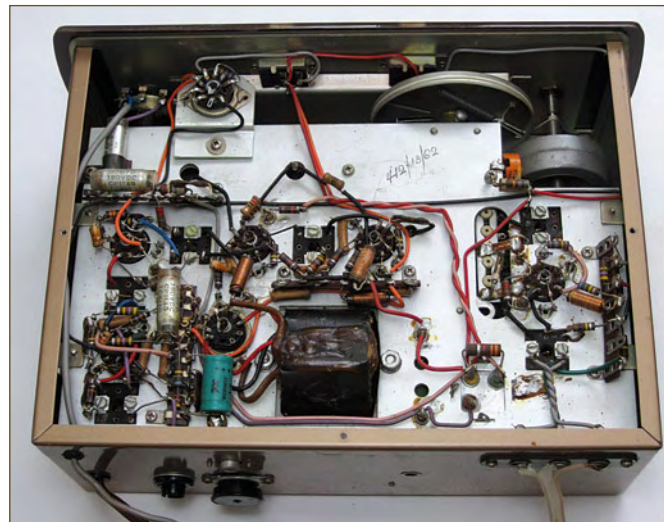
capacitor. LO drift was a problem especially as the tubes warmed everything up. To reduce drift the designers at Leak replaced the LO coil with a tapped transmission line resonator. Any innovation that helped lessen drift was a good selling point and the trough-line boasted a drift of 15 kHz or 3 kHz with AFC.

The first Leak trough-lines enclosed the center conductor on three sides (i.e. in a trough) but the resonant element used here and on later versions can be best described as a tapped coaxial stub, at 5.8 inches it is too short to be quarter wave. The taps are at one and two thirds with the open end connected in parallel with one gang of the tuning capacitor. The LO runs at 12.5 MHz above the received station frequency instead of 10.7 MHz. A Foster-Seeley discriminator recovers the baseband which supplies the multiplex O/P, the (mono) audio signal and (after filtering) the AFC feedback to help the LO maintain a constant IF. Later versions of the tuner contained a (poor) 3-transistor stereo decoder.

Harold Leak retired at the end of the 1960s and sold the company to the Rank Organisation, the name has since disappeared from hi-fi showroom shelves. The mono tuner's sound has always been well received by audio reviewers and there are now better decoder designs which improve stereo performance. There has also been something of a resurgence of interest in classic tuner designs recently. It may be that enough time has elapsed for their styling to be fashionable again or it could be seen as a reaction to the disappointing quality of many DAB stations. Perhaps it's not surprising that modern digital receivers, with all their frantic number crunching and signal processing still cannot improve on the sweet sound produced by a simple FM analog tuner designed half a century ago.

(090267-I)

---

products and services directory

Going Strong

A world of electronics from a single shop!
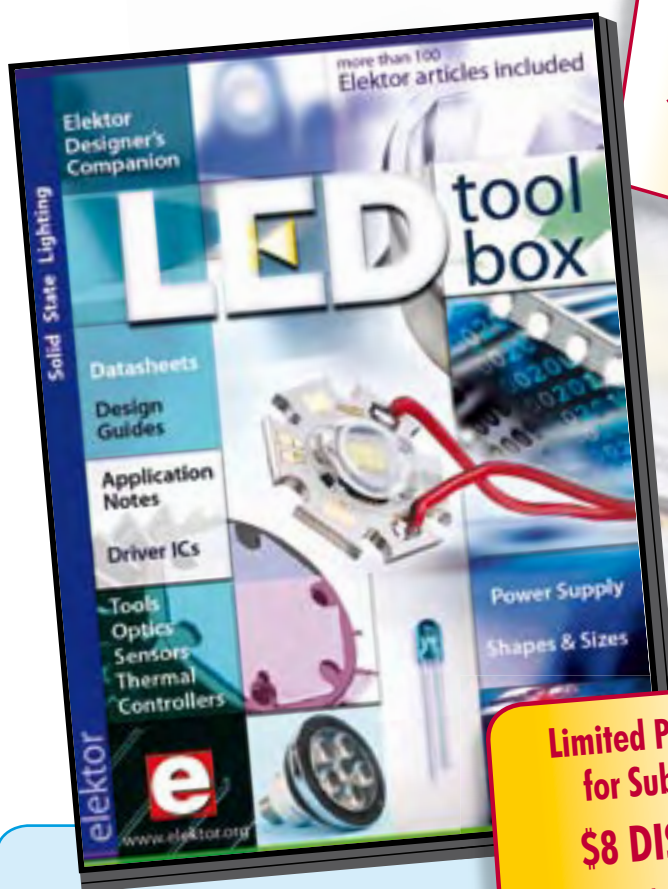
NEW!

Limited Period Offer for Subscribers!
$8 DISCOUNT
www.elektor-usa.com/led

See the light on Solid State Lighting!

DVD LED Toolbox

This DVD-ROM contains carefully-sorted comprehensive technical documentation (optical properties, electrical characteristics, mounting, life expectancy, etc.) about and around LEDs. For standard models (through-hole, SMD), and for a selection of LED modules (ribbons, light bars, bargraphs, and other LED clusters), this Toolbox gathers together data sheets from all the manufacturers, application notes, design guides, white papers and so on. LED Toolbox also offers several hundred drivers for powering and controlling LEDs in different configurations (buck, boost, charge pump, constant current, and so on), along with ready-to-use modules (power supply units, DMX controllers, dimmers, etc.). In addition to optical systems, light detectors, hardware, etc., this DVD also addresses the main shortcoming of power LEDs: heating. As a bonus, this DVD contains more than 100 Elektor articles on the subject of LEDs! All of the documents are PDF files. Browsing around the DVD is made easy by an HTML menu.

ISBN 978-90-5381-245-7 • $54.00

Incl. searchable i-TRIXX archive

DVD i-TRIXX
Freeware Collection 2009

This DVD contains 100 nifty freeware applications, tools and utilities for the Windows PC. And as a free extra, it contains the full and searchable (!) i-TRIXX archive, with all the editions up until week 8 of 2009 from i-TRIXX, the e-magazine published by Elektor. Do you feel the need for a decent and reliable antivirus program? A bandwidth monitor which keeps track of your current up and download rate? An application for recording, editing and converting video? Anonymous surfing from any internet access point from a USB stick? Checking, optimizing and cleaning up your computer? You can expect that and much more in the i-TRIXX Freeware Collection 2009.

ISBN 978-90-5381-244-0 • $39.50

All articles published in 2008

DVD Elektor 2008

This DVD-ROM contains all editorial articles published in Volume 2008 of the English, Spanish, Dutch, French and German editions of Elektor magazine. Using Adobe Reader, articles are presented in the same layout as originally found in the magazine. The DVD is packed with features including a powerful search engine and the possibility to edit PCB layouts with a graphics program, or printing hard copy at printer resolution. Thanks to the use of an Internet browser program, the requirements for the PC are minimal.

ISBN 978-90-5381-235-8 • $35.00

Prices and item descriptions subject to change. E. & O.E

Books

Bring your microcontroller to life

Artificial Intelligence

This book contains 23 special and exciting artificial intelligence machine-learning projects, for microcontroller and PC. Learn how to set up a neural network in a microcontroller, and how to make the network self-learning. Or discover how you can breed robots, and how changing a fitness function results in a totally different behavior. Several artificial intelligence techniques are discussed: expert system, neural network, subsumption, emerging behavior, genetic algorithm, cellular automata. roulette brains etc.

256 pages • ISBN 978-0-905705-77-4 • $46.00
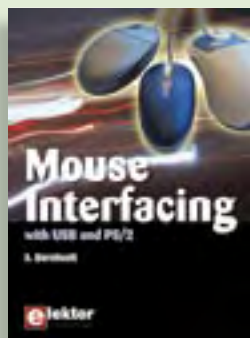
Connect your mouse into new embedded applications

Mouse Interfacing

This book describes in-depth how to connect the mouse into new embedded applications. It details the two main interface methods, PS/2 and USB, and offers applications guidance with hardware and software examples plus tips on interfacing the mouse to typical microcontrollers. A wide range of topics is explored, including USB descriptors, a four-channel, millivolt-precision voltage reference all with fully documented source-code. A wealth of clear illustrations enhance this highly readable text.

256 pages • ISBN 978-0-905705-74-3 • $42.00

Bestseller!

Experimenting with the MSP430

All the big electronics manufacturers supply microcontrollers offering a wide range of functions. Texas Instruments supplies handy USB evaluation sticks with related software for its low-cost MSP430 controllers. Unfortunately the I/O facilities are somewhat limited. These can be substantially enhanced with the help of the Elektor MSP430 board.

PCB, populated and tested

Art.# 080558-91 • $55.00

TI eZ430-F2013 Evaluation Kit

Art.# 080558-92 • $35.00

Automotive CAN controller

Since cars contain an ever increasing amount of electronics, students learning about motor vehicle technology also need to know more about electronics and microcontrollers. In collaboration with the Timloto o.s. Foundation in the Netherlands, Elektor designed a special controller PCB, which will be used in schools in several countries for teaching students about automotive technologies. But it can also be used for other applications, of course. The heart of this board is an Atmel AT90CAN32 with a fast RISC core.

Kit of parts, incl. PCB with SMDs prefitted

Art. # 080671-91 • $79.00

Software Defined Radio

SD radio receivers use a bare minimum of hardware, relying instead on their software capabilities. The Elektor SDR project (by Burkhard Kainka) demonstrates what's achievable, in this case a multi-purpose receiver covering all bands from 150 kHz to 30 MHz. It's been optimised for receiving DRM and AM broadcasts but is also suitable for listening in to the world of amateur transmissions. The designer's aim for this project was to create a receiver displaying high linearity and phase accuracy. Development was focussed on the characteristics that were most important for a top-notch DRM receiver and the end result is a receiver with remarkable interference rejection characteristics!

Ready-populated and tested board

Art. # 070039-91 • $145.00

Stand Alone OBD-2 Analyser

This handy analyser makes a simple job of rummaging through the information stored by the client-accessible part of your car's computer. It works with all current OBD-2 protocols and can read and erase trouble codes stored in the vehicle and reset the MIL display. All this without the help of a PC or a visit to a service station.

PCB, ready-assembled and tested (case and OBD-2 cable included)

Art. # 070038-93 • $119.95

Kits & Modules

Bestsellers

Books

1 **310 Circuits**
ISBN 978-0-905705-78-1 .... $45.00

2 **C# 2008 and .NET programming**
ISBN 978-0-905705-81-1 .... $44.50

3 **C Programming** for Embedded Microcontrollers
ISBN 978-0-905705-80-4 .... $52.00

4 **Artificial Intelligence**
ISBN 978-0-905705-77-4 .... $46.00

5 **PIC Microcontrollers**
ISBN 978-0-905705-70-5 .... $52.00

CD/DVD-ROMs

1 **DVD Elektor 1990 through 1999**
ISBN 978-0-905705-76-7 .... $99.00

2 **ECD 5**
ISBN 978-90-5381-159-7 .... $38.00

3 **DVD Elektor 2008**
ISBN 978-90-5381-235-8 .... $35.00

4 **DVD i-TRIXX** Freeware Collection 2009
ISBN 978-90-5381-244-0 .... $39.50

5 **FPGA Course**
ISBN 978-90-5381-225-9 .... $29.00

Kits & Modules

1 **MSP430**: PCB, populated and tested
Art. # 080558-91 .... $55.00

2 **MSP430**: TI eZ430-F2013 Evaluation Kit
Art. # 080558-92 .... $35.00

3 **Software Defined Radio**
Art. # 070039-91 .... $145.00

4 **DigiButler**
Art. # 071102-71 .... $52.50

5 **Automotive CAN controller**
Art. # 080671-91 .... $79.00

Order quickly and securely through
www.elektor-usa.com/shop
or use the Order Form near the end
of the magazine!

elektor SHOP

Elektor US
PO Box 876
Peterborough NH 03458-0876
USA
Toll-free: 888-924-9465
Phone: 603-924-9464
Fax: 603-924-9467
E-mail: custservus@elektor.com

Hearing Threshold Tester

Test your hearing abilities with this simple instrument. The circuit makes use of a PC (or a laptop) with a soundcard and specially developed software to record a person's hearing response. The measurement consists of a series of tone bursts with different levels and frequencies. Using a console the person tested indicates if he/she has heard the tone or not, and if a repeat is required. A pair of high quality headphones are an obvious requirement for repeatable results. The hardware is based on an MC9S08QG8 microcontroller from Freescale.
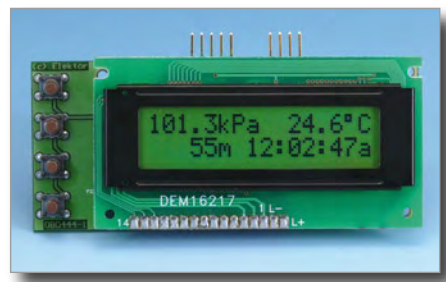
Digital Barometric Altimeter

In this project a piezoresistive transducer measures the atmospheric pressure, enabling a PIC18F2423 microcontroller to use the ISA model to tell you, on an LCD, your altitude above mean sea level (MSL). The barometer has a range of 15 kPa to 115 kPa and the altitude meter goes up to 11,000 m with a resolution of about 3 m. Temperature compensation and a real time clock are included in the project.

Audio Preamp with ATmega8

A microcontroller is perfect for controlling a preamplifier equipped with an electronic potentiometer and a display. This project uses an electronic pot type AD590 from Analog Devices, connected to an ATMega8 micro via a 3-wire bus link. The preamp has four stereo inputs selectable by way of relays, with a two-line LCD showing all settings and options for control. One setup allows each input source to be assigned a name from a predefined list. The preamp is ready for extending with an external DAC.

Article titles and magazine contents subject to change, please check 'Magazine' on www.elektor-usa.com for updates and announcements.

The October 2009 edition of Elektor is published on **Monday, September 21, 2009**. Elektor is available on subscription or from selected Borders and Barnes & Noble bookshops.

w.elektor-usa.com www.elektor-usa.com www.elektor-usa.com www.elektor-usa.co

Elektor on the web

All magazine articles back to volume 2000 are available online in pdf format. The article summary and parts list (if applicable) can be instantly viewed to help you positively identify an article. Article related items are also shown, including software downloads, circuit boards, programmed ICs and corrections and updates if applicable. Complete magazine issues may also be downloaded.

In the Elektor Shop you'll find all other products sold by the publishers, like CD-ROMs, kits and books. A powerful search function allows you to search for items and references across the entire website.

Also on the Elektor website:
- Electronics news and Elektor announcements
- Readers Forum
- PCB, software and e-magazine downloads
- Surveys and polls
- FAQ, Author Guidelines and Contact

Description	Price each	Qty.	Total	Order Code
310 Circuits NEW	**$45.00**			
C#2008 and .NET programming for Electronic Engineers NEW	**$44.50**			
DVD LED Toolbox NEW	**$54.00**			
DVD Elektor 1990 through 1999 NEW	**$99.00**			
C Programming for Embedded Microcontrollers	**$52.00**			
DVD i-TRIXX Freeware Collection 2009	**$39.50**			
Artificial Intelligence	**$46.00**			
	Sub-total			
	Shipping & Handling **$** *(Airmail from Europe)*		**20.00**	
	Total paid			

Prices and item descriptions subject to change. The publishers reserve the right to change prices without prior notification. Prices and item descriptions shown here supersede those in previous issues. E. & O.E.

_____
Name/Company

_____
Street & No.

_____

_____
City/State/Zip   Country

_____
Tel   E-mail

EL09   Date – –   Signature _____

METHOD OF PAYMENT
Remit in US $ drawn on a US bank. Please see next page for ordering information.

☐ **Check or Money Order**
☐ **Mastercard**
☐ **VISA**
☐ **Discover**
☐ **American Express**

Expiry date: _____

Verification code: _____

Send order form to:

Elektor US

PO Box 876

Peterborough NH 03458-0876

USA

Toll-free: 888-924-9465
Phone: 603-924-9464
Fax: 603-924-9467
E-mail: custservus@elektor.com
www.elektor-usa.com

![elektor electronics worldwide]

Yes, I want to subscribe to Elektor US for 1 year*

I would like:

☐ **Standard Subscription for $39.95 (11 issues)**

☐ **Plus Subscription for $54.95**
(11 issues plus the Elektor Volume 2009 DVD-ROM)
As a special limited period offer I also receive the Volume 2008 on DVD-ROM. Free of charge!

* Offer available in US & Canada only. Canada please add $11.00 per year for postage.

_____
Name/Company

_____
Street & No.

_____

_____
City/State/Zip   Country

_____
Tel   E-mail

EL09   Date – –   Signature _____

METHOD OF PAYMENT
Remit in US $ drawn on a US bank. Please see next page for ordering information.

☐ **Check or Money Order**
☐ **Mastercard**
☐ **VISA**
☐ **Discover**
☐ **American Express**

Expiry date: _____

Verification code: _____

Send order form to:

Elektor US

PO Box 876

Peterborough NH 03458-0876

USA

Toll-free: 888-924-9465
Phone: 603-924-9464
Fax: 603-924-9467
E-mail: custservus@elektor.com
www.elektor-usa.com

ORDERING INFORMATION

To order contact customer service:

Toll-free: 1-888-924-9465
On-line at www.elektor-usa.com
Fax: 603-924-9467
Mail: Elektor US
            PO Box 876
            Peterborough NH 03458-0876 USA
E-mail: custservus@elektor.com

Customer service hours: 8:00 AM–4:30 PM Monday–Thursday. Voice mail available at other times.
When leaving a message please be sure to leave a daytime telephone number where we can return your call.

PLEASE NOTE: While we strive to provide the best possible information in this issue, pricing and availability are subject to change without notice. To find out about current pricing and stock, please call or email customer service.

COMPONENTS

Components for projects appearing in Elektor are usually available from certain advertisers in the magazine. If difficulties in obtaining components are suspected, a source will normally be identified in the article. Please note, however, that the source(s) given is (are) not exclusive.

PAYMENT

Orders must be prepaid. We accept checks or money orders (in US $ drawn on a US bank only), VISA, Mastercard, Discover, and American Express credit cards. We do not accept C.O.D. orders.
We also accept wire transfers. Add $20 to cover fees charged for these transfers.

TERMS OF BUSINESS

Shipping Note: All orders will be shipped from Europe. Please allow 3–4 weeks for delivery. Shipping and handling via airmail: US $20.00 per order. **Returns** Damaged or miss-shipped goods may be returned for replacement or refund. All returns must have an RA #. Call or email customer service to receive an RA# before returning the merchandise and be sure to put the RA# on the outside of the package. Please save shipping materials for possible carrier inspection. Requests for RA# must be received 30 days from invoice. **Patents** Patent protection may exist with respect to circuits, devices, components, and items described in our books and magazines. Elektor accepts no responsibility or liability for failing to identify such patent or other protection. **Copyright** All drawing, photographs, articles, printed circuit boards, programmed integrated circuits, diskettes, and software carriers published in our books and magazines (other than in third-party advertisements) are copyrighted and may not be reproduced (or stored in any sort of retrieval system) without written permission from Elektor. Notwithstanding, printed circuit boards may be produced for private and personal use without prior permission. **Limitation of liability** Elektor shall not be liable in contract, tort, or otherwise, for any loss or damage suffered by the purchaser whatsoever or howsoever arising out of, or in connection with, the supply of goods or services by Elektor other than to supply goods as described or, at the option of Elektor, to refund the purchaser any money paid with respect to the goods.

SUBSCRIPTIONS (US & CANADA ONLY)

Subscription rates (1 Yr.)

Standard Subscription: $39.95
Plus Subscription: $54.95

Canada add $11 per year for postage

All subscriptions begin with the current issue. Expect 3–4 weeks for receipt of the first issue. Subscriptions, renewals, and change of address should be sent to:

Elektor US
PO Box 876
Peterborough NH 03458-0876 USA

E-mail: custservus@elektor.com

Order subscriptions on-line at www.elektor-usa.com

Subscriptions may be paid for by check or money order (in US $ drawn on a US bank only). We accept Mastercard, VISA, Discover and American Express credit cards.

For gift subscriptions, please include gift recipient's name and address as well as your own, with remittance. A gift card will be sent on request.

Subscriptions may be cancelled at any time for a refund of all unmailed issues.

INDEX OF ADVERTISERS