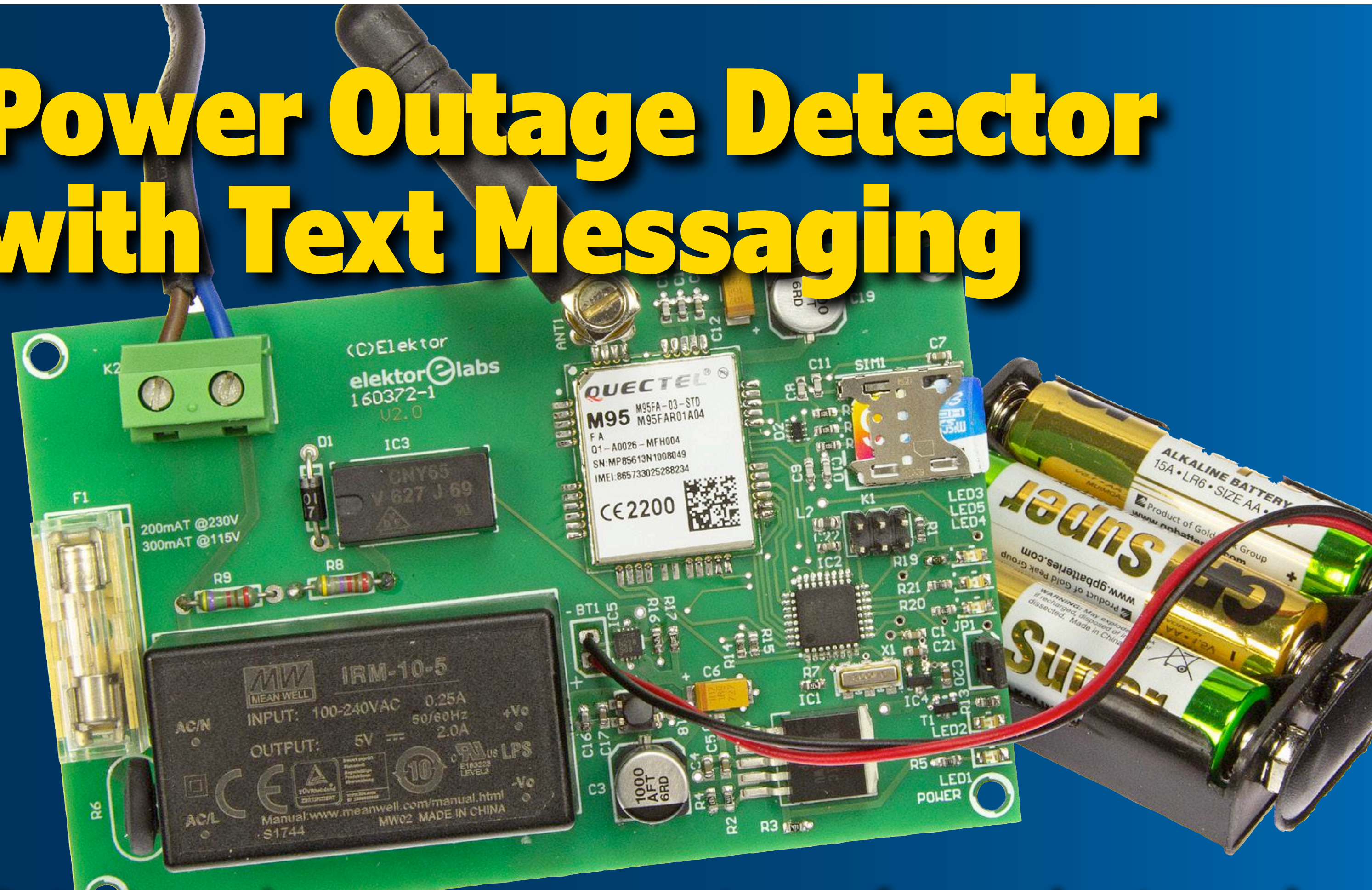
Edition 5/2018 | September & October 2018 | www.elektormagazine.com

elektorlabs magazine

DESIGN BUILD SHARE ELECTRONICS



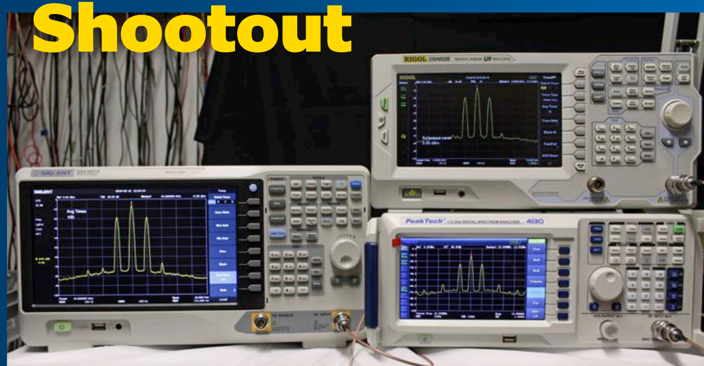
Power Outage Detector with Text Messaging

Detects and reports power outages (even short ones)

Find It!

Using 2.4 GHz to determine distance and direction to your lost item

Spectrum Analyser Shootout



Plenty of performance for pretty modest outlay

ESP8266 Web Server for NeoPixel LED Strips · Rain Detector · PWM Digital to Analogue Converter · The ESP32 Pico Discovery Board · LED Dimmers · Self-calibrating Frequency Meter · Phantom Power Supply · Sinusoidal FM LF Amplifier · Nostalgic LED Clock · Special Quality (SQ) Tubes · Hardware Design using (V)HDL · And More!

RND lab – PROFESSIONAL MEASURING TECHNOLOGY

HIGH QUALITY. LOW PRICES. SMART RESULTS.

TRMS multimeter, 6000 counts

RND lab

This professional multimeter convinces with a robust construction that can withstand industrial requirements. It also boasts a broad range of features for high precision and reliable measurements.

- Up to 1000 V AC/DC (±0.5 % + 2 digits)
- Non-contact voltage detection
- Analogue segment display on large white LED backlit display
- Intelligent data storage function
- AutoVolt: automatic AC/DC voltage selection

NCV NON CONTACT VOLTAGE

EN 61010-1 CAT III 1000 V

EN 61010-1 CAT IV 600 V

Order no.: RND 355-00003

PRICE TIP £ 87.09 (£ 72.57)

AC/DC clip-on ammeter

RND lab

- For conductors up to Ø 22 mm
- VoltSense for non-contact voltage detection
- 6000 count display
- Up to 300 A AC/DC, Up to 600 V AC/DC
- Backlit display

NCV NON CONTACT VOLTAGE

EN 61010-1 CAT III 600 V

Order no.: RND 365-00005

£ 73.36 (£ 61.14) TECHNOLOGY TIP

Non-contact voltage test

RND lab

When a voltage measurement is complete, this is indicated by brief illumination of the red LED in the tip of the detector, as well as a short signal tone.

- For testing from 100–1000 V AC
- Phase identification of 3-phase AC wires, multicore cables and outlets
- For measuring insulated cables and conductors

Order no.: RND 365-00003

£ 15.55 (£ 12.96)

EN 61010-1 CAT IV 600 V

EN 61010-1 CAT III 1000 V

Ideal for measurements in SMD assemblies!

LCR meter, 20,000 counts

RND lab

The toggle switch makes it easy to operate the device with only one hand.

- Integrated li-ion battery (up to 20 h charge)
- Resistance: up to 200 MΩ (resolution 1 mΩ)
- Capacity: up to 20 mF (resolution 0.001 pF)

MEASURES UP TO 2.5 TIMES PER SECOND!

Order no.: RND 365-00002

£ 132.94 (£ 110.79)

RND lab High quality, low prices, smart results. Discover RND lab ►

Daily prices! Price as of: 24. 7. 2018

Onlineshop languages:

Elektorlabs Magazine
(continued from: Elektor Magazine)

Edition 5/2018
Volume 44, No. 491
September & October 2018

ISSN 1757-0875 (UK / US / ROW distribution)

www.elektor.com
www.elektormagazine.com

Elektorlabs Magazine, English edition
is published 6 times a year by

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

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

Memberships:
Please use London address
E-mail: service@elektor.com
www.elektor.com/memberships

Advertising & Sponsoring:
Margriet Debeij
Phone: +49 170 5505 396
E-mail: margriet.debeij@eimworld.com

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

One letter ... come on!

Some argue that the BC547A, BC547B, BC547C and C847-CK180N are “some transistor” and the letter suffixes are just a clerical trick to suggest an extremely wide product range from the manufacturers mainly for printing massive catalogues and confounding competitors. Indeed any of the three ‘547s will work satisfactorily in a lowly application like an electronic switch to turn a 20-mA LED on and off. The same for a uA741 and a (fictional) UCLM741CC12NDP8-56C; or the case of a colourful 1-kΩ resistor from the nearby component tray *versus* a device with MCP25RRS102 printed on it but only legible using a magnifying glass. All these parts will let you get away with their use in simple, undemanding applications and you may wonder why the ‘difficult ones’ exist in the first place.
If you want to remain on the safe side in replicating a project or repairing a piece of electronics, stick with the exact specification of the designers as far as XXL part numbers go. If on the other hand you are adventurous and inquisitive and do not mind the odd e-mishap with an educational slant, then take your liberty to deviate from the book. The best result would seem to be when all is singing and dancing but with run of the mill components — “I told you”! In the worst case, nothing works but you can finally fathom and describe why the manufacturers have used such ‘weird’ components, while you can also abandon the thought of the big suppliers conspiring to throw dust at the unsuspecting hobbyist. Best case, worst case — it depends not on the outcome but on how you specify your final achievement: copycat or engineer?
The projects published in *ElektorLabs Magazine* are designed, written and produced (some in kit form) with an intention to also trigger “worst case” results at your end. Although we appreciate that you read the following pages solely intent on replicating the lab prototype and use the project in real life “no questions asked”, your effort at positively querying, modifying and enhancing the very same projects is valued higher. The place to post your success stories and mishaps is: www.elektormagazine.com/labs. In good engineering tradition you may do so using a nickname.

Jan Buiting, Editor-in-Chief

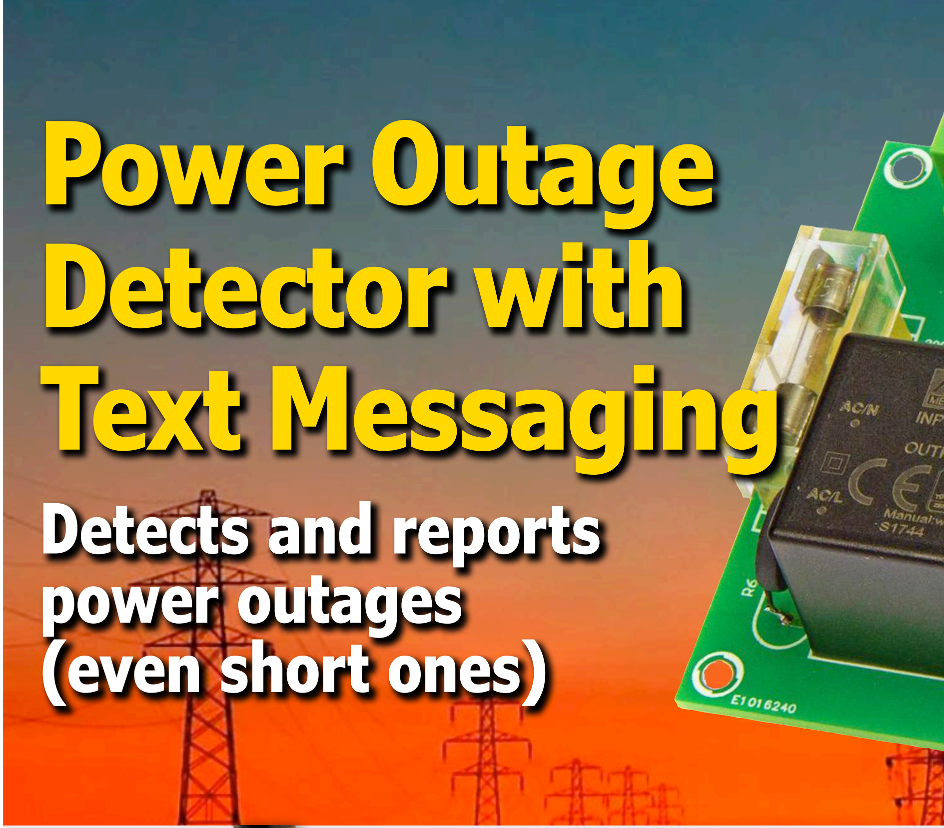
The Circuit

Editor-in-Chief:	Jan Buiting
Translators:	David Ashton, Jan Buiting, Martin Cooke, Ken Cox, Arthur deBeun, Andrew Emmerson, Tony Marsden, Mark Owen, Julian Rivers
Membership Manager:	Raoul Morreau
International Editorial Staff:	Eric Bogers, Mariline Thiebaut-Brodier Denis Meyer, Jens Nickel
Laboratory Staff:	Thijs Beckers, Clemens Valens, Ton Giesberts, Luc Lemmens, Jan Visser
Graphic Design & Prepress:	Giel Dols
Publisher:	Don Akkermans

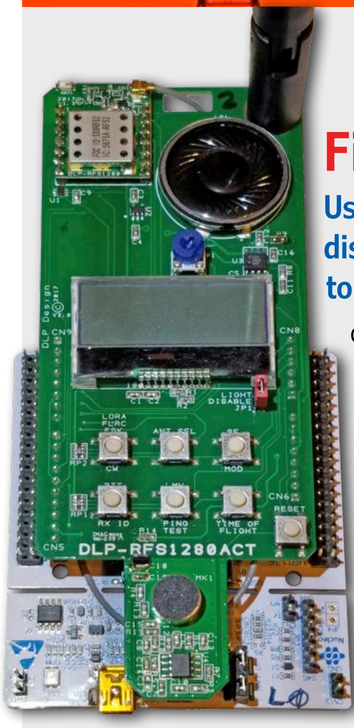
This Edition

Volume 44 – Edition 5/2018

No. 491 September & October 2018



Regulars



Find It!

Using 2.4 GHz to determine distance and direction to your lost item

6

Of the numerous technologies currently available to help you locate a lost item, most have the same limitations: they are only useful if the items are within Bluetooth's 30- to 100-foot (10 - 30 m) distance limit, or they require you to purchase a cellular modem and pay a monthly service fee so that you can send GPS data across the cellular network. This article will show you how to locate items that are up to miles away (without a cell phone, cellular network or GPS receiver) — not only showing you the distance, but also the direction to that item!

Features

Projects

RETRONICS

The pinnacle of tube technology

Special Quality (SQ) Tubes

104

elektorlabs magazine

Spectrum Analyser Shootout

Plenty of performance for pretty modest outlay

In the November & December 2017 issue of Elektor Magazine I examined the FFT function on modern digital oscilloscopes. Normally provided at no extra cost, this bonus feature is definitely useful for analysing signals but it also has its limits. Most electronicists who are used to working with a 'real' spectrum analyser will soon find these shortcomings a hindrance. In times gone by, dedicated analysers were prohibitively expensive, not only for a hobby lab but also for smaller firms, even if bought second-hand. Fortunately, those days are over and nowadays some pretty good devices can be had new at acceptable prices. For this reason, a comparison test is worthwhile.

Next Editions

ElektorLabs Magazine Edition 6 / 2018

Power-Saving Methods for Microcontrollers • Operational Amplifiers in Practice • Colour O-LEDs • VHDL Course (2) • ESP8266 in Model Building • Arduino does DDS • LED Dimmers (2) • Active Crossover Filter • Magic Colour Lamp • DIY Solder Station • AVO Tube Tester • USB to DMX Converter • RPI Buffer Board • Hamster Speedometer.

ElektorLabs Magazine edition 6/2018 covering November & December is published around 18 October 2018. Delivery of printed copies to Elektor Gold Members is subject to transport. Contents and article titles subject to change.

Elektor Business Edition 5/2018

Elektor Business Edition issue 5/2018 will focus on **Microcontrollers and Programming**. Contributions will come from companies and research institutions, as well as private authors, covering professional approaches to microcontroller-based design, new semiconductor technologies, embedded programming, and more. Plus, you'll find fresh instalments of all the EBE regulars like Infographics and Business Store.

Elektor Business Edition issue 5/2018 will be published in late September 2018 to Elektor Magazine Gold members in print, and Elektor Green members as a pdf download. The edition is also available for purchase at www.elektormagazine.com.

Find It!

Using 2.4 GHz to determine distance and direction to your lost item

By **Don Powrie** (USA)

Of the numerous technologies currently available to help you locate a lost item, most have the same limitations: they are only useful if the items are within Bluetooth's 30- to 100-foot (10 - 30 m) distance limit, or they require you to purchase a cellular modem and pay a monthly service fee so that you can send GPS data across the cellular network. This article will show you how to locate items that are up to miles away (without a cell phone, cellular network or GPS receiver) — not only showing you the distance, but also the direction to that item!

Applications for this type of locator are probably endless, but I think the ultimate would be a dog locator. A good friend of mine had a beautiful auburn Doberman that loved to chase rabbits through our suburban neighbourhood. Apollo was very large and very strong; and unless you had a tight grip on the leash at the moment he spotted a rabbit, he would be gone — out of sight — in an instant. Whenever this happened after the sun went down, it could take hours of searching the neighbourhood through a thousand potential hiding places between houses and in the adjacent alleys.

SX1280, 2.4 GHz and ranging

While it's true that 2.4 GHz has inherent limitations such as its inability to easily penetrate walls and other solid objects, it does have the advantage of being approved for license-free use worldwide. Going a step further, Semtech™ added LoRa® modulation to their SX1280 transceiver IC for applications that require greater link budget with a resulting improved range. There is a lot of information about LoRa modulation available on the web, so we need not go into it here. But, suffice it to say, the receive sensitivity for LoRa mode in the SX1280 is significantly improved as compared to most, if not all, non-LoRa 2.4-GHz transceivers. Granted, this improvement in sensitivity is derived via LoRa "spreading" within the digital realm, but the net effect is the same — vastly improved TX/RX distance between transceivers.

Figure 1. DLP-RFS1280 2.4-GHz LoRa Transceiver.

Long-range LoRa mode notwithstanding, I think the feature that sets the SX1280 apart from the rest is its industry-unique, built-in Ranging feature. Ranging works by sharing an ID between two SX1280-based transceivers by declaring one the Master, and then bouncing data packets off a Slave transceiver in LoRa mode and measuring the time of flight of those packets in order to calculate distance between the two. This may be an oversimplification, but the primary takeaway is the understanding that this feature is possible due to the high frequency clock in the Master (2.4 GHz) driving a 24-bit counter that measures the round-trip time of the data packet and the Slave's ability to respond to these "ranging" packets without host microcontroller intervention. Once placed in Ranging mode, an SX1280-based Slave transceiver will receive and retransmit ranging packets purely from silicon, thereby providing the fastest possible turnaround time for the packet.

For my locator device, I selected DLP Design's DLP-RFS1280 pre-certified module (**Figure 1**) due in large part to its on-board chip antenna. If this locating device performed as well as I had envisioned, I would want as small an antenna as possible to keep the overall device compact.

Pairing

For any two RF transceivers to work together exclusively (ignoring other transceivers within range), they must share their IDs with each other. This process is called pairing. In this application, the host microcontroller is the STMicroelectronics™ STM32L073 32-bit device, implemented on a Nucleo microcontroller development board. Each STM32L073 has a unique ID in silicon that can be read from address 0x1FF80050. Of the 32 bits read from this address, I discarded 16, keeping the other 16 for the ID that can be sent to another transceiver.

To perform the pairing, a standardised set of RF parameters (TX/RX frequency, modulation type, bandwidth, etc.) are selected, and the ID is transmitted after any time the microcontroller is powered up or reset. To pair two transceivers, the transceiver receiving the ID is placed in a wait mode with the standard parameters selected, and the other transceiver is simply reset. This process is repeated for the other transceiver, and the pairing process is complete. (Any non-broadcast packets transmitted from this point forward must have the destination transceiver's ID in the payload or the packet is ignored.)

Time of flight

In order to measure the time of flight of an RF packet travelling at nearly the speed of light, you need a very high-resolution counter. The use of a 2.4-GHz clock in the SX1280 incrementing a 24-bit counter that ticks until the master receives the Ranging reply yields a timing resolution of about 400 picoseconds. Considering that the speed of light is about one foot per nanosecond, this yields a distance resolution of about 6 inches (15 cms) — a perfect system for measuring the round-trip time of Ranging packets and determining the distance between transceivers.

One of the hurdles in measuring the time of flight of RF packets is the effect of multipath or reflections. A single packet arriving at the receiver may have taken more than one path to get there, bouncing off objects along the way. For this reason, the RF carrier frequency is changed for every Ranging packet sent for a total of 40 channels between 2.402 and 2.48 GHz, and the process is repeated several times. The resulting time of flight for all of these packets is accumulated and used to compute the mean value so that the estimated distance between transceivers can be calculated.

System calibration

A detailed description of how one goes about calibrating this Ranging system, whereby the hardware implementation is characterised so that the desired accuracy can be achieved, is beyond the scope of this article. However, one way to go about it is to measure the time of flight of a standard distance between transceivers with no possibility of multipath signals (e.g. when using a 100-foot (30-m) coax cable), and base calculated distances on this standard in conjunction with the known speed of light. Another is to collect real-world data from outdoor, line-of-sight environments across a large number of distances ranging from 2 feet to 2 miles (60 cm to 3.2 km), and then use curve fitting to create polynomial equations based upon this data to calculate the distance. I used the latter of these two methods to calibrate my system, and **Figures 2 and 3** show our data-collection setup.

Figure 2. Short-range (2 – 900 ft) (0.6 – 300 m) calibration data collection.

Figure 3. Long-range calibration data collection.

For distances out to 2+ miles (3+ km), we visited an area lake where we could be guaranteed good line of sight across long distances, see Figure 3.

Direction locator

Knowing the distance from your position to another transceiver is a good start towards finding your lost item, but in which direction should you begin walking? My solution was to create an electronic locator of sorts (**Figure 4**) that uses a high-gain Yagi antenna [1]. I found this antenna design on the web, and modified it for slightly higher directionality by adding a couple of directors. The idea for determining direction is to use the RSSI (receive signal strength) value as reported by the Slave transceiver. In this scenario, the Locator sends a LoRa packet with Spreading Factor 12 to the Slave that measures the RSSI level of that packet and sends this data back to the Locator. Since the Locator is transmitting using a directional antenna, the Slave receives more signal (higher RSSI) when the Locator's antenna is pointed directly at the Slave. The intensity of this RSSI signal is then reported by the Locator both as an audible tone and using visual indicators on an LCD display to indicate direction to the target.

Figure 4. Locator with Yagi antenna.

To operate the Locator, simply select a Slave ID from the list of learned IDs (see below), and horizontally sweep the Yagi antenna slowly in all directions around your body while listening to the tone. As the antenna points more directly at the target, the pitch of the tone will increase. Once the direction is known, set the switch to Range to get a read on the distance and start walking. The distance as you approach your target will be reported in real time on the display.

Learn mode

Instead of having one transceiver paired exclusively with another, I designed the Locator to be able to learn the IDs of up to 50 transceivers. To enter Learn mode, press and hold the Up button, and power up the Locator by flipping the power switch on the bottom of the board. The Locator will then broadcast a packet that is received by all Slave transceivers within range requesting their IDs. Each Slave will wait a random amount of time (2-64 ms) before transmitting its ID. The Locator receives and stores these IDs in EEPROM memory so that one can be selected for the Search process, and then sends a packet to each new ID instructing it to remain quiet (i.e. not respond to the next Learn request).

To select a transceiver to locate, simply use the Up and Down buttons to scroll through and select a Slave transceiver ID. The STM32L073's built-in EEPROM memory is also used to store the selected Slave's transceiver ID, so it is used by default on the next power-up.

Slave transceivers

For the Slave transceivers, I used the DLP Design DLP-RFS1280ACT (**Figure 5**) since its firmware is nearly identical to that which is utilised in the Locator device.

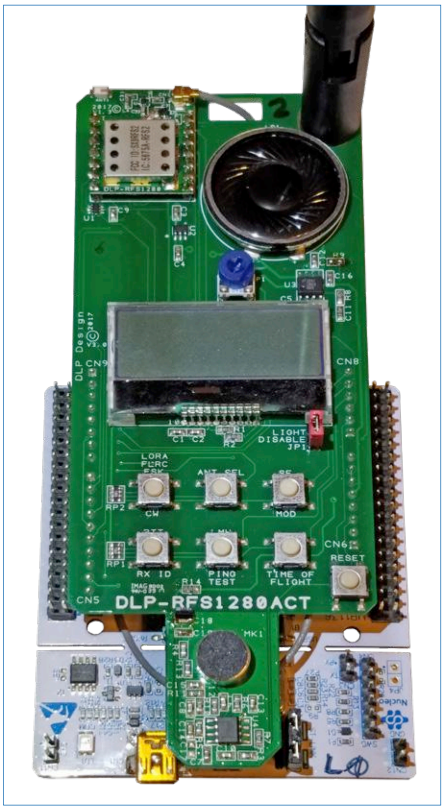


Figure 5. DLP-RFS1280ACT used as a Slave transceiver.

The project source code

For this project I started with C++ demo source code from Semtech for the SX1280 Demonstration Platform found on the Mbed website. (Most of the heavy lifting in the source code development was done by the engineers at Semtech using Mbed libraries.) All I had to do was convert it over from C++ to straight C code since I'm much more comfortable with the C programming language for microcontroller firmware development. Making this project even more appealing was the availability of a free C compiler from Keil for the STM32F0 and STM32L0 devices. Their MDK compiler [2] provides the ability to set breakpoints, single step through code while watching variables, etc. While Mbed is great for bringing up a new project quickly due to its built-in libraries, when you get into serious firmware development, a good debugger is indispensable.

The next step was to select a different STM32 microcontroller. The device used by Semtech for their demo hardware was the STM32L476 by way of the NUCLEO-L476RG. This is a very powerful micro with a ton of flash program memory (1 MB), but it's also a bit power hungry.

This is a great device for use in development tools when you don't know what you are going to need down the road and you want to make sure you don't run out of horsepower or memory. In this case, I chose to go with the STM32L073 to save power (which is appropriate for devices like a battery-driven pet collar), to save money and because Mbed supports the NUCLEO-L073RZ.

I created a nearly empty shell program (keeping the SPI interface code and a few other items) on the Mbed website, and immediately exported it to MDK. Doing so brought along the entire Mbed library, so I could take advantage of other handy Mbed features if needed. From there, the bulk of the code conversion to straight C was fairly simple. Again, most of the code for working with the SX1280 was already done for me, so the overall process only took a few days. The resulting C source code is fairly easy to follow and understand, and is available for download from the DLP Design website as well as the *ElektorLabs Magazine* web page supporting this publication [3].

Parting comments

By using a Yagi antenna and LoRa modulation, I was able to achieve very positive results with my locator system. I successfully located other transceivers through houses in our neighbourhood and between several sets of locations within our home.

Expanding upon the concept, I took the system to three different, large retail establishments to see how it would perform in environments with lots of metal shelving, customers, etc. (I'll call these three locations Store L, Store T and Store W.) In each of the three stores, I placed a Slave transceiver in one corner and took the Locator to the exact opposite corner of the store. In Store L, I was able to easily ping the Slave transceiver with no dropped packets at a reported distance of 505 feet (168 m). The problem was determining an exact direction to the Slave. Regardless of where I pointed the Locator, the Slave always reported approximately the same RSSI value. I was able to improve upon this result by moving to the centre of the store where the RSSI values were higher when pointing at the Slave. So, in this test environment, the Locator could immediately report the presence of the Slave target and the distance; but in order to determine the direction, I had to move around a bit. In Store T, I once again placed a Slave transceiver in one corner of the store. This time I was not able to ping the Slave from the opposite corner of the store at a distance of ~575 feet (~192 m). (It wasn't until I positioned myself in the centre of the store that I was able to get a ping reply. From there I could get good direction readings as well. Store W gave the same results as Store L with a corner-to-corner distance of 505 feet (168 m). From this I concluded that so long as I'm willing to walk around a bit, the Locator works well within a large retail environment, especially if I start at the centre of the store.

As expected, the best results are achieved when used outdoor using line of sight. There I was able to determine distance and direction to Slave transceivers located up to ~2 miles (~3.2 km) away.

In the end, while this system might not curb Apollo's yearning to catch that rabbit, it would certainly help my friend find his pet in the wee hours of the morning! ◄

180175-01

References and Web Links

[1] Application Note DN034 — SWRA350, 2.4 GHz YAGI PCB Antenna, By Richard Wallace & Steve Dunbar
www.ti.com/general/docs/litabsmultiplefilelist.tsp?literatureNumber=swra350

[2] ARM KEIL Compiler, www2.keil.com/stmicroelectronics-stm32/mdk

[3] Source code files for project: www.elektormagazine.com/180175-01

@ WWW.ELEKTOR.COM

→ **Dragino LoRa IoT Development Kit (868 MHz)**
www.elektor.com/dragino-lora-kit-868-mhz

→ **Elektor mbed Interface (150554-71)**
www.elektor.com/elektor-150554-71

→ **Dragino LoRa/GPS Shield for Arduino (868 MHz)**
www.elektor.com/dragino-shield-for-arduino-868-mhz

→ **STM Nucleo-L476RG Board**
www.elektor.com/stm-l476rg-board

'Top' Stencil on the Top

By **Jennifer Aubinais** (France)

Several years ago, during the development of the project 'Bluetooth Low Energy Wireless Thermometer' [1], I needed to position a small Bluetooth module with great precision on the printed circuit board before being able to reflow-solder it in the oven. The module in question has three locating holes with a diameter of 1.6 mm, so with the aid of three small screws of the same diameter as the holes I was able to position the module with great precision.

The project only used SMD components which I had wanted to solder in an oven. To apply the solder paste correctly on the printed circuit board, I was going to use a stencil. But how to place the stencil correctly on this small printed circuit board? Using these three little holes, of course!

I use Eagle for printed circuit board development. This software works with layers, for example I used one for the top copper layer '1 Top', one for the bottom copper layer '16 Bottom', etc. and also one for the top stencil, that was the layer '31 tCream'. However, in this layer you only find the lead outlines of the SMD components, there are no other openings, holes or other cutouts there.

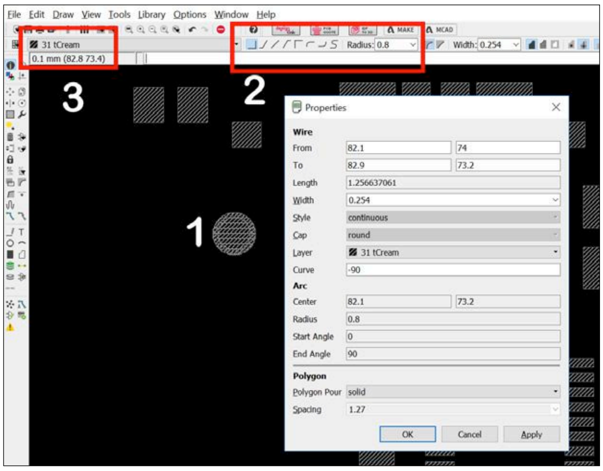


Figure 1. A 'circular' polygon (1) drawn in Eagle using the 'Wire bend' buttons and 'Radius' value (2) on the layer '31 tCream' (3).

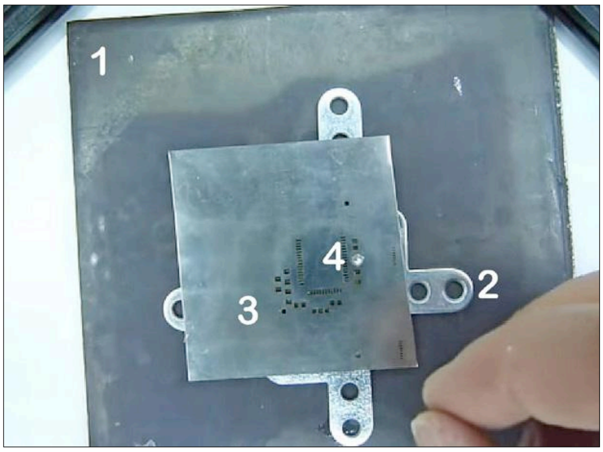


Figure 2. Screenshot from the video [2]: a magnetic sheet (1) four steel brackets (2) to hold the printed circuit board, then the stencil (3) and an aligning screw (4).

So I had to insert the holes manually, which is possible with the 'Polygon' function and a bit of patience (**Figure 1**).To draw a 'circular' polygon, experiment with the 'Wire bend' buttons and the value of 'Radius' (which appear at the top when you select the polygon function). Remember to adapt the thickness of the contour of the polygon to suit the desired final diameter. Leave enough space around these holes to allow the stencil to be fixed in position with screws and nuts (see below).

Another solution is to place the holes in the right places with the 'Hole' function as you would with any other mounting hole, and then to liaise with the manufacturer of the stencil. Communication is everything in this case. However you do it, it's a good idea to check the stencil before ordering it.

With the printed circuit board and your customised stencil in hand, proceed as follows:

- On your worktop, lay a magnetic sheet which is bigger than the printed circuit board. This must have a strong enough magnetic strength (for example 20 g/cm²) in order to hold steady the steel brackets (see step 3). This type of sheet is easily found on the Internet for a reasonable price. If the magnetic strength is not shown, choose another one.
- Place the printed circuit board in the middle of the magnetic sheet.
- Wedge the printed circuit board in place with four steel strips or brackets with a thickness ideally of 1.8 mm (get these from your favourite DIY store).
- Place the stencil on the printed circuit board.
- With the aid of the screws and nuts, position the stencil to exactly line up with the holes. The nuts on the screws serve to limit the length of the screws in the holes (**Figure 2**).
- Secure the position of the stencil on the brackets with some strong magnets (preferably neodymium type).
- Remove the screws.
- Spread the solder paste.

That's how to do some precision work at home with a bit of patience and imagination. I made a short video which shows how to do this [2].

180332-02

Web Links

[1] Thermometer with Bluetooth Low Energy: Elektor 1/2015: www.elektormagazine.com/140190

[2] Video: www.youtube.com/watch?v=0YlKxtYwQiE

[3] Support web page: www.elektormagazine.com/180332

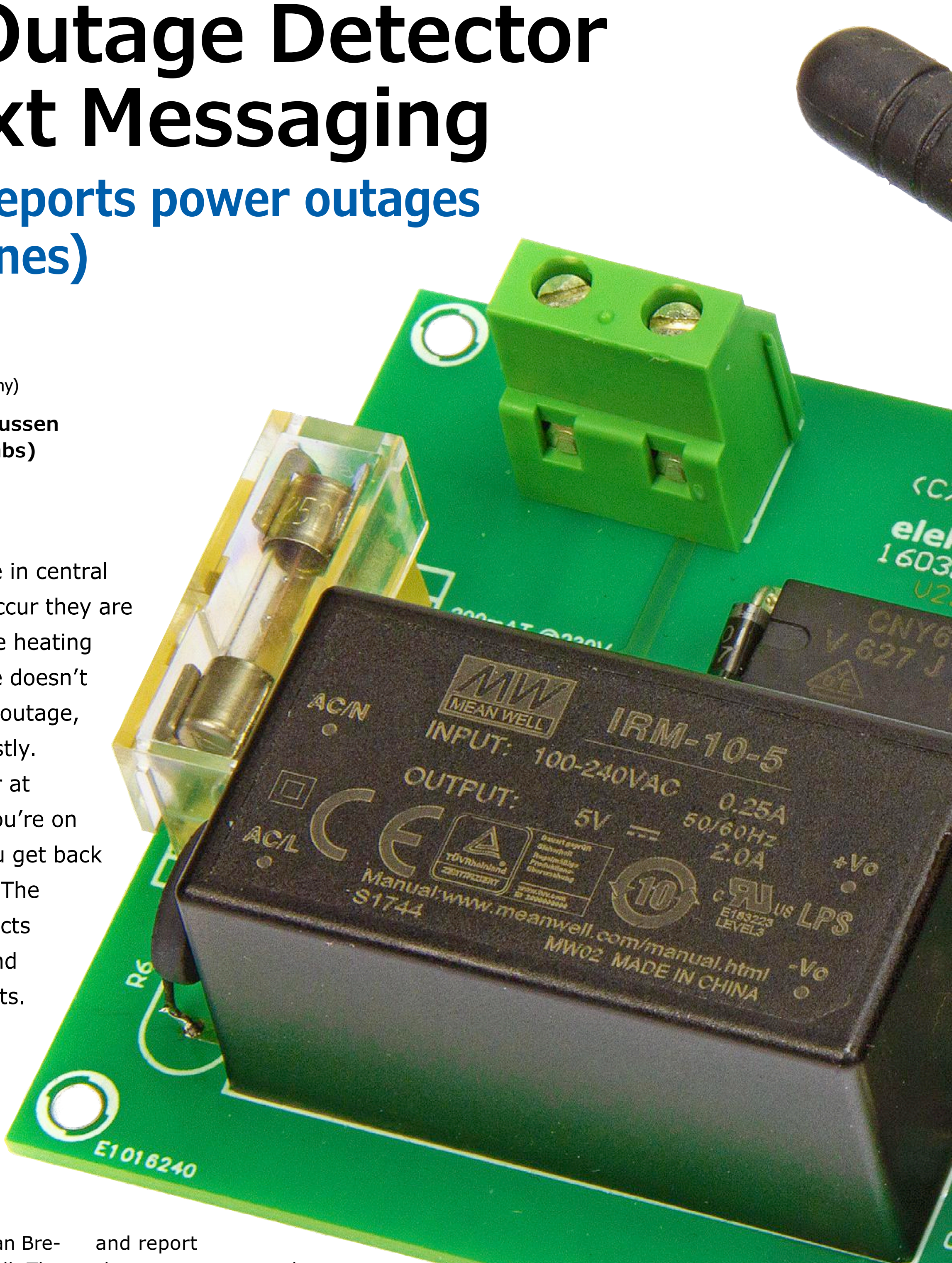
eLABS PROJECT

Power Outage Detector with Text Messaging

Detects and reports power outages (even short ones)

Idea: **Horst van Bremen** (Germany)

Development and text: **Mathias Claussen** and **Luc Lemmens (Elektor Labs)**



Power outages are fairly rare in central Europe, and when they do occur they are usually short. However, if the heating system of your holiday home doesn't start up again after a power outage, the consequences can be costly. And even if only your freezer at home stops working while you're on holiday, the stench when you get back can spoil your homecoming. The detector described here detects even short power outages and sends you text message alerts.

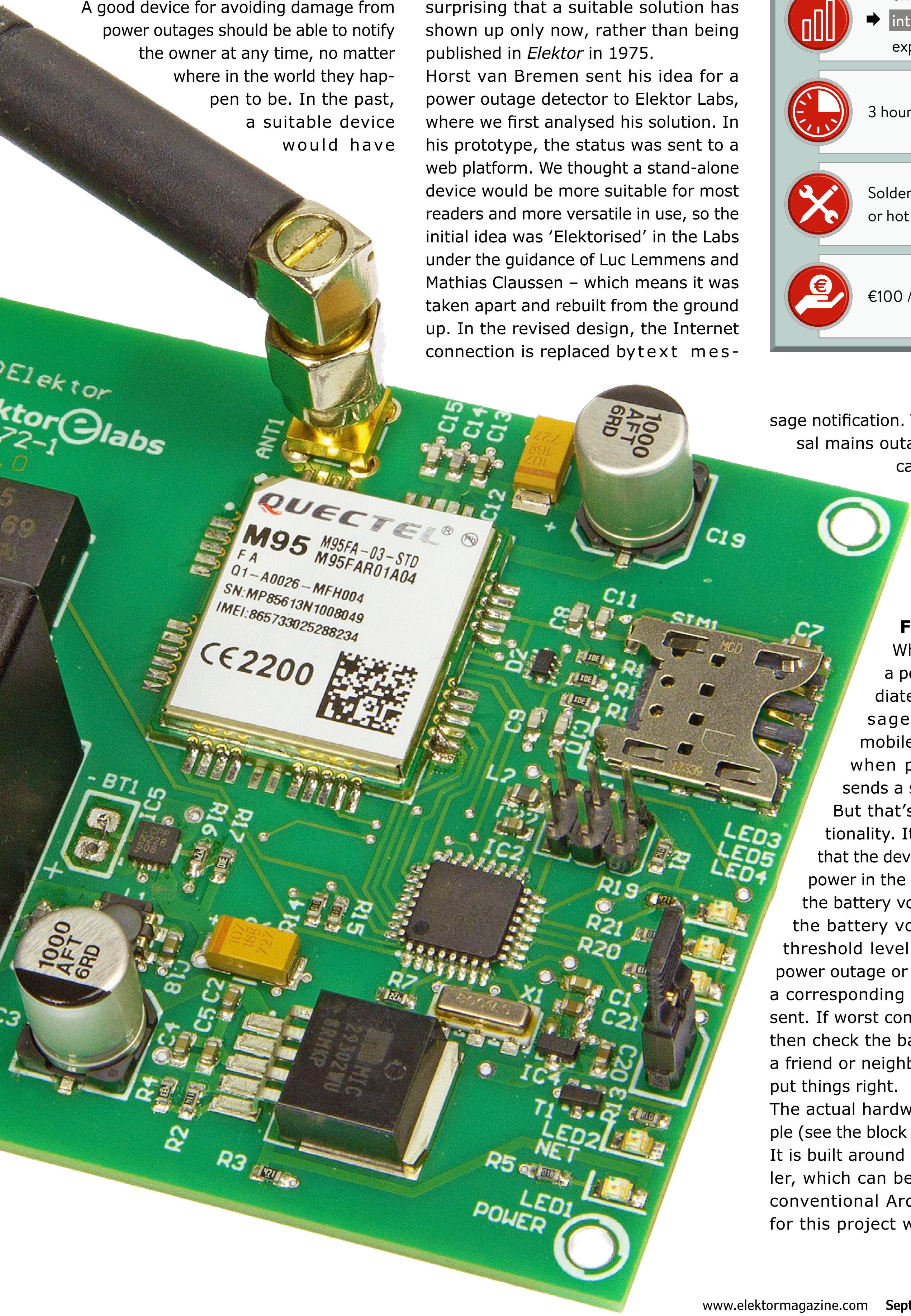
Elektor Magazine reader Horst van Bremen knows about this all too well. The heating system in his holiday home remained off after a power outage, and extensive renovation was necessary to clean up the resulting mould damage. After this costly experience, he decided it was time to take precautions, so he started working on a device that would detect even very short power outages and report the status to a web platform.

The problem situation

Electrical and electronic devices do not always restart automatically after a power outage. With some devices, even a short outage – just a few cycles of the mains voltage – is enough to cause trouble. That can have serious consequence – not only with heating systems in holiday houses, but also with some freezers or sprinkler systems – if the owners are away on a summer holiday in a far-away location. It's not always possible for a neighbour to check things every day, or they may not even notice that something has stopped running. The resulting damage may be severe enough to justify the cost of electronic prevention.

A good device for avoiding damage from power outages should be able to notify the owner at any time, no matter where in the world they happen to be. In the past, a suitable device would have been complicated and/or expensive, but now in the age of microcontrollers, widespread mobile phone coverage and low-cost, easy-to-use GSM modules, implementing this sort of project is not especially difficult. Even if the benefits vastly outweigh the costs, it is therefore hardly surprising that a suitable solution has shown up only now, rather than being published in *Elektor* in 1975.

Horst van Bremen sent his idea for a power outage detector to Elektor Labs, where we first analysed his solution. In his prototype, the status was sent to a web platform. We thought a stand-alone device would be more suitable for most readers and more versatile in use, so the initial idea was 'Elektorised' in the Labs under the guidance of Luc Lemmens and Mathias Claussen – which means it was taken apart and rebuilt from the ground up. In the revised design, the Internet connection is replaced by text message notification. The result is a universal mains outage detector that you can use wherever there is a mains installation to be monitored and access to a mobile phone network.



PROJECT INFORMATION

Mains monitoring
Text messaging
GSM modem
ATmega

entry level
→ intermediate level
expert level

3 hours approx.

Soldering iron with fine tip or hot air soldering station

€100 / £90 / $120 approx.

Features

When the device detects a power outage, it immediately sends a text message to a configurable mobile phone number, and when power is restored it sends a second text message. But that's just the basic functionality. It goes without saying that the device has built-in backup power in the form of a battery, and the battery voltage is monitored. If the battery voltage drops below a threshold level due to an extended power outage or as a result of ageing, a corresponding text message is also sent. If worst comes to worst, you can then check the battery yourself or call a friend or neighbour and ask them to put things right.

The actual hardware is relatively simple (see the block diagram in **Figure 1**). It is built around an AVR microcontroller, which can be programmed in the conventional Arduino IDE. However, for this project we developed a com-

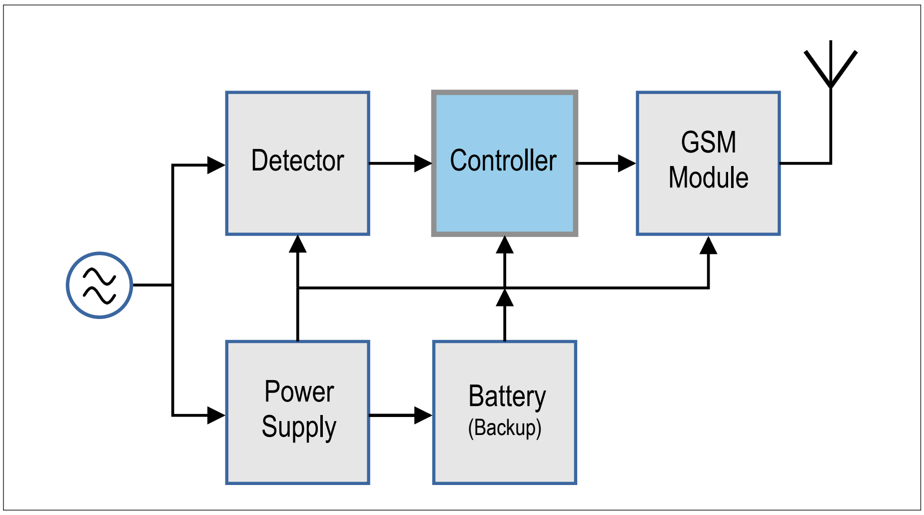


Figure 1. Block diagram of the power outage detector, consisting of five separate function modules.

pact program in C using AtmelStudio 7, as described in more detail later on in this article. The microcontroller needs to have sufficient resources to support the intended features of this project. We therefore switched from the original microcontroller, which had only 8 KB of flash memory and 512 bytes of RAM, to the tried-and-true ATmega328P, which boasts 32 KB flash and 2 KB RAM. Now there is also enough room for user extensions or modifications.

The firmware provides the following functions:

- text message on power outage;
- text message on power restoration;
- text message when the backup battery needs to be replaced (<3.9 V);
- text message when the backup battery is empty (<3.7 V);
- operation in low-power mode during power outage after sending the text message;
- configuration of phone number for sending alerts by text messaging;
- configuration storage in EEPROM.

To reliably avoid false alarms due to voltage spikes, waveform distortion or other reasons, the criterion for detection of a power outage is the absence of mains voltage zero crossings for a period of 250 ms. This means that a mains voltage dropout of more than 250 ms is regarded as a power outage. Along with simplicity, this form of detection has the advantage that it does not matter if the frequency of the monitored AC line is 50 Hz or 60 Hz.

LEDs

The yellow LED (LED1) is connected directly to the 5-V supply rail and is therefore lit when the AC line voltage is present. The green LED (LED2) is driven by MOD2 via T1 and indicates that the modem has logged in to a mobile phone network. Then there are the LEDs connected to the microcontroller. The red LED (LED3) is the battery status indicator, the orange LED (LED4) indicates the presence of the AC line voltage, and the green LED (LED5) shows the status of the GSM modem. Their functions are describe in the following table.

LED	Status	Function
LED3	Off	Battery OK or power outage
	Blinking	Weak battery, should be replaced soon
	On	Battery empty, text messaging no longer possible
LED4	Off	Power outage
	Blinking	AC line voltage present
LED5	Off	Modem not ready or power outage
	Blinking @ 1 Hz	Modem logged on but no number configured
	Blinking @ 10 Hz	Modem initialised but not logged on
	On	Modem logged on to GSM network and operational

Hardware

The highly popular ATmega 328P microcontroller (at the upper right in the schematic diagram in **Figure 2**) is the main component of the circuit. The positive half-waves of the mains voltage are sensed by optocoupler IC3, which provides galvanic isolation, and transferred to pin 1 of IC2. The input LED of the optocoupler has a low reverse breakdown voltage, so the negative half-waves are bypassed by diode D1. With a 230-V mains voltage, current limiting is provided by two resistors in series (R8 and R9) to avoid exceeding their maximum voltage rating. With a 115-V mains voltage, only one of the two resistors is necessary and the other can be replaced by a wire link. This detector configuration is virtually immune to voltage spikes on the mains, except in the very rare event of a direct lightning strike on the building service connection.

The encapsulated AC/DC converter [1] used here (MOD1) is a very practical alternative to a conventional transformer power supply. Along with fitting well on the circuit board, its ability to convert 115 or 230 VAC to 5 VDC makes it suitable for all mains systems in virtually every country in the world. In addition to protection by fuse F1, this module is protected against high inrush currents by the NTC thermistor R6, which is desirable because the converter module (MOD1) is a switching power supply with a capacitor after the input rectifier. On the secondary side, the low-drop voltage regulator IC1 is supplied with 5 V from MOD1, filtered by inductor L1 and a number of decoupling capacitors. IC1 converts this to 4.4 V, which supplies LED1 and LED2 over the V+ rail and is fed to the integrated fast switchover device IC5. This IC (type LTC4413 [2]) is described by the manufacturer as 'dual 2.6A, 2.5 to 5.5V fast ideal diodes'. That means it can switch the supply rail very quickly and without interruption to the 4.5 V from the battery BT1 in the event of a power outage. That happens when the voltage from IC1 drops below 3.5 V, which is sensed by pin 2 of IC5.

The 4.4 V output from IC5 powers the

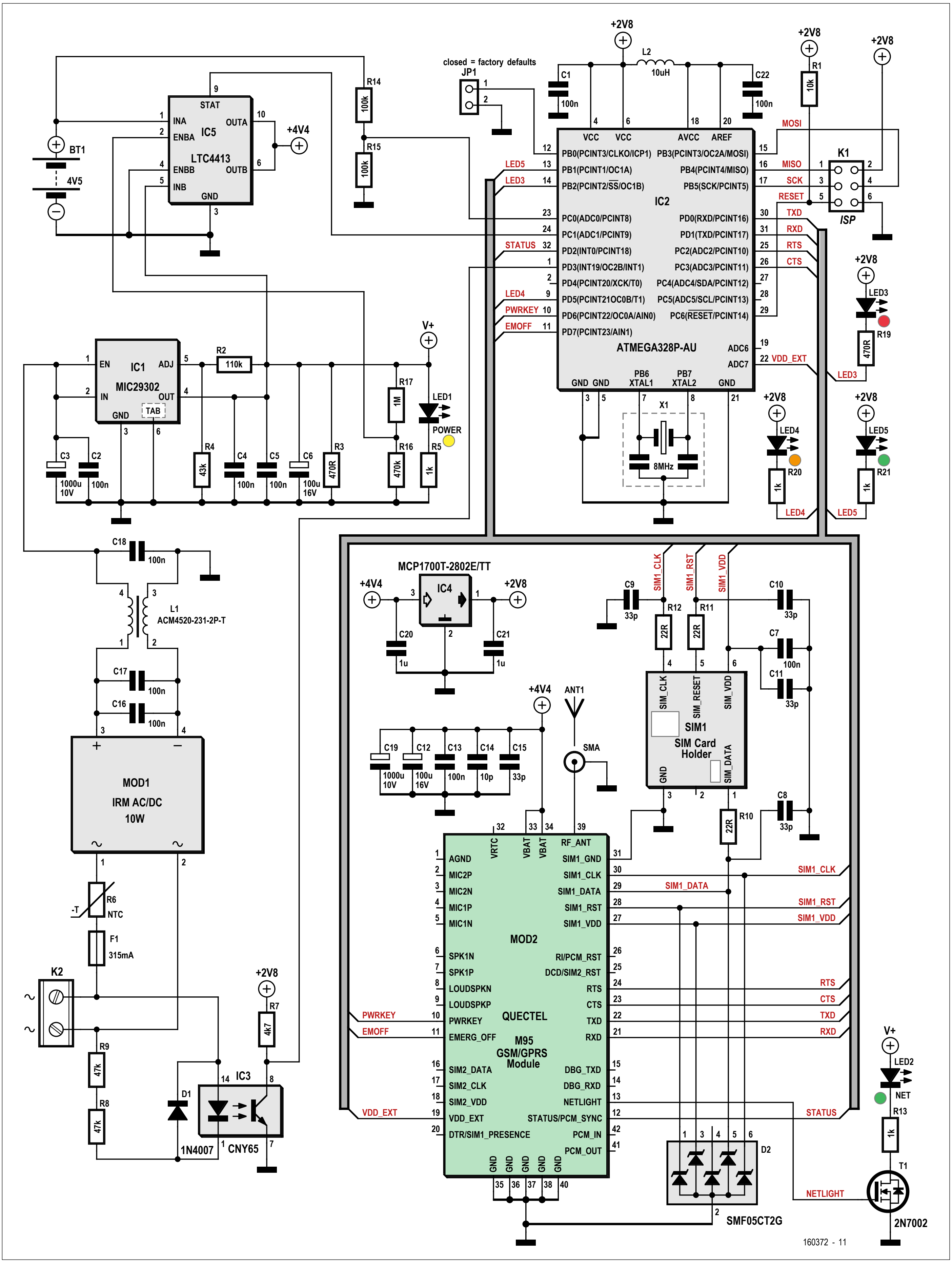


Figure 2. The circuit diagram of the power outage detector is fairly easy to understand after looking at the block diagram. The main component is the AVR microcontroller (IC2).

GSM module (MOD2) and the low-drop voltage regulator IC4, which in turn provides a 2.8 V supply voltage for the microcontroller and the LEDS (LED3–LED5). That is less than the maximum rated supply voltage of IC2, which helps keep power consumption low, and at the same time high enough for the 8 MHz clock rate set by resonator X1.

If you are wondering why we chose a 5 V version with 2-A load capacity for MOD1, the answer is that the GSM IC in MOD2 can easily draw a peak current of 1.6 A when transmitting to the mobile phone network. This brief power consumption is necessary to ensure adequate mobile range under poor conditions. That is also why an oversized electrolytic capacitor (C19) is provided as a buffer, since ordinary alkaline primary cells cannot handle such high peak currents on their own and the GSM module would automatically shut down if the supply voltage dropped below the 3.3-V threshold.

The rest of the circuitry around the M95 GSM module is largely the same as recommended by Quectel [3]. Its SIM card slot (SIM1) is protected against radiated RF interference by the combination of R10–R12, D2 and C8–C11.

The microcontroller monitors the voltage of the backup battery via the voltage divider R14/R15 and its analog input ADC0 (pin 23). The status of IC5 is sensed by pin 24. The status indications of the individual LEDs are explained in the **LEDs inset**.

Construction

The lab has designed a compact PCB (**Figure 3**) for the circuit in Figure 2, which can be purchased from the Elektor Store (where you can also purchase a pre-programmed microcontroller). Alternatively, you can use the layout files available from the Elektor web page for this project [4] to make your own PCB. As usual, all necessary code resources and the firmware hex file are also available on the web page.

As you can see from the picture of the finished prototype in **Figure 4**, assembling the board is not especially difficult. Although only a few of the components are through-hole types, at least there aren't any ultra-tiny SMDs or ICs with very narrow lead pitch. Hand soldering should therefore be possible, even with the SMD 0603 packages.

Mains voltage is present on the board, so it should be fitted in an insulated plastic

COMPONENT LIST

Resistors

Default: 1%, 0.1W, 50V, SMD 0603
R1 = 10kΩ
R2 = 110kΩ
R3,R19 = 470Ω
R4 = 43kΩ
R5,R13,R20,R21 = 1kΩ
R6 = NTC 80R B57236S0800M000, Epcos
R7 = 4.7kΩ
R8,R9 = 47kΩ, 5%, 0.6W, 250V, axial leads
R10,R11,R12 = 22Ω
R14,R15 = 100kΩ
R16 = 470kΩ
R17 = 1MΩ

Inductors

L1 = ACM4520-231-2P-T common-mode filter
L2 = BLM12AJ601SN1D, 10µH / 600Ω at 100MHz, , SMD 0805

Capacitors

Default: 50V, ceramic, SMD 0603
C1,C2,C4,C5,C7,C13,C16,C17,C18,C22 = 100nF
C3,C19 = 1000µF 10V, electrolytic, SMD, radial, 8mm diam.
C6,C12 = 100µF 16V, tantalum electrolytic, SMD 2312
C8-C11,C15 = 33pF, C0G/NP0
C14 = 10pF, C0G/NP0
C20,C21 = 1µF 10V

Semiconductors

D1 = 1N4007 diode, 1000V/1A
D2 = SMF05CT2G protection diode array
LED1 = yellow, SMD 0805
LED2,LED5 = green, SMD 0805
LED3 = red, SMD 0805
LED4 = orange, SMD 0805
T1 = 2N7002, n-channel MOSFET, SOT-23
IC1 = MIC29302WU TR, low-drop voltage regulator, adjustable
IC2 = ATmega328P-AU, microcontroller, programmed 160372-41
IC3 = CNY65, optocoupler
IC4 = MCP1700T-2802E/TT, low-drop voltage regulator, 2.8V
IC5 = LTC4413, dual fast diode / voltage changeover

Miscellaneous

MOD1 = AC/DC converter module, 5V/2A (Meanwell IRM-10-5)
MOD2 = Quectel M95 GSM module
X1 = 8MHz ceramic resonator, Murata CSTCC8M00G53-R0
SIM1 = micro SIM card socket, SMD (Molex 78723-1001)
ANT1 = SMA socket, PCB mount (Molex 73391-0070)
Matching GSM antenna with SMA plug (RF Solutions ANT-GHEL2R-SMA)
JP1 = 2-pin pinheader, vertical, 0.1" pitch
Jumper for JP1
F1 = fuse, T, 5×20mm, 315mA (230V) or 620mA (115V)
Fuse holder for F1 (Multicomp MC000827)
Fuse cover for F1 (MC000833) *
K1 = 6-pin (2×3) pinheader, vertical, 0.1" pitch
K2 = 2-way screw terminal block, 630V, 0.3" pitch
BT1 = battery holder for 3× AA cells
Clip for BT1
PCB 160372-1 V2.1

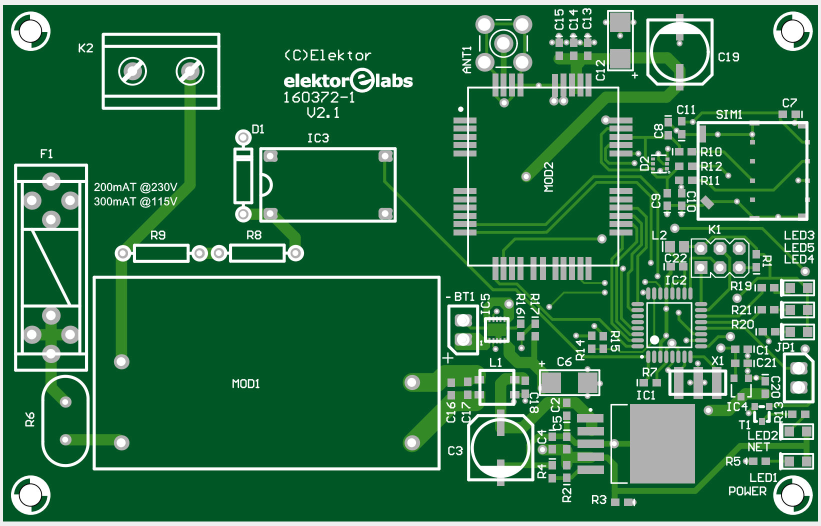


Figure 3. The PCB for the circuit in Figure 2 was designed in the Elektor Labs.

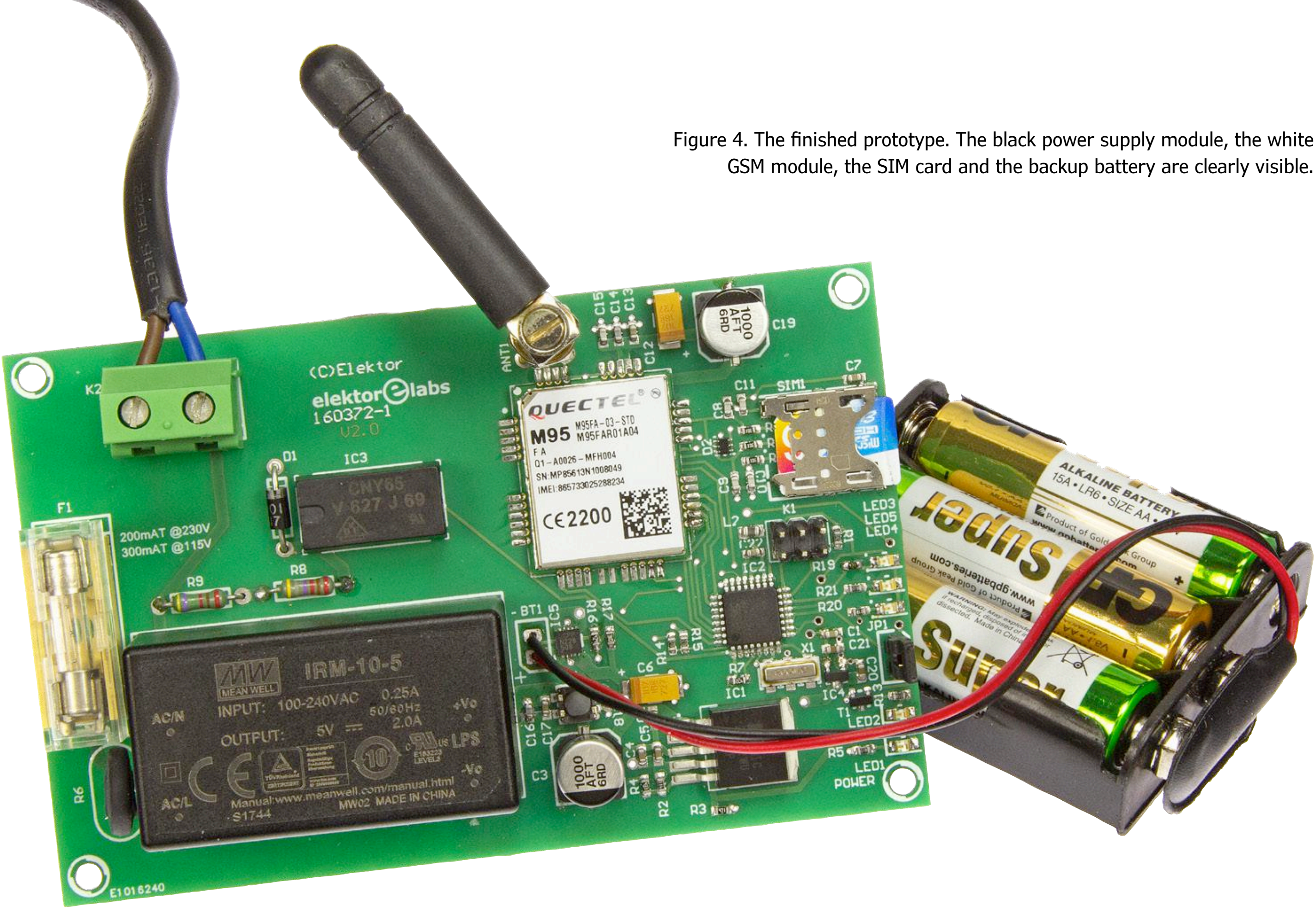


Figure 4. The finished prototype. The black power supply module, the white GSM module, the SIM card and the backup battery are clearly visible.

case. The battery holder for the three AA cells should be secured firmly so it does not fly around inside the case due to vibration or let go of any of the cells. The transparent cover for the fuse (visible on the left side) is also a good idea. Once the power outage detector is fully assembled and configured, you can take it with you anywhere and simply connect it to a mains power outlet with a suitable cable, ready to go.

Operation

Since backup power is provided by a battery (primary cells), the power consumption while operating in battery node during a power outage should be as low as possible. If the microcontroller is operated with a 2.8-V supply voltage and an external 8-MHz resonator, it draws about 4 mA. With a battery capacity of 1,500 mAh, that translates into just 375 hours (15.625 days) of operation in backup mode. The main load is the modem. It needs a maximum of 300 mA when transmitting a text message. However, it also draws that much current when it logs in to the mobile phone network or tries to log in. In the unrealistic situation of continuous modem operation, the total current consumption would be 304 mA, which means the battery would be fully discharged in about 5 hours. The battery cells would therefore have to be replaced after every power outage, if not earlier.

The current consumption of the microcontroller can be reduced by shutting down unnecessary portions. The ATmega328P has a Power Reduction Register (PRR) for this purpose. That way you can easily save 0.4 mA (10%). It also has an Idle mode, in which the processor core is shut down until an interrupt occurs. Using these options, the current firmware reduces the current consumption by about 50%, so the current is only 2 mA even though the program is still active. But the modem becomes a problem in the event of a power outage. Its current consumption of up to 300 mA is too much. For that reason, the modem is completely shut down after it successfully sends the power outage text message. On average the modem thus uses only a few microamps, which considerably extends the operating time in battery mode.

The final measure is to deactivate the LEDs in battery mode, since they also draw 2 to 3 mA. As the power outage detector is normally used stationary at a fixed site, it is not really necessary to indicate anything with the LEDs in the event of a power outage.

About the software

The flow chart in **Figure 5** shows the structure of the firmware written in C.

After initialisation, it retrieves the configuration data and arms the interrupts. To be on the safe side, it then remains in sleep mode until the voltage is present, so the battery is not drained if the system is not being powered from the mains. Then the GSM module is configured, and the battery voltage and the status of the mains network are acquired.
The microcontroller drives the indicator LEDs according to the currently available data. Now we get to the nitty-gritty: a text message is sent in the event of a power outage, restoration of power, weak battery or battery replacement. Then the firmware returns to sleep mode until something changes. If there is no mains voltage or the battery is empty, a reset occurs.
The software is divided into several task-specific modules. They are can be classified into four areas as shown in **Figure 6**: system monitoring, parameter storage, mains monitoring, and communication.
To detect a power outage, you can measure the time between zero crossings or specify a time interval within which two successive half-waves must occur. Both of these approaches are essentially based on the frequency of the AC voltage from the mains, but there is something that needs to be considered: switching processes sometimes occur in power grids to clear faults. These can cause brief transients, resulting in missing cycles and thus missing zero crossings. To avoid triggering a false alarm in such cases, you can simply increase the time interval to 250 ms and check if at least one zero crossing occurs within that period. With this method it is not necessary to distinguish between 50 Hz and 60 Hz power grids.
Mains monitoring (**Figure 7**) is handled in the C code by the components MAINS_MON (which does the actual monitoring) and a timebase in the module TIMER, which acts as a software watchdog. The Interrupt 1 input of the microcontroller, which is triggered by the falling edge of the applied signal, is used for this purpose. When a positive half-wave is present on the input, the optocoupler pulls the Interrupt 1 input low to produce a signal edge. In the interrupt routine, the distance between two successive interrupts is determine and the software watchdog of the TIMER module is reset. As long as this reset occurs within 250 ms, the watchdog does not time out.
But if there is a power outage, the reset does not occur, and then the watchdog times out and an error is indicated. Since the TIMER watchdog only responds to interrupts, power outage detection is fully independent of the main loop.

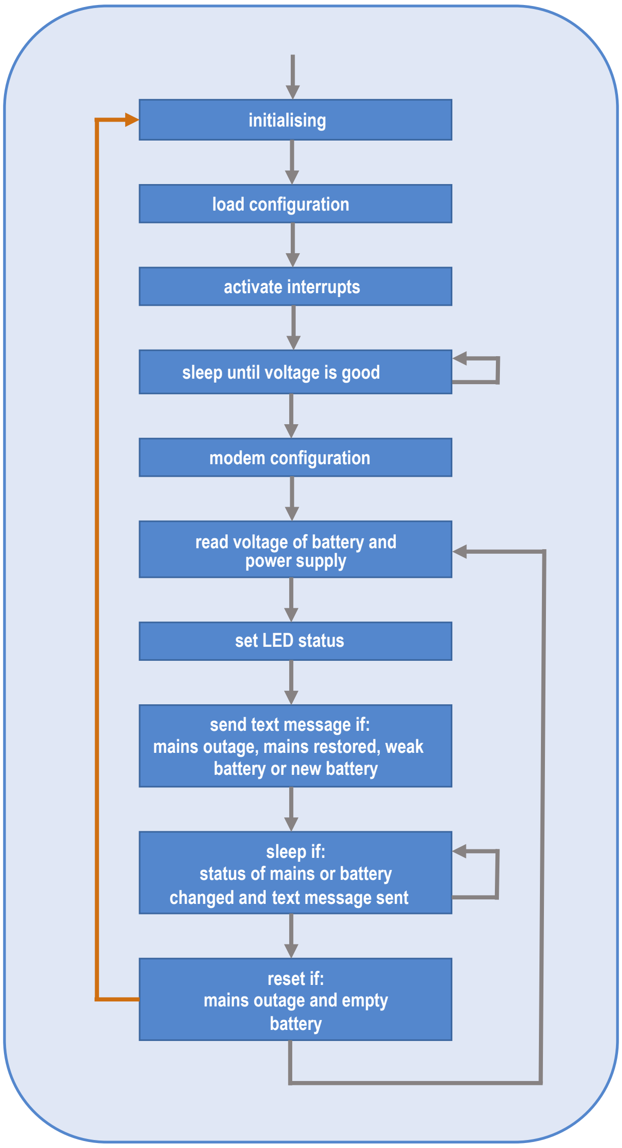


Figure 5. The flowchart of the firmware written in C.

Configuration

To get the system up and running, you need a valid SIM card without PIN protection. If the SIM is protected by a PIN, the software will not be able to use it. You can easily disable PIN protection of a SIM by inserting it in a mobile phone and then configuring the security settings accordingly.
The GSM module attempts to log in to a mobile phone network if the mains voltage is okay or the battery voltage is sufficient. It's a good idea to power the board from the AC line when you start it up the first time. If no AC line voltage is detected and the board is powered from the battery, the LEDs remain dark to save power, and that makes the configuration process difficult. Once the modem manages to connect to the mobile phone network, LED5 lights up. At this point you can call the phone number of the SIM card to check if the SIM card has been logged in properly and is

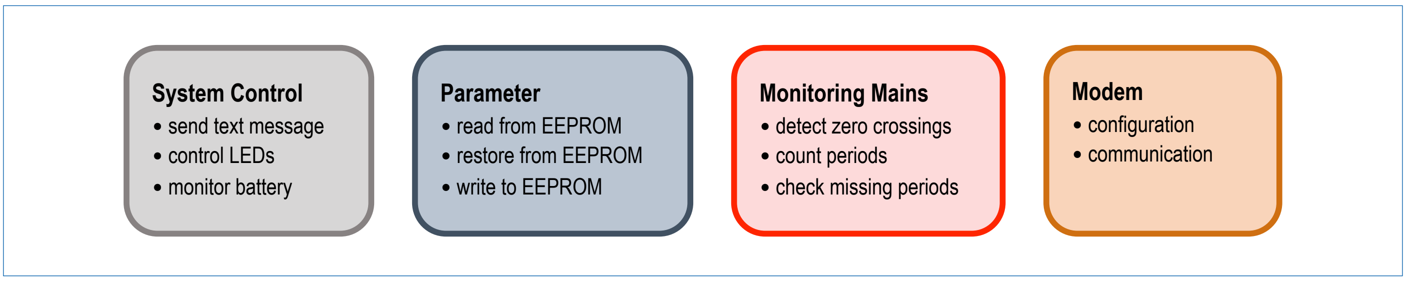


Figure 6. The firmware consists of four independent function modules.

not blocked by the network. However, the device will not issue any alerts because it does not yet have a valid phone number in the microcontroller EEPROM. This prevents the device from unintentionally sending text messages to an unknown number. The green LED should initially blink at a 1-Hz rate to indicate this status.

The phone number is set by a text message. For this you must use a mobile phone (with a different number) to send a text message with a specific content to the phone number of the SIM card inserted in the device. The format of the text message is ‘Config Remote xxx’, where ‘xxx’ stands for the phone number of the SIM card. Only the following characters are allowed here: +, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. If the text message is received and correctly processed, the device replies with the text message ‘Saved new number in EEPROM’. Now it is ready for use. After the reply text message has been sent, the green LED stops blinking and is constantly lit.

From this point on, the status of the system can be queried by text messaging. To do so, sent the text message ‘Request Status’ to the phone number of the SIM card in the device. The device will reply with the text message ‘Mains frequency is <STATUS> with <XX> Hz. Battery has <VBat> mV and>is <BAT_STATUS’.

To reset the current configuration, fit jumper JP1 and switch the device off and back on. A brief power interruption is necessary so that the state of JP1 can be sensed during booting. The green LED will again blink after this restart. You should then remove jumper so that the device will not be reset to the default configuration on the next restart.

For safety reasons (**mains voltage on the board**), you should only do this with the mains voltage disconnected. After this you can configure the board again as previously described.

(180344-03)

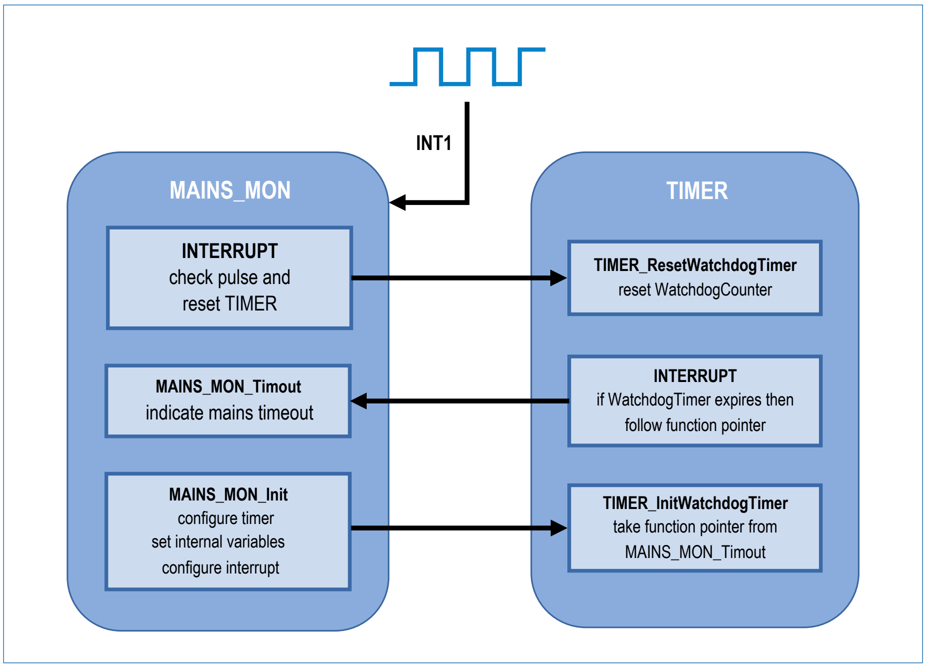


Figure 7. A watchdog implemented in software.

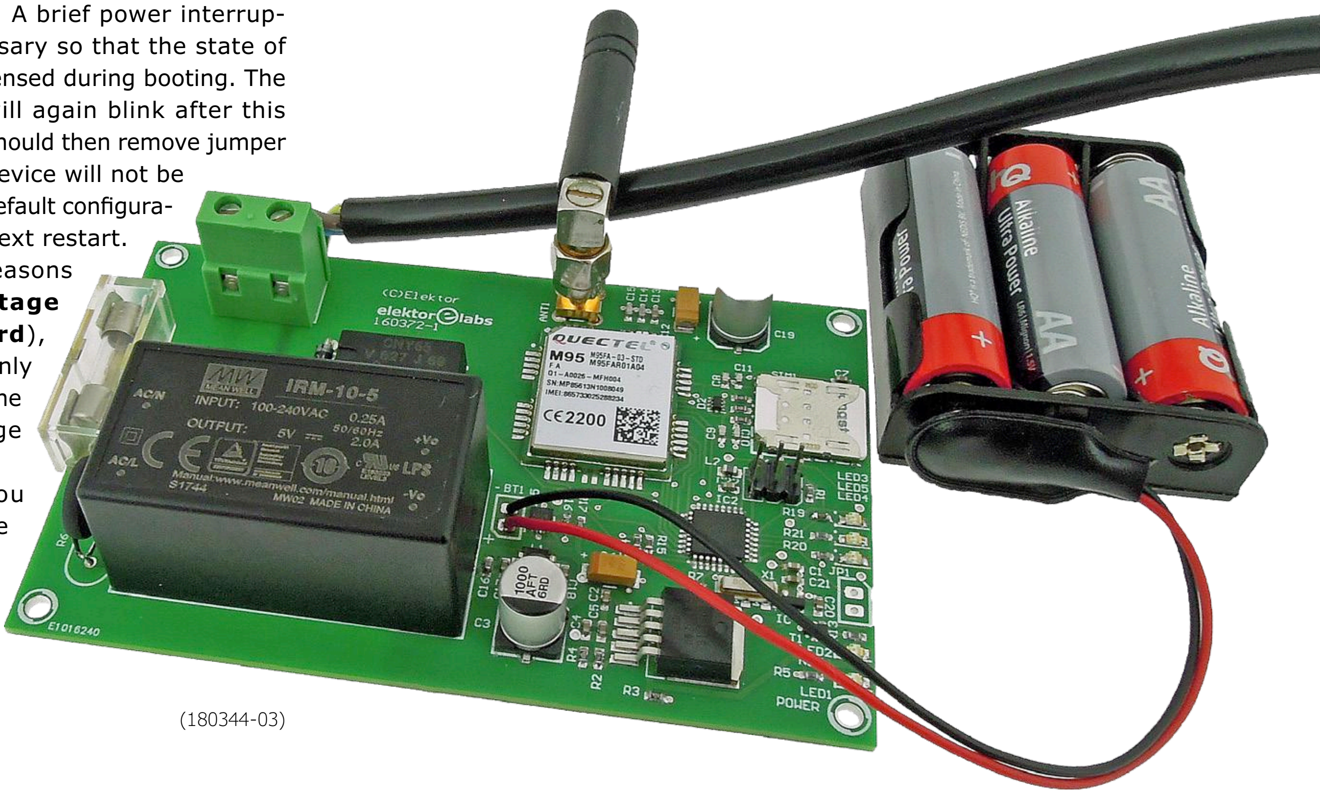
Web Links

[1] Meanwell IRM-10-5: https://bit.ly/2KsAsYr

[2] LTC4413: https://bit.ly/2Kfsiq6

[3] Quectel M95: www.quectel.com/product/m95.htm

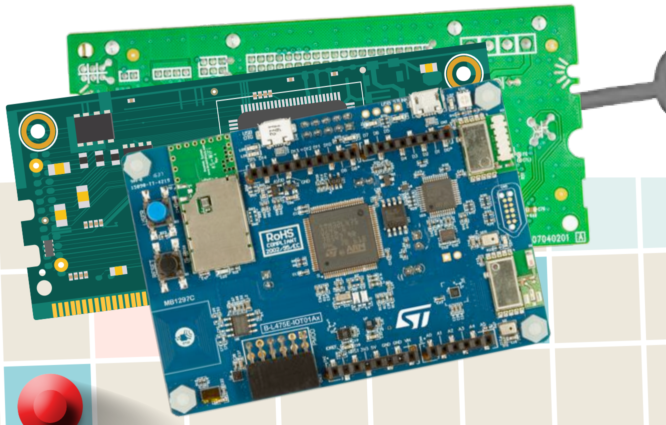
[4] Download: www.elektormagazine.com/180344-01



elektor

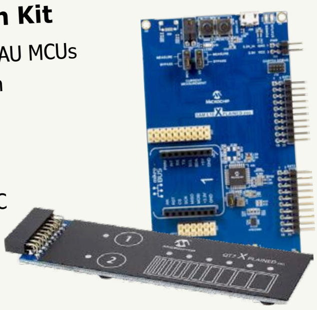
board games

POWERED BY **AVNET SILICA** AND **EBV ELEKTRONIK**



Turn your idea into a product!

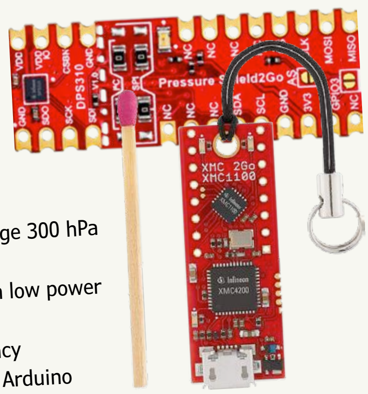
Microchip SAML10 Xplained Pro Evaluation Kit + QT7 Xplained Pro Extension Kit



- ATSAML10E16A-AU/ATSAML11E16A-AU MCUs
- 1 mechanical programmable button
- 1 QTouch button
- 1 user LED
- 32.768 kHz crystal
- ATECC508A CryptoAuthentication IC
- mikroBUS header
- Embedded debugger
- Embedded current measurement circuitry
- Self-capacitance water-tolerant touch + 8 LEDs

For datasheets and further product details:
www.elektormagazine.com/elektor-board-games-2018

Infineon XMC 2Go + S2GO Pressure DPS310



- XMC1100 MCU (ARM Cortex-M0 based)
- On-board J-Link Lite debugger
- 2 user LEDs
- 16-pin pinheader suitable for breadboard
- XMC software for Arduino IDE
- Ultra-high ±0.005 hPa resolution (equating to ±5 cm)
- Wide pressure operation range 300 hPa - 1200 hPa
- 3 µA current consumption in low power mode
- ±0.5 °C temperature accuracy
- Free and easy download of Arduino libraries

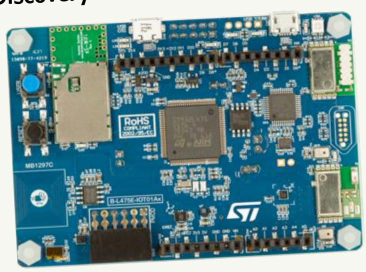
For datasheets and further product details:
www.elektormagazine.com/elektor-board-games-2018

Renesas Promotion Kit S5D9

- S5D9 Group MCUs with a 176 LQFP package
- Four extension connectors
- QVGA TFT touch screen
- 3 user LEDs
- Arduino Uno Shield compatible socket
- 2 mechanical switches
- 2 capacitive touch buttons + slider
- Audio output
- QSPI memory (8 MB)
- SPI, IIC, CAN, and SCI interface.

For datasheets and further product details:
www.elektormagazine.com/elektor-board-games-2018

STMicroelectronics STM32L4 Discovery Kit IoT Node



- STM32L4 Series MCUs (ARM Cortex-M4 based) in LQFP100 package
- Bluetooth V4.1 module (SPBTLE-RF)
- Sub-GHz (868 MHz or 915 MHz) low-power-programmable RF module
- 802.11 b/g/n compliant Wi-Fi module
- Dynamic NFC tag based on M24SR with its printed NFC antenna
- HTS221 capacitive digital sensor for relative humidity and temperature
- LIS3MDL 3-axis magnetometer
- LSM6DSL 3D accelerometer and 3D gyroscope
- LPS22HB absolute digital output barometer
- VL53L0X time-of-flight and gesture-detection sensor

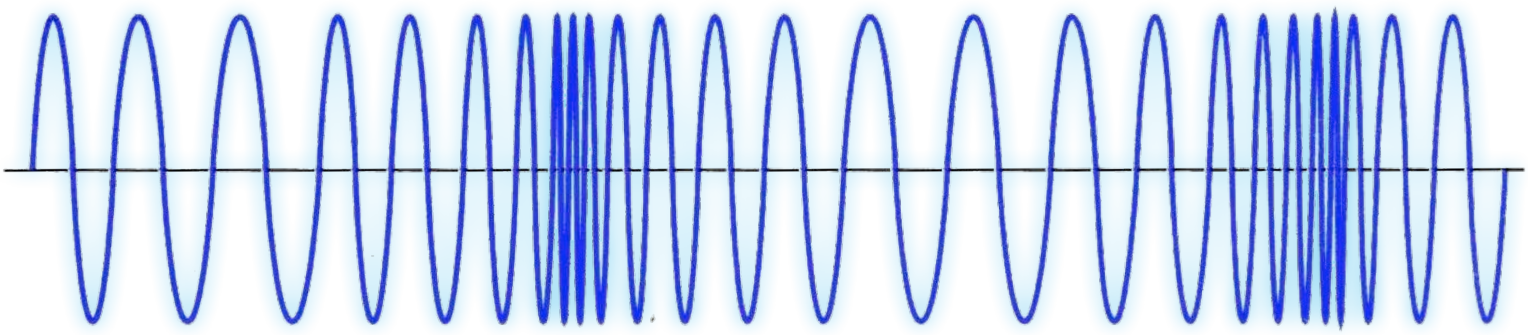
For datasheets and further product details:
www.elektormagazine.com/elektor-board-games-2018

Sinusoidal FM LF Amplifier

Analogue is not dead

Hans-Norbert Gerbig

Instead of converting a low frequency analogue signal to pulsewidth modulation (PWM) try using the signal to modulate a high-frequency oscillator. The circuit is simple and the signal quality benefits are impressive.



Digits, digits everywhere... nowadays it seems as though you can't get far when processing a signal level produced by any continuously changing physical process without sooner or later, converting it into its digital equivalent value.

One of the simplest and most popular methods of digitising an analogue signal is to convert it to a pulse-duration or pulse-width modulated (PWM) signal. In this case the value of a low-frequency analogue signal is used to define the on/off ratio of a constant frequency digital waveform. The amplitude of the digital waveform typically swings between 10% and 90% of the supply voltage.
Modern methods are not necessarily better than more traditional methods. Sinusoidal frequency modulation offers many benefits compared to PWM techniques. VHF and FM systems generally have stood the test of time. The beauty of this modulation method is that it allows impulse and other AM-based interference sources to be simply ignored so that they do not interfere with wanted FM signals.

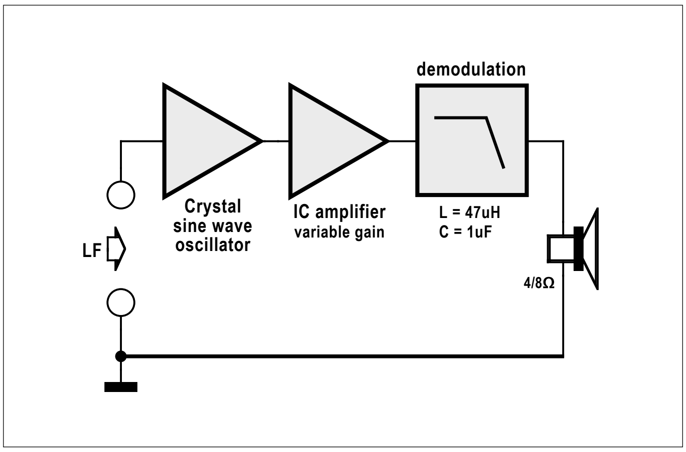


Figure 1. Sinusoidal self-oscillating FM amplifier.

Capacitive three-point oscillator

My own personal opinion is that VHF frequency modulation (FM) is one of the most important inventions in the field of electronics! Find out for yourself and probe a bit deeper by building the really simple experimental test setup shown in **Figure 1**. The sinewave generator is built using a capacitive 'three-point oscillator' circuit. The circuit of the self-oscillating sinewave generator (**Figure 2**) is built around a single transistor which produces a 'pullable' oscillator. This allows an externally applied voltage to modulate the crystal operating frequency. The 10-kΩ resistor between the base and collector of the transistor adds some damping to de-tune the oscillator network which has the effect of widening its pull-in range.

Some notable properties of this design include:

- The output signal level is very constant. The circuit attenuates any AM components in the signal by 60 dB. Superimposed interference spikes are effectively suppressed.
- When the input signal level is too low it has no affect on the oscillator frequency. No modulation occurs and the signal (i.e. the station) remains quiet.
- The oscillator is always pulled by the signal from the strongest (transmit) signal.
- Introducing more damping in the oscillator network has the effect of widening the circuit's pull-in range.

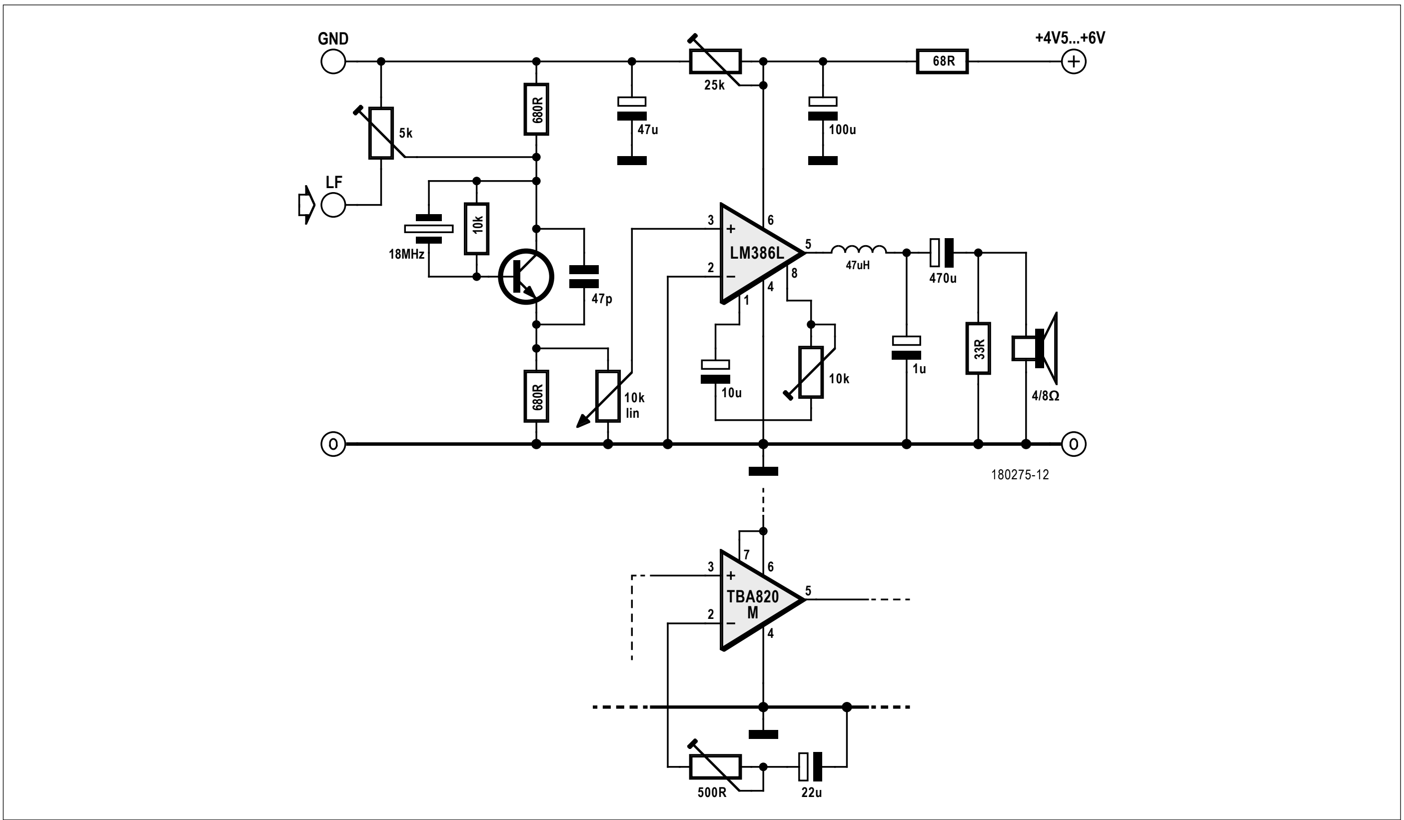


Figure 2. A frequency-modulated LF amplifier using an LM386 or TBA820M.

Other (receive) signals at the same frequency but of lower amplitude will have no effect on oscillator frequency. They therefore have no influence on demodulation and will be completely attenuated; even when the received field strength is only 30% below the desired (transmit) frequency. The result of these characteristics is an exceptional selectivity and the unmistakeable ‘VHF sound’.

Sinusoidal FM-LF amplifier

At the heart of this experimental circuit is a self-oscillating capacitive ‘three-point oscillator’ which, thanks to the quartz crystal, produces a stable 18-MHz constant amplitude sinewave (a higher frequency crystal could also be used). A 10.7-MHz crystal is not suitable because this frequency is usually in the first IF stage of VHF receivers and will produce interference in FM reception. The input audio signal is applied at the collector of the transistor and influences or modulates the oscillation frequency. This method of modulation exhibits good immunity to interference signals. Any unwanted AM interference is effectively suppressed because the oscillator circuit is only sensitive to input frequency changes and not signal amplitude changes. A feature of this design configuration is that the AF input is not galvanically isolated from the transistor collector (i.e. DC coupled without a capacitor) the same is true of the transistor emitter and the opamp input via the volume control. A standard audio frequency opamp IC is used here to boost the signal. You can use an LM386 or the slightly more powerful TBA820M. These two ICs require a slightly different circuit layout. I have therefore included two different layouts so you can build the circuit using perfboard (generated using the Lochmaster strip board designer tool) and also PCB layouts (using Sprint-Layout) which are all included in the download bundle [1] for this project.

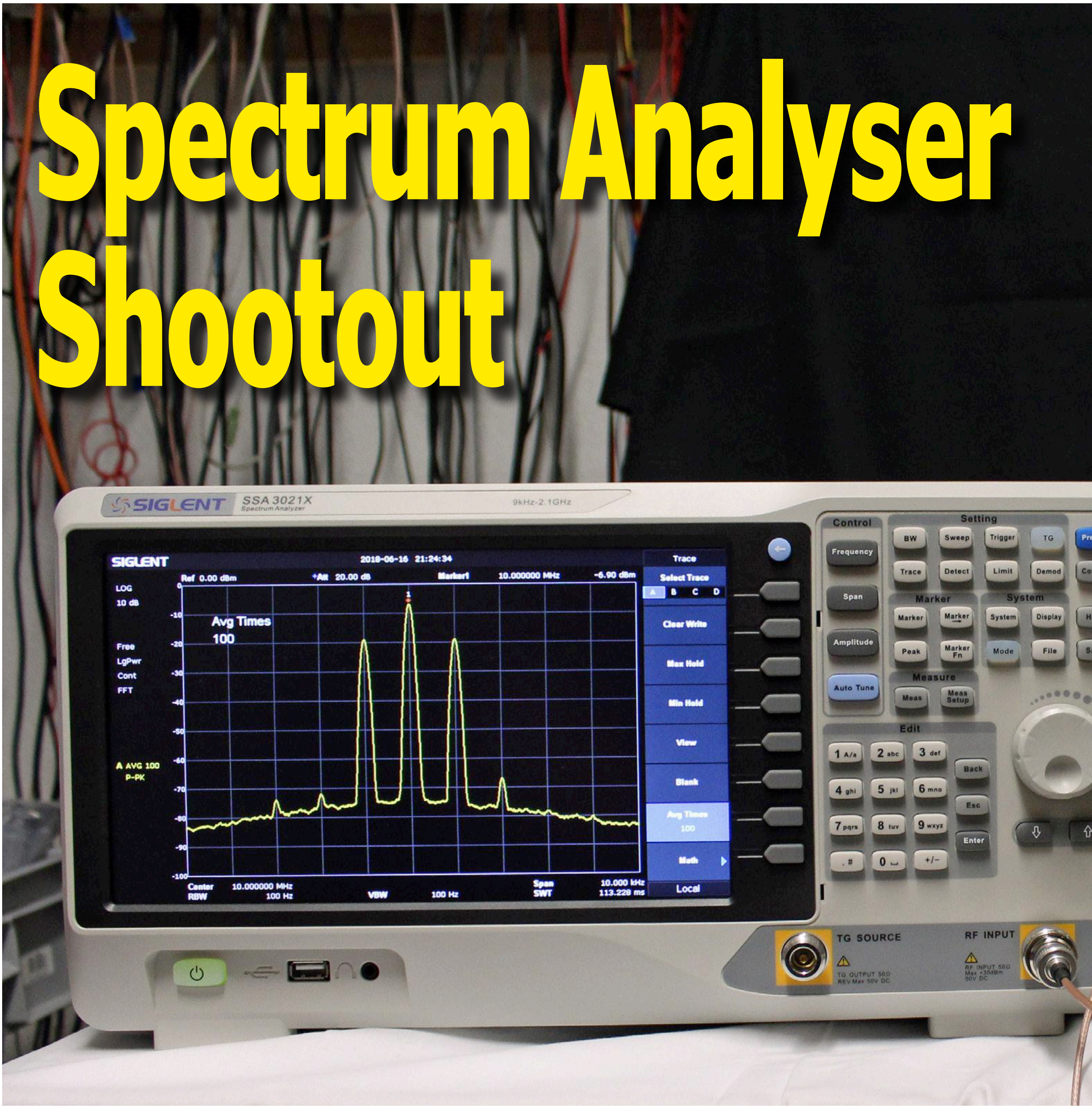
The frequency modulated sine wave signal output from the LM386 or TBA820M amplifier must now be demodulated further. The integrator network built from a low-pass filter consisting of L = 47 μH and C = 1 μF performs this function. It produces an output which is effectively the mean value of the frequency shift sequence. The 470-μF coupling capacitor serves to remove any DC voltage offset on the output signal which drives the loudspeaker. The low-pass filter also suppresses any residual high frequency components in the output signal. ◂

180275-03

Links and literature

[1] Project support web page: www.elektormagazine.com/180275-01

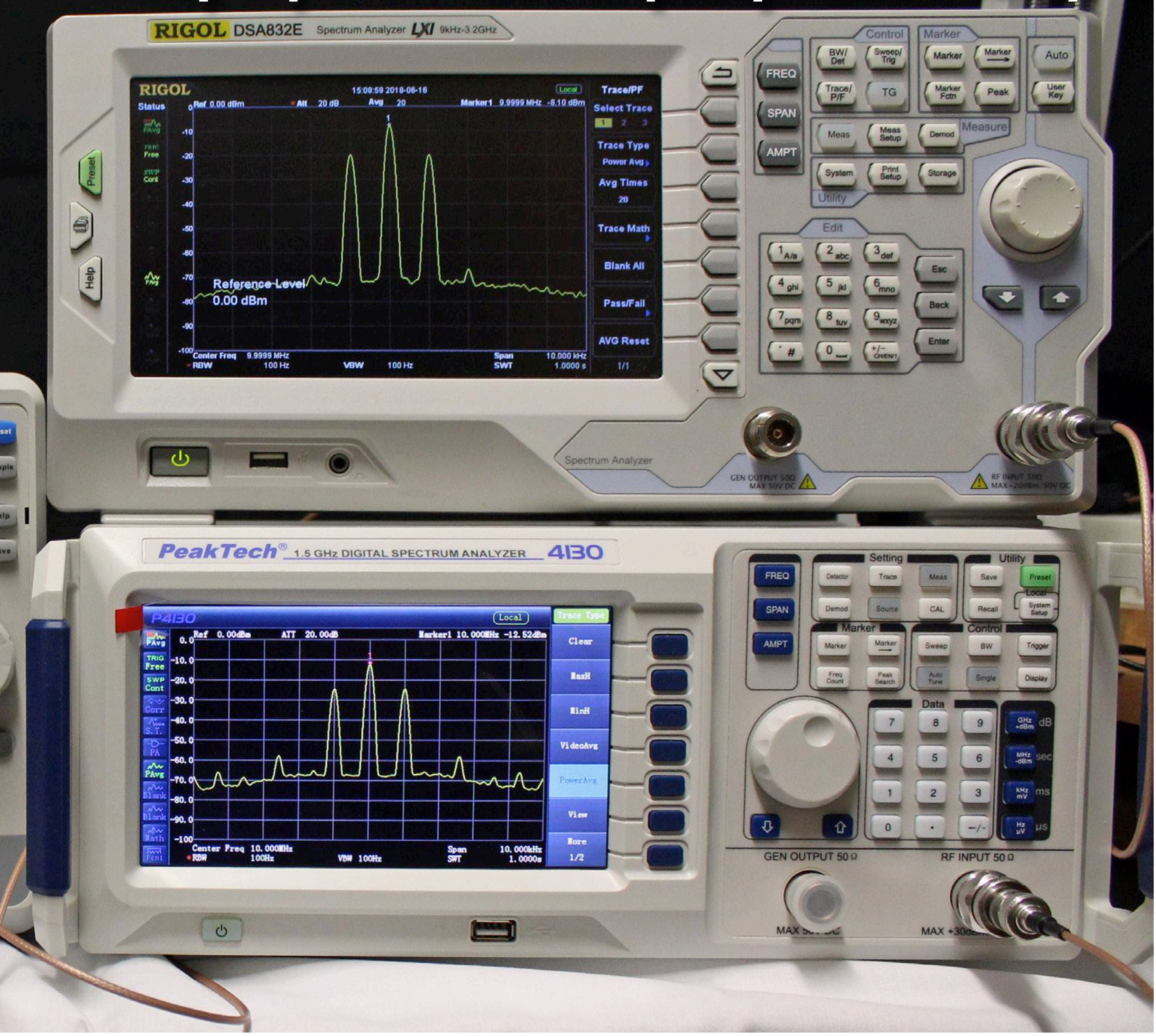
Spectrum Analyser Shootout



By **Alfred Rosenkränzer** (Germany)

In the November & December 2017 issue of Elektor Magazine I examined the FFT function on modern digital oscilloscopes. Normally provided at no extra cost, this bonus feature is definitely useful for analysing signals but it also has its limits. Most electronicists who are used to working with a ‘real’ spectrum analyser will soon find these shortcomings a hindrance. In times gone by, dedicated analysers were prohibitively expensive, not only for a hobby lab but also for smaller firms, even if bought second-hand. Fortunately, those days are over and nowadays some pretty good devices can be had new at acceptable prices. For this reason, a comparison test is worthwhile.

Plenty of performance for pretty modest outlay



Basically, there are (just as with modern oscilloscopes) a number of standalone devices equipped with screens and proper push buttons, as well as USB devices that can be operated only in conjunction with a PC or laptop. This article deals exclusively with standalone devices. First, however, we need to clarify what constitutes a spectrum analyser and which aspects determine its quality.

Frequency range

An oscilloscope enables us to visualise a signal being measured as a voltage over a period of time. Contrastingly, a spectrum analyser shows the amplitude of a signal across the frequency domain. Crucial to this is the upper cutoff frequency, up to which signal components can still be displayed. If, for example, we wished to display the first and second harmonics of a signal, then the signal frequency should occupy no more than one third of the analyser's total bandwidth. In fact, bandwidth is the variable that has the greatest influence on the price you pay — just as with oscilloscopes. In nearly all cases you cannot upgrade a device at a later stage to increase the bandwidth. So, before you buy, you need to consider precisely which signals you wish to measure.

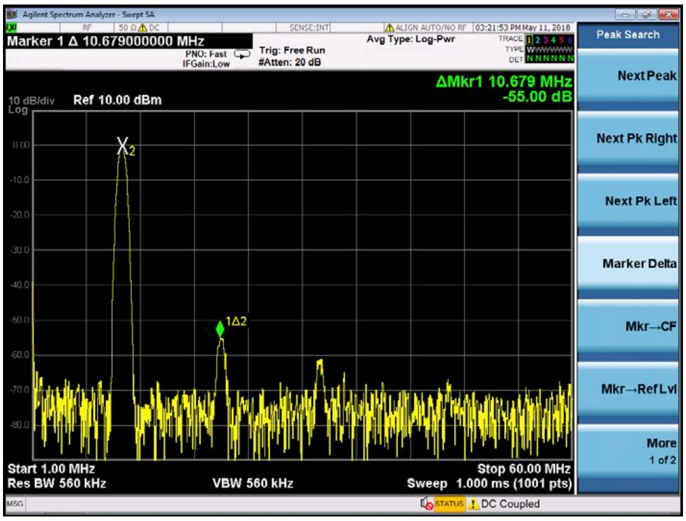


Figure 1. Spectrum of a 10.8-MHz signal without averaging.

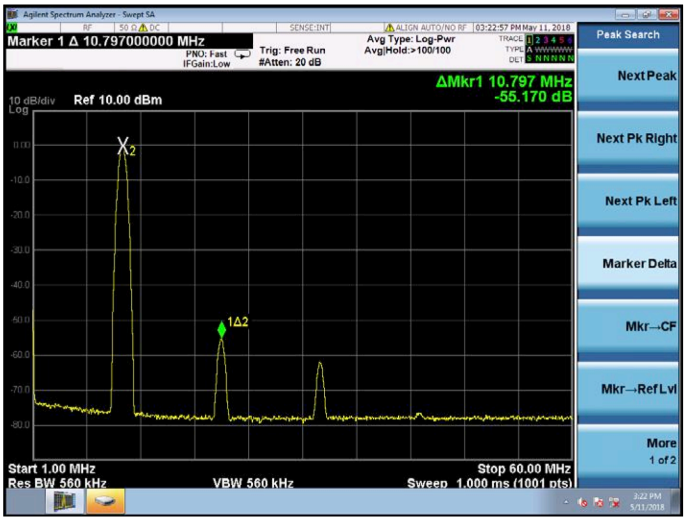


Figure 2. The same signal with an averaging factor of 100.

Attention should be paid to the lower cutoff frequency too. As a rule, the newer devices process signals from about 10 kHz upwards, although there are also some analysers whose spectrum starts only at 10 MHz. In the latter case, all signals below 10 MHz are ignored.
The frequency range to be analysed is usually defined by setting start and stop frequencies or, if appropriate, the centre frequency and span (frequency range).

Resolution

The so-called RBW (resolution bandwidth) determines the bandwidth of the bandpass filter used. If the usual automatic mode is switched on, this is normally set as a percentage of the span. However, you can set this manually at any time. The bandwidth of the bandpass filter is usually in a range starting from a few Hz up to a maximum of 1 MHz. The gradation also needs to be considered: inexpensive devices often offer a two-fold gradation per decade (i.e. ×1, ×3, ×10 and so on), whereas more expensive analysers provide significantly finer gradations. In this connection it's important to bear in mind the following.

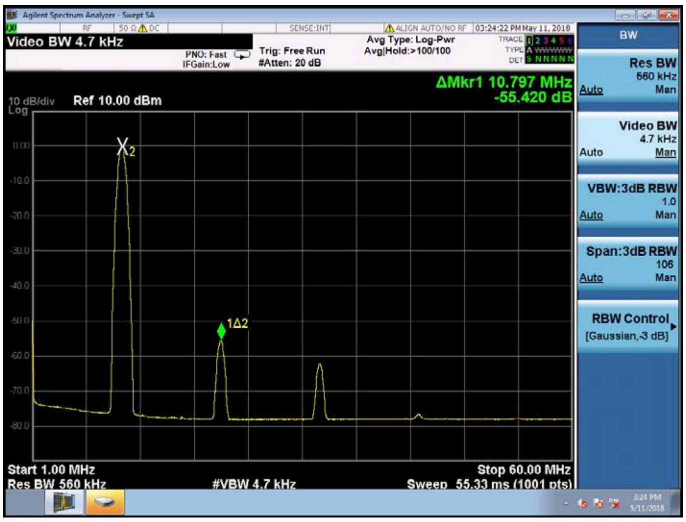


Figure 3. In comparison with the signal in Figure 2, the VBW has been reduced here from 560 kHz to 4.7 kHz.

A narrow-band filter takes more time to resonate (settle), in turn extending the time taken to sweep the span. Typically, the sweep time is adjusted automatically, to minimise the risk of incorrect measurements. With a narrower bandpass, the displayed noise level is reduced, enabling you to extract even weak (low level) signal components from the noise. By reducing the span and retaining the same sweep time, you can use a narrower filter to raise the signal-to-noise (S/N) ratio. The rule of thumb is that for reliable measurements, the S/N margin should be ≥10 dB. If the RBW is too large and signal components are tightly spaced, only a single peak will be seen, which should be avoided as far as possible in most situations. **Figure 1** shows the typical spectrum of a 10.8-MHz signal.

Averaging

Another way to reduce the noise displayed is the *Average* function. This simply takes and displays the average of several measurements. The statistically distributed and uncorrelated noise is reduced by a factor of $1 / \sqrt{n}$ (n = number of measurements). With the aid of the average factor n you can choose a compromise between noise amplitude and the time taken to measure. **Figure 2** shows the effect of an average factor of 100, which is very impressive with regard to noise.

Smoothing

With the aid of the VBW (video bandwidth) *Smoothing* function, rapid alterations of the curve shape can be levelled and balanced out. The result is a particularly smooth spectral profile, with very few spikes or jagged edges. In actual fact, it reduces the bandwidth of the spectrum to be displayed. **Figure 3** demonstrates how this achieves an exemplary spectral representation. There is of course a trade-off, because the processor in the analyser has to work harder and consequently the display speed and sweep frequency suffer a little.

Additional functions

Usually several so-called *detector* types are provided, such as Peak (positive and negative), RMS, Sample, Normal and so on. These settings affect the representation or weighting of the spectrum and are referenced to the signal that actually represents a display point.

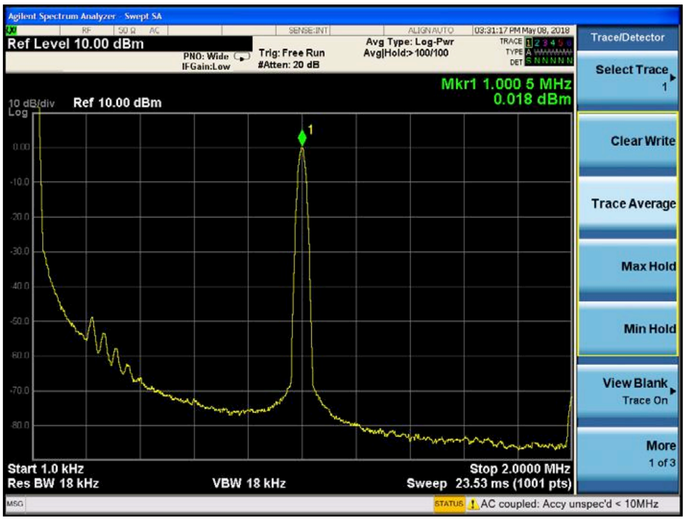


Figure 4. 1-MHz signal using AC coupling.

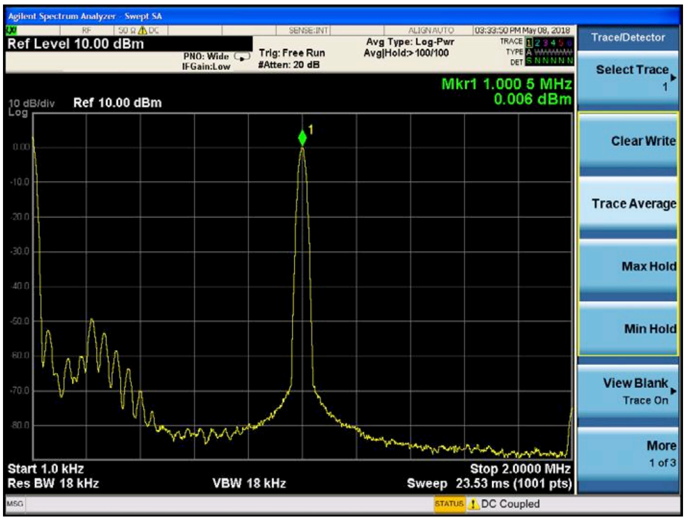


Figure 5. 1-MHz signal using DC coupling.

Input and level

The maximum level refers to the highest permissible amplitude that is allowed to enter the device input without risk of damage. If the signal to be measured is greater than this maximum, it is recommended to connect an external attenuator of appropriate rating. These can be had with N, BNC and SMA-type connectors. The switchable internal attenuator adjusts the input of the device to the signal level. In the past, a 10-dB step increment was common — today, 2-dB steps are more common. If the input signal is too large (insufficient attenuation), the input stage is overdriven and consequently can even generate harmonics that did not exist in the original signal. Incorrect measurements are the result. Keep an eye on the peak values of the input signal, which with high crest factors [1] can be significantly higher than the rms value. If too much attenuation is applied, small signals will be lost in the noise, because the amplifier stages in an analyser have only a finite S/N margin. Fine gradations make for an optimal compromise between distortion and noise, and are therefore not a needless luxury.

The frequency response of the analyser electronics should be as even as possible. This is important, because as far as possible, the displayed spectrum should not contain any frequency fluctuations that are not present in the original signal.

With small signals, such as those delivered by an antenna, it helps if you can switch in the (hopefully built-in) preamplifier. The input is generally hard-wired for AC coupling. Despite this, you should never connect any signal greater than the permissible DC voltage. The effect of charging capacitors while taking low-frequency measurement is not attractive (**Figure 4**). If the input can be switched to DC, you can prevent this (**Figure 5**). Nevertheless, caution should still be taken with regard to the voltage stability, so as not to destroy the input stage accidentally. An external DC block with a larger capacitor is a good compromise, but costs extra money.

Markers

Using the *Marker* functions, you can measure both absolute signal levels as well as the difference between two signals. Functions such as *Peak Search* will assist you to position the marker quickly and accurately. Then you can also determine the frequency obtained at these locations accurately.

Marker functions like *Noise Density* enable you to allow capture relevant amplifier parameters such as the noise density alone in dBm / Hz or µV / √Hz. In this case the internal noise of the analyser must be at least around 10 dB less than that of the signal to be measured.

The *Noise Power* function in an adjustable frequency range is useful to determine values such as SNR (signal to noise ratio). With large signals you carry this out in two steps. In the first step, the level of the signal is measured, then subsequently you do this without any signal applied (and with possibly less attenuation) so as to capture the noise alone.

Demodulation

If functions of this kind are provided, you can use these to demodulate amplitude and frequency modulated signals direct. Via an audio output you could even hear radio signals. Even more interesting is demodulating digital transmissions using modern protocols such as QAM16 [2] and so on. Expect to find features like this only on higher-priced models; a surcharge is often applied, even on expensive equipment.

Tracking generator

At extra charge, some devices offer the luxury of an output to show the amplitude curve of a filter or amplifier (**Figure 6**). If

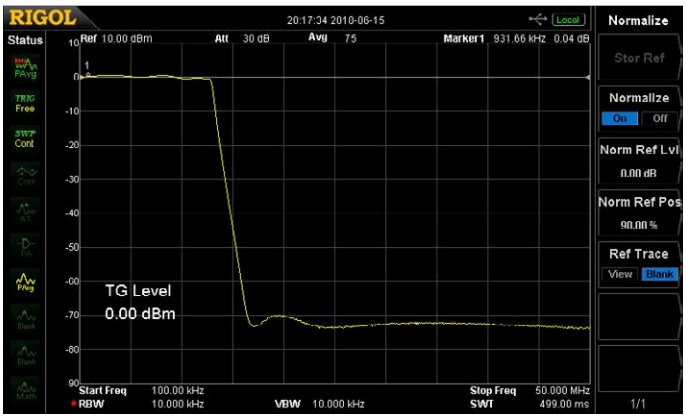


Figure 6. Frequency response of a lowpass filter with 10 dB per division.

Table 1. Technical Data

Manufacturer	Siglent	Rigol	Peaktech	Keysight (benchmark reference)
Model	SAA3021x	DSA832	4130	N9020A 3,6 GHz
Max. frequency (GHz)	2.1	3.2	1.5	3.6
Min. frequency (kHz)	9	9	9	10 Hz (DC), 10 MHz (AC)
Min. resolution (Hz)	1	1	1	2
Max. RBW (MHz)	1	1	1	3
Min. RBW (Hz)	10	10	10	1
Gradation (dB)	1/3/10	1/3/10	1/3/10	10%
Max. VBW (MHz)	3	3	3	3
Min. VBW (MHz)	10	10	10	1
Gradation (dB)	1/3/10	1/3/10	1/3/10	10%
Max. attenuation (dB)	51	30	50	70
Step increment / dB	1	1	5 (1)	10/2
Max. input voltage (V)	±50	±50	±50	0.2 (DC), 100 (AC)
Max. RF level (dBm)	33	30	30	30
Preamplifier (dB)	20	17	20	20
Phase noise (dBc/Hz)				
At 10 kHz offset	–95	–98	–80	–113
At 100 kHz offset	–96	–100	–	–116
At 1 MHz offset	–115	–	–	–135
Test signal harmonics (dBc)				
First harmonic	–74	–86 (–30 dB)	–87 (–25 dB)	–89
Second harmonic	–92	–102 (–24 dB)	–96 (–25 dB)	–105
Noise density at 100 MHz (dBm/Hz)				
Without preamplifier (data sheet)	–137/–141	–127	–120/–125	–151/–154

you do not have a network analyser, this feature is very useful and usually worth the extra cost. All the same, you can measure only the amplitude curve, not the group delay. With the Siglent analyser, the tracking generator is always built in — but activated only by entering a code supplied following payment. For other brands, this function can sometimes be retrofitted to hardware later. If neither is the case, you must make an informed decision before purchasing.

Display
Test results, settings and a software menu are shown on an LCD screen. As with all other displays, size, resolution, depth of colour palette, viewing angle and so on are the determining factors for clear and fatigue-free viewing.

Some spectrum analysers offer an additional video output (connection for an external display), which enables live presentations of test results to be made by video projector or large-screen TV.

Operation
Modern analysers are operated using a mixture of push buttons, rotary knobs and softkeys. The main settings should be made as directly as possible, but seldom-used functions can be concealed in sub-menus of the softkeys. Frequently you discover how well the operating scheme was thought through only after using it for a fairly long time. As a rule, direct comparisons are hardly an option for end users, because you hardly ever see the devices in question directly alongside each other and een then you won't have enough time to test them thoroughly.

Connections
Signal input and time generator outputs are normally equipped with N-type connectors. Patch cables fitted with these are not widely available, so when planning your purchase, your shopping list should include BNC or SMA adapters. These also protect the connectors on the device, as then it's only the adapters that suffer the wear and tear. For this reason, it makes very

Manufacturer	Siglent	Rigol	Peaktech	Keysight (benchmark reference)
Without preamplifier (measured)	−145	−148	−128	−153
With preamplifier (Data sheet)	−156/−161	−148	−135/−140	−163/−166
With preamplifier (measured)	−164	−168	−147	−165
Reference frequency				
Frequency (MHz)	9.9998883	–	9.9998878	–
Curve shape	Sinewave	Sinewave	Sawtooth/ squarewave	Sinewave
Level @50 Ω / mV_{pp}(mV RMS)	800/285	1.290/455	496/185	–
Tracking generator				
Max. frequency (GHz)	2.1	3.2	2.2	–
Min. frequency (kHz)	100	100	9	–
Max. level (dBm)	0	0	0	–
Min. level (dBm)	−20	−20	−20	–
Step increment / dB	1	1	1	–
Display				
Size (inches)	10.1	8.0	7.0	8.4
Resolution	1024 x 600	800 x 480	800 x 480	1024 x 768
Curve area (horizontal)	751	601	601	
Interfaces				
USB 2.0	A + B	A + B	A + B	A
Ethernet (Mbit/s)	100	100	100	–
Miscellaneous		GPIB (option)	RS−232	GPIB (option)
Dimensions (B × H × T) (mm)	393 × 207 × 116	361 × 178 × 128	363 × 154 × 327	368 × 177 × 426
Weight (kg)	4.6	5.15	6	18
Power consumption (W)	30	35/50	35	465

good sense to insist on top quality adapters, for as you know, no chain is better than its weakest link.

Additional connectors in the form of BNC sockets will be found on the rear of the devices. Particularly relevant are inputs and outputs for a frequency reference. These enable several devices to be hooked up to a 10-MHz signal. It goes without saying that the device with the highest stability or quality of the reference should be the 'master'. Often there is a USB socket on the front, which you can use to save your results and device settings on a USB stick.

For controlling equipment remotely, the professional (and expensive) GPIB bus was formerly standard. Today, more commonplace and inexpensive interfaces such as USB and Ethernet are used, or very rarely RS-232.

Quality

The mother of all questions is this: which quality standard should a spectrum analyser reach (for you personally) and what should you look for? The following aspects include some subjective considerations, because they depend on your own requirements. Since I do not know your particular applications, you yourself must weigh up the criteria that are relevant to you. For me the top priority has to be the harmonic performance. What I mean by that is which harmonics the analyser (incorrectly) indicates and at which level, even with a high-purity sinewave signal present at the input. To determine this, all devices were tested with a 10.8-MHz sinewave signal at a level of 0 dBm (**Figure 7**). Since the sine waveform of a generator is never good enough for this purpose, the test signal was first 'cleaned up' with an elaborate bandpass filter to produce a very good sinewave profile.

After checking the amplitude of the fundamental wave, the centre frequency was set to double the fundamental and the span to 100 kHz. The marker is set to the first harmonic using *Peak Search* (**Figure 8**) and the attenuation increased until the level of the first harmonic no longer alters (**Figure 9**). To

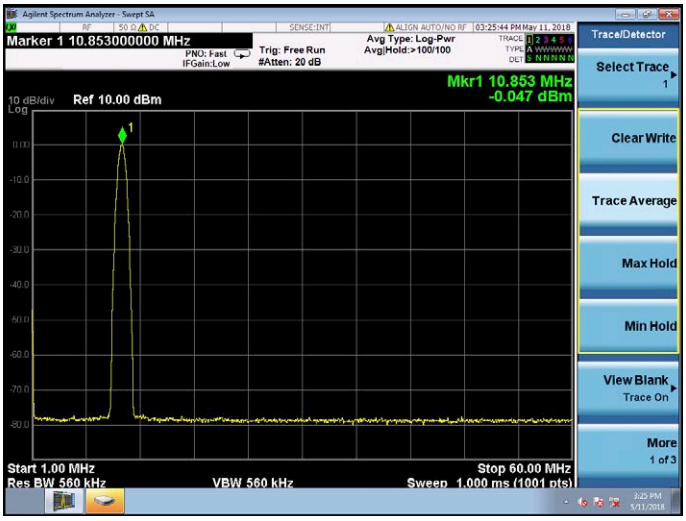


Figure 7. Test signal: 10.8-MHz sinewave at 0 dBm, filtered (first harmonic 135 dBc, second harmonic 125 dBc).

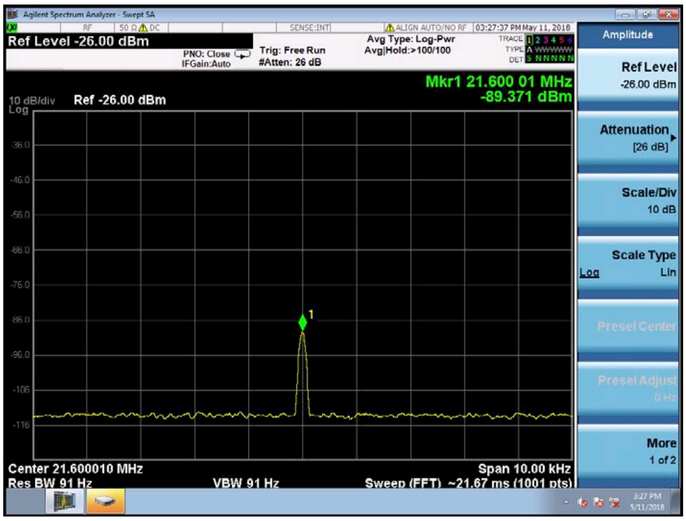


Figure 8. Level of the first harmonic measured using 20 dB attenuation: 83 dbc.

achieve greater headroom over the noise, the span can then be reduced. The second harmonic (= three times the signal frequency) is measured in the same way.

The second criterion for me is the internal noise of the device. To establish this, the Noise Density was measured at 100 MHz,with and without the preamplifier, and noted in **Table 1, Technical Data.**

Peaktech 4130

After powering up, the Peak Analyzer (**Figure 10**) reports for duty with noticeable fan noise. Although the menu structure struck me as somewhat unconventional, operation was simple and straightforward.

What stands out is that the attenuator can be adjusted only in 5-dB steps, using the arrow keys and the rotary knob. You can also use 1-dB steps via the keyboard. Unfortunately, the marker cannot be controlled by keyboard.

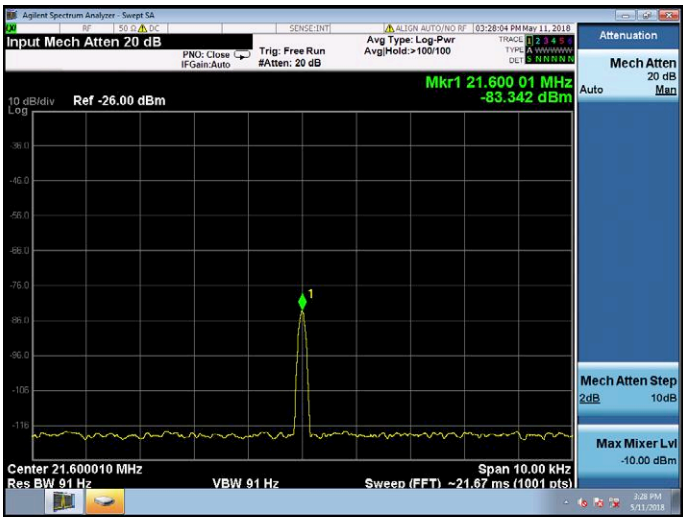


Figure 9. In comparison with Figure 8, using 26 dB attenuation reduces the level to 89 dbc.

The preamplifier can be used only from 100 MHz upwards — a significant limitation.

You can prevent fluctuations in the displayed level by performing a 'cal' after warming up. This is also recommended in the manual.

Unfortunately, it is not possible to save a screenshot on a USB stick. The company's technical support confirmed this to me when I asked.

The displayed harmonics of the test signal (sinewave at 10.8 MHz, 0 dBm) are quite low. However, the noise density at 100 MHz is unfortunately the worst of the three devices tested. The signal of the 10-MHz reference output has a strange curve shape. It's not a sinewave as on the other two devices.

Siglent SSA3021X

Once again, the cooling fan is audible when using this spectrum analyser. Its front elevation is dominated by the quite large 10.1-inch display. Despite that, there is enough space for adequately large control buttons (see **Figure 11**). The configuration of the menu is logical and the controls are intuitive, so long as you are familiar with using a spectrum analyser.

The measured first harmonic is at a relatively high level, the second harmonic being reasonable. The noise density is quite good. The levels measured at various RWB and VBW settings are very constant.

The preamplifier is usable even at low frequencies. Screenshots can be saved to a USB stick simply and conveniently.

Rigol DSA832E

Once again, the device will not operate without the fan running. The display is slightly smaller than the Siglent, but larger than that of the Peaktech (see **Figure 12**). Operation and menu structure are also logical and straightforward.

The harmonics and noise densities are the best in the trio, even better than my somewhat dated reference model (Keysight N9020A).

Negative points are the long time that the device takes to save a screenshot and how the *Peak Search* function fails with unclear signals.

About the author

Alfred Rosenkränzer has worked for more than 30 years as a development engineer, initially in the field of professional television technology. Since the end of the 1990s he has developed digital high-speed and analog circuitry for IC testers.

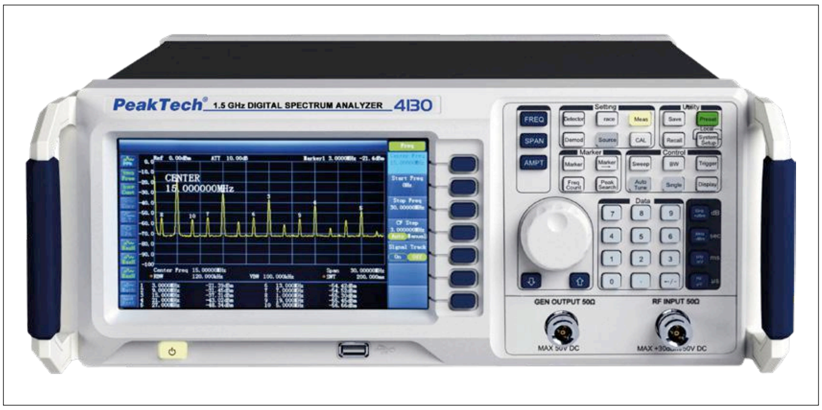


Figure 10. Peaktech 4130: the menu arrangements are a bit unconventional but remain easy to use.

The tester showed a small peak at 800 kHz without any signal connected, appearing to arise from the electronics of the device itself.
Comparing the rear panel with that of the Siglent analyser, the well-nigh identical arrangement of the connections is striking.

Figure 11. Siglent SSA3021X: this analyser has the advantage of an impressively big 10.1-inch display.

My verdict

I find it amazing how much virtually professional-grade test and measurement equipment you can buy today for relatively little money. The choice between the two front-runners, Siglent and Rigol, is a tough one. The Siglent analyser has a better display and clear presentation. The alternative from Rigol scores with the best measurement readings – albeit at price tickets starting at around €2,250 / £2,000 / $2,640.
The model from Peaktech brings up the rear — on one hand because of the missing ability to save screenshots without hooking up a PC and on the other hand because the preamplifier is usable only above 100 MHz. Both shortcomings could be remedied easily by the manufacturer. The internal noise level is also relatively high. From an operational viewpoint there nothing negative to reveal; in this respect the difference between them is very minor. ◄

(180290-01)

Figure 12. Rigol DSA832E: highest price plus best measurement characteristics.

Weblinks

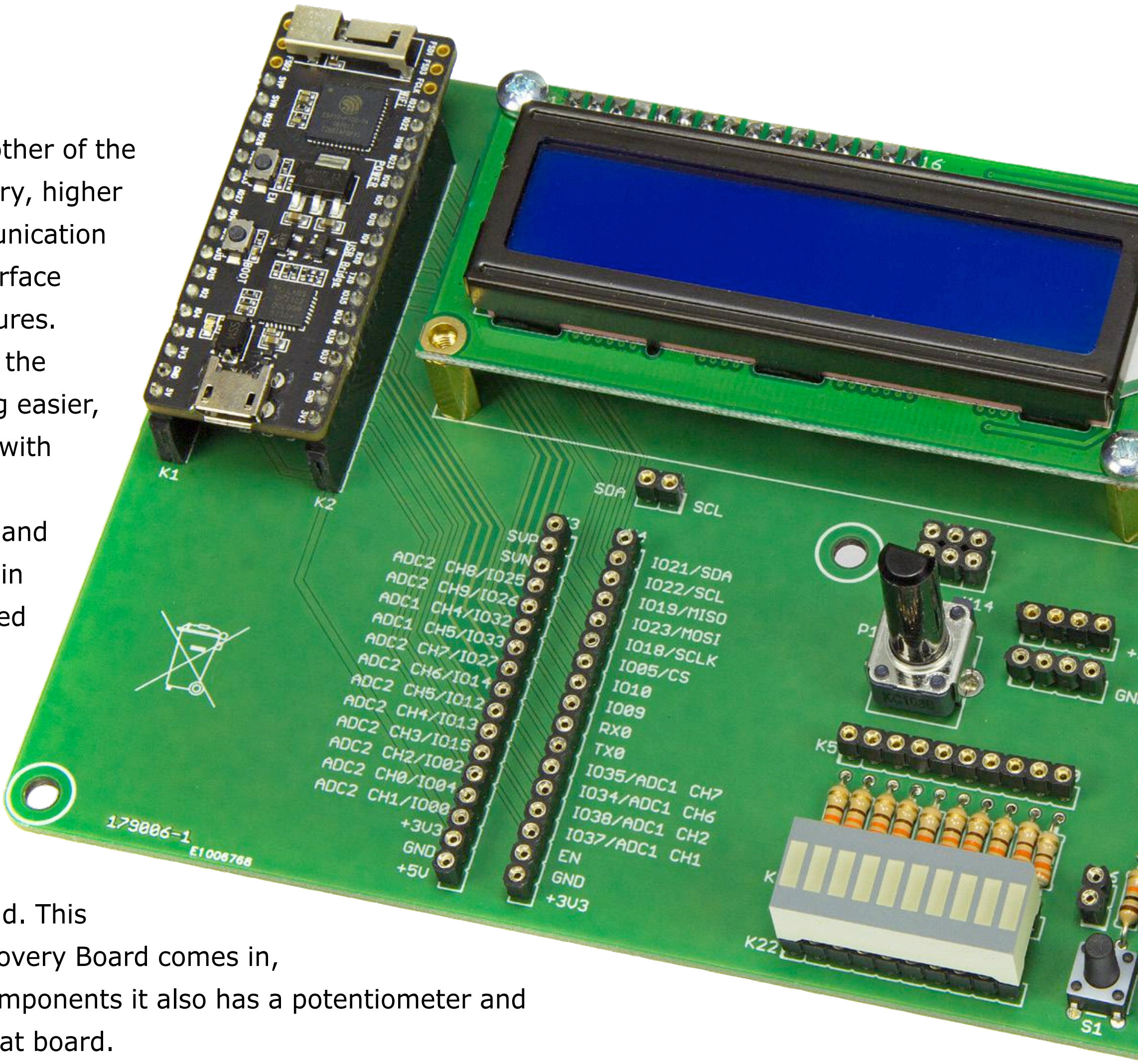
[1] Crest-Faktor: https://en.wikipedia.org/wiki/Crest_factor

[2] QAM: https://en.wikipedia.org/wiki/Quadrature_amplitude_modulation

The ESP32 Pico Discovery Board

Experimenting made simple

Erik Bartmann



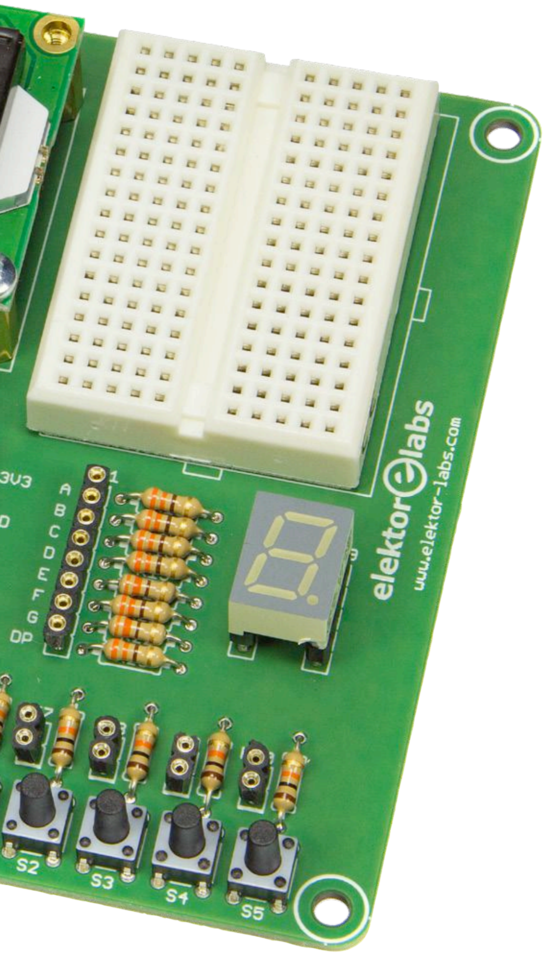
The ESP32 chip is the big brother of the ESP8266: more SRAM memory, higher CPU speed, Bluetooth communication and a greater number of interface pins are just some of its features. Development boards such as the ESP32 Pico Kit make handling easier, they come already equipped with all the essentials such as an antenna, a voltage regulator and a USB interface. The use of pin headers allows it to be plugged into a breadboard where an application circuit can be built around it. Some of the other necessary components such as pushbuttons, LEDs and a display are not always to hand. This is where the ESP32 Pico Discovery Board comes in, along with these essential components it also has a potentiometer and 7-segment display on one neat board.

The ESP32

Some of the key features of the ESP32 chip, more detailed information can be found in the technical reference [13].

- CPU: 32-Bit dual-core with two Harvard Architecture Xtensa LX6 cores
- Clock: up to 600 DMIPS
- External und internal memory
- Wi-fi: 2.4 GHz HT40
- Bluetooth: BLE 4.2 (Bluetooth-Low-Energy)
- Peripherals: SPI, I²C, I²S, UART, CAN 2.0 and Ethernet interface
- ADC: 12-Bit (Analog/Digital converter)
- Sensors: Touch, Hall and temperature
- PWM: 1 Hardware and 16 software PWM channels
- IO: GPIO pins (General Purpose Input Output)

The palette of hardware tools available for developing new applications and IoT functions is still dominated by Arduino and Raspberry Pi-based platforms. Newcomers appear but it is difficult for them to carve out a slice of the cake. Espressif Systems, a company based in Shanghai China is however making inroads into this market. They make microchips with built-support for communications protocols such as Wi-Fi and Bluetooth making them ideal for building IoT devices. For a number of years now the company has had raving success with their ESP8266 which provides a relatively pain-free way to add Wi-Fi capabilities to a project. Developer boards from various manufacturers soon appeared which helped made the chip easier to use. The first board, barely larger than a postage stamp was the ESP-01 it came equipped with an integrated antenna and provided connection to I/Os via a pin header. These boards are still very popular today. After the success of the ESP8266 Espressif were not content to sit back and blow smoke but soon introduced the ESP32 chip which added Bluetooth capability into the mix. Besides



Bluetooth this SoC chip has more SRAM, a faster processor clock, more interface pins (GPIOs), touch sensors, A/D and D/A converters (see box). This chip is now also featured in a whole raft of developer boards [1], most of which are a bit bigger than the ESP8266 versions. These boards offer additional facilities such as built-in USB-to-UART bridge, voltage regulator (LDO), power indicator LED and mini pushbuttons for reset and boot options. The board shown in **Figure 1** is a popular example, called the ESP32 Pico Board or ESP32 Pico Kit.

Cover all the bases

Most makers and homebrew circuit specialists these days will use large prototyping plug boards to fit and connect components without the need to fire up a soldering iron. Most development boards are fitted with rows of pinheaders so they easily fit into the plug board. For the beginner it may be something of a hurdle to get together all the peripherals and components to build a working system. For most designs you are likely to need basic components such as resistors, 7-segment displays, a potentiometer and pushbuttons. These are probably already lying around in a tray somewhere on your work bench or in amongst a box of components stashed under your desk. My idea was to build a neat universal development board containing an ESP32 Pico-board together with a range of pre-fitted components that would just about cover all the bases

PROJECT INFORMATION

ESP32 | ESP32 Pico Kit | Arduino | Dev-Board

➡ entry level
intermediate level
expert level

1 hour approx.

Soldering iron,
PC with Arduino-IDE,
patch leads

€30 / £25 / $40 approx.

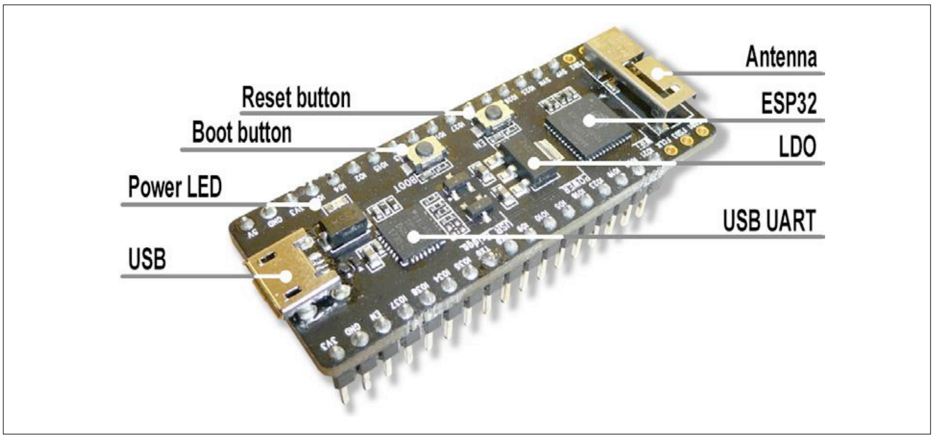


Figure 1. The ESP32 Pico-Board V4.

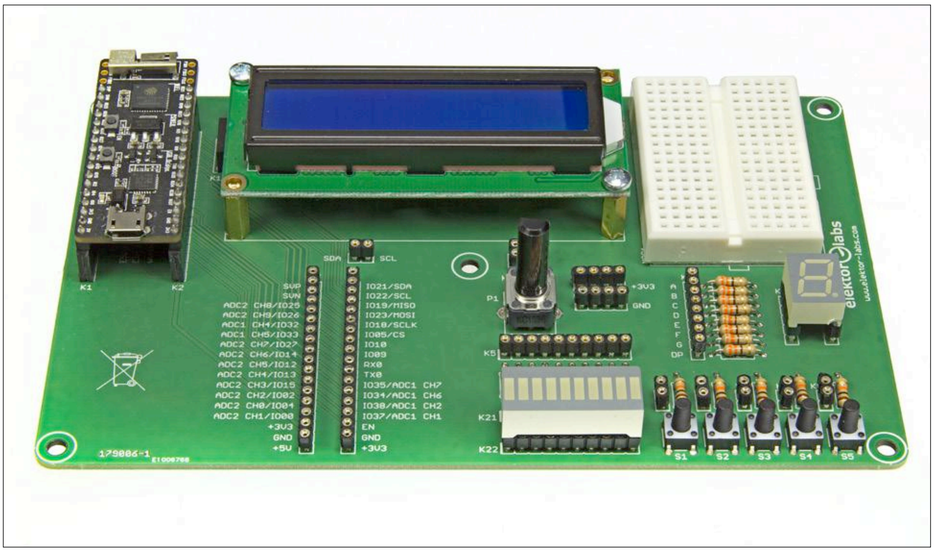


Figure 2. The ESP32 Pico Discovery Board.

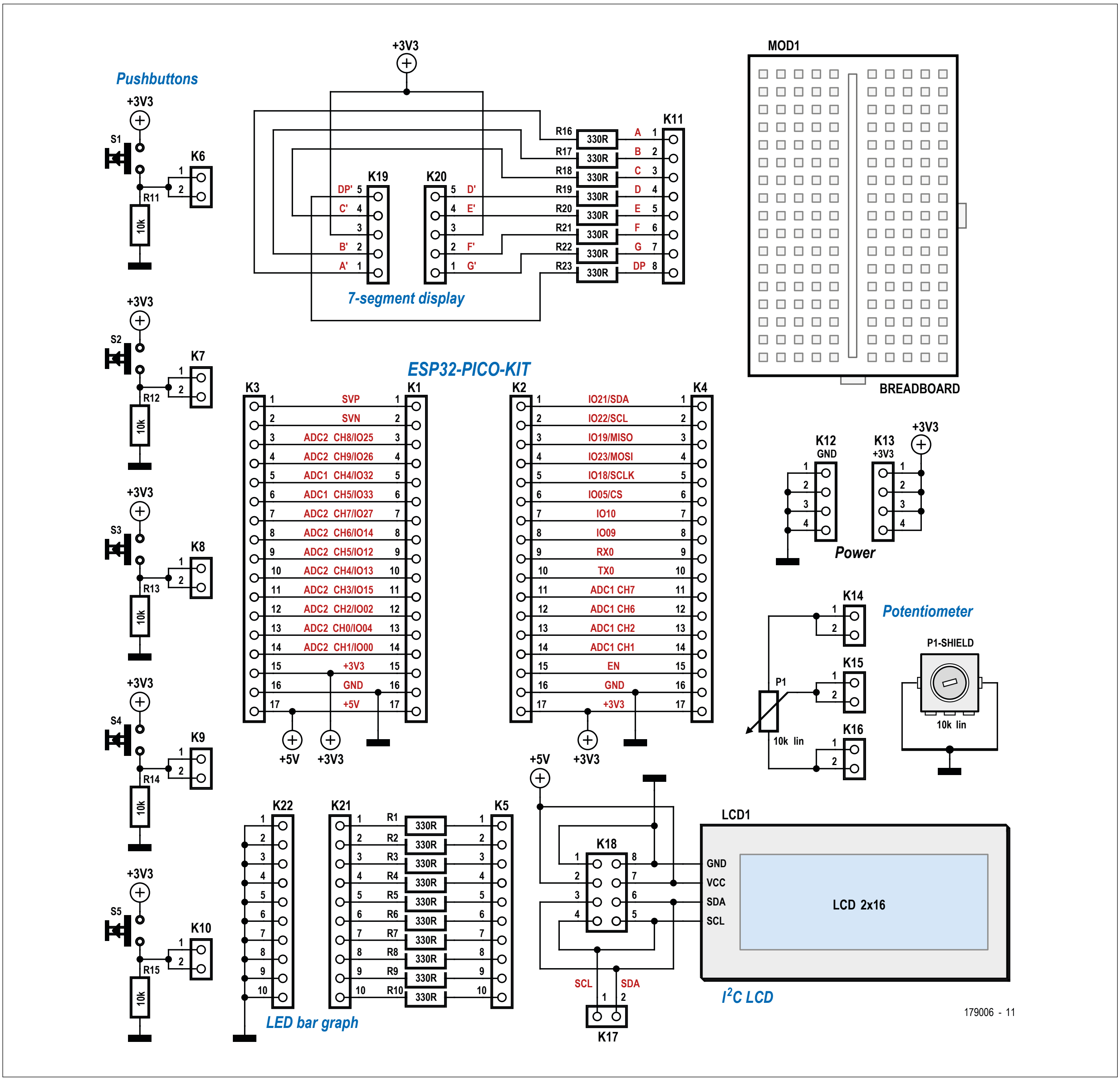


Figure 3. Schematic of the board. The pins of the ESP32 Pico board and the peripherals such as LEDs, buttons and display are routed to female connectors. In between can be freely wired.

in terms of basic system needs so that you can quickly and neatly configure a working system using jumper wires and be up and running on your project in no time. I have called this universal development board the *ESP32 Pico-Discovery board* (**Figure 2**).

The board includes all these components on one PCB:

- A socket to accept the ESP32 Pico-Board.
- Two rows of pin headers giving easy access to the pins of the ESP32 Pico-Board.
- 10 LEDs (bar graph) with series resistors and connection header pins.
- 5 micro pushbuttons with pull-down resistors and connection header pins.
- A potentiometer.
- A 7-segment LED display with series resistors and connection header pins.
- An LC display using an I²C-Bus interface with connection header pins.
- Power rails for 3.3 V and GND.
- A small prototyping plug board area, handy for adding components to build additional circuitry.

The circuit diagram of the board is shown in **Figure 3**. It is easy to see that the convenience of this board is that the connections between the controller and the peripherals, unlike most other development boards, are not hardwired. The user makes the connections with lengths of

COMPONENT LIST

Resistors

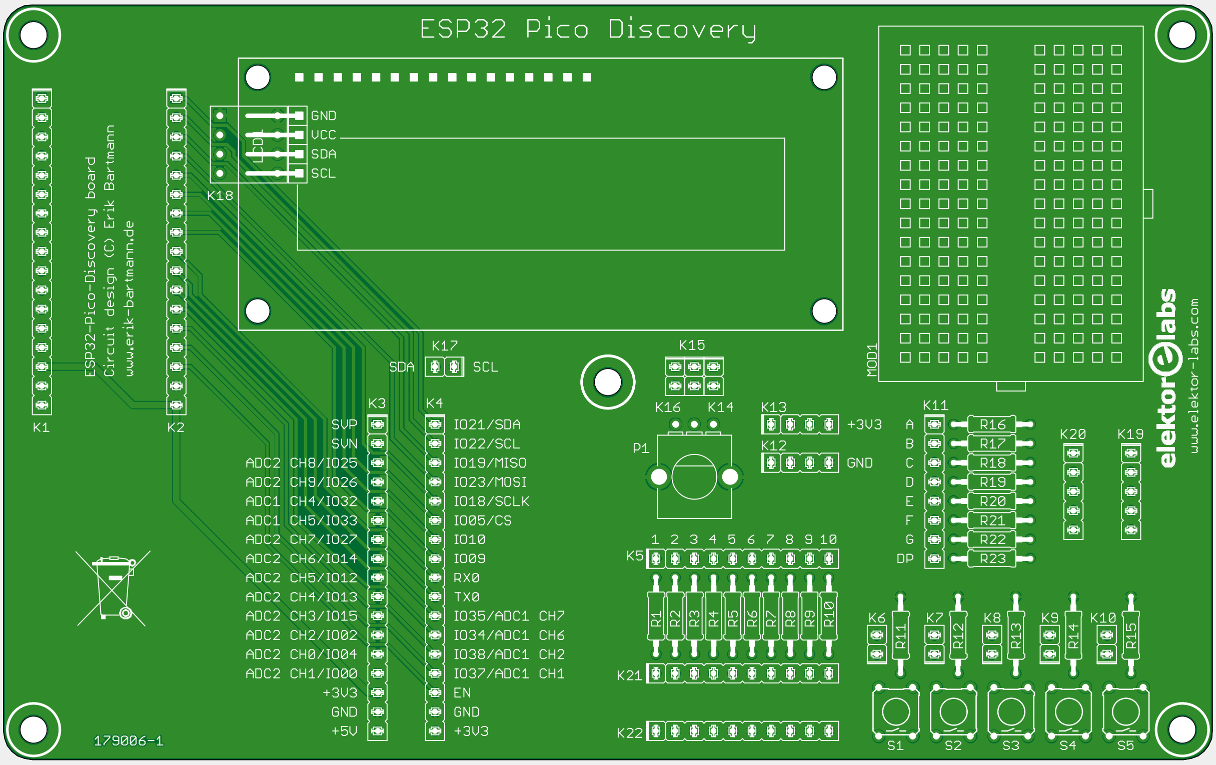
R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R16,R17,R18, R19,R20,R21,R22,R23 = 330Ω
R11,R12,R13,R14,R15 = 10kΩ
P1 = 10kΩ linear potentiometer

Semiconductors

SA39-11SYKWA (Kingbright) 7-segment display
DC-10GWA (Kingbright) LED-bargraph
LCD, alphanumerical, 2x16 characters, with I²C Interface

Miscellaneous

K6,K7,K8,K9,K10,K14,K15,K16,K17 = 2-way pinheader socket, 0.1" pitch
K12,K13 = 4-way SIL pinheader socket, 0.1" pitch
K11 = 8-way SIL pinheader socket, 0.1" pitch
K5 = 10-way SIL pinheader socket, 0.1" pitch
K1,K2,K3,K4 = 17-way SIL pinheader socket, 0.1" pitch
K18 = 8-way (2x4) pinheader socket, angled, 0.1" pitch
K19,K20 = 5-way SIL pinheader socket, 0.1" pitch
K21,K22 = 10-way SIL pinheader socket, 0.1" pitch, or 20-way IC socket
S1,S2,S3,S4,S5 = tactile-feedback pushbutton, 24V, 50mA, 6x6 mm
Breadboard 47x35 mm, 170 contacts (e.g. Kuongshun AA055)
Standoffs for LCD
ESP32 Pico-Board (www.elektor.com/esp32-pico-kit-v4)
PCB no. 179006-1 Rev 1.0



jumper wire — this degree of flexibility gives you almost unlimited development possibilities.

It is of course possible to build such a board from scratch yourself using a square of perf board but the finished result is unlikely to be as neat and usable as the *ESP32 Pico-Discovery-Board* .

Elektor hardware

Neat and simple is our motto and that is what you will get when you order the

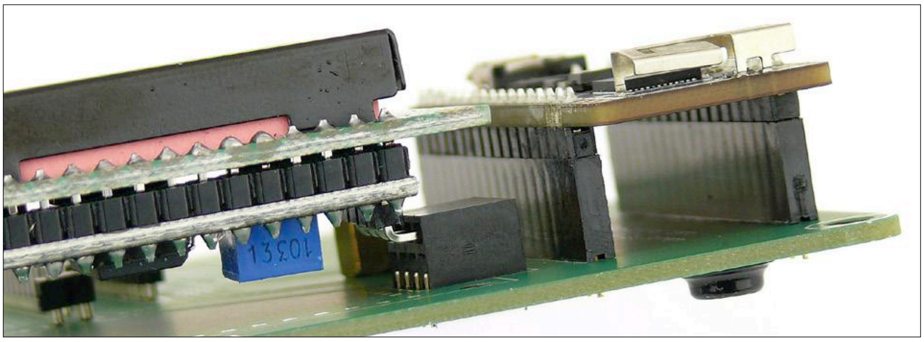


Figure 4. The I²C display (left) consists of the actual display unit and an I²C converter board, which is connected to the Discovery board via a 4-pin SIL header.

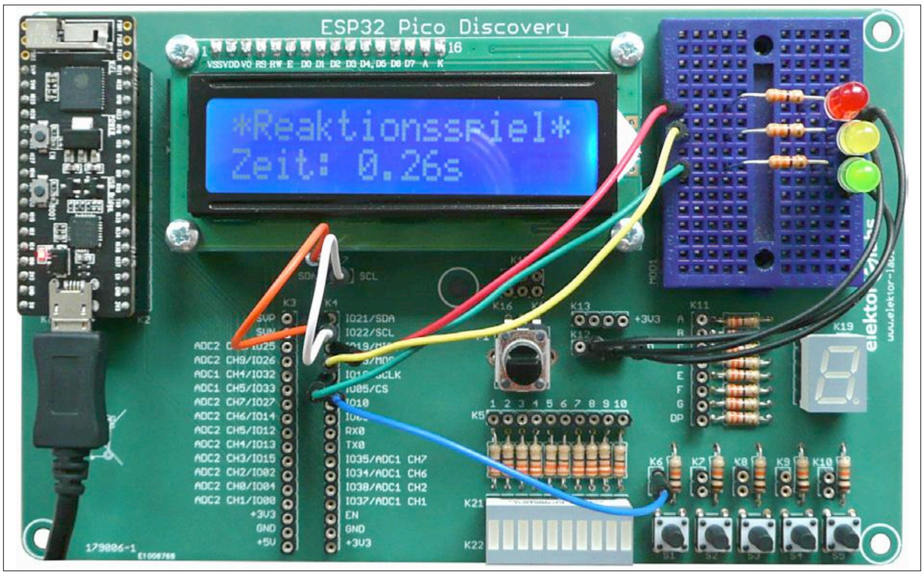


Figure 5. A Reaction Game on the ESP32 Pico Discovery Board.

discovery board kit from the Elektor Store. The kit contains the PCB and all the necessary components including an ESP32 Pico board (see box '**@ www.elektor.com**'). The PCB was designed here at Elektor together with help from the author of this project. We made sure the layout is easy to use and even paid attention to little details like rounded edges on the board to make handling more comfortable. Soldering the components in place will be good practice and should not pose any problem even for the absolute beginner.

Fitting the LCD: The kit includes a 2×16 character LC display which has a built-in adapter board allowing it to communicate via an I²C interface rather than the more usual parallel data transfer method. The adapter board connects to the Discovery board using a right-angled 4-pin header (see **Figure 4**). The kit includes some stand-off spacers for mounting the display and due to manufacturing tolerances these may need to be trimmed so that the display connectors line up properly.

A test application

Now you will see how easy it is to use the Discovery-Board to build a simple application. Shown in **Figure 5** is a game to test reaction times. It uses the LC display, three LEDs with series resistors and a micro pushbutton with a pull-down resistor.

Although there are 10 LEDs with series resistors already available on the board, I decided to add some different coloured LEDs to the small prototyping plug board area. This will give a more authentic traffic light sequence. This application allows you to easily measure levels of alertness by giving a direct readout of reaction times on the LC display. The display just needs two wires to transfer data (SDA und SCL) which cuts down on the number of interconnecting wires. The pull-up resistors needed for the two I²C bus lines are already included in the ESP32 module so no additional resistors are required when only one slave device is used on the bus. The software to program the ESP32 can be found on the Elektor webpage for this article [2], more about this in the next section.

Programming the ESP32 Pico board

When it comes to programming the ESP32 Pico board there are a number of alternative environments and programming languages available.

The ESP-IDF [3] Development Framework (ESP-IDF) is the official development environment from Espressif. It is a powerful tool and is somewhat complex with a correspondingly steep learning curve. Hard core C++ programmers should however find their way around with little difficulty. To get started a bit faster you can choose a very common development environment, widely used for programming many microcontrollers due to its ease of use and extension capabilities.

If you are familiar with the Arduino development environment [4] (it runs

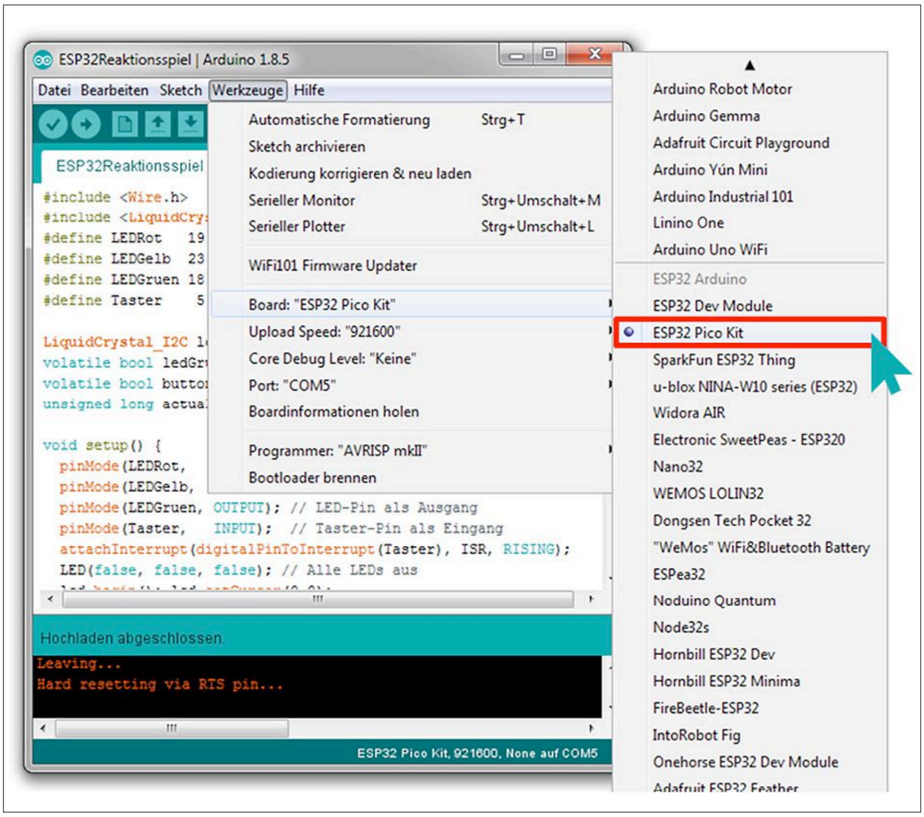


Figure 6. Selecting the ESP32 Pico Board (ESP32 Pico Kit) in the Arduino IDE.

on all major operating systems) and also enjoy programming in C ++, you will be pleased that there is an ESP32 Arduino core [5] for the ESP32. The integration process takes place quite quickly in about 15 to 20 minutes [5] [6]. Although not all the ESP32 functions are covered yet, most of the sketches that were written for the ESP8266 will run. Here are some examples of functions that can be programmed using familiar Arduino commands:

- the pin configuration with *pinMode;*
- access to digital pins with *digitalRead* and *digitalWrite*;
- interrupts with *attachInterrupt*;
- the serial interface;
- the I²C and SPI bus;
- Wireless communication via Wi-Fi and Bluetooth (the latter is still limited).

Over time, more functionality will be added. After successful integration of the ESP32 Arduino core (I am using Windows 7) I have the menu item *Tools -> Board* to show the list of available platforms. And lo and behold right at the bottom of the list some ESP32 boards have recently been added and the ESP32 Pico board or kit is also included (**Figure 6**).

As a rule, operating systems such as Windows 7 and later versions or Linux do not experience any driver problems and the ESP32 board is recognised correctly. Of course, this depends largely on the type of USB-to-UART module used. In any case, the Pico board is automatically detected and the *Silicon Lab CP210x USB to UART Bridge* driver [7] is installed.

An example sketch

In this small example sketch it shows how one of the digital to analogue converters in the ESP32 can be configured to generate a sine wave output signal. Both the DACs (DAC1 - GPIO25, DAC2 - GPIO26) have 8-bit resolution which allows the output voltage to be set to any one of 256 levels (0 to 255). The voltage range extends from 0 V to 3.3 V thereby giving the output level a resolution of 12.89 or 13 mV approximately.

The program sketch is shown in **Listing 1**. In line 7 within the *setup* function the GPIO pin 25 is configured as the DAC output using the key word ANALOG. This means that the *dacWrite* function in line 12 of the loop function will use this as its output pin. The *for* loop uses the specified range of values and the *sin* function to give the values for DAC1 which is then output to the GPIO pin. The resulting waveform is shown in **Figure 7**.

The sketch is automatically loaded to the ESP32 module when the upload button is clicked in the IDE user interface. It's not necessary to press any physical button on the board to start the firmware upload. This sketch and more is available from the resource at Github [8] where you will also be able to download the latest version of the software.

Listing 1. Program to generate a sinewave output signal.

```
#define DAC1       25
#define Steps      1
#define Offset     128
#define Amplitude  100

void setup() {
    pinMode(DAC1, ANALOG);
}

void loop() {
    for(int i = 0; i < 360; i = i + Steps)
        dacWrite(DAC1, int(Offset + Amplitude * sin(i * PI/180)));
}
```

Bluetooth? No worries

The ESP32 does of course support communication protocols for both Wi-Fi and

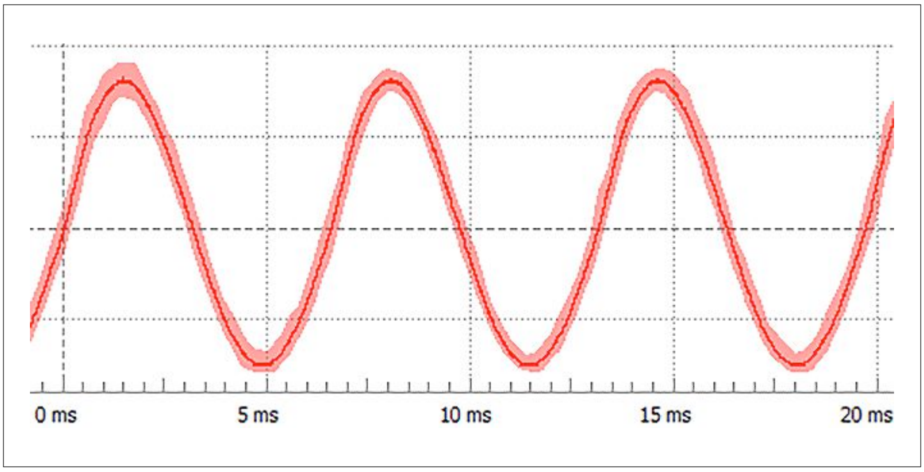


Figure 7. The output sinusoidal waveform from GPIO25.

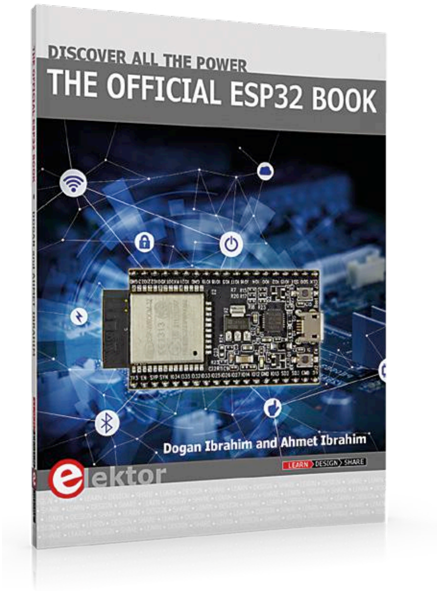


Figure 8. *The Official ESP32 (e-)Book* from Elektor.

Bluetooth. This capability allows the ESP32 to operate as a stand-alone device or as a slave.

The module has both Bluetooth 4.2 Low Energy (BLE) and classic Bluetooth capability. This functionality is of course catered for by the Arduino core ESP32 amongst others so that applications using Bluetooth connectivity can be easily implemented. At the Github address already mentioned you can find an example sketch demonstrating the use of Bluetooth in an application ('Hack 03'). The possibilities become even more intriguing with the help of development environments such as App Inventor [9] which allows you to quickly build Android apps without too much effort. The process of transferring data between an Android device and the ESP32 via Bluetooth is thereby made really simply. App Inventor will require the BLE extension [10] to be installed first. There are also lots of BLE application examples [11], which are well worth studying.

Go further with the ESP32

Now that your appetite for the ESP32 has been whetted you may want to discover more. Available from the Elektor Store is *The Official ESP32 Book* [13]. This is an introduction to the ESP32 processor and describes the main hardware and software features of this chip. Its main aim is to teach the reader how to use the ESP32 hardware and software in practical projects, especially using the popular ESP32 development board. Many basic, simple, and intermediate level projects are given in the book based on the ESP32 DevKitC development board, using the Arduino IDE and also the MicroPython programming language (**Figure 8**). ◂

(180341-03)

Web Links

[1] www.espressif.com/en/products/hardware/development-boards
[2] www.elektormagazine.com/180341-01
[3] https://esp-idf.readthedocs.io/en/latest/
[4] www.arduino.cc/en/Main/Software
[5] https://github.com/espressif/arduino-esp32
[6] www.elektormagazine.com/160454
[7] www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers
[8] https://github.com/erikbartmann/ElektorESP32/
[9] http://ai2.appinventor.mit.edu/
[10] http://appinventor.mit.edu/extensions/
[11] http://iot.appinventor.mit.edu/#/bluetoothle/bluetoothleintro
[13] www.elektor.com/the-official-esp32-book
[14] www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

Elektor Labs Pipeline

From simple to complex, from measuring to controlling, Elektor Labs is a place for all projects about electronics. Check out this selection of projects, there may be something in it that you can use.

Slow-start your audio amplifier

At power-on, the current surge demanded by a circuit presenting a large capacitive load to the power supply can destroy the former or trigger its short-circuit protection. In both cases the result is the same: the circuit will not start. A solution is to insert a soft-start circuit to control the inrush current. This soft-start circuit with many additional features was designed to protect an audio amplifier and you can build it too.

@ Elektor Labs: https://goo.gl/Pknx2g

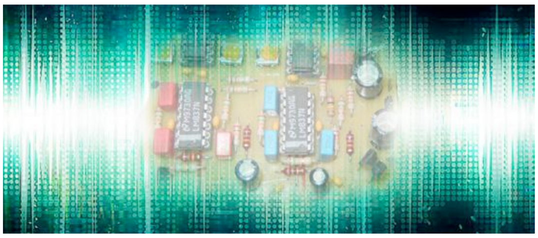
Simple water detector and alarm

Did you know that many projects published on the Elektor Labs website include the word “simple” in their title? The one we highlight here is just one of them. Reading the project’s description reveals how you can transform a commercially available magnetic door switch into a water detector and alarm by adding just two resistors...

@ Elektor Labs: https://goo.gl/RC5kiQ

Active audio crossover filter

Besides small SUVs, crossovers are also filters used in audio to divide the audio signal into two or more frequency bands. A typical example is a home cinema audio system where a subwoofer handles the low bass frequencies while other speakers reproduce the mid and high frequencies. Here is an active 3-way crossover design with three or four stages based on state-variable filters.



@ Elektor Labs: https://goo.gl/yNh27S

Build a Raspberry-Pi-based high-voltage curve tracer

With a Raspberry Pi, a touch screen and “some” extra effort like developing a high-voltage power supply, writing a graphical interface, and tooling metal you can come up with great results like this high-voltage curve tracer...

(180295)

@ Elektor Labs: https://goo.gl/LsxtG5

Fast Forward 2018

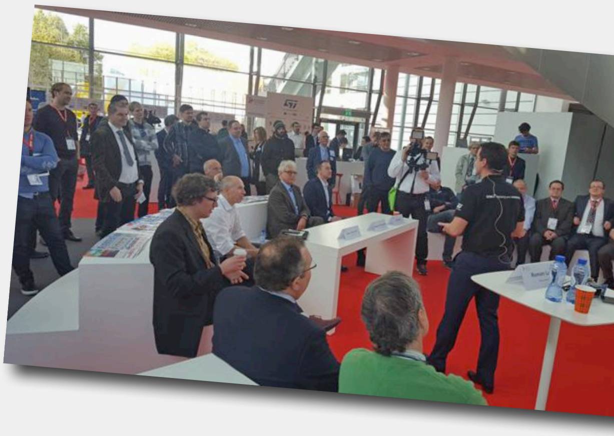
Do you have a great business idea?
Join our Start-Up and Prototype Contest

Get your business idea off to a flying start on the international marketplace as we are on the lookout for innovative Prototypes and Start-ups worldwide!

electronica Fast Forward, the Start-up Platform powered by Elektor unites the worldwide electronics marketplace, innovative technologies and an international media presence to bring your start-up to the attention of the movers and shakers of the industry.

This call goes out to creative thinkers, developers and start-ups the world over actively working on an electronics-related project. Send your ideas before September 9, 2018, to us at **www.elektormagazine.com/e-ffwd**. More than 50 participants from 15 countries already did so!

A jury comprising of ElektorLabs Magazine editors and design engineers will sift through the entries and evaluate them. The creators of best entries will be invited to participate in the contest final at **electronica** 2018 — the world leading trade trade show for electronic components, systems and applications held every two years in Munich, Germany.



Participating in ***electronica Fast Forward, the Start-up Platform powered by Elektor*** brings a unique opportunity to establish worldwide contacts and attend a wide range of exhibition events. You will also be enabled to pitch your creative ideas to an international audience of visitors and future customers.

On November 13, 2018, an international jury will select three winners from the finalists attending the **electronica** trade show. Besides receiving a prestigious 'electronica fast forward Start-up Award' to expose in your office, the winners will also benefit from international PR, consulting and marketing services with an estimated total value in excess of 150,000 euros. On top of that they will be supported by Elektor's global network of over 250,000 subscribers.

The overall winner will be awarded an international Elektor crossover marketing campaign worth 75,000 euros plus a free exhibition booth at **electronica** 2020. Second-place winner will be awarded an Elektor Media campaign valued at 50,000 euros while the third-place winner will benefit from media exposure through Elektor valued at over 25,000 euros.

Interested? Visit **www.elektormagazine.com/e-ffwd** and complete the sign-up form.
It is also the place to look for more information like the Terms and Conditions, and news.

We look forward to hosting the **e-ffwd 2018 edition** and welcome you at the **electronica** trade show in Munich this November.

ARROW ELECTRONICS | AVNET SILICA | EBV Elektronik – An Avnet Company | TRINAMIC MOTION CONTROL | Messe München International

(almost) everything you wanted to know about...

Buying Second Hand Test Equipment

Answers from **Robert Lacoste** (France)

photo : shutterstock.com

Q *Many manufacturers, particularly in Asia, are now offering new test equipment at very modest prices. So why should we be interested in the second-hand market?*

A Let's just say that they are not mutually exclusive. Test equipment has long been out of reach of the budget of most amateurs. Buying second-hand was not a choice, rather a necessity. For a few years now, thanks to the web, affordable test equipment has been easily available. Often, the big advantage of new equipment is that it has many 'bells and whistles' mainly in the software: a USB port to make screen copies, cursors and advanced mathematical functions, etc. On the other hand, for the same price, the intrinsic specifications are often inferior to those of older test equipment often a few tens of years old, especially in the analogue domain. And what you want from test equipment is usually just to measure something...

Q *For an oscilloscope for example, when would you advise buying new and when second-hand?*

A For an oscilloscope, buying second-hand often doesn't make a lot of sense. In fact, for a reasonable price you can buy a good new one with a bandwidth sufficient for everyday use (say 100 MHz). On the other hand, I think looking at the second-hand market is justified in two cases: when you're on a really tight budget, or when you have some specific requirements. Let me explain. Imagine you wanted to buy an oscilloscope, but with a budget limited to 200-250. You could buy a new one in this price range, but it would be an entry-level Asian made model, probably not too reliable in terms of both measurement accuracy and performance over time. For the same price, you could find a second-hand professional digital oscilloscope around fifteen years old with a 500 MHz bandwidth, such as a Tektronix TDS520. The other case where second-hand equipment is justified is to fill a very specific requirement. Let's say you need to buy an oscilloscope with a very high bandwidth, for example 20 GHz. You'll certainly find a new one, but you'll have to spend, oh, around 100 K… If you can accept the limitations of slightly older sampling digital oscilloscopes, you'd probably find a second-hand HP 54120B and a 20 GHz HP 54121A head for probably under 1 K.

Q *And for a spectrum analyser?*

A Well there I think the choice is simpler. What you're after in a spectrum analyser is really good linearity and the absence of parasitic signals produced by the analyser itself. That implies lots of screening and high quality components, impossible to find in new equipment at a cheap price. So for an amateur on a tight budget, second hand equipment makes a lot of sense. An antique HP 8560 covers the whole range up to 2.9 GHz and you'd find one for around 1 K, and you won't find a new one with anything approaching those specifications at that price. Similarly, an HP 70000, the 'Rolls-Royce' of spectrum analysers twenty years ago, can be found for not much more than that and can be used up to 100 GHz with suitable accessories, which are easy to find on the web! If you need to do more complicated tasks than such an analyser can do itself, it's always possible to do a bit of 'DIY" and connect a newer digital oscilloscope to the 'Intermediate Frequency' output of such an analyser, used thus as an excellent receiver. That gives you the best of both worlds…

Q *Are there any other types of test equipment for which the second-hand market is a particularly good choice?*

A The more exotic the equipment is, the more reason to look at buying second-hand. New test equipment for a specific function is usually very expensive, because not many are sold, but conversely you can find them for peanuts on the second-hand market, because demand is not high, even world-wide, especially when their first users no longer need them. This is very noticeable, for example, in the field of fibre-optic measuring equipment, where prices on the second-hand market are as much as 1000 times less than new prices, simply because the users are big companies who only buy new and change often to newer technology. It also applies to telecommunication companies. For example some time ago I acquired for my company a Rhode & Schwarz CMU200 3G Network Emulator… I wouldn't even dare to guess at the new price. As the big operators who usually use this type of equipment have moved on to 4G, it was let go at €900 (£800; $1000). What makes an equipment like this especially tasty is knowing that it has a great spectrum analyser and RF generator that can be used separately… but you have to know about this.

Q *So where and how do you buy second-hand?*

A Obviously eBay is a good starting point. From experience, don't be afraid to offer vendors prices well below their displayed prices if there are no bidders, several times I've been pleasantly surprised when trying this. There are also brokers who specialise in test equipment, some very good, the better of whom can be a good choice. Public auctions can also offer good bargains, if you have nerves of steel!

Q *But isn't there a risk in buying second-hand, particularly by correspondence?*

A Obviously there are crooks around, but my experience is that 99% of people are serious and honest! I speak from experience, because I have bought around a hundred items of test equipment during my career. If you're of a nervous disposition, then look for local discount vendors or direct sellers where you can see and test the equipment, but this will greatly limit your opportunities. Don't forget that most online transactions can be insured using certain payment methods, such as Paypal with eBay.

Q *Is it possible to find real bargains?*

A If you're a handyman and are lucky, and have a bit of spare time, one solution is… to buy equipment listed as faulty. At least you know what you're getting, and the price should match. With a bit of inside knowledge and expertise, or luck, I've had a few good experiences. For example the first digital oscilloscope I bought was a second-hand Lecroy 9350 listed as defective. I asked for details of the fault and the vendor told me that nothing happened when it was switched on. A power fault? That should be fixable, so I took the risk. On arrival, by lucky chance, it took me five minutes to find that the mains input filter was faulty. The repair was trivial and that oscilloscope served me well for many years.

Q *But you must have had some bad experiences?*

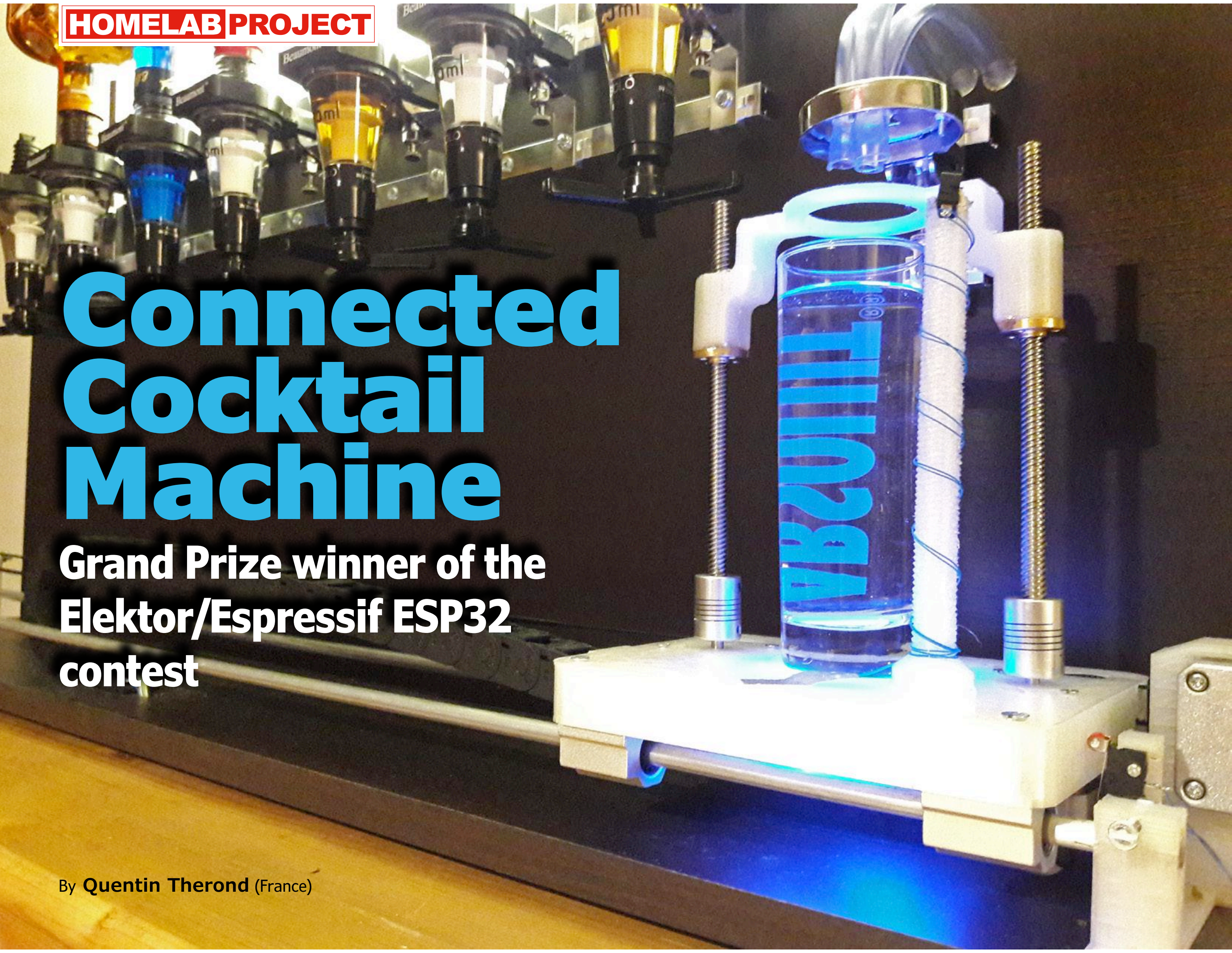
A Yes, two or three. The worst was a really dodgy organisation which pirated some legitimate eBay accounts (with good feedback profiles) to supposedly sell some very expensive equipment. I was a bit naïve in agreeing to pay by bank transfer for a vector network analyser. The seller and his bank account vanished immediately. I had prudently asked for a copy of an identity card but that turned out to be false, as were the company's credentials. On enquiry, it seemed that several hundred people were swindled on the same day by the same business. So yes, it does happen, but not very often.

Q *And the risk of faults? Second-hand means no guarantee!*

A Of course, it is best if you're a bit a handyman when you buy older equipment. That said, if you stick to the better known brands and devices, it's easy to find documentation on the Internet along with helpful guys who can assist you with repairs.

A final word: I'm not sure if a new bottom-of-line equipment has less risk of failure just after the guarantee expires than an older equipment which has already worked for 30 years without a problem! On the other hand, repairing newer equipment is sure to be a challenge. ◄

(180294-01)



HOMELAB PROJECT

Connected Cocktail Machine

Grand Prize winner of the Elektor/Espressif ESP32 contest

By **Quentin Therond** (France)

In your ultra connected house, ask your personal assistant to prepare your favourite cocktail. When it's ready, take it from the machine! You're not dreaming! It's possible with Quentin, grand prize winner of the ESP32 Contest 2018, with his connected cocktail machine. The jury was particularly impressed by the complete and detailed documentation for this project. Thanks to the information supplied (circuit diagrams, exhaustive parts list, source code, mechanical design and videos) you can make your own machine to dispense various cocktails to match your sitting room.

Everything started when a friend, Adrien, showed me a video of some robot arms preparing cocktails in a bar. Captivated by this machinery, I looked at some other videos of automatic cocktail systems, each better than the last. After several hours of this, I was motivated like never before to start a new project. Nonetheless, at home, I had to take into account the WAF (Wife Acceptance Factor). I suggested two versions to my wife (round or rectangular) along with a plan of how it could fit into the sitting-room. After considerable discussion, I got the green light for the rectangular model.

To cap it all, by chance I stumbled upon the Elektor site which was holding a competition with the manufacturer Espressif around the ESP32 chip, ideal for the Internet of Things (IoT).

The project

My new design had to allow ordering a cocktail from a smartphone, a tablet, a PC or a voice assistant (Google Home, Alexa…). The cocktails are previously loaded in memory by you and are presented on a web page. Once the desired drink is selected, the glass is put on a tray which moves to be filled

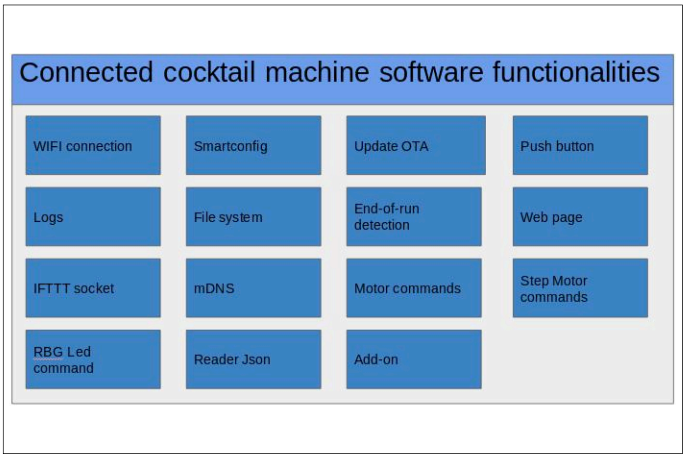


Figure 1. The software functions.

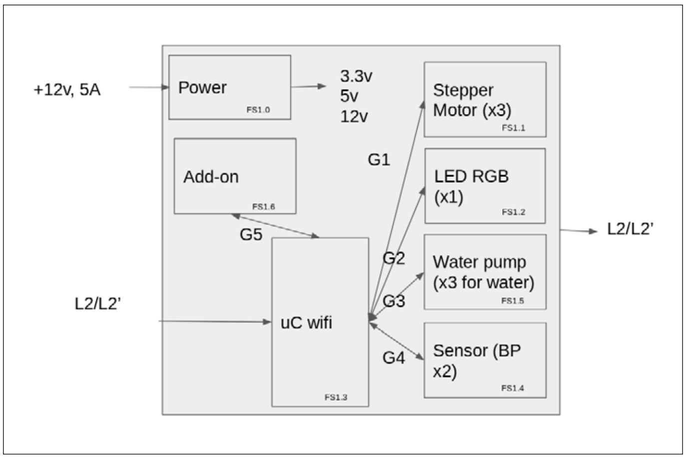


Figure 2. Functional diagram.

by the various available liquid ingredients.
The mechanical parts are made with a 3D printer, and ball bearings and stepper motors are currently offered at reasonable prices. For the structure of the machine I chose melamine-covered chipboard, often used in the construction of furniture. For the transport tray for the glass, I found in my internet searches some 3D printable parts offered by DIY Machines [1]. Finally, I drew some diagrams with an overview of the specifications and functions of the software (**Figure 1**) and the hardware (**Figure 2**).

Mechanics and structure

The structure of the machine can be made with various materials like profiled aluminium or wood (which I chose to match the décor of my sitting room). The two 1-metre wide boards are assembled with eight 90° metal brackets. The 3D printed parts are fixed on the bottom board. The aluminium shafts which support the glass transport tray are 95 cm long. While testing, I added a ball (**Figure 3**) under the tray to support it. What was happening is that when the tray was in the middle, the aluminium rods were not firm enough and collapsed. I think this comes from the quality of these rods!
Depending on the brands of tot measures used, different pressure was needed to deliver a tot. I added an end-of-travel detector on the Z-axis (**Figure 4**) to get round this problem. This also avoids having to calibrate this axis for different tot measures.
The addition of these two functions (not present in the prototype) is not essential, but it improves the reliability of the system over long use.
On the top board, I installed three pipes for the water pumps at position 0 and seven bottle tot measures spaced 10 cm apart from each other (positions 1 to 7).

Electronics

The master part of the system (cf. circuit diagram, **Figure 5**) is based on the Espressif module type ESP-WROOM-32 which contains the ESP32 SoC, which comprises two 32-bit LX6 microprocessors from Xtensa, with 4 MB of external flash memory and 512 KB of RAM. The module supports Wi-Fi and Bluetooth communication, and all this at a clock speed of 240 MHz. It assures, thanks to its programmed handler (WROOM-32 software), the uploading, processing, and downloading of data. Six of its GPIOs are available on a connector to allow adding further functions via an extension board.
I chose the ESP-WROOM-32 because if offers numerous advantages: competitive price (for a module with Wi-Fi); sufficient memory; simple and full IDE with plenty of examples; and stable Wi-Fi connection. It is thus an excellent choice for your own connected cocktail machine.
The board is powered by an AC line power supply with 12 V output. At least 3 A are needed to drive the stepper motors and the water pumps. On the circuit diagram you will see two regulators: 5 V and 3.3 V. The board thus offers three power voltages: 12 V (motors, pumps), 5 V (pumps) and 3.3 V (WROOM32). The three jumpers P2, P4 et P5 allow use of external power for the 12 V, 5 V and 3 3 V supplies. This is because at the start, I did not know the total consumption of the project. I didn't have technical documentation for some of the components and I did not want to redraft the printed circuit board if there was a problem. After the tests, an external power supply was considered unnecessary (jumpers P2, P4, P5 on pins 1 and 2). The ULN2803 is a very practical integrated circuit to attack problems of driving LEDs or a motor if you don't want to get tied up with a power control circuit. That said, it can switch at

Figure 3. Ball wheel for supporting the glass transport tray.

Figure 4. End of travel sensor for the Z axis.

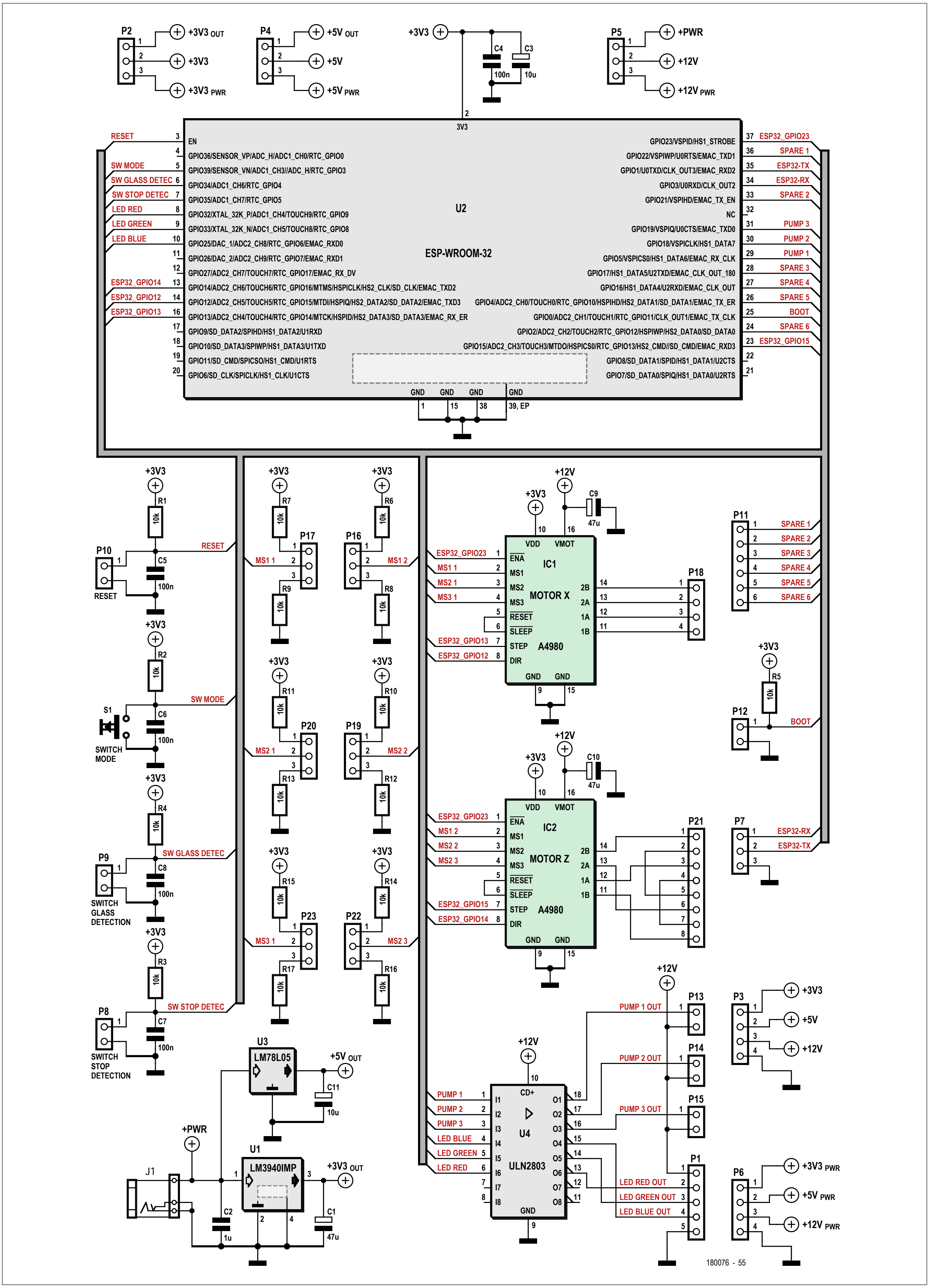

P2
+3V3 OUT
+3V3
+3V3 PWR
P4
+5V OUT
+5V
+5V PWR
+3V3
C4
100n
C3
10u
P5
+PWR
+12V
+12V PWR
U2
ESP-WROOM-32
3V3
EN
GPIO36/SENSOR_VP/ADC_H/ADC1_CH0/RTC_GPIO0
GPIO39/SENSOR_VN/ADC1_CH3/ADC_H/RTC_GPIO3
GPIO34/ADC1_CH6/RTC_GPIO4
GPIO35/ADC1_CH7/RTC_GPIO5
GPIO32/XTAL_32K_P/ADC1_CH4/TOUCH9/RTC_GPIO9
GPIO33/XTAL_32K_N/ADC1_CH5/TOUCH8/RTC_GPIO8
GPIO25/DAC_1/ADC2_CH8/RTC_GPIO6/EMAC_RXD0
GPIO26/DAC_2/ADC2_CH9/RTC_GPIO7/EMAC_RXD1
GPIO27/ADC2_CH7/TOUCH7/RTC_GPIO17/EMAC_RX_DV
GPIO14/ADC2_CH6/TOUCH6/RTC_GPIO16/MTMS/HSPICLK/HS2_CLK/SD_CLK/EMAC_TXD2
GPIO12/ADC2_CH5/TOUCH5/RTC_GPIO15/MTDI/HSPIQ/HS2_DATA2/SD_DATA2/EMAC_TXD3
GPIO13/ADC2_CH4/TOUCH4/RTC_GPIO14/MTCK/HSPID/HS2_DATA3/SD_DATA3/EMAC_RX_ER
GPIO9/SD_DATA2/SPIHD/HS1_DATA2/U1RXD
GPIO10/SD_DATA3/SPIWP/HS1_DATA3/U1TXD
GPIO11/SD_CMD/SPICSO/HS1_CMD/U1RTS
GPIO6/SD_CLK/SPICLK/HS1_CLK/U1CTS
GPIO23/VSPID/HS1_STROBE
GPIO22/VSPIWP/U0RTS/EMAC_TXD1
GPIO1/U0TXD/CLK_OUT3/EMAC_RXD2
GPIO3/U0RXD/CLK_OUT2
GPIO21/VSPIHD/EMAC_TX_EN
NC
GPIO19/VSPIQ/U0CTS/EMAC_TXD0
GPIO18/VSPICLK/HS1_DATA7
GPIO5/VSPICS0/HS1_DATA6/EMAC_RX_CLK
GPIO17/HS1_DATA5/U2TXD/EMAC_CLK_OUT_180
GPIO16/HS1_DATA4/U2RXD/EMAC_CLK_OUT
GPIO4/ADC2_CH0/TOUCH0/RTC_GPIO10/HSPIHD/HS2_DATA1/SD_DATA1/EMAC_TX_ER
GPIO0/ADC2_CH1/TOUCH1/RTC_GPIO11/CLK_OUT1/EMAC_TX_CLK
GPIO2/ADC2_CH2/TOUCH2/RTC_GPIO12/HSPIWP/HS2_DATA0/SD_DATA0
GPIO15/ADC2_CH3/TOUCH3/MTDO/HSPICS0/RTC_GPIO13/HS2_CMD/SD_CMD/EMAC_RXD3
GPIO8/SD_DATA1/SPID/HS1_DATA1/U2CTS
GPIO7/SD_DATA0/SPIQ/HS1_DATA0/U2RTS
GND
39, EP
RESET
SW MODE
SW GLASS DETEC
SW STOP DETEC
LED RED
LED GREEN
LED BLUE
ESP32_GPIO14
ESP32_GPIO12
ESP32_GPIO13
ESP32_GPIO23
SPARE 1
ESP32-TX
ESP32-RX
SPARE 2
PUMP 3
PUMP 2
PUMP 1
SPARE 3
SPARE 4
SPARE 5
BOOT
SPARE 6
ESP32_GPIO15
P10
RESET
R1
10k
C5
100n
R2
C6
S1
SWITCH MODE
R4
C8
P9
SWITCH GLASS DETECTION
R3
C7
P8
SWITCH STOP DETECTION
P17
P16
R7
R6
R9
R8
MS1 1
MS1 2
P20
P19
R11
R10
R13
R12
MS2 1
MS2 2
P23
P22
R15
R14
R17
R16
MS3 1
MS3 2
U3
LM78L05
C11
10u
+5V OUT
U1
LM3940IMP
+3V3 OUT
C1
47u
C2
1u
J1
+PWR
+12V
C9
47u
IC1
MOTOR X
A4980
VDD
VMOT
ENA
MS1
MS2
MS3
RESET
SLEEP
STEP
DIR
2B
2A
1A
1B
P18
C10
IC2
MOTOR Z
P21
ESP32_GPIO15
ESP32_GPIO14
MS1 2
MS2 2
MS2 3
P11
P12
R5
P7
U4
ULN2803
CD+
P13
P14
P15
P1
PUMP 1 OUT
PUMP 2 OUT
PUMP 3 OUT
LED RED OUT
LED GREEN OUT
LED BLUE OUT
P3
P6
180076 - 55

Figure 5. Circuit diagram of the electronics board at the heart of the connected cocktail machine.

most 500 mA. For this reason it only controls the relays, and not the water pumps directly. As the 5 V pumps I originally chose did not come up to expectations, I replaced them with more powerful 12 V models. I had to manually modify the power to the ULN2803 on the printed circuit board.

The stepper motors are controlled by an A4988 module using a clock signal and a choice of direction. The precision attained is almost surgical (±1 mm in full step). The entries 'MS_x' allow you to configure the resolution in steps if you don't need so much precision. The wiring on P21 allows control of two stepper motors at the same time. It's important to avoid desynchronizing the Z-axis screws. That would cause the destruction of the mechanical part that pushes on the tot measures.
You can find many other details of the electronic circuit in the file that you can download at [2].

Cabling and colour code of the RGB LED of the machine

There are a few things connected to the electronics board (**Figure 6**) to know about:

- Three stepper motors
- Two end of travel detectors
- a maximum of three water pumps
- Some jumpers to select the power supply mode and the resolution of the stepper motors.

The RGB LED fixed under the transport tray gives information on the status of the system:

- Red on: starting up (initializing), machine not connected to the Wi-Fi network
- Red flashing: Sending SSID and password to the Wi-Fi network router
- Blue on: system initialised and ready to work
- Blue flashing: in the process of making a cocktail
- Green on: waiting for a Wi-Fi update
- Green flashing: update in progress.

WROOM-32 software

The software architecture (**Figure 7**) shows that the use of one of the Espressif IDEs is obligatory. I chose the SDK-IDF (version 2.1) because I already know the Arduino IDE. If you wish to use a different one, the Hardware Abstraction Layer (HAL) of the programme will easily let you use another IDE.

Configuration of the software and traces

Before programming the WROOM-32, I suggest you open the following files:

- `Board.h`: choice of all the inputs/outputs for control of the pumps, stepper motors, end of travel detectors and the LEDs
- `Esp32.c`: Setting of the static IP and of the mDNS
- `MotorHandling.c`: calibration of the stepper motors and the pumps.

If you don't have your home network on 192.168.1.x with a subnet mask of 255.255.255.0, you will need to change the information of the function 'IP4_ADDR' and in `Html.c` :

```
ESP_ERROR_CHECK(tcpip_adapter_dhcpc_stop(TCPIP_ADAP-
TER_IF_STA));
   tcpip_adapter_ip_info_t info = { 0,};
IP4_ADDR(&info.ip, 192, 168, 1, 51);
IP4_ADDR(&info.gw, 192, 168, 1, 1);
IP4_ADDR(&info.netmask, 255, 255, 255, 0);
   ESP_ERROR_CHECK(tcpip_adapter_set_ip_info(TCPIP_ADAP-
TER_IF_STA, &info));

mdns_server_t * mdns = NULL;
mdns_init(TCPIP_ADAPTER_IF_STA, &mdns);
   ESP_ERROR_CHECK(mdns_set_hostname(mdns, "mybar"));
   ESP_ERROR_CHECK(mdns_set_instance(mdns, "mybar"));
```

I'd advise you to use the section `#define BAR_DEBUG` in `Debug.h` and connect the board to your PC with a USB serial adapter on P7. This will allow you to better understand any eventual

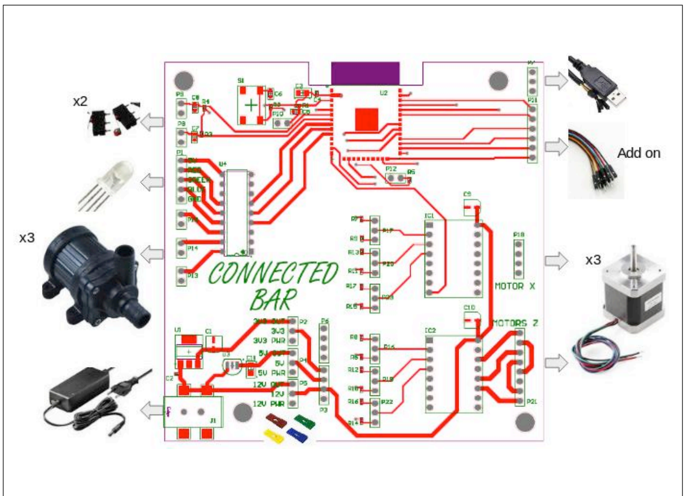


Figure 6. Cabling of the external parts.

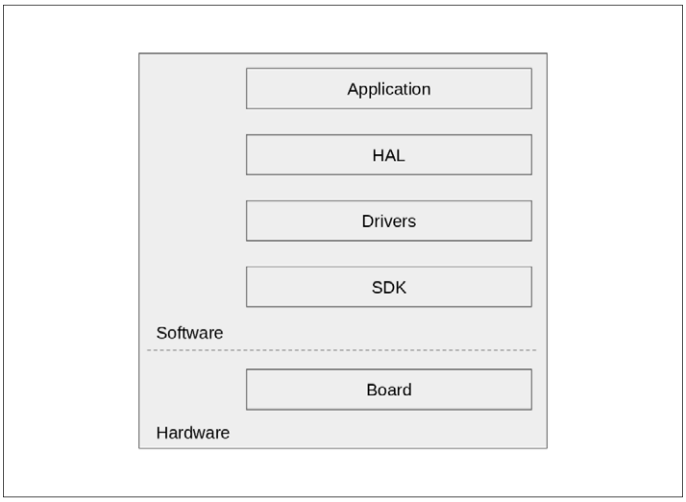


Figure 7. Software architecture.

Figure 8. Web page of the connected cocktail machine.

Listing 1. JSON table of bottles

```
{
   "bottles": [{
          "bottle": {
              "name": "eau",
              "note": "1",
              "position": 0
          }
      }, {
          "bottle": {
              "name": "menthe",
              "note": "0%vol",
              "position": 1
          }
      },
            ….
     }]
}
```

software problems. Open a serial console (115200 b, 8N1) and power up the board. The traces will follow.
At system startup the LED is red while the X and Z axes are not in the home position and the system is not connected to Wi-Fi.

Connection to Wi-Fi

Connection to the cocktail machine is is made via SmartConfig. The smartphone application is available on the App Store (iOS) and Google Play (Android). You will need to download and install 'ESP8266 SmartConfig' on your phone. Once the SSID and the password of your internet router have been sent, the system saves them in flash. At this stage, the LED should be constant blue. You can now order a cocktail. If sending the 'SSID' and 'password' doesn't work, use the function `Wifi_saveSSIDAndPass("VOTRE SSID", "VOTRE MDP");` in `Initialize.c`.

Order a cocktail

To order cocktails, I have chosen to use a web page. That allows compatibility with any smartphone, tablet or PC. The web page (**Figure 8)** is accessible on the address `http://mybar.local/` or `http://192.168.1.51` on your LAN.
The HTML and CSS code is produced by the C code of the WROOM-32. When the module starts up, the software scans the tables for 'bottles' and 'cocktails' in JSON format to create a dynamic HTML table of three columns and n lines. The first column contains CSS Buttons with the name of the cocktail (available in the JSON). The second column contains the list of ingredients to add manually (not available in the list of bottles). The third column lists the ingredient(s) available in the cocktail machine (available in the list of bottles).
In the first line of the HTML code, you can see the title and a link to an image:

```
<title>Connected bar</title><link href="https://url.png"
rel="icon" type="image/x-icon" />
```

When we add the link to the web page on a smartphone, this line allows us to have a nice shortcut with a name and logo (**Figure 9**).
You can change the log by changing the URL in `link href`.
Note: the current system does not have the ability to detect the glass. You could add this by connecting a card to the extension connector.

Positioning the bottles and creating the cocktails

Before getting into the details of the JSON data structure, it's worth knowing that you can find JSON courses and correctors of JSON syntax on the internet. If the syntax is not right, the system won't display your cocktails.
There are two JSON structures in the project (`CocktailJson.h`). The first defines the placement of the bottles, the second the list of ingredients for each cocktail.

Listing 2. JSON table of cocktails

```
{
   "cocktails": [{
          "cocktail": {
              "name": "sirop de menthe",
              "ingredients": [{
                      "ingredient": {
                          "name": "menthe",
                          "measure": 1
                      }
                  },
…
                  {
                      "ingredient": {
                          "name": "eau",
                          "measure": 5
                      }
                  }
              ]
          }
      }, {
            ....
    }]
}
```

The JSON ‘bottles’ table (**Listing 1**) is a list of the bottles with a name, a position and a note. Position 0 is the initial position of the tray. The ‘note’ field allows, when you’re in positon 0, to choose a pump (1 = pump one, 2 = pump two…). This explains why there are several bottles at the same position. Note: the position must physically exist on the cocktail machine..

The JSON ‘cocktails’ table (**Listing 2**) is a list of cocktails, each composed of a name and a table ‘ingredients’. Each ingredient has a name (which is not necessarily available in the cocktail machine) and a ‘measure’, which is the quantity of liquid to

Listing 3. Making of a cocktail

```
OsQueueReceive(pCtx->xQueueCocktailEventQueue, &QueueCocktail, OsPortTimingPeriod);
LedRGBHandling_ExecuteLedTaskFromISR(BLUE_LED_FAST_BLINKING);
MotorHandling_setInitialPosition();
int nbIngredients = Cocktail_getDispoIngredients(bottleList.bottle, bottleList.position, bottleList.
   measure, bottleList.note, QueueCocktail);
int goToPosition = 0;
int currentPosition = 0;

for(int i = 0; i < nbIngredients; i++)
   {if(currentPosition != bottleList.position[i])
        {
        goToPosition = bottleList.position[i] - currentPosition;
     MotorHandling_setPositionOnX(goToPosition);
        currentPosition += goToPosition;
        CpuDelay_ms(500);
        }

        {
        if(currentPosition != 0)
     {
        MotorHandling_getAMeasureOnY(bottleList.measure[i]);
        }
        else
        {
        MotorHandling_setInitialPosition();

           if(bottleList.note[i] == '1')
           {
           MotorHandling_getAMeasureOnPump(bottleList.measure[i], MOTOR_PUMP_3);
           }
           else if(bottleList.note[i] == '2')
           {
           MotorHandling_getAMeasureOnPump(bottleList.measure[i], MOTOR_PUMP_2);
           }
           else if(bottleList.note[i] == '3')
           {
           MotorHandling_getAMeasureOnPump(bottleList.measure[i], MOTOR_PUMP_1);
           }
           else
              {
              BarDebug_err("Pump not found\n");
              }
           }
        }
   }

MotorHandling_setInitialPosition();
LedRGBHandling_ExecuteLedTaskFromISR(BLUE_LED);
```

Figure 9. It is possible to personalize the shortcut to the web page.

Figure 10. Configuration of Webhooks.

dispense (1 = 1.25 ml, 2 = 2.5 ml...). Note: a cocktail is made up in order of the list of ingredients.
Therefore, to add or change bottles and cocktails, you need to modify the file `CocktailJson.h` in `src/app/cocktail/`.

When a cocktail is ordered, the order is sent to the task `QueueCocktail_receivedTask` (**Listing 3**) via a FreeRTOS queue. The task causes the LED to flash blue and positions the X and Z axes to the home positions. Thereafter, having retrieved the list of positions, the number of ingredients and the quantities, the programme goes into a loop. At each iteration, the tray is moved to the desired position as a function of the current position and the desired volume of liquid is dispensed.

Updating the software and the cocktails via Wi-Fi

It is possible to reprogram the memory of the WROOM-32 over Wi-Fi to allow modification of the software, the bottles or the cocktails. One advantage of this method is being able to update the system without a wired connection.
When you click the 'update' button (at the bottom of the web page), the system restarts and launches the update task. The LED turns green.

From your PC, you can execute the update script by giving the IP address and the software for the cocktail machine. During the execution, the computer sends packets of 4096 bytes to the WROOM-32 which saves them in sequence in a new partition. The significance of this packet size is that it is exactly the size of a memory page. Once the software upload is finished, the application signals the bootloader that it can run the application at the address of the new partition.

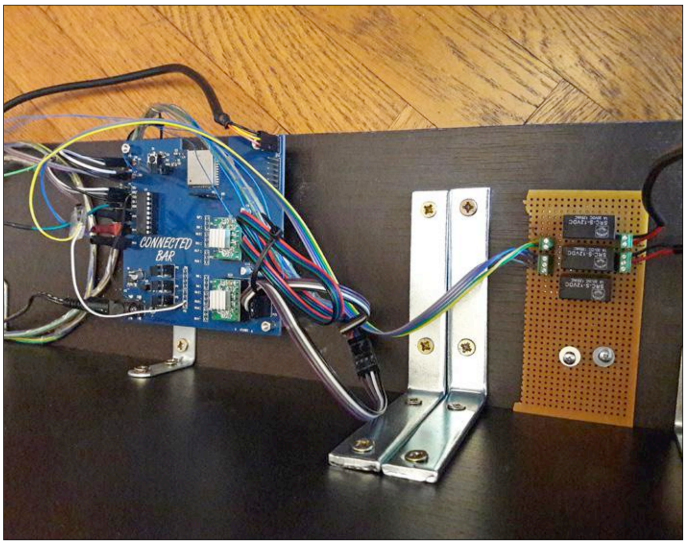
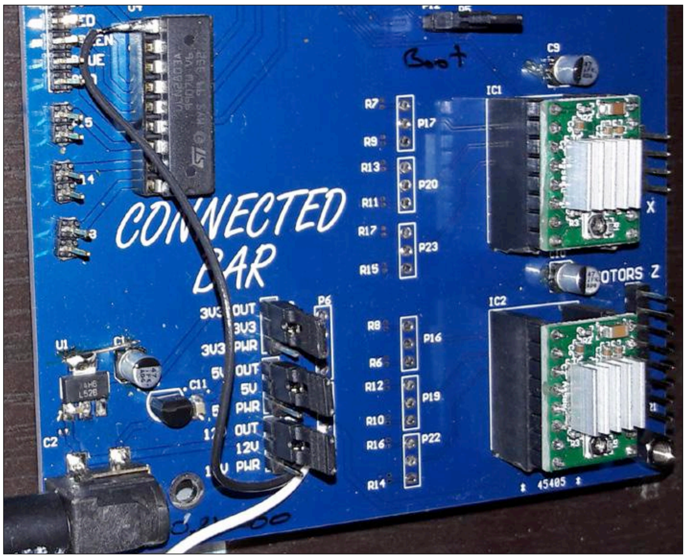


Figure 11. Control and Wi-Fi access board *in situ*.

I took the example of 'classycodeoss' [3] for this function. You can find a copy of his update script `update_firmware.py` in the directory `utils/`.

Making the system compatible with a voice assistant

The service IFTTT [4] (if this then that) allows to start an action as a result of a condition from any of your IoT devices which are IFTTT compatible. There is a long list. I limit myself here to its use on a voice assistant, but you could create other



Web Links

[1] Parts in 3D from DIY Machines : http://www.thingiverse.com/thing:2478890
[2] Project engineering page at Elektor Labs : http://www.elektormagazine.com/labs/connected-cocktail-machine
[3] Source code for update utility: https://github.com/classycodeoss/esp32-ota
[4] IFTTT : https://ifttt.com/

applets. To use the IFTTT service, you need to redirect a port on your internet router to the connected cocktail machine (IP:192.168.1.51, port:4551).
To create a new applet on IFTTT with Google Home, you need to open an account on their site [4]. Click on 'New Applet'. You will see: if 'this' then 'that'. Click on 'if', and select 'Assistant Google' follow the instructions. Click on 'that', select 'Webhooks', and follow those instructions (**Figure 10**).
Webhooks allows you to send an HTTP command to the cocktail machine. When the system receives a command of type POST sent by IFTTT, it verifies if the name of the cocktail is available in the list of cocktails. If the cocktail exists, it returns the code HTTP 'HTTP/1.1 204', if not, 'HTTP/1.1 400'.

Modifications for the V2 machine

The first version of the cocktail machine is working, but it shows that certain improvements should be looked at for the second version. On the hardware side, I need to respin the printed circuit board for the modification of the power to the ULN2803 (12 V instead of 5 V). I need to add some relays for the control of the pumps, and remove the possibility to use an external power supply.
On the software side, a secure connection between the WROOM-32 and IFTTT would be more professional. I'd also like to be able to update the bottles and cocktails without having to reprogram all the software. I also need to change the section `#define MY_IP` with a function `getIp();`.
If you construct this magnificent machine, think about posting a photo of your creation on the Elektor Labs website.

(180076)

The author thanks

- Elektor and Espressif for organizing the contest.
- Adrien for his help with the electronics.
- DIY Machines for the modelling of the 3D parts.
- Classy Code GmbH for their sample code for OTA (Over The Air updates).

@ WWW.ELEKTOR.FR

- → ESP32-WROOM-32 Module
  www.elektor.com/esp-wroom-32
- → ESP32-Pico-Kit board
  www.elektor.com/esp32-pico4
- → ESP32-WROVER V4 module
  www.elektor.com/esp32-wrover-v4
- → ESP32 DevKitC board (based on ESP-WROOM-32)
  www.elektor.com/esp32-devkitc
- → ESP-WROVER-Kit development board
  www.elektor.com/esp-wrover-kit
- → VFD / ESP32 clock kit with acrylic case
  www.elektor.com/vfd-tube-clock
- → M5Stack – ESP32 based development kit
  www.elektor.com/m5stack-basic-devkit

Tips and Tricks

From readers for readers

Another neat solution to a tricky problem

Improve Relay Efficiency

.. and reduce energy losses

By **Dr. Martin Oppermann** (Germany)

With the addition of a couple of cheap components it's possible to reduce the power losses in a relay driver circuit. The modification ensures that these trusty electromechanical components will not be out of place in state-of-the-art electronic circuit designs.

Amongst all the high-tech smart devices and IoT there is still applications that rely on the humble electromagnetic relay. This single component can easily switch high power levels without much fuss and provides galvanic isolation from the switched circuits. They are favoured for low-noise, low-distortion audio and video signal switching at amplifier input stages. The electromechanical relay has however a number of performance shortcomings; it is relatively slow to switch, the relay contacts suffer from bounce when the relay switches and the coil consumes a relatively high level of energy when the relay is driven at its specified operating voltage.

Energy efficient switching

We can definitely do something about this latter disadvantage. Once a relay has switched it is possible to reduce the voltage applied to the coil significantly and still keep the relay contacts closed. For example, the manufacturer specifies the following values for the popular universal relay type RE030012 (which in this case has a 12 V coil) [1]:

- Specified coil operating voltage: 12 V
- Coil operating voltage: 8.4 V
- Release voltage: 1.2 V
- Coil resistance: 720 Ω
- Power dissipated in coil at 12 V: 200 mW.

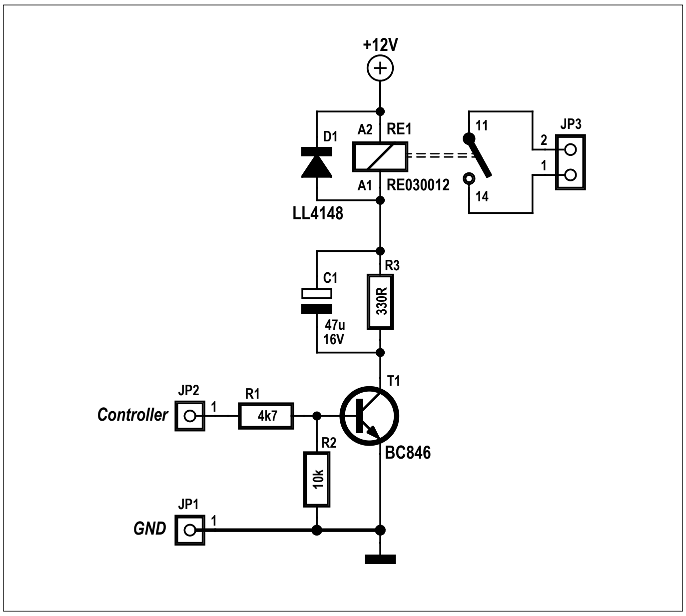


Figure 1. Schematic of the energy-saving relay switching circuit.

In order to minimise contact bounce and reduce wear it's recommended to apply the specified operating voltage at switch on. This achieves the maximum electromagnet pull and ensures good switching characteristics of the contacts. Once the relay has been energized, the coil voltage and thereby the operating current and power losses can be reduced. A number of relay-driver ICs [2] and [3] are commercially available which implement this energy-saving technique. It's even easier to make your own as **Figure 1** shows.

The relay is operated via a signal from a microcontroller (such as an Arduino) which switches transistor T1 via a logic high (3.3 to 5 V). At the moment the relay is switched on the electrolytic capacitor C1 is in a discharged state so that the charging current of the capacitor I_{in} is largely determined only by the operating voltage V_{op}, the relay coil resistance R_{coil} and the equivalent series resistance (ESR) of the electrolytic capacitor. With low ESR capacitor types such as those used on PC mainboards, this resistance is so low it can be neglected. At the moment voltage is switched to the relay coil, capacitor C1 therefore effectively acts as a short circuit across resistor R3 ensuring the full supply voltage is applied to the coil.

As long as T1 is conducting C1 will charge up to the steady state voltage drop produced across R3 and the current through R3, and the coil resistance can be determined as follows.

Current at switch on (inrush current, with C1 discharged):

$I_{in} = \mathrm{V}_{op} / \mathrm{R}_{coil} = 12 / 720 = 16.6$ mA.

Operating current (holding current, C1 charged):

$I_{op} = \mathrm{V}_{op} / (R_{coil} + \mathrm{R3}) = 12 / 1050 = 11.4$ mA,

which gives the power dissipated in the relay drive current path:

Power at switch-on:

$P_{in} = V_{op} \times I_{in} = 12 \times 0.016 = 199$ mW.

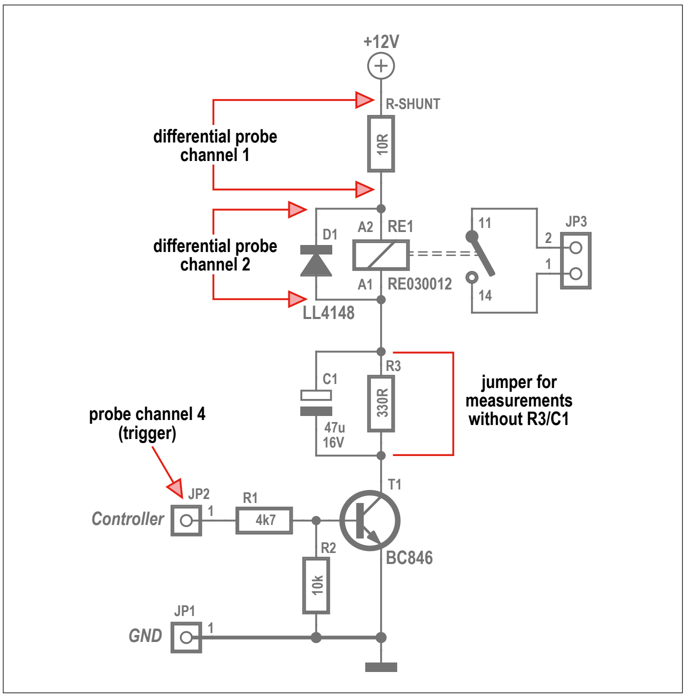


Figure 2. Measurement test points. The current level is determined by measuring the voltage drop across a 10-Ω shunt resistor.

Continuous operating power:
$P_{op} = V_{op} \times I_{op} = 12 \times 0.0114 = 137$ mW.

This indicates that the continuous power loss in the relay driver stage is now reduced to about 2/3 of the power drawn at switch on! This is about the same saving that can be expected by using a commercial relay driver IC [2]. The power reduction can be demonstrated by the measuring circuit in **Figure 2**.

The waveforms shown for channel 1 and channel 2 were produced using the differential current sense probe described in the September & October 2016 issue of *Elektor Magazine* [4]. The waveforms on the four channels are as follows:

Channel 1 (yellow trace): Voltage across the shunt resistor giving the circuit current.
Channel 2 (cyan trace): Voltage across the relay coil.
Channel 4 (blue trace): Relay control signal from the controller (H = on) / Trigger signal
Channel M (lilac trace): Power dissipation derived from channels 1 and 2.

In **Figure 3**, the upper waveform shows the on and off cycles without the energy-saving RC network, below are the equivalent waveforms with the RC network. The reduced power losses following the switch-on surge can be clearly seen. C1 has no effect on the relay turn-off time and C1 discharges through R3 in around 100 ms. The relay is then ready to be switched again. ⏮

180204-02

Web Links

[1] TE Connectivity: RE030012 datasheet : www.te.com/

[2] iC-Haus: iC-JE PWM Relay / Solenoid Driver datasheet: www.ichaus.de/

[3] Microchip: HV9901 datasheet: www.microchip.com/wwwproducts/en/HV9901

[4] Rosenkränzer, Alfred: Current Probe for Oscilloscopes, Elektor Magazine July & August 2016: www.elektormagazine.com/150182

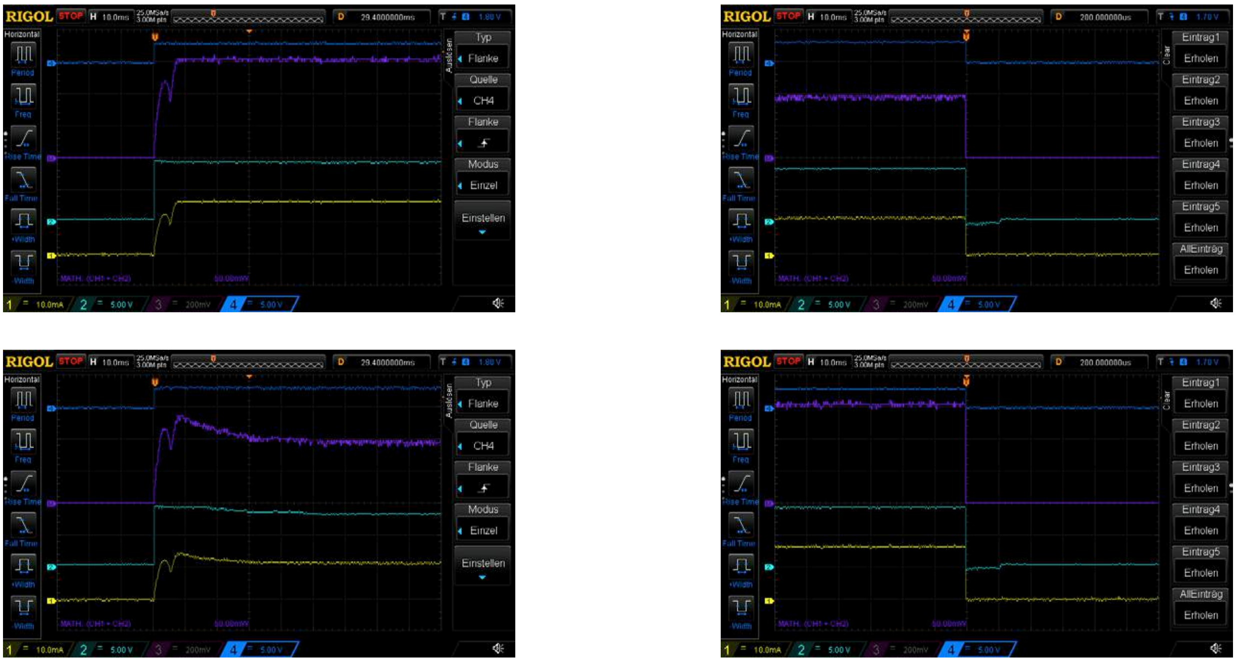


Figure 3. Waveforms showing relay switching with and without the RC network.

HomeLab Helicopter

Ham Radio 2018 and The Bodensee Maker Faire

A tour of this event held beside idyllic Lake Constance

By **Dr. Guido Schönwälder**

Friedrichshafen calling … and in response, some 180 exhibitors and hobby organisations from 32 countries, along with 5,460 visitors, all invaded this lakeside resort at the foot of the Alps. From 1 to 3 June 2018 the location hosted the 43rd HAM RADIO show, together with the Bodensee Maker Faire. The HAM RADIO extravaganza is the leading fair in Europe for amateur radio fans – and an event not just for radio hams!

The annual HAM RADIO event is not only about lectures and sales stalls; some people go there just to gather up information, go on a treasure hunt at the flea market, make a sightseeing flight in the zeppelin balloon or simply meet friends from all over the world in person – folk that they knew previously only at the far end of a radio linkup.

Learning process

Traditionally the HAM RADIO event always offers training opportunities for radio amateurs, with plenty of teach-ins and workshops. This year the range of topics ranged from 'Solar modules and their interesting effects in the IR and UV range' through the topic 'Simple circuits for THz radio communication' right up to the major project 'Es'hail-2 using the AMSAT P4-A transponder'.

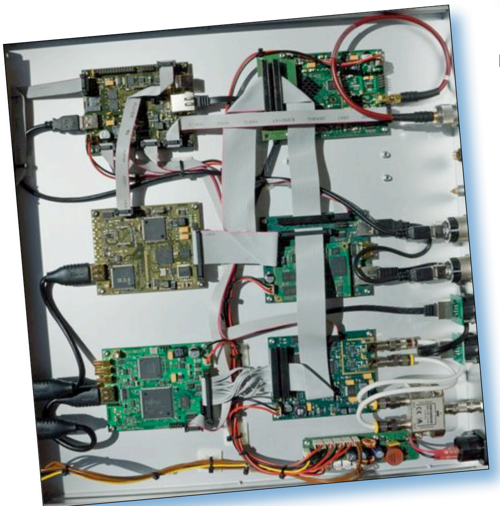


Figure 1. Home designed hardware for digital amateur television (Es'hail-2 satellite project using the AMSAT P4-A transponder).

These topics show that the radio amateur is anything but an ageing nerd experimenting with obsolete components. 'Es'hail-2 using the AMSAT P4-A Transponder' will be the first geostationary amateur radio satellite. In addition to the classic operating modes provided by numerous existing amateur radio satellites, digital amateur radio television (DATV) in DVB-S2 will also be achieved in this project. Home-designed hardware (**Figure 1**) was on display at the show, with the developers patiently answering many questions from the visitors. The partic-

Figure 2. High-end amateur radio transceiver.

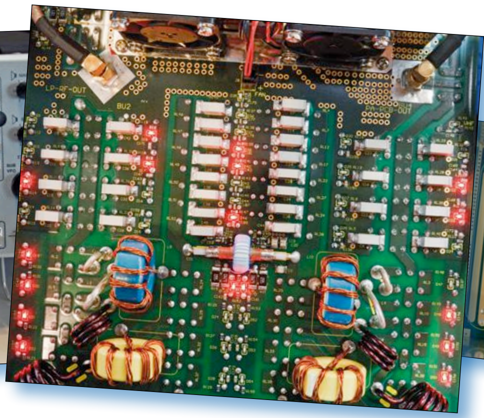


Figure 3. Inner workings of the high-end rig.

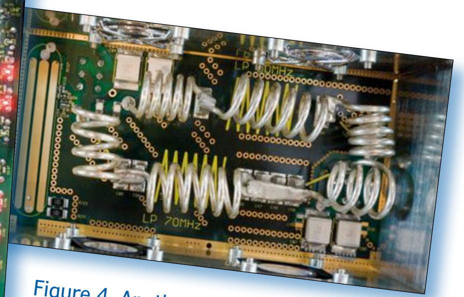


Figure 4. Another peep inside this high-end rig.

ular challenge for amateur radio satellite projects is not just developing the ground-based equipment, but also designing unconditionally reliable, space-ready hardware. One of the earth stations for satellite control will be installed in Bochum, Germany. Further information about this major project and other amateur radio satellites can be found at [1].

Broadcasting by hams

The ideal partner for HAM RADIO is DARC, the German Amateur Radio Club [2]. The club strives to arouse interest in technology through amateur radio and provide knowledge about technical relationships, thus guiding more young people into an engineering or scientific education. The club is also committed to integrating people with disabilities and widening international understanding among people. It also provides support for civil protection contingencies. To do this, the DARC employs not only the standard communication media such as the Internet and its own club magazine, but also makes a weekly broadcast on Radio DARC [3], every Sunday from 11.00 am in the 49-metre Broadcast band (6070 kHz). The program goes out over the Austrian Broadcasting Corporation's 100-kW transmitter near Vienna and is received reliably throughout Europe. The lecture programme at HAM RADIO offered listeners of Radio DARC the opportunity to find out how the broadcasts are produced to a high standard and now much technical effort is needed to produce a radio program. They also got a detailed insight of how things can still go wrong during a broadcast and how this is resolved professionally.

Something for everyone

In the main hall you could find everything needed for the amateur radio hobby – from antennas, through entry-level ham radio gear, right up to high-end amateur radio rigs (**Figure 2**), that arouse enthusiasm and desire from the very first glance at their well thought-out and high-quality construction (**Figure 3** and **Figure 4**). But it's not only radio amateurs who get their money's worth at HAM RADIO. Even electronics enthusiasts who are not RF-minded could strike rich here. On offer were tons of electromechanical components through to test equipment and much more. Laboratory power supplies of varying price ranges could be seen live (**Figure 5**), whilst spectrum analysers and oscilloscopes, which can mitigate many measurement problems and time-consuming troubleshooting, were available for direct comparison, trial and purchase (**Figure 6**). The ICOM company displayed, among many other devices, their software-defined broadband communication receiver IC-R8600 (**Figure 7**), covering the frequency range

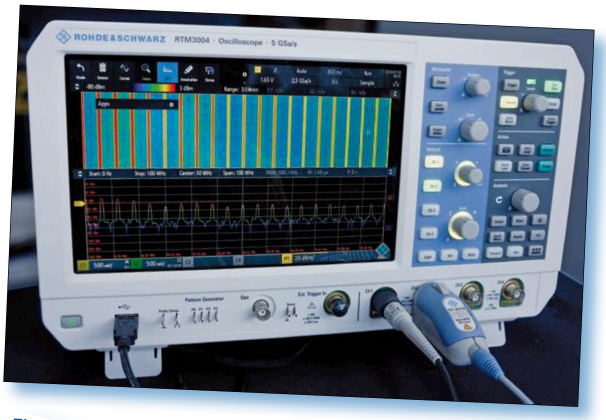


Figure 6. Rohde & Schwarz RTM3004 oscilloscope.

Figure 5. Rigol DP832 lab power supply.

Figure 7. ICOM IC-8600 SDR receiver, covering from 10 kHz up to 3 GHz.

Figure 8. Test setup on the German amateur radio club's EMC working group for measuring interference from LED lamps.

Figure 9. Interference level when LED bulb no. 2 from the March & April 2018 issue of Elektor Magazine was not switched on.

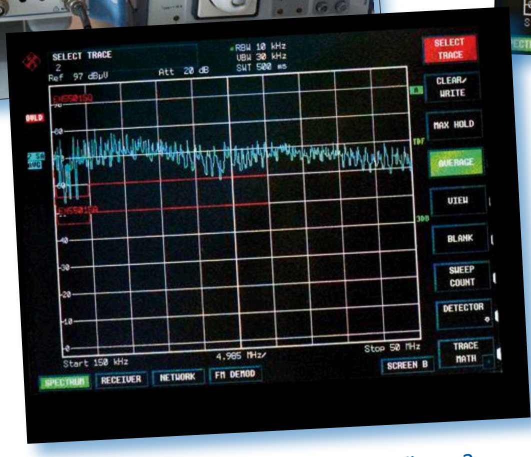


Figure 10. Interference level with LED bulb no. 2 illuminated.

between 10 kHz and 3 GHz, making it possible to both hear and display signals. This decodes various digital transmission protocols including P25 (Phase 1), NXDN, dPMR, D-STAR, and DCR (Digital Convenience Radio) and demodulates conventional signals such as USB, LSB, FSK, CW, AM, S-AM (synchronous AM), FM and WFM in tuning steps of 1 Hz minimum.

EMC test on LED lamps

The DARC's EMC working group were showing a test setup based around LED light bulb no. 2 (**Figure 8**) seen in the *Elektor Magazine* article 'EMC interference from LED lamps' [4], which discussed the EMC interference caused by LED lamps in detail. More than a dozen types were measured at the event. The noise level with LED lamp no. 2 switched off can be seen in **Figure 9**, whilst **Figure 10** shows the higher level with the LED lamp turned on.

In its June 2018 issue, the DARC's club magazine *CQ DL* invited readers to send suspect LED bulbs to the Elektor. The German Federal Communications Agency, as the competent authority of market supervision, has welcomed the call for submission of suspicious examples. Elektor Labs will now examine the bulbs submitted and publish an update if necessary, as well as informing the agency about the test results.

Back in September 2017, a DARC press release pointed to the evidently growing level of interference to radio communication, other radio services and DAB radio reception caused by non-EMC-compliant LED bulbs and thereby set the ball rolling.

For tinkerers

The Bodensee Maker Faire, which took place at the same time, had around 150 booths with inspired inventiveness and creative ideas. Starting with simple craftsmanship all to way to technically ambitious projects, the fifth Maker Faire beside Lake Constance embraced everything to beguile creative minds and hobby-tinkerers of all ages. The emphasis was on participation, not just watching. Accordingly, a number of improvised workshops (**Figure 11**) were arranged, inviting visitors to 'do it yourself'. Children especially found great pleasure in having hands-on access to technical activities.

Figure 11. Improvised workshop at the Bodensee Maker Faire.

Modifying and repurposing

The year-round rivalry of the case modding scene enabled visitors to view computer cases in a new light (**Figure 12**). Case modding is a discipline that focuses on changing the appearance of the PC in order to enhance it visually. Thought-provoking materials are used, such as wood, and the processing methods place no limits on the imagination. The craftsmanship level was remarkably high in the exhibits shown.

Another big thing at the Bodensee Maker Faire is the steampunk contingent. This is a phenomenon that might seem bizarre. Stemming originally from a literary movement that first appeared in the 1980s, it has developed into a genre of great artistry. Steampunk style combines elements and futuristic technical functions that are entirely modern with resources and materials of the Victorian era, thereby achieving an unashamedly retro-look for the technology created (**Figure 13**). Steampunk thus falls within the scope of the so-called retro-futurism, in other words a view of the future, as it might have been created in earlier times [5]. The artworks exhibited are normally fully functional.

Figure 12. Case modding taken to the ultimate.

Figure 13. Steampunk exuberance.

Hunters and collectors

The flea market at HAM RADIO offered not only components of all sorts in vast numbers, but also devices that seemed to come from another era (**Figure 14**). Old radios were waiting for new owners, perhaps over several HAM RADIO shows (**Figure 15**)! Some of the treasures discovered might even have escaped from a museum (**Figure 16**). But anyone who believes that the flea market is merely an antique fair is mistaken. There were also rare components on sale, for instance microwave radio parts. Also to be found in the flea market were many dealers with interesting components, sub-assemblies or other amateur radio goodies and related stuff. For genuine bargain hunters the flea market is always the first port of call at HAM RADIO, to be reached as early as possible. You soon find out how you can discover 'must have' items here that you never even knew existed and have to buy instantly.

Figure 14. Moving-coil instruments.

Figure 15. A radio rig offered in the flea market.

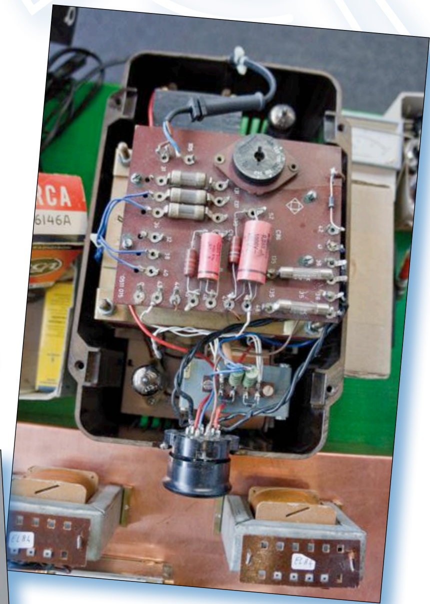


Figure 16. Vintage hardware in the flea market section.

Same place, (not) same time

The next HAM RADIO event will be held again at Friedrichshafen but this time it will be from 21-23 June 2019, meaning it reverts to its usual time slot (late June). The Bodensee Maker Faire takes place on the same festive weekend. ⏮

(180212-05)

Weblinks

[1] www.amsat-dl.org/index.php

[2] www.darc.de/home/

[3] www.darc.de/nachrichten/radio-darc/

[4] www.elektormagazine.de/160610

[5] http://de.wikipedia.org/wiki/Steampunk

Elektor Store Highlights

Electronics and the Senses

Once again we aim our spotlight on a few items you can buy in the Elektor Store.

By ElektorLabs

Waterun F800 Fume Extractor

Even as a hobbyist you should be just as concerned about your health as about the nasty smells your housemates associate with your "scientific" activity in the basement or circuit cellar. Three ways exist basically to get rid of the smells you love but others hate.

1. Place a small ventilator close to your work area. Like 12 volts and/or pulled from a PC. This will pull the fumes across a distance of about 10-15 cms so at least it does not get in your eyes and cause irritation (sensitivity differs between people). But in your room… it will remain, as will the smell, just weakened. Not recommend even if you can open a window very close by.
2. *Ditto* with the ventilator in place but connected to a dryer hose to exhaust the polluted air preferably to outdoors. Cheap.
3. A device that not only removes the polluted air from your workspace but also cleans it, i.e. catches the offending particles in a filter. Provided a strong cleaning activity is guaranteed by the filter (like 99%) the cleaned air can be exhausted to the workspace.

@ WWW.ELEKTOR.COM

→ Waterun F800 Fume Extractor
www.elektor.com/f800-fume-extractor

The Chinese-made Waterun F800 falls into category 2. It's a tabletop device with a weight of just under 5 kg and a footprint of 500 x 270 mm at a height of 300 mm. It can displace 110 m^3 of air per hour at a filter efficiency of 99.7%. It consumes 80 W of power and is reasonably quiet at under 59 dB. The vapours first pass through a white dust filter and then through a carbon filter. The clean air is blown out through a grille at the back. The carbon filter can be replaced if necessary — it is in any case easily accessible.
The device can be used in two ways:

- Horizontal extraction: The F800 is mounted on the workbench, at a maximum distance of 30 cm from the workpiece. The 290 x 45 mm intake opening effectively extracts vapours from a small, flat-fitting workpiece. However, this method is less effective for larger workpieces.
- Extraction via a flexible hose: The F800 can be mounted on the workbench. The supplied hose (90 cm long) is clicked into the intake opening. Thanks to the clever construction, the trunk remains in any position, so that its opening can be placed neatly above the soldering spot.

Especially thanks to its flexible hose, the F800 is a semi-professional extractor that will not look out of place on the workbench of the frequent solderer!

Nanosound DAC Pro

Numerous DACs are available for the Raspberry Pi: digital-to-analogue converters that allow you to play high-quality audio on your Pi. It's always interesting to come across a DAC that's a little different. Nanomesher's NanoSound DAC Pro clearly distinguishes itself.
Like many DACs, the Nanosound DAC pro comes as a HAT expansion board for the Raspberry Pi, and fits snugly on the

@ WWW.ELEKTOR.COM

→ Nanosound DAC Pro
www.elektor.com/nanosound

GPIO pins. This HAT covers the entire Raspberry Pi (B+, 2, 3 ...) and employs the USB and Ethernet ports. This is necessary because it contains more options, namely a small LCD screen and some 'physical' buttons. These buttons and the screen are some of the most interesting parts of this solution. They enable the Pi and DAC to be used together as a music player, which you can then connect to your favourite speakers. There's even a 3D printed housing, for which you can download the design files for free. It all works with Volumio, an open source music player that runs on the Pi and is optimised to play music at the highest possible quality.
With a few adjustments you can have Volumio accept entries from the DAC buttons and display information about the song currently playing on the screen. The kit even comes with a remote control that allows you to control the system as well.
In addition to the fully equipped Pro version, there is also a cheaper Basic version without a screen if you don't need it. This version comes with the buttons and the remote control, which are more essential than the screen.
In short, the NanoSound DAC, in both Pro and Basic versions, is a great kit if you want to make a music player! It's a fun all-in-one music solution for your Raspberry Pi, adding everything you need (except speakers). And it sounds good too!

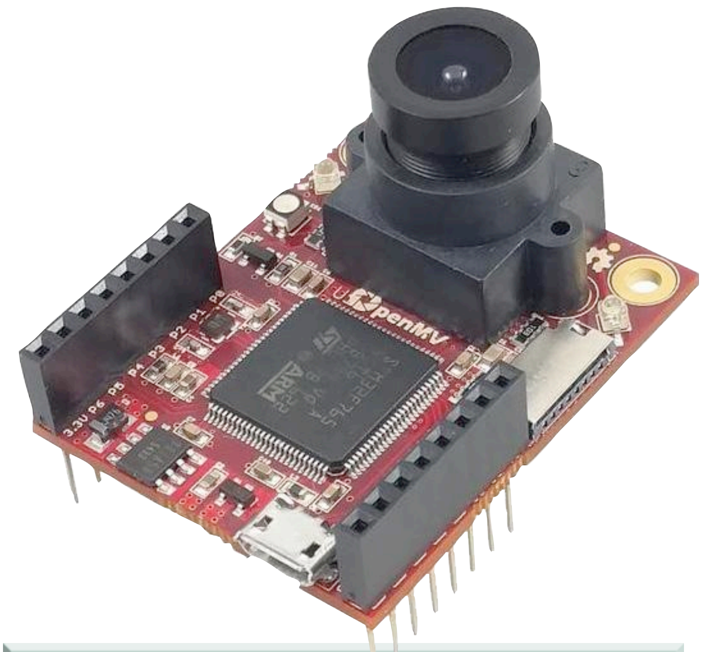
OpenMV Cam

The OpenMV Cam is a small and low-power microcontroller board that allows you to build machine vision applications quickly and easily. The OpenMV Cam is programmed with Python scripts (thanks to the MicroPython operating system); C/C++ is not required. This makes it easier to process the complex output of machine vision algorithms, and to handle high-level data structures. Nevertheless, in Python you have full control over the OpenMV Cam and its I/O pins. It's a piece of cake to trigger photos and videos on external events.
The board is equipped with an STM32F765VI ARM Cortex M7 processor running at 216 MHz and providing 512 KB of RAM en 2 MB of flash memory. The I/O pins carry 3.3 V but tolerate 5 V. The processor has the following I/O interfaces:

- Full speed USB (12 Mb/s) for interfacing with your computer. After connection, the OpenMV Cam will appear as a virtual COM port and USB Flash drive.
- A microSD card socket that supports 100 Mb/s write/read operations that allow the OpenMV Cam to easily record video.
- An SPI bus (up to 54 Mb/s) that lets you stream fine video data to the LCD display, WiFi display, or other microcontroller.
- An I²C, CAN and asynchronous serial bus (TX/RX) for interfacing with other microcontrollers and sensors.
- A 12-bit ADC and ditto DAC.
- Three I/O pins for servo control.
- Interrupt and PWM support on all I/O pins.
- One RGB LED and two powerful 850-nm IR LEDs.

The OV7725 image sensor can record 640 x 480 pixel, 8-bit grayscale images or 640 x 480 pixel, 16-bit RGB565 images at 60 frames per second (at a resolution of 320 x 240 or higher) and 120 frames per second at a resolution of less than 320 x 240.
The OpenMV Cam comes standard with a 2.8-mm lens in a standard M12 mount; for specialist applications, you can purchase and mount other lenses.

(180292-02)



@ WWW.ELEKTOR.COM

→ OpenMV Cam
www.elektor.com/openmv-cam-m7

PWM Digital to Analogue Converter

An IC with a PWM input and an analogue output

By **Tam Hanna** (Slovakia)

Pulsewidth modulation (PWM) is a simple method for creating analogue signals using purely digital means. When the duty cycle of a squarewave signal changes, its average voltage changes in proportion. Now, a PWM signal requires just a single digital output on an IC and so it is hardly surprising that these days the digital I/O pins of microcontrollers often come with extensive PWM capabilities. However, there are also downsides: for example, DACs are faster and more accurate. But now an IC is available that offers a simple upgrade to a PWM-based design.

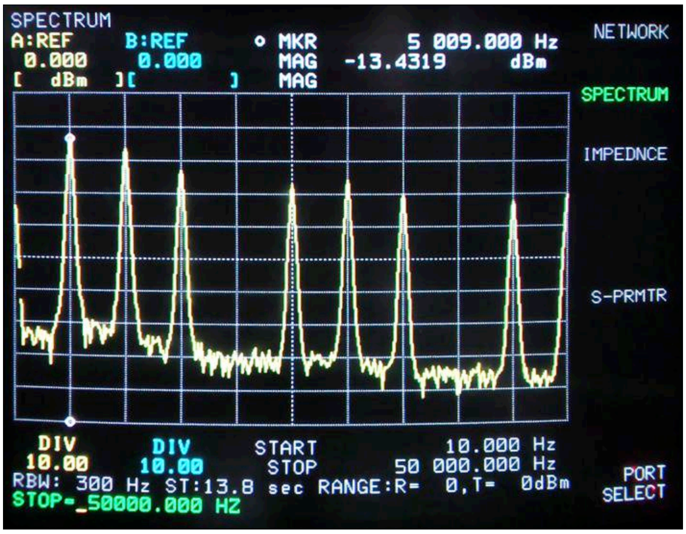


Figure 1. Spectrum of a 5-kHz PWM signal with a duty cycle of 25%.

Table 1. Ripple measurements.

Duty cycle (%)	4.4 kHz low-pass filter		2.2 kHz low-pass filter	
	Ripple (V_{pp})	DC (V)	Ripple (V_{pp})	DC (V)
10	0.95	0.11	0.52	0.14
20	1.38	0.30	0.84	0.34
30	1.62	0.49	1.02	0.55
40	1.78	0.76	1.13	0.67
50	1.82	0.96	1.16	0.90
60	1.77	1.16	1.13	1.09
70	1.64	1.35	1.01	1.29
80	1.38	1.54	0.82	1.49
90	0.94	1.73	0.53	1.68

As we mentioned, a PWM signal is just a squarewave whose duty cycle is modulated, and the effective voltage can be calculated from the peak voltage output and the duty cycle. In order to obtain a half-way decent analogue output, it is usual to insert a low-pass filter circuit after the PWM output. It is easy to see that the cutoff frequency of this filter should be low in comparison to the underlying frequency of the PWM signal, and as we lower this cutoff frequency the response of the output becomes more sluggish. The residual ripple depends on the filter characteristics, not just on the ratio between the PWM frequency and the cutoff frequency, but also on the sharpness of the filter response. These relationships derive from the fundamental principles of electronics.

Filtered PWM

In order to study these effects more closely, it is worth taking a look at the spectrum of the PWM signal. **Figure 1** shows the spectrum of an unfiltered PWM signal with a basic frequency of 5 kHz and a duty cycle of 25%. As you can see, the harmonics at multiples of the basic frequency gradually fall off in amplitude at higher frequencies. It is also worth noting that at this particular duty cycle the harmonics at 20 kHz and 40 kHz completely disappear: this is not the case for other values of duty cycle. At extreme values of duty cycle, for example 1% or 99%, where the signal consists of very narrow positive- or negative-going spikes, the spectrum consists of a series of low-amplitude peaks all of approximately the same height.

Depending on the ratio of its cutoff frequency to the PWM frequency, and on the sharpness of its response, a low-pass filter will pass part of the spectrum of the PWM signal and attenuate certain higher spectral components. As a result the residual ripple on the output signal will depend on the characteristics of the filter and on the PWM duty cycle. To give an idea of the extent of this effect, **Table 1** shows some measured results using two ordinary RC low-pass filters with a rolloff of 6 dB/octave and cutoff frequencies of 4.4 kHz and 2.2 kHz. The 'rip-

ple' figure is the peak divergence of the output signal from the median value of the DC level generated. With a duty cycle of 100% the DC output level should in theory be 2.0 V.
You might think that by using a filter with a sharper cutoff we could obtain a clear improvement, and indeed that is indeed the case, at least for some values of 'clear'. Using a 12-dB/octave filter improves the ripple at 2.2 kHz and a 50% duty cycle to just 0.50 V_{pp} and the DC level to 0.92 V. Of course it is possible to play around with Bessel and Chebyshev filters and the like, but it is questionable whether this is really worth the effort. All those extra components cost money, and, so important these days, real estate on your printed circuit board. On the other hand, if we significantly reduce the cutoff frequency of the filter, the sluggishness of its response prevents us making rapid changes to the DC output value.

A DAC is better

What we learn from the above discussion is that a real digital-to-analogue converter is often a better choice. This applies in many situations even though strictly speaking an 8-bit DAC is less precise than the 16-bit PWM signal offered by many microcontrollers (albeit at low frequencies). With a resolution of eight bits and a maximum output level of 2 V, for example, we are looking at errors of only around 8 mV.
It is usually easy to add a 'real' DAC to a microcontroller circuit. Devices are available with parallel interfaces, or, if pins are at a premium, serial interfaces. However, in many situations the requirement will be to revise and improve an existing solution without wholesale modifications to the program code. Here we might consider the possibility of converting a PWM signal purely digitally into a DC signal generated by a DAC. In that way the software can be left completely untouched.
How do we go about this? Linear Technology (now part of Analog Devices) has announced a complete family of converter devices with part numbers LTC2644 and LTC2645 [1]. **Table 2** shows the differences between the various members of the family. There are parts with 8-, 10- and 12-bit resolution, each with two or four integrated converters. The basic idea is 'PWM in, DC out'. Driving the device requires a single pin, which is even more economical and more direct than using an I^2C or SPI bus.
Doubtless many brows will be furrowed by this point as readers wonder how this trick is pulled off. Although the exact interior structure of the chip is not published, the block diagram (see **Figure 2**) lets us make a few inferences and educated guesses. Essentially the device consists of a microcontroller with integrated DACs running dedicated firmware. A counter counts clocks for the duration of the incoming pulse and for the duration of the gap between the pulses, enabling an estimate of the duty cycle to be computed. If this number changes, then the value sent to the integrated DACs is changed correspondingly. The DC voltage thus directly represents the duty cycle of the PWM signal: the process is illustrated in **Figure 3**. We only need to add filter capacitors and the like if the nature of the signal we are trying to generate demands them.
So all very simple: we just need to get hold of the right IC and we are set. There is just one (very tiny) fly in the ointment: the devices are only available in MSOP packages. Soldering sixteen pins on a package measuring 4 mm by 5 mm is not for the faint of heart!

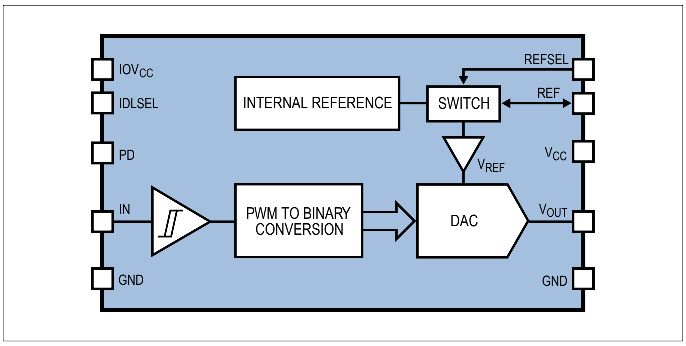


Figure 2. Block diagram of the PWM DAC IC (source: Linear Technology).

Table 2. The device family.

Part number	Number of DACs	Resolution (bits)	Settling time (µs)	Price/1Ku ($)
LTC2644-8	2	8	7.0	2.45
LTC2644-10	2	10	7.4	2.60
LTC2644-12	2	12	7.8	2.75
LTC2645-8	4	8	7.0	2.55
LTC2645-10	4	10	7.4	3.25
LTC2645-12	4	12	7.8	3.95

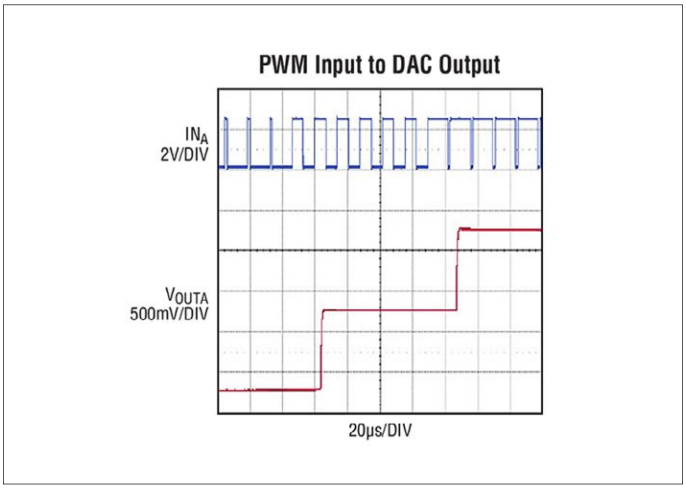


Figure 3. How the LTC2644/5 converts a PWM signal to an analogue voltage (source: Linear Technology).

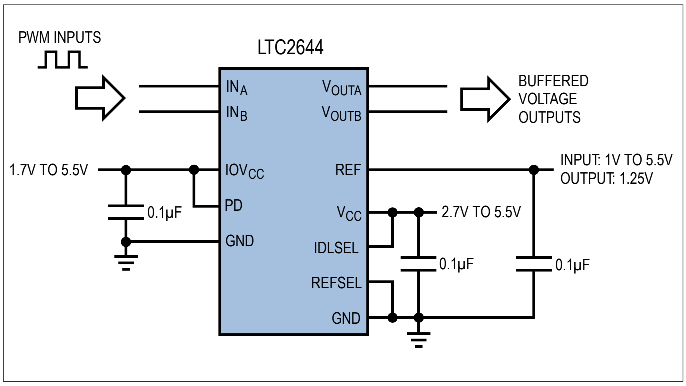


Figure 4. Basic circuit using a LTC2644 (source: Linear Technology).

Table 3. PWM.		
Resolution (bits)	Maximum non-linearity (LSBs)	Maximum PWM frequency (kHz)
8	0.5	100.00
10	1.0	25.00
12	2.5	6.25

Hardware

The basic circuit of the two-channel version of the converter shown in **Figure 4** shows that connecting up the device is not complicated. At the upper left are the two PWM outputs from the microcontroller and at the upper right the two buffered DC outputs. In addition there is a reference input/output pin and separate power supply pins for the digital side of the circuit and for the analogue outputs. Note that the use of a 12-bit DAC demands special attention be paid to the design and construction of the circuit: not only do we need power supply decoupling, but we must also ensure that the circuit board is clean and free of flux residues, as otherwise leakage currents can reduce performance.

When it comes to the input side of the device the performance is nothing spectacular: the datasheet says that the PWM frequency can be between 30 Hz and an upper limit that depends on the desired output resolution: see **Table 3**. Logic levels on the PWM signal inputs are determined by IOV_{CC}, which can be at most 5.5 V.

We should mention a couple of features of the special-function pins. Linear Technology is known for its voltage references, and this IC family offers a built-in reference with a voltage of 1.25 V and a maximum error of 10 mV, with a temperature drift of just 10 ppm/K. If the REFSEL pin is taken logic low then the internal reference voltage will appear on the REF pin, where it should be decoupled using a capacitor with a value of at least 100 nF (and ideally 10 µF) to ground. The analogue output voltage will then be able to swing up to a maximum of 2.5 V, as the voltage is in effect doubled in the DAC. If, on the other hand, REFSEL is taken logic high, then the REF pin becomes an input, to which a voltage between 1 V and V_{CC} (maximum 5.5 V) can be applied. In this case there is no effective doubling of the reference voltage.

Trying it out

A simple way to experiment with the PWM DAC chip is to connect it up to a PWM signal produced by a function generator, and indeed this is the first thing I tried. However, it is more interesting to connect it to a microcontroller, as that is the configuration for which it is really intended. And what better digital Swiss Army knife to use for testing than an Arduino?

The PWM facilities offered by the Arduino programming environment are not particularly extensive. If you want to realise a wider range of possibilities you will need to talk directly to the hardware module on the AVR microcontroller. At [3] you will find a helper function that we will use in the examples below without further explanation. The basic test program is straightforward: first comes the set-up routine, which configures as outputs both pin 9 (which is driven by the PWM hardware itself) and pin 11, which we will use to trigger an oscilloscope.

```
void setup() {
     pinMode(9, OUTPUT);
     setPwmFrequency(9, 8);
     pinMode(11, OUTPUT);
}
```

In the main loop we alternately generate a very ‘weak’ PWM signal and a very ‘strong’ PWM signal. We do not quite go to the minimum and maximum values permissible, as those would trigger a special mode in the converter which we would like to avoid for now.

```
void loop() {
    analogWrite(9, 1);
    digitalWrite(11, LOW);
    delay(50);
    analogWrite(9, 254);
    digitalWrite(11, HIGH);
    delay(50);
}
```

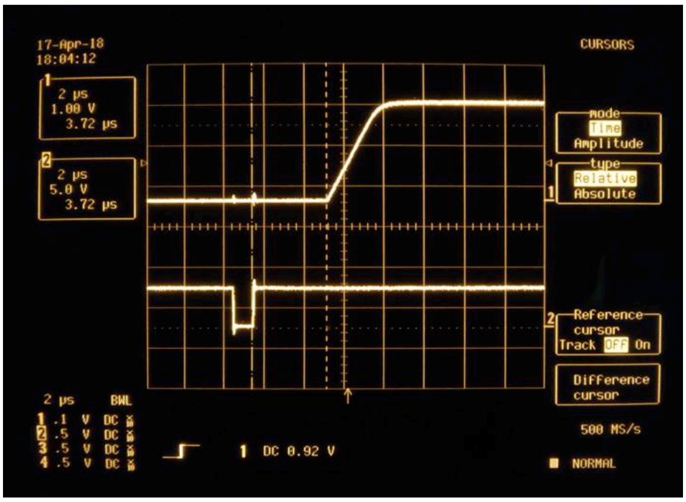


Figure 5. Rising edge at the output of the DAC.

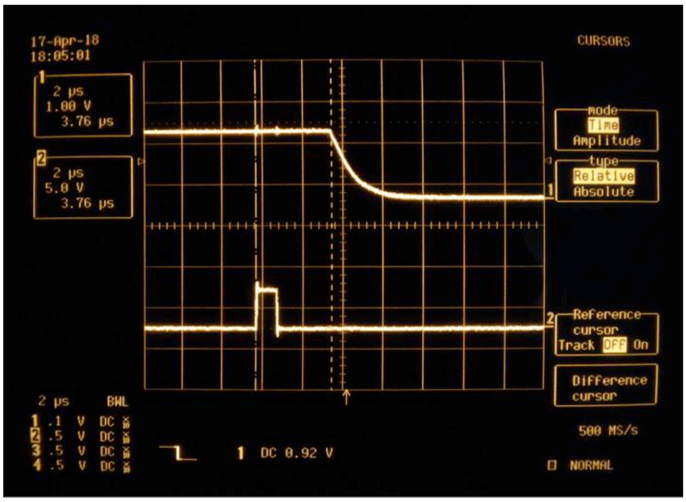


Figure 6. Falling edge at the output of the DAC.

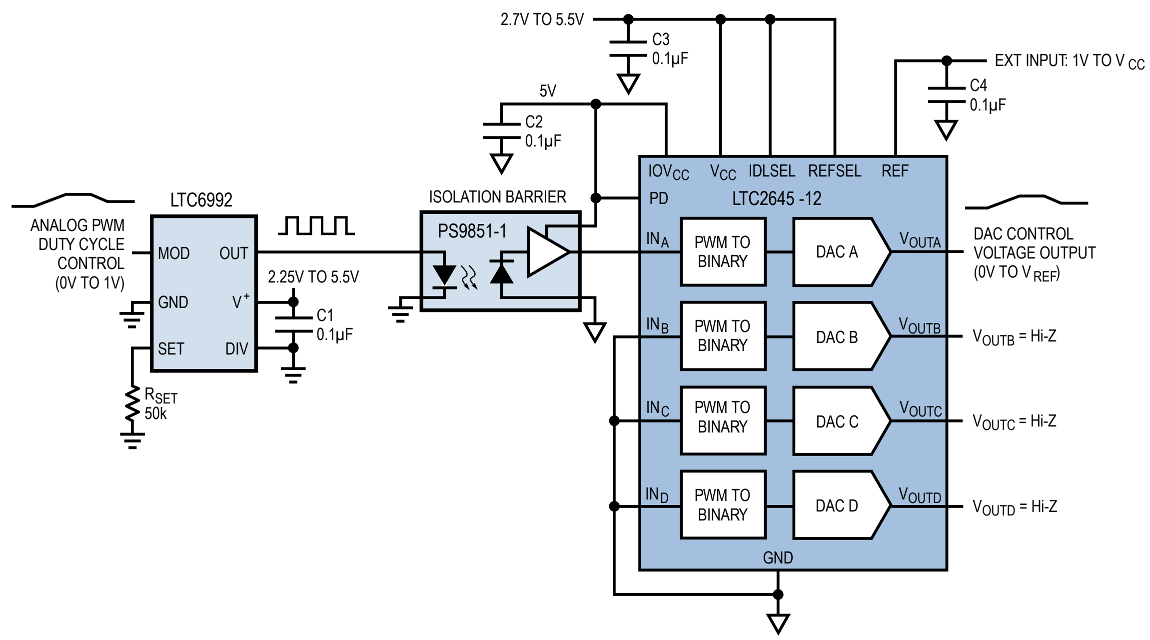


Figure 7. Linear and galvanically isolated transmission of an analogue signal using a low-cost optocoupler (source: Linear Technology).

If you now connect up an oscilloscope you will see an unfortunate effect: the call to the analogWrite function does not necessarily reset the PWM counter, and the practical consequence of this is that there is a noticeable jitter between the rising edge on pin 11 and the change in PWM duty cycle. To work around this problem we can trigger on the output channel instead. Then we observe the behaviour shown in **Figure 5** and **Figure 6**: persistence mode is enabled for this oscilloscope trace, which clearly demonstrates that the response of the device is very consistent.

... and more besides

The behaviour of the device in the presence of a PWM signal is easy to understand. More interesting is what happens at power-up or if no PWM signal is applied for a period of 60 ms or more. In these cases the converter goes into 'idle' mode, with behaviour determined by the logic level on the IDLSEL pin. The IC's datasheet goes into detail about exactly what configuration options are available.

Last but not least let us look at an interesting application of the device, which gives a hint as to the possible uses that the manufacturer had in mind when designing it. **Figure 7** shows an example circuit from the datasheet in which an analogue value is transmitted across an isolation barrier using an optocoupler. To avoid the use of an expensive linear optocoupler, the analogue signal is first converted to PWM, which is then passed through a low-cost universal optocoupler and then converted back to analogue. This idea works very well in practice: for example, multimeters from Solartron employ a similar technique.

The LTC2644 and LTC2645-series ICs are not just suited to upgrading existing systems. They can also be used to reduce traffic on system buses, for example. Convert a high-resolution PWM output to a high-resolution DAC output, and the microcontroller's pins that are freed up can be used to drive a display or for any other purpose. ◄

(180342-02)

Web Links

[1] LTC2644: www.analog.com/en/products/digital-to-analog-converters/special-function-dac/ltc2644.html

[2] LTC2645: www.analog.com/en/products/digital-to-analog-converters/special-function-dac/ltc2645.html

[3] PWM helper function: https://playground.arduino.cc/Code/PwmFrequency

HOMELAB PROJECT

Self-calibrating Frequency Meter

No alignment necessary

By **Willem den Hollander** (The Netherlands)

In the May & June issue of ElektorLabs Magazine we described a 10-MHz reference frequency source that uses the signals from GPS satellites to provide an extremely accurate reference frequency. The author has designed a self-calibrating frequency counter based on that circuit.

Features

- Self-calibrating with GPS signal
- Input frequency range 5 Hz to 80 MHz
- Input signal amplitude 0.2 to 30 V
- Resolution 6, 7 or 8 digits
- Accuracy (after locking) ±1 count pulse

There are various ways to measure signal frequencies. They all have one thing in common: they require a stable and precisely known reference frequency. A suitable reference frequency source was described in the May& June 2018 issue of *ElektorLabs Magazine* [1]. Here we use it as the basis for a full-fledged frequency counter with self-calibration capability.

A brief history

The 10-MHz reference frequency generator uses a precise 20-MHz voltage controlled temperature compensated crystal oscillator (VCTCXO) that is locked to the 1 pps output signal from a GPS receiver module. Under ideal conditions, the output signal has an accuracy of 1 part in 10^{10}. A commercial reference frequency generator with comparable accuracy is a lot more expensive than our DIY version.

Reciprocal measurement

The easiest (and conventional) way to measure signal frequency is to count the number of cycles of the input signal during a precisely defined 1-second time window. The uncertainty with this measurement method is ±1 digit. That corresponds to an error of ±1 Hz, which is of course unacceptable for measurement of low frequencies.

Another measurement method became popular with the advent of microcontrollers: reciprocal frequency measurement. With this method, the input signal defines a time window with a duration that depends on the desired accuracy. Two counters run during the gate time of this window. The first counts the number of cycles of the input signal, while the second counts the number of cycles of the reference signal. The frequency of the input signal can then be determined from:

$$frequency = \frac{input\ pulse\ count}{(reference\ pulse\ count) \times t_c}$$

Here t_c is the period of the reference frequency, which is equal to $1/f_{ref}$. Assuming a reference frequency of 10 MHz, the resolution of the frequency measurement is 6, 7 or 8 digits with a measurement window (gate time) of 0.1, 1 or 10 seconds, respectively.

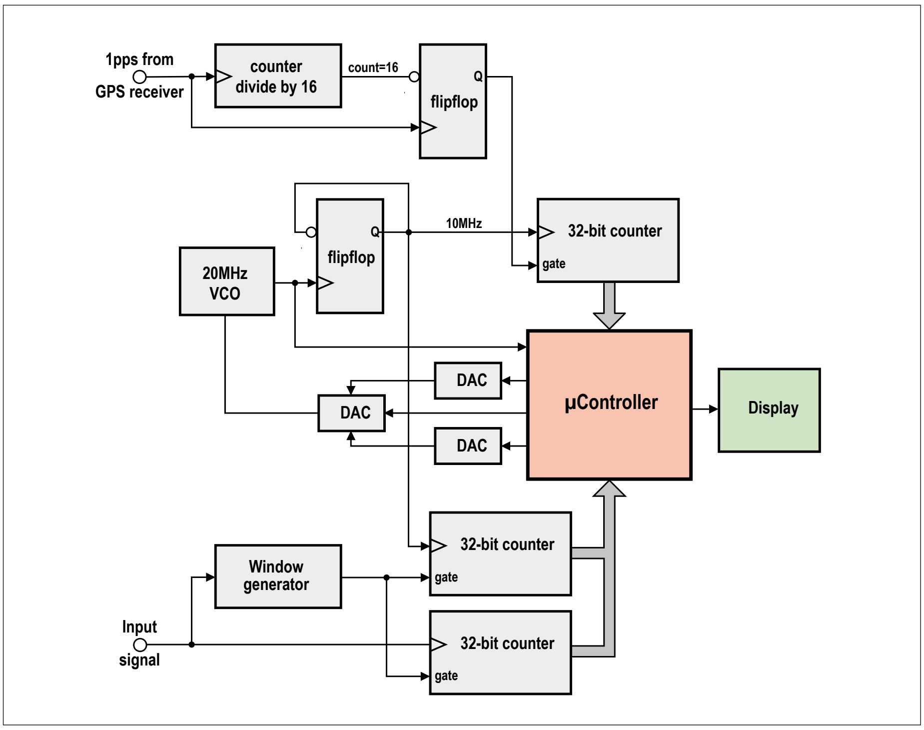


Figure 1. Block diagram of the frequency counter. Except for the display, everything shown here is located in the microcontroller.

Block diagram

The block diagram of the digital portion of the frequency counter is shown in **Figure 1**. The counter essentially consists of the gate timing logic and the two subsequent 32-bit counters, along with part of the microcontroller. The other blocks are responsible for generating the reference frequency. For more information about that, please consult the article in the May & June issue [1]. It's worth noting that with the exception of the VCTCXO (marked '20 MHz VCO' in the block diagram), everything shown on the block diagram is integrated into the microcontroller.

The maximum frequency that the microcontroller inputs can handle is 16 MHz, so the oscillator frequency is divided by 2 to obtain a 10-MHz signal. That is well within the specified capability of the microcontroller.

By the way, the microcontroller datasheet is not entirely clear on the maximum allowable frequency for the I/Os. Sometimes you think it is 16 MHz, and sometimes you think it is 20 MHz. That makes it anybody's guess. However, the author's prototype certainly did not have any problem with frequencies of 20 MHz or even higher, corresponding to 80 MHz or more at the input.

Input amplifier

As we all know, digital circuits are only happy with digital signals, so the input signal has to be conditioned before it is applied to the microcontroller port. **Figure 2** shows the relevant part of the circuit (top half). The display module is also shown here in the bottom half, because that part of the circuit is located on the display board. You might wonder why it is on the display board. The answer is that the circuit needs to be as close as possible to the input connector, because long wire leads should always be avoided with high-frequency analogue signals.

The signal from the BNC connector J1 is limited by the dual diode D1 and buffered by the FET T1. This input circuit is required to obtain an input impedance of 1 MΩ and to avoid overloading due to high-amplitude input signals.

Transistors T2 and T3 provide enough gain to allows the comparator IC1 to convert the analogue signal into a respectable digital signal. The amplifier does not need to be especially linear, since all we actually need is the timing of the rising and falling edges. Counter IC2 divides the signal frequency by a factor of 4 to avoid confronting the microcontroller input with frequencies too high for it to handle. The frequency range of the input circuit, from the BNC input to the counter output, is at least 5 Hz to 80 MHz. The amplitude of the input signal should be between 0.2 V and 30 V.

The display used is a DOG type from Electronic Assembly with two rows of six-

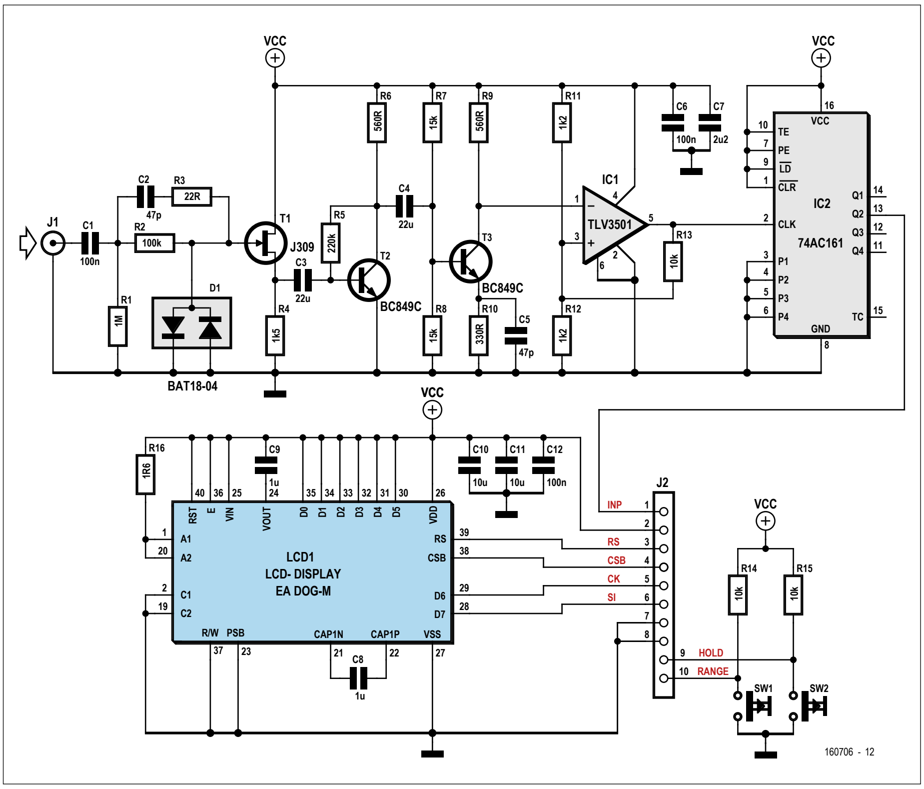


Figure 2. Schematic diagram of the input amplifier. It is shown on this diagram because it is located on the display board.

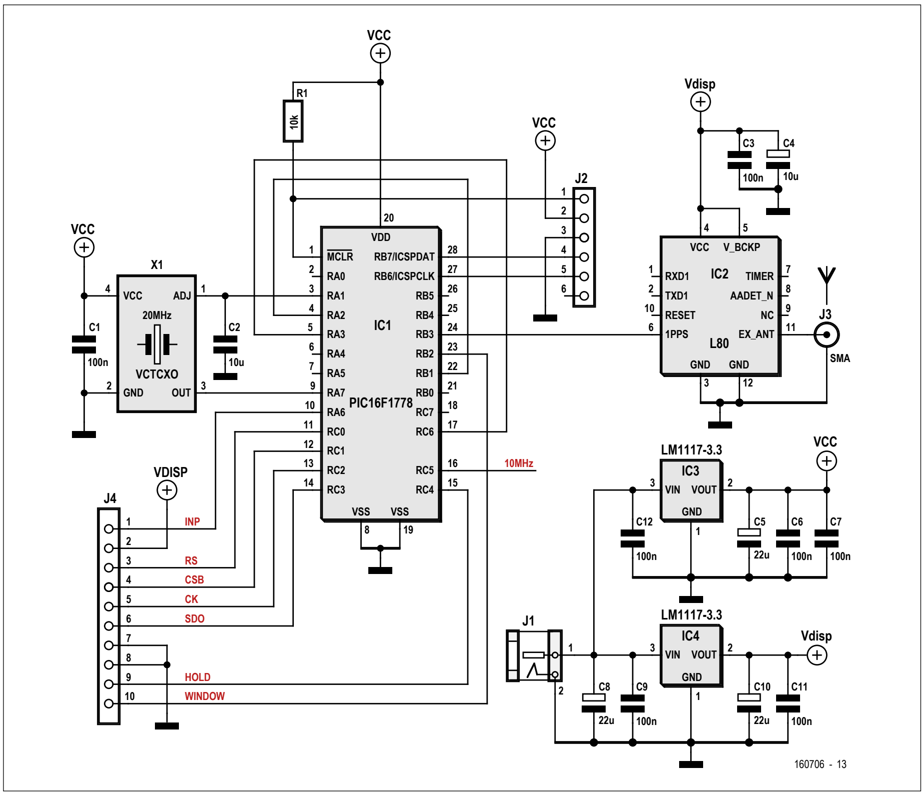


Figure 3. Schematic diagram of the microcontroller portion. There aren't many components, so the PCB is remarkably empty.

teen characters and white LED backlighting. The display is driven by the microcontroller over an SPI bus (connector J2).

Microcontroller portion

The microcontroller, the clock crystal and the GPS receiver are located on the main circuit board. The schematic diagram is shown in **Figure 3**. It is largely the same as the 10 MHz reference frequency project in the May & June issue. For more details, see the article in that issue. The only differences are:

- the two indicator LEDs are deleted (the microcontroller pins are needed for the SPI interface (J4);
- the buffered 10-MHz and 1-MHz outputs are also deleted because they are no longer necessary, but a 10-MHz signal is still available on the RC5 output of the microcontroller (pin 16);
- a second 3.3-V voltage regulator (IC4) has been added to provide the supply voltage for the display, the input amplifier and the GPS receiver. That way the VCTCXO and the microcontroller have their own power supply and the stability of the reference frequency is assured.

Connector J2 is intended for in-circuit programming of the microcontroller, and J4 provides the connection to the display board.

Construction

As previously mentioned, the input amplifier and the display are located on the display board. With the exception of the connectors and switches, which should preferably be leaded types because they are subject to physical stress, all components are SMDs. That keeps the overall package pleasantly compact (see **Figure 4**). It's a pity that for proper operation the GPS receiver has to be kept as far away from the oscillator as possible; otherwise it could have also been mounted on this circuit board.

Figure 5 shows the microcontroller board. It is very sparsely populated, but that is beneficial for the GPS receiver.

The boards are fitted in a Teko 011 case, with the original aluminium front and back panels replaced by acrylic sheets covered with aluminium-coloured self-adhesive film because the GPS receiver needs access to the outside world. The power source is a standard 5-V USB AC line adapter.

Figure 6 gives an impression of the author's prototype.

Firmware

The firmware residing in the microcontroller, which performs all the functions required for proper operation of the frequency counter, is written in assembly language because using a high-level language would have a number of serious disadvantages.

First of all, it would not be possible to control the timing of the various functions with sufficient accuracy, and secondly, it would take up more memory space. And the software would run slower – also

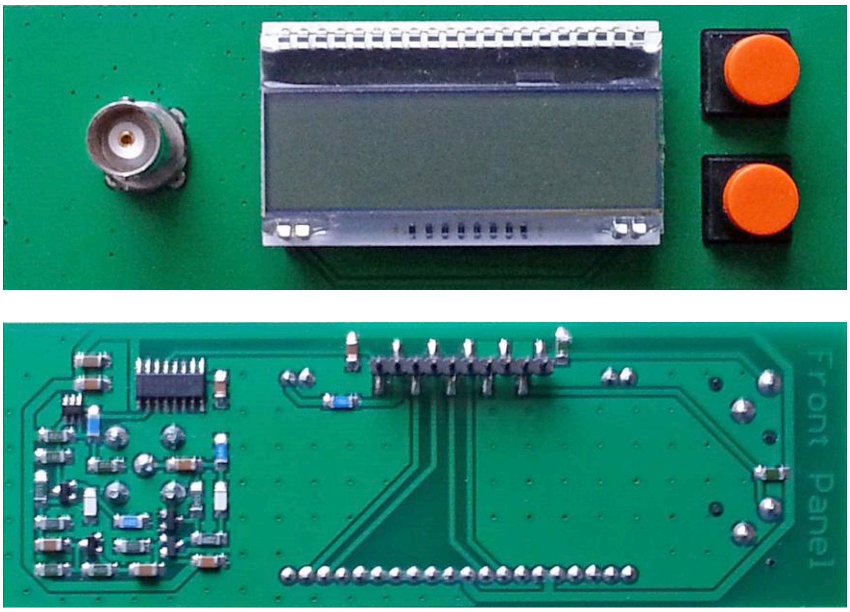


Figure 4. The display board with the input amplifier.

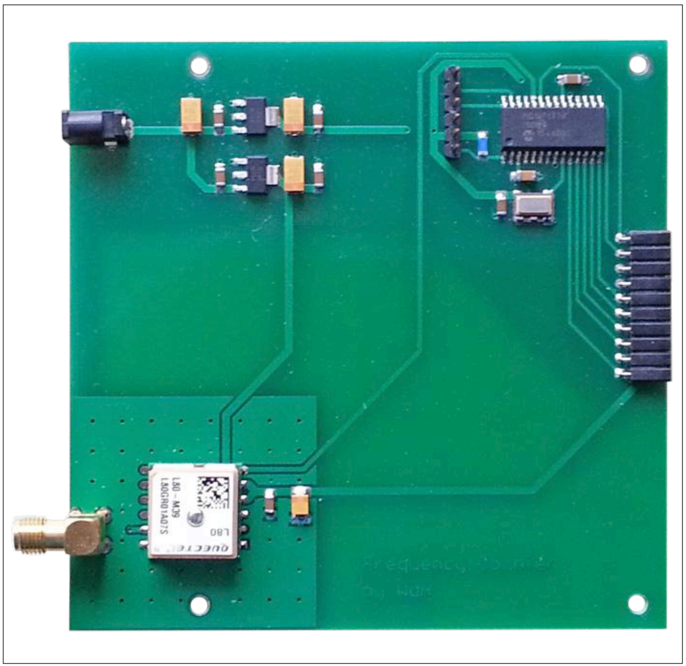


Figure 5. The microcontroller board.

Figure 6. The front and rear of the case.

Never calibrate again

a significant consideration. In particular, the multiplication and division operations that are necessary to calculate the frequency shown on the display are coded the way they need to be for this application. Multiplying the content of the 32-bit input counter by 10 and then dividing the result by the content of the second 32-bit counter takes less than 1 millisecond, including BCD conversion. All in all, the firmware in its present form occupies only 8% of the microcontroller memory. The firmware and the PCB layouts can be downloaded free of charge from the project page for this article [2].

Use

The frequency counter is easy to use. The duration of the measurement window (the gate period during which pulses are counted) can be selected with the Time button. That determines the resolution of the measurement (see **Figures 7a,b,c**). The selected window time is indicated by the number of dots in the bottom line of the display.

The small symbol at the left end of the top line appears for about 200 ms when a measurement has been completed and the display is updated (**Figures 7a and 7c**). The Hold button freezes the reading on the display (Figure 7d). Note that this only stops updating of the display; the counter keeps on making measurements.

The symbol at the left end of the bottom row (**Figures 7d and 7e**) indicates that the VCTCXO is locked to the GPS signal. But even if this symbol is not shown, accurate measurement is still assured for a relatively long time. Once the frequency counter has been locked to the GPS signal, the D/A converter settings are stored in the non-volatile memory of the microcontroller as described in the article in the May/June issue. Each time the counter is powered up, these settings are read from the memory and the converters are configured accordingly.

Of course, the frequency of the VCTCXO will gradually change over time — that is a normal ageing process. But as long as the oscillator is locked to the GPS signal at more or less regular intervals, the latest calibration factors for the D/A converters are always stored in memory. This way the accuracy of the frequency counter is guaranteed to be ±1 clock pulse. ⏮

180343-02

Web Links

[1] www.elektormagazine.com/160595

[2] www.elektormagazine.com/180343-01

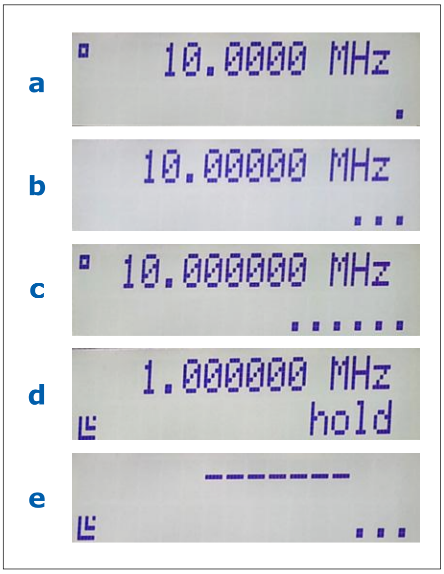


Figure 7. a: 0.1-s measurement window;
b: 1-s measurement window;
c: 10-s measurement window;
d: oscillator locked, readout frozen;
e: oscillator locked, no input signal.

@ WWW.ELEKTOR.COM

→ GPS board
www.elektor.com/gps-board-eb056

→ GPS antenna
www.elektor.com/trimble-gps-antenna

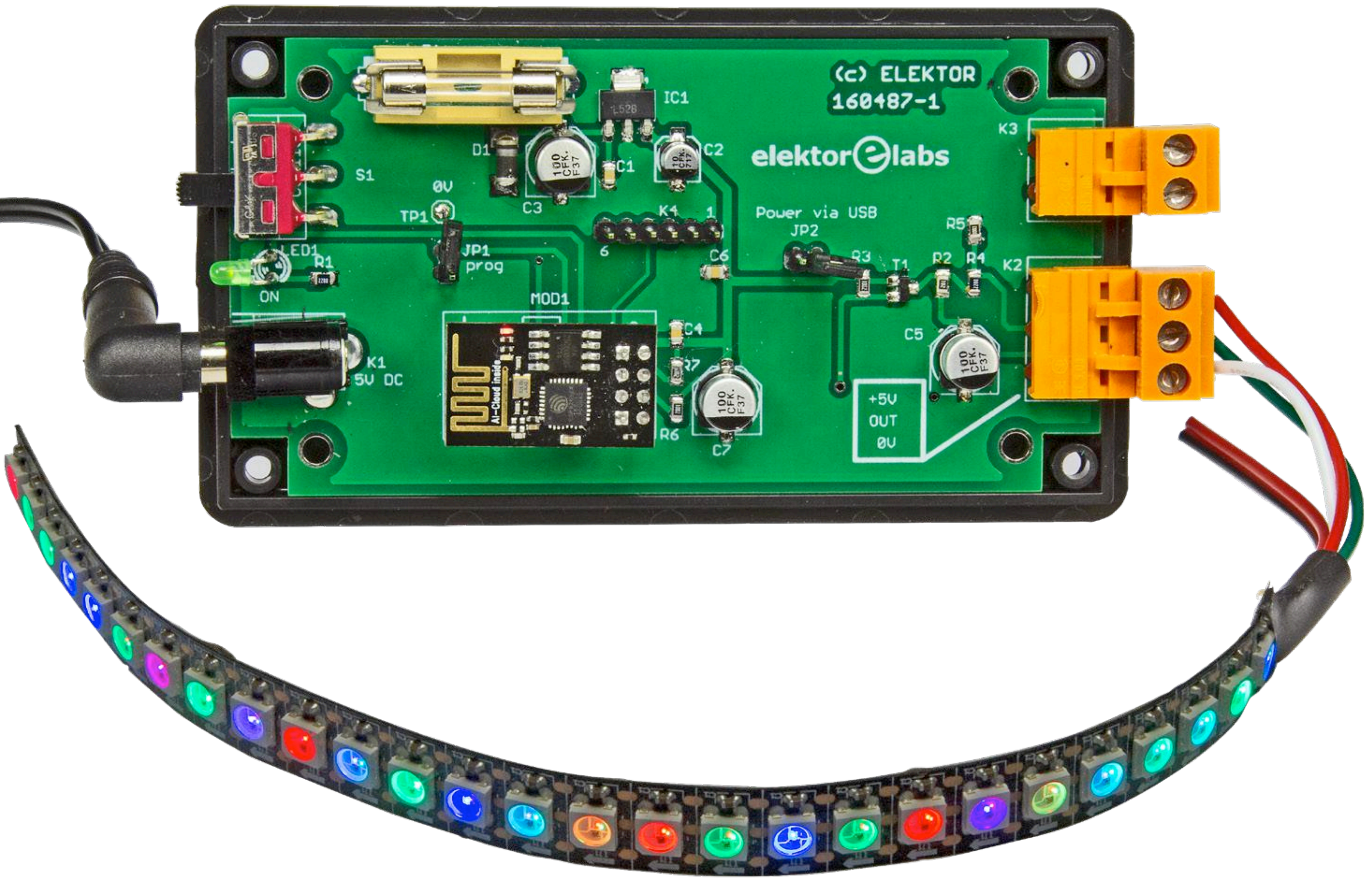
→ PIC multiprogrammer
www.elektor.com/pic-eb006

→ TL866A universal programmer
www.elektor.com/tl866a-programmer

ESP8266 Web Server for NeoPixel LED Strips

Driving WS2812 chips with the ESP-01

By **Pascal Rondane** and **Bastian Bouchardon** (France)



This circuit allows you to drive a strip of WS2812 RGB LEDs (also called NeoPixels [1]) via Wi-Fi. Using the module ESP-01 module (using the ESP8266 chip), the board becomes a Wi-Fi access point. You only have to open a web page on a smartphone to select one of 44 light effects and one of 5 brightness levels of the LEDs The length of the strip can be from one to five metres (15 ft.); in general, the number of LEDs per metre is 30, 60 or 140.

Bridge board

The circuit diagram (**Figure 1**) shows that the board is not very complicated. The ESP-01 module based on the ESP8266 (MOD1) does all the work, but there are a number of other components that help it get there.

First of all the power supply: you can connect a 5 V / 2 A supply to connector K1. Diode D1 is for protection: it will blow the fuse F1 if the power supply polarity is incorrect.. The WS2812 LEDs (also available in through hole format) consume at least 60 mA each at full intensity, so a 2 A power supply will suffice for about 30 LEDs (a 1-m strip). If you connect a 2-m strip, you must avoid putting all LEDs at full intensity at the same time. If you need more power (meaning more LEDs, more than 2 m), you can connect a beefier power supply on K3 instead of K1. In this case, don't forget to replace F1 with a higher rated one – you need one rated just over the par maximum current that will be needed). Whatever power supply you use, don't exceed 4 A on K1! See also the sidebar 'Now that's power!'

If you only need power for programming and debugging the ESP8266, don't worry about K1 and K3. In this case, fit jumper JP2 and connect a 3.3 V FTDI cable to K4. Note: the 5-V line on pin 3 will not take more than 50 mA, so disconnect K2 (which connects the LEDs) when you use K4 to power the circuit! Also important: disconnect the FTDI cable and JP2 before connecting a power source to K1 or K3!

The jumper JP1 must be in place when you power on if you want to program the ESP module, don't forget to remove it after programming is done. If JP1 is shorted when power is applied, the ESP8266 starts in programming mode and your application will not run.

Low dropout regulator IC1 supplies 3.3 V for the ESP-01, LED1 will be lit when this voltage is present. Filtering by capacitors C4, C6 and C7 avoids unwanted and untimely resets.

The function of transistor T1 is to convert the level of the output signal of the ESP (3.3 V, pin GPIO2) to the voltage level of the data input to the LED strip connected to K2, which is 5 V. Although this level change would work both ways, in this case the signal is only an output (that is to say, 3.3 V to 5 V).

During start-up, resistor R5 holds the serial output of the ESP8266 to a low level; during changing effects, decoupling capacitor C5 filters the current demand of the LED strip.

Even though it uses some SMD components, wiring of the printed circuit board should not pose any problems. The circuit is finally installed in a Hammond case.

Problems with level changing

Initially the circuit used a 2N7002 MOSFET with a pullup resistor of 10 kΩ connected to the 5-V and 3.3-V lines. It wasn't a good solution: as the NeoPixel data signal is a digital signal at 800 kHz, the level at the drain of the FET hardly reached the 3.5-V level needed by the LEDs for a high logic level, let alone the 5 V expected. Apparently the cutoff voltage of this FET (max. 2.5 V) is too high for this application, a BSS138 with one of 1.5 V works better (though still not perfect!).

Software

The Arduino sketch for this project uses (amongst others) the WS2812FX library, which contains 44 light effects which can be selected for the LED strip.

The length of the strip (i.e. the number of LEDs) is held on line 44 of the sketch, here 180 (strip 5 m long):

```
#define LED_COUNT 180
#define LED_COUNT 180
```

The sketch works even if this constant does not correspond to the actual num-

PROJECT INFORMATION

LED strip
web server
ESP8266
ESP-01
Wi-Fi

➡ **entry level**
intermediate level
expert level

1 hour approx.

Standard tools,
PC with Arduino IDE

£35 / €40 / $45

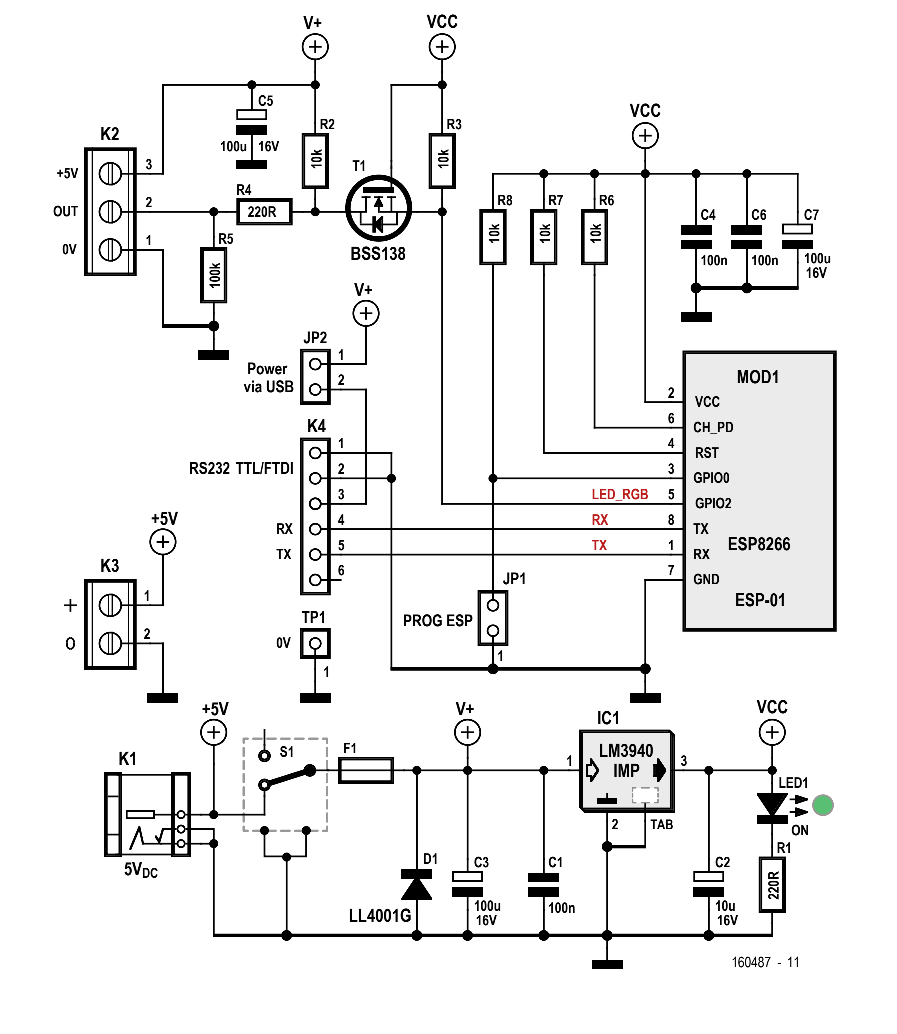


Figure 1. Circuit diagram of the board that physically connects the LED strip.

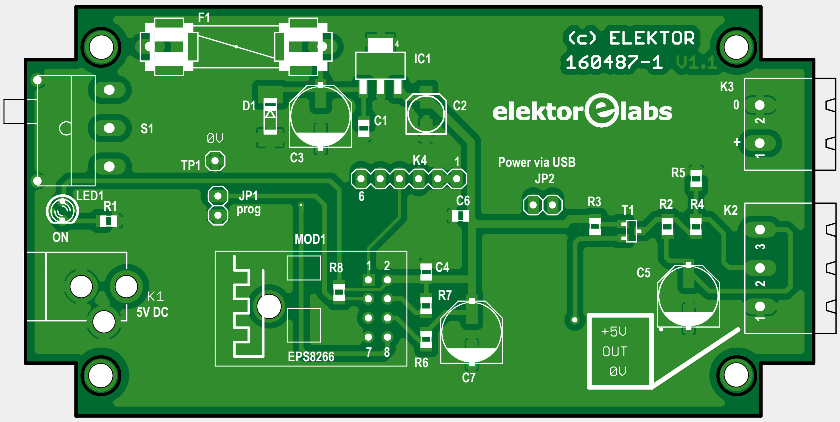
COMPONENT LIST

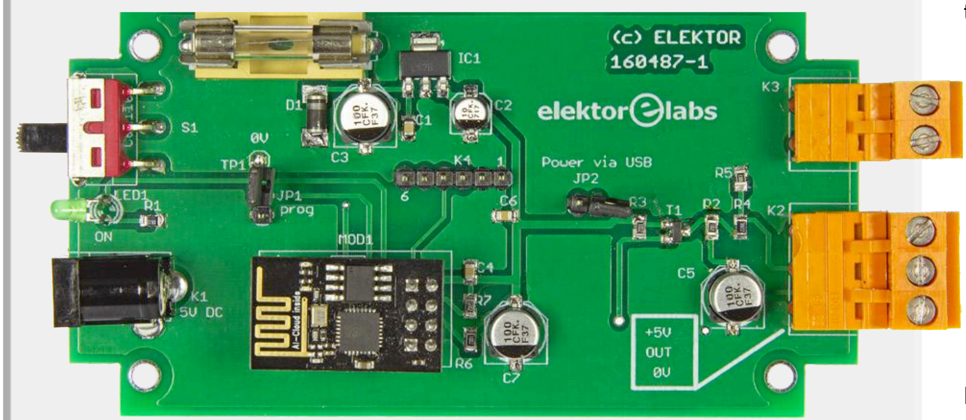
Resistors
R1,R4 = 220Ω, thick film, 5%, 0.1W, 150V
R2,R3,R6,R7,R8 = 10kΩ, thick film, 5%, 0.1W, 150V
R5 = 100kΩ, thick film, 5%, 0.1W, 150V

Capacitors
C1,C4,C6 = 100nF, 50V, X7R, 0805
C2 = 10µF, 16V radial can SMD, Panasonic FK series
C3,C5,C7 = 100µF, 16V radial can SMD, Panasonic FK series

Semiconductors
D1 = LL4001, diode, 50V, 1A
LED1 = green, 3mm
T1 = BSS138LT1G N-channel MOSFET, 50V, 200mA
IC1 = LM3940IMP-3.3, 3.3V LDO, 1A
MOD1 = ESP-01, ESP8266-based WiFi module

Miscellaneous
F1 = holder for 5 x 20mm fuse, PCB mount, 22.6mm pitch
F1 = fuse, 5 x 20mm (see text)
S1 = SPDT slide switch, through hole, angled (C&K 1101M2S4AQE2)
K1 = power jack 2.35mm, 4A (Multicomp MJ-180PH)
K2 = wire-to-board connector 0.2" pitch, 3 ways
K2' = pluggable terminal block 0.2" pitch, 3 ways
K3 = wire-to-board connector 0.2" pitch, 2 ways
K3' = pluggable terminal block 0.2" pitch, 2 ways
K4 = 6-pin pinheader, 0.1" pitch, vertical
JP1,JP2 = 2-pin pinheader, 0.1" pitch, vertical
JP1',JP2' = jumper
Enclosure Hammond 1591 112x62x31
PCB 160487-1 V1.1

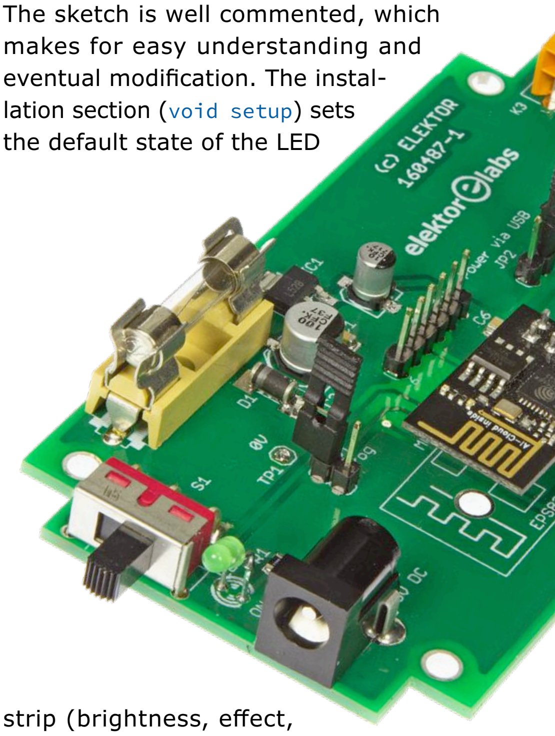




ber of LEDs. If the strip contains more LEDs, the extra LEDs will not light. If the number of LEDs in less than in LED_COUNT, the light effect will not be exactly as selected; a part of the sequence will be lost.
On lines 27 and 28 of the sketch, you can modify the name and password of the ESP Access point:

```
const char *ssid = "Neopixel
   Elektor"
const char *mot de passe =
   "Tours2017"
```

Thanks to the web server, you can make a connection through a router and you can send a command from any smartphone. To create the server, the sketch uses the libraries `WiFiClient` and `ESP8266WebServer`. The library `ESP8266WiFi` permits control of the ESP-01 module. The sketch is well commented, which makes for easy understanding and eventual modification. The installation section (`void setup`) sets the default state of the LED

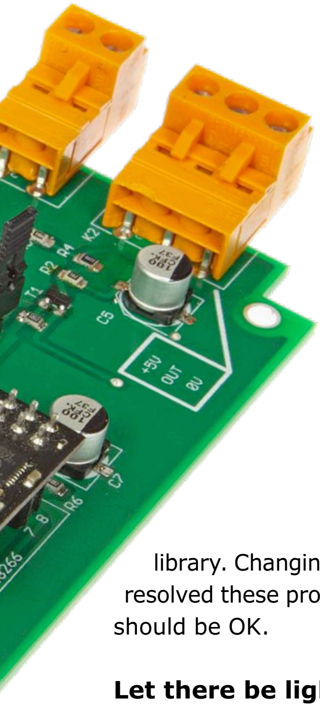


strip (brightness, effect, speed), the data rate of the serial port (115200) as well as activating the Wi-Fi connection and launching the web page. Following that the main section (`void loop`) waits to action requests form clients.

Compilation errors

Elektor Labs had some problems compiling the source code for this project following a new installation of the Arduino IDE and the Arduino ESP8266 plugin, errors in the use of the 'min' and 'max' function, to be more precise. It was due to the installation (by default!) in the board controller of a version of the ESP8266 plugin that was incompatible with the WS2812FX

Figure 2. Screenshot of the server's web page on a smartphone.



library. Changing to version 2.4.0-rc2 resolved these problems, other versions should be OK.

Let there be light!

Connect the LED strip to the 3-way connector. Note: do this without power applied! Next connect the power and switch on. As long as there is no web connection, the strip will be in 'demo' mode.

First connect yourself to the WiFi network of the ESP8266 access point, the default SSID is 'Neopixel Elektor' with the password 'Tours2017' (you can modify these in the sketch). Open your browser and enter the IP address 192.168.4.1 in the address bar. Select the LED brightness (0%,25%,50%,75%,100%) and the light effect (between 1 and 44), and press the 'Send' button to transfer these parameters to the server (**Figure 2**). Effect '100' will sequence through all the available effects. Note you must refresh the web page after each command. ◀

(180355-01)

Web Links

[1] NeoPixels (in French): https://wiki.mchobby.be/index.php?title=NeoPixel-UserGuide

[2] All about NeoPixels: https://learn.adafruit.com/adafruit-neopixel-uberguide/overview

[3] WS2812 datasheet:https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf

[4] Project page on Elektor Labs website: www.elektormagazine.com/labs/esp8266-web-server-for-neopixel-led-strips-160487

[5] Article suppport page:: http://www.elektormagazine.com/160487

Now that's power!

Image credit: Sparkfun, CC BY 2.0

The WS2812B [3] is an 'intelligent' LED light source: the driver and the RGB LED are integrated in a 5050 format SMD package. The package has four contacts: two for power (+5 V and 0 V), one for DATA IN and one for DATA OUT which allow the transmission of data along a strip of LEDs. The signal is unidirectional, the LEDs don't send back any information.

Each LED consumes 20 mA when its brightness is set to the maximum (255), so the total current drawn is thus 60 mA (R+G+B). For a strip a metre long with 60 LEDs, you will need to supply 3.6 A if all the LEDs are set to 100%! For 120 LEDs (2 m), it will be 7.2 A; for 180 LEDs (3 m), 10.8 A.

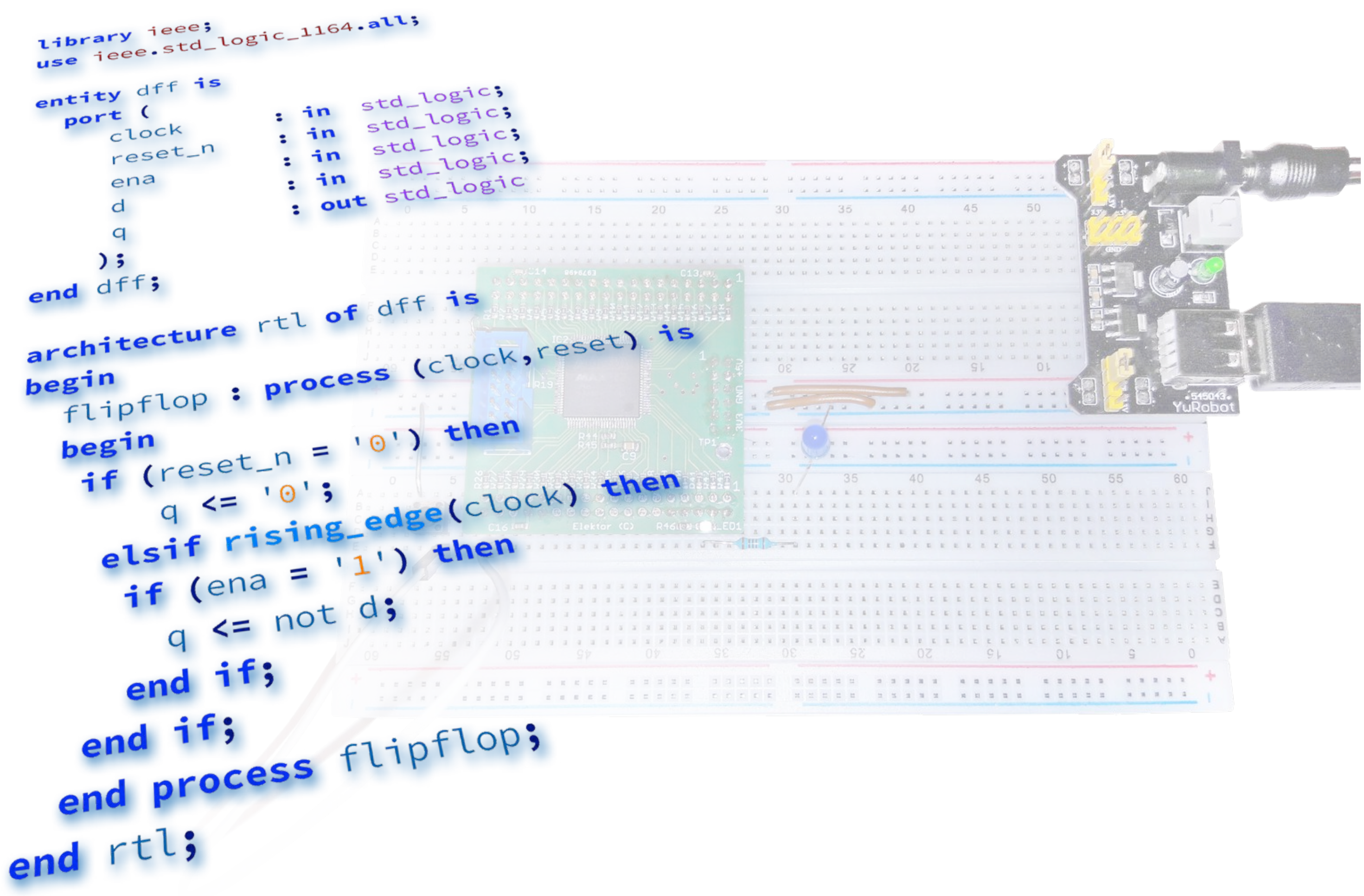
You can experimentally determine the power supply requirements: set the brightness and the effect that consumes the most power and use a regulated power supply to measure the current consumption.

With a strip of 144 LEDs (5 m), the author was obliged to connect power at both ends of the strip, as he found that there was a large voltage drop between the two ends. The ten LEDs at the end furthest from the power source had some random behaviour. This problem is linked to the quality of the LED supports and the size of the tracks.

A last word from the author: for NeoPixel strips, choose a power supply with an output voltage no higher than 5.2 V. The author has had problems with switching power supplies 'Made in China' – very cheap, but with an output voltage of 5.3 to 5.4 V. No problems were found with phone power supplies or those of a good brand. It may be necessary to add a 5-V regulator to the circuit.

Hardware Design using (V)HDL (1)

First steps with Quartus and ModelSim



By **Jörg Zollmann** (Germany)

Programmable logic has for a long time been the domain of professionals and geeks, not least because of the fact that the ICs involved are somewhat pricey and difficult to solder manually. Moreover, in order to bring the devices to life it is necessary to use a special-purpose programming language. The CPLD breakout board published in Elektor Magazine issue 3/2018 makes using these chips considerably easier, and so it is appropriate for us now to help our readers get to grips with the programming languages that are used to configure the internal structure of CPLDs and FPGAs.

With a little background knowledge and a few experiments we will rapidly show that using CPLDs in DIY projects does not involve any black magic, and that in fact they open up many new opportunities. The hardware platform for this course will be the CPLD board described in the previous issue of Elektor [1]. Just as the 'Hello World' program is the *pons asinorum* of software development, so the flashing LED is the starting point for hardware development; and so it is on that example that we will base our first foray into the world of hardware description languages.

Simple hardware

The download accompanying this article [2] contains all the examples we will be looking at. The best practice for small-to-medium-sized projects is to dedicate

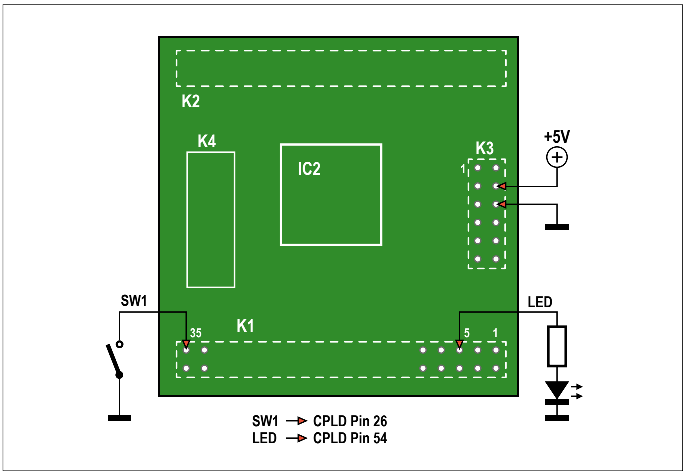


Figure 1. Circuit diagram with button, LED and resistor.

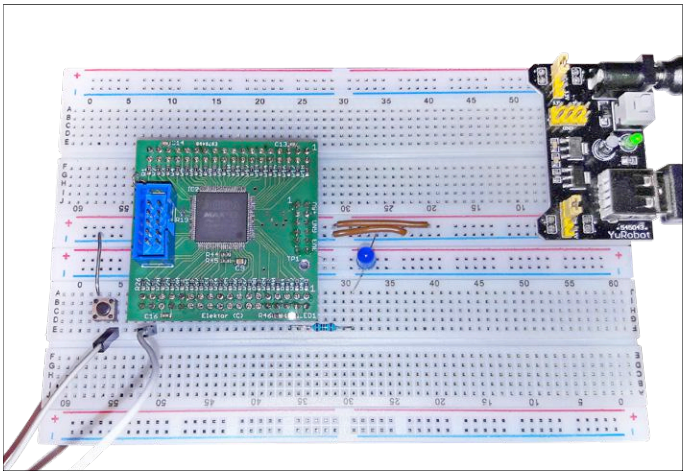


Figure 2. The CPLD breakout board mounted on a breadboard.

a directory to each project. This directory will have subdirectories called 'sim', 'src' and 'quartus'. On the hardware side things are very straightforward: the CPLD board is provided with power and extended with a pushbutton and a LED with its series resistor (see **Figure 1**); the final set-up of the evaluation board is shown in **Figure 2**. We will arrange things so that pressing the button will turn the LED on and off.

Installation

Installing the development environment requires about 1.5 GB. This includes Quartus Prime Lite and ModelSim Starter Edition. To download these it is necessary to register on the Intel website and agree to the terms and conditions. It is important to make sure that you select the correct devices during the installation process (we need just the MAX II family) and to install the help package.

Sequential synchronous digital logic

At the most fundamental level digital circuits consist of only two different types of component: combinatorial logic blocks and edge-triggered sequential logic blocks. **Figure 3** shows that the logic elements (LEs) of the MAX II CPLD contain both these kinds of block. The combinatorial part is implemented using a look-up table (LUT); in older devices this was implemented using AND and OR gates. The sequential part is implemented using a D-type flip-flop. Strictly speaking the device also allows the possibility of describing and synthesizing clock-level-sensitive logic, but latches and similar constructs are frowned upon in an otherwise fully synchronous design and should be avoided where possible unless you really know what you are doing. The rationale for this is that the static timing analysis implemented in currently-available synthesis tools does not cope well with latches. When designing hardware using VHDL it is important to try to picture the logic that will be synthesized in your mind's eye; in our examples the logic will consist of a sequence of combinatorial logic blocks and edge-triggered flip-flops.

Processes

Listing 1 and **Listing 2** show precisely how these building blocks are implemented in a VHDL description. The first is a D-type flip-flop with asynchronous reset and a clock enable input, while the second is a 'cloud' of combinatorial logic. The `entity` declaration describes the interface of the design to the outside world, that is, to the next higher level of the hierarchy. This section contains the signal or pin names in the `port` list, as well as an indication of which are inputs and which are outputs. This is comparable to an IC's package description, or to an API in the world of software. The function itself, describing the implementation of the circuit, is in the `architecture` section. At the heart of the architecture description are VHDL

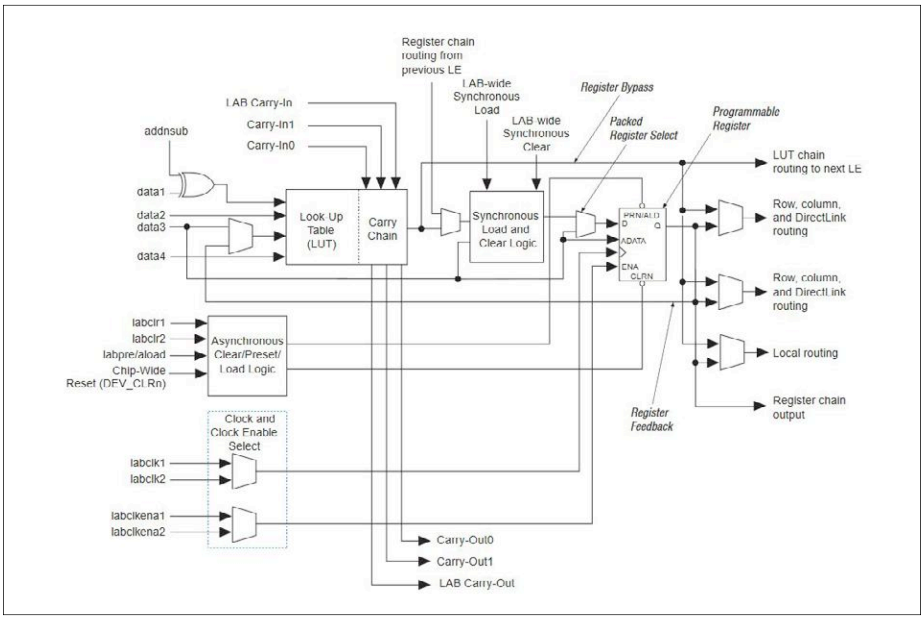


Figure 3. Inside a MAX II logic element (from the Altera/Intel datasheet [3]).

Listing 1. D-type flip-flop with inverting Input

```
library ieee;
use ieee.std_logic_1164.all;

-- VHDL comments are introduced by two dashes
entity dff is -- entity declaration
  port (
     clock      : in  std_logic;
     reset_n    : in  std_logic;
     ena        : in  std_logic;
     d          : in  std_logic;
     q          : out std_logic
    );
end dff; -- "dff" must match the name given after the "entity" keyword

architecture rtl of dff is -- the name of the architecture "rtl" is
arbitrary
          -- <--- internal signals can be declared here (see below)
begin
  flipflop : process (clock,reset) is
  begin
   if (reset_n = '0') then
       q <= '0';
    elsif rising_edge(clock) then
     if (ena = '1') then
       q <= not d;   -- the pushbutton input is 0 when it is pressed;
                     -- the "not" here
     end if;          -- inverts the logic level
   end if;
  end process flipflop;
end rtl;             -- must match the name of the architecture
```

processes. A process can be an assignment (denoted '<='), a procedure call, a component instantiation or, in the most direct case, a `process` clause. All VHDL processes within a single architecture block run simultaneously. A process is executed whenever one of the signals in its 'sensitivity list' changes state. The sensitivity list is given in brackets after the `process` keyword. In this case the two signals in the sensitivity list are `clock` and `reset`. For programmers used to working in C, this is often the part that leads to headaches: VHDL is inherently multi-threaded. Communication between processes is via signals (introduced using the `signal` keyword).

A rule of thumb for processes is that there are only three variants of the sensitivity list. The first, which includes `clock` and `reset` signals, generates sequential circuit components with an asynchronous reset. The second variant uses the `all` keyword: since VHDL-2008 this has been available to describe processes implementing combinatorial logic. The third variant is the empty sensitivity list: this is used in simulation (see below).

Data types

Only a couple of data types are really needed to describe hardware, but VHDL nevertheless offers a wide range of types. Also, VHDL is a strongly-typed language, which means that there are no implicit type conversions and each assignment must operate on identical data types; otherwise, an explicit type conversion of the operands must be included. The most fundamental data types in VHDL are the bit and the `bit_vector`. Signals of type bit can take on the values '0' or '1'. Although this type on it own is sufficient for logic synthesis, it is more usual to use the type `std_logic`. A signal of this type can take on nine different states: as well as '0' and '1' there are (among others) '-' ('don't care'), 'u' (undefined), and 'x' (unknown). These extra states make it easier to define complex boolean logic formulas, and to track down missing reset signals ('u') and signals with multiple drivers ('x'). The first thing for us to note is that `std_logic` is not defined by default. In order to use a signal of this type we must first include the IEEE library (using the `library` keyword). This library, which can be compared to a directory on a PC, contains multiple packages. The use keyword brings in the definitions from one of these packages. Note that VHDL is not sensitive to case. In software development there are control structures such as `if/elsif/else` and `case` statements to control the flow of execution. VHDL has constructs with a similar syntax, that can only be used within processes. Listing 2 illustrates the use of both. In order to avoid the unwanted instantiation of latches, another simple rule of thumb comes into play: each signal must be the subject of an assignment in each branch of the construct. In a `case` block the `others` branch must be specified, and `an if` block must have a corresponding `else` part. Alternatively, it is possible to assign default values to the signals at the beginning of the process.

Modelsim

A significant advantage of VHDL is that the description can be completely simulated. This means that before we program the design into a FPGA or CPLD, or bake it into an ASIC, we can test its functions thoroughly. However, again there are some traps here for the unwary: it is not necessarily the case that VHDL code that gives the correct results in simulation can be synthesized, or turned into hardware. But, as long as we write our VHDL code so that the design is composed of the fundamental blocks described above, there should be no difficulties. Stand-alone tools are used for simulation: in this case we use ModelSim, supplied free of charge by Intel and originally by Mentor Graphics (now part of Siemens). There is not space in this article to describe all the features of

Programming Adapter

A programming adapter is required to program the CPLD board. The company Terasic produces a suitable unit, which is recommended by Altera. Chinese clones are available on eBay for ten dollars or so and reports from the Internet community indicate that they appear to work under Linux, but that they can unfortunately lead to 'blue screens of death' under Windows 10. Sometimes it is necessary to reinstall the USB Blaster driver manually: in the Hardware Manager enter the path Intel Installation → Drivers and everything should work.

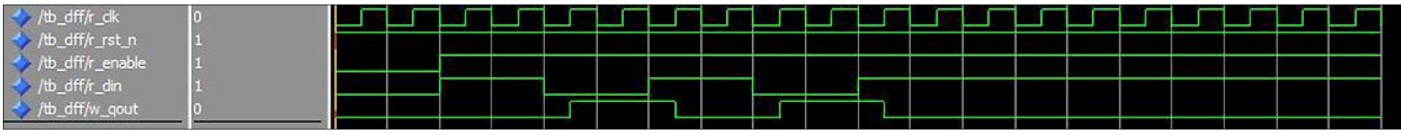


Figure 4. Simulation of a D-type flip-flop in ModelSim.

this software; but to start with we only need a couple of commands, which we will put in a .tcl (Tool Command Language) file. Tcl is a scripting language which is widely used in ASIC and FPGA development. To simulate a circuit we first construct a 'testbench'. This is a special piece of VHDL code that simulates a circuit external to the device (or unit) under test (DUT or UUT) to exercise its operation. **Listing 3** shows a simple testbench for the D-type flip-flop which generates a clock signal and toggles the input a couple of times. The testbench is inside a process that has an empty sensitivity list, and which must therefore contain a wait statement. The testbench also illustrates one of the several types of hierarchical structure in VHDL: here the DUT is instantiated using a component declaration. To start the simulation first change the current directory to the 'sim' subdirectory. Then, in the 'Transcript Window' enter the command sim_dff.tcl, which will cause the series of commands given in that file to be executed in sequence (see **Listing 4**). This will start the simulation and show the results graphically. The two most important commands are the invocation of the compiler `vcom` (to which you can pass a parameter specifying the version of VHDL to be compiled), and `vsim` to start the simulator. You can admire the results of the simulation in **Figure 4**. In the 'Simulation Window' and the 'Object Window' you can right-click to select further signals to be displayed graphically in the Wave Window. Clicking on the diskette icon in the Wave Window (showing the age of the tool somewhat!) saves the currently-displayed signals for the next simulation run. Once the circuit is behaving as it should in simulation, it is time to move on to programming the CPLD.

The Quartus development environment

Intel offers a complete IDE for developing FPGA and CPLD applications. The best way to get to know your way around a development environment is always to try actually using it. To create a new project, start the Project Wizard (File → New Project Wizard) and click your way through, answering the various questions: in particular, it is here that you can set the 'working directory' to 'quartus'. This is where Quartus will store all sorts of files related to the project. The most important of these are the project description file (*.qpf), the settings file (*.qsf) and, if necessary, any other constraint files (*.sdc). You will also need to point Quartus to any source files already

Listing 2. A cloud of combinatorial logic

```
library ieee;
use ieee.std_logic_1164.all;

entity logic_cloud is
  port (
      a           : in  std_logic;
      b           : in  std_logic;
      c           : in  std_logic;
      d           : out std_logic;
      e           : out std_logic
    );
end logic_cloud;

architecture rtl of logic_cloud is

signal abc    : std_logic_vector (2 downto 0) := (others => ‘0’);
-- abc is a 3-bit-wide bus, or vector, of std_logic signals
-- elements 2 to 0 are initialized to ‘0’

begin

  abc   <= a & b & c ;  -- concatenation of a, b and c into a bus

  combinatorial  : process (all) is -- "all" has been available since
                                    --  VHDL-2008 and is
  begin                             -- designed for implementing
                                    -- combinatorial logic

  if abc = "001" then
    d <= ‘1’;
  elsif abc = "101" then
    d <= ‘1’;
  else  -- IMPORTANT: without the "else" branch a latch will be
        -- synthesized
    d <= ‘0’;
  end if;
  end process combinatorial;

  mux  : process (all) is
  begin
    case (abc) is --
       when "000"  => e  <= ‘1’;
       when "011"  => e  <= ‘1’;
       when others => e  <= ‘0’; -- here the rule is not so hard and
fast but the
    end case;                    -- "others" statement avoids any
                                 -- possible error

  end process mux;

end rtl;
```

Listing 3. Testbench for the D-type flip-flop

```
library ieee;
use ieee.std_logic_1164.all;
library work;

entity tb_dff is    -- a testbench has no port list
end tb_dff;

architecture behave of tb_dff is

  constant c_CLOCK_PERIOD : time := 20 ns;  -- constants to improve
                                            -- readability

  signal r_clk    : std_logic := '0';
  signal r_rst_n  : std_logic := '0';
  signal r_enable : std_logic := '0';
  signal r_din    : std_logic := '0';
  signal w_qout   : std_logic;

  component d_ff is  -- component declaration
    port (
      clock      : in  std_logic; -- semicolons required except at
                                  -- the end
      reset_n    : in  std_logic;
      ena        : in  std_logic;
      d          : in  std_logic;
      q          : out std_logic  -- no semicolon
      ); -- semicolon
  end component d_ff;
begin

  clk : process is  -- no sensitivity list - a wait statement is
                    -- required
  begin
    wait for c_CLOCK_PERIOD/2;
    r_clk <= not r_clk;
  end process clk;

  dut : d_ff       -- component instantiation: assign signals using
                   -- the port map
      port map (
      clock     => r_clk, -- commas required except at the end
      reset_n   => r_rst_n,
      ena       => r_enable,
      d         => r_din,
      q         => w_qout -- no comma
    );   -- semicolon

  process is
  begin
    r_rst_n  <= '1';
    r_din    <= '0';
    wait for 40 ns;
    r_enable <= '1';
    r_din    <= '1';
    wait for 40 ns;
    r_din    <= '0';
    wait for 40 ns;
    r_din    <= '1';
    wait for 40 ns;
    r_din    <= '0';
    wait for 40 ns;
    r_din    <= '1';
    wait ;
 end process;

end behave;
```

stored in the src directory and select the correct target device: and then you are good to go.

There is one further setting of interest: under Assignments → Settings → Compiler Settings we have the ability to set the version of VHDL. Although one might think that VHDL-2008 would be universal by 2018, that is unfortunately not the case. Quartus' main window normally consists of a text editor in the middle, the Message Window along the bottom edge, and the Tasks and Project Navigator Windows on the right-hand side. Of course, the arrangement of windows is freely configurable and View → Utility Windows lets you recover lost windows and create new ones. The Tasks window gives a convenient overview of the steps involved in creating a bitstream file for the programmer. Clicking on the small play symbol allows these individual steps to be executed manually one by one. For initial debugging of a design the 'Analysis and Synthesis' step is often all that is required, saving time when finding syntax errors. A useful tool for learning the language is the RTL Viewer, which is hidden away in the NetlistViewers directory (or it can also be reached using the menu system under Tools → NetlistViewers → RTL Viewer). This tool shows graphically how Quartus is interpreting your source code as RTL (Register Transfer Level). **Figure 5** and **Figure 6** show the views that result from the code in Listing 1 and Listing 2 respectively. This view gives direct feedback to the designer on how the VHDL constructs you have used are mapped into hardware.

The synthesis tool cannot know which pins of the CPLD are connected to the pushbutton or to the LED, and so we

Listing 4. Modelsim .tcl file

```
# Simulation Control Script
file delete -force work
vlib work
vmap work work

puts "Compile DUT"
vcom -2008 ../src/d_ff.vhd

puts "Compile Testbench"
vcom -2008 tb_dff.vhd

puts "Starting Simulation"
vsim -t 10ns tb_dff
do wave.do
run 15 us
```

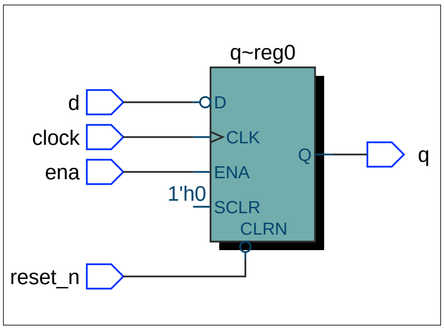


Figure 5. RTL Viewer result from code in Listing 1: a D-type flip-flop with asynchronous reset.

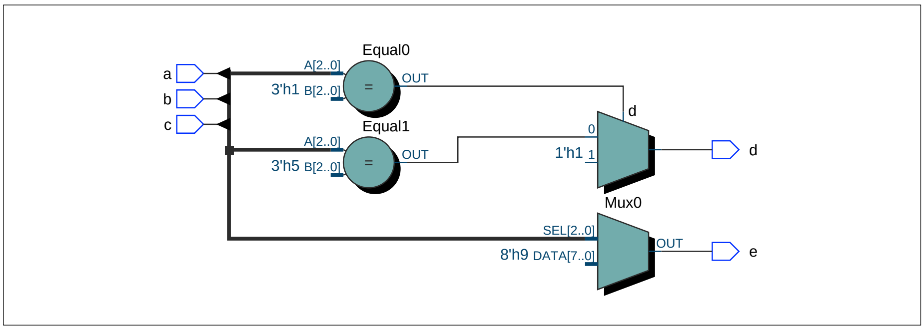


Figure 6. RTL Viewer result from code in Listing 2: a cloud of combinatorial logic.

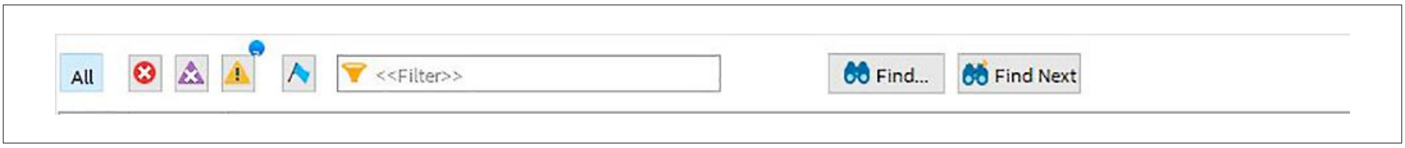


Figure 7. Quartus' Message Window: take control of those warnings!

must find a way to give this information to Quartus. The best way to do this is, after the analysis process has been completed successfully at least once, to go to the Pin Planner (Assignments → Pin Planner), and in the left-hand column you will see all the top-level signals that appear in the design. As well as assigning each of these signals from the VHDL design to a physical pin on the CPLD, a further column makes it also possible to assign an 'IO Level' to the signal. For the examples in this article we will use the default setting of 3.3 V. An alternative way to specify the pin assignments is provided by the Assignment Editor (Assignments → Assignment Editor). Here you cannot just bind signals to pins (using 'Location Assignments'), you can also configure various special properties such as whether the internal pull-up resistor is enabled. In our example we will take advantage of exactly this feature, so that the pushbutton can simply be connected between the pin and ground without the risk of an undefined logic level appearing on the pin. **Important:** in order to avoid damage to your hardware, it is essential to specify the configuration of Device Pins that are not explicitly used in the design. The default setting in Quartus is that these pins are driven to ground, and, depending on the external circuitry connected, this can have disastrous consequences. It is safer to use the 'Assignments → Device → Device and Pin Options → Unused Pins: As input tri-stated' setting.

Errors, critical warnings, warnings... and help

After clicking on the play button in Quartus for the first time (or Ctrl-L → Start Compilation), a series of messages will be displayed in the Processing tab of the Message Window (ALT-3). This tsunami of information can be somewhat disconcerting, but have no fear: most of the messages are really for information purposes only and so can be ignored. The symbols 'X', 'x' and '!' at the top edge of the Message Window (see **Figure 7**) lets you sort the messages into errors, critical warnings and warnings. This is the order in which the messages should be investigated and dealt with. Your aim should be at least to ensure that there are no critical warnings that apply to your design; hoping to avoid any warnings at all is usually unrealistic. Quartus' help page (F1) often gives a good explanation of an error or warning given its code. Once all errors have been dealt with and Quartus has successfully performed a complete compile-run cycle, the programmer file (the configuration data for the CPLD) can be transferred to the hardware. To do this, open the 'Programmer' and select 'USB Blaster' as the hardware configuration. When the connection is correctly set up a click on 'Auto-Detect' should display 'EPM570T100'. Now select the .pof file that has just been created and click on the Start button, and that's it! The LED on the CPLD board should light.

All done!

This simple example project has illustrated how a little bit of software can bring the CPLD to life. Now a flashing LED is hardly likely to blow anyone's socks off, but it forms a solid foundation and we are now in a position to proceed with a series of more entertaining and interesting projects, learning some more about VHDL on the way. ◀

180285-02

Web Links

[1] www.elektormagazine.com/160425

[2] www.elektormagazine.com/160674

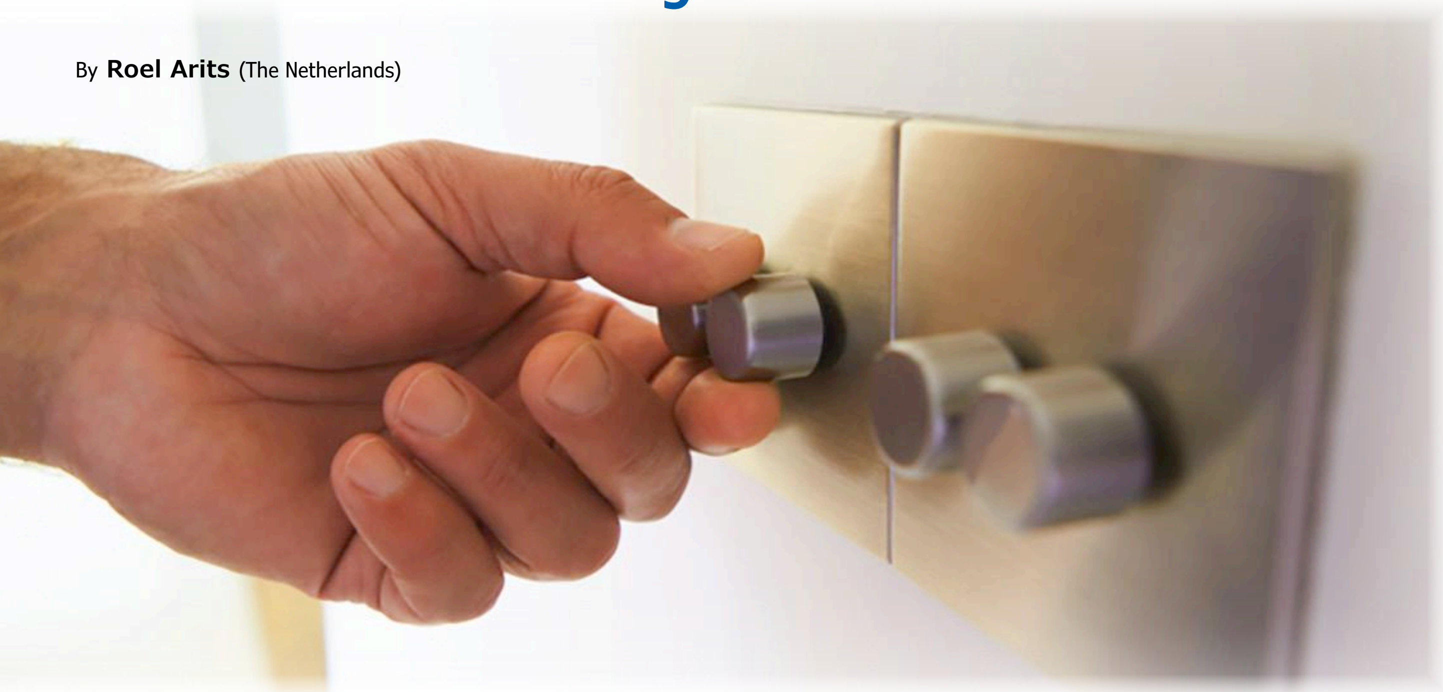
[3] https://bit.ly/2ri1edP

HOMELAB PROJECT

LED Dimmers (1)

...old-fashioned analogue

By **Roel Arits** (The Netherlands)



In August last year, we at Elektor Labs organised a small competition — a challenge to design an analogue LED dimmer. In this year's January & February issue [1] we published the winners — and Roel Arits, with the most entries the uncrowned King of the LED Dimmers. In a couple of articles we will shine some light on his interesting creations.

When controlling the brightness of an LED (or of an incandescent lamp) you will notice something peculiar. When we increase the current through the lamp or LED linearly (and it doesn't matter whether that is done through analogue or digital means), we don't observe the brightness changing linearly. In fact the opposite: initially the observed brightness increases rapidly, but when we continue to increase the current at the same rate, the change in brightness appears to reduce. Or expressed differently: to observe a change in brightness for an already bright LED requires a greater increase in current than seeing a change in brightness of a weakly lit LED.

Weber-Fechner law

What is happening here? Well, the gentlemen Weber and Fechner looked into this between 1830 and 1860 and formulated the law now named after them [2]. It turns out that our senses are not linear biological sensors, but have more of a *logarithmic* characteristic. And that is a good thing too, because this is the reason our eyes have such an exceptional large dynamic range, varying from star light on a moonless night to the bright summer sun on the beach.

It may be obvious that this (approximate) logarithmic characteristic of our eyes has consequences for the control of the brightness of an LED (because incandescent lamps are now 'out', we limit ourselves in this contribution to that of LEDs). When we want to control the brightness such that we observe a linear increase, we have to increase the current in the LED *exponentially*!

That is, by the way, also the reason that the volume controls in audio amplifiers use logarithmic potentiometers; our ears have, just like our eyes, an approximate logarithmic sensitivity response.

In **Figure 1** we have sketched the observed change in brightness of an LED with a linear change (increase and decrease) of the current through the LED, and in **Figure 2** the observed change in brightness with an exponential change in the LED current.

Characteristics

- Different methods for dimming LEDs using analogue techniques
- Using only standard components
- Invites further experimenting...

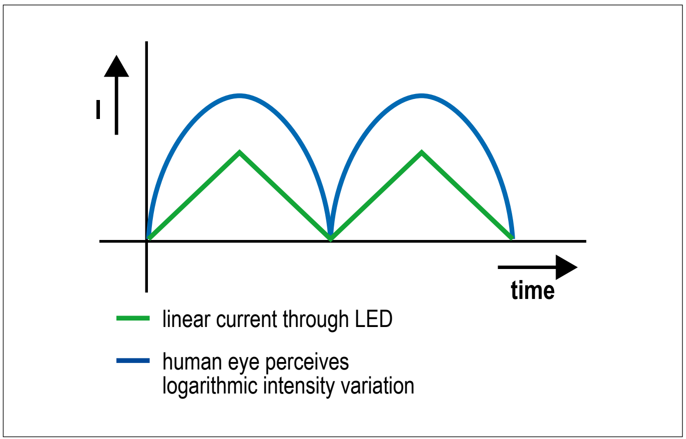


Figure 1. When the current though an LED increases or decreases linearly we observe the brightness as a logarithmic change.

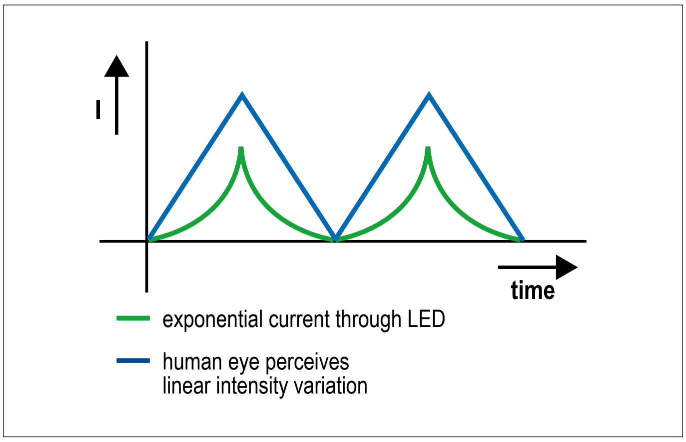


Figure 2. Here the opposite happens: with an exponential change in current we perceive a linear change in brightness.

From biology to electronics

When we want to control the brightness of an LED we therefore need a circuit that, in one way or another, adjusts the current through that LED in such a way that we observe a linear change of brightness (such as that sketched in Figure 2). In the present era of microcontrollers this would be a piece of cake: a *look-up table* in the memory of the controller contains a neat correction factor for each brightness value. When we use that to build a dimmer that is operated with a potentiometer, then an equal amount of rotation of the potentiometer corresponds to an equal change in observed brightness.
But how do we do that in an analogue circuit? (This was exactly the challenge posed by the Elektor Labs competition!)
The best way (and that is really true for any design) is to work systematically, and first draw a block diagram; later on we can worry about filling in the details of the various blocks. ('Worry'? Well, not really — in fact only then does it become really interesting!)

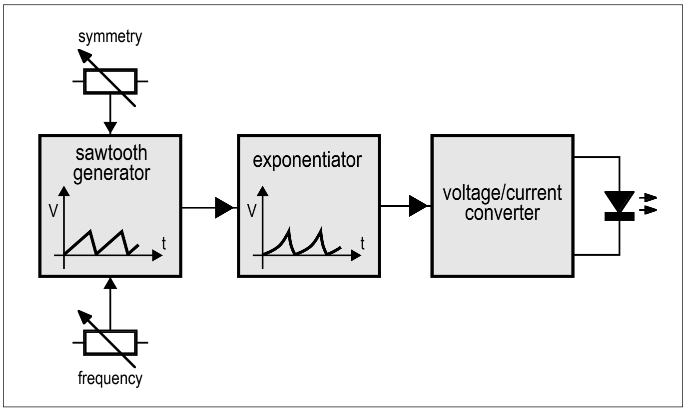


Figure 3. The general block diagram for all our analogue dimmer circuits.

Block diagram

This is drawn in **Figure 3**. We begin with a sawtooth generator (or triangle generator, if you prefer) with adjustable frequency and symmetry (in a 'real' dimmer the brightness is of course adjusted with a potentiometer; in our demo projects we have automated this manual adjustment). The resulting sawtooth is subsequently converted through an 'exponentiator' into a more or less exponentially increasing and decreasing voltage. It is also possible to use a squaring circuit here; this also gives a reasonable approximation of an exponential change. Oh, you know what – we will just do both...
Finally, the so obtained exponentially increasing and decreasing voltage has to be converted into a corresponding current for driving the LED. *That's All, Folks!*

- A sawtooth generator can be built without much problem using a pair of opamps (an integrator and a Schmitt-trigger). The frequency of the generated sawtooth can be adjusted with a potentiometer (that is, the rate at which the brightness of the LED increases and decreases again). With a second potentiometer the symmetry of the sawtooth can be adjusted.
- The exponentiator or squaring circuit turns the triangle into an exponential or quadratic changing signal respectively. The exponential change can, for example, use the logarithmic resistance characteristic of an LDR; for a quadratic change we can use, for example, the quadratic gate-voltage/drain-current characteristic of a FET.
- For the voltage/current converter we could, of course, use an ordinary resistor in series with the LED, or something more luxurious, utilising a current mirror or a voltage-controlled current source.

Alternating dimmer

As you can see in the schematic of **Figure 4**, we use here two methods to obtain the desired dimmer characteristic. For one method we use an LDR and in the other a current mirror, which, when correctly adjusted, has a quadratic characteristic. Because the voltages obtained this way happen to be 180° out of phase, this gives a nice alternating effect (one LED comes on while the other one goes off, and the other way around).
The sawtooth or triangle generator is built around opamps IC1.A and IC1.B, where the first is configured as a Schmitt-trigger and the second as an integrator. Because the circuit is powered from a single-ended power supply voltage of 9 to 12 VDC, we first make a reference voltage for both opamps with resistors

Warning

The circuits described in the article are proof-of-concept designs, intended for a low-voltage power supply (9–12 VDC) and for driving a single LED. The circuits are **absolutely not suitable** for controlling LED lamps and are **completely unsuitable** for connection to the AC line voltage!

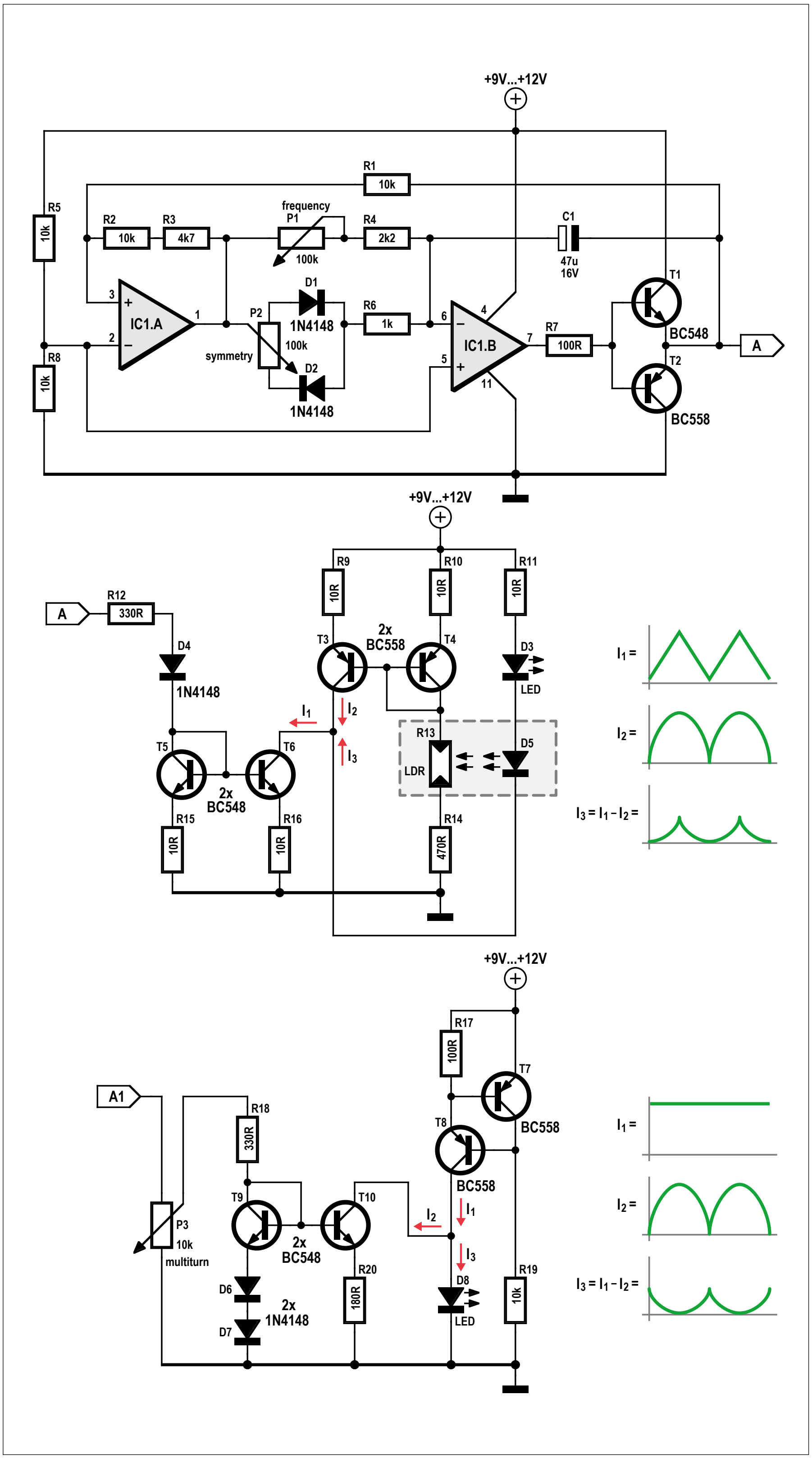


Figure 4. The schematic of the alternating dimmer.

R5 and R8, that is equal to half the power supply voltage.

The integration time of the integrator IC1.B is determined by C1 in combination with P1 and R4. Because C1 is a fairly 'fat' capacitor, we have added a voltage follower (T1/T2) to the output of IC1.B to ensure that sufficient current is always available. Potentiometer P1 is used the set the frequency of the triangle (with the indicated values between about 0.3 Hz and 3.8 Hz). P2 is responsible for the symmetry of the triangle. Resistor R6 determines the minimum rise and fall times of the triangle. Because the frequency and symmetry adjustments are connected in parallel to each other, and because of the values selected for the potentiometers, both these settings do not influence each other much.

The output of the integrator is fed back via R1 to the non-inverting input of the Schmitt-trigger. The output of which is, through R2 and R3, fed back to the same input. The positive feedback, together with R1, provides hysteresis around the reference voltage. It is, ultimately, this hysteresis that determines the amplitude of the generated triangle.

When the output of the Schmitt-trigger is high, the voltage at the output of the inverting integrator will fall – and the same is true for the voltage at the non-inverting input of the Schmitt-trigger. Once this voltage drops below the reference voltage on the inverting input, the output of IC1.A will flip and becomes low. Through the feedback via R2 and R3 the non-inverting input will be pulled even lower.

At the same time the output voltage of the integrator will increase because the Schmitt-trigger output is now low. As a consequence the voltage on the non-inverting input of the Schmitt-trigger increases again – until this is greater than the reference voltage. Then the output of the Schmitt-trigger will flip over again and becomes high. In this way the entire process continues to repeat itself.

The signal at the output of the voltage divider has, with the values indicated, an amplitude of about 6.5 V; the signal oscillates up and down between 3 and 9.5 V. This signal ('A' in the schematic) goes to both the exponentiator as well as the squaring circuit.

Exponentiator

This subcircuit is formed by the four transistors T3 through T6 and associated parts. We use an LDR (R13, a light dependent resistor based on cadmium sulphide) for

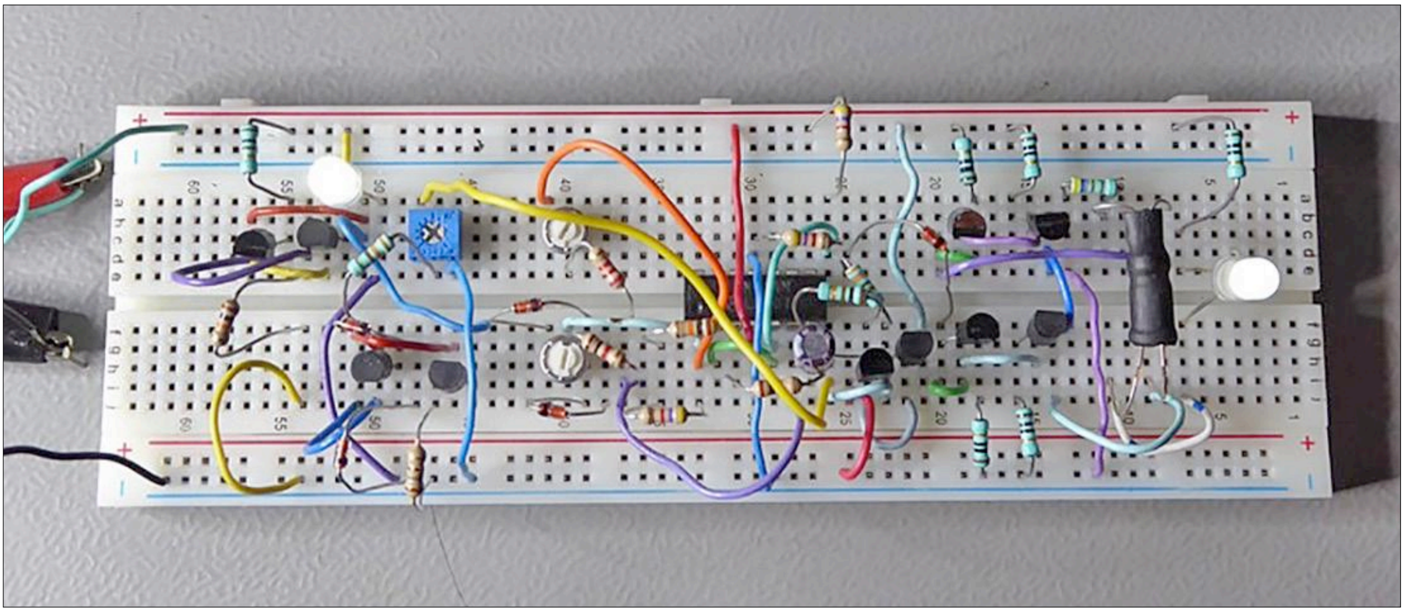


Figure 5. The alternating dimmer on the breadboard.

the 'translation' from a triangle to a more or less exponential wave shape. The light from the LED is fed back via the light dependent resistor and so brings its logarithmic behaviour into play.

For this purpose we added a second LED (LED2) in in series with the first one; because we are using two identical LEDs (same type, same colour), we may assume that they are of equal brightness when an equal current flows through them (which is automatically the case in a series circuit). To prevent ambient light from interfering, this LED and the LDR are facing each other and 'encapsulated' in a piece of black heatshrink tubing.

Because in order to control the brightness of an LED we need to control the current through the LED and not the voltage across the LED, we are 'thinking' here in terms of currents and not voltages. This is therefore the reason why we are using two current mirrors (T3/T4 and T5/T6). A nice feature of current mirrors is that the current in one branch is always equal to the current in the other branch. In the ideal case the transistors in a current mirror are 'matched' (selected for having identical electrical characteristics, as much as possible) and thermally coupled; however, for our proof of concept circuit this is not really necessary.

The current through the current mirror on the left (T5/T6) is determined by the triangle voltage that (via R12 and D3) is supplied from the generator. The only function of D3 in this context is to shift the triangle signal 0.7 V down so that the current will become low enough for the LEDs to turn off completely.

The current through the other current mirror (T3/T4) is determined by the current through the series circuit of LDR and resistor R14. This current has, as a consequence of the LDR characteristic, a logarithmic shape when the brightness of the LED changes linearly.

Now the advantage of thinking in currents is demonstrated: at the node of both current mirrors and the LEDs we may simply add all the currents (this is, by the way, Kirchhoff's first law). Therefore: the current I_3 through the LEDs in series is equal to the difference of the current I1 through the left current mirror and the current I2 through the current mirror on the right:

LED there be light!

$$I_3 = I_1 - I_2$$

And as we have drawn adjacent to the schematic, this current has in inverse logarithmic and therefore exponential shape.

Quadratic circuit

Transistors T7/T8 together form a constant-current source that supplies a constant current I_1 through the LED. This current determines the maximum brightness of the LED; with the component values indicated, this current is about 7 mA. To the left of the current source we see again a current mirror (T9/T10), where the emitter connection of T9 contains two diodes (D4, D5). These ensure that the current through this branch of the current mirror, again derived from the triangle from the generator, has a quadratic shape. Potentiometer P3 and series resistor R18 enable the current to be adjusted such that the LED turns off at exactly the right moment.

Just as with the exponentiator, the final current I_3 through the LED is again equal to the difference between the constant current I_1 and the squared current I_2:

$$I_3 = I_1 - I_2$$

From the small charts adjacent to both the subcircuits in **Figure 4** it is clear that the currents through the LEDs are 180° out of phase, which creates a nice affect.

Figure 5 shows the circuit built on a breadboard; because these are (just as more variants yet to be described) proof of concept circuits, no circuit board has been designed for this.

And now with FETs...

After the alternating dimmer the author had caught the bug;

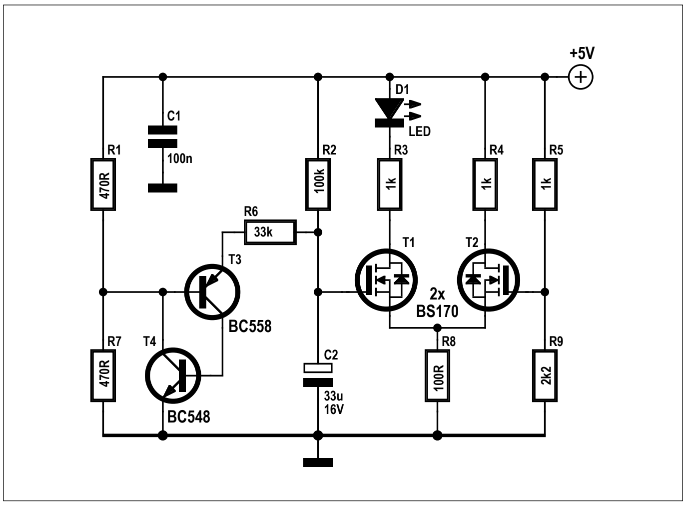


Figure 6. Schematic of an LED dimmer circuit using FETs.

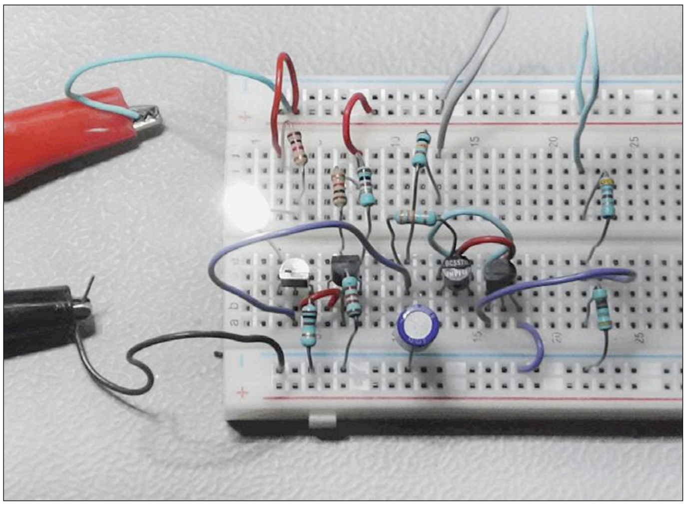


Figure 7. The FET dimmer on the breadboard.

the variant described below (refer to the schematic in **Figure 6**) uses the square-law relationship between the gate voltage and drain current of a FET.

But we first need, of course, a triangle generator; here we build it with the aid of a PUT – that is a Programmable Unijunction Transistor, a member of the thyristor family [3].

Although PUTs as three-legged components (anode, cathode, gate) are still reasonably easily available, we nevertheless build a discrete version using two common-or-garden variety transistors (T3 and T4). Resistors R1 and R7 form a voltage divider which sets the trigger voltage of the PUT to half of the supply voltage. After switching the power supply voltage on, capacitor C2 will slowly charge via resistor R2;The voltage across C2 will increase accordingly. Once the voltage across C2 is equal to the trigger voltage of the PUT plus 0.7 V (the latter is as result of the BE-junction of T3), T3 will begin to conduct. As a consequence the base of T4 also receives current, which causes T4 to conduct also. This causes the base of T3 to be pulled low with the result that T3 will conduct even more. Now T4 receives even more base current — and so on: both transistors enhance each other's conduction so that the PUT will latch itself into maximum conduction.

The instant that the PUT starts to conduct, C2 is discharged via resistor R6 and the PUT to ground. This discharge will continue until the voltage across C2 drops far enough so that there is insufficient current through the base-emitter junction of T3. At this moment the PUT unlatches and both transistors block. Then C2 can be charged again via R2.

This process repeats as long as the power supply voltage is applied; across C2 is a more or less triangular voltage ('more or less' because the rising and falling slopes are not linear, but follow the charge and discharge curves of capacitor C2).

With the component values shown in the schematic, the period of the triangle is about 3 seconds; the fade-in and fade-out times are then about 1.5 s each. The triangle has an amplitude of about 0.9 V; this signal oscillates up and down between about 2 V and 2.9 V.

T1 and T2 form a MOSFET differential amplifier; on one input is our triangle; on the other input a fixed voltage that corresponds with the gate-threshold voltage of the BS170. This fixed voltage is derived from the power supply voltage of 5 V through voltage divider R5/R9, and amounts to about 2.7 V. It so happens that the quadratic relationship between the gate voltage and drain current of a MOSFET starts just when the gate threshold is exceeded. The differential amplifier amplifies the difference between the triangle voltage and the threshold voltage, where the triangle voltage just oscillates up and down around the gate threshold voltage.

The current through the LED shows a quadratic wave shape with respect to time, as a result the change in brightness appears linear to the eye.

The frequency of the triangle generator can be changed by changing the value of C1; for the LED a high-efficiency type is preferred, because these give a decent light output at modest currents. In **Figure 7** you can see the prototype of the dimmer on a breadboard (neither was a circuit board designed for this circuit).

To be continued....

Hopefully this contribution has encouraged you to experiment with analogue electronics again! In the next instalment we will describe two more ways of dimming LEDs the analogue way.

(170404-01)

Web Links

[1] Overview of contest entries: https://goo.gl/JWo8oC

[2] Weber-Fechner Law description: https://en.wikipedia.org/wiki/Weber-Fechner_law

[3] Progammable Unijunction Transistor: https://en.wikipedia.org/wiki/Programmable_unijunction_transistor

Phantom Power Supply using Switched Capacitors

Voltage tripler using three ICs

Thomas Scherer

If you wish to connect a low-cost, large-diaphragm microphone with decent sound quality to your PC you will often find that it requires a phantom power supply at a higher voltage than the PC can deliver. A ready-made phantom power supply with its own separate power brick or USB connection it is a bit too elaborate and expensive for applications such as Skype: you can build one yourself! What's more, it can be made sufficiently small that it can be mounted directly inside a low-cost USB sound card.

The microphone input of a PC or laptop sound card typically delivers a phantom supply at 3.3 V with a series resistance of between 2 kΩ and 3 kΩ. This is enough to operate the usual low-cost electret and large-diaphragm microphones, but only at very low gain, and the resulting low output level results in poorer sound quality than would otherwise be possible. The nominal operating voltage of these microphones is 48 V, but they will work well down to voltages as low as 9 V with a 3.3-kΩ series resistor. Now that gives us some ideas!

Figure 1. My low-cost large-diaphragm microphone has a retro look.

Starting point

In the search for better sound performance for applications such as Skype than the built-in microphone of my Logitech webcam could offer, I made a snap decision to purchase a low-cost large-diaphragm microphone (see **Figure 1**). Despite only setting me back some €25 (about £20 or US$30) this was a definite step forwards, given that I had no desire to invest in professional-level equipment for this experiment. The microphone's size and retro looks gave a good first impression, but its output level left something to be desired. The reason for this was easy enough to find: the microphone socket on my PC delivered a supply at only just over 3 V through around 3 kΩ. Under the load of the microphone the voltage dropped almost to 2 V, which was too little: I estimate that the impedance converter built in to the microphone was attenuating the signal by a good 20 dB. I found that a dedicated external 'sound card' in the form of a USB stick with microphone input and headphone output designed for a Plantronics headset delivered nearly 5 V. Using this, the output level from my new microphone was much improved, but still not good enough; and in any case I needed the USB stick in order to use the headset.

A little research on the Internet revealed that most low-cost large-diaphragm microphones work very well with a phantom power supply at 9 V or higher, and a quick test using a 9-V battery and a 2.2-kΩ resistor was indeed successful. So all I had to do was rustle up a simple voltage converter and everything should be fine.

Initial considerations

A maximum of 5 V is available from a PC's external USB connector. However, a step-up converter using an inductor would create too much interference on the audio connections and would also take up space. So my idea was to build a phantom power supply inside a low-cost external USB sound card of the sort that

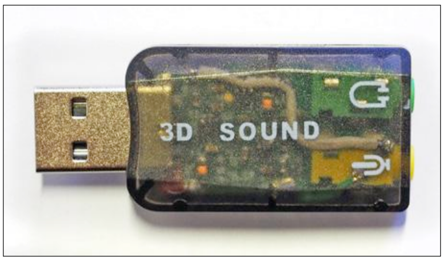


Figure 2. This compact USB sound card is available very cheaply.

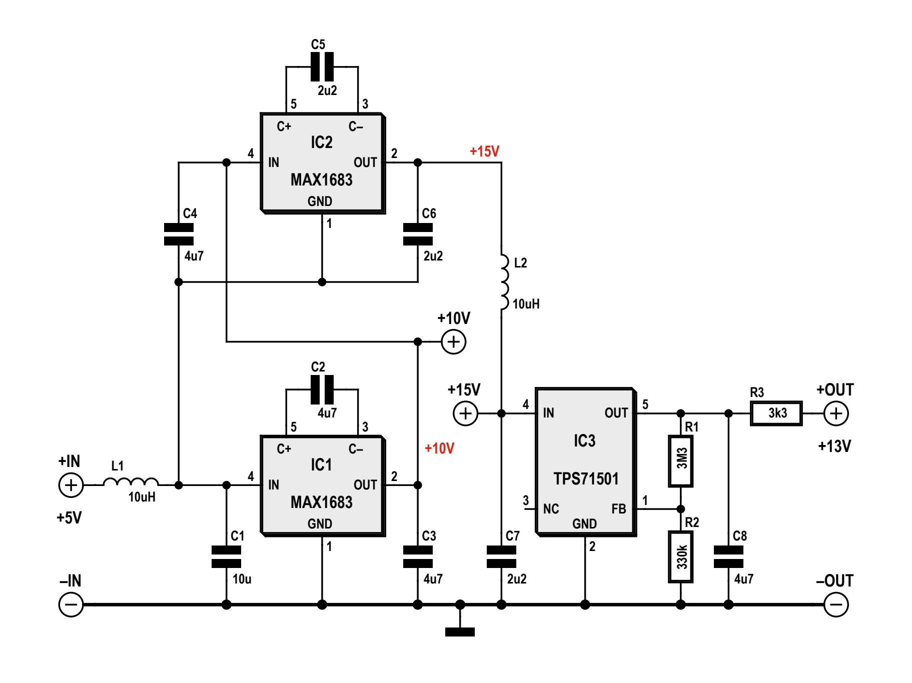


Figure 3. The ICs make the switched-capacitor phantom power supply design very simple.

COMPONENT LIST

Resistors
All SMD 0603
R1 = 3.3MΩ
R2 = 330kΩ
R3 = 3.3kΩ

Capacitors
All ceramic SMD 0603
C1 = 10µF, 10V
C2,C3,C4,C8 = 4.7µF, 25V
C5,C6,C7 = 2.2µF, 25V

Inductors
L1,L2 = 10µH, SMD 0805

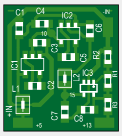


Figure 4. The board can be made very small!

Semiconductors
IC1,IC2 = MAX1693, SOT23-5
IC3 = TPS71501, SC70-5

Miscellaneous
External USB sound card (see text)
Printed circuit board (not available from the Elektor Store)

can be bought for less than the price of a pint on eBay (see **Figure 2**). A realistic approach would be to use a simple squarewave generator followed by a voltage multiplier, or, more elegantly, a voltage doubler [1] using a switched capacitor arrangement. The principle of operation of this circuit is easy to explain. In a voltage doubler a capacitor is first charged up to the supply voltage. Then its negative terminal is taken to the positive supply voltage and as a result the voltage on its positive terminal becomes equal to the supply voltage plus the voltage across the capacitor. This capacitor then partially discharges into a reservoir capacitor. Finally the capacitor is then switched back to the supply and the process continues from the beginning again. The difference between a classical voltage doubler and the variant with switched capacitors is that the latter uses transistors instead of diodes, which introduce losses due to their forward voltage. This increased efficiency is particularly helpful when starting from a relatively low voltage such as 5 V. It is no surprise that ready-made IC voltage doublers that use FETs as the switching elements can offer an (open-circuit, at least) output voltage of very nearly a full 10 V when operating from 5 V.

Design criteria

A 10 V output is indeed adequate for our purposes, but unfortunately the USB power output from a PC is swimming in digital interference. We therefore need to generate a bit more than 10 V so that we can use a small linear regulator to clean up the supply. To be on the safe side we therefore want to look at tripling the input voltage as a minimum. We should then have nearly 15 V at our disposal, from which we can generate a clean 13 V supply. This should work in 98 % of cases as typically the current draw is very low.
The frequency at which our IC switches needs to be above the audio spectrum so that we can remove any residual RF interference with a low-pass filter. Also, we would prefer the IC to be small, as we need two of them in addition to the linear regulator: that calls for SMDs in the smallest packages we can find. One device that fits the bill is the MAX1683 [2] which operates at a clock frequency of over 30 kHz.
Unfortunately this part has a maximum input voltage of 5.5 V, and so we cannot simply connect two devices one after the other and thereby quadruple our input voltage to 20 V. But a simple trick (suggested by the manufacturer) can be used to make a voltage tripler that works very well, and the resulting 15 V is perfect for

Weblinks

[1] Voltage multiplication: https://en.wikipedia.org/wiki/Voltage_doubler

[2] MAX1683: www.maximintegrated.com/en/products/power/charge-pumps/MAX1683.html

[3] TPS715: www.ti.com/sitesearch/docs/universalsearch.tsp?searchTerm=tps71501

[4] Project support page: www.elektormagazine.com/180165-01

our purposes. A TPS71501 low dropout ultra-low-power linear regulator [3] completes the very tiny and very clean 13-V phantom power supply.

Connecting the three ICs

The circuit is shown in **Figure 3**. After a first stage of filtering with L1 and C1, the 5 V USB supply goes to the input of IC1. C2 is the switched capacitor, which charges output reservoir capacitor C3 to 10 V. And here comes the trick: since IC2 cannot be supplied directly with 10 V, the ground of IC2 is connected to +5 V and its input to the 10 V present on C3: the effect of this is that IC2 only sees a permitted 5 V across its input. Across C6, as C3, a potential of 10 V is developed, but since C6 sits above the 5-V input voltage the positive terminal of C6 is at +15 V with respect to ground.

After further filtering by the 12 dB/octave low-pass filter formed by L2 and C7, IC3 produces a stable 13 V from its 15-V input; C8 provides further noise rejection. R3 is the series resistor for the phantom power supply: with the suggested value of 3.3 kΩ about 12 V should appear across the microphone.

Construction

Thanks to the straightforward circuit and the small components it is possible to design a very small circuit board (see **Figure 4**). Measuring just 16 mm by 17 mm, it is the smallest I have ever worked on. It fits neatly in the USB stick, which you can see opened up in **Figure 5**. In order to be able to use the yellow input socket it is necessary to desolder the existing 3.3 V phantom power series resistor, which is marked 'R' in the figure.

Figure 6 shows the stick with the board fitted. The components are tiny, but I found it fairly easy to solder the 0603-size resistors and capacitors by hand. The two coils, both 0805-size (their DC resistance is not critical in this application), are even more straightforward. The only component that is tricky to solder is IC3. It is important to note that the best point to pick up the ground connection is from the furthest pin on the microphone socket (white wire); the 13 V phantom power supply is taken to the microphone socket on the green wire. Once all is ready and the stick is plugged in to a 5-V supply, you should be able to measure 13 V at that point. Finally refit the plastic lid and there you have it: a compact external sound card that can provide phantom power suitable for supplying a low-cost large-diaphragm microphone.

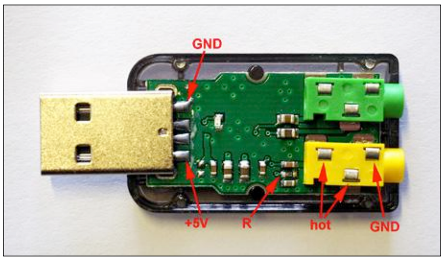


Figure 5. Inside the USB sound card. The series resistor marked 'R', which provides the original 3.3-V phantom supply, must be desoldered.

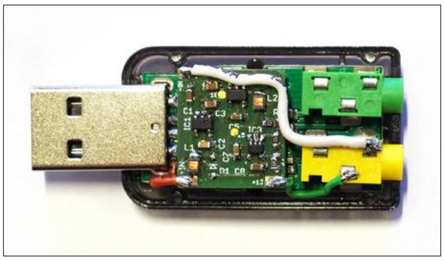


Figure 6. The assembled phantom power supply wired into the USB sound card.

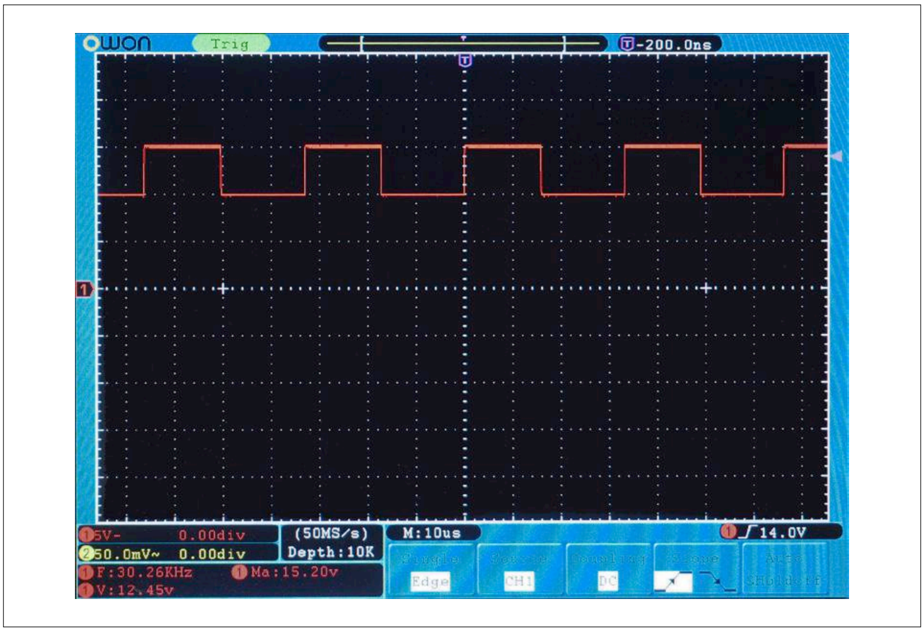


Figure 7. Oscilloscope trace of the signal on switching capacitor C5.

Final thoughts

When provided with a test input voltage of exactly 5.0 V my prototype circuit produced 14.96 V at the input to the linear regulator and, with the component values indicated, 13.2 V at its output. The quiescent current consumption was just an economical 350 µA. The circuit is short-circuit proof: not even 10 mA will flow if you bridge the output. **Figure 7** shows the switching signal on capacitor C5 at pin 5 of IC2. The switching frequency is 30.26 kHz, which is within the expected range. When powered for real with 5.1 V the peak voltage across C5 was 15.2 V, again as expected, and the circuit has given trouble-free operation for several months. It is usually not necessary to install a driver for the USB sound card as both Windows and OS X come with a suitable driver already installed. And if you want to modify the circuit for your own purposes, you will find the layout design files available in Eagle 7 format for free download from the Elektor web page created for this article [4]. ◀

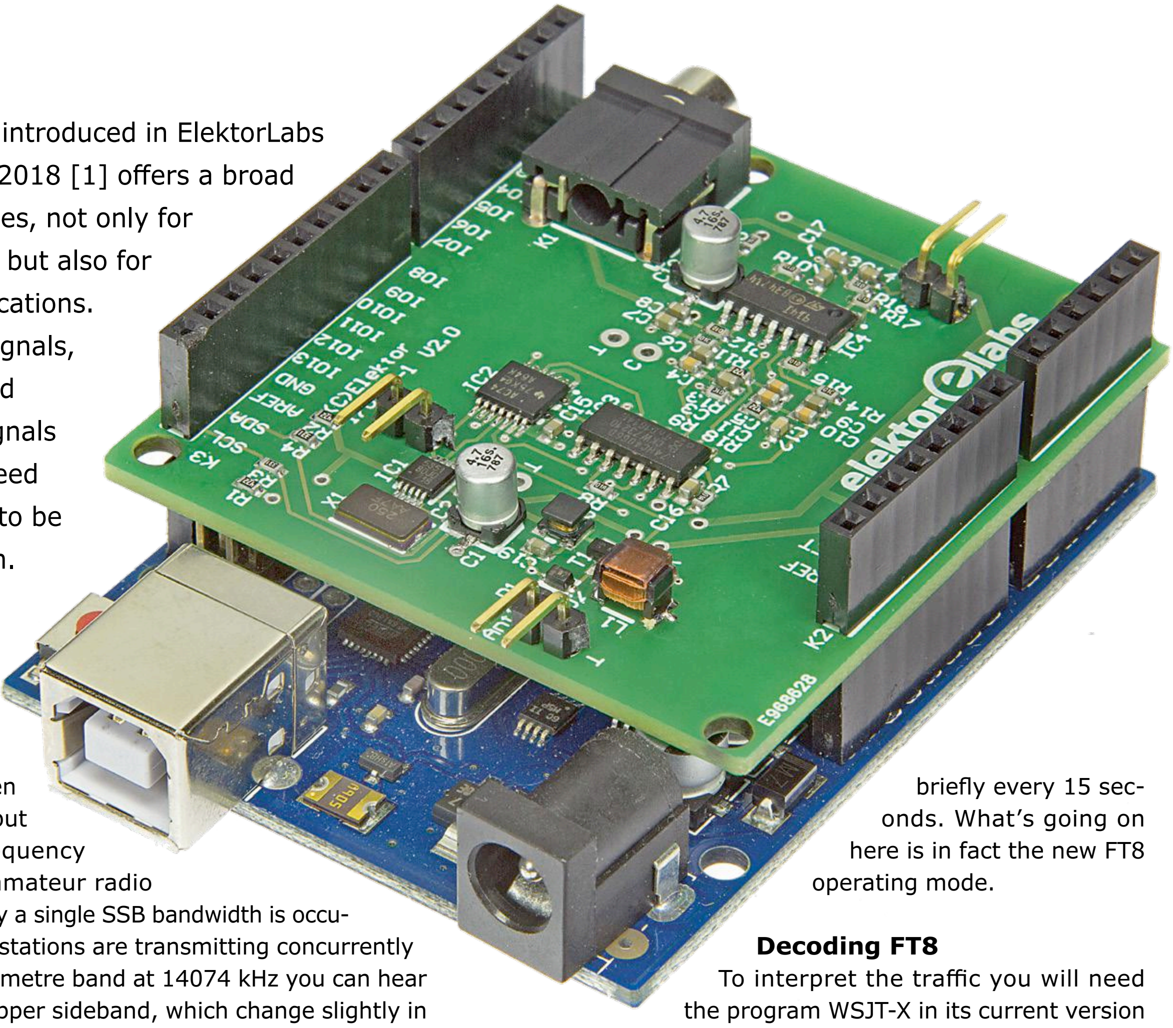
(180165-03)

Elektor SDR Shield 2.0 (2)

Digital data transmissions on the short waves

Burkhard Kainka

The SDR Shield 2.0 introduced in ElektorLabs Magazine edition 4/2018 [1] offers a broad range of opportunities, not only for broadcast reception but also for amateur radio applications. You can hear SSB signals, also listen to CW and numerous digital signals for which you will need additional software to be able to decode them.



Lately people have been discovering a narrow but particularly active frequency spot on many of the amateur radio bands. In each case only a single SSB bandwidth is occupied, on which several stations are transmitting concurrently (**Figure 1**). On the 20-metre band at 14074 kHz you can hear strange tones on the upper sideband, which change slightly in rapid succession. You'll also notice how the signals are silenced briefly every 15 seconds. What's going on here is in fact the new FT8 operating mode.

Decoding FT8

To interpret the traffic you will need the program WSJT-X in its current version (1.8.0). You can download this for Windows,

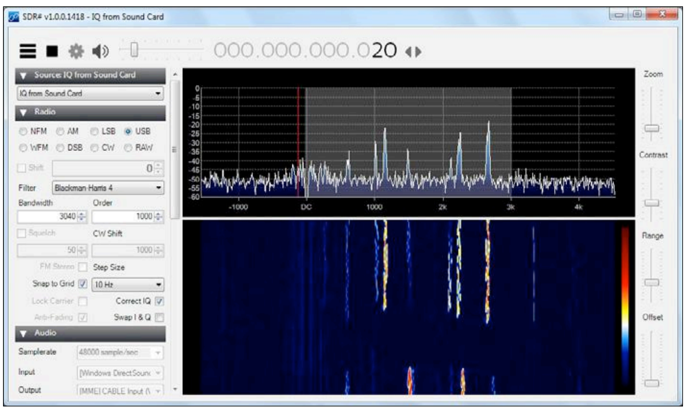


Figure 1. FT8 signals on the 20-m band.

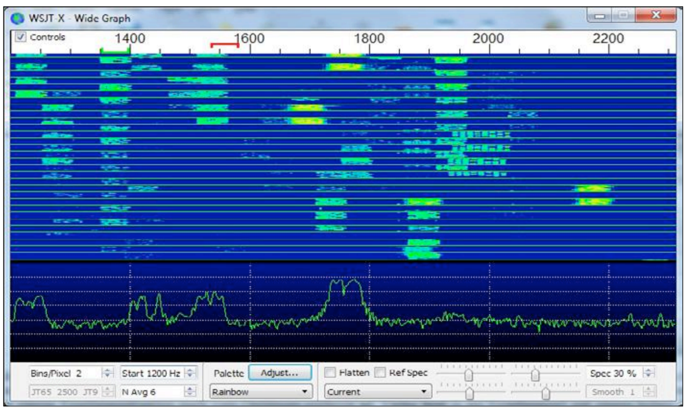


Figure 2. FT8 signals in their timeframe.

Linux and Mac machines at [2]. Installation is plain sailing. The crucial settings can be found in the (not very easy to find) menu by pressing the F2 key. WSJT-X works in a number of modes and for our purpose you need to select FT8. The software produces its own spectrum display (**Figure 2**). FT8 signals can be recognised by their width of around 50 Hz, in which eight frequencies make up an FSK signal. Additionally, each station occupies two transmission blocks, a scant 15-s long, per minute. Its partner station replies during the intervening time. A typical radio connection lasts for only one minute more or less.
If the level lies in the middle range, messages should be decoded. It's essential, however, that the system time is set absolutely exactly. If nothing is decoded, this is usually a problem with the PC clock, which still needs to be synchronised. As soon as data is displayed, the current time deviation DT is also displayed in fractions of a second (**Figure 3**).
If everything is installed correctly, you can now sit back and, without needing to make further frequency searches, quietly observe who is contacting whom. Of particular interest is your own station's signal-to-noise ratio in dB compared with those reported by the stations it's in contact with. In this way you get an impression of how far removed your own receive set-up is from the optimum. This depends largely on your receive location, which may suffer a high level of QRM (interference), whilst other stations may be enjoying better conditions and fully optimised antennas.
You can check out the system even without a transmitter or amateur radio licence, so to speak, by clicking on a station that is calling CQ. The software sets up the five text messages, Tx1 to Tx5. If you have arranged everything correctly and press Enable-TX, the transmit signal will be heard direct through the loudspeaker. Using the PWR control you can select the volume, just as you would the transmit power of an SSB transmitter. With good hearing you can also detect the eight adjacent frequencies.

Decoding WSPR

The WSPR transmission technique (Weak Signal Propagation Reporter, pronounced 'whisper') was developed to span long distances with low power and minimal bandwidth. There are always many stations active, who report their reception results to [3]. Very rapidly you can gain an overview of where a signal can be received and how 1000 km (600 miles) is genuinely possible with only 10 mW. With 200 mW you can reach anywhere in Europe without difficulty.
Signals can also be read using WSJT-X. With a bandwidth of only about 5 Hz, they are extremely narrow and in total occupy only 200 Hz (from 1400 Hz to 1600 Hz). The spectrum in **Figure 4** indicates that the calibration of the receiver is correct, which is quite a challenge.
Stations received are displayed with the signal-to-noise ratio achieved, their call sign, location, transmit power and the distance in km. For this purpose, your own location in the configuration must be specified as a QTH locator, which you can determine, for example, at [4]. The locator code for my receiver in Essen was JO31MK.
Among the stations received (**Figure 5**) were several with a transmit power of only 200 mW or less, with which many managed distances of 100 km. The check mark (tick) for 'Upload spots' should be set only if you are logged into Wsprnet. Then your own reception results will be stored in the database. Otherwise you may prefer to listen in privately. But it is always interesting to refer to the Wsprnet website to see which other stations were heard.
The example in **Figure 6** shows where the station DJ5AM was received. A comparison of the displayed reception frequency

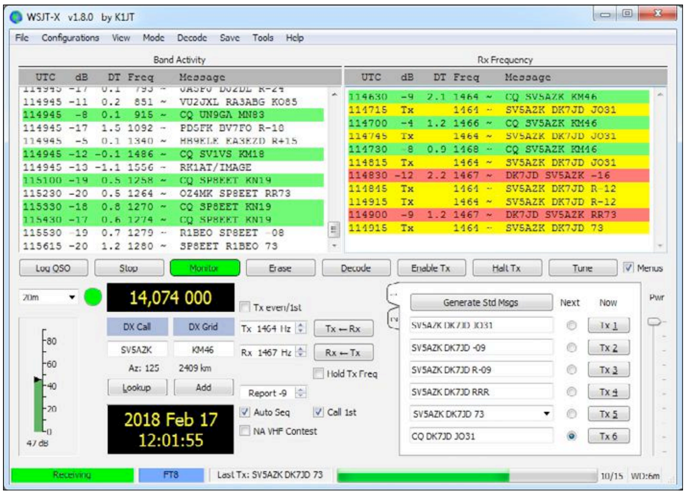


Figure 3. FT8 radio traffic across Europe.

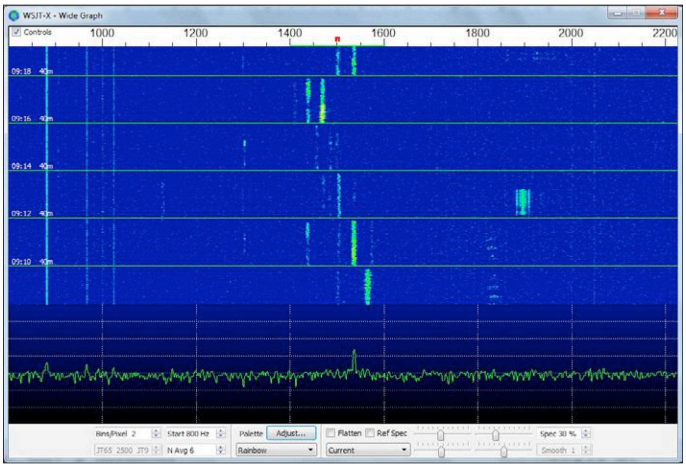


Figure 4. Waterfall diagram of WSPR signals.

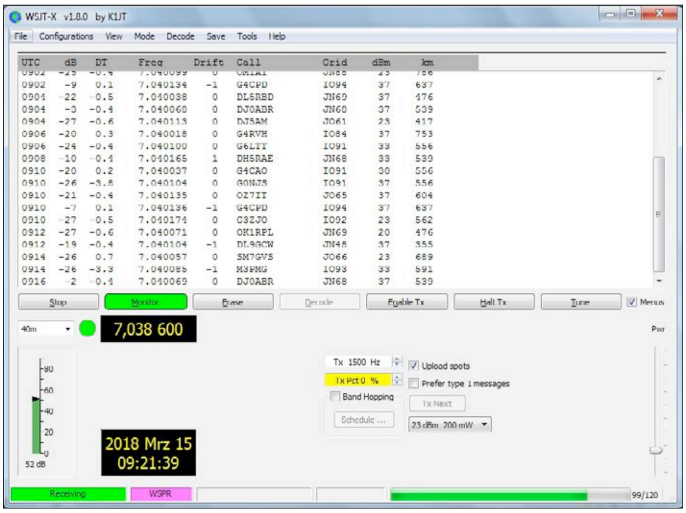
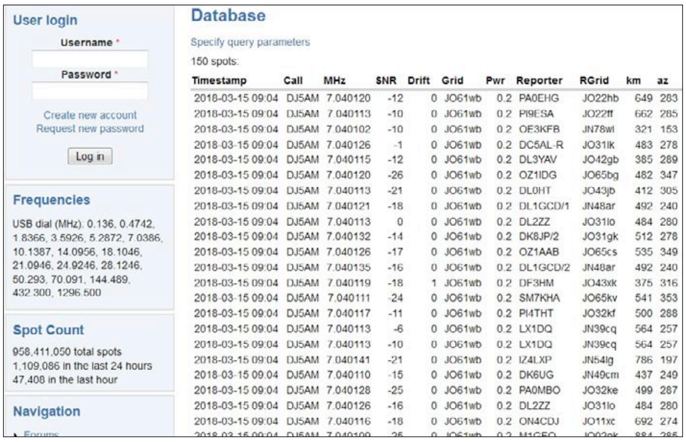


Figure 5. WSPR reception stations.



Timestamp	Call	MHz	SNR	Drift	Grid	Pwr	Reporter	RGrid	km	az
2018-03-15 09:04	DJ5AM	7.040120	-12	0	JO61wb	0.2	PA0EHG	JO22hb	649	283
2018-03-15 09:04	DJ5AM	7.040113	-10	0	JO61wb	0.2	PI9ESA	JO22ff	662	285
2018-03-15 09:04	DJ5AM	7.040102	-10	0	JO61wb	0.2	OE3KFB	JN78wl	321	153
2018-03-15 09:04	DJ5AM	7.040126	-1	0	JO61wb	0.2	DC5AL-R	JO31lk	483	278
2018-03-15 09:04	DJ5AM	7.040115	-12	0	JO61wb	0.2	DL3YAV	JO42gb	385	289
2018-03-15 09:04	DJ5AM	7.040120	-26	0	JO61wb	0.2	OZ1IDG	JO65bg	482	347
2018-03-15 09:04	DJ5AM	7.040113	-21	0	JO61wb	0.2	DL0HT	JO43jb	412	305
2018-03-15 09:04	DJ5AM	7.040121	-18	0	JO61wb	0.2	DL1GCD/1	JN48ar	492	240
2018-03-15 09:04	DJ5AM	7.040113	0	0	JO61wb	0.2	DL2ZZ	JO31lo	484	280
2018-03-15 09:04	DJ5AM	7.040132	-14	0	JO61wb	0.2	DK8JP/2	JO31gk	512	278
2018-03-15 09:04	DJ5AM	7.040126	-17	0	JO61wb	0.2	OZ1AAB	JO65cs	535	349
2018-03-15 09:04	DJ5AM	7.040135	-16	0	JO61wb	0.2	DL1GCD/2	JN48ar	492	240
2018-03-15 09:04	DJ5AM	7.040119	-18	1	JO61wb	0.2	DF3HM	JO43xk	375	316
2018-03-15 09:04	DJ5AM	7.040111	-24	0	JO61wb	0.2	SM7KHA	JO65kv	541	353
2018-03-15 09:04	DJ5AM	7.040117	-11	0	JO61wb	0.2	PI4THT	JO32kf	500	288
2018-03-15 09:04	DJ5AM	7.040113	-6	0	JO61wb	0.2	LX1DQ	JN39cq	564	257
2018-03-15 09:04	DJ5AM	7.040113	-10	0	JO61wb	0.2	LX1DQ	JN39cq	564	257
2018-03-15 09:04	DJ5AM	7.040141	-21	0	JO61wb	0.2	IZ4LXP	JN54lg	786	197
2018-03-15 09:04	DJ5AM	7.040110	-15	0	JO61wb	0.2	DK6UG	JN49cm	437	249
2018-03-15 09:04	DJ5AM	7.040128	-25	0	JO61wb	0.2	PA0MBO	JO32ke	499	287
2018-03-15 09:04	DJ5AM	7.040126	-16	0	JO61wb	0.2	DL2ZZ	JO31lo	484	280
2018-03-15 09:04	DJ5AM	7.040116	-18	0	JO61wb	0.2	ON4CDJ	JO11xc	692	274

Figure 6. Reception reports from multiple receiving stations.

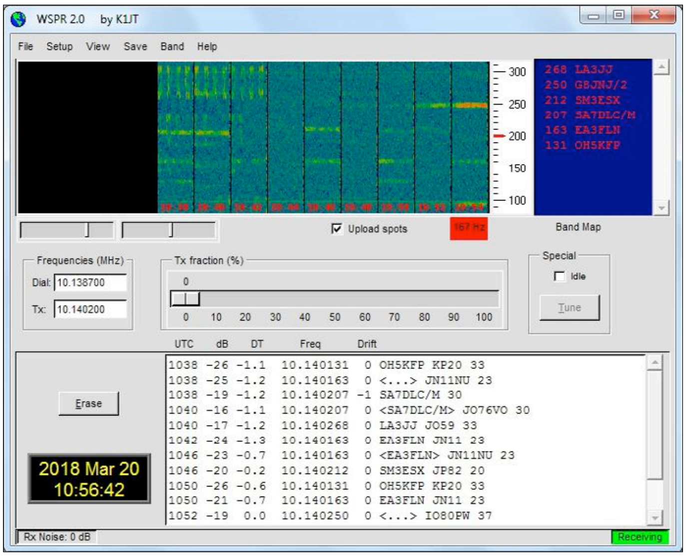


Figure 7. WSPR signals direct and via SDR#.

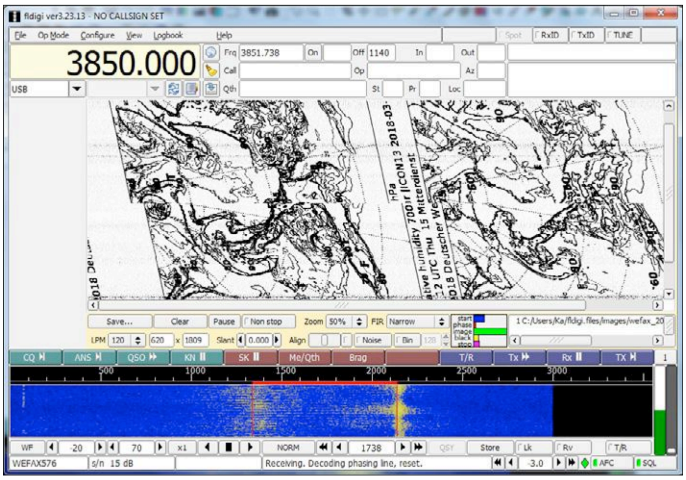


Figure 8. Weather chart with minor errors.

proves that my own display of 7040113 Hz is well in range. The calibration is therefore in line with achievable accuracy. In my first reception attempts, despite my initial calibration, a deviation of 100 Hz was discovered, which was then easy to correct. A comparison of the measured signal-to-noise ratios is also interesting. On average, other receive stations at comparable distances had signal-to-noise figures that were better by 10 dB to 20 dB, which indicates better antenna positions with less background noise. In my case, a shortened dipole plus balun were used in the garden. Despite this there is still plenty of wideband noise on the antenna input.

Virtual audio cable

Many programs that have been developed specially for amateur radio require the use of an SSB receiver. Usually these provide reception from 300 Hz to 2700 Hz. You switch the receiver to USB and feed the receiver's audio signal into the sound card. The SDR Shield does have a connection for the sound card but as such it is not a fully-fledged SSB receiver. Nevertheless, you can use it for this purpose. There are two methods for this: a direct connection or else using SDR software such as SDR# or G8JCFSDR.
The direct connection is the simplest. You set the VFO so that the wanted signal lies within the range of the SSB bandwidth. The Shield then produces the desired SSB signal direct, albeit with the addition of the other sideband. The decoding software does not normally extract an IQ signal, only a mono signal. In many situations this works very well nevertheless, because there is no signal present in the other sideband.
Sometimes the better method is to use an SDR program with an IQ input and pass the already processed audio signal to the decoder. In this way you achieve effective suppression of the unwanted sideband. But now a new problem appears. You really need an extra sound card or a second PC, since the first sound card is already tied up with the SDR Shield. A solution lies in a virtual audio cable, such as the VB CABLE virtual audio device [5]. In this way you get an extra output device and in the SDR program you enter the *Cable Input* instead of the usual sound card outputs. Correspondingly, in the decoding software you select the *Cable Output* instead of the sound card input.
Theoretically halving the noise bandwidth should produce a signal improvement of 3 dB. A comparison with WSPR signals confirms this differential exactly. On occasions the improvement was even clearer, because other signals present in the wrong sideband were now suppressed. In other cases the signal was definitely degraded, as the programs involved did not work optimally with the virtual audio cable.
Teaming up SDR# with WSPR2.0 produced good results. **Fig-**

Web Links

[1] www.elektormagazine.com/160577
[2] https://physics.princeton.edu/pulsar/k1jt/wsjtx.html
[3] http://wsprnet.org/
[4] www.hb9eyz.ch/g_locator.php
[5] www.vb-audio.com/Cable/
[6] https://sourceforge.net/projects/fldigi/

ure 7 shows the difference made. First the signal was received direct from the sound card. Here you can see the interfering FT8 signals in the wrong sideband. Then we switched over to the virtual audio cable, that is, to the already processed USB signal. In that way the desired signals were decoded free of interference.
If you feel like working with G8JCFSDR, you cannot use direct tuning of the VFO to the USB frequency because the program shows its centre IF of 12 kHz in the middle of the spectrum. However, the decoded USB signal can be routed to any program of your choice for further processing.

Digital operating modes with fldigi

The ‘classic’ operating modes in amateur radio are CW and SSB. Somewhat later radio teletype (RTTY) came along. In the meantime numerous new digital operating modes have appeared. Many of these can be received using fldigi, which you can also obtain on the Net [6]. Besides CW, the program also decodes RTTY and the much-loved PSK31, weather fax plus many other signals.
It’s not only amateur radio signals that can be decoded with fldigi. **Figure 8** shows a reception test of a weather chart of the German weather service. The result indicates some fine tuning is necessary here.
Morse telegraphy can be read too. This works well with relatively strong signals. On the other hand, weak Morse signals are one of the last bastions where humans remain superior to machines. Nevertheless, fldigi can help if you cannot keep up with high Morse speeds or are just starting out. After a while the program adjusts to the current transmission speed. During any gaps in transmission, it attempts to read something from the noise, which can produce somewhat hit or miss results. **Figure 9** shows an example with, in places, perfectly legible decoding.
Gaining in popularity lately is PSK31, a phase shift technique of 31 Baud, which corresponds more or less to the speed of typing on a keyboard. Because PSK31 corresponds to a bandwidth of only 31 Hz, its dependability is even better than CW, for which you need around 200 Hz. The outcome is you can make contacts even with strong noise and low transmit power (**Figure 10**).
PSK31 uses phase shift keying and shifts the phase by 180 degrees for smooth transitions. The oscillogram (**Figure 11**) shows that the amplitude has a zero crossing at each phase shift. The bandwidth thus corresponds only to the baud rate. Incidentally, the signal comes from fldigi, which generates transmit signals via the sound card, without connection to a transmitter.

(180284)

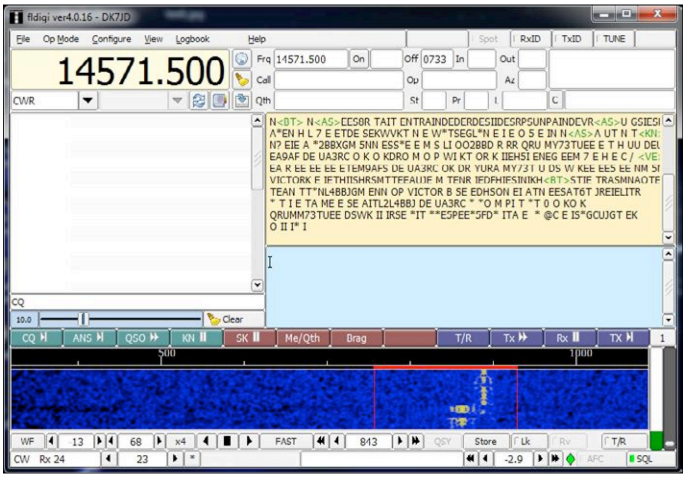


Figure 9. Decoding Morse code signals.

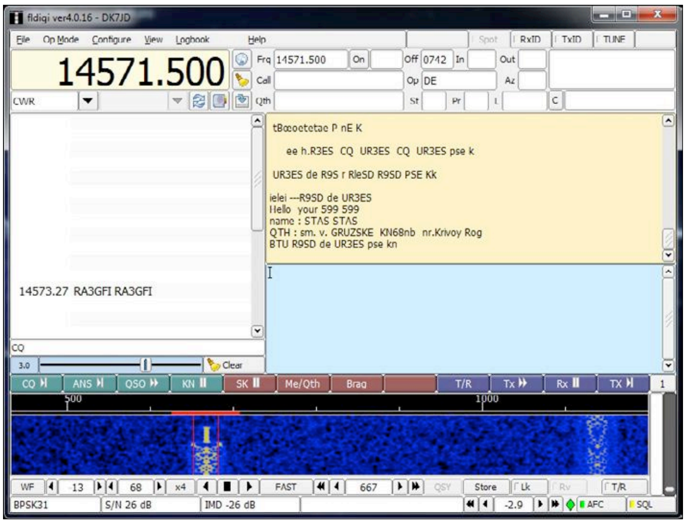


Figure 10. Decoding a PSK31 signal.

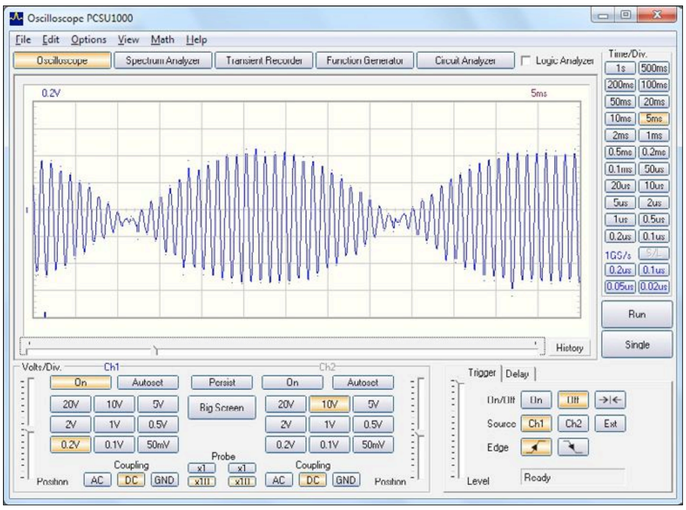
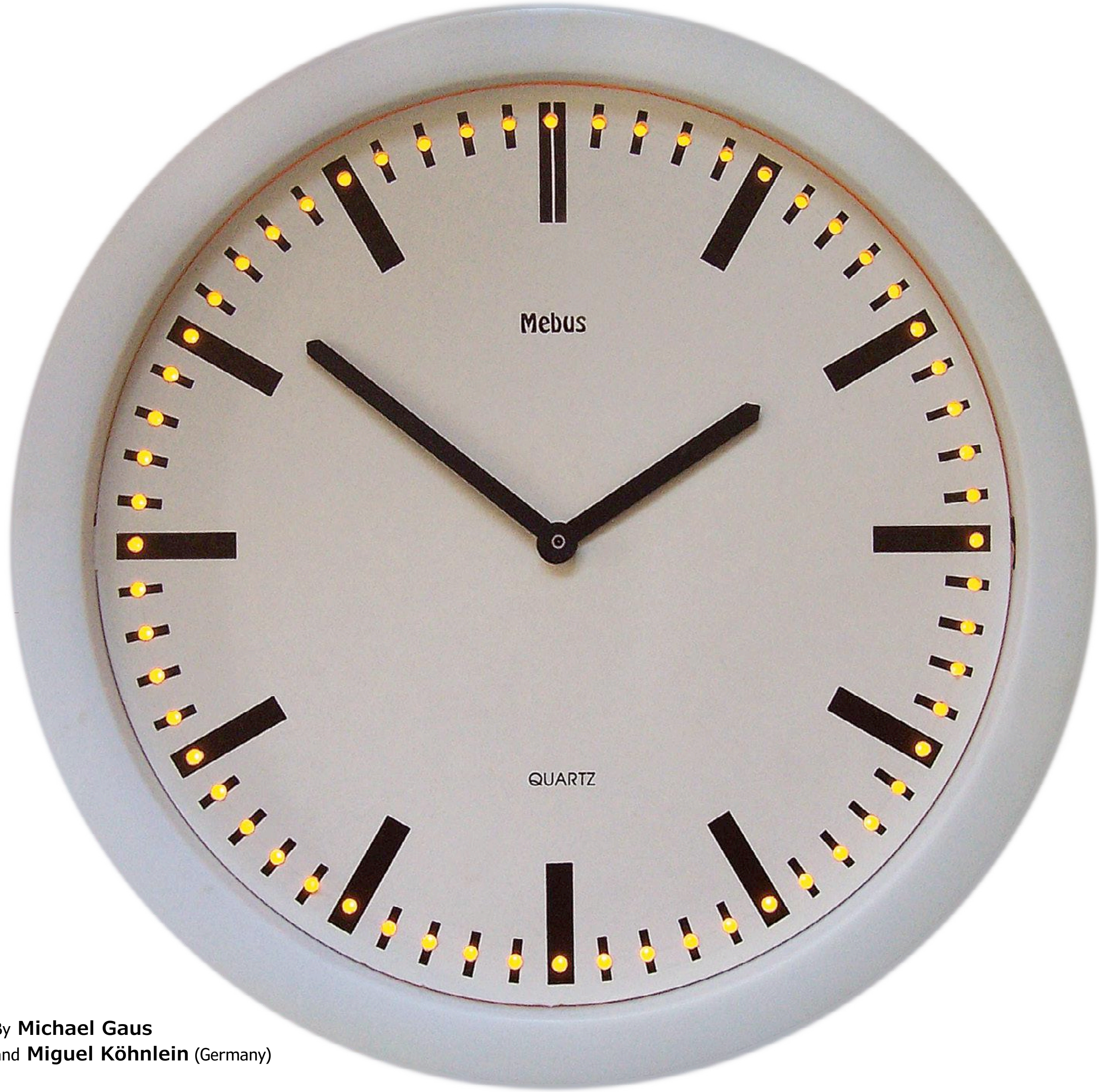


Figure 11. Oscilloscope image of a PSK31 signal.

HOMELAB PROJECT

Nostalgic LED Clock

74xx around the clock

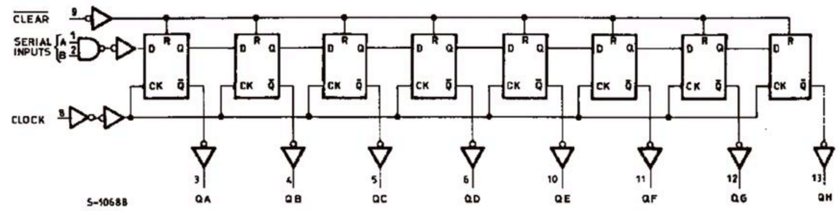


By **Michael Gaus**
and **Miguel Köhnlein** (Germany)

In this project, an analogue wall clock (train station clock) is fitted with 60 LEDs that light up to simulate the movement of a seconds hand around the clock face. To keep to the spirit of this project you should try hard and resist the temptation to use a microcontroller; just rely on good-ol' 74xx TTL logic to build a stand-alone clock. For those with less willpower the circle of LEDs can be microcontroller or PC controlled, this allows you to be a bit more creative with lighting sequences and special effects.

The clock can be configured for one of two modes of operation using jumpers. In **mode 1** using only 74xx chips, all the LEDs light up after switch on. At 1 second intervals one LED after another is switched off, after a minute all LEDs are off. In the following minute, the LEDs are switched on again one after another at intervals of one second, until all the LEDs are lit up. This effect is achieved by using an inverter in the shift register's feedback path. For authenticity we've used a 4060 binary counter chip to provide the clock signal.

Mode 2 is a nod to modern times: all 60 LEDs can be controlled by an external microcontroller or PC to generate any desired lighting patterns. The chain of 74164 shift registers receives the CLK and DATA signals from the external controller via connector J1. Jumpers JP1 and JP2 must link pins 2-3 and JP3 can be left open in this mode.



FUNCTION TABLE

INPUTS				OUTPUTS		
$\overline{CLEAR}$	CLOCK	A	B	Q_A	Q_B ...	Q_H
L	X	X	X	L	L	L
H	L	X	X	Q_{A0}	Q_{B0}	Q_{H0}
H	↑	H	H	H	Q_{An}	Q_{Gn}
H	↑	L	X	L	Q_{An}	Q_{Gn}
H	↑	X	L	L	Q_{An}	Q_{Gn}

H = high level (steady state), L = low level (steady state)
X = irrelevant (any input, including transitions)
↑ = transition from low to high level.
Q_{A0}, Q_{B0}, Q_{H0} = the level of Q_A, Q_B, or Q_H, respectively, before the indicated steady-state input conditions were established.
Q_{An}, Q_{Gn} = the level of Q_A or Q_G before the most-recent ↑ transition of the clock; indicates a one-bit shift.

Figure 1. The 74LS164 block diagram and truth table. Courtesy Texas Instruments TTL data book circa 1974.

Old friends, reunited

The 74 series of logic IC's aren't used much these days so let's take a closer look at the 74LS164 shift register we will be using in this design. As a quick reminder (or as an introduction to budding engineers) the chip was launched back the early 1970s. Inside the IC (**Figure 1**) are eight flip-flops with common clock and reset lines. The flip-flops are cascaded: Each data input is connected to the Q output of the previous flip-flop. The inverted output signal of each flip-flop connects to a pin to the outside. Altogether, the user has eight data outputs QA through QH in parallel. The data input of the first flip-flop is the serial data input to the system. There are actually two serial inputs A and B which drive a NAND gate, these two inputs are connected together in our circuit.

At each rising edge at the clock input CLK, the logic state of each flip flop is moved or shifted one position right to the next flip flop in the chain towards QH. At the same time, the state of the input signal is loaded into the first flip-flop with its output at QA. The shift register truth table in **Figure 1** also shows when a logic low level is put on the CLEAR input all the flip flops are cleared and their outputs go low.

The internal circuitry of the 74LS164 suggests you can cascade multiple shift register ICs to give more than one eight-bit wide output. All that's necessary is to connect the last output QH to the A / B input of the next 74LS164. The CLK and CLEAR inputs are then connected together and used synchronously.

As the circuit diagram in **Figure 2** shows we have used a total of eight 74LS164s in series for the LED clock. This effectively builds one large shift register with a 64-bit wide output. Eight LEDs are connected to each 74LS164 except for the last shift register, which only drives four LEDs. Each LED is connected to +5 V via a 1-kΩ series resistor so that they light up when the corresponding flip flop output goes low. The shift register is driven by a 1-Hz clock signal generated by the counter chip.

A Binary counter generates the clock

Let's first look at the clock working in stand-alone „normal mode" or mode 1. All three jumper positions link pins 1 and 2. A reset network, consisting of R65 and C6, provides a reset pulse at switch on to ensure all the shift registers are cleared. Pushbutton S1 provides a manual reset. In this initial state, all the shift register outputs are low (i.e. all LEDs are lit). The last output for LED 60 is fed back to the input of the first 74LS164 via the 7404 inverter (IC9A) and jumper position JP1. The A / B input of IC1 is therefore at a logic high. At the next rising clock edge, this high level is shifted to the output QA, so that the LED goes out. This logic level is now at the input to the next flip flop, which will change state when the next clock edge arrives. The change of state propagates along the shift register with each clock pulse. At the 60th clock edge, even though the output of LED60 goes high and the last LED goes off, the level at the A / B input of IC1 now becomes low, so that for the next 60 seconds the low level is sent down the shift register chain to switch on each LED in turn.

Clock pulses are generated by the 14-stage binary counter IC10 (a 74HC4060). Although this chip performs a completely different function, internally (**Figure 3**) it's also made up of flip flops like the shift register chip. The 74HC4060 uses a chain of toggle-type flip-flops, these have their data input connected to the flip flop output /Q . Each flip-flop changes state at every falling edge of the CLK input the output signal is therefore a waveform half the frequency of the CLK signal. Each flip flop in the chain provides this divide-by-two function so that at the 14th flip-flop a signal with the frequency of $1/2^{14}$ of the clock signal appears.

The 74HC4060 binary counter has featured in many digital clock schematics over the years where it is used to provide the 1 second reference timebase. It just needs a 32,768 Hz clock crystal connected at its clock input so that at Q13 a frequency of ... no, alas not 1 Hz, but $32768/2^{14}$ = 2 Hz is output. The signal needs to be further divided by 2 to give a 1 Hz signal, a 7474 D-type flip-flop (IC11A) does this job. It has its $\overline{Q}$ output connected to its D input to form a toggle flip flop. Via jumper JP2 the resulting 1-Hz signal is fed to the shift register clock inputs.

Modern times

Maybe you're getting a bit tired of all this nostalgia, you can of course opt for mode 2 and control the LED clock using a real state-of-the-art microcontroller (or PC). First disconnect the clock signal by removing the two links between pins 1 and 2 on both JP1 and JP2 and placing them between pins 2 and 3. This disconnects the on-board clock generator and also the feedback path to the first shift register. You can also (but don't need to) remove jumper JP3 if you want the circuit to be reset by a signal from the microcontroller.

In mode 2, all 60 LEDs are externally controlled. The signals CLK and DATA (and optionally also CLEAR) are supplied to the shift register via connector J1. When you write the program for the controller, set a timer interrupt to generate the clock signal and remember that the DATA signal level must be set and stable before the positive edge of the CLK signal is supplied. The video [2] shows some possible lighting and display effects, which can be achieved using external control.

Power and assembly

The 74LS family of ICs is quite demanding when it comes to supply voltage and signal levels. The power supply should be in the range 4.75 to 5.25 V. Under no circumstances may the signal level exceed the supply voltage! This must be taken into account especially when external control signals, for example from a PC are interfaced, this is also relevant when interfacing microcontrollers that are not operated at 5 V. Signal levels above 2 V are detected by the LS logic as High, levels below 0.8 V as Low.

The supply voltage is generated by a 7805 linear regulator. The input voltage V_{IN} must be at least 8 V for this regulator, but don't go much higher than 15 V otherwise power dissipation in the regulator becomes excessive. It may be necessary to fit a small heatsink to IC12 if this is the case.

The clock face has 60 holes drilled at the seconds positions. Make sure the drill size is the same at the LED body diameter so that the collar at the base of the plastic lens stops the LED from falling through. They can then be fixed in position using two-part adhesive (**Figure 4**). Each LED cathode is connected to a shift register output using enamelled copper wire. The series resistors are soldered directly to each LED anode and to the inner ring of copper wire connected to +5 V. The

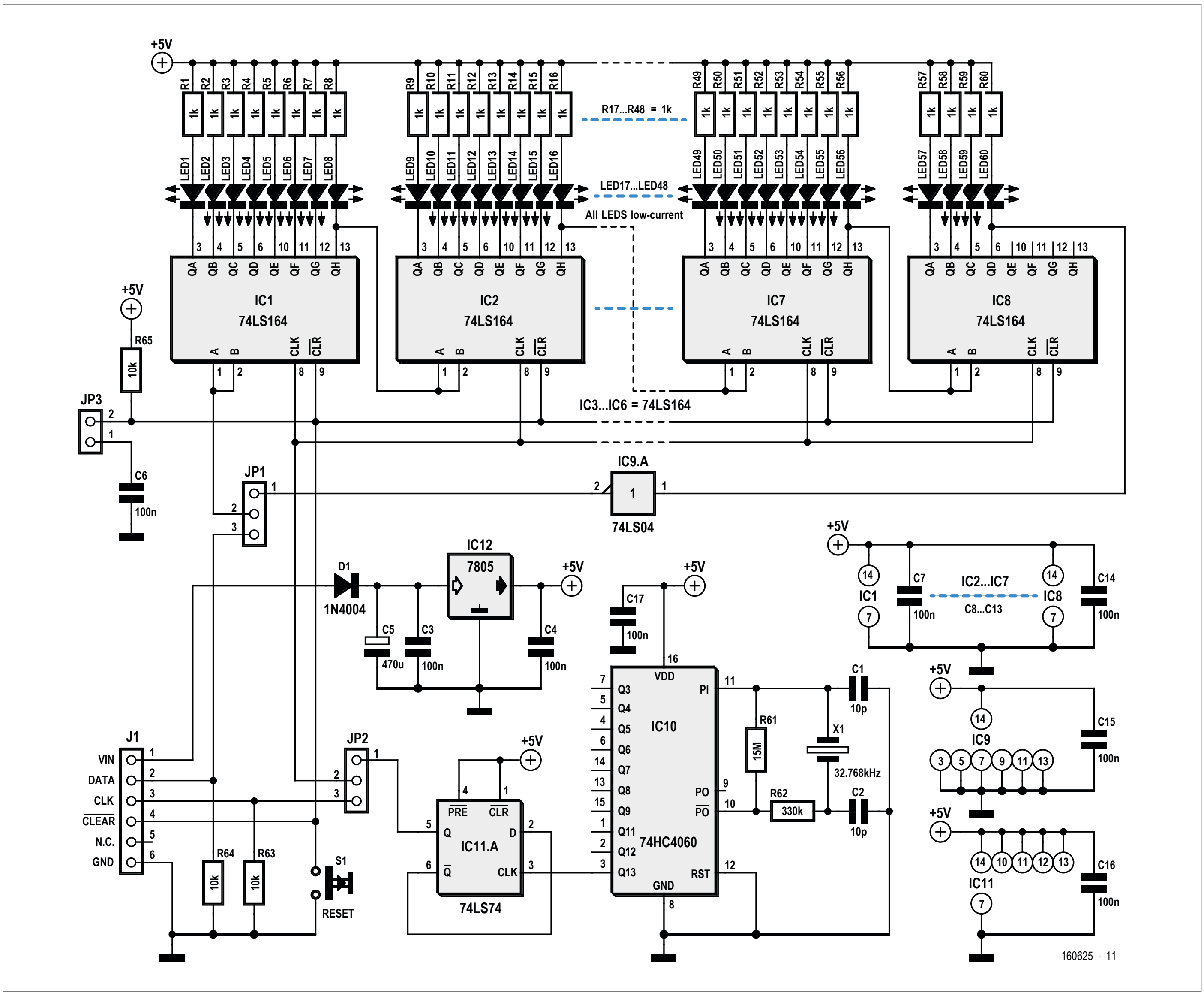


Figure 2. The 74xx TTL clock circuit diagram.

shift register ICs are mounted on small squares of perf board and distributed around the clock perimeter, this helps reduce wire lengths to the LEDs. Only wires carrying the DATA and CLK signal need to run all the way 'around the clock'.

(160625 || 180419)

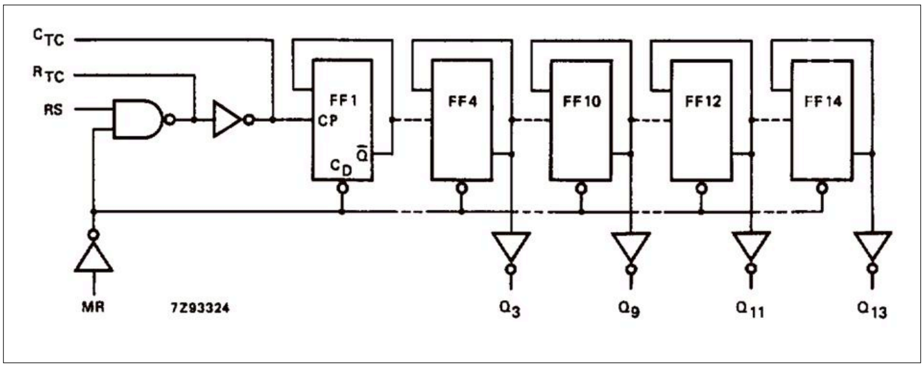


Figure. 3. Internal block diagram of the 74HC4060 binary counter (from a recent Philips Semiconductors datasheet).

Web Links

[1] Video 1: www.youtube.com/watch?v=LTaV84mTj2w

[2] Video 2: www.youtube.com/watch?v=UEWgwRypzHk

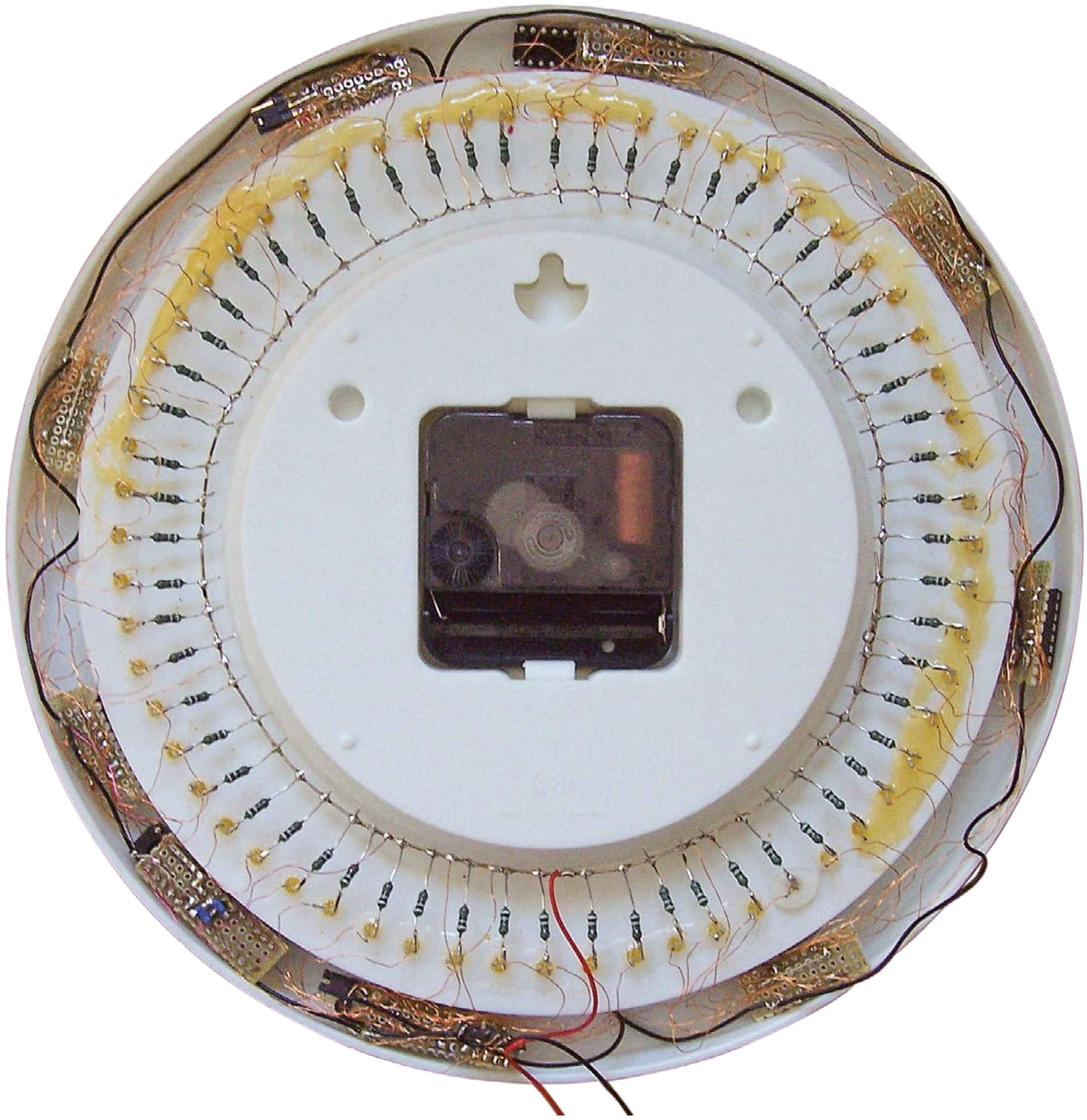
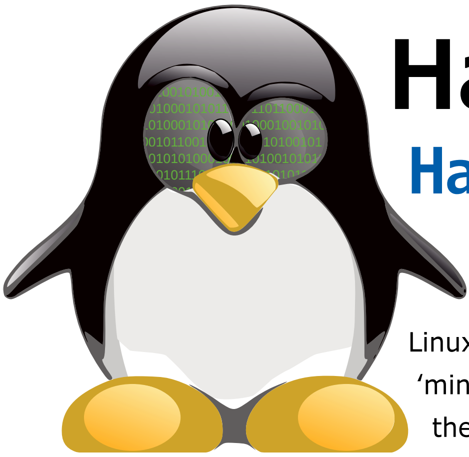


Figure. 4. Wiring the 60 LEDs to the shift registers.



HangTux Mk. 2

Hangman on the Raspberry Pi

By **Roy Aarts** (Elektor Labs) & **Thijs Beckers** (Elektor)

Linux will always face a challenge. However, the emergence of many 'minicomputers' such as the Raspberry Pi (and variants), the Beagle Bone, the DragonBoard and the Odroid, which all use a Linux operating system, has given a healthy boost to the number of Linux users. If you haven't joined them yet, here is an interesting and simple application to get your teeth into: hangman.

In the 'HangTux' article published in the January 2013 edition [1] we showed the use of the Elektor Linux board to play a Hangman-ish game. The game itself ran on the board, while a Linux PC acted as the display via a terminal. The score was also shown on the Elektor Linux board via an LED VU-Meter. The following article builds on this concept, although the Elektor Linux board and the Linux PC have been replaced with a Raspberry Pi. This circuit does exactly the same as the above-mentioned HangTux: it plays a game of hangman in Linux. Here, the Raspberry Pi should be configured as a 'mini-PC', with a display, keyboard and mouse, using a Raspberry Pi compatible Linux distribution.

Hardware

The hardware hasn't been changed and we've shown the schematic again in **Figure 1**. The LEDs are driven by an 8-bit Microchip MCP23S08 microcontroller that communicates with the Raspberry Pi via an SPI link. The printed circuit board we designed previously is easy to populate due to the use of leaded components and the DIP18-packaged microcontroller. Even a beginner in soldering shouldn't find it difficult to construct.

The board takes it supply from the 3.3-V bus on the Raspberry Pi. This works well but do keep in mind that the current consumption increases slightly. If you were using a lower power adapter for the Raspberry Pi, it would be better if you replaced it with a more powerful version when you're connecting the board.
The MCP23S08 communicates with the Raspberry Pi via the SPI bus, using 24-bit words. The first byte addresses the chip and indicates if we want to read or write. We only need to write, so the first byte is '0100 0000'. For this reason, address bits A0 and A1 are also always '0'; pins 4 and 5 of IC1 are connected to Ground. The second byte is the write address of the register. We use just two registers: IODIR (I/O Direction) to configure the pins as outputs, and OLAT (Output Latch) to set the required bit pattern on the

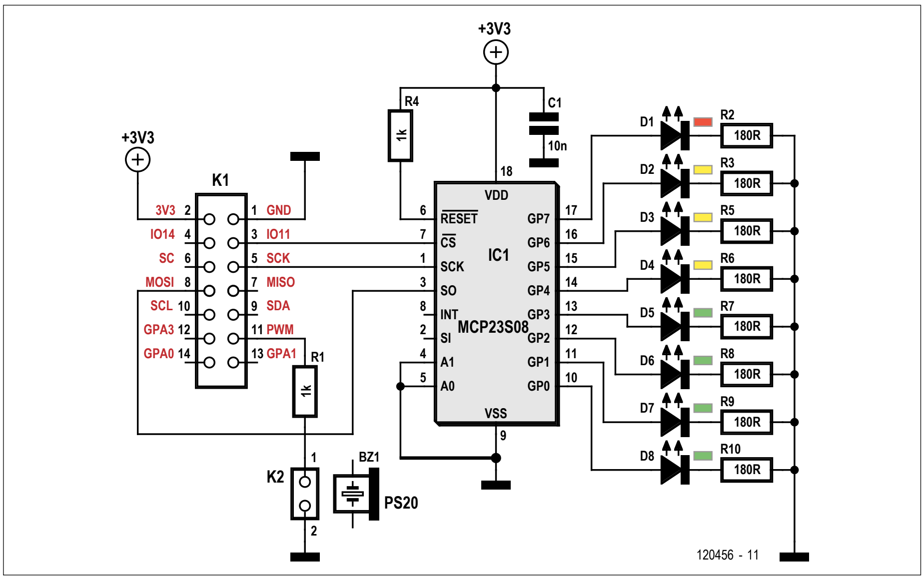


Figure 1. The circuit hasn't changed. A simple microcontroller controls the scoreboard LEDs.

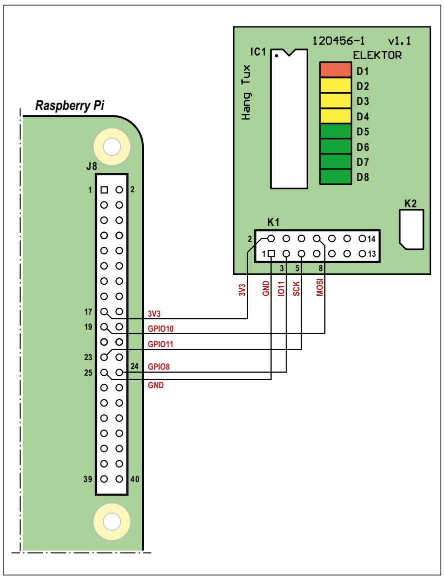


Figure 2. This is how the hardware should be connected to the Raspberry Pi.

Figure 3. The game starts with the main menu.

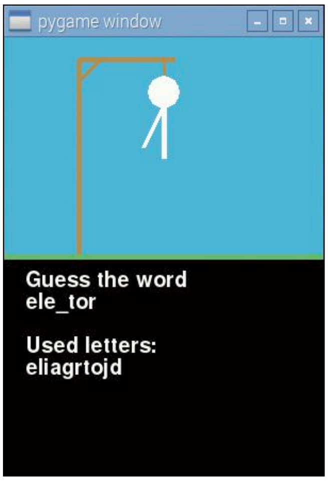


Figure 4. Can you guess what the word is?

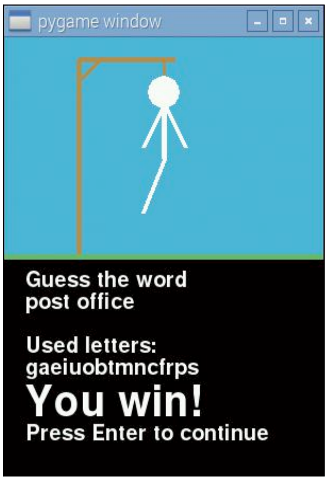


Figure 5. Congratulations! You've won.

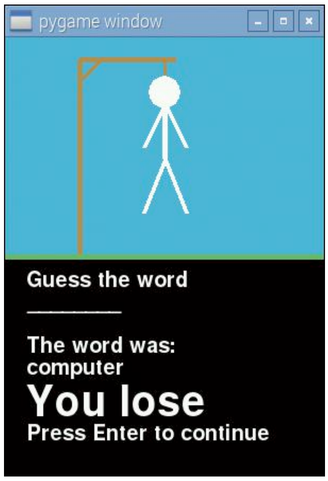


Figure 6. Hard luck — why not try again?

outputs. The CS (Chip Select) input is driven by the Raspberry Pi.

Software

The software, which can be freely downloaded from [2], was successfully written in Python. The installation procedure is detailed below. First of all, the SPI has to be configured. Open a terminal window on the Raspberry Pi and enter the following command:

```
sudo raspi-config
```

Then go to *advanced* → *SPI*. You'll be asked if you want to turn on SPI, to which you should answer 'yes'. Go back to the command prompt and enter the following commands:

```
sudo apt-get update
sudo apt-get install python-dev
python3-dev
git clone git://github.com/doceme/
py-spidev
cd py-spidev
sudo python setup.py install
sudo reboot
```

Copy the *hangman* directory from the download [2] to the Raspberry Pi. You could do this with the help of a USB-stick or using winSCP. The directory should be put in '/home/pi'. Next, reboot the Raspberry Pi.
Once the Raspberry Pi is up and running again, copy the file 'hangman.sh' from the *hangman* directory in the download to the desktop. Then open a terminal window and enter the following commands:

```
sudo cd Desktop
sudo chmod +x hangman.sh
```

Now connect the scoreboard to the Raspberry Pi according to the wiring diagram in **Figure 2**.
You're now ready to begin a game by double-clicking on the hangman.sh file that you copied to the desktop.

Your own words

The game uses 'dictionaries'. In the download we have included English and French dictionaries, but there is no reason why you can't create your own dictionary. To do this, create a simple .txt file, where each of your words is on its own line, and store it as 'dictionary.txt' in the directory where the game is. In contrast to the original game that was limited to 100 words, this version can use an unlimited number of words.
The game always uses the file called dictionary.txt, which means you will have to rename the files when you want to change the dictionary used.

Get ready to play!

The game starts with the main menu (see **Figure 3**). From here you can start a new game or close the program. The game uses the familiar gallows on the screen to show how many letters you've tried, which ones were correct, and how many tries you have left before the stick-man is complete (see **Figure 4**). The LED bar graph is another indicator that shows how many attempts you have left. The LED board is not vital for the operation of the game, but it does act as a good introduction to driving hardware via SPI. And since more and more electronics need a (Linux) operating system, this is a good, but simple, exercise.

(160127)

Web Links

[1] www.elektormagazine.com/120456
[2] www.elektormagazine.com/160127

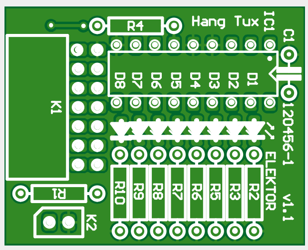
COMPONENT LIST

Resos
R1,R4 = 1kΩ
R2,R3,R5–R10 = 180Ω

Condensatoren:
C1 = 10nF tantalum

Semiconductors
D1 = LED, red, rectangular, 5mm
D2,D3,D4 = LED, yelllow, rectangular, 5mm
D5–D8 = LEDd, green, rectangualr, 5mm
IC1 = MCP23S08, programmed

Miscellaneous
P1 = 2-pin pinheader, 0.1" pitch
K1 = 14-pin (2x7) pinheader, 0.1" pitch
PCB # 120456-1

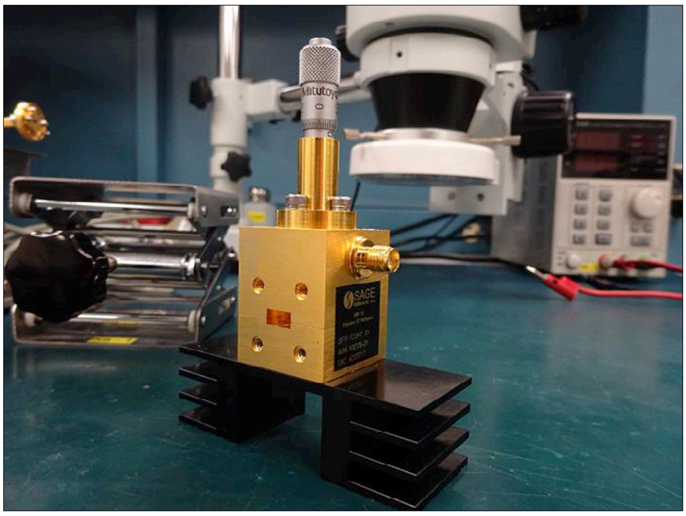


Gunn Diodes

Peculiar Parts, the series

By **Neil Gruending** (Canada)

I’ve talked about different types of diodes in previous *Peculiar Parts* instalments because there just seems to be so many unique types made in the early days of semiconductor devices. This time I wanted to look at Gunn diodes which are common in RF circuits but can barely be classified as a diode at all.
Normally a diode is a two pin device that contains a PN silicon junction. A Gunn diode also a two-pin device but differs because it only contains N silicon junctions since it uses the transferred electron effect and therefore doesn't’t need the electron holes in P silicon. The junctions are made from two heavily doped outer layers that are separated by a lightly doped layer like in **Figure 1**. As voltage is applied to the device, the current through the inner layer increases until the applied voltage is high enough to increase its resistivity and the current will decrease.
This construction gives the transfer characteristic shown in **Figure 2**. A normal diode will continuously increase its current with an applied voltage, but a Gunn diode has a peak and valley points where the current actually goes down as the applied voltage increases which creates a **negative resistance**. This behaviour is called the Gun Effect and is named after the physicist J. B. Gunn who first documented it in 1962 while trying why his measurements contained high-frequency noise.
One common usage for Gunn diodes is to use their negative resistance properties to cancel out the positive resistance of a load to cause spontaneous oscillations at microwave frequencies and so make microwave oscillators. The frequency is controlled by the diode middle layer properties but it can also be tuned externally. In practice the diode is placed in a cavity or waveguide so that it can be tuned to the desired frequency like in the photo.



Gunn diodes were typically used in 1 GHz+ RF applications but now are used very high frequency (10 GHz+) circuits. Besides oscillators, they are also used as a radar signal source for police speed radar guns, safety detection systems, etc. which are available as surplus if you would like to experiment with one. Some radio amateurs even use them as the microwave source for 10-GHz and 24-GHz radio transmitters (Gunnplexers). Have fun!

180297-01

Weblinks

[1] www.radio-electronics.com/info/data/semicond/gunndiode/gunndiode.php

[2] www.quora.com/What-is-a-Gunn-diode

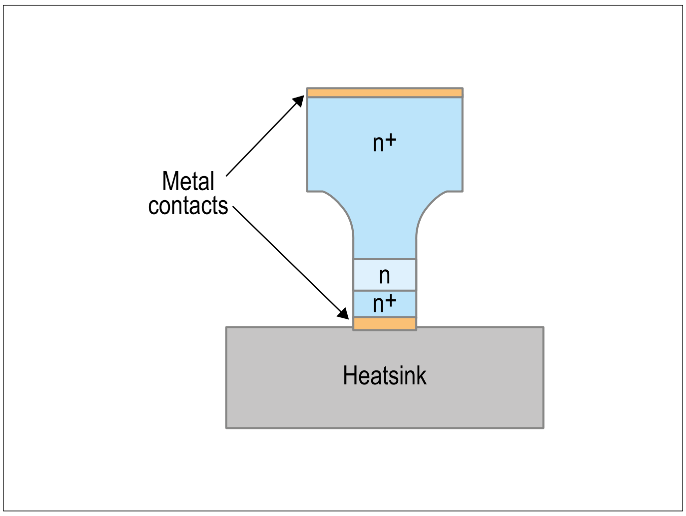


Figure 1. Gunn diode construction [1].

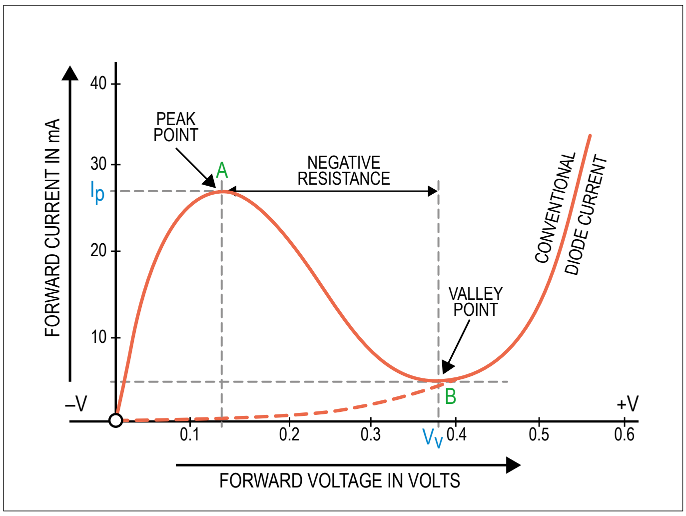


Figure 2. Gunn diode transfer characteristic [2].

HOMELAB PROJECT

Rain Detector

different from the norm

By **Reinier Ott** (The Netherlands)

The climate is changing — everyone talks about this but no one is doing anything about it. There are plenty of small-scale projects, but these are largely ineffective. Unfortunately, worldwide there is too little effort to prevent climate change. And the consequences are noticeable in and around the house.

One of these consequences is a clear increase in heavy rain and thunder storms — downpours — with precipitation rates of more than 25 mm per hour. To get this straight: these few millimetres may not look like much, but we're talking here about 25 litres per square meter, and that starts to add up. Such a shower could cause quite a bit of trouble when you are not at home and you inadvertently left a window open. Reason enough for the author to develop a rain detector with a temperature sensor that can be used to close the windows automatically.

Part of a larger whole

The detector described here is a small part of a larger whole: an intelligent and energy-efficient controller for the air conditioner and the opening/closing of the window in the bedroom of the author. However, for a single article this is too much of a good thing; here we limit ourselves to the rain detector and if there appears to be sufficient interest we will return for the remainder of the system.

The principle

In a 'normal' rain detector, rain drops are counted in a light trap, or the drops fall into a small container that is mounted on a kind of see-saw — if memory serves, *Elex*, a former sister magazine of *Elektor*, published a design like this once. All these detectors have in common that a considerable (fine) mechanical skill is required. Despite the arrival of 3D printers, this mechanical finicky work is often something that electronics enthusiasts have an aversion to. Additionally, the mechanics will invariably cease up when it is not regularly maintained. Because our intention here is to build a detector that only detects a (heavy) downpour, and not build a scientific instrument that measures the amount of precipitation exactly, we make use of a completely different principle, that has no moving parts and requires no (well, not much) mechanical effort.

Characteristics

- reliably detects downpours
- also detects hail
- effective during warmer periods
- without moving parts
- wireless signal transfer
- maintenance free

A capacitive detector could be considered one of the possibilities; with this, the capacitance of the sensor changes when the surface becomes wet. A nice principle, but it has a significant disadvantage: condensation can corrupt the measurement. To prevent that from happening the sensor surface needs to be heated — but that makes the whole thing more complicated and that is not something we want. Also, the enclosure for the rain detector contains a temperature sensor and the proper operation of which would be disturbed by the heater.

That is why another principle was selected: measuring a temperature difference. Here two temperature sensors are placed adjacent to each other; under normal (dry) conditions we can assume that both sensors have the same temperature. When it starts to rain, the rain drops (collected in a funnel) fall on one sensors, while the other sensor remains dry. Because rain drops area at a lower temperature than the ambient temperature (and therefore the sensor), which is generally the case for a downpour during a warm summer day or night, and also because the evaporation of the moisture on the sensor takes heat out of the sensor, the 'wet' sensor will have a lower temperature than the 'dry' sensor. A difference signal generated from this can be used for all kinds of purposes. An additional benefit is that this difference measurement is not affected by condensation on the sensors, since any condensation will be present on both sensors in equal measure.

The signal from the rain detector plus the outside temperature

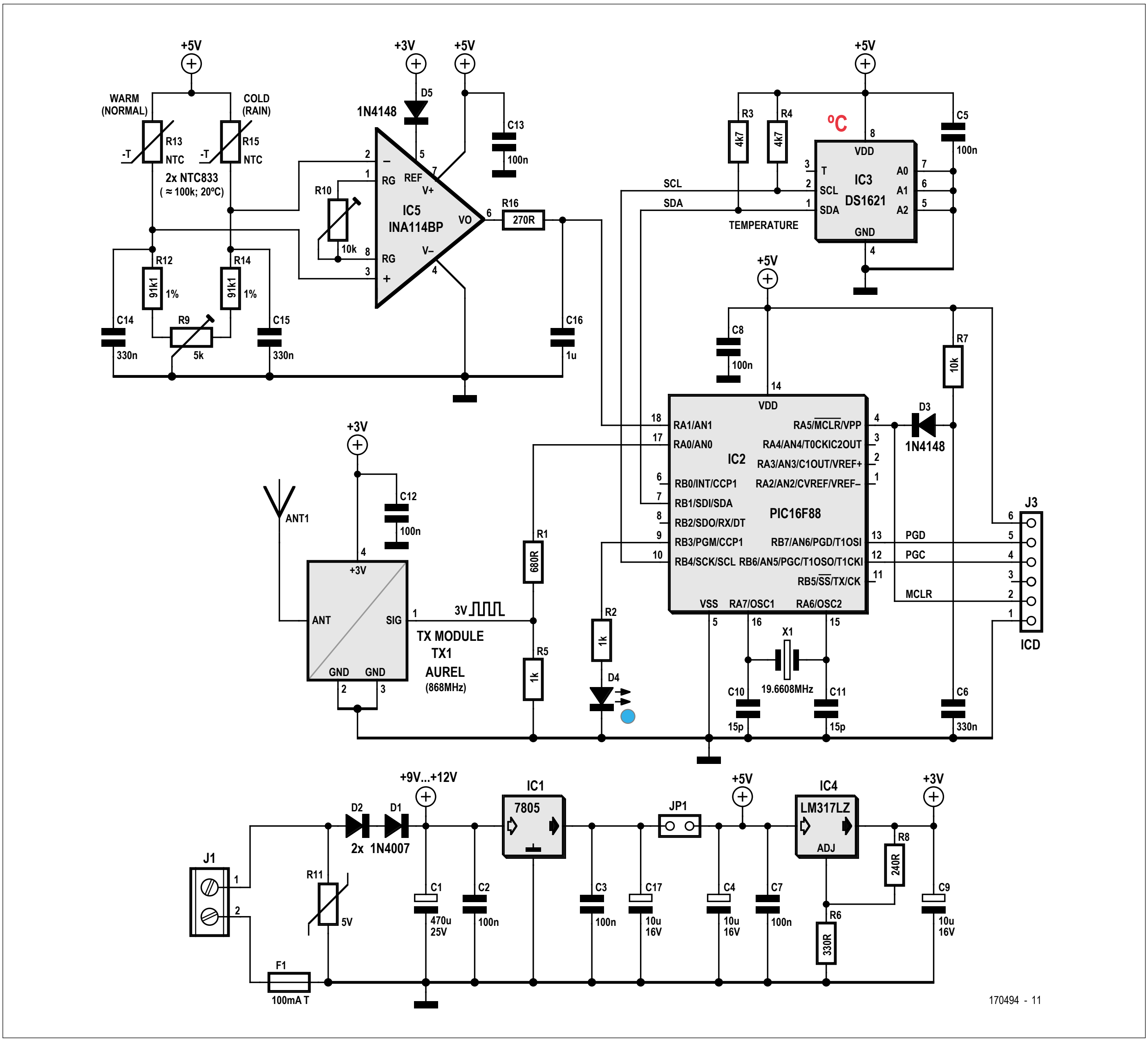


Figure 1. The schematic of the rain detector. We are mainly concerned with IC5 and surrounding components.

measured by the temperature sensor are transmitted wirelessly to a central unit in the house.

The schematic

In **Figure 1** we have drawn the schematic for the electronics. Starting with the power supply. The power supply voltage comes from a mains power adapter (9 VDC). The long cable between the power adapter in the house and the detector in the garden make it essential that the circuit is protected against induced disturbances (thunder storms!) This protection has three parts:

- in series with the input voltage are two diodes (1N4007) with a reverse-voltage rating of 1000 V each;
- a varistor (18 VDC) is built in;
- and finally there is also an ordinary fuse.

In all other aspects the power supply is a regular design, using a 7805 voltage regulator that supplies 5 V for practically all the electronics, and an LM317, which is set such that it supplies a voltage of 3 V for the transmitter module.
The temperature sensor is straightforward — a two-wire I2C sensor type DS1621, which has already been used in various Elektor circuits. Just to be sure: this sensor has nothing to do with the rain detector. This (calibrated) sensor sends the absolute temperature in the form of 2 bytes to the processor. The first byte (MSB) gives the temperature in whole degrees Celsius, the second byte the fractional value after the decimal point. In the datasheet [1] for this IC you can read how the temperature value is encoded in these two bytes and how to control this IC.
The actual rain detector consists of two compact NTC resistors (NTC833, R13 and R15) with a (specified by the manufacturer) response time of only 0.7 s [2]. Together with two fixed resistors (R12, R14), and a potentiometer for calibration (R9), these form a Wheatstone bridge; the exact balance point of which is adjusted with R9. The nodes of the NTCs and fixed resistors are connected to the inputs of an instrumentation amplifier type INA114BP (IC5). The gain of this is given by:

$$G = 1 + \frac{50{,}000}{\mathrm{R10}}$$

With a value of 10 kΩ for potentiometer R10, the gain can be adjusted from 6× to 500×. Because the detection of rain is not a particularly high-frequency process, the bridge is low-pass filtered with capacitors C14 and C15, and the output of the instrumentation amplifier with C16. This suppresses any potential oscillations.
The output signal from the instrumentation amplifier in the author's circuit goes to a microcontroller (a PIC16F88); this processes the signal (together with that from the DS1621 temperature sensor) so that it can be transmitted wirelessly to the indoor unit. An 868-MHz module from Aurel is used for this [3]. In this article we focus on the actual rain detector; for an extensive description of the transmission protocol we refer you to the documentation provided by the author, which is available as a free download (including the software for the microcontroller) [4].

Practical implementation

Figure 2 gives an impression of the housing for the rain detector, which the author fabricated using a 3D printer. The enclosure with the electronics is mounted on a small angle, while the rain is directed onto only one of the NTC833 temperature sensors through a funnel. The entire assembly is positioned under a small roof so that the 'dry' temperature sensor does indeed stay dry during the rain and no rain water remains on the 'wet' sensor.
The two sensors themselves are mounted in a kind of over-sized, hollow 'drawing pin' (see **Figure 3**). These 'drawing pins' have a diameter of 20 mm and are made from 0.5 mm thick silver, for the sake of the least possible heat capacity and the lowest possible thermal resistance. The tubes, in which the actual NTC resistors are mounted with heat-conducting thermal paste, are soldered (with silver) on the silver discs. Subsequently the sensors are attached with epoxy glue on the outside of the enclosure, only the tubes pass through to the inside.
The third temperature sensor (the DS1621) is also not just mounted in its socket; this is glued to an aluminium disc and subsequently attached via a ribbon cable to its socket. The photo in Figure 5 gives a good impression of this.
Figure 4a and **Figure 4b** give an impression of how the author

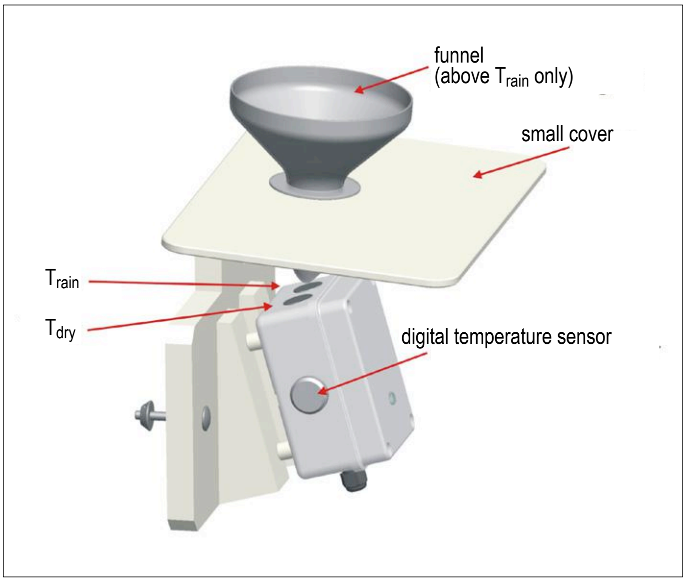


Figure 2. This is how the rain detector can be built in practice.

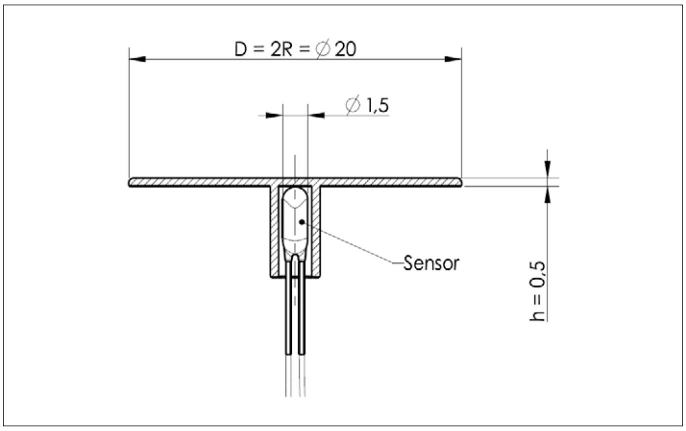


Figure 3. Both of the temperature sensors that are at the heart of the rain detector, are pushed into small tubes that are soldered to small silver discs.

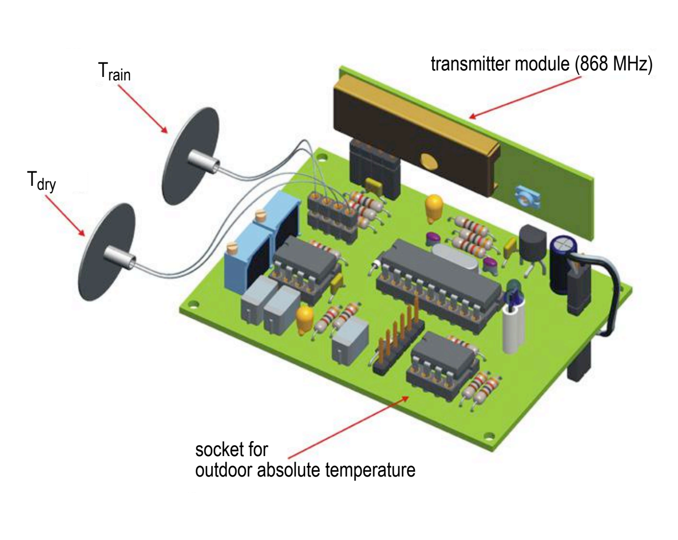


Figure 4a. The components can be mounted on prototyping board like this.

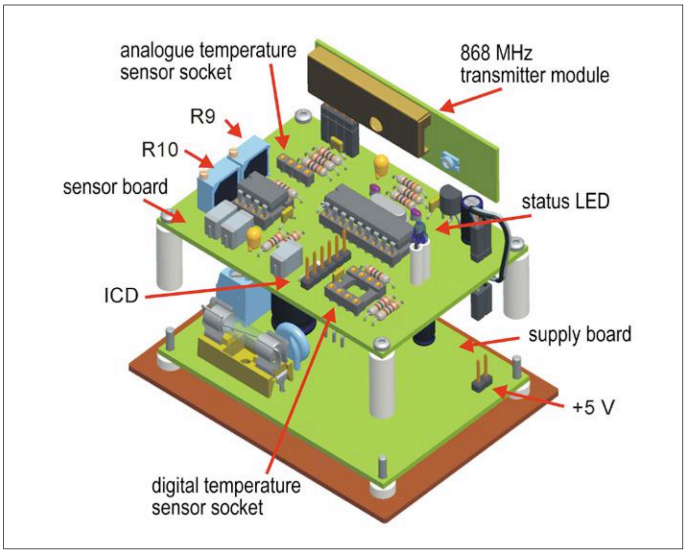


Figure 4b. The circuit board with the parts for the power supply is underneath the sensor board.

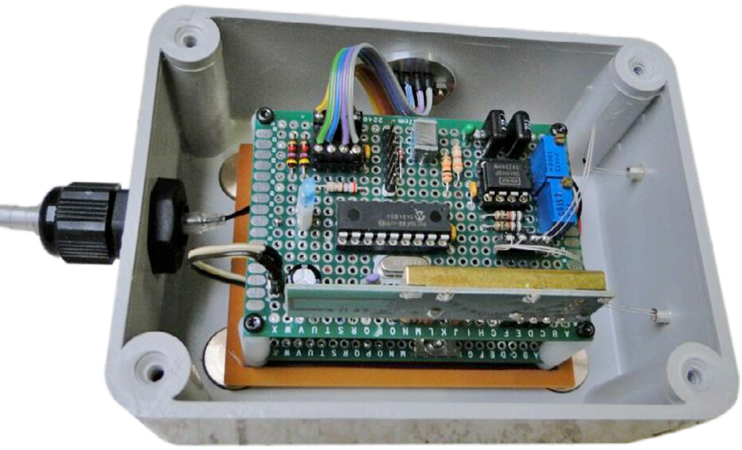


Figure 5. The prototype made by the author.

has built his version; in **Figure 5** you can see that prototyping board was used for this.

Calibration and practical use

For stable and reliable operation it is important that the output of the instrumentation amplifier, across the entire temperature range, is in the range from 2 V to 3 V (at a gain that is set initially to about a 100 times, P10 rotated clockwise to a value of about 500 Ω). The balance of the Wheatstone bridge is set with R9 so that when both NTC resistors are at the same temperature, the output voltage of the instrumentation amplifier is equal to the reference voltage at pin 5 of the amplifier (the reference voltage is equal to the power supply voltage for the transmitter module minus the voltage drop across diode D5, therefore about 2.4 V). At the correct setting the detector will react practically instantly to a minimal temperature difference between the two sensors (which can be checked by touching one of the sensors briefly with a finger).

In practice it may be necessary the reduce the gain to prevent the detector from reacting too soon; the author arrived at a gain of about 37 times with his version. It is also possible to set R9 such that there is a small imbalance in the bridge, which results in a small threshold before the sensor reacts.
The prototype of the sensor has now been in use for more than two years. At appears that especially in summer (and the circuit was designed for this in the first instance) rain drops are decidedly colder than the ambient temperature. That means that the sensor reacts reliably. A prerequisite is that the collection funnel is a good thermal insulator so that it doesn't heat the collected rain drops (too much). A version 3D-printed using white ABS works very well. When the rain continues for a longer period the temperature difference will decrease but that is of minor importance because for the closing of the bedroom window (using a servo motor) only the initial detection is important. ◀

(170494-01)

Web Links

[1] https://pdfserv.maximintegrated.com/en/ds/DS1621.pdf

[2] www.conrad.nl/p/temperatuursensor-b-b-thermo-technik-praezisions-ntc-40-tot-100-c-radiaal-bedraad-188506

[3] www.conrad.nl/p/zendmodule-aurel-tx-8l25ia-191564

[4] www.elektormagazine.com/170494-01

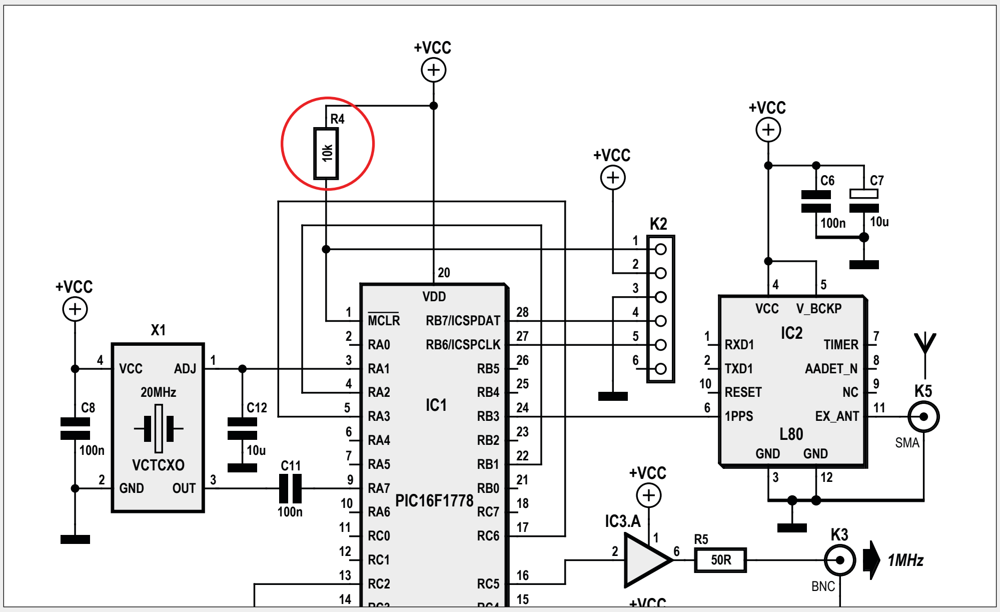
Err-lectronics

Corrections, Updates and Feedback to published articles

10-MHz-Reference

Elektor 3/2018 (May & June), p. 32 (160594)

UPDATE. There is an error in the circuit diagram, Figure 3. The value of the resistor R4 must be 10 kΩ, not 820 Ω.



Struggling with LED Snake Lights

Elektor 3/2018 (May & June), p. 84 (160669)

FEEDBACK. Once again I read - not only in Elektor Magazine - the term „low voltage“ in the wrong context. Thomas Scherer writes about LED lamps at 230 V and then about strips that would (mostly) be operated with low voltage. In fact, the 230 volts belong to the low voltage range; the range designated by the author as low voltage, probably the later 12 V, is low voltage. Please do not be influenced by the term „Low Voltage“ from the English-speaking world!

Franz Babener (Germany)

Editorial reply:

You are absolutely right; it must be called Extra Low Voltage (ELV). In the case of Safety Extra Low Voltage (SELV), this must not exceed 25 V for AC and 60 V for DC. Voltages up to this level are still considered harmless when touched, which is why it is not necessary to take any protective measures against contact.

Ralf Schmiedel

Tips & Tricks: Door alarm detects water

Elektor 5/2018 (July & August), p. 72 (160470)

FEEDBACK. To prevent the electronics from getting wet even at higher water levels, the circuit can be inserted with the sensor feet through a piece of polystyrene so that the electronics floats and remains dry itself.

Kees de Groot

Special Quality (SQ) Tubes

The pinnacle of tube technology

By **Jan Buiting** (Retronics Editor)

Partly because of their pleasant appearance and high stroke factor, LF amplifier tubes (valves) in particular are very popular with audiophiles who speak about their favourite tube as if were is a bottle of ChateauNeuf-du-Pape bought at Sotheby's. Top of the heap are the rarer versions with gold-plated pins, the so-called SQ types, which are fanatically searched for, and often "fetch" more in cash than the heavy equipment from which they were pulled. Lesser known are the origins of these SQ tubes and for what purpose they were originally developed, which duly appears in this article, in summary.

We go back to 1956 when Klaas Rodenhuis, today recognised as Grandmaster of Practical Tube Technology at Philips Netherlands, together with two colleagues, first published an article in the legendary Philips Technical Review [1]. The article provides such a good basis for the general knowledge about tubes that it appears here in summarised and partly retranslated form.

1956... the well-known Philips Miniwatt 'E' tubes are already manufactured in hundreds of thousands, not only for use in our (grand)parents' radio and TV sets of but also gradually in industry. While production for the consumer market was cheap & easy, questions from professional users prompted Rodenhuis and his team to take a close look at the reliability of "amplifier tubes" as he called them at the time. Back then, the following was considered professional: 'telephone amplifiers, electronic calculators, industrial monitoring and control systems, and mobile transmitters and receivers'. Special Quality tubes were gradually developed for this purpose. Indeed, "... The serious consequences that failure of a tube in professional equipment can have imply that the requirements usually imposed on tubes, e.g. requirements concerning steepness, power output, suitability for wide band amplification, *etcetera*, are not sufficient. It is evident that reliability is the extra requirement for professional tubes, since the equipment must be ready for operation at all times and must not fail suddenly. Furthermore, characteristics such as long service life, close tolerance of electrical characteristics and high impact resistance may be important, although not to the same extent in all four areas of application." So here's a case of damage claims, not from Joe Bloggs at the Radiospares counter, but big cases from the industry!

The concepts of reliability and service life

Assume a huge piece of electronic equipment equipped with a large number of S_0, (say 1000) new tubes of the same type that are operated at a given time for a long period. Each failure of a tube is recorded, so that at any given moment *t* it be seen how many of the original tubes are still in operation (S). If S / S_0 is plotted on a logarithmic scale as a function of t, a curve is created as shown in **Figure 1**, a few of which are derived from earlier research. Strange is the curve number 1 found by Lewis in 1948; it is an almost straight line, the equation of which is:

$$S / S_0 = e^{-Pt}$$

in which P is a constant. For the number of tubes that fail per unit time, i.e. $-dS/dt$, we find: $P \cdot S_0 \cdot e^{-Pt} = P \cdot S$. As a result, the relative failure per unit time (the failure rate, analogous to the mortality rate of a population), $-(dS/dt) / S$, is equal to the constant P and therefore independent of time!

Without considering the integral-type derivation of the average service life L_m here in detail, curve 1 yields: $P = 0{,}135 \cdot 10^{-3} \cdot h^{-1}$ (i.e. failure rate 13.5% per 1000 hours), and L_m then becomes: 1000 / 0.135 = 7500 hours (and your iPhone...?)

The number of survivors at instant $t = L_m$ equals

$$S_{Lm} = S_0 \cdot e^{-1} = 0.368 \cdot S_0$$

meaning that 36.8% of the original number of tubes is still in operation. When $t = 2 \cdot L_m$ the number is $S \cdot 2 \cdot L_m$, $= S_0 \cdot e^{-2}$, *etc.*: in each period L_m decreases the percentage of survivors by a factor *e*. Consequently, tubes that have operated over thousands of hours are in no way different from completely new tubes and that it is pointless to replace them with new ones. Of course, this only applies to a straight line, such as 1.

More realistic, however, would be the curve no. 3 published by Knight, also from 1948. It shows a steep decline during the first few hundred hours, and then a long, fairly straight section with a small slope, which after many thousands of hours changes into a steeply descending curve. And that looks more practicable. Initially, some tubes show the consequences of manufacturing faults that escaped attention during inspection (e.g. poor welding, near short-circuits between electrodes, *etc.*). After the failure of these tubes, a 'coincidental' failure occurs only occasionally, until finally certain types of 'non-accidental' failure are considered, namely those related to slow-moving physical and chemical processes in and near the cathode, such as a gradual decrease in emissions, a gradual decrease in insulation resistance between electrodes, *etc*.

The following can be said to hold true about the curve:

- In the straight middle part of the curve, the tubes behave like tubes to which the exponential law $S / S_0 = e^{-Pt}$ applies.
- Previously, during the first 100 to 1000 hours, the dropout rate was higher. Tubes that have worked for several hundred hours are therefore more reliable than new ones.
- It is only beyond the straight line, when the failure rate increases, that it becomes advisable (unless certain precautions have been taken) to replace the survivors *en bloc* with new tubes.

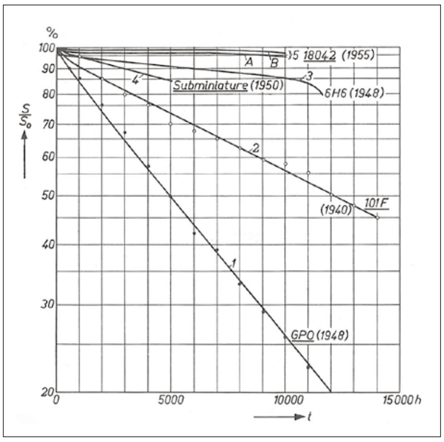
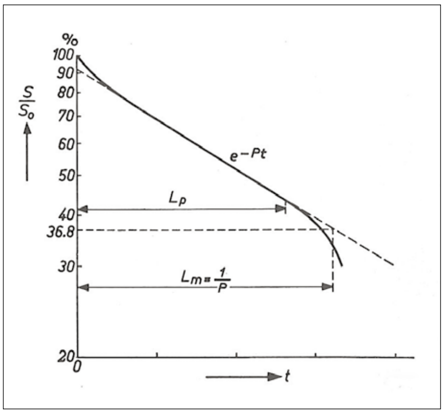


Figure 1. Percentage S / S_0, on logarithmic scale, as a function of time *t*, in life cycle tests with different types of tubes. Curves 1 though 5 are taken from earlier literature [1]. The curves show the type of tube and the year of publication. The data for curves 5A and B were obtained from a trial in which the Dutch P.T.T. kindly cooperated. [1].



Figuur 2. Illustrating the terms "average life span" L_m and "practical life span" L_p (L_p ends when the failure rate starts to increase). Source: [1].

The practical life span L_p is defined as the time at which the failure rate starts to increase (**Figure 2**) and is therefore more useful than the aforementioned L_m. This time is of course not only dependent on the quality of the pipes, but also on the deviations that may occur in cer-

ESTD 2004

www.elektor.tv

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

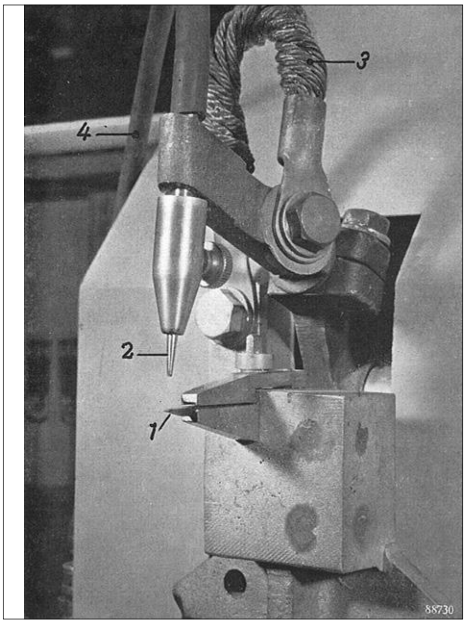


Figure 3. Spot-welding machine at Philips. Item 1 is the fixed electrode; 2 is the movable electrode with power supply 3. A reducing gas mixture ($H_2 + N_2$) is fed through hose 4. Source: [1].

tain tube properties before the circuit in which the tubes are used no longer functions properly. Two tube properties that are important in this respect are the steepness and the control grid current. To have an objective criterion, we usually find a tube to be less or unusable when the steepness has fallen to 70% of the nominal value, or when the control grid current (under defined conditions) has risen to 1 µA. If necessary, other criteria may be chosen for particular applications. The failure rate mainly determines the frequency of malfunctions that will occur as a result of a tube failure. The reciprocal value is simply the reliability — the flatter the curve, the greater the reliability. As an example of the behaviour of a tube with high reliability, Figure 1 shows the curves 5A and 5B, both of which apply to type 18042 (an SQ penthode for telephone amplifiers). The two curves relate to different applications. The curve 5A was found from a test in which the tubes operated in a circuit that is very sensitive to insulation errors in the tube. Therefore, if a certain insulation resistance had dropped to 1 MΩ in each tube, a relay indicated that it had "failed". The curve 5B applies to equally heavily loaded tubes 18042, now, however, in an ordinary low-frequency amplifier, in which a decrease of the insulation resistance to 1 MΩ could not cause any harm and was therefore not detected. In the straight section 5A shows a slope of 0.5%, and 5B only 0.25% per 1000 hours. After 10,000 hours there was no tendency to turn around.

At Philips it was believed that the failure rate of professional tubes could (should) be reduced to around 0.1% per 1000 hours, and that "... there will not be many applications for which this implies insufficient reliability". But for the booming industry, reliability was the Number One issue.

The gradually occurring faults

Certainly in 1956 most tubes had an indirectly heated oxide method. It consists of a layer of BaO and SrO (and sometimes CaO) mixing crystals applied to a nickel tube, which is heated from the inside by a tungsten wire to a temperature of 750 to 800 °C. The saturation current density of such a cathode is between 2.5 and 25 A/cm^2. What can be done there to improve quality and thus prevent failures?

- Reduce all symptoms of cathode poisoning by positioning the electrodes and any screens in such a way that no electrons get onto the glass or mica. Aim for the highest possible purity of the components used to build the tube.
- Select the lowest cathodic temperature that is still permissible to avoid poisoning (a certain margin must of course be left here due to variations in the filament voltage).
- Prevent the formation of an undesirable interlayer in the cathode by using "passive" nickel for the cathode tube, i.e. nickel with approx. 0.03% Mg and not more than 0.01% Si (ordinary nickel contains up to approx. 0.1% Si). Passive nickel takes longer to activate the cathode in manufacturing, but after 19,000 hours of normal cathodic temperature operation, no resistance with measurable value has been formed.

And now for a few more factors with which the equipment designers can get started right away.

- Keep the filament voltage within ±5% of the rated value. Stabilised within 1%, even 5% to 10% below the nominal value is "fully justified", according to Rodenhuis and his crew. So 6.00 V and nicely stabilised ... fine. A company like Tektronix did it like this for years, with known results. And... use direct voltage!
- Keep the balloon temperature below 170 °C at all times in order to limit the release of gas through the glass. Caution should therefore be exercised when using shielded cover placed over the tubes.
- Avoid excessive anode and/or screen grid temperatures and cathode temperatures due to excessive dissipation, which may result in gas release or barium vaporisation. With regard to cathode poisoning, it is not indifferent whether a certain dissipation is obtained at high voltage and low current, or at low voltage and high current. High voltages are more likely to cause poisoning, probably because fast electrons in particular are able to break down material that has been absorbed by the anode or screen grids into substances that affect the cathode. A favourable condition for the tube is therefore: small dissipation at low voltage; the cathode current can be large without any problem.

The best possible prevention of insulation errors, grid emissions and ion currents are the shared responsibilities of the tube manufacturer and the appara-

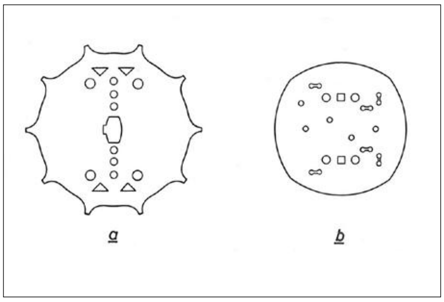


Figure 4. Old form (a; left) and new form (b; right) of the mica support plates. With the new shape there is less flaking. Source: [1].

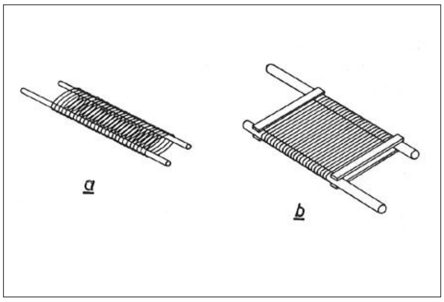


Figure 5. A grid of normal construction (a; left) and a modern “window grid” (b; right). Source: [1].

tus designer. The best way for the manufacturer to exercise control is to store the tubes for one month before delivery and then measure the grid current. If desired, the leakage check can be carried out quicker with the so-called argon test. The designer can keep the influence of inadequate insulation and grid current small by using the smallest possible grid resistance.

Fun lessons and trivia:

- ExxC(C) / ExxL indicates an SQ tube.
- An ECCxxx can be an SQ tube.
- Not all SQs follow the type numbering ExxC(C) / ExxL.
- Not all SQ tubes have gold plated pins.
- A 4-digit xxxx number or a CVxxxx can also be an SQ tube.
- Today, the ExxL numbering is incorrectly used by smart manufacturers from certain countries to designate a "quality tube".

After 1960, the development and production of the SQ series continued for several years at Philips, Valvo, Mullard and Telefunken, among others, and an ever higher degree of perfection was sought and achieved. If you have SQ tubes that are not listed in the table, inform the Retronics editor: jan.buiting@elektor.com. I came across two of them: the **E34L** and the **E288CC**, the last of the terminal production phases of the famous Philips Heerlen-Molenberg factory. Audiophiles wave their Gold credit cards when they see two of those tubes, NOS but without a box, as I recently found out at the Rosmalen radio flea market.

Sudden failure... hey, no warning!

Sudden failure of a tube is usually caused by: electrical interruption (e.g. a detached weld), short circuit between two electrodes, glass errors, and breakthrough between cathode and filament. We will take a brief look at these factors.
In order to avoid **interruptions**, it is important to place good welds and to take samples in order to maintain the assembly quality. The spot welding equipment and the materials used must also meet high standards. For example, the phase of the switch-on moment of the spot-welding machine (**Figure 3**) is fixed, i.e. it is independent of the moment when the assembler presses the foot pedal.
Short-circuits between two electrodes (only 100 µm and even 50 µm apart) are easy to detect in production and rarely occur 'outside the factory gate'. A much more serious problem are conductive compounds, which are caused by loose particles in the tube. These can exhibit transition resistances that not only vary widely from the order of 10^7 to 1 ohm, but also intermittent occurrences, which makes it very difficult to detect them. Dust-free assembly is therefore recommended. Likewise, it is absolutely necessary to avoid loose particles in the tube that arise during manufacture, such as droplets of metal that have splashed away during welding, chipped cathode material and flaked off pieces of mica (these become conductive in the long run, if atomized cathode material deposits on them). Sprayed metal indicates incorrect adjustment of the welding machine; crumbled cathode material indicates careless fabrication of the cathode. Mica washers of a new shape (**Figure 4, right**) blister much less than mica in the old shape (**Figure 4, left**).
Cathode-filament breakthrough is unfortunately a notorious "fault" in many tubes, often occurring after decades of heavy use. At Philips, the optimal composition of the cathode and filament materials was carefully examined in Rodenhuis' lab and an unconventional route was followed for the "professional" tubes with smaller cathodes: not only the filament, but also the inside of the cathode tube is coated with alumina (aluminium oxide). This is more favourable for several reasons than applying a thicker layer of alumina to the filament:
(a) The internal layer on the cathode-ray tube is at a lower temperature than the layer applied to the filament, which increases the breakthrough voltage.
(b) The increased radiation absorption of aluminium oxide relative to nickel improves heat transfer, allowing lower filament temperature.

In practice, in the case of cathodes with an internal layer of alumina (e.g. with the SQ tubes types E80CC and E80L), it appears that it takes 5 to 10 times as long for a breakthrough to occur or that, for the same service life, V_{kf} can be made appreciably higher.

Spread in the characteristics

Differences in the characteristics of tubes of the same type are due to differences in electrode dimensions or distances, in output potential of the control grid and in emissivity of the cathode. In particular, the g1 exit potential may be less known to many tube users. It depends, among others, on the extent to which the control grid is covered with deposited cathode material, i.e. the cathodic temperature. The grid is more sensitive to poisoning than the cathode, as it cannot produce free barium itself and is therefore only "fed" by the cathode. The purpose of allowing the tube to work for a few days at the factory is to bring the surface of the control grid into normal working condition; once this has been achieved, the output potential does not change much anymore.
Until about 1956, almost all of Philips' tube grids were constructed in accordance with **Figure 5**, on the left: the grid consists of a screw-wound molybdenum wire, supported by two bars made of nickel, copper sheathed wire or similar material. The windings are held secure due to to resting in grooves in the bars, which are then clamped tight. With the improved construction, the grid windings no longer have any mechanical function. This is entirely entrusted to a solid window (**Figure 5, right**) consisting of two rods, which are held at the right distance from each other by four strips. The wire is very thin, e.g. 7.5 µm, and is wrapped tightly around the window. In combination with measures to reduce the tolerance in the dimensions of the cathode and the mica plates, the window-grid construction makes it possible to manufacture tubes with a cathode-grid spacing of 50 µm, i.e. half the size that was feasible at the time with normal grids. Of course, the professional tubes (ExxCC / ExxL / SQ series) were perfect for this new grid technology with much less spread in the characteristics.

Vibration resistance

When the use of tubes in aircraft in particular increased enormously, the issue of

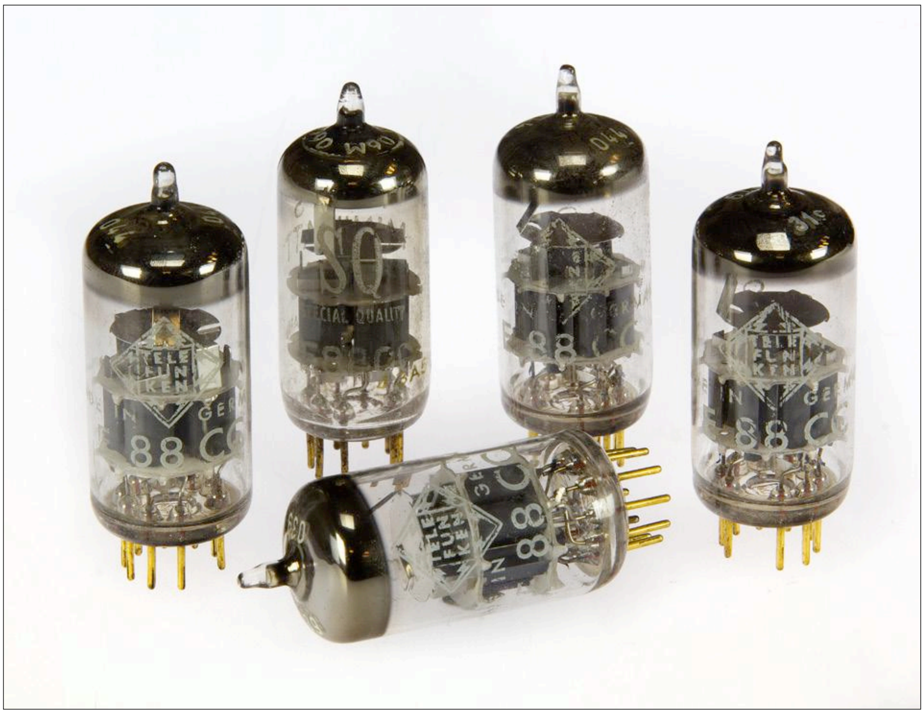


Figure 6. A quintet of E88CC tubes from Telefunken and Philips. Any offers, please!

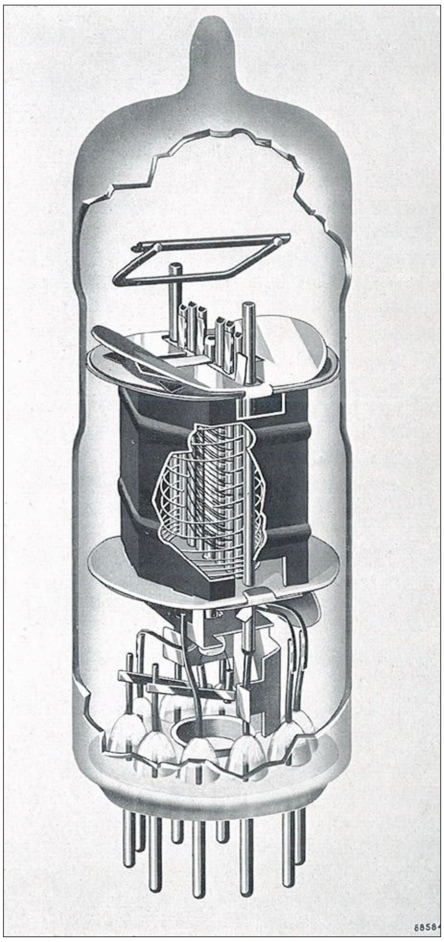


Figure 7. Penthode type E80F, with improved mica washers according to Figure 4b. The upper washer fits snugly into a slightly pinched area of the glass balloon. Source: [1].

vibration resistance demanded increasing attention.
If there is any play between the mica washer and the balloon, or between the washer and any part it holds, the mica will wear out when the tube is exposed to vibration, increasing its play. First of all, then, the feared 'microphony' effect is on the increase. Later on, the play may increase to the extent of creating a short circuit between two or more electrodes. Mica can also emit gas and microscopic flakes can poison the cathode. The mica shape pictured in Figure 4b proved to be a great improvement, and a particularly good fit of the washer in the balloon is obtained by pinching the balloon locally where the diameter is 0.1 to 0.2 mm smaller than the largest diameter of the mica washer. **Figure 7** shows the inside of the SQ tube type E80F in which this has been applied. In order to prevent mechanical fatigue of the filament, it is necessary that it has little play with respect to the cathode tube. The so-called cathode tail can be saved from breaking off by giving it a generous bend.

Quality Control

In the manufacture of "Special Quality" tubes, quality control is of the utmost importance. Some examples have already been mentioned, such as checking the installation and the tensions in the glass envelope. However, this has already been preceded by a quality check of the components, in which the most important properties of these components have been determined on a random basis. After pumping and burning clean, the tubes are examined for short circuits, interruptions, crackling, etc., and measured for certain electrical properties.
The tubes are then exposed to vibration for 5 minutes to determine whether there are any loose particles in the balloon. The behaviour during 48 hours of operation, in which the characteristic properties are stabilised, provides an important indication of the reliability. After this period, which is a short lifetime test for all the tubes produced, the emission, steepness, anode current and insulation are checked, and a new crackling test is carried out. Each week's production is stored separately. From this, a number of tubes intended for sampling are taken. Some of these are measured in terms of capacitance, grid emission, noise, micro-

phony, etc., while others are subjected to a 500-hour lifetime test. Tubes with high mechanical stress requirements shall be subjected to random vibration and shock tests.
The results of all these samples determine whether the relevant weekly production may be delivered. Before delivery from the factory, the main characteristics of the tubes are examined again to see whether they have been stored in a way that has had a harmful effect.

Today

Here we return to the year 2k018 and find that tubes that were initially specially made for use in industrial, telecom and computer equipment, have inadvertently acquired a cult status, especially by "them golden pins" (they are: gold-plated pins).
The development of the SQ Tube line at Philips yielded two benefits: on the one hand, it was possible to sell to industry at high prices, and on the other hand, the improvements also partly benefited the "ECC/PCC consumer glass" sold in millions. The now famous SQ tubes are often among the last to be manufactured by and for Philips at the end of the 70s; examples of this are the beautiful E80L for use in telephone exchanges, and the E90CC in certain "computing gear".
While Philips produced its SQ and "ruggedized" tubes specially and sold them at premium prices, some large customers taking Philips "regular" stuff simply ordered large batches from which they picked the best ones for "hi-rel" and truly critical applications. A good example of this is Tektronix, which by itself conducted a punitive quality regime on tube microphony and steepness in particular, and gave matched tubes their own stock number (**Figure 8**). ⏮

180299-05

Golden Oldies

The table shows an overview of “tubes for professional applications” I found in the 1960 edition of the *Philips Tube Pocket Booklet*. The Eindhoven sales division went to great lengths to use every term and classification imaginable to trigger potential customers. Only “military” is not seen here, there was a separate telephone line for it, or “KLU” (Dutch Royal Airforce) was stamped on the tube.

Qualification (according to Philips)	Tube types
Reliable long life	18040; 18042; 18045; E81L; E83F
Reliable, ruggedized and long life	E80CC; E80CF; E80F; E80L; E88CC; E130L; E180F, E188CC
Reliable, ruggedized	E90F; E99F; 5654; 5726; 6201
Computers	E90CC; E91H; E92CC; E180CC; E182CC; ECC186; 6211; 6463

Figure 8. XXL consumers like Tektronix dodged the SQ-tube price tags by selecting the best ones from regular-run production batches, which were mainly tested for steepness and microphony in their own factories. Here’s a pair of matched 12BY7A tubes I borrowed from an old oscilloscope.

Reference

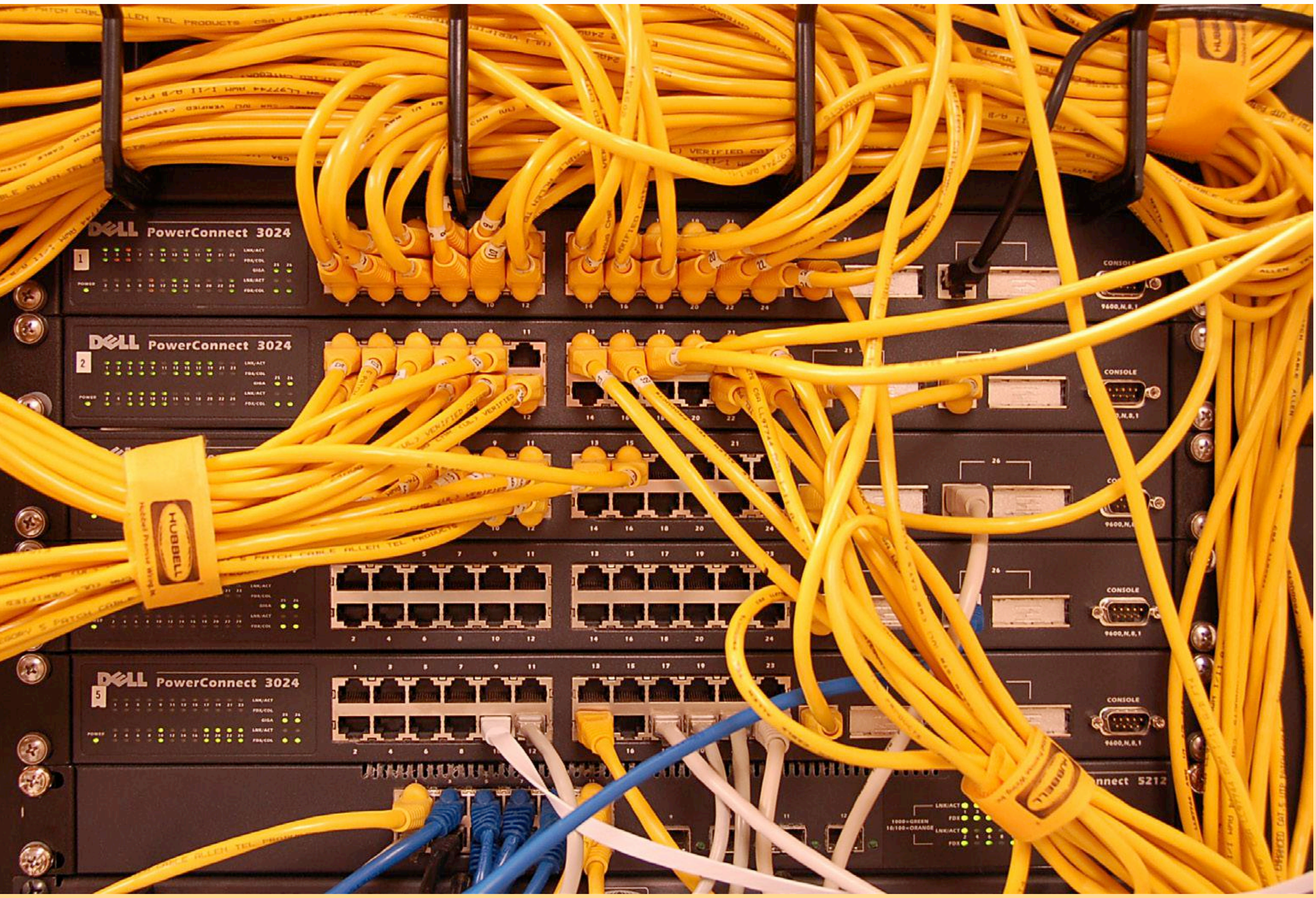
[1] Reliability and longevity of professional amplifier tubes, K. Rodenhuis, H. Santing and H. J. M. van Tol, *Philips Technical Review*, edition May-June 1956.

ELEKTOR ETHICS

Trust in the Smart City

How Amsterdam is working on a trustworthy algorithm

Tessel Renzenbrink



Data storm. Photo: Dave Herholz. https://www.flickr.com/photos/dherholz/450303689/ CC licence: BY-SA 2.0

Student admissions to secondary schools are determined by an algorithm in Amsterdam, according to Tamas Erkelens. He is Programme Manager Data Innovation for the Municipality of Amsterdam. There is not enough capacity to allow every student to attend the school of their choice. Up until 2014, admissions were determined by lottery for each school individually. Students who lost out in the lottery had to make do with an unpopular school that still had room. In 2015, a central lottery and matching system for all students in Amsterdam was established. The students state their twelve preferred schools, and then an algorithm assigns specific schools to the students.

One of the main differences is that only one preferred school could be stated in the old system. In the new system, a student who misses out on their first choice still has a chance of being admitted to one of the eleven other preferred schools. In 2017, 97.7 per cent of the applicants were admitted to one of their top five schools.

Amsterdam is striving to become a smart city. As in many other cities, data and algorithms are being used to improve life in the city. But how can you achieve trust in digital technologies amongst citizens and companies? And convince them that the smart city deserves their trust? A variety of speakers addressed these questions at the We Make The City festival on 20–24 June, 2018, in Amsterdam [1]. Public official Tamas Erkelens talked about a practical example of an algorithm that is being used in Amsterdam. Professor Valerie Frissen described a number of reasons why people find it difficult to trust technology. And Professor Sander Klous talked about what Amsterdam can do to increase that trust.

Every year there is a lot of discussion about the lottery because there are always some students who lose out and do not end up in the desired school. That was also true with the old system, but then it was simply bad luck. With the new system, they can always blame the algorithm. Students and their parents may think that the algorithm is not fair or not to be trusted.

Trust in technology declining

However, the city needs this trust. Without public support, it is difficult to carry out processes based on data and algorithms – but trust in digital technologies has been declining in recent years. Valerie Frissen, Professor of ICT and Social Change at Erasmus University, described a number of reasons for this. “Trust is based on give and take. But in the data society, a lot of transparency is expected and demanded from us without corresponding transparency on the other side.” And data is becoming increasingly important for many aspects of our lives. Data determines which advertisements and news items are shown to us. Data can be a factor in taking out a loan or applying for a job. But this data does not belong to us. We do not know if the data that is collected about us truly reflects who we are.

Another issue, according to Frissen, is that technology is constantly becoming less visible. As a result of the emphasis on intuitive interfaces for electronic consumer devices, we have less and less understanding of how the underlying technology works. Tablet computers are so intuitive that young children can use them, but even most adults do not go beyond the functions preprogrammed for them by the designers. A modern smartphone has more processing power than the systems that were used for the first moon landing, but the average user can’t do much more with it than swiping over the screen. Frissen: “Ease of use has become so dominant that we no longer know what’s happening inside and what social relationships are hidden behind the interface.” Finally, systems are becoming more and more complex. Frissen: “That makes trust blind. And if trust is blind, it shouldn’t be violated or repeatedly put under pressure.”*swipen*

Working on trust

Sander Klous, Professor of Big Data Ecosystems for Business and Society at the University of Amsterdam, talked about what a city like Amsterdam can do to win the trust of its citizens. He used the lottery and matching algorithm as a practical example for that. The problem with an algorithm is that it is a black box. The algorithm processes a data set and generates an outcome, but what happens between input and output is unclear. The first question is: Who determines what the algorithm should do? In the case of the matching algorithm, that is the municipal council.

The next question is: How is the algorithm checked to ensure that it works the way it should? That can be done by carrying out an audit, says Klous. In addition to examining the algorithm, the audit looks at the operating context of the algorithm. For example, it checks if the data that is used by the algorithm is correct. It also checks if the underlying processes run as they should. For example, if students are properly informed that they have to state twelve preferred schools. The third question is: How is the process made understandable to the citizens? Klous does not have a clear answer to that question, because understanding at the level of the algorithm itself is not feasible. Hardly anyone can follow the complex mathematical logic of that algorithm. Understanding therefore has to be created at another level. But when do people feel that they have been given enough information to be able to trust the algorithm? Is it if they know that 97.7 per cent of the students are assigned a school from their top five choices? Or if they can read about the goals set by the municipal council and know that the implementation has been checked by an external party? Or should consideration be given – as suggested by a member of the audience – to making the algorithm easier to understand, at the cost of complexity? Even if that means that the algorithm will not work as well?

The smart city is still in its infancy. At this stage it is difficult to say what is necessary to achieve broad public trust. However, there is growing awareness that trust is a prerequisite for the success of the smart city. And that's a good start. ⏮

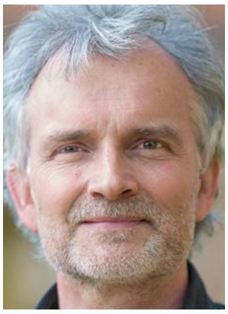
(180300-02)

Weblinks

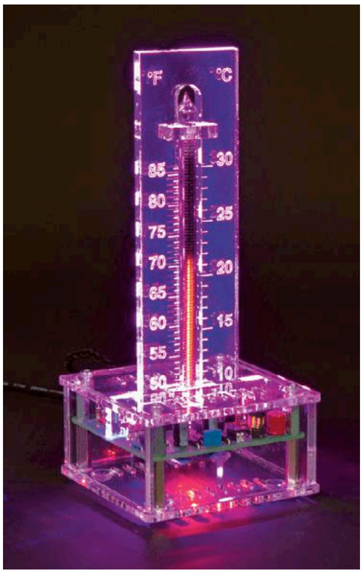
[1] https://wemakethe.city/en/

welcome in your ONLINE STORE

EDITOR'S CHOICE



The Nixie Bargraph Thermometer is a thermometer with a traditional degrees C/F column but a "warm" readout. A great gem for in the living room and a perfect match of the old and the new.



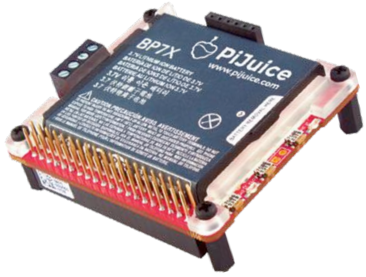
Nixie tubes are always fascinating. Nowadays they are mostly used for clock displays. The 'Nixie' bargraph tubes for analogue readout in the form of a column of light are less well known. Strictly speaking, they are not Nixie tubes because they do not display numerals, but they have the same warm retro allure because they are also filled with neon gas. This thermometer uses a Russian IN-9 tube and is a nice alternative to the usual clock projects.

Jan Buiting (Editor-in-Chief)

www.elektor.com/nixie-thermometer

Elektor Bestsellers

1. PiJuice HAT
   www.elektor.com/pi-juice-hat



2. Raspberry Pi 3 (model B+)
   www.elektor.com/rpi3b-plus
3. The Official ESP32 Book
   www.elektor.com/esp32-book
4. Nixie Bargraph Thermometer
   www.elektor.com/nixie-thermometer
5. Programming with STM32 Nucleo Boards
   www.elektor.com/stm32-nucleo-book
6. Mastech MS8911 SMD Tester
   www.elektor.com/ms8911
7. Raspberry Pi 3 Basic to Advanced Projects
   www.elektor.com/rpi-projects-book

Programming with STM32 Nucleo Boards

The book covers many projects using most features of the STM32 Nucleo development boards where the full software listings for Mbed and System Workbench are given for every project. The projects range from simple flashing LEDs to more complex projects using modules and devices such as GPIO, ADC, DAC, I²C, LCD, analog inputs and others. Comes with a FREE STM32 Nucleo L476RG Board for a limited time only!

member price: £30.95 • €34.95 • US $41

www.elektor.com/stm32-nucleo-book

Acoustics in Performance

This book helps those responsible for providing good acoustics in performance and worship spaces to understand the variables and choices entailed in proper acoustic design for performance and worship. Practicing acoustical consultants will find the book a useful reference as well. The level of presentation is comfortable and straightforward without being simplistic.

member price: £19.95 • €22.46 • US $27

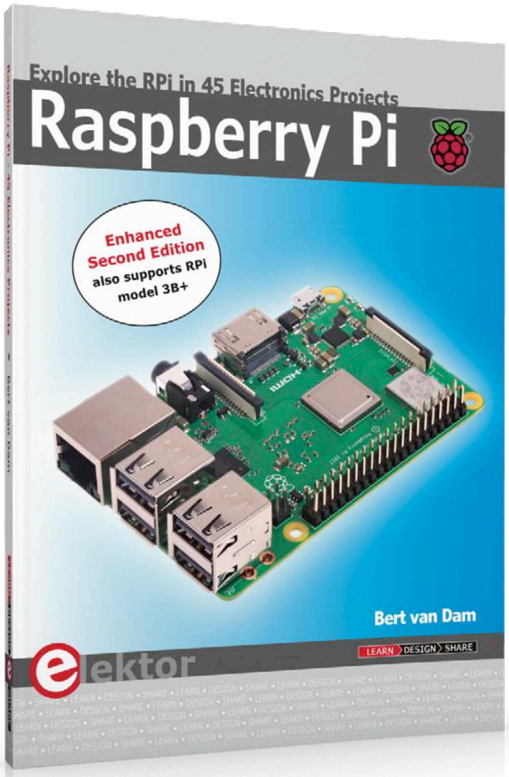
www.elektor.com/acoustics-in-performance

PeakTech 8200 Measuring Accessories Set

This comprehensive collection of test accessories for the electronics area ensure the best suitable measuring adapter for the desired application for each digital multimeter. All clips and cables are manufactured to the latest safety standards and provide a safe implementation in every application for the user. The material used is a flexible and non-slip plastic that can endure the harshest kind of usage.

member price: £35.95 • €40.46 • US $48

www.elektor.com/peaktech-8200-set



Explore the RPi in 45 Electronics Projects

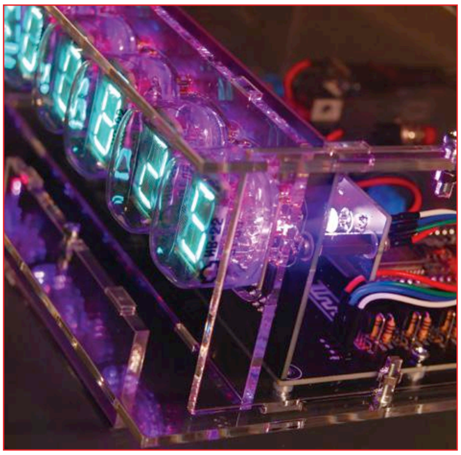
Revised and Expanded Edition

In this book, we concentrate on one of the more powerful aspects of the Raspberry Pi: the combination of programming and electronics. There are 45 exciting and fun projects in this book. It varies from flashing lights to a motor speed controller; from processing and creating analog signals to a CGI web server and client-server programs. You can use this book as a project book, building the projects and using them in practical applications. The clear explanations, circuit diagrams, and photos of the setup on a breadboard make building the projects an enjoyable experience. The book is also very handy as a reference guide. You can easily find projects via the index, which can then be used as a starting point for your own projects. Even if you've built all the projects, it still deserves a place next to your RPi for this reason.

MEMBER PRICE: £27.95 • €31.46 • US $37

www.elektor.com/explore-the-rpi

VFD Tube Clock with ESP32

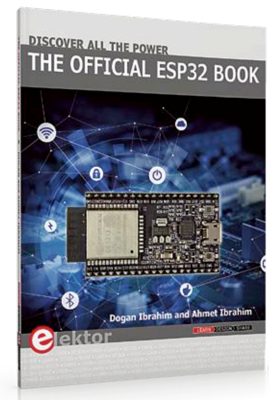


This VFD-Tube clock uses an NTP server on the Internet to keep the time synchronized. It is built around an ESP32-DevKitC module. This Arduino compatible board not only has a powerful processor but also has Wi-Fi and Bluetooth onboard. The Wi-Fi connection is used to synchronize the time. When the clock is switched on, it connects to the network and synchronizes with the NTP server instantaneously. This is better than synchronization via radio time servers or even via GPS.

member price: £135.95 • €152.96 • US $179

www.elektor.com/vfd-tube-clock

The Official ESP32 Book

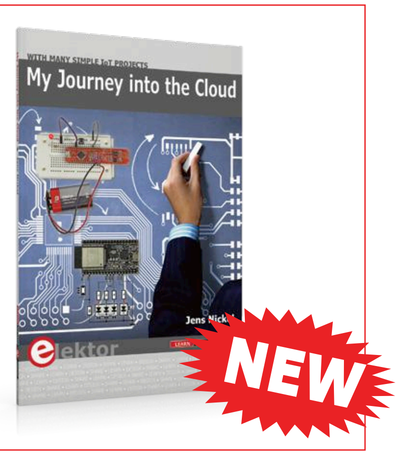


This book is an introduction to the ESP32 processor and describes the main hardware and software features of this chip. The book teaches the reader how to use the ESP32 hardware and software in practical projects. Many basic, simple, and intermediate level projects are given in the book based on the ESP32 DevKitC development board, using the highly popular Arduino IDE and also the MicroPython programming language.

member price: £27.95 • €31.46 • US $37

www.elektor.com/esp32-book

My Journey into the Cloud



Our German Editor-in-Chief has made its way into the IoT. As part of a series of articles, he developed several demo projects - from the lamp control in the home network up to a autarkic sensor board that sends datas to the cloud. With the slogan „Learning by Doing", topics such as TCP/IP, MQTT, control via smartphone, WiFi access, connection of a cloud service, object-oriented programming and much more are treated. In this book, the first 24 episodes of this IoT series compactly summarized.

member price: £11.95 • €13.45 • US $16

www.elektor.com/journey-into-cloud

Hexadoku The Original Elektorized Sudoku

Traditionally, the last page of Elektor Magazine is reserved for our puzzle with an electronics slant: welcome to Hexadoku! Find the solution in the gray boxes, submit it to us by email, and you automatically enter the prize draw for one of five Elektor book vouchers.

The Hexadoku puzzle employs numbers in the hexadecimal range 0 through F. In the diagram composed of 16 × 16 boxes, enter numbers such that **all** hexadecimal numbers 0 through F (that's 0-9 and A-F) occur once only in each row, once in each column and in each of the 4×4 boxes (marked by the thicker black lines). A number of clues are given in the puzzle and these determine the start situation.

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

Solve Hexadoku and win!

Correct solutions received from the entire Elektor readership automatically enter a prize draw for five Elektor Book Vouchers worth **$70.00 / £40.00 / €50.00 each**, which should encourage all Elektor readers to participate.

Participate!

Ultimately September 20, 2018, supply your name, street address and the solution (the numbers in the gray boxes) by email to:
hexadoku@elektor.com

Prize Winners

The solution of Hexadoku in edition 4/2018 (May & June) is: **E3B9F**.

The €50 / £40 / $70 book vouchers have been awarded to: Nico Kirchhorf (Germany); Tiago Ferreira (Portugal); Casimir Schmid (Switzerland); J.W. van Tuijl (Netherlands) and Steven Venter (South Africa).

Congratulations everyone!

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

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

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

MPLAB® Mindi™ Analog Simulator

Microchip's Free Software for Circuit Design

MPLAB® Mindi™ Analog Simulator reduces circuit design time and design risk by simulating analog circuits prior to hardware prototyping. The simulation tool uses a SIMetrix/SIMPLIS simulation environment, with options to use SPICE or piecewise linear modeling, that can cover a very wide set of possible simulation needs. This capable simulation interface is paired with proprietary model files from Microchip, to model specific Microchip analog components, in addition to generic circuit devices. Finally, this simulation tool installs and runs locally, on your own PC. Once downloaded, no internet connection is required, and the simulation run time is not dependent on a remotely located server. The result is fast, accurate analog circuit simulations.

MPLAB® MINDI™ ANALOG SIMULATOR

Key Benefits

- Perform AC, DC and transient analysis
- Validate system response, control and stability
- Identify problems before building hardware

microchip DIRECT
www.microchipdirect.com

MICROCHIP®

www.microchip.com/mindi