www.elektor-magazine.com

.magazine

October 2013 | £ 4.90

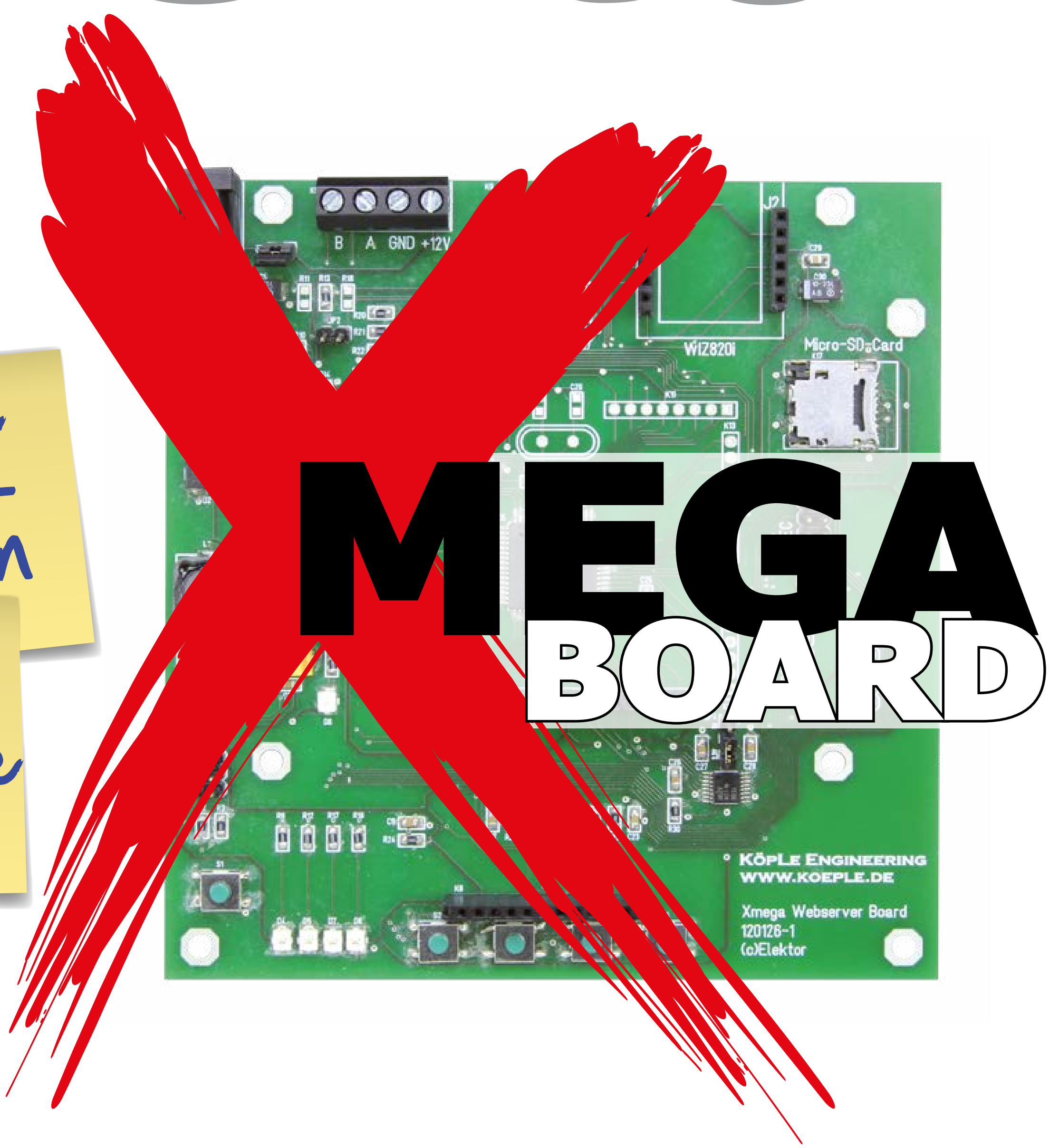
elektor

Sound & Media

Home Automation

Lights & Stage

Web Server



XMEGA BOARD

● **8x8 Two-color LED Matrix** | Numitron Clock & Thermometer | Android ElektorCardi♥scope (3) | **Modular RF Link using Manchester Code (2)** ● DesignSpark Tips & Tricks ● Negative Energy LPC800 Getting Started ● **Heathkit IT-28 Capacitor Checker**

10
9 770268 451173

Shop languages:

reichelt.co.uk elektronik

Your competent online partner for

Components
Power supply systems
Measuring technology

Shop & soldering technology
Home & security technology
Network technology

PC technology
Sat / TV technology
Communication

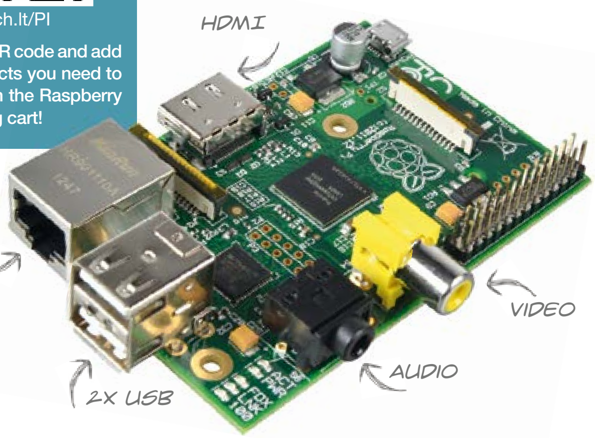
CONVENIENT ORDERING!

http://rch.lt/PI

Just scan the QR code and add all of the products you need to get started with the Raspberry Pi to the shopping cart!

"Raspberry Pi" Mini-PCs

Implement your project!



Raspberry Pi — Model B
with 512 MB RAM

- Broadcom BCM2835
- 700 MHz ARM, Dual Core
- Open GL ES 2.0, OpenVG
- 10/100 BaseT Ethernet jack
- HDMI/RCA composite video jack
- SD card slot
- 2x USB 2.0

€ 38.95 (~ £ 33,72)
RASPBERRY PI B

BE INVENTIVE!

Raspberry Pi – a credit card-sized board with virtually infinite potential applications:

After initial setup and configuration you can, for example, collect all of your media files in one location and then use the Raspberry Pi as an HD media player in your entertainment area, such as the living room. Or use the Raspberry Pi as a control center for network services.

Raspberry Pi has many capabilities – for file and printer access, as a network printer interface, for wireless AirPrint printing or the use of wireless speakers via AirPlay.

"Raspberry Pi" case
TEKO

- Perfect for model A and B
- Snap-on system
- Flame-retardant material
- Status display window
- Ventilation holes

TEK-BERRY	white	€ 4.95	(~ £ 4,29)
TEK-BERRY SW	black	€ 4.95	(~ £ 4,29)
TEK-BERRY TR	transparent	€ 4.95	(~ £ 4,29)

Suitable VESA adapter (50/75/100):
TEK-BERRY VESA transparent € 5.50 (~ £ 4,76)

Raspberry Pi not included; more colours online!

Raspberry Pi OS

- Linux operating system for Raspberry Pi A/B
- Pre-installed on SD card (4 GB)

RASPBERRY PI OS € 18.50 (~ £ 16,02)

WLAN USB adapter, 150 Mbit/s
EDIMAX

- Supports QoS-WMM & WMM power save mode
- Chipset: RTL8188CUS
- Supports WEP, WPA, WPA2
- WPS compatible

EDIMAX EW-7811UN € 8.95 (~ £ 7,75)

TEKO

4-part cooling set

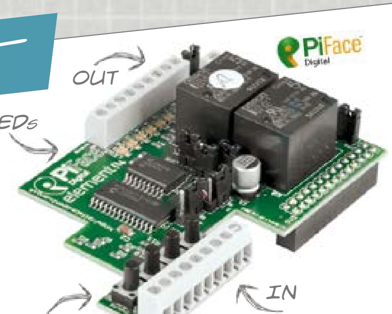
- Compatible with Rev. 1 + 2
- Extremely flat profile
- Easy to install

TEK-BERRY COOL € 4.95 (~ £ 4,29)

YOUR RASPBERRY PI — CONNECTED TO THE WORLD!

PiFace Digital
The I/O add-on for the Raspberry Pi.

- GPIO base • 2 relays • 4 switches
- 8 digital inputs • 8 outputs
- 8 LED displays

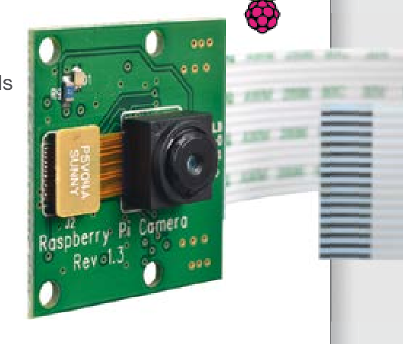


RASPBERRY PIFACE € 35.95 (~ £ 32,12)

TOP ADD-ON:

HD camera
for the Raspberry Pi

- 5 megapixels
- Photo: 2592 x 1944 pixels
- Video: 1080p (Full HD), up to 30 frames/s
- Port: 15-pin thin ribbon cable to the serial MIPI camera port



RASPBERRY PI CAM € 29.95 (~ £ 25,87)

Order now! **www.reichelt.co.uk**

English Hotline: **+49 (0)4422 955-360**

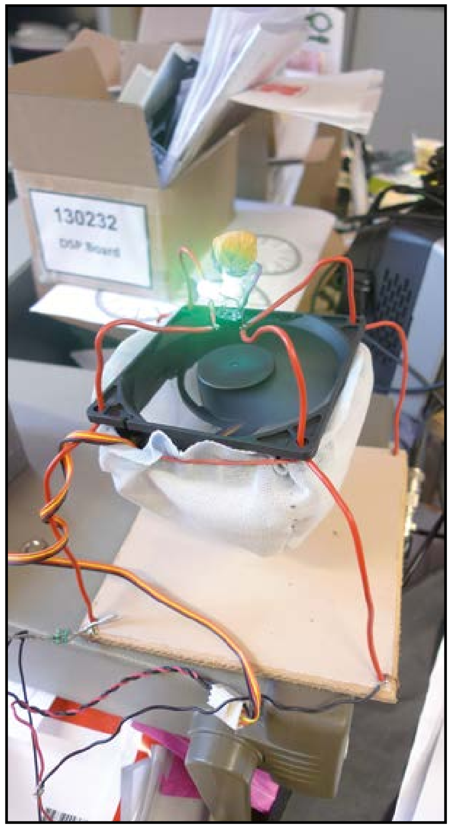
International payment methods:
VISA PayPal

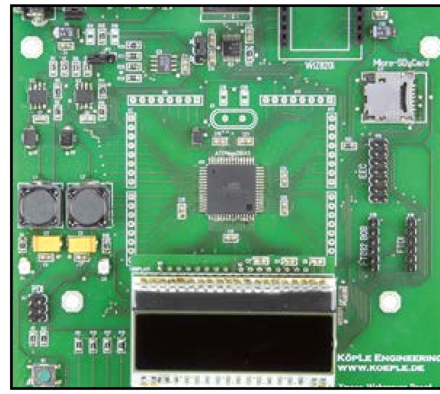
Daily prices! Prices as of: 12/08/2013

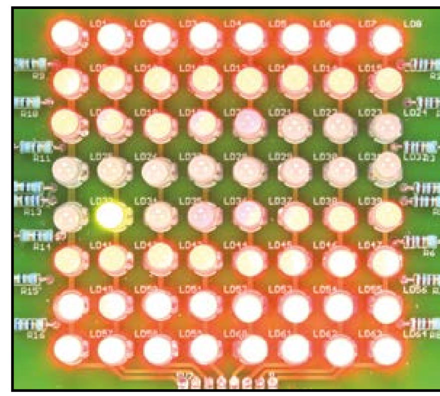
For consumers: The statutory right of withdrawal for consumers shall apply. All stated prices in € include the legal value added tax, ex works Sande/Germany, plus forwarding charges for the entire shopping cart. Our general terms and conditions shall apply exclusively (under www.reichelt.com/agb). Subject to prior sale. All brands, product names and logos are property of the respective manufacturers. Images can be similar. Subject to misprint, errors and changes in prices. reichelt elektronik GmbH & Co. KG, Elektronikring 1, 26452 Sande/Germany (HRA 200654 Oldenburg)

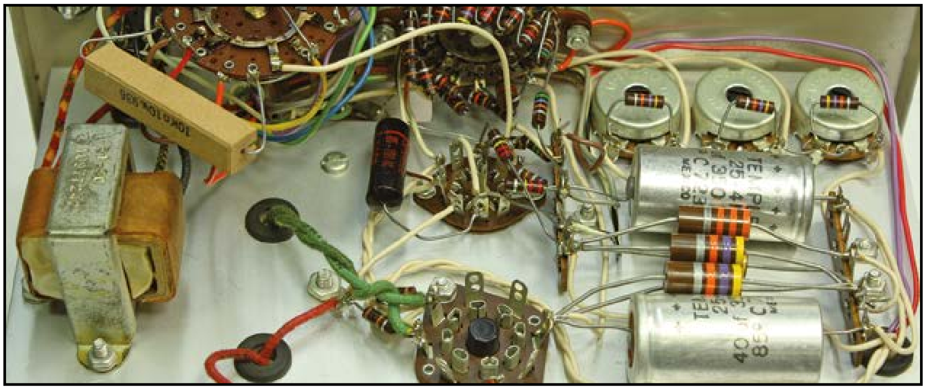

FTDI Chip
THINK DISPLAY DESIGN IS DIFFICULT?
DISPLAY AUDIO TOUCH
FTDI
FT800
3 FUNCTIONS, 1 CHIP
FTDI Chip
LEAVE IT TO
EVE EMBEDDED VIDEO ENGINE
Unparalleled Value in Embedded Display Technology
• Complex graphics made easy
• Elimination of expensive frame buffer memory
• Object orientated design methodology
• Allows low cost uC
WATCH EVE DEMOS ONLINE
See how easy display development can be with EVE, visit www.ftdichip.com

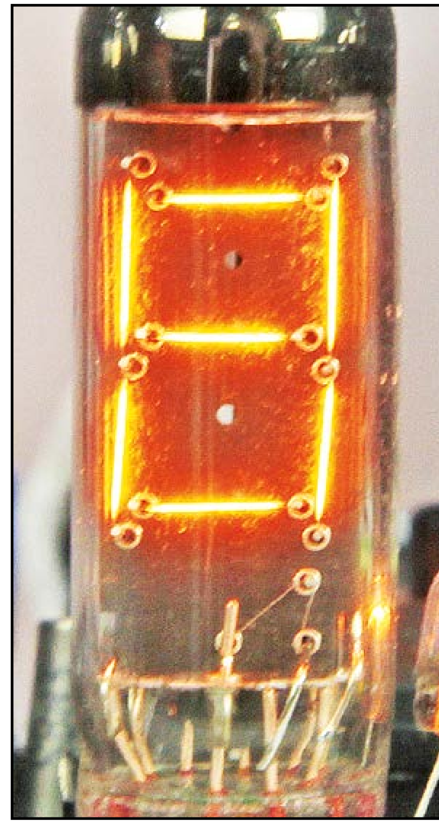
Contents

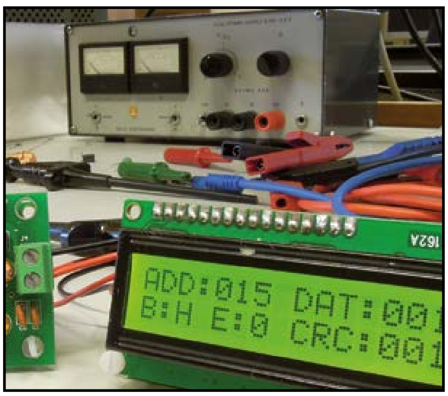


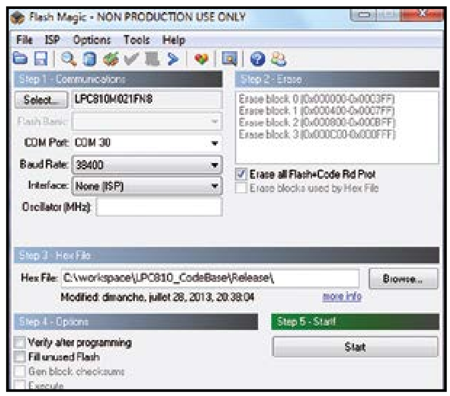












Labs

Industry

Magazine

Volume 39, No. 441
September 2013

ISSN 1947-3753 (USA / Canada distribution)
ISSN 1757-0875 (UK / ROW distribution)
www.elektor.com

Elektor Magazine is published 10 times a year including double issues in January/February and July/August, concurrently by

Elektor International Media
111 Founders Plaza, Suite 300
East Hartford, CT 06108, USA
Phone: 1.860.289.0800
Fax: 1.860.461.0450

and

Elektor International Media
78 York Street
London W1H 1DP, UK
Phone: (+44) (0)20 7692 8344

Head Office:
Elektor International Media b.v.
PO Box 11
NL-6114-ZG Susteren
The Netherlands
Phone: (+31) 46 4389444
Fax: (+31) 46 4370161

USA / Canada Memberships:
Elektor USA
P.O. Box 462228
Escondido, CA 92046
Phone: 800-269-6301
E-mail: elektor@pcspublink.com
Internet: www.elektor.com/members

UK / ROW Memberships:
Please use London address
E-mail: service@elektor.com
Internet: www.elektor.com/member

USA / Canada Advertising:
Peter Wostrel
Phone: 1.978.281.7708
E-mail: peter@smmarketing.us

UK / ROW Advertising:
Johan Dijk
Phone: +31 6 15894245
E-mail: j.dijk@elektor.com

www.elektor.com/advertising
Advertising rates and terms available on request.

Copyright Notice
The circuits described in this magazine are for domestic and educational use only. All drawings, photographs, printed circuit board layouts, programmed integrated circuits, disks, CD-ROMs, DVDs, software carriers, and article texts published in our books and magazines (other than third-party advertisements) are copyright Elektor International Media b.v. and may not be reproduced or transmitted in any form or by any means, including photocopying, scanning and recording, in whole or in part without prior written permission from the Publisher. Such written permission must also be obtained before any part of this publication is stored in a retrieval system of any nature. Patent protection may exist in respect of circuits, devices, components etc. described in this magazine. The Publisher does not accept responsibility for failing to identify such patent(s) or other protection. The Publisher disclaims any responsibility for the safe and proper function of reader-assembled projects based upon or from schematics, descriptions or information published in or in relation with Elektor magazine.

© Elektor International Media b.v. 2013
Printed in the USA Printed in the Netherlands

Red Fruits, Italian Heroes, Penguins, and Texan Bones

Ever since Steve Wozniak assembled a 6502 based microprocessor system, and Steve Jobs literally created a market for it, we techno-inclined folks have delighted in creating and hearing product names with a disarming, if not charming, ring: apple, raspberry, acorn, penguin, Captain Zilog, KIM, Junior. I'm convinced a good number of the names given to microprocessor systems and platforms from the early days of computing have helped significantly to unnerdify the craft of programming and staring at command lines for hours on a 15-inch CRT screen propped up by pizza boxes. The Linux community in particular has set the bar in creative product naming with every new release of "their" operating system. Where the "men in suits" simply put the next higher number behind the product name, a letter "b", or a year, the followers of Tux the Penguin came up with names you'd expect from a Tolkien book.
The main embedded platforms with clearly defined entry levels and educational aims are Raspberry Pi and Arduino, and both are covered extensively in Elektor**.**Magazine and Elektor**.**POST. However, in good engineering tradition there's more to choose from in a diversified market. Elektor's Embedded Linux board is linked this month to a range of extension boards through its "Gnublin" connector (that'll be a young goblin running *GNU's Not Unix*). The same boards, we're proud to say, also connect seamlessly to the Raspberry Pi and—as we've just discovered—to the BeagleBone Black. Have a look at the article on page 28 to see how our modules for controlling relays, displays, stepper motors, I/O devices and temperature sensors can be connected to the latest embedded microcontroller systems running Linux as the abstraction layer. Keeping abstraction and fantasy apart was never easier though as Penguin, Beagle and Gnome seem to get along very well, at least in this edition of Elektor. Let's hope no unadapt, badly named creatures appear on stage, like T-Roll or CEO2B.
More creatures and creations in this issue!

Jan Buiting, Editor-in-Chief

The Team

Managing Editor:	Jan Buiting (editor@elektor.com)
International Editorial Staff:	Harry Baggen, Thijs Beckers, Eduardo Corral, Wisse Hettinga, Denis Meyer, Jens Nickel, Clemens Valens
Design staff:	Thijs Beckers, Ton Giesberts, Luc Lemmens, Tim Uiterwijk, Clemens Valens, Jan Visser
Membership Manager:	Raoul Morreau
Graphic Design & Prepress:	Giel Dols, Jeanine Opreij, Mart Schroijen
Online Manager:	Daniëlle Mertens
Managing Director:	Don Akkermans

United Kingdom
Wisse Hettinga
+31 (0)46 4389428
w.hettinga@elektor.com

USA
Hugo Vanhaecke
+1 860-875-2199
h.vanhaecke@elektor.com

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

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

Netherlands
Harry Baggen
+31 46 4389429
h.baggen@elektor.nl

Spain
Eduardo Corral
+34 91 101 93 95
e.corral@elektor.es

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

Sweden
Wisse Hettinga
+31 46 4389428
w.hettinga@elektor.com

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

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

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

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

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

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

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

Our network

elektor | VOICE COIL | CIRCUIT CELLAR
hifi.nl MUZIEK - FILM - LIFESTYLE | audioXPRESS | techthefuture

Connects you to

Supporting Companies

Beta Layout
www.pcb-pool.com 57

Crystalfontz
www.crystalfontz.com 61

Eurocircuits
www.elektorpcbservice.com 27

FTDI
www.ftdichip.com. 3

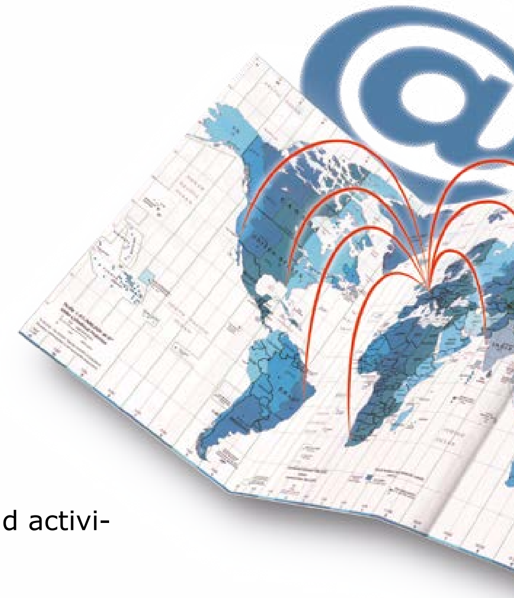
Intelligent Soc s.l.
www.soclutions.com 61

Labcenter
www.labcenter.com. 84

Pico Technology
www.picotech.com/PS235 83

Not a supporting company yet?

Contact Peter Wostrel (peter@smmarketing.us, Phone 1 978 281 7708,
to reserve your own space in Elektor Magazine, Elektor•POST or Elektor.com



Elektor World

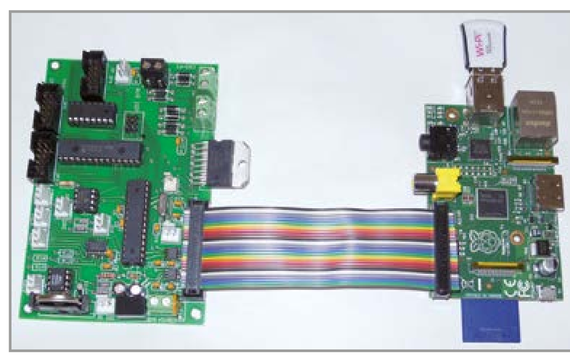
Compiled by
Wisse Hettinga

Every day, every hour, every minute, at every given moment designers and enthusiasts are thinking up, tweaking, reverse-engineering and developing new electronics. Chiefly for fun, but occasionally fun turns into serious business. Elektor World connects some of these events and activities — for fun and business.

The Iso-Pi Board

When *Circuit Cellar* contributor Brian Millier received his first Raspberry Pi in late 2012, he started a project that would inspire his two-part series “Raspberry Pi I/O Board”, which appears in *CC*’s August and September 2013 issues. Millier, a former instrumentation engineer at Dalhousie University in Halifax, Canada, outlines his own Pi learning curve and shares the versatile I/O board he designed for the single-board computer. “In the time since I received my Raspberry Pi, one of the board’s developers has designed an I/O board called the Gertboard,” Millier says. “I feel my board is quite distinct and has some advantages over the Gertboard.”

For example, Millier says, his board “provides complete galvanic isolation between all the I/O devices on-board and the Raspberry Pi itself (which protects the Raspberry Pi board).” You can find out how to configure Millier’s “Iso-Pi” in *Circuit Cellar’s* September 2013 issue.



A short ribbon connects the Raspberry Pi board (right) to Millier’s “Iso-Pi” I/O board.

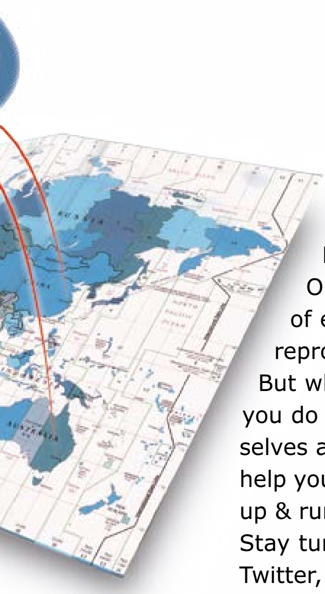
Xpressly for Audio

The long History of Audio in Elektor magazine enters a new era with the re-launch of *audioXpress*. In 2011 Elektor acquired *audioXpress*, *Voice Coil*, *Loudspeaker Industry Sourcebook*, *World Tube Directory*, assorted audio books, and more. Believing that the work of enthusiasts should serve as a model for the industry in the excellence of design and quality of constructions, the titles were founded in the US by Edward T. Dell (1923-2013) and, for over 35 years, served the build-it-yourself audio fan as well as those working in the industry, with great articles, projects, tips and technologies.

A new editorial team reinforced by selected authors from Elektor’s network is currently working on a redesign of the publication with an expanded format. The new *AudioXpress* reaches out to the global audio engineering community, not forgetting to cover the R&D efforts in the industry in many new application areas.

“audioXpress restyled” will be launched at the forthcoming AES convention in NY (October 17 – 20, 2013) with a new graphic layout in print as well as all-digital formats, including a regular newsletter to (currently) over 30,000 members. *audioXpress* is already engaging with the global audio community though Twitter (@audioXP_editor) and Facebook (facebook.com/audioxpresscommunity).

www.audioxpress.com



The Next Step: Arduino.next

For sure, Arduino has become one of the main gateways to Technology World for many young students. One key to its success may be the ease of creating an app in minutes. You don't need a deep knowledge of electronics or programming. Almost everything is on the Internet—simply copy and paste some code, reproduce a simple circuit and your application is ready to run.

But what if you want to go a little deeper? How can you change the behavior of your application? What can you do to make it work in a different way? What circuit do you need to run a new function?... We asked ourselves all these questions and are working to come up with proper answers, simple and easy to understand to help you to take the next step on your favorite embedded platform. That step is called Arduino.next and will be up & running soon—powered by Elektor.

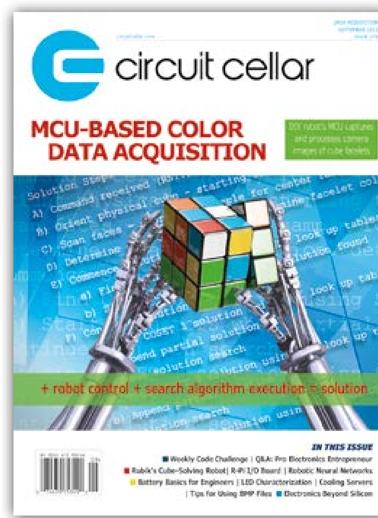
Stay tuned to our communication channels! Follow us on Facebook, www.facebook.com/arduinonext, and on Twitter, @ arduinonext, and check out the Arduino products already on sale at www.elektor.com.

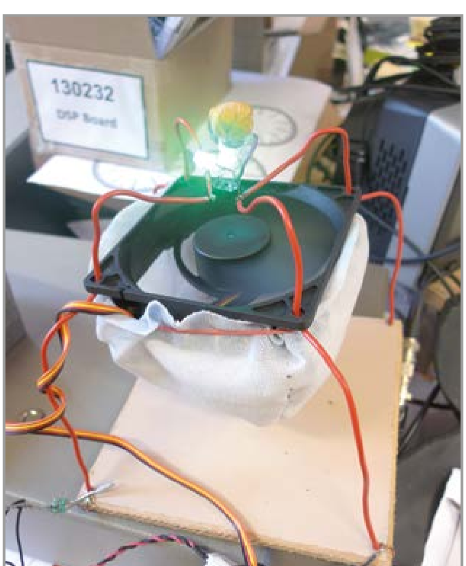
Circuit Cellar 'refreshed'

With the September 2013 issue, *Circuit Cellar* magazine unveils a bold new layout and fresh content for electrical engineers, academics, and serious embedded systems designers everywhere. Together with the Elektor International Media staff, the *CC* team delivers a modern, clean layout that makes studying photographs and analyzing schematics easier and more exciting. Built into the layout are also handy direct links (QR codes as well) to a variety of essential online resources, such as source code, videos, and parts lists.

As for new content, *CC* is delivering two informative columns: Green Computing by Ayse Coskun (Issue 278) and Programmable Logic in Practice by Colin O'Flynn (Issue 279). Another new feature is CC World (p. 8). Much like the Elektor World section of *Elektor* magazine, *CC* now provides monthly updates on topics of interest to the community, such as the *CC Weekly Code Challenge* (http://bit.ly/1brGEIU).

The team hopes you enjoy the refreshed *CC*. To submit articles and projects ideas, write to editor@circuitcellar.com.





Animal-Friendly Mosquito Trap

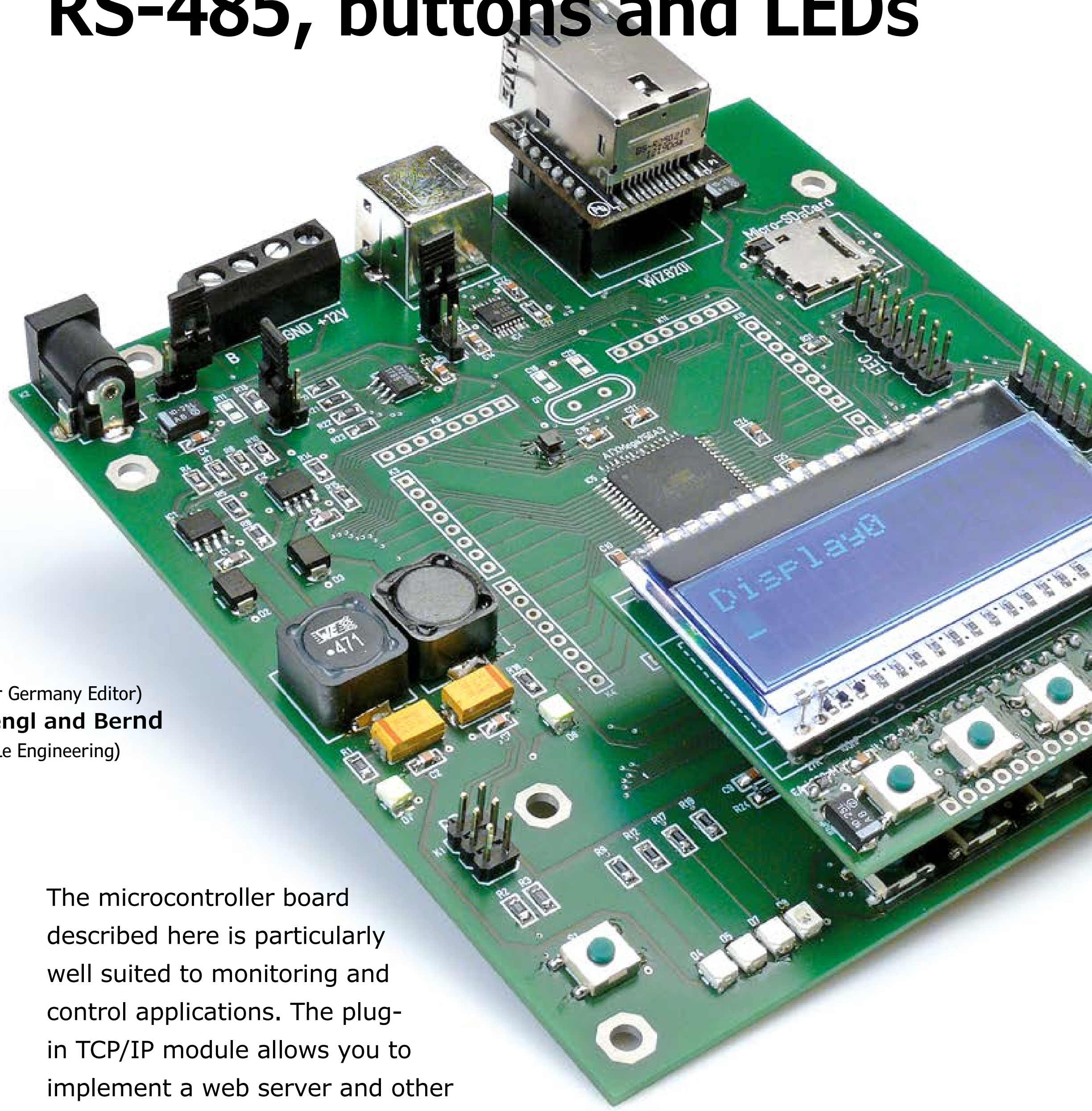
For a lot of you the annual 'August fight' with the mosquitos has begun. Good luck—you will probably wake up every night, exploring the bedroom for them small little !@#$$$%'s that are after your blood.

Or... do like Aurélien Moulin, our Elektor Labs trainee from France. Always on the look for new projects he proposed his 'ultimate bug killer'; an LED and an old computer ventilator. When the jokes had subsided we asked him if had tried this for real—his answer was simply, 'yes' (our French trainees are invariably deadly serious). The idea is simple—the LEDs attract the mosquitoes, then the ventilator sucks them down into a small net. Aurélien made an undemanding prototype and to our great surprise the contraption caught around 120 mosquitoes (say, one hundred and twenty!) on the first night!

But now the $\$10^6$ dollar question! All mosquitos caught in the net were still alive. Question to you—how do you think these mosquitoes defied the killing speed of the vent blades?

XMEGA Web Server Board

Display, SD card, Ethernet, RS-485, buttons and LEDs

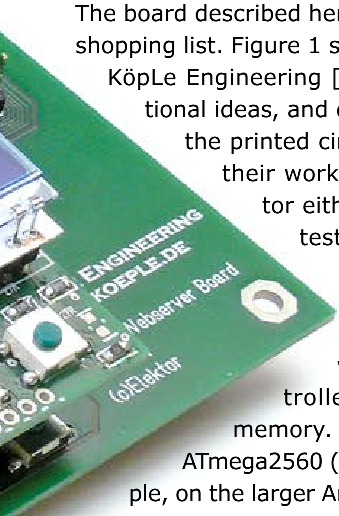


By **Jens Nickel** (Elektor Germany Editor)
Development: **Achim Lengl and Bernd Köppendörfer** (KöpLe Engineering)

The microcontroller board described here is particularly well suited to monitoring and control applications. The plug-in TCP/IP module allows you to implement a web server and other network-oriented applications, and a microSD card provides mass storage. Four LEDs, four buttons, and a removable display provide the user interface options. And of course the board comes with a wide range of external interfaces.

The popularity of the ElektorBus demonstrated that there is significant demand among our readers for monitoring and control applications. Of course it is always possible to use a PC as a central controller, but for many applications this is overkill, in terms of cost, size and noise. For some applications the Elektor Embedded Linux board is a good choice, although not everyone will be comfortable using its free operating system. And in many cases all that is needed is an 8-bit microcontroller, such as one from the AVR series. External interfaces are essential on a board like this, so that remote sensors and actuators can be accessed. Here RS-485 is a good choice, and of course an Ethernet interface is useful. An SD card interface makes it easy to store sensor readings, and a text display, along with buttons and LEDs, allows the construction of simple menu-based user interfaces.

Based on an XMEGA



The board described here answers to the above shopping list. Figure 1 shows its block diagram. KöpLe Engineering [1] brought some additional ideas, and designed the circuit and the printed circuit board. The fruit of their work is available from Elektor either as a ready-built and tested board or as a blank PCB [2]: see Figure 2.

With web server applications in mind we chose a microcontroller with plenty of flash memory. We decided against the ATmega2560 (which is used, for example, on the larger Arduino boards) and went instead for the ATxmega256A3. This device has 256 KB of flash memory and 16 KB of RAM, as well as a few nice extra features such as an advanced event system [3]. This allows us, for example, to count level transitions on any of the GPIO pins. The interrupt controller with configurable priority levels is also useful in more sophisticated applications. At first it seemed rather a disadvantage that the register layout is not compatible with older members of the ATmega family, and even programmers familiar with AVR devices might need a few hours' study of the new datasheet to get up to speed. A quicker approach is to use the drivers for the UART, SPI and other interfaces provided by the manufacturer, who also provides a freely-downloadable

Features

- ATxmega256A3 with 256 KB flash and 16 KB SRAM
- Four pushbuttons and four LEDs
- Plug-in display module, three lines of sixteen characters, with LED backlight
- RS-485 driver with screw terminals for A and B signals, plus 12 V and GND (ElektorBus compatible)
- Optional header for attaching FTDI USB-to-TTL cable
- Optional header for Elektor BOB USB-to-TTL converter module
- Additional UART pins brought out to optional mini-DIN socket
- Optional headers for connection to almost all pins of the microcontroller
- MicroSD card slot connected to SPI interface
- Socket for WIZ820io module, available from Elektor (# 130076-91)
- Embedded extension connector with three ADC inputs, two GPIOs, SPI and I²C on a 14-pin (2x7) header, range of expansion boards available from Elektor
- Printed circuit board fits Hammond enclosure 1598REGY and RS part number 220-995
- Programmable using low-cost AVRISP programmer and free Atmel Studio software
- Software library in C for all peripheral modules available for free download

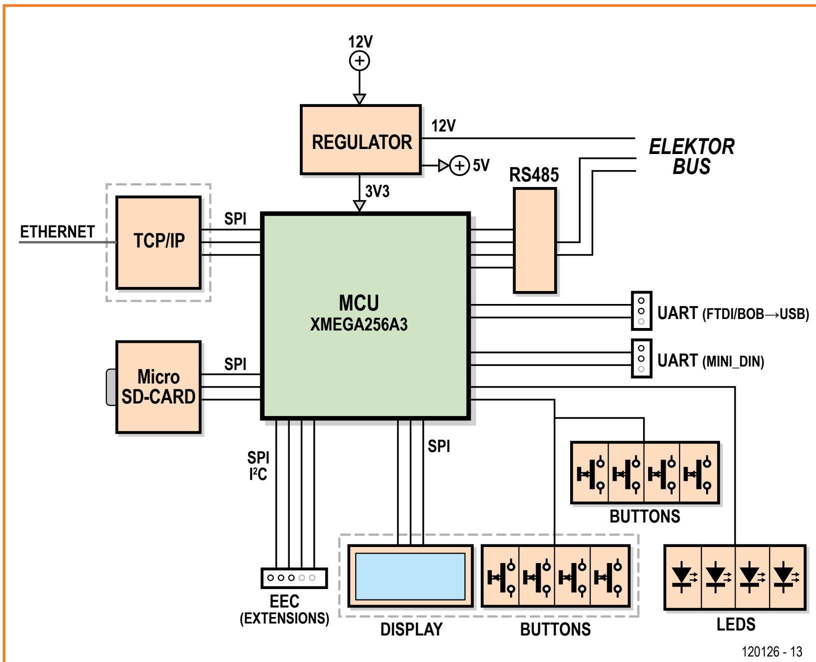


Figure 1. The Elektor XMEGA board is available ready built and tested. It can be expanded with additional modules via optional pinheaders.

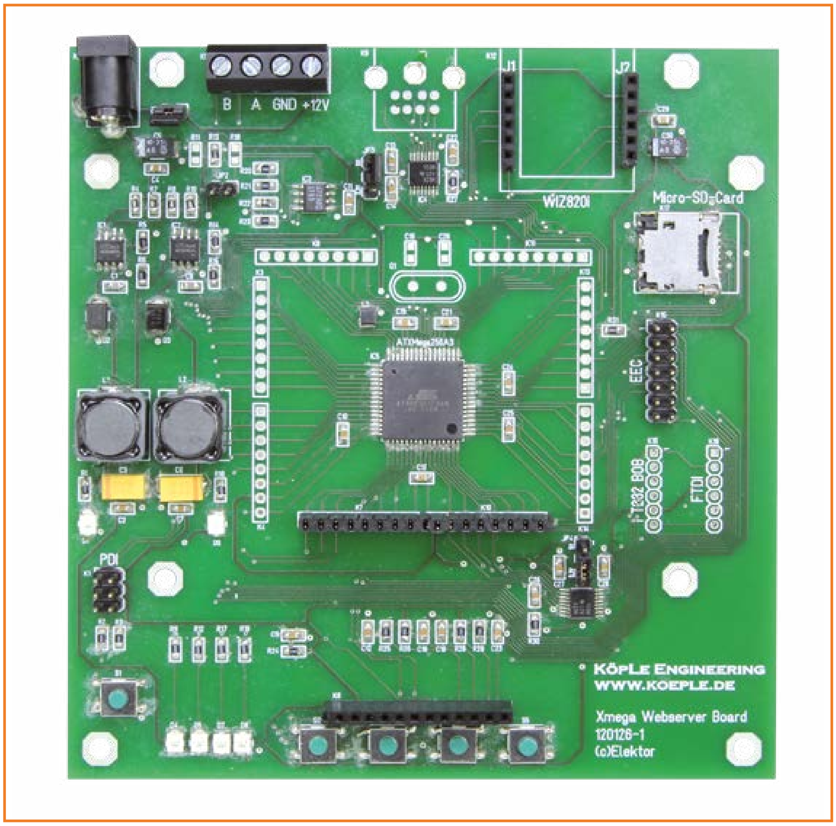


Figure 2. Block diagram of the Elektor XMEGA board.

development environment called Atmel Studio 6. At Elektor we have made things even simpler: we provide an API (application programming interface) for the microcontroller and a driver file for the board, covering all the device's peripherals. More on this below.

Power supply

As is *de rigueur* for modern electronics, the XMEGA microcontroller requires a power supply voltage of 3.3 V. It was therefore clear from the outset that we would have to have supplies at both 3.3 V and 5 V on the board, just so that we could communicate with existing 5 V electronics. Hence there are two power supply circuits, each built around an energy-efficient MC34063A [4] switching regulator: see the circuit diagram in **Figure 3**. There is a handy tool to help with component selection for this circuit available on the Internet [5].

On the input side both regulators are fed with 12 V. This supply can come either from a terminal block (see below) or from a jack: observe the correct polarity! JP1 is used to select the power source.

Connector K1 is the 6-way programming and debugging interface, which allows in-circuit programming of the microcontroller. The pinout of this connector is different from the familiar in-system programming connector for the ATmega series, but nevertheless the low-cost AVRISP mkII programmer [6] can still be used: it will automatically work with the 3.3 V supply of the XMEGA device. And, of course, there is a reset button.

Interfaces

The tried-and-tested UART interface is still the most popular for board-to-board communication. The programming required is simple thanks to the hardware UARTs built into the microcontroller. Furthermore, there are many converters available, for example to RS-485 and USB. Our microcontroller contains six UART modules, of which three are used directly on the board.

Signals PC2 and PC3 from one of the UARTs are taken to an RS-485 driver device, with the RS-485 A and B lines brought out on a terminal block. Ground is also available on this terminal block, which makes it easy to connect to other RS-485 or ElektorBus boards. For example, our RS-485-to-USB converter [7] can be connected using three wires, allowing remote control, possibly over a considerable distance, using a PC; the Elektor AndroPod module [8] allows similar control using a tablet or smartphone. A fourth connection on the terminal block allows the board to be powered from the 12 V rail on the ElektorBus.

The digital signals DE and /RE control transmission and reception by the RS-485 interface chip. JP2 enables a 120 Ω termination resistor on the bus. The optional resistors R11 and R18 pull the A and B lines to a defined level if no other node on the bus is transmitting, which reduces the board's sensitivity to interference. In any case, in our experiments on the ElektorBus we did not notice any problems without this biasing in place.

We have wired the pins of a second UART to a mini-DIN socket (not fitted as standard on the board), along with four GPIO signals. This is convenient for communicating with other electronics over a suitable cable. For example, the mini-DIN socket is compatible with the mini-DIN socket on the Andropod module. For maximum compatibility JP3 selects whether the signal levels are suitable for 3.3 V or 5 V logic.

A third UART (on pins PD2 and PD3) is designed to be connected to a USB-to-UART converter. We have given constructors the choice between an

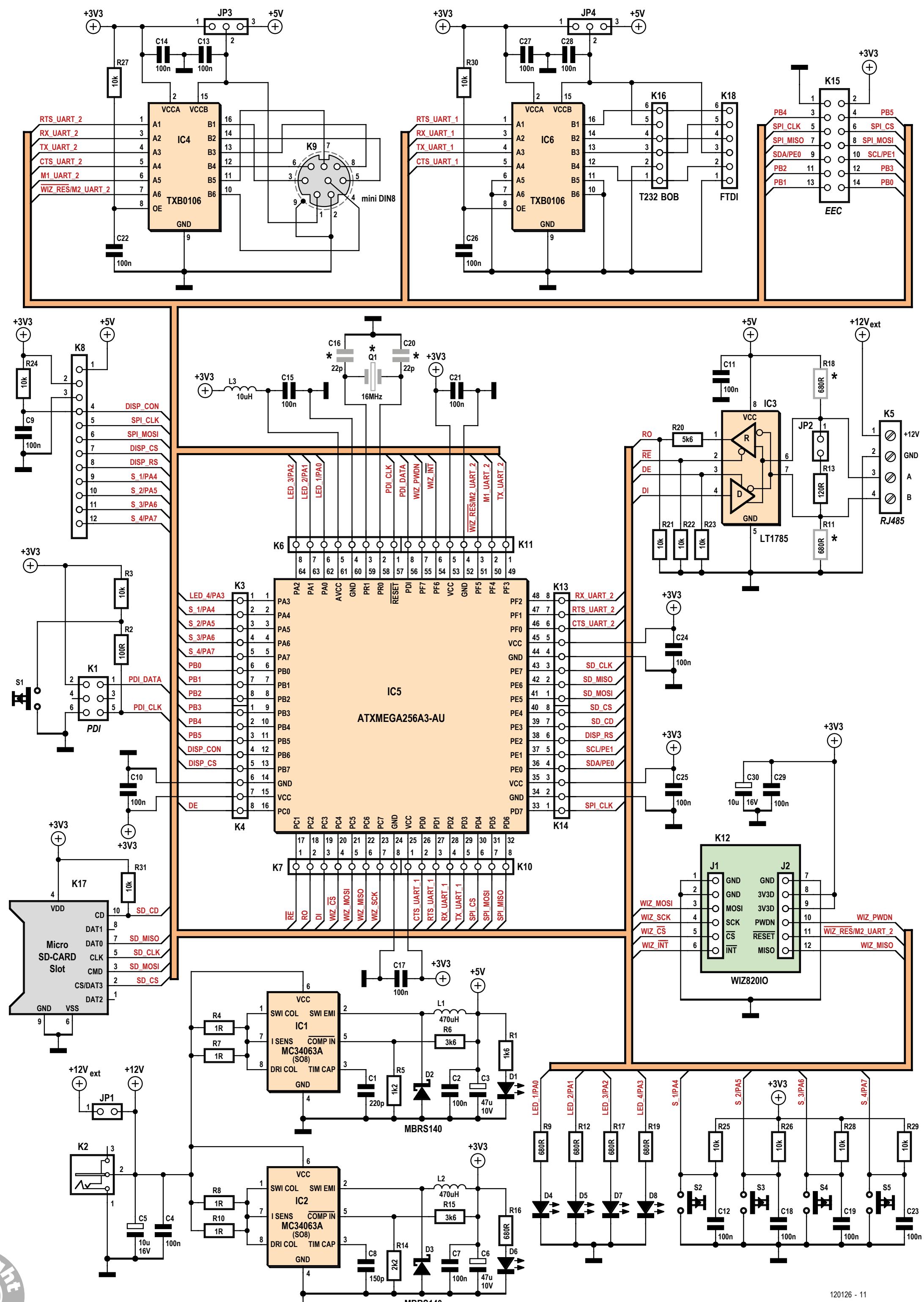


Figure 3. Circuit diagram of the XMEGA board. Extensive use is made of the microcontroller’s I/O facilities, including three SPI modules, three UARTs and an I²C module.

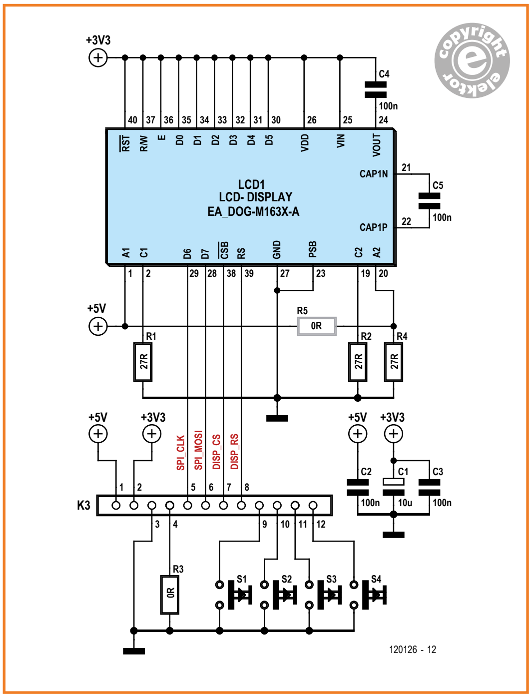


Figure 4. Circuit diagram of the display module. The four buttons allow a menu system to be implemented.

... Web server and other network applications the simple way ...

FTDI USB-to-TTL adapter cable and the Elektor 'BOB' USB-to-TTL converter board, both of which are available from Elektor Online Store [2]. Again a level shifter is provided for compatibility with both 3.3 V and 5 V logic levels: the desired voltage is configured using JP4. The constructor can fit straight or right-angled headers, whichever is more convenient for the application.

Expansion

Although the above interfaces will prove enough for many applications, it is easy to imagine situations where the board's capabilities need to be expanded by adding extra devices. Mostly these devices will be controlled over I²C or SPI interfaces, and so we decided to bring out the corresponding controller pins to a header. At this point we hit upon the idea of using the same pinout for the connector as Benedikt Sauter had used in the Elektor Embedded Linux board. This means that we can use the 14-way 'embedded extension connector' (EEC) to connect any of the expansion boards designed for the Elektor Linux board by the Embedded Projects team. Among those available from Elektor are an expansion board with a display, a port expander and a real-time clock [9], a relay board [10], a stepper motor driver board, a temperature sensor and more: see earlier articles [11] and [12].

It is also possible to fit further headers or sockets around the microcontroller for testing, debugging and expansion: these provide access to practically all of the pins of the XMEGA. As supplied by Elektor a socket is only soldered in the row along the bottom edge of the chip, but this is purely to provide additional mechanical support for the display module described below; it does not have any electrical function when used in this way.

User interface

It is useful to have a display connected to the board if it is to be used at the center of a monitoring and control application. But not all applications will need a display, and so we have made it a separate module that can be fitted to the controller board if required. This also gives the option of mounting the display elsewhere, for example in a separate enclosure.

Header K8 for the display has twelve pins, of which three are reserved for ground and the two supply voltages. The display itself is controlled over SPI: as well as the usual SPI signals MOSI, SCK and CS there is a fourth signal RS which specifies whether a given byte is to be interpreted by the display as a command or data. More information on this can be found in the datasheet [13].

When the display module is attached to the controller board it takes pin 4 on the header to ground. This pin is connected to port PB6 of the microcontroller. If PB6 is configured in software as an input and the internal pull-up resistor is enabled, then the microcontroller can detect whether the display is fitted by checking the level on this input.

Four further pins are dedicated to the four pushbuttons that are also fitted on the display mod-

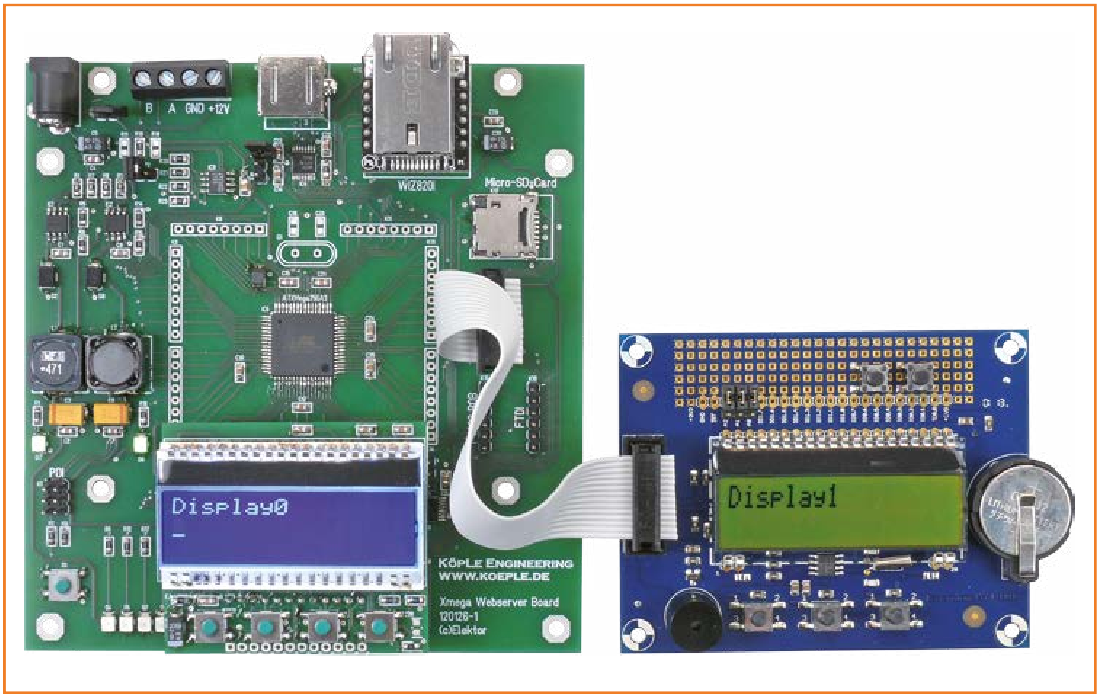


Figure 5. A range of expansion boards can be connected via the 'embedded extension connector'.

ule (see **Figure 4**). The buttons are arranged directly below the display so that a menu system can easily be implemented.

Even if an application does not make use of a display, the pushbuttons might still be required. For this reason the buttons are duplicated on the main board, connected to the same port pins on the microcontroller. Capacitors C12, C18, C19 and C23 serve to debounce the contacts.

Last but not least in this section we will mention the LEDs, which are indispensable not just as status indicators in an application, but also for debugging code. The board sports a total of four LEDs

SD card and network

The SD card slot accepts microSD cards, which serve as mass storage for our board. The SD card is driven in so-called 'SPI mode', with the four signals MISO, MOSI, SCK and CS connected to a hardware SPI peripheral module on port E of the XMEGA. As with the other peripheral units on the microcontroller we have included support in the form of a small library. The library deals only with raw data stored on the card: in other words, only supporting reading from and writing to the card on the board itself. If you wish to develop the driver further, you will find a good introduction to the subject at [14]. The CD pin on the SD card slot is taken to ground when a card is inserted, and this signal is connected to port pin PE3 on the XMEGA microcontroller.

A special feature of the board is the space provided to fit a WIZ820io network interface module. This module, also available from Elektor Online Store (# 130076-91) [2], is a self-contained processor board incorporating a TCP/IP stack. This relieves the XMEGA microcontroller of the task of dealing with network traffic at the protocol level. All the microcontroller has to do is tell the module when to open a socket (specifying an IP address and a port), when to send characters, and so on. Data can be received over a network socket in a similar way. Communication between the XMEGA and the network interface module is again over an SPI bus, in this case connected to port C of the microcontroller. Space does not permit a full description of the module here, but documentation can be obtained from the Korean manufacturer WIZnet [15]. Drivers are available in C for a wide range of microcontrollers, providing an interface to application code (a web server, for example) in the form of functions such as `SocketOpen()`. We have adapted the back end of the driver for our board and for the XMEGA, and have extended the front end by adding a couple of simple extra functions. We will cover the use of the board in a home network and connected

Listing 1. Demonstration with LEDs, pushbuttons and display

```
int main(void)
{
  Controller_Init();
  Board_Init();
  //Extension_Init();

  ApplicationSetup();

  while(1)
  {
    ApplicationLoop();
  }
};


void ApplicationSetup(void)
{
  LEDButton_LibrarySetup(ButtonEventCallback);

  Display_LibrarySetup();
  Display_WriteString(0, 0, "Display0");
  //Display_WriteString(1, 0, "Display1");

}

void ApplicationLoop()
{
  ButtonPollAll();
}

void ButtonEventCallback(uint8 BlockType, uint8 BlockNumber,
uint8 ButtonPosition, uint8 Event)
{
  //Buzzer(BuzzerBlockFirstIndex, 1000,
BUZZER_TONEMODE_RAMP);

  if (Event == EVENT_BUTTON_PRESSED)
  {
    ToggleLED(0, 0);
    Display_WriteNumber(0, 1, BlockNumber);
    Display_WriteNumber(0, 2, ButtonPosition);

  }
}
```

to the Internet in a separate article to be published in the near future.

Software

The existing ElektorBus library, a small display library from KöpLe and the WIZnet driver all had a role in fostering the development of the 'Embedded Firmware Library', which has been the subject of two previous articles in Elektor [16][17]. Since the development of this framework has been focused on support for the XMEGA web server board, we are now in the happy position of being able to offer library modules for all the peripheral blocks on the board.

The current EFL code base can be downloaded at [2] and [18]. This includes the individual code modules and a demonstration application for the board. The XMEGA microcontroller API is as usual contained in a pair of files, ControllerEFL.h and ControllerEFL.c. In this case the two files reside in the subdirectory Xmega256A3. Functions are provided to set and read digital outputs and inputs, to take ADC readings, to send and receive data using the UART blocks, and much more besides. And all this without having to read a datasheet! The board file contains code that calls these microcontroller functions. Low-level functions to talk to the peripheral blocks are provided for the use of higher levels of the EFL. So, for example

```
void Display_SendByte(uint8
DisplayBlockIndex, uint8 ByteToSend, uint
DATABYTE_COMMANDBYTE)
```

is a function which sends a byte over the SPI block that is connected to the display (the internal peripheral block table contains a reference to this unit). The function also sets the RS signal to the correct level according to whether a command or data byte is to be sent: to do this the function has to look up which pin of the microcontroller is connected to the RS line on the display. The higher-level code now does not need to know the wiring of the board, and the application can use the display library in a hardware-independent fashion. Also, as far as the application is concerned, it does not matter whether the display is connected over an SPI interface or a (four bit wide) parallel port when it calls the function `Display_SendByte()`.

Similar low-level functions are provided in the

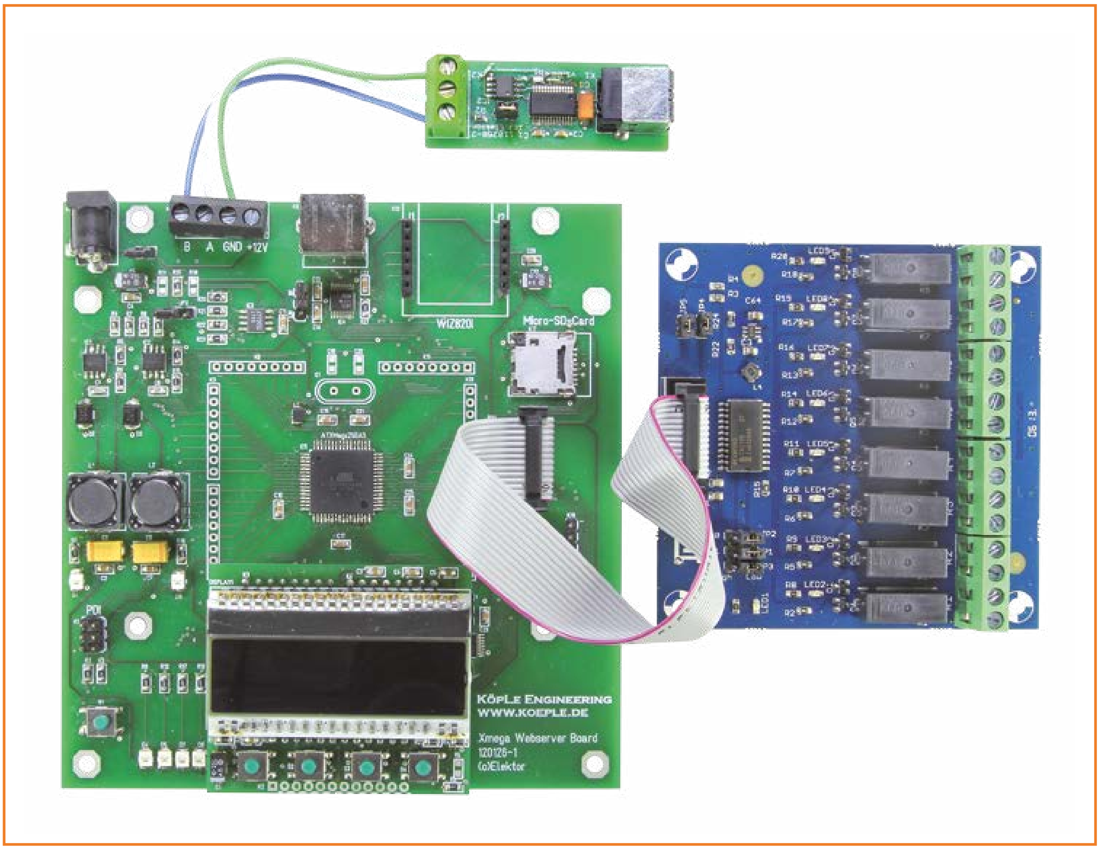


Figure 6. By adding eight relays a powerful control unit can be constructed. The unit can also be controlled from a PC over an RS-485 bus.

board file to drive the SD card and the network module. The code base includes documentation for the functions, produced using Doxygen. The 'Manuals' directory also includes an extra document describing the internals of the EFL, in English, French and German.

My first program

The hardware-independent display library files (DisplayEFL.h/.c) are located in the directory 'Libraries' in the code base.

The application code initializes the library with the call

```
Display_LibrarySetup();
```

and can then talk to up to four displays, numbered from zero to three, located on the controller board or on an extension board. The demonstration application 'XmegaDemo' shows this off: in the code base click on the directory 'Applications' and find the file 'XmegaDemo.atsln'. A double click on this file opens the project in Atmel Studio 6. When the microcontroller has been successfully flashed with the hex file, the first line of the display should show 'Display0'. If you press one of the buttons its number will be shown on the display and the first LED on the board will turn on and off.

Listing 1 shows the source code. In the application setup function we initialize both the Display library and the LEDbutton library. As part of the initialization we tell the library which function in the application is to be called when a button is pressed.

To ensure that the buttons are polled sufficiently frequently the main loop in the application must include a call to `ButtonPollAll()`.

The function `ButtonEventCallback()` contains the code that is to be executed when a button press is recognized. The argument `ButtonPosition` gives the number of the button that has been pressed on the XMEGA web server board (from zero to three). The variable `Event` can take on the values `EVENT_BUTTON_PRESSED (= 1)` or `EVENT_BUTTON_RELEASED (= 2)`. The application can therefore react differently to the press and the release of a button.

About the Development Team

Bernd Köppendörfer and Achim Lengl studied electronics and information engineering at the Georg Simon Ohm Institute of Technology in Nuremberg, Germany, graduating in 2009. In 2010 they founded their own company KöpLe Engineering GbR in Oberasbach, close to the city. Since then they have worked as consultants to various companies, undertaking development work in the field of analog and digital circuit design, from simple modules to complex real-time image processing systems implemented in FPGAs.

Expansion

If you are the lucky owner of a Linux extension board [9], you can connect it via the 'embedded extension connector' using a ribbon cable: see **Figure 5**. A pair of files 'ExtensionEFL.h/.c' accompanies this board, providing the necessary low-level driver functions for its peripheral blocks. These files are already included in the project, and all we have to do is remove the comment characters at the beginning of the line containing the call to `Extension_Init()`. We then do the same with the other commented-out lines in the main source file.

With the program recompiled and flashed into the XMEGA the three buttons on the extension board will now also be polled. The function `ButtonEventCallback()` can determine which group of buttons was responsible for a callback using the argument `BlockNumber`, where a value of 0 means the buttons on the main board, and a value of 1 means the buttons on the extension board.

As you can see, it makes no difference to the application code whether the buttons or display are located on the main board or on the extension board. What makes this even more noteworthy is that the buttons on the Linux extension board are connected to analog inputs on the microcontroller rather than digital inputs (see the section on virtualization in the extra EFL documentation).

Listing 2. Control over RS-485 or UART

```
int main(void)
{
  Controller_Init();
  Board_Init();
  Extension_Init();

  ApplicationSetup();

  while(1)
  {
    ApplicationLoop();
  }
};

void ApplicationSetup(void)
{
  UARTInterface_LibrarySetup();
  UARTInterface_SetBaudrate(0, 38400);

  BlockProtocol_LibrarySetup(UARTInterface_Send, 0,
UARTInterface_GetRingbuffer(0));
}

void ApplicationLoop()
{
  BlockProtocol_Engine();
}
```

Plug and play

As described in previous articles, we can now connect the board to a PC using its RS-485 interface and the RS-485-to-USB converter [6], and connect the relay board described in the previous issue [10] to the expansion connector. The set-up should appear as shown in **Figure 6**.

The application we will use is 'XmegaRelay.atsln'. The code in the main source file is remarkably short: see **Listing 2**. The application set-up code and the application loop code are described in detail in the article on the EFL in the June 2013 edition [17]. As well as setting up the UART interfaces on the board (both for RS-485 and for the FTDI cable or BOB), we initialize a library that implements a simple control protocol called 'BlockProtocol'.

In the application loop function the line

```
BlockProtocol_Engine();
```

polls to determine whether a new command has been received by the board from the PC.

Once the hex file has been flashed into the microcontroller we run a terminal emulator program on the PC, select the appropriate COM port and set the baud rate to 38400. The terminal emulator must also be configured so that when the Enter key is pressed the complete line of characters is

sent, followed by a carriage return (ASCII 13). On entering

```
R 0 0 + <ENTER>
```

the first relay should pull in, and on entering

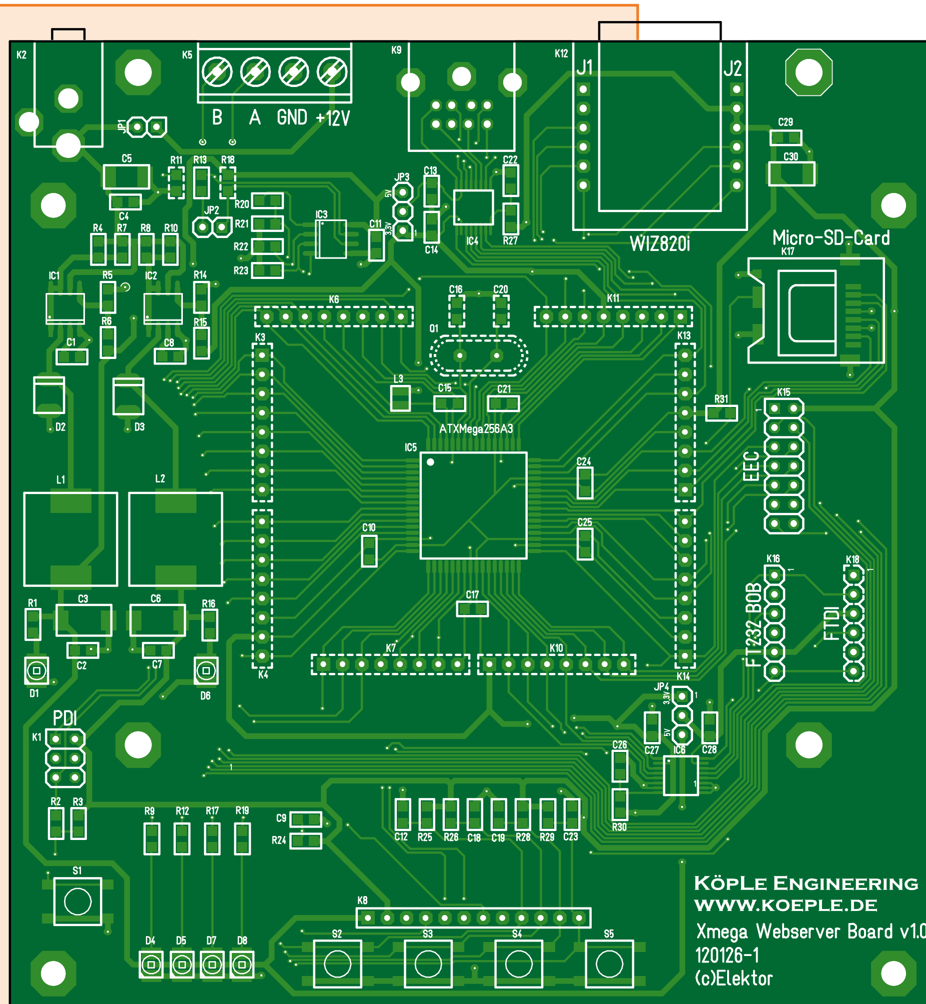
```
R 0 0 - <ENTER>
```

it should drop out again. The other relays can be controlled using commands of the form R 0 x ..., where x ranges from one to seven.

If you do not have a USB-to-RS-485 converter, you can use an FTDI cable or BOB to connect the PC. The code needs to be modified to use the second UART interface block (numbered '1') rather than the first, which is simply a matter of changing a single line of code from

COMPONENTS LIST

CONTROLLER BOARD



Resistors

All SMD, 0805
R1 = 1.6kΩ
R2 =100Ω
R3,R21–R31= 10kΩ
R4,R7,R8,R10 = 1Ω
R5 = 1.2kΩ
R6,R15 = 3.6kΩ
R9,R12,R16,R17,R19 = 680Ω
R11,R18 = 680Ω (optional)
R13 = 120Ω
R14 = 2.2kΩ
R20 = 5.6kΩ

Capacitors

C1 = 220pF (0805)
C2,C4,C7,C9–C15,C17,C18,C19,C21–C29 = 100nF (0805)
C3,C6 = 47µF 10V tantalum (SMD-D/E)
C5,C30 = 10µF 16V tantalum (SMD-C)
C8 = 150pF (0805)
C16,C20 = 22pF (optional)

Inductors

L1,L2 =470µH (Ferrite, PIS4728)
L3 = 10µH (LQH3C)

Semiconductors

D1,D4–D8 = LED, type LG T67K (PLCC2)
D2,D3 = MBRS140
IC1,IC2 = MC34063A (SO8)
IC3 = LT1785CS8 (SO8)
IC4,IC6 = TXB0106 (TSSOP16)
IC5 = ATXmega256A3-AU (TQFP64)

Miscellaneous

JP1,JP2 = 2-pin pinheader, 0.1'' pitch, with jumper
JP3,JP4 = 3-pin pinheader, 0.1'' pitch, with jumper
K1 = 6-pin pinheader (2x3), 0.1'' pitch
K2 = 2.5-mm-jack socket, solder mounting
K5 = 4-way PCB screw terminal block, 0.2'' pitch, solder mounting
K8 = 12-pin pinheader, 0.1'' pitch
K9 = 8-way mini-DIN socket (optional)
K12 = 2 pcs 6-way socket, 0.1'' pitch for Wiz820io module
K15 = 14-pin pinheader, (2x7), 0.1'' pitch
K16 = 6-pin pinheader, 0.1'' pitch, for Elektor 'BOB' USB/TTL converter (optional)
K17 = hinge socket for microSD card
K18 = 6-pin pinheader, 0.1'' pitch, for FTDI USB/TTL cable (optional)
Q1 = 16MHz quart crystal (optional)
S1,S2,S3,S4,S5 = SMD short-action pushbutton
PCB # 120126-1

Alternatively
Elektor # 120126-91 Controller Board, ready assembled and tested

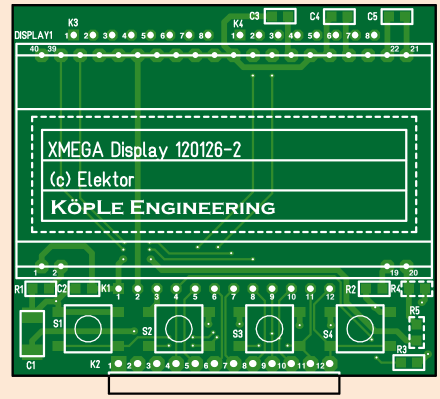
COMPONENTS LIST

Display Module

Resistors
(SMD, 0805)
R1,R2,R4 = 27Ω
R3 = 0Ω
R5 = 0Ω (optional)

Capacitors
C1 = 10µF 16V tantalum (SMD-C)
C2–C5 = 100nF (0805)

Miscellaneous
DISPLAY1 = Display EA-DOG-M163X-A with LED backlight LED55X31
20-way SIL receptacle 0.1'' pitch + 2 pcs 2-way, for mounting DISPLAY1
K1 = 12-pin pinheader, 0.1'' pitch
K2 = 12-pin pinheader, 0.1'' pitch (optional)
K3,K4 = 8-pin pinheader, 0.1'' pitch
S1,S2,S3,S4 = short-action pushbutton



PCB, Elektor #120126-2

Alternatively
Elektor # 120126-92 Display Module, ready assembled and tested

```
BlockProtocol_LibrarySetup(UARTInterface_
Send, 0, UARTInterface_GetRingbuffer(0));
```

to

```
BlockProtocol_LibrarySetup(UARTInterface_
Send, 1, UARTInterface_GetRingbuffer(1));
```

Messages will now be received over the second UART channel. Alternatively we can arrange to receive messages over TCP/IP by changing the line to

```
BlockProtocol_LibrarySetup(IPInterface_
Send, 0, IPInterface_GetRingbuffer(0));
```

We will look in more detail at the possibilities this opens up in the next article in this series.

(120126)

Internet Links

[1] www.koeple.de (website in German)
[2] www.elektor.com/120126
[3] www.atmel.com/Images/Atmel-8331-8-and-16-bit-AVR-Microcontroller-XMEGA-AU_Manual.pdf
[4] www.onsemi.com/pub_link/Collateral/MC34063A-D.PDF
[5] www.nomad.ee/micros/mc34063a/
[6] www.atmel.com/tools/AVRISPMKII.aspx
[7] www.elektor.com/110258
[8] www.elektor.com/110405
[9] www.elektor.com/120596
[10] www.elektor.com/130157
[11] www.elektor.com/130212
[12] www.elektor.com/gnublin
[13] www.lcd-module.de/eng/pdf/doma/dog-me.pdf
[14] http://elm-chan.org/docs/mmc/mmc_e.html
[15] www.wiznet.co.kr/WIZ820IO
[16] www.elektor.com/120668
[17] www.elektor.com/130154
[18] www.elektor-labs.com/efl

NEW
LabVIEW 2013
NOW AVAILABLE!

Low Cost.
High Performance.

PC-based measurements from only £69

Multifunction DAQ

NI USB-6008

12-Bit, 10 kS/s, Multifunction DAQ

- 8 analogue inputs; 2 analogue outputs; 12 digital I/O; 32-bit counter
- Bus-powered for high mobility; built-in signal connectivity
- NI-DAQmx driver software and LabVIEW SignalExpress LE interactive data-logging software
- *£99*

ALSO AVAILABLE:

NI USB-6009 14-Bit, 48 kS/s, Multifunction DAQ - *£179*
NI USB-6210 16-Bit, 250 kS/s, Multifunction DAQ - *£399*
NI USB-9201 12-Bit, 8 Channels, 500 kS/s - *£585*

Sensor-based measurements

NI USB-TC01

Thermocouple Measurement Device - *£104*

ALSO AVAILABLE:

NI USB-9211A 24-Bit, 4 Channels, Thermocouple Input - *£545*
NI USB-9219 4 Channels, Universal Analogue Input - *£1125*

Digital I/O

NI USB-6501

24 Channel, 8.5 mA, USB Digital I/O - *£69*

ALSO AVAILABLE:

NI USB-6525 8 Solid-State Relays, 8 DI, Counter, Channel-to-Channel Isolated - *£260*
NI USB-9421 8 Channels, 11-30V Digital Input - *£370*
NI USB-9472 8 Channels, 6-30V Digital Output - *£370*

Instruments

NI USB-5132

50 MS/s Bus-Powered Digitiser/Oscilloscope - *£563*

ALSO AVAILABLE:

NI USB-5133 100 MS/s Bus-Powered Digitiser/Oscilloscope - *£763*
NI USB-4065 6$^1/_2$-Digit USB Digital Multimeter - *£1040*

Instrument Control

NI GPIB-USB-HS

GPIB Controller for Hi-Speed USB 2.0 - *£349*

ALSO AVAILABLE:

NI USB-232 Single-Port RS232 Interface for USB - *£132*
NI USB-485 Single-Port RS485 Interface for USB - *£132*
NI USB-8451 I2C/SPI Interface - *£365*

At lower cost than you might think, National Instruments provides an intuitive platform that can connect to thousands of sensors and instruments to acquire, analyse and process signals, and present results graphically. Whether programming in C, Basic or LabVIEW on Windows, Mac or Linux, get the latest in PC-based data acquisition, instrumentation and instrument control, starting at **less than £70**.

>> Buy online and explore the range at **uk.ni.com/lowcost**

01635 517300
uk.ni.com
info.uk@ni.com

NEW: LabVIEW 2013 now released.
Find out more at **uk.ni.com/labview/whatsnew**

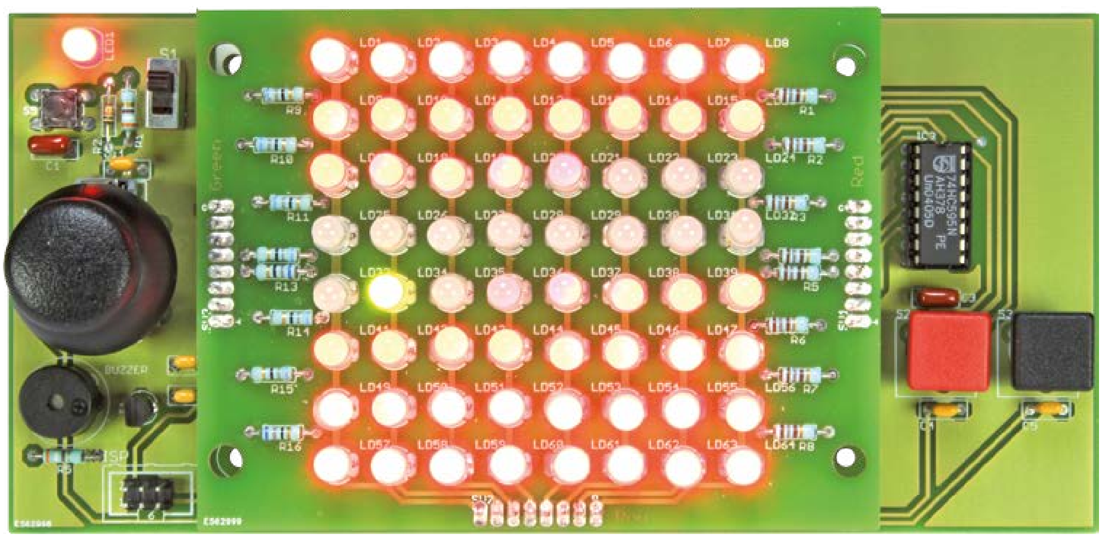
JOIN THE CONVERSATION: *Follow us on* *Search* **niukie**

NATIONAL INSTRUMENTS

©2013 National Instruments Corporation. All rights reserved. National Instruments, NI, and ni.com are trademarks of National Instruments. Other product and company names listed are trademarks or trade names of their respective companies. Prices subject to change. For latest prices, check online at ni.com/products.

8x8 Two-color LED Matrix

With an ATmega328P



By
Ruben van Leeuwen
and
Cederique Prevoo
(Netherlands)

This article describes an alternative method for driving a matrix consisting of a large number of LEDs, while using only a few I/O lines from a microcontroller. As an example application for this circuit, a small game was developed in which an LED can be directed across the matrix using a joystick.

An LED matrix is a good starting point to gain experience in the details of driving LEDs. Such a matrix will quickly accumulate a large number of LEDs and since a microcontroller will usually have too few I/O pins to control them all individually, it is necessary to use some type of multiplexing method. Here we chose a less well-known multiplexing variant, namely charlieplexing (invented in 1995 by Charlie Allan from Maxim). In the more usual multiplexing, the rows of a matrix are controlled with drive signals (Low or High), while the columns are activated one at a time in the rhythm of the refresh frequency. With charlieplexing the functions of row and column signals change all the time, which allows a larger number of LEDs to be controlled with the same number of I/O lines. **Figure 1** shows the basic design. Instead of only one LED, there are now two anti-parallel LEDs between each junction in the matrix. By changing the level of the row or column signal, you can turn on either one of the LEDs, or turn them both off by pulling both drive signals High or Low simultaneously.

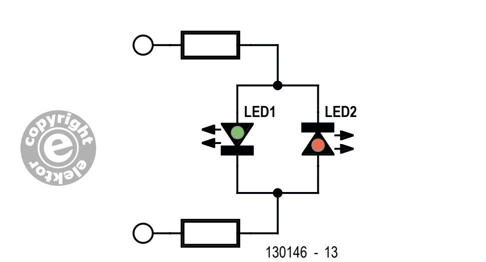


Figure 1. Design principle of charlieplexing.

Matrix

The LED matrix used here comprises 8 x 8 two-color LEDs. This type of LED has both a red and a green LED chip in each LED package, and is connected as shown in **Figure 2** (these two-color LEDs have three connections: two anodes and a common cathode—the connection for the red chip is on the side of the package that has

the flat). The schematic for the matrix, which is accommodated on a separate circuit board, is shown in **Figure 3**.

Control electronics

The control of the LEDs is provided by an ATmega328P, an 8-bit AVR microcontroller with 32 KB programmable flash memory, 1024 bytes of EEPROM and 2 KB of SRAM. In this application the internal oscillator of the ATmega is used, so an external crystal is not necessary. The controller drives three 8-bit shift registers, type 74HC595, one for the green LEDs, one for the red LEDs and one for the common cathodes (IC3, IC4 and IC5 in the schematic of **Figure 4**). With this arrangement we save a large number of port pins on the microcontroller, which now remain available for

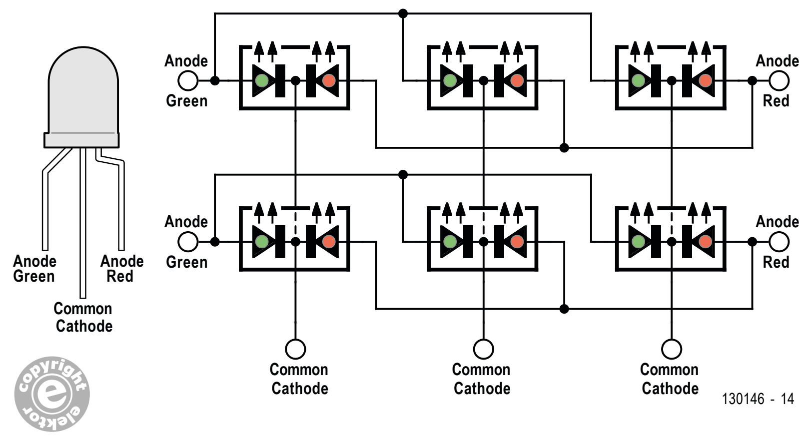
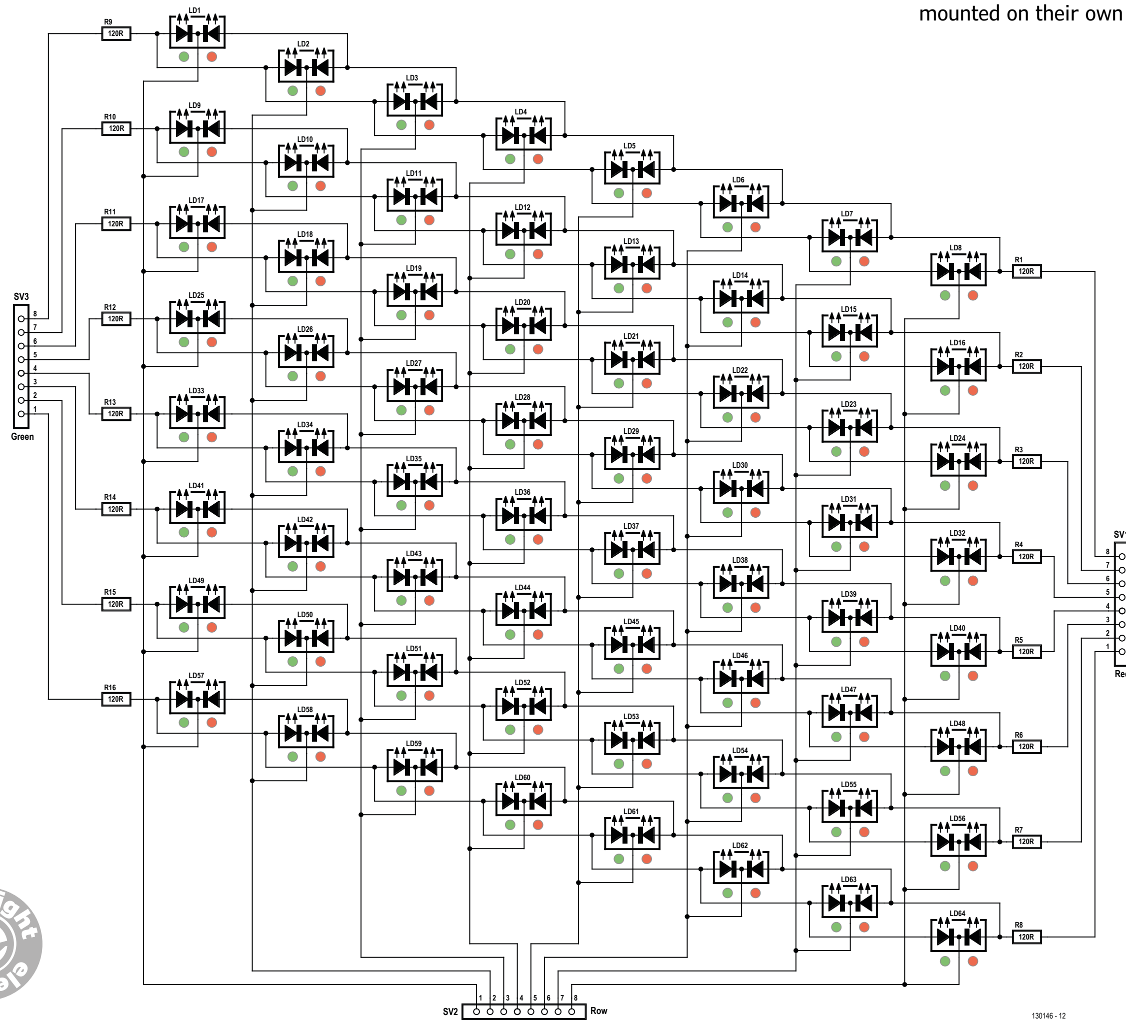


Figure 2. This is how the red/green two-color LEDs are connected in the matrix.

Figure 3. The schematic for the matrix with the 64 two-color LEDs mounted on their own circuit board.



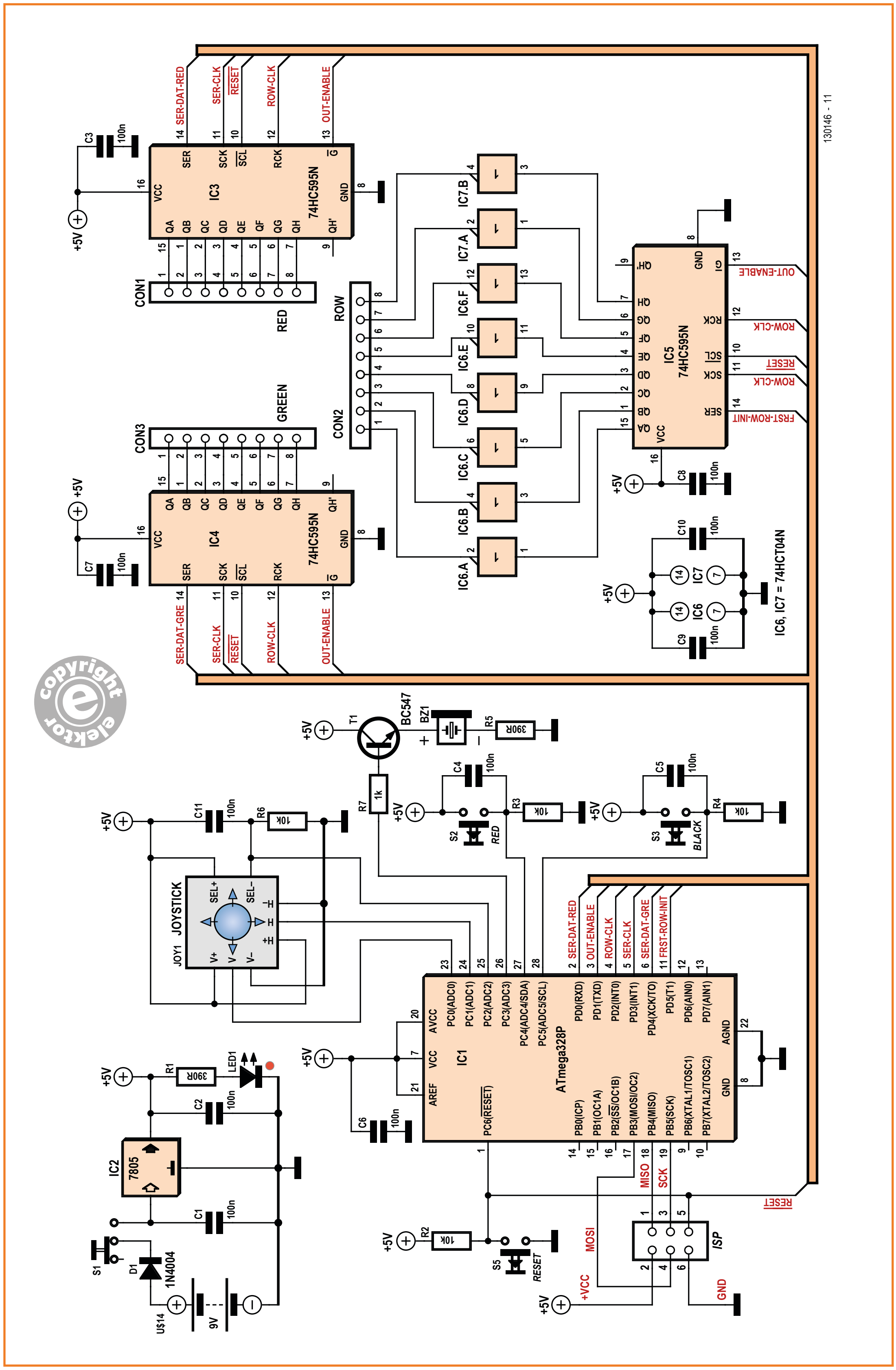


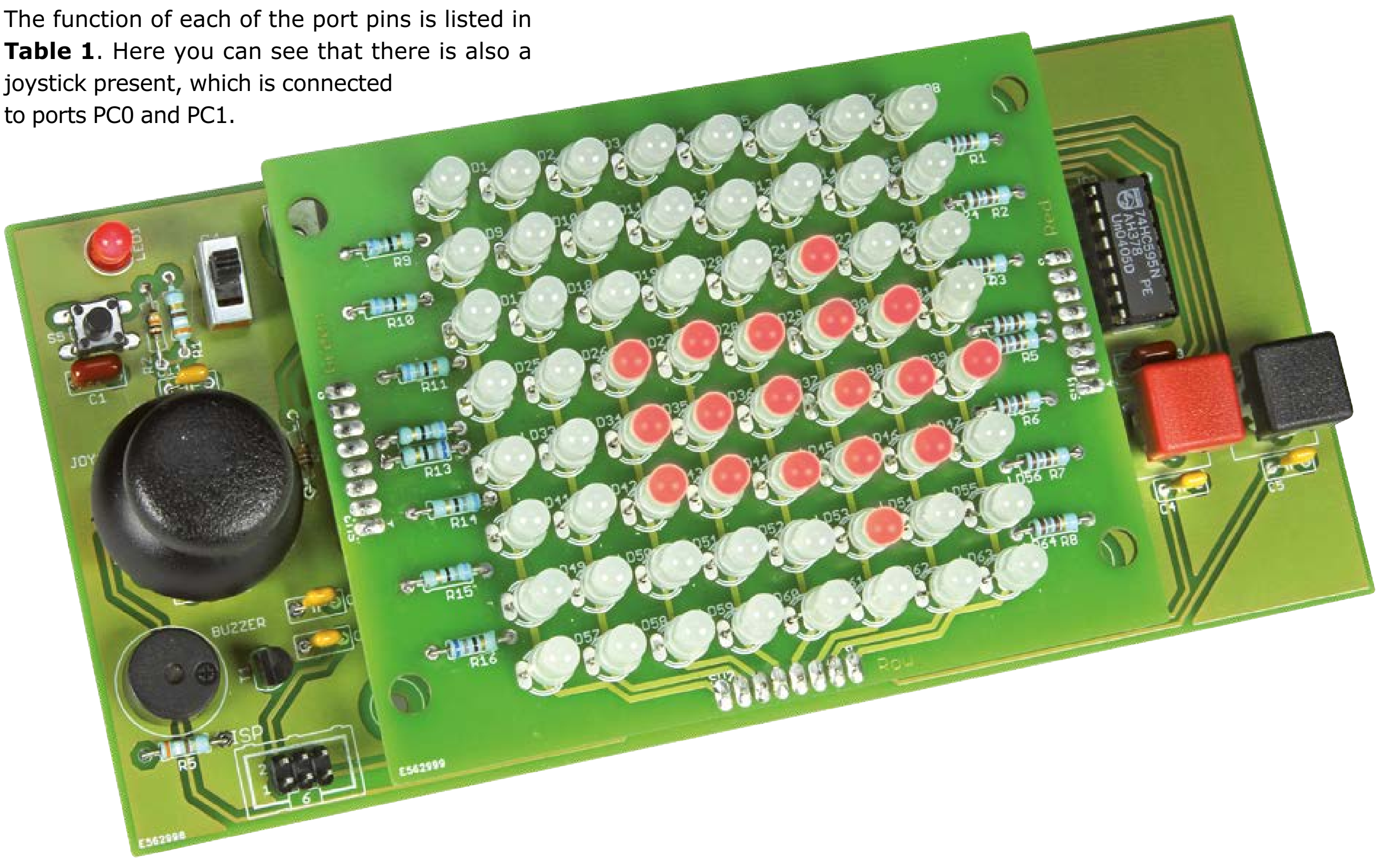
Figure 4.
The driving electronics consists mainly of an ATmega microcontroller and three shift registers.

some other applications.
The microcontroller continually sends serial 8-bit data to the shift registers of the two colors to prepare the information for each row. After these eight bits are sent, a pulse is applied to the Output Enable inputs of the two-color registers and the common-cathode shift register so that a complete row of LEDs can turn on.
When the common-cathode shift register drives a row to the activate state, this row sees a Low signal level; an inactive row has a High signal level. This ensures that there is a voltage drop of 0 volts across the LEDs in the inactive rows (so these LEDs will therefore not turn on). In the active row there will be a voltage difference of 5 V, so the LEDs that are connected to this row can light up.
This whole process takes place eight times per 'frame' so that all rows are driven in turn. A 'frame' in this case is a user-programmed code which represents an entire picture of 8 x 8 red and green LEDs.
The turning-on of the LEDs takes place at a frequency that is above 60 Hz, so that to the human eye it appears that all the LEDs are turned on simultaneously, instead of only one row at a time. This means that 60 frames per second are displayed, which is equivalent to one frame per 0.017 s.
The function of each of the port pins is listed in **Table 1**. Here you can see that there is also a joystick present, which is connected to ports PC0 and PC1.

Table 1. Port connection

Inputs:	Microcontroller port
Black button	PC5
Red button	PC4
Joystick horizontal axis	PC0
Joystick vertical axis	PC1
Outputs:	**Microcontroller port**
Buzzer	PC3
First Row pulse	PD5
Serial data red	PD0
Serial data green	PD4
Serial clock	PD3
Row clock	PD2

That is because on these ports there is also an A/D converter available, which makes it possible to read the position of the joystick as an analog value. In addition, the circuit also contains a DC buzzer to generate sounds. This type has the advantage that it contains a built-in oscillator, but the disadvantage is that it can produce only one fixed frequency. In the schematic we can also find the ISP connector for the in-circuit programming of the ATmega. In this way changes to the software can be programmed quickly and then tested.



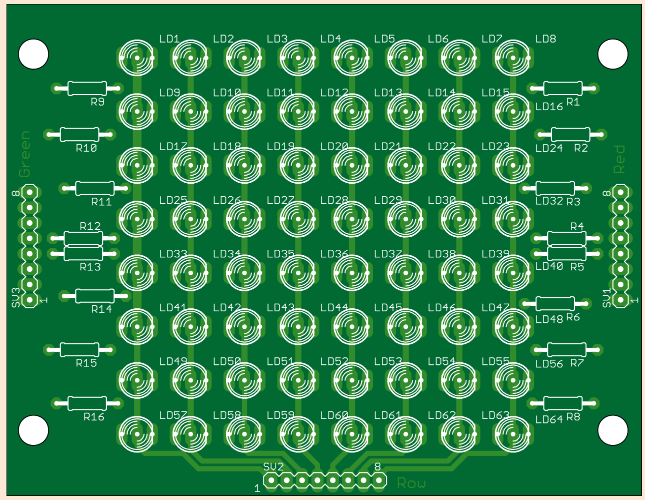


Figure 5.
The printed circuit board for the LED matrix. Make sure that the LEDs are fitted in straight lines and at the same height.

Note: both circuit boards shown at 80% of actual size.

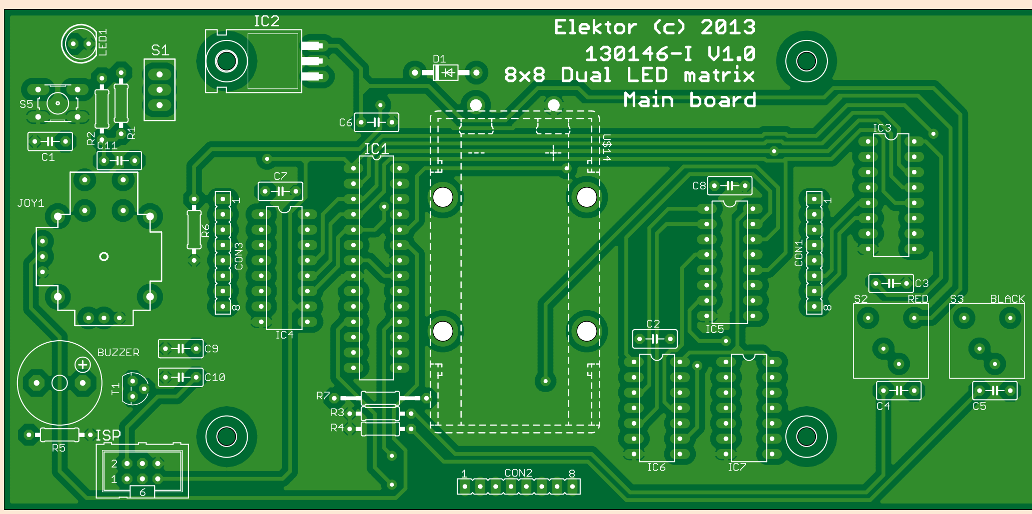


Figure 6.
The main circuit board has been designed in such a way that the matrix board can be attached in the middle on top of it.

COMPONENT LIST

Main Board

Resistors

R1,R5 = 390Ω
R2,R3,R4,R6 = 10kΩ
R7 = 1kΩ
JOY1 = 2-axis joystick 2x10kΩ w. pushbutton (e.g. Conrad Electronics # 425637-89, matiching button # 710047-89)

Capacitors

C1–C11 = 100nF, 5mm pitch

Semiconductors

D1 = 1N4004
LED1 = LED, red, 5 mm
T1 = BC547
IC1 = ATmega328P-PU, programmed, Elektor # 130146-41
IC2 = 7805
IC3,IC4,IC5 = 74HC595
IC6,IC7 = 74HCT04

Miscellaneous

Buzzer = DC buzzer, 30mA / 5V
ISP = 6-pin (2x3) boxheader
S1 = slide witch w. make contact
S2 = pushbutton w. make contact, red cap
S3 = pushbutton w. make contact, black cap
S5 = miniature pushbutton w. make contact
CON1,CON2,CON3 = 8-pin SIL connector
9-V battery w. holder
PCB # 130146-1 [1]

Display Board

R1–R16 = 120Ω
LD1–LD64 = dual LED green/red, IF = 10 mA (e.g. Conrad Electronics # 156269-89)
SV1,SV2,SV3 = 8-pin pinheader
PCB # 130146-2 [1]

The power supply is taken care of by a 5-V regulator of the type 7805. The current consumption of the circuit, including the LEDs, amounts to a maximum of 50 mA. The input voltage has to be between 7 and 12 VDC (from a wall adapter, for example), but for the occasional game a 9-V battery is also sufficient (there is an option for fitting this battery on the back of the board).

Printed circuit boards

Two separate circuit boards have been designed for this circuit, one for the LED matrix (**Figure 5**) and one for the control circuit (**Figure 6**). Since only leaded components are used the construction should not present any problems. It is best to use sockets for the digital ICs. For powering the circuit it is possible to mount a 9-V (6LR22) battery holder on the solder side of the circuit board using four small nuts and bolts. Make sure that on the matrix board you mount the LEDs in neat rows and columns: both the rows and the columns have to be in straight lines, and all the LEDs have to be fitted at the same height. To help with this you can slip a piece of cardboard between the legs of the LEDs before you solder them in place. Also make sure that you mount the LEDs all the same way around; the connection for the red LED corresponds with the flat side of the package. So this side of the LED has to face the same way for all the LEDs on the board (in the direction of the 'Red' screen print label).

Program

There are a few important parts to the program which we will explain in detail here: the serial data control for the matrix, the reading of the analog value from the joystick and the game that is used as the application example to demonstrate this circuit.

Three inputs of the 74HC595 shift register are used, namely Serial-Data, Serial-Clock and Output-Enable. The Serial-Clock input is used to shift the eight bits present in the shift register by one bit on every clock pulse, so that the LSb is lost and the current level of the Serial-Data input is added as the MSb.

After eight new bits have been shifted in, and when the Output-Enable input is activated, the values of these bits in the shift register will be copied to the output buffers of the IC.

For each frame, an array of 8 characters is processed and analyzed, so that it is clear which LEDs have to be turned on for each row. The

Listing 1.

```
if(RowNmbr!=0) //check if the current Row is not the first
{
  //Set First row out to High = PD5 (DDRD 0b00100000)
  PORTD&=0b11011111;
}
else if(RowNmbr==0)
{
  //Set First row out to Low = PD5 (DDRD 0b00100000)
  PORTD|=0b00100000;
}
//supply a row clock pulse
PORTD|=0b00000100;
PORTD&=0b11111011;

for(I=0;I<8;I++) //repeat the routine 8 times to get all the
bits
{
//get the right bit value by bit shifting the bit
if(((RedRowData>>(7-I))%2)==1)
{
//set the serial Red output High = PD0 (DDRD 0b00000001)
PORTD|=0b00000001;
}
else
{
//set the serial Red output Low = PD0 (DDRD 0b00000001)
PORTD&=0b11111110;
}
//get the right bit value by bit shifting the bit
if(((GreenRowData>>(7-I))%2)==1)
{
//set the serial Green output high = PD4 (DDRD 0b00010000)
PORTD|=0b00010000;
}
else
{
//set the serial Green output Low = PD4 (DDRD 0b00010000)
PORTD&=0b11101111;
}
//give a serial clock pulse = PD3 (DDRD 0b00001000)

  PORTD|=0b00001000;
  PORTD&=0b11110111;
}
```

program first checks whether it is for the first row (in this case a pulse will be provided for the common-cathode shift register) and subsequently each bit in a row is tested for the active state; a '1' indicates active and a '0' for inactive. When it has been determined that a particular bit is active, the Serial-Data input of the corresponding color, together with the Serial-Clock input, will be activated; if the bit is inactive only the Serial-Clock input will be activated.
A section of the serial control software can be seen in Listing 1.
In order to read the position of the analog joystick, the existing functions in the avr/io.h library of the AVR development software are used (AVR Studio 5). The following global variables are declared in the program.h header file:

- PortJoyV – The actual vertical value of the joystick,
- PortJoyH – The actual horizontal value of the joystick.

The code shown in **Listing 2** can be found in ProgramInit.cpp.

Listing 2.

```
//analog reading setup
  ADMUX=0b01000000;  // For Aref=AVcc;
  ADCSRA=0b10000110; // prescale div factor = 64
```

Listing 3.

```
//Reading the analog signal of the joystick Vertical-Axis

  //Reads vertical joystick (port 0)
  ADMUX&=0b11111100;

  //Start Single conversion
  ADCSRA|=0b01000000; //ADSC = high

  //Wait for conversion to complete
  while(!(ADCSRA&0b00010000));

  //Clear ADIF by writing "one" to it
  ADCSRA|=(1<<ADIF);
  ADCSRA&=0b10111111;

  PortJoyV=ADC;
```

The reference voltage for the A/D-converter is set to 5 V, which is equal to the power supply voltage for the microcontroller (AVcc). The 'prescale div factor' furthermore ensures that the A/D-conversions take place at the correct speed. The code in the file DataIn.cpp subsequently reads the position for each axis of the joystick. In **Listing 3** you can find an example showing how the position of the vertical axis is read in.

The example program for this project, which is available as a free download from [1], is a little game in which you have to steer a green LED across the matrix using the joystick in such a way as to avoid the oncoming red LEDs.
After an introductory animation the game will begin when the red button is pressed. The first 'level' that is found in the file LoadLevel.cpp is loaded into an array of 117 characters, this therefore represents the values of eight bits high (the complete height of the LED matrix and 117 bits wide. The present position of the green LED is also stored in a character variable and this can be changed by moving the joystick up or down. After each frame, the position of the green LED is bit-wise ANDed with the next row of the level. If these happen to be the same, the loop will end and the game will begin again, restarting with the same level that was being played.

There is also a test function built in. If you press and hold the black button after playing the game or after turning-on of the power supply voltage, you can move a block of four LEDs across the entire matrix using the joystick.

So enjoy the game, but also have fun programming your own applications.

(130146)

The development of this project was the practical project part of an electronics course at the ROC Leeuwenborgh in Sittard, the Netherlands.

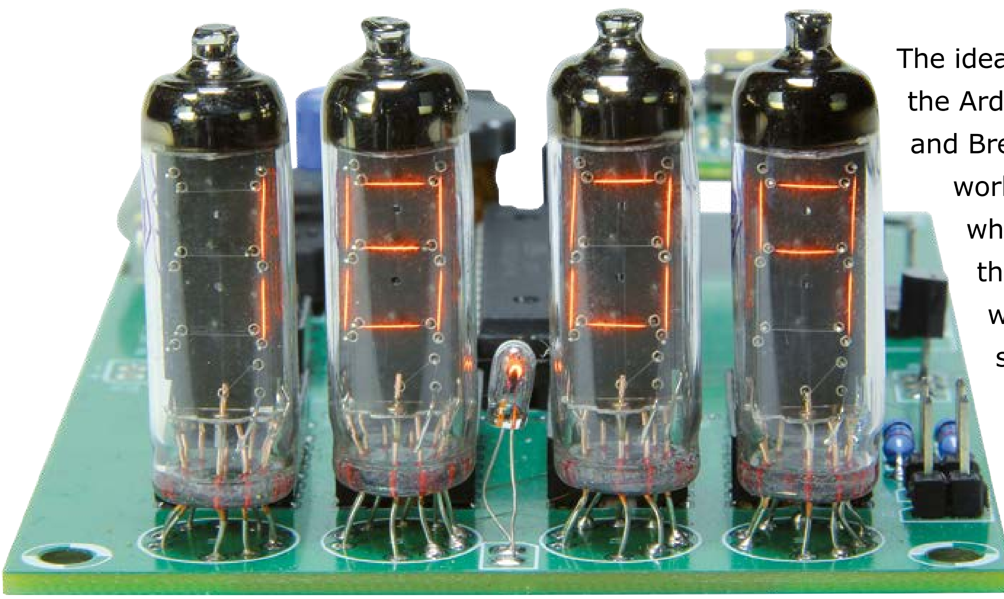
Internet Link

[1] www.elektor.com/130146


THEY SAY THE FUTURE DOESN'T FIT IN A BOX.
Tuesday, January 7
through Friday, January 10, 2014
Las Vegas, Nevada • CESweb.org • #CES2014
GOOD THING LAS VEGAS IS FAR FROM SQUARE.
Over four day, those who shape the future gather in a city built on reinvention. Here, brands, markets and economies converge in what's far more than a tradeshow. And in 2014, there's more opportunity than ever to connect with those who matter. The only question now, why wouldn't you fit it in your future?
Register today at CESweb.org.
2014 International CES
THE GLOBAL STAGE FOR INNOVATION
PRODUCED BY CEA

Numitron Clock & Thermometer

Developed on the Arduino platform



The idea for this project was to take the Arduino world off of its Shields and Breadboards and out into a world of Elektor DIY projects while also hopefully appealing to the diehard AVR hackers. These worlds are so close but are so often treated as separate. The project happily combines today's microcontroller technology with 1950's Soviet tubes now available 'NOS' from Ebay.

By **Paul Court** (UK)

For the clock I decided against the traditional 7-segment LEDs in favor of something much more beautiful in the form of ex-Soviet Numitron tubes. Numitrons are cousins of the Nixie tubes featured in several Elektor projects, but run on about 4 volts rather than >150 volts, hence are more suited to experimentation. Just like the Nixies, Numitrons are readily available on sites such as eBay at very reasonable rates.

Features

- Four IV-9 Numitron tubes
- Date/Time/Year/Seconds/Temperature readout
- ATmega328 microcontroller
- Developed on Arduino platform
- On-board adjustable SMPSU for optimal Numitron brightness setting
- Powered from 9-12 VDC 500 mA power adapter
- No high voltage used
- All development code freely available
- Design powered by Elektor.Labs

So here we have the project, a Numitron bulb, clock & thermometer built on a single board. The project is *based on* the Arduino UNO, meaning the UNO and its IDE were used as development tools only. In the final project there are no shields, no modules and no plugins—just an ATMega328 with peripheral hardware devices around it, although that may be oversimplifying the build ahead, as we'll see.

The next 'tron': Numitron!

Dekatron, *Klystron*, *Thyratron*, *Magnetron*, *Trochotron*, Whatever next, *Annoyatron*? the low-voltage generation may ask, and everyone invariably ending up at www.radiomuseum.org for a glimpse at more baffling, *–tron* suffixed vacuum tube devices from the dark pre-Internet ages.

You don't have to master classic Greek to fathom that a *Num*itron does something with *num*bers. And indeed a Numitron tube is capable of displaying numbers 0 through 9 and a few letters,

using seven discrete segments, which are actually little incandescent bulbs. On many Numitrons, a sort-of-comma (decimal symbol, ds) is included as an 'eighth segment' in the right-hand bottom corner. The segments in a Numitron share a common positive supply line, and their other ends are brought out to wires—not pins. Like many Nixies, Numitrons are wire tubes.

Numitrons are the 'warm-light' variants of the solid-sate 7-segment LED display. Originally developed in the Cold War period, i.e. in the tube era, they are still available as NIB (new in box) a.k.a. NOS (new old stock) items, particularly from Russian suppliers—mostly Ukrainian—active on the Internet. The type IV-9 Numitrons Luc Lemmens wanted in order to replicate the project at his desk in the Elektor Labs were ordered through EBay (a US company for sure) and arrived from Russia intact despite rudimentary but amusing packaging methods applied by the 5-Kstar rated seller. See the photo story in **Figure 1** triggered by your Retronics Editor: LOL.

The main technical specs of the IV-9 numitron used here are given in **Table 1**. As we mentioned in the article on the Nixie VU Meter using the IN-9 [1], what we like to think of as "design data" on these Sovjetisk tubes is subject to (1) wide interpretation, (2) even wider tolerance, and (3) stencil printing in Cyrillic.

Circuit description

Looking at the circuit diagram in **Figure 2**, you might be surprised to see that no multiplexing is applied to the display elements. Instead, each of the four digits of the clock has its own driver IC type SN74LS47D. You can't really multiplex the IV-9, being a tungsten filament bulb it's just too slow and either flickers, or is too dim. Each channel therefore has its own driver and a set of four BCD drive lines, except V1, the highest clock digit, which needs only two. While using four 74LS47's and no latch or multiplexer uses up more Atmega pins, it is easier to understand when it comes to coding. Each tube having its own BCD lines, it can be directly addressed by toggling the pins High or Low.

On supply voltages, the circuit has two internally: +5 V from an ordinary 78L05 regulator (IC1) and +4.00 V (adjustable on P1) from a little switch-mode supply around the familiar MC34063

Figure 1.
Hey, the Numitrons have arrived all the way from Dot-ua Land. And a good read included free! Paid for in US$ using a credit card, and cleared by customs, the little tubes came all the way from the Ukraine.

Table 1. IV-9 Main Specifications (attempt @)

Type	IV-9 (ИВ-9)
Brand	Reflector/ Sovtec
Substitute(s)	-
Displayed symbols	7 segment, ds
Symbol height (length)	10 mm (0.400 ")
Overall dimensions	11 mm (0.433 ") dia. x 35 mm (1.38 ")
Pin diameter	0.5 mm (0.020 ")
Typ. (max.?) supply voltage	4.5 V
Typ. current per segment	19.5 mA
Base	TO-100, pin 1 in front
Socket	Njet. Wires—no socket needed

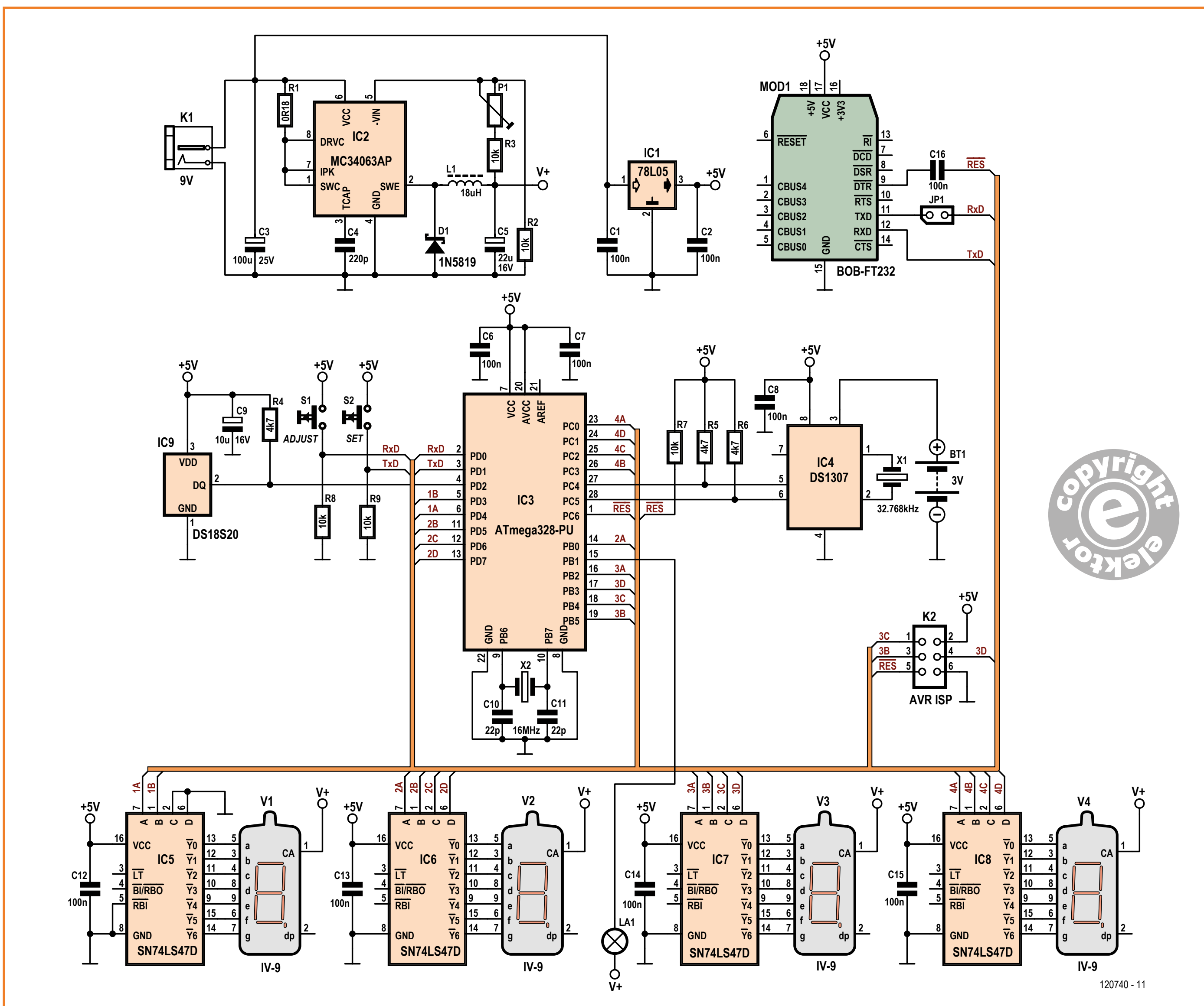


Figure 2.
Schematic of the Numitron Clock & Thermometer. Note that each Numitron V1-V4 has individual drive, rather than multiplexed.

copyright elektor

(IC2). The IV-9's filament voltage range is actually 3.15-4.50 volts and you want to set P1 to a level that best suits your viewing. Each filament draws about 19.5 mA. The whole clock requires a DC input of 7.5 V unregulated at about 500 mA. The internal supply voltages were chosen as 5 volts and 4 volts (3.00 V-4.50 V), and not 3.3 V only, because

- the higher voltage allows the Atmega to work at 16 MHz;
- 3.3 V is too low considering the 74LS47D LS-TTL 'open collector' outputs powering the filaments from the V+ do not quite pull down all the way to 0 V when On.

The heart of the clock is an ATmega328P, which is the chip used in the UNO and Duemilanove series of Arduino boards. If purchasing or using a chip with an Arduino Boot Loader pre-installed the Duemilanove version is preferred.
Programming of the ATmega328P is via the standard 6-pin ISP header and AVR Studio, or via the optional Elektor FT232 BOB, the BOB being used by the Arduino IDE software (V1.0 or greater) with the Duemilanove Bootloader pre-programmed on the ATMega. As far as the Arduino IDE knows it's talking to a standard 'Arduino Duemilanove w/ ATmega328' Board.
The real-time clock (RTC) is a DS1307, and the temperature sensor, a DS18B20—both components are well supported in the Arduino and AVR communities. Their 'One-Wire' data is read by the ATMega controller on the last two available port lines, PD2 (temperature) and PC4 (RTC).
The expected ISP (in-system programming) connector is available (K2) to allow the ATmega chip to be programmed without removing it from its socket.
MOD1, an Elektor BOB-FT232 (break out board), is optional. It provides USB connectivity and is useful to have if you are keen on working on the clock software within your Arduino environment.

The software

The source and hex files for the ATmega328 are available from the Elektor website [1]. Ready programmed ICs are also available: Elektor Store # 120740-41.

The code is written to loop every 500 ms (½ second)—this is most obvious by the indicator bulb La1 that flashes on and off once per second, plus this delay gives time for the RTC to recover between reads. If you read too often you get rubbish back! BTW La1 is a "grain of wheat" (GOW) style miniature light bulb as used in dollhouses and model railways.
On each cycle the code starts by checking if the SET button (S2) has been pressed and if not then updates the indicator bulb. At this point the RTC is read and, using some basic Integer division math, the hours and minutes are each split into Tens and Units, each value is then sent to the correct Numitron digit.
For simplicity of the code each digit has its own routine but they all work in a similar way with the only exception being the 'Hour Tens' that only has two binary bits (on IC5/V1). Finally using another useful math trick (Bitwise AND) we then convert the decimal number into Binary ready for the 74LS47's. More info on Bitwise operations may be found at [2]. While it looks a little long winded it's actually very simple, and a really useful way to convert Decimals to individual Binary digits.
We have two extra functions to add value and interest. The first is called on the 10-second and 30-second points, reading and displaying the temperature in degrees Celsius. This works as above using Mod and Bitwise math to take the reading, convert it to binary and display it via the 74LS47s, the exception to this being the lowest digit that is sent a decimal '10' which makes the 74LS47 on that digit (V4) to display a 'c' thus completing the display to show for example '22.5c'. The 74LS47 is unable to display 'f', hence a Fahrenheit readout is not supported.
Lastly on the 50th second the showdate() routine is called that displays the date, month and then year.

The code is very basic and I'm sure you could find other functions for the clock (Egg timer?), so why not join the project at Elektor.Labs [4] and show us your code/updates/tweaks, or just share your ideas and suggestions.

Build it, use it

Being SMD-free the project should be easy to build at home, at school, at TechShop San Jose CA, or in a small lab. The circuit board shown in **Figure 3** tallies with the Component List. **Figure 4** shows the benchmark you're up against as far as tidiness, component selection and proper soldering are concerned. Initially, set preset P1 for +4.00 volts on the V+ rail.

COMPONENT LIST

Resistors

R1 = 0.18Ω 3W
R2,R3,R7,R8,R9 = 10kΩ
R4,R5,R6 = 4.7kΩ
P1 = 4.7kΩ preset, top adjust

Capacitors

C1,C2,C6,C7,C8,C12,C13,C14,C15,C16 = 100nF
C3 = 100µF 25V radial
C4 = 220pF
C5 = 22µF 25V radial
C9 = 10µF 50V radial
C10,C11 = 22pF

Inductors

L1 = 18uH choke, 3.4A, 0.036Ω (Panasonic type ELC10D180E)

Semiconductors

D1 = 1N5819
IC1 = 78L05
IC2 = MC34063
IC3 = ATmega328-PU, programmed, Elektor Store # 120740-41 [2]
IC4 = DS1307
IC5,IC6,IC7,IC8 = 74LS47
IC9 = DS1820

Miscellaneous

V1,V2,V3,V4 = IV-9 Numitron tube
JP1 = 2-pin pinheader + jumper cap, 0.1’’ pitch
X1 = 32.768kHz quartz crystal
X2 = 16MHz quartz crystal
MOD1 = BOB-FT232R (optional), Elektor Store # 110553-91
Bt1 = CR2032 battery
S1,S2 = pushbutton, SPST, chassis mount
K1 = power jack, 2.1mm, PCB mount
K2 = 6-pin (2x3) boxheader
La1 = 5V 300mW light bulb, GOW
PCB # 120740 [2]

www.elektorpcbservice.com

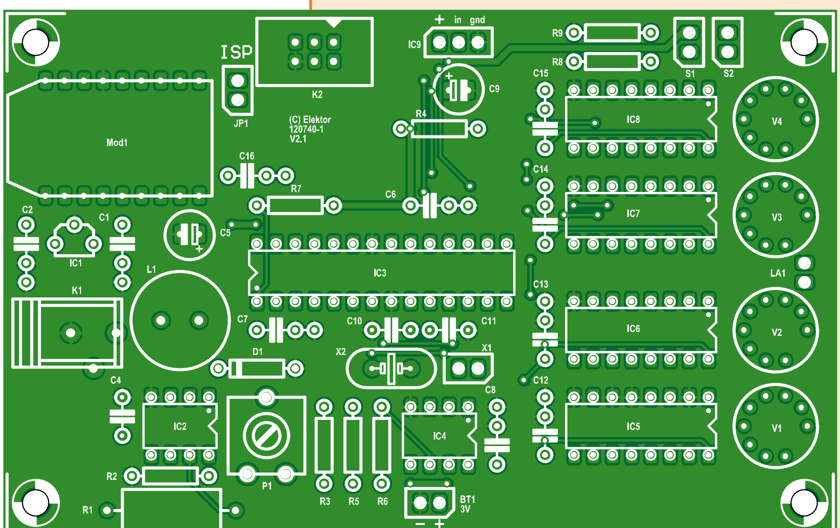


Figure 3.
The double sided printed circuit board designed for the project by Elektor Labs.

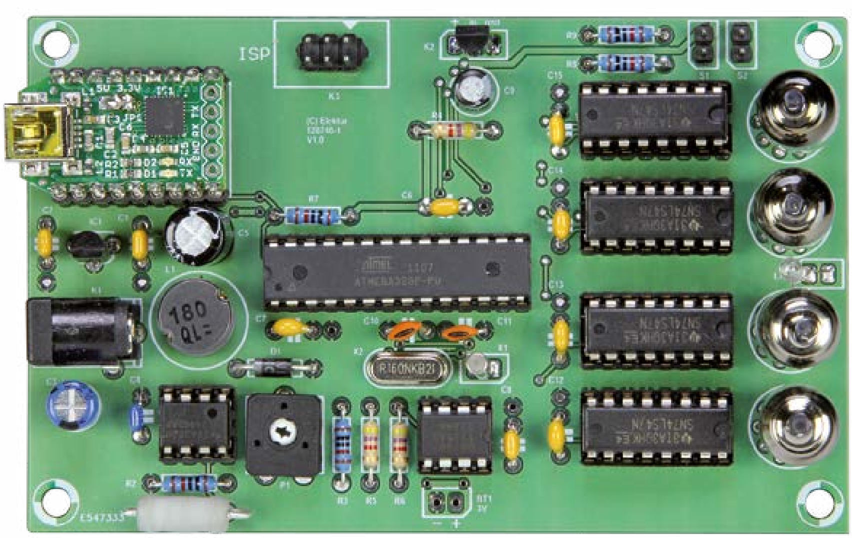


Figure 4.
The assembled and tested board, ready for use.

To set the clock, jumper JP1 must be removed (‘Run’ mode). Pressing the SET button S2 will cause the setclock() routine to be called, firstly displaying the Minutes. Pressing the ADJUST button S1 counts up by 1 on each press or keeps adding 1 every ½ second if holding the button.

Small delays in the code de-bounce the buttons and make everything smooth.

Once the minutes are correct, another press on SET switches to the Hours, then Day, Month and finally, Year. On completion of this cycle the new values are written to the RTC and the code returns to the main Loop.

(120740)

Internet Links

[1] Nixie VU Meter, Elektor November 2012, www.elektor.com/110744

[2] www.elektor.com/120740

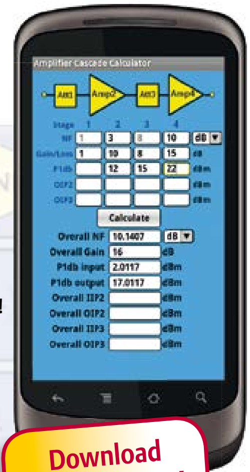
[3] http://playground.arduino.cc/Code/BitMath#bitwise_and

[4] www.elektor-projects.com/project/numitron-arduino-clock-and-thermometer-120740.12460.html

Elektor RF & Microwave Toolbox

for Android

The RF & Microwave Toolbox contains 55 calculation and conversion tools for RF, microwave and electronics in general. Whether you are an RF professional, radio-amateur, astronomer or hobbyist, this app puts some of the most important tools right at your fingertips. Literally!



Highlights:

- Amplifier cascade (NF, Gain, P1db, OIP2, OIP3)
- Field intensity and power density converter (W/m2, V/m, A/m, Tesla, Gauss,dBm, W)
- PCB Trace calculator (impedance/dimensions)
- PI and T attenuator
- Antenna temperature (Kelvin)
- EMC (EIRP, ERP, dBµV/m)
- Filter Design (Butterworth, Chebyshev, prototype)
- And much more

Download your app now!

ANDROID APP ON Google play

Further information at

www.elektor.com/rf-app

EMBEDDED SYSTEM & ELECTRONICS DESIGN SERVICES

50 Man Years of Professional Experience, From INDIA

Fastest "Proof of Concept" Demonstration

Services Offered

- Microprocessor / DSP Selection
- Hardware Software Co-Design
- Functional Pseudo Code / Firmware
- C / C++ Code
- PCB Design & Population
- Prototype Making
- Design Verification & Validation
- Maintenance of Embedded System
- Post Design Production Support

Recently Completed Projects

- Digital Engraving Machine Controller - dsPic
- Elevator Controller - ARM 7
- AMF Controller - dsPic
- Automobile - Water Separator in Diesel Pump - AtMega
- Industrial PA System - AtMega
- Weighing Scale - Nuvoton

100% Customer Satisfaction

SHIDORE MICROSYS ELECTRONICS PVT. LTD.

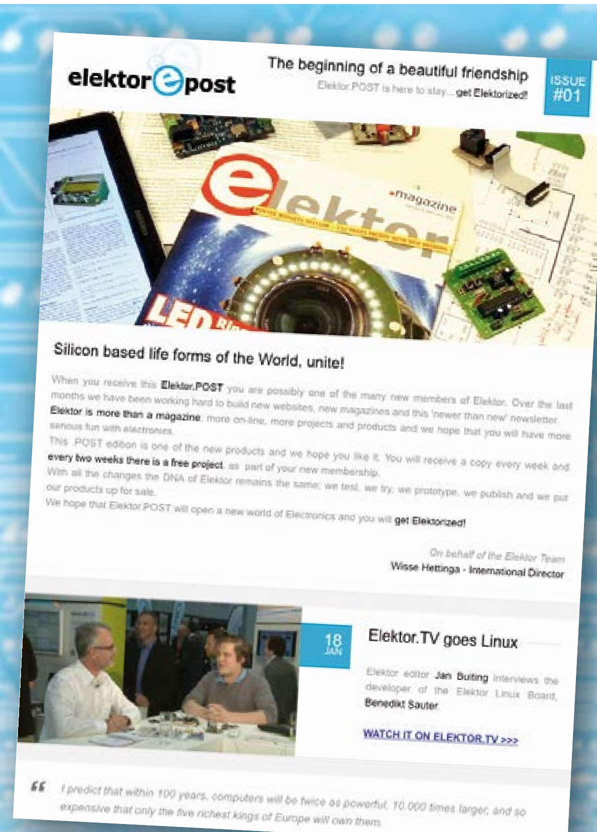
AN ISO 9001 COMPANY

Plot No. 15, Bhusari Colony, Paud Road, Kothrud, Pune 411 038. (India)
Tel. : +91 20 25283572 /94. Email : info@shidoremicrosys.com
www.shidoremicrosys.com

Take out a FREE membership to Elektor.POST

- The latest on electronics and information technology
- Videos, hints, tips, offers and more
- Exclusive bi-weekly project for GREEN and GOLD members only
- Elektor behind the scenes
- In your email inbox each Friday

elektor



elektor post

The beginning of a beautiful friendship

Elektor.POST is here to stay... get Elektorized!

ISSUE #01

Silicon based life forms of the World, unite!

When you receive this **Elektor.POST** you are possibly one of the many new members of Elektor. Over the last months we have been working hard to build new websites, new magazines and this 'newer than new' newsletter. **Elektor is more than a magazine**, more on-line, more projects and products and we hope that you will have more serious fun with electronics.

This .POST edition is one of the new products and we hope you like it. You will receive a copy every week and **every two weeks there is a free project**, as part of your new membership.

With all the changes the DNA of Elektor remains the same: we test, we try, we prototype, we publish and we put our products up for sale.

We hope that Elektor.POST will open a new world of Electronics and you will **get Elektorized!**

On behalf of the Elektor Team
Wisse Hettinga - International Director

18 JAN

Elektor.TV goes Linux

Elektor editor **Jan Buiting** interviews the developer of the Elektor Linux Board, **Benedikt Sauter**.

WATCH IT ON ELEKTOR.TV >>>

I predict that within 100 years, computers will be twice as powerful, 10.000 times larger, and so expensive that only the five richest kings of Europe will own them.

Register today at www.elektor.com/newsletter

Android elektorcardioscope

(Part 3)

Construction, adjustments, and operating instructions

Here now is the last installment of the article series. As far as we know nothing comparable exists commercially, neither in terms of performance nor price.

By **Marcel Cremmel** (France)
in co-operation with
Raymond Vermeulen (Elektor Labs)

Construction: a few comments

For constructing the elektorcardioscope (also referred to as the ECG interface), half the work has already been done for you, as the module is available assembled, tested, and ready-to-use [8]. All the components for the circuit published in the first article [9] fit on a modestly-sized board (100×60 mm) (Figure 19) that's been designed to fit (without needing screws!) into a common case with a battery compartment (see component list). The only thing to be done for the interface board itself is to connect up the two supply cables. If you fancy the challenge of building the PCB yourself instead of buying the ready-to-use module, only attempt this if you have solid experience in the subject. The regulator IC12 must be an MCP1640BT (lower noise version). It is possible to use a different Bluetooth module from the one suggested (Figure 4c in the first article), just as long as it uses the SPP protocol. The advantage of this one is that it offers the possibility of reset (S3). By not fitting or by unsoldering the ferrite bead interference suppression inductor L3, it is easy to test the various parts of the circuit separately. The pin header on the underside of the PCB is optional. It allows you to reprogram the microcontroller "in circuit" (ICSP) using a compatible tool (e.g. PICkit2 or PICkit3 from Microchip). The functions of the three buttons and two LEDs is shown in the front panel drawing (Figure 20): S1 = Off (and microcontroller reset), S2 = On, S3 = Bluetooth Reset, D3 = Data Transmission to the Android Terminal (TX) and D4 = Bluetooth Module Status (BT).

You will still need to make yourself a connector for your electrodes, and maybe the electrodes themselves – we'll come back to this later. First, we're going to get our ECG interface up and running. Once power has been applied from the two AA cells, all you have to do is briefly press S2 and the assembled module will show signs of life by flashing D4 (2 Hz), showing that the Bluetooth module is able to be identified.

Thanks are due to Aurélien Moulin, I2 student at ESEO, Angers (France), trainee at Elektor Labs, for his active participation in testing and debugging the first few assembled modules supplied by the elektorPCBservice.

COMPONENTS LIST

Resistors *(Default: SMD 0603 shape, 1%)*
R1,R13,R15,R18,R19,R20,R33,R34,R54 = 10kΩ 0.25W
R2, R12 = 3.3kΩ
R3 = 523kΩ
R4 = 300kΩ
R6 = 150Ω
R7–R11,R23 = 1kΩ
R14,R53,R55,R56,R57,R59–R62 = 100kΩ
R5,R16,R17,R35 = 1MΩ
R21,R36,R37,R38 = 330kΩ
R22 = 100Ω
R24,R65 = 390kΩ
R25,R29 = 47MΩ 5%
R26,R30 = 10MΩ
R27,R31 = 2.2MΩ
R28,R32 = 470kΩ
R39,R40,R49–R52 = 47kΩ
R41,R42,R45,R46 = 28.7kΩ
R43,R44 = 1.4MΩ
R47,R48 = 45.3kΩ
R58 = 9.1kΩ
R63,R64 = 27Ω
P1,P2 = 5kΩ 20% adjustable (Vishay TS53YJ502MR10)
P3 = 2kΩ 20% adjustable (Vishay TS53YJ202MR10)

Capacitors
Default: SMD 0603
C1,C13 = 33µF 6.3V, tantalum (case A)
C2,C4,C7,C9,C12,C14 = 10µF 6.3V, X5R
C3,C5,C6,C8,C10,C15,C21,C22,C23,C24,C39,C40,C41,C42,C43,C44 = 100nF 25V, X7R
C11,C16,C25,C26,C31–C38 = 1µF 10V, X5R
C17–C20 = 1nF 50V, X7R*
C27, C29 = 470pF 50V, 5%,NP0
C28, C30 = 47nF 25V 5%, X7R

* C18, C19, C20 are 1 nF 50V, not 100 nF 50V as shown in the schematic.

Inductors
L1 = 4.7µH 20% 0.5A (Wuerth 744032004)
L2–L9 = ferrite bead, 30Ω @ 100MHz (Murata BLM18PG330SN1D)

Semiconductor
D1, D2 = BAV99S
D3, D4 = LED, red, (PLCC-4)
T1 = PSMN6R5-25YLC NMOSFET
IC1 = PIC24FJ32GA002-I/SS, programmed, Elektor # 120107-41
IC2 = TPS60403DBVT
IC3–IC7 = TLC2252AIDRG4
IC8 = LMC6482AIMX/NOPB
IC9 = DG4053 AEQ-T1-E3
IC10, IC11 = CD74HC4052PW
IC12 = MCP1640BT-I/CHY
IC13 = LTC1981ES5#TRMPBF

Miscellaneous
K1 = 5-pin pinheader, 0.1'' pitch
K2 = 6-pin pinheader, 0.1'' pitch
K3 = 2-pin pinheader, 0.1'' pitch
MOD1 = Bluetooth module, Roving Networks/Microchip type RN-42
S1,S2,S3 = pushbutton, Omron type SPNO B3FS-1052) with cap, Omron type B32-2010
Case, Pactec PPL-2AA
PCB, Elektor # 120107-1 or
Assembled module, ready for use, Elektor # 120107-91

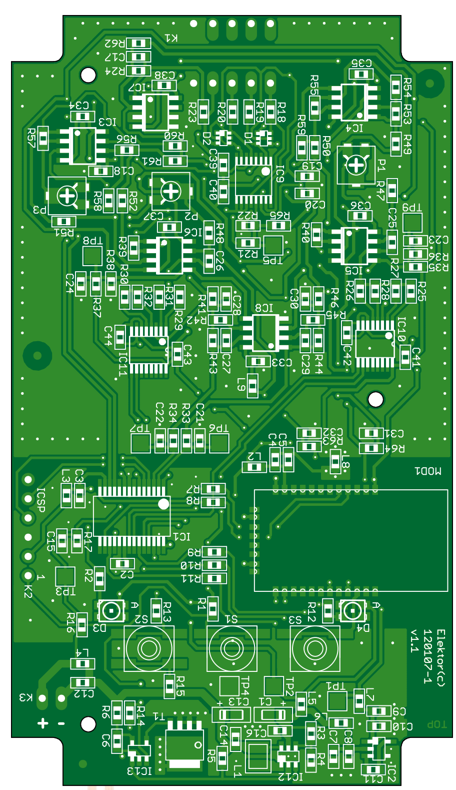


Figure 19. The ECG interface fits into the palm of your hand. Connoisseurs will appreciate the good separation in the routing between the analog and digital parts of the circuit, vital here.

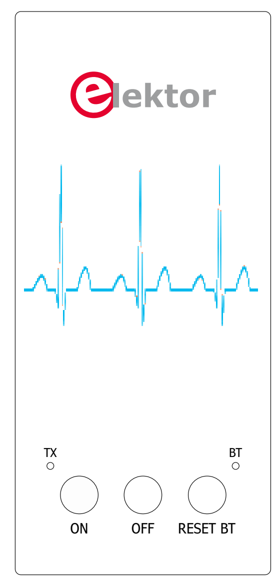


Figure 20. Front panel design with 3 buttons and 2 LEDs.

The numbering of the figures and links continues from the two previous articles.

Figure 21. Menu for selecting the Bluetooth link to be established between the ECG interface and a nearby Android terminal.

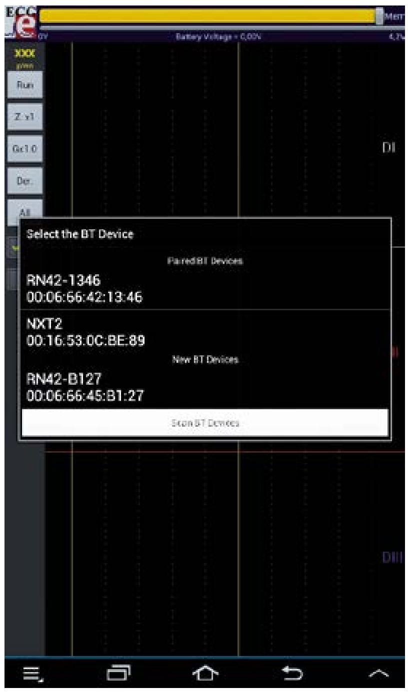
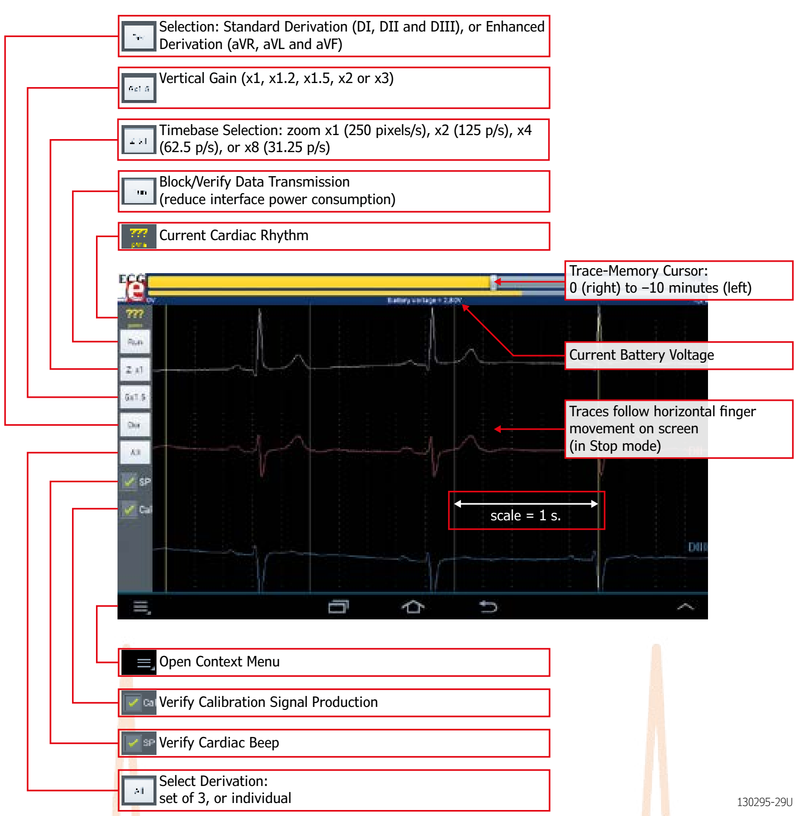


Figure 22. Overview of the touch-screen interface functions.



Android application

Installation is conventional: download the Android application package (file ANDROECG.apk) [10], copy it into the root of the Android terminal and select it using the File Explorer. It will be installed automatically (assuming you have already allowed installation of non-Market applications in your Android terminal's Security menu).

You can then place a shortcut wherever you like. If the terminal's Bluetooth interface is not enabled when the application is run, the application tells you. Of course, you must enable it.

At this stage, the BT radio link between the ECG interface and the Android terminal has not yet been established. To do this, you must open the menu and request Paired BT Devices (Figure 21). The connection should be established straight-away after you have chosen the ECG interface's Bluetooth module. In this case, the graph starts to scroll and the BT (State) LED on the interface stays lit. At the first connection, you'll need to enter the PIN code (here: 1234) for the BT module. The BT peripheral is now registered in the Android terminal and you won't be asked for the PIN again. The interface's microcontroller is not involved in this protocol. The MAC code printed on the BT module and displayed on the Android terminal may make it easier to identify the ECG interface if the list of BT devices is long.

If no BT connection is established within 5 minutes, the interface will be powered down automatically.

It won't take you long to get to grips with the menus and functions of the elektorcardioscope software on the Android terminal, thanks to the intuitive "Instructions for use" in Figure 22. There's also a demonstration video [11].

Now all that remains is to move on to adjusting the interface, which involves just two simple operations: setting the common-mode rejection ratio (CMRR), and balancing the gains.

Setting the CMRR for each channel

The first adjustment consists in optimizing the CMRR for each differential amplifier with the help of a function generator. Start by constructing the calibration accessory on the left in Figure 23 (BNC + pin header with no resistors). Plug it into socket K1, making sure you get it the right way round: pin 1 is on the right when you are holding the interface with the buttons toward you. The

RA, LA, and LL inputs are connected together so as to inject a common-mode signal via pins 1, 2, and 3 of the calibration jig; its pin 4 is connected to the body of the BNC socket and hence to the ground of the LF function generator, while pin 5 (the ECG interface ground) is left floating.
Adjust the generator to obtain an AC sinewave at 50 Hz with an amplitude of 1 V. Power up the ECG interface and run the application ANDROECG. Establish the Bluetooth connection and observe the DI and DII leads at maximum amplification. Then adjust P1 and P2 to reduce the peak-to-peak amplitudes as much as possible. They should be barely visible at ×10 amplification (Figure 24).
In the absence of an LF function generator, you can work as follows:

- touch the interface GND with a finger on one hand;
- with a finger on the other hand, touch the common point of the three electrodes RA, LA, and LL: by so doing, you are injecting a common-mode signal picked up by your body from the 50 Hz power line;
- observe the DI and DII signals on the Android terminal;
- using your third hand (maybe you'll need a guinea-pig!), adjust P1 and P2 to obtain the flattest possible curves.

Balancing the gains

The total gain of amplifiers in each channel must be identical, as the DI and DII signals are used for calculating the other leads (see Leads box [9]). A signal generator is recommended for this adjustment, with an attenuator, as the signal to be injected must be very low: 1.4 mVpp (on right in Figure 23, BNC + pin header + resistors). This attenuator, formed from two 1.5 kΩ and 150 kΩ resistors connected as shown in Figure 25, must be inserted between the function generator and the ECG interface in socket K1: pin 1 is on the right when you are holding the interface with the buttons toward you.
Set the generator to produce a 1 Hz sinewave at an amplitude of 140 mVpp. If not already done, connect your Android terminal to the ECG interface and observe the DI, DII, and DIII leads using ×2 amplification. You should observe 1 Hz sinewaves on DI and DII as seen at image 1 in Figure 26. Now adjust P3 to minimize the trace for the DIII lead. In point of fact, the Android terminal calculates DIII = DII − DI: DIII should

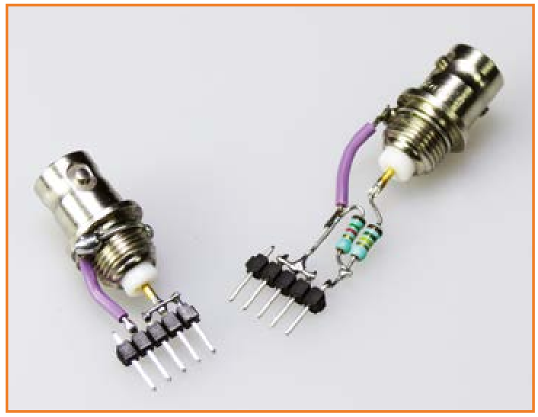


Figure 23. You'll only need to turn on your soldering iron to build these two calibration jigs.

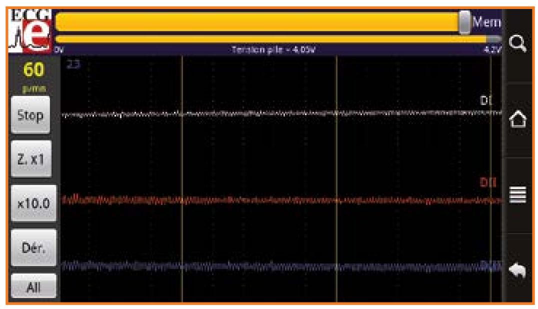


Fig 24 = The amplitude of residual 50 Hz components must be minimized.

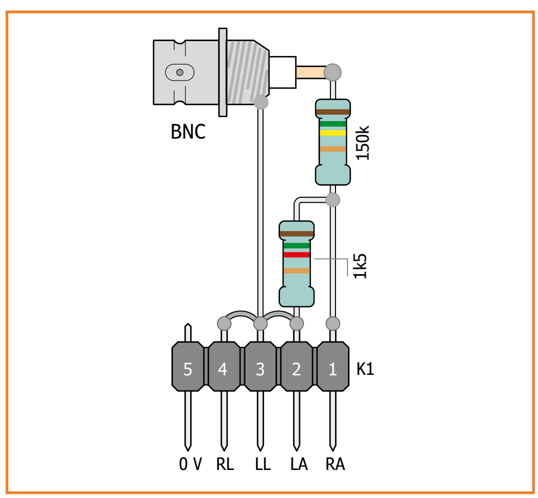


Figure 25. This adjustment jig divides the injected calibration signal by 100.

ATTENTION

Manufacturing and using medical devices are governed by national and international laws [14]. The Elektorcardioscope has no medical approval and hence is not intended for professional use.
In order to comply with protection class III, it must be battery powered only, and may be used only for the purpose of personal study and experimentation. Under no circumstances shall the author of this article or the Publishers be able to be held liable for the consequences of using this interface.

The gain balancing requires injecting the highest possible level without driving the amplifiers into saturation. Before the adjustment, ensure that the sine waves displayed on the terminal are free from saturation effects or distortion (particularly DI). Reduce the signal level if necessary. In principle, saturation should not occur as there is a small drive margin left when injecting 1.4 mV.

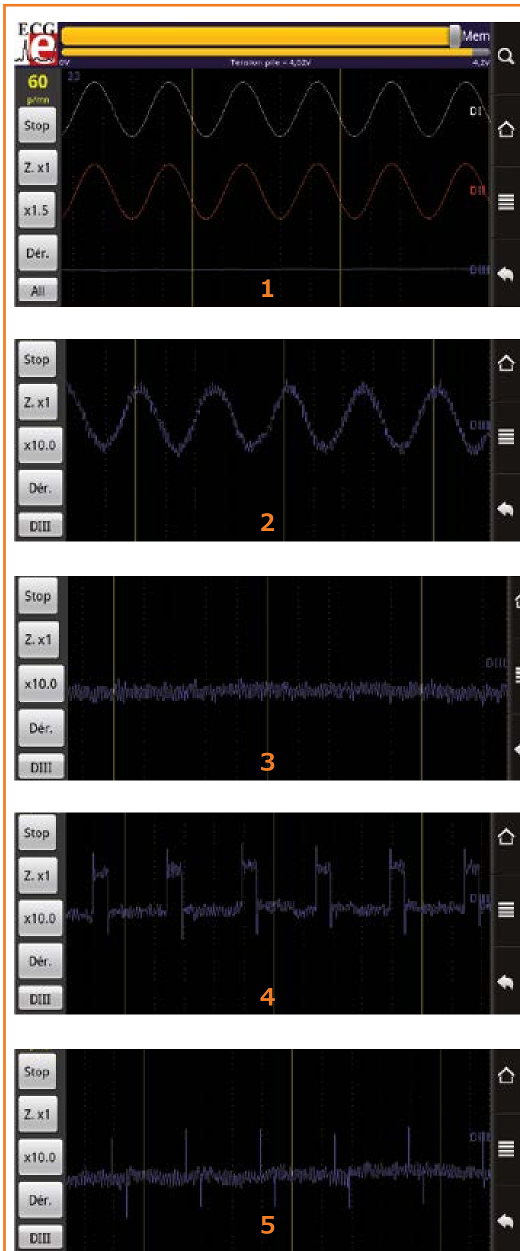


Figure 26. These readings will help you balance the amps IC3 and IC4 (DI and DII channels): the trace for the DIII lead must be minimized.

Figure 27. Every minute, the interface produces a 2 Hz/1 mV CAL signal for 10 s, which can be used for balancing the gains (see Figure 26).

be null when DI = DII. The amplification can be increased to obtain greater sensitivity, and display only DIII. Images 2 and 3 represent DIII before and after adjustment.

In the absence of a signal generator, wait for the periodic injection of the CAL calibration signal at the ECG interface input (Figure 27, see also Figure 9 in the second article) and adjust P3 to obtain as little residual CAL signal as possible on DIII. Images 4 and 5 (Figure 26) show the results before and after adjustment; note that it isn't possible to eliminate the fine spikes seen in image 5.

Note: impatient users force an immediate injection signal by un-ticking and again ticking the "Cal." Button.

The elektorcardioscope is now operational and you can use the various touch buttons on the Android terminal screen (Figure 22) to choose the graphs, amplify the amplitudes, change the timebase, and move around within the trace memory. But to be able to observe your electrocardiograms on your Android terminal's screen, you'll need to equip yourself with some electrodes—and find a willing patient.

Electrodes

The electrical signals we're going to be picking up with the help of the electrodes are infinitely weaker than the ones we're usually familiar with. Hence it will only be possible to obtain a good electrocardiogram using good electrodes, properly positioned, and correctly wired. As a reminder, on the electrodes connector (Figure 28a), the ground is on the left when you are holding the ECG interface with the buttons toward you. At the patient end, get into the habit of adhering to the color code [9]:

- red → RA (right arm)
- yellow → LA (left arm)
- green → LL (left leg)
- black → RL (right leg)

If you don't want to go to the expense of buying four commercially-available electrodes – the clamps, very handy with children, are not cheap – you can easily make some yourself. Any electrode with its connecting cable is also a great antenna, and here you'll have four: so it's essential to use shielded cable, to minimize the influence of

unwanted signals. The shield is connected at the interface end only; at the electrode end, it must be insulated so as to avoid any contact with the skin. Watch out! Although audio cable is electrically suitable, it is mechanically fragile. Strain reliefs reduce the risk of failure (Figure 28b).
The 4 mm banana plugs allow you to use commercially-available accessories (Figure 29) like banana to spring-terminal adaptors [12] or screw-fit bananas (RS Components ref. 641-8053).
If you've still got some nickel-alloy coins around you can also produce these accessories yourself more cheaply. The only tricky part if to solder a 4-mm socket onto the coin (Figure 30), which will then be held in place on the wrists and ankles by four elastic bracelets (suspender elastic + Velcro strip). You can also use rings cut from a motorbike or scooter inner tube. Fit the electrodes to the wrists and ankles, if possible using conducting gel, which improves the quality of the ECGs by significantly reducing noise and contact voltages.

Saving and playing back ECGs

These operations are simple, no options are offered: you save or playback the 10 min of samples for the DI, DII, and DIII leads. The index of the most recent sample (the current one) within the circular ECG memory is also saved in order to be able to get back the same appearance of the traces in playback (latest sample represented on the right of the screen: see Figure 17 in the second article).
The save and playback operations are offered in the menu. A window offers you the choice between an existing file and a new file, for which you just have to enter the name (Figure 18 in the second article [13]).

Other functions

The "CAL" checkbox gives you the choice to periodically inject a calibration signal (Figure 22) in place of the ECG signals. Cardiologists are very familiar with this calibration signal, with an amplitude of exactly 1 mV, which is used as a reference against which to compare ECGs.

The Android application calculates the cardiac rhythm using an algorithm based on the derivative of the DI signal. This algorithm can sometimes go wrong, together with its display on the screen, and an audible 'beep' generated. As this noise can become annoying, it can be turned off using the SP button.

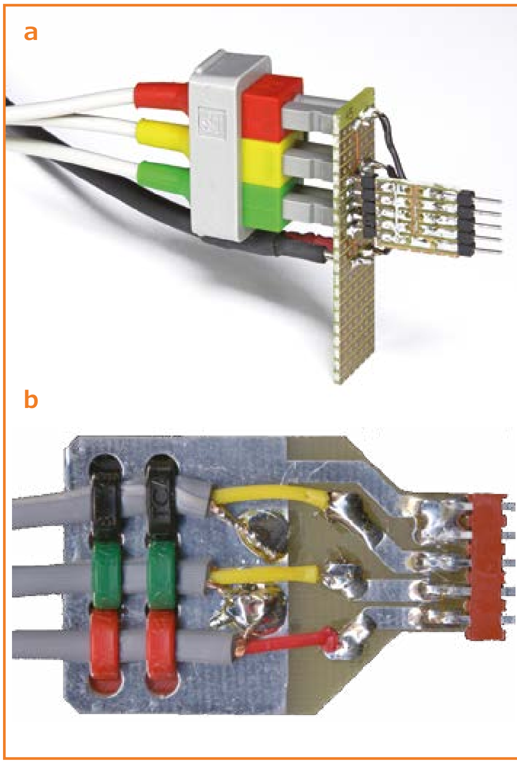


Figure 28. We made up this adaptor for a commercially-available electrode connector (Figure 28a). Note the cable clamps on the robust home-made connector (28b) produced in 2006 for the Gameboy version, which used one less electrode.

Future functions

A project like this is constantly evolving. In its current state, the only display application available is the Android version. However, Elektor would be delighted to publish versions for iPhone,

Figure 29. Commercially-available accessories are not cheap, but do look the part.

Figure 30. Satisfactory electrodes can be obtained using nickel-silver coins, as long as you can manage to solder to them a 4-banana socket.

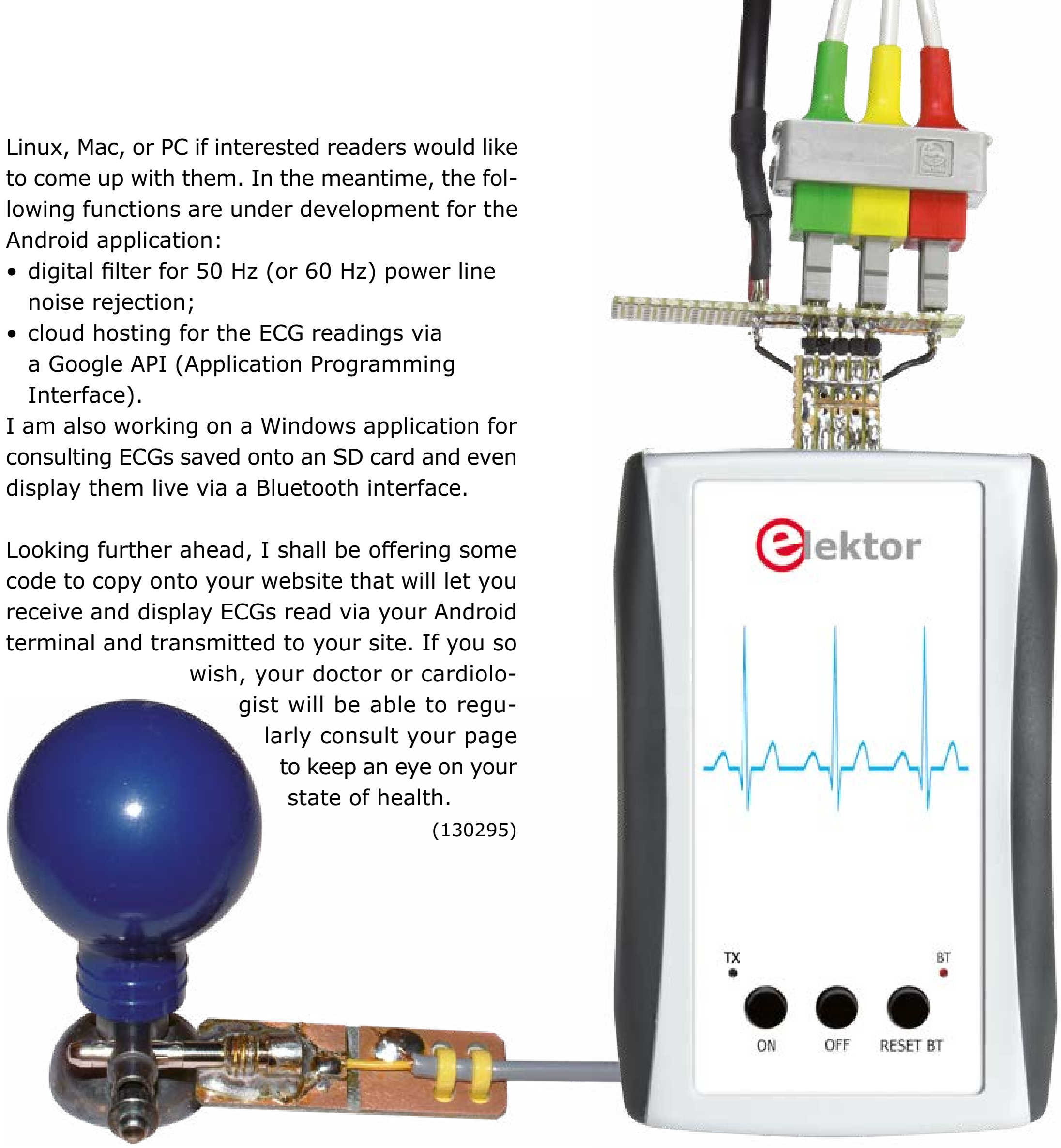
Linux, Mac, or PC if interested readers would like to come up with them. In the meantime, the following functions are under development for the Android application:

- digital filter for 50 Hz (or 60 Hz) power line noise rejection;
- cloud hosting for the ECG readings via a Google API (Application Programming Interface).

I am also working on a Windows application for consulting ECGs saved onto an SD card and even display them live via a Bluetooth interface.

Looking further ahead, I shall be offering some code to copy onto your website that will let you receive and display ECGs read via your Android terminal and transmitted to your site. If you so wish, your doctor or cardiologist will be able to regularly consult your page to keep an eye on your state of health.

(130295)



The author Marcel Cremmel is a qualified Electrical Engineering instructor, electronics option, on the Higher Diploma in Electronics course at the Louis Couffignal College in Strasbourg. Website: http://electronique.marcel.free.fr/ Email: marcel.cremmel.llc@free.fr

Internet Links

[8] www.elektor.com/120107

[9] Part 1: Elektor July & August 2013, www.elektor.com/120107

[10] ANDROECG.apk application: www.elektor.com/130295

[11] Demo video: in production, please check the “elektorim” channel on Youtube.

[12] Electrodes: www.praxisdienst.fr/fr/home/

[13] Part 1: Elektor September 2013, www.elektor.com/130227

[14] Legal guidance: http://homeusemedical-devices.com/humd.html

Banana to Snap-On adapters:

http://goo.gl/5WWYFu or

http://www.praxisdienst.com/en/Diagnosis/Specialised+diagnosis/ECG+devices+and+accessories/oxid+oxid/Press+-stud+adapter+for+ECG+red.html

Limb clamps:

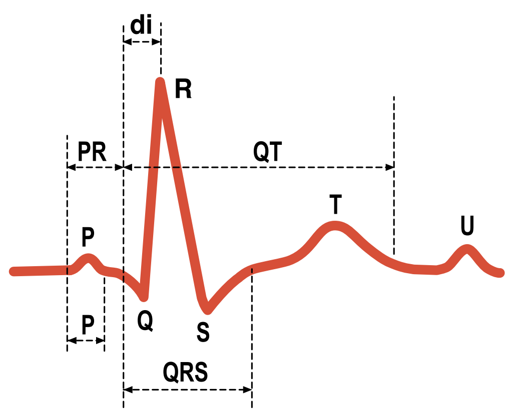
http://goo.gl/bZIKXf or

http://www.praxisdienst.com/en/Diagnosis/Specialised+diagnosis/ECG+devices+and+accessories/oxid+oxid/Limbs+-Clip+electrode+Adult+red.html

A normal electrocardiogram

Rhytm and Rate

The normal resting heart rate is between 50 and 100 beats per minute. Below 60 (sometimes, 50) is *sinus bradycardia*, over 100 is *sinus tachycardia*.



Atrial depolarization: P wave

Normal duration is less than or equal to 0.1 s.
Its normal amplitude is less than or equal to 0.25 mV, i.e. ¼ of the amplitude of the CAL calibration signal.
The P wave is generally maximum in DI, DIII, and aVF. The P wave is always positive in DI and DII and negative in aVR.

PR or PQ interval

The normal duration of the PR interval is between 0.12 and 0.20 s. It is measured from the start of the P wave to the start of the QRS complex. It corresponds to the time taken to conduct the influx from the atrium to the ventricles. This may reduce when the heart rate increases with exertion. Above 0.20 s, it indicates a problem with atrioventricular conduction.

QRS amplitude

In the front leads, the amplitude is very variable. With an amplitude lower than 0.5 mV (½ CAL) in all of these leads, we speak of micro-voltage.

Duration of the QRS complex

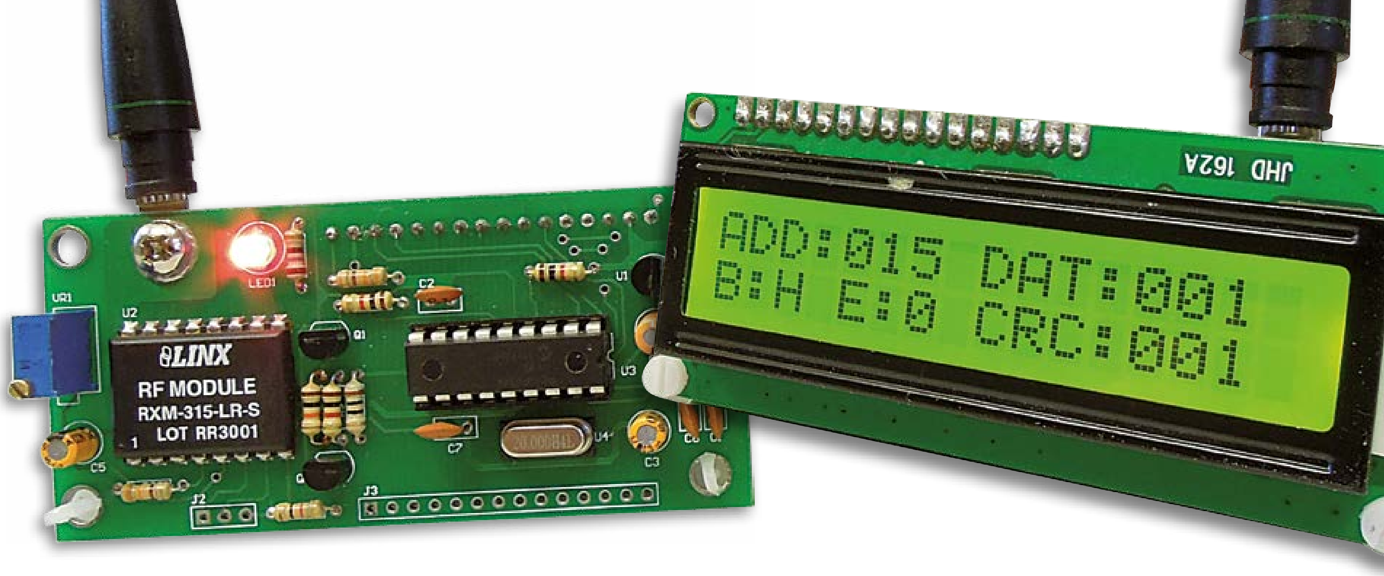
This is on average 0.08 s; it must be below 0.12 s. Above this, it is most often an asynchronism in the depolarization of the two ventricles, associated with a problem with intra-ventricular conduction.

Ventricular repolarization: ST-segment – T wave – U wave

The ST-segment separates the QRS complex from the T wave. It starts at the end of the QRS.
The T wave is usually of low amplitude, asymmetrical with the rising slope lower than the falling slope, and in the same polarity as the QRS. It is normally positive in DI, DII, DIII, and aVF. A T wave that is biphasic or negative in DIII must be considered as physiological.
The U wave, inconsistent, follows on from the T wave. It has the same polarity, but a lower amplitude. Its significance is debated.
The QT interval (start of QRS, end of T) varies according to the heart rate. For a rate of around 60 bpm, the duration of the QT interval is around 0.4 s.

Modular RF Link using Manchester Code (2)

Part 2: Software



In the previous installment we elaborated on the hardware side of this project. Now it is time to go software. While the proper hardware design and board layout guarantees the correct radiation and reception of the RF signals, the software (sometimes referred to as firmware) plays a fundamental role in the reliability of the message being carried by the signal.

By
Marcelo Maggi (USA)

As mentioned in the first installment, the RF modules proper have no intelligence included, so if our radio communication system is supposed to have 'brains' it must be provided by the microcontrollers' software. In this respect, 'intelligence' includes the selection of the proper coding protocol, the link speed (bit rate), data organization and error detection/recovery mechanisms.

The software has been designed in a modular way with all the essential functions organized in 'drivers' that remain unchanged. The user has complete freedom to implement any routine in the main area. A call on the TX driver sends the information. On the receiver side, the RX driver also takes care of all the complicated tasks, returning just the useful data to the main program.

Manchester code

The Manchester Code has proven to be a basic but very reliable way to send data over a radio link. With this project speeds of up to 5,000 bps have shown good stability, being the top bit rate of our link; a lower speed option is available for added versatility. For transmission the data is organized in a macro structure called *frame*. An error correction byte is included. Its logic is quite simple: every bit is represented by a transition, instead of a logic level.

Figure 1 shows the logic behind the Manchester Code and the two conventions to represent each bit. We will be using the IEEE 802.3 definition: a logic '1' will be represented by a Low-to-High transition, while a logic '0' will be the opposite

(High-to-Low). The advantages of this form of coding are quite obvious:

- The signal clock is always present in each transition, no matter the bit sequence, and it is easily recovered at the receiver end.
- The average DC level of the signal is constant, around 50%.
- Using an OOK (On-Off Keying) RF link, like the one we use here, the average transmitting power is reduced. This is useful not only to save power in portable applications, but also to stay within the power limits mandated by local regulations for ISM band usage, while the peak power may be larger to increase range.

The Linx modules are rated up to 10,000 bps. While this speed is possible, our link will have a maximum speed of 5,000 bps. This is due to the coding, since each bit will have two logic levels; with a simple binary transmission the bit rate would double. However, given the advantages the Manchester Code, we sacrifice transmission speed for simplicity and reliability. At 5,000 bps an RF link over distances in excess of 600 ft (180 m) proved reliable.
A lower speed (2,500 bps) is provided to be used in case of very noisy environments or to extend the range. Linx claims that distances up to 3,000 ft (just under 1 km) are feasible with the proper hardware setup.

Features

- One transmitter, multiple receivers possible.
- Range up to 600 ft (180 m).
- Bit rate selectable between 2,500 bps and 5,000 bps (software).
- Error correction can easily be implemented.
- PCB artwork files and example software code available free of charge [2].

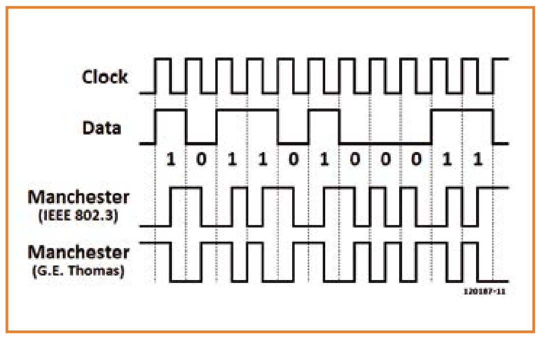


Figure 1.
Manchester Code logic and conventions.

Data format

The data to be transmitted is composed of three elements, 1-byte long each: *address*, *data* and *CRC*. The *address* byte is intended to indicate to which receiver the transmission is directed. Being 1-byte long, there would be 256 possibilities. In most situations this would be a waste of bits, so the most efficient way would be to use the *address* byte to indicate receiver *and* function:

- Upper nibble (4 bits) used to call the receiver (16 addresses).
- Lower nibble used to call the function within the receiver (16 options).

This way receiver number 5 can be ordered to run command number 9 using *data*, which may be turning on a servo mechanism and position it based on the contents of *data*.
In our example code, *address* will be used simply to indicate the receiver; it contains the number 15, just to show how it may be used. At the receiver side, when 15 is read from the *address* byte, *data* will be considered valid and processed by the code. In binary form 15 is represented 00001111, which is easily recognized on an oscilloscope. In Manchester code this is represented by the sequence 1010101001010101.
The *data* byte carries the actual information and can be a fixed value, such as an order to execute a defined task, or a variable one, like the output of an A/D converter. The example code will send a different *data* byte based on the status of pin B3 (RB3/CCP1, pin 9): if Low, *data* will contain a fixed '1'; if High, *data* will alternate between '0' and '1', using an internal timer to control the change. On the receiver side this will be used to turn LED D1 ON (1) and OFF (0). This is a very simple way to test the link and demonstrate how to use the *data* byte.
Despite the simplicity of the provided example code, take note of its abilities. By properly using the *address* byte it's easy to send more than one data byte at a time. The lower nibble may be used to indicate which byte you are sending, so the receiver can properly reassemble and complete the data. Sending more than one data byte

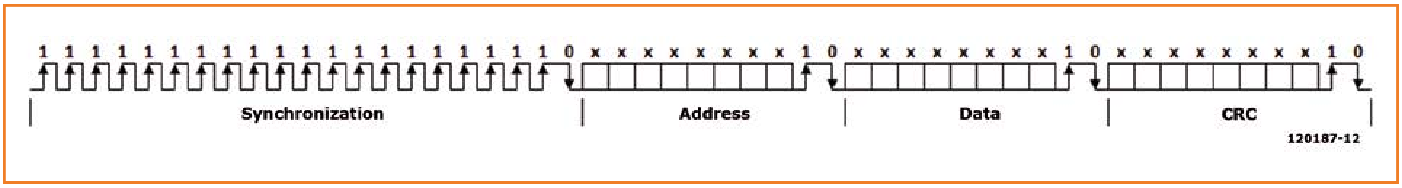


Figure 2.
Complete frame structure.

is much more efficient than sending one byte at a time. In this example we demonstrate the basics of the RF link, but a few changes in the code will allow any number of data bytes to be sent at once.

CRC, or *C*yclic *R*edundancy *C*heck, contains information useful to validate the previously received *data* byte (or bytes) and may provide elements to recover a missing bit. The various methods and algorithms generally available for creating a *CRC* byte are beyond the scope of this article, but a proper data stream should have one, so it is included. In the example code *CRC* is a copy of *data*. At the receiver end, if *CRC* is not equal to *data*, you know that something went wrong. However, there is no way to tell which one is wrong.

Composing a frame

Now to put the main three elements together in a macro structure called *frame*. Since every RF link is susceptible to noise, the receiver must contain elements to determine whether a frame is valid, and where it starts. Hence 'frame synchronization' is added preceding each frame. This synchronization is a non-changing sequence that cannot be emulated by any combination of the previously described three bytes.

We will use bit sequence 11111111111111111110.

A 'spacer' is added to the frame. This is a short sequence, just '1' and '0', added at the end of each byte. A synchronization conflicting sequence (when *address* and *data* are both 11111111 and *CRC* is 11110XXX) will thus be prevented: 11111111 10 11111111 10 11110XXX 10. The long sequence of '1's is broken by the spacer, thus guaranteeing that no combination will emulate the synchronism sequence. **Figure 2** shows a complete frame and its components in Manchester Code.

Bit rate

The transmission speed is selected at the transmitter (TX) side. It is predefined by the firmware, but can also be selected using input B2 (RB2/TX/CK, pin 8) of the microcontroller, accessible via K2 pin 11. If low, the bit rate will be 2,500 bps, if high, it will be 5,000 bps. The receiver detects the speed automatically. Since the detection is based on time measurements using the local oscillator as the reference, it is very important to keep the 20-MHz quartz crystal unchanged.

The transmitter needs to send the frames at certain intervals for the link to work. An isolated frame may not be received correctly, as may frames spaced more than 10 to 15 ms apart. The receiver must be 'awake' and have the right gain settings dialed in to properly receive and demodulate the signal. After 10 ms or more of inactivity the receiver may not be ready for an incoming frame.

The simple solution is to send the frames repeatedly at intervals of 10 ms or less. While this works, it has two major drawbacks:

- A microcontroller task requiring more than 10 ms is not allowed.
- Besides transmit power, local or national legislation in place for ISM band usage may prohibit the time a transmitter is active at a set frequency.

The solution is simple: set an internal timer in the main program and transmit two or three frames in a single burst, spaced 10 ms or less. Then, let the transmitter idle for a few seconds.

Note: The provided example, downloadable from [1], is only to show how the link works and to make a functional test. The main code should not be used in an actual device as is. Please check your local ISM band regulations so your final device is fully compliant. It is <u>your</u> responsibility.

Transmitter firmware (TX)

Both the fundamental transmitter and receiver routines have been grouped in a separate file, like PC hardware drivers. These routines contain the most complex parts of the code and can remain unchanged, no matter the application. The main program, with only a few lines to perform the previously described basic functions, is provided to show how the link works. Here is where you should write your own code.

All software has been written in C, using the CCS C compiler. Comments are added in the code pro-

RX driver

The *frame* is received and decoded in the receiver driver. All the complexity of the link resides in this driver, which incorporates comments to explain each section. The receive function *mc_rx*, called from the main program, calls the rest of the bit-gathering functions.

First the bit rate is detected: *baud_detect* counts the duration of two halves of one bit. All these activities are triggered by the rising edge of an incoming signal. The first bit of a *frame* is a '1'— in Manchester Code, low to high— so when the interrupt is triggered, the first half of this first bit is already gone. Therefore, *baud_detect* counts the second half of one bit and the first half of the next.
Isn't it enough to measure one half, since the bits are symmetrical? Yes and no. If we send data continuously, the receiver will be continuously active and the internal DC levels will be stable. Then the bits are fairly symmetrical. However, if the frames are spaced, let's say by 10 ms, the receiver may not be completely ready when each new frame arrives. As a consequence, the first bits of the frame may not be symmetrical, as shown in **Figure 3**. While the link still works, if we do not measure a complete bit to recover the bit rate there is a real possibility that we have the wrong figure, thus making any subsequent bit detection impossible. In our example code we send the frames 1 ms apart, so there is no major issue. However, the *baud_detect* function has been designed with those extreme cases in mind.
After the first half of the second bit *timer0* will contain the bit duration... sort of. Some simple math:

- A 20-MHz oscillator results in a 5-MHz instruction clock.
- The period of one cycle is 1/5,000,000 = 0.2 µs.
- *Timer0* increases every 0.2 µs×16 = 3.2 µs.
- One bit at 5,000 bps lasts 200 µs, so after one complete bit *timer0* = 200/3.2 = 62.5.

Being an 8-bit integer, *timer0* may either be 62 or 63. But this is not really relevant. By nature, radio signals will exhibit a phenomenon called jitter, random shifting of the edges (transitions) of the signal, shortening or stretching the bits. So *timer0* varies around 62.5. Any of these surrounding values are interpreted as the actual bit length as calculated. The program implements this by accepting a range, between 55 and 70. If *timer0* falls within this range, the bit duration is 200 µs, so half a bit will be 100 µs. This is exactly what is stored in the variable *semi*, returned by the function, and named *semiperiod*. For 2,500 bps *semiperiod* is 200 µs and the detection range for *timer0* is 118 to 133.
Now that the bit time is known, the detection of the bits is easy: Read the signal status, wait until it changes, read the next status, compare. Every bit will have a signal status change in the center of the bit period. Before reading the next bit we wait an interval *semiandjitter* to ensure that the next bit is read and not the tail of the previous. If status 1 is higher than status 2 (high to low transition, *in1* and *in2* in the code), the received bit is '0', otherwise it is '1'. Then this bit is added to a 32-bit variable (*three_byte_rx*), which is then shifted to the left so it is ready to accept the next bit. By carefully eliminating the spacers, *three_byte_rx* will contain *address*, *data* and *CRC* with the remaining 8 bits left empty.
Provided there is no error in the process, *three_byte_rx* is now called *ad_da_cr_rx* and moved to *frame_rx* in the *mc_*rx function. Its contents are then shifted 8 bits to the left (×256) and *semiperiod* is added. *Frame_rx* is now ready to be returned to the main program where it will be known simply as *frame*. In case of an error, the routine forwards this to the main program by sending an empty *frame_rx*.

A final note: as with the transmitter program, to avoid conflicts please check the names of the variables used here and do not to use them in the main program.

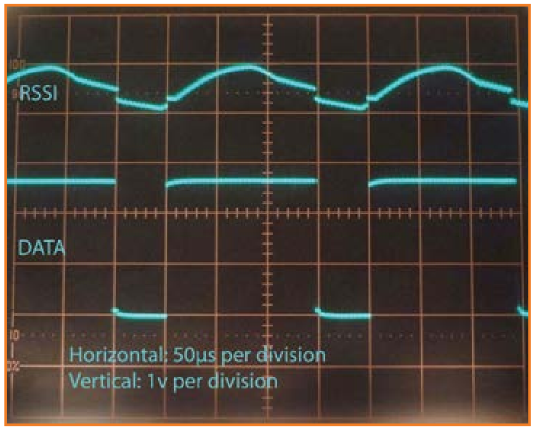


Figure 3.
Received signal with frames spaced 10 ms apart, as captured on an oscilloscope.

viding additional information. Let's examine the transmitter code, split into three files:

- Manchester_Link_TX.c—main program,
- Manchester_Link_TX.h—PIC setup,
- MAN_TX.c—transmitter driver.

The main program is quite simple. After the initial definitions the program enters an infinite loop, where pin B2 is read to set the bit rate, *address* is fixed at 15, pin B3 is read to determine the value of *data* and *CRC* is made equal to *data*. With all four variables set, the transmission function is called with `mc_tx(baud, address, data, crc);`. Here the main program sends the variables to the transmitter driver. No matter how simple or complex the main program may be, this is the only line required to transmit the information. The last line is just a 1 ms delay before starting all over again.
The transmitter driver is the true value of this design. It will remain unchanged, no matter what application the user may come up with. It needs to be included in the main program at the very top, after the PIC setup. The driver contains all the elements to create a frame in Manchester Code using just the four variables received from the main program. The bit rate is converted into half bit time (*semiperiod*), so the Manchester Code is easily obtained. The frame synchronization is constructed by calling function *one* twenty times, and then one time function *zero*.
Each byte is analyzed bit by bit, starting at the MSB side. Depending on the result, the proper function is called (*one* or *zero*), inserting the spacer after each complete byte. And that's it! A complete frame with all the bells and whistles previously described is sent out via pin B0 at the speed set by the bit rate variable. There is no need to change the driver program to send a standard frame. If more data bytes have to be included in the frame, only a few changes will be required.
Additionally, an LC display is implemented in order to display the encoded information (*address*, *data* and *CRC)*.

Receiver software (RX)

Similar to its hardware counterpart, the receiver (RX) code is just a little more complex than the transmitter. But not to worry: the complexity resides in the driver, which is discussed in the text frame *RX driver*. The same three-file structure is maintained here:

- Manchester_Link_RX.c (main program);
- Manchester_Link_RX.h (PIC setup);
- MAN_RX.c (receiver driver).

Again, the main program is quite simple. After the LCD procedures, the standard definitions are stated. Note that the Interrupt Service Routine (ISR) function is defined in `void detection_isr();`. This is important since the received information will enter through pin B0 (RB0/INT, pin 6) by triggering the external interrupt on the rising edge. The program will jump to this function as soon as this edge is detected.
Another important line that must be included is the setting of *timer0* inside the main function: `setup_timer_0(RTCC_INTERNAL|RTCC...`. This defines *timer0* as an 8-bit timer that is incremented every 16th period of the instruction clock. This internal timer is used to determine the bit rate of the incoming signal (by measuring the length of one bit).
Since we use interrupts, these must be enabled using `enable_interrupts(GLOBAL);`. Then the program enters an infinite loop, in which the incoming data is monitored. User program code can be entered here, provided none of the activities disable the interrupts. Some activities, like showing data on an LCD, may disable interrupts, hence these must be programmed inside the ISR, after the information is received and validated. Remember that the receiver is waiting for a signal to arrive.
When a rising edge is detected, the program jumps to the ISR, where the interrupts are disabled and the receive function is called: `frame=mc_rx(frame);`. This function is located in the receiver driver and it returns the 32-bit variable *frame*. The beauty of this is its simplic-

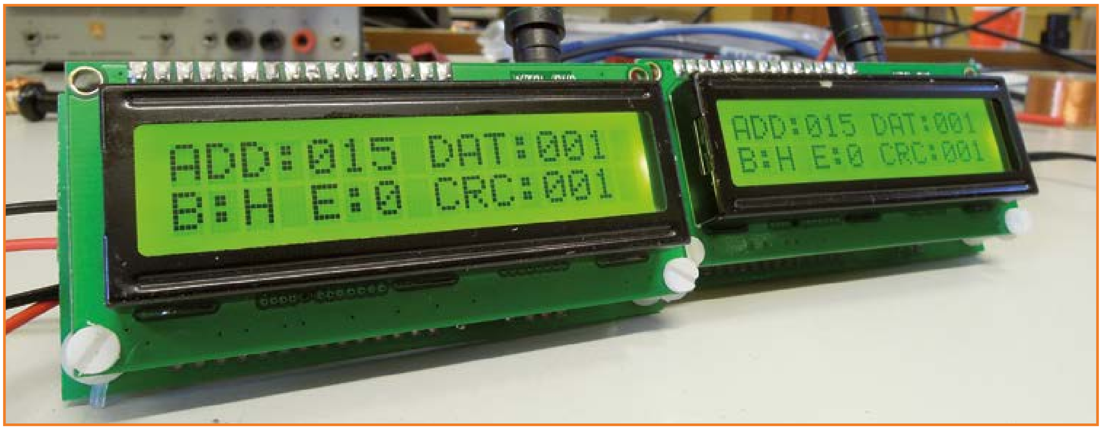


Figure 4.
First prototypes with LCD.

ity. A signal arrives, the *mc_rx* function is called, and *frame* contains all the information that was received.

Being a 32-bit variable, *frame* contains 4 bytes: *address*, *data*, *CRC* and *halftime* of the received frame. The bytes are extracted from the *frame* as follows. We copy the 32-bit frame to an 8-bit variable, so only the 8 least significant bits (LSBs) are copied: `halftime=frame;`. *Halftime* now contains the fourth byte of *frame*. Then we shift the contents of *frame* 8 positions to the right and repeat the copying process to extract the next byte.

Halftime contains the duration of half a bit, so the bit rate can be calculated from there. The rest of the instructions show how to use the received information in a very basic way. If there is an error in the received frame, the function will return a '0', so HALFTIME will be 0. This is used here to set the error flag.

If *address* is 15 (as set in the transmitter) and there is no error, then *data* contents are used to turn the LED ON and OFF. *CRC* is not used in this brief example.

The LCD continuously displays received information (address, *data* and *CRC*) after the interrupt is enabled. If the receiver does not receive any data, the LCD shows Error is '1' and Bit status is 'L'.

Improvements and applications

While the link is fully functional and proved to be extremely reliable during various tests, there is plenty of room for improvements and customizations, like sending more than one *data* byte at a time. Additional bytes could easily be added to a *frame*, you just need to know how many to expect to avoid missing any. Obviously a single 32-bit variable will not be able to contain all bits, so a structure should be considered.

The applications are endless, limited only by the user's imagination; this is just a building block for many bigger projects.

Figure 4 shows the prototypes assembled at Elektor Labs with an LCD display showing *address* = 15, *data* and *CRC* = 1, Bit rate (B) is high (H = 5,000 bps) and Error (E) = 0 (no error). This simple application shows the potential of these units and it is very useful to debug the code when things do not turn out as expected.

This concludes the second and last part of this project. Should you have any questions, concerns or just comments regarding the hardware or software presented, do not hesitate to join our topic on [2] or visit the author's web page at [3]; PCB artworks and software routines will be available for direct download.

(120187)

Correction to Part 1
(Elektor September 2013)

In the previous installment, the caption with Figure 5 mentioned that the "Transmit Output Power can be trimmed using *R3*". This should be corrected to read: … *R1*.

Internet Links

[1] www.elektor.com/120187

[2] www.elektor-projects.com/120049

[3] www.magusporta.com

Arduino
Now Available @ Elektor!

FREE SHIPPING until October 18th!

ARDUINO UNO
The most popular board with its ATmega328 MCU

Features

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Channels	6
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (of which 0.5 KB used by bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

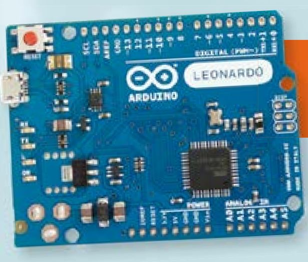
£24.40 • € 27.35 • US $39.70

ARDUINO MEGA
Like the Uno but with more memory and I/O

Features

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB (of which 8 KB used by bootloader)
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

£47.00 • € 52.77 • US $76.50

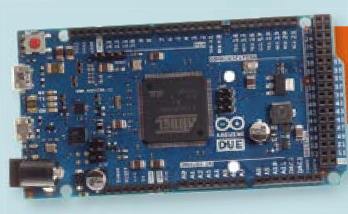


ARDUINO LEONARDO
Especially good for USB applications

Features

Microcontroller	ATmega32u4
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	20 (of which 7 provide PWM output)
PWM Channels	7
Analog Input Pins	12
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (of which 4 KB used by bootloader)
SRAM	2.5 KB
EEPROM	1 KB
Clock Speed	16 MHz

£22.10 • € 24.81 • US $36.00



ARDUINO DUE
32-bit power thanks to an ARM processor

Features

Microcontroller	AT91SAM3X8E
Operating Voltage	3.3V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 12 provide PWM output)
PWM Channels	12
Analog Input Pins	12
Analog Outputs Pins	2 (DAC)
DC Current per I/O Pin	130 mA
DC Current for 3.3V Pin	800 mA
DC Current for 5V Pin	800 mA
Flash Memory	512 KB (all available for the user applications)
SRAM	96 KB (two banks: 64KB and 32KB)
Clock Speed	84 MHz

£47.00 • € 52.77 • US $76.50

ARDUINO ETHERNET
Networking has never been easier

Features

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 4 provide PWM output)
Arduino Pins reserved	10 to 13 used for SPI 4 used for SD card 2 W5100 interrupt (when bridged)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (of which 0.5 KB used by bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

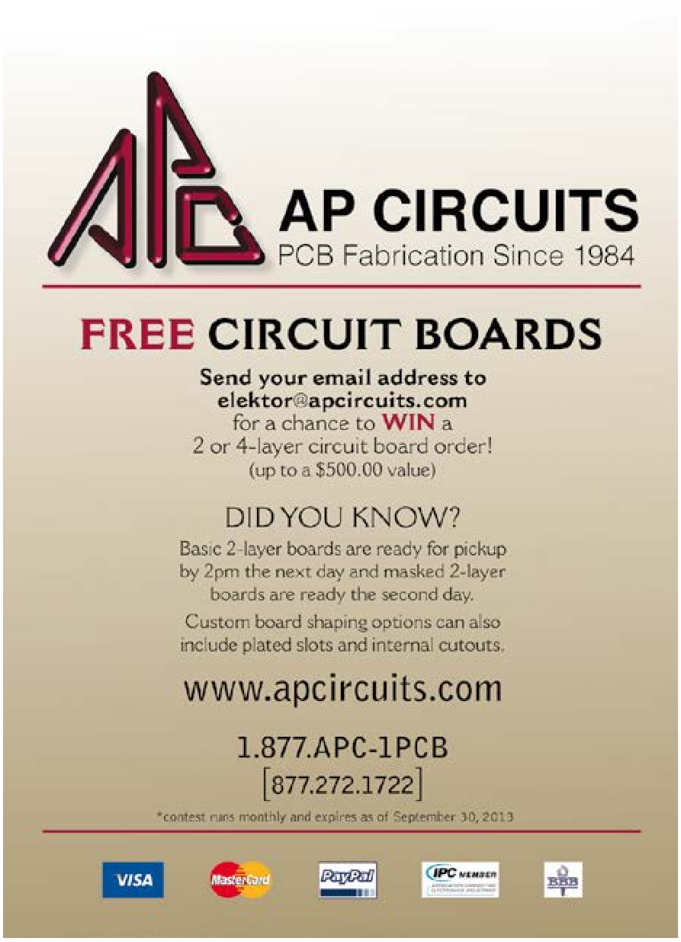
£48.10 • € 53.98 • US $78.30

10% OFF for GREEN and GOLD Members

ARDUINO

elektor

Further Information and Ordering at www.elektor.com/arduino


AP CIRCUITS
PCB Fabrication Since 1984
FREE CIRCUIT BOARDS
Send your email address to
elektor@apcircuits.com
for a chance to WIN a
2 or 4-layer circuit board order!
(up to a $500.00 value)
DID YOU KNOW?
Basic 2-layer boards are ready for pickup by 2pm the next day and masked 2-layer boards are ready the second day.
Custom board shaping options can also include plated slots and internal cutouts.
www.apcircuits.com
1.877.APC-1PCB
[877.272.1722]
*contest runs monthly and expires as of September 30, 2013
VISA
MasterCard
PayPal
IPC MEMBER


THINKING AUDIO OVER IP APPLICATIONS ARE HARD?
solutions™
EVEN KIDS CAN DO IT WITH SOCLUTIONS
ISL0A15E
ISL0A11E/SE
One-way voice-band
IP-PA System (PA over IP)
ISL0A12E/SE
One-way broad-band
IP-PA System (PA over IP)
ISL0A15E/SE
Two-way peer-to-peer (P2P)
voice-band and broad-band
(Intercom)
All solutions available in clear and secure modes
www.soclutions.com


THE ORIGINAL SINCE 1994
PCB-POOL®
Beta LAYOUT
create : electronics
FREE! SMT laser stencil
with every PCB Prototype order
Assembly service
From as little as one component
Embedded RFID
authenticate, track & protect your product
www.magic-pcb.com
NEW!
Free Phone UK: 0800 389 8560
sales@pcb-pool.com
RS-274-X
cadence
PULSONIX
DIPTRACE
www.pcb-pool.com
Beta LAYOUT
create : electronics
PCB-POOL® is a registered trademark of Beta LAYOUT GmbH


eSTORE®
Beta LAYOUT
Develop, assemble, solder
€ 38,95
Raspberry Pi Model B
512 MB RAM
Reflow Controller
€ 129,00
UHF RFID Starterkit Basic
€ 279,00
Large Beta-Reflow-Kit
€ 129,00
eSTORE® is a registered trademark of Beta LAYOUT GmbH
www.beta-eSTORE.com
Beta LAYOUT
create : electronics

DesignSpark Tips and Tricks

Day #4: a simple project

By **Neil Gruending**
(Canada)

Last time I talked about how to setup and use libraries in DesignSpark. Today we'll make a simple bi-color LED driver to learn how to use the schematic and circuit board editors. There are several ways to drive a bi-color LED and today we will use an H bridge variation which is hardwired to turn one of the LEDs on.

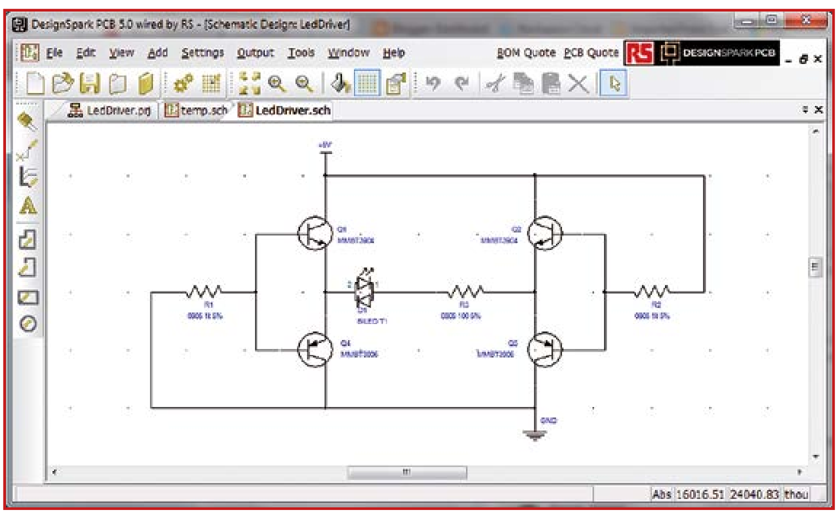


Figure 1.
Schematic of the bi-color LED driver.

Drawing the schematic

The first thing we need to do is to create a project file to link the schematics and PCB by using the "File->New" command. You then add a schematic to the project by using the "File->New" command again, but make sure that you check the "Add To Open Project" button. At this point you can also choose the technology file to use for the schematic like we talked about in an earlier article. **Figure 1** shows the schematic we will be using. Designspark has a schematic entry tutorial at [1] which covers how to add components and edit the schematic.

Moving visible component fields like reference designators around in a DesignSpark schematic is different than in some other packages because all of the fields are moved in 1 block. For example, in the example schematic there's a part number and reference designator visible for each transistor. Clicking on either one will highlight both fields and they can be dragged to a new position as a group. This can cause problems when mirroring components, which DesignSpark calls flipping, because the text can be incorrectly lined up. Fortunately the text alignment can be easily changed by right clicking the text and selecting the "Properties" menu. There in the "Text" tab you will find an alignment field that will let you choose between "Left", "Center" and "Right" text alignment.

Also, don't forget that power and ground symbols are components in DesignSpark. The default ones are in the DesignSpark schema library, but you can also create your own library of symbols to your liking. Note that if you connect a power symbol to an existing net DesignSpark will warn you that it will rename the net even though that's what you want.

The LED component

For the transistors and resistors I used some existing schematic symbols and PCB footprints from the DesignSpark libraries. However, for the LED I modified an existing DesignSpark LED symbol and then made a custom PCB footprint for it. Making a custom PCB footprint is a lot easier when you use the footprint wizard. You can access the wizard by opening the PCB library where you want to save the footprint with the library manager and then clicking on the "Wizard..." button. The PCB footprint wizard will then ask a series of

questions to make a footprint and since they are a generic as possible it's important to pick the closest type possible to minimize later editing. In the case of my LED I used an axial component with 2.54-mm (0.1'') lead spacing so that all I needed to do was to edit the silkscreen and mark the polarity on pin 1.
DesignSpark also includes similar schematic symbol and component wizards.

Getting ready for layout

Now we're ready to layout our circuit board by creating a new PCB file using the "Tools->Translate to PCB" menu which will start the New PCB Wizard. We are going to create a 2-layer metric design that's a 20 mm square. If you tell the wizard to place the components outside of the board you will get something like in **Figure 2.**
I like to place components on a 0.25 mm grid so I changed the working grid to 0.25 mm before placing the components on the circuit board to get an arrangement like in **Figure 3**.

Before routing the board I want to talk about the routing grid used when placing the copper traces on the circuit board. DesignSpark doesn't include an interactive autorouter which means that you have to set the routing grid to the width of the trace you're routing. This way when two traces touch the spacing is 0 mm, and when there's a gap between them the spacing is at least the trace width. This works because the routing grid is applied to the center of the trace instead of the edges. Therefore, if you want to route a 0.2 mm trace then you would set the routing grid to 0.2 mm spacing to get 0.2 mm trace spacing. The downside of this technique is that all the trace widths should be multiples of the smallest size. For example, 0.2 mm and 0.6 mm would work but 0.2 mm and 0.35 mm would not. Also, be sure to create a style for each trace width that you want to use in the design technology settings (Settings->Design Technology... and then select the "Track Styles"). That makes the different trace widths much easier to manage in more complicated designs because you can change the current trace width by just changing the style. You can change the current track style by pressing "s" while routing and then choosing the new style you want.
The same is also true for vias and in DesignSpark you configure the via styles using the "Pads Styles" tab in the "Design Technology" window.

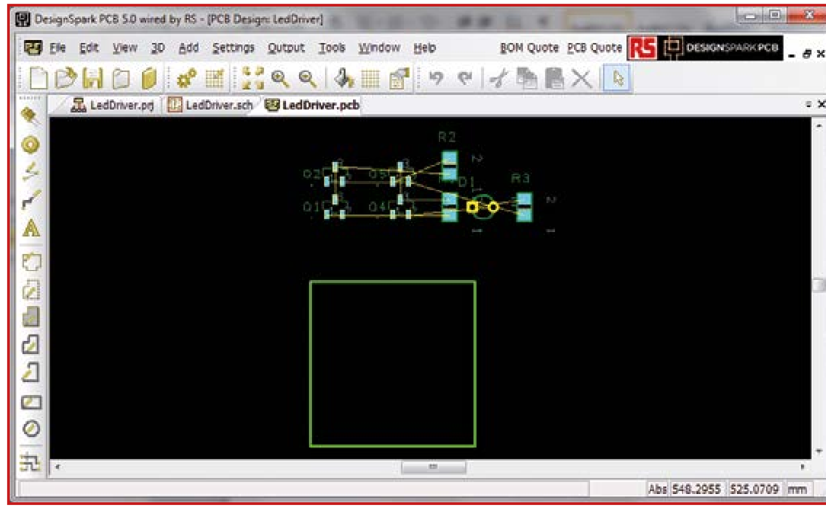


Figure 2. Parts are initially placed outside the box.

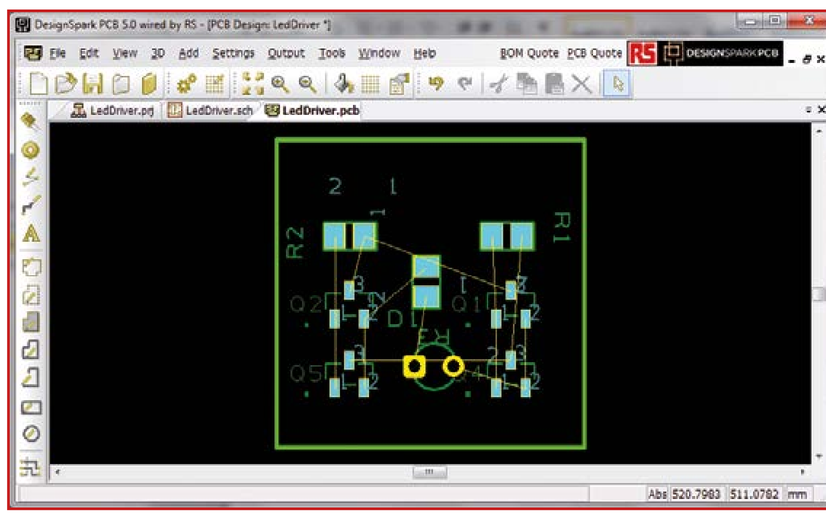


Figure 3. The parts are now inside the box at a modified 0.25 mm grid.

I recommend making a "SignalVia" style and any other styles that you may need. I chose to make the signal vias in this design with a 0.45 mm hole and a 0.95 mm pad. You can change the current style used for vias by right clicking while routing and going into the "Change Via Style" menu to choose the style you want.
DesignSpark uses the "Settings->Defaults" menu to set the track and via defaults, although I haven't been able to get DesignSpark to recognize the new settings even with a restart.
More information is available from the DesignSpark website about PCB setup and placement is available at [2].

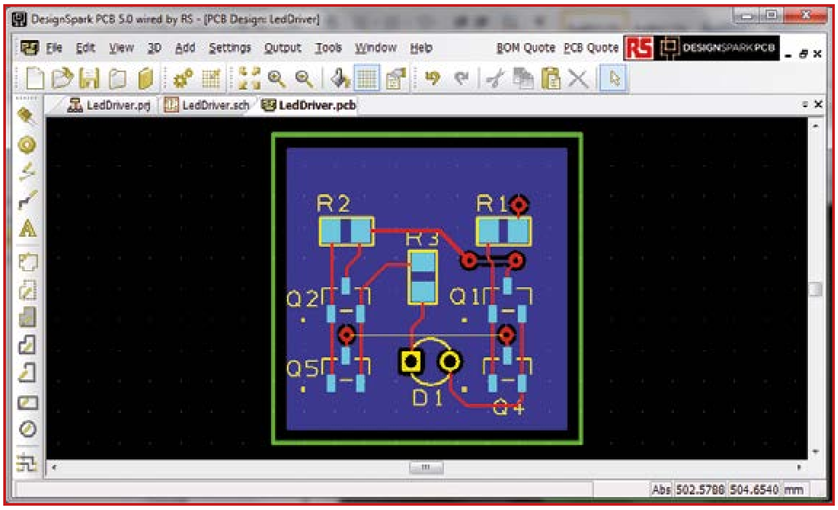


Figure 4. End result of the board design work.

Figure 5. Applying the Design Rule Check (DRC) tool.

Layout

Once everything is configured it's time to lay out the circuit board. The result is pictured in **Figure 4**.

I routed all of the signal traces on the top layer and used a polygon pour to make a bottom ground plane. When routing the board it's important to make sure that you double click on the "rat's nest" interconnect line when starting to route a trace. As you run the trace you can change how DesignSpark deals with corners by right clicking and choosing a different segment mode.
If you look closely at the layout you can see a "rat's nest" line between Q4 and Q5 for the ground connection which means that DesignSpark thinks that those two transistors aren't connected to ground. Fortunately DesignSpark includes a design rule check (DRC) in the tools menu that can verify all of the board connections—see **Figure 5**.

Once you click the "Check" button DesignSpark will then verify that the layout meets all of the selected design criteria and it will generate a report summarizing any errors. Errors are also marked in the layout and if you mouse over them the error message will be displayed. All of the clearance rules are set in the design technology settings "Spacing" tab where there's a table listing all of the clearances between different object types.

For more information about routing a board with DesignSpark, a tutorial is available at [3].

Conclusion

Today we created a simple PCB from a schematic and then verified the design using DesignSpark's verification tools. Next time we'll generate a BOM and the Gerber files so that we could build the design.

(130230)

Internet References

[1] www.designspark.com/eng/tutorial/schematics-entry

[2] www.designspark.com/tutorial/pcb-setup-placing-components

[3] www.designspark.com/tutorial/pcb-routing

elektor PCB Service
powered by Eurocircuits

PCBs
Multilayer
Prototypes
Small series

Professional Quality
Trusted Service
Secure Ordering

Elektor PCB Service at a glance:

- 4 Targeted pooling services and 1 non-pooling service
- Free online PCB data verification service
- Online price calculator available
- No minimum order value
- No film charges or start-up charges

Delivery from 2 working days

Further information and ordering at www.elektorpcbservice.com

Getting Started with the LPC800 Mini Kit

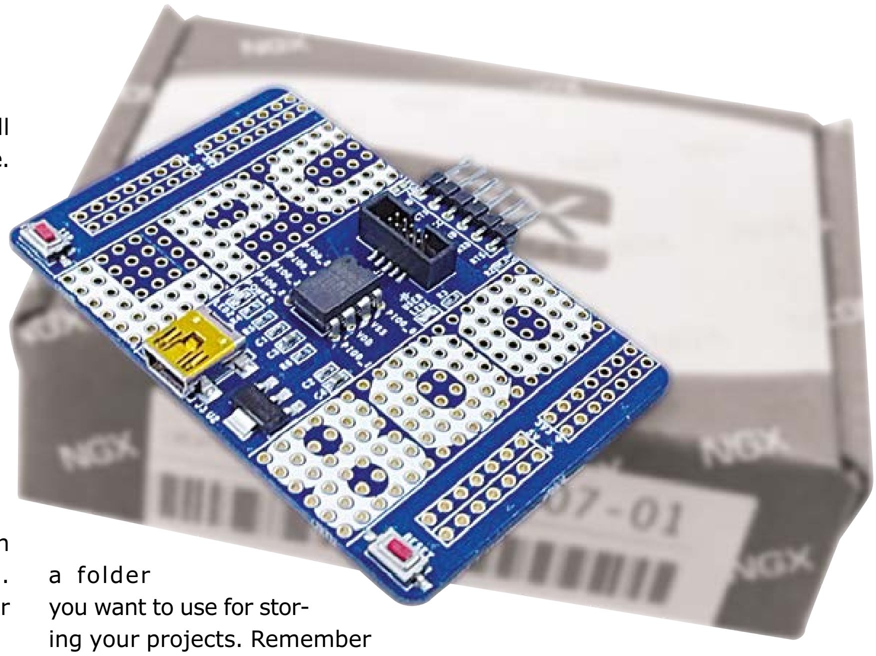
Just before the summer started Elektor offered its members the opportunity to receive a little microcontroller board for free. The board was an LPC800 Mini Kit sporting an LPC810 32-bit ARM Cortex-M0+ microcontroller (MCU) in an 8-pin DIP package, a voltage regulator, two pushbuttons, an LED and two small prototyping areas. The campaign was a huge success: all boards were gone in about 45 minutes…

By
Clemens Valens
(Elektor.Labs)

If you were among the Fast & Lucky having managed to get your hands on one of these boards, you have had all summer to experiment with it. Now, we know how these things go: as soon as you receive the kit, you open the box, have a good look at the board, connect it to a PC to see the LED blink, and then put it on your desk for later use. And, for many of those kits, this is where they still are. Therefore, to save these kits from e-oblivion, here is a short tutorial to explain you how to get started with this little kit. To play with the Mini Kit you will need these things:

- a Mini Kit;
- a PC with a 3.3 V logic-level compatible serial port;
- a 5 V power supply;
- the Serial Wire Debug (SWD) pod (optional).

You can tick off the first line—to tick off the second you either need a PC with a real serial port connected to a level adapter like a MAX3232 powered at 3.3 V, or—much easier as it also takes care of the third item—a 3.3 V "FTDI" cable (available from the Elektor Store, # 080213-72). You can also use our "BOB" USB/Serial Bridge (Elektor Store # 110553-91), which is more flexible, but needs a hot soldering iron because you have to set its solder jumper to the 3.3 V position. Both the FTDI cable and the BOB can provide the 5 V supply voltage needed to power the Mini Kit. The Mini Kit has a connector compatible with a 3.3-V FTDI cable, but I prefer to use a BOB mounted on a small adapter board on which I placed some jumpers that allow me to disconnect the serial port data lines without cutting the power (**Figure 1**). That's useful for experimenting as some functions of the MCU share pins with the serial port. It is also possible to select the voltage on the VCC (0 V, 3.3 V or 5 V) pin. If you use this adapter circuit, place a jumper on JP1 pins 1 and 2 to select 5 V as VCC. This is necessary to make the Mini Kit's on-board voltage regulator work.
The serial port is needed for programming the MCU—real debugging is not possible. If you own a Serial Wire Debug (SWD) pod you can use that instead of the serial port to load your programs into the MCU. The pod also allows debugging



them. Sadly I do not have an SWD pod—I will use the serial port in the remainder of this article. Once you have all the hardware, you need to set up some software. All you need is free, you only have to download it. Go get:

- LPCXpresso IDE (big, requires registration) [1];
- Flash Magic [2];
- LPC800 Mini Kit code base [3].

Needless to say that you need the latest versions of all tools and libraries. Install the first two tools in random order (people planning on using an SWD pod do not need Flash Magic). Unpack the code base somewhere on a disk—your project folder would be a good place.

With Flash Magic installed you can check the communication between the Mini Kit and the PC. Connect the board to the serial port (using a BOB, FTDI cable or your own level adapter) and make sure that it is powered. You can also power the board through the USB connector if you prefer (this connector only provides power—the data lines are not connected). The on-board power LED should light up and if your board is brand new the user LED will start blinking.

Press the ISP button and keep it depressed while you press the Reset button. The blinking LED should stop blinking (if it was blinking). Launch Flash Magic, 'Select…' the correct MCU (LPC810M021FN8) and the right 'COM Port'. The 'Baud Rate' can be 115,200 baud but I experienced problems at that speed; 38,400 baud always worked for me. 'Interface' should be 'None (ISP)' and the 'Oscillator' field can be left blank (**Figure 2**). From the 'ISP' menu select 'Read Device Signature…'. A window pops up and if all is well it should get filled with some data. If you get an autobaud error chances are that either your cable is not working properly or that the MCU is not in ISP mode. Try again after checking your cable and maybe on a different speed. Possibly the MCU's Reset pin has become disabled (due to your earlier experiments). In this case you should disconnect the power (or the serial cable), then hold the ISP button down while you reconnect the power (or the cable). This trick will always put the MCU in ISP mode. If you still can't read the device's ID, then you must have a connection problem.

Now it is time to launch the LPCXpresso IDE. When you are asked for a workspace, point it to a folder you want to use for storing your projects. Remember the path because you will need it later. The IDE takes a while to start, but when it is finally ready it offers a quick access menu named 'Start here' containing the most important functions (and

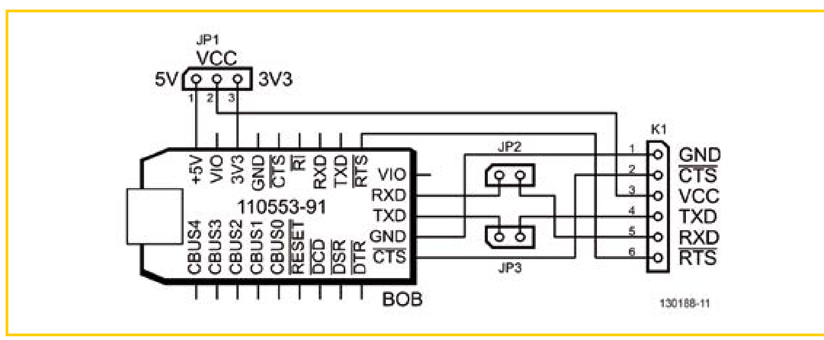


Figure 1.
The Elektor BOB adapter board used to program the LPC800 Mini Kit.

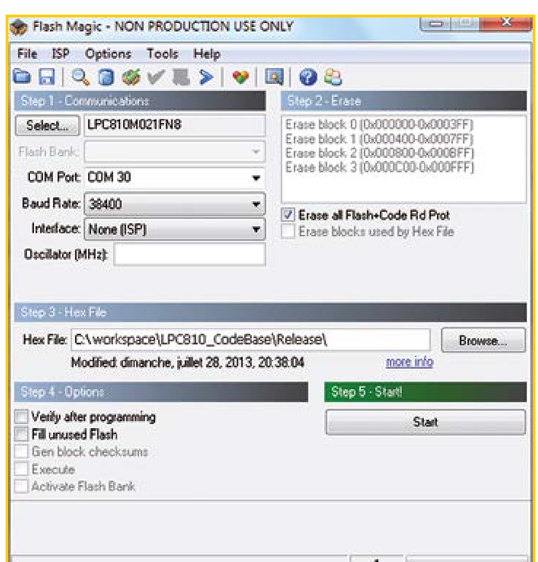


Figure 2.
Flash Magic and the settings that work for me.

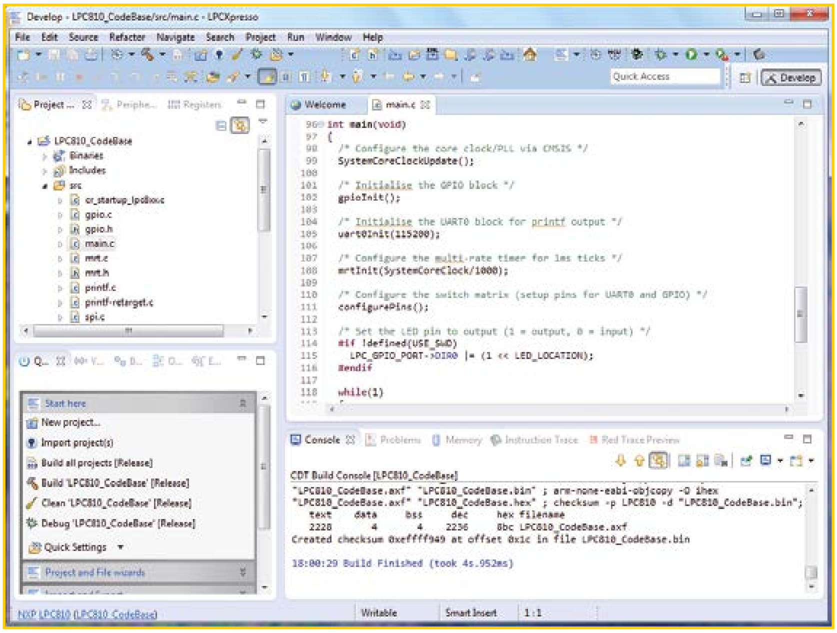
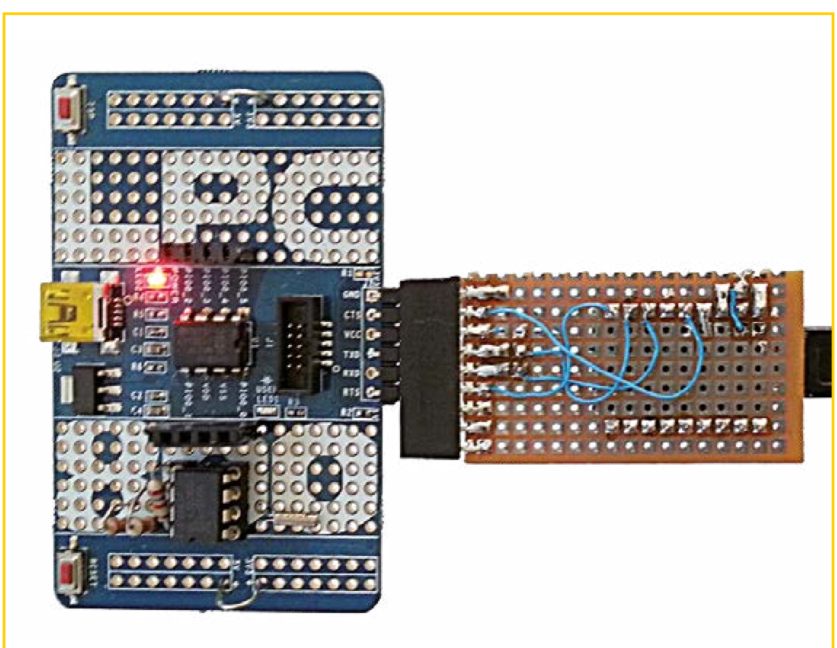


Figure 3. The LPCXpresso IDE showing a successful build of the LPC800_CodeBase project.

Figure 4. My LPC800 Mini Kit development setup with the BOB on its adapter board (yes, I know it's upside down). The IC on the lower part of the board is a type PCF8563 I²C real-time clock (RTC).



www.elektor-labs.com

some more) that you will use often, like new project, build & debug. Here you will also find an option to import example projects. Click the link 'Import project(s)' to open the import dialog, then click the 'Browse...' button to the right of the 'Root directory' field if you unpacked the code base (if you kept it as an archive you can click the other 'Browse...' button) and navigate to the LPC800 Mini Kit Code Base folder. Once you have found it click 'Ok' followed by 'Next'. Make sure the LP810_CodeBase project is ticked before clicking 'Finish'.

You will now have a project called 'LPC810_CodeBase' in the 'Project Explorer' window. Select it to build it from the 'Start here' menu. Observe the messages that scroll through the 'Console' window; there should not be any errors or warnings (**Figure 3**). If for some reason you do have an error or a warning, click on the 'Problems' tab to get more information. Double clicking a line in this window will take you to the offending code. After a successful build, you will find a HEX file in the 'Release' folder of your project that's been copied into your workspace during the import. Click the 'Browse...' button in Flash Magic to navigate to your HEX file. Put your Mini Kit in ISP mode before clicking 'Start'. If all is well the MCU will be programmed with your new HEX file. Press the Reset button on the board to launch the program.

You are now ready to start developing your own projects. The Elektor.Labs website [4] has a few get-u-going projects using this board. If you do something awesome, useful, interesting or whatever with the Mini Kit we would like to know about it. To inform us please post your LPC800 projects on Elektor.Labs.

(130188-I)

Internet Links & References

[1] LPCXpresso IDE: http://lpcxpresso.code-red-tech.com/LPCXpresso

[2] Flash Magic: www.flashmagictool.com

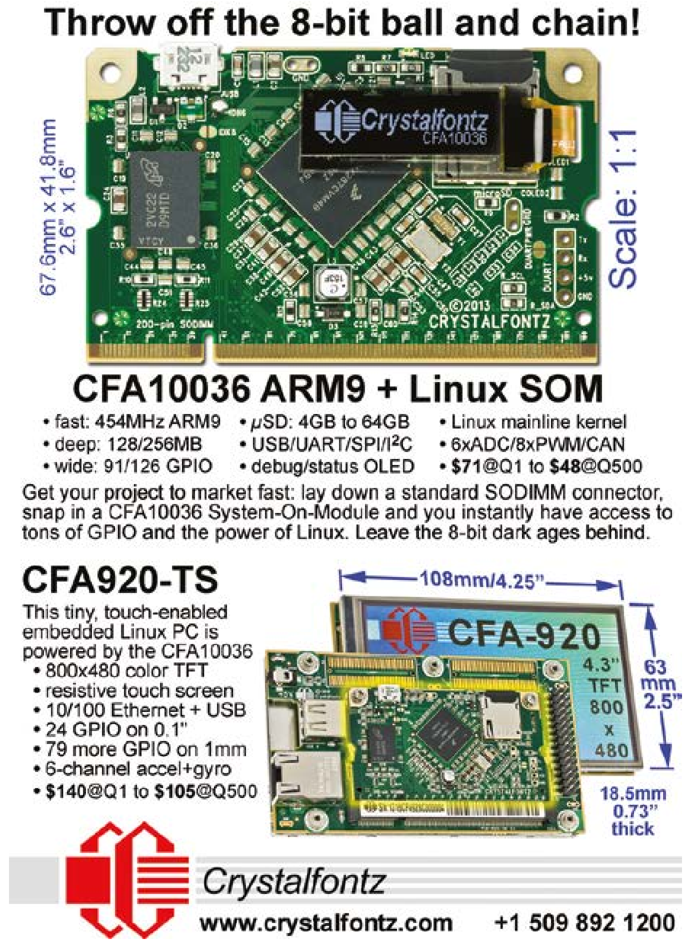
[3] LPC810 Code Base: http://lpcware.com/lpc800-mini-kit

[4] Elektor.Labs: www.elektor-labs.com

Tips & Tricks

The LPCXpresso IDE is based on Eclipse, a popular albeit–in my humble opinion–absolutely horrible tool. I therefore recommend the inexperienced user to use the example project as a starting point for new projects. You can copy & paste a project in the 'Project Explorer' window using right mouse clicks—that way you will be sure to get the settings right.
Projects created from scratch will not produce a HEX file. To correct this, copy the settings from the example project: select the sample project; on the menu click 'Project' then 'Properties'. Expand 'C/C++ Build', click 'Settings' and then the 'Build Steps' tab. Copy the content of the 'Command' field of the 'Post-build steps' area and copy it into Notepad or something similar so as not to lose it during the steps that follow. Close the 'Properties' dialogue. Now select the new project and repeat the steps above to return to the post-build steps, but this time of the new project. Replace the post-build steps command line with the line you just copied. Click 'OK' to save the settings. Repeat this procedure for every configuration you may have created (release, debug, other).
Only build for Release or you will quickly run out of program memory (the LPC810 has only 4 KB). Of course, if you have an SWD pod you may want to build debug versions too, but Flash Magic users have nothing to gain here.
Adding existing source code files to a project is rather counterintuitive (or should I say counterproductive?): 'File' → 'Import…' → select 'Filesystem', click 'Next', browse to the file's location, check the file you want and click 'Finish'. Since the IDE does not remember the last path you used you may need a lot of mouse clicks to get things done. But, you can also copy the file directly into your projects folder using your favorite file manager (Total Commander in my case). After copying the files press 'F5' to refresh your project and the new files appear.
The MCU's reset function can be disconnected from the pin so that the latter can be used for something else (see the MCU's switch matrix). The Reset button on the board then becomes useless and entering ISP mode is more difficult. In this case you should switch off the power to the board, and hold the ISP button down while you switch the power back on. This trick will always put the MCU in ISP mode. This will also re-enable the SWD interface in case you had it disconnected.

Advertisement



Negative Energy

By
Dr. Thomas Scherer
(Germany)

It just so happened that on the last visit to my in-laws their electric doorbell had just stopped working. It is a wireless type I installed around two years ago at their side entrance which never gets much use. The curious thing is that I remember replacing the batteries just two months ago. My father in law, as ever takes the pragmatic approach and suggests replacement batteries. However, he has clearly underestimated my determination—there is an anomaly here that needs resolution. “You don’t happen to have a multimeter in the house” I asked.

He had, I can’t remember the last time I used one of these analog moving-coil multimeters. Engraved in the heavy plastic casing were the words ‘Battery Tester’, and I figured the instrument should do the job nicely as long as it was still working. With both parts of the doorbell now on a table in front of me and in the absence of any sophisticated test equipment I confidently assumed that the flashing LED is evidence that the bell push and transmitting half of the unit is functioning correctly. With the receiver half of the doorbell now right next to the transmitter I think we can rule out any issues of range. The finger of suspicion looks to be pointing at the receiver unit, showing no activity whatsoever, be it on a LED or an attempt at plying the Big Ben chime .

As I think, a tick list of possible causes is already starting to scroll up on that imaginary screen inside every engineer’s head. In this case it went something like this:

1. Gramps playing a trick on me? (possible but unlikely)
2. A bad set of batteries. (gramps scores—duh)
3. Electronic failure. (always possible)
4. Something else. (can’t rule that one out)

What’s your guess?

With the battery compartment opened the three AA cells I fitted not two months ago are still in position and before you ask… Yes, they are all the right way round. With the battery tester set to 4.5 V full-scale I touch the test prods across the three series connected cells and read a measly 1.5 V.

So hypothesis #2 seems probable. My father in law is now starting to look a little too pleased with himself. I take out all three cells, set the meter to the 1.5 V range and measure each battery individually. The first one reads 1.5 V… curious, does that mean that the next two are completely flat? The second one also reads 1.5 V, now I’m confused, two batteries each reading 1.5 V but the voltage across all three is just 1.5 V! I measure the third battery and watch the needle swing the wrong way and try to wrap itself around the end stop. This one seems to be outputting a negative voltage; I switch the probes around with the red one to the negative terminal and the black one on the positive to get a reading of 1.5 V.

1.5 V + 1.5 V – 1.5 V = 1.5 V. No wonder!

The problem was fixed by just replacing the faulty battery. The question remains: is it possible for a battery to reverse its polarity? Well no, cell chemistry defines the direction of current flow. I took the faulty battery home to look at it a bit more closely. Next morning it still showed a negative voltage but my DVM was now indicating just -35.9 mV after the cell had been resting out of the circuit for about eight hours.

The most likely explanation is that the faulty cell had probably suffered a manufacturing fault leaving it with very low capacity. Once its initial energy had been used up the current flow produced by 3 V from the other two healthy cells (the bell receiver is permanently on) had the effect of reverse charging the faulty cell.

(130108)

Join the Elektor Community

Take out a GOLD Membership now!

Your GOLD Membership contains:

- 8 Regular editions of Elektor magazine in print and digital
- 2 Jumbo editions of Elektor magazine in print and digital (January/February and July/August double issues)
- Elektor annual DVD-ROM
- A minimum of 10% DISCOUNT on all products in Elektor.STORE
- Direct access to Elektor.LABS
- Direct access to Elektor.MAGAZINE; our online archive for members
- Elektor.POST sent to your email account (incl. 25 extra projects per year)
- An Elektor Binder to store these 25 extra projects
- Exclusive GOLD Membership card

ALSO AVAILABLE:

The all-paperless GREEN Membership, which delivers all products and services, including Elektor.MAGAZINE, online only.

Take out your Membership now at www.elektor.com/member

Improved Power Factor Controller

Intersil Corporation's new active power factor controller type ISL6730A features a patent-pending breakthrough in negative capacitance technology which reduces EMI filter size, improves THD and PF and provides maximum efficiency over a wide input supply range and output power (input voltages from 85 VAC to 270 VAC and power range from 50 W to 2 kW). This technology also minimizes zero crossing distortion, compensates for input filter capacitance PF displacement error and reduces the magnetic component size by up to 66 percent. For example, in an 85 W power supply the EMI filter inductor can be reduced from 150 µH to 56 µH. The small external components result in a lower cost design with improved performance.

The device's excellent power factor correction capability is compatible with the requirements for a power factor higher than 0.9, as currently required by ENERGY STAR Program Requirements for Computers Version 5.0. The ISL6730A also achieves excellent light load efficiency using integrated skip-mode, and includes an internally clamped 12.5 V gate driver delivering 1.5 A peak current to the external power MOSFET.

The ISL6730A enables a reliable system that is fully protected with features such as cycle-by-cycle over-current, power limiter, over-temperature, input brownout, output over-voltage and under-voltage protection.

www.intersil.com (130167-VII)

Toradex Embedded Design Challenge

Here's your chance to transform your ideas into reality. Toradex has launched the "Embedded Design Challenge" which is open for students and other individuals at education and research institutions. All the student innovators from all over the world are invited to participate. We welcome any innovative idea, there is no required theme.

To participate in the challenge all you need to do is to submit the product proposal of what you want to build. On the selection of your proposal you will receive a free Toradex Colibri NVIDIA Tegra™ Kit (worth $250 USD) to bring your idea to life. Along with the kit Toradex also offers technical support from our engineering team, access to support libraries and online knowledge base.

Twice a year the projects are judged and the winner will receive a prize of $20,000. Additionally there will be four selected honorable mentions which will receive a prize of $5,000 each.

Toradex is currently accepting project proposals! Compete individually, or in teams! The process to submit your proposal is very simple, entries have to be submitted online at:

www.challenge.toradex.com (130306-V)

Ultra-Low Noise Dual JFETs

Linear Integrated Systems, a leading full-service manufacturer of specialty linear semiconductors, announces the immediate availability of its LSK489 1.8 nV at 1 kHz, low-capacitance, N-channel monolithic dual JFET. This is part of a family of ultra-low-noise, dual JFETs specifically designed to provide users better-performing, wider bandwidths and cheaper solutions for obtaining tighter I_{DSS} (drain-source saturation current) matching and better thermal tracking than matching individual JFETs.

Available packaged in surface mount and ROHS compliant versions, the LSK489 is an ideal improved functional replacement for the similar JFETs that have similar noise characteristics but greater gate-to-drain capacitance. The LSK489 SOT-23 and SOIC packages are ideal for space-limited circuits in audio and instrumentation applications. LSK489 available packages are: TO-71; SOT-23-6L, SOIC-8L.

The most significant aspect of the LSK489 is how it combines a noise level nearly as low as the LSK389 while having much lower gate-to-drain capacitance, 4

pF versus the 25 pF, Hall said. While the LSK389 provides ultra-low noise of less than a 1 nV at 1 kHz, the capacitance is high enough to cause designers to have to use a cascode feature to handle higher bandwidths without intermodulation distortion.

Like the Linear Systems LSK389, the LSK489 features a unique Monolithic Dual design construction of interleaving both JFETs on the same piece of silicon to provide excellent matching and thermal tracking and a low-noise profile having nearly zero popcorn noise.

Lead-Free, ROHS compliant versions are available. Linear Integrated Systems' in-house fab and domestic factory stock guarantee short lead times, ensuring no disruption in production schedules.

Summary of Features:

- Low noise (typically 1.8 nV/√Hz @ 1kHz)
- Nearly zero popcorn noise
- IDSS (drain-source saturation current) matching to 10% max
- Low offset/tight matching ($|V_{gs1} - V_{gs2}|$ = 20 mV max)
- Low capacitance (C_{ISS}=4 pF)
- High input impedance
- High breakdown voltage (BV_{GSS} = 40 V min)
- Monolithic Dual (2 JFETS on one piece of silicon, better matching and thermal tracking)
- Low noise, reduced device count alternative for the classic dual JFET cascode configuration
- Improved replacement for Siliconix U401 series
- Surface mount SOIC versions and the smaller SOT23-6 package
- Lead-free ROHS compliant versions available

Applications include microphone amplifiers; phono preamplifiers; audio amplifiers and preamps; discrete low-noise operational amplifiers; battery-operated audio preamps; audio mixer consoles; acoustic sensors; sonic imaging; and instrumentation amplifiers; wideband differential amplifiers; high speed comparators; impedance converters.

www.linearsystems.com (130306-II)

First LIN Slave Companion IC to Support Automotive ISO26262 Requirements

ams AG announced that the AS8530 is the world's first miniature power/transceiver IC to support LIN slave applications, which complies with the ISO26262 functional safety standard.

The AS8530 is a power management and communication device that includes a LIN 2.1 transceiver, a 50mA LDO to supply a local micro and a reset generator in an 8-pin SOIC 8 package. As a differentiator, the AS8530 offers a series of system management functions through a shared pin serial interface, all within the same small, 8-pin SOIC8 package. The addition of enhanced diagnosis functions provides built-in support for the requirements of ISO26262. A two-wire serial port routed through shared Enable and TX pins allows the device to read out status registers and provide diagnosis information to the system's microcontroller. It also supports additional features such as a window watchdog function and access to back-up registers to store data when the microcontroller shuts down.

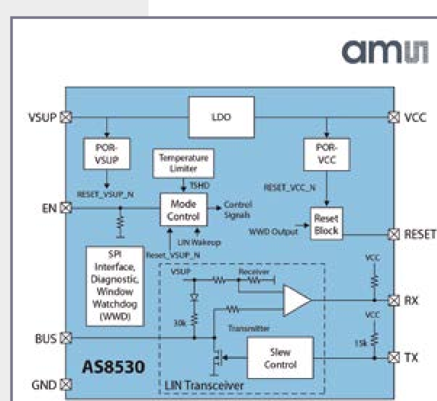
These features are crucial in the design of ISO26262-compliant systems, which must be able to cease operation safely when in a fault condition, before the vehicle is at risk of endangering passengers or other road users.

The AS8530 is suitable for any LIN-networked sensor and actuator slaves including those found in door modules, sunroofs and headlight positioning units. It is ideally suited for safety-critical systems that require an ASIL grading under the provisions of the ISO26262 standard.

The on-chip LDO provides a factory-selectable 3.3V or 5V output. The chip also offers a reset generator and an output voltage monitor and has a unique chip ID to support traceability requirements. Power-saving, stand-by and sleep modes, as well as the normal operating mode, can be triggered via the Enable pin.

The AS8530 LIN IC is in volume production now. It is priced at $2.21 for 1,000 pieces.

www.ams.com/Interface/LIN-Bus-System/AS8530 (130306-IV)



Chip Quik Removes SMDs Safely & Easily

The Chip Quik® SMD removal kit enables SMD parts to be removed from circuit boards with no more than a soldering iron.

Suitable for low temperature reworking, the product removes QFP's , PLCC's, SOIC's, chip components, and other surface mount configurations. Chip Quik removal alloy melts at 300°F (150°C). Fast, safe, and easy to use, Chip Quik eliminates the need for complex expensive equipment. You can learn how to remove SMD's in minutes; there is no need to stock a large inventory of expensive tips & nozzles. No damage to PC boards or adjacent components, no more burning of boards, lifting pads or lands, reflowing adjacent components, damage to double sided boards, throwing PC boards away because of unreliable removal methods.

Chip Quik is now used extensively worldwide throughout the Electronics Industry. The product is a low temperature removal alloy with excellent wetting ability. When melted into the existing SMD solder connections, both alloys combine, resulting in a new lower temperature alloy. Each pin/pad connection stays molten long enough to easily remove the SMD.

SMD 1 Kit Contains: 2.5 ft. Chip Quik Removal Alloy (removes approx. 8 SMD's); 2cc syringe of Chip Quik No CleanTack Flux; Alcohol Pads for Clean Up; Complete Instruction for SMD Removal & Clean Up.

Chip Quik was tested at Elektor Labs with excellent results.

www.chipquik.com (130306-I)

CapSense Express Mechanical Button Replacement

Cypress Semiconductor Corp. announced a new CapSense® Express™ capacitive touch-sensing controller optimized to replace mechanical buttons in front panels for industrial and consumer applications, portable medical devices, gaming devices and home automation systems. The new low-power CY8CMBR2110 device supports up to 10 buttons and drives up to 10 LEDs with fully configurable LED effects.

Cypress also introduced the EZ-Click™ customizer tool, GUI-based software that combines device configuration, visual feedback, and production line testing for streamlined register configuration of the CY8CMBR2110 controllers, thereby accelerating time-to-market. Designers can use the tool to implement customized LED effects and buzzer output for audio feedback. Controllers in the Mechanical Button Replacement (MBR) family leverage Cypress's SmartSense™ auto-tuning algorithm, which completely eliminates the requirement for manual system tuning and is the only solution that maintains optimal button performance during run-time.

The CapSense Express MBR family includes the CY8CMBR2016 matrix keypad solution, the CY8CMBR2010 ten-button controllers and the CY8CMBR2044 four-button hardware configurable controllers. Devices in the family offer the industry's lowest power consumption with supply current in run mode of 15 µA per button and a 100 nA Deep-Sleep mode. The devices operate over a 1.71 V to 5.5 V range, making them ideal for a wide range of regulated and unregulated battery applications, and enabling them to operate from a single coin cell battery. The family delivers robust sensing in noisy environments using Cypress's patented CapSense Sigma Delta (CSD) sensing method, ensuring superior immunity to conducted and radiated noise. These devices also feature an integrated voltage regulator to address power supply noise, as well as filters for any spurious noise.

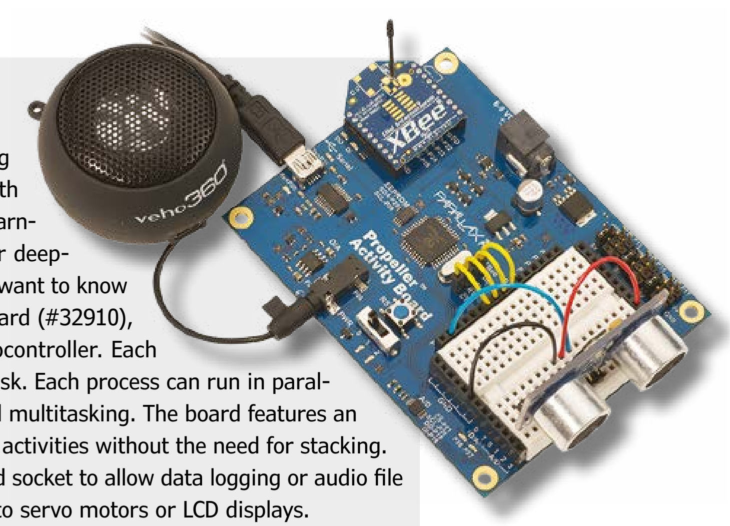
The MBR family features SmartSense auto-tuning,

which dynamically optimizes the capacitive baseline and detection threshold for each button. The algorithm adjusts for the optimal capacitance sensing range at power-up and during runtime as environmental conditions change, including noise, temperature, and humidity. Eliminating the need to tune is a significant advantage for large and small manufacturers alike, as it saves engineering time and yield loss that can occur with even slight variations in manufacturing tolerances. This savings is greatly multiplied for customers with a global factory footprint and multi-sourced supply chain. SmartSense auto-tuning can eliminate the need for additional test steps required by competing solutions to address manufacturing variations in PCBs and overlays.

Cypress offers the CY3280-MBR2 CapSense Express with SmartSense Auto-Tuning Evaluation Kit to support the CY8CMBR2110 controller. The MBR family's accompanying Design Toolbox is a simple, interactive spreadsheet that provides detailed resources to ensure optimal performance and validate CapSense systems. The toolbox delivers advanced system debug features and offers application specific guidelines for capacitive buttons, allowing customers to take designs directly to production for significantly shorter time-to-market. The CY8CMBR2110 CapSense Express MBR controller is currently in production in a 32-pad QFN package.

www.cypress.com/go/capsense (130306-III)

Propeller C Learning System



Parallax' new Propeller C Learning System consists of a programming tool and a suite of tutorials featuring simple circuits and libraries with code samples for core devices and sensors. The program simplifies learning to program in the C language for new users, but also allows for deep-dive exploration of the libraries and background code for those that want to know more. The program is being launched with the Propeller Activity Board (#32910), a new hardware platform featuring the Propeller multi-core microcontroller. Each core of this on-board chip can be dedicated to process a different task. Each process can run in parallel, providing truly seamless processing for maximum efficiency and multitasking. The board features an ideal balance of on-board peripherals to complement the program's activities without the need for stacking. Among them an XBee socket for wireless capabilities, a microSD card socket to allow data logging or audio file playback and breadboard to allow the easy solder-less connections to servo motors or LCD displays.
Visit the Propeller C Learning System page for all the resources or visit the Propeller Activity Board product page for hardware specification or to order this exciting new hardware from Parallax. Visit Parallax.com search "Activity Board." Retail: $49.99

www.parallax.com (130306-VI)

Microchip: World's First USB2 Controller Hubs

Available in three families, optimized for mainstream USB2, mobile USB2 and simultaneous USB2 and HSIC designs, Microchip's programmable USB2 Controller Hubs (UCH2) help to overcome the challenges of extending battery life and supporting multiple platforms and communication protocols.
The on-chip "Quad Page" OTP Flash memory eliminates the need for external configuration memory by reserving space for interoperability and enabling easy customisation using Microchip's free ProTouch Configuration Editor software tool. The UCH2 hubs also provide direct I/O bridging to I²C™, SPI, UART and general-purpose I/O, with support for vendor-specific messaging and FlexConnect for port reversals.

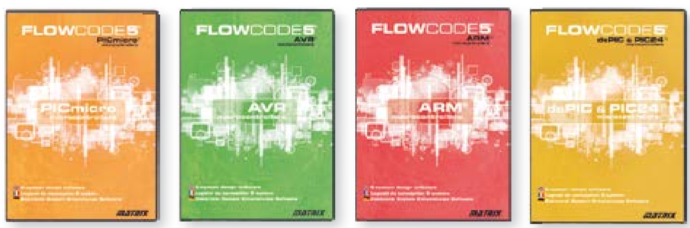
Low-power modes, such as Link Power Management (LPM), and advanced BC1.2, Apple®, SE1 and China battery-charging modes, enable UCH2 hubs to extend battery life and provide a replacement to wall chargers.
Free and low-cost development tools provide fast time to market for multi-platform USB2 designs:

- Free ProTouch Configuration Editor at www.microchip.com/get/2M4E
- USB2534 Eval Board (EVB-USB2534) for USB-charging designs

www.microchip.com/get/euUCH2 (130306-IX)

Developing and learni

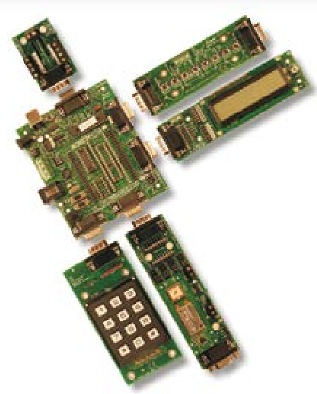
FLOWCODE5



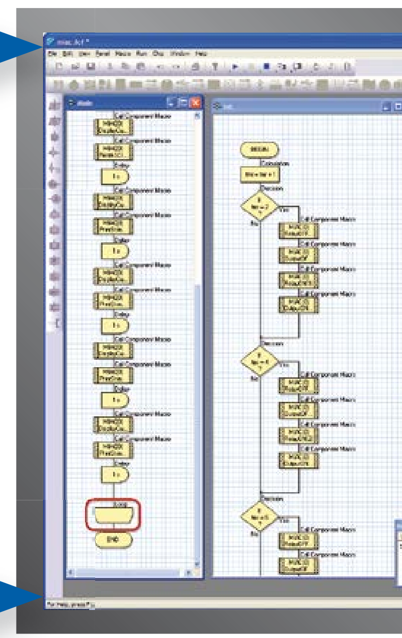
Flowcode 5 is one of the world's most advanced graphical programming languages for microcontrollers (PIC, AVR, ARM and dsPIC/PIC24). The great advantage of Flowcode is that it allows those with little to no programming experience to create complex electronic systems in minutes.

www.elektor.com/flowcode

… for electronics



E-Blocks are small circuit boards each of which contains a block of electronics that you would typically find in an electronic or embedded system. There are more than 40 separate circuit boards in the range; from simple LED boards to more complex boards like device programmers, Bluetooth and TCP/IP. E-blocks can be snapped together to form a wide variety of systems that can be used for teaching/learning electronics and for the rapid prototyping of complex electronic systems. Separate ranges of complementary software, curriculum, sensors and applications information are available.



… for industrial control

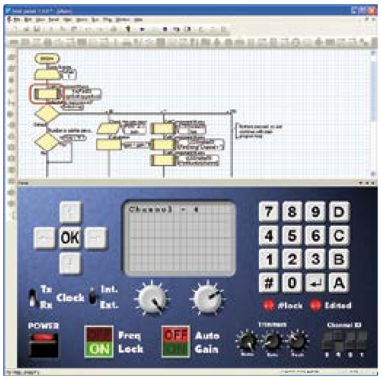
MIAC (Matrix Industrial Automotive Controller) is an industrial grade control unit which can be used to control a wide range of different electronic systems including sensing, monitoring and automotive. Internally the MIAC is powered by a powerful 18 series PICmicro device which connects directly to the USB port and can be programmed with Flowcode, C or assembly. Flowcode is supplied with the unit. MIAC is supplied with an industrial standard CAN bus interface which allows MIACs to be networked together.

Flowkit

Flowkit provides In Circuit Debugging for a range of Flowcode applications for PIC and AVR projects:

- Start, stop, pause and step your Flowcode programs in real time
- Monitor state of variables in your program
- Alter variable values
- In circuit debug your Formula Flowcode, ECIO and MIAC projects

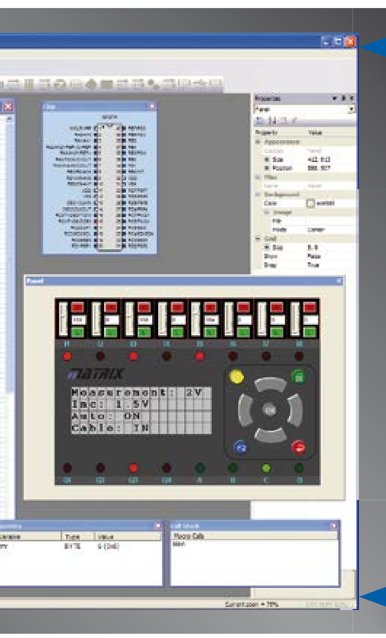
ng with Flowcode 5



New features in Flowcode 5

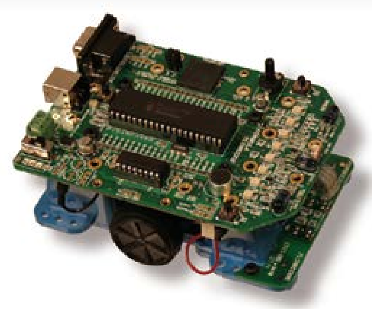
Flowcode 5 is packed with new features that make development easier including:

- New C code views and customization
- Simulation improvements
- Search and replace function
- New variable types and features, constants and port variables
- Automatic project documentation
- New project explorer makes coding easier
- Implementation of code bookmarks for program navigation
- Complete redesign of interrupts system allows developers access to more chip features
- Compilation errors and warnings navigate to icons
- Disable icons feature
- Improved annotations
- Improved links to support media
- Support for MIAC expansion modules and MIACbus



… for robotics

Formula Flowcode is a low cost robot vehicle which is used to teach and learn robotics, and to provide a platform for competing in robotics events. The specification of the Formula Flowcode buggy is high with direct USB programming, line following sensors, distance sensors, 8 onboard LEDs, sound sensor, speaker and an E-blocks expansion port. The buggy is suitable for a wide range of robotics exercises from simple line following through to complete maze solving. E-blocks expansion allows you to add displays, connection with Bluetooth or Zigbee, and GPS.



… for USB projects

ECIO devices are powerful USB programmable microcontrollers with either 28 or 40 pin standard DIL (0.6") footprints. They are based on the PIC 18 series and ARM 7 series microcontrollers. ECIO is perfect for student use at home, project work and building fully integrated embedded systems. ECIO can be programmed with Flowcode, C or Assembly and new USB routines in Flowcode allow ultra rapid development of USB projects including USB HID, USB slave, and USB serial bus (PIC only). ECIO can be incorporated into your own circuit boards to give your projects USB reprogrammability.

More information and products at:

www.elektor.com/eblocks

Heathkit IT-28
Capacitor Checker

By **Dan Koellen** (USA)

Figure 1. The IT-28 weighs a hefty 5 lbs (2.2 kg) and needs to be plugged in, while the battery powered Pico C-Super is a relative lightweight physically but not in measurement capabilities. The open magic eye shown here occurs when the bridge is balanced.

Recently I had the pleasure of building the Elektor Pico C-Super instrument [1]. I was impressed with the range of capacitance that could be measured, the accuracy and precision, and the ease of measurement; the frequency counter and signal generator were nice additions as well. While characterizing and using the Pico C-Super I remembered that I had an old Heathkit capacitor checker somewhere in my garage workshop. After searching I found the IT-28 capacitor checker a bit caked in dust and spider webs but looking in pretty good condition considering it was once a drenched victim of a fierce storm that blew away the backyard shed that it was residing in. The original assembly manual was in very good shape. The IT-28 and Pico C-Super are pictured together in **Figure 1**.

The glory years

In the 1960's and 70's the Heathkit Company was a well-known source of electronic kits successfully covering consumer devices, amateur radio, and test equipment. Though homebrewing was popular back then, projects such as an amateur radio transceiver could be difficult and costly for a hobbyist to build on their own from a schematic and locally sourced parts only. By contrast, kits came with highly detailed assembly manuals and all components and mechanical parts in a box delivered by the mailman. And since electronics were mostly hand assembled, kits could be offered at competitive prices.

Table 1. IT-28 Measurement Ranges and Standards for Capacitance and Resistance
Note: µfd = µF; µµfd = pF; nfd = nF.

Capacitance			**Resistance**		
Selection	**Standard**	**Range**	**Selection**	**Standard**	**Range**
C × .0001	200 µµfd (200 pfd) mica	10 µµfd (10 pfd) to 0.005 µfd	R × 1	200 Ω 1%	5 Ω to 5000 Ω
C × .01	.02 µfd (20 nfd) mylar	0.001µfd (1 nfd) to 0.5 µfd	R × 100	20 kΩ 1%	500 Ω to 500 kΩ
C × 1	2 µfd mylar	0.1 µfd to 50 µfd	R × 10 kΩ	2 MΩ 1%	50 kΩ to 50 MΩ
C extended	2 µfd mylar + 9 kΩ 1%	20 µfd to 1000 µfd			
External Standard		Maximum 25:1 ratio to external known standard	External Standard		Maximum 25:1 ratio to external known standard

From IT-11 to IT-28

Heathkit introduced the IT-11 Capacitor Checker kit in 1961. Minor changes were made in 1968 with the model number changing to IT-28, which was offered until 1977. The changes included a three-wire power plug, a spring clamp on the 6AX4 rectifier, upgraded capacitors, 120 V or 240 V operation, a new paint color and other cosmetic changes. The three tube IT-28 is an AC bridge circuit driven by an internal 60 Hz (mains) signal from a 1:2 transformer connected to the 6.3 VAC filament supply (**Figure 2**). The bridge may also be driven from a front panel sourced external signal. The IT-28 is much more versatile than its name implies since it is able to also measure resistance, inductance and transformer turns ratio. Capacitance and resistance are measured against internal standard components, while inductance and transformer turns ratio depend on external standards. A 1-kohm precision wire-wound potentiometer is split between the remaining two legs of the bridge to balance the bridge. Each measurement range is shown in **Table 1**; notice that each range is quite wide covering 500× for capacitance and 1000× for resistance.

The balance potentiometer's position is the front panel scale from which you obtain the value of the test capacitor, resistor or ratio. The potential across the bridge is AC coupled to the grid of the triode section of the 6BN8 tube acting as an AC amplifier. The other two sections of the 6BN8 tube are diodes connected as a half-wave doubler of the output of the triode AC amplifier. The resulting DC from the half-wave doubler is connected to the control grid of a 6E5 magic eye tube.

A magic eye tube, also called a cat's eye or tuning eye, had been used in radio receivers to indicate signal strength. The tube's glowing phosphor 'eye' closes as the control grid potential gets more negative. As a radio signal strength indicator, the more the eye is closed the better, i.e. the stronger the signal. But in this application bridge balance is indicated by the widest non-glowing section, i.e. the nulling of the bridge gives us an 'open eye'. The 6E5 'magic eye' driver circuit in the IT-28 is shown in **Figure 3**. It is a prominent element on the IT-28's front panel—see **Figure 4**.

Those leaky electrolytics

In addition to measuring capacitance the checker is able to determine if a capacitor is exhibiting leakage at working voltages from 3 volts to 600 volts. While checking for leakage the user

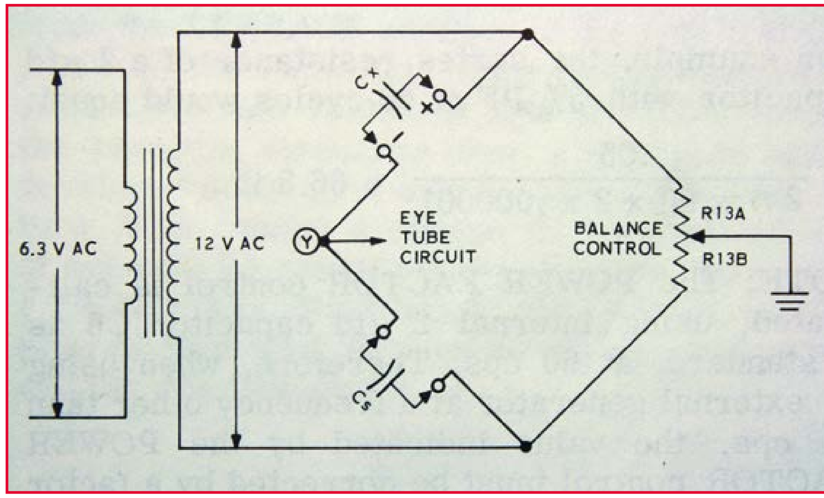


Figure 2. The bridge circuit as shown in the assembly manual. The unknown component value is read from the position of R13 on the front panel scale. The bridge is balanced when reactances are balanced: $X_{unknown} = X_{standard} \times (R_{13A}/R_{13B})$.

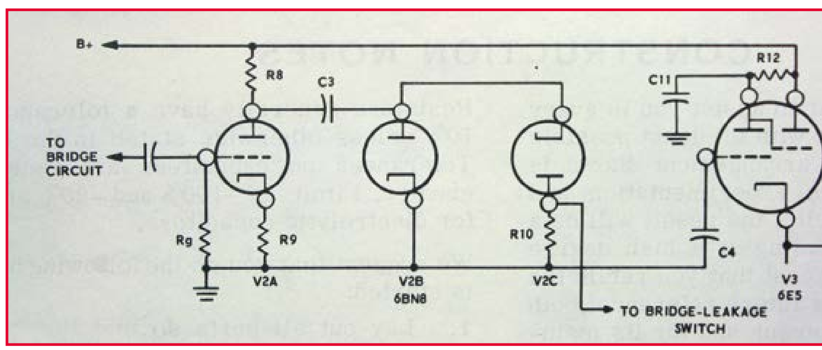


Figure 3. The 6E5 'magic eye' circuit from the assembly manual. The 6BN8 triode section is an AC amplifier followed by a voltage doubler using the same tube's two diode sections. A negative potential on the 6E5 magic eye control grid closes the eye.

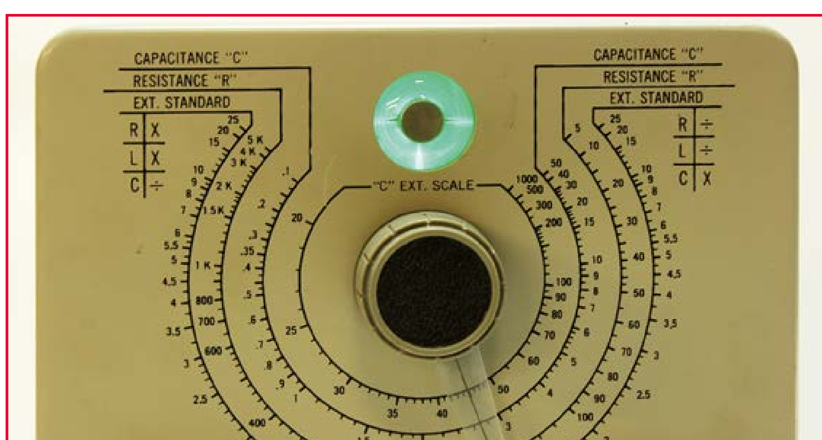


Figure 4. A close up view of the front panel scale and a closed eye.

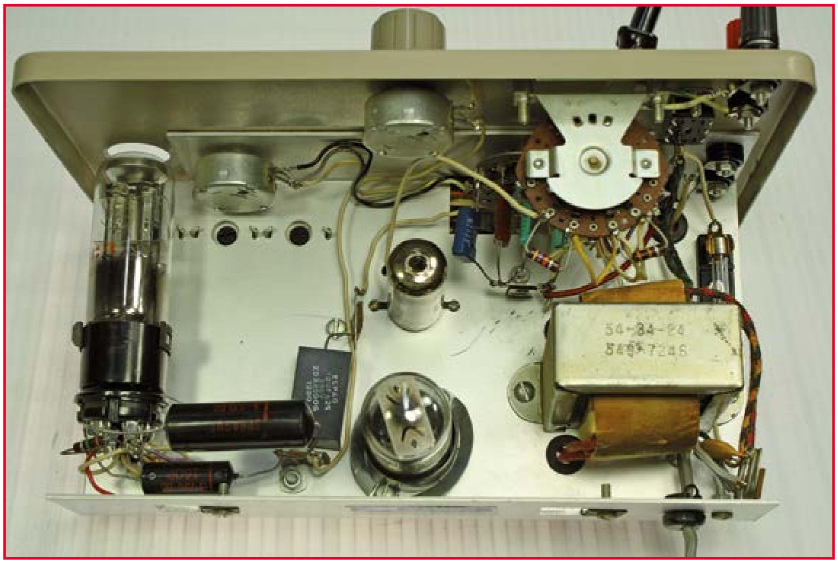


Figure 5. A view of the tube side of the chassis. The 6E5 magic eye is the horizontally monted tube on the left of the chassis. The center tube is the 6BN8 and the lower tube is the 6AX4 high voltage rectifier. A few of the standard capacitors and resistors can be seen at the right of the 6BN8.

has to choose whether an electrolytic, 'min.'lytic' or paper/mica capacitor was being measured. Initially I was puzzled by the term 'min. 'lytic' which I assumed to be a small value electrolytic capacitor. The Heathkit assembly manual somewhat cleared up the conundrum:

Figure 6. A view of the component side of the chassis. The leftmost wafer is the Bridge/Discharge/Leakage switch. The middle wafer switch is the voltage ladder for selecting the working voltage. The three potentiometers in the upper right adjust the leakage threshold for each capacitor type.

"NOTE: A MIN. 'Lytic (miniature electrolytic) can be distinguished from an electrolytic by its high capacitance, low working voltage and small size. Miniature electrolytics are usually encased in ceramic or plastic and are completely sealed."

Times have changed, now with our low operating voltages 'min. 'lytic' capacitors are the more commonly used electrolytic around. Back when this checker was designed, tube based equipment required power supplies delivering hundreds of volts, so electrolytics of very high working voltages and tens of microfarads were commonly used.

Leakage is measured by monitoring the charging current through the test capacitor. The charging current flows through a resistor to ground; the potential across the resistor is applied to the grid of the 6BN8 triode section. During leakage testing the 6BN8 is configured as a DC amplifier whose output is directly connected to the control grid of the 6E5 magic eye; the 6BN8 diode sections are not in circuit. Initially the charging current is high which causes the eye to close; when the capacitor is fully charged this current drops to zero so the eye opens back up. If the capacitor is leaky, current through the grid resistor will continue to flow keeping the eye closed. The value of the grid resistor is different for each type of capacitor as selected on the front panel, providing different leakage thresholds.

Power factor can also be measured, which is basically a measurement of effective series resistance (ESR). ESR has to be calculated using the formula given in the assembly manual.

The IT-28 awakened

After cleaning the outside of the metal enclosure I took a look inside; it was surprisingly clean—see **Figure 5**. All the solder joints looked good and the wiring appeared to be correct; there were no signs of burnt or broken components or other obvious problems. The date codes on the components are from the third quarter of 1972 so I estimate that it was built in the late part of that year, or in 1973.

One potential problem I found was a blown line fuse. I checked for any obvious shorts, checked the electrolytics for leakage using an analog ohm meter and verified the resistance of all the bleeder resistors. The two smaller standard

capacitors were verified to be good with the Pico C Super, a DMM was used to verify the standard resistors and the 2 µF capacitor standard. After replacing the fuse I crossed my fingers and, lacking a variac, plugged the unit directly into the wall outlet and flipped on the power switch. Fortunately there was no smoke, the fuse did not blow and I saw life in the tubes. An occasional arc and erratic switch action led me to more diligent cleaning of the chassis and switch contacts. After this second cleaning (**Figure 6**) there was no more arcing and the operation was much more consistent.
With all the tubes looking good and the 6E5 revealing a beautiful closed eye a sonnet came to mind:

All filaments are aglow
Not one red hot plate
No tubes with a purple glow
Now this will work great!

It was entertaining to watch the magic eye open when measuring various resistors and capacitors from my junk box. For small capacitors the eye opening can be easily missed. The manual suggests using 1000 cps (1 kHz) external signal for better eye opening with small capacitors, I did not try that but will be kept in mind for future measurements.

Enter Pico C
Now that the unit appears to work it was time to check the functionality and calibration. For calibration of component measurement the kit was shipped with a 200-ohm 1% precision resistor. The procedure was to simply measure this resistor and position the pointer on the bridge potentiometer until it is over 200 on the panel scale and assume the calibration will hold for all ranges and for capacitors as well. I verified the calibration with resistors and capacitors that I had checked with my DMM and the Pico C Super.

Advertisement

Table 2. Measurement results near range centers
Note: μfd = μF; μμfd = pF; nfd = nF.

Range	DMM	Elektor Pico C-Super	Heathkit IT-28
C ×.0001	-	210 pF	205 μμfd
C ×.01	-	20.8 nF	.0205 μfd
C × 1	1.96 μF		2.1 μfd
R x 1	220 Ω		223 Ω
R × 100	22.6 kΩ		22,700 Ω
R × 10 k	2.32 MΩ		2,350 kΩ

In the center of the range (and scale) the agreement was quite good (**Table 2**) but the performance degraded as one goes to the extremes of the range. This would be expected since the measurement is the ratio of the resistance on either side of the wiper of the balancing potentiometer. This is not linear, and quickly goes to zero or infinity at either scale extreme making the scale more crowded. Measurements away from the range center are good enough to determine if the test component is okay but will not provide much accuracy or precision. Another difficulty is the multiplicity of scales on the front panel and that the capacitance scale direction is opposite of resistance and ratio. This and the need for interpolation not only makes the IT-28 harder to use than the Pico C-Super or a DMM, but one is less confident of the resolution and precision of the measurement. Though the specification for minimum capacitance is 10 pF (10 μμF in old money) I would not trust the measurement near that extreme.

I had several unmarked inductors in my junk box that I was able to compare to a newly bought 'standard' inductor. This would have been very handy when I was experimenting with the AVR SDR project where I had to determine an inductor's value by finding the resonance with a known capacitor that was measured with the Pico C-Super.

Capacitor reforming

The calibration procedure for leakage was simply adjusting the threshold of current at which the magic eye just closes for each capacitor type. I also verified that the charging potentials were all within 10% of their selected value. To verify leakage operation I hoped a 1962 date code NOS 10 μF paper encased electrolytic from my junk box would be a good leakage candidate. I was thrilled when the magic eye would not open showing it was leaky. After repeating the measurement a few times I was quite surprised to find that the capacitor was no longer leaky! After a bit of research on the web I found that I had reformed the capacitor. The recommended procedure is to start at a low working voltage and work in steps up to the full working voltage rather than initially applying the full working voltage like I did. This is a handy feature of the IT-28 allowing you to reform your old capacitor stock.

Lightweight wins

While the IT-28 is a welcome addition to my bench I will still be using the Pico C-Super for most capacitance measurements. The ability of the much smaller and lighter Pico C-Super to measure capacitance under 1 pF while compensating for leads, the superb resolution, and the easy to read digital readout makes it the preferred instrument for capacitors smaller than 500 nF. The Pico C-Super is also a handy frequency and period meter and I use the square wave generator quite often. Likewise I will continue to use my DMM for resistance measurements and capacitors larger than 500 nF. But I will be powering up the IT-28 when I need to determine the inductance of a coil, find the turns ratio of an unmarked transformer, check capacitors for leakage, reform an old electrolytic from my junk box, or when I simply want to indulge in the pleasure of opening and closing the magic eye.

(130193)

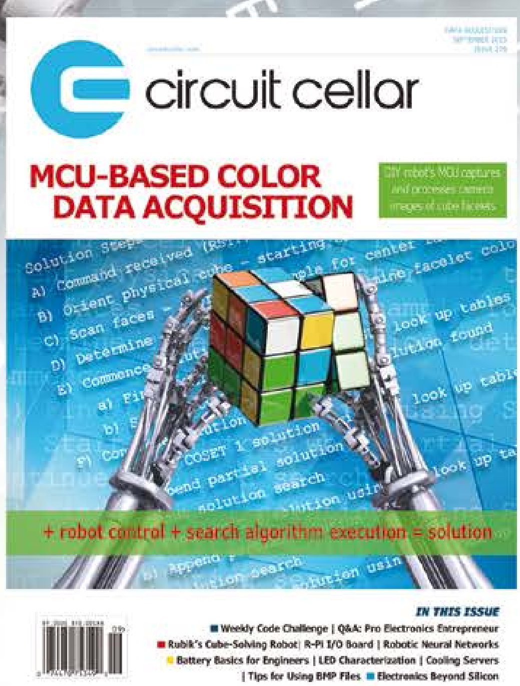
Internet Reference

[1] Pico C-Plus and Pico C-Super, Elektor February 2012, www.elektor.com/110687

ESTD 2004

Retronics is a monthly section covering vintage electronics including legendary Elektor designs. Contributions, suggestions and requests are welcome; please telegraph editor@elektor.com

We're celebrating a new look — and the next 25 years.



25% OFF CIRCUIT CELLAR

Whether it's **programming advice** or **design applications**, you can rely on *Circuit Cellar* for solutions to all your electronics challenges. **Raspberry Pi**, **embedded Linux**, **low-power design**, **memory footprint reduction** and more! Become a member, and see how the hottest new technologies are put to the test.

JOIN TODAY!

www.circuitcellar.com/sepN13

Hexadoku Puzzle with an electronic touch

With wintertime almost upon us it's a good idea to prepare for being outside less and devote time to the thought phase of projects and creations. One way of slowly adapting to deep thinking again is to solve our Hexadoku puzzle. Find the solution in the gray boxes, submit it to us online, and you automatically enter the prize draw for one of four vouchers.

The Hexadoku puzzle employs numbers in the hexadecimal range 0 through F. 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 4×4 boxes (marked by the thicker black lines). A number of clues are given in the puzzle and these determine the start situation.

Correct entries received enter a prize draw. All you need to do is send us **the numbers in the gray boxes**.

Solve Hexadoku and win!

Correct solutions received from the entire Elektor readership automatically enter a prize draw for one Eurocircuits PCB voucher worth **$140.00 (£80.00)** and three Elektor book vouchers worth **$60.00 (£40.00)** each, which should encourage all Elektor readers to participate.

Participate!

Before November 1, 2013,
supply your personal details and the solution (the numbers in the gray boxes) to the web form at
www.elektor.com/hexadoku

Prize winners

The solution of the July-August 2013 Hexadoku is: **3B4CD**. The Eurocircuits $140.00 (£80.00) voucher has been awarded to József Nagy (Hungary). The Elektor $60.00(£40.00) book vouchers have been awarded to Mary Chang (USA), Olavi Parkka, and Jacqueline Deletombe (France).

Congratulations everyone!

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

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

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

CIRCUIT CELLAR

[The CC Weekly <Code/> Challenge is underway!]

Each week, you'll find a new snippet of source code that contains one error.

```
#include <stdio.h>

int main (int argc, char *argv[])
{
  char ch;
  int i;

  for (i = 0; i < argc; ++i) {
    printf ("Argument #%d: '%s'\n", i, *argv[i]);
  }
}
```

sample code only

If you can find the error, you could be a winner!

Follow *Circuit Cellar* on Facebook and Twitter for information about each week's challenge, prizes, and winners announcements.

@circuitcellar
@editor_cc

circuitcellar

For complete details, visit
circuitcellar.com/cc_weekly_code_challenge

SPONSORED BY IAR SYSTEMS

Breaking the Laws of Physics

By **Gerard Fonte** (USA)

The crude quip for the three laws of thermodynamics is: 1) You can't win, 2) You can't break even and 3) You can't get out of the game. So you can imagine my surprise when I found out that every major manufacturer of "1000 watt" home theater systems (with speakers) specifies more output power than it consumes. They've apparently invented an entirely new game of thermodynamics!

The Line-up

Let's examine the JVC model TH-G31, LG model LHT-854, Panasonic model SC-BT730, Philips model HTS3371, Samsung model HT-C5500 and the Sony model HT-SF370. All of these are 5.1 channels, rated at 1000 watts of output power (class D) and include speakers. All of these, except the Philips, consume between 75 watts and 130 watts. These output power and input power values come directly from the operator's manual that is provided by the manufacturer (and is available at the manufacturer's website). The LG-854 input power was taken from the chassis nameplate because it is not listed (!) in the operator's manual. The Philips consumes 180 watts.

There are those that will dismiss this as just creative marketing. And that it's just a matter of how the power is distributed to the individual channels. It is true that it is marketing and that it is creative. It is also impossible. You can't get more power out than what you put in—Law #1 above. Every model, except the Philips, claims more power out for each and every individual channel, than it consumes. The per-channel power varies from a low of 125 watts to a high of 250 watts. The Philips outputs 167 watts for every channel (with 180 watts input).

The Federal Trade Commission (FTC)

But aren't there regulations that are supposed to protect the US consumer from fraud and misrepresentation? Yes. That's the FTC's job. And they have defined the testing standards required for amplifiers in CFR (Code of Federal Regulations) Title 16 Part 432 "Power Output Claims for Amplifiers Utilized in Home Entertainment Products" (available on-line). All "associated" channels must be driven to maximum output power for five minutes. According to FTC Staff Attorney Mr. Jock Chung, at least the right and left channels must be "associated" as specified in the Federal Register Notice of June 26 (Volume 75, page 3985). So, at least two channels must be driven at maximum power for five minutes. This makes things twice as impossible for all of them, except the Philips. It's only plain impossible to get 334 watts out (167 x 2) with 180 watts in, for that.

Perhaps the amplifiers consume more power during the test? Perhaps. But that raises a safety issue. The amplifiers would be using several times the power than was specified. This could overload branch circuits and potentially be the source of electrical fires. Product safety is something that manufacturers generally take very seriously.

Then there's that pesky FTC Part 432.5 clause. "No performance characteristics to which this part applies shall be represented or disclosed if they are not obtainable when the equipment is operated by the consumer in the usual and normal manner without the use of extraneous aids."

The Consumer Electronics Association (CEA)

The CEA is an international association of manufacturers that sets standards for their products. This is a voluntary agreement. These standards are not enforceable. The apparent CEA Standard is CEA-490-A R-2008 "Test Methods of Measurement for Audio Amplifiers". Unfortunately, "Self-powered loudspeakers...as well as manufacturer-packaged audio and home theater systems (systems that include loudspeakers) are specifically not covered by CEA-490-A." I talked to Mr. David Wilson of the FTC and he agreed that any home theater system that included speakers was exempt from CEA-490. The apparent reason was because the amplifier/speaker combination provided a fixed speaker impedance to the amplifier and made it more difficult to test. (Note that this means that computer amplifier/speaker systems are also exempt. They have some fantastic output power specifications as well.)

RCA RT2870

This all started when I bought an RCA RT2870 system rated at "1000 watts RMS" some time ago. The performance was beyond abysmal and when I investigated I was shocked and awed. (Note that the RCA name/logo was purchased by a Canadian company [Venturer Electronics Inc.] which has since disintegrated.) The speakers that are supposed to handle 167 watts RMS are marked as "8 ohm 60 W". So it is clear that the there never was any possibility of 167 watts going to any speaker.

I contacted Mr. David Hanna, Director of Consumer Affairs for RVA/Venturer with a written letter (yes--real snail mail). I eventually got a response from him that was illuminating. It included this sentence: "Furthermore there are far too many inconsistent tests done on consumer equipment for an engineer to put his or her faith into a consumer grade product." This is a remarkable statement. The RT2370 is clearly a consumer grade product. And what if you aren't an engineer?

FTC Again

I sent in a formal, written complaint to the FTC some months ago. The complaint was very detailed and consisted of about 75 pages of documentation. To date, I have not received any response at all. The FTC rules (432.4-c) state: "The rating and testing methods or standards used...are neither intended nor likely to deceive and confuse the consumer...". Do you think you are deceived and confused? If so, join the club.

(130310)

ORDERING INFORMATION

To order, contact customer service for your region:

USA / CANADA
Elektor US
111 Founders Plaza, Suite 300
East Hartford, CT 06108
USA
Phone: 860.289.0800
E-mail: service@elektor.com

Customer service hours:
Monday-Friday 8:30 AM-4:30 PM EST.

UK / ROW
Elektor International Media
78 York Street
London W1H 1DP
United Kingdom
Phone: (+44) (0)20 7692 8344
E-mail: service@elektor.com

Customer service hours:
Monday-Thursday 9:00 AM-5:00 PM CET.

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 for your region.*

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.

TERMS OF BUSINESS

Shipping Note:
All orders will be shipped from Europe. Please allow 2–4 weeks for delivery.

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, magazines, online publications and presentations. Elektor accepts no responsibility or liability for failing to identify such patent or other protection.

Copyright
All drawings, photographs, articles, printed circuit boards, programmed integrated circuits, discs, 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 educational 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.

MEMBERSHIPS

Membership renewals and change of address should be sent to the Elektor Membership Department for your region:

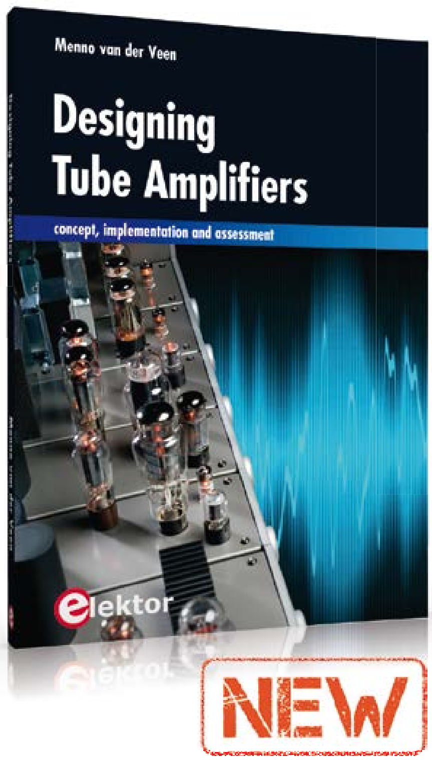
USA / CANADA
Elektor USA
P.O. Box 462228
Escondido, CA 92046
Phone: 800-269-6301
E-mail: elektor@pcspublink.com

UK / ROW
Elektor International Media
78 York Street
London W1H 1DP
United Kingdom
Phone: (+44) (0)20 7692 8344
E-mail: service@elektor.com

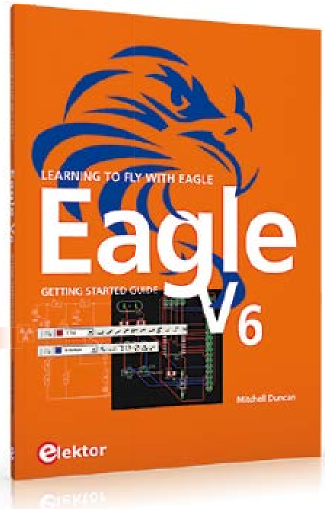
Do you want to become an Elektor GREEN or GOLD Member or does your current Membership expire soon?

Go to www.elektor.com/member.

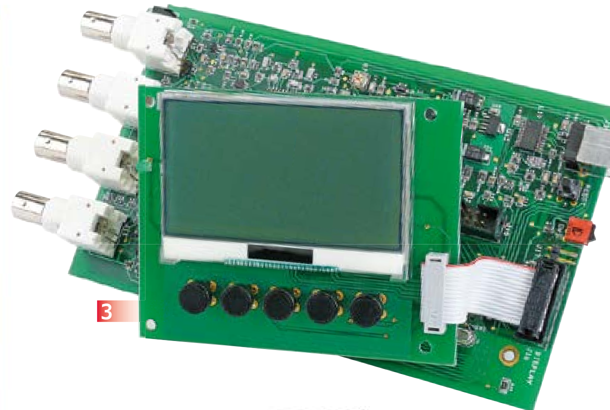
1



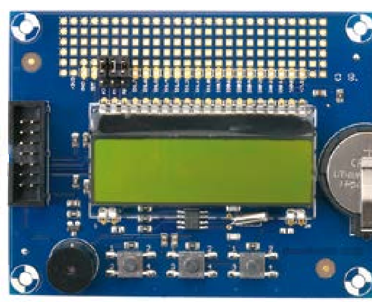
2



3



4



5

Concept, implementation and assessment

1 Designing Tube Amplifiers

This book looks at tube amplifiers from more than just a theoretical perspective. It focuses primarily on the design phase, where decisions must be taken with regard to the purpose and requirements of the amplifier, and it addresses the following questions: How do these aspects relate to subjective and objective criteria? Which circuits sound the best, and why? If you want to develop and market an amplifier, what problems should you expect? What are the significance and meaning of measurements? Are they still meaningful, or have they lost their relevance? Thanks to the enormous processing power of computers, we can now measure more details than ever before. How can these new methods be applied to tube amplifiers? Menno van der Veen will give you all the answers!

188 pages • ISBN 978-1-907920-22-6
£29.50 • € 34.50 • US $47.60

Learning to fly with Eagle

2 EAGLE V6 Getting Started Guide

This book is intended for anyone who wants an introduction to the capabilities of the CadSoft's EAGLE PCB design software package. This book will quickly allow you to obtain an overview of the main modules of EAGLE: the schematic editor; layout editor and autorouter in one single interface. You will apply your knowledge of EAGLE commands to a small project, learn more about some of the advanced concepts of EAGLE and its capabilities and understand how EAGLE relates to the stages of PCB manufacture. After reading this book while practicing some of the examples, and completing the projects, you should feel confident about taking on more challenging endeavors.

208 pages • ISBN 978-1-907920-20-2
£29.50 • € 34.50 • US $47.60

The luxury of precision within everyone's reach

3 500 ppm LCR Meter

The remarkable precision of this device and its amazing ease of use are the result of careful design. It works so well behind its uncluttered front panel that one could almost forget the subtleties of the measurement techniques employed. A dream opportunity for our readers who are passionate about measurement to enjoy themselves. If, like us, you wonder at the marvels modern techniques bring within our reach, come along and feel the tiny fraction of a volt.

Set: main board and LCD board, assembled and tested
Art.# 110758-93
See www.elektor.com/lcrmeter

Display, buttons, real time clock and more

4 Elektor Linux Board Extension

This extension board was developed to further propel our Embedded Linux series of articles and the matching GNUblin board. It has a display, buttons, a real time clock and 16 GPIOs. Linux devotees, switch on your solder irons. The Linux extension board includes everything needed to provide the user interface for a wide variety of projects!

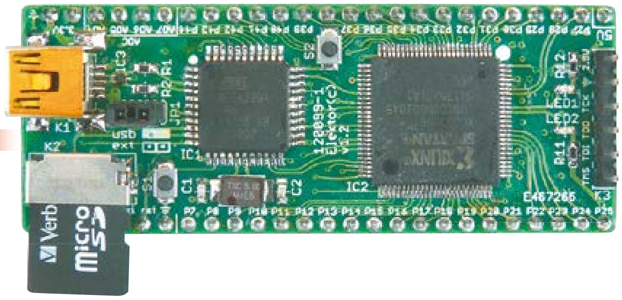
Module, SMD-populated and tested board, incl. LCD1, X1, K1-K4, BZ1, BT1 for home assembly
Art.# 120596-91
£31.10 • € 34.95 • US $50.20

140 Minutes video presentation and more

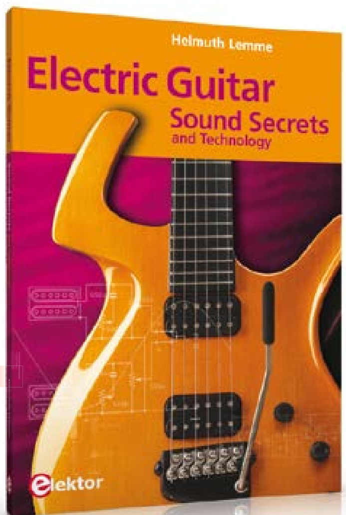
5 DVD Feedback in Audio Amplifiers

In this Masterclass we address several aspects of feedback in audio amplifiers. The focus of this Masterclass, although not entirely math-free, is on providing insight and understanding of the issues involved. Presenter Jan Didden provides a clear overview of the benefits that can be obtained by feedback and its sibling, error correction; but also of its limitations and disadvantages. Recommended to

6

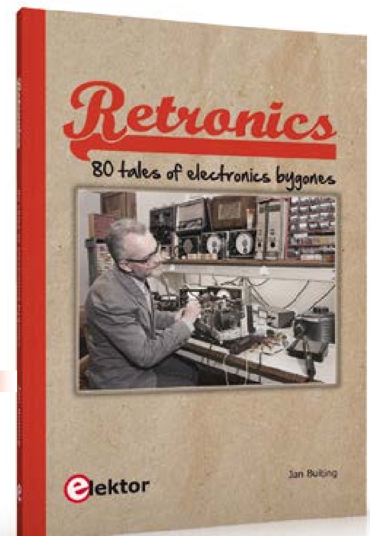


7



8

9



audio designers and serious audio hobbyists!

ISBN 978-907920-16-5

£24.90 • € 29.95 • US $40.20

Taming the Beast

6 FPGA Development Board

FPGAs are unquestionably among the most versatile but complex components in modern-day electronics. An FPGA contains a maze of gates and other circuit elements that can be used to put together your own digital circuit on a chip. This FPGA development board (designed in the Elektor Labs) shows how easy it is for any electronics enthusiast, whether professional or amateur, to work with these programmable logic devices.

Module, ready build and tested Art.# 120099-91

See www.elektor.com/fpgaboard

Sound Secrets and Technology

7 Electric Guitar

What would today's rock and pop music be without electric lead and bass guitars? These instruments have been setting the tone for more than forty years. Their underlying sound is determined largely by their electrical components. But, how do they actually work? This book answers many questions simply, in an easily-understandable manner. For the interested musician (and others), this book unveils, in a simple and well-grounded way, what have, until now, been regarded as manufacturer secrets.

The examination explores deep within the guitar, including pickups and electrical environment, so that guitar electronics are no longer considered highly secret. With a few deft interventions, many instruments can be rendered more versatile and made to sound a lot better – in the most cost-effective manner.

287 pages • ISBN 978-1-907920-13-4

£29.50 • € 34.50 •US $47.60

More than 75,000 components

8 CD Elektor's Components Database 7

This CD-ROM gives you easy access to design data for over 11,100 ICs, 37,000 transistors, FETs, thyristors and triacs, 25,100 diodes and 2,000 optocouplers. The program package consists of eight databanks covering ICs, transistors, diodes and optocouplers. A further eleven applications cover the calculation of, for example, zener diode series resistors, voltage regulators, voltage dividers and AMV's. A colour band decoder is included for determining resistor and inductor values. All databank applications are fully interactive, allowing the user to add, edit and complete component data.

ISBN 978-90-5381-298-3

£24.90 • € 29.50 • US $40.20

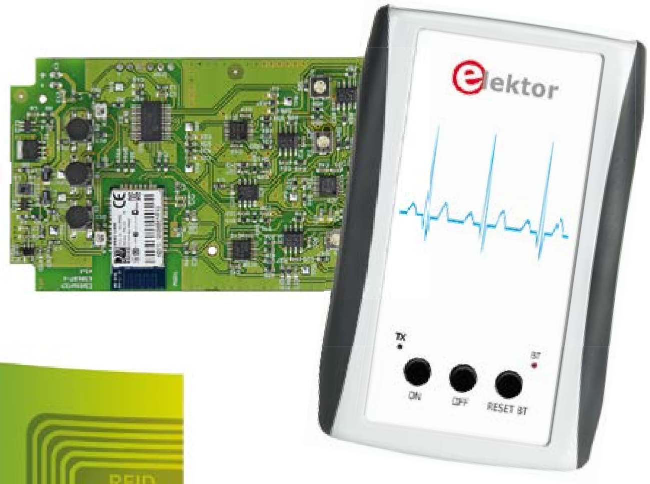
80 tales of electronics bygones

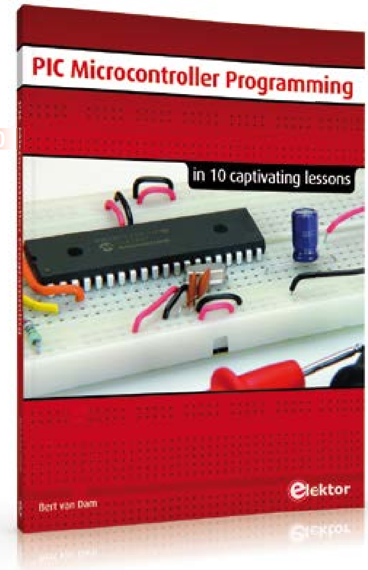
9 Retronics

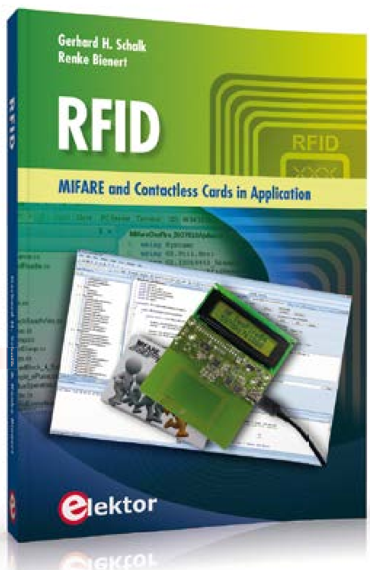
Quite unintentionally a one-page story on an old Heathkit tube tester in the December 2004 edition of Elektor magazine spawned dozens of 'Retronics' tales appearing with a monthly cadence, and attracting a steady flow of reader feedback and contributions to the series. This book is a compilation of about 80 Retronics installments published between 2004 and 2012. The stories cover vintage test equipment, prehistoric computers, long forgotten components, and Elektor blockbuster projects, all aiming to make engineers smile, sit up, object, drool, or experience a whiff of nostalgia. Owners of this book are advised to not exceed one Retronics tale per working day, preferably consumed in the evening hours under lamp light, in a comfortable chair, with a piece of vintage electronic equipment close and powered up.

193 pages • ISBN 978-1-907920-18-9

£24.80 • € 29.95 • US $40.00

9


10


11


12


Wireless and button-free

9 Android Elektorcardioscope

Instructive, fascinating, and potentially useful to everyone: perform your own electrocardiograms on your Android smartphone or tablet! The project involves skillfully combining a small PIC interface to control an analog input stage with a great deal of software. Our ECG interface is available in the form of a ready-to-use module to which you just have to add four electrodes and an Android application for smartphone or tablet; there's no physical connection between this terminal and the interface, as it uses Bluetooth communication!

Ready assembled board • Art.# 120107-91

See www.elektor.com/elektorcardioscope

10 captivating lessons

10 PIC Microcontroller Programming

Using the lessons in this book you learn how to program a microcontroller. You'll be using JAL, a free but extremely powerful programming language for PIC microcontrollers. Assuming you have absorbed all lessons you should be confident to write PIC microcontroller programs, as well as read and understand programs written by other people. You learn the function of JAL commands such as include, pin, delay, forever loop, while loop, case, exit loop, repeat until, if then, as well as the use of functions, procedures and timer- and port interrupts. You make an LED blink, build a time switch, measure a potentiometer's wiper position, produce sounds, suppress contact bounce, and control the brightness of an LED. And of course you learn to debug, meaning: how to spot and fix errors in your programs.

284 pages • ISBN 978-1-907920-17-2

£29.50 • € 34.50 • US $47.60

MIFARE and Contactless Cards in Application

11 RFID

MIFARE is the most widely used RFID technology, and this book provides a practical and comprehensive introduction to it. Among other things, the initial chapters cover physical fundamentals, relevant standards, RFID antenna design, security considerations and cryptography. The complete design of a reader's hardware and software is described in detail. The reader's firmware and the associated PC software support programming using any .NET language. The specially developed PC program, "Smart Card Magic.NET", is a simple development environment that supports sending commands to a card at the click of a mouse, as well as the ability to create C# scripts. Alternatively, one may follow all of the examples using Visual Studio 2010 Express Edition. Finally, the major smart card reader API standards are introduced. The focus is on programming contactless smartcards using standard PC/SC readers using C/C++, Java and C#.

484 pages • ISBN 978-1-907920-14-1

£44.90 • €49.90 • US $72.50

110 issues, more than 2,100 articles

12 DVD Elektor 1990 through 1999

This DVD-ROM contains the full range of 1990-1999 volumes (all 110 issues) of Elektor Electronics magazine (PDF). The more than 2,100 separate articles have been classified chronologically by their dates of publication (month/year), but are also listed alphabetically by topic. A comprehensive index enables you to search the entire DVD. What's more, this DVD also contains the entire 'The Elektor Datasheet Collection 1...5' CD-ROM series, with the original full datasheets of semiconductors, memory ICs, microcontrollers, and much more.

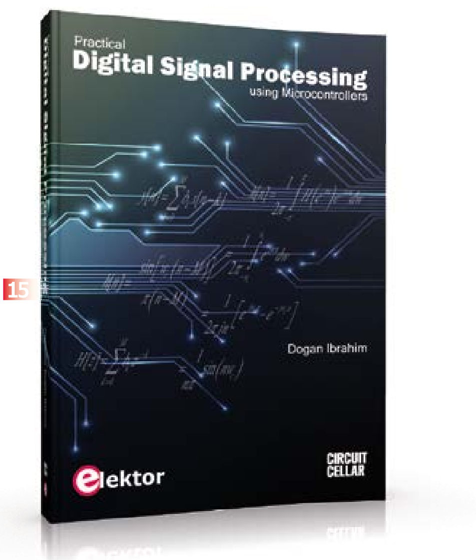
ISBN 978-0-905705-76-7

£69.00 • € 89.00 • US $111.30

13

14

15



Ultrasensitive wideband E-smog detector

13 TAPIR Sniffs it Out!

Attention boy scouts, professionals and grandfathers! This ultrasensitive wideband E-smog detector offers you two extra senses to track down noise that's normally inaudible. TAPIR — short for Totally Archaic but Practical Interceptor of Radiation — also makes a nice project to build: the kit comprises everything you need. Even the enclosure, ingeniously consisting of the PCB proper! Using the TAPIR is dead easy. Connect the headphones and an antenna and switch it on. Move it around any electrical device and you'll hear different noises with each device, depending on the type and frequency of the emitted field.

Kit of parts, incl. PCB
Art.# 120354-71
£13.30 • € 14.95 • US $21.50

LabWorX 2

14 Mastering Surface Mount Technology

This book takes you on a crash course in techniques, tips and know-how to successfully introduce surface mount technology in your workflow. Even if you are on a budget you too can jumpstart your designs with advanced fine pitch parts. Besides explaining methodology and equipment, attention is given to SMT parts technologies and soldering methods. Many practical tips and tricks are disclosed that bring surface mount technology into everyone's reach without breaking the bank. A comprehensive kit of parts comprising all SMT components, circuit boards and solder stencils is available for readers wishing to replicate three projects described in this book.

282 pages • ISBN 978-1-907920-12-7
£29.50 • € 34.50 • US $47.60

Ideal reading for students and engineers

15 Practical Digital Signal Processing using Microcontrollers

This book on Digital Signal Processing (DSP) reflects the growing importance of discrete time signals and their use in everyday microcontroller based systems. The author presents the basic theory of DSP with minimum mathematical treatment and teaches the reader how to design and implement DSP algorithms using popular PIC microcontrollers. The author's approach is practical and the book is backed with many worked examples and tested and working microcontroller programs. The book should be ideal reading for students at all levels and for the practicing engineers who may want to design and develop intelligent DSP based systems. Undergraduate students should find the theory and the practical projects invaluable during their final year projects. Similarly, postgraduate students should be able to develop advanced DSP based projects with the aid of the book.

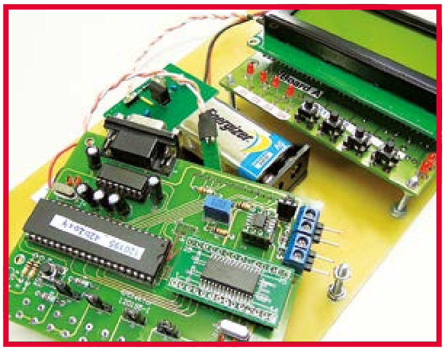
428 pages • ISBN 978-1-907920-21-9
£44.90 • € 49.90 • US $72.50

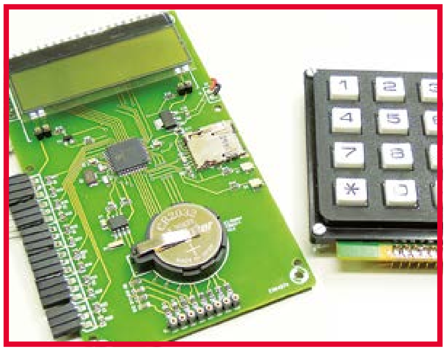
Further Information and Ordering: www.elektor.com/store
or contact customer service for your region

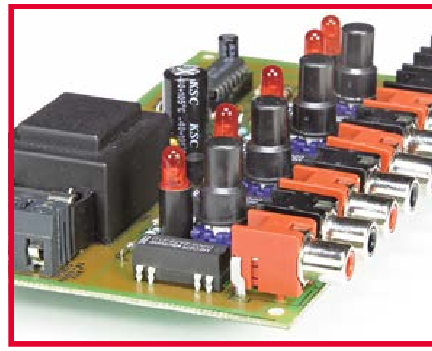
UK / ROW
Elektor International Media
78 York Street
London - W1H 1DP United Kingdom
Phone: +44 20 7692 8344
E-mail: service@elektor.com

USA / CANADA
Elektor US
111 Founders Plaza, Suite 300
East Hartford, CT 06108 USA
Phone: 860.289.0800
E-mail: service@elektor.com

NEXT MONTH IN ELEKTOR MAGAZINE







CAN Tester

Modern vehicles are moving networks really, with a variety of control units all interconnected via wires and even wirelessly. Many car manufacturers employ the CAN bus (Controller Area Network). CAN is marked by its high degree of immunity to interference. The downside of CAN is its complexity and the inherent difficulty to track down and repair failures. That's one of the reasons why the CAN Tester got developed.

Multi-Channel Temperature Logger

Logging multiple digital temperature sensors type DS18B20 at the same time? No problem with this 6-channel data logger with built-in Real Time Clock. A keypad is used for controlling the circuit, and a 4x20-character LCD for displaying the necessary information. Data is stored on an SD card, which makes it easy to open for processing on the computer.

Audio Switchboard

Got a faulty input selector switch on your amplifier? Or are you looking for just a simple but decent selector switch for your sound system? This circuit, built from conventional parts only (i.e. no SMD parts) lets you select four input source. It's even possible to make two sources active simultaneously without shorting them out.

Article titles and magazine contents subject to change; please check the Magazine tab at www.elektor.com.

Elektor November 2013 edition is published to Members on October 15, 2013.

See what's brewing @ Elektor Labs 24/7

Check out

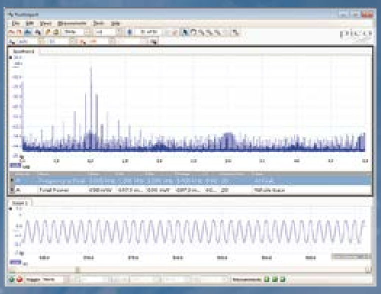
www.elektor-labs.com

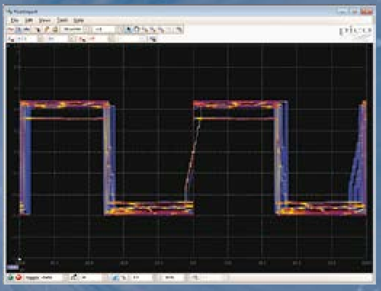
and join, share, participate!

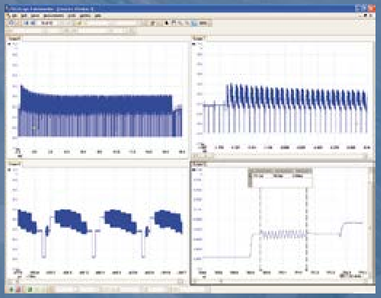
pico Technology

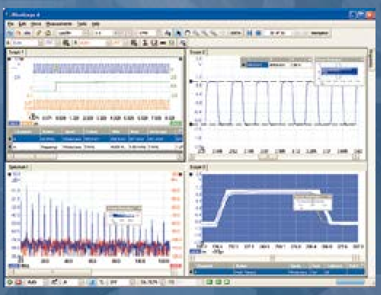
PicoScope® 6000 Series

- High-performance PC oscilloscope
- 500 MHz with 5 GS/s
- Industry-leading 2 Gsample buffer memory!
- High-performance function /arbitrary waveform generator
- SuperSpeed USB 3.0 interface







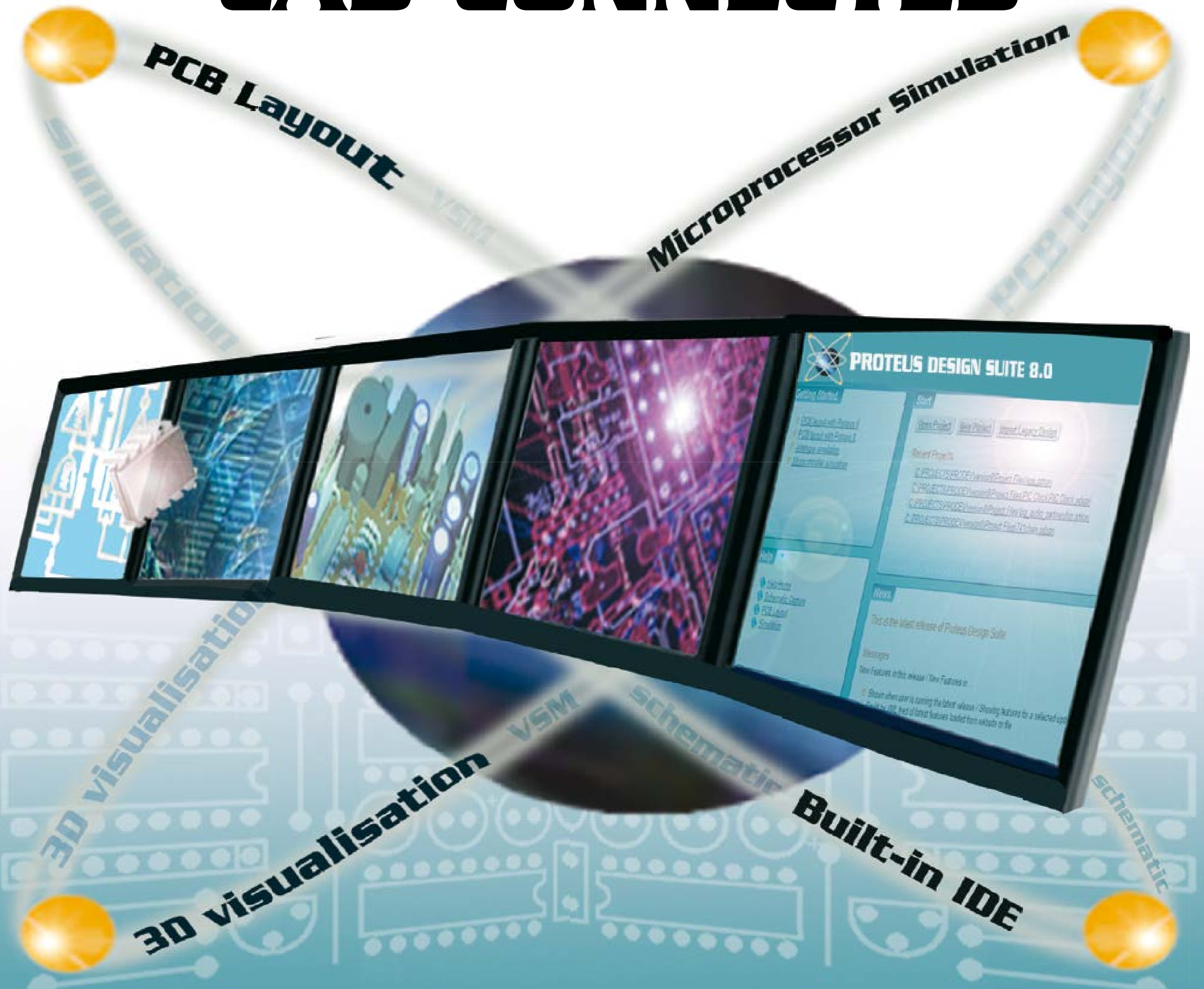


PicoScope	6402C	6402D	6403C	6403D	6404C	6404D
Channels	4					
Bandwidth	250 MHz		350 MHz		500 MHz	
Sampling Rate	5 GS/s					
AWG or Function Generator	FG	AWG	FG	AWG	FG	AWG
Memory	256 MS	512 MS	512 MS	1 GS	1 GS	2 GS
Price	£1995	£2495	£2995	£3495	£3995	£4495
Compatibility	USB 2.0 & 3.0, Windows XP, Windows 7 & Windows 8					

ALL MODELS INCLUDE PROBES, CARRY CASE, FULL SOFTWARE AND 5 YEAR WARRANTY. SOFTWARE INCLUDES MEASUREMENTS, SPECTRUM ANALYZER, SDK, ADVANCED TRIGGERS, COLOR PERSISTENCE, SERIAL DECODING (CAN, LIN, RS232, I²C, I²S, FLEXRAY, SPI), MASKS, MATH CHANNELS, ALL AS STANDARD, WITH FREE UPDATES.

www.picotech.com/PS236

CAD CONNECTED



PROTEUS DESIGN SUITE VERSION 8

Featuring a brand new application framework, common parts database, live netlist and 3D visualisation, a built in debugging environment and a WYSIWYG Bill of Materials module, Proteus 8 is our most integrated and easy to use design system ever. Other features include:

- Hardware Accelerated Performance.
- Unique Thru-View™ Board Transparency.
- Over 35k Schematic & PCB library parts.
- Integrated Shape Based Auto-router.
- Flexible Design Rule Management.
- Polygonal and Split Power Plane Support.
- Board Autoplacement & Gateswap Optimiser.
- Direct CADCAM, ODB++, IDF & PDF Output.
- Integrated 3D Viewer with 3DS and DXF export.
- Mixed Mode SPICE Simulation Engine.
- Co-Simulation of PIC, AVR, 8051 and ARM MCUs.
- Direct Technical Support at no additional cost.

labcenter Electronics

www.labcenter.com

Visit our website or phone 01756 753440 for more details

Labcenter Electronics Ltd. 21 Hardy Grange, Grassington, North Yorks. BD23 5AJ.
Registered in England 4692454 Tel: +44 (0)1756 753440, Email: info@labcenter.com