25 pages on communication

February 2011 AUS$ 14.50 - NZ$ 17.50 - SAR 102.95 £ 4.80

+ Telephone-to-VoIP Adapter

elektor

TimeClick
Programmable Camera Controller

Geolocalization without GPS
Where am I? Where am I headed?

Light Alarm Clock
Programmed Sunrise

+ Ultimatic CW Keyer
For high-speed telegraphy operators

+ Contactless Thermometer
Are you running an infrared temperature?

R08 02

LPC11C2x CAN MCU Series

Industry's first integrated CAN microcontroller and transceiver with CANOpen® Drivers.

- Features high-speed CAN transceiver and 32-bit LPC1100 MCU
- Higher code density powered by Cortex-M0
- Superior performance than 8/16-bit microcontrollers
- Enable development of a wide range of industrial networking applications

www.nxp.com/microcontrollers

Writing your project with mikroC PRO for PIC

is a piece of cake

Useful tip:

If you buy some of our development systems, you get the mikroC PRO for PIC compiler at a discount price of

$199

$249

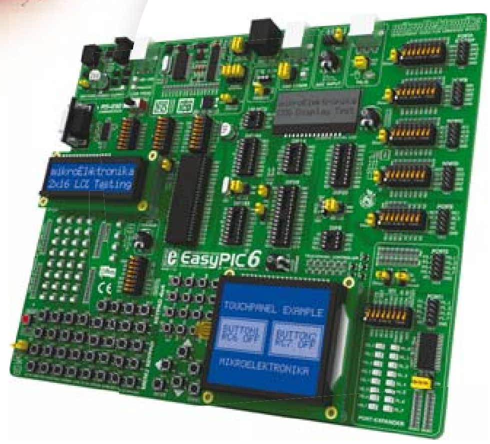
Ingredients:

1 x Intuitive IDE
1 x Powerful ANSI C compiler
11 x Software Tools
28 x Hardware Libraries
12 x Software Libraries
1 x Comprehensive Help file
139 x Ready To Use Examples

Preparation:

Get mikroC PRO for PIC. Open existing example you wish to start your project with. Include additional libraries and add custom code. Use Free Product Lifetime Tech Support as much as you need. Compile the project. Pour the generated HEX file into the desired microcontroller on EasyPIC6 board and enjoy!

BEST SERVED with EasyPIC™6 board



MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS
...making it simple

GET IT NOW
www.mikroe.com

Communicate

Surveys and feedback indicate that Elektor readers expect to find a large variety of articles, viewpoints and technical approaches in the magazine. For this February 2011 edition we have done our utmost to pack the pages ahead of you with a wide variety of topics with a slight accent on communication, our announced theme of the month. Let's see what's communicated on communication.
A telegram-style explanation is presented on the famous OSI layer model, which is widely used in computers and networks. There's also a handy VoIP adapter that enables the good old analogue telephone set gathering dust in the cupboard to be used for state of the art VoIP communication. The adapter features a USB port for connecting to the computer and is designed to work under Linux. If you like to mess around with old computer gear you're sure to have fun with the article on a DIY Texting (SMS) gateway based on a scrap PC and a cellphone your spouse or children have classified as 'RIP' or 'unfashionable'. There's plenty of ICT scrap material around, and it's often free for collection. Communication at the hands-and-feet level is also covered in this edition with the AlphaLED (AlphabetLED) letter shaker, a small board that 'writes' messages in the air if you wave it. There are a few messages in memory, but you can also create your own words or text by on the fly programming.
Finally we have a circuit that bridges 100+ years effortlessly, happily combining Morse (the pundits say CW) with PIC microcontroller technology. The Ultimatic keyer is for experienced CW fans, maintaining the correct time relations between dots, dashes, words and lines, besides doing a lot more. The project was designed to support the famous Ultimatic mode, a system that reduces hand movement on part of the telegraph operator and so allows amazing speeds of up to 100 wpm to be achieved.
And more ... just browse this edition because there are many more interesting articles I am unable to communicate at the risk of exceeding the 360 word limit the page layout colleagues have communicated over coffee and a piece of OSI cake (page 14).

Jan Buiting, Editor

elektor

CONTENTS

Volume 37
February 2011
no. 410

elektor | international media bv

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

Volume 37, Number 410, February 2011 ISSN 1757-0875

Elektor aims at inspiring people to master electronics at any personal level by presenting construction projects and spotting developments in electronics and information technology.

Publishers: **Elektor International Media, Regus Brentford, 1000 Great West Road, Brentford TW8 9HH, England. Tel. (+44) 208 261 4509, fax: (+44) 208 261 4447 www.elektor.com**

The magazine is available from newsagents, bookshops and electronics retail outlets, or on subscription.
Elektor is published 11 times a year with a double issue for July & August.

Elektor is also published in French, Spanish, American English, German and Dutch. Together with franchised editions the magazine is on circulation in more than 50 countries.

International Editor:
Wisse Hettinga (w.hettinga@elektor.nl)

Editor: **Jan Buiting (editor@elektor.com)**

International editorial staff: **Harry Baggen, Thijs Beckers, Eduardo Corral, Ernst Krempelsauer, Jens Nickel, Clemens Valens.**

Design staff: **Christian Vossen (Head), Ton Giesberts, Luc Lemmens, Jan Visser.**

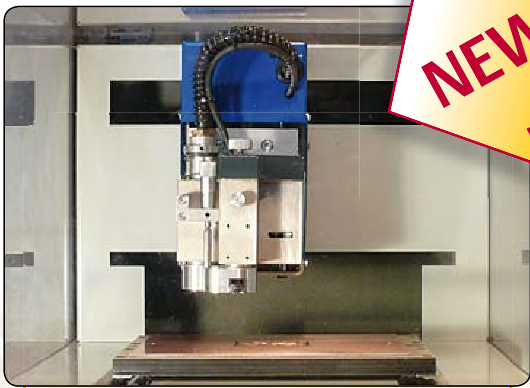
Editorial secretariat:
Hedwig Hennekens (secretariaat@elektor.nl)

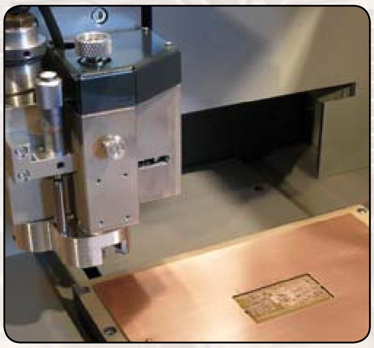
Graphic design / DTP: **Giel Dols, Mart Schroijen**

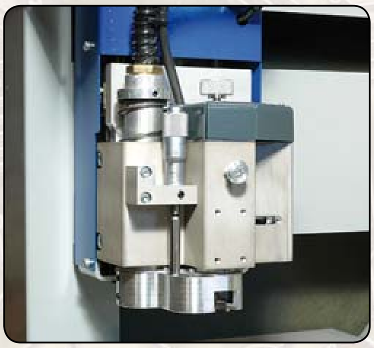
Managing Director / Publisher: **Paul Snakkers**

Marketing: **Carlo van Nistelrooy**

Subscriptions: **Elektor International Media, Regus Brentford, 1000 Great West Road, Brentford TW8 9HH, England. Tel. (+44) 208 261 4509, fax: (+44) 208 261 4447 Internet: www.elektor.com/subs**







Email: subscriptions@elektor.com
Rates and terms are given on the Subscription Order Form.

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

Distribution:
Seymour, 2 East Poultry Street, London EC1A, England
Telephone:+44 207 429 4073

UK Advertising:
Elektor International Media b.v.
P.O. Box 11 NL-6114-ZG Susteren The Netherlands
Telephone: (+31) 46 4389444, Fax: (+31) 46 4370161
Email: t.vanhoesel@elektor.com
Internet: www.elektor.com
Advertising rates and terms available on request.

External 48 & 60 watt power supply series meets latest energy standards

XP Power recently announced the launch of the highly efficient single output AFM series of 48 & 60 watt external AC-DC power supplies designed for a wide range of IT and medical equipment. The ranges offer output voltages of 12, 15, 18 and 24 VDC. Offering a typical efficiency of up to 88%, these highly efficient units meet the latest stringent energy efficiency standards such as Energy Star Level V, EISA2007 and CEC2008 in the United States and the ErP Directive for Europe. These standards define

the average energy efficiency and the maximum no load power consumption. The AFM45 has a no load power consumption of less than 0.3 W, and the AFM60 less than 0.5 W.

In addition, the AFM series comply with the internationally recognized safety approval standards for IT and commercial equipment IEC60950-1 / UL60950-1 / EN60950-1. They also comply with the IEC60601-1 / UL60601-1 / EN60601-1 medical safety standards.

The units offer multiple input cable interface options allowing designers to specify either IEC320-C14, -C6 or -C8 connectors. The -C8 option provides a Class II grounding method. An optional cable restraining clip is also available.

XP Power is an Energy Star partner and has an approved certification facility that allows in-house testing for compliance to the energy efficiency standards.

To help designers keep up to date with the changing energy efficiency legislation, XP Power provides a number of useful resources and information on their web site. In addition, XP Power applications and support staff can assist customers in understanding the various 'green' initiatives.

The AFM series is available from Farnell or direct from XP Power. The units have a 3 year warranty.

www.xppower.com (100820-IX)

Smart current sink LED backlighting platform

Semtech Corp. recently announced the industry's first smart current sink LED backlighting platform with on-chip digital lighting effects for high-end handheld displays. This new platform incorporates Semtech's patent-pending, smart Automatic Dropout Prevention (ADP) technology to enable a new-generation of high-quality current sink drivers that can replace boost converters and charge pumps in high-end handhelds, while providing high-quality display backlighting. The new SC667 and SC668 current sinks with ADP technology reduce the total parts count and extend battery life compared to boost converters or charge pumps, and offer far superior illumination quality compared to conventional current sink drivers. Additionally, on-chip digital lighting effects provide the flexibility to incorporate fade, breathe and blink effects without changing the firmware.

White LEDs used in backlighting applications typically have a forward voltage up to 3.6 V. When the battery voltage declines in portable devices, the supply voltage must be boosted to ensure the white LEDs have sufficient voltage to illuminate the display. Charge pump or inductive boost converter devices have typically been used to provide this voltage boost function. In an effort to maintain constant output power, these circuits increase current draw as the battery voltage declines, shortening battery life. Improvements in white LEDs have resulted in forward voltages as low as 3.0 V, reducing the threshold at which conventional LED drivers need to boost the battery voltage. Because of this, LED backlight drivers increasingly are operating in a non-boost mode, making current sink drivers an attractive alternative. Current sink drivers eliminate the capacitors and inductor associated with the boost circuitry, reducing component count, board size and sys-

tem cost, with the added benefits of eliminating any switching noise and extending operating time.

The SC667 and SC668 are the first current sinks to incorporate ADP technology. These devices also integrate a number of functions to enable high-end features on portables, including an ambient light sensing/control circuit that sets backlight brightness based on surrounding lighting conditions. A PWM dimming interface that incorporates a digital low-pass filter is also included, providing the capability to perform content-adaptive brightness control (versus 'always-on' illumination).

The SC668 provides eight current sinks, while the SC667 features seven current sinks plus an interrupt request indicator signal to tell the host processor when an ambient light threshold has been crossed. Both devices include an on-chip digital lighting-effects engine to control LED fade-in/fade-out, breathe, blink, auto-dim full, and auto-dim partial; an I^2C interface to program and control the LEDs; and four programmable, 200 mA low-noise LDO regulators to manage the power for multiple embedded peripherals.

Key Features of the SC667 and SC668 include

- Patent-pending Automatic Dropout Protection (ADP) technology
- On-chip digital lighting effects (fade, breathe, blink) without changing firmware
- Ambient light sensing option can automatically adjust backlight brightness
- Low parts-count: only one low-voltage capacitor needed, no inductor required
- I2C programming interface (fast and

standard modes)
- ±0.5% (typ.) LED current matching, ±1.5% (typ.) LED current accuracy
- Four low-noise LDO regulators
- Ultra-small, low-profile 20-pad MLPQ package with exposed thermal pad: 3 x 3 x 0.6 mm

The SC667 and SC668 are available immediately in production quantities.

www.semtech.com (100820-X)

MEMS oscillator covers 1 MHz to 800 MHz frequency range

IQD has announced immediate availability of a newly developed high frequency MEMS Oscillator on the opening day of Electronica 2010. The new IQMS-900 series MEMS (Micro Electro Mechanical Systems) oscillator is available in an exceptionally wide frequency range from 1MHz up to 800MHz. The new model can be factory programmed with either a LVDS (low voltage differential source) or LVPECL (low voltage positive emitter coupled logic) output.

The plastic packaged IQMS-900 series is available in two sizes: either a 7 x 5mm or the increasingly popular 5 x 3.2mm. Due to its MEMS technology design, phase jitter is low being typically 0.7 ps at 200 MHz. Two supply voltages are available, 3.3 V & 2.5 V.

Frequency stabilities can be specified at either ±10 ppm over an operating temperature range of 0 to +70 degrees C or ±15 ppm over –40 to +85 degrees C.

MEMS technology is based upon the process techniques of CMOS (Complementary Metal Oxide Silicon) and as such can be produced on standard production lines in foundries that are more used to producing standard or custom integrated circuits. This of course lends itself to the ability of producing product in vast quantities at an economic rate.

The IQMS-900 range is eminently suitable for the incorporation into infrastructure equipment where it will provide very accurate timing processes coupled with the ability to drive the very latest high-speed processors that will enable the transfer of high data rates — the fast rise and fall times contribute to this. Typical applications include Fibre Channel, Ethernet 10G, HDMI, SATA/SAS & USB3.

www.iqdfrequencyproducts.com (100820-XI)

Factory modification services for enclosure panels

Standard front panels for enclosures and sub racks, typically conforming to sizes specified in international standards, are produced by many companies, giving users ready availability and a wide choice of suppliers. All such panels have one thing in common; they will usually need to be modified to suit the application before they can be used. Vero Technologies offer a rapid turn-round factory customisation service for its own manufactured panels, a

modification service for panels from other suppliers and bespoke panel fabrication to customer drawings with a maximum size of 480 x 400 mm.

Panels can be punched, drilled, multi-colour screen-printed, engraved, engraved with a coloured infill, fitted with polyester overlays, rebated to accept a flush-fitting membrane keypad and fitted with EMC gaskets. Conductive and non-conductive finishes, to both sides or just to the rear can be provided or the panels can be painted. Cutouts can be within 1mm from the panel edge without distortion by using laser machining or extend through the edge by using three-dimensional CNC machining, enabling rebates and apertures to be created. Both plastic and metal panels can be modified; plastic panels and enclosures can be moulded in custom colours. Specialist finishes can be applied to give EMC/RFI protection or anti-bacterial capability for use in medical instrumentation.

Card mounting brackets, conductive and non-conductive fixings, handles, gaskets and sealing strips can all be supplied to meet specific project requirements.

www.verotl.com (100820-XII)

100-watt low profile PCB-mountable power supplies

Power supply expert TDK-Lambda France has upped the output power of its ZPSA series of compact PCB-mountable AC-DC power supplies with the introduction of a 100 watt model. The new single-output ZPSA100 accepts a wide input voltage range, has a very low profile (26.6mm) and industry standard footprint (127 x 76.2 mm), making it an ideal choice for applications such as LED signage and lighting, point-of-sale equipment, datacom, video/audio routers and test & measurement equipment.

These 100-Watt supplies are available with the most popular single output voltages from 5V to 48Vdc. Offering a typical efficiency of up to 90%, these models are well-suited for operation in both convection or customer air cooled environments from 0°C up to 70°C, with appropriate derating. A green LED is provided as an indicator that the power supply is on, and other standard features include over voltage and short-circuit protection. These supplies are offered in an open-board configuration with Molex input/output connectors.

Accepting a wide input voltage range of 90-264 VAC (47–440 Hz) or 120–370 VDC, the ZPSA100 series is ready for use globally with no further configuration or input selection. Its industry standard footprint makes it an ideal choice as a drop-in replacement for existing supplies, while its low profile

means that it can be installed in the most compact of applications.

TDK-Lambda's ZPSA100 series is approved to national and international safety approvals, including IEC/EN/UL/CSA60950-1 (edition 2) meets conducted and radiated EMC requirements of EN55022-B and FCC Class B (without additional filtering or components) and meets EN61000-4 immunity specifications for greater reliability. All models in the ZPSA100 series carry the CE mark, according to the LV Directive, and come with a two-year warranty.

www.uk.tdk-lambda.com/zpsa (100820-XIII)

WLCSP 2.4 GHz chips target space-constrained sports, fitness, and health applications

Ultra low power (ULP) RF specialist Nordic Semiconductor ASA (OSE: NOD) recently announced that it is to expand its existing 2.4 GHz RF and ANT™ product line-ups in Q1 2011 with a new set of ultra miniaturized, wafer-level chip scale package (WLCSP) options designed to meet the highly space-constrained needs of both existing and emerging sports, fitness and health applications such as wireless watches, bike computers, sensors, hearing aids and other devices designed to be worn

on or near the body.

Sampling in Q1 2011 and available for volume orders in Q2 2011 will be the nRF24AP-2WLCSP (1- and 8-channel) and nRF24LE1 WLCSP (Flash or OTP) options.

The new 1- and 8-channel nRF24AP2 WLCSP package options will represent the world's small single chip ANT solutions (nRF24AP2-1CHC32 and nRF24AP2-8CHC32) featuring 400µm pitch (regular array) 32-ball BGAs with a thickness of 0.5 mm and a flat footprint area of just 2.6 x 2.7mm (7 mm²) that is over 5x smaller than the footprints of competing 6 x 6 mm (36 mm²) packaged products.

The new nRF24LE1 WLCSP option (Flash or OTP) will again be a 400µm (regular array) 32-ball pitch BGA measuring 2.7 x 2.7 (7.3 mm²) in footprint area for the Flash version and 2.6 x 2.7 (7 mm²) for the OTP version. Both devices are 0.5 mm in thickness.

www.nordicsemi.no (100926-I)

RAP for ADAM mobile robot

RMT Robotics® a Cimcorp Oy company, launches ADAM RAP (Reactive Audio Playback). The programmable sound system, designed exclusively for the RMT Robotics ADAM (Autonomous Delivery and Manipulation) mobile robot, includes interactive voice messages and a mobile "vehicle in motion" jukebox for every mood and season.

"In developing the ADAM RAP module, we wanted to create a new 'vehicle in motion' function that not only improves safety but leverages the power of the ADAM platform to enhance the interactive experience between humans and the mobile robots that they work with every day," said Bill Torrens, Director of Sales & Marketing for RMT Robotics.

AGVs have a standard beeper-based 'vehicle in motion' alert system which is mandated by most international safety standards. In most operations however, noise proliferation combined with the monotonous beep of vehicles tends to diminish the alertness of workers with constant exposure. An enhanced audio system that could generate a variety of sounds and automatically associate these sounds with vehicle position or function would dramatically improve the effectiveness of the warning system.

RMT Robotics developed the ADAM sound application to play 'text to speech' messages, sound bites or musical interludes through its mobile robot that can be either actively or passively triggered in reaction to a variety of operational conditions and system inputs. Through the ADAM Commander mapping software, regions on

the facility map can be embedded with commands to automatically load specific tunes or prompt a series of voice messages. For example, ADAM may play a song in the aisle ways and then switch to a text to speech-based message saying, "Excuse me I am coming through" as it approaches the region of a doorway. After successfully moving through the doorway, ADAM will automatically revert to playing the original song.

ADAM RAP also has the ability to react to inputs. As another example, if a worker (or object) is fully obstructing the doorway and ADAM is unable to reconcile an alternative path, ADAM will politely enunciate, "Please move, I cannot get around you." As another example, if the emergency stop button is pressed, ADAM will state, "My emergency stop has been pressed," or if the Enter Destination button is pressed on the vehicle keypad, ADAM will state, "Thank you, see you later" before proceeding on its mission.

"Now that ADAM can "speak" and interact with workers, there is an improved harmonization between the work force and the robots that serve them," says Torrens. "The fact that ADAM can also entertain while enhancing safety and efficiency in the process is an added bonus."

www.rmtrobotics.com (100926-II)

Contactless sensing for 360° navigation in human-machine interfaces

austriamicrosystems has announced the AS5013, a contactless magnetic encoder IC that monitors the displacement of a magnet incorporated in a knob relative to its centre position and provides x and y position information via an I²C interface. The AS5013 Hall-sensor IC is used in the EasyPoint(TM) module, which consists of a mechanical stack incorporating a naviga-

iPad app for Engineers

Apple brought a whole new format to the world of personal computing when they introduced the iPad. The handy tablet computer can be taken anywhere and the interactive touch screen makes it dead easy to use. Add to that the convenience of the thousands of downloadable software apps available to solve real-world problems the iPad can be without doubt a very useful tool.

Elektor have added another to the firmament of apps with their 'Elektor Electronic Toolbox'. This one should be indispensable to engineers and hobbyists alike. It comprises 28 applications, any one of which can be selected from the opening screen.

The app contains a data bank of over 45,000 electronic components, including bipolar transistors, FETs, triacs, thyristors, diodes and ICs. A component can be selected from the lists in different categories on the left hand side of the display. As usual for iPad you can drag your finger to scroll through the list and select a component by tapping the screen whereupon the pin-outs and important electrical characteristics appear on the right side of the display. All data is contained in the app so an internet connection is not necessary. Also included is a special data bank containing pin assignments of the majority of connectors used in the fields of audio, video, computer and telephone engineering.

To add to that an interactive component calculator is included which simplifies the design of resistive circuits (series, parallel, bridge), high and low pass filters (R/C, R/L and L/C) and transistor circuits (with base resistor values and voltage divider calculations). The ubiquitous NE555 is also included as a basic circuit building block as are all the characteristics of the different colour LEDs.

The electronic toolbox not only provides assistance in component selection but also supports some engineering design activities. For example circuits often require a regulated DC linear power supply and this app has all the tools necessary to make the job a cinch.

Other useful tools include a virtual resistor colour-code clock, units of measurement converter, circuit diagram symbol data bank plus much more.

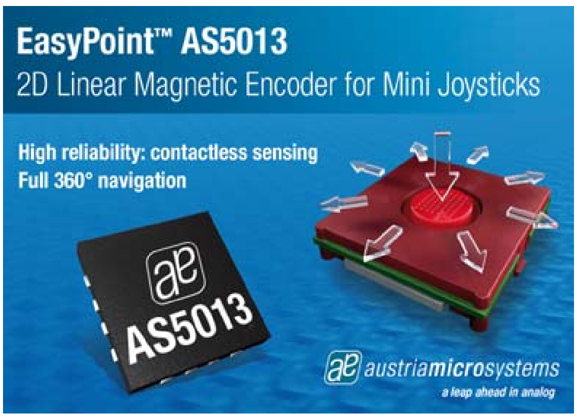
The Elektor Electronic Toolbox can be downloaded from the Apple iTunes Store for just $4.99. Go to the apps web page for more information.

UK & European readers: http://apps.elektor.com/Toolbox/?c=en&d=3&l=en

US & Canada readers: http://apps.elektor.com/Toolbox/?c=us&d=3&l=en

tion knob, a magnet and the AS5013 magnetic encoder IC. Its simple construction and contactless sensing technology give the module very high reliability and is designed for any kind of 360° navigation input device.

The AS5013 is a complete Hall-sensor IC for human-machine interface (HMI) applications requiring low power. This simplifies software integration challenges for joystick navigation in products such as cell phones and other handheld devices, e.g. MP3 players, PDAs, GPS receivers, Digital Still Cameras, gaming consoles, and remote controls. The AS5013 single-chip IC includes user-selectable power-saving modes, 5 integrated Hall sensing elements for detecting up to ±2 mm lateral displacement, a high-resolution analogue-to-digital converter (ADC), an XY coordinate and motion detection engine, and a smart power management controller. The XY coordinate registers and mag-



netic field information for each Hall-sensor element are transmitted over the I²C interface to the host processor.

The HMI device also provides two interrupt modes (motion detect, data ready) and two operating modes: idle mode, with less than 3 µA current consumption, and low-power mode, with selectable readout rate. The low-power AS5013 operates over a power supply range of 2.7 to 3.6 V, and down to 1.7 V peripheral supply voltage.

Specified over an operating temperature of –20 to +80°C, the AS5013 Hall-sensor IC operates from a 3.3 V power supply and is offered in a 16-pin 4 x 4 x 0.55 mm QFN package. The EasyPoint(TM) module is available as small as 8 x 8 x 1.5 mm.

www.austriamicrosystems.com/AS5013/Hall-sensor-IC

(100926-VI)

OSI from ISO
Seven layers is all it takes

As with any structured approach to such a problem it would be best to identify and divide the job into manageable tasks. Processes to deal with error control and data packet addressing for example can be easily defined. Some of the lower level tasks would be concerned with details of the physical transfer of bits over the medium while other higher level tasks would deal with the interface to the application software. In between the data would pass through a series of intermediate processes. This gives rise to a functional hierarchy of the program structure which Elektor editorial team have chosen to illustrate using this seven-layered cake.

Now we come to the International Organization for Standardization (ISO) which as

1 Physical

Radio, optical or cabled, it doesn't matter which medium you use, this layer defines how a digital 0 and 1 is represented in the media so that transmitter and receiver can communicate with one another.

2 Data link

This layer is responsible for detecting corruption of the transmitted message. The interference may come from an external source or from more than one transmitter talking at the same time on the network (data collision). Techniques such as data packetisation, message headers, checksums and CRC algorithms all help here.

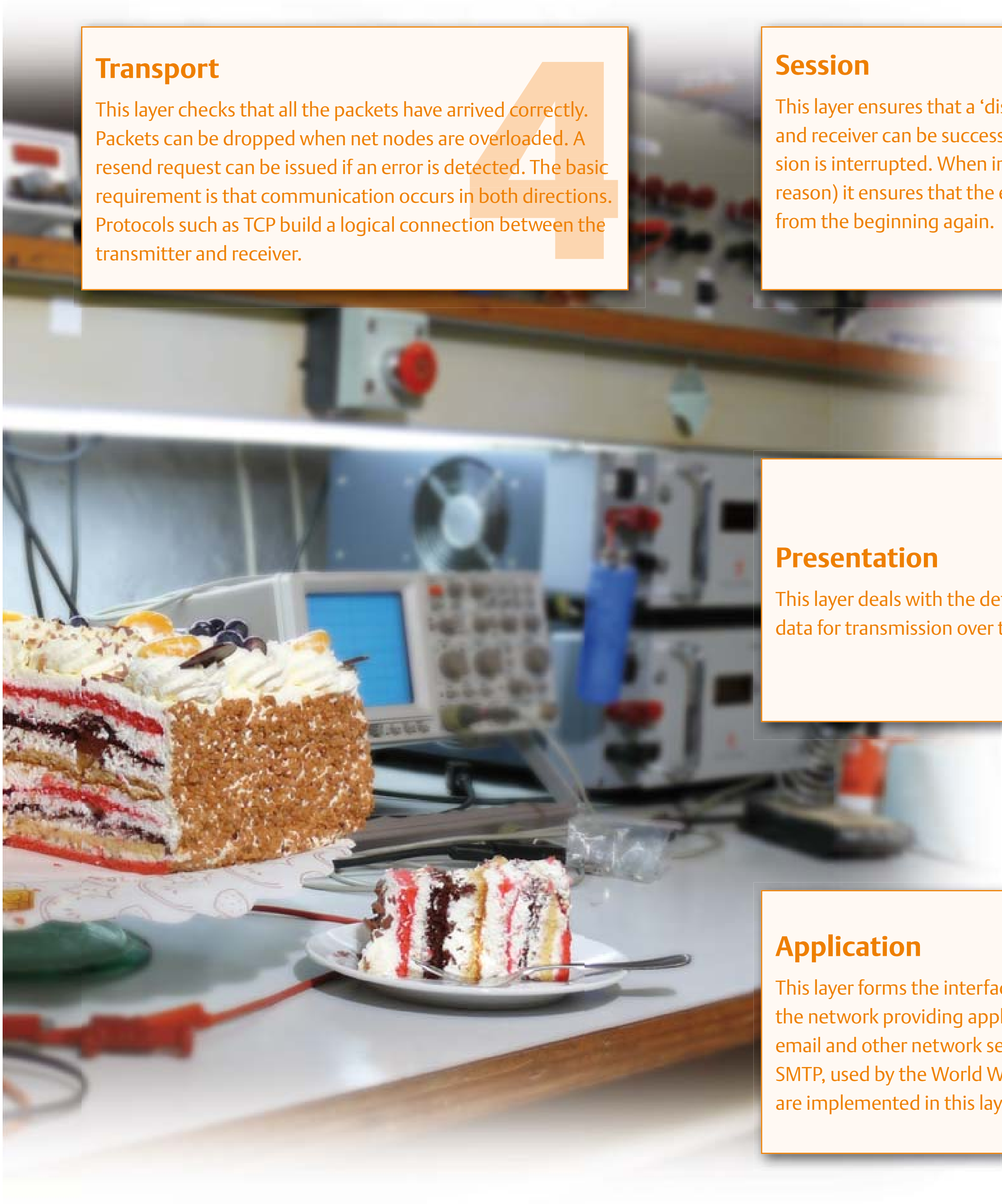
3 Network

This layer ensures that each packet arrives at the correct receiver. In larger networks there may be a hierarchy of network nodes relaying the data packets. Similar to how a postal address and code uniquely identifies a delivery address, the destination address is comprised of several fields e.g. a series of characters or character sequence.

When computers communicate it's important that messages are exchanged reliably, efficiently and securely (to protect against eavesdropping). To build such a complex communication protocol from scratch represents more than a weekend's work for sure.

usual has a thing or two to say about technical standards. At the beginning of the 1980s they released the OSI layered model (Open Systems Interconnection Model) which subdivides such a communication system into seven distinct 'layers'. When all the interfaces are strictly adhered to (i.e. the function of each layer and the data between the layers) it is possible to substitute a layer from one implementation of the standard model to another without problem. This promotes the concept of interchangeable software building blocks which can simply be linked to fulfil a particular communication requirement. A similar model called TCP/IP uses fewer layers and is responsible for data transfers most notably over the internet.

(100781)



Transport

4

This layer checks that all the packets have arrived correctly. Packets can be dropped when net nodes are overloaded. A resend request can be issued if an error is detected. The basic requirement is that communication occurs in both directions. Protocols such as TCP build a logical connection between the transmitter and receiver.

Session

5

This layer ensures that a 'discussion' between the transmitter and receiver can be successfully concluded, even if the session is interrupted. When interruption occurs (for whatever reason) it ensures that the exchange does not need to start from the beginning again.

Presentation

6

This layer deals with the details of formatting and encrypting data for transmission over the network.

Application

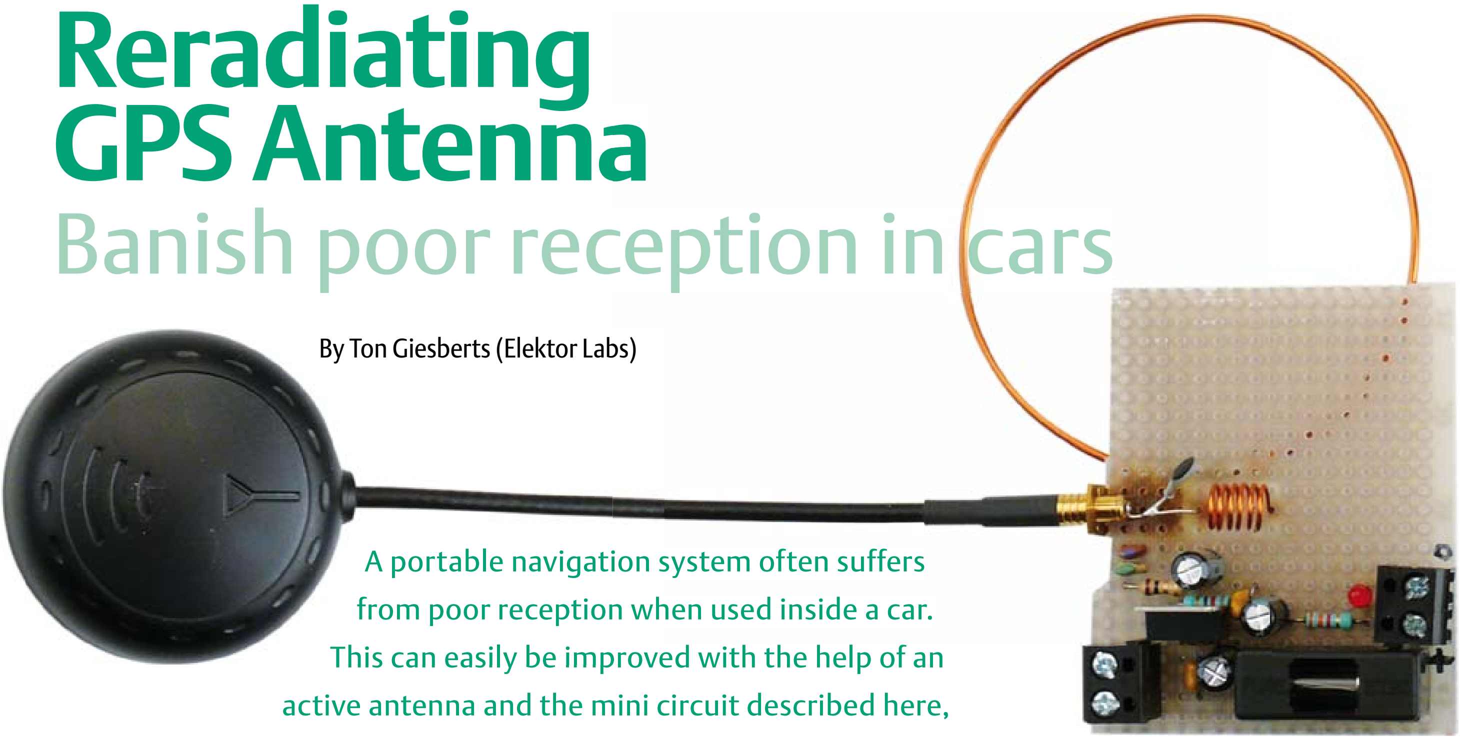
7

This layer forms the interface between the user software and the network providing application facilities for file transfer, email and other network services. Protocols such as HTTP and SMTP, used by the World Wide Web (e.g. to transfer emails) are implemented in this layer.

Reradiating GPS Antenna

Banish poor reception in cars

By Ton Giesberts (Elektor Labs)



A portable navigation system often suffers from poor reception when used inside a car. This can easily be improved with the help of an active antenna and the mini circuit described here, which doubles as the battery charger.

The signal strength of a portable navigation system such as a TomTom, a Garmin eTrex or a PDA/mobile phone with built-in GPS-receiver is often poor inside a car because of the metal coating which is often applied to the windscreen of a modern car. A possible solution is an active, external antenna. This type of antenna is available in all sorts of shapes and sizes. However, many portable GPS-receivers do not have a connection for an external antenna (any more); PDAs are not all that likely to have one either.

In order to provide a navigation system without an external antenna connection with a stronger satellite signal, it is possible to use a so-called re-radiating antenna. A loop antenna is installed either in, behind, or in front of the navigation system (depending on the position of the receiving antenna inside the navigation system) which then re-radiates the received and amplified GPS signal from an active antenna mounted on the outside of the car. In this way the navigation system is still able to receive a sufficiently large signal so that it can function properly.

For this design we only need a commercial active GPS antenna and a tuned circuit which will re-radiate the output signal that is available on the connector at the end of the active antenna cable. In addition, we require a regulated power supply for powering the active GPS antenna. And while we have one of those, we could, at the same time, also use it to charge/power the navigation system or PDA while in the car. The circuit is so small that all of it is easily accommodated on a small piece of prototyping board of a few square centimetres.

The circuit basically consists of a standard application for the LM317 voltage regulator (C3 for less noise and higher ripple/interference suppression) and a pass-through circuit for the signal from the active antenna to the antenna which will be mounted near the navigation system. A loop antenna is used for the radiating element. This is made from a sturdy piece of enamelled copper wire soldered to the prototyping board on which the remainder of the circuit is built. The wire is 19 cm long, equal to the GPS signal's wavelength. It is normally recommended to keep the loop between 1/8 and 1/4 of the wavelength to prevent self resonance. But with a larger dimension it is easier to bend it into a shape that suits the GPS receiver better (the loop has to be positioned near the location of the internal antenna).

The power supply for the active antenna is first RF decoupled with R4 and C5/C6. R4 is

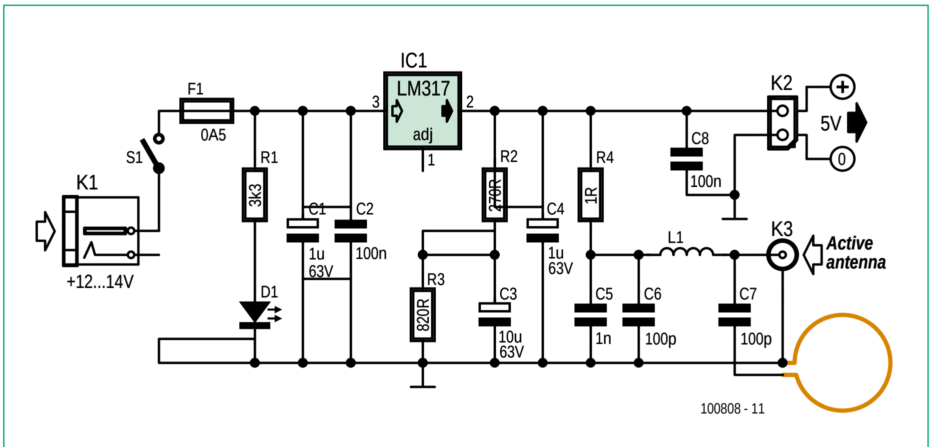


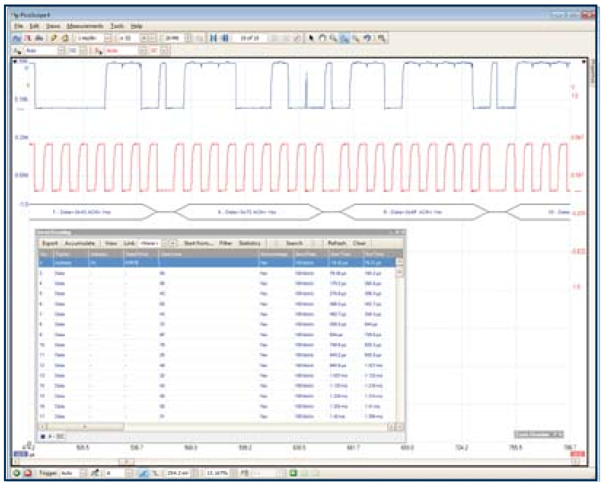
Figure 1. The schematic for the circuit: a voltage regulator and an antenna circuit which re-radiates the GPS signal from an active antenna.

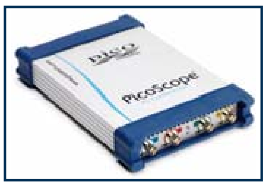
a normal, through-hole resistor which is mainly used as an inductor. By keeping the value of R4 low, the voltage drop across it is kept as small as possible. L1 decouples the active antenna signal from the power supply voltage. L1 is an air-cored coil, made by winding a few turns (6 or so) around a small diameter drill bit (diameter 5 mm). Keep the individual turns apart by at least the thickness of the wire used (for example 0.5 mm CuL). This is to minimise the internal capacitance. A standard 100-pF-capacitor is used to couple the signal from the active antenna to the loop antenna. None of this is very critical.

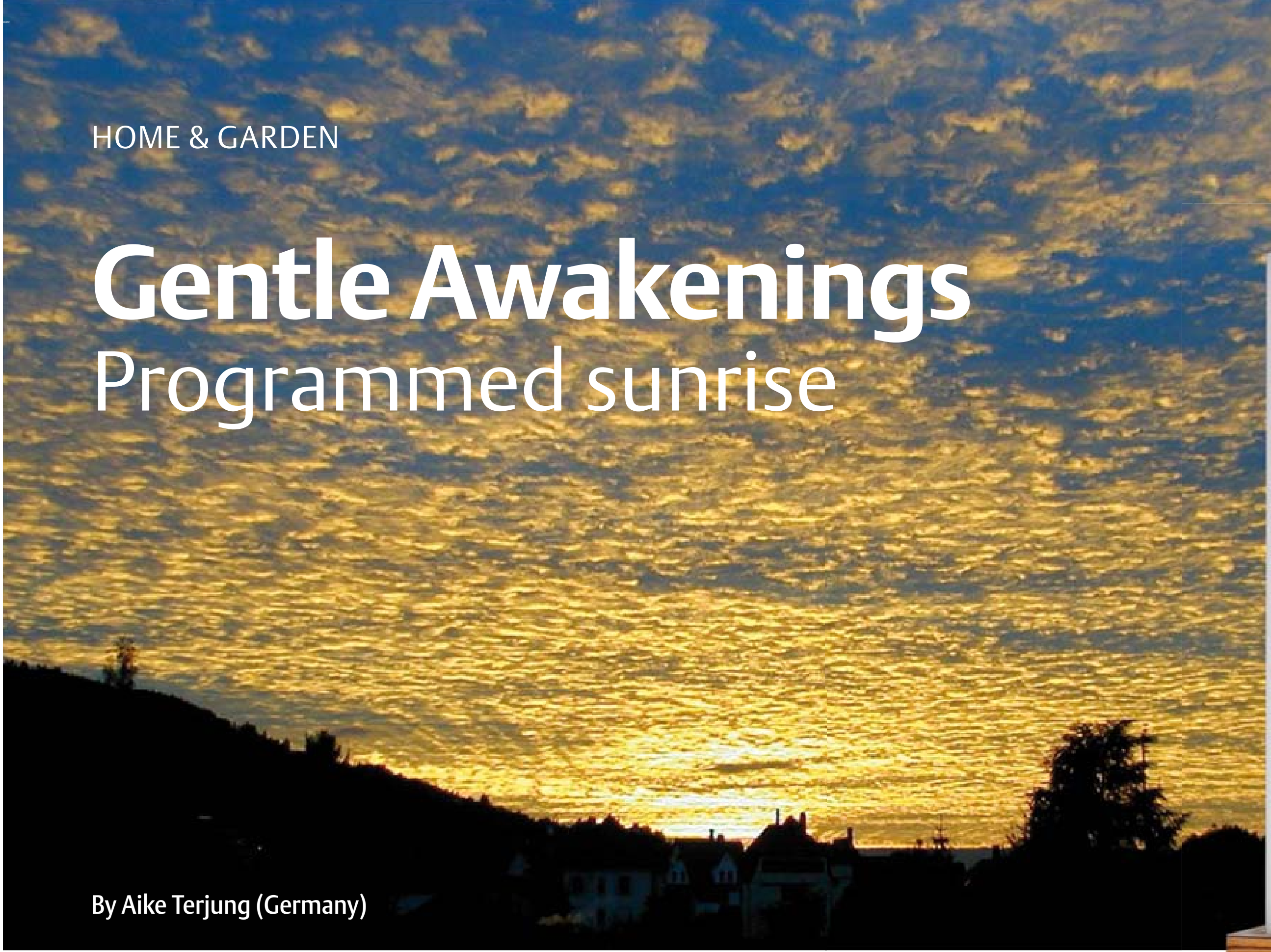
The photo shows the assembled prototype. The circuit was found to work superbly. We have tested it within the thick walls of our castle with, among others, a Pocket PC Mio P350. This had no reception inside the building, but by placing it next to the loop antenna it was able to find six satellites quite quickly. The active antenna was placed near a window. The 5-V voltage on K2 can be used as the power supply for the Pocket PC (or GPS receiver or some other device with GPS functionality). This is often a mini-USB connector which is also intended for charging the battery. Fuse F1 is there for safety, so that when experimenting you will not easily blow one of the car's fuses. In our prototype, the power consumption (without load on K2) amounted to just below 28 mA.

There appears to be no standard for the connector at he end of the cable of the active antenna. The same antenna is often available with different types of connectors. For testing we used an older type made by Trimble (39265-50) which has 5 meters of cable and an MCX connector. There are also active antennas with SMA, SMB, etc. So before you buy an active antenna, check that you can also obtain the corresponding chassis-mount connector.

(100808-I)







Gentle Awakenings

Programmed sunrise

By Aike Terjung (Germany)

In the natural world, our biological clocks are controlled by daylight. The light alarm clock described here imitates a natural sunrise to wake you gently from your slumbers in a natural manner. Start your day better by waking to the light instead of that horrid alarm clock.

We all know the unpleasant feeling of being rudely awakened from our sleep by the buzzing, beeping or ringing of an alarm clock. This often happens when you're in deep sleep, which frequently results in not feeling properly awake or groggy the whole day long. However, it doesn't have to be this way. If you sleep in the summer with the windows unobstructed, you often awaken spontaneously when it becomes light outside.

When you awaken in this natural, gentle manner, you also feel much better, and this feeling often persists all day. Obviously, it would be better to start your day with a simulated sunrise if the real thing isn't available.

Features

- Output connector for one or more dimmable 230 V / 115 V lamps (80 W max.)
- DCF77 radio time synchronisation
- Touch sensor for switching off the alarm
- ATmega168 microcontroller
- PCB and programmed microcontroller available from Elektor
- Firmware and source code available free from Elektor
- Can also be used on 115 V power grids

Light and time

The light alarm clock described here is built around a microcontroller and can switch and dim an existing 230-V lamp (or lamps) fitted with an incandescent bulb (normal or halogen). Although it might seem more logical to use power LEDs, there are good reasons for using an incandescent lamp. Aside from low cost (money and effort) and the possibility of using a lamp you already have on hand (such as a bedside lamp), the colour characteristics of incandescent lamps are better for this purpose. The colour of the light from an incandescent lamp varies when it is dimmed, gradually changing from a strong red hue to nearly white as the brightness is increased. This is similar to what happens at sunrise, and in any case it is more attractive than what you see with an LED dimmed under PWM control. If you wanted to simulate this effect with LEDs, you would have to use RGB LEDs or LEDs with several different colours. Things would be even more complicated if you also wanted to be able to continuously vary the brightness. The power savings that could be achieved by using LEDs are anyhow very limited because the alarm clock does not keep the lamp lit for hours on end.

In addition to a lamp, the light alarm clock naturally needs to know what time it is. A DCF77 receiver module is a good solution here, since it allows us to use a fairly simple, software-controlled time base in the microcontroller. There's no need for a real-time clock, since the microcontroller clock frequency (controlled by an external quartz crystal) is sufficiently accurate to allow it to manage for several hours without a proper DCF77 radio signal.

Table 1. Touch sensor functions		
State	**Touch duration**	**Action**
Display backlight off	< 1 second	Switch on display backlight for 1 minute
Lamp off; alarm not active	> 3 seconds	Switch on lamp at full brightness
Lamp on (including alarm phase)	> 3 seconds	Switch off lamp
Wake-up alarm	< 1 second	Switch off alarm (with optional snooze function)

Sensor buttons and dimmer

There's more to the alarm clock circuit than just a microcontroller. A display and buttons are necessary for the user interface. In the schematic diagram (**Figure 1**) of the alarm clock, you can right away see that the user interface takes the form of an inexpensive (and ubiquitous) LCD module with two lines of 16 characters (LCD1) along with four pushbutton switches (S1–S4), which are connected to the main circuit board (**Figure 2**) by plug-and-socket connectors (K7 and K8) and a length of 5-way cable if necessary. Each switch pulls an I/O pin of the ATmega168 (IC2) to ground. Pull-up resistors (R7–R10) provide defined high levels when the switches are in the quiescent state.

The alarm clock is controlled by five button functions in total. The fifth 'button' is a capacitive sensor based on an idea described on a German microcontroller forum [1]. The sensor consists of a metallic surface connected to the Sensor input (near R13), which is connected to an I/O pin (PC5) of the ATmega microcontroller via R13 and a protection network consisting of D2, D3 and R15. The microcontroller firmware (written in C in the WinAVR development environment) periodically switches the output of this pin from logic 1 (High) to logic 0 (Low) and then repeatedly samples the level on the I/O pin while it is returning to the High level. The capacitance of the signal line and the connected metallic sensor surface is charged via resistor R15, and the logic level changes back to logic 1 when the voltage crosses the switching threshold. The software can determine the capacitance of the sensor circuit connected to the I/O pin by measuring how long this process takes. If you touch the sensor surface, the capacitance increases and the charging time increases accordingly.
The author used the aluminium front panel of his light alarm clock as the sensor surface.

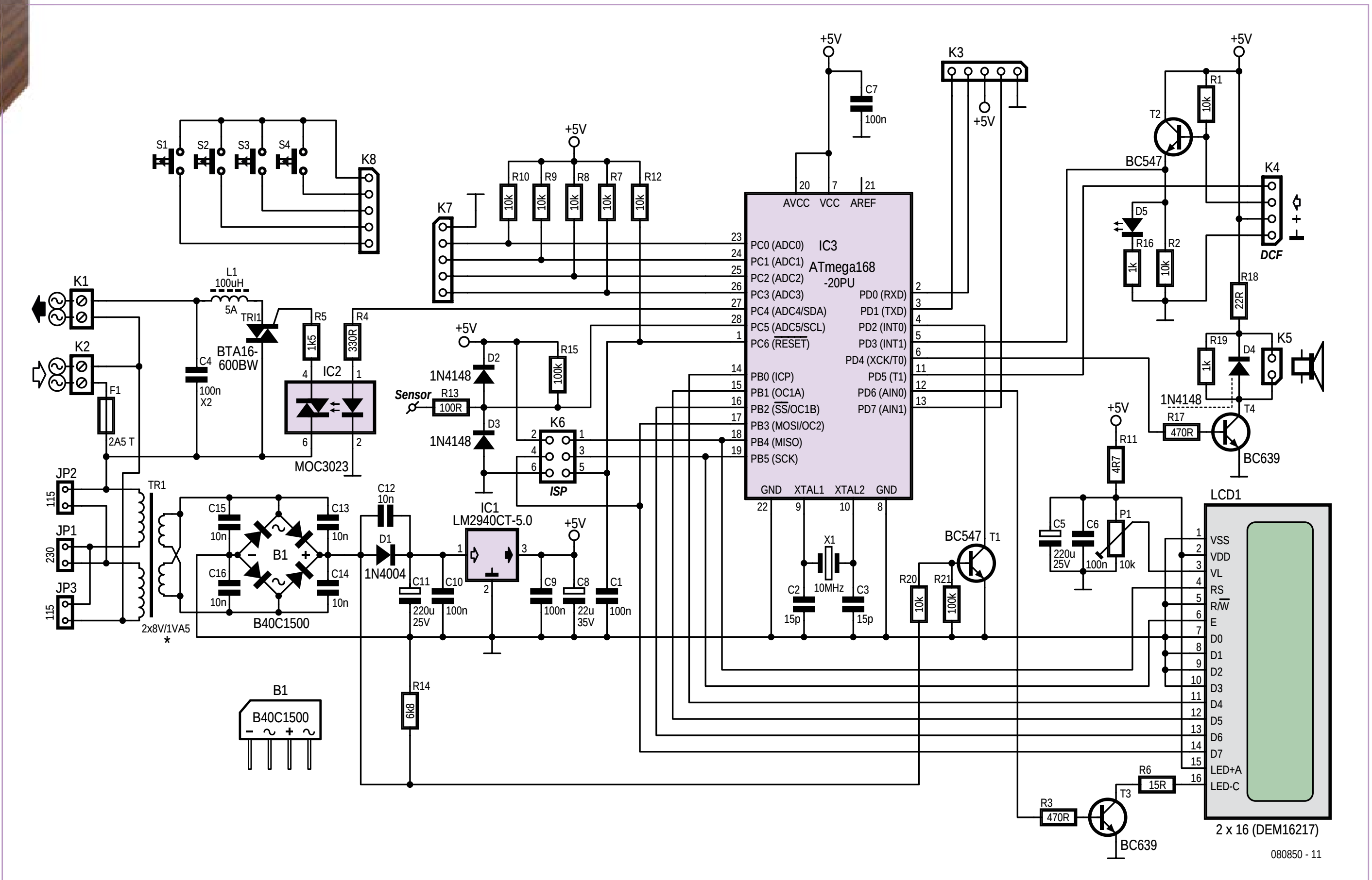


Figure 1. The circuit of the light alarm clock essentially combines the functions of a dimmer controlled by a microcontroller and a timer clock. Both of these functions are implemented in microcontroller firmware.

COMPONENT LIST

Resistors
R1,R2,R7,R8,R9,R10,R12,R20 = 10kΩ
R3,R17 = 470Ω
R4 = 330Ω
R5 = 1.5kΩ
R6 = 15Ω
R11 = 4.7Ω
R13 = 100Ω
R14 = 6.8kΩ
R15,R21 = 100kΩ
R16,R19 = 1kΩ
R18 = 22Ω
P1 = 10kΩ trimpot

Capacitors
C1,C6,C7,C9,C10 = 100nF, ceramic, lead pitch 5mm
C2,C3 = 15pF 2%, ceramic, 5mm lead pitch
C4 = 100nF, X2 class, 275VAC rated, 15mm lead pitch, 5mm wide
C5,C11 = 220µF 25V radial, 3.5mm lead pitch
C8 = 22µF 35V, radial, 2.5mm lead pitch
C12,C13,C14,C15,C16 = 10nF ceramic, 5mm lead pitch

Inductors
L1 = 100µH 5.4A triac suppressor coil (e.g. Murata Power Solutions type 1410454C)

Semiconductors
D1 = 1N4004
D2,D3,D4 = 1N4148
D5 = LED, low current, diam. 5mm, Kingbright type L-53LSRD
B1 = 40V 1.2A bridge rectifier, Semikron type SKBB40C1500L5B
T1,T2 = BC547B
T3,T4 = BC639
TRI1 = BTA16-600BW (e.g. STMicroelectronics BTA16-600BWRG)
IC1 = LM2940CT-5.0 (National Semiconductor)
IC2 = MOC3023 (Fairchild)
IC3 = ATmega168-20PU, programmed, Elektor # 080850-41 *

Miscellaneous
K1,K2 = 2-way PCB terminal block, lead pitch 7.5mm (0.3 in.)
K3,K4 = 4-pin SIL pinheader, 0.1 in. pitch
K5 = 2-pin pinheader, 0.1 in. pitch
K6 = 6-pin (2x3) pinheader, 0.1 in. pitch
K7,K8,K9 = 5-pin SIL pinheader, 0.1 in. pitch
LCD1 = 16-pin SIL pinheader, 0.1 in. pitch
S1,S2,S3,S4 = pushbutton, SPNO, e.g. Multicomp MCDTS6-5N
X1 = 10MHz quartz crystal, HC-49/S case, 50ppm, 12pF load capacitance, e.g. AVX HC49SFWB10000H0PESZZ)
F1 = fuse, 2.5A slow, 5x20 mm, with PCB mount holder and protective cap
TR1 = power transformer, 2x8 V sec. 2x115 V prim., 1.5VA (e.g. Block AVB 1.5/2/8) or 1x9 V sec., 1x230 V prim., 1.5VA (e.g. Block VB 1.5/1/9, see text)
LCD1 = 2 x 16 characters (DEM16217), e.g. Elektor # 030451-72
Piezo buzzer, e.g. Kingstate KPEG827, or 8Ω loudspeaker (min. 0.1W)
PCB # 080850-1 *

* Elektor Shop, www.elektor.com/080850

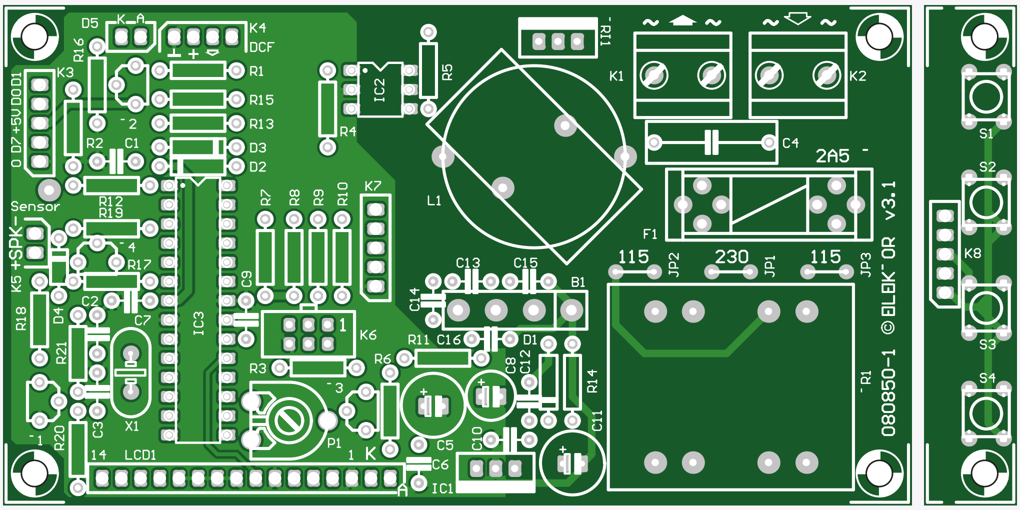


Figure 2. The PCB (available from the Elektor Shop) has a narrow strip for the four pushbuttons, which can be separated from the main PCB.

The functions listed in **Table 1** can be activated by touching the sensor surface. One of these functions is switching the backlight of the display module, which is controlled by pin PD6 of the ATmega microcontroller and transistor T3.

The 230 V or 115 V lamp connected to K1 (maximum power 80 W) is dimmed by a standard phase-control triac dimmer circuit, although here the trigger time of the triac (TRI1) is controlled by the microcontroller via an optocoupler (IC2) for mains isolation, instead of by the customary diac and potentiometer. The dimmer circuit is based on the 'Semitone Crystal' open source design [2]. However, in the alarm clock circuit we dispensed with an optocoupler for zero crossing detection and used a simple transistor (T2) instead.

Connections

We already mentioned that the 230 V (115 V) lamp(s) is (are) connected to K1 and the buttons are connected by K7 and K8. That leaves us with connectors K2 to K6. K2 provides the AC power connection to the alarm board (115 V or 230 V). An approved AC power cord with strain relief is connected to these terminals.

K3 is a serial port for programming and debugging. It is compatible with the FTDI USB to TTL adapter cable [3].

The DCF77 receiver module is connected to K4. This port is laid out for the well known Conrad Electronics DCF77 receiver module (order number 641113) shown in **Figure 3**. The data sheet and circuit diagram of this module are available on the Conrad website for reference. Only pins 1 to 3 of the DCF77 receiver module are used; they are connected to pins 1 to 3 of K4. The inverted output of the DCF77 module (inverted output) is not used. Pin 4 of K4 is connected to PD5 of the ATmega microcontroller and provides an Enable signal, which the author needed for a different module. DCF77 can be received across most of Central Europe and the UK.

A small loudspeaker (8 Ω) or a piezoelectric buzzer can be connected to K5. This acts as a sort of fail-safe for people who sleep with an eye mask or as an acoustic alternative to the light alarm. K6 is a six-way ISP port, which for example can be used in combination with the Elektor USB AVRprog interface [4] for programming and debugging.

Power supply

The mains transformer (TR1) on the PCB receives AC grid power via 115/230-V power connector K2. This transformer has two 115-V primary windings and can be configured for either 115 V or 230 V operation using jumpers. For 230 V operation, JP1 must be fitted and JP2/JP3 must be left open; this connects the two primary windings in series. For 115 V operation, leave JP1 open and fit JP2 and JP3 to connect the two primary windings to the mains voltage in parallel.

The two 8 V secondary windings of the transformer are connected in parallel. The rest of the power supply circuit is conventional, with a bridge rectifier, electrolytic capacitor and 5 V voltage regulator. The only unusual element is diode D1 between the bridge rectifier and storage capacitor C11. This allows the pulsating DC voltage at the output of the bridge rectifier to be tapped off for zero crossing detection by transistor T1 and I/O pin PD2 of the microcontroller, before it is smoothed by the capacitor.
The LCD module is also powered from the 5 V rail. Trimpot P1 provides a variable voltage derived from the 5 V supply voltage for adjusting the contrast of the display module.

Assembly

A programmed ATmega168 and the PCB for this project are available from the Elektor Shop (see the components list). Of course, you may also program the microcontroller yourself; the source code and hex file can be downloaded free of charge from the Elektor website [5].

The PCB consists of two parts. In addition to the main PCB for the light alarm clock, there is a small strip designed to hold the four pushbutton switches and a header (see Figure 2). No SMDs are used in this circuit, so you can solder everything the same way you did 30 years ago. As usual, you should pay attention to the orientation and/or polarity of the components. This applies in particular to the bridge rectifier, since devices with nearly the same shape but different pin configurations are available. The right one is shown in Figure 1.

Our European readers can fit a regular 230 V PCB mains transformer with single primary and secondary windings instead of the international version with dual primary and secondary windings, as long as it has a compatible pin configuration. In this case, **omit** the three jumpers. An example of a suitable 230 V transformer is given in the component list, and any similar EI 30 transformer rated at 1.3 or 2 VA with 230 V on pins 1 and 5 and 9 V on pins 7 and 9 can also be used.

Initial operation

AC line voltage (230 V or 115 V) is present on the PCB and on some components, so the light alarm clock must be fitted into an enclosure for protection against contact with live voltages before it is connected to the mains or put into service. The enclosure must comply with the applicable safety regulations. In this regard, see the electrical safety instructions published on the Elektor website at www.elektor.com/electrical-safety.

If the board is assembled properly and the microcontroller is programmed correctly, you will see the time of day ‘00:00:00’ on the display after you plug in the power cable. If you don’t see anything on the dis-

Figure 3. A DCF77 receiver module is used for precise time synchronisation. This module is based on the proven Temic DCF77 receiver IC.

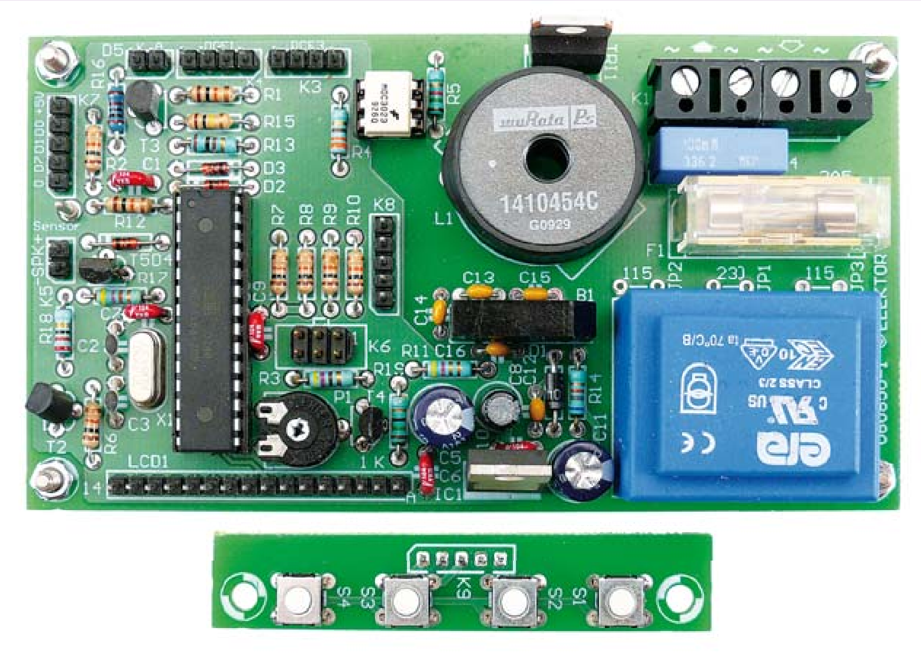


Figure 4. The fully assembled Elektor prototype board.

play, first check whether the contrast is adjusted properly (with P1).
Next the alarm clock tries to receive the current time from the DCF77 module. After it receives a correct data set, a small transmitting tower symbol is displayed next to the time to indicated that data is being received. However, it may take several minutes after this symbol appears before the correct time is displayed, since the program (to be on the safe side) waits until it has received two successive data sets before it updates the time.
If you now briefly touch the sensor surface, the backlight of the display will light up for approximately one minute.
If you touch the sensor surface for longer than around three seconds, the lamp connected to K1 is switched on. To switch it off again, repeat this action.
The front panel of the author's prototype acts as the touch sensor surface. It is connected to the sensor port of the PCB.

Control menu

The menu settings can be configured using the four buttons, whose assigned functions are listed in **Table 2**.
First press S2 to display the alarm clock menu. Then press S2 again to open the Alarm submenu, or press S3 to open the Settings menu. A user guide in the form of a detailed overview of the menus is available for downloading at [5]. The basic menu structure is as follows:

Alarm
Alarm active
Alarm time

Settings
Set alarm
- with light
- with sound
- Dimmer advance

Debug

Use the **Alarm** menu to enable the alarm and set the alarm time. When setting the alarm time, bear in mind that the simulated sunrise is programmed to end at the set alarm time, so the alarm clock starts the alarm process earlier than the set time. For this reason, the light will remain completely dark if the time interval until the set alarm time is shorter than the duration of the simulated sunrise (the 'dimmer advance' time).

Use the **Settings** menu to configure the basic alarm clock settings. In the 'Set alarm' submenu you can enable or disable the light alarm, enable or disable the supplementary acoustic alarm, and specify the duration of the light alarm phase. This alarm phase ('Dimmer advance') can be set in the 'Settings' menu. The default setting is 15 minutes, but you can also select 30, 45 or 60 minutes.

The **Debug** menu is displayed only when the alarm clock is operating in debug mode. Among other things, this menu shows the number of detected bits since the last start marker of the DCF77 signal and the last detected bit in this signal. If no information is displayed in Debug mode, there is a problem with DCF77 reception. This may be due to the location of the receiver or the antenna orientation, or it may be caused by misconnection of the receiver module or a defective receiver module.

Table 2. Menu button functions

Button	Function
S1	Back (return to previous menu item)
S2	OK (confirm)
S3	> (larger or upwards)
S4	< (smaller or downwards)

Your own ideas

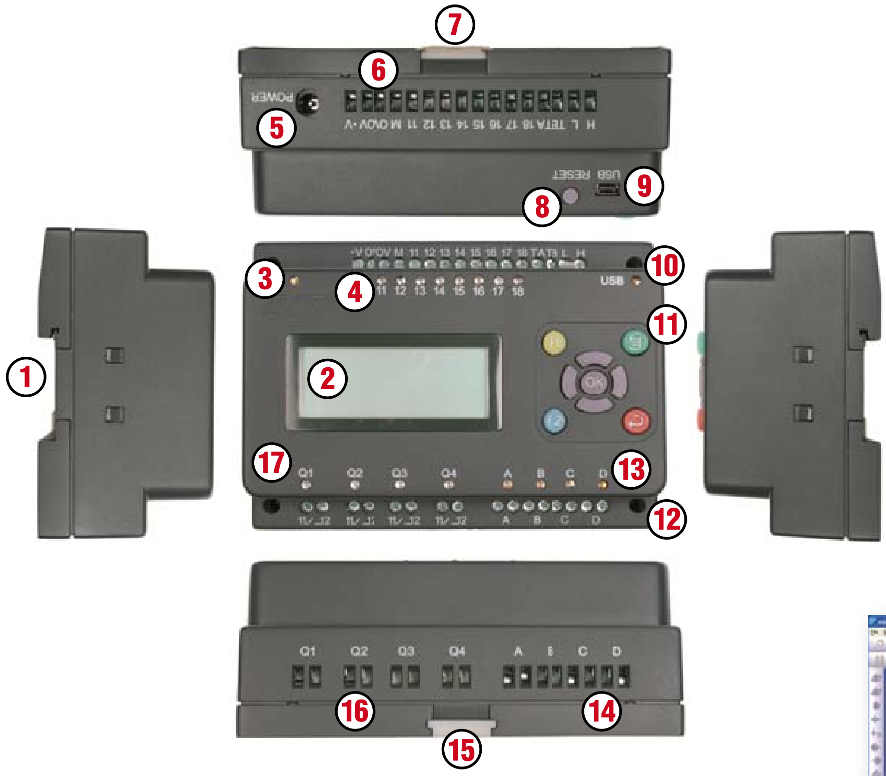
The software for this project, including the source code, can be downloaded from the Elektor website and was generated using a free C compiler (GCC), so there is nothing to stop you from modifying it as desired – for example, you could completely remodel the control menus and add new functions. One very nice change would be to replace the brutal acoustic alarm signal with a gentle wave noise signal whose amplitude increases gradually along with the increasing light intensity. For stubborn sleepyheads, you could also add a 'Maximum volume' menu item with the options 'Hurricane' or 'Jet fighter take-off'.

Finally, Elektor USA readers are encouraged to develop software based on their national time signal stations like WWV and WWVB. Let us know.

(080850-I)

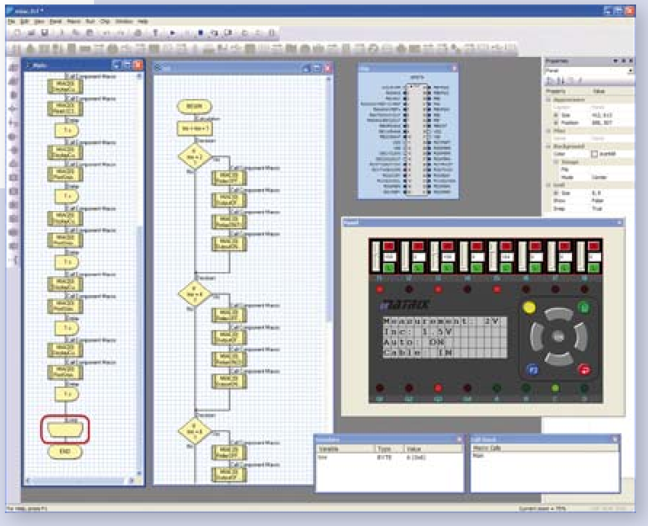
[1] www.mikrocontroller.net/topic/25045 (in German)
[2] www.engbedded.com/semitone
[3] www.elektor.com/080213
[4] www.elektor.com/080083
[5] www.elektor.com/080850

MIAC – the rugged PIC



What does it do?

MIAC is an industrial grade control unit which can be used to control a wide range of different electronic systems. It has a lots of applications in industrial control and automation and is perfect for hobbyist PIC projects that need a little oomph.



Flowcode *– the graphical programming language supplied with MIAC*

Key

1. Top hat rail mounting recess
2. 16 character x 4 line LCD display
3. Power LED
4. Input status LEDs
5. 2.1mm power jack
6. Screw terminal inputs
7. Top hat rail retainer clip (upper)
8. Reset / run switch
9. USB socket
10. USB transfer LED
11. Control keys
12. M3 mounting holes
13. Motor status LEDs
14. Motor output screw terminals
15. Top hat rail retainer clip (lower)
16. Relay output screw terminals
17. Relay output status LEDs

Using MIAC with FlowKit to give full IN-Circuit Debug with Flowcode

Benefits

- Flexible and expandable
- Easy to program with flowcharts, C or Assembly code
- Physically and electrically rugged

Features

- Programmable from USB
- Based on PIC18F4455
- Shipped with a free copy of Flowcode (worth £119)
- Compatible with third party C compilers
- 8 digital or analogue inputs
- 4 relay outputs @10amps
- 4 motor outputs @500mA with speed control
- 4 line LCD display and control keys

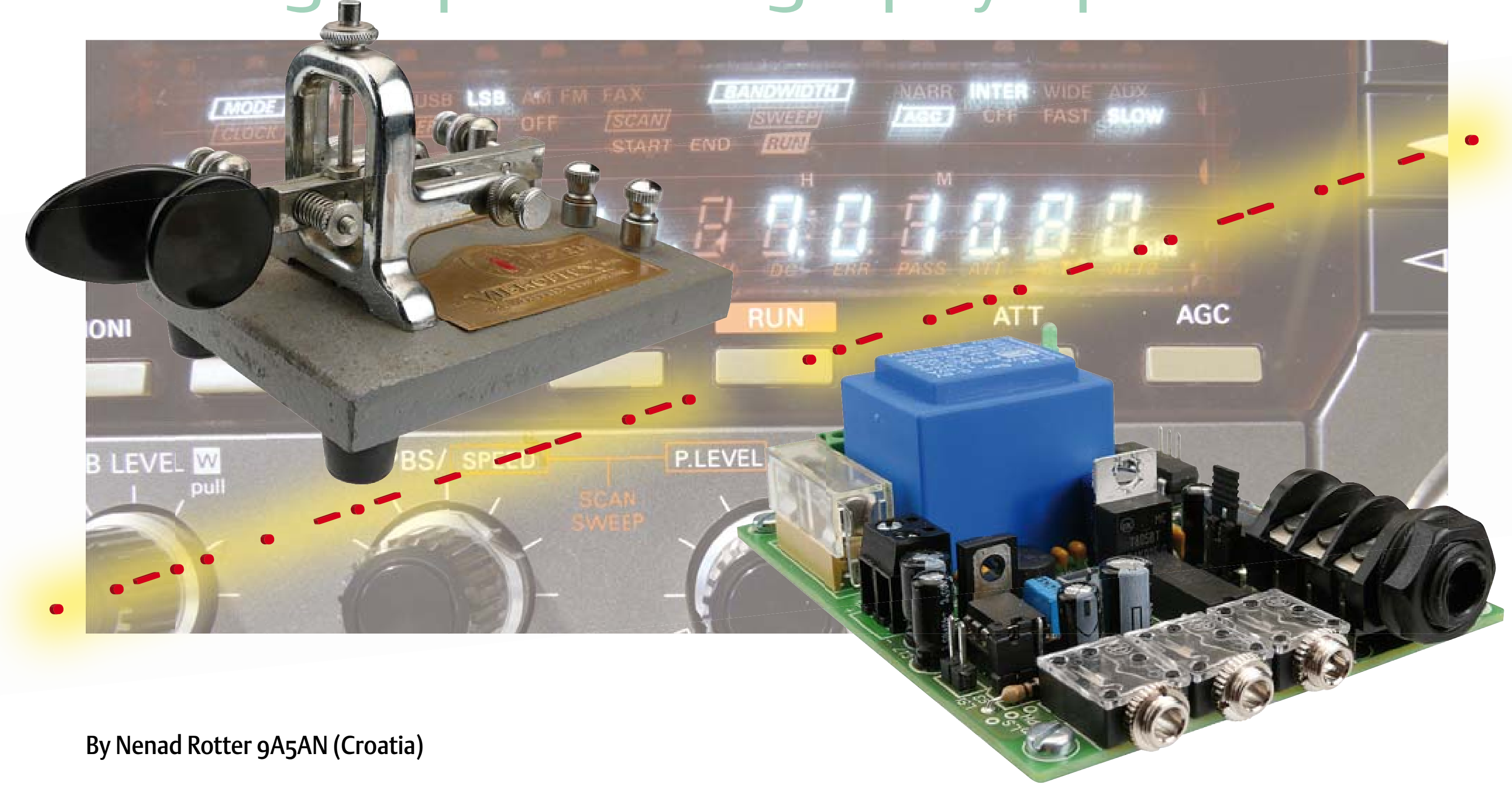
Create your own PIC projects with the advantageous MIAC Bundle. This package consists of a MIAC Module (in enclosure) and the graphical programming language 'Flowcode 3 for PIC' (Professional-Version)!

Bundle Price:
Only £154.00

Order now at
www.elektor.com/miac-bundle

Ultimatic CW Keyer

For high-speed telegraphy operators



By Nenad Rotter 9A5AN (Croatia)

There are still many radio amateurs who enjoy CW (which does not fully equate to 'Morse'). Those who are proficient at it achieve incredible speeds and typically like to use paddles enabling the dots and dashes to be generated via separate levers using thumb and finger. The circuit discussed here was developed specially for this type of key. It looks after a lot of time related issues such as the pauses between dots and words, fully supporting the renowned Ultimatic mode. A small monitor amplifier is also accommodated on the PCB.

The idea to develop this automatic CW (continuous wave) keyer arose after some discussions with fellow QRQ (QR-Quick = high-speed CW) telegraphers. They complained about commercial keyers on market being limited to Iambic mode only when the 'Ultimatic keyer' is what they were after. 'Ultimatic' represents a real squeeze technique keyer.

Having listened to ('copied') the complaints and wishes, the author summarised the main characteristics of a new keyer to be developed as follows:

- Ultimatic algorithm in case of squeeze keying;
- keying speed from 5 to 100 wpm (words per minute);
- high quality CW monitor;
- open to improvements and personal preferences.

As further considerations, the keyer should be simple to use, with no bells & whistles that a high-speed telegrapher can't be bothered about. Therefore, all parameters of the CW code are defined according to a standard and can't be changed. No functions for CW contest are implemented because nowadays radio amateurs are using dedicated programs on PCs for this purpose. The new keyer should be suitable for practicing to achieve correct pauses between characters and words, as well as encourage the telegrapher to use real 'squeeze' keying technique.

The Ultimatic algorithm

The Ultimatic algorithm is not generally known among radio amateurs, hence the short discussion below. The Ultimatic algorithm, as well as the 'Iambic' algorithm developed later, is based on squeezing two levers (or 'paddles') rather than straight closing and releasing of a traditional single lever key. The basic idea of squeeze keying is the possibility of *simultaneously* pressing the levers for dot and dash to reduce the total number of presses of the levers necessary for CW code keying. The resulting reduction of hand movements allows higher key-

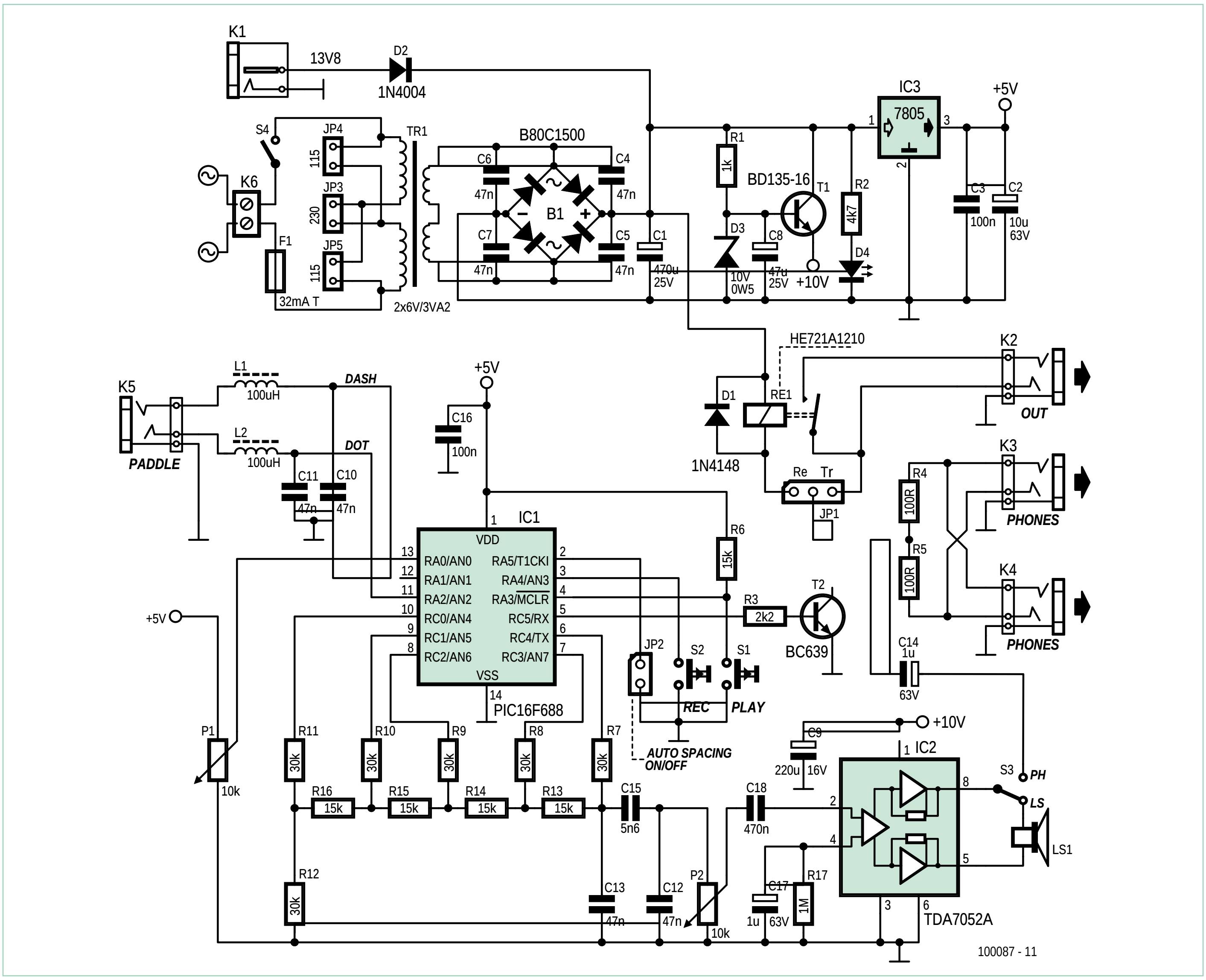


Figure 1. The Ultimatic CW keyer packs an impressive amount of intelligence into a small microcontroller, hence the compactness of the circuit. Anyone got the blueprint of the W6SRY 1955 tubed original handy?

Quick Features

- Operating mode:
  standard or Ultimatic in case of 'squeeze' keying
- Keying speed:
  5–100 wpm (words per minute), adjustable with the external potentiometer in increments of 1 wpm up to 36 wpm and thereafter in increments of 2 wpm
- Dot/Dash ratio: 1:3
- Pause between two elements: 1 dot
- Automatic generation of pause between characters / words (auto spacing); can be disabled
- Automatic pause between characters: 3 dots
- Automatic pause between words: 7 dots
- Frequency of monitor tone:
  600–2000 Hz, adjustable in increments of 5%
- Tone waveform for monitoring: sine, amplitude modulated
- Max. audio output power:
  0.15 watts (optional: 0.5 watts) (8 Ω speaker), external volume control
- Headphone connectors:
  2; connectivity with radio station's audio output
- Memory: one, capacity approx. 16 words
- TX output: transistor or reed relay
- Minimum pulse at the dot / dash input: 3 µs
- Power supply: Powerline (230/115 VAC) or 13.8 VDC from rig
- Power consumption: 0.3 W typical, max. 2 W

Boastware: Ultimatic algorithm advantages

To make the Ultimatic algorithm advantages more obvious, let's examine all characters. Letters E, H, I, M, O, S and T as well as numbers 0 and are 5 always keyed with one lever only and need not be considered here.

A single squeeze (both levers pressed and released at same time) is enough for the keying of letters A, B, D, J, N and W as well as numbers 1 and 6. In the case of the Iambic algorithm, these are only letters A, C and N as well as full stop (point), semicolon and plus.

All other characters can be keyed out with the lever of the initial element continuously pressed. In the case of the Iambic algorithm that's possible only for letters F, G, K, L, Q, R, U, V and Y as well as numbers 4 and 9

More than two lever presses are necessary for letter C as well as parenthesis, full stop (point), semicolon, slash, plus sign and comma.

Comparing by number of lever presses, the Ultimatic algorithm is more effective than the Iambic algorithm except for the letter C, full stop, semicolon and plus sign. However, the Ultimatic algorithm allows keying of all characters with true squeeze technique while with Iambic algorithm letters B, D, J, P, W, X and Z, numbers 1, 2, 3, 6, 7 and 8 as well as minus sign, question mark, parentheses, equal, slash, comma and exclamation mark can't be keyed out using squeeze technique at all because during keying it's necessary to release the opposite element lever, meaning these characters should be keyed with standard single-lever technique.

A	• ▬	N	▬ •	0	▬ ▬ ▬ ▬ ▬
B	▬ • • •	O	▬ ▬ ▬	1	• ▬ ▬ ▬ ▬
C	▬ • ▬ •	P	• ▬ ▬ •	2	• • ▬ ▬ ▬
D	▬ • •	Q	▬ ▬ • ▬	3	• • • ▬ ▬
E	•	R	• ▬ •	4	• • • • ▬
F	• • ▬ •	S	• • •	5	• • • • •
G	▬ ▬ •	T	▬	6	▬ • • • •
H	• • • •	U	• • ▬	7	▬ ▬ • • •
I	• •	V	• • • ▬	8	▬ ▬ ▬ • •
J	• ▬ ▬ ▬	W	• ▬ ▬	9	▬ ▬ ▬ ▬ •
K	▬ • ▬	X	▬ • • ▬	.	• ▬ • ▬ • ▬
L	• ▬ • •	Y	▬ • ▬ ▬	,	▬ ▬ • • ▬ ▬
M	▬ ▬	Z	▬ ▬ • •	?	• • ▬ ▬ • •

100087 - 12

ing speeds to be achieved with relative ease. All relative to the traditional up/down key, of course.

The squeeze technique and associated Ultimatic algorithm were developed by John Kaye, W6SRY. Back in 1955, Kaye published his article *The All-Electronic "Ultimatic" Keyer* in *QST* magazine. The all-tube keyer had an integrated twin lever paddle and was able to generate CW code according to the Ultimatic algorithm philosophy. Acceptance was slow and twin lever paddles did not appear widely on the market until around 1964.

The Iambic algorithm, known to almost every radio amateur, was worked out by John Curtis, K6KU. In 1968 he made a semiconductor keyer called the TK-38 which generated CW code based on the Iambic algorithm. In the following years, Curtis even developed a few integrated circuits supporting the Iambic algorithm. This is probably why the Iambic algorithm became widespread and today's transceivers have integrated electronic keyers built in that work according to the Iambic algorithm. However, this is also the main reason for so many telegraphers actually not using squeeze keying because the Iambic algorithm does not enable all CW characters to be keyed out with genuine 'squeeze' technique. The Iambic algorithm simply generates dots and dashes ('elements') alternatively when both levers are squeezed together. By contrast, 'Ultimatic' generates one dot and a series of dashes, or *vice versa*. Looking at the possible combinations for the levers being pressed:

- in case of both levers squeezed 'simultaneously', the element belonging to the lever first pressed will be generated, followed by continuous generation of the other element as long as the lever for it is held pressed. After that, the first element will continue to be generated as long as first element lever is held actuated.
- with one lever continuously pressed it is possible to insert one or more of the opposite elements at any desired moment.
- when both levers are released 'together', the last element to be generated is the one associated with the lever pressed last.

To make the principle easy to understand, here are few samples of CW code generation based on the Ultimatic algorithm:

? (··−−··) the dot lever was pressed first and generation of dots start. During the generation of the second dot, the dash lever was pressed also and generation of dashes starts. Both levers are pressed! When generation of the second dash starts, the dash lever is released and keyer is generating dots again. During the second dot generation, the dot lever was released also.

1 (·−−−−) the dot lever was pressed just before the dash lever. Then both levers are pressed. The keyer generates one dot and continues with generating dashes. Both levers can be squeezed and released simultaneously when the fourth dash starts, or the dot lever can be released at any time keeping only the dash lever pressed until the fourth dash starts.

Q (−−·−) the dash lever is pressed and the keyer generates dashes. While the second dash is being generated, the dot lever is pressed briefly. After the second dash is finished, one dot gets generated followed by

the last dash. At that time the dash lever is released also.
Since it is not required for a lever to be released to enable generation of the 'opposite' element, lever handling can be more relaxed with all types of overlap that are not time critical. There is just one 'must', and that's to stick to correct pressing of both levers in order to properly generate a CW for each character.

Schematic diagram

Looking at the schematic of the Ultimatic CW Keyer in **Figure 1**, the complexity is only apparent and mostly due to the passive components in the keying monitor. The heart of keyer is a Microchip microcontroller type PIC16F688. It was chosen because of its internal 8 MHz oscillator, 10-bit A/D converter, 14-pin housing and 12 I/O lines, the latter ample to connect the CPU to input and output signals as well as to an R-2R network for audio signal generation.

The squeeze paddle is connected to the keyer circuit through 6.3 mm jack socket K5. The dot and dash lever contacts are connected to microcontroller inputs trough RF filters L1/C10 and L2/C11 to eliminate any RF noise induced in the paddle cable. The keying speed is adjustable with potentiometer P1 which sets the voltage on PIC analogue input AN0. With 64 voltage levels available, 64 keying speeds in the range from 5 to 100 wpm have been defined. Resistors R7 through R16 (1% tolerance) together form a 5-bit R-2R network for sinewave audio signal generation. Capacitors C12 and C13 are part of a low-pass filter to suppress the sinewave sampling frequency at 32×f. C15 acts mostly as an attenuator.
Potentiometer P2 is the audio volume control. The CW monitor ('listen-in') amplifier is based around the TDA7052A which conveniently requires few external parts to operate. Output power in the configuration shown is about 0.15 watts. Those hard of hearing or wishing to impress a larger circle of bystanders may increase the output power to 0.5 W by trebling the value of C15 or connecting a 100 kΩ resistor between pin 4 of the TDA and the +5 V rail. Doing so will stretch the limits of the simple 10 V stabiliser around T1 though, and the TDA supply voltage is likely to drop below 6 volts at full output power.
Switch S3 allows the internal speaker to be silenced — the audio signal from the keying monitor is then redirected to connectors K3 and K4. These are wired in parallel to enable connection of transceiver headphone audio output simultaneously with the keyer. In this way you can listen to the receiver audio signal during reception as well as the keying monitor's audio signal during CW transmitting. Many transceivers have poor keying monitors built in, especially on higher CW speeds.
The keying output to the transceiver can be by transistor or by reed relay depending on the setting of 3-pin jumper JP1. Although the relay indicated here has an integrated back-emf diode, a separate diode D1 is provided in the circuit to accommodate pin compatible relays that might not have back-emf protection.

For most transceivers 'TX to ground' provided by the transistor will be okay but the reed relay can ensure a potential-free contact for special cases. The relay will also ensure galvanic isolation between keyer and transceiver, if needed. Bear in mind that the relay has to be *fast,* hence a type with a maximum contact close and release time of 1 ms is specified here.
The on-board power supplies allow the keyer to be powered either from your AC power socket (230 VAC or 115 VAC, set wire links JP3-4-5 as required) or from the 13.8 VDC taken from the transceiver or its power supply. Transistor T1 acts as a rudimentary voltage stabiliser and filter for the audio amplifier. The 5-V stabilised supply rail for the PIC micro is provided by a 7805 regulator, which may be replaced with a 78L05, the 16F688 being 'QRP' i.e. a few milliamps only. Green LED D4 is the ON indicator taking into account that the power switch is on the back side of the box.
The keyer has only two pushbuttons, S1 (PLAY) and S2 (REC). Their function will be explained below, as well as that of jumper JP2.

Software and Operation

For those with access to PIC programming tools, the software for the project is a free download from the Elektor website [1]. The CW keyer source code is fairly complex and can be divided into several sections. Only three will be discussed briefly, with notes on operating and configuring the keyer added. The **Operator's Chart** printed here is useful to have handy with the keyer. You may want to copy it and stick it on a piece of cardboard for easy reference.

Ultimatic CW Keyer Operator's Chart

CW
[REC] ⭳1s ⇢ 🔈 S
EE ⇢ RAM: [PLAY] ⇢ 🔈 R
■ : [REC] ⇢ 🔈 R

CW
[PLAY] ⇢ 🔈 - · - · ..
■ : [·] or [-]

f ≈
[REC] + [PLAY] ⇢ 🔈 ≈
f↑ : [·] *f*↓ : [-]
■ : [REC]

WPM info
[REC] + [-] ⭳ ⇢ 🔈 *n* . *n* .

RAM ⇢ EE
[REC] + [·] ⭳5s ⇢ 🔈 SETUP
[REC] ⇢ 🔈 MSG .. ⇢ 🔈 R

The purpose of the **auto spacing routine** is to insert correct time pauses between characters and words based on the CW standard. The routine executes in the main program loop and can be disabled with jumper J2. If J2 is open, the routine is enabled. Auto spacing can significantly improve the quality of your CW code and can help the telegrapher to get a good sense of timing for the specified pause. Proper pauses between characters and words are the biggest problem during keying.
A **SETUP routine** has been added to keyer

COMPONENT LIST

Resistors

R1 = 1kΩ
R2 = 4.7kΩ
R3 = 2.2kΩ
R4,R5 = 100Ω
R6,R13-R16 = 15kΩ 1%
R7-R12 = 30kΩ 1%
R17 = 1MΩ
P1 = 10kΩ potentiometer, mono, linear (not on circuit board)
P2 = 10kΩ, potentiometer, mono, logarithmic (not on circuit board)

Capacitors

C1 = 470µF 25V radial, electrolytic, lead spacing 5mm
C2 = 10µF, 63V, radial, electrolytic, lead spacing 2.5mm
C3 = 100nF 50V, ceramic, lead spacing 5mm
C4-C7 = 47nF 50V, ceramic, lead spacing 5mm
C8 = 47µF 25V, radial, electrolytic, lead spacing 2.5mm
C9 = 220µF, 16V, radial, electrolytic, lead spacing 2.5mm
C10-C13 = 47nF, ceramic or MKT, lead spacing 5mm or 7.5mm
C14,C17 = 1µF 63V, radial, electrolytic, lead spacing 2.5mm
C15 = 5.6nF 5%, polyester/MKT, lead spacing 5 or 7.5mm
C16 = 100nF 50V, ceramic, lead spacing 5mm
C18 = 470nF 63V, polyester, lead spacing 5mm

Inductor

L1,L2 = 100µH choke, axial, e.g. Bourns type 79F101K-TR-RC

Semiconductors

D1 = 1N4148
D2 = 1N4004
D3 = 10V 0.5W zener diode
D4 = LED, green, 3mm
T1 = BD135-16
T2 = BC639
IC1 = PIC16F688-I/P, programmed, Elektor # 100087-41, see [1]

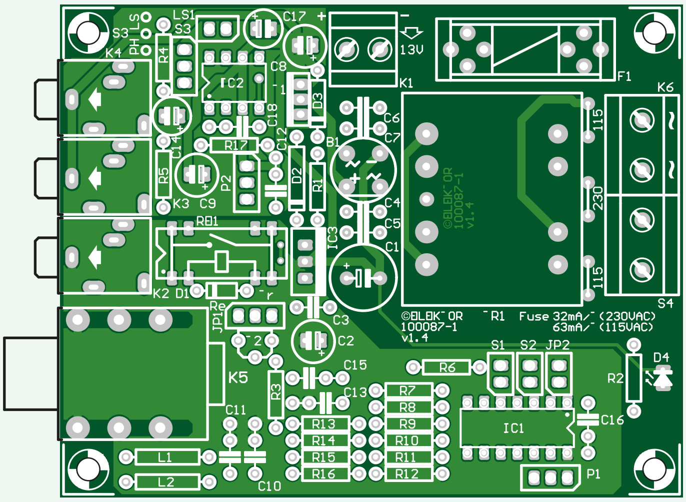


Figure 2. This small board was designed by Elektor labs to build the Ultimatic keyer. Note that controls like pots and switches are externally connected to the board. The copper track artwork is a free download from [1].

IC2 = TDA7052A/N2 ('A' suffix device must be used)
IC3 = 7805
B1 = 100V 1.5A bridge rectifier, Vishay General Semiconductor type W01G

Miscellaneous

F1 = fuse, 32mA (230VAC), 63mA (115VAC), slow blow, with PCB mount holder and cap
D4,S1,S2,JP2 = 2-pin pinheader, lead spacing 0.1'' (2.54mm)
D4,S1,S2 = 2-way socket SIL, straight, lead spacing 0.1'' (2.54 mm)
JP1 = 3-pin pinheader with jumper, 0.1'' (2.54 mm)
JP2 = 2-pin pinheader with jumper, 0.1'' (2.54 mm)
K1 = 2-way PCB terminal block, lead spacing 5mm
K2,K3,K4 = 3.5mm stereo jack socket, PCB mount, e.g. Lumberg 1503 09
K5 = 6.3mm (1/4'') jack socket, switched, 3-way, PCB mount, e.g. Cliff Electronic Components type S2BBBPCA
K6,S4 = 2-way PCB terminal block, lead spacing 7.5 mm, Camden Electronics CTB0110/2
LS1 = miniature loudspeaker, 1W / 8Ω (not on circuit board)
RE1 = reed relay, 12VDC, SPST-NO, e.g. Hamlin Electronics type HE721A1210.
S1,S2 = pushbutton, 1 make contact, e.g. APEM type 9633NVD with black cap type U482 (not on circuit board)
S3 = SPST (toggle) switch (not on circuit board)
S3,P1,P2 = 3-pin pinheader, 0.1'' (2.54mm)
S3,P1,P2 = 3-way socket, 0.1'' (2.54mm)
TR1 = AC power transformer, PCB mount, prim. 2x115 V, sec. 2x6V 2.3VA, e.g. Block type AVB2.3/2/6. Strap primary for local AC line voltage.
PCB # 100087-1, see [1]

software. It allows copying of the message saved in RAM to EEPROM, as CW keyer user could permanently save frequently used messages and make RAM memory free for temporary messages which will be automatically erased after keyer is switch off. The SETUP routine is also used for PIC oscillator frequency adjustment which will be explained below. Launching of the routine takes 5 s and the routine is active only 3 s, this is done to prevent accidentally start it. The SETUP routine is started by pressing REC and the dot lever simultaneously. After that, REC can be released but Dot lever must remain pressed for at least 5 seconds, until the keyer starts to key 'SETUP'. After that, REC] must be pressed briefly within 3 s. The keyer will key 'MSG' and the message saved in RAM will get copied into EEPROM. When done you will hear 'R' keyed out and the keyer is ready for normal use. The message saved in EEPROM can be overwritten with a new message at any time by repeating the procedure described. The message in EEPROM can be erased by copying previously erased RAM into it. Microchip guarantees one million write cycles to EEPROM.

Internal oscillator frequency adjustment. The PIC 16F688, like many other new types of PIC, has an internal oscillator with a default frequency of 8 MHz, and the use of an external quartz crystal is optional. Here, the accuracy of the internal oscillator is sufficient especially after frequency adjustment with the software tool. Frequency can be changed in increments of approximately 60 kHz and after adjusting the CW speed error will not exceed ±0.5%. The procedure for oscillator frequency adjustment is not too complicated and described in a free supplementary document [1].

Construction and testing

The circuit is built on an Elektor-designed printed circuit board of which the component overlay is shown in **Figure 2**. Only through-hole components are used, so assembling this board should not cause problems provided you work accurately as it has to be admitted the component arrangement is fairly dense in places. When gathering the components for the project, pay attention to the pin arrangement of jack sockets K2, K3, K4 (3.5 mm) and K5 (6.3 mm). Also note the size and lead pitch of C18 for which a space of about 3.5 mm is available on the board. The programmed PIC micro is preferably mounted in a 14-pin DIL socket.

The completed board is fitted in a metal enclosure, see **Figure 3** for an impression of the author's prototype. The jack sockets are lined up at the board edge to enable them to protrude from the front panel. The two pots and the two pushbuttons are connected to the board by way of 0.1-inch pinheaders and receptacles.
Care and attention should be given to the 230/115 VAC power connections to the board. All wiring between the board and external AC power switch S4 must be secure and rated at 250 VAC minimum. On/off switch S4 must be rated and approved for AC line voltage (230 VAC / 115 VAC) switching. Do not even dream of going sloppy here.
The 7805 and the BD135 both have an easy job and do not require a heatsink.
Ready-programmed PICs for the project are available from Elektor, see [1].
The 13.8 VDC input is by way of a screw terminal block because a zillion different connector systems exist on radio rigs.

The component list includes references to all 2-pin and 3-pin 0.1-inch pinheaders to controls and components mounted off the board — see, for example, potentiometer P1 which gets connected to the board with a 3-way combination of a SIL pinheader and mating socket.
Before installing it in its case, the assem-

Figure 3. Suggested mounting in a metal case (author's prototype).

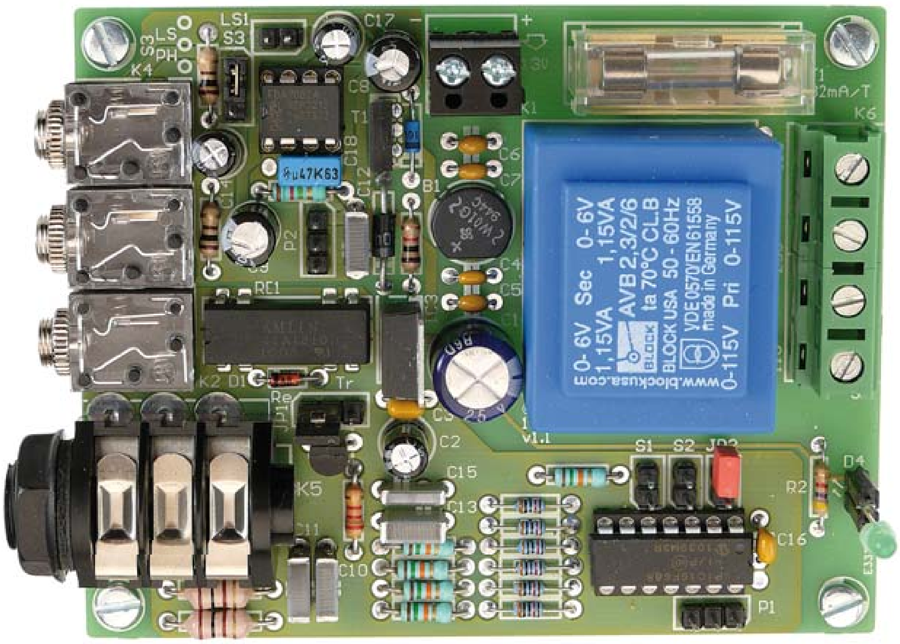


Figure 4. A close look at the finished and tested prototype of the keyer assembled by Elektor Labs. When in doubt, look closely at this photograph - HI.

bled board (**Figure 4**) can be given a quick test by running over some of the routines described in the Software and Operation section above. This requires all controls and a small loudspeaker to be connected, if necessary, in a temporary fashion.

Connecting to a radio station

Connection between keyer and radio station can be done in several ways depending on your desires and the type of radio station.

The first connection method involves connecting the keyer to the radio station with two cables. First, the 'control' cable connects the CW output (Out) with paddle input (Key) on the radio station and serves to activate the transmitter while the other, 'audio', cable connects the headphone jack (Phones) with an output for headphones (Phones) on the radio station to listen to the receiver and the keyer monitor on the same headphones. Using this method of connecting the keyer and radio station, grounds are joined via the connecting cable! This method is recommended because it ensures proper keyer grounding while galvanic isolation from the AC grid is afforded by the transformer.

If the radio station has a 13.8 V output for powering small external loads you are in luck as the keyer is switched on and off together with the radio station.

When the ground lines of both devices are connected together, the use of reed relay as the keyer output is pointless because the transistor output provides a better defined CW output (in terms of pulse timing).

When using the transistor output, the control and audio cables are identical and standard cables with 3.5 mm diameter stereo connectors at both ends. Such cables are used to connect a PC audio card and video monitor which have a built-in speakers and/or a microphone, and can be purchased ready-made in PC accessory outlets. It is important to emphasise that before connecting the keyer to a radio station, two parameters in the radio's menu need changing: (1) turn off internal keyer; (2) turn off the internal monitor.

The second way of connecting up is to provide full galvanic isolation between keyer and radio station. In this case, the audio cable should not be used, and the reed relay selected for the CW output. To ensure galvanic isolation of the CW output from the keyer ground, the reed relay contacts are connected to the tip and ring of a 3.5 mm stereo connector and the control cable should be made separately. In this case, it is desirable for the keyer ground to be connected to the ground rail in the radio shack (not: Radioshack).

The third method of connecting is a combination of first and second methods and is applicable in the event that the radio station requires a potential-free contact on its paddle input. This may occur at older radio stations which have not internal electronic keyer. The reed relay should be used for the CW output, and the control cable should be made as described above (second way of connecting) while the audio cable can be used as stated in the first method.

(100087)

Note: paddle keys used for circuit testing and photography kindly provided by Anton Klok, PA3AQV.

Internet Link

[1] www.elektor.com/100087

Figure 5. A collector's item and a QRQ delight, this vintage Vibroplex paddle keyer is easy to adjust to your personal preferences and rock solid on its cast iron base. Courtesy Anton Klok PA3AQV.

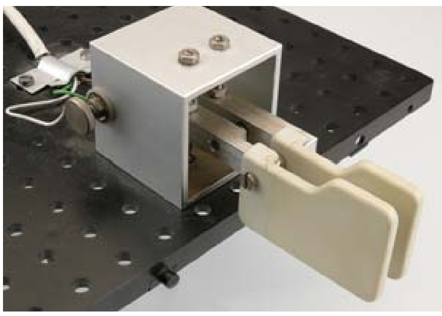


Figure 6. A low cost 'utility' paddle keyer mounted on a plastic base. Great for practicing and "getting up to speed" as some say. Courtesy Anton Klok PA3AQV.

Educational Expansion Board

With handy general-purpose peripheral functions

By Grégory Ester (France)

This expansion card, which is designed to be used with the Elektor ATM18 board, should come in handy for all sorts of projects. The combination provides a platform that is very suitable for both rapid prototyping and educational use.

Due to its suitability for educational applications, we have dubbed this board 'EduCard'. The basic idea is very simple: each subsystem of the circuit (see the schematic diagram in) provides a function that is often used in a wide variety of electronic systems. All inputs and outputs are brought out to PCB headers for easy access. The addresses, digital inputs and so on are configured by jumpers. Using a normal printer, you can easily generate an overlay for the PCB with appropriate signal names and I/O labels, which you can place on top of the board. The link between the EduCard and the ATM18 board requires only 19 leads. The ATM18 board was described in the April 2008 issue of *Elektor* and is available from the Elektor Shop (item number 071035-2). Once you have the ATM18 board, you can immediately start developing the software for your application. The modules fit together very nicely, and the result is something you can show with pride.

Mating modules

All of the PCB headers are located at the edge of the PCB, so they can easily be marked with the appropriate signal names or I/O labels. This way you can see exactly which lines are connected to the ATM18 board or to peripheral devices. For powering the board, you can choose from a PCB terminal strip (K2) for use with an 8–12 V power source, a male PCB header (K1) or female PCB headers (K3 and K4), all three of which can be used with a 5 V power source. If you aren't overly fond of cutting and stripping short lengths of wire and you're tired of seeing them break after being used just a few times, we have a handy tip for you. You can buy ready-made breadboard jumpers from Sure [2] and save yourself time and trouble. They are available in both male and female versions (). All of the connections necessary to use this board properly

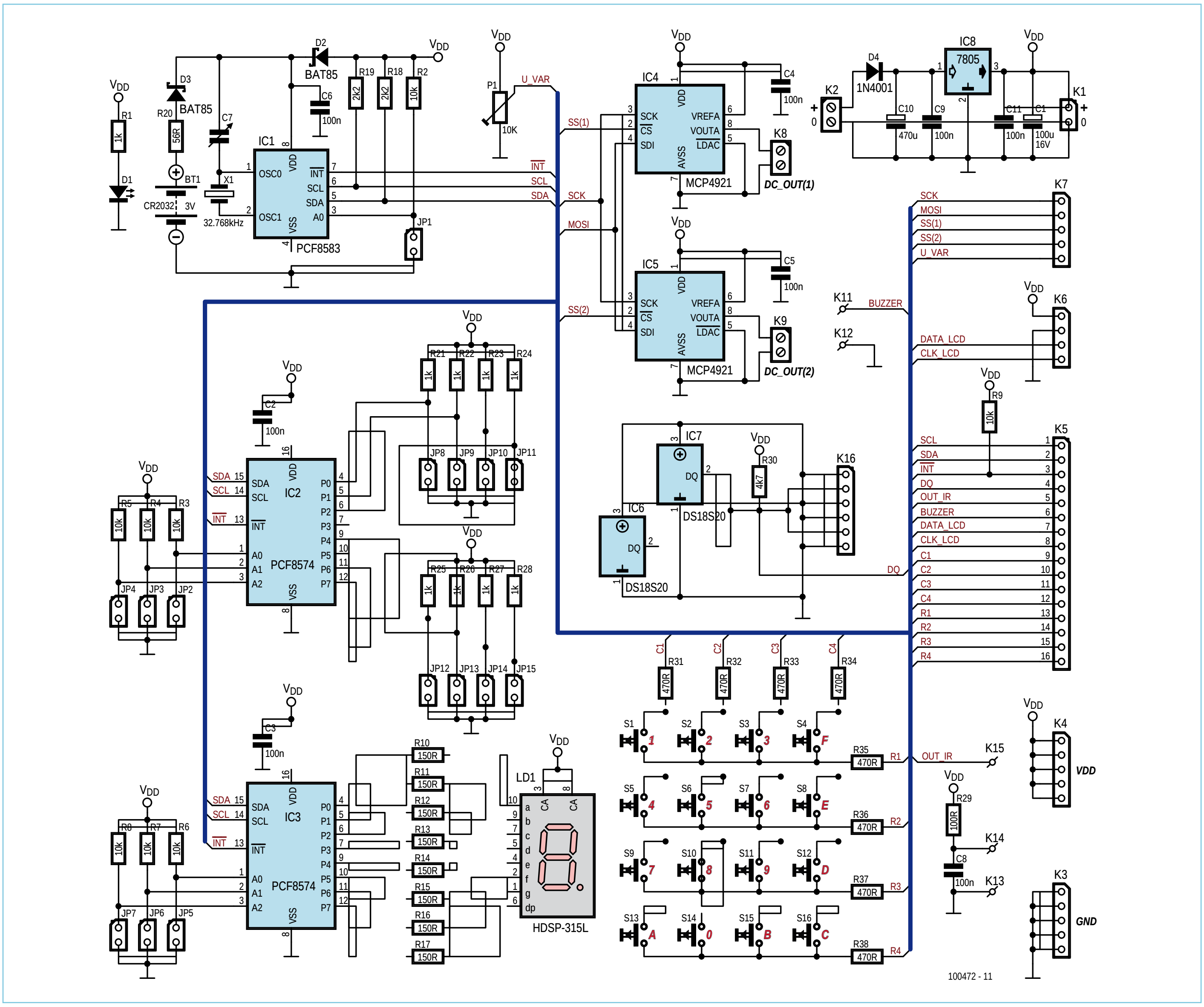


Figure 1. EduCard schematic diagram.

with the ATM18 board are listed in Table 1 for easy reference.

Just one wire

The 1-Wire bus was developed by Dallas Semiconductor and Maxim; it also goes by the name 'MicroLAN bus'. The nice thing about this bus is that you can connect a nearly unlimited number of I/O devices to your board with just one twisted-pair cable, which has a maximum length of several dozen metres. This is what is called a 'multi-drop' system. Under certain conditions, the bus can be extended as far as 300 metres (1,000 ft.) or so. Each device on the bus has a unique 65-bit identification number. Another handy feature is that the ID number is marked on the device package, so you always know where to find it. The 1-Wire devices transmit their digital data over the bus. The device identifier is protected by a cyclic redundancy check (CRC) code to prevent address errors on the bus.

A free API called 'TMEX' is available from Dallas/Maxim. You can use the attractive iButton Viewer user interface to access and program all 1-Wire devices on the bus, although in this case the ATM18 board looks after this task for you. 1-Wire devices that do not draw very much current can take their operating power directly from the bus. This is called 'parasitic power mode', and it utilises the fact that the signal level on the bus is often in the High state, so a current of a few milliamperes flows through the bus lead. The two sensors shown in operate in this mode. Connector K16 is provided to allow you to connect additional 1-Wire devices. Although individual 1-Wire devices usually draw less than 100 µA, there is nat-

Features

- 2 temperature sensors on 1-Wire bus
- 2 DACs with SPI interface
- RC5 sensor
- Matrix keypad
- 7-segment LED display
- 8 digital inputs on I2C bus
- Real-time clock
- Potentiometer
- 5-V supply voltage

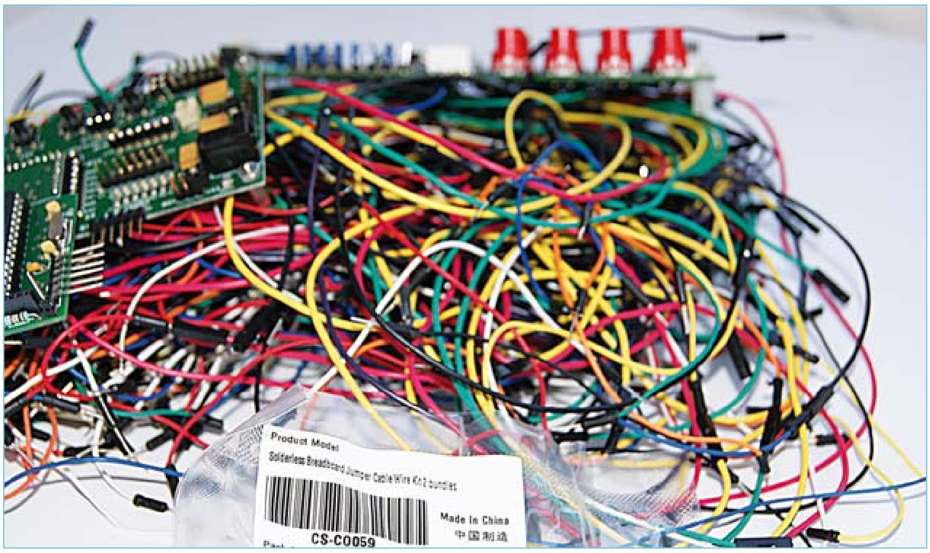


Figure 2. Ready-made breadboard jumpers.

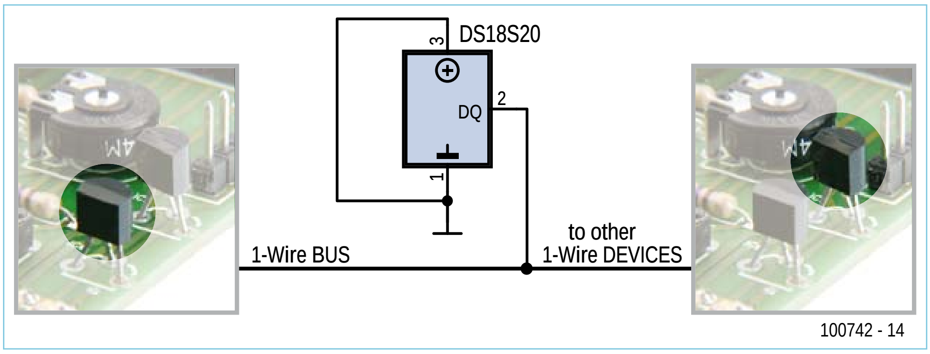


Figure 3. Basic operating principle of the 1-Wire bus.

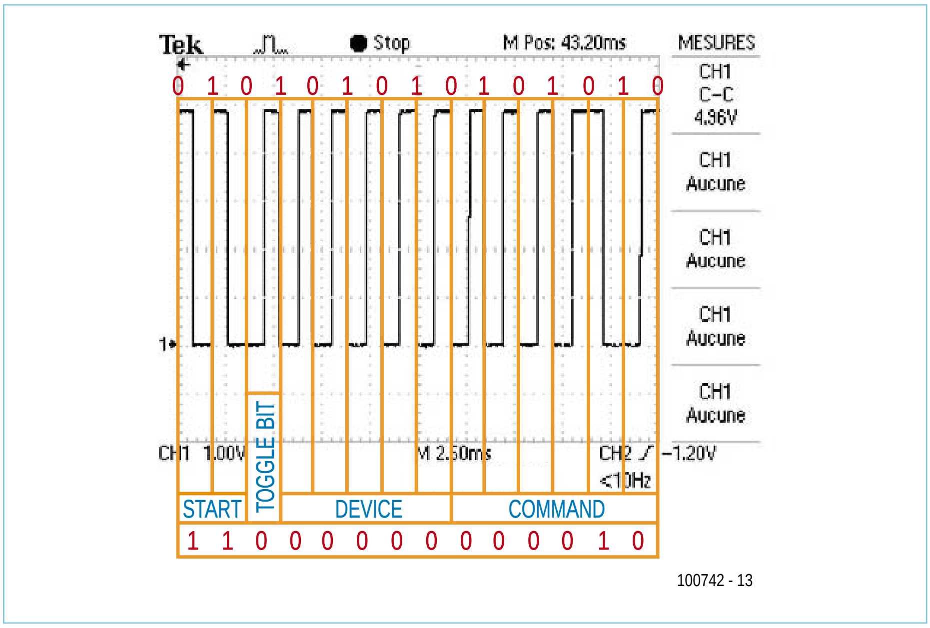


Figure 4. RC5 frame format.

urally a limit to how many devices operating in parasitic power mode can be connected to the bus. It is always possible to power some of the devices on the bus from a separate 3-V or 5-V power supply. Two DS1820 temperature sensors are present on the board as standard. Here we should mention that the DS18S20 as opposed to the DS1820, normally takes around 500 to 750 ms to convert a temperature reading to a bus signal.

Everyone on the bus

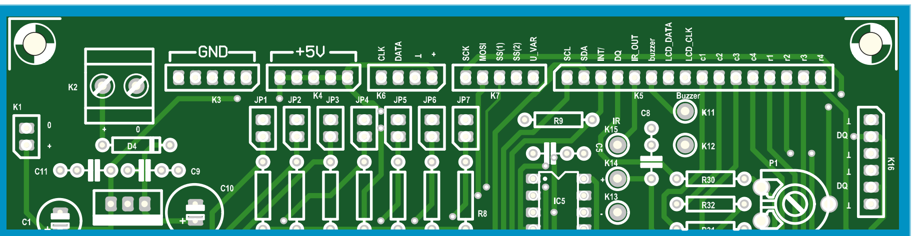
The Inter Integrated Circuit Bus, usually designated 'I^{2}C bus', was developed in the early 1980s by Philips for use in consumer electronics and home automation systems, in particular to provide a convenient way to link the various circuits in modern television sets to a microcontroller. Atmel and some other companies call this system 'Two Wire Interface'. The I^{2}C bus is a synchronous bus with two leads plus ground. One lead is designated 'SDA' and carries the data, while the other lead is designated 'SCL' and carries the clock signal. Addressing is used to ensure that the data arrives where it is supposed to go.

The EduCard has three components connected directly to an onboard I^{2}C bus: two PCF8574 devices and one PCF8583. The PCF8574s provide two 8-bit digital I/O ports (one each) for general purpose use. One port drives a seven-segment display, while the other can be configured and used as desired. It can be accessed via eight jumpers (JP8–JP15). The PCF8583 contains a real time clock and calendar and is equipped with a backup battery (a CR2032 3 V lithium cell fitted in battery holder BT1). The appropriate address assignments are listed in . Note that there is also an 'A' version of the PCF8574, which uses a different addressing scheme. Its most significant nibble is set to a fixed value of '0011' binary (hex 7x).

Invisible remote control

The RC5 standard for infrared data transmission is also a Philips invention. The fourteen data bits are biphase coded (Manchester coded) and available on pin 5 of connector K5. If you have a universal remote control unit programmed for controlling a Philips television set (TV1 mode), pressing the '2' button will cause

Table 1.
Connectors, jumpers and their functions.



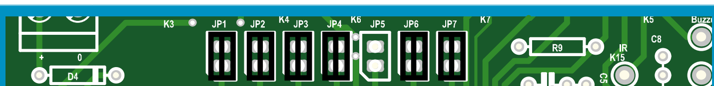
Function	Connections	Description
Power	K1, K2, K3, K4	5 VDC or 8 - 12 VDC external source; indicated by LED D1
Keypad	K5 pins 9–16	C1, C2, C3, C4, R1, R2, R3, R4
I2C PCF8583 (IC1): RTC	K5 pins 1–3	SCL, SDA, INT
	JP1	Address: on: A0 = 0; off: A0 = 1
	K10	CR2032 button cell
I²C PCF8574 (IC2): 8 digital inputs	K5 pins 1–3	SCL, SDA, INT
	JP2, JP3, JP4	Address: A0, A1, A2
	JP8–JP15	8 digital inputs: P0_E–P7_E
I²C PCF8574 (IC3): 8 digital outputs	K5 pins 1–3	SCL, SDA, INT
	JP5, JP6, JP7	Address: A0, A1, A2
		The states of the 8 outputs are shown by 7 segments and decimal point of the 7-segment display.
SPI MCP4921 (IC4 and IC5): two DACs	K7 pins 1–4	SCK, MOSI, SS(1) (IC4), SS(2) (IC5)
	K8 and K9	DC_OUT1, DC_OUT2 (2 analogue outputs)
1-Wire DS1820 (IC6 and IC7): two digital temperature sensors connected in parasitic power mode	K16 pins 1–6	GND, DQ, GND, GND, DQ, GND
	K5 pin 4	DQ
RC5: infrared receiver	K13, K14, K15	TSOP2236 to GND, VCC and OUT_IR
	K5 pin 5	OUT_IR output
Buzzer	K11 and K12	Intended for buzzer (+ and GND)
	K5 pin 6	Buzzer input
Potentiometer	K7 pin 5	Output range 0–5 V
Two-line Elektor LCD module	K6 pins 1–4	VDD, GND, DATA_LCD and CLK_LCD for driving the LCD module; DATA_LCD and CLK_LCD are available on K5 pins 7 and 8

the waveform shown in to appear on the output connected to pin 5 of connector K5. The first two bits always have a value of ‘1’ and are used for synchronisation. They are followed by a toggle bit, which changes state when a button is pressed (or pressed again). This means that the toggle bit stays the same as long as a button is held pressed. The toggle bit is followed by five address bits to select the device that should execute the command, and finally six bits corresponding to the pressed button (in this case ‘2’).

Analogue outputs

The EduCard has two 12-bit D/A converters implemented with type MCP4921 ICs. The analogue outputs are available on connectors K8 and K9. The converter ICs are configured as bus slaves and are enabled by pulling their Chip Select (CS) inputs Low. Data from the microcontroller is clocked into the Master Out Slave In (MOSI) input of each converter by the clock signal on the SCK line. The data is formatted as 16-bit words. The first 4 bits contain configuration information, while the remaining 12 bits contain

Table 2.
I²C device addresses.



Component									ADDRESS
PCF8583 (IC1) : RTC	1	0	1	0	0	0	JP1	$\overline{RW}$	$A0
PCF8574 (IC2) : 8 inputs	0	1	0	0	JP4	JP3	JP2	$\overline{RW}$	$40
PCF8574 (IC3) : 8 outputs	0	1	0	0	JP7	JP6	JP5	$\overline{RW}$	$42

Table 3. Connections between EduCard and ATM18 board.

Function	Educard	ATM18
Two-line LCD module	DATA_LCD, CLOCK_LCD	PB2, PB1
Matrix Keyboard	C1, C2, C3, C4, R1, R2, R3, R4	PD0, PD1, PD2, PD3, PD4, PD5, PD6, PD7
Buzzer	BUZZER	PC5
DS1820 (1-Wire)	DQ	PC4
RC5 (infrared)	OUT_IR	PC3
I^{2}C	SDA, SCL	PB0, PC2
SPI	SS(1), SS(2), MOSI, SCK	PC0, PC1, PB3, PB5

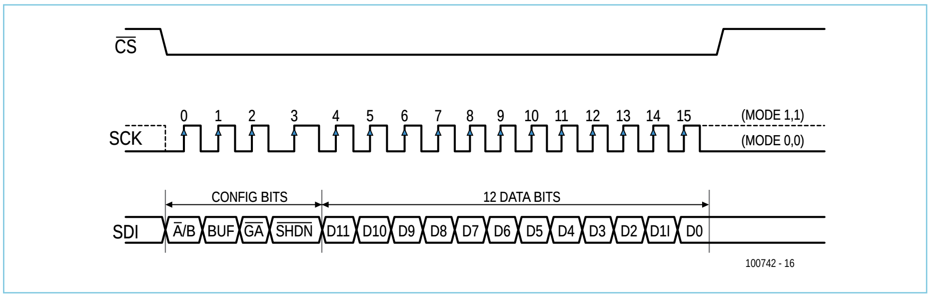
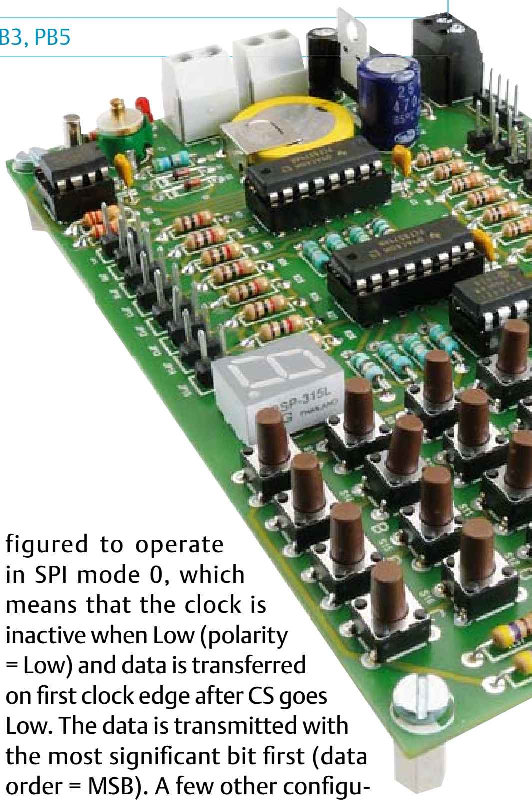


Figure 5. Two successive bytes on the SPI bus.



the value to be converted (see).
An SPI link supports four different operating modes depending on the values of two parameters. The first parameter determines the clock polarity (active High or active Low), while the second parameter determines whether data is clocked in or out on the first edge or the second edge of the clock signal when after the CS input goes active (Low). Here these converters are configured to operate in SPI mode 0, which means that the clock is inactive when Low (polarity = Low) and data is transferred on first clock edge after CS goes Low. The data is transmitted with the most significant bit first (data order = MSB). A few other configuration settings are:

/SHDN = 1	Disable sleep state
A/B = 0	Write data to DAC_A; there is actually no other choice with only one DAC per IC
BUF = 0	No buffer for V_{REF}
/GA = 1	Gain = 1; $V_{OUT} = (V_{REF} \times 1 \times D)/4096$

If D = 100011111111 (binary) = 2303 (decimal) and V_{REF} = 5 V, the voltage at the output of the converter should therefore be 2.811 V.

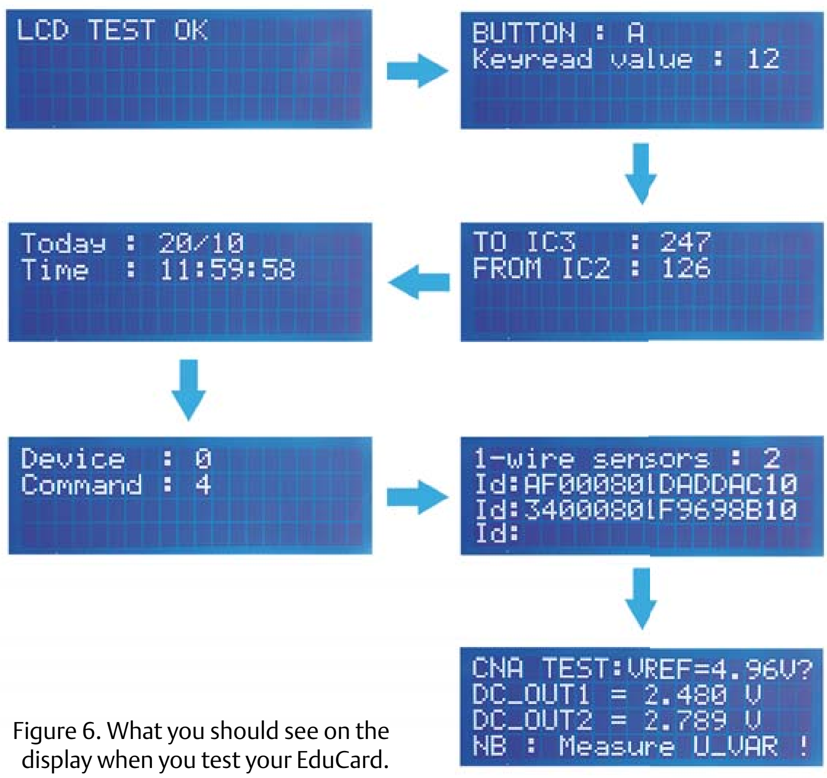
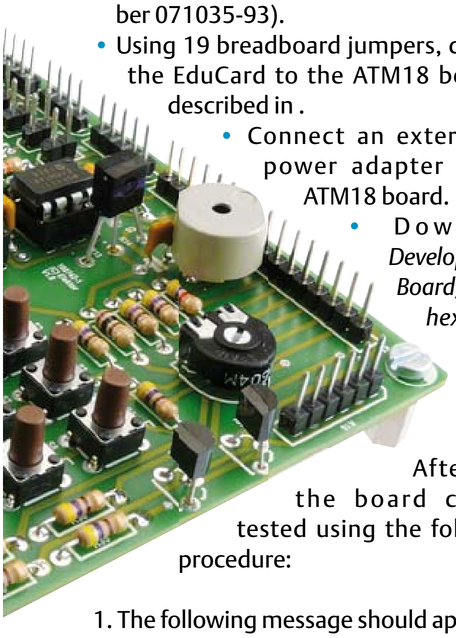


Figure 6. What you should see on the display when you test your EduCard.

The glue

The firmware is the glue that bonds all these components together. We generated the firmware using BASCOM-AVR 2.0.1.0, which has a solid track record.
The firmware includes test routines for all of the components on the EduCard. shows what you should see on the display after power-up. Use the following procedure to prepare for testing the EduCard:



- Connect the LCD module to K6 and the buzzer to K11/K12 on the EduCard (the LCD module was described in the May 2008 issue of *Elektor* and is available from the Elektor Shop under item number 071035-93).
- Using 19 breadboard jumpers, connect the EduCard to the ATM18 board as described in .
- Connect an external AC power adapter to the ATM18 board.
- Download *Development_Board_Test.hex* to the flash memory.

After this, the board can be tested using the following procedure:

1. The following message should appear on the LCD: 'LCD TEST OK'.
2. Matrix keyboard test: press 'F' to proceed to this test.
3. PCF8574: the segments of the seven-segment LED should light up in sequence. If the value read is '255', which means that jumper positions JP8 and JP9 are both open, the routine proceeds to the next test.
4. PCF8583: the display shows the date (DD/MM format) and time (e.g. 20/10 11:59:55). At 12:00:00 the routine proceeds to the next test.
5. RC5 mode TV1: press the '0' button on the remote control unit (which must be configured in TV1 mode) to proceed to the next test.
6. 1-Wire: the serial numbers of the two temperature sensors are displayed. Press '8' on the remote control to proceed to the next test.
7. DAC via SPI: write '2048' to DAC1 and '2303' to DAC2. With V_{REF} = 4.96 V, the measured values should be 2.480 V on K8 and 2.789 V on K9.
8. Finally, use a voltmeter to measure U_VAR. Adjust P1 to vary the voltage.

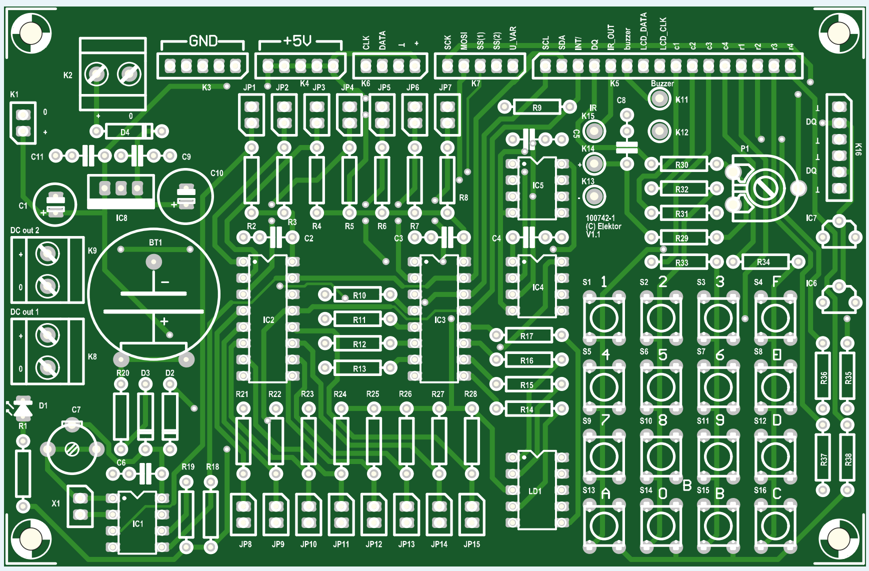
COMPONENT LIST

Resistors
R2–R9 = 10kΩ
R18,R19 = 2.2kΩ
R20 = 56Ω
R1,R21–R28 = 1kΩ
R10–R17 = 150Ω
R31–R38 = 470Ω
R29 = 100 Ω
R30 = 4.7kΩ
P1 = 10k horizontal, trimpot

Capacitors
C1 = 100µF 16V radial
C2–C6,C8,C9,C11 = 100nF
C7 = 22pF trimmer or 22pF capacitor
C10 = 470µF 25V radial

Semiconductors
D1 = LED, low current, green
LD1 = 7-segment display, Avago type HDSP-315L
D2,D3 = BAT85, Schottky diode
D4 = 1N4001
IC1 = PCF8583
IC2,IC3 = PCF8574
IC4,IC5 = MCP4921
IC6,IC7 = DS18S20+
IC8 = 7805

Miscellaneous
X1 = 32.768kHz quartz crystal
CR2032 Lithium button cell, 3 V
S–S16 = pushbutton, PCB mount, push to make, e.g. SPNO-B3S series (Omron), Farnell # 118-1016
K2,K8,K9 = PCB solder pins or 2-way PCB terminal block
K1 = 2-pin pinheader, lead pitch 0.1'' (2.54mm)
K3,K4,K7 = 5-pin socket strip, lead pitch 0.1'' (2.54mm)
K6 = 4-pin socket strip, lead pitch 0.1'' (2.54mm)
K5 = 16-pin socket strip, lead pitch 0.1'' (2.54mm)
K10 = PCB mounting for CR2032 cell
K11-K12 = DC buzzer, 5 V/4 kHz
K13-K14-K15 = TSOP2236 or equivalent
K16 = 6-pin socket strip, lead pitch 0.1'' (2.54mm)
JP1–JP15 = 2-pin pinheader with jumper, lead pitch 0.1'' (2.54mm)
3 pcs 8-pin DIL IC socket
2 pcs x 16- pin DIL IC socket
PCB, # 100742-1 (see [1])



If your EduCard passes all of these tests, it is working properly.

(100742-I)

Internet Links

[1] www.elektor.com/100742
[2] www.sureelectronics.net/goods.php?id=841

Geolocalization without GPS

Where am I? Where am I headed?

By Clemens Valens (Elektor France Editor)

These days, the simplest way to find out your geographical position is to use a GPS receiver or 'satnav'. A GPS (let's drop the 'receiver') is accurate and works anywhere in the world. GPSs are getting smaller and smaller and performing better and better, and new applications are constantly being found for them. But in spite of how powerful it is, the GPS is not the solution to every geolocalization problem. Where the signals from the GPS satellites can't get through properly, like indoors or in places surrounded by tall buildings, GPS receivers won't work correctly. Luckily, there are other solutions.

As we've often seen in TV detective series, it is indeed possible to find someone's position using their mobile phone. Knowing the positions of the cellphone network towers ('repeaters') with which it is in contact, we can find out roughly where the telephone is. And if these repeaters are able to compare between them the strength or the arrival time of the phone signals, it's even possible to get a quite accurate estimate of the position.

It works the other way round too. If the mobile phone has a database containing the positions of the repeater stations, it can calculate its own position using the signals transmitted by the repeaters nearby. The operators take advantage of this technique to offer automatic pedestrian guidance or local information services.

And what works with mobile phones and repeater stations can also be used with other wireless communication systems like Wi-Fi, ZigBee, or Bluetooth networks. GPS works all over the world — it's a global system (remember that GPS stands for Global Positioning System); in the same vein, a positioning system using a local network is called a Local Positioning System or LPS.
In this article, we're going to be taking a look at some systems that make it possible to locate an object (or person) at any moment, i.e. in real time, and anywhere within the area covered by the cellphone network. This type of system is known as Real Time Location System or RTLS (in this context, real time means periodically). Hence this excludes those systems that use, for example, RFID tags or barcodes to track the position of an object or beacon systems that make it possible to find an object by means of a mobile beacon detector.

A short history of geolocalization

Before launching into a description of LPS, its worth taking a little trip back into history, for the navigation techniques we use today were developed for the first mariners who sailed the seas and oceans.
Until the 15th century, sea journeys were almost always coastal: they would sail from port to port without every getting too far away from the coast. They navigated by observing the stars, the wind, the sea, the land, and the behaviour of birds and sea mammals. Basic tools like the star chart (used by the Arabs) or a wind rose (in the Mediterranean) made it possible to formalize good practice a little.

In the Northern hemisphere, the Pole Star allows "constant latitude" navigation; in the Southern hemisphere, they managed using other stars and constellations. Then the first instruments capable of measuring the angle of a star made an appearance: the kamal, the cross-staff ('Jacob's staff'), the nautical astrolabe, the quadrant, the octant, and lastly the sextant. These instruments make it possible to calculate the latitude with suitable precision.
By the late 12th century, the lodestone was already being used to find magnetic North and thereby deduce the ship's heading. By adding a compass card to it, it becomes a real compass, making it possible to read the ship's heading off directly.
Speed measurement arrived with the invention of the ship's log. These two elements, heading and speed, allow dead-reckoning navigation, but this still wasn't accurate enough for longer voyages.
In 1759, an Englishman, John Harrison, invented the marine chronometer, capable of keeping accurate time during long months aboard a ship. This allowed improved accuracy for these navigational approaches and significantly reduced the risk of running aground. With this kind of timepiece, you are able to measure the longitude by using the principle of time differential.
Later, the accuracy of these methods was improved and calculation methods refined. First employed during the First World War, the gyroscopic compass made it possible to get around the difficulties encountered with both the declination of the Earth's magnetism, and the influence of metal masses present aboard ships, which distorted and complicated the measurements.
The Second World War led to the emergence of devices exploiting radio waves, like radar and so-called 'hyperbolic' radio navigation systems like GEE, LORAN, and DECCA. These made possible an accuracy varying between a few metres and a few kilometres. Then they in turn were supplanted by more accurate satellite positioning systems.

The first GPS satellite was launched in 1978 by the United States. The current system comprises 30 satellites orbiting at an altitude of 20,200 km. The Russian equivalent GLONASS, comprises, at the time of writing, 26 satellites (20 of them operational) in orbits at 19,130 km. Europe is lagging behind with Galileo, supposed to be operational in 2014, but so far, no satellite has been placed into orbit. Satellite positioning systems offer great accuracy, to the nearest metre, or even better.

Triangulation, trilateration, or multilateration?

LPS and GPS (not just the US system) both use several transmitters to enable a receiver to calculate its geographical position. Several techniques are possible, each with its advantages and drawbacks. The important thing in all these techniques is the notion of a direct path (Line of Sight or LoS). In effect, if the transmitter signal has not taken the shortest path to the receiver, the distance between them calculated by the receiver will be incorrect, since the receiver does not know the route taken by the radio signal.
Three mathematical techniques are usually used for calculating the position of a receiver from signals received from several transmitters: triangulation, trilateration, and multilateration. The last two are very similar, but should not be confused.

Triangulation

Triangulation (Figure 1) is a very ancient technique, said to date from over 2,500 years ago, when it was used by the Greek philosopher and astronomer Thales of Miletus to measure (with surprising accuracy) the radius of the Earth's orbit around the Sun. It allows an observer to calculate their position by measuring two directions towards two reference points. Since the positions of the reference points are known, it is hence possible to construct a triangle where one of the sides and two of the angles are known, with the observer at the third point. This information is enough to define the triangle completely and hence deduce the position of the observer.
Using triangulation with transmitters requires the angle of incidence (Angle of Arrival or AoA) of a radio signal to be measured. This can be done using several antennas placed side by side (an array of antennas, for example, Figure 2) and to measure the phase difference between the signals received by the antennas. If the distance between the antennas is small, the incident front of the signal may be considered as straight, and the calculation of the angle will be fairly accurate.

It's also possible to use a directional antenna to determine the position of a transmitter. The antenna orientation pro-

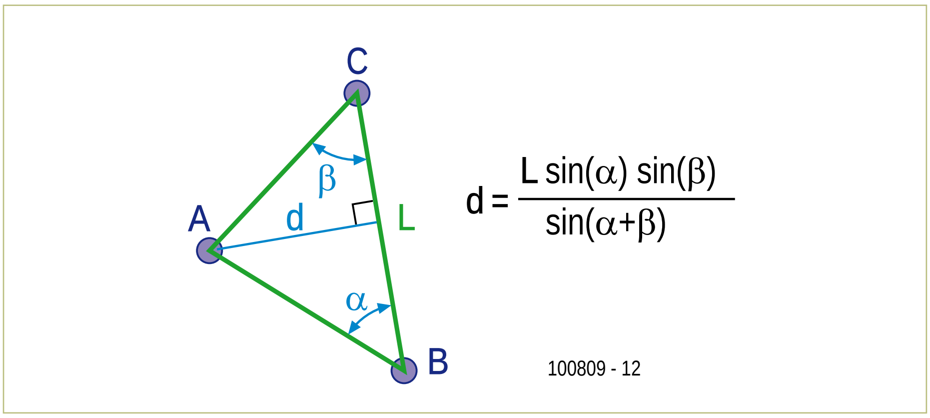


Figure 1. Triangulation: you are at A, from where you can see B and C. If you know their geographical positions, you can find out your own position with the help of a compass. Oh yes you can!

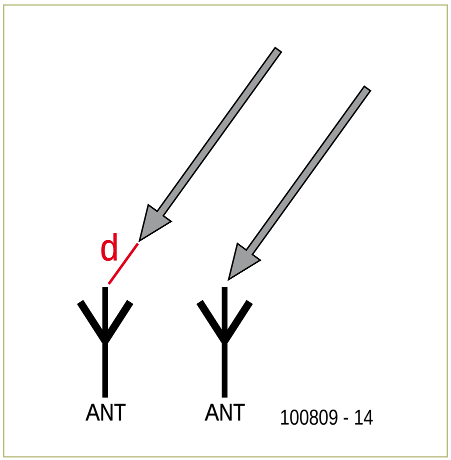


Figure 2. An antenna array makes it possible to measure the angle of incidence of a radio signal, and hence its direction.

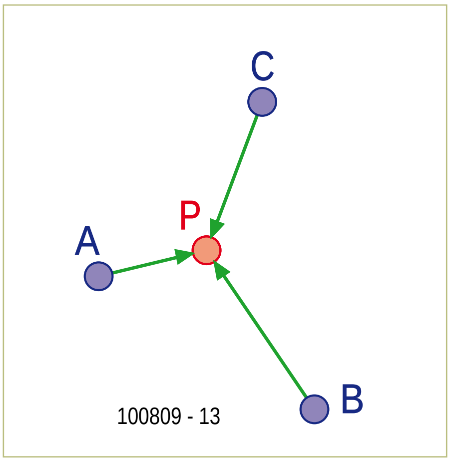


Figure 3. The length of the arrows corresponds to the arrival time at receiver P of the signals broadcast by three transmitters A, B, and C. It forms a measurement of the distances between the transmitters and the receiver.

ducing the strongest signal indicates the direction of the transmitter. All you then have to do is take two measurements from known transmitters in order to be able to apply triangulation.

Trilateration

This technique requires the distance between the receiver and transmitter to be measured. This can be done using a Received Signal Strength Indicator (RSSI), or else from the time of arrival (ToA, or Time of Flight, ToF, Figure 3) of the signal, provided that the receiver and transmitter are synchronized — for example, by means of a common timebase, as in GPS.
Thus, when receiving a signal from a single transmitter, we can situate ourselves on a circle (for simplicity, let's confine ourselves to two dimensions and ideal transmission conditions) with the transmitter at the centre. Not very accurate. It gets better with two transmitters — now there are only two positions possible: the two points where the circles around the two transmitters intersect. Adding a third transmitter enables us to eliminate one of these two possibilities (Figure 4).

When we extend trilateration to three dimensions, the circles become spheres. Now we need to add one more transmitter in order to find the position of the receiver, as the intersection of two spheres is no longer at two points, but is a circle (assuming we ignore the trivial point when they touch). This explains why a GPS needs to 'see' at least four satellites to work.

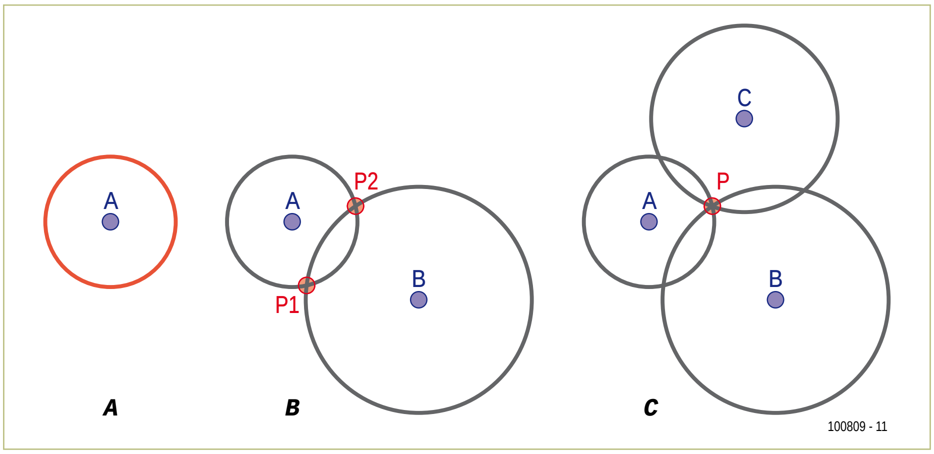


Figure 4. Two-dimensional trilateration. In 3D, another transmitter has to be added in order to determine a position unambiguously.

Multilateration

Using a single receiver listening to the signals (pulses, for example) from two synchronized transmitters, it is possible to measure the difference between the arrival times (Time Difference of Arrival or TDoA) of the two signals at the receiver. Then the principle is similar to trilateration, except that we no longer find ourselves on a circle or a sphere, but on a hyperbola (2D) or a hyperboloid (3D). Here too, we need four transmitters to enable the receiver to calculate its position accurately.
The advantage of multilateration is that the receiver doesn't need to know at what instant the signals were transmitted — hence the receiver doesn't need to be synchronized with the transmitters. The signals, and hence the electronics, can be kept simple. The LORAN and DECCA systems, for example, work like this.

LPS using Wi-Fi and RSSI

With the advent of Wi-Fi, we now find radio networks everywhere, and some people have had the idea of using these wireless networks to make an LPS. Often in these cases the 'L' of local is limited to a building or just a few rooms. These projects almost all use the RSSI signal strength indicator, available in the majority of receivers. The energy in a radio signal from a transmitter broadcasting uniformly in all directions is inversely proportional to the square of the distance from the transmitter (in effect, the area of a sphere is equal to $4\pi r^2$). Hence the further we are from the transmitter, the weaker the signal. So the RSSI signal gives us a measure of the distance between the receiver and the transmitter, and thus can be used for trilateration.

In reality, RSSI trilateration is not as simple as that, as the RSSI signal is not accurate enough. Already, the manner in which the RSSI depends on the intensity of the radio signal is not necessarily inversely proportional to the square of the distance from the transmitter, and in addition, the RSSI is influenced by obstacles like partitions or ceilings.

One way of remedying this is to map the RSSI over the whole area where the positioning system is required to operate. Microsoft's RADAR project uses this principle. The receiver measures the RSSI and then searches for the position that best matches on the map (or in a table). To improve the chances of getting the right position, the system takes account of the recent history of the receiver's movements and the environmental factors that have a direct influence on the RSSI map, like the number of people present or the ambient temperature. You can watch an animation about RADAR on the Internet [1]. To produce this animation (based on actual measurements), it was necessary to ensure that the receiver was at all times receiving at least four Wi-Fi transmitters (access points).

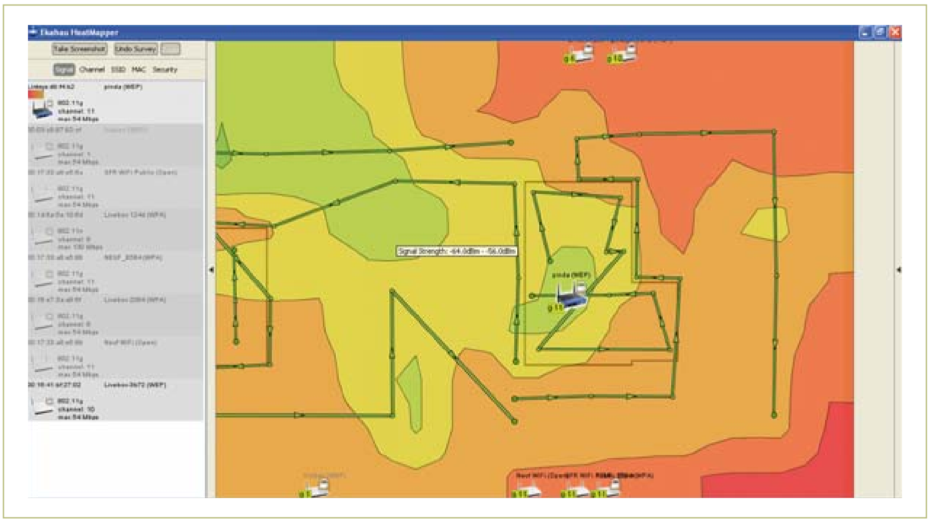


Figure 5. Wi-Fi coverage in and around the author's house, drawn with the help of the HeatMapper software. It's surprising to discover that the neighbours have a hidden Wi-Fi AP in their garden ('hidden' means that the SSID is concealed)!

Ekahau, a company spawned by research at Helsinki University (Finland), is offering a free software application [2] for easily producing a map of the Wi-Fi coverage in your home, based on the RSSI. On a grid or with the help of a previously-prepared plan, the

WPS

One means of exploiting Wi-Fi access points (APs) for geolocalization on a larger scale has been developed by Skyhook [6]. As Google is currently busy doing for its Street View project, Skyhook too is sending cars out to go around towns, but looking for Wi-Fi APs. The geographical positions of the APs and their names (SSIDs) are stored in a database, which already contains over 250 million APs! In this way, the company has created the Wi-Fi Positioning System (WPS). In order to find out your position, all you have to do is send the WPS the APs 'seen' by your computer or mobile phone. The database will tell you (roughly) where you are.

To improve the system, the geographical positions of phone network relay antennas have also been entered into the database. According to the company, the accuracy of the WPS varies between 10 and 20 metres (30 and 60 feet).

software (which weighs in at a hefty 100 MB — it's hard to understand why it should be so huge?) plots a brightly-coloured image while you walk around your home with your computer (Figure 5). You can then use this plot for experimenting with a robot fitted with a Wi-Fi receiver that provides the RSSI.

Some commercial LPSs

Several companies market LPSs or RTLSs based on wireless networks. RTLSs are standardized in ISO/IEC 24730. They are often used for tracking objects or persons (like Big Brother). In this scenario, the receiver position is not determined for navigating around, but so it can be transmitted to a master system. Hospitals, for example, are very keen on these types of system to avoid losing track of their patients or equipment. Here are a few examples.

Ekahau RTLS [3], the flagship product of the Ekahau company mentioned above, uses Wi-Fi to determiner the position of persons and objects. The system works broadly like Microsoft's RADAR, using RSSI mapping, but has been extended with, among other things, the notions of 'problematic paths' (an object cannot pass through walls) and 'relevant places' (an object cannot be at multiple places at the same time). More than anything, this makes it possible to limit the calculation power needed, since the system itself is capable of tracking thousands of objects at once. The manufacturer also sells Wi-Fi tags to allow remote tracking of the movements of an object or person.

Figure 6. This RTLS Wi-Fi tag (802.11 b/g/n) attached to an object makes it possible to locate it with a range of around 50 m (150 ft.) indoors and up to 150 m (450 ft.) in free field.

Zebra Technologies [4] is the owner of WhereNet, which markets the WhereLAN RTLS. This product is compatible with Wi-Fi, but adds proprietary access points that use the difference in the radio signal arrival time (DToA), instead of the RSSI signal. WhereLAN complies with the ISO/IEC 24730-2 standard.

Zebra also offers a proprietary ultra-wideband (UWB) radio technique: Dart UWB. This system works like a radar with transponders. A network of transmitters sends short pulses of UWB electromagnetic energy to 'wake up' active RFID tags so they can then be read. This system offers accuracy of 30 cm (12 in.) and reading distances up to 100 m (300 ft.).

One RTLS based on ZigBee is being sold by Awarepoint [5]. In this system, a building is fitted with detectors and their position is indicated on a plan. Objects to be monitored are fitted with a beacon that sends out a signal periodically — every five seconds if the beacon is moving, or every minute if it is stationary. The detectors then use this signal to accurately determine the location of the beacon. The position is reported back to the central unit, where the database of all the objects being monitored can be consulted. The grid network of detectors also monitors the RF conditions and is automatically adapted when the environment changes.

(100809-I)

Internet Links

[1] research.microsoft.com/en-us/um/people/bahl/MS_Projects/RadarDemo/demo.htm

[2] www.ekahau.com/products/heatmapper/overview.html

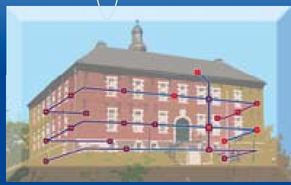
[3] www.ekahau.com

[4] zes.zebra.com/technologies/location/index.jsp

[5] www.awarepoint.com

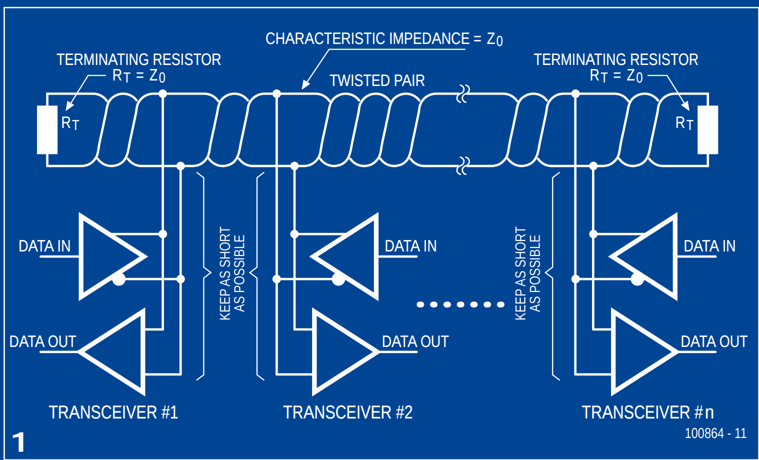
[6] www.skyhookwireless.com

Here comes the Bus! (2)

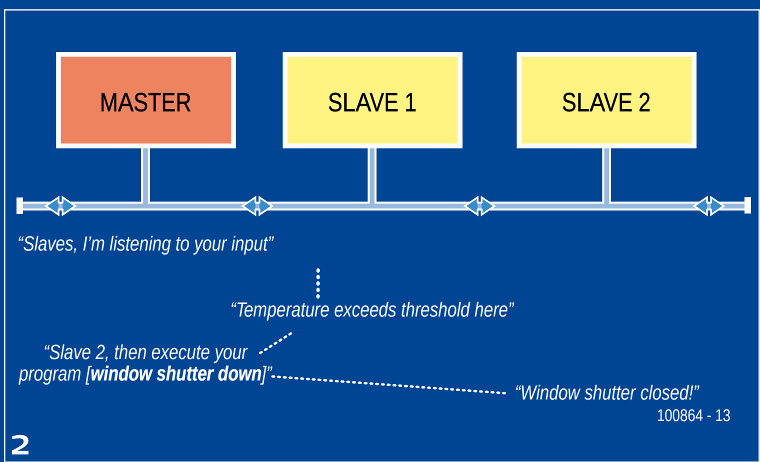


By Jens Nickel

Readers whose memories stretch back to our previous issue will recall that in the first part of this series our small but highly effective team decided that electrically the ElektorBus would be based on the RS-485 standard, operating over a twisted pair. To provide reliable communications each of our bus participants needs to be able to send and receive data. The bus is wired as shown in Figure 1, which is based on a Maxim application note [1]. The screenshot on the next page shows how not to do it.



1



2

With all bus nodes connected to the same pair of wires, the obvious sixty-four thousand pound question is: how do we make sure that only one bus node is talking at any given time? Unlike the CAN bus standard, the RS-485 standard does not specify a mechanism for detecting collisions, and without such a mechanism we are in danger of losing data.

As you might suspect, we spent some time discussing the problem, coming up with several alternative solutions.

The simplest approach is to make one of the nodes the boss, with the underlings simply doing what they are told and speaking only when spoken to. The master-slave arrangement has the advantage that the slave nodes can be kept very simple and to a large extent standardised, which in turn relieves the developer of a considerable burden: all the nodes can use the same microcontroller, and even be running identical firmware. The master simply issues commands like 'take port pin PB5 High', or 'take a reading from ADC1 and send it to me'. The software in the slave microcontrollers then simply has to parse the commands (of which there need only be a few different types) and then suit the action to the word).

However (as you might have guessed from the length of this article), there are some serious downsides to this quasi-direct access of the bus master to the slave's I/O pins. The most significant of these is that the master must know exactly how each slave is wired. For example, if a slave includes a temperature sensor, the master must somehow know how to convert a raw A/D converter reading into a temperature value. It also makes for a lot of bus transactions. Consider, for example, the task of raising a roller blind until a limit switch is actuated. The conversation between master and slave might go something like this: "Set port pin PB5 High." "Done that." "Now, is port pin PC1 High?" "No." "How about now?" "Yes." "Okay, take port pin PB5 Low at once."

So as you can imagine this idea was rapidly sent on its way to the shredder. After all, what we have is more of an inter-microcontroller communications protocol aimed at a certain narrow range of applications than a true bus system. In my mind's eye I was picturing a fully-fledged home automation system, with the slaves having at least a modicum of intelligence. This means that a node should for example translate a raw A/D converter reading into a physical quantity so that different types of sensor, converter and microcontroller can be mixed on the same bus without the bus master needing to know the details. It would also be desirable to implement simple control loops running within the slave (of the form 'set output X low until input Y goes high'), which would be enough to cover cases such as the roller blind example above.

It also seemed at first sight to be a little impractical to have the slaves only send messages on request. When values need to be monitored, this means that the master must interrogate the slave on a regular basis, which, besides feeling inelegant, might result in latencies unacceptably great for applications such as alarm systems. In my vision of the bus system (Figure 2) the master can go into a 'listen mode', waiting for a range of events such as 'input 2 on slave 1 has gone low' or 'temperature at slave #3 has gone over 100 °C'. By design these events should be relatively rare, which will help minimise the number of collisions on the bus.

"That should be enough for a modest home automation system," said my French colleague Clemens, "but what else could we do with the bus?"

"Well," I said, "we could provide a 'fast transmit mode' to allow rapid point-to-point communications." That would allow us to send rapidly-changing data to the master, at least from one of the slaves.

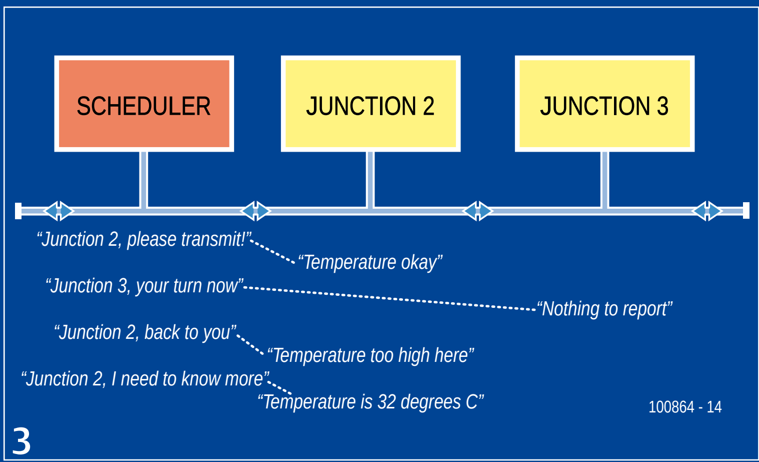
It was obvious to us both that in this situation the bus would not be available for other activities and that we could in some circumstances miss important event notifications from other nodes. "We need some kind of prioritisation on the bus,' said Clemens, 'we need nodes that are workers, under-managers, over-managers, under-over-managers, over-under-managers..." So, like the CAN bus in a car, where (in the fullest sense of the word) vital data can always get through?

"And what happens," asked Clemens, warming to his point, "when we have more than one node making decisions? In your design only the master collects data, but if all the nodes have access to all data packets, there isn't really a single master any more."

Oh my giddy aunt, I thought to myself, things are starting to move quickly. If the nodes are allowed to talk to other nodes without going via the master, then we are getting close to designing a network that can tolerate faulty nodes. How do we stop the nodes from chattering away uncontrollably to one another on the bus? "How about allocating defined time-slices?" suggested Clemens, "though that would of course put a limit on the number of devices we could support simultaneously."

There followed an hour of furious googling and sending one another links. We discovered, for example, a Siemens patent on a time-slice-controlled symmetric bus architecture which could be used for sending home-automation commands at the same time as multimedia data streams [2]. Also of interest was a presentation on 'Time-triggered CAN' [3].

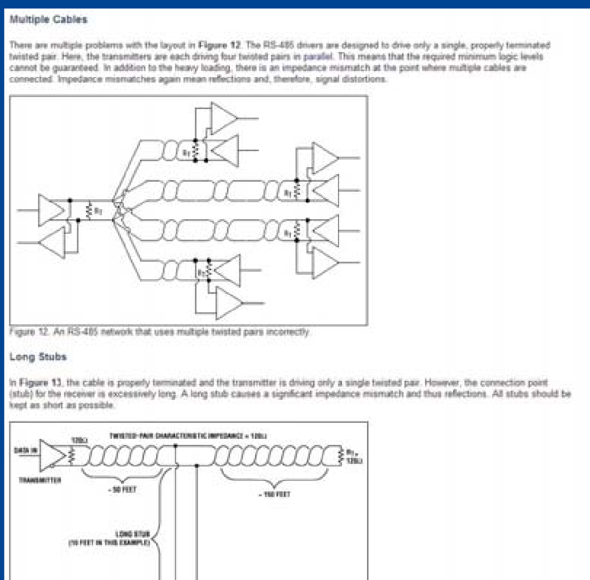
The advantages of a time-slice architecture were very seductive, but the details involved in synchronisation would be fiddly. I also felt that it would be difficult to get such a system up and running with a reasonable amount of development and debugging time, both for us and for our readers.



3

"All right then," said Clemens, "what about using some sort of scheduler?" This would allocate to each of the other nodes a time to speak based on the importance of what it might have to say, with the nodes being scheduled in order of decreasing priority. "Something like a pre-emptive multitasking scheduler," he explained.

I had to concede that this approach to collision avoidance was not bad, even though it was almost exactly the opposite of what I had originally envisaged. Nevertheless, the problem still remained of how the scheduler could stop a node from talking if the bus was needed for something more important.


Multiple Cables

There are multiple problems with the layout in Figure 12. The RS-485 drivers are designed to drive only a single, properly terminated twisted pair. Here, the transmitters are each driving four twisted pairs in parallel. This means that the required minimum logic levels cannot be guaranteed. In addition to the heavy loading, there is an impedance mismatch at the point where multiple cables are connected. Impedance mismatches again mean reflections and, therefore, signal distortions.

Figure 12. An RS-485 network that uses multiple twisted pairs incorrectly.

Long Stubs

In Figure 13, the cable is properly terminated and the transmitter is driving only a single twisted pair. However, the connection point (stub) for the receiver is excessively long. A long stub causes a significant impedance mismatch and thus reflections. All stubs should be kept as short as possible.

The solution to this came to me somewhat later: each node would only be allowed to send a fixed number of bytes before having to give way to the next node (Figure 3). If a node reports a higher-priority event, the scheduler would then give it permission to speak. All we needed to do now was to try out these fine ideas to see if they would actually work in practice...

(100864)

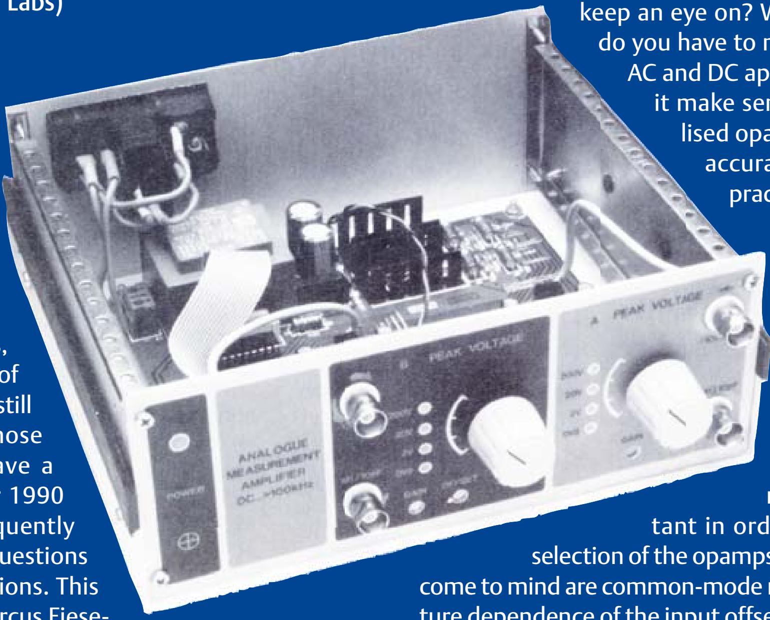
What do you think? Feel free to write to us with your opinions and ideas.

[1] http://www.maxim-ic.com/app-notes/index.mvp/id/763
[2] http://www.patent-de.com/20030320/DE10126339A1.htm (in German)
[3] http://www-lar.deis.unibo.it/people/crossi/files/SCD/An%20Introduction%20to%20TTCAN.pdf

E-LABs INSIDE

Design tips for instrumentation amplifiers

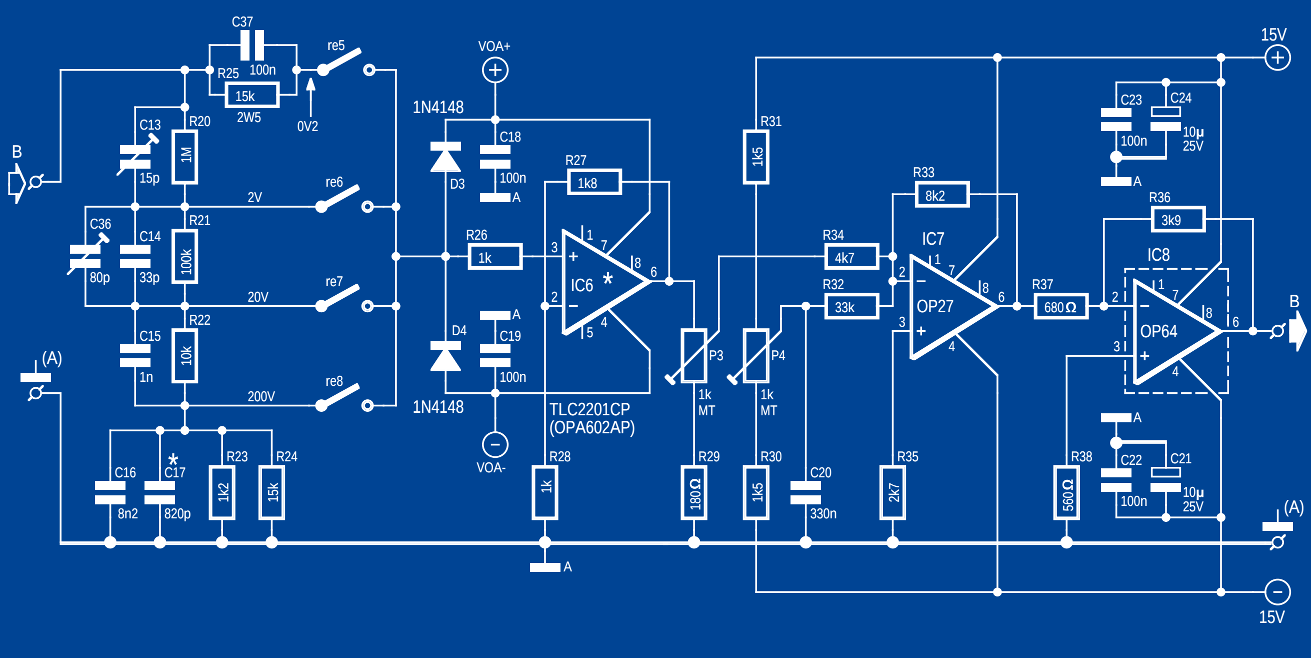
By Ton Giesberts (Elektor Labs)



Many Elektor readers faithfully keep all their old issues and also frequently refer back to them when searching for a circuit for a particular application. Younger readers have now also discovered how to find these older circuits, because these are often of excellent quality and are still relevant to build (for those who are interested: have a look at the DVD Elektor 1990 through 1999). We frequently receive comments and questions about these old publications. This is how Elektor reader Marcus Fieseler used the circuit from the 'Universal instrumentation amplifier' from January 1992 as the basis for his own measuring circuit. While he was experimenting he came up with a number of questions: "I would really like to know how to dimension an instrumentation amplifier. What are the criteria when selecting an opamp and what are the most important specifications in the datasheet that I need to keep an eye on? When choosing the opamp, do you have to make a distinction between AC and DC applications? Also, when does it make sense to use a chopper-stabilised opamp? What is the maximum accuracy that can be obtained in practice?"

For these types of questions there is one designer in the Elektor lab who specialises in this subject: Ton Giesberts!

When designing instrumentation amplifiers, a number of criteria are important in order to facilitate the correct selection of the opamps to be used in it. Criteria that come to mind are common-mode range, input offset, temperature dependence of the input offset, bias current, temperature dependence of the bias current, bandwidth and power supply range. For battery-powered circuits the choice will quickly narrow to rail-to-rail types. However, with many of the opamps of this type, the characteristics of the amplifier change when operating close to the supply rail. When used in accurate measuring systems this is something that certainly has to be taken into account. In these situations you could consider using an invert-



E-LABs INSIDE

ing amplifier as input amplifier. There are then also no problems the with common-mode dependency of some characteristics.

With some opamps, in particular the bipolar types, the bias current depends on the common-mode voltage at the inputs. In a non-inverting amplifier there is a greater chance that the output voltage deviates, depending on the values of the resistors used. This results in distortion with AC voltage signals, and with DC voltage measurements a gain that appears to change with the change in input voltage. This is nowhere to be found in most datasheets.

A good example of where this information is provided in the form of a chart (BIAS AND OFFSET CURRENT vs INPUT COMMON-MODE VOLTAGE) is the OPA111 from Burr-Brown (Texas Instruments these days). But with a maximum bias current of 1 pA this is not a problem in most applications. When we look at an OP27, where the bias current can amount to several tens of nA (and because of internal bias correction this can be either positive or negative), then we can expect a problem with DC voltage stability when using larger values of resistors. Because of noise it is better to choose lower impedances, and you will then also have less of a problem with DC voltage variation due to bias current. Very good opamps are available for extremely stable DC voltage measurements, such as the OPA177, a device with an offset of only 25 µV (F-type).

For even more accurate DC voltage measurements you could consider using chopper-stabilised opamps. Devices such as the TLC2654 or ICL7650 have and offset of less than5 µV. There are also related models such as the AD8551 from Analog Devices, which possess a special auto-correction circuit (this one has an offset of only 1 µV). Their disadvantage however is that none of these are particularly fast amplifiers. With chopper-stabilised opamps it's recommended to stay at least a factor of 10 below the chopper frequency. This is perhaps the reason why there are so few chopper-stabilised opamps (for example, the MAX420 and ICL7650 have disappeared from the Maxim line-up). These days there are also normal opamps with equal, if not better, specifications.

More bandwidth is usually a trade-off with reduced DC voltage stability, so you need to strike a compromise. There are now opamps that have a bandwidth of several 100 MHz. If a lot of gain and good linearity are required, then you could consider dividing the gain across two or more separate stages. The overall bandwidth will be greater for any given gain. For the total gain this has the disadvantage that the accuracy is reduced by about 2 % with each stage when using 1 % tolerance resistors. An amplifier stage will now need to have a potentiometer added to allow for calibration. The impulse response of the total amplifier is also more complex. When searching for the right opamp it is often difficult to find the desired impulse response. It is usually necessary to use frequency compensation in the feedback circuit (or some other method). With multiple amplifier stages it is necessary to tune the individual frequency compensations with each other. And then we haven't even mentioned frequency compensated voltage dividers with multiple ranges.

When the instrumentation amplifier functions as the front-end for an A/D converter, then the amount of noise determines the maximum resolution of the A/D converter. If you want a bandwidth that's as large as is possible, then you will quickly bump into the theoretical noise of the amplifier stages and not in the last place the impedance of the voltage divider for the measuring ranges, too. The noise of a resistor is determined by the well-known formula $\sqrt{(4KTBR)}$. For example, let's take a resistor as a noise source and an 8-bit A/D-converter. With a 1 V reference and a desired bandwidth of 1 MHz, the value of the resistor may amount up to 115 MΩ. Above that the noise will clearly exceed 1 LSB. With this calculation we have assumed 'sine-shaped' noise and have divided the size of the resolution by $2\sqrt{2}$ (peak-to-peak). We'll ignore for the moment that the noise can have bigger peaks in practice. It's a good idea to choose the impedance of the voltage divider as half the calculated value or even lower. We will also assume for the moment that the voltage divider is not frequency compensated. For the resistor R we have

$$R = ((V_{ref} / 2^N) / 2\sqrt{2})^2 / (4 \times 1.38 \times 10^{-23} \times 300 \times 1 \times 10^6)$$

where 300 K = T; 1.38 x 10^{-23} = Boltzmann constant K, and N is the number of bits.
For 12 bits a resistance of up to 450 kΩ is allowed, at 16 bits, 1.76 kΩ and at 24 bits, just 26.8 mΩ (with a 1 kHz bandwidth this equates to 26.8 Ω). Clearly, with a resolution of 12 bits and an even higher bandwidth — for example a digital oscilloscope and a 10:1-probe — a certain amount of noise will be visible. In the active parts there is further contribution to the noise level mainly from the first amplifier stage.
However, in practice, the frequency compensation will be around 10 kHz, for example 1 MΩ and 15 pF. If we calculate the number of bits, only taking the noise contribution from the voltage divider:

$$V_{ref} / 2^N = 3.64 \times 10^{-5}.$$

From this follows 2^N = 27474 or $N = (\log_{10} 27474 / \log_{10} 2)$ = 14.75 bits.

Add to this the noise from the amplifiers and the chaotic nature of noise, you soon realise why the resolution of those top of the range oscilloscopes is usually around 12 bits.

When using microcontrollers for taking measurements of sensors, you could even use the noise to your advantage to obtain a higher resolution than that of the internal A/D converter. By using a considerable amount of oversampling and averaging you can obtain a much more accurate measurement. In addition, software offers the possibilities of correcting non-linearities and other defects. But that is a different story.

(100812-I)

Internet Links

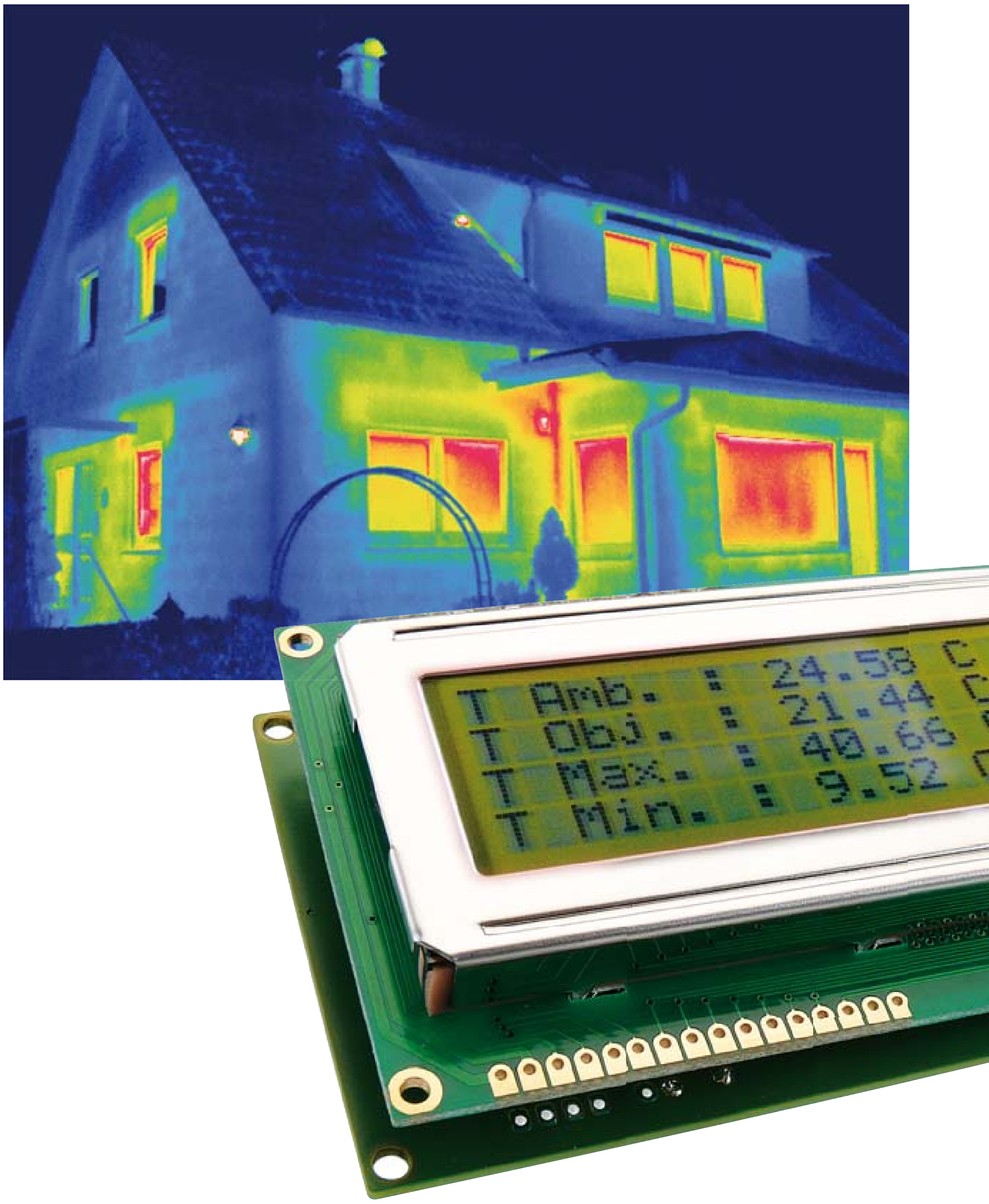
www.analog.com/library/analogDialogue/archives/39-05/Web_ChH_final.pdf

http://en.wikipedia.org/wiki/Operational_amplifier

Contactless Thermometer

Are you running an infrared temperature?

By Christian Tavernier (France)



It's easy these days to find several cheap sensors for contactless thermometers, also called infrared thermometers. These sensors, which measure the infrared radiation from objects, make it possible to build a contactless thermometer yourself with performance easily as good as its commercial counterparts.

Technical specifications

- Infra-red detecting thermometer
- PIC16F876A microcontroller
- Four line × 20 character LCD display
- Displays ambient and object temperatures
- Stores minimum and maximum temperatures
- Runs off two 1.5 V cells
- Open-source software

Our thermometer measures at the same time both the ambient temperature and the temperature of any object placed within its 'field of view'. And even though the ambient temperature range 'only' goes from −40 to +125 °C, the object temperature can be from −70 to +380 °C, and all with an accuracy of 0.5 °C and a measurement resolution of 0.02 °C.
In order to be self-contained and portable, it runs off batteries or rechargeable cells, and for even greater convenience, our thermometer automatically remembers the maximum and minimum temperatures of objects, and displays everything on a backlit LCD display with four lines of 20 characters. Thanks to the sensor used, it only needs two ICs: a perfectly ordinary PIC microcontroller and a switching regulator to power it.

MLX90614 sensor

The sensor we've chosen is the MLX90614 from Melexis, and it's this that gives our thermometer its excellent performance. This IC, which comes in a metal TO-39 package with a window, should not be regarded as just any old temperature sensor, like a thermistor, for example, as it includes a whole load of processing and shaping circuitry (**Figure 1**).
The sensor proper is an infrared thermopile (or two, depending on the IC version) that delivers a very low, non-linear signal which would thus be difficult to use directly. This signal is first amplified by a chopper-stabilized opamp. It is then converted to digi-

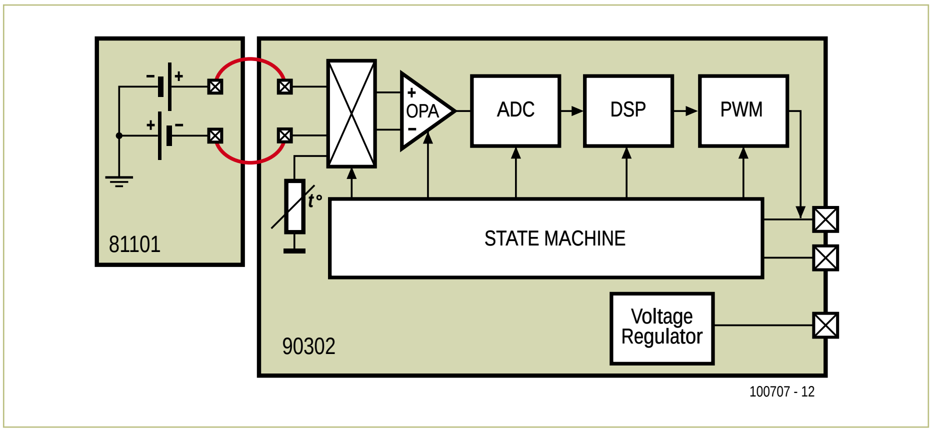


Figure 1. Internal block diagram of the MLX90614 sensor.

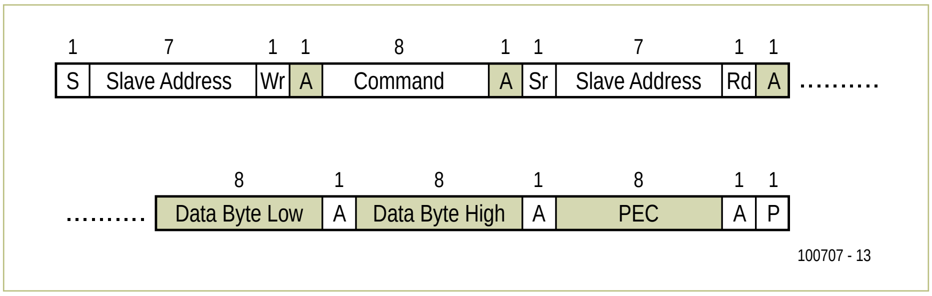


Figure 2. Principle of one of the sensor's RAM read frames.

tal in a delta-sigma type converter before being applied to a digital signal processor (DSP). After noise filtering and sensor signal processing performed by this DSP, the temperature information is available in a directly-usable digital form.

To simplify interfacing, the sensor can provide this information via a 2-wire SMBus (virtually identical to the I²C) or in the form of a PWM (pulsewidth modulated) signal. Although the latter mode does make it simpler to connect up the MLX90614, the PWM signals are trickier to process than those from the SMBus. What's more, the resolution in PWM mode is only 0.14 °C, as against 0.02 °C in SMBus mode.

Depending on the version, this IC runs off a single power rail of either 3 V or 5 V, so you need to pay great attention to which type you've got before fitting it into this circuit — we nearly learnt the hard way...

Talking to the sensor

If we decide to communicate with the sensor in SMBus mode — which is what we've done in our thermometer — the syntax to be used is relatively simple, provided we don't want to modify the internal parameters, which are factory-set but perfectly suitable for our needs. To read the ambient temperature and the temperature of the objects the sensor is aimed at, all you have to do is read from two different locations in its internal RAM, which is done using an SMBus frame similar to the one in **Figure 2**. After sending the sensor its slave address, all you then have to do is send it a command, chosen from those proposed in **Table 1** as far as temperature measurement alone is concerned. Sending its slave address again then lets you receive back two bytes containing the LSB and MSB of the temperature, followed by a check polynomial, marked PEC in Figure 2, which we won't be using here.

The temperatures, expressed in Kelvin, are represented by unsigned 15-bit words. If we call the 15-bit word output N, and given the sensor resolution in SMBus mode, the measured temperature T expressed in K is given by:

$$T = 0.02 \times N$$

But as it's more user-friendly to read a temperature in degrees centigrade (Celsius), the thermometer software simply uses the formula:

$$T = 0.02 \text{ x } N - 273.15$$

If you're not familiar with the SMBus, don't worry, it's 99 % identical to the better-known I²C bus. It only differs in a few subtleties in the protocol, which are unimportant here, and by a small difference in terms of the electrical levels; a difference that very fortunately PIC microcontrollers with an I²C interface are able to handle, as long as we correctly program one bit in one of the control registers in their MSSP interface.

Table 1. The main commands for reading the temperatures.

Command	Code (hexadecimal)
Raw ambient temperature	0x03
Temperature, IR sensor 1	0x04
Temperature, IR sensor 2	0x05
Linearized ambient temperature	0x06
Linearized sensor 1 temperature	0x07
Linearized sensor 2 temperature	0x08

Thermometer circuit

Clearly, the sensor's high degree of integration simplifies the circuit of our thermometer, which can therefore be based on a simple PIC microcontroller, as long as it has an I²C interface. Here the 16F876A was chosen.

As shown in **Figure 3**, our PIC is used in crystal clock mode (20 MHz) and has a manual reset command via the button S1, included to allow the memories containing the maximum and minimum temperatures to be reset.

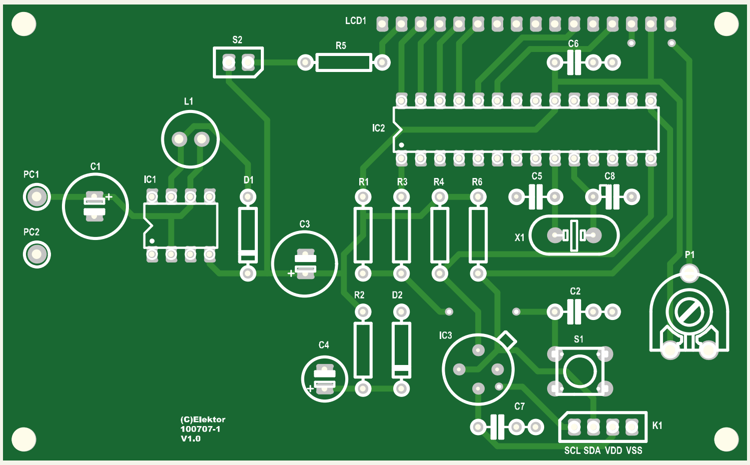
COMPONENT LIST

Resistors (0.25W 5%)
R1,R4,R6 = 10kΩ
R2, R5 = 100Ω
R3 = 1.5kΩ
P1 = 10 kΩ trimpot, horizontal

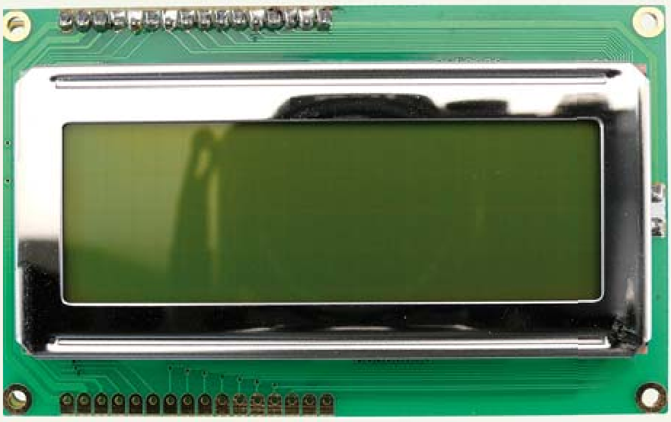
Capacitors
C1,C3 = 100µF 16V, radial, lead pitch 2.5mm
C2 = 470nF 63V, MKT, lead pitch 5 or 7.5mm
C4 = 10µF 25V, radial, lead pitch 2.5mm
C5, C8 = 22pF ceramic, lead pitch 0.2” (5.08mm)
C6 = 100nF ceramic, lead pitch 5 or 7.5mm
C7 = 10nF ceramic, lead pitch 5 or 7.5mm

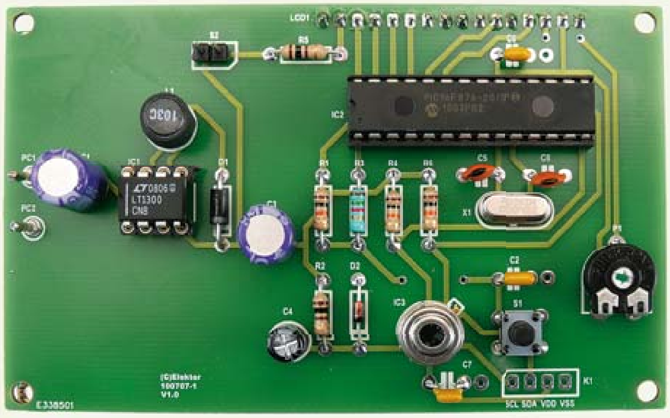
Inductors
L1 = 10µH, Panasonic type ELC08D100E (RS Components)

Semiconductors
IC1 = LT1300
IC2 = PIC16F876A-I/SP, programmed, Elektor # 100707-41
IC3 = Sensor type MLX90614ESF-BAA or MLX-90614ESF-AAA (see text)
D1 = 1N5817 (must-have Schottky)
D2 = 3.3V 0.4W zener diode



Miscelllaneous
LCD1 = LCD, 4 lines of 20 characters, e.g. Displaytech 204A
X1 = 20MHz quartz crystal, HC18/U case
S1 = pushbutton, 1 make contact, ITT type D6 if fitted on PCB
S2 = swich, changeover, or wire link
DIL IC sockets: 1 pc 8-way; 1 pc 28-way
Pinheader pins, lead pitch 0.1” (2.54mm)
Pinheader sockets, lead pitch 0.1” (2.54mm)
PCB, Elektor # 100707-1





The display used is an LCD type, with or without backlight as you prefer, depending on whether or not switch S2 is closed (or a link is fitted). Ideally, you should use a perfectly standard type with four lines of 20 characters, but the circuit also works with a pin-compatible two-line, 16-character type. In this instance, the two lines displaying the minimum and maximum are not visible, which is a bit of a shame. The display is used in 4-bit mode, driven from port B of the microcontroller.

The two bus lines coming from the sensor terminate at RC4 and RC3 of the PIC respectively, the inputs to parallel port C, which are shared with its internal I²C interface.
The circuit is powered at 5 V from two 1.5 V cells (or two 1.2 V NiMH rechargeables) by way of the LT1300 switching step-up DC-DC converter. This IC provides a stabilized 5 V output from any input voltage between 2 and 5 V. It is capable of supplying a current of 400 mA, i.e. a great deal more than is needed for our thermometer.
Although the sensor does exist in a 5 V version, the most readily available at the

moment is the 3 V version. This explains the reason for resistor R2 and its associated zener diode (D2). Note here that the SMBus load resistors R4 and R6 are returned to the 5 V rail all the same, in order to guarantee correct electrical levels at the PIC input. But there's no risk to the MLX90614, as it has internal limiting diodes.

Software

The software (copiously annotated source code and HEX file) is available for free downloading from [1] and [2]. It has been written in MikroBasic from Mikroelektronika, which has the advantage of having available a perfectly functional I²C library. Note that the size of the compiled software is over 2 KB and so it can't be compiled using the demo version of this compiler.

Judging by the content of the Internet forums devoted to the MLX90614, some users seem to have encountered difficulties, so we thought it would be worth commenting here on the relevant section of code.

The procedure for reading the temperature is called using the parameter 'com' for the chosen command. The procedure then scrupulously adheres to the instructions from Melexis, namely:

- send a start signal to start the I²C/SMBus transaction;
- send the IC's slave address (Melexis specifies in the data sheet that all the ICs respond to the address 0x00) with Write mode selected (R/W = 0);
- send the command contained in the variable 'com' (0x06 for the ambient temperature and 0x07 for the temperature of the object);
- send a repeated start signal;
- send the IC's slave address, this time with Read mode selected (R/W = 1);
- receive a series of three bytes: the LSB and MSB of the temperature, then the PEC (not used in our application);
- send a stop signal to terminate the I²C/SMBus transaction;
- and finally, reconstruct the 15-bit word containing the temperature by concatenation of the two bytes received.

We'll leave you to analyse the rest of our program with the help of our copious notes, and move on to the practical aspects of construction.

```
Sub procedure Read_temp(dim com as byte)
  I2C1_Start()             ' issue I2C start signal
  I2C1_Wr(0x00)            ' send address (device address + W)
  I2C1_Wr(com)             ' send command
  I2C1_Repeated_Start()    ' issue I2C signal repeated start
  I2C1_Wr(0x01)            ' send address (device address + R)
  SensorLow = I2C1_Rd(1)   ' Read temp. low byte (acknowledge)
  SensorHigh = I2C1_Rd(1)' Read temp. high byte (acknowledge)
  PEC = I2C1_Rd(1)         ' Read PEC (not used)
  I2C1_Stop()              ' issue I2C stop signal
  SensorRaw = SensorLow + (SensorHigh << 8) ' Build temp. word
End sub
```

Construction

With the aim of simplifying the mechanical side of building the thermometer, we've designed a PCB the same size as the display board, so it can be mounted on the back of it.

Sourcing the components should not present any problem. The display used is a Displaytech 204A, but in theory at least any 4 line × 20 character LCD display using a standard interface (ST7066, HD44780, or KS066 controller) will do, as well as any 2 line × 16 character LCD display, as indicated above, albeit with the loss of the min. / max. display.

The MLX90614 exists in numerous versions, distinguished by the part number suffixes. The commonest and cheapest version is the MLX90614ESF-BAA. The letter B indicates that it runs on 3 V. If you come across an MLX90614ESF-AAA, this is a 5 V version that can still be used in our circuit, but in that case you'll need to remove D2, C4, and C7, and replace R2 by a wire link.
Note that L1 must be capable of carrying a current of 800 mA without saturating. Otherwise the LT1300 will work very badly, or not at all.

The sensor can be remoted to the case via the four connecting pins provided for the purpose, but to avoid possible interference and distortion of the SMBus signals, it's preferable not to extend its connections more than a few tens of cm.

We fitted 2.54 mm (0.1") pitch sockets on the back of the display and pins at the same pitch on the copper side of the PCB. In this way, you can produce an assembly that's easy to remove in the event of problems.

Use and adapting to your own needs

The thermometer operates as soon as power is applied, and the first line of the display gives you the ambient temperature, i.e. that of the sensor case. The second line shows the temperature of the object the sensor is pointing at, i.e. the average of the temperatures of the objects in the sensor

Can also be read without contact, but using a mobile phone.

window's field of view. The angle of view of the standard version (MLX90614ESF-XAA) is not stated. For the MLX90614ESF-XAC it is 35°, and 10° for the MLX90614ESF-XAF.

The maximum and minimum object temperatures are stored automatically, and display on the last two lines of the display. They are updated at the same time as the measurements, which take place once per second.

To reset the minimum and maximum, all you have to do is press the reset button. You can adapt the software to your own needs and make the thermometer behave quite differently. However, if you want to modify the procedure that handles the communication with the sensor, only do so if you really know what you're doing, as it is possible to write to it, and incorrect writing may destroy or modify its factory-set calibration parameters, rendering any subsequent measurements inaccurate.

(100707)

Internet Links

[1] www.elektor.com/100707

[2] www.tavernier-c.com

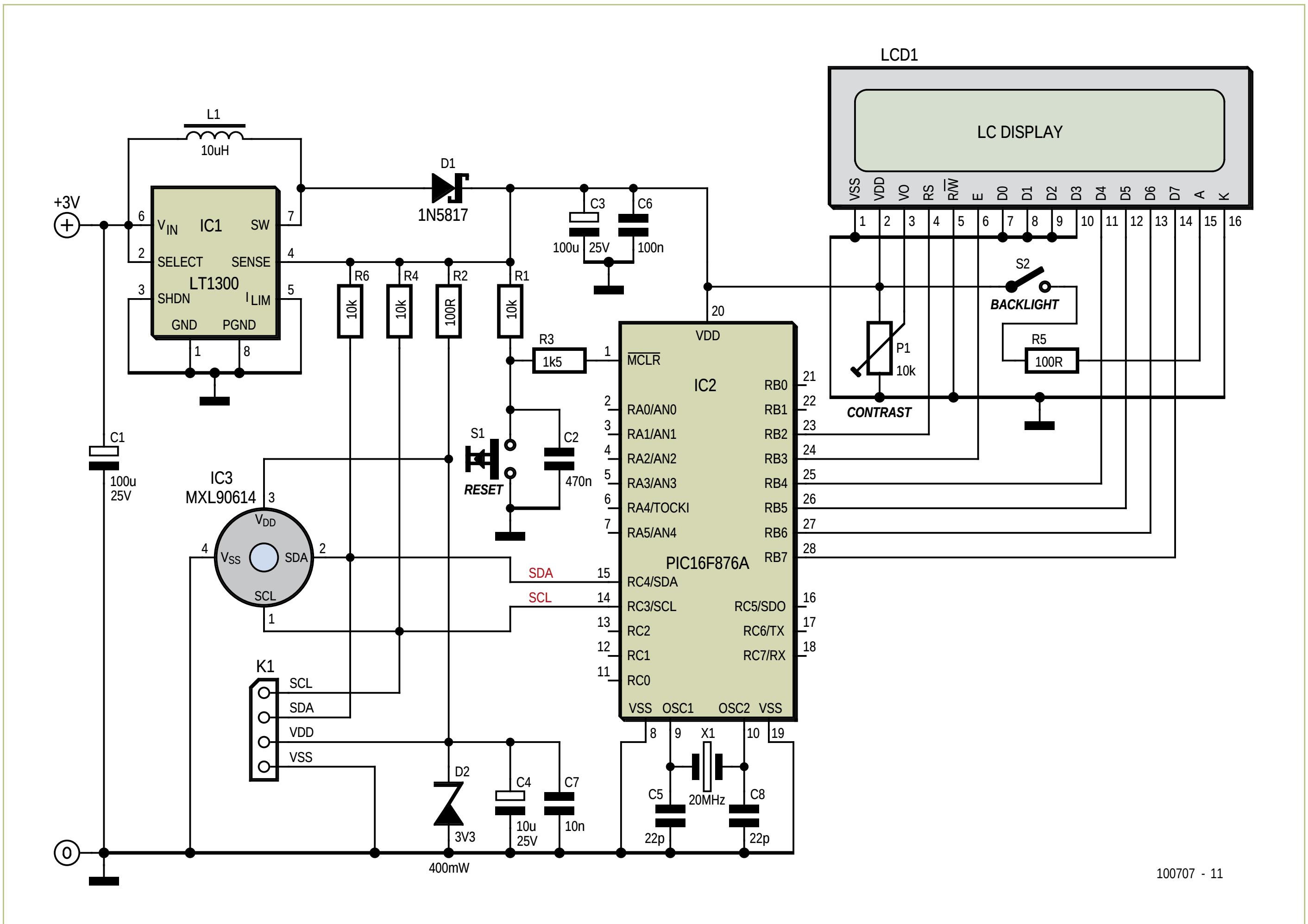


Figure 3. Complete circuit of the contactless thermometer.

TimeClick
Programmable camera controller

Carlos Ladeira (Portugal)

This project dubbed TimeClick controls a digital SLR camera without human intervention using a wired connection. It can take photographs at fixed or random time intervals or in response to sensor input, which makes it suitable for various purposes from HDR photography to sound-triggered pictures.

This project came about after having taken too many photos randomly at events like parties. This way of operating a camera can lead to funny results at best. However, as the project developed, the author started to have new ideas and added several new features to extend the functionality, such as:

- Fixed delay between photos
- Random delay between photos
- Sensor input to trigger photos
- Manual operation for use as wired remote
- Bulb mode
- Mirror lock
- Exposure bracketing
- 12 presets to save different sets of configurations

Operation is remarkably simple and once the device is configured and wired to the camera, you simply choose the right spot for the camera on a tripod and let it do all the work. Depending on the power source used for the device and the camera, you can have the camera in operation for days.

Tools used

The circuit started out on a breadboard and later evolved into a complete PCB design. For the hardware design (schematics and PCB) open-source CAD program Kicad was used. It is very easy to use, even for first time

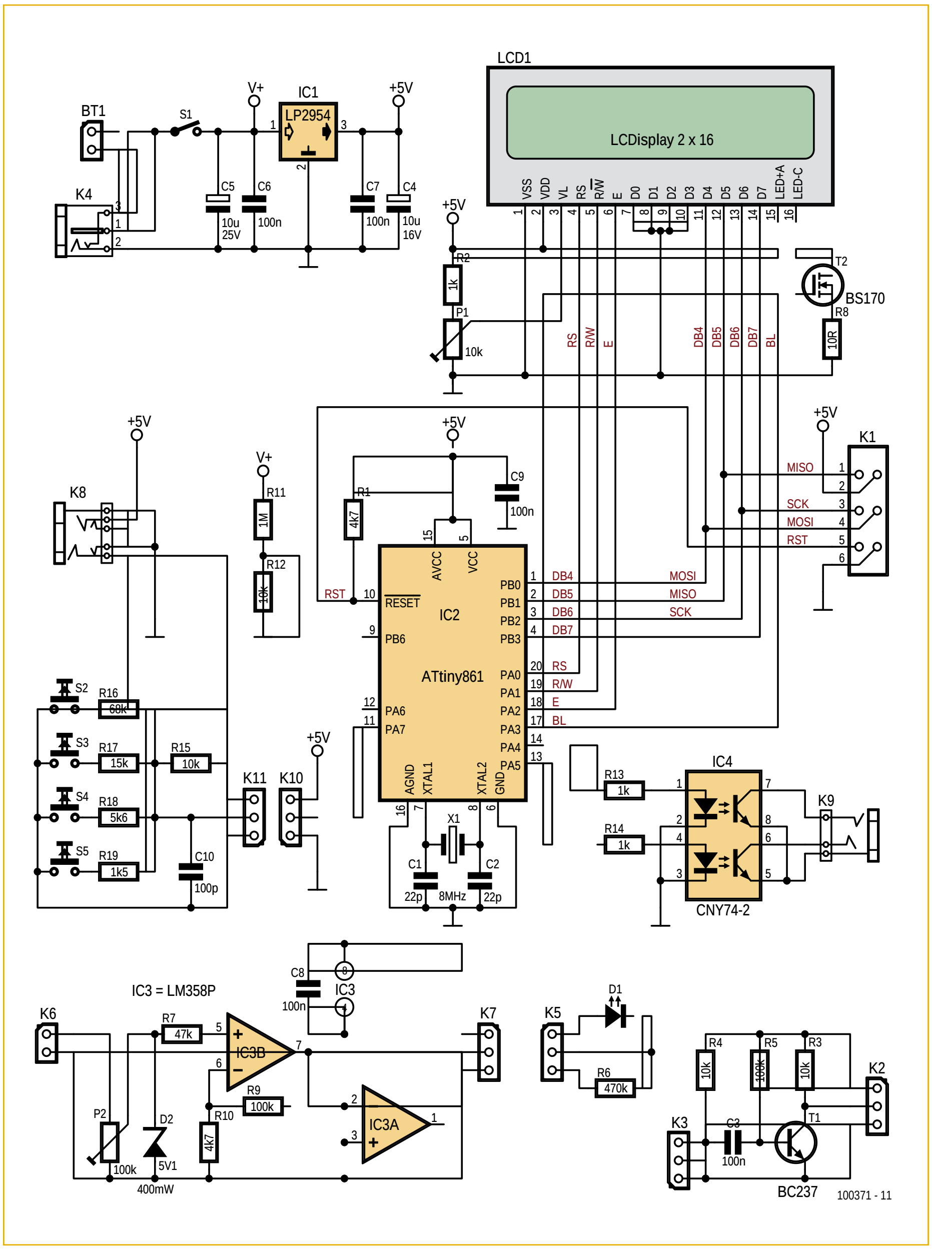


Figure 1. Schematics

users. Since the author has done everything at home (PCB making using the *ironing method*), no metallised holes have been used and all connections between both layers of the PCB are made using wires or component pins.

For the software design AVR Studio 4 was the development tool. The microcontroller programming was done with Atmel's AVR Dragon using ISP mode. In the beginning the author experienced some troubles with AVR Studio and AVR Dragon. Sometimes AVR Studio seemed to be able to connect to the AVR Dragon but unable to communicate with the microcontroller. This problem was solved once by restarting the computer and ultimately by completely reinstalling AVR Studio. From what can be found in postings on the Internet by users with similar problems, we've a hunch it was caused by a bug in AVR Studio mixed with inappropriate procedures.

The advised procedure for programming the microcontroller can be found under *Assembling & Programming*.

Hardware description

Starting with the power supply, there are two options for powering the device. One is using an internal 9-V battery, the other is using an external power supply. A power switch has been added in this section of the circuit, see **Figure 1**. The author's prototype used six AAA batteries instead of a 9-V block battery, mostly because of the poor battery life of the 9-V block type (IEC 6LR22).
Under normal operation the circuit consumes about 10 mA with the LCD backlight off. If the LCD backlight is on, current consumption rises to about 100 mA. To save energy during operation, the LCD backlight is turned off automatically when no buttons are pressed for 10 seconds. To turn the backlight on again, just press any of the device's buttons.

Since the LCD needed 5 volts and the same voltage was suitable for powering the rest of the circuit, the author went for a standard voltage regulator. Thus, the first pick was the well known 7805. However, a quick change of mind occurred after a close look at the datasheet. A 7805 needs at least 7 volts at its input to be able to stabilise its output at 5 volts. This of course isn't any good when you want to power your circuit from batteries. Moreover, the 7805 is not known for its high efficiency...

As result of all this, an LP2954 seemed much more appropriate. This is a regulator with reverse battery protection and a low dropout voltage which helps to extend battery life.

There is a battery level indication option included using R11, R12 and a free ADC port (ADC9) of the microcontroller.
The heart of this circuit is an ATMEL ATtiny861 which fits the bill exactly. The main reason for the author to choose this microcontroller instead of another from the Atmel family was mostly because of its availability in SOIC-20 package. This package is not too hard to solder and still is relatively small. The microcontroller operates at 1 MHz. Here, you have two options: just use a crystal of 1 MHz or an 8 MHz device and set the CKDIV8 fuse when programming.
The 2x16 LCD is used to show information and allow the user to configure the device. It's being used in 4-bit mode and the LED backlight is controlled by the microcontroller through N-channel FET T2.

The keyboard input is implemented using one ADC input (ADC6) instead of several (digital) ports. That way there is no need for a controller with more input ports and consequently no larger package is needed. The voltage the ADC reads depends on the key pressed. When no key is pressed, the ADC will read roughly 5 V. All ADCs work in 10 bit mode, which means the value read lies between 0 and 1023. With 5 V on the ADC pin, the software will read 1023.
If a key is pressed, the voltage on the ADC pin may be calculated using the formula:

$$V_{ADC} = V_{CC} - (R15 \times V_{CC}) / (R15 + R_{SW})$$

where V_{CC} = 5 V, R15 = 10 kΩ and R_{SW} = 1.5 kΩ, 5.6 kΩ, 15 kΩ or 68 kΩ depending on the switch pressed.

The four keyboard switches have the following functions: MENU, MINUS, PLUS and ENTER, with which you can adjust the setting of the TimeClick.

ADC5 is used for sensor input. The sensor is connected through a 3.5-mm jack socket. At the tip there is 5 V available for powering the sensor. The ring carries the sensor output signal and the shield is grounded.
There are three types of sensor included in the schematic: a light sensor, a sound sensor and a vibration sensor (piezo). But of course the sensor range can be extended to whatever sensor you need. To avoid connection mistakes, the sensor input jack and the output jack have different sizes.
The output signal is available at a 2.5-mm jack socket, as found on some Canon cameras. For safety reasons the output of the microcontroller is coupled to the jack via an optocoupler.
This device can be used with many different camera models (see inset ***Camera Compatibilty Guide***); you only need to have the right adapter cable and the camera should be able to work with the implemented protocol.

This protocol is rudimentary: it uses three pins: 1—ground; 2—ring and 3—tip. When pins 1 and 2 are shorted, the camera behaves the same as when the shutter button is half way pressed. When pins 1 and 3 are shorted, the camera behaves like the shutter button is fully pressed.

COMPONENT LIST

Resistors
R1,R10 = 4.7kΩ
R2,R13,R14 = 1kΩ
R3,R4,R12,R15 = 10kΩ
R5,R9 = 100kΩ
R6 = 470kΩ
R7 = 47kΩ
R8 = 1Ω
R11 = 1MΩ
R16 = 68kΩ
R17 = 15kΩ
R18 = 5.6kΩ
R19 = 1.5kΩ
P1 = 10kΩ trimpot
P2 = 100kΩ trimpot

Capacitors
C1,C2 = 22pF
C3,C6,C7,C8,C9 = 100nF
C4,C5 = 10µF 25V radial
C10 = 100pF

Semiconductors
D1 = BPW16N photo transistor
D2 = 5.1V 400mW zener diode
T1 = BC547
T2 = BS170
IC1 = LP2954IT
IC2 = ATTINY861-20SU, programmed, Elektor # 100371-41
IC3 = LM358P
IC4 = CNY74-2

Miscellaneous
K1 = 6-pin (2x3) pinheader, lead pitch 2.54mm (0.1'')
K2,K3,K4,K5,K7,K10,K11 = 3-pin pinheader, lead pitch 2.54mm (0.1'')
K6,Bt1 = 2-pin pinheader, lead pitch 2.54mm (0.1'')
S1 = switch, SPST
S2,S3,S4,S5 = 6 mm tactile switch type MC32830
K8 = PCB jack socket, 3.5 mm, stereo, e.g. Lumberg type 1503 09
K9 = PCB jack socket, 2.5 mm, stereo, e.g. Lumberg type 1501 06
X1 = 8MHz quartz crystal
LCD1 = LCD, 2 x 16 characters, Lumex type LCM-SO1602DSF/A

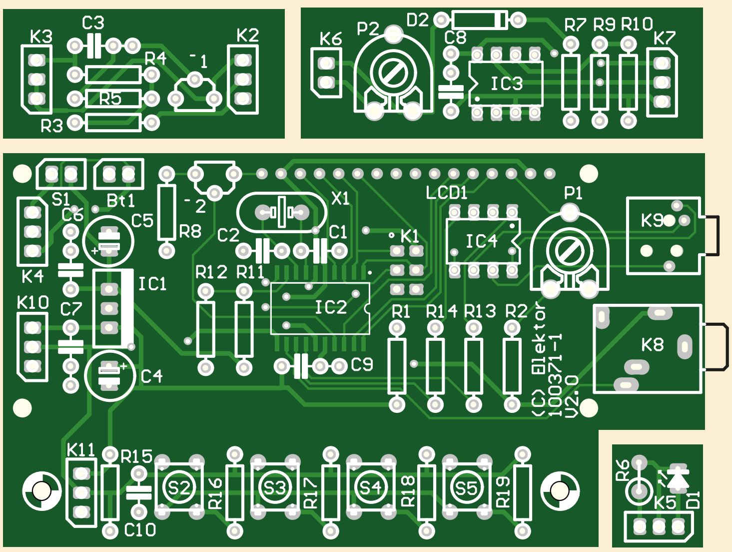


Figure 2. Component layout

Sensor operation

The way the sensors work is very easy to understand. The microcontroller reads the ADC input receiving the voltage a sensor generates. Regardless of the type of sensor used the microcontroller waits for a transition (rise or fall) across a trigger value. When this condition is met, it acts accordingly depending on the configuration, i.e.

Camera Compatibility Guide

TimeClick was successfully tested with a Canon 500D, which has a 2.5 mm jack intended for a remote control (E3-type) and a Canon 7D that has a Canon N3 connector for the same purpose. As a result, all cameras from Canon with 2.5 mm jacks should be compatible with TimeClick. This includes models 1000D, 550D, 500D, 450D, 350D, 300D and 60D.

The N3 connector is present in Canon models EOS 1D, 5D, 5Dmk2, 7D ,10D, 20D, 30D, 40D and 50D, so these should be compatible too. Of course in this case — since N3 connectors are difficult to obtain — the easiest way is to buy a special connecting cord or a cheap remote trigger and just use the N3 connector. Connecting cords are available from [3] for example (go to 'Remote Accessories' in the 'Remote Cords/Wireless/Infrared' section).

Other brands haven't been tested, but a search using Google revealed that all Pentax cameras use the same pinout as the Canon cameras, so it's very likely they are compatible. Nikons have different connectors, but they have the same basic functionality. They use an MC-DC1 connector on models D70, D70s and D80, a MC-DC2 connector on models D90, D3100, D5000 and D7000 and a 10-pin connector with different names (MC-20, MC-22, MC-30, MC-36) on models D200, D300, D700, D3 and D3x... So perhaps they can be used with TimeClick too. We would like to hear from you if they do (or don't). Everyone's invited to post their findings on the Elektor forum.

waits a configured time expressed in ms and then fires the shutter.

Always turn off the power before plugging and unplugging the sensors. This way short-circuits inside the mini-jack connectors are avoided.

Software description

The source code, freely available from [1], was written completely in C language using the very efficient compiler avr-gcc. About 99% of the 8 K flash space of the microcontroller is used to store the program. It was hard to squeeze all the features into the device. Several code optimisations were required. All configuration data are preserved in the EEPROM of the microcontroller.

The program operation is clearly commented; the program does the initial setup and then enters an endless loop. Inside this loop is where all the action happens. TIMER0 is configured to generate a pulse every second to take care of all timing multiples of 1 second.
To allow for maximum flexibility, there are 12 different profiles where the user can save different operating configurations. Each profile can be renamed to indicate the function it was created for. For example, the user can have a profile for 'Lightning', one for 'Drops', and so on. Included in the source code at [1] is a file with the EEPROM contents of several preconfigured modes.

Assembling & Programming

A printed circuit board and a programmed microcontroller can be purchased from the Elektor Shop [1]. When assembling your device, start with the lowest profile components and work your way up to the tallest ones. **Figure 2** shows the PC board component layout. The LCD and switches need to be mounted on the copper side of the PCB. Once the board is populated, it is time to put some intelligence in! The programming of the Atmel microcontroller is done by way of ISP connector K1. Assuming you're using AVR Studio and AVR Dragon / AVR ISP programmer, here is what you should do:

First choose the appropriate device in the menu *Project -> Configuration options*.
Next, the programmer and the TimeClick device should be connected. However, since the author experienced some mysterious occurrences in this step, he advises to stick to the following order:

1. With both TimeClick and AVR Dragon/ISP powered OFF, connect the ISP cable between them.

2. Then connect the AVR Dragon/ISP to a USB port capable of supplying more than 300 mA.

3. Now you can power up TimeClick.

Next, go to the menu *Tools -> Program AVR -> Connect* to choose the appropriate programmer/port and press Connect. **Figure 3** shows the window you should see right now. It's important that the programming mode is set to ISP and that the ISP frequency is set to 125 KHz (< ¼th of the device clock). If everything checks out, an 'OK' should be returned after you press *Read Signature.*

Now we're ready to set the fuses to the correct values as shown in **Figure 4**. SPIEN is active as factory default. If you used an

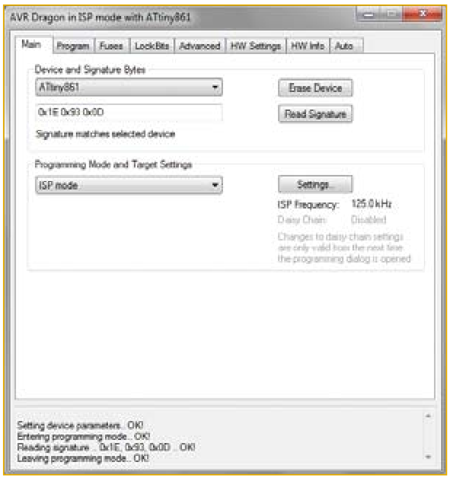


Figure 3. Connecting to the device.

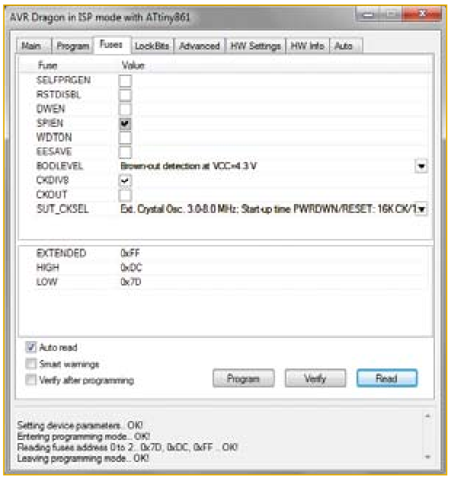


Figure 4. Setting the fuses correctly.

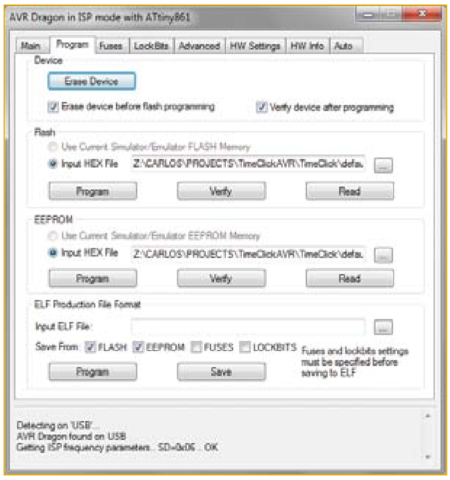


Figure 5. Programming the microcontroller.

8 MHz crystal, you should also check the CKDIV8 fuse as shown in the screen capture. It is also important to set the correct clock source, in this case select 'External Crystal 3.0-8.0MHZ'. The Brown-out detection could avoid EEPROM corruption with batteries at the end of life, so set this to 4.3 V.

Finally, it's time to program the device. All the needed files are inside the ZIP-file associated with this article (100371-11.zip, see [1]). Download and extract this file. In the flash area, select the file TimeClick.hex and hit the Program button (**Figure 5**). At the end, the device will reboot.

If you want to use the preconfigured EEPROM settings, in the EEPROM area select the file TimeClick_Configured_eeprom.hex (also inside the downloadable zip package) and hit the Program button.

Software Operation

With the device programmed, it's ready for use. When the ENTER key is pressed and held during device power on, a reset to factory defaults of all configuration possibilities can be carried out. This option will erase all saved configurations, even the preconfigured EEPROM settings mentioned before. In the main screen the following information is always visible:

- Battery level in steps of 25%
- Sensor mode
- Operation mode
- Active profile
- Number of photos taken so far
- Metering setting
- Bulb setting
- Mirror lock setting
- Exposure bracketing setting

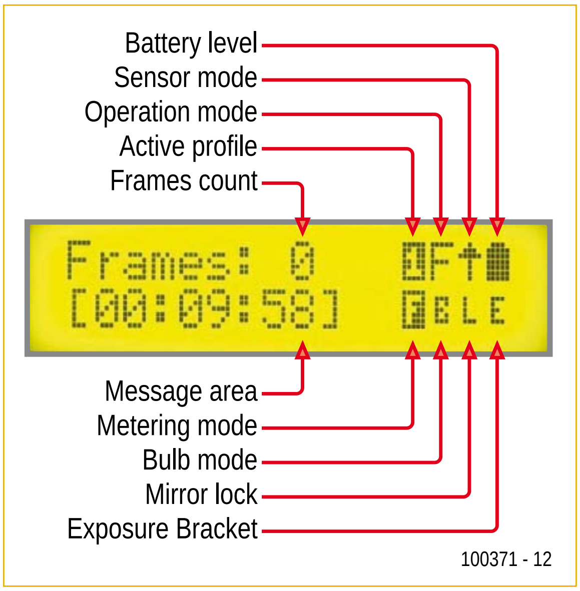


Figure 6. Information shown on display.

Some of this information is presented using custom characters defined in the software.

Pressing the MENU key while the main screen is being displayed, enables the user to enter the menu where the configuration takes place. The menu is intuitive and easy to navigate. Press MENU to go back, PLUS and MINUS to change values and ENTER to go forward. For a more complete reference about the operation of the TimeClick, there is a user manual available at [1] and the author's website [2].

(100371)

Internet Links

[1] www.elektor.com/100371

[2] http://timeclick.no.sapo.pt

[3] www.enjoyyourcamera.com

MIAC Controlled Underfloor Heating System

Totally programmable with Flowcode

By Ben Rowland (United Kingdom)

This heating system sketched in this article keeps you warm in cold times and with the help of Flowcode software is designed for total adaptability to heating capacity and other parameters.

The MIAC (Matrix Industrial Automotive Controller) supplied by Elektor is an industrial grade control unit similar to a PLC but more feature packed and easier to program without having to resort to using ladder logic. It's based on the powerful 18F4455 PIC microcontroller and can be directly connected to USB, making programming — via Flowcode, C or Assembly — a breeze. An LCD, pushbuttons, four relay outputs, eight inputs — selectable analogue or digital — and a CAN connection complete the system. The main purpose for the MIAC is industrial applications. Hence it makes use of voltages of 12 volts instead of the 5 volts normally applied in PIC microcontroller systems.

Practical application...

In this article we demonstrate how we can replicate an expensive underfloor home heating system at a fraction of the normal costs. This heating system consists of a few major key elements:

- a boiler
- electronic valves
- a thermostatic mixing valve
- a central heating pump
- an air release valve
- PEX underfloor piping
- an AC power residual current device (RCD)
- temperature sensors
- a MIAC
- compression fittings to assemble manifold

...and implementation

Figure 1 shows a basic schematic of a two loop underfloor heating system. We let the MIAC use a lookup table technique to read the temperature of thermistors t1 and t2, which are situated in the floor near the PEX heating loops. The lookup table data is generated using an Excel spreadsheet with values matching those of the thermistors (included in the free download 100871-11.zip from the Elektor website [1]).

When the temperature of the thermistors drops below a threshold value, we check to see if the individual loops are enabled. If they are, we open the valves connected to the individual loops. We then switch on the pump and the boiler. An example program — *Heating System.fcf* — is included in the download from [1].

As the water from the boiler output starts to heat up, the thermostatic mixing valve does its work and starts to mix the cold water from the output of the PEX loops with the hot water from the boiler. We can monitor the temperature of the water running through the PEX loops by reading thermistor t3. When this temperature has reached the required level, we can shut off the boiler and we can also shut off the pump.

Every so often we can activate the pump for a bit to circulate the water and ensure that it is still up to temperature.

Please note that the RCD is an essential part of the system, as it could make the difference between a nasty shock and death if you were to come into contact with a live cable.

With reference to the existing boiler and thermostat, wiring should be implemented so that the room thermostat can still work when the underfloor heating system is not running.

The example program is very basic and simply checks the return temperature t3. When t3 drops below a threshold value, the temperatures of t1 and t2 are read into the system. Depending on these temperatures the valves to the PEX loops are opened and then the boiler and the pump are switched on. When the return temperature t3 returns to a value above the switch off temperature the pump and boiler are switched off.

Improvements

The program could be improved by allowing loop1 and loop2 to be enabled or disabled separately to allow for zones to be left unheated if required. Another way to improve the system would be to add a timer functionality to allow the temperatures to vary, for example drop slightly during the night.

The file *MIAC_Underfloor_v1.1.fcf*, found in the download 100871-11.zip on [1], contains the latest version of the author's thermostatic heating controller program for the MIAC. It has lots of functions and allows you to save up to 40 programmable events, directly control the system, it has fill and drain modes, temperature settings for each zone, back-off temperature settings and a lot of other bits and pieces. The green menu button accesses the main menu that allows you to set up the device.

The author decided to simply use thermistors for each zone for now rather than create a CAN network of sensor nodes mainly because of time limitations on the project. The thermistors he used (Rapid # 61-0410) are simply connected between 12 V and the MIAC input terminal and the lookup table provided in the file works to translate the readings to degrees centigrade.

If a warm-water based system is not for you, then the MIAC can also be used to directly drive electrical underfloor heating elements. The relay contacts are rated up to 1800 – 2000 W at AC grid voltage allowing you to drive small through to large,

Elektor Products & Services

- MIAC, ready assembled*
- MIAC and Flowcode 4 Bundle*
- 3x MIAC and Flowcode 4 Bundle*
- USB A to B mini lead*
- Flowcode program: # 100871-11.zip**

* available at www.elektor.com/miac
** accessible through www.elektor.com/100871

high power heating mats. To do this, you would connect the Neutral and Ground signals from the AC powerline to the heating mat. Then connect the Live from the AC grid through one of the MIAC relays to the Live from the heating mat.

If you want full electrical separation when the mat is not in use, then you will have to use a second relay to connect and disconnect the Neutral connection. Again an RCD should be used along with a great deal of precaution to avoid any injury caused by contact with AC powerline voltage. Electrical underfloor heating mats should also be placed on an insulated layer to avoid a high percentage of the heat escaping directly into the ground.

Word of warning

Please note that all electrical work will need to be inspected by a qualified electrician. Electrical regulations may differ from country to country, so make sure you check your local legal requirements. Also the wattage quoted for the electrical heating mat is subject to a 250 VAC system. The amperage on the relay outputs, the screw terminals and the PCB tracking of the MIAC is rated to 8 A. Therefore at 250 V the theoretical power is 250 V × 8 A = 2000 W. At 110 V the wattage drops to a theoretical maximum of 880 W.

The author is in no ways a qualified plumber or a qualified electrician so got advice from qualified personnel when installing the system and then signed off the electrical and heating systems with approved professionals. If you decide to do similar things using domestic AC grid voltages or plumbing systems, then please ensure to get help and advice from qualified professionals in your area before you begin and again before you switch on or commission the system.
Some pictures of the author's installation work can be found on [2], where more information about MIAC and Flowcode is also available and help is on stand-by.

(100871)

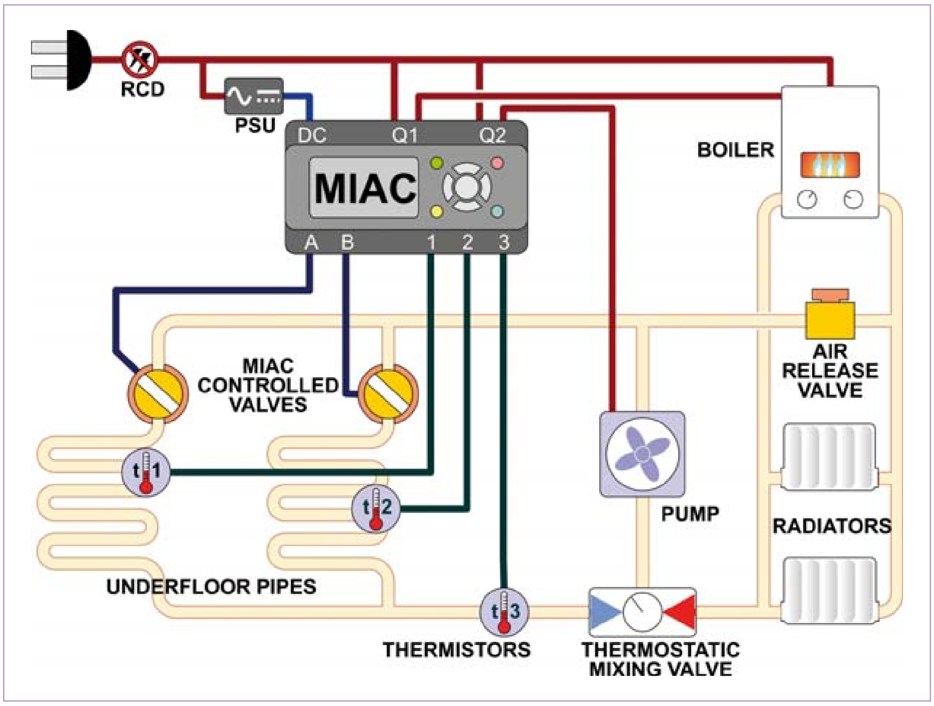


Figure 1. Schematic diagram of a heating system using an Elektor MIAC as the controlling unit.

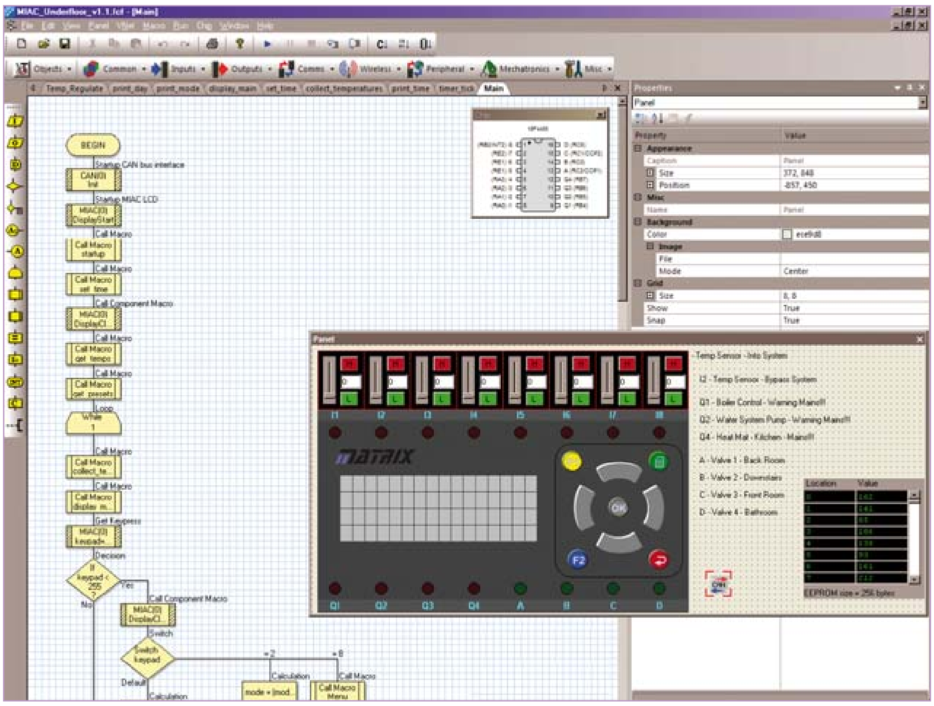


Figure 2. Using Flowcode for programming the MIAC is quite straightforward.

Internet Links

[1] www.elektor.com/100871

[2] www.matrixmultimedia.com/mmforums

Linux'ed Telephone-to-VoIP Adapter

By Angelos Varvitsiotis (Greece)

This open hardware & software project is a USB-connected interface that links a Voice-over-IP (VoIP) system to an analogue phone set or similar equipment like an analogue exchange. The powerhouse board works under Linux using the renowned Asterisk IP PBX software, and at a stroke enables you to use your home telephone set to connect to the VoIP world.

Are you ready for some acronyms? Engineers use them all the time and the telephony/telecomms industry has plenty! Like: an FXS (Foreign eXchange Subscriber) interface (the plug on the wall) delivers POTS (plain old telephone service/system) from the local phone company's CO (Central Office) and must be connected to subscriber equipment like telephone sets, modems, and fax machines. In other words, an FXS interface points to the subscriber. An FXS interface provides the following primary services to a subscriber device: dial tone, battery current and ring voltage. Note that all three services come with different values and parameters in countries around the world. Sometimes the FXS acronym is also rendered as Foreign eXchange System.

Now, an FXO (Foreign eXchange Office) interface (the plug on the phone) receives POTS, typically from a CO of the PSTN (Public Switched Telephone Network). In other words, an FXO interface points to the Telco

Elektor Products & Services

- PCB, bare: # 100761-1
- PCB artwork: # 100761-1.pdf
- PIC18F2550-I/SO, programmed: # 100761-41
- Source code: # 100761-11.zip
- Hyperlinks in article

Items accessible through www.elektor.com/100761 office. An FXO interface provides just one primary service to the Telco network device: on-hook/off-hook indication (loop closure). As illustrated in **Figure 1**, a telecommunications line from an FXO port must connect to an FXS port in order for the connection to work. Similarly, a line from an FXS port must connect to an FXO port in order for the connection to work. When the FXO port on your analogue telephone is connected to the FXS port in the wall, you receive FXS service from the Telco — and assuming your bill is paid, you hear a dial tone when you pick up the phone. Note the arrows in Figure 1, they illustrate the pointing.

If you connect an FXS device to another FXS device, the connection will not work. Likewise, if you connect an FXO device to another FXO it will not work either. So, for example, you can not plug a standard analogue telephone (FXO) directly into a standard analogue telephone (FXO) and talk phone-to-phone.
That's FAB (full acknowledgement of broadcast, tnx Thunderbirds) but with the arrival of VoIP (voice over internet protocol) we don't need the Telco anymore and that begs the question: can I connect an analogue phone (system) to VoIP? The answer is: YES (yes, exquisitely so) using Linux and a dedicated converter designed for use on the USB (universal serial bus).

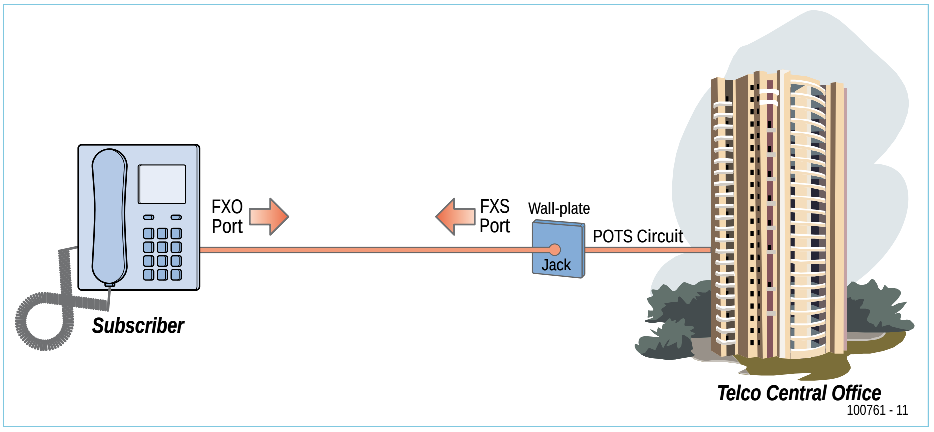


Figure 1. With the arrival of VoIP this is all history but you may appreciate a bit of remedial teaching on basic telephony and the quirky acronyms used.

The circuit

The schematic of the adapter is shown in **Figure 2** — this should be a treat for all fans of microcontrollers, embedded applications and open source platforms including Linux. The analogue phone is connected to J1 and the PC's USB port to... 'USB'! Simple as that, the Linux environment on the PC and the firmware running inside the PIC do all the work and you can start phoning for free using your trusted analogue phone.
At the heart of the circuit is a Silicon Labs Si3210 microcontroller [1] in its default application circuit. This chip, also called a SLIC (subscriber line integrated circuit), controls the telephony-related functions of the board, namely:

- a DC-DC converter that generates the necessary voltage to drive the subscriber line;
- the actual subscriber line;
- the analogue-to-digital and digital-to-analogue conversion (PCM codec).

The SLIC is complemented by an analogue line driver type Si3201, again from SiLabs. The circuit also comprises the DC-DC converter analogue circuitry, consisting mainly of D1, L1, Q7 and Q8 plus surrounding components. The converter is driven by a PWM signal from the 3210's DCDRV output. Besides the analogue telephony interface, the chip uses two digital buses to communicate to the (digital) world: a PCM bus and an SPI bus. Both are controlled by a PIC18F2550 microcontroller running some clever firmware.
The parts marked with an asterisk (*) have values optimised for a phone line length of up to 2,000 feet (approx. 700 m) and a ring voltage of 45 V_{rms}.

The PIC (ticking at 20 MHz) and its firmware accomplish a large number of tasks: packing PCM 1-ms audio samples into USB pack-

How 2 make it come alive

There are plenty of reasons why an otherwise perfect board would not work at first try. The first step in giving life to the board is to bring up the bootloader, as explained in the blog at [5b].

This is done by switching on S1b (i.e. the DIP switch closest to the USB plug) in order to invoke the bootloader and enable the use of a USB bootload utility like PICDEM or fsusb (the latter on Linux) to load the firmware.

However, this applies to a pre-programmed PIC. In the case of a fresh, empty PIC, you first need to flash the bootloader firmware from here:

http://openusbfxs.googlecode.com/svn/trunk/PIC18FSource/Bootloader-FXSMOD/bootloader.hex

using a PIC programmer that will do In-Circuit-Serial-Programming (ICSP). Once the bootloader has been flashed, you need to switch S1b and plug the board into a USB in order to invoke the bootloader.

The final step is to use PICDEM-FS or fsusb to load the actual adapter firmware pulled from [8].

Flashing just the FXS firmware without the bootloader will cause the board to fail to work. This is because the bootloader firmware takes care of jumping into the right places within the FXS firmware during reset and interrupt sequences. Obviously, if the bootloader is not installed, this is not going to work.

Other than that, please check the author's blog page for more hints and advice in the rare case that your adapter does not work straight off.

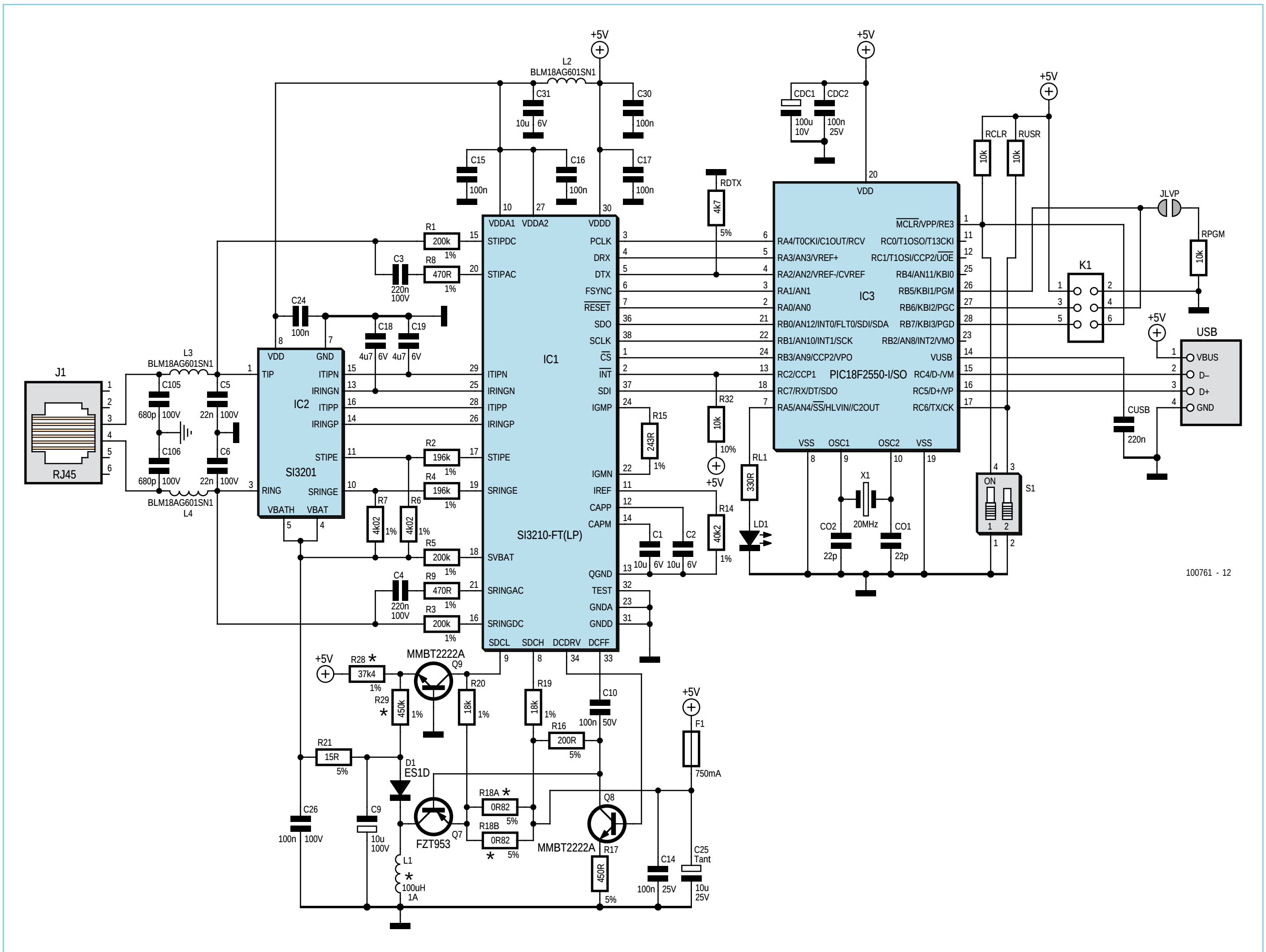


Figure 2. The core of the circuit is a SLIC (subscriber line integrated circuit) from SiLabs while most of the intelligence resides in the PIC18F2550 microcontroller.

ets (and vice versa), driving the 3210's PCM bus clock, transmitting and receiving isochronous data over USB, keeping the PCM bus time in synchronism with the USB bus 'ticking' and, finally, managing all other generic functions required for USB communication. While the SiLabs ICs are seen in a more or less standard constellation following the manufacturer's datasheets and reference circuits, the originality of the circuit of course resides in the PIC firmware, which manages all the above functions in real time, and thus obviates the need for an expensive FPGA device (sooo corny now in telephony designs) and keeps the whole cost very low. Pinheader array K1 is the PIC ICSP (in-circuit programming) connector.

Construction – yes you can, too

All Linux fans should want to build this, so a nice compact board (60 × 30 mm) got designed for the adapter and best of all it's available ready-made from Elektor (# 100761-1) and the same goes for the programmed PIC (# 100761-41). Figure 3 shows the board stuffing plans for the top and bottom sides. The board is brimful of SMD parts at both sides, including the three integrated circuits so assembly might not be too easy if you've never worked with these tiny devices before. The Si3210 is troublesome with its extremely fine pitch leads, which may push your limits in terms of accuracy and power of sight.

All fears aside, the prototype was successfully built by Jan Visser of Elektor Labs using manual soldering throughout at both sides of the board and the 'solder braid trick' to separate the Si3210 pins electrically after mass-soldering. FYI Jan is spectacled and at the safe side of 50, but just. The result is shown in Figure 4. Choke L1 is very conspicuous on the board with its two large solder pads.

As a tip-off, SiLabs operate a sampling service for their ICs. Our ICs however were obtained through Mouser with a little help from CJ and Valerie at our sister magazine Circuit Cellar based in Vernon, CT.

Firmware

The PIC firmware may be downloaded free from the Elektor website [2] and is upgradable over USB, using a Microchip-supplied tool called PICDEM [3]. This function is accomplished by DIP switch S1(b), which controls whether the board boots in bootloader or normal mode. Firmware upgrades can also be performed by a utility program. The other DIP switch, S1(a) resets the PIC, it should be kept in the OFF position.

COMPONENT LIST

Resistors

(default: SMD 0805)
R1,R3,R5 = 200kΩ
R2,R4 = 196kΩ
R6,R7 = 4.02kΩ
R8,R9 = 470Ω
R14 = 40.2kΩ
R15 = 243Ω
R16 = 200Ω
R17 = 453Ω*
R18A,R18B = 0.82Ω*, shape 1206
R19,R20 = 18kΩ
R21 = 15Ω, shape 1206
R28 = 37.4kΩ
R29 = 453kΩ*
R32,RCLR, RPGM, RUSR = 10kΩ
RL1 = 330Ω, shape 1206

Capacitors

(default: SMD 1206)
C1,C2,C31 = 10µF 6V
C3,C4= 220nF 100V, shape 1812
C5,C6= 22nF 100V, shape 1812
C9 = 10µF 100V radial
C10,C14,C26 = 100nF 100V, shape 1210
C15,C16,C17,C24,C30= 100nF, shape 0603
C18,C19 = 4.7µF 6V
C25 = 10µF 25V tantalum bead
C105,106 = 680pF 100V
CDC1 = 100µF 10V radial
CDC2 = 100nF 25V
CO1,CO2 = 22pF
CUSB = 220nF

Inductors

L1 = 100µH 1A, SMD
L2,L3,L4 = 150µH 1A, SMD, type BLM18AG-601SN1, 0603 shape

Semiconductors

(all SMD)
D1 = ES1D (SMB)
IC1 = Si3210-FT/GT (TSSOP38-LP), SiLabs, Mouser # 634-SI3210-GT
IC2 = Si3201-FS/GS (ESOIC-16T), SiLabs, Mouser # 634-SI3201-GS
IC3 = PIC18F2550-I/SO, programmed, Elektor # 100761-41, see [2]
LD1 = LED, green (1206 CHIPLED)
Q7 = FZT953 (SOT230P700X160-4N)
Q8,Q9 = MMBT2222A (SOT95P280X13-3N)

Miscellaneous

F1 = fuse, 0.75A, shape 1210
J1 = RJ-11/12 socket, PCB mount
JLVP = jumper or temporary wire link
K1 = 6-pin (2x3) pinheader block, 0.1"
S1 = 2-way DIP switch, SMD
U1 = USB-A-H socket
X1 = 20MHz quartz crystal
PCB, Elektor # 100761-1, see [2]

* for phone cable length up to 2,000 ft. and V(ring) = 45 Vrms.

Figure 3. Component mounting plan of the VoIP adapter board. The bare board is available from Elektor.

Linux driver and Asterisk

The board is accessible via a Linux device driver. The author has chosen to integrate the board with the Dahdi [4] device driver family, so that the board can be used under the open-source Asterisk IP PBX system. Extensive instructions on how to build the driver and integrate the board into a Linux system can be found in the author's blog [5]. When the 'oufxs' device driver module is compiled and loaded into the Linux kernel, the system recognises the USB FXS as soon as it is plugged. The verbosity of debug messages is tuneable: while terse by default, with the 'debuglevel' parameter set

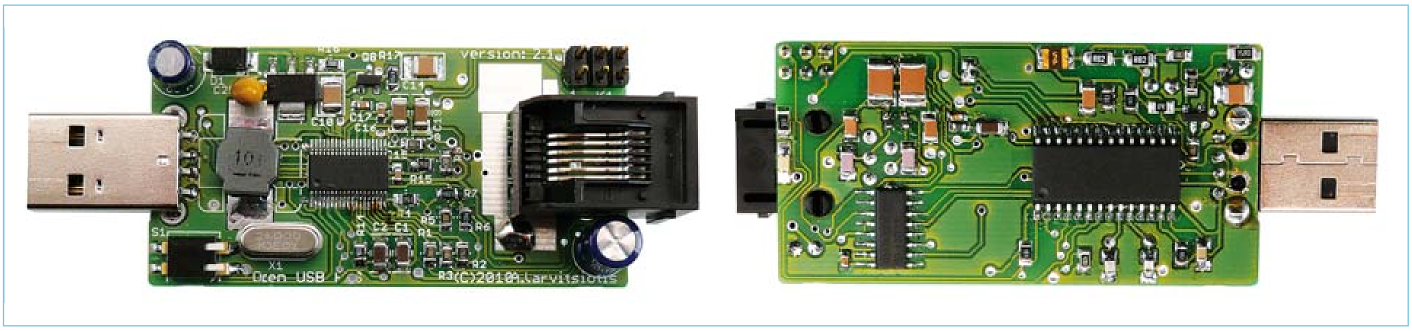


Figure 4. A close look at both sides of the board. Check your soldering against this!

to 4, the driver displays in detail all its steps while initializing the board (Figure 5). By now, you have a new Dahdi device, that you can see and manage with utilities like dahdi_scan.

The next step is to configure the device's signalling, i.e. the electrical method by which the system tells the subscriber that the line has become available, or that the other party has hung up. While a phone set does not care much about these, analogue exchanges do, so Dahdi support various signalling methods. The author uses 'fxols' signalling, which stands for 'Loop-Start'. To do this, the '/etc/system/dahdi.conf' file must be created or edited and a line reading 'fxols=1' must be added at the end. If you would like to change the default ring and dial tones as well, that's the place to do it, by selecting a 'tonezone' other than the default 'us'. You can also add an echo canceller e.g. by adding the following line 'echocanceller=oslec,1'. Finally, the 'dahdi_cfg' utility must be run.

You are now ready to start with Asterisk. Follow the author's instructions for setting up and configuring Asterisk [9], then start Asterisk in debug console mode ('e.g. asterisk –vvvvvvc'). Pick up the phone; Asterisk should note an off-hook event, and you should be listening to a dial tone. Then, dial '600' (and wait a bit for an inter-digit timeout); Asterisk will log a console message and start an 'echo' application. You can then speak on the phone and listen back to your own voice with a few tens of milliseconds delay.

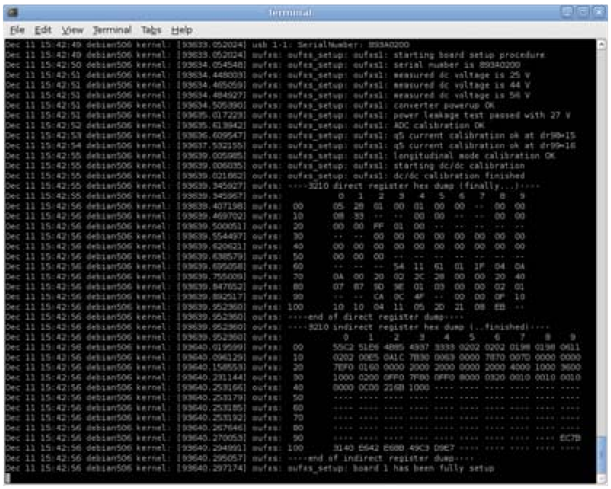


Figure 5. The Linux 'oufxs' driver sees a board plugged and takes action.

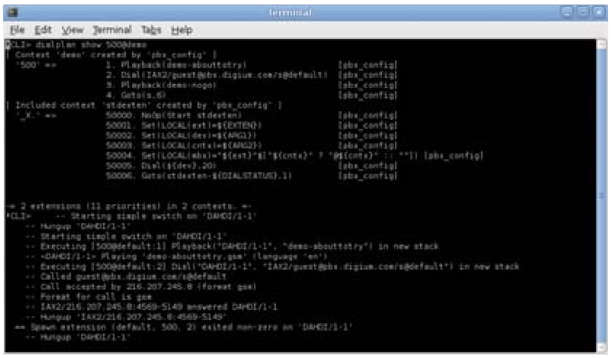


Figure 6. Displaying the dialplan and logging a call from Asterisk's console.

Listening to one's voice may be a great tool for debugging, but it is not of much use, is it? So what about placing your first free international VoIP call using IAX (the Inter-Asterisk-eXchange protocol)? Hang up, then pick up the phone again and dial 500. This will route a VoIP call to Digium's demo IAX server in the United States (Figure 6). Digium are the people behind Asterisk. You will hear a ring tone, and then Digium's own Asterisk system answers the call (beware: this is a real VoIP PBX, and if you dial an extension, you will probably reach a Digium employee).

An open project

All the parts of the design, including the schematic and PCB design files in Cadsoft Eagle format [6], the firmware [7] and the Linux driver software are open-source, meaning the work is licensed under the GNU Public Licence (code) and Creative Commons licenses (PCB, documentation, etc.). The source code can also be found at [8]. All Elektor readers are invited to improve and extend the software to their heart's content and let the Editor, the author and members of the community know by way of the Elektor forum (main topic: Microcontrollers & Embedded).

(100761)

Internet Links

[1] www.silabs.com/Support%20Documents/TechnicalDocs/si3210.pdf

[2] www.elektor.com/100761

[3] Microchip PICDEM FS: www.microchip.com/Microchip.WWW.SecureSoftwareList/secsoftwaredownload.aspx?device=en021940&lang=en&ReturnURL=http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en021940&part=DM163025#

[4] http://downloads.asterisk.org/pub/telephony/dahdi-linux/releases/dahdi-linux-2.3.0.1.tar.gz

[5] (a) http://openusbfxs.wordpress.com/ and (b) http://openusbfxs.wordpress.com/dyi-setup-and-debugging-guide/

[6] http://code.google.com/p/openusbfxs/source/browse/#svn/trunk/Eagle-OPENUSBFXS-Dongle

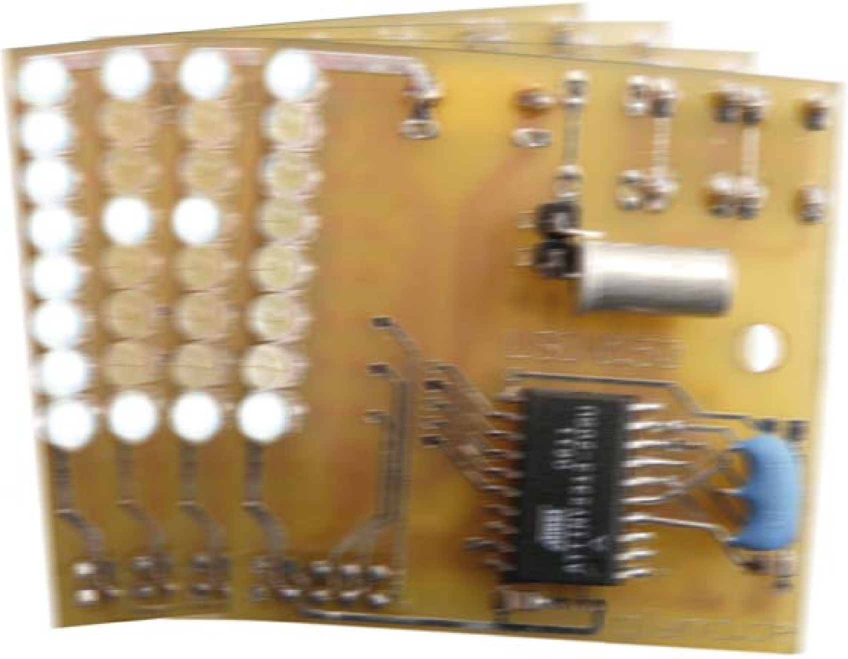
[7] http://code.google.com/p/openusbfxs/source/browse/#svn/trunk/PIC18FSource/OPENUSBFXS-FMWR

[8] http://code.google.com/p/openusbfxs/source/browse/#svn/trunk/LinuxDahdiDriver

[9] http://openusbfxs.wordpress.com/getting-started-with-asterisk/

AlphaLED Shaker

Magic message machine



By Kurt Schuster (Germany)

Give the board a shake, and letters fall out? Not quite: but you can make your favourite message appear as if suspended in thin air.

The basic principle will be familiar: as a row of LEDs is moved from side to side, the LEDs are driven in such a way as to exploit the persistence of human vision and create the illusion of letters, symbols and other graphics floating in mid-air. This Reader's Project is a new twist on that idea: the board carrying the row of LEDs is moved by hand rather than being driven by a motor. We use a fairly sophisticated motion detection system to let us synchronise the display of text with the movement of the unit, and the text can scroll to allow longer messages to be displayed. As a bonus, it is possible to edit up to four short messages 'on the fly'.

Shaken, not stirred

The author persuaded some friends of his to try out the device: an occupational therapist, a graphic artist in the advertising industry, and a gaggle of children. Impressions were positive: the occupational therapist thought it would be a good way to motivate his patients, particularly the younger ones, with the combination of the larger-scale shaking moves and the delicate motor skills required to operate the pushbuttons to enter messages having 'definite potential'. That the device might also encourage younger children to learn the alphabet through play he saw more as a side-effect. The graphic designer also saw potential in the unit, and had the witty idea of mounting one on the side of a cocktail shaker to display an advertising message. Children, especially older children who have already learned to read, quickly got the hang of using the buttons and shaking the device to create their own messages. The ease with which the text could be read at a distance was mentioned as a particularly good feature, and they were inspired to make the board into a toy by decorating it with colourful characters and animals. Calling all *Elektor* readers with a sideline in toy design…

Sensor and sensitivity

In principle the moving-LED idea could be applied to a yo-yo, with the text of a message being spelt out as the yo-yo goes up and down. However, the motion of a yo-yo proved too slow to produce a satisfactorily stable visual effect, and hence we settled on the letter shaker approach.

The basic ingredients, comprising a few LEDs and a microcontroller, were easy enough to find and solder together. Harder to solve was the problem of finding a suitably sensitive shake sensor without having to resort to a fully-featured accelerometer. Two homebrew attempts, one using a steel ball and the other using a magnet in a tube with switches (both mechanical and reed-contact) at either end, were not really up to the job. Several commercial tilt switches were tested, and the readily-available (from Farnell or Conrad Electronics) Assemtech CW1300-1 was found to work well. Surprisingly, taking the sensor apart reveals two apparently gold-plated balls, one small and one large. The smaller one bridges the contacts, while the larger one presumably ensures that contact is made with sufficient pressure.

The software was written in AVR assembler and was, at least initially, kept very simple. A fixed text graphic was read from memory and output sequentially to the LEDs when the ball in the (home-made) sensor pressed one of the two switches. To get a stable display it proved necessary to compensate for the inertia of the sensor using a variable delay in the code to synchronise the

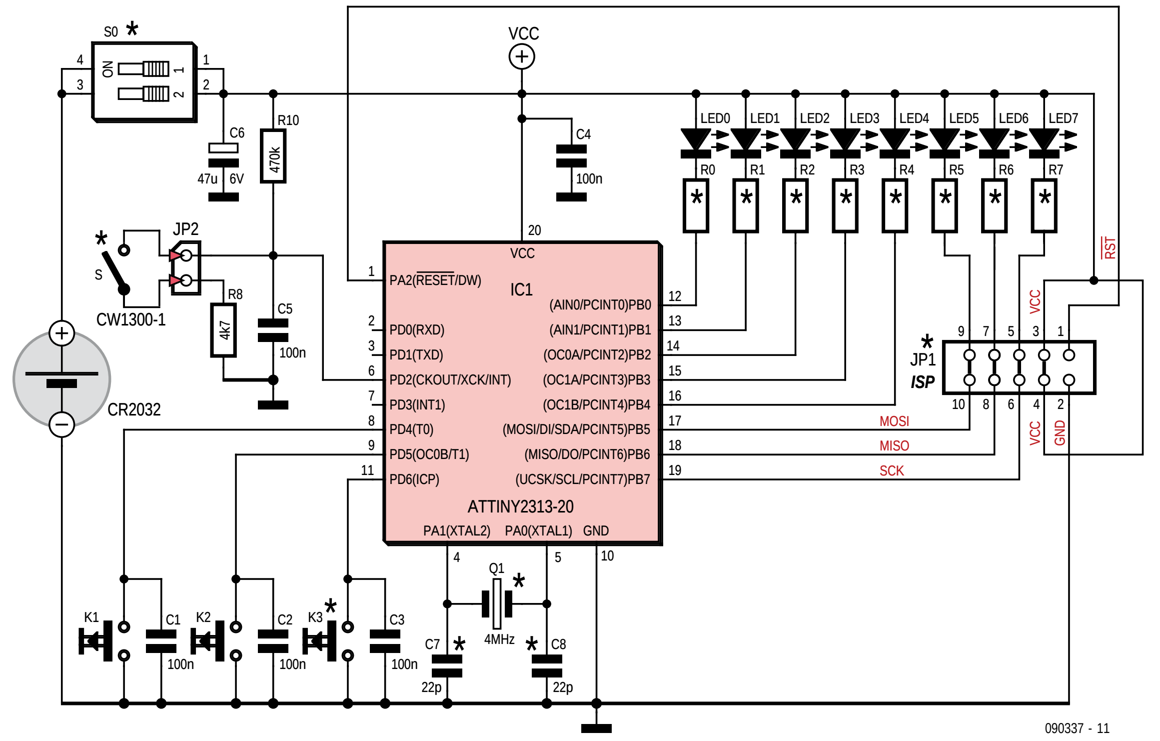


Figure 1. A tilt switch controls the drive timing of a row of LEDs via an ATtiny microcontroller. The key to the circuit is in the software.

LED data output. It was particularly tricky to arrange things so that the message did not appear in two different positions as the device was moved to and fro, reducing the overall perceived brightness. As the program was developed more and more functions were added, with the bulk of the (thoroughly commented) code [1] concerning the user interface, control logic, character sets, message scrolling, LED drive and polling the control buttons. Shake detection is done under interrupts. Sensor inertia compensation accounts for very few lines of code, but a disproportionately large amount of know-how, including a detailed analysis of the shake movement using an ADXL320 accelerometer. It was only armed with this knowledge that we could make the final version of the device work with just a tilt switch.

Circuit and construction

The tilt switch is connected across the pins of JP2 (see the circuit diagram in **Figure 1**). It is debounced by R8, R10 and C5 and then taken to port pin PD2 of the ATtiny microcontroller. Two miniature buttons (K1 and K2) connected to PD5 and PD8 provide the user controls for entering text. K3 is not used in the current version of the software, and so is not fitted. A DIP switch (S0) on the printed circuit board (**Figure 2**) is used to turn the device on and off; the two-pole version would also be suitable. Power is provided by a 3 V type CR2032 button cell, and of course observing correct polarity is essential when soldering its holder (type SMTU-2032-1). The best LEDs to use are high brightness 3 mm types with a wide viewing angle; these are connected to the microcontroller outputs PB0 to PB7. To make the resulting display as pleasingly solid as possible, the LEDs are mounted tight up against one another and round LEDs may need a bit of filing to get them to fit. Alternatively, rectangular LEDs can be used. The value of the series resistor depends on the LED colour: 47 Ω is suitable for red LEDs, 27 Ω for green, 22 Ω for white and 10 Ω for blue. Since the blue LEDs in particular are being driven at 3 V, below their rated voltage, it is recommended to select devices manually for uniformity of brightness. It is also a good idea to do this for green and white LEDs, but it is less critical for red LEDs. The tilt switch is mounted horizontally, soldered to the pins of JP2 (see **Figure 3**). The ISP connector on the board (JP1) need not be fitted, and the relevant

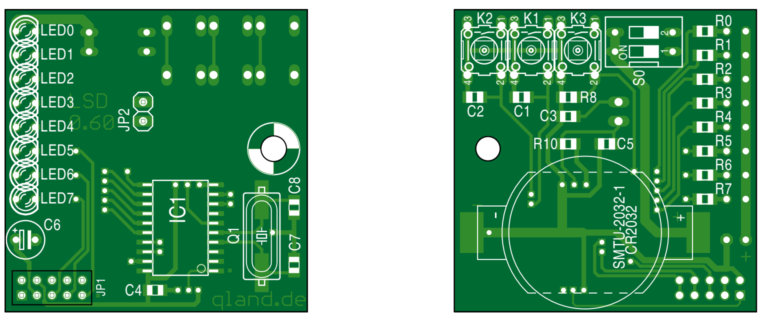


Figure 2. The double-sided printed circuit board mainly uses SMDs that can be soldered by hand.

Users’ guide

Switching on
- Set S0 to the ON position.

Displaying a message
- Hold the board so that LED 0 is at the top. Then quickly but regularly shake the unit to and fro.

Selecting a message *(single LED flashes)*
- Hold the board still, and LED 7 will continuously emit single flashes.
- **Hold down button K2** until LED 7 flickers, and then release the button. LED 6 will then start to emit single flashes continuously, and message 2 is ready for display.
- **Hold down button K2** until LED 6 flickers, and then release the button. LED 5 will then start to emit single flashes continuously, and message 3 is ready for display.
- **Hold down button K2** until LED 5 flickers, and then release the button. LED 4 will then start to emit single flashes continuously, and message 4 is ready for display.

Four messages are available. Message 1 is hard-programmed in, while messages 2 to 4 can be changed at will by the user. If a message is not programmed the letter ‘E’ (for ‘empty’) will be displayed.

Programming a message *(double LED flashes)*
- Select message 2, 3 or 4 according to the instructions above.
- **Hold down K1 and K2 simultaneously** until the LED flickers and then release the buttons. LED7 will now continuously emit double flashes which means that the unit is ready to be programmed with a capital letter.
- Shake the board, and the currently selected letter (‘A’ initially) will be displayed.
- **Repeatedly press button K1 or K2 briefly**. This changes the selected letter forwards or backwards through the sequence ‘A’, ‘B’, ‘C’, and so on to ‘Z’. To program the selected letter, **hold down K1** until the LED flickers. To switch between programming modes, **hold down K2** until the LED flickers. The programming modes are indicated as follows:

- LED7: programming mode for the capital letters ‘A’ to ‘Z’ and the space character (the space character is displayed as a rectangle during programming);
- LED6: programming mode for the lowercase letters ‘a’ to ‘z’ and the space character;
- LED5: programming mode for the digits ‘0’ to ‘9’ and the space character;
- LED4: programming mode for symbols, accented characters, punctuation and the space character;
- LED3: erase functions B (‘backspace’, deletes the most recently entered character) and C (‘clear’, deletes the entire message). To execute the erase function, **hold down K1** until LED3 flickers.

Leaving programming mode
- **Hold down buttons K1 and K2 simultaneously** until the LED flashes, and then release both buttons. The most recently selected (or edited) message will be displayed.

connections are made by tracks on the circuit board. If the ATtiny2313 is only to be programmed after it is soldered to the board, the four tracks joining the header pads need to be cut and a 2mm pitch header soldered on. After programming the broken connections must be restored: this can be done by plugging a suitably-wired socket to the programming connector.
The resistors are all 0805 SMD types, as are all the capacitors with the exception of C6 (a 47 µF 6 V electrolytic, radial, 2.5 mm pitch). Soldering the microcontroller in its SOIC package requires a fine bit and a little confidence in SMD work. Pin 1 of the microcontroller is indicated by a small dot on the printed circuit board. With a little skill with the soldering iron the SMD crystal can be replaced by a wired device or by a ceramic resonator. If a resonator is used capacitors C7 and C8 should be removed, and the middle contact of the resonator should be connected to the ground via near C7 and C8 (see Figure 3). It is a good idea to fit the microcontroller before the crystal as otherwise the room left for soldering is a bit tight.

Programming and operation

The text box gives an overview of how to operate the device. Of the four stored messages, messages 2, 3 and 4 can be edited using the buttons; message # 1, however, is hard-programmed into the device and cannot normally be changed or erased. There is, however, a special way to change this message: in quick succession press K2, K2, K2, K1, K1 and then K2. LED7 will then flicker briefly to confirm that the special mode has been entered, and message 1 can now be edited. To protect the message once more, use the same magic button sequence. It is also possible, of course, to enter a message directly into the EEPROM of the ATtiny microcontroller using a programmer.

(090337)

[1] http://www.elektor.com/090337 (includes software download)

About the author
Kurt Schuster is a self-employed electronics engineer and software developer.

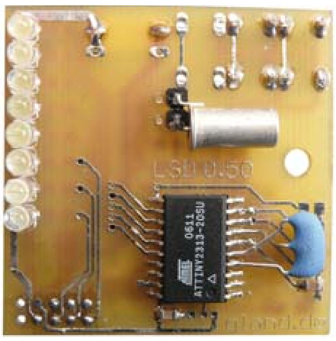


Figure 3. The top side of the printed circuit board, with horizontally-mounted tilt switch and 4 MHz ceramic resonator.

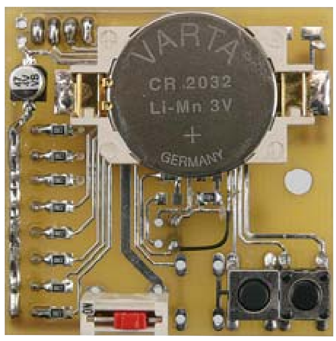


Figure 4. The underside of the printed circuit board, with the button cell, smoothing capacitor, on-off switch and the two buttons that are used to operate the device and enter messages.

How to Get Your Own USB ID

Options and costs

By Harry Baggen (Elektor Netherlands Editorial)

Every device with a USB interface needs to have a set of ID numbers that enable it to register with a host (computer or other equipment) so the host can take the appropriate action. Is it also necessary to have these ID numbers for devices you develop yourself, and if so, how can you get your own ID numbers for your products? Here we report on the results of a brief survey.

Nowadays you find products with USB interfaces just about everywhere, ranging from practical devices such as external USB hard disks to frivolous gadgets such as USB coffee mug heaters. Every USB device that you connect to a computer (known in USB terminology as a host) uses two ID codes to register with the computer: a vendor ID (VID) and a product ID (PID), each of which is a 16-bit number (for example, 0x0424 and 0x0531). From this set of ID codes, the operating system of the PC determines what sort of device is connected, what designation should be assigned to it, and what driver should be used for it.

These numbers are administered by the USB Implementers Forum (USB IF [1]), an organisation that was founded by various computer companies and ensures that manufacturers of USB devices comply with the formulated USB standards.

Buying your own VID

If you develop a device with a USB port and you want to market it commercially, you can request your own VID from the USB IF. There are several options, although they are actually oriented toward mass production and not intended for devices such as prototypes. Briefly, the options are:

- Become a member of the USB IF (membership fee: US$ 4000 per year). You will be assigned a VID at no additional cost.
- Purchase a USB logo licence (fee: US$ 2000). This licence is good for two years.
- Purchase a VID alone (fee: US$ 2000). With this option, you are not allowed to show the USB logo on your products.

Once you have been assigned a VID, you receive a large block of PID numbers that you may assign as you see fit. The number of PIDs is large enough (around 65,000) that manufacturers of USB products don't have to worry about using them up too quickly. Only one number is necessary for each product type or model, so each individual product does not need to be assigned a separate number.

Prototypes and small-scale production

What can you do if you want to use your own VID and PID for a single prototype or small volume of products? You probably don't wish to spend a large amount of money for this. In the past, there were a few companies that bought their own VIDs and then sold small blocks of PIDs to people who wanted to use them for their prototypes and their own products, but the USB IF disapproved of this arrangement. Some time ago it prohibited this form of trading in USB numbers and added a corresponding clause to the regulations. However, there is one company that still does this (MCS Elec; see [2]), based on the stance that the rules were amended after it bought its VID.

If you use Atmel microcontrollers, there is also another option. If you use the V-USB driver [3], you receive a VID/PID set free of charge if you agree to adhere to the conditions of the GNU general public licence (GPL) governing the V-USB project. If you do not wish to release the software you develop, as required by the terms of the GPL, you can also purchase one or more VID/PID sets. For hobby use, each set costs about £/€ 10.

What about the situation where you develop a device with a USB interface IC in its circuit? Usually the IC manufacturer has its own VID and assigns a separate PID to each individual product with a USB interface. You can then use these products to develop prototypes of devices or circuits.

The next question is: what if you wish to make products on a modest scale? We found two manufacturers who are willing to do a bit more for their customers in this regard. Microchip has a large number of PICs with integrated USB interfaces in its product line. On the Microchip website you can find a document [4] that you can use to apply for a sublicence. With this sublicence, you receive the Microchip VID and your own PID, which you can use for your product. This

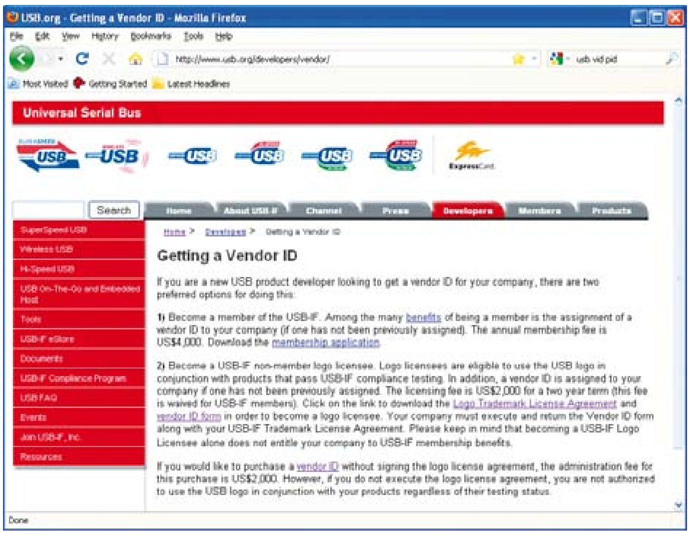
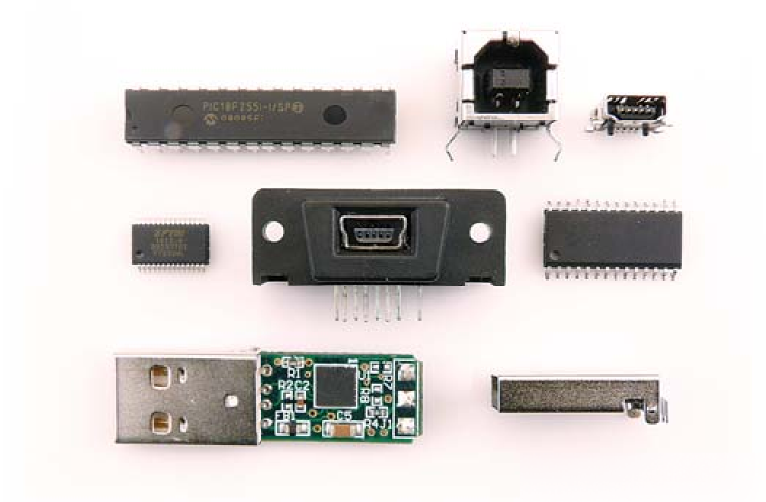


Figure 1. The USB Implementers Forum administers all vendor IDs.



is handy if you want to manufacture products on a small scale. In this regard, the only condition imposed by Microchip is that the production volume must not exceed 10,000 units. As far as we know, there are no other costs associated with this, although Microchip naturally makes money on the ICs you buy from them.

FTDI is a semiconductor manufacturer that has become very well known for its USB interface ICs, which make it very easy to provide a USB connection for a device with a serial interface. FTDI also supplies RS232 connectors with integrated USB ICs, and of course various types of adapter cables, such as USB to TTL). On FTDI's website you can find a PDF document [5] that clearly explains which VIDs and PIDs are used by FTDI and what options are available to users of FTDI ICs. Here again, it is possible to apply for a unique block of product IDs for use in prototypes and small-scale production, with no specific quantities mentioned.

With other manufacturers that also have a lot of ICs with USB interfaces in their product lines, such as Atmel, Analog Devices, Freescale and TI, we were unfortunately not able to find any option for applying for your own block of PIDs. With these products, the VID/PID pairs programmed into the ICs can only be used for circuit development. In most of the documentation, it is simply noted that you need a separate VID/PID set for each product you develop, with a recommendation to contact the USB IF or visit their website. Perhaps these companies could take a cue from Microchip and FTDI to make things a bit easier for users who manufacture products on a modest scale (as long as the USB IF doesn't decide to prohibit this sort of arrangement).

If you are curious about the VIDs assigned to all sorts of manufacturers, you can visit the website at [6] to view a list.

(100718-I)

Internet Links

[1] www.usb.org/home

[2] www.mcselec.com (look in the shop under Hardware/USB)

[3] www.obdev.at/products/vusb/license.html

[4] ww1.microchip.com/downloads/en/AppNotes/Application%20for%20USB%20Vendor%20ID%20Sublicense.pdf

[5] www.ftdichip.com/Support/Documents/TechnicalNotes/TN_100_USB_VID-PID_Guidelines.pdf

[6] www.linux-usb.org/usb.ids

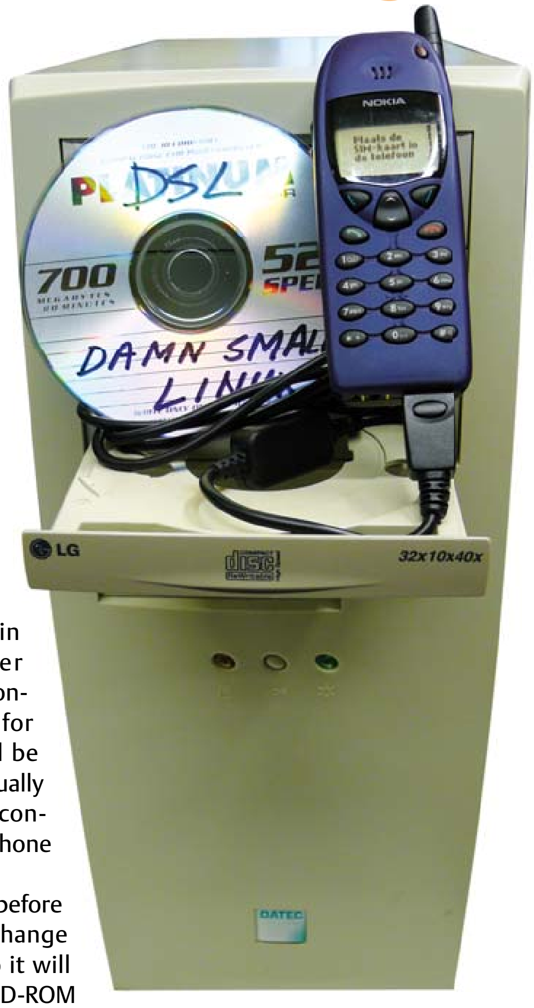
Index of Advertisers

TEXT Me! from 1, PC Junkyard

An £0.00 SMS gateway centre using Linux and a recycled PC

By Hans Henrik Skovgaard (Denmark)

In this small project a discarded PC together with an old mobile phone will both be granted a second lease of life. With the 'Damn-Small-Linux' (DSL) variant running on the PC and the mobile phone attached, the basics of a small, totally free SMS gateway will be demonstrated. Fasten your seatbelts.



If like me you refuse to throw away electronic assemblies that are functional but 'less fashionable', you will most likely too have managed to accumulate several old PCs and maybe a few old mobile phones. Of course, you will have hidden your clunkers in an artful manner from viewing by a house-proud partner. A veritable 2011 Aladdin's cave.

After several years, some of my own such treasures were very close to reaching their final destination at the 'recycling place' (aka 'Old-Silicon Heaven'). That was until I saw the "Remote control by Mobile Phone" article in Elektor's November 2008 issue, where a mobile phone got attached to a dedicated piece of hardware. However since I'm currently into Damn Small Linux coding I wanted to connect a phone to a PC, run Linux and make something useful out of it all. It is now up to you to decide if this is clever, crafty or crazy, but here I got DSL up and running, attached the mobile phone to the PC, installed software and built the basics of my very own small SMS gateway. Here's how you can do it, too.

Installing DSL

People already into Linux, in particular Damn Small Linux (DSL), will possibly scoff at the level of detail in this article, but the aim is to enable you to set up such a system all by yourself even if you're new to Linux.

In **Table 1** you can see the hardware configuration used for this project. The PC was originally a Dell Dimension XPS T600r but in good PC Junkyard fashion the only bits of it left by now are the motherboard and the power supply. No preferences are expressed here — check out what you have lying around, dig it out and see if it works for you.

There is one thing though that needs to be in place. Your old PC must have either a USB or an RS232 connector. You need to align this with the method of connecting the mobile phone to the PC. Nowadays USB is the port of choice but not so long ago it was RS232 — like in the November 2008 article. Configuring Linux for both cases will be described —actually three ways to connect a mobile phone to the Linux PC.

One final thing before we continue: change the PC BIOS so it will boot from the CD-ROM drive.

Damn Small Linux is a stripped down version of another Linux distribution called Knoppix. It's a free Linux distribution for the x86 family of personal computers and fits inside a 50 MB live CD.

One of the reasons why DSL is so small is that instead of using KDE or GNOME as its desktop it employs one of the two 'lightweight' desktops that go by the names of 'Fluxbox' and 'JWM'. This makes DSL an ideal choice to run on junked hardware and thereby bring back new life to it. You may choose to use another Linux distribution where there will most likely be differences but nothing that can't be managed. A link to the DSL home page can be found at [1].

At the time of writing there are two maintained versions of DSL: an older version called DSL-3.4.12 which uses Linux kernel version 2.4.26 and a newer one called DSL-4.4.10 which uses Linux kernel version 2.4.31. Sure, there exists a newer Linux kernel but this is how DSL is configured in order to keep the system small and fast.

For the purpose of this article, version DSL-3.4.12 was downloaded from [2], the file you want is identified as "dsl-3.4.12.iso". The reason for choosing the DSL-3.4.12 version is mainly due to the author's 'C' development environment used for the control centre described in his book [3]. Once downloaded, use your favourite CD burning soft-

ware to make a bootable CD containing DSL. Note that you have downloaded an ISO image file, this is essential to know when you burn the CD-ROM for later use.
The CD-ROM you just made is a Live CD allowing you to boot up a fully functional Linux system running completely in RAM.
As you probably want to install a permanent system, the next thing to do – after the first boot up — is to install DSL on your old PC's hard disk. Now that's straightforward and can be found under the menu item:

```
APPS->Tools->Install to Hard Drive.
```

Note that the menu is found by right clicking anywhere on the desktop. Please refer to the description at the end of this article if you need help installing DSL.

Connect the mobile phone

Like in the November 2008 article a Siemens mobile phone is being used, in this case a C65 mainly because the author contributed to developing it. Not surprisingly the AT command set [4] interface and the phone proper are 'old stomping ground'. Don't feel tied to using the C65 phone in this project though, and as you can see when we get to the SMS gateway software, one of the most commonly used mobiles are [... drumroll ...] Nokia phones.
In **Figure 1** you can see what the interface cables to the C65 look like. To the right in the picture is the RS232 connector and in the centre, the actual phone connector. To the left there is a small power adaptor in order to keep the phone running so you never need to turn it off and recharge it.
If you attach the phone to the RS232 interface right away you will then be able to communicate with the phone via the device interface called /dev/ttyS0.
If you have done some C programming (if not there's a good Elektor book on this [7]) you will know how to read from and write to a file. It is exactly the same thing here. In order to read or write to the phone you read or write to /dev/ttyS0.
In order to verify that the phone is connected and working there is a small program called 'microcom' you can use, see **Figure 2**.
In case your PC has a USB connector and no RS232 interface you can use an USB-RS232 converter. If you do that you will get a different device interface this time called **`/dev/ttyUSB0`**. In order to use 'microcom' to test your connection this time you need to tell 'microcom' which interface to use. This is done as shown in **Figure 3**.
If your phone and PC both have USB connectivity, just connect them together and your device interface will then be /dev/ACM0. You verify the connection the same way as with the USB-RS232 converter.
When connecting the phone via the USB interface you need to set up your phone in 'modem' mode or similar. In many cases a mobile phone has a USB mode setting where you can choose between 'mass storage', 'modem' and maybe some more modes. You may have to do some testing to see what's working. All in all it gives you possibilities listed in **Table 2**.
You will be able to install support for both IrDA and Bluetooth as

Table 1. Junkyard PC hardware configuration (example)

CPU	Pentium III 600MHz
RAM:	384 Mbytes
BIOS:	PhoenixBIOS 4.0 release 6.0
Hard disk:	10 GByte (Samsung)
Network card:	Realtek 100Mbit card
Display adaptor:	
CD-ROM drive:	NEC DVD drive

Figure 1. Siemens C65 mobile phone interface cables.

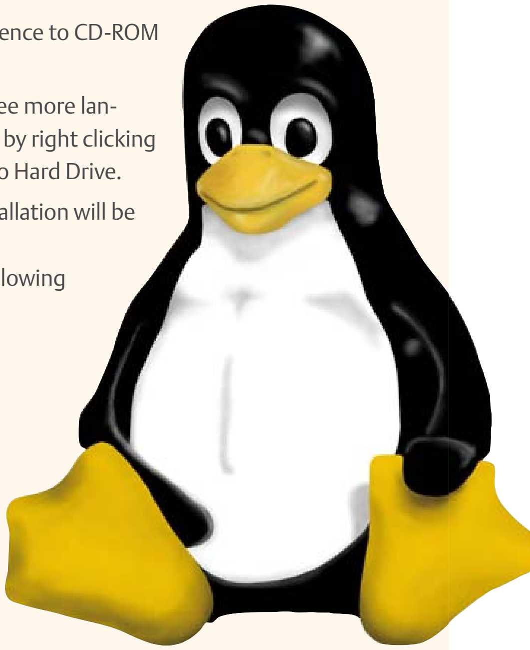
Figure 2. microcom w. /dev/ttyS0.

Table 2. Possible device interfaces.

Interface	Device
Std RS232	/dev/ttyS0
USB - RS232	/dev/ttyUSB0
USB	/dev/ttyACM0

DSL step by step installation

Listen up PC users, DSL here means *Damn Small Linux*, not *Digital Subscriber Line*.



1. Insert the Live CD in you CD-ROM drive and power up the PC. Remember to set the boot sequence to CD-ROM first.
2. Set the language like “boot: dsl lang=xx” (where xx = language code) then hit Enter. You can see more language definitions by pressing F2. This should bring up you DSL desktop. DSL menus are found by right clicking anywhere outside a program on the desktop. Find the following menu: APPS->Tools->Install to Hard Drive.
3. You are now ready to install DSL on your hard disk. Please note all data present before the installation will be lost after the installation.
   During the installation you will be asked some questions which most likely should have the following answers:

   Enter the target partition: hda2 (or hda1)

   Do you wish to support multiuser login: n

   Use journalized ext3 filesystem: n

   Continue: y

   (This was your last warning to save the content on the PC)

   Proceed to install a boot loader: y

   Use [G]rub MBR or [L]ilo Active Partion: g

   (If you selected hda2 above the following will appear)

   Do you have windows installed: no

   Reboot: yes
4. You should now remove the CD and see your system reboot for the first time. The first picture will be the Grub boot loader startup picture. After that you will be asked to assign your system a root and a user dsl password. After that you should then get your DSL desktop up.

well but that’s maybe another article.
You should hopefully now have a running Linux system and a working mobile phone attached. High time to install some software.

Installing gnokii

The SMS gateway software we’re going to use is the software from gnokii [5]. In the following a lot of file manipulation will take place. You can do this either via a terminal window using Linux commands or use the built-in file manager called Emelfm. This file manager can be accessed by the Emelfm icon on the desktop.
If you are a hardcore Linux user you will now download the code and compile it yourself. You can still do that but fortunately you will also be able to find prefab software packages made by the DSL community.
The ready to run software packages can be found via the MyDSL (Extension tool) icon on the desktop. Once started you will be able to see lots of precompiled (almost) ready-to-install applications. The installations we are interested in here can be found under the ‘Testing’ tab!
Please remember that you need to have Internet access in order to use the MyDSL tool. If not, you will be unable to see the list of precompiled applications and therefore unable to download them.
Under ‘testing’ you will find the gnokii-0.6.25.uci package. The UCI extension indicates that it is a Universal Compressed ISO image. Extensions with the .uci format are mounted as a separate file system to minimize RAM usage. On mounting, see [6].
You download the software package by selecting it and after having read the instructions hit ‘download’. You should save the software in a directory where you can find it later as you need to include the location in a boot file. The default download location is /tmp.
As indicated in the DSL description for gnokii you need to download the following software packages as well:
- gtk+-2.12.9.uci
- bluez-utils.uci.

Like the gnokii software these are also located under ‘testing’ in the MyDSL Extension tool and should be saved in the same directory as the gnokii software. Once downloaded the software packages will be installed in the /opt directory — or mounted as they will not be present there following the next reboot.
In order to have the newly installed software available after a reboot you need to add the following

```
-mydsl-load /tmp/gnokii-0.6.25.uci
-mydsl-load /tmp/gtk+-2.12.9.uci
-mydsl-load /tmp/ bluez-utils.uci
```

at the end of the file /opt/bootlocal.sh. If hardcore you can use the ‘Beaver’ or ‘VI’-editor to edit /opt/bootlocal.sh. You can verify that your software packages are fully mounted by right clicking the MyDSL icon and selecting ‘UCI tool’. This displays which UCI packages are loaded.
Next thing to do is to set-up gnokii. This is done by copying the file:
`/opt/gnokii-0.6.25/gnokiirc`
to the home directory for the user dsl — which is /home/dsl — and rename the file to “.gnokiirc” (<dot>gnokiirc). In case you want to run the software as root you need to copy the file to the root-home

directory – which is /root.
In the config file you need to specify the correct port. You can see which port you should select in Table 2. It's recommended to specify:

```
port=/dev/ttyUSB0
```

when a USB-RS232 converter is used.
You also need to specify which model you will be using. Here you need to read the documentation and/or consult the gnokii homepage.
I specified

```
model=AT
```

as I wanted to use the AT-command mode, which is fully supported by the C65 phone.
By default it should not be necessary to change more in the config file. The rest of the parameters are explained if you want to experiment with them.
To verify that your system is working you should now run the following command:

```
gnokii --identify
```

in a terminal window. Remember to have your mobile phone switched on.

You will hopefully see lots of AT commands flying across the screen, ending with a listing of your phone's IMEI number, Manufacturer, Model and Product name.
If it is not working you're in for a debug session. Some hints to a conclusion can be found in the file:

```
/var/log/messages
```

You may also want to increase the gnokii debug information in the gnokii config file mentioned earlier.
After installation you will have noticed two additional icons (Gnocky and Xgnokii). They are the entry points to a GUI interface. In order to use them you need to do some initialisation of the new GTK library before you reboot your PC for the first time after the download and installation of the software.
The initialisation is done via the MyDSL menu, which has been extended by two new entries:

```
gnokii-0-6-25
GTK+-2.12.9-setup
```

The actual initialisation of the new GTK library is achieved via the menu item:

```
MyDSL->GTK+-2.12.9-setup->GTK+-2.12.9-setup
```

And then just follow the instructions.
After the initialisation of the GTK library you can use the two GUI interfaces (Gnocky and Xgnokii) either via the icons on the desktop or via the MyDSL menu.
One final tip before we continue using the actual software: in case you want to keep the new menu items in the MyDSL menu listing, you should save a copy of the file

```
/home/dsl/.fluxbox/mydsl.menu
```

and replace the myDSL.menu file that will be created after a reboot of the PC.

Using gnokii

If everything you've done so far has been successful you should now be ready to send your first Text (SMS) message. Before continuing, be cautioned that **"depending on your mobile phone subscription, sending excessive numbers of Text messages may cost you a fortune if you are not careful. No monies returned."** Having said that, let's continue.

Sending Text

In order to text somebody, start a terminal window so you can enter commands. If you just enter the following:

```
gnokii
```

you will see all the arguments that can be used with gnokii. The ones we are interested in is the `–sendsms` argument. So, to send a Text message (SMS) you need to enter the following:

```
echo "enter text here" | gnokii --sendsms
+4412345678
```

Where +4412345678 is the phone number with the country code included (+44 for United Kingdom). Note that there is no space between '—'and 'sendsms'.

In the terminal window you should hopefully once again see many AT commands flying across the screen ending with the following text before the command prompt returns:

```
Message sent (reference: 2)
Send succeeded!
Serial device: closing device.
```

The number after `reference:` may differ.

Receiving Text

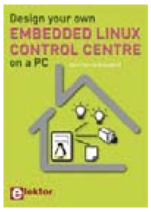
In order to receive Text (SMS messages) we need to make use of one of the other arguments to gnokii:

```
--smsreader
```

So to receive Texts you should enter the following command in a terminal window:

```
gnokii --smsreader
```

Figure 4. If you like working with old computers and Linux, this book comes highly recommended.



This makes gnokii look continuously for incoming Text msgs and save them into a mailbox under
`/tmp/sms/*` (actual filename varies).
Such a file could look like this:

```
/tmp/sms/sms_4512345678_1189_0
```

and will contain the content as you would see it on your mobile phone. There will be no additional information present.

If everything works as expected, incoming Texts are never actually saved in the phone.
You exit the gnokii smsreader mode by pressing <ctrl>–<c>.
Please note that you will not be able to receive and send Text messages at the same time. This is due to the way Linux works. When you start the gnokii program it will lock the device specified with the port command in the gnokii config file, thereby preventing other programs from using the port.
What I have shown so far is the basic stuff that must be present in order to set up a small rudimentary TEXT Gateway. In a further article I will show how to install an Apache server and make the received Texts available to 'the public' and present a more streamlined interface to send Texts around — overcoming the need to use the command line interface.
If you're interested in the full details of how to design your own Embedded Linux Control Centre on a PC, check out the Elektor book with that title [3] (**Figure 4**).

(090939)

Internet Links

[1] DSL homepage: http://damnsmalllinux.org/

[2] DSL download link: http://distro.ibiblio.org/pub/linux/distributions/damnsmall/current/dsl-3.x/

[3] Design your own Embedded Linux Control Centre on a PC: www.elektor.com/products/books/computer/embedded-linux.529463.lynkx

[4] AT command set decription: http://en.wikipedia.org/wiki/AT_commands

[5] gnokii homepage : http://www.gnokii.org/

[6] Two descriptions of mounting file systems: http://en.wikipedia.org/wiki/Mount_%28computing%29 http://en.wikipedia.org/wiki/Mount_%28Unix%29

[7] C Programming for Embedded Microcontrollers: www.elektor.com/products/books/microcontrollers/c-programming-for-embedded-microcontrollers.868705.lynkx

Hexadoku

Puzzle with an electronics touch

Is your hexadecimal calculus a bit rusty? Not to worry, for this 'electronified' puzzle the requirements are limited to counting from 0 to F, persistence and some logic reasoning to arrive at the solution. Enter the right numbers in the puzzle, send the ones in the grey boxes to us and you automatically enter the prize draw for four Elektor Shop vouchers. Have fun!

The instructions for this puzzle are straightforward. Fully geared to electronics fans and programmers, the Hexadoku puzzle employs 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 draw for a main prize and three lesser prizes. All you need to do is send us the numbers in the grey boxes.

Solve Hexadoku and win!

Correct solutions received from the entire Elektor readership automatically enter a prize draw for one Elektor Shop voucher worth £ 80.00 and three Elektor Shop Vouchers worth £ 40.00 each, which should encourage all Elektor readers to participate.

Participate!

Before March 1, 2011, send your solution (the numbers in the grey boxes) by email, fax or post to

Elektor Hexadoku – 1000, Great West Road – Brentford TW8 9HH

United Kingdom.

Fax (+44) 208 2614447 Email: hexadoku@elektor.com

Prize winners

The solution of the December 2010 Hexadoku is: **381F0**.
The £80.00 voucher has been awarded to: B. Horn (Germany).
The £40.00 vouchers have been awarded to: Karin Menzel (Germany), Serge Sussel (France), Christian Klems (The Netherlands).
Congratulations everyone!

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

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

Slide Rules & the Electronic Engineer

By Reginald W. Neale (USA)

Engineers translate ideas into products. They're limited by the tools available for facilitating that translation. Over the span of my engineering career, those tools have undergone a dramatic transformation. I hope you will agree that it's interesting to review on this month's Retronics pages some of the rich history of those changes. First-hand knowledge of much of this material is fast disappearing. Fortunately, the web has some remarkable archives that will help to preserve it.

Nothing new, just faster!

Let's take a brief look at the technology that produced radios, rockets, bridges and automobiles sixty years ago, when many of today's tools were yet to be invented. Consider that your computer, calculator and software design package don't do anything fundamentally new, they just make the old tasks easier and orders of magnitude faster. To one of today's newly minted engineers, the pace of yesterday's product life cycles would appear glacial; the design time and effort per product would seem exhausting. And there were once entire disciplines, like high-performance filter design, whose commercial potential was unrealized because the calculations could be so tedious.

Obviously, the tool revolution has been powered by continuous advances in semiconductor technology. Transistors, integrated circuits and microcontrollers have relentlessly driven increases in the computing power available to electronic engineers. Moore's Law predicts that semiconductor capability roughly doubles every two years. The cumulative effect of this exponential increase has been profound. T.J. Rodgers of Cypress Semiconductor notes that if automotive technology had followed the same trajectory as computer technology, today you would be able to buy a Chevy for a penny and it would get 10,000 miles per gallon. Sounds like hyperbole? In 1948, ENIAC cost $500 k and performed 0.002 MIPS. The current Digi-Key catalogue lists embedded controllers for $0.50 that perform 2 MIPS. Apply those same factors to the 1948 Chevy at $2 k and 15 mpg and Rodgers' claim doesn't look so preposterous.

COMMON LOGARITHMS OF NUMBERS

550 — 600

N.	0	1	2	3	4	5	6	7	8	9
550	74 036	044	052	060	068	076	084	092	099	107
551	115	123	131	139	147	155	162	170	178	186
552	194	202	210	218	225	233	241	249	257	265
553	273	280	288	296	304	312	320	327	335	343
554	351	359	367	374	382	390	398	406	414	421
555	429	437	445	453	461	468	476	484	492	500
556	507	515	523	531	539	547	554	562	570	578
557	586	593	601	609	617	624	632	640	648	656
558	663	671	679	687	695	702	710	718	726	733
559	741	749	757	764	772	780	788	796	803	811
560	819	827	834	842	850	858	865	873	881	889
561	896	904	912	920	927	935	943	950	958	966
562	974	981	989	997	*005	*012	*020	*028	*035	*043
563	75 051	059	066	074	082	089	097	105	113	120
564	128	136	143	151	159	166	174	182	189	197
565	205	213	220	228	236	243	251	259	266	274
566	282	289	297	305	312	320	328	335	343	351
567	358	366	374	381	389	397	404	412	420	427
568	435	442	450	458	465	473	481	488	496	504
569	511	519	526	534	542	549	557	565	572	580
570	587	595	603	610	618	626	633	641	648	656
571	664	671	679	686	694	702	709	717	724	732
572	740	747	755	762	770	778	785	793	800	808
573	815	823	831	838	846	853	861	868	876	884
574	891	899	906	914	921	929	937	944	952	959
575	967	974	982	989	997	*005	*012	*020	*027	*035
576	76 042	050	057	065	072	080	087	095	103	110
577	118	125	133	140	148	155	163	170	178	185
578	193	200	208	215	223	230	238	245	253	260
579	268	275	283	290	298	305	313	320	328	335
580	343	350	358	365	373	380	388	395	403	410
581	418	425	433	440	448	455	462	470	477	485
582	492	500	507	515	522	530	537	545	552	559
583	567	574	582	589	597	604	612	619	626	634
584	641	649	656	664	671	678	686	693	701	708
585	716	723	730	738	745	753	760	768	775	782
586	790	797	805	812	819	827	834	842	849	856
587	864	871	879	886	893	901	908	916	923	930
588	938	945	953	960	967	975	982	989	997	*004
589	77 012	019	026	034	041	048	056	063	070	078
590	085	093	100	107	115	122	129	137	144	151
591	159	166	173	181	188	195	203	210	217	225
592	232	240	247	254	262	269	276	283	291	298
593	305	313	320	327	335	342	349	357	364	371
594	379	386	393	401	408	415	422	430	437	444
595	452	459	466	474	481	488	495	503	510	517
596	525	532	539	546	554	561	568	576	583	590
597	597	605	612	619	627	634	641	648	656	663
598	670	677	685	692	699	706	714	721	728	735
599	743	750	757	764	772	779	786	793	801	808
600	815	822	830	837	844	851	859	866	873	880
N.	0	1	2	3	4	5	6	7	8	9

Proportional parts

	8
1	0.8
2	1.6
3	2.4
4	3.2
5	4.0
6	4.8
7	5.6
8	6.4
9	7.2

	7
1	0.7
2	1.4
3	2.1
4	2.8
5	3.5
6	4.2
7	4.9
8	5.6
9	6.3

.74 036 — .77 880

1

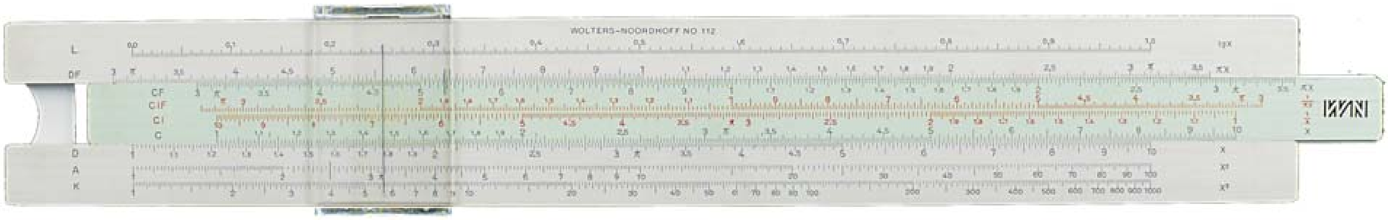
2

Reference tables and pencils

Compared to the design world we take for granted today, the environment of a mid-twentieth century designer was awkward and inefficient. Whatever technical information you couldn't keep in your head had to be retrieved from reference books. New advances in technology took months or even years to trickle down from laboratories to articles printed in trade magazines and scientific publications. There was also an overwhelming gender bias — engineers were almost exclusively male.

Here's an example of how painful it could be to complete even a simple calculation, back when you had to do it the hard way. Suppose you need to find an accurate value for the reactance of a 220 pF capacitor at 6.085 MHz. Picking up your paper and pencil, you write down $X_C = 1/(2\pi fC)$. You open up your book of logarithm tables. You look up the logarithm of each factor and you do the arithmetic, grinding out a result of 2.07514.

In **Figure 1**, the nearest value in the table is .07518. Using the Proportional Parts chart at the right, you determine that the antilog is



approximately 1189. Your calculation results in a characteristic or exponent of 2, so the solution is X_C = 118.9 ohms. Whew! And you have to hope you didn't make any dumb mistakes in your arithmetic or in using the tables. No wonder reactance nomograms and other charts were so popular in mid-century technical publications. Usually they could get you close enough for building a prototype circuit, and you could tweak the value from there.

Slide rulez

Unless the highest precision was needed, most mid-century engineers would have skipped the above hand calculation in favour of using a slide rule. Slide rules are nearly as old as logarithms. A slide rule is simply a mechanical analogue computer that adds and subtracts logarithms by representing them as distances. Much less accurate than log tables; but again, close enough for many purposes and far, far quicker. Slide rules don't have decimal points, so the user is forced to keep track of it mentally. That might sound like a nuisance, but it's a mental skill that engineers should still exercise today as a reality check.

Figure 2 shows a selection of slide rules: a miniature combination slide rule and caliper that fits into a pocket protector, and two standard 10-inch rules with user manuals. Circular rules were also available, along with elaborate helical rules boasting spiral scales equivalent to 10 feet in length! Accuracy depends on the skill of the user, the size of the rule, and how carefully the scales are

F TRIGONOMETRIC FUNCTIONS

(I) - sin(θ) and cos(θ)

If $0° \le \theta \le 45°$

and if $x = \theta/100$

then:

(6) $\sin(\theta) = 1.7453293x[1 - 0.50758x^2(1 - 0.149x^2)]$

(7) $\cos(\theta) = 1 - 1.523087x^2[1 - 0.25382x^2(1 - 0.1x^2)]$

If $45° \le \theta \le 90°$

use the identity

(8) $\sin(\theta) = \cos(90 - \theta)$

and use equation (7) above to calculate sin(θ).

Use the identity

(9) $\cos(\theta) = \sin(90 - \theta)$

and equation (6) above to calculate cos(θ).

STEP-BY-STEP PROCEDURE FOR CALCULATING SIN(θ) USING EQUATION (6):

STEP	OPERATION	STEP	OPERATION
1	enter x	6	multiply by x
2	square x	7	multiply by -0.50758
3	multiply by -0.149	8	add 1
4	add 1	9	multiply by x
5	multiply by x	10	multiply by 1.7453293

The maximum error in calculating sin(θ) or cos(θ) from equations (6) and (7) is approximately 0.0002% (two parts per million).

(II) - tan(θ)

If $0° \le \theta \le 30°$

and if $x = \theta/100$

then

(10) $\tan(\theta) = 1.7453293x[1 + 1.007796x^2(1 + 1.46465x^2)]$

3

4

aligned. A couple of significant figures for the standard-sized rule would be typical. What a contrast to the calculator accessory in my three year old PDA, which cranks out twelve significant figures as soon as you hit the ENTER key!

From Burroughs & Co. to HP and TI

The typical engineering student proudly carried his rule hanging from his belt, in a leather scabbard. If 2010 students have even heard of slide rules, they probably associate them with the era of powdered wigs and quill pens.

In the early '70s when simple four-function handheld calculators were flooding the market, scientific calculators were still priced out of reach for most engineers. The geek community soon developed clever algorithms to implement transcendental functions (**Figure 3**) on the cheap four-bangers from Burroughs and other manufacturers. But just as these shortcuts hit their stride, the first affordable scientific handheld calculators finally arrived: the Hewlett-Packard HP-35 and the Texas Instruments SR-50 (**Figure 4**). It would take almost another decade for real personal computers to become widely available, but the handhelds made logarithm tables and slide rules obsolete almost overnight.

(100802)

Retronics is a monthly column covering vintage electronics including legendary Elektor designs. Contributions, suggestions and requests are welcomed; please send an email to editor@elektor.com

ELEKTOR SHOWCASE

To book your showcase space contact Elektor International Media

Tel. 0031 (0) 46 4389444 Fax 0031 (0) 46 4370161

ASTROBE V3.0

www.astrobe.com

Windows Development System for LPC2000 microcontrollers.

- Develop high-and low-level software without C or assembler
- Generate fast 32-bit native ARM code
- No special programming hardware required
- Personal, Standard and Professional Editions

ATOMIC PROGRAMMING LTD

www.atomicprogramming.com

- AP-114 ISP/JTAG Programming System
- JTAG Programming and Testing
- Boundary Scan Testing
- Universal In-System Programming
- EEPROM and SPI Flash Out-of-Circuit Programming
- Generic GDB Proxy Server
- Jennic JN5148 ZigBee Development Applications
- Training Platform available

AVIT RESEARCH

www.avitresearch.co.uk

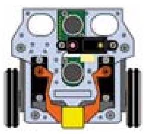
USB has never been so simple...
with our USB to Microcontroller Interface cable. Appears just like a serial port to both PC and Microcontroller, for really easy USB connection to your projects, or replacement of existing RS232 interfaces.

See our webpage for more details. From £10.00.

BLACK ROBOTICS

www.blackrobotics.com

Robot platforms and brains for research, hobby and education.



- Make your robot talk!
- TalkBotBrain is open-source
- Free robot speech software
- Robot humanisation technology
- Mandibot Gripper Robot

TO BOOK YOUR SHOWCASE SPACE CONTACT ELEKTOR INTERNATIONAL MEDIA

Tel. 0031 (0) 46 4389444
Fax 0031 (0) 46 4370161

CEDA

www.ceda.in
ceda@vsnl.com

CEDA e-learning
PADS | OrCAD | ALLEGRO
PCB layout@ $5 Hourly

- PCB Layout & library service @$5 Hourly
- PCB Reverse Engg, MCU Code/ckt. Extraction service
- Learn Signal Integrity with Hyperlynx, PCB Design with PADS & Allegro. Support by email & web meeting

DESIGNER SYSTEMS

http://www.designersystems.co.uk

Professional product development services.

- Marine (Security, Tracking, Monitoring & control)
- Automotive (AV, Tracking, Gadget, Monitoring & control)
- Industrial (Safety systems, Monitoring over Ethernet)
- Telecoms (PSTN handsets, GSM/GPRS)
- AudioVisual ((HD)DVD accessories & controllers)

Tel: +44 (0) 845 5192306

EASYSYNC

http://www.easysync.co.uk

EasySync Ltd sells a wide range of single and multi-port USB to RS232/RS422 and RS485 converters at competitive prices.

ELNEC

www.elnec.com

Europe's leading device programmers manufacturer:

- reliable HW: 3 years warranty for most programmers
- support over 58.000 devices
- free SW updates
- SW release: few times a week
- excellent technical support: Algorithms On Request, On Demand SW
- all products at stock / fast delivery

www.elektor.com

EMBEDDED ADVENTURES

www.embeddedadventures.com

From news and tutorials to modules, components and kits, we have everything for your next microcontroller based project.
Your embedded adventure starts here.

embedded adventures

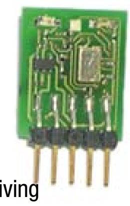
FIRST TECHNOLOGY TRANSFER LTD.

http://www.ftt.co.uk

First Technology Transfer Ltd.

- Training and Consulting for IT, Embedded and Real Time Systems
- Assembler, C, C++ (all levels)
- 8, 16 and 32 bit microcontrollers
- Microchip, ARM, Renesas, TI, Freescale
- CMX, uCOSII, FreeRTOS, Linux operating systems
- Ethernet, CAN, USB, TCP/IP, Zigbee, Bluetooth programming

FLEXIPANEL LTD



www.flexipanel.com

TEAclippers - the smallest PIC programmers in the world, from £20 each:

- Per-copy firmware sales
- Firmware programming & archiving
- In-the-field firmware updates
- Protection from design theft by subcontractors

FUTURE TECHNOLOGY DEVICES

http://www.ftdichip.com

FTDI designs and sells USB-UART and USB-FIFO interface i.c.'s.
Complete with PC drivers, these devices simplify the task of designing or upgrading peripherals to USB

HAMEG®
Instruments
A Rohde & Schwarz Company

- ☑ Oscilloscopes
- ☑ Power Supplies
- ☑ Spectrum Analyzers
- ☑ RF Instruments
- ☑ Programmable Measuring Instruments

Great Value in Test & Measurement

www.hameg.com

products and services directory

HEXWAX LTD

www.hexwax.com

World leaders in Driver-Free USB ICs:

- USB-UART/SPI/I2C bridges
- TEAleaf-USB authentication dongles
- expandIO-USB I/O USB expander
- USB-FileSys flash drive with SPI interface
- USB-DAQ data logging flash drive

MQP ELECTRONICS

www.mqp.com

- Low cost USB Bus Analysers
- High, Full or Low speed captures
- Graphical analysis and filtering
- Automatic speed detection
- Bus powered from high speed PC
- Capture buttons and feature connector
- Optional analysis classes

www.elektor.com

ROBOT ELECTRONICS

http://www.robot-electronics.co.uk

Advanced Sensors and Electronics for Robotics

- Ultrasonic Range Finders
- Compass modules
- Infra-Red Thermal sensors
- Motor Controllers
- Vision Systems
- Wireless Telemetry Links
- Embedded Controllers

ROBOTIQ

http://www.robotiq.co.uk

Build your own Robot!

Fun for the whole family!

Now, available in time for X-mas

- Arduino Starter Kits *NEW!!*
- Lego NXT Mindstorms
- Affordable Embedded Linux Boards
- Vex Robotics (kits and components)
- POB Robots (kits and components)

email: sales@robotiq.co.uk Tel: 020 8669 0769

STEORN SKDB LITE

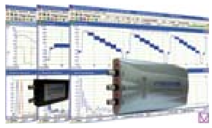
Join the SKDB Lite, the place to understand, discuss and experiment with magnetics.

- Learn more about magnetics and electromagnetics
- Participate in developer forums and discussion surrounding magnetics and related topics.

For FREE access to SKDB Lite:
https://kdb.steorn.com/ref25

VIRTINS TECHNOLOGY

www.virtins.com



PC and Pocket PC based virtual instrument such as sound card real time oscilloscope, spectrum analyzer, signal generator, multimeter, sound meter, distortion analyzer, LCR meter. Free to download and try.

www.elektor.com

SHOWCASE YOUR COMPANY HERE

Elektor Electronics has a feature to help customers promote their business, Showcase - a permanent feature of the magazine where you will be able to showcase your products and services.

- For just £242 + VAT (£22 per issue for eleven issues) Elektor will publish your company name, website address and a 30-word description
- For £363 + VAT for the year (£33 per issue for eleven issues) we will publish the above plus run a 3cm deep full colour image - e.g. a product shot, a screen shot from your site, a company logo - your choice

Places are limited and spaces will go on a strictly first come, first served basis. So-please fax back your order today!

I wish to promote my company, please book my space:
• Text insertion only for £242 + VAT • Text and photo for £363 + VAT

NAME: .... ORGANISATION: ....

JOB TITLE: ....

ADDRESS: ....

....

.... TEL: ....

PLEASE COMPLETE COUPON BELOW AND FAX BACK TO 00-44-(0)1932 564998

COMPANY NAME ....

WEB ADDRESS ....

30-WORD DESCRIPTION ....

....

....

....

....

Going Strong

A world of electronics from a single shop!

NEW!

Limited Period Offer for Subscribers!

£4 DISCOUNT

www.elektor.com/february

CD/DVD-ROMs

RFID, NFC, Zigbee, GPS and more

DVD Wireless Toolbox

On this DVD-ROM you'll find a number of technical documents and tools that will enable you to add wireless data exchange to your electronics systems. The choice of equipment depends on the transmission distance: a few centimetres using Near Field Communication (NFC) or Radio Frequency Identification (RFID), tens of metres with the Bluetooth, Wi-Fi or ZigBee systems, or indeed thousands of kilometres using a module for receiving GPS data. In accordance with the principle of our Toolbox series, we've brought together technical documentation (spec. sheets, application notes, user guides, etc.) on various devices according to the frequency and/or protocol used. All of the documents are PDF files (in English). Browsing around the DVD is made easy by an HTML menu.

ISBN 978-90-5381-268-6 • £28.50 • US $46.00

A must-have for audiophiles

DVD Masterclass High-End Valve Amplifiers

In this Masterclass Menno van der Veen will examine the predictability and perceptibility of the specifications of valve amplifiers. The DVD represents 3.5 hours of footage filmed in the grand meeting room of Elektor House. Bonus elements on the DVD include the complete PowerPoint presentation (74 slides), scanned overhead sheets (22 pcs), AES Publications mentioned during the Masterclass. Not forgetting the bombshell: 25 Elektor publications about valves.

ISBN 978-0-905705-86-6

£24.90 • US $40.20

110 issues, more than 2,100 articles

DVD Elektor 1990 through 1999

This DVD-ROM contains the full range of 1990-1999 volumes (all 110 issues) of Elektor Electronics magazine (PDF). The more than 2,100 separate articles have been classified chronologically by their dates of publication (month/year), but are also listed alphabetically by topic. A comprehensive index enables you to search the entire DVD.

ISBN 978-0-905705-76-7

£69.00 • US $100.00

75 Audio designs for home construction

DVD The Audio Collection 3

This DVD contains more than 75 different audio circuits from the volumes 2002-2008 of Elektor. The articles on the DVD-ROM cover Amplifiers, Digital Audio, Loudspeakers, PC Audio, Test & Measurement and Valves. Highlights include the ClariTy 2x300 W Class-T amplifier, High-End Power Amp, Digital VU Meter, Valve Sound Converter, paX Power Amplifier, MP3 preamp and much more. Using the included Adobe Reader you are able to browse the articles on your computer, as well as print texts, circuit diagrams and PCB layouts.

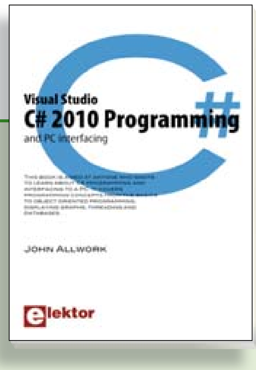
ISBN 978-90-5381-263-1
£17.90 • US $28.90

More than 75 power supply designs

CD The Power Supply Collection 1

This CD-ROM contains more than 75 different power supply circuits from the volumes 2001-2005 of Elektor. Highlights include the Cuk Converter, Automatic Battery Switchover, Battery Voltage LED, Digital Benchtop Power Supply, Lithium-Ion Charger, Electronic Fuse, High Voltage Regulator, Power Supply for USB Devices, Step-up Converter for White LEDs, Vehicle Adapter for Notebook PCs and much more. Using the included Adobe Reader you are able to browse the articles on your computer, as well as print texts, circuit diagrams and PCB layouts.

ISBN 978-90-5381-265-5
£17.90 • US $28.90



Visual Studio

C# 2010 Programming and PC interfacing

This book is aimed at anyone who wants to learn about C# programming and interfacing to a PC. It covers programming concepts from the basics to object oriented programming, displaying graphs, threading and databases. The book is complete with many full program examples, self assessment exercises and links to supporting videos. All code examples used are available – free of charge – from a special support website. Professional quality software tools are downloadable –also free of charge– from Microsoft. The Microsoft Visual Studio 2010 environment is extensively covered with user controls and their properties, methods and events. Detailed guidance is provided for those wishing to control hardware from a PC with PC interfacing chapters which explain the legacy serial and parallel ports, analogue interfacing using the sound card and use of Microsoft DirectX drivers. Interfacing to the ubiquitous USB port is explained in-depth with a detailed hardware and software design for a USB connected PIC-based hardware target included.

306 pages • ISBN 978-0-905705-95-8
£29.50 • US $47.60

More information on the Elektor Website:

www.elektor.com

Elektor
Regus Brentford
1000 Great West Road
Brentford
TW8 9HH
United Kingdom
Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
Email: sales@elektor.com

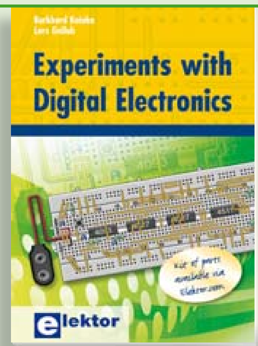
elektor

Associated starter kit available

ARM Microcontrollers

This is the perfect book for people who want to learn C and who want to use an mbed ARM microcontroller in an easy and fun way. The mbed NXP LPC1768 uses cloud technology, a revolutionary concept in software development. This means you do not need to install software on your PC in order to program the mbed! The only thing you need is a browser such as Microsoft Internet Explorer, and a USB port on your PC. No previous experience or knowledge required. You can get access to your project from any PC anywhere in the world and continue working on it. When you are done a few mouse clicks transfer the program to your mbed hardware.

250 pages • ISBN 978-0-905705-94-1
£29.50 • US $47.60



An introduction to digital control electronics

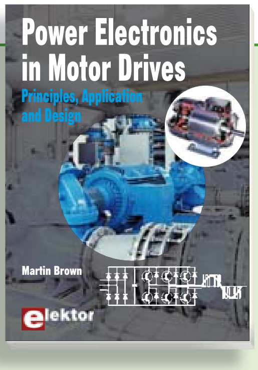
Experiments with Digital Electronics

This book presents fundamental circuits using gates, flip-flops and counters from the CMOS 4000 Series. Learning these fundamentals is best done using practical experiments. Each of the 50 experiments presented in this book has a circuit diagram as well as a detailed illustration of the circuit's construction on solderless breadboard. Building these digital circuits will improve your knowledge and will be fun to boot.

176 pages • ISBN 978-0-905705-97-2
£26.50 • US $42.80

Books

Books

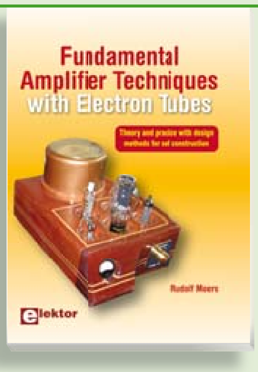


Principles, Application and Design

Power Electronics in Motor Drives

This book is aimed at people who want to understand how AC inverter drives work and how they are used in industry. The book is much more about the practical design and application of drives than about the mathematical principles behind them. The detailed electronics of DC and AC drive are explained, together with the theoretical background and the practical design issues such as cooling and protection.

240 pages • ISBN 978-0-905705-89-7
£29.50 • US $47.60



The ultimate tube amplifier reference book

Fundamental Amplifier Techniques

The aim of this book is to give the reader useful knowledge about electron tube technology in the application of audio amplifiers, including their power supplies, for the design and DIY construction of these electron tube amplifiers. This is more than just building an electron tube amplifier from a schematic made from the design from someone else. No modern simulations, but because you first understand the circuit calculations, then you can work with your hands to build the circuit .

834 pages • ISBN 978-0-905705-93-4
£65.00 • US $104.90

NetWorker

(December 2010)

An Internet connection would be a valuable addition to many projects, but often designers are put off by the complexities involved. The 'NetWorker', which consists of a small printed circuit board, a free software library and a ready-to-use microcontroller-based web server, solves these problems and allows beginners to add Internet connectivity to their projects. More experienced users will benefit from features such as SPI communications, power over Ethernet (PoE) and more.

Module, ready assembled and tested

Art.# 100552-91 • £53.00 • US $85.50

Digital Multi-Effects Unit

(September 2010)

It's a simple fact: every recording sounds better with the right sound effects. Here we prove that it's possible to generate a variety of effects digitally, including hall, chorus and flanger effects, without having to work yourself to the bone with DSP programming. The circuit is built around a highly integrated effects chip and features an intelligent user interface with an LCD. The result is a treat for the eye and the ear.

Kit of parts including PCBs, programmed controllers and EEPROM

Art.# 090835-71 • £165.00 • US $266.20

The Elektor DSP radio

(July/August 2010)

Many radio amateurs in practice use two receivers, one portable and the other a fixed receiver with a PC control facility. The Elektor DSP radio can operate in either capacity, with a USB interface giving the option of PC control. An additional feature of the USB interface is that it can be used as the source of power for the receiver, the audio output being connected to the PC's powered speakers. To allow portable 6 V battery operation the circuit also provides for an audio amplifier with one or two loudspeakers.

PCB, assembled and tested

Art.# 100126-91 • £149.00 • US $240.40

Reign with the Sceptre

(March 2010)

This open-source & open-hardware project aims to be more than just a little board with a big microcontroller and a few useful peripherals — it seeks to be a fast prototyping system. To justify this title, in addition to a very useful little board, we also need user-friendly development tools and libraries that allow fast implementation of the board's peripherals. Ambitious? Maybe, but nothing should deter you from becoming Master of Embedded Systems Universe with the help of the Elektor Sceptre.

PCB, populated and tested, test software loaded (excluding Bluetooth module)

Art.# 090559-91 • £89.00 • US $143.60

Kits & Modules

Product Shortlist

	£	US $
February 2011 (No. 410)		
+ + + Product Shortlist February: See www.elektor.com + + +		
January 2011 (No. 409)		
Nixie Tube Thermometer		
090784-1 ...... Printed circuit board	12.40	20.00
090784-41 .... Programmed controller AT89C2051/24PU	8.75	14.10
Flight Data Recorder		
071035-91 .... ATM18 controller module	7.30	15.40
090773-91 .... PCB, populated and tested with programmed bootloader	56.00	90.00
100653-1 ...... Printed circuit board	12.95	20.90
Low-cost Headphone Amp		
100500-71 .... Elektor Project Case	16.80	25.80
100701-1 ...... Printed circuit board	8.75	14.10
Wireless ECG		
080805-1 ...... Printed circuit board	8.75	14.10
Support Board for Arduino Nano		
100396-1 ...... Printed circuit board	18.00	29.00
December 2010 (No. 408)		
NetWorker		
100552-91 .... Module, ready assembled and tested	53.00	85.50
Heating System Monitor		
090328-41 .... ATmega328-20AU (TQFP32-08), programmed	11.00	17.80
Stroboscopic PC Fan		
100127-1 ...... Printed circuit board	4.50	7.30
100127-41 .... ATtiny 2313, programmed	14.20	8.75
ARM Freephone Control		
080632-91 .... ECRM40 module, ready assembled and tested	32.00	51.70
Modular LED Message Board		
100664-41 .... MC9S08SH32CWL, programmed	8.75	14.20
Speed Controller for Small DC Motors		
100571-41 .... ATtiny44-20PU, programmed	8.75	14.20
November 2010 (Nr. 407)		
Micro Fuel Cell Measures Oxygen Concentration		
090773-91 .... PCB, populated and tested with programmed bootloader	56.00	90.40
The 5532 OpAmplifier (2)		
100124-1 ...... Amplifier board (one channel)	23.00	37.10
100124-2 ...... Power supply board	17.95	29.00
Camera Interval Timer		
081184-41 .... PIC16F886-I/SP, SPDIP28, programmed	8.00	12.90
October 2010 (No. 406)		
CL-3 Digital Rotary Combination Lock		
100026-41 .... Atmel ATTINY2313-20PU, programmed	8.00	12.90
Wheelie GT		
100479-71 .... Kit of parts upgrade kit controller board + 2x Hall sensor board	105.00	169.40
September 2010 (No. 405)		
Elektor Project Case		
100500-71 .... Predrilled Lexan sheets with standoffs	14.90	24.10
Digital Multi-Effects Unit		
090835-31 .... EEPROM 24LC32	4.00	6.50
090835-41 .... ATmega8-16PU	8.30	13.40
090835-42 .... ATtiny2313-20PU	8.30	13.40
090835-71 .... Kit of parts including PCBs, programmed controllers and EEPROM	165.00	266.20
Dual Voltage/Current Display		
100166-71 .... Kit of parts incl. PCB, item -41, LCD	62.00	100.00
Vision System for Small Microcontrollers		
090334-1 ...... PCB 19.90 32.10		
090334-41 .... PIC16F690-I/P, programmed	8.00	12.90

Bestsellers

Books

1. **ARM Microcontrollers** — ISBN 978-0-905705-94-1 .... £29.50 .....US $47.60
2. **Experiments with Digital Electronics** — ISBN 978-0-905705-97-2 .... £26.50 .....US $42.80
3. **C# 2010 Programming and PC interfacing** — ISBN 978-0-905705-95-8 .... £29.50 .....US $47.60
4. **Fundamental Amplifier Techniques with Electron Tubes** — ISBN 978-0-905705-93-4 .... £65.00 ...US $104.90
5. **Power Electronics in Motor Drives** — ISBN 978-0-905705-89-7 .... £29.50 .....US $47.60

CD/DVD-ROMs

1. **CD The Power Supply Collection 1** — ISBN 978-90-5381-265-5 .... £17.90 .....US $28.90
2. **DVD The Audio Collection 3** — ISBN 978-90-5381-263-1 .... £17.90 .....US $28.90
3. **DVD Elektor 1990 through 1999** — ISBN 978-0-905705-76-7 .... £69.00 ...US $100.00
4. **DVD LED Toolbox** — ISBN 978-90-5381-245-7 .... £28.50 .....US $46.00
5. **Masterclass DVD High-End Valve Amplifiers** — ISBN 978-0-905705-86-6 .... £24.90 .... US $40.20

Kits & Modules

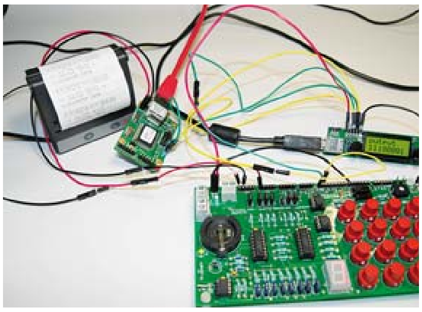
1. **NetWorker** — Art. # 100552-91 .... £53.00 .....US $85.50
2. **Digital Multi-Effects Unit** — Art. #090835-71 .... £165.00 US $266.20
3. **Reign with the Sceptre** — Art. # 090559-91 .... £89.00 ...US $143.60
4. **Elektor DSP radio** — Art. # 100126-91 .... £149.00 US $240.40
5. **InterSceptre** — Art. # 100174-71 .... £116.00 US $187.10

elektor

Elektor
Regus Brentford
1000 Great West Road
Brentford TW8 9HH • United Kingdom
Tel. +44 20 8261 4509
Fax +44 20 8261 4447
Email: sales@elektor.com

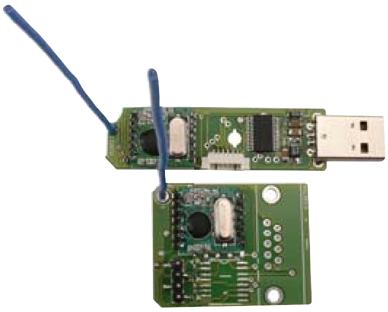
SatLocator

Anyone who regularly needs to align a mobile satellite dish (for example, on top of the caravan), is forever busy locating these 'birds' in the sky. This handy circuit is linked to a database containing popular TV satellites and employs GPS data to calculate the angle at which the satellite can be received. The hardware consists of little more than a GPS receiver module, an ATmega8 microcontroller and an LCD. Very handy for use on your next vacation!



Minimod18 Web Server

In this instalment of the Minimod18 (ATM18) series we put this AVR controller 'brick' at the heart of a basic web server. For the client we propose the Firefox browser which ensures error-free HTML viewing. A type EZL-70 module is in control and ensures a smooth adjustment between the various components.



Wireless Telemetry for Elektor Wheelie

ElektorWheelie, our popular self-balancing two-wheel vehicle gets a transmitter/receiver combination that allows lots of live system data such as battery voltage, speed, power per engine and tilt angle to be transmitted wirelessly to a laptop. A related program processes and visualizes data, giving a good impression of how your Wheelie is behaving and what software remains to be tweaked.

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

Elektor UK/European March 2011 edition: on sale February 17, 2011. *Elektor USA March 2011 edition: published February 9, 2011.*

w.elektor.com www.elektor.com www.elektor.com www.elektor.com www.elektor.com w

Elektor on the web

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

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

Also on the Elektor website:

- Electronics news and Elektor announcements
- Readers Forum
- PCB, software and e-magazine downloads
- Time limited offers
- FAQ, Author Guidelines and Contact

Order Form 02-2011

Description		Price each	Qty.	Total	Order Code
DVD Wireless Toolbox1	NEW	£28.50			
CD The Power Supply Collection 1	NEW	£17.90			
Fundamental Amplifier Techniques with Electron Tubes	NEW	£65.00			
ARM Microcontrollers	NEW	£29.50			
Experiments with Digital Electronics		£26.50			
C# 2010 Programming and PC interfacing		£29.50			
LEDs 1 (Special Project)		£9.90			
		Sub-total			
		P&P			
		Total paid			

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

Name

Address + Post code

Tel. *Email*

Date – – *Signature*

EL02

METHOD OF PAYMENT

(see reverse before ticking as appropriate)

- ☐ **Bank transfer**
- ☐ **Cheque** (UK-resident customers ONLY)
- ☐ VISA ☐ EUROCARD MasterCard

Expiry date: ____________________

Verification code: ______________

Please send this order form to*
(see reverse for conditions)

Elektor
Regus Brentford
1000 Great West Road
Brentford TW8 9HH
United Kingdom

Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
www.elektor.com
sales@elektor.com

*USA and Canada residents should use $ prices, and send the order form to:

Elektor US
PO Box 180
Vernon CT 06066
USA
Phone: 860-875-2199
Fax: 860-871-0411
E-mail: sales@elektor.com

Subscription Order Form 02-2011

Yes, I am taking out an annual subscription to Elektor and receive the E-book Microcontroller Basics totally free!*

(Please fill in your emailaddress below)

I would like:

- ☐ **Standard Subscription (11 issues)**
- ☐ **Subscription-Plus**
  (11 issues plus the Elektor Volume 2010 DVD-ROM + exclusive access to www.elektor-plus.com) TIP

* *Offer available to Subscribers who have not held a subscription to Elektor during the last 12 months. Offer subject to availability. See reverse for rates and conditions.*

Name

Address + Post code

Tel. *Email*

Date – – *Signature*

EL02

METHOD OF PAYMENT

(see reverse before ticking as appropriate)

- ☐ **Bank transfer**
- ☐ **Cheque** (UK-resident customers ONLY)
- ☐ VISA ☐ EUROCARD MasterCard

Expiry date: ____________________

Verification code: ______________

Please send this order form to

Elektor
Regus Brentford
1000 Great West Road
Brentford TW8 9HH
United Kingdom

Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
www.elektor.com
subscriptions@elektor.com

ORDERING INSTRUCTIONS, P&P CHARGES

All orders, except for subscriptions (for which see below), must be sent BY POST or FAX to our Brentford address using the Order Form overleaf. Online ordering: www.elektor.com/shop
Readers in the USA and Canada should send orders, except for subscriptions (for which see below), to the USA address given on the order form. Please apply to Elektor US for applicable P&P charges. Please allow 4-6 weeks for delivery.
Orders placed on our Brentford office must include P&P charges (Priority or Standard) as follows: Europe: £6.00 (Standard) or £7.00 (Priority) Outside Europe: £9.00 (Standard) or £11.00 (Priority)

HOW TO PAY

All orders must be accompanied by the full payment, including postage and packing charges as stated above or advised by Customer Services staff.
Bank transfer into account no. 40270211 held by Elektor International Media BV with The Royal Bank of Scotland, London.
IBAN: GB96 ABNA 4050 3040 2702 11. BIC: ABNAGB2L. Currency: sterling (UKP).
Please ensure your full name and address gets communicated to us.
Cheque sent by post, made payable to Elektor Electronics. We can only accept sterling cheques and bank drafts from UK-resident customers or subscribers. We regret that no cheques can be accepted from customers or subscribers in any other country.
GCredit card VISA and MasterCard can be processed by mail, email, web, fax and telephone. Online ordering through our website is SSL-protected for your security.

COMPONENTS

Components for projects appearing in Elektor are usually available from certain advertisers in this magazine. If difficulties in the supply of components are envisaged, a source will normally be advised in the article. Note, however, that the source(s) given is (are) not exclusive.

TERMS OF BUSINESS

Delivery Although every effort will be made to dispatch your order within 2-3 weeks from receipt of your instructions, we can not guarantee this time scale for all orders. **Returns** Faulty goods or goods sent in error may be returned for replacement or refund, but not before obtaining our **consent.** All goods returned should be packed securely in a padded bag or box, enclosing a covering letter stating the dispatch note number. If the goods are returned because of a mistake on our part, we will refund the return postage. **Damaged goods** Claims for damaged goods must be received at our Brentford office within 10-days (UK); 14-days (Europe) or 21-days (all other countries). **Cancelled orders** All cancelled orders will be subject to a 10% handling charge with a minimum charge of £5.00. **Patents** Patent protection may exist in respect of circuits, devices, components, and so on, described in our books and magazines. Elektor does not accept responsibility or liability for failing to identify such patent or other protection. **Copyright** All drawings, photographs, articles, printed circuit boards, programmed integrated circuits, diskettes and software carriers published in our books and magazines (other than in third-party advertisements) are copyright and may not be reproduced **or transmitted** in any form or by any means, including photocopying and recording, in whole or in part, without the prior permission of Elektor in writing. Such written permission must also be obtained before any part of these publications is stored in a retrieval system of any nature. Notwithstanding the above, printed-circuit boards may be produced for private and personal use without prior permission. **Limitation of liability** Elektor shall not be liable in contract, tort, or otherwise, for any loss or damage suffered by the purchaser whatsoever or howsoever arising out of, or in connexion 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 in respect of the goods. **Law** Any question relating to the supply of goods and services by Elektor shall be determined in all respects by the laws of England.

January 2011

SUBSCRIPTION RATES FOR ANNUAL SUBSCRIPTION

	Standard	Plus
United Kingdom & Ireland	**£51.00**	**£63.50**
Surface Mail Rest of the World	**£65.00**	**£77.50**
Airmail Rest of the World	**£82.00**	**£94.50**
USA & Canada	See www.elektor.com/usa for special offers	

HOW TO PAY

Bank transfer into account no. 40270211 held by Elektor International Media BV with The Royal Bank of Scotland, London. IBAN: GB96 ABNA 4050 3040 2702 11. BIC: ABNAGB2L. Currency: sterling (UKP). Please ensure your full name and address gets communicated to us.
Cheque sent by post, made payable to Elektor Electronics. We can only accept sterling cheques and bank drafts from UK-resident customers or subscribers. We regret that no cheques can be accepted from customers or subscribers in any other country.
Credit card VISA and MasterCard can be processed by mail, email, web, fax and telephone. Online ordering through our website is SSL-protected for your security.

SUBSCRIPTION CONDITIONS

The standard subscription order period is twelve months. If a permanent change of address during the subscription period means that copies have to be despatched by a more expensive service, no extra charge will be made. Conversely, no refund will be made, nor expiry date extended, if a change of address allows the use of a cheaper service.

Student applications, which qualify for a 20% (twenty per cent) reduction in current rates, must be supported by evidence of studentship signed by the head of the college, school or university faculty.
A standard Student Subscription costs £40.80, a Student Subscription-Plus costs £53.30 (UK only).

Please note that new subscriptions take about four weeks from receipt of order to become effective.
Cancelled subscriptions will be subject to a charge of 25% (twenty-five per cent) of the full subscription price or £7.50, whichever is the higher, plus the cost of any issues already dispatched. Subsciptions cannot be cancelled after they have run for six months or more.

January 2011