www.elektor-magazine.com

.magazine

April 2014

elektor

E-Lock
the First Elektor Chip

Some Say… E-Lock is unbreakable

You Crack E-Lock—You Win $25K
www.elektor.com/e-lock

● **E-Lock** | Platino-based Experimenter's PSU | **Motion-Detector Camera Trigger**
LED's Replace That Fluorescent Tube | **Microcontroller BootCamp** | Current Transformer Calculations | Precision Adjustable DC Current Source | **ATmega on the Internet**
Raspberry Pi Emulator | Elektorize Your Scope Probe
● **DuckDuckGo** | **A 2-value C?** | Unwrapping RepRap
● Atwater Kent Model 30 Radio | The Nearly Lost World of Elektor

US $9.00 - Canada $10.00
04
7 25274 24965 7

INNOVATIONS

Shop languages:

reichelt electronics

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

No hand free? No problem!

The integrated HF sensor activates this light simply by detecting motion!

LED motion detection light

The triggering microwaves pass through glass, plastic and even thin walls.

- ✓ Long life
- ✓ Low power consumption
- ✓ Low maintenance

- **Replaces conventional 40W light bulb**
- Socket: E27
- Angle of detection 360°
- Max. radius of detection 8 m
- 7 W, 460 lumens, 3000 K, warm white

15,92 (~ £ 13.21)
LED E27 7W HF

A EEC
E-27

OUR BLOCKBUSTERS

SMD LED spotlight
DELOCK

A EEC
GU 10

- Replaces 35 W halogen bulb
- Power consumption: 4 W
- Life: 20,000 hours
- 290 lm, warm white

DELOCK 46337 **4,41** (~ £ 3.66)

A+ EEC
GU 5,3

- Replaces 20 W halogen bulb
- Power consumption: 2.5 W
- Life: 30,000 hours
- 180 lm, warm white

DELOCK 46345 **3,74** (~ £ 3.10)

LED recessed spotlight
goobay

A+ EEC
G 4

- Replaces 20 W light bulb
- Power consumption: 2 W
- 170 lm, warm white

GB 30588 **3,03** (~ £ 2.52)

LED light
HEITRONIC

A EEC
E-14

- Replaces conventional 20 W light bulb
- Power consumption: 3 W
- Life: 25,000 hours

HEIT 16966 **4,29** (~ £ 3.56)

LED bulbs

Products online for virtually every application:

http://rch.lt/led

LED floodlight

made of weatherproof aluminium, for indoor and outdoor use

- Hardened protective glass
- Life: 30,000 hours
- Angle of radiation: 120°
- Adjustable metal clip

A EEC

10 W, cold white, 920 lm HEIT 37083 **12,56** (~ £ 10.43)

10 W, warm white, 870 lm HEIT 37009 **12,56** (~ £ 10.43)

HEITRONIC

SMD LED globe light
goobay

Energy efficiency class A

- **Replaces conventional 40W light bulb**
- Socket: E27
- Life: 30,000 hours
- Light colour: warm white
- 6.7 W, 470 lumens, 3200 K

SAVE ENERGY AT BEST PRICES!

A+ EEC
E-27

5,- (~ £ 4.15)
GB 30285

PRICE BUSTERS

Order now! **www.reichelt.com**

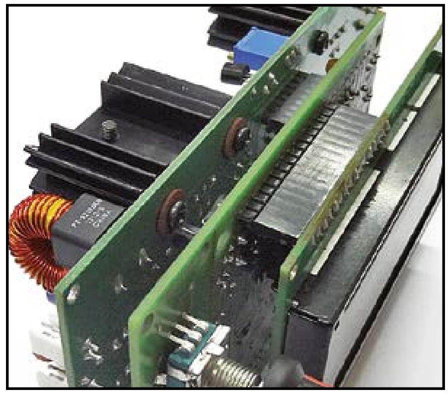
English Hotline: **+49 (0)4422 955-333**

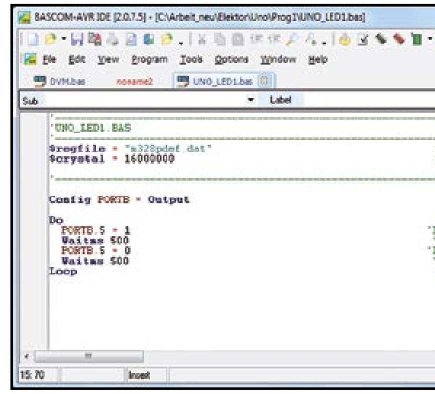
International payment method: PayPal

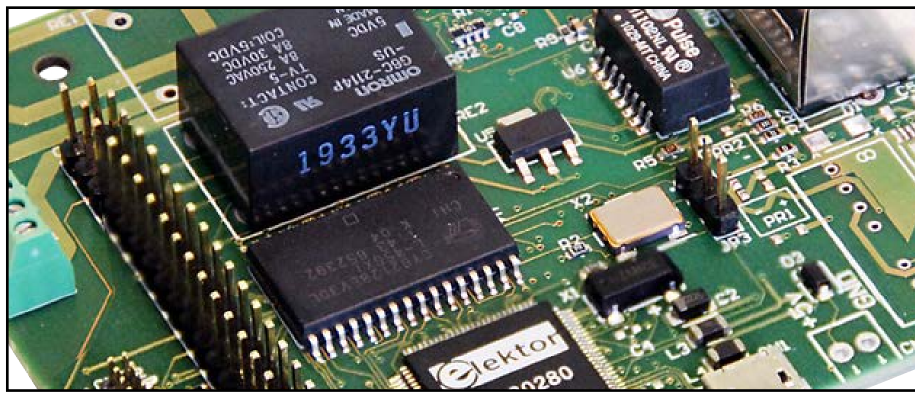
Daily prices! Prices as of: 10/02/2014


we CHALLENGE you!
Try beating our price.
If you do, we will match PLUS give you $100 towards your next order.
Just go to www.PCB4u.com and see why our pricing can not be beaten.
From Same Day Quick turn prototype to production in under 10 days
Full CAD and CAM review plus design rule check on ALL Gerber files.
Materials: Fr4, Rigid, Flex, Metal Core (Aluminum), Polymide, Rogers, Isola, etc.
HDI Capabilities: Blind/Buried Mocrovias, 10+N+10, Via-in-Pad Technology, Sequential Lamination, Any Layer, etc.
Our HDI Advantage: Direct Laser Drilling , Plasma De-Smear Technology, Laser Micro via Conductive Plate Shut, etc.
Accutrace inc.
www.PCB4u.com sales@PCB4u.com 408-748-9600

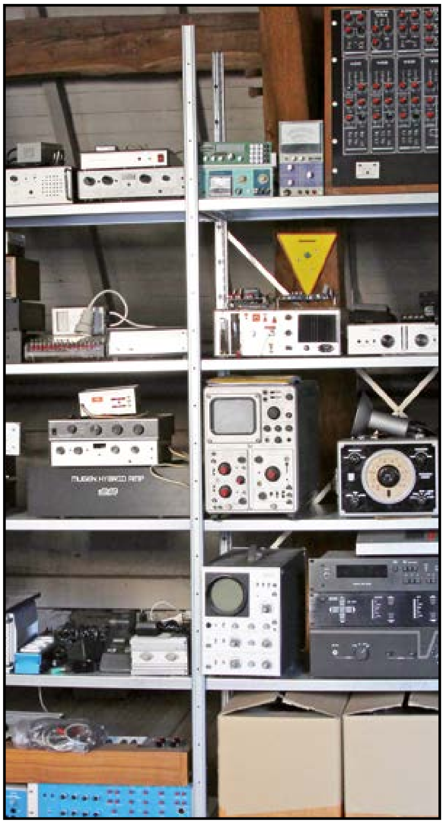
Contents

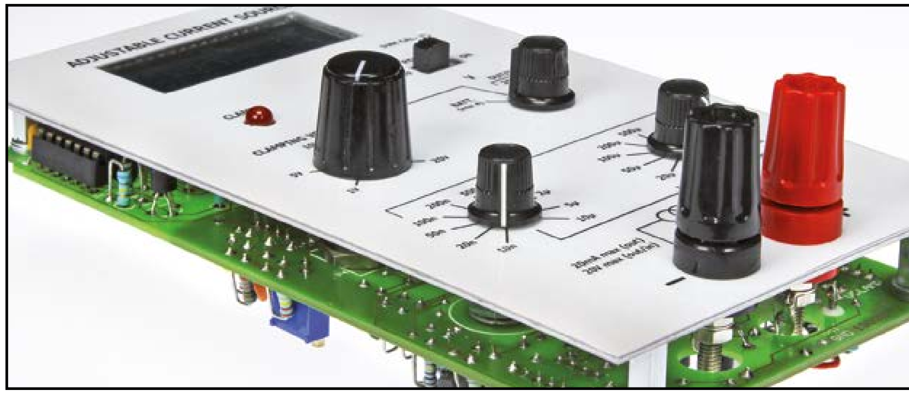


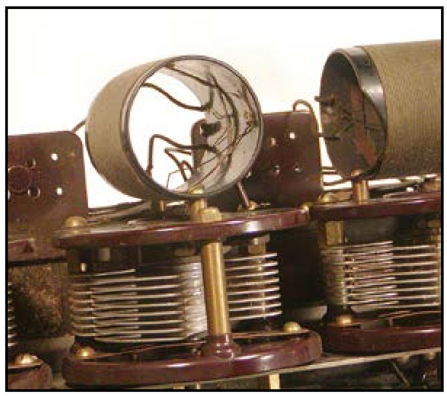

File Edit View Program Tools Options Window Help
'UNO_LED1.BAS
$regfile = "m328pdef.dat"
$crystal = 16000000
Config PORTB = Output
Do
PORTB.5 = 1
Waitms 500
PORTB.5 = 0
Waitms 500
Loop


1933YU
elektor

Volume 40 – No. 448 April 2014







Community

Volume 40, No. 448
April 2014

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

Printed in the USA   Printed in the Netherlands

From A to M and back in 92 pages

What used to be a hard division, Analog versus Microcontrollers, is mellowing to say the least. Our virtual e-hatchery website www.elektor-labs.com is a fine place to observe the trend: we find all-analog designs like audio amplifiers, bench instruments and radio projects receiving good ratings from the crowd there and progressing quickly to publication in Elektor magazine. On the other hand, totally microcontroller-oriented designers there are jubilant on discovering that a 2 dollar 8-pin IC called an ADC exists and can add considerably to the glory of their product.
As you will be able to confirm from browsing this copy of Elektor, the former sworn enemies A and M are beginning to reach out to each other with good results mutually. Let's see. The *Microcontroller BootCamp* course launched in this issue by veteran author Burkhard Kainka is aimed at those of you with classic electronics skills but hesitant till now about jumping the microcontroller bandwagon. After a good deal of deliberation we decided to steer clear of designing yet another microcontroller board specially for the course (what PIC, AVR, ARM?) and instead go for a cheap, powerful commercial product you should all know by now: Arduino. For software, we fire up Bascom because of its delightful simplicity.
Evidence of more interfacing between a microcontroller and analog circuitry is found in the *Platino Experimenter's Power Supply* on page 16. This clever little benchtop PSU has all the output voltages, currents and cooling typically needed by the microcontroller crowd: 0-15 V adjustable at 1 amp, and 3.3 V or 5 V selectable at .5 amps conveniently on a USB connector.
The articles *Current Transformer Calculations* (page 38) and *Precision Adjustable DC Current Source* (page 42) are a small and a large jewel of analog design respectively. Now, can we use a *RepRap 3D Printer* (page 66) to produce a plastic collet for the *Elektorized Scope Probe* (page 60)? After all, you really can't separate A and M.
Finally, the right-wing engine is not on fire and there is no April spoof in this issue. If you think there is, let me know the page number.
Enjoy reading this edition of Elektor

Jan Buiting, Editor-in-Chief

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

United Kingdom
Carlo van Nistelrooy
+44 20 7692 8344
c.vannistelrooy@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
Ferdinand te Walvaart
+31 46 43 89 444
f.tewalvaart@elektor.nl

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

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

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

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

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

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

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

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

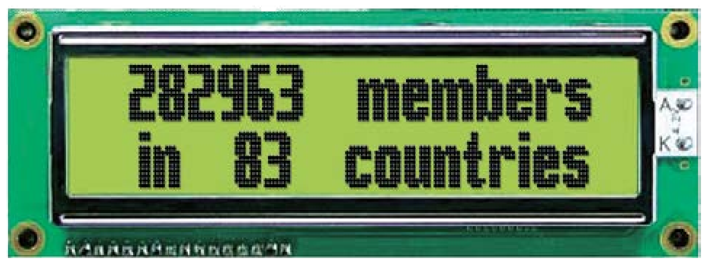
South Africa
Johan Dijk
+31 6 1589 4245
j.dijk@elektor.com

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

Our Network

elektor · VOICE COIL · circuit cellar · hifi.nl MUZIEK - FILM - LIFESTYLE · audioxpress · techthefuture

Connects You To



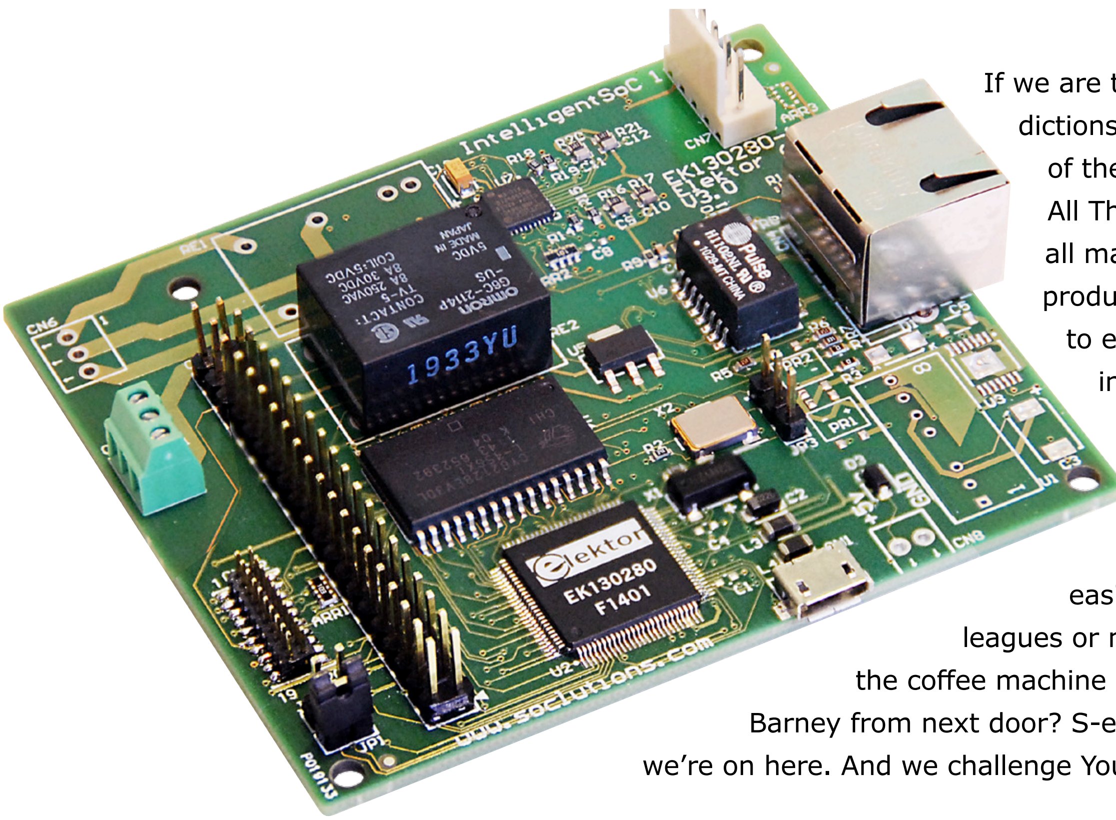
Supporting Companies

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

E-Lock, the First Elektor Chip

Some Say... E-Lock is unbreakable*



If we are to believe all the predictions, 2014 will be the year of the Internet of Things ... All Things! As a consequence all manner of electronic products will start chatting to each other, exchanging information over the Internet, the premise being that communicating with your coffee machine is easier than with your colleagues or neighbors—but how will the coffee machine know it is you and not Barney from next door? S-e-c-u-r-i-t-y, that's what we're on here. And we challenge You Hacker.

By
Eduardo Corral
(Spain);
Developed by
Intelligent SoC
(info@soclutions.com)

That coffee machine might be the least of your concerns, but when 'things' are valuable, you need take some serious precautions. Elektor has given this a good deal of thought and together with Intelligent SoC developed a highly dedicated chip to protect your 'things': meet E-Lock, the first Elektor Chip!

The E-Lock chip allows you to connect your control system to the 'Network of Networks' and monitor and control it from anywhere on or around the globe using your computer, tablet or smartphone without having to worry about the security of the connection and in full assurance of protection against intruders.

Remarkably, the levels of security we are applying to our computers and mobile devices has not been extended to the millions of webbed embedded systems out there, which represents a serious risk to the healthy expansion of the Internet of Things (IoT). Who in his/her right mind connects an embedded device to the Internet if it is wide open to attacks by hackers attempting to take control or modify its behavior?

E-Lock is the answer

Enough doom & gloom, let's get back to technology. We got in touch with embedded security specialists Intelligent SoC [1] and decided to jointly develop E-Lock, a chip—did we mention it's

** Courtesy Top Gear. We say: Hack E-Lock --> Crack the Code --> Win 61A8hex dollars ($25K). Check www.elektor.com/e-lock for details.*

the first Elektor chip?—specially designed for the IoT based on two fundamental conditions; 1) to allow the management of a device remotely via Internet and 2) do it in a completely secure way. The E-Lock chip is able to establish a secure connection over the Internet (TCP/IP) using the Transport Layer Security (TLS) encryption protocol. The chip also offers a 7-line GPIO (general purpose input/output), four 16-bit ADC channels, one 12-bit DAC and one I²C bus that allow control and communication with other peripheral devices. **Figure 1** shows E-Lock's functional diagram. The E-Lock chip (a SoC) also incorporates a real time clock (RTC) that takes the data from an Internet time server as reference, using the Simple Network Time Protocol (SNTP).

Evaluation board

The E-Lock chip comes in a 100-pin LQFP package, which is not easy to handle for manual assembly. That's why we designed the Evaluation (and Application) Board introduced here. The Evaluation Board will allow you to test the E-Lock chip while also serving as a basis for your own secure IoT projects. The board is available from the Elektor Store as number **130280-91**. Deeply technical descriptions of the board are available for downloading at [2].
As a get-u-going demo the board will allow you to control up to two relays and monitor the temperature of the room in which it is installed in a remote and secure way. Also on board is an expansion connector to host future extensions or simply open the door to developing your own applications. **Figure 2** shows the functional diagram of the evaluation board. We can distinguish the following sections:

E-Lock — The heart of the board. The device manages the communications and the operation of devices connected to it. E-Lock consist of two different communication protocols:

- Raw Ethernet configuration protocol; it's used to configure the main network and security parameters, such as IP address, Gateway address, DNS server, SNTP server as well as certificate, key and CA. It is done by using E-Lock unique MAC address.
- Secure TCP/IP client-server application protocol. Once E-Lock has been successfully configured, it acts as a secure TCP/IP server following the application protocol explained in detail in datasheet document [2].

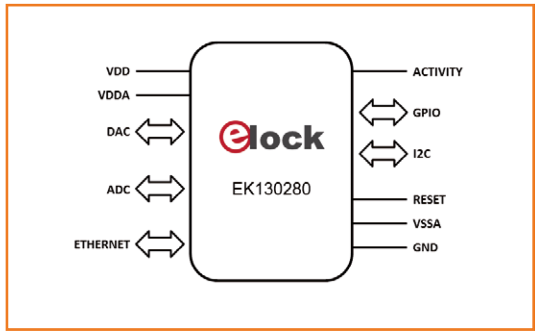


Figure 1.
E-Lock chip functional diagram.

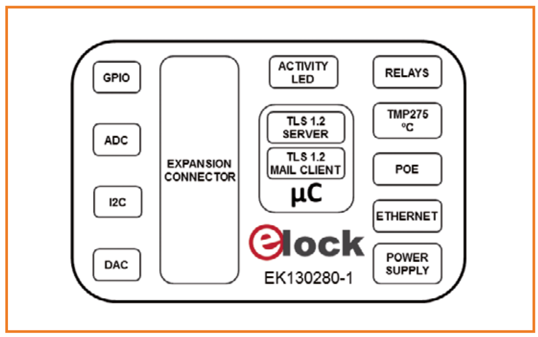


Figure 2.
E-Lock Evaluation Board functional diagram.

E-Lock Evaluation Board Key Features

- Secure bidirectional TCP/IP communication using TLS 1.2
- Software Watchdog
- Unique MAC address
- Raw Ethernet network configuration
- Supply Voltage: 5.0 V
- Temperature range: –40 ºC to 85 ºC
- RoHS compliant
- Ethernet Controller with RMII interface to PHY
- Power on Ethernet (POE) circuit (Option)
- 50 MHz System clock
- 1 I²C Temperature sensor
- 2 Relays (one on board)
- 7 General Purpose Input/Output lines
- 4 Analog-to-Digital Converters, 16-bit
- 1 Digital-to-Analog Converter, 12-bit
- 1 Inter-Integrated Circuit (I²C) bus
- External temperature sensor (with connection cable)
- Expansion connector
- Self configurable
- Supplied ready-assembled and tested through Elektor Store
- Key component to participation in $25K competition at **www.elektor.com/e-lock**

If you are designing a dead secure

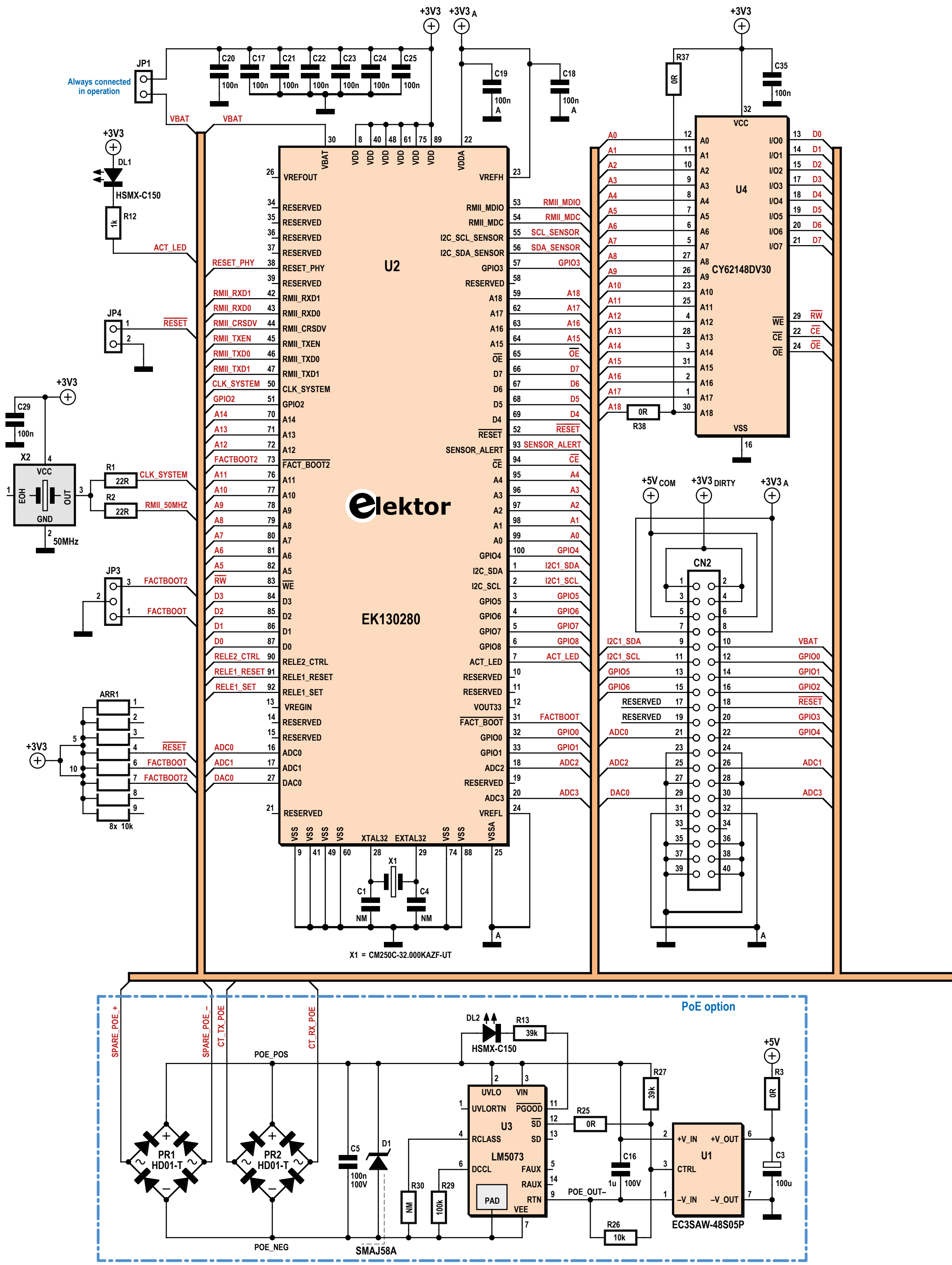
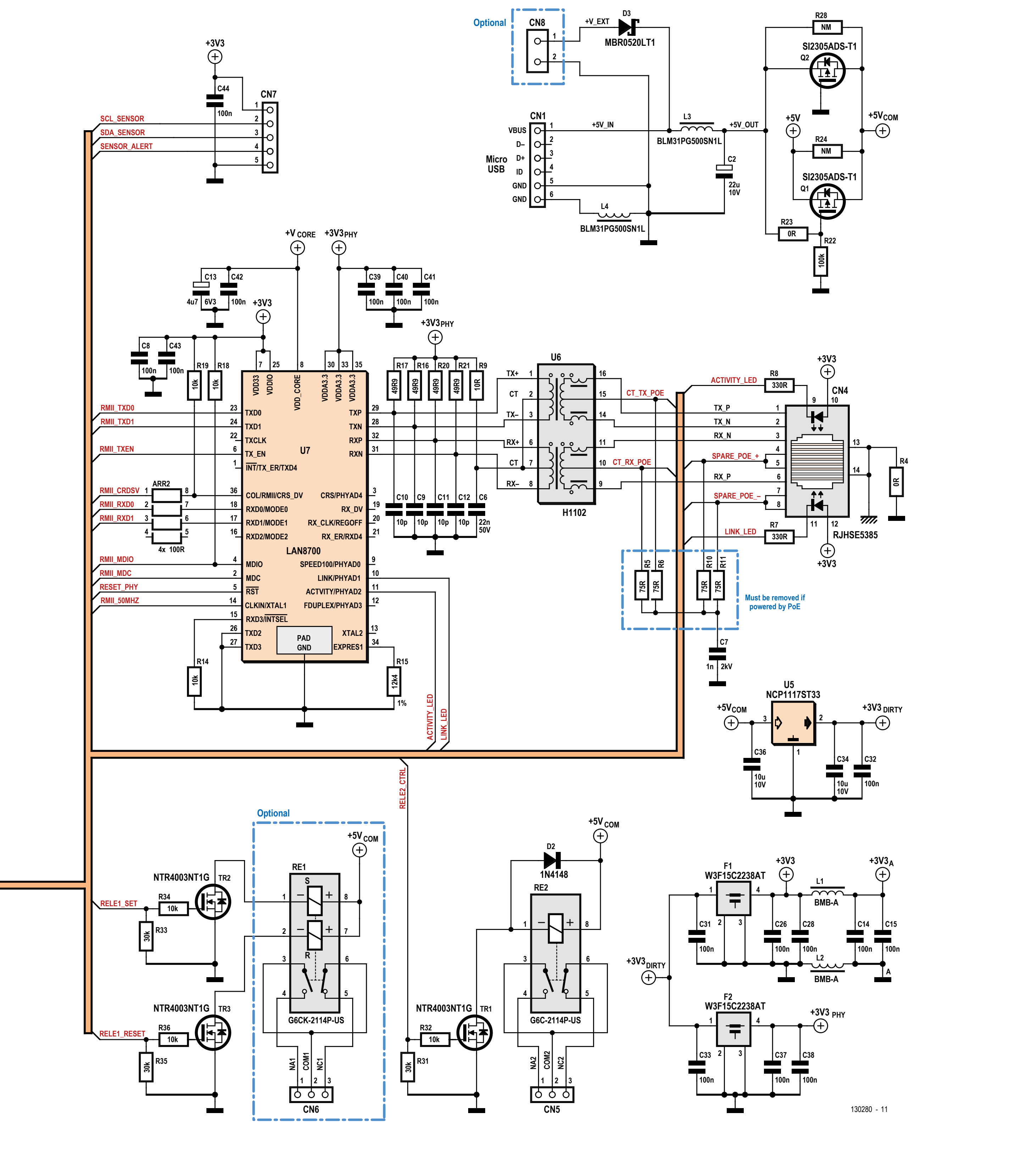


Figure 3.
Full schematic of the
E-Lock Evaluation Board.

IoT device, E-Lock is the solution.



The basic commands supported by E-Lock to manage on-board devices are:

- Set sensor configuration;
- Set relay configuration;
- Get sensor configuration;
- Read sensor value;
- Get relay status;
- Set relay status.

The commands to act on devices connected to expansion connector are:

- Set Configuration: Initialize GPIOs, ADC channels, DAC channels and I²C devices with specific device parameters;
- Get Configuration: Used to know current device configuration;
- Read: Capture GPIs, ADC and DAC channels and I²C devices current values;
- Write: Set GPOs, DAC channels and I²C devices values.

Additional system commands are:

- Set system configuration;
- Get version;
- New Certificate;
- New Private Key;
- New CA (Certification Authority) Certificate;
- EMail notify;
- Bootloader (download new firmware).

Power Supply — The board requires 5 volts for operation, conveniently supplied by a power adapter as used on smartphones, through micro USB connector CN1 or directly by connecting a 5-V stabilized power supply on the CN8 terminals—paying attention to the polarity indicated on the silkscreen.

Ethernet — The board is connected to a 10/100 Base-T Ethernet network through a standard RJ45 connector, which incorporates the corresponding 'Activity' and 'Link' LED indicators. Between the connector and the E-Lock chip we find the usual transformer block and the Physical Layer Transceiver (PHY) chip.

PoE — The board is Power over Ethernet (PoE) ready, although the version currently available thought Elektor Store does not include the 48 to 5 V DC/DC converter.

Output circuit — The board has two output circuits that control two relays: RE1, which is a dual coil latching type, and RE2, which is a non-latching type. Both are configured as SPDT (single-pole, double-throw*)* switches and contacts are available on the terminals of CN6 and CN5 connectors respectively, where the central terminal is Common.

On the version made available through the Elektor Store only one of these sections is assembled: RE2 relay and its corresponding connector CN5.

Temperature Sensor — The I²C temperature sensor has been separated from the main board to avoid the heat dissipated by the components installed on the motherboard affecting the sensor measurements. Because the TMP275 is an SMD device, it comes assembled on a small PCB and is connected to the main board on CN7 connector using a 7 inch long (approx. 20 cm) patch cable.

Expansion Connector — The goal we have pursued since the beginning of the design process is for the board to be more than a means of demonstration and eventually serve as a platform for our readers' IoT projects and/or your own extension projects. Thus, E-Lock pins not used for on-board peripherals have been routed to expansion connector CN2. It puts +3.3 V and +5 V supply voltages at your disposal, as well as an additional I²C bus, four ADCs, one DAC, seven GPIOs and the Reset signal.

The practical implementation of the functional diagram is found in the hefty schematic in **Figure 3**. Rather than a wall poster showing the Things Embedded Dreams Are Made Of, the schematic is printed here in support of all Elektor readers seriously interested in Internet security AND winning $25K in the competition at **www.elektor.com/e-lock**.

Installation and configuring

Before applying power to the E-Lock Evaluation Board, download the demo software for PCs, which is available on the project page [2] as archive file 130280-W.zip, and unzip it to a folder on your hard drive.
In the *ISLElektor* folder, find all the files needed to configure and test your board. The 'doc' folder contains 'network-configuration.pdf', which describes the E-Lock configuration process, and

'ISLElektor_130280-applicationnote.pdf', which describes the demonstration application. It is highly recommended to read them before you start using the board.
You need to configure the board before testing it and to do so the board needs to know some information about your network: router IP address, a free IP address for your board and DNS IP address.

After collecting all this information, connect the power supply and Ethernet cable to the board. Then run the application configuration *ISLRaw.exe* and you hopefully will see the window shown in **Figure 4**.
The application is able to configure various devices attached to the same LAN. The first step is to identify any MAC addresses by clicking on *Scan MAC*; doing so, a message will be broadcast by the application and the chart situated at the bottom of the window will be filled with the information given by the devices that are waiting to be configured.
Once all the information has been received select the device you want to configure in the list and the *Remote MAC* field will be filled automatically. Finally write the desired IP address and press *Set Remote IP* in order to configure the IP address of the device. If process was successful, the list will be updated with the new IP.
A new window shows up to select security parameters, that is: certificate, key, CA certificate, SNTP server, DNS address and Gateway address. Once all fields mentioned above are filled correctly, press *Send* to set up the security parameters on E-Lock. You can create your own certificates following the instructions described in the configuration document but for testing the board you will find sample certificates and key files in the *certs* folder, select *ISLserver-cert.pem* for the *Cert* field, *ISLserver-key.pem* for *Key* field and *ISLca-cert.pem* for the *CA* field.

Demonstration software

Once the setup process is completed, proceed to test the operation of the E-Lock board running the demo application *ISLElektor.exe*. **Figure 5** shows the main window of this application. Before running the application, the temperature sensor board must be plugged into the motherboard.
ISLElektor is an application used to interact with the E-Lock Evaluation Board, exchanging information through a TCP/IP channel in Secure Mode, based on a communication protocol described in full in the board's datasheet [2].
When *ISLElektor* is launched, all the buttons are disabled except *Connect*. First of all, it is necessary to enter the IP address you have assigned to your board in the previous step, then press *Connect* to establish a TCP/IP connection with the device. When the connection is ready, a new window is shown similar to **Figure 6**, and you have to set the server certificate and key files. As for the board, you will find the sample certif-

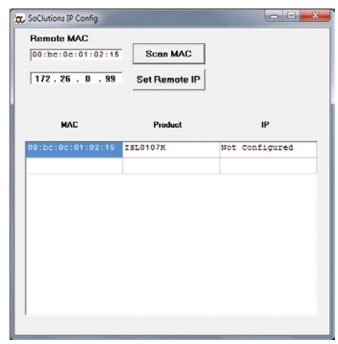


Figure 4.
E-Lock Evaluation Board configuration.

Figure 5.
E-Lock Evaluation Board demo application main window.

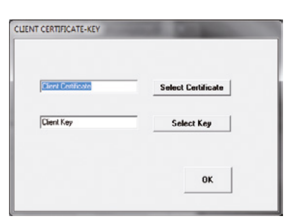


Figure 6.
Setting up client certificates on the E-Lock Evaluation Board.

Figure 7.
E-Lock Evaluation Board successfully connected.

The security trap

Being secure and feeling secure are different things. Together with millions of others we use the Internet to access and convey information, meet up with family and friends and cheerfully complete transactions with banks and online stores. For peace of mind we look at the https sign in the upper left part of the screen… and feel secure. But are we? Very likely, not. And there are more ways to get to your valuable stuff. Many banks have taken a lot of precautions to secure their ATMs and then some crooks do a ram-raid and run with the money—so much for security.

With the IoT taking off at terrific speed, there will be many highways and byways to get into systems and houses. So, what we present here is a solution to secure the connection to your Things, for example the electronics to open your garage door, but you will have to keep checking the windows!

About Intelligent Soc

IntelligentSoC was founded in Madrid, Spain in 2011 as a spinoff of the R&D department of **Datatech SDA**, an engineering company which is specialized in the development and design of communication and data processing systems with a strong focus on security.

The main goal of the company is to design, develop and manufacture IPs, chips and modules with the latest authentication technologies based on elliptic curve math (ECC), Diffie-Hellman algorithms, and various AES 128/256 encryption techniques.

IntelligentSoC's products are currently installed where security is a must-have, such as avionics or military systems, and are currently expanding this technology to other areas like the Internet of Things.

icate and key in the *cert* folder *(ISLclient-cert.pem and ISLclient-key.pem*).

If the connection process is successful, a *Connected* tag will appear in the *Connection Status* box with a blue background, and all the onboard functionalities will be available (**Figure 7**). If not, an *Error* tag will be displayed. If so, no sweat, check the IP address, the Ethernet cable and TCP/IP client Certificate and Private Key and then retry the connection. If you are sure all parameters are correct but the error persists, use the factory settings recovery jumper JP3 (install it, remove it) to initialize E-Lock and repeat the above configuration process.

Click on *Disconnect* to finish the previous connection or modify the IP address field and press *Change IP* to boot the device with the newly filled IP address. In this last case, it is mandatory to connect again using the new IP address.

Now you can test the operation of all available elements on the board, change their initial configuration, play with the expansion connector … detailed information about all the possibilities offered by this demonstration software is found in the 'ISLElektor_130280-applicationnote.pdf' document found in the *doc* folder.

The folder called *source* contains the source files of this application and we encourage you to make your changes or use them as a basis for your own applications. We have used Borland C++ Builder6.

More info …

We have just skimmed the surface of E-Lock on these pages. Magazine space is at a premium and there is a lot more to describe in terms of what E-Lock has to offer. I highly recommend downloading all documents available on the project page [2] including the PCB design data, the E-Lock board datasheet and demo and support software. Reading these documents will not just enable you to understand the real power of the first Elektor chip, but also provide essentials to win $25,000 in cash [3].

(130280)

Web Links

[1] www.soclutions.com

[2] www.elektor.com/130280

[3] www.elektor.com/e-lock

Platino-based Experimenter's Power Supply

A handy supply for the embedded electronics worker

By **Sunil Malekar / Krishnachandran**
(Elektor India Labs)

Although many benchtop power supplies boast a microcontroller of some sort, the MCU's function is usually limited to controlling the V/A readout. Not so in this project, which employs Elektor's versatile 'Platino' ATmega powered board to do all of the PSU's control functions, including its internal V and A regulation and fan control.

The small power supply described here was inspired by the "Lab PSU for Embedded Developers" published earlier in Elektor [1]. The obvious shortcoming of that design is that the activity of the microcontroller is limited to the display. By contrast, in the Platino-based PSU described here, the MCU rules over output voltage, maximum output current, and a lot more. And a powerful controller it is: Elektor's very own "Platino" [2].

Design considerations

Designing an MCU into a benchtop power supply is all about adding a few extra parts in order to reduce the number of potentiometers and switches. The process is fraught with problems but thanks mainly to Platino, some Bascom programming and a component called Digital Pot, we were successful and managed to minimize component count and board space.
The PSU delivers the most common outputs for 'light' lab work—meaning 0 V to 15 V DC in 0.1-V increments, with a 0 mA to 1000 mA selectable current limit. The PSU also provides for fixed 3.3 V and 5 V voltages on a USB connector with 0 to 500 mA adjustable current limit. The instrument display offers two modes of operation: Setup Mode and Normal Mode.

Circuit description

The Platino Benchtop PSU has a hardware and a software component—both interact strongly to guarantee stability and accuracy of user selections and output values. The heart of the project is Elektor's Platino MCU board powered by an ATMega328p.
Starting with the hardware, let's do some sightseeing of the schematic in **Figure 1**. But before hopping on the tour bus, note that the input voltage for the unit comes from a standard 18-VDC adapter normally used with a laptop PC, or an equivalent adapter.

Main power input and auxiliary supply generator

The raw 18 VDC input voltage is connected to circuit on PCB terminal block K1. Diode D1 protects the circuit form reverse polarity. The raw DC is applied to regulators IC3 and IC4. IC3, an LM7805, supplies the main 5-V power for the Platino board and all other circuitry operating at the same voltage. IC4, an LM7812, is responsible for the regulated 12 VDC needed by the fan control circuit.

Negative auxiliary voltage generator

The circuit around IC5 and IC6 converts +5 V, via a –5 V intermediate level, down to –1.27 V ($-V_E$), which is needed by the positive voltage regulator section to Make its output go down to 0 V. The LM337L negative voltage regulator has its output voltage adjusted to –1.27 V by preset P2.

Temperature measurement

Temperature is measured by an LM335 precision temperature sensor, IC7. The output of the

'335 is applied to Platino's PC4 input, enabling the MCU to read and process the sensor voltage and then display the temperature. The measured temperature is used to automatically shut the power supply down when the PSU temperature reaches a predetermined high level. You can set the shutdown level in Setup Mode under "Threshold Temperature".

Fan control

A small fan fitted in the enclosure is used to cool the unit and prevent damage by overheating. The 12-VDC fan is powered by a BS170 MOSFET (T1) via Platino's PD1 port pin. Connector K8 may be used as a jumper to set the fan operation in Manual Mode by applying a permanent High (no jumper) or Low level (jumper installed) at Platino's PC6 pin.

The operation mode of the fan can be set to Auto or Manual by entering the instrument's Setup Mode. In Automatic mode the fan is Platino-controlled and is switched on at 20 °C below the user-defined Threshold Temperature value. Below that level, the fan remains off in Auto mode. If Manual mode is selected the software checks the status of K8 to turn the fan on or off.

Precision reference generator

An LM336 IC in position IC11 generates the 'AREF' reference voltage for Platino, which does all of its calculations related to voltages based on that value. The LM336 is a precision reference generator in a space saving TO-92 package style and requiring a minimum of external components. Preset P1 adjusts AREF to exactly 5.00 V (use a good voltmeter). This is a onetime setting.

Positive voltage regulators

The power supply has two outputs, hence two separate positive voltage regulators are used. For the sake of simplicity identical regulators are used. The LM2576 (IC1, IC2) is an adjustable step-down switching regulator used here as a positive voltage regulator on the two sections of power supply. The LM2576 in position IC1 is configured to output 0 V to 15 V at 1 A (max.),

Specifications & Features

- Output 1: 3.3 V or 5 V / 500 mA max., on USB-A connector
- Output 2: 0–15 V / 1 A max., on banana sockets
- Both outputs short-circuit resistant
- Power supply output capacity: 17.5 watts
- DC input: 18 V / 2 A laptop PSU
- Readout: 20 x 4 LCD
- Single-knob control
- High temperature automatic shutdown
- Auto or Manual mode for fan & control
- Internal temperature sensor
- Setup Mode for fan threshold and temperature settings
- Normal Mode for V/A adjustment in daily use

all adjustable. IC2 in the other section provides the 3.3 V (500 mA max.) and 5 V (500 mA max.) selectable output voltages on K7. Both sections are quasi-digitally controlled by Platino.

The 0 V to 15 V adjustable output voltage is available on connector K6. The negative auxiliary voltage $-V_E$ is connected to the LM2576's GND pin (3) to make the output voltage start at 0 V instead of +1.23 V as the datasheet says. Both IC1 and IC2 take +18 VDC after diode D1 as their input voltage. Their output filter circuits follow the same layout with differences in current rating for the coils (L1, L2) and buffer capacitors (C11, C21).

Auto-shutdown control

Automatic shutdown is implemented for the protection of the power supply and the user. Protection is afforded against short circuits, over temperature and over current. Platino's PB6 pin is used to effectively switch IC1 on and off via MOSFET T2. Since T2 has the $-V_E$ voltage at its source terminal, zener diode D8 is necessary to avoid any negative voltage on Platino's PB6 pin.

Likewise PB7 is used to switch IC2 off in case of a short circuit, over temperature event, or over current detected at its output.

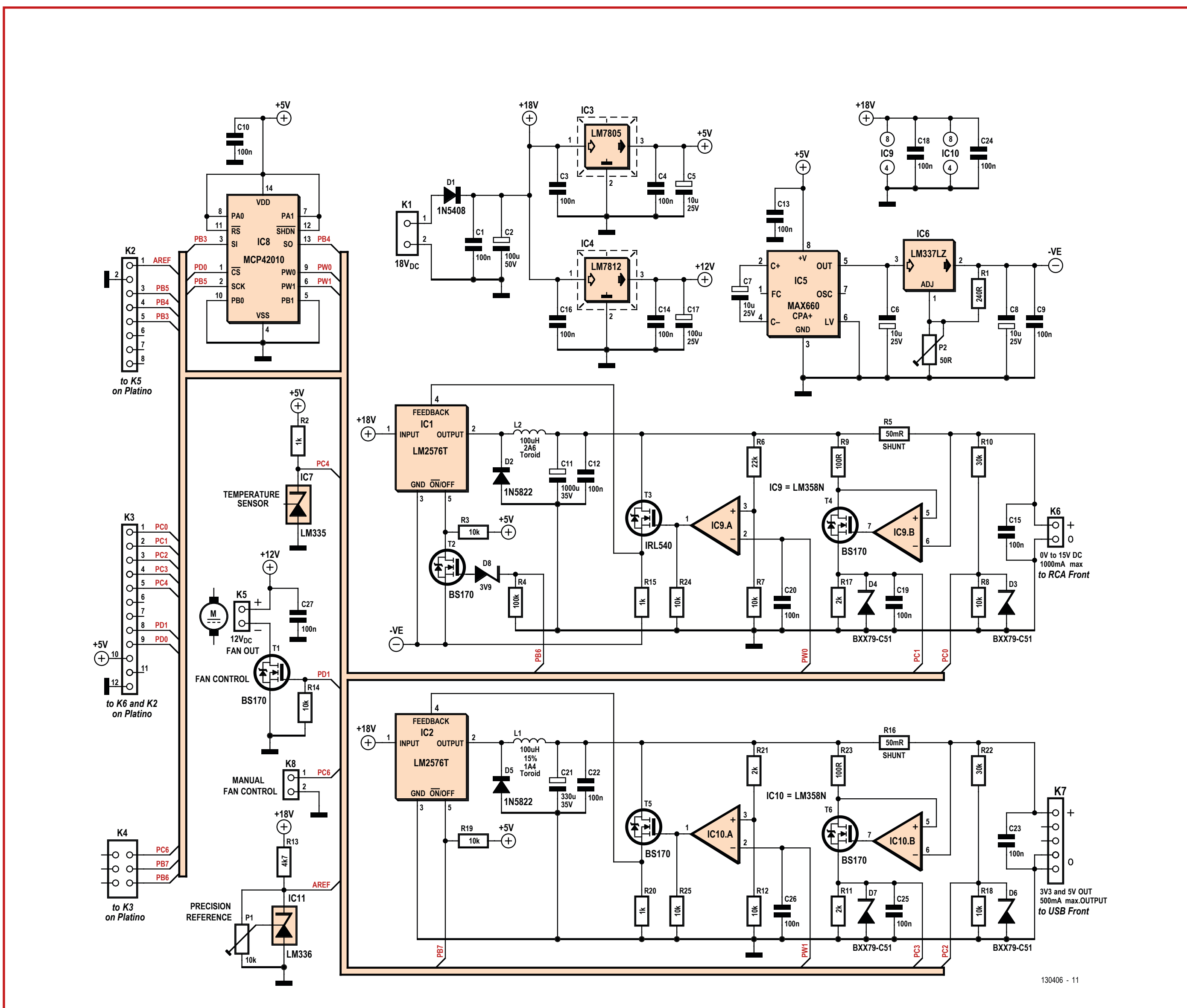


Figure 1.
Schematic of the Platino-controlled Benchtop Power Supply. Input power is furnished by a common 18 V / 2 A laptop supply.

Feedback & error correction

This section is responsible for controlling the output voltages and currents of both regulators. The main parts in this section are an opamp and a MOSFET. This section generates and controls the feedback of IC1 and IC2 using a control signal supplied by Platino.

For IC1, PW0 is the 'digital potentiometer' signal to control the feedback voltage. The current measurement signal gets fed to the MCU via pin PC1; the voltage measurement signal through PC0.

In the case of IC2, PW1 is the 'digital pot' signal; current measurement is by way of PC3, and voltage measurement by way of PC2.

The operation of the feedback & error correction section is as follows. Opamps IC9a and IC10a compare the voltage at their non-inverting terminals to that at the inverting terminals and then adjust their outputs according to the differences. The outputs determine the impedance seen by the regulator's feedback pins. Consider a voltage between 0 and 5V at the inverting terminal of the opamp; this will cause the opamp output to drop to a level well below the switching threshold of the MOSFET. Consequently the MOSFET's d-s path represents a high impedance. In order to meet the 1.23 V at feedback terminal, the regulator drives the output High but this output is again fed back to the non-inverting terminal. This will ramp up the output of the op amp resulting in a lower impedance. Consequently the overall circuits strive to maintain equal voltages at the opamp's inverting and non-inverting terminals.

The current sensing section consists mainly of opamps IC9b and IC10b. The current flowing through the load is converted to voltage by shunt resistors R5 for IC1 and R16 for IC2. The voltage dropped by R5 and R16 is fed to the inverting terminals of the opamps. The voltages cause the opamp outputs to ramp up, resulting in lower MOSFET impedance, resulting in turn in a voltage drop at the non-inverting terminal. The loop action ideally yields equilibrium between the voltages at the non-inverting and inverting terminals. A voltage drop identical to that across the series resistor is developed across 100-Ω resistor R9 for IC1 and R23 for IC2. Hence the current flowing through the 100-Ω resistor is proportional to the load current, which also flows through 2-kΩ resistor R17 for IC1, and R11 for IC2. The current-proportional voltage makes the microcontroller aware of the current demand on the PSU.

In order to protect the relevant microcontroller terminals 5.1-V zener diodes D3, D4, D6 and D7 are provided. The 0-15 V regulated output voltage is scaled to 0 to 5 V which van be handled safely by the microcontroller.

Digital potentiometer section

The MCP42010 (IC8) is a dual digital 50-kΩ potentiometer with SIP interference. Here, it supplies the feedback voltage under control of the Platino MCU. Digital Pot line PW0 serves IC1 through IC9a. Dig Pot line PW1 serves IC2 through IC10a. PB3, PB4, PB5 and PD0 are SIP interface signals from the MCU to control the digital potentiometer 'wiper' such that it supplies the voltage levels needed in the regulation loop.

Output section

The output of regulator IC1 is connected to the load via banana type connectors on the front panel (black and red) by way of PCB-mount connector K6. The output voltage of IC2 appears on a standard USB type connector also on the enclosure front panel, through PCB-mount connector K7.

Software

The software for the project was written in BASCOM AVR for ATMEGA328P microcontroller. The Platino board can be used for further development of the project. The software part is divided into a number of sections discussed below. The BASCOM program is available for downloading from [3].

Display section

The display section comprises the Platino board and a 20 x 4 LCD. This section of the program initializes the LCD and first outputs the message: "Platino Instrument series 1.0" followed by the name of the project: "Platino Adjustable Bench Power Supply". After the initialization and greetings, the display shows the Menu screen with its "Setup Mode" and "Normal Mode" options. Select any option by operating the rotary encoder (R/E) and its integrated pushbutton on the Platino board. At the selected option, press the pushbutton to view the options given by Normal or Setup Mode. A long press of the button returns you to the Main Menu.

Setup Mode

In this mode you configure the settings of voltage, current, fan and temperature threshold by

Component List

Resistors

Default tolerance, wattage: 5%, 0.25 W
R1 = 240Ω
R2, R15, R20 = 1kΩ
R3,R7,R12,R14,R19,R24,R25 = 10kΩ
R4 = 100kΩ
R5, R16 = 50mΩ shunt precision resistor, 1% 2W
R6 = 22kΩ
R8, R18 = 10kΩ 1%
R9, R23 = 100Ω 1%
R10, R22 = 30kΩ 1%
R11, R17, R21 = 2kΩ 1 %
R13 = 4.7kΩ
P1 = 10kΩ trimpot
P2 = 50Ω trimpot

Capacitors

C1,C3,C4,C9,C10,C12-C16,C18,C19,C20,C22-C27 = 100nF
C2 = 100µF 50V, radial
C5-C8 = 10µF 25V, radial
C11 = 1000µF 35V, radial
C17 = 100µF 25V, radial
C21 = 330µF 35V, radial

Inductors

L1 = 100µH, 1.4A (2062790)
L2 = 100µH, 2.6A (2215967)

Semiconductors

D1 = 1N5408
D2 = 1N5822
D3,D4,D6,D7 = 5.1V zener diode, 0.4W
D5 = 1N5819
D8 = 3.9V zener diode, 0.4W
T1,T2,T4-T6 = BS170
T3 = IRL540NPBF
IC1, IC2 = LM2576T-ADJ (Texas Instruments) (9488146)
IC3 = LM7805
IC4 = LM7812
IC5 = MAX660CPA+
IC6 = LM337LZ
IC7 = LM335
IC8 = MCP42010-E/P (Microchip) (1332110)
IC9, IC10 = LM358N
IC11 = LM336BZ-5.0

Miscellaneous

K1 = 2-way PCB terminal block
K5,K6,K8 = 2-pin pinheader
K2 = 8-pin pinheader
K3 = 12-pin pinheader
K4 = 6-pin (3x2) pinheader
K7 = 5-pin pinheader
Miniature fan, 12V, 40x40x15mm
2 pcs. banana socket, 1 red, 1 black, panel mounting (1176431) (1176430)
USB-A Socket, panel mounting (1667928)
Enclosure, 75x133x-110mm, Bopla ref. 26160000 (1217479)
Switch, on/off, rocker, panel mount, e.g. Schurter ref. 1301.9205 (1162728)
PCB, Elektor Store # 130406
Platino board, Elektor Store # 110645

(numbers in round brackets are Farnell/Newark order codes)

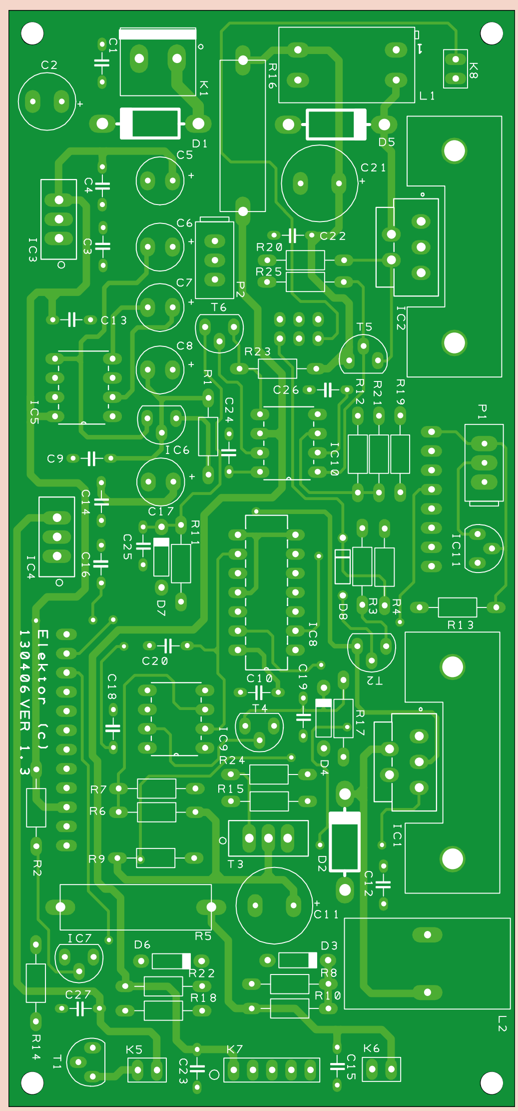


Figure 2.
PCB design for the supply.

selecting relevant options. A long press on the R/E button returns you to the Main Menu.

Normal Mode

In this mode the outputs are switched on for powering the load(s). You can use the rotary encoder to adjust the 0 V to 15 V output as required. A long press of the R/E button returns you to the Main Menu. All outputs are available in normal mode only.

Measurement Section

The microcontroller obtains information on the output voltage and current feedback from the Feedback & error correction hardware sections. It also measures the voltage and current relative to the reference signal on the AREF pin. Temperature measurement is carried out by the software, and the result appears on the LCD.
Measuring signals are scaled to the 0–5 V range by hardware and sent to MCU's ADC.

Control section

Depending upon the ADC input value the MCU generates the control signals for digital pot IC8 to control the output voltage and current. It also generates the auto shutdown signals for IC1 and IC2 depending upon current and temperature values and the limit value set in Setup Mode. It generates the Fan control signal according to the configuration in Setup Mode.
Using suitable software the microcontroller reads the output voltage and increments or decrements the digital pot in order to meet the desired voltage level. It also monitors the current and limits the output voltage if the load current exceeds the current limit. If a short circuit is detected (excessive current) the microcontroller shuts down the particular regulator and resets the digital pot to zero-out. Also the microcontroller shuts down the regulator, if it detects high temperature. The upper current limits and temperature limits can be programmed by the user.
A quick overview of microcontroller pins usage jointly by hardware and software is presented in **Table 1**.

Construction

First build the Platino with its LCD, rotary encoder with pushbutton, ATMEGA328P MCU and all other components, then 'jumper' it as shown in **Table 2**. Refer to [2] for the Platino showoff pages.

Table 1. Microcontroller Pins Usage

Pin designation	Function
PB0–PB2	Encoder with pushbutton
PB3, PB4, PB5, PD0	microcontroller <-> digital pot SPI comms. Data in = PB4 Data out = PB3 Chip select = PD0 Clock = PB5
PB7, PB6	regulators on/off control
PC0–PC3	IC1 and IC2 output voltage and current measurement
PC4	temperature measurement
PC5	LCD backlight control
PC6	fan controls

Figure 3. Looking inside the enclosure from the right. The three boards that make up the little supply (LCD, Platino, PSU) are fitted vertically behind the front panel.

The add-on board that forms the actual power supply is constructed next, see the PCB layout and Component List in **Figure 2**. The board is designed to fit exactly to the back of Platino with the help of pinheader connectors. A Bopla enclosure accommodates Platino with its 20 × 4 LCD, rotary encoder/pushbutton, the PSU board, a small 12 VDC fan, the power jack, on/off switch, banana connectors, and USB (power-only!) connector.

Figures 3 and 4 and various other photographs in this article illustrate the method of construct-

Table 2. Jumper settings on Platino

JP3	PC5
JP4	PB0
JP5	PB1
JP6	PB2
JP10	PB6
JP9	PB7
JP8	RESET
JP11	PB7
JP12	PB6

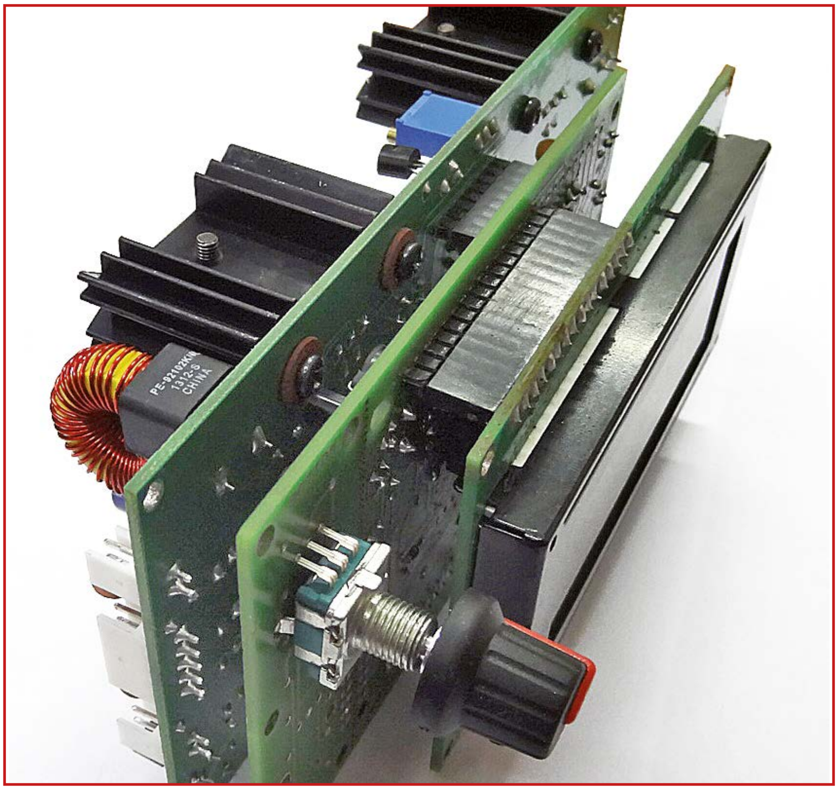


Figure 4. A close up of the three boards. The assembly is best tested in this state, i.e. prior to building it in the Bopla case.

ing the little benchtop supply. The boards sit vertically behind the enclosure front panel, in this order: LCD, Platino, PSU.

The front panel requires three rectangular cutouts: LCD, USB power-out, on/off switch. And three round holes: rotary encoder spindle, banana sockets (2x). Blocking elements like plastic cable feet should be glued to the bottom of the case to keep the 3-board assembly securely in place. In the back panel, make round holes for the fan (approx. 37 mm diam.) and the DC input connector (approx. 10 mm diam.). To assist ventilation, drill 12 2-mm holes in a 40-mm x 40 mm cross shape in the back panel, above the DC input socket.

Internal wiring is limited to the fan, the on/off switch, the USB power-out connector, the 18 VDC input and the red and black banana socket.

Testing

Test your power supply rigorously under full-capacity load conditions, observing all relevant safety precautions. Load the 0–15 V output with 1 amp by connecting a 15-ohm 30-watts power resistor (**Figure 5**). Load the 5 V output with 500 mA using a 10-ohm 10-watt power resistor. Also run a 1-kHz dynamic load test for 15 V (1A); 10 V (1 A) and 5 V (500 mA). Check the short-circuit recovery and auto-shutdown features by shorting outputs to ground.

Check over-current detection and response by setting the limit value, then dropping the output voltage a little and check if the set current is maintained.

(130406)

Figure 5. Maximum-output i.e. 15 V @ 1 A test using a 15-ohm 30-watts power resistor. Don’t leave it lying on the desk for too long while the test runs (and don’t pick it up either).

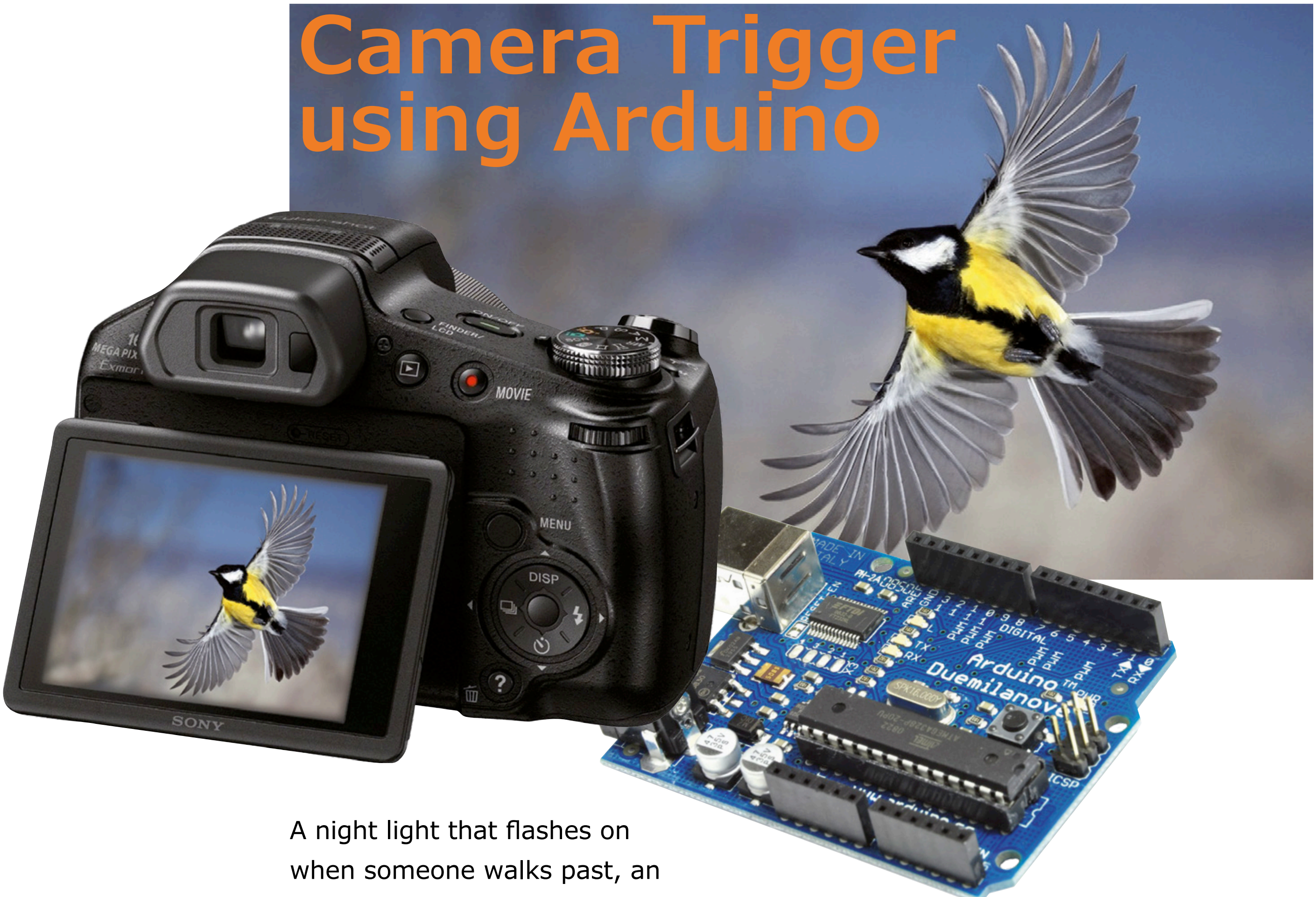
Web Links

[1] Lab PSU for Embedded Developers, Elektor April 2012 (UK edition), www.elektor-magazine.com/110645

[2] Versatile Board for AVR Microcontroller Circuits, Elektor October 2011, www.elektor-magazine.com/100892

[3] Project software: www.elektor-magazine.com/130406

Motion-Detector Camera Trigger using Arduino



A night light that flashes on when someone walks past, an IR-LED, a few resistors, a capacitor and an Arduino with a bit of software: that's all you need to take pictures using motion detection.

By **Rolf Blijleven**
(Netherlands)

I consider the Arduino to be very similar to Lego: you can make all manner of things with it. This is useful and instructive to do, and it results in something useful. You can then use it for a while, and whenever you feel like it, you can take it apart again and use it to make something else.
A while ago, I made an infrared remote control for my Nikon D80 using an Arduino. Not because such a remote control is that expensive, but because making one yourself is much more fun and using an Arduino it offers many more possibilities. Moreover, it turned out to be child's play: An IR-LED, a resistor and a piece of software that I found on the internet, nothing else is required. A considerable advantage of the Arduino, compared to other embedded platforms, is that there is a tremendous wealth of firmware to be found, just free from the Internet.
After this, I wanted to trigger the remote control using a motion detector. There are many nice solutions to be found for this on the DIY market and on the internet, but these usually require an AC power adapter. I didn't want that, it had to work without any wires. By chance, I spotted at the supermarket a night light powered from batteries, with a motion sensor, for a couple of dollars (**Figure 1**). That won't break the bank. It just went with the weekly groceries.

Night light hacking

The first thing you do, naturally, is look what is inside it. This tuned out to be much better than expected. No difficult SMD parts or—worse—COBs, but instead an IC with legs, normal resistors and capacitors, a PIR-sensor and a photo diode.
The part number printed on the IC was TL0001. A quick Internet search and sure enough there appears to exist a datasheet [1]. In Chinese, but that is not a problem: just copy the text and paste into Google Translate. This results in an horrendous translation but is still good enough so that you can get the gist of it. There was even an *Application Note* in there, that although it was not exactly the same is my night light, it was nevertheless very close.
The night light did three things that I didn't want: it only operated in the dark; it gave a pulse that was several minutes long, while I needed a much shorter pulse and a turned on three bright LEDs when you walked past.
The latter was easily solved. The three LEDs shared one series resistor. I removed two and replaced the series resistor with one that has a higher value (2.2 kΩ, A in **Figure 2**), so that the remaining LED would still switch on with each trigger, but not as bright.
Then the *inhibit*-function during daylight. In the example schematic (**Figure 3**), R3 is an LDR. This I couldn't find anywhere, but I did find a photo diode, also a device that exhibits a lower resistance as the amount of light on it increases. So I replaced that with a reasonably large resistor (220 kΩ, B in **Figure 2**). That worked too: Even in bright light the lamp turned on when motion was detected.

Figure 1.
Available from your Walmart or discount store: a super cheap night light with motion detector.

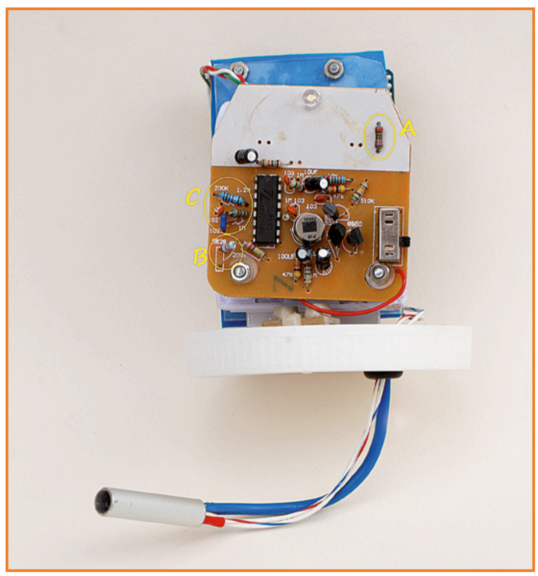


Figure 2.
The same night light after a few modifications.

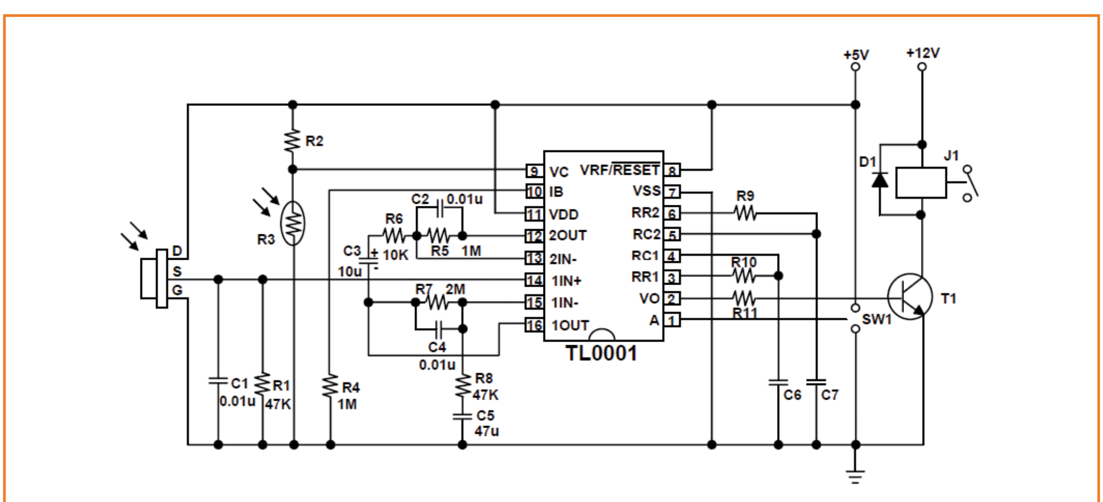


Figure 3.
A sample schematic from the datasheet for the TL0001 made by the Chinese company Treasure Link Technology. Although it looks very similar, it does not correspond completely with the actual circuit in the night light.

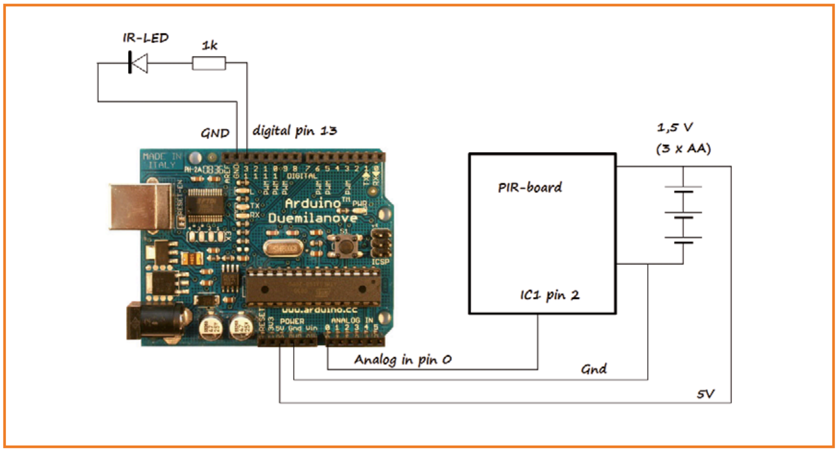


Figure 4.
The complete circuit with Arduino, IR LED and PIR-board (formerly a night light).

Now the pulse duration. The light turns on for about 5 seconds. That will be an RC-time constant somewhere, but which R and which C? I had already observed that it was not possible to turn the light back on immediately after it had turned off. So there had to be another RC-time constant to suppress new triggers for a while (*trigger-inhibit*). Now I really had to resort to reading the datasheet. And sure, it was all there. Machine translated from the Chinese: "Output delay time Tx R9 and C7 by an external sizing, value Tx ≈ 24576xR9C7; triggered by an external blocking time Ti R10 and C6 resizing is Ti ≈ 24xR10C6." That's clear, isn't it?

Time for some *reverse engineering*: not designing a printed circuit board from a schematic, but the other way around: drawing a schematic based on the circuit board. This is a little easier when you can see both sides of the circuit board side by side. A photocopy of the copper side is very helpful.
After a little drawing, calculating and soldering I had identified R9/C7 and R10/C6 and replaced them with 'improved' values. That's what I thought. However, it wasn't right. At any rate, the light had completely lost the plot. While I was checking everything again, my eye caught some text in the datasheet: "BISS0001 chip it is fully compatible with". And sure enough, the datasheet for the BISS0001 had the correct formulas: Tx ≈ 24576xR10C6 and Ti ≈ 24xR9C7. In the Chinese datasheet R10 and C6 where swapped with R9 and C7! With R10C6 = 1 kΩ x 100 nF and R9C7 = 270 kΩ x 1 nF resulted in Tx ≈ 24 ms and Ti ≈ 0.5 s (C in **Figure 2**). Right on the money. C7 could have remained unchanged.
An attempt to increase the gain of the PIR-sensor somewhat resulted in a modest improvement. The gain stage comprises two steps. 1IN+, 1IN- and 1OUT in the IC are an opamp (see datasheet), the gain of which is about equal to R7/R8. Using 1 MΩ/12 kΩ this became 84 (was 40 with the original 2 MΩ/47 kΩ). The makes the PIR-sensor not more sensitive, of course, but it does amplify small signal more. A consequence was that larger signals would hit the power supply rails. The second stage is also an opamp with a gain equal to R6/R5, originally 100, but with 15 kΩ for R6 this became 67. This solved the clipping problem.
The result is that the sensor is able to detect movement indoors over a somewhat larger distance than before, but that doesn't mean much by itself. PIR-sensors are especially sensitive at seeing heat *differentials*. A cat walking past on a frosty day is detected from many meters further away compared to the same cat on a warm summer's day.
The output of the IC is on pin 2, which is soldered to a wide copper track. It is therefore very easy to attach a wire to it. With another two wires for +5 V (after the switch) and ground the night light had become a PIR-board. Not bad for an investment of € 2.65 and some time figuring it all out.
I fully expect that in your part of the world the type of motion detector or night light that you can buy will be entirely different. But the above story has at least shown a method you can use to figure out how it operates and adapt some of the features of the circuit to your needs.

Work horse Arduino

It turned out that connecting the PIR-board to an analog input of the Arduino was more convenient than using a digital input. One digital output of the Arduino is used for the IR-LED with a series resistor, which is used to operate the camera. The three AA-batteries for the night light also serve as the power supply for the Arduino. The circuit is otherwise simplicity itself (see **Figure 4**).
The timing for the IR-pattern for the camera trigger is borrowed from [3] and [4]. Movement a few meters from the PIR-sensor generates a trigger for the camera, which, in turn, could also be a few meters away from the IR LED. This also works through glass, so you can keep your camera inside and the sensor/remote control outside.
I have mounted the IR-LED on a piece of thick electrical wire, so that the LED can be bent into a different direction compared to the direction of the PIR sensor.

My camera, a Nikon D80, turned out to have some unexpected characteristics. When you switch this camera into IR Remote mode it waits a while for an IR-command. If that does not happen it will switch out of IR-mode by itself. Any command after that interval is ignored. For my application (nature photography) this was undesirable. The waiting time can be adjusted in the camera up to a maximum of 15 minutes.
That is why the firmware will give the IR-command 'stay awake' if there has been no motion detected for 14 minutes. In this way you can leave the camera for days, waiting for that one rare animal to go by.
This interval can also be made shorter. Without the PIR-board you can then also use it to make time-lapse movies of, for example, flowers that grow and open.

To allow this interval to be changed in the code in a more obvious way, requires a little bit of computation. We are using Timer1, this is a 16-bit timer, so counts from 0 to 65536. If we let the timer start from a `timerPreload` of 3036 it will count 65536 - 3036 = 62500 clocks and give an interrupt. The Duemillenove runs at 16 MHz; with a prescaler at 1024 this becomes 15625 Hz, so we will have a Timer1-interrupt exactly every 62500/15625 =4 seconds (ignoring any inaccuracies of the clock). In the code this is implemented as follows:

```
#define four_sec 1
#define twelve_sec 3 * four_sec
#define minute 5 * twelve_sec
#define quarter 14 * minute
```

This is a quarter of an hour that lasts only 14 minutes, because at 15 my camera would just leave IR-mode. With `timeCounter` we keep track of the time. In the interrupt service routine we give the intial value `timerPreload` (=3036) and incerment `timeCounter`. The counter value times four is the time elapsed in seconds.

```
ISR(TIMER1_OVF_vect) {
  TCNT1 = timerPreload;
  timeCounter +=1;
}
```

In the main loop we take a picture when there is a trigger from the PIR-sensor or when a quarter of an hour has gone by.

```
if (val > 200 || timeCounter == quarter
) {
      timeCounter = 0;
      takePicture();
      delay(500);
}
```

The source code for the firmware for this project is a free download from the Elektor website [5]. The binary file is only 4 KB, so with an Arduino with 32 KB flash memory there is plenty of spare space for to add your won features. A trigger based on sound would be another possibility.

(130265)

Internet Links

[1] www.treaslink.com/Upload-Files/2010531152721141.pdf

[2] www.e-ele.net/DataSheet/BISS0001.pdf

[3] www.bigmike.it/ircontrol

[4] http://luckylarry.co.uk/arduino-projects/arduino-ir-remote

[5] www.elektor.com/130265

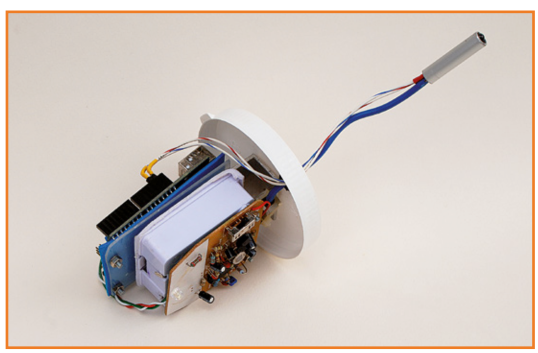
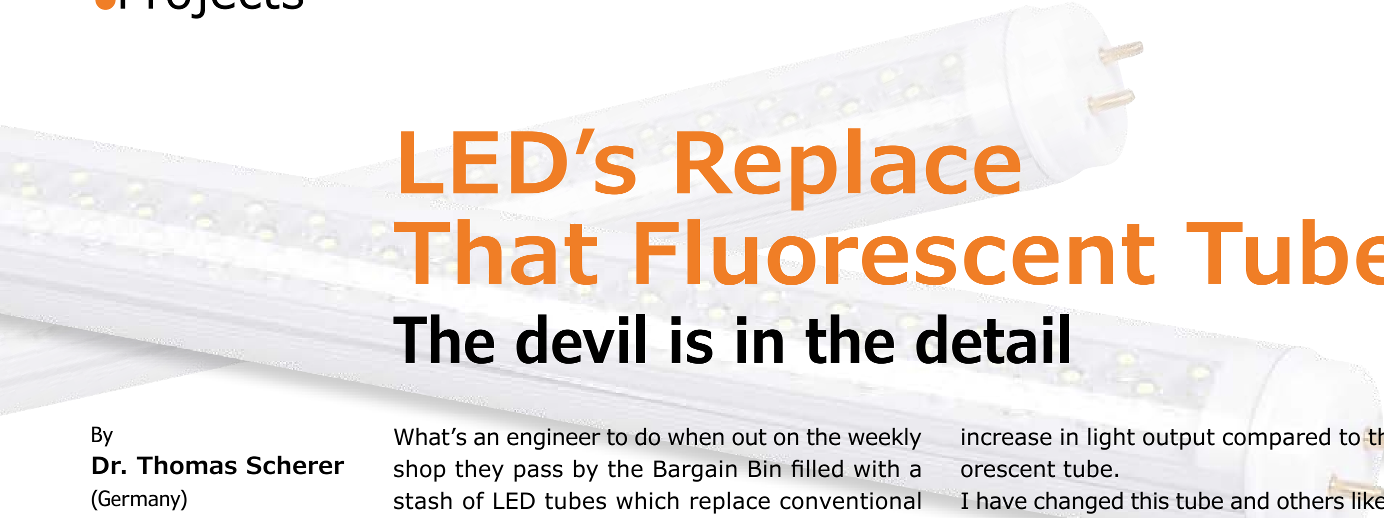


Figure 5.
Arduino and PIR-board with battery housing, mounted back-to-back on a piece of L bracket.

Figure 6.
The entire circuit housed in a splash-proof enclosure.



LED's Replace That Fluorescent Tube

The devil is in the detail

By
Dr. Thomas Scherer
(Germany)

What's an engineer to do when out on the weekly shop they pass by the Bargain Bin filled with a stash of LED tubes which replace conventional fluorescent tubes? Take a closer look of course. Picking up the long thin box I scrutinize the labeling to check the tech specs. There never seems to be enough information—I finally submit to my hunter-gather instincts and place one in the shopping cart alongside my other finds. They really are a good price. A few minutes later I'm back at the Bargain Bin putting another one in the cart. Who was it that said a bargain is something you don't need at a price you can't resist?

Attempt #1

Back at home I plan to fit one LED tube to our bathroom cabinet which I know uses an 18-watt fluorescent tube. According to the label the 10-watt LED replacement tube uses less energy and will produce more light. When all was said and done and the tube was finally up and running I measured (at the same distance) a 30 % increase in light output compared to the old fluorescent tube.

I have changed this tube and others like it before. It shouldn't be difficult—take off the diffuser, rotate the tube through 90°, pull it out, insert the LED replacement tube rotate it back through 90°, refit diffuser, job done… Not so fast!

It can be that easy, but not necessarily—it depends on the type of light fitting you have. Older fittings use a coil or ballast together with a pre-heat starter device, if yours is like this then you should have no problems. **Figure 1** shows the wiring in this type of fitting. The tube has a heating filament at each end—when AC power is applied the voltage across the starter unit produces a glow discharge which heats up a normally open set of bimetal contacts. The contacts close, supplying current to the heating filaments at each end of the tube and through the inductive ballast. With the starter contacts closed, glow discharge in the starter unit stops and the contacts cool down. After a second or so they open, producing a voltage spike between the filaments, initiating the main discharge in the fluorescent tube. With this type of fitting it's normal to hear it click and see the light flash two or three times when you switch on before the tube strikes.

Figure 2 shows the internal wiring of the LED replacement tube; on the left is the LED driver module instead of a heating filament. On the right is just a short circuit instead of the second heating filament. Just fitting this in place of a standard fluorescent tube would most likely cause the lamp to flash until the induced voltage spikes eventually kill the LED driver primary side. It would probably not last too long and any case it wouldn't make a good restful light to shave by. To get round this, the LED tube comes with its own starter unit (**Figure 3**) consisting of nothing more than a wire link. Using this instead of the standard starter unit solves the problem: The AC input now connects to the LED driver input irre-

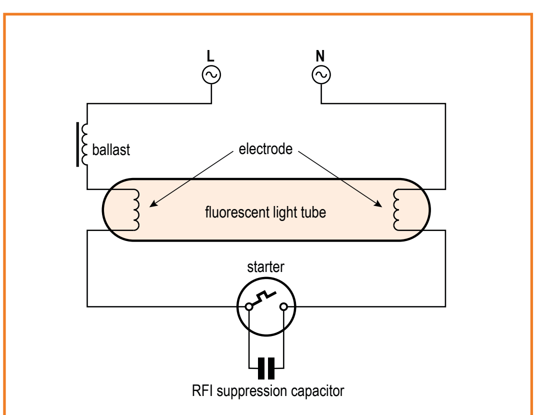


Figure 1.
Internal wiring of a conventional fluorescent fitting showing the ballast, starter and tube showing the heating filaments.

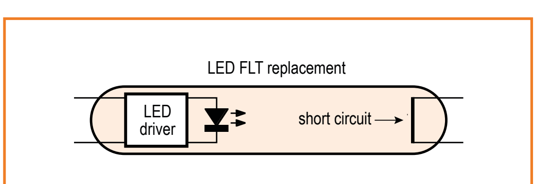


Figure 2.
Internal wiring of the LED tube replacement. The two connections on the right are shorted.

spective of which way round the tube is fitted. So problem solved.

Just one tiny complication: my modern bathroom cabinet fitting doesn't have a starter… My first attempt at replacing the lamp ends in failure.

All good things come in twos

When I originally bought the lamp fitting I chose the latest technology with an **electronic** ballast. These light up in a few milliseconds without all that clicking and flashing associated with a ballast and starter. What was an advantage has now turned out to be something of a problem; there isn't a starter device to replace. I wasn't brave or foolhardy enough to just fit the new tube and fire it up. As a last resort I turned to the instruction sheet that came with it. There I read a warning that the tube should not be used in fittings with an **electronic** ballast. Indeed it states that any electronic ballast must be taken out of the circuit before the LED lamp can be used.

Armed with this knowledge and a screwdriver I set to work (**Figure 4**). The 'Electronic Ballast' module has two wires connecting to the AC input and two lengths of twin cable connecting to the filaments at either end of the fluorescent tube. **Figure 5** shows it in circuit with the tube. You can see why it would not have been a good idea to just plug in the LED tube. The LED lamp in **Figure 2** would short one side of the ballast and probably generate smoke signals.

So it now looks obvious (to me) that I should take out the electronic ballast and wire the AC input to the input side of the LED tube which connects to the internal LED driver... wait a minute, what if the tube gets taken out and put back the other way round? Not only would there be no light, the built-in short circuit at the other end of the LED tube would now be across the AC input, tripping the lighting circuit breaker for sure. Not good. With the ballast now removed, wire the fitting as shown in **Figure 6**. The LED tube can now be inserted either way round without any problem. The circuit wiring looks like a conventional fitting with a short circuited pseudo-starter and no ballast. Success at the second attempt! With the help of a screwdriver and a couple of terminal blocks the rewiring was complete, the lamp shone forth and the engineer was happy (and clean shaven).

A cautionary note

It is possible that sometime in the future someone may want to replace the tube. The modified fitting is now no longer suitable for a fluorescent tube. It's important to make this clear otherwise it will be a safety hazard. Without a ballast or starter in circuit the filaments will glow continuously bright orange (probably for not very long). Before you fit the LED tube, use a permanent marker and write a cautionary note on the fitting: '**Only Suitable for LED Tubes!**'. Choose somewhere that will be hidden by the tube but can be seen when the tube is taken out.

(130403)

Figure 3.
The starter supplied with the LED lamp is just a wire link.

Figure 4.
The electronic ballast unit in the fitting.

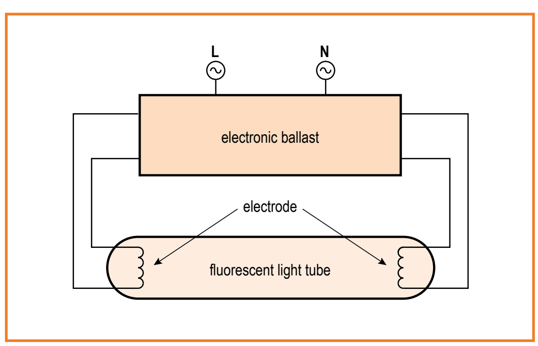


Figure 5.
The electronic ballast (Figure 4) wiring to the fluorescent tube.

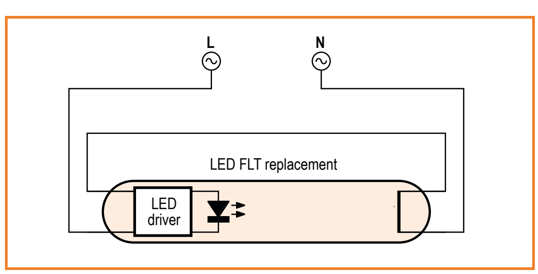
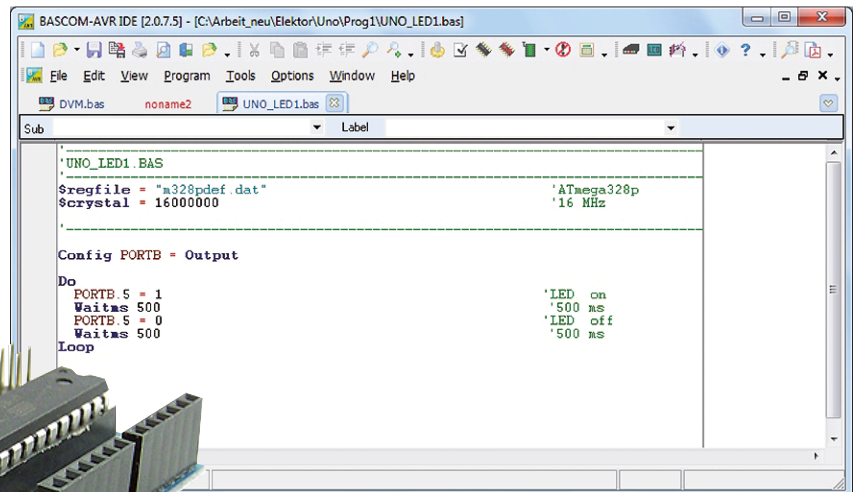


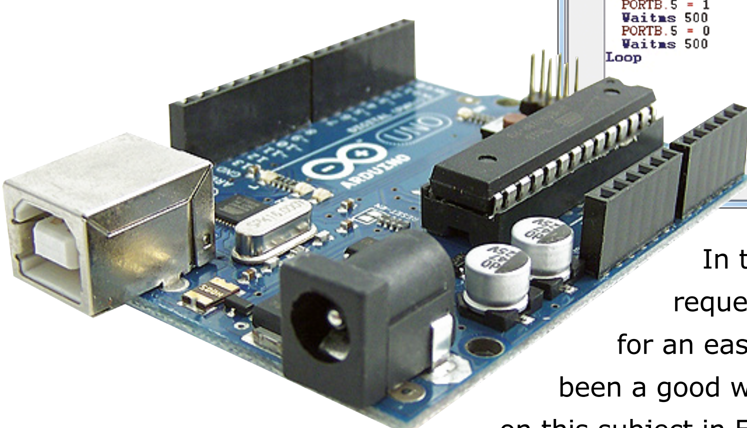
Figure 6.
Final wiring for the LED tube.

Microcontroller BootCamp (1)

Arduino and Bascom

By **Burkhard Kainka**
(Germany)





In the Editorial Office we receive a fair number of requests from electronics enthusiasts who are looking for an easy way to get started with microcontrollers. It's been a good while since we last ran a large series of articles on this subject in Elektor, and the associated hardware is vintage, so it's high time to run a new series. This series is aimed at our readers who already have some experience with analog electronics and now want to start using microcontrollers in their own circuits.

You might ask why you should use a microcontroller when it's possible to do so much with ordinary analog electronics. At first glance, this looks like a good question. New microcontrollers with more features, higher performance, higher clock rates and even more memory are appearing all the time, but the first demo program for every one of them invariably makes a LED blink. This leads to the justified criticism that you could get the same result by simply taking an NE555 timer IC and adding a couple of resistors and a capacitor. That's absolutely right, and the comparison is better than you might think because a lot of the elements of an NE555 timer IC can also be found in a microcontroller.

For comparison: the NE555 timer IC

First of all there's the output. When you look at the internal block diagram of the NE555 on the data sheet (**Figure 1**), you can see that it has a push-pull output stage that can actively switch high and low. Microcontrollers have exactly the same kind of outputs. They are called ports, where the name "port" can stand for a set of outputs or for pins that can be configured either as inputs or as outputs. The circuitry connected to the output for a LED blinker application is also the same: a LED with a series resistor, connected either to ground (GND) or to the supply voltage (Vcc). The NE555 also has a second output driven by an open-collector transistor, which can only switch something to ground. Many microcontrollers can also emulate this function. Actually the only difference is that microcontrollers usually have several outputs but the NE555 has only one, since the push-pull output and the open-collector output are not independent.

Next we have the inputs of the NE555, which consist two inputs to a comparator that controls an internal flip-flop. A typical application for this is an astable multivibrator, which hobbyists often call a blinker circuit. You can put all this together in almost no time. You just plug the numbers into a couple of formulas to determine the right component values, and then the NE555 does exactly what you want. Determining the component values for an NE555 circuit and programming a microcontroller are actually comparable tasks. There's another thing that's very similar: both devices (NE555 and microcontroller) have a Reset input that you can use to set everything back to the starting point. And with both devices the Reset input is usually high in the quiescent state and must be actively pulled to ground.

Figure 2 shows a simple square-wave generator in the form typically used as an LED blinker. The NE555 data sheet also shows several other basic circuits, including a monostable and a pulse-width modulator—all of which are typical tasks for microcontrollers. If you further consider the countless NE555 applications you can find somewhere on the Internet, you certainly have to agree that in many cases all you need is a 555 timer IC and a microcontroller is overkill. For everything from light curtains to servo controls or processing analog sensor signals, there are many things that can be done with this IC and similar devices. And you can be sure that there are still lots of potential applications that haven't been worked out yet.

Reducing development time

The similarities between the NE555 and a microcontroller are summarized in **Table 1**. You could formulate the result of a fair comparison as follows: The microcontroller has a bit more of everything and is therefore generally the better choice for complex tasks. For example, a single microcontroller could probably handle a task that would take ten NE555s. Above a certain level of complexity, the microcontroller solution is also smaller and cheaper than the analog solution. On the other hand, an analog electronics solution is a better and more economical choice for quite a few simple tasks.

Anyone with a bit of experience in microcontroller development can also mention another advantage of microcontrollers: once the circuit is complete, you don't need to touch the soldering iron again. From that point on, all you do is write and test code. Changes to device functions can be implemented and tested very quickly. Microcontrollers are general-purpose and versatile computation workhorses. The learning curve is worth the effort, since you ultimately save time. Your first exposure to a microcontroller data sheet may put you off, since it can easily amount to 300 pages or more. Fortunately, there are

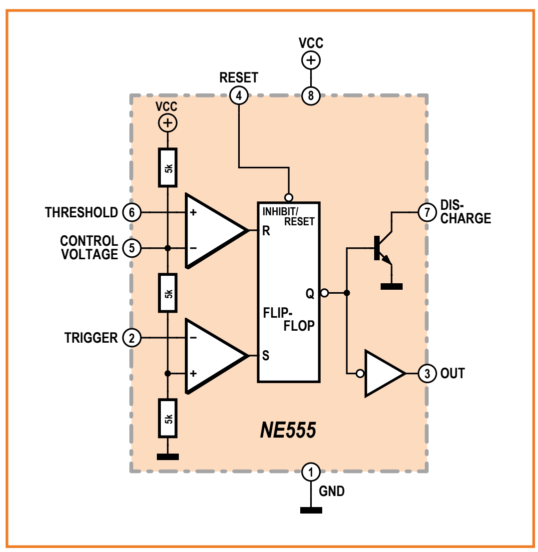


Figure 1.
Block diagram of the NE555.

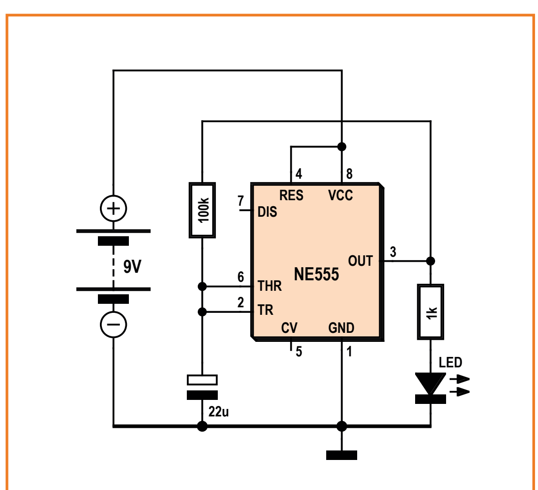


Figure 2.
Blinker circuit with the NE555

Table 1. Comparison of NE555 and microcontroller.	
NE555 chip	**ATmega microcontroller**
Switching output	Port outputs
Inputs	Port inputs
Timing control by RC networks	Internal timer driven by crystal-controlled clock
Reset input	Reset input
Comparator inputs	Comparator; analog inputs
PWM function	PWM outputs
Analog signal processing	Analog to digital converter
Flip-flop	Memory cells

approaches to the world of microcontrollers that do not require you to understand everything right up front. In many programming languages the first steps are very easy. You just need to have enough confidence to try something even when you don't entirely understand how it works. The main thing is to have a couple of positive experiences at the beginning, and after that it just goes automatically. That's doubtless the reason why the first demo task for every microcontroller is a LED blinker. Here as well we remain true to this tradition, if only to maintain the comparison with the NE555.

Arduino and Bascom

Microcontrollers actually come in all sorts and sizes. At the start of a series such as this we are faced with the choice of which system to use, and there are many different options. We spent a long time talking about which microcontroller, which circuit board (existing or new), and which programming language we could specifically recommend for beginners. Our discussions led to the following proposal: hardware: Arduino Uno; software: Bascom.

Arduino has become the most popular system in the hobby environment. The programs are developed on a PC using a simple programming language, and they can be downloaded directly to the microcontroller over a USB connection. A large variety of boards and extension boards (Arduino shields) are also available at amazingly low prices. The entire system is based on the open-source concept, so the software and the hardware are fully documented. The low-cost Arduino Uno board [1] is a good choice for our course because it is equipped with a widely used microcontroller. The ATmega328 is an AVR microcontroller made by Atmel. Programming this device is quick and quite easy. The microcontroller has enough memory to allow relatively large programs to be executed later on (see **Table 2**). There is a dedicated development environment available for the Arduino. A development environment in this context, also called an integrated development environment (IDE), is a PC program that is used to develop programs for a microcontroller. In the development environment, the microcontroller programs are typed in using an editor and then converted into bytes by the compiler, after which they are downloaded to the microcontroller.

To ensure that the compiler understands what the microcontroller is supposed to do, you must adhere to the syntax rules of a particular programming language when you write the pro-

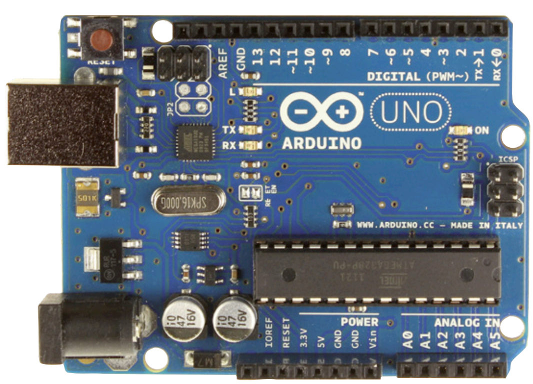


Figure 3.
The Arduino Uno board.

Table 2. Arduino Uno at a glance.	
Microcontroller	ATmega328
Supply voltage	5 V
USB port	5 V supply and device programming
External power supply	7–12 V
Digital I/O pins	14 (of which 6 can be used for PWM output)
PWM channels	6
Analog inputs	6
Current per I/O pin	40 mA (max.)
3.3-V output	50 mA (max.)
Flash memory	32 KB (with 0.5 KB occupied by the boot loader)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock speed	16 MHz
Price (for Elektor Members)	$39.70 / £24.75 (check [1])

gram. The Arduino IDE uses a simple version of the C programming language, which is the language used by most professional programmers. It also has several special commands that make program development easier. Many users have become familiar with this programming language and have no trouble working with it. Nevertheless, for this course we chose the Basic programming language. There are many reasons for this choice. One is the Bascom Basic compiler for AVR microcontrollers, which is widely used and popular. The learning curve is especially easy, in part because Bascom includes many special commands for sending characters from a microcontroller to a PC, for showing letters and numbers on a display, and much more. However, perhaps the most important reason for using Bascom is that it makes you fit for all AVR controllers. The selected Arduino board (**Figure 3**) is used here as a learning platform, but in the end you can use any desired AVR microcontroller on other commercially available boards or on your own boards. You also don't have to limit yourself to ATmega devices. For example, you may be able to manage with the compact ATtiny13, which has only eight pins.

A special feature of the Arduino board is the USB port. In addition to downloading programs to the microcontroller as described below, you can use it to send characters back and forth between the PC and the ATmega328. Among other things, you can use this to control the microcontroller remotely from a PC program or to send measurement values captured from sensors by the microcontroller to the PC and display them on the PC. Another thing is that the board can be powered over USB if you so wish. All you need is a USB cable, and you're ready to go.

Your first program

One option for putting together a blinker circuit with an ATmega328 is shown in **Figure 4**. To make it easier to see what matters here, the circuit diagram shows the microcontroller without all the peripheral circuitry of the Arduino board. The LED is already present on the Arduino board, so you don't have to put anything together.
Similar simplified circuit diagrams are also used for the other example applications described later on. The idea behind this is that you can also try out all of these examples using a bare microcontroller, for example on a breadboard or on a piece of prototyping board. This means that if you wish, you can follow the entire course without the Arduino board. However, it's more convenient and easier for beginners to use the Arduino Uno board.
Listing 1 shows the Bascom program for the LED blinker. It starts off with two directives for the compiler, which define what is called the envi-

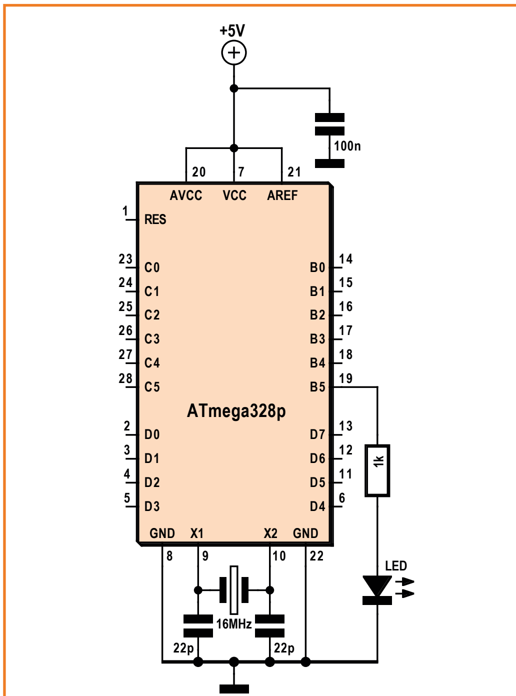


Figure 4.
linker circuit with a microcontroller.

Listing 1. LED blinker

```
'___________________________________
'Uno_LED1.BAS

'___________________________________
$regfile = "m328pdef.dat" 'ATmega328p
$crystal = 16000000       '16 MHz
'___________________________________

Config Portb = Output

Do
  Portb.5 = 1      'LED on
  Waitms 500       '500 ms
  Portb.5 = 0      'LED off
  Waitms 500       '500 ms
Loop
```

ronment. This is necessary because the Bascom compiler needs to know the target microcontroller for compiling the program. The Arduino Uno is equipped with an ATmega328P, so the directive is:

```
$regfile = "m328pdef.dat"
```

You also have to tell the compiler what clock speed the microcontroller runs at. The higher the clock speed, the faster the microcontroller executes the instructions. This can be important when dealing with things that require exact timing. An example is sending characters to the PC at a particular data rate. The board has a 16-MHz crystal that sets the clock speed of the controller.

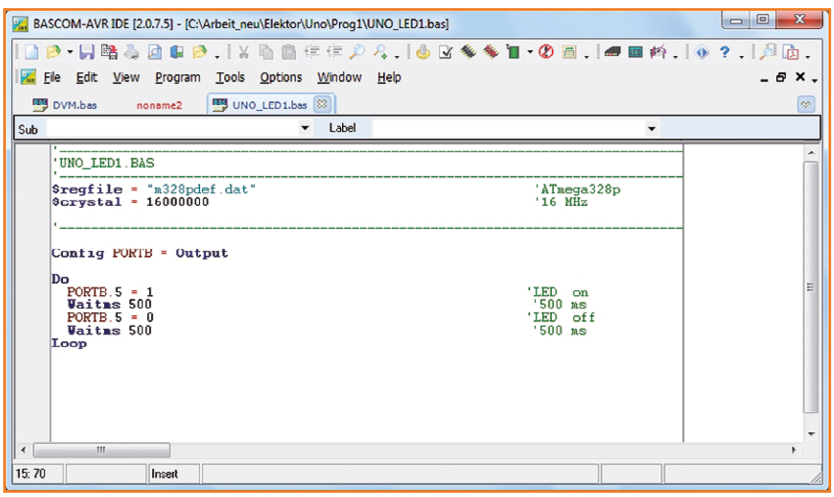


Figure 5.
The first program in the Bascom editor.

The corresponding directive for the compiler is:

```
$crystal = 16000000
```

Before you get to the actual program, there's another important configuration issue: the port pins of the microcontroller can be used as inputs or as outputs. All port pins are initially configured as inputs after a restart. Here we need an output, so the corresponding port must be configured accordingly:

```
Config Portb = Output
```

This instruction configures all six pins belonging to port B as outputs. The program actually only uses port pin PB5; all of the other pins remain unused. Why did we choose PB5 in particular? Because a yellow LED is already connected to this pin on the Arduino Uno board.

```
Portb.5 = 1
Waitms 500
Portb.5 = 0
Waitms 500
```

The individual line to the LED is switched to the "high" voltage level (close to the supply voltage of the microcontroller) by the instruction *Portb.5 = 1*. This causes a current to flow through the LED. The port pin is switched back to the "low" voltage level (close to the ground level) by the instruction *Portb.5 = 0*. There is also a wait instruction to delay program execution. *Waitms 500* is self-explanatory; it causes a time delay of 500 ms. All of this is built into a loop structure. Everything between "Do" and "Loop" will be repeated indefinitely. The only way to stop the program is to switch off the supply voltage or press the Reset button.

Software: the compiler

So far, so good. But how is the program converted into an executable form, and how do you get it into the microcontroller? The Bascom Basic compiler was developed by Mark Alberts for the 8051 family and for AVR microcontrollers. First you have to get a copy of this PC software. It is available from the website of MSC Electronics [2]. You can opt for the paid full version or the free demo version. The demo version is fully adequate for trying out the software and for getting starting with programming. It is limited to programs that occupy up to 4 KB (4,096 bytes) after compilation. By comparison, the Arduino has room for up to 32 KB. However, 4 KB is not peanuts; it takes a fair amount of programming effort to fill it up. Most of the examples in this series will be much smaller.

Installing Bascom is very easy. After you launch the program, the first thing you see is an empty Editor window where you can type in your own program. Of course, you can also import an existing program into the Editor (see **Figure 5**). These program files (with the suffix .bas) are plain text files that can also be viewed with Windows Notepad or another editor program. This is called "source code" because the compiler uses the contents of these text files as the source for compilation. The compiled program is called "hex code" because the bytes are often shown in hexadecimal notation, always with two hex characters in the range 0–9 and A–F per byte. Numbers in the form of bits and bytes and the various notations used for them will be a frequent topic in this series on microcontroller programming.

You can type in the program in Listing 1 yourself or download it from the Elektor page. The file *UNO_LED1.bas* is located in the zip folder [3]. Compiling the program is very easy: simply click Program/Compile, click the corresponding icon on the toolbar (a black IC), or press the F7 key. If there is any sort of error in the source text, an error message will be displayed. If there are no problems, a pop-up window shows what percentage of the memory is occupied. In this case it is less than 1%, which is rounded down to 0%. What matters is the resulting hex file *UNO_LED1.hex* or the binary file *UNO_LED1.bin*, which are two different file formats with the same content. They contain the executable code for the microcontroller. Now you have to load this code into the microcontroller's flash memory. There are many ways to do this, and for now we only describe the simplest way. Other options will be described in subsequent instalments.

The simplest way: use the boot loader

This program must be located in the microcontroller's flash ROM in order to run on the microcontroller. Flash ROM is a special type of memory, similar to EEPROM, in which electrical charges ensure that the memory contents are retained reliably for many decades. Flash ROM can be rewritten repeatedly ("flash" refers to very fast writing), so programs stored in flash memory can always be altered at a later date. Special programming devices are available for programming flash memory. They are connected to specific pins of the microcontroller, and the program bytes are received from the development environment on the PC via the USB link.
However, this can also be done without a programming device. When the Arduino board has a USB connection to the PC as mentioned above, the program bytes can be sent to the microcontroller over the USB link as long as there is a small program running in the microcontroller that receives the data and writes it to the microcontroller's flash memory.
This small program is called a boot loader, which is related to the expression "booting up" for starting up a computer. This term originates from the word "bootstrap" in the saying "pull yourself up by your bootstraps". Of course, pulling yourself out of the mud by tugging on your bootstraps doesn't work in practice, and likewise a microcontroller with nothing in its program memory cannot program itself. However, this is possible if a boot loader is already present in memory, and the Arduino comes with a built-in boot loader.
The development environment on the PC also has to be able to download program code using the Arduino boot loader. Fortunately, Bascom developer Mark Alberts already guessed that someone would want to program an Arduino board in Bascom at some point in time, so this capability is already incorporated. After you configure the right settings, everything is quite easy. Here we describe how this works with the demo version, since there is a small difference with the full version.

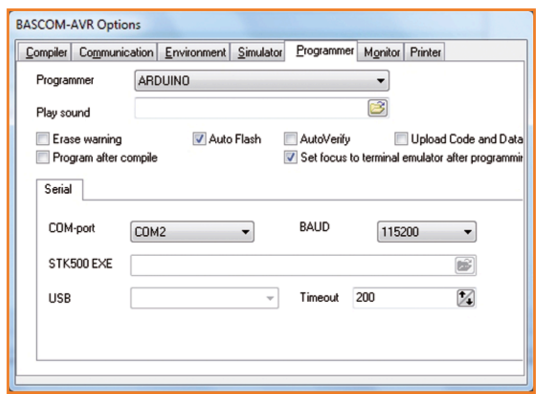


Figure 6.
The settings for the Arduino boot loader.

First you have to install the original Arduino software available from [4], which includes the USB driver for the Arduino Uno board. Then you connect the Arduino Uno over a USB cable. The driver is loaded automatically, after which the Arduino Uno should be visible in the Windows Device Manager window. There you can see which COM port number (e.g. COM2 or COM3) has been assigned to the Arduino Uno. If you wish you can change the COM port number in Device Manager, but this is usually not necessary. However, you should note or write down the COM number. If you wish, you can also open the Arduino IDE and try out a couple of sample programs. However, here we want continue straightaway with Bascom.

In Bascom you can choose from a large variety of programming devices and boot loaders under the menu item Options/Programmer. The right setting is "ARDUINO" (not "Arduino STK500/2"), as shown in **Figure 6**. It's also important to con-

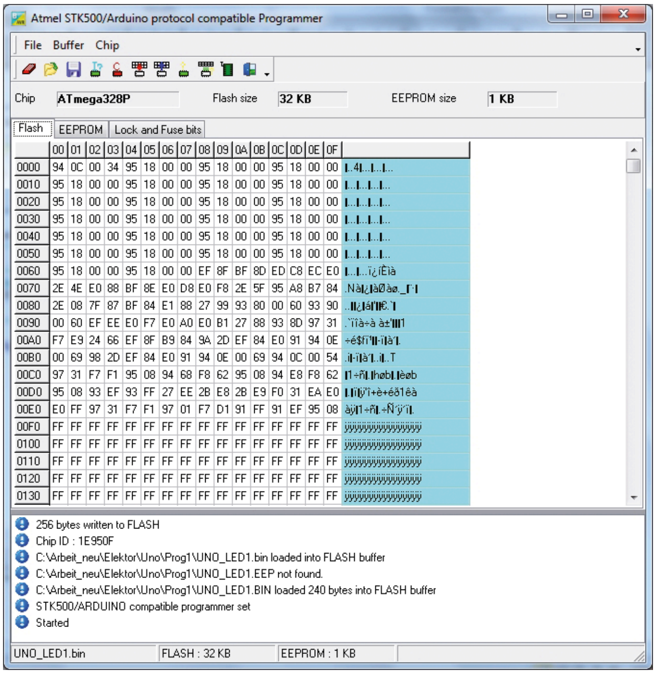


Figure 7.
The compiled program in the form of hex numbers.

figure the right COM port (in this example COM2), and it's essential to configure the right baud rate (115,200). This is because both parties must agree on the data transmission rate in bits per second (baud). We'll come back to this subject later on in the course when we talk about sending messages and measurement data to the PC. If you only go by the Bascom help and the examples included with Bascom, some of your settings will be wrong here because the Arduino Uno is relatively new and the Arduino camp has only recently changed to the highest standard transmission rate of 115,200 baud. All in the interest of fast data transfer and correspondingly superfast device programming. Time is scarce in our fast-paced era.

In Figure 6 you can see that the option "Auto Flash" has also been ticked. This saves a mouse click later on, and it reduces the risk of doing something wrong. The "Terminal Emulator" option is also very helpful if you want to have programs send messages and data to the PC later on. Once the right settings have been selected, close the window with ESC. That's the previously mentioned difference between the demo version and the full version, which has an "OK" button. If you don't know about using the ESC key, it looks like the program is stuck. Incidentally, ESC also works with the full version.

Now you're done, and it's time for action. As previously mentioned, press F7 to compile the source code. Then press F4 to start the programmer. Alternatively you can click the small green PCB symbol "Program Chip"; the result is the same. In any case, a new Programmer window opens. There the compiled program (*UNO_LED1.bin*) is already nicely loaded and displayed in the form of hexadecimal numbers (**Figure 7**).

Good job: it works!

After this you can sit back and relax; the rest is automatic. However, you can also do everything yourself. If you opt for this, you must be very careful because there is a function here that can completely erase the microcontroller memory, which means that the Arduino boot loader is also gone. Stay well clear of anything with the word "erase" in it. The only right option is "Chip/Write Buffer into Chip".

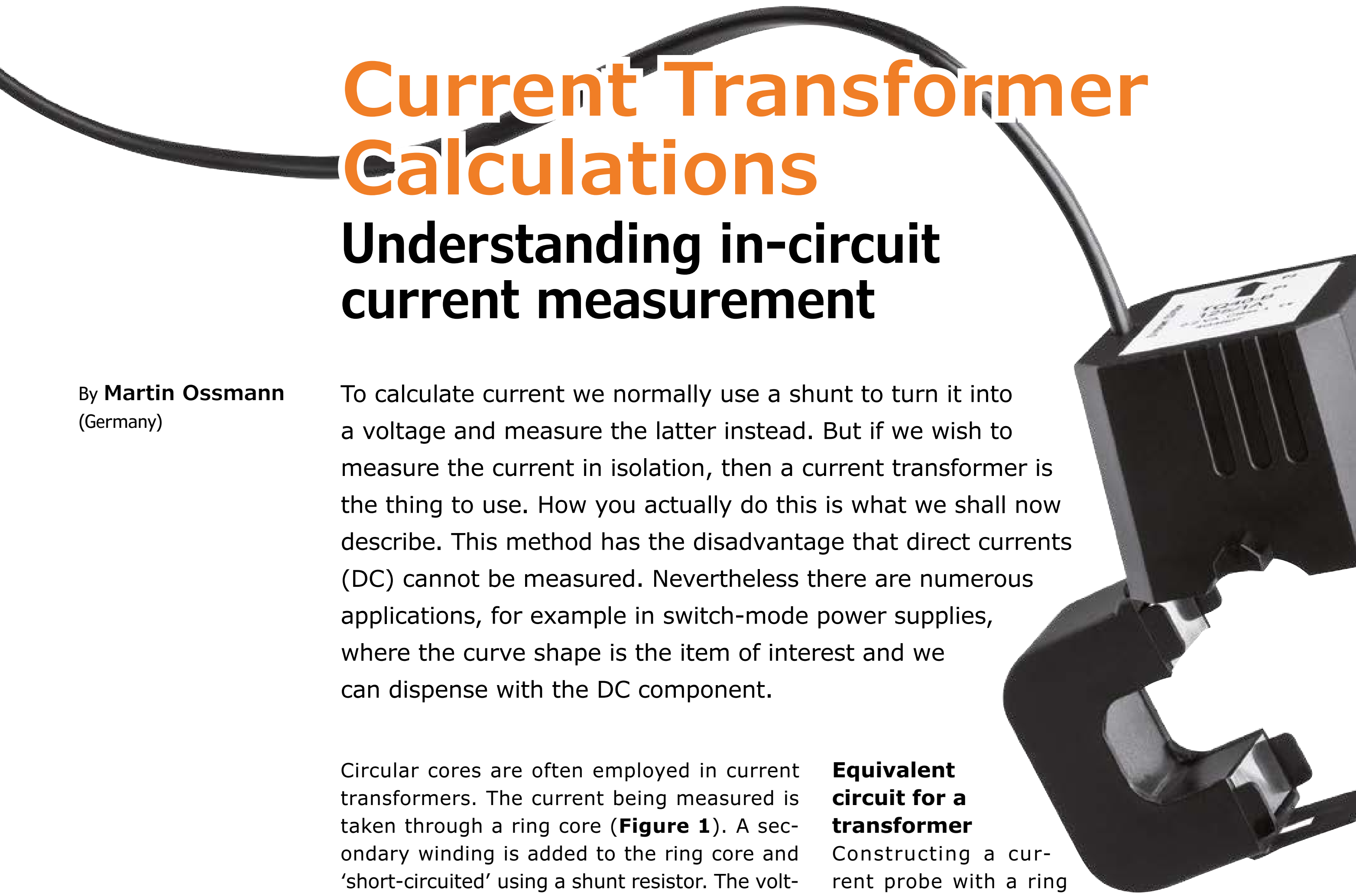
Now you will see a long chain of messages in the programmer window, with the lovely word "Started" at the end. During the programming process you can see from the activity of the Tx and Rx LEDs on the Uno board that a lot of data traffic is going on. At the end the window closes again.

The downloaded program starts running immediately after the end of programming, and the yellow LED blinks. It may be a tiny program, but it's a big step for you as the programmer. If you're not quite sure about all this, try making some small changes to the program. For example, you can change the blinking rate. For really fast blinking, change *Waitms 500* to *Waitms 100*, or for very slow blinking change it to *Waitms 2000*. After editing the code, recompile it and reprogram the microcontroller, and then check the result. If the LED does exactly what you programmed it to do, there's no room for doubt: it really works! In the next instalment we turn our attention to inputs.

(120574-I)

Web Links

[1] www.elektor.com/arduino
[2] www.mcselec.com
[3] www.elektor-magazine.com/120574
[4] http://arduino.cc/en/Main/Software



Current Transformer Calculations

Understanding in-circuit current measurement

By **Martin Ossmann**
(Germany)

To calculate current we normally use a shunt to turn it into a voltage and measure the latter instead. But if we wish to measure the current in isolation, then a current transformer is the thing to use. How you actually do this is what we shall now describe. This method has the disadvantage that direct currents (DC) cannot be measured. Nevertheless there are numerous applications, for example in switch-mode power supplies, where the curve shape is the item of interest and we can dispense with the DC component.

Circular cores are often employed in current transformers. The current being measured is taken through a ring core (**Figure 1**). A secondary winding is added to the ring core and 'short-circuited' using a shunt resistor. The voltage across this shunt is measured with the oscilloscope. **Figure 2** illustrates the result of this kind of measurement.

Our pulse voltage source covered in the January/February issue [1] provides the 10 V impulses used here (blue trace). Using a 20 Ω resistor produces 500 mA current impulses, arising from which we get shunt measurement impulses of 50 mV (red trace). The current transformer produces pulses of 5 mV. The frequency is 250 kHz. In fact the result looks very promising, with the current transformer doing a good job of reproducing the shape of the square-wave signal pulses. A problem arises if we drop the frequency to 50 Hz (**Figure 3**); the current transformer no longer reproduces the square-wave signal correctly (green trace).

The reason for this will be discussed in a moment. Using circuit modeling we can establish how to dimension a current transformer of the kind required.

Equivalent circuit for a transformer

Constructing a current probe with a ring core calls for a good concept model of the current transformer that will result from it. That's what we will now discuss. A transformer consists of a primary winding (Index 1) and a secondary winding (Index 2). The number of turns in each coil is n_1 and n_2. We can measure the so-called no-load inductances L_{1o} and L_{2o} (**Figure 4**). If the primary and secondary windings are fitted to the core in the same way, we can also calculate these with the help of the value A_L. That is to say:

$$L_{1o} = n_1^2 \times A_L \quad \text{and} \quad L_{2o} = n_2^2 \times A_L$$

The ratio of the number of turns is then $N = n_2 / n_1$. For instance, with a core made of Vitroperm with A_L = 80 µH / turns² the following arises:

$$n_1 = 1 \quad L_{1o} = 80\ \mu H \quad n_2 = 25 \quad L_{2o} = 50\ mH$$
$$N = n_2 / n_1 = 25$$

Another characteristic of interest with transformers is the coupling factor k. We can determine k by measuring the short-circuit inductance. If we short out the secondary, in our example we find L_{1k} = 0.2 μH. On the other hand, if we short out the primary we measure L_{2k}. However, this value is not independent of the other three and is in fact:

$L_{2k} \times L_{1o} - L_{1k} \times L_{2o} = 0$

This relationship enables us to calculate, for example, L_{2k}. In our example this turns out to be:

$$L_{2k} = \frac{L_{1k} \times L_{2o}}{L_{1o}} = 125\mu H$$

The coupling factor arising from this is:

$$k = \sqrt{1 - \frac{L_{1k}}{L_{1o}}} = 0.9987492178$$

In this case it turns out to be very close to unity (1), meaning that a current transformer using a ring core is virtually ideal.

The three characteristics L_{1o}, L_{2o} and L_{1k} complete the description of the transformer. To define the relationship of transformer we use an equivalent circuit. The generic form of this is shown on the left-hand side of **Figure 5**.

It consists of an ideal transfer device (transformer) with a transfer ratio of 1:M. We then have a main (or magnetization) inductance L_m, the primary stray inductance L_{1s} and the secondary stray inductance L_{2s}. Since the equivalent circuit contains four characteristics, we cannot specify it unambiguously but can select certain characteristics freely. Consequently the elements do not have direct physical counterparts.

From now on we will use the equivalent circuit shown on the right-hand side of Figure 5. This has the primary stray inductance L_s, arising directly from the short circuit measurement $L_s = L_{1k}$. The main inductance L_m results from the no-load measurement / turns formula:

$L_m = L_{2o} = n_2^2 \times A_L$.

The transfer ratio M is worked out as:

$M = N / k = 25.0313... \approx N$.

As the coupling factor is virtually 1, M is almost

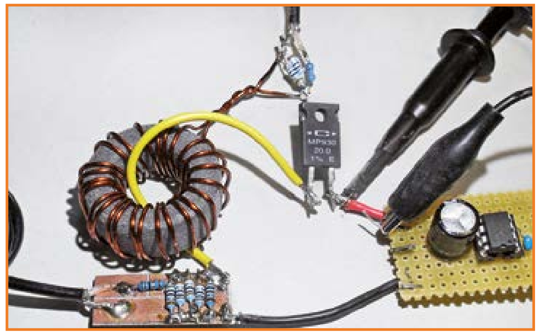


Figure 1.
Ferrite ring used as a current transformer.

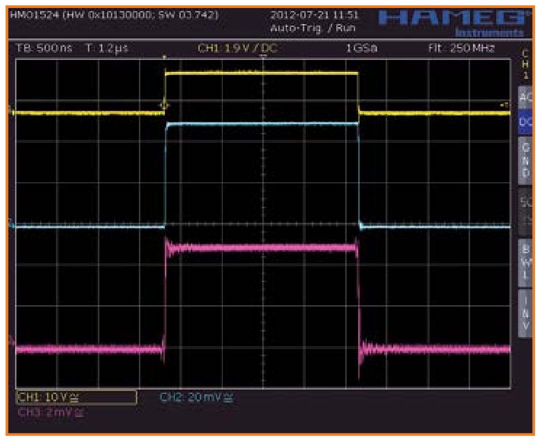


Figure 2.
Current measurement of a 250 kHz square-wave signal using a current transformer.

Figure 3.
Measuring a 50 Hz square-wave signal with a ferrite ring.

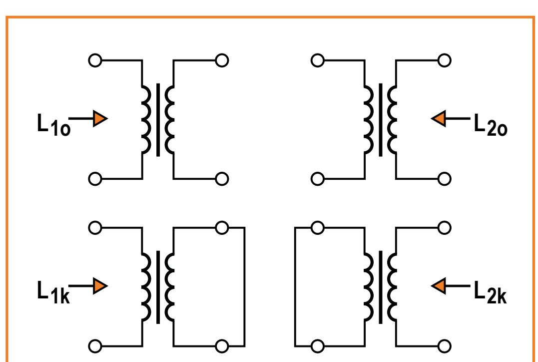


Figure 4.
No-load and short circuit measurement in a transformer.

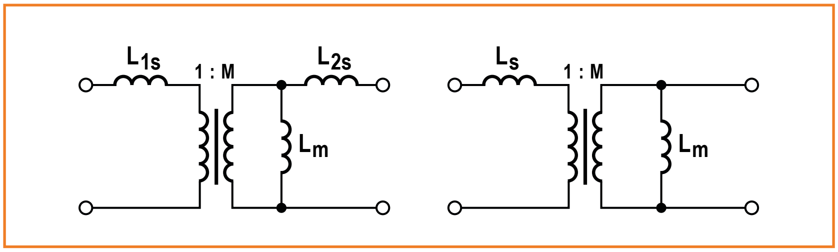


Figure 5.
Equivalent circuit of the transformer.

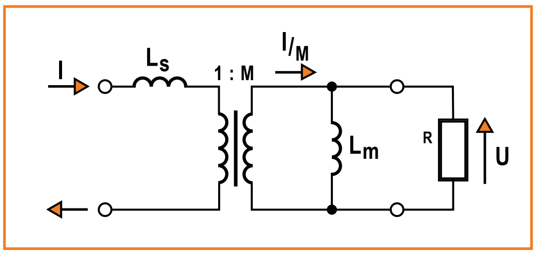


Figure 6.
Equivalent circuit of the current transformer.

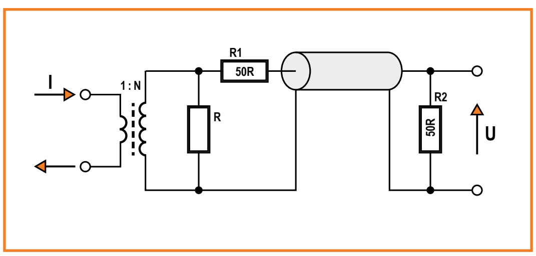


Figure 7.
Current measurement set-up using transformer.

exactly the same as the ratio of the number of turns. We have now defined all the elements of the equivalent circuit (**Figure 6**). Using these elements we can now carry out simulations in SPICE for example.

We now allow the current being measured (I) to flow through the primary. On the secondary side we connect a (small) resistor R and measure the voltage on it. Since the stray inductance lies in series, it has no influence on the current flowing through the transformer. The transformed current I/M on the secondary side passes through the parallel circuit of L_m and R to produce the voltage measured U. For current transformers we normally have $n_1 = 1$, producing $N = M = n_2$, which we shall now take as a given.

When the frequency of current I is sufficiently high that the coil impedance is significantly greater than R, we can ignore L_m and simply say $U = R \times I / N$. The transfer impedance is thus $R_{TR} = U / I = R / N$. Using a complex AC calculation we can now recalculate that a system of this kind constitutes a high-pass with the cut-off frequency

$$f_g = \frac{R}{2 \cdot \pi \cdot L_m}$$

$$R_{TR} = \frac{R}{N} \qquad f_g = \frac{R}{2 \cdot \pi \cdot A_L \cdot N^2}$$

Dimensioning

With these insights we can next assess how to dimension a current transformer. If we make R larger, the voltage signal increases, but so does the lower cut-off frequency, meaning we can no longer measure low frequencies (simultaneously the voltage drop on the primary side increases). The choice of R is therefore a compromise. Since oscilloscopes often display voltages in the mV range poorly, we should make R sufficiently large that moderate currents produce a voltage of several tens of mV.

Now we come to the number of turns. Raising N reduces the transfer impedance and with this, the voltage produced. At the same time, however, L_m gets larger and that reduces the lower cut-off frequency. Here too we must accept a compromise. Additionally we note that a higher value of A_L helps, because it lowers the boundary frequency, without affecting the transfer impedance.

Ferrite ring core

By way of example we will calculate the parameters (using a ferrite ring core) with a value for A_L of around 3 µH / *turns*². If R = 0.5 Ω and N = 25 we get:

$$R_{TR} = \frac{R}{2N} = \frac{0.5\,\Omega}{50} = 0.01\,\Omega$$

$$f_g = \frac{R}{2 \cdot \pi \cdot A_L \cdot N^2} = 42\,Hz$$

The factor of 2 in the denominator for the transfer impedance arises from the two 50 Ω resistors R1 and R2. These match the signal to a 50-Ω cable, creating a voltage divider that reduces the signal by half (**Figure 7**).

At a current of 1 A we obtain 10 mV on the oscilloscope, which is not a lot. The lower cut-off frequency is around 50 Hz, so with a ring core of this kind 50 Hz signals can be measured only with significant error.

Vitroperm, a magic material

A German company called Vacuumschmelze GmbH manufactures a material called Vitroperm, which possesses very high permeability (at low frequencies). Below 1 kHz their ring core (type T60006 L2025-W380) has the A_L value of around 80 μH/turns². This reduces, with other parameters, the boundary frequency to around 2 Hz. With this kind of core we can now construct a current probe that can be deployed without problem below 50 Hz (**Figure 8**).

A few words still need to be said on the 'internal' resistance of the current probe. Shunt R on the secondary is transformed on the primary side by a factor of $1/N^2$. So if we use $R = 0.5\ \Omega$ and $N = 25$, the effective resistance on the primary side amounts to precisely 0.0008 Ω, which can be ignored in normal situations. Admittedly there is still a stray inductance $L_{1k} = 0.2\ \mu H$ in series with this, representing an impedance of 1.2 Ω at 1 MHz. The viability of the current transformer for higher frequencies is restricted by parasitic capacity and so on, but our simple example will make measurements into the MHz region without any problem.

(130410)

Figure 8.
Vitroperm core.

Web Links

[1] www.elektor-magazine.com/130005

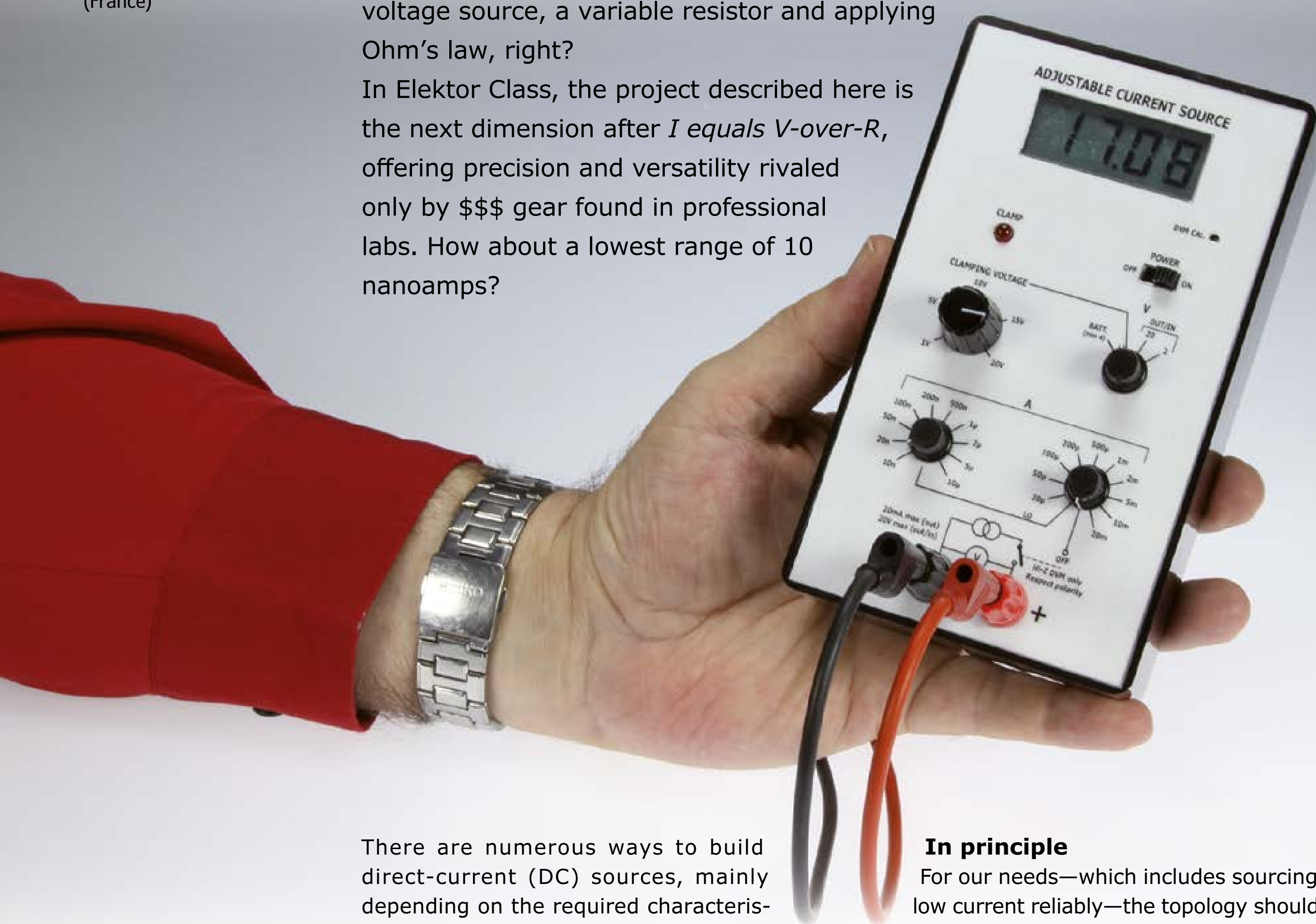
[2] www.elektor-magazine.com/130410

Precision Adjustable DC Current Source

With integrated hi-Z voltmeter

By **Henry Morizot**
(France)

Parroting the Science Class teacher an adjustable current source consist of any old voltage source, a variable resistor and applying Ohm's law, right?
In Elektor Class, the project described here is the next dimension after *I equals V-over-R*, offering precision and versatility rivaled only by $$$ gear found in professional labs. How about a lowest range of 10 nanoamps?



There are numerous ways to build direct-current (DC) sources, mainly depending on the required characteristics: current strength, fixed or adjustable, single or dual polarity, floating or grounded load, precision, anything else?

The lowliest of schematics is no more than a voltage source and a resistor. But as soon as you want some precision regardless of load current, you have to resort to a voltage-to-current converter using one or more operational amplifiers. The reference paper from Jerald Graeme [1] is recommended reading on the subject.

In principle

For our needs—which includes sourcing very low current reliably—the topology should take into account measuring the voltage across the load without stealing current from it. A theoretical configuration as shown in **Figure 1** is based on two operational amplifiers. The first one (U1) generates the current and the second one (U2) isolates the measuring circuit from the load. Both amplifiers are unity-gain (×1) connected, hence do not require any precision resistors.
The principle of operation: a secondary current source I_1 feeds reference diode D1 connected to amplifier U1's input with the diode return con-

nected to amplifier U2's output, whose input is connected to the load. A resistor *R*s is connected between amplifier U1's output and the load. Disregarding errors in the amplifier, the output voltage on each of them is the image of their respective input voltage. So the voltage across R_s becomes equal to the reference voltage V_{ref} from D1.
This constant voltage developed across R_s induces a constant current

$$I = \frac{V_{ref}}{R_s}$$

sourced by amplifier U1 and flowing exclusively through load Z_{Ld}, provided the bias current with amplifier U2 is insignificant.
Manual switching of different values for R_s affords outputting any current across a very large range.

The reference voltage selection is all but trivial. Appearing like a wasted voltage, you could attempt to make it as small as possible, but this is defeated by multiple error sources. First of all, the amplifiers "suffer from" initial offset voltage and common mode rejection ratio (CMRR). Actually, the voltage developed across R_s resistor is not exactly the same as that present at the reference. More precisely, this voltage could even deviate up to the following value:

$$V_{(Rs)} = V_{ref} + V_{os(U1)} + V_{os(U2)} + \frac{V_{ex}}{CMRR_{(U1)}} + \frac{V_{ex}}{CMRR_{(U2)}}$$

where
V_{os} = initial offset voltage;
V_{ex} = common-mode voltage;
CMRR= common-mode rejection ratio.

For example, if both amplifiers are specified for an offset voltage of 2 mV and a common mode rejection ratio of 78 dB; and if voltage extension is 20 V, we could obtain in the worst case: $V_{(Rs)} = V_{ref}$ + 9 mV, and if we choose V_{ref} = 100 mV, we can have an error up to 9%!

Fortunately, because all terms of the sum are polarized and may compensate themselves, this worst-case deviation is statistically unlikely, but it gives a good idea about a possible loss of precision.

Secondly, it is important to take into account the mechanical switching of the resistors defining the output current. For example, to obtain a current value of 20 mA, again using the 100-mV reference will require a resistance R_s of 5 Ω. If the rotary switch represents a contact resistance of 100 mΩ, this yields an additional 2% error, to which we should add the error induced by intrinsic printed circuit track resistance. Conversely, if we choose a higher reference voltage, say 10 V, added to some wasted energy, we will be faced with the difficulties inherent to tiny currents. First there is the value of R_s: to obtain, say, 10 nA (nanoamps) we'll need a 1-GΩ resistor—expensive and not easy to find with a tight tolerance. Next, at the resistor switching level, the finite insulation resistance of the rotary switch will be in parallel with this very large resistance and affect the required value. And still here, do not forget that the printed circuit itself is not a perfect dielectric.

Finally, we should reject any "exotic" value for the reference voltage (for example 1.235 V) because this value inevitably leads to "exotic" values for the R_s resistors themselves.
In this project, the reference voltage was chosen at 1.00 V, a satisfactory compromise.

Main Specifications

- Low Ranges (10): 10nA – 10µA
- High Ranges: (10) 10µA – 20mA
- 3.5 Digit LCD Readout
- Battery Powered (4x AAA)
- Extendable as a High Impedance Voltmeter (optional)
- Microcontroller Free

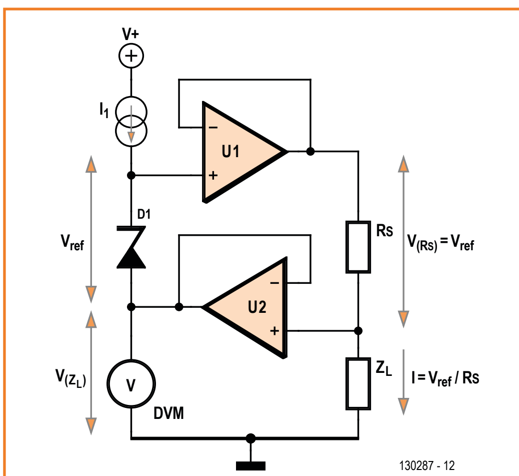


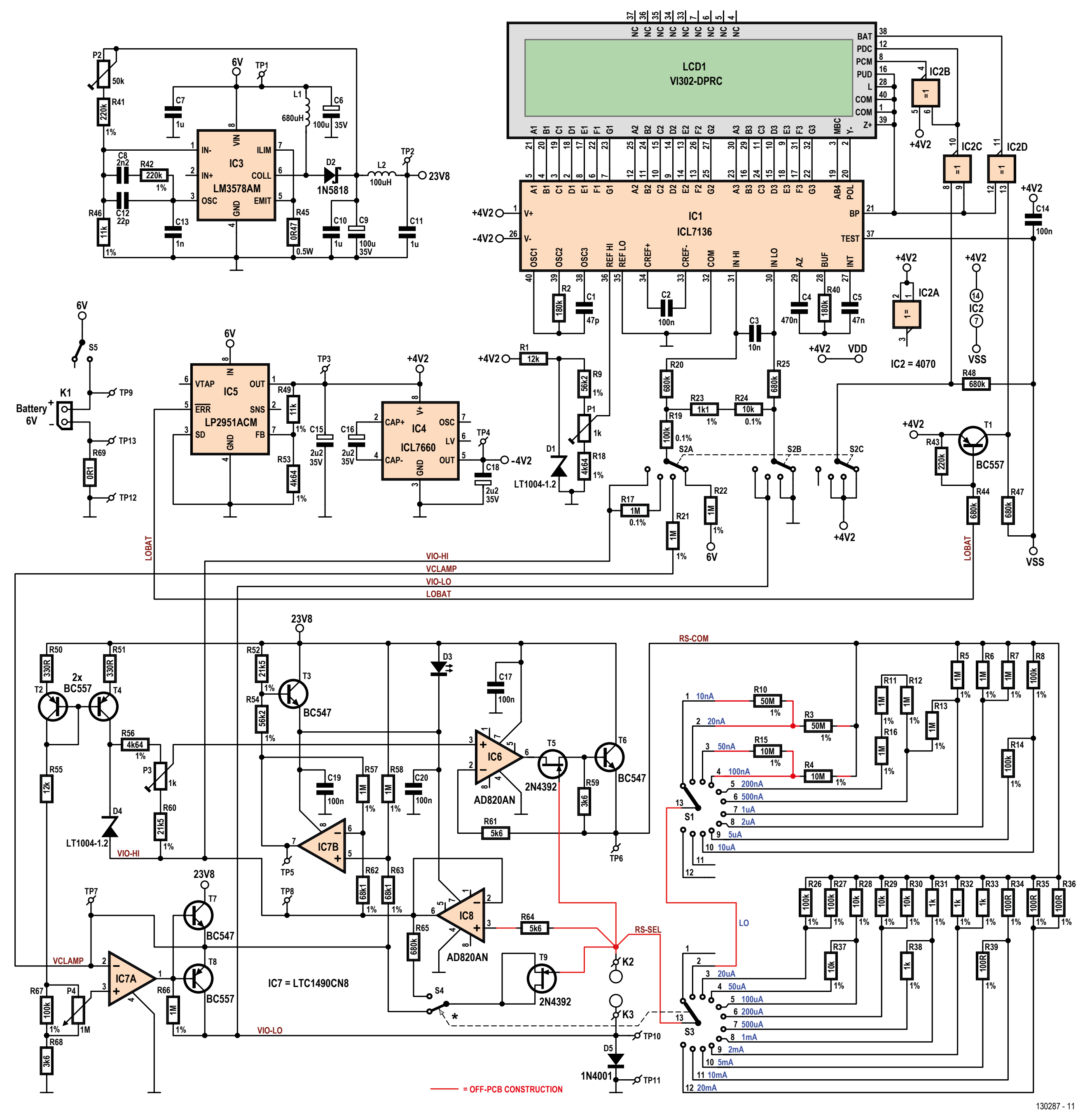
Figure 1.
Theoretical model of a near perfect current source.

Into the schematic

Figure 2 shows the schematic of the instrument. Its design is Analog Electronics Paradise and deserves a thorough discussion. Not a microcontroller in sight.

References

The current needed by the floating source reference is obtained from a current mirror comprising transistors T2 and T4. The master current of 215 μA through T2 comes from resistor



This circuit contains a FETish diode

chain R55, R67 shunted with P4, and R68. R67 minimizes impact from the large tolerance of pot P4, which delivers the output limiter adjustment voltage typically spanning 0.8 V – 20.4 V.
The 1:1 ratio of the current mirror output at T4's collector supplies the bias current for reference diode D4. T2 and T4 not being matched, emitter resistors R50 and R51 counteract a possible unbalance.
The resistor divider made up of R56, P3 and R60 allows precise adjusting of the source reference to 1.000 V. We've chosen an LT1004CZ voltage reference from Linear Technology, but it is also possible to use a slightly less temperature compensated type like the LM385BZ-1.2 from National Semiconductor.

Measuring amplifier
The bias current of measuring amplifier IC8 being an error term algebraically added to the source current, it is important to keep it insignificant relative to the minimum output current of 10 nA. For this reason, the amplifier is an AD820AN from Analog Devices which has JFET inputs having an extremely low bias current. It's quite a vintage device (20-year old), but possibly the only one to accept a single supply up to 30 V with rail-to-rail output and rather high precision. Very likely, it's still available because of that, even in a DIP package.
Neglecting the offset voltage, the output voltage of the unity-gain connected amplifier is the image of its input voltage. This output sinks the return current from the voltage reference. It sources some current for the voltmeter input attenuator and also for the negative branch of voltage limiter indicator's differential amplifier.
In the measuring circuit, diode D5 inserted between load return and Ground plays an important role, assuming in fact a triple function. Its main object is to raise the load return potential of a few hundreds millivolts, allowing the measuring amplifier IC8 to always work in a linear region. Everyone knows that operational amplifiers exhibit a lot of defects when approaching supply rail levels, especially the open-loop gain which collapses drastically. This causes an output offset voltage of a few up to hundreds of millivolts, depending on amplifier design and load conditions, mainly if the output has to source or sink current, the latter being the worst case around ground rail potential.
The sourced current—even the tiniest—always flows through D5, except when using the instrument as a voltmeter only. In that case, there is nothing except the current caused by the voltmeter's input attenuator, unfortunately proportional to measured voltage. In order to get a significant voltage drop across the diode, a tiny current is added, made up of two components: the collector current from T8, defined by the input resistance R58 + R63 with differential amplifier IC7B (and inversely proportional to the clamping voltage), and on the other hand the current sourced by amplifier IC7A through R66 (and proportional to the clamping voltage). The sum of these currents is nearly constant and about 20 microamps, causing a minimum voltage drop of approximately 420 millivolts.

Secondly, the voltage drop across D5 compensates for the voltage drop across 'clamping diode' (FET) T9 of the voltage limiter, thus making the clamping voltage across the load independent of the current sourced. Actually, this is not perfect because this compensation is only effective for currents above the smallest current forced through D5 as previously described. Also, the characteristics of diodes T9 and D5 are slightly different. That's why D5 is a type 1N4001, its voltage *vs* current slope resembling that of clamping diode T9 closer than a small signal diode like a 1N4148. Ideally, D5 would be identical to T9, but considering the cost of the latter, this is not essential because that compensation is only a secondary parameter.
Finally, in the case the instrument used as a voltmeter only, D5 together with resistors R61 and R64 protects the components against external voltages up to about 30 V, irrespective if the instrument is powered or not.

Figure 2. Schematic of the Precision Adjustable Current Source. The circuit has a lot of clever analog design aspects to it. The light lines indicate wiring in the air.

Normally, a high-impedance voltmeter is only used with high-impedance sources themselves and having such a low short-circuit current suggests inability to cause damage, but one never knows...

Current generator

Output current is obtained from amplifier IC6, the same type as that used for the measuring amplifier (AD820AN), the characteristics needed being nearly the same. It is unity-gain connected and buffered by transistor T6, the amplifier itself being unable to source the maximum required current of 20 mA.

Neglecting the amplifier offset voltage, the output voltage on T6's emitter is the image of the reference voltage at the wiper of P3, and this output is connected to the high-end common point of current selection resistors. On the other hand, through unity-gain measuring amplifier IC8, the low-end of the reference voltage is itself the image of the voltage present at the low end of the current selection resistor. Thus, the voltage across the current selection resistor is the image of the reference voltage, and this very voltage is applied to that resistor which in turn generates the required current.

Base-emitter resistor R59 with T6 implements a selection according to the output current. For currents up to 100 µA, the amplifier alone sources the current through this resistor, and then starting from 200 µA, the transistor operates as a buffer. Moreover, R59 sinks the leakage current of transistor T6 if it is greater than the required output current. This phenomenon may appear under exceptional circumstances, like if the instrument outputs the maximum current of 20 mA for some time into a zero or almost-zero load—the transistor dissipates about 460 mW, which raises its junction temperature of nearly 115°C, causing significant leakage current. If immediately afterwards you want the instrument to output the minimum current of 10 nA, the leakage current may be temporarily greater than this value, requiring the amplifier not to source current, but rather to sink the excess.

Surge current limiter

When the ratio between 'open-circuit' and 'load-applied' voltages is significant, the resulting surge current may vastly exceed the final value during the few microseconds needed for the amplifiers to stabilize. Despite more than 10

years experience on the previous instrument version without any damage on checked components, a surge limiter is included here in the form of a JFET transistor which acts by momentarily cutting off the load from the current generator. JFET transistors are subject to large tolerance in terms of electrical characteristics. Consequently users have to check that the transistor to be mounted is able to fulfill its function. This can be done easily with the test circuit described further on. As with measuring amplifier IC8, the gate current of transistor T5 is an error term added to the sourced current. However, with the type of transistor mentioned, this current is extremely low because of the small reverse voltage applied (1.7 V max.).

Another simple and efficient but more constraining way to eliminate the surge current consists of adjusting the voltage limiter to the minimum

before applying the load, and then increase it until the clamping indicator turns off.

Current selectors

Output current selection is obtained by manual switching of resistors. Because of the large number of current values (20), this selection is split into two ranges: high and low, allocated to rotary switches S3 and S1 with 12 positions.

An “OFF” position outputting no current allows the instrument to be used as a simple high-impedance voltmeter, but with some limitations (see the section on the voltmeter). In that mode, there is also a voltage limiter disconnect option consisting of a microswitch actuated in the “OFF” position. As you can see, the spread of the current selection resistor values applied is rather low, thanks to some series and parallel combinations, even if it means slightly more resistors. The precision of these resistors directly affects the output current precision. For such an instrument, a 1% tolerance is enough and moreover easy to get.

Voltage limiter

Voltage limitation is done by the unity-gain connected amplifier IC7A buffered by complementary transistors T7 and T8, the output being connected to the load through T9 acting as a clamping diode. Voltage limiting is adjustable with the front panel mounted potentiometer P4 from approximately 1 V up to 20 V.

Transistor T8 begins to sink the output current as soon as the voltage across the load is greater than that present at T8’s emitter, the clamping diode T9 getting forward biased. Transistor T7 is only useful during the reverse recovery period of T9 to extract the charges previously accumulated. This time, the chosen amplifier is an LT1490A from Linear Technology (everyone has to live), a micropower dual version with rail-to-rail input and output, operating off a single supply up to 44 V and still available in a DIP package.

Like the measuring amplifier input, the leakage current of FET-ish diode T9 is an error term which must be insignificant up to a reverse voltage of 20 V. In the previous instrument version, this diode was an ultra-low leakage type BAV45, but unfortunately it is today quite impossible to find. A good alternative is to use the gate-source diode from JFETs like 2N4391..4393 or J201...203 (source connected to drain being the cathode). Caution! Unless having them selected, do not use the plastic case series PN4391...4393, which are specified with a gate reverse current up to 10 times higher.

Voltage limit visual indicator

The voltage limit indicator is comprised of differential amplifier IC7B, transistor T3 and LED D3. By way of unity-gain connected amplifier IC8, the differential amplifier measures the voltage across “diode” T9. This amplified voltage appears between amplifier output and the positive supply rail, and not ground as you might expect. The arrangement saves energy, the LED current being drawn from the supply current of amplifiers IC7 and IC8 (about 1 mA total), these amplifiers suitable for powering with a voltage slightly smaller than that required for current source amplifier IC6. The only requirement is to use a high efficiency red LED with an intrinsic low

forward voltage (about 1.6 V at 1 mA).
By means of resistors R52 and R54 this configuration referenced to the positive supply rail also provides one of the components needed for the minimum current required through D3.
If T9 is reverse biased (limiter off), the output of IC7B swings high, switching T3 into conduction, effectively shorting out the LED. Conversely, as soon as T9 becomes forward biased at about 250 mV, the amplifier output voltage drops low, switching the transistor off and causing the LED to light. This 250-mV threshold across the diode corresponds to a current of about 10 pA, equal to 0.1% of the minimum output current. Consequently the LED lights as soon as there is a hint of voltage limiting going on.
It's fair to ask why have gain on the differential amplifier IC7b and then kill it by a resistor divider to drive T3. Why not have the composite gain and a direct drive of the transistor? It's merely because the output of the amplifier cannot go beyond the supply rail and so cannot drive the base of the transistor more positive than its emitter, meaning the LED would remain on all the time. Another reason for the presence of the resistor divider is to not approach the avalanche voltage of T3's base-emitter junction if the amplifier output drops low. When the instrument is used as a voltmeter only, this could occur if the voltage limiter disconnect option is installed and the limiter not adjusted to the maximum value. Exactly in that case the LED indicator could even be used as a voltage threshold indicator by adjusting the limiter accordingly.

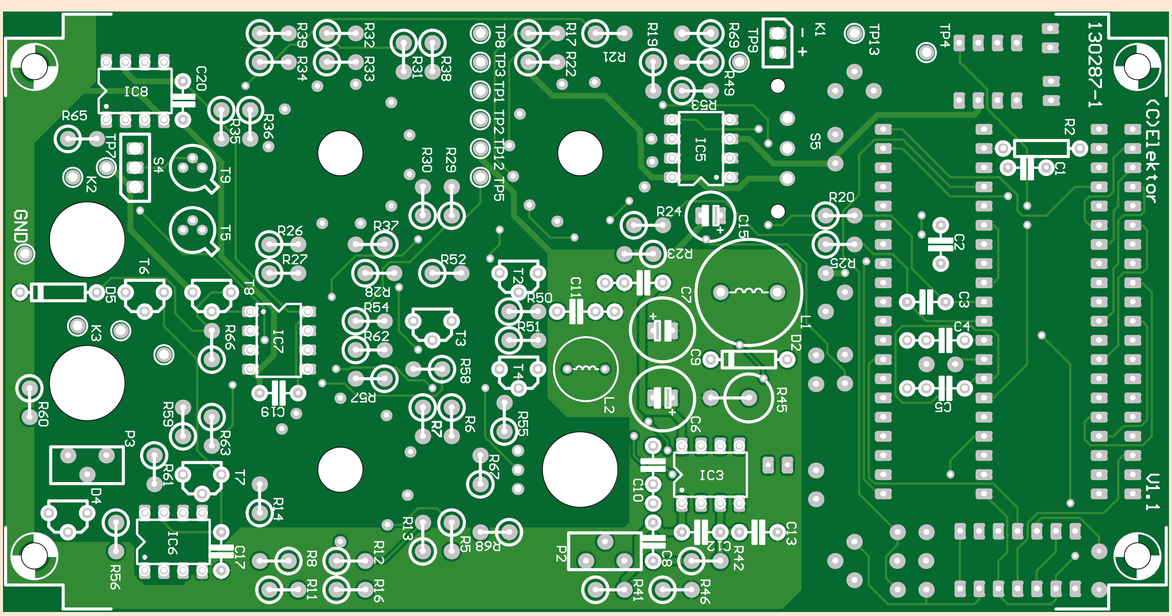


Figure 3.
The printed board is populated at two sides. See also the notes and photographs covering components "in the air".

Component List

Resistors
R1,R55 = 12kΩ 5% 400mW
R2,R40 = 180kΩ 5% 400mW
R3,R10 = 50MΩ 1% 1W
R4,R15 = 10MΩ 1% 600mW
R5,R6,R7,R11,R12,R13,R16,R21,R22,R57,R58 ,R66 = 1MΩ 1% 500mW
R17 = 1MΩ 0.1% 250mW
R8,R14,R26,R27,R67 = 100kΩ 1% 500mW
R19 = 100kΩ 0.1% 250mW
R9,R54 = 56.2kΩ 1% 600mW
R18,R53,R56 = 4.64kΩ 0.5% 250mW
R20,R25,R44,R47,R48,R65 = 680kΩ 5% 500mW
R23 = 1.1kΩ 1% 250mW
R24 = 10kΩ 0.1% 250mW
R28,R29,R30,R37 = 10kΩ 1% 500mW
R31,R32,R33,R38 = 1kΩ 1% 500mW
R34,R35,R36,R39 = 100Ω 1% 400mW
R41,R42,R43 = 220kΩ 5% 500mW
R45 = 0.47Ω 5% 1W
R46,R49 = 11kΩ 1% 500mW
R50,R51 = 330Ω 5% 500mW
R52,R60 = 21.5kΩ 1% 600mW
R59,R68 = 3.6kΩ 1% 500mW
R61,R64 = 5.6kΩ 5% 500mW
R62,R63 = 68.1kΩ 1% 600mW
R69 = 0.1Ω 5% 1W
P1,P3 = 1kΩ multiturn preset
P2 = 50kΩ multiturn preset
P4 = 1MΩ miniature cermet potentiometer

Inductors
L1 = 680µH 980mA 0.46Ω
L2 = 100µH 580mA

Capacitors
C1 = 47pF 5%
C2,C14,C17,C19,C20 = 100nF 20%
C3 = 10nF 10%
C4 = 470nF 5%
C5 = 47nF 1% 160V polypropylene
C6,C9 = 100µF 35V radial 20%
C7,C10,C11 = 1µF 63V 10%
C8 = 2.2nF 100V 10%
C12 = 22pF 200V 5%

Power supplies

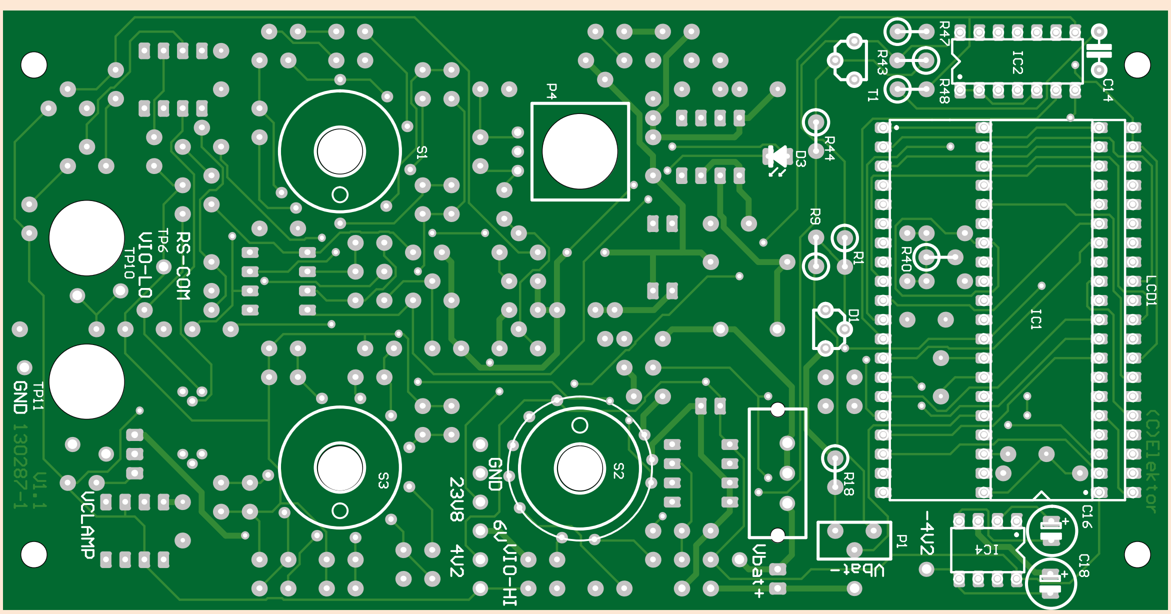
The instrument is runs off 6 volts nominal supplied by four alkaline 1.5-volt AAA cells. Three internal voltages are generated : +23.8 V for the current source, +4.2 V and – 4.2 V for the voltmeter. The 23.8 V voltage is the minimum value required to ensure correct operation in the worst case. This voltage is obtained by a classic switching regulator type LM3578A from National Semiconductor, and it is adjustable with trimpot P2. Additional output filtering by L2 and C11 reduces the ripple on the supply rail.

The +4.2 V supply is obtained from an LP2951 micropower adjustable linear regulator from National Semiconductor. The output voltage is fixed by resistors R49 and R53. This regulator also includes a comparator, switching whenever the output falls out of regulation by about 5%. The comparator output is used to activate the Low Battery indicator on the voltmeter display. The 4.2 V level causes the LO BATT indicator to be turned on when the battery voltage falls below 4 V, which leaves a comfortable time margin, the instrument remaining fully functional down to approximately 3.5 V.

The −4.2 V rail also needed for the voltmeter supply is obtained from the +4.2 V section by means of an ICL7660 charge pump converter from Intersil.

Integrated voltmeter

The integrated voltmeter is a very classic 2-Kcounts type. It is based on an ICL7136 A/D converter from Intersil, directly driving a ½-inch



C13 = 1nF 100V 5%
C15,C16,C18 = 2.2µF 25V radial tantalum

Semiconductors

D1,D4 = LT1004CZ-1.2
D2 = 1N5818
D3 = LED, red, low current, 3mm
D5 = 1N4001
T1,T2,T4,T8 = BC557
T3,T6,T7 = BC547
T5,T9 = 2N4392
IC1 = ICL7136
IC2 = CD4070
IC3 = LM3578AM
IC4 = ICL7660
IC5 = LP2951ACM
IC6,IC8 = AD820AN
IC7 = LT1490CN8

Miscellaneous

S1,S3 = 1x12 rotary switch, Grayhill 56SD30-01-1-AJN (Digikey # 440-7657)
S2 = 3x4 rotary switch, TE Connectivity Alcoswitch (1186546)
S5 = PCB mount slide switch, Knitter Switch MFP201N (134-0113)
LCD1 = LCD module 3.5 digit, transflective, Varitronix VI302-DPRC (Digikey # 1183159)
TP1,TP2,TP3,TP4,TP5,TP6,TP7,TP8,TP9,TP10,TP11,TP12,TP13 = test pin, 1mm diameter, Stelvio Kontek 3110014000540 (Digikey # 305-0913)
K2 = test terminal PTFE, ITT Canon 011-1004-040FB9 (1347796)
Insulated terminal 30A, black (2112490)
Insulated terminal 30A, red (2112491)
Black cap knob, 1/4” shaft (1209791)
Black cap knob, 11.6mm for 1/8” shaft (Digikey # 259-6812)
2x holder for 2 AAA batteries
Enclosure: Teko Coffer A/7, dim. 160x95x45mm
4 x Hexagonal PCB spacers (see text)
PCB # 130287-1 from Elektor Store

(Numbers only in round brackets are Newark/Farnell order codes)

Figure 4.
Not a PCB design error, this: To avoid stray leakage currents, components and a wire are soldered directly to the pins of the range selector switch. PCB tracks simply are not up to this task.

Figure 5.
The + input of the instrument is a star junction with components connected in the air rather than to lossy PCB tracks.

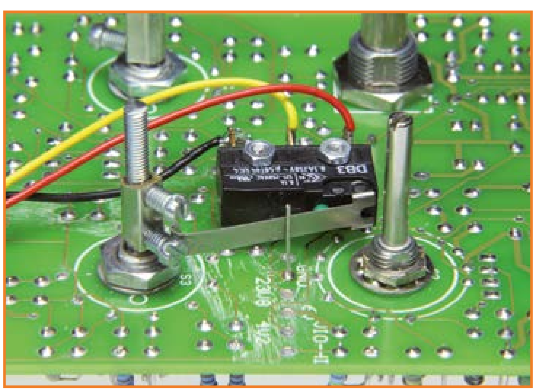


Figure 6.
A microswitch attached to the PCB and wired into the circuit creates a high-impedance voltmeter option on the instrument.

high 3½-digit LCD. The measuring reference voltage is obtained from an LT1004CZ reference diode, the same as that used for the current source. The resistor divider made up of R9, P1 and R18 allows precise adjustment to 100.0 mV. The front panel accessible trimpot permits voltmeter calibration without opening the case.
Quadruple XOR gate IC2 is required for the display to drive both the decimal points and the Low Battery indicator. The latter is driven by the comparator in IC5, via T1 doing the level conversion.
Voltage measurements at the instrument terminals capitalizes on the intrinsic capability of differential inputs on the A/D converter, the load return not being grounded, but instead connected to the anode of diode D5.
The 4-position rotary switch coupled with an input attenuator allows selection of either the 2 V or 20 V range for output/input voltage, and the 20 V range for limiter voltage and battery voltage.
Thanks to its current source disconnection capability, the instrument can be used as a voltmeter only with a very high input impedance. However, this function comes with the following restrictions:

- the current selector must be in the “OFF” position;
- the voltage limiter must be adjusted to the maximum value of 20 V (except if the voltage limiter automatic disconnect option is installed);
- the voltage to be measured must respect the instrument polarity (however, it is still possible to measure a reverse voltage up to approximately 400 mV);
- the input voltage must not exceed 20 V (nevertheless, the instrument powered or not is protected against voltages up to about 30 V).

Construction

The care and precision used in designing this instrument should be reflected fully by your construction on board # 130287-1 of which the **two overlays** (front side and rear side) are pictured in **Figure 3**, together with the **Parts List.**
For obvious reasons the range selector section should not exhibit high contact resistance, jitter, leakage resistance or stray capacitances.

Figure 7.
The board and the front panel are secured with hex PCB spacers to make up a solid assembly.

Hence the switches high-end types you should not attempt to replace by cheap devices. Basically for the same reason the 10-MΩ and 50-MΩ resistors are directly wired on the rotary switch contacts. What you see in **Figure 4** is a not a quick & dirty fix but a necessity.

A star point is available at K2, the + measurement terminal of this circuit. Look at **Figure 5**. A red, insulated banana socket is mounted on the front panel. There is a hole in the PCB close to that for the red socket. A PTFE (Teflon™) test terminal is press fitted into this hole and the following connections are soldered to it:

- the gates of T5 and T9;
- R64, which is ‘mounted in the air’ between this terminal and IC8. Note that pin 3 of IC8 is bent out of its socket terminal to solder straight to R64;
- a wire to the common contact of S3;
- a wire to the red terminal on the front panel.

Components are mounted on both sides of the PCB—check the PCB overlays and the pictures in **the photographs** at various places here to see which component goes where. The LCD is mounted on top of IC1 using two stacked socket strips to get it at the correct height.

Some remarks on S4—this switch can be used to use the circuit as a high-impedance voltmeter. Note: S4 is drawn in ‘current source’ position in the schematic. The most elegant solution (also mechanically most complicated) is to use a microswitch that registers position 1 of rotary switch S3. This microswitch is attached to the PCB or mounted to the back of the front panel. An extender attached to the shaft of S3 (discussed below) actuates the lever on the microswitch. Luc Lemmens at Elektor Labs glued two M2 bolts to the PCB—it’s a bit tricky to get the microswitch to register in the correct position, but it can be done, see **Figure 6**. Another, slightly less com-

plicated option is to add an extra SPDT (slide) switch for S4 to the front or side of the case. If the voltmeter feature is not wanted or needed, a simple jumper wire on the PCB can short S4 to permanently set current source mode.
The shafts of rotary switches S1 and S3 are too short to protrude through the front panel. Extensions for 1/8″ shafts are difficult to find, although suitable ones may be found in RC model shops. Diameters of 1/8″ are commonly used there for shafts and spindles—Luc found some couplings there but these were too high to fit between the switch and front panel. Therefore we used an old, simple trick by removing the plastic from a small screw terminal block and using its metal parts as a coupling between the shaft of the switches and a 3.2-mm diameter metal shaft (a decapitated M3 bolt will do too). Space between PCB and front panel is tight; you may need to file the ends to make the coupling fit there. On S3, the shaft extender coupling can be used to actuate microswitch S4.

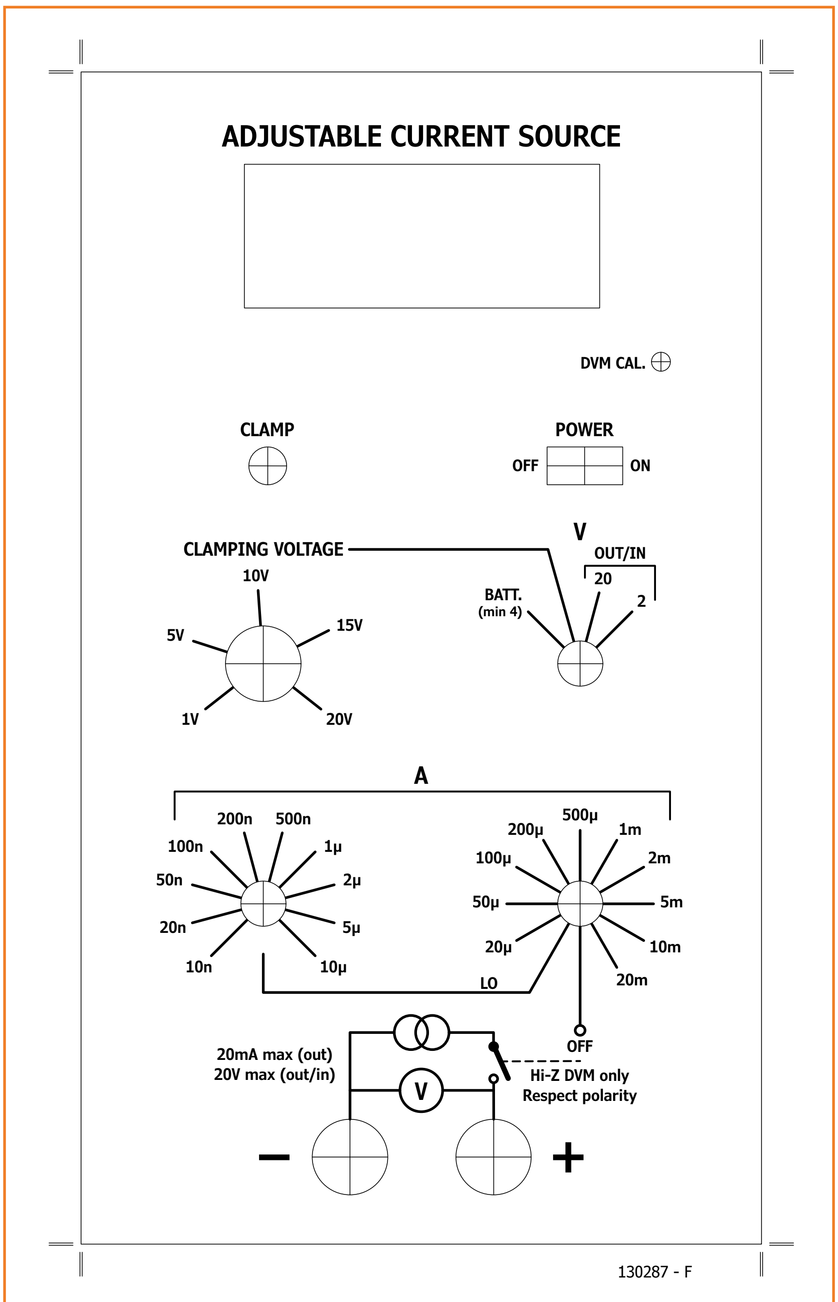


Figure 8.
Suggested front panel layout printed here at actual size.

The PCB is mounted to the front panel using four hex studs (12 mm height). We glued them to the backside of the panel with 2-component glue (for aesthetics' sake on the pictures). Sure, countersunk bolts through the panel present a more durable solution. We used some washers between the studs and the PCB to be able to mount the LCD flush with the backside of the front panel. The finished board and front panel assembly is pictured in **Figure 7**. The arrangement of controls on the front panel is shown in **Figure 8**.

Slide switch S5 (power) is originally a DPDT type, but we cut off three pins of the second switch to save space on the PCB. Two battery holders in series for two AAA cells each power the current source. One is mounted on the bottom of the case to the left and one to the right of the capacitors underneath the LCD. In a taller enclosure one holder for four batteries (even AA types) may do the job too.

(130287)

Reference

[1] Precision DC Current Sources, Jerald Graeme, Burr-Brown Corp., *EDN* April 26 and May 10, 1990.

ATmega on the Internet (2)

Using Raspberry Pi as a network gateway

By
Dieter Holzhäuser
(Germany)

All you need for remote control of an ATmega32 microcontroller from your home LAN or the Internet is a second computer connected to the ATmega32 microcontroller over a serial link. Here the second computer is a bare-bones design in the form of a Raspberry Pi. In this instalment we show you how to integrate the hardware described in the first part of this series into your home LAN and into the Internet.

At the end of last month's instalment we had a microcontroller connected to a Raspberry Pi board over a serial link, and we used it to blink an LED under the control of a terminal. Controlling the demo program on the ATmega microcontroller using a keyboard and monitor connected to the Raspberry Pi is pretty close to remote control. However, if you really want to control your own electronic devices remotely your best option is to use your home LAN, or the Internet if you want access from anywhere in the world. Here we tell you how it works.

Secure shell / SSH

The terminal remote control arrangement described in the previous part can be extended to a LAN and to the Internet by using a secure shell (SSH). In the simplest case, SSH allows command lines from one computer to be transferred over a network and appear on another computer. Data exchange with SSH is encrypted and is relatively secure.

The term SSH stands for a network protocol and for the programs that use this protocol. The SSH program running in the background on the Raspberry Pi is the SSH server. It provides the SSH service.

The service offered by the server is used by an SSH client. The computer hosting the client may be located in the same LAN as the Raspberry Pi or somewhere else, linked over the Internet. If you do not have Linux on your remote control computer, you can simply boot with a Linux Live CD. These CDs are often included with popular computer magazines. With the usual distributions you can open a console without any special installation and run the SSH client as described below.

The Linux system on the Raspberry Pi has a built-in SSH server. It can be activated under *raspi-config* (see Part 1).

SSH on the LAN

Naturally, a Raspberry Pi does not offer all the features of a full-fledged desktop PC. Nevertheless, for some applications it makes sense to operate devices on the LAN under remote control via SSH without a monitor or keyboard. A LAN is a private local network, typically with Internet access, in which a network router such as the FRITZ!Box looks after all the details.

The same protocols are used on LANs as on the Internet: TCP and IP. Devices on a LAN (called nodes) typically have their own private IP addresses. The router separates the LAN from the WAN, which means the Internet. In this way it screens the LAN and protects it against access from the Internet. Only the router appears on the Internet as a node with a public IP address (host). Although we focus on the German FRITZ!Box in this article, it is certainly possible to use other Internet routers with similar capabilities for the purpose described here.

The detailed settings for the Raspberry Pi can be configured in the user interface of the FRITZ!-Box. Its IP address can also be seen there. You should configure the router so that the Raspberry Pi is always assigned the same local IP address. If the settings are configured as shown in **Figure 1**, the SSH client can be called from a Linux computer on the LAN by typing the following command line:

```
ssh pi@192.168.178.28
```

Another option is:

```
ssh pi@raspberrypi
```

The command line prompt of the Raspberry Pi appears after the user enters the password *pi*. Next you should launch *picocom* on the Raspberry Pi. If the demo program *suidemo1.c* is running on the ATMega32, the lines with three blinking asterisks shown in part 1 will appear on the monitor. After closing *picocom*, you can type *exit* to close the *ssh* program.

SSH on the Internet

Establishing a SSH link over the Internet is basically the same as on a LAN. However, the settings for this are considerably more complex because you have to connect a SSH client on the Internet to the SSH server on the LAN. This works the opposite way as the usual situation in which a browser on the LAN, acting as a client, requests pages from a web server and these pages are sent to the client over the Internet. The Internet router is prepared to receive incoming data because it comes in response to a request from the LAN.

By contrast, unknown data arriving from the Internet, such as data from a SSH client somewhere else in the world, represents a security risk. For this data to be received, it must pass through ports that are normally blocked on LANs. As components of complete internet addresses, ports correspond to numbers that are assigned to services on networks. For example, the standard port for SSH is 22. To allow “unexpected” incoming SSH data to be forwarded to port 22 on the Raspberry PI, the router must be explicitly configured to enable forwarding for this port. The SSH server of the Raspberry Pi then listens to this port.

Figure 2 shows the FRITZ!Box port forwarding enable page for the SSH server on the Raspberry Pi. Remember to disable port forwarding when you no longer need it.
Along with the number of ports for which forwarding has been enabled, the public IP address of the FRITZ!Box is shown on the overview page

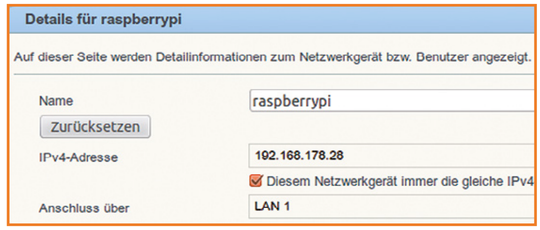


Figure 1.
Setting the IP address in the FRITZ!Box. Zuruckzetzen = Clear. Anschluss uber = Connect Via.

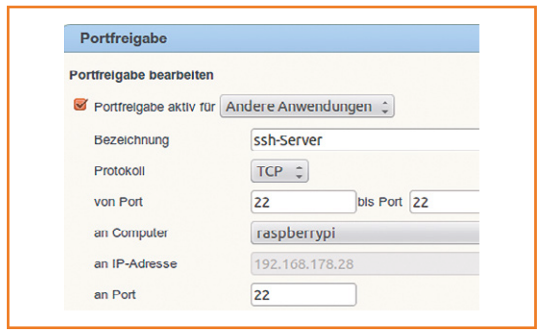


Figure 2.
Enabling port forwarding in the FRITZ!Box for the SSH server of the Raspberry Pi. Andere Anwendungen = Other Applications. Bezeichnung = Label. an = to.

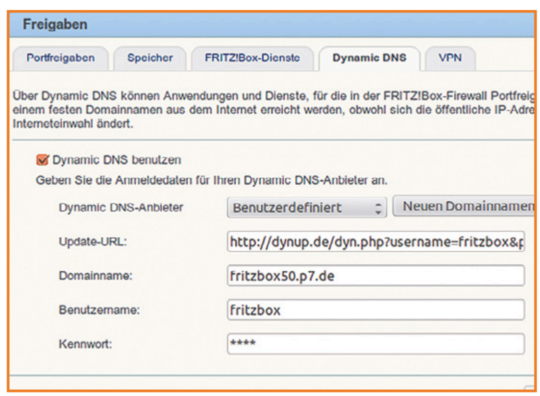


Figure 3.
DDNS configuration settings in the FRITZ!Box. Benutzername = User Name. Kennwort = Password.

of the router configuration. When an SSH server on the Internet issues a call with this IP address (with the remainder of the address as previously described), it gains access to the SSH server of the Raspberry Pi, The same call also works on the LAN to which the Raspberry Pi belongs because the FRITZ!Box sends it to the Internet and then receives it as the addressee.

The fly in the ointment here is that with a dial-up Internet connection, the IP address of the FRITZ!-Box changes almost every time a link is established (depending on the provider). Even a permanent connection does not prevent this, since the connection is normally broken every day by the provider and this results in the assignment of a new address.

Unfortunately, the IP addresses of routers are not recorded in the public Domain Name System

(DNS). The only remedy here is to use Dynamic DNS (DDNS). With this option the FRITZ!Box is assigned a fixed domain name. This name is first used to contact a DDNS server, which subsequently calls the FRITZ!Box with its current IP address. This is possible because the FRITZ!Box (assuming it is suitably configured) notifies the server for the DDNS service of every change to its IP address.

Here's how this works in practice: DDNS providers (such as the German *dyndns:free*) offer a portion of their services free of charge at a specific time. First you have to register. The DDNS access data is sent by e-mail. After this, configuring the FRITZ!Box is relatively easy (see **Figure 3**). First you have to enter the desired domain name for accessing the FRITZ!Box. This consists of the name you chose followed by the predefined part of the domain name. You also have to create a DDNS user account in the FRITZ!Box and enter the associated user name and password. Finally, you have to enable DDNS in the FRITZ!Box and enter the data necessary for logging in to the DDNS server.

Right after the data is accepted, the FRITZ!-Box tries to log in to the DDNS server. This can also be done under software control. The URL of *dyndns:free* is:

```
http://dynup.de/dyn.php?username=xxx&password=xxx&hostname=xxx.xxx.de
```

where the *xxx* terms are placeholders for the actual names.

To see whether the login was successful, check the *Online Monitor* page or the overview page. Now you can establish a connection from the Internet to the SSH server of the Raspberry Pi with the agreed name by entering the following command:

```
ssh pi@xxx.xxx.de
```

Here again the *xxx* terms are placeholders.

There are also SSH apps for smartphones. The program on the ATMega32, the shell and *picocom* expect to be served by a terminal. This means that the arrow keys as well as the Ctrl key, the Esc key and so on must be available. Hardly any alphanumeric keyboards on smartphones have these keys by default, so they must be specially configured or retrofitted. For example, the Android app *vSSH* comes with an extended keyboard.

Final remarks

Naturally, there's no reason for going to the trouble of setting up remote access to an ATmega32 program from the Internet unless it runs a program that does something genuinely useful, instead of the simple demo program.

The program *mmdemo1.c* (which can be downloaded free of charge from [1]) implements a serial user interface with four sample communication scenarios. It needs roughly 300 lines of code for this. It can also be extended to do other things if you maintain the structure of the program. One of its features is support for communication between a terminal and several microcontrollers over a low-impedance wired-AND bus, which also makes large distance between the bus nodes possible. This program is described in detail in the comments at the end of the code section.

Another important note: If you want to use this approach for remote control of real devices instead of just a couple of LEDs, you can increase the security by requiring a second password entry before allowing remote access to the hardware, in addition to the password required for SSH access to the Raspberry Pi. You shouldn't ignore security, as otherwise it's entirely possible for jokers and people with dubious intentions to do unpleasant things while remaining anonymous.

(130481-I)

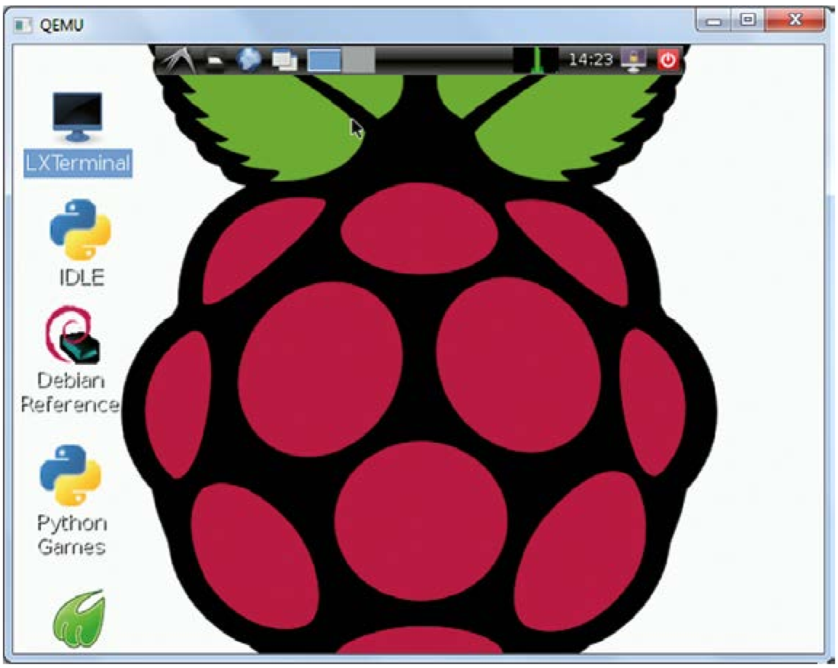
Web Link

[1] **Elektor web page for this project:** www.elektor-magazine.com/130481

Raspberry Pi Emulator

By **Bert van Dam**
(Netherlands)

You don't have a Raspberry Pi, but you are nevertheless curious how this popular platform works? With the aid of the open-source processor-emulator Qemu it is possible to emulate this wildly popular single-board-computer on a Windows PC.



Many Elektor readers will have become acquainted by now with the Raspberry Pi through the various Elektor.Post-articles distributed to Members every 14 days [1] or the Elektor book 'Explore the RPi in 45 Electronics Projects' [2]. For those of you who have not yet acquired an RPi but are nevertheless keen to play with this multifaceted system, we describe here the possibility of emulating the Raspberry Pi on a Windows-PC. Of course, this emulation will not be 100% correct because some hardware is not physically present, but this method will still give you a very good impression of the possibilities.
To make it easy for you, all the necessary software can be downloaded from the Elektor website. In the instructions described below, we assume you are using the SD card which is available with the Raspberry Pi book (also on [2]) as the Linux operating system that normally runs in the RPi. But you can also use any other Debian-based distribution for the RPi.

- Go to the Elektor website [3] and download the Qemu software package and the program DiskImager. Both programs are suitable for Microsoft Windows (tested with Win7 64-bit).
- Make a new folder, for example `c:\qemu` and unzip both software packages into this folder. You do not need to install the programs.
- Use the DiskImager program to make a copy of the SD card from the book. (When starting the program it is possible that Windows will ask whether you agree to allow this program to make changes to your computer. Answer with 'yes'. After that it is possible that a nasty error message appears, you can ignore that one; in spite of this the program will operate correctly.) Name the copy `rpi-bookbertvandam.img` and store this in the qemu folder.
- Start the program `fix.bat` in the qemu folder by double-clicking it. The emulation will now begin and stops with the final line reading: `root@(none):/#`
  This is the Linux prompt.
- Enter the following command:
  `nano /etc/ld.so.preload`
  This starts the text editor, which contains the following line (or something very similar):
  `/usr/lib/arm-linux-gnueabihf/libcofi_rpi.so`
  Insert the hash (or pound) symbol (#) in front of this line and subsequently press Ctrl-X. The press Y followed by the Enter key.
- The editor will now close again and you are back at the prompt: `root@(none):/#`
  Now enter the command `halt`, press Enter and wait until the prompt reappears (ignore the error message). Close the window using the small red cross at top right.
- Start the program `run.bat` in the qemu folder by double-clicking it. The program will now automatically load the copy of the SD card

and after a short time the famous Raspberry Pi screen will appear. The message 'failed' when loading the file system and error messages when loading libmod are normal. These are caused by certain hardware that is not available but is expected by the software (and which of course is present on a real RPi).

You can now experiment to your heart's content in the emulation window. It is even possible to go through the first few chapters of the 'Raspberry Pi' book. Of course you cannot make any of the electronics projects from the book, because Qemu only emulates the Raspberry Pi and not the electronics from the book.
When you use the mouse in the emulation window all its commands are used for the Raspberry Pi emulation. You can return the mouse to Windows by pressing the key-combination Ctrl-Alt.
The emulation program requires quite a bit of computing power from your PC, so it is possible that everything is running and reacting somewhat slower compared to a real Raspberry Pi.

From now on you can start the emulation using `run.bat` and stop it using the command `sudo shutdown -h now` in a terminal window or via the task bar (the red power symbol on the far right) and then 'Shutdown'. Wait until the message `system halted` appears before closing the window.

(130539-I)

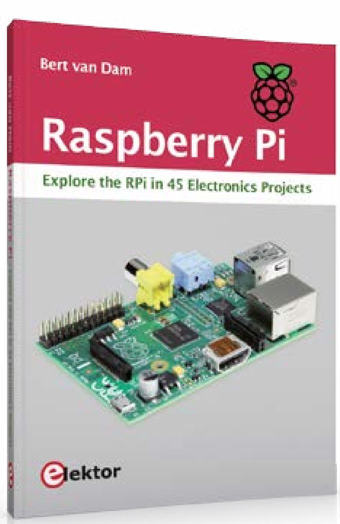
With thanks to xecdesign for the preload information.

Web Links

[1] www.elektor-magazine.com/post

[2] www.elektor.com/rpibook

[4] www.elektor.com/130419



Elektorize Your Scope Probe

By
Alfred Rosenkränzer
(Germany)

Measuring high-speed digital signals with probes is prone to produce errors. The ground connection of the probe has special significance. If the standard flying-lead-and-croc-clip is used, the scope images often show wild overshoots definitely not present in the signal. More expensive probes have a "ground spring" to establish a ground connection that's as short as possible. Such a ground spring is eligible for Elektorizing!

Measuring signals with an oscilloscope is daily bread and butter to an engineer either at work on in their spare time.
The output from a signal generator can generally be directly connected to a scope input using a coax cable with a BNC plug at either end. For low-frequency signals such as audio is isn't necessary to terminate the cable with its characteristic load. Higher frequency signals such as video or digital signals generally need a load termination at the end of the cable—ideally the scope will have a switchable load option built into its input stage or else use a 50-Ω feed-thru terminator.

Figure 1.
10-MHz square wave signal using coax and 50-Ω termination at the scope input (A); The same signal measured with a standard probe and a flying Earth lead (B).

Figure 2.
An Agilent probe with the Earth lead removed, using a ground spring.

In most cases for circuit debugging, a high impedance scope probe is used to trace a signal path through a circuit. Any part the circuit will 'see' the probe tip as a 10-MΩ impedance to ground. This is made up of a 9-MΩ series resistor in the probe body (in parallel with a capacitor of around 10 pF) and the 1-MΩ scope input impedance. Altogether these two impedances form a 10:1 voltage divider network so that the signal at the scope input is only one tenth of its actual value.

The signal is sampled with the tip of the probe and the Earth connection is a flying lead, usually around 10 cm (4 inches) long terminated in a crocodile clip. It is assumed that the scope probe has already been tweaked for optimal response using the scope test output signal.

When measuring fast digital pulses you notice that the signal shown on the scope changes depending on where you attach the Earth lead, particularly noticeable on overshoot at the signal edges. The suspicion is that the displayed waveform is not a true picture of what is happening in the circuit (**Figure 1**).
More expensive probes such as those made by Agilent or Tektronix provide a tip ground or ground spring option. This removes the long Earth lead and improves the waveform by allowing a

shorter Earth connection nearer to the point of measurement (**Figure 2**). The resulting waveform is shown in **Figure 3**.

The author hasn't seen this type of earthing feature on any of the standard low-cost probes, but luckily they do have an Earth collar just back from the probe tip which can be used for this purpose. Wind a length of bare copper wire around the collar a few times and twist the ends to make it grip the collar (**Figure 4**). For convenience this Earth connection can be temporarily soldered into the circuit under test. It costs practically nothing and if it falls off it only takes a minute to replace.

Bandwidth

These improvements will not alter the fact that the scope probe bandwidth usually limits high frequency measurements.

To check this out use a square wave generator and measure the rise or fall time on the scope display first using a BNC to BNC coax with a terminator load and then with the scope probe. If the rise time is slower with the probe then you can be sure the probe bandwidth is having an effect. Professional solutions include probes with an input impedance of 500 or 1000 Ω. These at first seem very low compared to standard 10 MΩ probes, but at high frequency the probe's input capacitance has a greater influence on signal loading. Also signals in this range tend to be in the 1 V_{pp} range and the ICs can easily drive the extra load.

The cost of such a probe is in the region of $100 Euro, which is rather a lot if it only gets occasional use. You can build a test probe with similar characteristics quite cheaply. First take a length of 50-Ω coax fitted with a BNC connector at one end (alternatively cut a 2 m (7 ft.) BNC to BNC cable in half to make two probes). Trim back the insulation from the cut end and solder a resistor to the center conductor. A small length of heat shrink sleeving will cover the joint and resistor. Tease out the shielding wire strands and solder to a short length of wire to make the Earth connection on the PCB. Another length of heat shrink over all the bare wires will finish the job (**Figure 5**). The value of resistor used here can be either 450 or 950 Ω. Together with the 50 Ω termination resistor at the scope input these resistor values produce a 10:1 and 20:1 voltage division of the measured signal. The non standard resistor values are available from component stockists. Failing that you can use values of 470 and 1000 Ω and accept that the voltage division ratios will be slightly out. If something breaks, it's a simple job to trim the cable back and make another.

Figure 6 shows a low-impedance scope probe from HP. In the white plastic shell is a swappable resistor. Connection to the scope is via an SMA connector and a length of SMA cable (the cost of this probe puts it in the 'Professional Use only' camp). Using an SMA connector, cable, a resistor, wire and some heat shrink sleeving it should be simple to produce something similar (providing you have some SMA-diameter cable and a BNC adapter).

(130112)

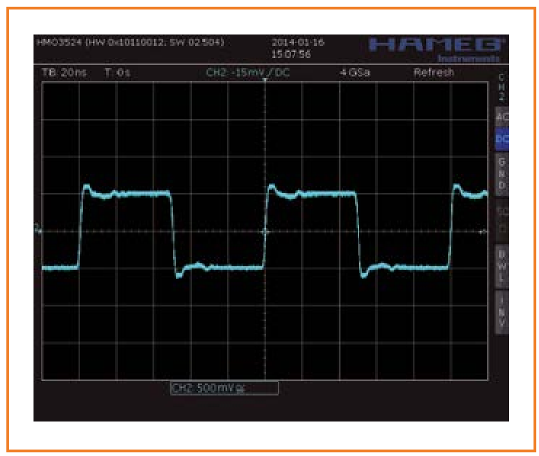


Figure 3.
The 10-MHz signal measured with a short Earth lead on the probe collar.

Figure 4.
A standard Hameg probe with DIY tip-ground modification. The Earth lead is removed.

Figure 5.
The DIY low-impedance probe.

Figure 6.
An old HP low-impedance probe.


electronics
engineering
embedded
DISCUSS
design tips
tutorial
software
mobile
contests
engage
engineering
tools
system
audio
business
networking
media
COMMUNITY
data
social media
talk
information
product news
projects
Want to talk to us directly?
Share your interests and opinions!
Check out our New Social Media Outlets for direct engagement!
CIRCUIT CELLAR / AUDIOXPRESS / ELEKTOR

Big Brother Be Gone

By
Clemens Valens
(Elektor.Labs)

The search function on Elektor.Labs has switched to DuckDuckGo, an Internet search engine claimed to provide real anonymous search. Our first impressions are very positive: fast, accurate and clean results without tracking. No more Big Brother following you around the Internet. Why don't you give it a try?

Collecting information is easy; making it accessible is difficult. Since its launch Elektor.Labs has collected quite a lot of useful information about electronics and projects, and obviously good tools are needed to exploit this data. Since Elektor.Labs is a laboratory we can experiment with search engines to find out which solution suits us best.
The Elektor.Labs website started out with the default search engine integrated in the website's 'Drupal' content management system. This search engine works fine as long as you know exactly what you are looking for. Punch in the search phrase and you will be rewarded with all the articles containing that exact phrase. Typo? Nothing found. Wildcards? Not allowed.
While investigating the rationale behind the absence of wildcards we discovered that there are (at least) two schools of thought for search engines: few accurate results *versus* many less-accurate results. Our website adhered to the first school whereas we preferred the second, which we felt would be more suited for people not too sure about the exact term they are looking for.
So we replaced the default search engine by Google's Site Search. Initially we got spectacularly precise results; we found everything we wanted and even some fans of the first school were impressed with the quality of the pages returned. OK, Google added some commercial links to the results, but the precision of the results outweighed this inconvenience.
However, as we were soon to discover, Google Site Search turned out to be too much of a good thing—it even presented pages we didn't know the existence of, like site-wide lists of new posts. At first we thought that that was all right, but quickly these listings started to outnumber the useful results. Deactivating these listings did not solve anything because Google had memorized them all and so we pulled the plug on Google Site Search.

Enter DuckDuckGo. This is a rather recent search engine that boasts anonymous search while trying to limit spam to a minimum. They offer a site search function as well, although not as neatly integrated as Google. When you click the Elektor.Labs search button you are taken to our DuckDuckGo site search page. Try it and be amazed by its lightning speed. There is still a trifling inconvenience though: make sure you do not delete "site:www.elektor-labs.com" from the search box.

(130461-I)

www.elektor-labs.com/node/3795

What's up with this cap?

By Thijs Beckers
(Elektor Labs)

Loyal readers will be aware that Elektor have been offering printed circuit boards, programmed microcontrollers, kits and modules *et cetera* for some considerable time as a complementary (not: complimentary) service to Elektor Magazine. This service we hope increases the chances of success when assembling your own circuit at home or in your workspace. While we cannot fly over and assist with your soldering, at least our semi-kits ensure that the correct components are available to those interested. One of my duties at Elektor Labs is to randomly check these semi-kits for completeness and make sure everything is in agreement with the published BOM when they arrive from our suppliers.

For our recent ADAU1701 Universal Audio DSP Board ([1]) we had one of our trusted suppliers do the parts picking for the semi-kit. When the kits arrived at our warehouse I randomly checked a couple for correctness and completeness. I then stumbled upon something very odd with the 100-nF (0.1 µF) capacitors. The picture shows two of them, each from a different side. Now I'd like to believe I'm not a complete knucklehead on electronic components, but this capacitor included in the kit had me flabbergasted. It was supposed to be a 100-nF capacitor, but as you can see in the photo, it has a "105" marking on one side and cheerfully "104" on the other side. Every 100-nF capacitor in every sample kit I checked showed the same 104 / 105 print as shown here, so it was not just a misprint on one of them, and it's a long time to April 1.

Trawling the net for a possible explanation didn't result in an answer, nor did any of my colleagues at here at Labs know what was going on here. According to our LCR meter (the fancy one we published about in the March, April and May 2013 editions [2]), the capacitor is indeed 100 nF. Your DSP kit should be okay.

So here's my call out to you, our well-informed reader and/or omniscient professional, to explain what's going on with this component. Is it just an ex-factory misprint? Or is there a method to the madness?

Send your insights to editor@elektor.com, subject: madcap.

(130460)

Web Links

[1] www.elektor-magazine.com/130232

[2] www.elektor-magazine.com/130093

Unwrapping RepRap

By **Patrick Wielders**
(Elektor Labs)

A couple of weeks ago Labs took delivery of RepRap 3D printing machine [1] from RS Components / Allied Electronics [2] to experiment with. Here were cover the unpacking and assembling of the mini manufacturing machine.

The first thing you come across after opening the box is a packing list, which should be checked thoroughly to make sure all items listed are present and intact. All parts that make up the kit are packed in transparent bags, and on every bag there's a content list, which is very useful.

The letter of introduction that comes with the kit includes a link to the online Assembly Manual [3]. Initially we looked for clues to a printable version of the manual, but it soon dawned upon us that there is no such thing. The disadvantage of this approach is that you need to have a computer with Internet access up and running while assembling the printer. But it didn't take long for us to see the benefits of the online-only manual: for example, the images are displayed at a higher resolution after you click on them, offering more detailed visual support.

The manual provides all the details needed for an easy step-by-step assembly procedure. It features lots of pictures, loads of useful tips and in general is well structured. We reckon anyone who ever played with Lego or built something from Ikea drawings can assemble this 3D printer. Check our time-lapse video of the steps to assembly at [4].

A nice touch of the RepRap concept is that all the parts printed by a 3D printer are available as open-source 3D designs and the links to their locations on the web are included. So, if you feel up to it, you can download the files and print (spare) parts yourself. This can be useful in case a part is worn out or when an improved version of the part becomes available. Imagine, downloading upgrades and (3D-)printing them yourself for the purpose of improving your own 3D-printer... ain't that a blast?

(130562)

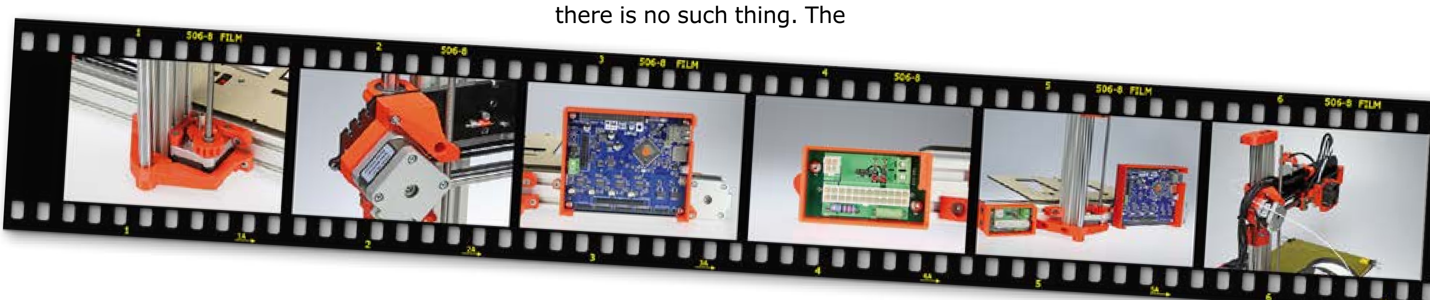
Web Links

[1] www.reprap.org

[2] www.rs-components.com

[3] http://reprappro.com/documentation

[4] http://youtu.be/eLEMP8VmtM8



DesignSpark Tips & Tricks

Day #9: Custom Outlines Continued

By
Neil Gruending
(Canada)

Let's finish our board outline that we started last time.

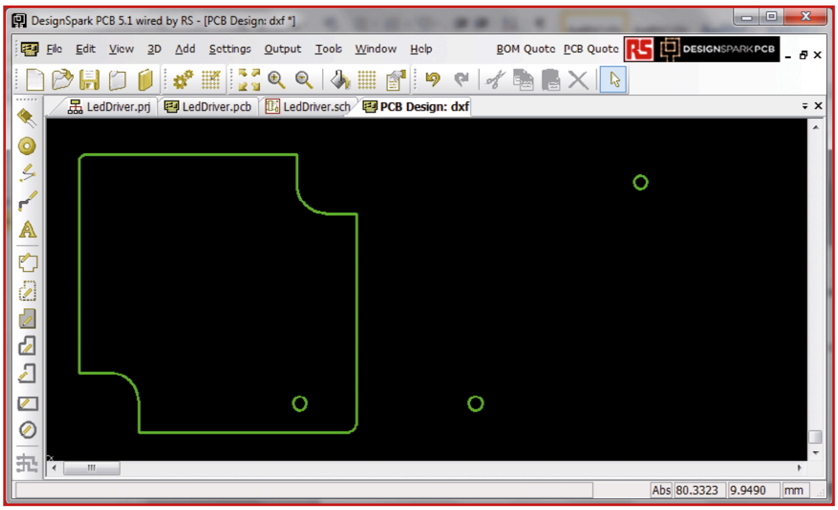
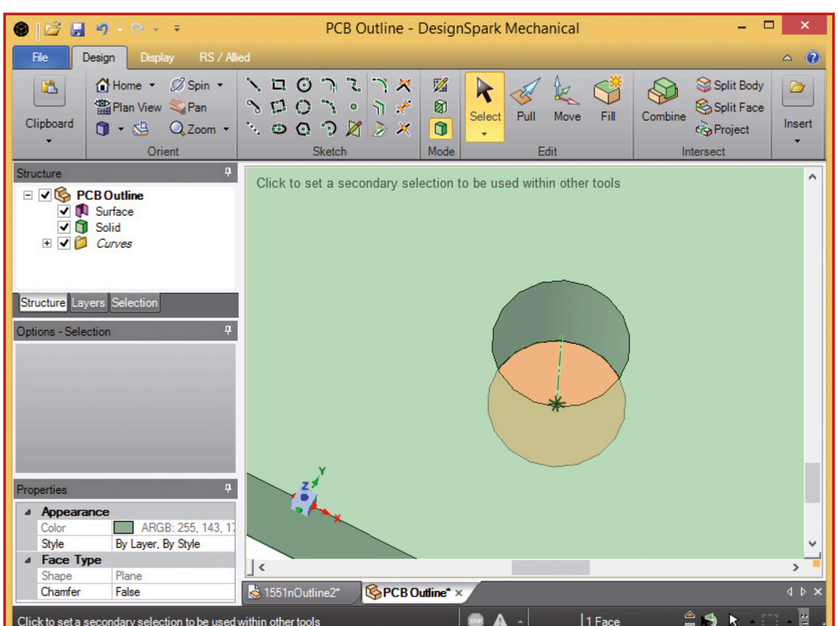


Figure 1. Failed DesignSpark PCB Import.

Figure 2. Deleting the plane covering the mounting holes.



Last time we started a PCB board outline for a Hammond 1551N enclosure. We started by importing the enclosed STEP model into DesignSpark Mechanical and then we modeled the circuit board to make sure it would fit. Today we are going to finish importing the board into DesignSpark PCB so that we can the board ready for components.

Fixing the import

Figure 1 shows where we left off last time. The PCB outline is correct but the mounting holes are obviously wrong. I tried some other packages and they correctly imported the DXF file we created last time. Exporting the DXF from those packages would also import properly into DesignSpark PCB so something in our DXF file must be confusing it.

After some experimentation I found that there are a couple of errors with the model I had created in DesignSpark Mechanical. The first problem is that the mounting holes weren't really holes in the board model because they are covered by a thin plane. It was hard to see until I rotated the model a bit and saw that the bottom of the holes were shaded instead of being white. The second problem is that DesignSpark PCB doesn't think the board shape is a closed shape since it raises a warning after the import. Moving the corners showed that two of them aren't connected properly because the unconnected lines won't resize properly.

So let's start by fixing the mounting holes in the DesignSpark Mechanical PCB model. The first step is to use the pull tool to pull the PCB outline 1.6 mm upwards from the sketch plane which creates a solid PCB object. Then select the circles from the original sketch plane and delete them like in **Figure 2**.

The trick is make sure that you select the circle covering the mounting hole has a solid and not just circle outline otherwise you won't be able to

delete it properly. You will know when hole is clear because you will see the white background show through the hole. Once you've cleared both holes right click on "Curves" heading in the PCB outline structure and select delete to remove the extra lines that we don't need anymore.
Before you export the new DXF you have to select the top view of the PCB (Design→Orient→Top). Otherwise DesignSpark Mechanical will export the DXF as a 3D perspective of the PCB which won't work. **Figure 3** shows what the corrected board looks like in DesignSpark PCB with the new DXF. It still raises the same import warning but that's due to the mounting holes—not a problem since we'll be deleting them later anyways.

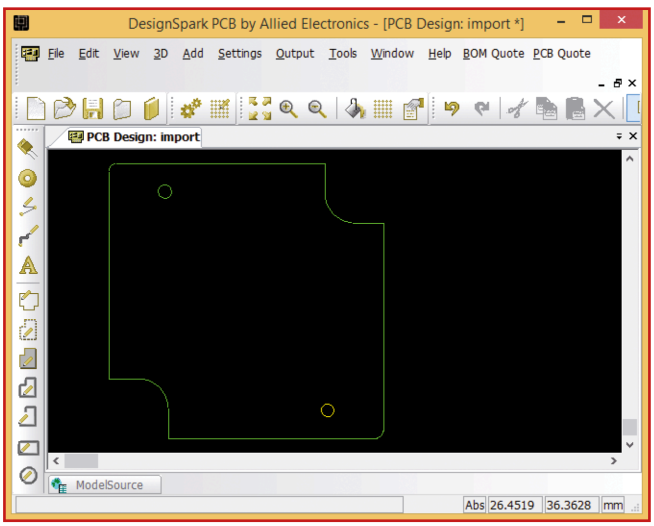


Figure 3. Fixed board import.

Adding the mounting holes

I usually like to set up my mounting holes so that they're unplated and without any copper pads under the screw head to avoid tearing the copper off of the board when tightening the screw. These types of holes are usually made as a library component because they require copper keep-outs on the outer layers. DesignSpark PCB can do this but it's difficult because you can't add keepout elements to a DesignSpark PCB library component. Instead you would have to manually add them to every hole you make. Today we'll just use a plated hole since they're much easier to use in DesignSpark. You can make them unplated as well but be careful that the copper pads don't tear off the PCB.

Start by creating a new PCB symbol for our mounting hole. We're going to use ANSI #2 screws so add a pad with a 2.5-mm hole and a 5.5-mm pad which corresponds to the recommended size [1] for a pan head screw and nut. For all practical purposes these sizes suit the metric M2 screw also. Once you finished editing the pad save it your library. Next

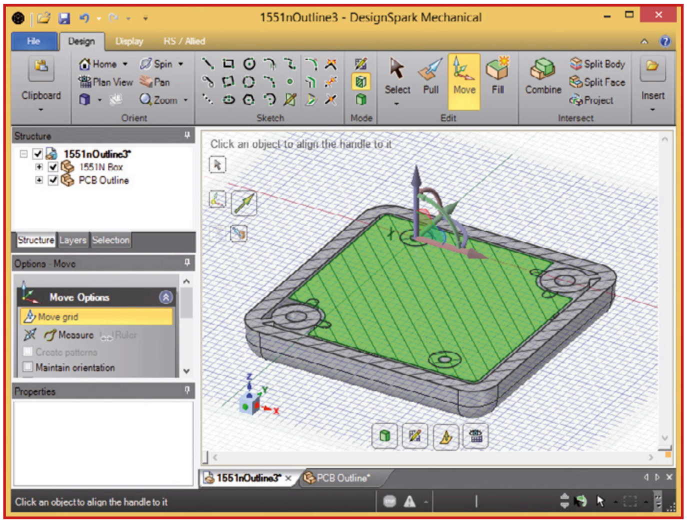


Figure 4. Creating a new cross section with keepout information.

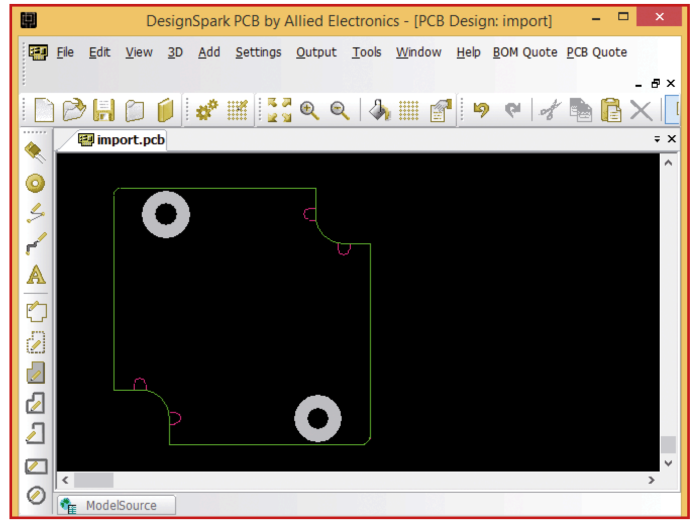


Figure 5. PCB with mounting holes and keepouts.

you will need to create a PCB only component for the mounting hole. You do that by adding a new component to the library like usual except that you uncheck the "Schematic Symbol" box in the "New Component" window. Alternatively you could add it to the PCB technology file instead but if you use multiple technology files then you would have to copy to each technology file. Using a library means there's only one copy of the hole.
Now we can add the two mounting holes to the design. I manually placed them on the PCB and then adjusted them to the correct locations by manually entering the hole co-ordinates to be the same as the imported hole co-ordinates. You can get the co-ordinates of the imported holes by right clicking on them, selecting "Properties" and noting the center co-ordinate shown. Then you can set the mounting hole component co-ordinates by right clicking them, selecting "Type Coordinate". After the new components are placed properly you can delete the old imported holes.

Adding the keepout areas

The last step today is to add the component keepout areas for the mounting bosses. You do this in DesignSpark PCB by adding the keepout information to a documentation layer and making sure we don't violate the keepout while working on the design. DesignSpark cannot check for violations via a DRC.
First we'll need to use DesignSpark Mechanical to create a cross section of where the board meets the enclosure mounting bosses just like in **Figure 4**. Then we can create a new component with the board outline, mounting holes and mounting bosses. There's no point in making it a 3D shape this time so just export a DXF of the 2D shape even though there will be some errors with the mounting holes after the import. Make sure to select the bottom documentation layer in the "To Design Layer" field when importing the DXF into DesignSpark PCB.
After a little bit of editing the final result in DesignSpark PCB will look like **Figure 5**. The mounting holes now have clearance for the screw heads and the keepout areas are visible on the bottom documentation layer which is shown in pink.

Conclusion

Today we fixed our import issues and finished adding the necessary mechanical information to our PCB. Next time we'll import it back into DesignSpark Mechanical to make sure we didn't make any errors.

(130463)

Web Link

[1] http://blogs.mentor.com/tom-hausherr/blog/tag/pcb-mounting-holes/

Nixie Displays

Weird Component #4

By
Neil Gruending
(Canada)

Nixie displays are great for retro style clocks but they were actually developed back in the vacuum tube era. For a long time they were one of the only ways to display 'sort of' alphanumeric characters. They've since been replaced by seven-segment LED displays but they're still fascinating. Nixie displays are a lot of fun but they operate at around 170 V_{DC} which can be lethal. Play safe and follow high voltage safety rules [1].

A modern seven-segment display uses seven LED segments that can be used to represent to represent a character. It makes a nice and compact display but the downside is that a decoder (hardware or software) is required to convert a character to its 7-segment representation. This isn't a problem with modern electronics but back in the 1950's and 60's this was a lot harder. Nixie displays solve this problem by having one control line per digit. So for example a numeric display would have one control line for each number between 0 and 9.

Internally a nixie display tube has a bunch of stacked elements. The topmost one is the anode or where the positive voltage is applied. Below that is a plate for each character which is then brought out to their respective control lines. This construction gives the nixies their nostalgic character shapes and friendly orange glow.

Nixes operate on the same principals as a neon bulb. You apply a large enough DC voltage to the nixie anode and cathode and once you reach the neon breakdown voltage the currently selected character will illuminate. The trick is that you have to either use a current limited anode power supply or an anode supply with a series resistor because a nixie tube will act like a zener once it starts to conduct. Otherwise it will fail. The required DC voltage to make a nixie conduct is quite high (usually over 150 V) which makes the numerical device a little tougher to interface to modern digital logic but there are lots of examples available on the Net and in Elektor magazine, like the *Sputnik Time Machine* (**Figure 1**) [2] and various thermometers [3],[4].

The anode current is a critical parameter for nixie tubes. If it's too high or too low it can drastically reduce the tube operating lifetime. It also controls the brightness and focus of each digit. But with the correct drive circuitry a long life type tube can last up to 200,000 hours or over 20 years.

Figure 2 shows an illuminated Russian IN-12A nixie display tube which is common in clocks and is also still used in some nuclear power plants. I used a 170 V_{DC} power supply and a 40-kΩ resistor connected to the anode and I grounded the pin to illuminate the number 7. In it you can see the anode grid at the top and then some of the numbers that are stacked on top of the seven but not illuminated.

It's amazing that it's still possible to find brand new nixie tubes (New Old Stock; NOS) from various online sources. And the growing clock enthusiast community is making it even easier to try and use nixie tubes.

(130536)

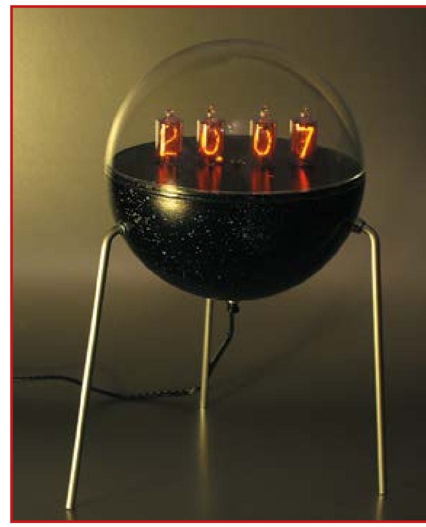


Figure 1. The Elektor Sputnik clock published in, ermm, 1957+50 = 2007.

Figure 2. Illuminated IN-12A nixie display.

Web Links

[1] www.repairfaq.org/REPAIR/F_safety.html

[2] Sputnik Time Machine, Elektor January 2007, www.elektor-magazine.com/050018

[3] Nixie Tube Thermometer, Elektor January 2011, www.elektor-magazine.com/090784

[4] Nixie Thermometer/Hygrometer, Elektor June 2012, www.elektor-magazine.com/110321

90-°C Oven Controlled Xtal Oscillators

Precision Devices Inc., a member of the Avrio Technology Group LLC, has announced the high temperature (to +90°C) VN01 and VN03 series of Oven Controlled Crystal Oscillators (OCXO). This product is ideal for applications like satellite communications and a variety of military systems where extremely high temperature operating environments are the normal.

"Many harsh environment applications require higher operating temperatures than the typical +70 or 80 °C products generally available in the market. Moving to 90°C is not a trivial thing," according to Barry Arneson, Director, Oscillator Engineering. "Due to the nature of how an SC cut crystal functions in conjunction with the oscillator, this is a significant step forward."

www.pdixtal.com (130448-III)

Electronic Component Selector Updated

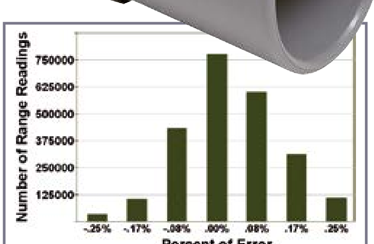
Würth Elektronik eiSos have released a revised and expanded version of their free Component Selector. Six new modules have been added to the four existing ones, making this selection tool one of the world's most comprehensive component databases to be provided by a single manufacturer. In the field of EMC components, cable ferrites and common mode chokes are now available in addition to PCB ferrites. These modules allow users to analyze and compare technical values and impedance curves for all products. Within the component characteristic curve, filters help users find the right components quickly and easily. In the power inductors module, the right inductor for non-insulated DC/DC converters can easily be determined. The Component Selector also calculates wire and core losses and, based on these losses, calculates the temperature increase of the component. In the field of power supplies, there is also a module to help quickly and easily configure flexible transformers for isolated DC/DC converters to suit the relevant input and output values. For use in active PFC circuits, there is now also a module for determining the right PFC choke.

The most significant extension of the software was for signal and communications products. As well as the existing RF inductor module for selecting components according to frequency, for adjusting antenna for example, three further modules have now been introduced. For signal and backlight LEDs, by entering the resistance and input voltage values for every freely definable forward current, the expected brightness of the LED can be displayed. A particularly practical addition is the module for discrete LAN transformers or integrated RJ45 LAN transformers, which enables users to search and filter components by the internal connections or land pad. This means that the right solutions for existing products can be found in no time at all.

Users can also save and store existing settings for component selection. Within all the modules, the relevant data sheet can also be consulted online and S-parameters can be downloaded. The integrated online update function ensures that the product database is kept up to date at all times. Users without an internet connection can get updates from the manufacturer's website at any time and install them manually. The Component Selector is designed to be self-explanatory and easy to get to grips with.

Elektor readers can install the free Component Selector by downloading the software from the link below.

www.we-online.de/component-selector (130448-VI)



High Accuracy Snow Depth Sensor

The newly released MaxBotix snow depth sensors are rated to operate in temperatures between –40C° to +65 °C with a maximum range of 5000 mm. Our sensors have been proven and optimized, yielding design solutions that previously hindered competing sensors, in a real world snow environment tested at Maxbotix'facility located in the heart of Minnesota. MTBF of 200,000+ hours and the low power consumption easily allow the sensor to operate in battery based systems for extended periods of time (or solar powered systems indefinitely) easing maintenance requirements and allowing for remote installations. At a low cost of $149.99 (MSRP) the use of these new snow sensors include improved performance during heavy snowfall conditions, responsive accurate temperature compensation, and continued performance during high wind conditions. The sensors are equipped to apply to any of these interface controls: pulsewidth, analog voltage, serial data - RS232 (MB7354) or TTL (MB7374).

www.maxbotix.com/snow (130448-IV)

PicoScope oscilloscopes now run on Linux

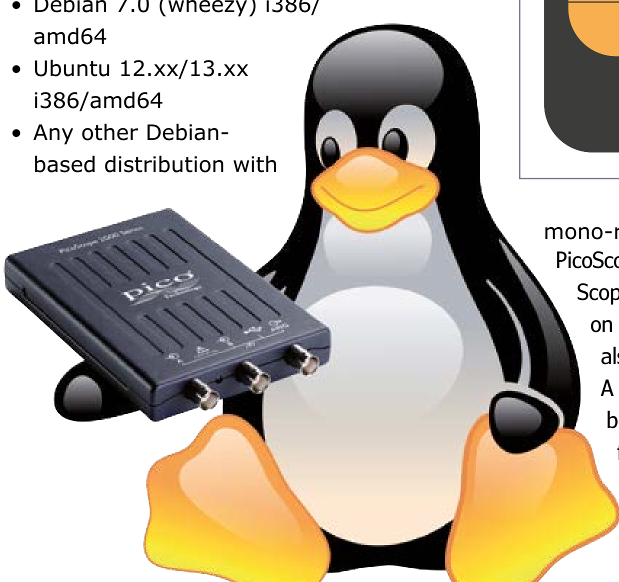
Pico Technology has just released a beta version of the PicoScope® 6 oscilloscope software for Linux. This is in response to requests from a large number of customers in the scientific and educational fields where Linux is widely used.

PicoScope 6 is a powerful application that converts your PC into an oscilloscope, FFT spectrum analyzer and measuring device. On-device buffering, using deep memory on some devices, ensures that the display is updated frequently and smoothly even on long timebases. The most important features from PicoScope for Windows are included—scope, spectrum and persistence modes; interactive zoom; simple, delayed and advanced triggers; automatic measurements; and signal generator control—and we are working on adding more. You can save captures for off-line analysis, share them with other PicoScope for Windows and PicoScope for Linux users, or export them in text, CSV and Mathworks MATLAB 4 formats.

The only additional hardware you need is a compact, economical USB oscilloscope from the PicoScope range. PicoScope oscilloscopes are available with bandwidths up to 1 GHz, up to 4 input channels, hardware vertical resolutions up to 16 bits, sampling rates up to 5 GS/s, buffer sizes up to 2 GS, and built-in signal generators. Other features available on some models include flexible hardware resolution, switchable bandwidth limiters, switchable high-impedance and 50 ohm inputs, and differential inputs.

The new software is packaged for easy installation on the following distributions:

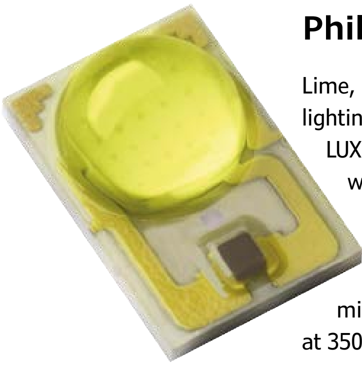
- Debian 7.0 (wheezy) i386/amd64
- Ubuntu 12.xx/13.xx i386/amd64
- Any other Debian-based distribution with mono-runtime >= 2.10.8.1



PicoScope includes drivers for all current scopes from the PicoScope 2000 Series to the 6000 Series and we are working on adding support for some older devices. The drivers can also be used on their own to create custom applications.

A number of customers are already using the new beta and contributing to a lively discussion on the Pico forum at http://www.picotech.com/support/.

www.picotech.com/linux.html (130458-I)



Philips LUXEON Lime LEDs break 200 lm/W Barrier

Lime, the newest addition to the widely renowned LUXEON color portfolio of LEDs from Philips Lumileds, enables lighting designers to take the next step in delivering the highest quality, tunable white light in bulbs and fixtures. LUXEON Rebel ES Lime is the proprietary LED technology in the revolutionary Philips hue bulb, where it combines with LUXEON Rebel Red-Orange and Rebel Royal Blue emitters to deliver over 16 million color options— all controlled from an iOS device. Philips hue can use color tunable Light Recipes to help set mood and energy level in the home, office, retail, classroom and hospital environments.

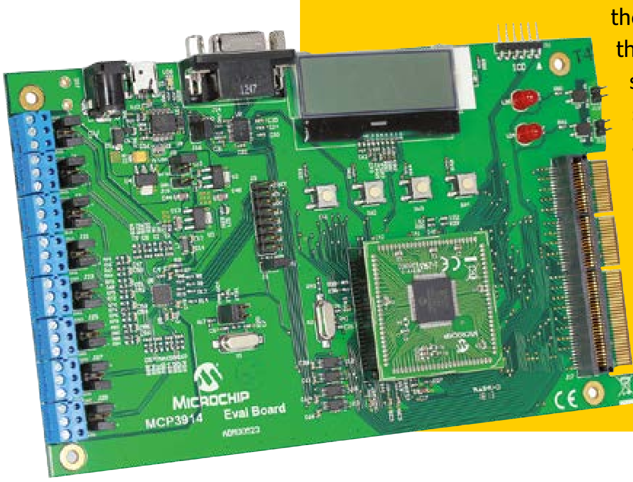
Lime is the highest efficacy LUXEON LED manufactured to date. Therefore it enables highly efficient color mixing by providing a convenient above-blackbody color point with optimal standalone efficiency of 200 lm/W at 350 mA and 85°C. The spectral output of Lime is closely aligned with the wavelength that human eye cones are most sensitive to, 555 nm. In addition to LUXEON Rebel ES, the Lime technology is offered in the LUXEON Z format, an undomed, 2.2 mm^2 LED that is 75% smaller than most high power LEDs. In spotlight and downlight applications, the LUXEON Z enables tighter packing density and better color mixing control. The LUXEON Z Lime can be combined with Red and Blue LEDs to achieve a broad spectrum of saturated colors. Alternatively, tunable white light with high efficacy can be achieved from 1800-6500K along the blackbody curve.

www.philipslumileds.com/LUXEONRebelColors www.philipslumileds.com/LUXEONZ (130458-III)

Microchip Analog Front Ends Offer High Accuracy for 3-Phase Smart Meters

Microchip Technology Inc. Announce their next-generation family of energy-measurement Analog Front Ends (AFEs) with industry-leading accuracy. The MCP3913 and MCP3914 integrate six and eight 24-bit, delta-sigma Analog-to-Digital Converters (ADCs), respectively, with 94.5 dB SINAD, –106.5 dB THD and 112 dB SFDR for high-accuracy signal acquisition and higher-performing end products. The MCP3914's two extra ADCs enable the monitoring of more sensors with one chip, lowering cost and size. Additionally, the programmable data rate of up to 125 ksps with low-power modes allows designers to scale down for better power consumption or to use higher data rates for advanced signal analysis, such as calculating harmonic content. These AFEs also feature a CRC-16 checksum and register-map lock, for increased robustness.

As the energy-metering infrastructure is being upgraded worldwide, designers are demanding increased AFE accuracy, performance and flexibility to develop the latest generation of smart meters. These features are also required by the designers of advanced power-monitoring systems for applications such as server power supplies and power distribution units, uninterruptible power supplies, smart power strips and data-acquisition products in the industrial and commercial markets. Microchip's new AFEs improve application performance with their industry-leading accuracy, while providing the flexibility to adjust the data rate to optimize each application's rate of performance versus power consumption.



Microchip also announced two new tools to aid in the development of energy systems using these AFEs. The MCP3913 Evaluation Board (part # ADM00522) and MCP3914 Evaluation Board (part # ADM00523) can each be purchased today for $99.99.

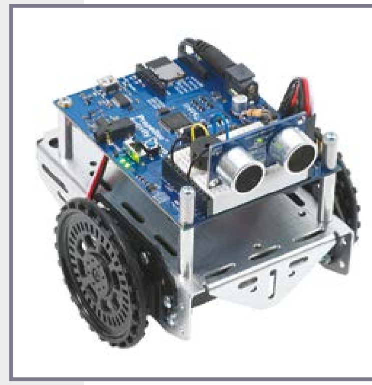
www.microchip.com/get/EUJ9 (130458-IV)

New Items from Parallax

ActivityBot (# 32500) (rrp: $199.00)

Learn real-world engineering skills with the friendly, capable, and peppy ActivityBot. It's a great option for first-time robot-builders, as well as for an intro to technology and engineering courses in high schools and colleges.

Step-by-step web tutorials take you through programming its multicore Propeller chip in C, wiring circuits on a breadboard, and building sensor systems so your robot can navigate on its own. Following the checkmarks gets you to the fun fast, with optional links for added learning.



Key Features:

- Easy to program in C on Windows, Mac, or Linux with the SimpleIDE software and custom Simple Libraries
- Multicore Propeller control board makes it quick to integrate sensors, motors, and more
- High-speed servo motors with optical encoders provide fast, consistent maneuvering
- Breadboard and 3-pin headers let you experiment with common electronics parts, no soldering or proprietary connectors required
- Component kit makes navigation systems that use touch, visible light, infrared light and ultrasonic sensors
- Built-in SD card slot and microSD card are ready for data-logging and file storage

Y-6 Multicopter (# 80100) (rrp: $799.00)

The Elev-8 Y-6 Multicopter is a flying robotics platform that is lifted and propelled by six fixed pitch rotors. Unlike other 'Tri-copter' platforms, the Y-6 has a motor mounted on the top and bottom of each boom. This 'double-stack' of the motors gives you added payload capabilities, and makes your system redundant in case of a sudden motor loss, making the platform safer to fly. Aircraft stabilization is electronically controlled by the HoverFly board with Parallax's Propeller multicore microprocessor. The benefit to this system is a stable platform with no mechanical linkages, for a small, maneuverable, and agile aircraft.

Key Features:

- Open underbody for accessory and camera mounting
- Redundant flight platform enabled by the use of two motors per boom arm
- 4 lbs. payload capacity
- Open source design files available for you to download and use
- LED strips wrap horizontally around the motor booms giving you precise orientation markers
- Uses common 4-40 hardware that can be found in most local hardware stores
- Modular frame allows you to repair and modify your platform quickly and easily

Gimbal Joystick (# 27808) (rrp $24.99)

The Quad-Bearing Gimbal Joystick is a high-quality, two-axis human input device. It is useful for sending continuously-variable settings, such as position, direction, or throttle values to a connected microcontroller. The joystick interfaces readily with analog-to-digital converters as well as simple RC timing circuits. It is mechanically adjustable for a customized action and feel. The included adapter kit provides a connection option to breadboards or thru-hole boards with 0.1" spacing (simple soldering required).

Key Features:

- 5 k-ohm linear-taper potentiometer on each axis provides even resistance transitions without skips
- Ball-bearing action on both axes make for silky-smooth, jitter-free operation
- Selectable spring return-to-center on vertical axis for rudder-type or throttle-type inputs
- Selectable detents and/or friction on vertical axis to hold joystick in place when not being manually held

www.parallax.com /[item #]   (130448-II)

Atwater Kent Model 30 Radio (1926)

By
Gerard Fonte
(USA)

In the mid 1960's I was a teenager and getting interested in electronics. I had just read about rebuilding old radio receivers (*Popular Electronics* May 1964) when I saw a mammoth Silvertone console radio on the curb, just waiting to be collected by the "sanitation engineers". Unfortunately I couldn't ask the school-bus driver to stop. I probably couldn't have gotten it on the bus anyway. It was about four feet tall (1.3 m), two feet wide (0.6 m) and weighed at least 50 pounds (20 kg). But, being a resourceful student, I called my parents from school and begged and pleaded for them to pick it up for me before someone else did or it got trashed. I was well rewarded for my histrionics.

With some help from a friend, I was able to replace the shorted filter capacitor that was making the regulator tube's plates get red hot. (For those unfamiliar with vacuum tubes, or "valves", only the filaments were supposed to glow.) The radio had fantastic tone and volume. But better still, it had several shortwave bands—very important for a budding engineer in those times. With such a fantastically successful start, I continued collecting old radios and suchlike.

Buying Blind

Some time later I went to my first auction and there was a large console radio that was in very sad shape. It had been stored in the barn for many years and the wood was falling apart and the back was closed up. I couldn't see any of the electronics. The bidding started at two dollars. Not knowing any better, I bid two dollars. Nobody else said anything. So, I got it for two dollars. One bid and one auction win! Pretty good for my first time.
The thing was nearly as big as me and as I was dragging it to the car, I realized that something was loose inside. When I got it to the car I pulled off the backing. Inside was a pristine Atwater Kent Model 30 radio. I don't know what it would have auctioned for, but I am pretty sure it would have been a lot more than two dollars—which was about all I could afford.

AK-30 Basics

The Atwater Kent model 30 was built around 1926. It's a six-tube, Tuned Radio Frequency (TRF) type of receiver. What that means is that is pretty much an amplified crystal radio. Looking at the redrawn schematic, **Figure 1** [1] (original also available, *Ed*.), there are three stages of tuned RF amplifiers, followed by a "grid-leak" detector, followed by two Audio Frequency (AF) amplifiers. This makes six stages with one tube for each stage.

The tuning of the three RF amplifiers is nothing more than a parallel L-C (inductor/capacitor) network (exactly like a crystal radio). The three air-variable capacitors were adjusted for tuning. Putting three identical tuned amplifiers in series increases gain (obviously) and selectivity (i.e. the ability to separate signals close in frequency). Each tuning stage narrows the passband considerably. With three stages, the selectivity is pretty good. Of course, there weren't all that many broadcasters to interfere with each other at that time. This also meant that each capacitor had to be properly adjusted to tune in a station. The Model 30 was the first Atwater Kent to use only one dial to adjust all three simultaneously. The capacitors were connected with pulleys and a metal band to accomplish this (more on this later). There was an RF gain control that varied the filament voltage to the three RF amplifiers.

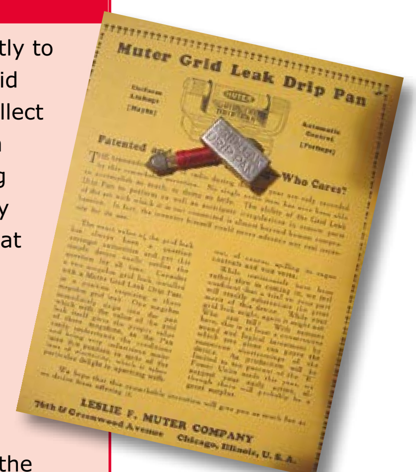
The grid leak detector (**inset**) was a very simple and cheap way to demodulate the RF signal. The procedure was to place a small bias on the grid by connecting a high value resistor to ground. The resistor in the model 30 was about 2 megohms. The effect is to change the operating point of the tube to only amplify the negative half of the signal. In this way there was detection as well as amplification. And while this did work, there were many things that affected the performance. This included signal strength, tube degeneration and

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

Leakage! Drip-drip!

The real grid leak resistor lead directly to the development of the fictitious "grid leak (drip) pan" that was used to collect all of these electrons and keep them from falling on the floor and creating a mess. Like the left-handed monkey wrench and the board-stretcher, great enjoyment was had by all (except the victim) in the futile search by the new-hire in the radio repair shop. There are rumors that these pans did exist and that some unscrupulous souls would actually sell these grid leak pans to the unsuspecting public. There are currently "audiophile" AC power cords that sell for \$1699.99. Perhaps, more reasonably, parts vendors and distributors may have given pans away as humorous promotional items.



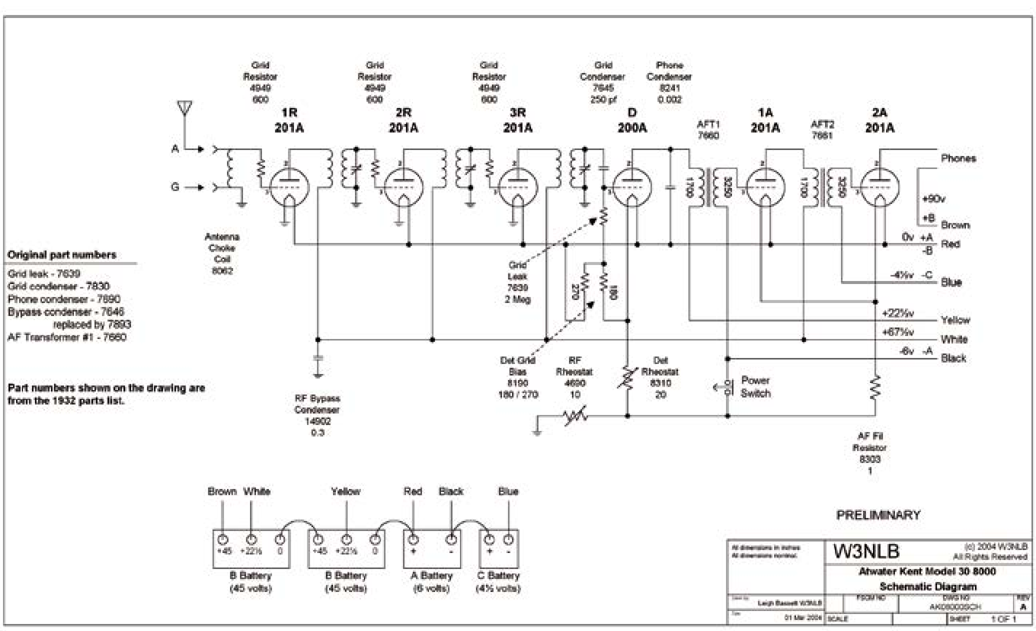


Figure 1. Redrawn AK-30 schematic, reproduced with permission from Leigh, W3NLB and Atwaterkent.info.

Figure 2.
View of my AK-30, hinged lid opened.

grid-resistor variation (these resistors changed with temperature and humidity).

After the leaky-grid stage came two stages of audio amplification. The final result was a signal that could drive headphones well or a loud-speaker poorly. Of course the loud-speakers of that era were really not much more than a headphone attached to an acoustic horn. The volume control varied the voltage on the filament of the detector rather than the input voltage to the AF amplifier stages (as is normally done today).

My AK-30

As you can see from the pictures in **Figures 2 and 3**, the case was nearly perfect and all the tubes were in place. However, there were some problems. The most significant was that the metal pulleys connecting the variable capacitors were crumbling (**Figure 4;** thanks Allen [2]). I had never seen (and still haven't) anything like this before. I suspect that it was due to metal fatigue. Especially since the metal appeared to be sintered rather than cast or machined. (Sintered metal starts as a powder that is pressed and heated into its final shape.) Cycling between a hundred degrees in the summer and zero degrees in the winter (40 °C to –20 °C) could certainly do that.

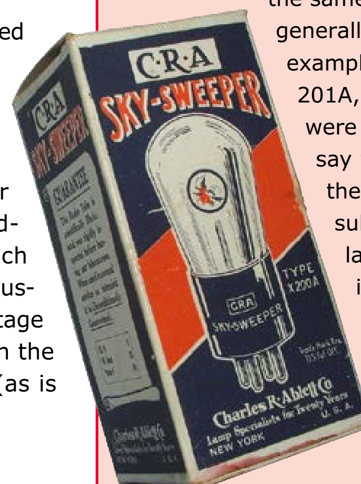
The second problem was that one of the tubes had an open filament. This is not a major problem today. It's easy to go to eBay or to one of the many antique radio sites on the internet to find a replacement. However, when I got this the internet didn't exist. Trying to locate a tube

Figure 3.
View of the AK-30, radio chassis pulled out.

Tubes

You can't really understand the radios of the era without understanding a bit about the critically important vacuum tubes. First, they were expensive. In the early 1920's they could cost as much as $9.00 each which is about $120.00 today. By 1930 the cost dropped to about $1.50 ($20.00 today). An early radio could rival a Model T Ford in price. A mid-20's radio like the Model 30 was a significant financial investment.

There were variations in tube manufacturing but eventually most producers settled on the four-long-pin base. And any tube that had the same last two digits was generally interchangeable. For example the UX-201, UX-201A, UX-301, UX-301A, etc. were reasonably similar. I say "reasonably" because the filament current was substantially less in the later tubes. This was very important since these radios operated from batteries. The early five-volt filaments used a full amp of current. The later versions reduced this to a quarter amp. With six tubes,



power for the filaments usually came from a 6-volt automobile battery. Filaments were wired in parallel, so there was no problem with substituting different tubes which had different filament ratings.
Tubes had to be devoid of oxygen. Otherwise the filament would oxidize (burn!) and fail. There were two general methods for accomplishing this. The first was to put a vacuum into the tube. Unfortunately, vacuums of the day weren't all that good. So, in order to eliminate the remaining oxygen a "getter" was inserted into the envelope. The getter was a chemical that reacted and bound up all the free oxygen. Getters caused the discoloration on the inside of the tube envelope. Different getters created different colors. Generally the color was silver-black. But some were bright blue and others were iridescent.
The second method was to place an inert gas into the tube—typically argon. This eliminated the getter and the glass envelope is perfectly clear. Of course, the problem with argon is that it breaks down at high voltage and glows a nice blue (very similar to the classic neon lamp). For some reason, detector tubes (UX-200A) usually used argon and RF (UX-201A) tubes used getters. Perhaps because detectors used lower voltages and there was less chance of breakdown/ionization.

that hadn't been manufactured for decades was not a trivial task.

The last problem wasn't really a problem. The two AF tubes were not the ones specified. They were apparently substitutes. All of the tubes of that time were simple triodes and many were interchangeable. They could amplify by a factor of about 8. (The minimum gain of a 2N2222 transistor is 50.) Instead of the two UX-201A tubes, there was a UX-200A and an SX-112A tube (**Figure 5**). The SX-112A was a newer tube designed for audio output. It could provide a whopping 200 milliwatts of power.

Batteries

The Model 30 needs five direct voltages: 90, 67.5, 22.5, 6 and –4.5. These are supplied by batteries. The 90 volts is used to drive the headphones. The 67.5 volts goes to the plates of the RF amplifiers and the first AF amplifier. The detector tube uses the 22.5 volts. The 6 volts powers the filaments. The –4.5 volts biases the final AF amplifier. (It draws no appreciable current and can last for years.)

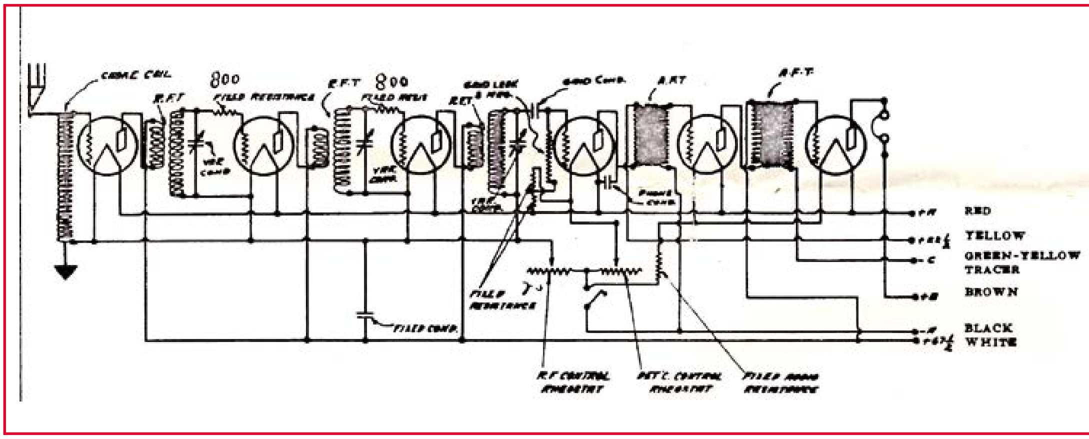
Obviously, batteries had to be replaced on a regular basis. So, the operating costs of an early radio were significant. It wasn't until a few years later that AC powered sets became widely available. And that was because of the development of rectifier tubes that could handle the power requirements.

Construction Details

There are some interesting construction details surrounding the RF coils. The back of the case is routed out in three places by about a quarter of an inch to provide clearance space for the

Figure 4.
Disintegrated pulleys: photograph reproduced with permission of Allen Wooten, WD4EUI.

Figure 5. Tube complement in my AK-30.



coils. **Figure 6** shows the chassis removed from the case. This seems like a design error. I would think that the case would normally be designed to accommodate the coils to begin with.
The second detail is that the three coils are mounted at right angles to each other (**Figure 7**). This is done to reduce inductive coupling between them. Without this, there would probably be leakage that would feedback and initiate oscillation. Oscillation was a significant problem in early radios.

Lastly, the RF coils are actually transformers. There is another winding inside of the coil. I do not know if this was the standard method at the time or if this is associated with Atwater Kent. It is certainly an unusual technique.

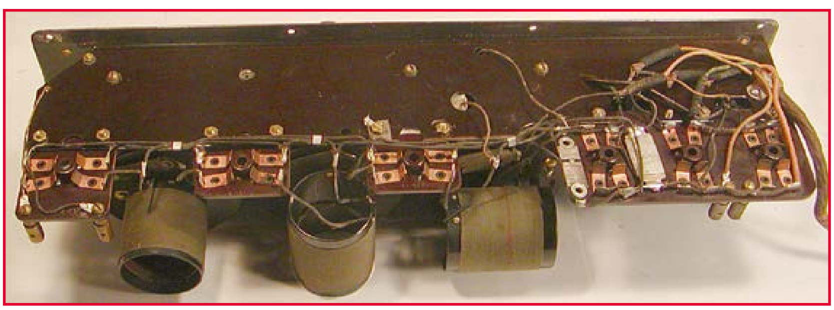


Figure 6. Clearly visible here are the tuning coils sticking out from the “chassis”.

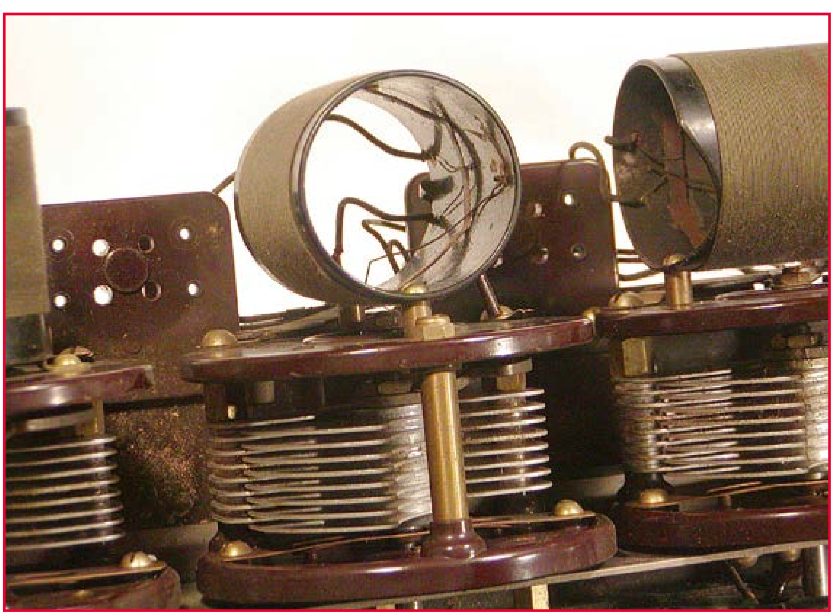


Figure 7. Detail of tuning coils mounted at right angles to each other.

Turning the Page

The battery-operated TRF radios were pretty much gone by 1930. They were replaced with regenerative circuits that provided high gain and selectivity with fewer tubes, or with superheterodynes that were similar to today’s radios. AC operation became standard and the prices for radios plummeted.

Unfortunately, my collection of huge radios didn’t survive my going to college and living in a small apartment. Of the three special pieces I had (the Silvertone, the AK-30 and the Ediphone) I was only able to keep the AK-30 and Ediphone. I gave the colossal Silvertone to my older sister. However, I don’t think she appreciated the gift. It was gone in a year or so. Too bad. Currently the going price for that in near perfect physical and electrical condition is about $1000.

(130426)

Web Links

[1] www.atwaterkent.info

[2] www.wd4eui.com

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

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.

Elektor World

Elektor's Nearly Lost World

By **Wisse Hettinga**

During Elektor's 60+ years' existence the lab and editorial teams must have built, handled and demoed hundreds of prototypes for the projects that made their magazine famous. A lot of hardware built in the old days got junked in 2007 during a move to new offices A few items escaped our cleansing frenzy though, mainly thanks to a few headstrong and proud guys from the lab aided and abetted by a zealous *Retronics* Editor. In the huge attic here at Elektor House we still cherish a small collection of Elektor hardware and projects and related stuff that defined the his-

tory of the lab, the magazine and the publishing company as a whole.

We hope you recognize some of the old stuff here: we have the Formant, the Elektor Junior Computer, the 'Mondrian' Plotter, items from high-end audio series, the Edwin amplifier, the Filmnet Decoder, The ElektorScope, a whole pile of vintage measurement equipment and more.

The picture is available at high resolution on www.elektor-labs.com/attic so you can browse the shelves from the comfort of your chair …. only look, don't touch!

We love e-stories. Under the topic specially created for this article at www.elektor-labs.com, please post a comment, story or whatever you feel fitting as a tribute to any old Elektor project you recognize in the picture (it's free for all Elektor Members). The posting, comment and/or story we like best wins a totally contemporary piece of hardware, the J2B board.

(130459)

Hexadoku The Original Elektorized Sudoku

Hexadoku participants are diehards. Every month a large number of them succeed in solving this mentally grueling puzzle. Join the crowd, find the solution in the gray boxes, submit it to us by email, and you automatically enter the prize draw for one of five Elektor book 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 five Elektor Book Vouchers worth **$70.00 (£40.00 / €50.00)** each, which should encourage all Elektor readers to participate.

Participate!

Before May 1, 2014,
supply your name, street address and the solution (the numbers in the gray boxes) by email to:

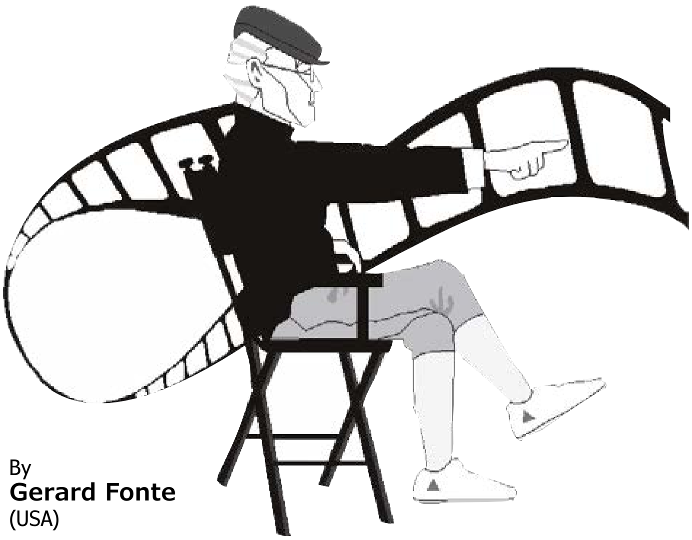
hexadoku@elektor.com

Prize winners

The solution of the January & February 2014 Hexadoku is: **1F734**.
The €50 / £40 / $70 book vouchers have been awarded to: Arun Annaji (India), Per Troelsen (Denmark), Arwin Vosselman (Netherlands), Brian Unitt (UK) and Udo Altmann (Germany).
Congratulations everyone!

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

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



By
Gerard Fonte
(USA)

Production

So you've decided that your ePod is going to make money for you. You're going to manufacture them and sell them yourself. And with a little luck and a lot of perspiration, you're hoping to retire early. Look out Bill Gates! You know that poor management is the root cause of 80% of the failures in new companies. But, you're not going to make that mistake because you've taken finance courses and read all about business management. Let's look at the failures not caused by bad management.

Product Quality

A good product for a good price is the hallmark of success. But an inferior product, at any price, will have a difficult time. It cannot be overstated that the quality of a new product must be exceptional. You're challenging existing products so your ePod must stand out in some way. It must clearly superior in performance or else sell for considerably less than similar products. Customers must get more bang for their buck or else they will not change their buying habits. Examine your own buying habits. What will it take for you to change your grocery store?

A product that has bugs or performs poorly is devastating in two major ways. The first is that it immediately clobbers your sales volume. You'll get few sales and a lot of returns. (Which you can't re-sell as a new unit. But many start-ups do anyway. Can you say bad management?) The second is that it gives your business a poor reputation. So even if you correct the product, you've alienated your customer base. Never forget—your customers give you money! Cherish them and cultivate them. People expect and deserve a professional quality product. By going into production you've said to the world that you're a professional and that your product is just as good, or better, than competing products. If you can't deliver that, don't try. All that will do is frustrate you and give your company a bad name.

Pricing

Start-ups often have difficulty in pricing their product. For example suppose your product costs $25 in parts and takes you an hour to assemble. Should your minimum selling price be $50 or $100? If you said $50 you'll be out of business very quickly. Here's why.

Suppose you get an order for 1000 ePods. That's 1000 hours of assembly time or 25 weeks of full-time effort. Very few customers are willing to wait half a year for delivery. So you will have to hire and train assemblers. Setting up your own company with employees, offices, insurance, equipment, taxes, etc. is not fast or cheap. An hour of technical labor plus overhead will be at least $25 an hour. So, right out of the gate, all your mark-up is gone. Even if you raise your price for other customers, you will have no profit from sales for at least six months. That's a huge financial hurdle to overcome.

Alternatively, you could contract out the assembly, locally. This will save you the set-up time, labor and overhead costs. However, the contractor is now paying for salaries and overhead. He expects to see a profit, so he will charge you more than what he is paying. Again you will see no profits for half a year. It's very possible that you will actually lose money.

Finally, you could contract with an overseas assembly house. Their labor costs are very low and nowadays they are fairly easy to work with and no longer require huge volumes (10 Kunits or more). Of course, there are the problems of language, import/export rules, contract negotiation, shipping costs, etc. Then there is the concern of intellectual property protection. It is not unheard of for these contracting houses to manufacture your product and sell it for themselves. With their name on it! It's no fun competing with your assembler. Initiating an international lawsuit is ridiculously expensive and time-consuming. Lastly, these cheap assemblers are notorious for poor workmanship. The bottom line on pricing is to assume a minimum cost of $25/hr for labor (which is still quite low). Your MINIMUM selling price should be twice the parts and labor costs. If this means that your selling price is not competitive, then you have a problem.

Design for Production and Test

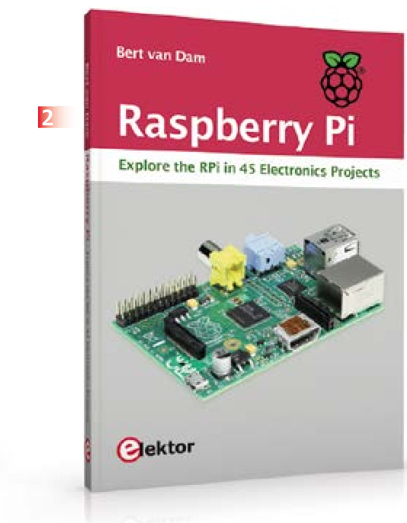
Probably the single most important factor in reducing labor costs is to design your product to be easy to assemble and test right from the start. Labor costs usually have the biggest influence in product pricing. So, if you can significantly reduce the labor, you can significantly reduce your assembly costs and make your product more financially attractive to consumers.

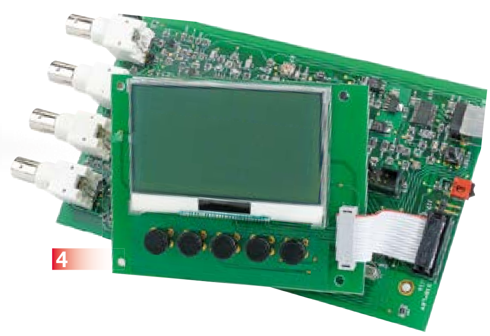
Obviously, there is way too much information about this to discuss here. But there are plenty of books on the subject. If you are serious about producing your ePod, then you should be willing to spend a few dollars for the books and a few weekends reading them. You may have to re-design your product. But, a few up-front hours spent in improving the ePod can save you thousands, or tens of thousands of dollars, in the long term. And then there's the engineer's mantra: "After you built it the first time, you know how to build it right."

Look around your home or office. Examine how the products are put together and learn from them. These are successful products. They have to be successful products or you wouldn't have bought them.

(130570)

SPECIAL OFFER
for GREEN and GOLD Members!
www.elektor.com/zeroes





110 Elektor Editions, Over 2500 Articles

1 DVD Elektor 2000 through 2009

This DVD-ROM contains all circuits and projects published in Elektor magazine's year volumes 2000 through 2009. The 2500+ articles are ordered chronologically by release date (month/year), and arranged in alphabetical order. A global index allows you to search specific content across the whole DVD. Every article is printable using a simple print function. This DVD is packed with ideas, circuits and projects that are ideal for any electronics enthusiast, student or professional, regardless of whether they are at home or elsewhere.

ISBN 978-1-907920-28-8

£77.95 • € 89.00 • US $121.00

Explore the RPi in 45 Electronics Projects

2 Raspberry Pi

This book addresses one of the strongest aspects of the Raspberry Pi: the ability to combine hands-on electronics and programming. No fewer than 45 exciting and compelling projects are discussed and elaborated in detail. From a flashing lights to driving an electromotor; from processing and generating analog signals to a lux meter and a temperature control. We also move to more complex projects like a motor speed controller, a webserver with CGI, client-server applications and Xwindows programs. Each project has details of the way it got designed that way. The process of reading, building and programming not only provides insight into the Raspberry Pi, Python, and the electronic parts used, but also enables you to modify or extend the projects any way you like.

288 pages • ISBN 978-1-907920-27-1

£34.95 • € 39.95 • US $56.40

Lots of power with low distortion

3 Q-Watt Audio Power Amplifier

Elektor has a long history with audio power amplifiers. We are proud to present yet another fully analog circuit developed entirely in house. Despite the simple design of this audio power amp with just one pair of transistors in the output stage, Q-Watt can deliver over 200 quality (Q!) watts into 4 ohms with exceptionally low distortion thanks to the use of a special audio driver IC. Due to popular demand, this project is now supported by a semi-kit containing everything you need to build a stereo amplifier.

2 PCBs and all electronic parts

Art.#110656-72

£151.25 • € 169.95 • US $244.00

The luxury of precision within everyone's reach

4 500 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

All articles in Elektor Volume 2013

5 DVD Elektor 2013

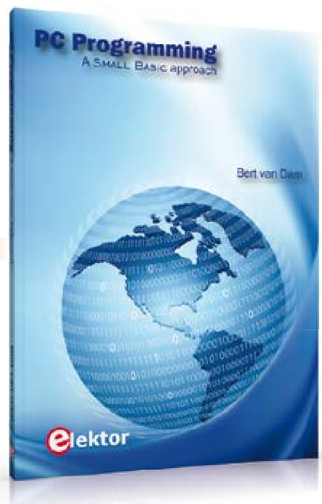
This DVD-ROM contains all editorial articles published in Volume 2013 of the English, American, Spanish, Dutch, French and German editions of Elektor. Using the supplied Adobe Reader program, articles are presented in the same layout as originally found in the magazine. An extensive search machine is available to locate keywords in any article. With this DVD you can also produce hard copy of PCB layouts at printer resolution, adapt PCB layouts using your favorite

5


6


7


8


9


graphics program, zoom in / out on selected PCB areas and export circuit diagrams and illustrations to other programs.

ISBN 978-90-5381-277-8

£23.50 • € 27.50 • US $37.90

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

6 XMEGA Web Server Board

This microcontroller board 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.

Controller Module

Art.# 120126-91

See www.elektor.com/xmega

Programming step-by-step

7 Android Apps

This book is an introduction to programming apps for Android devices. The operation of the Android system is explained in a step by step way, aiming to show how personal applications can be programmed. A wide variety of applications is presented based on a solid number of hands-on examples, covering anything from simple math programs, reading sensors and GPS data, right up to programming for advanced Internet applications. Besides writing applications in the Java programming language, this book also explains how apps can be programmed using Javascript or PHP scripts. When it comes to personalizing your smartphone you should not feel limited to off the shelf applications because creating your own apps and programming Android devices is easier than you think!

244 pages • ISBN 978-1-907920-15-8

£34.95 • € 39.95 • US $56.40

MIFARE and Contactless Cards in Application

8 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

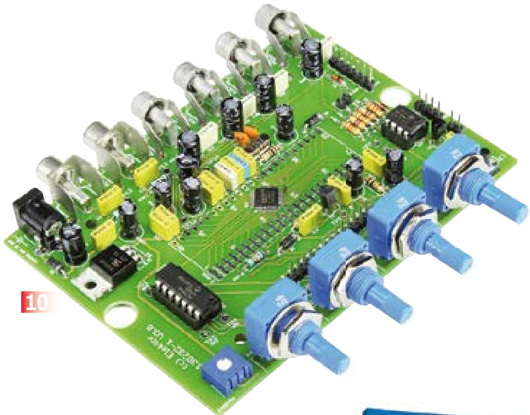
A Small Basic approach

9 PC Programming

There are many different PC programming languages available on the market. They all assume that you have, or want to have, a knack for technology and difficult to read commands. In this book we take a practical approach to programming. We assume that you simply want to write a PC program, and write it quickly. Not in a professional environment, not in order to start a new career, but for plain and simple fun... or just to get a task done. Therefore we use Small Basic. You will have an application up and running in a matter of minutes. You will understand exactly how it works and be able to write text programs, graphical user interfaces, and advanced drivers.

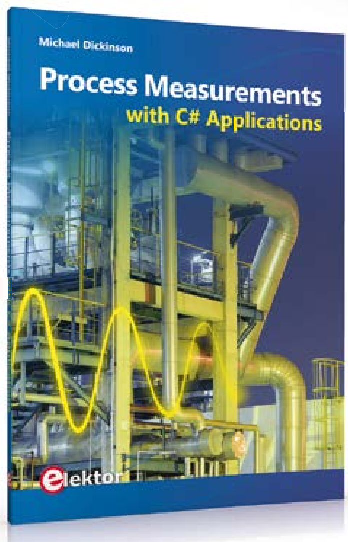
194 pages • ISBN 978-1-1-907920-26-4

£29.50 • € 34.50 • US $47.60

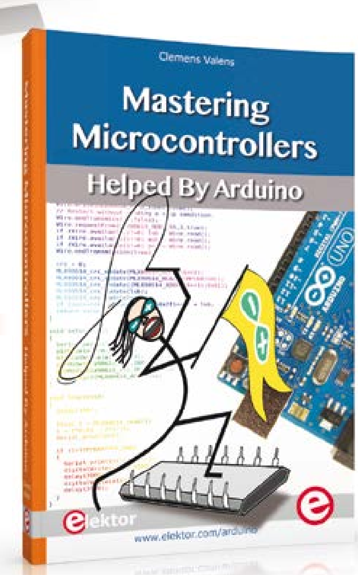


10

11



12



13

Drag 'n Drop

10 ADAU1701 Universal Audio DSP Board

Always wanted to start out with DSPs, but afraid of the SMD? Here's the answer: Elektor's all-DIY Universal Audio DSP Board! Based on the Analog Devices ADAU1701 DSP chip and drag-and-drop software, this board will ease your way to becoming a sound craftsman, guaranteed maths-free.

Semi-kit:
all TH components, PCB with DSP preassembled
Art.# 130232-71
£64.95 • € 169.95 • US $104.80

140 Minutes video presentation and more

11 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 audio designers and serious audio hobbyists!

ISBN 978-907920-16-5
£24.90 • € 29.95 • US $40.20

An essential source of reference material

12 Process Measurements with C# Applications

Measurement is vital to the successful control of any process. This book introduces PC based measurement systems and software tools for those needing to understand the underlying principles or apply such techniques. Throughout the book, the C# programming language is used to give the reader immediate practical desktop involvement. C-Sharp has a wide support base and is a popular choice for engineering solutions. The basics of measurement and data capture systems are presented, followed by examples of software post-processing. Application examples are provided from a range of process industries, with reference to remote monitoring, distributed systems and current industrial practices.

144 pages • ISBN 978-1-907920-24-0
£24.50 • € 27.50 • US $39.50

Helped By Arduino

13 Mastering Microcontrollers

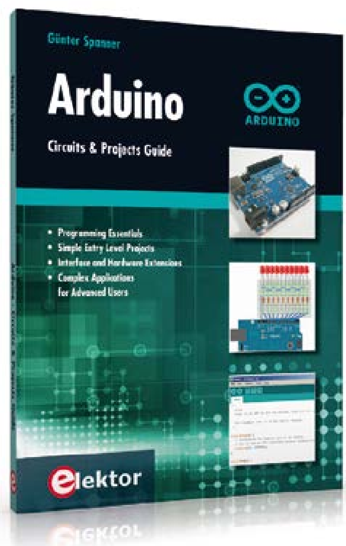
The aim of this book is not only to let you enter the World of Arduino, but also to help you emerge victorious and continue your microcontroller programming learning experience by yourself. In this book theory is put into practice on an Arduino board using the Arduino programming environment.
Having completed this fun and playful course, you will be able to program any microcontroller, tackling and mastering I/O, memory, interrupts, communication (serial, I²C, SPI, 1-wire, SMBus), A/D converter, and more. This book will be your first book about microcontrollers with a happy ending!

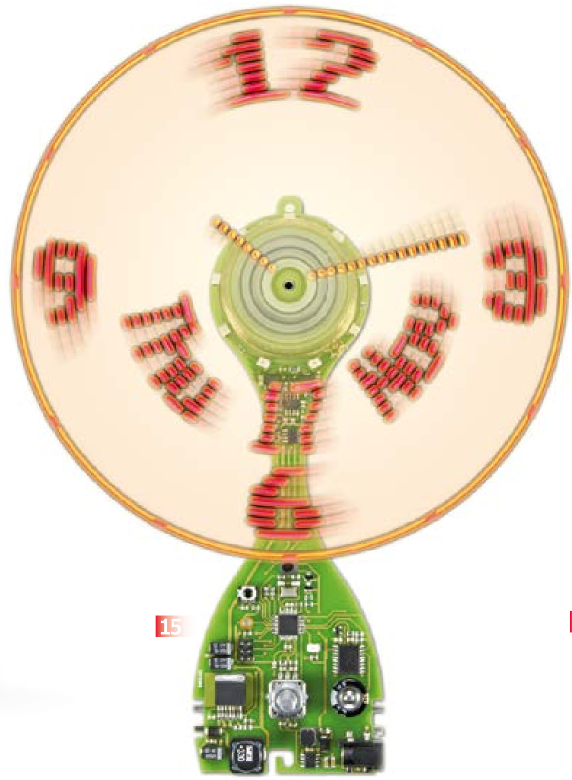
348 pages • ISBN 978-1-907920-23-3
£34.95 • € 39.95 • US $56.40

Circuits & Projects Guide

14 Arduino

The Arduino user is supported by an array of software libraries. In many cases, detailed descriptions are missing, and poorly described projects tend to confuse rather than elucidate. This book represents a different approach. All projects are presented in a systematical manner, guiding into various theme areas.

14 

15 

16 

In the coverage of must-know theory great attention is given to practical directions users can absorb, including essential programming techniques like A/D conversion, timers and interrupts—all contained in the hands-on projects. In this way readers of the book create running lights, a wakeup light, fully functional voltmeters, precision digital thermometers, clocks of many varieties, reaction speed meters, or mouse controlled robotic arms. While actively working on these projects the reader gets to truly comprehend and master the basics of the underlying controller technology.

260 pages • ISBN 978-1-907920-25-7
£34.95 • € 39.95 • US $56.40

Time & Date Floating in the Air

15 UltiProp Clock

Electronics is never so fine as when it skillfully combines magic with physics, mechanics with software, imagination with thoroughness and precision, and a taste for beauty with good workmanship. This timepiece was designed to display the time and date in an original way—but we admit we did also design it to draw cries of amazement from the visitors who find it in our laboratory. We'll bet many of you will want to do the same in your own homes!

Module, ready assembled and tested
Art.# 120732-91
See www.elektor.com/ultiprop

Ideal reading for students and engineers

16 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

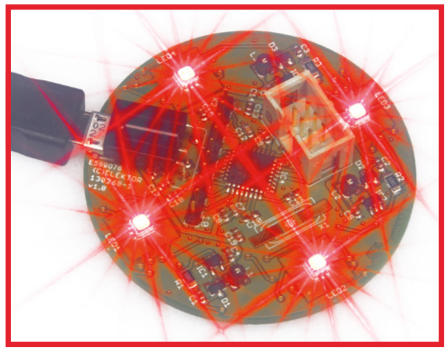
UK / ROW
Elektor International Media
78 York Street
London - W1H 1DP United Kingdom
Phone: +44 20 7692 8344
E-mail: service@elektor.com

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

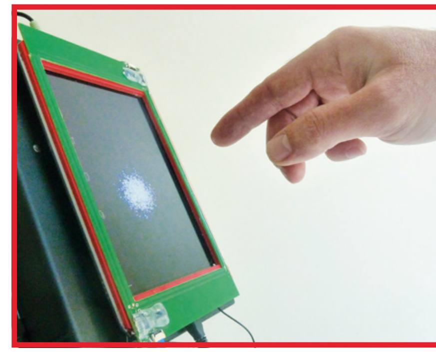
Arduino Experimenter's Shield

To perform simple experiments, such as those in the newly started Microcontroller BootCamp course, Elektor Labs developed a small experimental PCB for plugging on to practically any Arduino board. The shield contains two buttons, two LEDs, one potentiometer and level shifters for adjusting the logic levels of the Arduino board. In addition, the shield is equipped with a universal ECC connector.



Mini RGB Lamp with IR Remote Control

Lighting circuits with colored LEDs are all the craze nowadays. There are plenty of such RGB lamps for sale, but an electronics engineer naturally designs and builds such a device himself. This circuit is small and employs four LEDs connected in series and powered at just 5-V thanks to a simple voltage converter. The control is effectively via a standard RC5 remote control that allows, among others, color and brightness to be adjusted.



Touchfree Display Control

The ***Ootside Box*** mentioned in the March 2014 installment of Elektor World is a smart circuit that lets you detect hand movements from a distance by means of a metal frame around a tablet computer. The system allows you to operate programs and games with simple gestures. Next month we present the hardware and software designed for this purpose.

We regret that 'Intelligent Cuelight System' and '433 MHz Gateway' could not be published in the current edition as scheduled.

Article titles and magazine contents subject to change, please check www.elektor-magazine.com for updates.
Elektor May 2014 is processed for mailing to US, UK and ROW Members starting April 22, 2014.

See what's brewing @ Elektor Labs 24/7

Check out
www.elektor-labs.com
and join, share, participate!