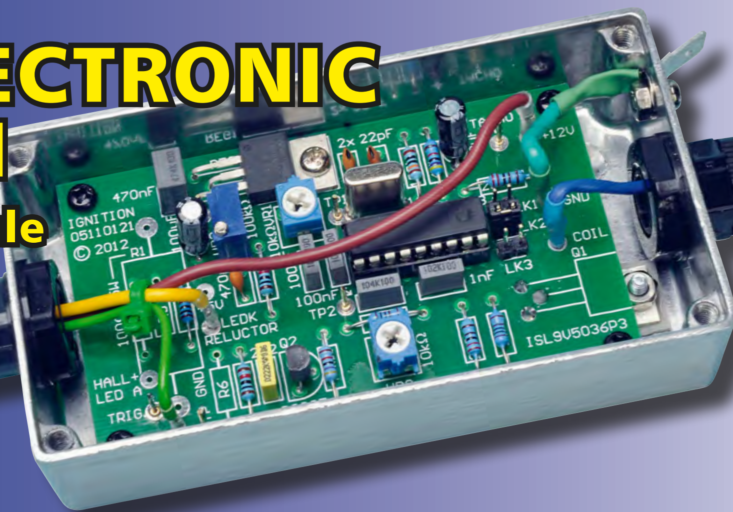
THE No 1 UK MAGAZINE FOR ELECTRONICS TECHNOLOGY & COMPUTER PROJECTS

EPE EVERYDAY PRACTICAL ELECTRONICS

www.epemag.com

HIGH-ENERGY ELECTRONIC IGNITION SYSTEM

Replace a failed ignition module
Upgrade a mechanical system
Innovative spark test mode
Latest IGBT drive technology



MOBILE PHONE LOUD RINGER!

NEVER MISS A CALL

WIN A MICROCHIP dsPICDEM MCLV-2 DEVELOPMENT BOARD

HIGH RESOLUTION FREQUENCY/PERIOD COUNTER

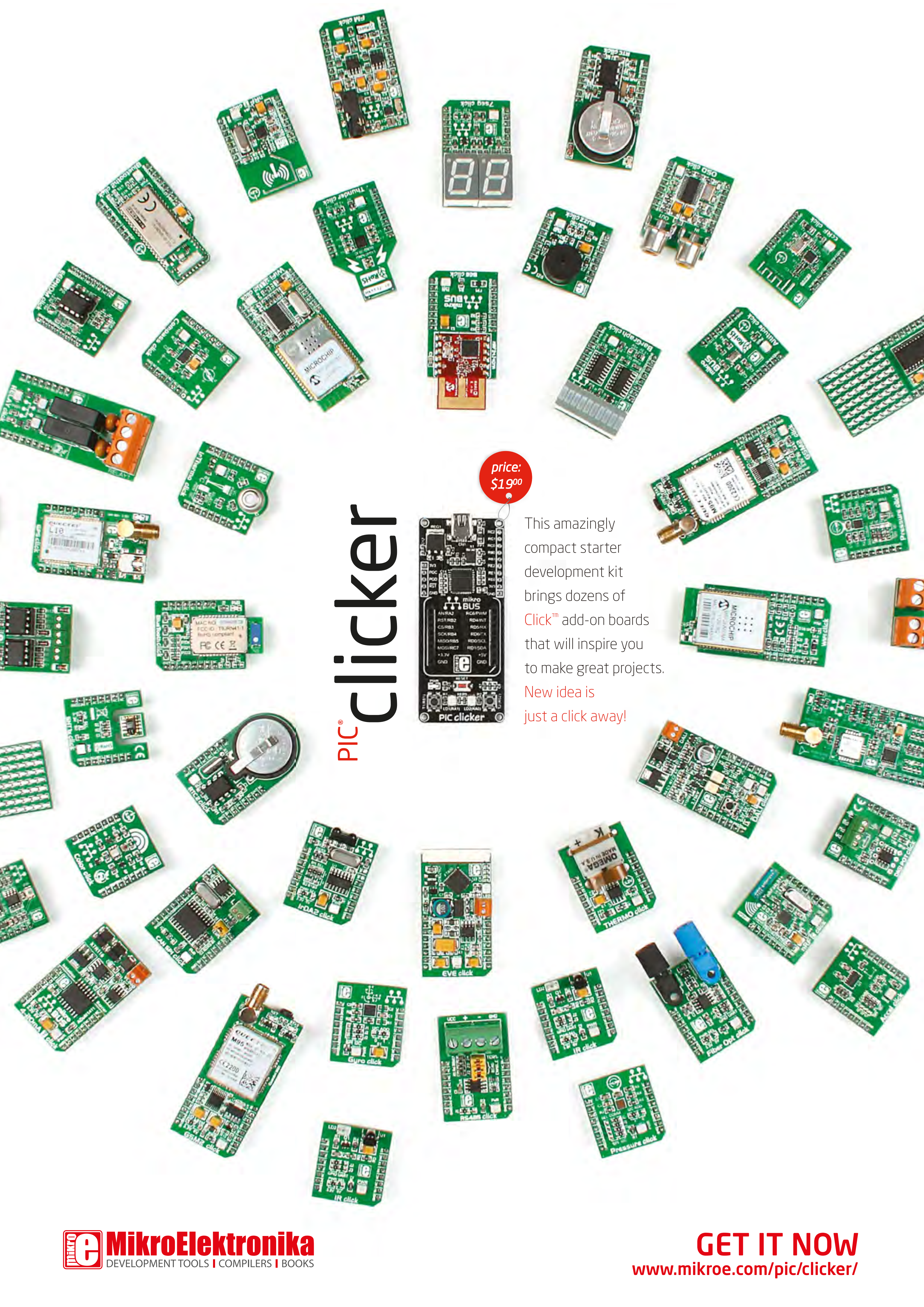
PART 2 – CONSTRUCTION AND SET UP PROCEDURE

Teach-In 2014 Raspberry Pi – Part 5

EPE's comprehensive guide to Raspberry Pi

FEB 2014 £4.40
02
9 770262 361188

PLUS: NET WORK, CIRCUIT SURGERY, TECHNO TALK
PIC N' MIX, READOUT AND INTERFACE

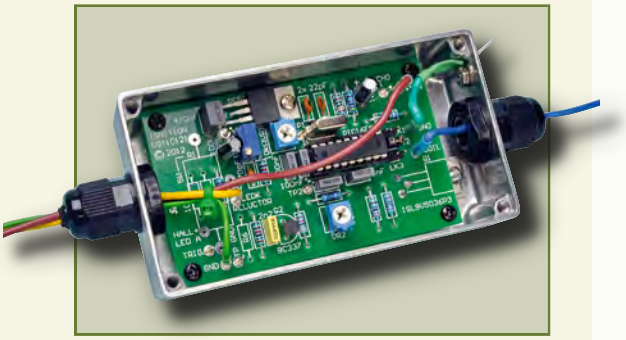

PIC® clicker
price: $19⁰⁰
This amazingly compact starter development kit brings dozens of Click™ add-on boards that will inspire you to make great projects.
New idea is just a click away!
MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS
GET IT NOW
www.mikroe.com/pic/clicker/

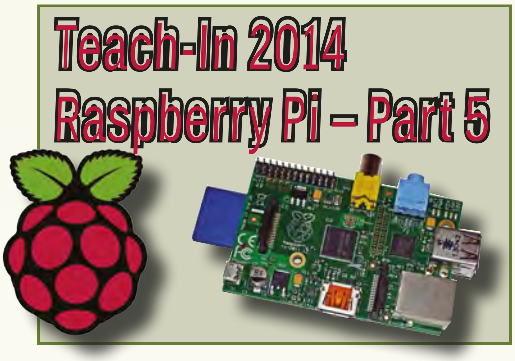
ISSN 0262 3617
• PROJECTS • THEORY •
• NEWS • COMMENT •
• POPULAR FEATURES •
VOL. 43. No 2 February 2014

EPE EVERYDAY PRACTICAL ELECTRONICS

INCORPORATING ELECTRONICS TODAY INTERNATIONAL

www.epemag.com





Our March 2014 issue will be published on Thursday 06 February 2014, see page 72 for details.

Projects and Circuits

Series and Features

Regulars and Services

QUASAR electronics

Solutions for Home, Education & Industry Since 1993

Quasar Electronics Limited
PO Box 6935, Bishops Stortford
CM23 4WP, United Kingdom
Tel: 01279 467799
Fax: 01279 267799
E-mail: sales@quasarelectronics.co.uk
Web: www.quasarelectronics.co.uk

All prices INCLUDE 20.0% VAT.
Postage & Packing Options (Up to 0.5Kg gross weight): **UK Standard 3-7 Day Delivery** - £3 95; **UK Mainland Next Day Delivery** - £8.95; **Europe (EU)** - £12.95; **Rest of World** - £14 95 (up to 0.5Kg).
Order online for reduced price Postage (from just £1)
Payment We accept all major credit/debit cards. Make PO's payable to Quasar Electronics Limited.
Please visit our online shop now for full details of over 1000 electronic kits, projects modules and publications. Discounts for bulk quantities.

01279 Credit Card Sales 467799

PIC & ATMEL Programmers

We have a wide range of low cost PIC and ATMEL Programmers. Complete range and documentation available from our web site.

Programmer Accessories:
40-pin Wide ZIF socket (ZIF40W) £14.95
18Vdc Power supply (PSU121) £22.95
Leads: Parallel (LDC136) £3.95 / Serial (LDC441) £3.95 / USB (LDC644) £2.95

USB & Serial Port PIC Programmer

USB/Serial connection. Header cable for ICSP. Free Windows XP software. See website for PICs supported. ZIF Socket and USB lead extra. 18Vdc.
Kit Order Code: 3149EKT - **£49.95**
Assembled Order Code: AS3149E - **£64.95**
Assembled with ZIF socket Order Code: AS3149EZIF - **£74.95**

USB Flash PIC Programmer

USB PIC programmer for a wide range of Flash devices—see website for details. Free Windows Software. ZIF Socket and USB lead not included. Powered via USB port - no external power supply required.
Assembled with ZIF socket Order Code: AS3150ZIF - **£64.95**

ATMEL 89xxxx Programmer

Uses serial port and any standard terminal comms program. 4 LED's display the status. ZIF sockets not included. Supply: 16Vdc.
Kit Order Code: 3123KT - **£28.95**
Assembled Order Code: AS3123 - **£39.95**

Introduction to PIC Programming

Go from complete beginner to burning a PIC and writing code in no time! Includes 49 page step-by-step PDF Tutorial Manual, Programming Hardware (with LED test section), Win 3.11—XP Programming Software (Program, Read, Verify & Erase), and 1rewritable PIC16F84A that you can use with different code (4 detailed examples provided for you to learn from). PC parallel port.
Kit Order Code: 3081KT - **£16.95**
Assembled Order Code: AS3081 - **£24.95**

PIC Programmer Board

Low cost PIC programmer board supporting a wide range of Microchip® PIC™ microcontrollers. Requires PC serial port. Windows interface supplied.
Kit Order Code: K8076KT - **£34.95**

PIC Programmer & Experimenter Board

The PIC Programmer & Experimenter Board with test buttons and LED indicators to carry out educational experiments, such as the supplied programming examples. Includes a 16F627 Flash Microcontroller that can be reprogrammed up to 1000 times for experimenting at will. Software to compile and program your source code is included.
Kit Order Code: K8048KT - **£34.95**
Assembled Order Code: VM111 - **£44.95**

Controllers & Loggers

Here are just a few of the controller and data acquisition and control units we have. See website for full details. 12Vdc PSU for all units: Order Code 660.446UK £10.95

USB Experiment Interface Board

5 digital input channels and 8 digital output channels plus two analogue inputs and two analogue outputs with 8 bit resolution.
Kit Order Code: K8055NKT - **£27.95**
Assembled Order Code: VM110N - **£40.95**

Rolling Code 4-Channel UHF Remote

State-of-the-Art. High security. 4 channels. Momentary or latching relay output. Range up to 40m. Up to 15 Tx's can be learnt by one Rx (kit includes one Tx but more available separately). 4 indicator LED 's. Rx: PCB 77x85mm, 12Vdc/6mA (standby). Two *& Ten Channel versions also available.*
Kit Order Code: 3180KT - **£54.95**
Assembled Order Code: AS3180 - **£64.95**

Computer Temperature Data Logger

Serial port 4-channel temperature logger. °C or °F. Continuously logs up to 4 separate sensors located 200m+ from board. Wide range of free software applications for storing/using data. PCB just 45x45mm. Powered by PC. Includes one DS1820 sensor.
Kit Order Code: 3145KT - **£19.95**
Assembled Order Code: AS3145 - **£26.95**
Additional DS1820 Sensors - **£4.95 each**

Remote Control Via GSM Mobile Phone

Place next to a mobile phone (not included). Allows toggle or auto-timer control of 3A mains rated output relay from any location with GSM coverage.
Kit Order Code: MK160KT - **£10.72**

Most items are available in kit form (KT suffix) or pre-assembled and ready for use (AS prefix).

4-Ch DTMF Telephone Relay Switcher

Call your phone number using a DTMF phone from anywhere in the world and remotely turn on/off any of the 4 relays as desired. User settable Security Password, Anti-Tamper, Rings to Answer, Auto Hang-up and Lockout. Includes plastic case. 130 x 110 x 30mm. Power: 12Vdc.
Kit Order Code: 3140KT - **£79.95**
Assembled Order Code: AS3140 - **£94.95**

8-Ch Serial Port Isolated I/O Relay Module

Computer controlled 8 channel relay board. 5A mains rated relay outputs and 4 opto-isolated digital inputs (for monitoring switch states, etc). Useful in a variety of control and sensing applications. Programmed via serial port (use our new Windows interface, terminal emulator or batch files). Serial cable can be up to 35m long. Includes plastic case 130x100x30mm. Power: 12Vdc/500mA.
Kit Order Code: 3108KT - **£74.95**
Assembled Order Code: AS3108 - **£89.95**

Infrared RC 12–Channel Relay Board

Control 12 onboard relays with included infrared remote control unit. Toggle or momentary. 15m+ range. 112 x 122mm. Supply: 12Vdc/0.5A
Kit Order Code: 3142KT - **£64.95**
Assembled Order Code: AS3142 - **£74.95**

Audio DTMF Decoder and Display

Detect DTMF tones from tape recorders, receivers, two-way radios, etc using the built-in mic *or* direct from the phone line. Characters are displayed on a 16 character display as they are received and up to 32 numbers can be displayed by scrolling the display. All data written to the LCD is also sent to a serial output for connection to a computer. Supply: 9-12V DC (Order Code PSU303). Main PCB: 55x95mm.
Kit Order Code: 3153KT - **£37.95**
Assembled Order Code: AS3153 - **£49.95**

3x5Amp RGB LED Controller with RS232

3 independent high power channels. Preprogrammed or user-editable light sequences. Standalone option and 2-wire serial interface for microcontroller or PC communication with simple command set. Suitable for common anode RGB LED strips, LEDs and incandescent bu bs. 56 x 39 x 20mm. 12A total max. Supply: 12Vdc.
Kit Order Code: 8191KT - **£29.95**
Assembled Order Code: AS8191 - **£39.95**

Hot New Products!

Here are a few of the most recent products added to our range. See website or join our email Newsletter for all the latest news.

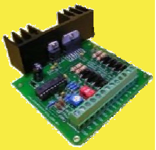
4-Channel Serial Port Temperature Monitor & Controller Relay Board

4 channel computer serial port temperature monitor and relay controller. Four inputs for Dallas DS18S20 or DS18B20 digital thermometer sensors (£3.95 each). Four 5A rated relay outputs are independent of sensor channels allowing flexibility to setup the linkage in any way you choose. Simple text string commands for reading temperature and relay control via RS232 using a comms program like Windows HyperTerminal or our free Windows application software.
Kit Order Code: 3190KT - **£84.95**
Assembled Order Code: AS3190 - **£99.95**

40 Second Message Recorder

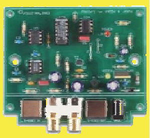
Feature packed non-volatile 40 second multi-message sound recorder module using a high quality Winbond sound recorder IC. Standalone operation using just six onboard buttons or use onboard SPI interface. Record using built-in microphone or external line in. 8-24Vdc powered. Change a resistor for different recording duration/sound quality. Sampling frequency 4-12 kHz.
Kit Order Code: 3188KT - **£29.95**
Assembled Order Code: AS3188 - **£37.95**
120 second version also available

Bipolar Stepper Motor Chopper Driver



Get better performance from your stepper motors with this dual full bridge motor driver based on SGS Thompson chips L297 & L298. Motor current for each phase set using on-board potentiometer. Rated to handle motor winding currents up to 2 Amps per phase. Operates on 9-36Vdc supply voltage. Provides all basic motor controls including full or half stepping of bipolar steppers and direction control. Allows multiple driver synchronisation. Perfect for desktop CNC applications.
Kit Order Code: 3187KT - **£39.95**
Assembled Order Code: AS3187 - **£49.95**

Video Signal Cleaner



Digitally cleans the video signal and removes unwanted distortion in video signal. In addition it stabilises picture quality and luminance fluctuations. You will also benefit from improved picture quality on LCD monitors or projectors.
Kit Order Code: K8036KT - **£24.70**
Assembled Order Code: VM106 - **£36.53**

Most items are available in kit form (KT suffix) or assembled and ready for use (AS prefix).

Motor Speed Controllers

Here are just a few of our controller and driver modules for AC, DC, Unipolar/Bipolar stepper motors and servo motors. See website for full details.

DC Motor Speed Controller (100V/7.5A)

Control the speed of almost any common DC motor rated up to 100V/7.5A. Pulse width modulation output for maximum motor torque at all speeds. Supply: 5-15Vdc. Box supplied. Dimensions (mm): 60Wx100Lx60H.
Kit Order Code: 3067KT - **£19.95**
Assembled Order Code: AS3067 - **£27.95**

Bidirectional DC Motor Speed Controller

Control the speed of most common DC motors (rated up to 32Vdc/10A) in both the forward and reverse direction. The range of control is from fully OFF to fully ON in both directions. The direction and speed are controlled using a single potentiometer. Screw terminal block for connections.
Kit Order Code: 3166v2KT - **£23.95**
Assembled Order Code: AS3166v2 - **£33.95**

Computer Controlled / Standalone Unipolar Stepper Motor Driver

Drives any 5-35Vdc 5, 6 or 8-lead unipolar stepper motor rated up to 6 Amps. Provides speed and direction control. Operates in stand-alone or PC-controlled mode for CNC use. Connect up to six 3179 driver boards to a single parallel port. Board supply: 9Vdc. PCB: 80x50mm.
Kit Order Code: 3179KT - **£17.95**
Assembled Order Code: AS3179 - **£24.95**

Computer Controlled Bi-Polar Stepper Motor Driver

Drive any 5-50Vdc, 5 Amp bi-polar stepper motor using externally supplied 5V levels for STEP and DIRECTION control. Opto-isolated inputs make it ideal for CNC applications using a PC running suitable software. Board supply: 8-30Vdc. PCB: 75x85mm.
Kit Order Code: 3158KT - **£24.95**
Assembled Order Code: AS3158 - **£34.95**

AC Motor Speed Controller (600W)

Reliable and simple to install project that allows you to adjust the speed of an electric drill or 230V AC single phase induction motor rated up to 600 Watts. Simply turn the potentiometer to adjust the motors RPM. PCB: 48x65mm. Not suitable for use with brushless AC motors.
Kit Order Code: 1074KT - **£15.95**
Assembled Order Code: AS1074 - **£23.95**

See www.quasarelectronics.com for lots more DC, AC and Stepper motor drivers

QUASAR electronics
The Electronic Kit Specialists Since 1993

Credit Card Sales 01279 467 799

Electronic Project Labs

Great introduction to the world of electronics. Ideal gift for budding electronics expert!

500-in-1 Electronic Project Lab

Top of the range. Complete self-contained electronics course. Takes you from beginner to 'A' Level standard and beyond! Contains all the hardware and manuals to assemble 500 projects. You get 3 comprehensive course books (total 368 pages) - *Hardware Entry Course*, *Hardware Advanced Course* and a microprocessor based *Software Programming Course*. Each book has individual circuit explanations, schematic and connection diagrams. Suitable for age 14+.
Order Code EPL500 - **£179.95**
Also available: 30-in-1 **£17.95**, 50-in-1 **£29.95**, 75-in-1 **£39.95**, 130-in-1 **£49.95** & 300-in-1 **£79.95** (see website for details)

Tools & Test Equipment

We stock an extensive range of soldering tools, test equipment, power supplies, inverters & much more - please visit website to see our full range of products.

Advanced Personal Scope 2 x 240MS/s

Features 2 input channels - high contrast LCD with white backlight - full auto set-up for volt/div and time/div - recorder roll mode, up to 170h per screen - trigger mode: run - normal - once - roll ... - adjustable trigger level and slope and much more.
Order Code: APS230 - ~~£374.95~~ **£324.95**

Handheld Personal Scope with USB

Designed by electronics enthusiasts for electronics enthusiasts! Powerful, compact and USB connectivity, this sums up the features of this oscilloscope. 40 MHz sampling rate, 12 MHz analog bandwith, 0.1 mV sensitivity, 5mV to 20V/div in 12 steps, 50ns to 1 hour/div time base in 34 steps, ultra fast full auto set up option, adjustable trigger level, X and Y position signal shift, DVM readout and more...
Order Code: HPS50 - ~~£289.96~~ **£204.00**

See website for more super deals!

No.1 FOR KITS

www.quasarelectronics.co.uk

Secure Online Ordering Facilities • Full Product Listing, Descriptions & Photos • Kit Documentation & Software Downloads

EPE EVERYDAY PRACTICAL ELECTRONICS BACK ISSUES

We can supply back issues of *EPE* by post, most issues from the past five years are available. An *EPE* index for the last five years is also available at **www.epemag.com**. Where we are unable to provide a back issue a photocopy of any one article (or one part of a series) can be purchased for the same price. Issues from Jan. 99 are available on CD-ROM or DVD-ROM – and back issues from recent years are also available to download from **www.epemag.com**. Please make sure all components are still available before commencing any project from a back-dated issue.

DID YOU MISS THESE?





BACK ISSUES ONLY £5.00 (each inc. UK p&p.)

Overseas prices £5.80 each surface mail, £6.80 each airmail.

We can also supply issues from years: 2006 (**except** Jan., Feb., Mar., Apr., May, July). 2007 (**except** June, July, Aug), 2008 (**except** Aug, Nov, Dec) 2009 (**except** Jan, Mar, Apr.). 2010 (**except** May, June, July, August, Oct, Nov, Dec) 2011 (**except** Jan).

Where we do not have an issue a photocopy of any ***one article*** or ***one part*** of a series can be provided at the same price.

DEC '12

PROJECTS • Universal USB Data Logger – Part 1 • Hot-Wire Cutter • Digital Lighting Controller – Part 3 • Hearing Loop Level Meter – Part 2 • Ingenuity Unlimited
FEATURES • Jump Start – Mini Christmas Lights • Techno Talk • PIC N' Mix • Circuit Surgery • Interface • Max's Cool Beans • Net Work

JAN '13

PROJECTS • 3-Input Stereo Audio Switcher • Stereo Compressor • Low Capacitance Adaptor For DMMs • Universal USB Data Logger – Part 2
FEATURES • Jump Start – iPod Speaker • Techno Talk • PIC N' Mix • Raspberry Pi – Keypad and LCD Interface • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work

FEB '13

PROJECTS • Semtest – Part 1 • Crystal DAC • 10W LED Floodlight • Built-In Speakers • Universal USB Data Logger – Part 3
FEATURES • Jump Start – Logic Probe • Techno Talk • PIC N' Mix • Raspberry Pi – Software Investigation • Circuit Surgery • Interface • Max's Cool Beans • Net Work

MAR '13

PROJECTS • Lightning Detector • SemTest – Part 2 • Digital Spirit Level • Interplanetary Voice • Ingenuity Unlimited
FEATURES • Jump Start – DC Motor Controller • Techno Talk • PIC N' Mix • Raspberry Pi – Further Investigation • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work

APR '13

PROJECTS • SoftStarter • 6-Decade Resistance Substitution Box • SemTest – Part 3
FEATURES • Jump Start – Egg Timer • Techno Talk • PIC N' Mix • Circuit Surgery • Interface • Max's Cool Beans • Net Work

MAY '13

PROJECTS • Electronic Stethoscope • PIC/AVR Programming Adaptor Board – Part 1 • Cheap, High-Current Bench Supply • Ingenuity Unlimited
FEATURES • Jump Start – Signal Injector Probe • Techno Talk • Raspberry Pi • PIC N' Mix • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work

JUNE '13

PROJECTS • Mix-It • PIC/AVR Programming Adaptor Board – Part 2 • A Handy USB Breakout Box • Converter For Neon Lamp Experiments • Ingenuity Unlimited
FEATURES • Jump Start – Simple Radio Receiver • Techno Talk • PIC N' Mix • Circuit Surgery • Interface • Max's Cool Beans • Net Work

JULY '13

PROJECTS • 6-Decade Capacitance Substitution Box • Soft Starter For Power Tools • High Power Brushless Motors From Old CD/DVD Drives • High-Current Adaptor For Scopes And DMMs
FEATURES • Jump Start – Temperature Alarm • Techno Talk • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work

AUG '13

PROJECTS • Driveway Sentry • Milliohm Meter Adaptor For DMMs • Build A Vox • Superb Four-Channel Amplifier – On The Cheap
FEATURES • Techno Talk • Circuit Surgery • Interface • Max's Cool Beans • Net Work • Computer Error: Reliable Digital Processing – Part 1

SEPT '13

PROJECTS • Digital Sound Effects Module • USB Stereo Recording & Playback Interface • Vacuum Pump From Junk • Minireg 1.3-22V Adjustable Regulator • Ingenuity Unlimited
FEATURES • Techno Talk • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work • Computer Error: Reliable Digital Processing – Part 2

OCT '13

PROJECTS • LED Musicolour – Part 1 • High-Temperature Thermometer/Thermostat • Ingenuity Unlimited
FEATURES • Teach-In 2014 – Part 1 • Techno Talk • Circuit Surgery • Interface • Max's Cool Beans • Net Work • Computer Error: Reliable Digital Processing – Part 3

NOV '13

PROJECTS • CLASSiC-D Amplifier – Part 1 • LED Musicolour – Part 2 • Mains Timer For Fans Or Lights • Ingenuity Unlimited
FEATURES • Teach-In 2014 – Part 2 • Techno Talk • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work

DEC '13

PROJECTS • Six Test Instruments In One Tiny Box • Virtins Technology Multi-Instrument 3.2 • CLASSiC-D Amplifier – Part 2 •
FEATURES • Teach-In 2014 – Part 3 • Techno Talk • Circuit Surgery • Interface • Max's Cool Beans • Net Work

JAN '14

PROJECTS • 2.5GHz 12-Digit Frequency Counter With Add-on GPS Accuracy – Part 1 • The Champion Amplifier • Simple 1.5A Switching Regulator •
FEATURES • Teach-In 2014 – Part 4 • Techno Talk • Circuit Surgery • Practically Speaking • Max's Cool Beans • Net Work • PIC N' Mix • Net Work

ORDER FORM – BACK ISSUES – PHOTOCOPIES– INDEXES

☐ Send back issues dated . . . .

☐ Send photocopies of (article title and issue date) . . . .

Name . . . .

Address. . . . .

. . . . Tel: . . . .

☐ I enclose cheque/P.O./bank draft to the value of £ . . . .

☐ Please charge my Visa/Mastercard/Maestro £ . . . .

Card No . . . . Maestro Issue No . . . .

Valid From . . . . Card Expiry Date . . . . Card Security Code . . . .
(The last 3 digits on or just under the signature strip)

SEND TO: **Everyday Practical Electronics, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU.**
Tel: 01202 880299. Fax: 01202 843233.
E-mail: orders@epemag.wimborne.co.uk **On-line Shop**: www.epemag.com

Payments must be in £ sterling – cheque or bank draft drawn on a UK bank. Normally posted within seven days of receipt of order. Send a copy of this form, or order by letter if you do not wish to cut your issue. M02/14

UK readers you can SAVE 81p on every issue of *EPE*

How would you like to pay £3.59 instead of £4.40 for your copy of *EPE*?




Well you can – just take out a one year subscription and save 81p an issue, or £9.80 over the year. You can even save £1.08 an issue if you subscribe for two years – a total saving of £26.10.

Overseas rates also represent exceptional value.

You also:

- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Ensure your copy, even if the newsagents sell out**

Order by phone or fax with a credit card or by post with a cheque or postal order, or buy online from www.epemag.com (go to the Online Shop).

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in the UK: 6 months £23.50, 12 months £43.00, two years £79.50; Europe Airmail: 6 months £27.00, 12 months £50.00, 24 months £95.00; Rest Of The World Airmail: 6 months £37.00, 12 months £70.00, 24 months £135.00.

Cheques or bank drafts (in **£ sterling only**) payable to *Everyday Practical Electronics* and sent to EPE Subs. Dept., Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU. Tel: 01202 880299. Fax: 01202 843233. **Email:** subs@wimborne.co.uk. Also via the **Web** at: **www.epemag.com**.

Subscriptions start with the next available issue. We accept MasterCard, Maestro or Visa. (For past issues see the Back Issues page.)

ONLINE SUBSCRIPTIONS

Online subscriptions, for reading the magazine via the Internet, £19.99 for one year, visit **www.epemag.com** for more details.

SUBSCRIPTION ORDER FORM

☐ **6 Months: UK £23.50, Europe £27.00 (Airmail), Rest Of The World £37.00 (Airmail)**

☐ **1 Year: UK £43.00, Europe £50.00 (Airmail), Rest Of The World £70.00 (Airmail)**

☐ **2 Years: UK £79.50, Europe £95.00 (Airmail), Rest Of The World £135.00 (Airmail)**

To: Everyday Practical Electronics, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel: 01202 880299 Fax: 01202 843233
E-mail: subs@epemag.wimborne.co.uk

I enclose payment of £ .... (cheque/PO in £ sterling only), payable to *Everyday Practical Electronics*

☐ Please charge my Visa/Mastercard/Maestro

My card number is: ....

Please print clearly, and check that you have the number correct

Signature ....

Card Security Code .... Valid From Date.......................

(The last 3 digits on or just under the signature strip)

Card Ex. Date ....Maestro Issue No. ....

Name ....

Address ....

Post code .... Tel. ....

Subscriptions can only start with the next available issue.

VOL. 43 No. 02 FEBRUARY 2014

Editorial Offices:
EVERYDAY PRACTICAL ELECTRONICS EDITORIAL Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU
Phone: (01202) 880299. **Fax:** (01202) 843233.
Email: fay.kearn@wimborne.co.uk
Website: www.epemag.com
See notes on **Readers' Technical Enquiries** below – we regret technical enquiries cannot be answered over the telephone.

Advertisement Offices:
Everyday Practical Electronics Advertisements
113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU
Phone: 01202 880299 **Fax:** 01202 843233
Email: stewart.kearn@wimborne.co.uk

Editor:	MATT PULZER
Subscriptions:	MARILYN GOLDBERG
General Manager:	FAY KEARN
Graphic Design:	RYAN HAWKINS
Editorial/Admin:	01202 880299
Advertising and Business Manager:	STEWART KEARN 01202 880299
On-line Editor:	ALAN WINSTANLEY
Publisher:	MIKE KENWARD

READERS' TECHNICAL ENQUIRIES
Email: fay.kearn@wimborne.co.uk
We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine. We regret that we cannot provide data or answer queries on articles or projects that are more than five years' old. Letters requiring a personal reply must be accompanied by a stamped self-addressed envelope or a self-addressed envelope and international reply coupons. We are not able to answer technical queries on the phone.

PROJECTS AND CIRCUITS
All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

A number of projects and circuits published in EPE employ voltages that can be lethal. You should not build, test, modify or renovate any item of mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD adaptor.

COMPONENT SUPPLIES
We do not supply electronic components or kits for building the projects featured, these can be supplied by advertisers.

We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

EPE EVERYDAY PRACTICAL ELECTRONICS

Welcome to the new year

A warm welcome to the first issue of 2014 from all of us at *EPE*. I hope Christmas brought you a few electronic goodies, but even if Santa missed your silicon-based hints, you can be sure that your favourite magazine is all set for a whole year of semiconductor fun and project inspiration. So what can we look forward to in 2014? Well, if you need an extra iPod charger, an RF detector probe, a mini audio mixer or an over-the-Internet relay controller, then these examples are just a small sample of the clever and useful projects coming your way. I won't spoil the fun with any more revelations, but we do have some real treats coming your way!

Bright spark

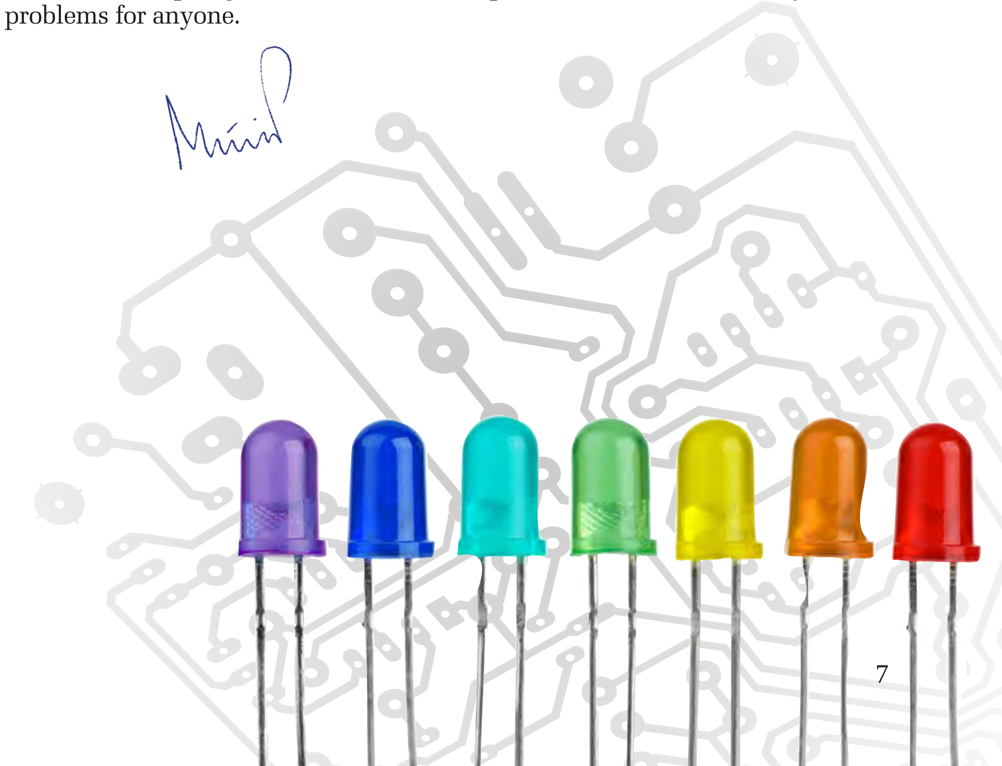
It's been a little while since we had a major automotive project, but car electronics is returning with a mighty zap of electron-based power! Our compact *High-Energy Electronic Ignition System* offers you the option of driving your spark plugs with the very latest in high-voltage finesse. From upgrading your current system to adding a touch of reliability to an old banger rebuild, this project will help you stay on the road reliably.

Light power

I enjoyed reading Mark Nelson's report on improving solar photovoltaic cells in this issue. It reminds me of an oft-heard criticism of UK-based solar power – why invest in solar power in Britain? The sun never shines here! Well, no one has told our plant life. This is a famously '*green* and pleasant land' – and it's green because nature's solar-powered plants thrive here. Green plants are green to capture and use the power of sunlight, and they seem to do it very well. True, you may get even more power from the Sahara desert or other cloud-free environments, but there is plenty of opportunity for usefully catching the sun's power here in Britain too. Next time you need a remotely-powered device, why not consider building in solar cells.

Mains Timer for fans or lights

Our thanks to readers who spotted an error in the *Mains Timer for fans or lights* project in the November 2013 issue of *EPE*. On the circuit diagram of Fig.3 page 32, CON1 is mislabeled. From top to bottom, the correct annotation should be: Lperm, Lload, Lsw, N. The wiring diagram on Fig.4 is correct. We apologise for the error and hope it did not cause too many problems for anyone.



NEWS

A roundup of the latest Everyday News from the world of electronics

Bright, brighter, brightest – by Barry Fox

Projection remains the most cost-effective way to show really big pictures. There are two competing technologies; either light is shone through small red, green and blue LCD panels, or light is reflected through a spinning wheel of colour filters by a DLP array of micro-mirrors. The DLP camp claims that LCD panels degrade in the intensely hot light, and the LCD camp points to visible artefacts like the 'rainbow effect' seen with some DLP systems.

In the battle for projector market share, Epson is claiming its LCD panel technology delivers the greatest brightness – providing the 'correct' measurement technique is used.

The Epson approach

Epson, which sells only three-panel LCD projectors, is now gearing up to attack the manufacturers of single chip DLP projectors on the grounds that they are over-claiming light output by using out-dated measuring techniques.

'Our LCD projectors are three times brighter than single chip DLP projectors' claimed Paul Wilson Epson UK's product manager at a recent London seminar held to air Epson's beef. 'This is shown when you use CLO (colour light output) measurements. WLO (whole light output) does not guarantee bright colour images'.

Epson's argument is that the filter wheels used in single-mirror-chip DLP projectors put a large clear segment between the red, green and blue filter segments to make the picture look brighter. But the picture only looks brighter because of the high white content, not combined RGB colour, and traditional light measurements do not take account of this.

Measurement battle

The traditional (WLO) method of measuring projector brightness is in ANSI lumens, and was devised by the American National Standards Institute. A white light screen is divided into nine sectors, like a chess board, and nine independent measurements taken, in lux or lumens per square metre. The nine measurements are then averaged to give the lumens rating.

CLO uses a standard published by the Society for Information Displays (SID) in May 2012. As with WLO, CLO is measured at nine specific points on a projected grid, but this time the grid is made up of the primary colours (red, green and blue). The three patterns are displayed in turn and the measurements taken, nine on each pattern. The results are then averaged to give the overall colour brightness level in lumens.

'We ran a 'blind' test with 200 people in the UK and asked the independent Intertek Labs to take measurements' Paul Wilson explains. 'We compared an Epson three-panel LCD projector with five DLP models. When brightness is measured by CLO, DLP projectors lose on average two thirds of their claimed brightness rating. Taking LCD as 100%, DLP is on average 33%'

Business *vs* consumer models

Point of sale placards shown by Epson at the seminar promoted a '3×' light gain for LCD over DLP. However, under questioning, Epson qualified this claim by saying that the '3×' figures are based on business and data projectors, rather than home cinema models. When home cinema projectors are compared, their CLO averages around 50% of the manufacturers' claimed brightness, because home cinema projectors have larger colour filter segments than business models.

Epson now says that it will claim 'up to 3×' more brightness in its projector advertising.

Mind-controlled 'robot' arm

Corporal Andrew Garthwaite, who lost his arm while bravely serving his country in Afghanistan, has become the first person in the UK to receive a mind-controlled prosthetic limb.

Andrew was severely injured by a rocket-propelled grenade on operations in 2010. He lost his entire right arm but was given the opportunity to participate in a revolutionary nerve transfer surgery known as 'targeted muscle reinnervation'. After 18 months of rehabilitation, he is now able to control movement of his prosthetic arm with his mind. Focusing his thoughts on the nerves connected to muscles in his chest, he can open and close his right hand.

Targeted muscle reinnervation is a surgical procedure to redirect the nerves that were originally used to control the limb to a new muscle group. For those who have lost their arm at the shoulder, the nerves are usually redirected to the chest muscles. The selective nerve transfer surgery connects the existing functioning nerves, which previously controlled arm and hand movements, to other muscles that can read the signals. After the surgical procedure, electrodes sense the signals from the redirected nerves to drive the arm's motors.

Mind-body-electronic control of a prosthetic arm (Photo: Crown copyright)

Growth in mobile smart wearable devices

Google Glass is leading the way with wearable smart devices, but other devices, particularly 'smart watches' are in the pipeline

New findings from hi-tech analysts, Juniper Research, has revealed that smart wearable device shipments, including smart watches and glasses will approach 130 million by 2018, 10 times more than 2013. The report notes that this change in adoption levels is attributed to heightened consumer awareness of wearable technology and better visibility of product adoption.

Single charger for notebooks to reduce e-waste

IEC, the international standards and conformity assessment body for all fields of electrotechnology, has announced the publication of the first globally relevant technical specification for a single external charger for a wide range of notebook computers and laptops

Each year, billions of external chargers are shipped globally. Power supplies for notebooks weigh typically around 300 grams but sometimes up to 600 grams. They are generally not usable from one computer to the next. Sometimes they get lost or break, leading to the discarding of computers that may still work perfectly well.

It is estimated that the total e-waste related to all kinds of chargers of ICT devices (information and communication) exceeds half a million tons each year; basically the equivalent of 500,000 cars.

This new IEC technical specification covers critical aspects of external chargers for notebook computers, their connector and plug, as well as safety, interoperability, performance and environmental considerations.

Tiny spacecraft sends signals to Bletchley

A group of amateur radio and space enthusiasts based at Bletchley Park has been celebrating the successful launch in Russia of their very own satellite – the Funcube-1.

FUNcube-1 is a small 'CubeSat', which weighs less than 1kg and measures only 10×10×10cm. The first signals, picked up by volunteer radio enthusiasts at monitoring stations at locations including South Africa and Hawaii, gave the team watching its progress at the National Radio Centre in Bletchley Park detailed information about its position, temperature and the performance of the battery.

Funcube-1's primary mission is educational outreach to schools and colleges. This outreach is intended to encourage interest in STEM (science, technology, engineering and maths).

The volunteers behind the project intend that, in time, schools will be able to receive the signals directly using a simple USB dongle receiver and a small aerial in the playground.

Further information is available at: **http://funcube.org.uk**

New Microchip motor controllers

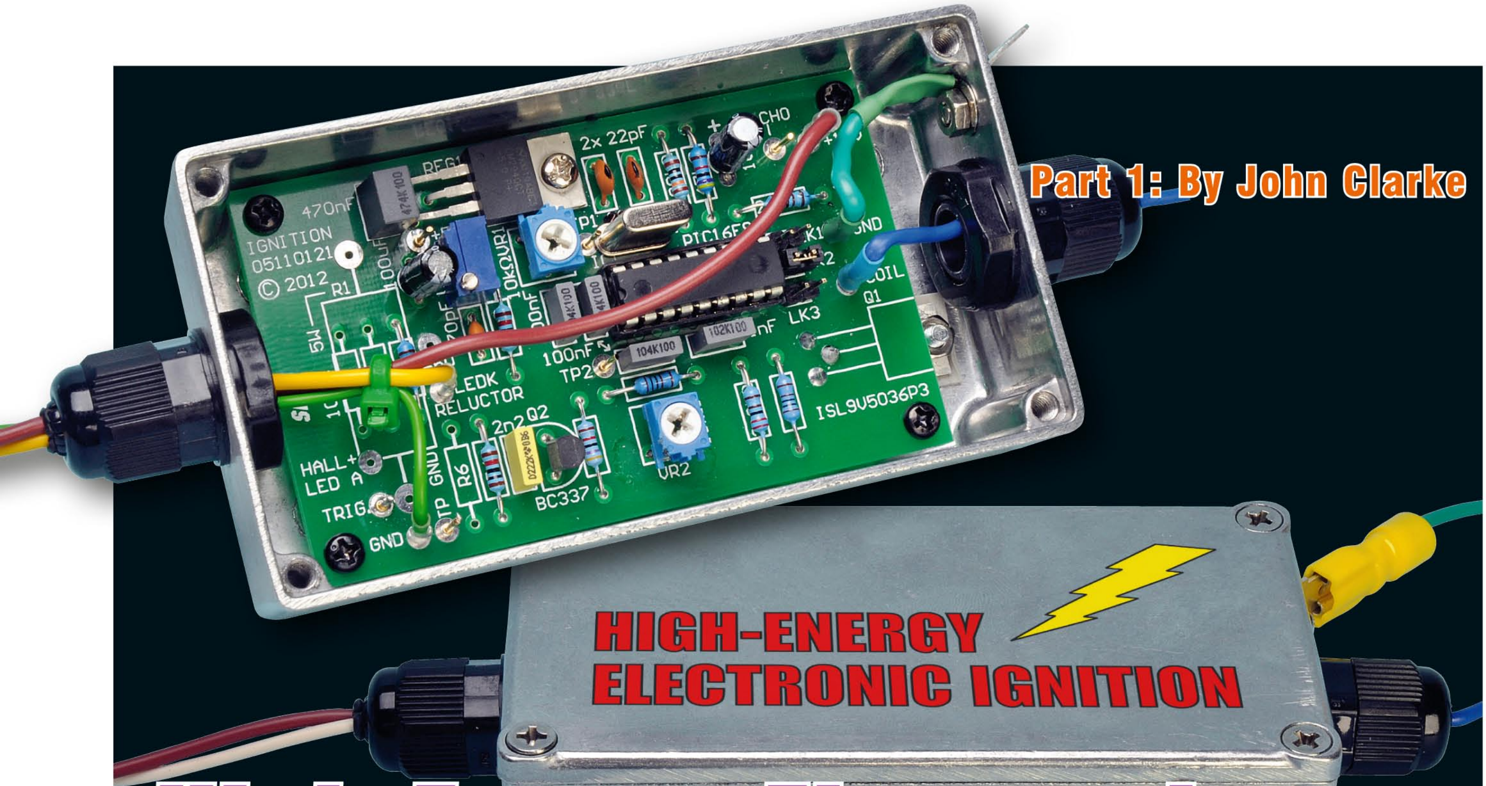
Microchip has announced a new set of dsPIC33 digital signal controllers (DSCs): the dsPIC33EP512GM710 family. The family adds higher levels of integration for motor-control applications, efficient dual motor control with 12 motor-control PWM channels (6 pairs), dual 12-bit ADCs and multiple 32-bit quadrature encoder interfaces.

The controllers support 512KB of Flash memory, 48KB of RAM, and four op amps. Other features of this family include enhanced analogue functionality with two independent ADC modules, configurable as 10-bit 1.1Msps with four sample and hold circuits, or 12-bit, 500ksps with one sample and hold, supporting up to 49 channels.

For more information, visit Microchip's website at: **www.microchip.com/get/V37V**

Microchip has also announced a new three-phase BLDC motor gate driver with power module: the MCP8024. This new device includes functions that power dsPIC digital signal controllers (DSCs) and PIC microcontrollers with capabilities to drive six N-channel MOSFETs.

The MCP8024 operates across a wide voltage range of 6V to 28V and can withstand transient voltage up to 48V. It is supported by Microchip's MCP8024 TQFP BLDC Motor Driver Evaluation Board, priced at $99.00. For more information, visit: **www.microchip.com/get/RTAS**



Part 1: By John Clarke

High-Energy Electronic Ignition System

This new circuit improves upon a traditional high energy electronic ignition system. It uses an IGBT ignition driver rather than the expensive high-voltage Darlington used in older designs. You can use it to replace a failed ignition module or to upgrade a mechanical ignition system when restoring a vehicle.

IT'S HAPPENED TO many of us – one day, you are driving around in a perfectly serviceable if older vehicle and then it quits on you, or it simply won't start the next morning. You take it to your local friendly mechanic who tells you that the ignition module has failed and will need to be replaced, but because of the age of the vehicle (and possibly its overseas origin) the repair job will cost you many hundreds.

However, you are an *EPE* reader, so you have a big advantage; you can build this substitute module for a fraction of the cost. Or maybe you have an older vehicle which has the old points ignition and you want to upgrade it to electronic ignition. Once again, our new module is the answer.

This new *High-Energy Ignition* suits vehicles with points, Hall effect/Lumenition sensors, optical sensors (eg, Crane or Piranha) and reluctor pick-ups. In fact, it will work with virtually any ignition system that uses a single coil, even those controlled by an engine management computer.

Better and simpler

We've improved on older electronic designs in a number of important ways. The main change is the use of an IGBT (insulated-gate bipolar transistor) ignition driver. This features integrated protection and is the type of device used in virtually all new cars.

The Darlington transistors used in older designs are not only larger and more expensive, but require a string of Zener diodes to protect them against the high-voltage back-EMF from the ignition coil. Plus, they require extra driving circuitry, some of which is bulky, that the IGBT simply does not need. The resulting, much smaller module will be much easier to install, especially in motorcycles.

We have also built a self-test feature into this unit, which means you can do a bench test to check it's working without needing a signal source to

Features

- Multiple trigger source options
- Trigger invert option
- Adjustable dwell time
- Option for output to follow input
- Spark test mode
- Tachometer output
- Adjustable debounce period
- Dwell compensation for battery voltage
- Simplified design using ignition IGBT to switch the coil
- Coil switch-off with no trigger signal

drive it. Similarly, it can be used as a stand-alone ignition coil tester.

This system uses a PIC16F88 microcontroller as the 'smarts', and naturally we have worked hard to improve the software over older designs.

Advantages of the IGBT

Older electronic ignition designs used a Darlington transistor to switch the ignition coil (eg, the BU941P and the MJH10012) – both are high-voltage transistors specifically intended for use in automobile ignition systems. But that approach has been obsolete for some time and all new cars now use IGBT ignition drivers, enabling a much simpler circuit.

Our previous Darlington circuits (eg, *EPE*, Sep-Nov 2009) were similar to that shown in Fig.1(a). The 100Ω 5W resistor provides 120mA of base drive to ensure that the Darlington transistor switches on fully – ie, it is saturated. Transistor Q2 is driven from a 5V signal and when on, shunts Q1's base drive to ground to switch it off. Q1 also required four series 75V Zener diodes to clamp the coil voltage to about 300V (to protect the transistor).

With an IGBT coil driver (Fig.1(b)), none of this extra circuitry is required. The IGBT is effectively a cross between a transistor and a MOSFET (a hybrid, if you like). Like a MOSFET, it is easy to drive from a voltage source, but it has the high-voltage performance of a bipolar transistor and is capable of switching the inductive load of the ignition coil.

As with a logic-level N-channel MOSFET, it is switched on when 5V is applied to its gate terminal via the 1kΩ resistor, while a low gate voltage switches it off.

The Zener diodes are no longer necessary because this type of IGBT incorporates internal voltage clamping to protect both the gate and the collector. When the collector voltage exceeds about 360V, an internal Zener diode conducts and switches the IGBT on to shunt the current to ground. The gate is protected from over-voltage with internal back-to-back Zener diodes.

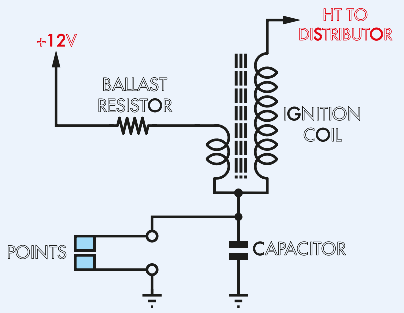


Fig.2: the Kettering ignition system uses points to interrupt the current through a coil. When the points open, the coil's magnetic field collapses and this produces a high voltage in the secondary, which is fed to the spark plugs via the distributor and the plug leads.

Kettering system

Fig.2 shows the arrangement for a Kettering ignition, which is the good old-fashioned points system. It comprises points (operated by a cam in the distributor), a capacitor (also known as the 'condenser'), an ignition coil and a distributor.

The primary winding of the ignition coil is connected to the +12V supply, and when the points are closed, current flows through the coil, causing energy to be stored in its magnetic field. This field collapses when the points open, generating a high voltage. The coil secondary has many more turns

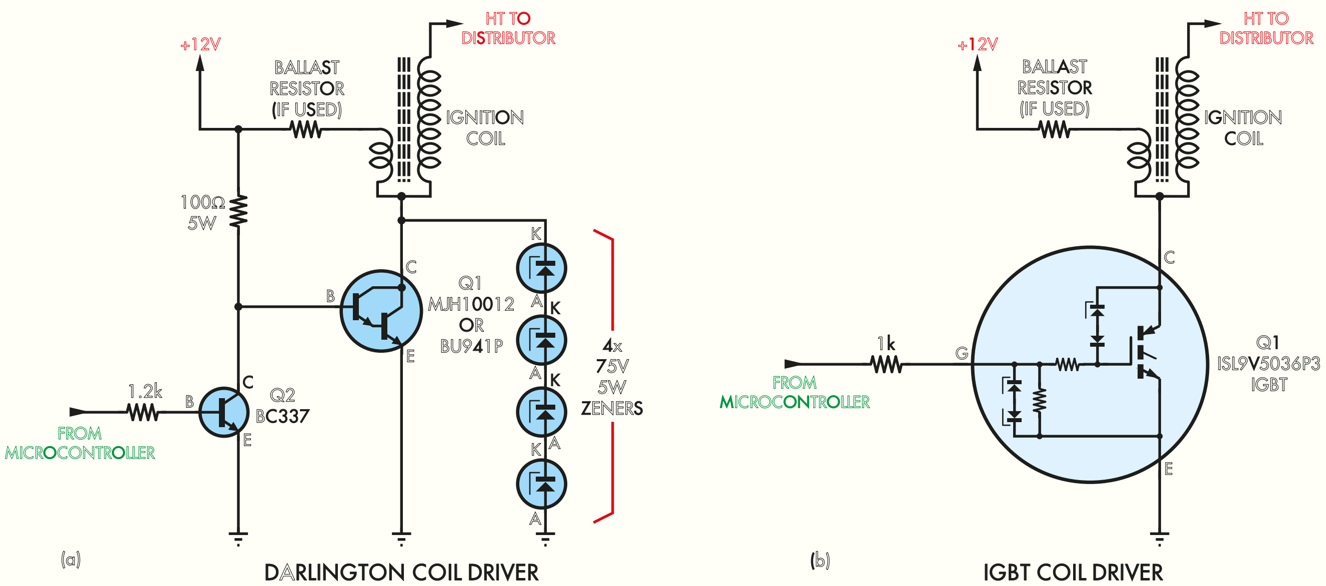


Fig.1: most cars of the last 20 years use an electronic trigger pick-up and an engine management computer to drive an electronic power device to switch the ignition coil on and off. Our previous designs used a Darlington transistor to switch the ignition coil as shown at (a), while our latest design uses an IGBT ignition coil driver to do the job, see: (b). The IGBT has in-built protection and this greatly reduces the parts count, increases reliability and simplifies construction.

Specifications

Debounce: 166µs to 5ms in 30 steps

Dwell: 129µs to 26ms in 200 steps (graded for more resolution at the lower values) for signals above 3.125Hz. Below 3.125Hz, the dwell automatically increases to the full period between firing minus the 1ms spark period.

Latency from trigger edge to firing: 18µs (10µs due to the IGBT response time)

Spark test rate: 15-75Hz (adjusted using trimpot VR2)

Spark test dwell: 129µs to 26ms (no dwell extension with battery voltage included)

Coil switch-off delay: after 10s with no trigger signal for debounce period above 2ms; after 1s for debounce period below 2ms

Dwell extension with battery: progressively increases from 2x below 12V through to 4x at 7.2V supply and below

Spark period: 1ms minimum

Maximum RPM for 1ms debounce and 1ms spark: 15,000 RPM for 4-cylinder, 10,000 RPM for 6-cylinder and 7500 RPM for 8-cylinder engine (4-stroke)

than the primary and so it produces a higher voltage again, creating a spark across the spark plugs in the engine.

The capacitor is there to prevent unnecessary arcing across the points, which would otherwise quickly become pitted and worn. Even so, there will always be some contact damage to the points due to sparking and so they need to be replaced on a regular basis – unless, that is, you install our electronic ignition module.

The coil charge period and the spark duration is set by the points' opening and closing periods. These are determined by the distributor cam lobe design and the points gap setting. During the dwell period, the points are closed to charge the coil. This dwell period reduces as RPM increases; at high RPM, spark energy can drop off badly as the coil does not have sufficient time to fully charge between each spark.

Refinements to the Kettering system allow the ignition timing to vary with RPM and manifold vacuum (ie, engine load). The RPM advance uses a system of centrifugal weights that move outward with higher rotational speed. These weights then advance the position of the cam and its lobes relative to the distributor drive shaft from the motor.

To vary the spark with engine load, a vacuum-driven actuator can rotate the points relative to the camshaft to produce timing changes with varying manifold pressure.

When starting the engine, the high starter motor current draw drops the battery voltage, reducing the spark voltage. This effect is worst right when maximum spark energy is needed; especially starting in cold weather. To solve this problem, the ignition coil is designed to deliver a healthy spark even with a ballast resistor connected in series with the 12V supply. During starting, the ballast resistor is shorted to increase the coil current drive and thus maintain sufficient spark energy.

Electronic ignition

Adding a switching transistor to a Kettering ignition system has many advantages. The main one is that the points no longer need to carry a high current – only enough to switch the transistor (and to keep the points clean). This minimises points wear, so that the only significant wear is to the rubbing block. That wear is insignificant and so the engine doesn't need to be re-tuned anywhere near as often.

Alternatively, the points can be replaced by Hall Effect, reluctor or optical triggering, thereby reducing ignition system maintenance to virtually nothing.

A secondary advantage of electronic ignition is that the dwell and spark duration are much more consistent, giving smoother engine running. The effect of reduced spark energy at higher RPM can also be alleviated, since with the electronic ignition module, coil charging can begin immediately after spark firing if necessary and the spark period can be kept low (1ms).

Features

Note that this particular design does not incorporate programmable timing. Instead, it uses the existing timing advance curve that is incorporated into the distributor. If you need a programmable electronic ignition system, we published a suitable design in the Sep-Nov 2009 issues of *EPE*.

This new unit includes an adjustable debounce period, adjustable dwell time and increased dwell with low battery voltage. It also features a special 'follow' operational mode for points if the distributor shaft, points cam and points are badly worn (more on this later). In addition, there is a spark test facility that allows the dwell to be easily adjusted to suit the ignition coil in use.

The spark test feature also allows an ignition coil to be tested on the bench over a range of spark frequencies.

During normal operation, the ignition coil is switched on for a sufficient dwell period just before firing. This allows the coil to charge fully without consuming any more power than necessary or overheating due to high saturation current. If the engine RPM becomes so high that the dwell period cannot fit between successive firings of the coil, the dwell period is reduced. The firing period is a minimum of 1ms, sufficient for the coil to deliver a healthy spark.

Dwell time

The dwell can be set between 129µs and 26ms in 200 steps, with more resolution available for the shorter periods. Some coils require a minimum 4ms dwell, while high-performance sports coils need less. The spark test feature basically allows the dwell to be adjusted to its optimal value while watching the spark delivered from the coil across a spark plug gap.

The dwell time is automatically extended when the battery voltage falls below 12V, to compensate for the longer charging period required. This helps maintain spark energy when starting the engine. This is disabled in the spark test mode.

Another important feature with this electronic ignition module is that the coil is not energised until after the engine has begun to turn. This prevents the coil from overheating and possibly burning out when the ignition is first switched on but the engine is not turning over (ie, not being cranked). Also, if the engine stops with the ignition still switched on, the coil is automatically switched off after one second.

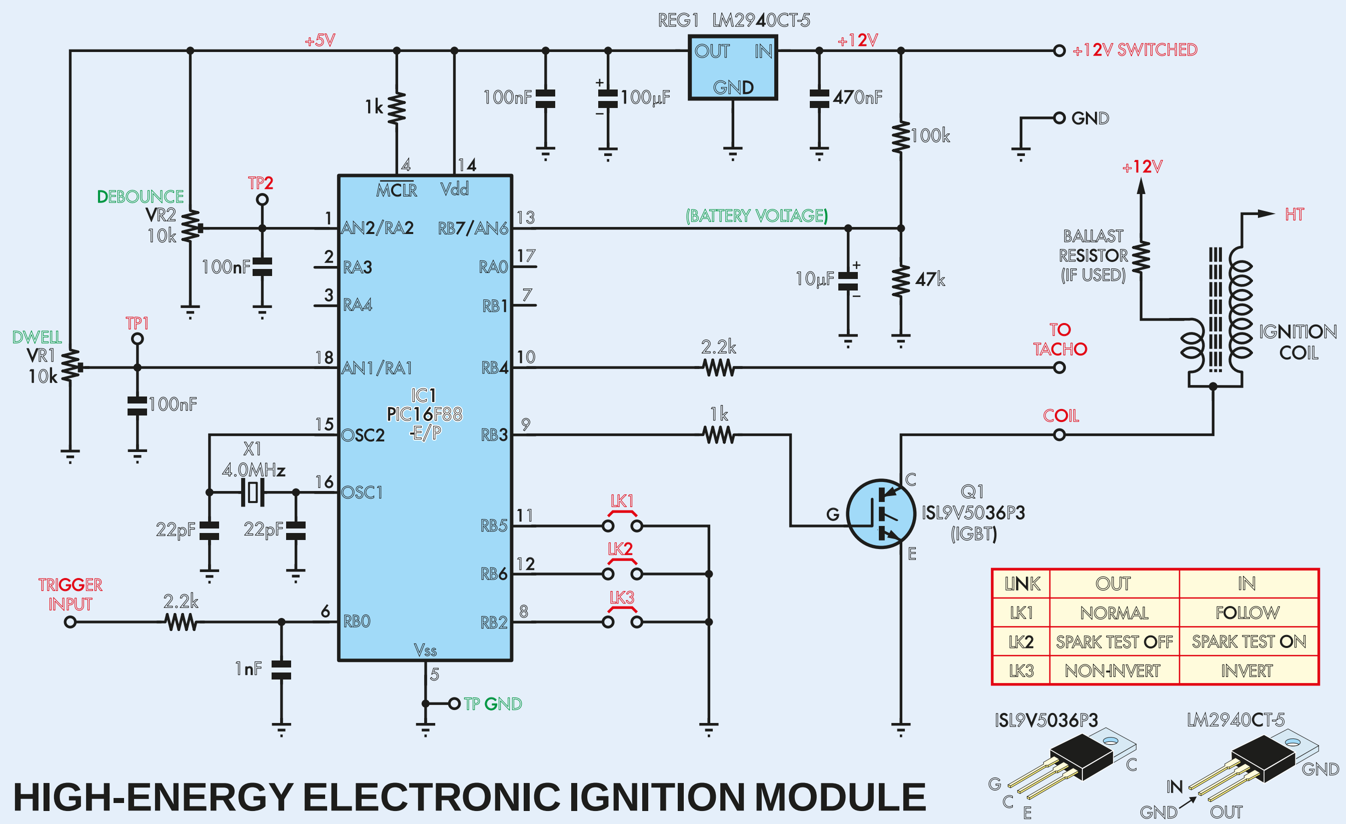


Fig.3: the circuit is based on PIC microcontroller IC1. It accepts the trigger signal, calculates the dwell and the spark period and produces an output at RB3 to drive IGBT transistor Q1. Output RB4 provides the tachometer drive signal.

However, this one-second period could be too short for a single-cylinder motorcycle engine to start when kick-starting. To solve this, if the debounce setting is more than 2ms, the coil switch-off delay is increased to 10s. In this case, the ignition coil must be able to withstand the application of 12V for 10s. Most coils designed for use with points are suitable as they are designed to cope if the motor stops with the points closed.

Debounce is included to prevent the ignition from being re-triggered due to noise on the trigger input. A 0.5ms period can be used with most sensor types, but a longer period is needed for points as they do not tend to open or close cleanly. Instead, points can bounce back open after closing and this can result in a series of rapid openings and closings.

The debounce feature enables the circuit to ignore this. However, there is a limit to the length of this debounce period. If it is made too long, then the upper RPM range can be severely limited as the time between plug firings approaches the debounce period.

A 2ms debounce period for a single-cylinder engine will not present such a problem. In fact, the upper RPM limit with a 5ms debounce period and a 1ms spark duration is 20,000 RPM for a single-cylinder 4-stroke engine.

Follow mode

In order to cope with severe points bounce, we have provided a 'follow' mode. When this mode is selected, the ignition system's output simply follows the input. This means that the coil begins charging as soon as the points close and the spark duration is not limited to 1ms.

In other words, much of the internal 'smarts' which attempt to optimise coil charging are disabled in the follow mode. However, the debounce setting is still effective, to prevent false triggering.

Note that the follow mode should only be selected when using points that produce erratic firing with the normal setting.

Finally, the system also includes an option to invert the input sense, so that the coil can fire on either the rising or falling edge of the input signal. For points, coil firing always occurs when the points open (ie, on the rising edge). However, for other triggers, you may need to fire on either the rising edge or the falling edge.

Circuit description

Refer now to Fig.3 for the main section of the *High-Energy Electronic Ignition System* circuit. The various trigger section options are shown in Fig.4.

Microcontroller IC1 is at the heart of the circuit. As shown, the trigger signal is applied to its pin 6 input (RB0). IC1 then processes this trigger signal and produces an output signal to drive the IGBT (Q1) at pin 9 (RB3).

The pin 6 input is protected from voltage spikes by a 2.2kΩ resistor. This limits the current if the internal clamping diodes between the input and each supply rail conduct. The associated 1nF capacitor provides high-frequency filtering to prevent false triggering.

In operation, IC1's RB3 output is alternately switched high to +5V to turn on Q1 and charge the coil, then to 0V in order to turn off Q1 and fire the spark plug when required. In addition, a second output is made

Restoring an older vehicle

Ignition systems for cars and motorcycles have improved greatly over recent years, with increased spark energy across the entire rev range of the engine. Much of this improvement has been achieved by using separate ignition coils for each spark plug. The 'old-fashioned' single coil and distributor is now rapidly becoming a relic.

But some older cars and motorcycles have a particular appeal and many are still in regular use. Enthusiasts often claim that these vehicles have more 'personality' and are more 'fun' to drive than modern counterparts.

So, restoring an older vehicle to its former glory has a certain appeal. Commonly restored cars include the original VW Beetle and Kombi vans with air-cooled horizontal engines, early model Vauxhall, Ford and Leyland vehicles, and classic marques such as MG, Morgan, Ferrari, Lancia, Citroen, Jaguar, Porsche and others.

Similarly, motorcycle enthusiasts revere the Norton Commando, Triumphs, BMWs, Moto Guzzis, Ducatis, Indians and Harley Davidsons. Many of these companies are still in business, but their older models are still popular.

These older cars and motorcycles use a Kettering ignition system, ie, one that comprises points, an ignition coil and distributor (Fig.1). This system usually benefits greatly with the addition of an electronic ignition module and that's where this project's unit comes into play.

Note, however, that this ignition module is not suitable for use with, or as a replacement for, a magneto ignition or a capacitor discharge ignition (CDI). These are found on some older motorcycles and in particular 2-strokes. To cater for these units, we published a replacement CDI module in the October 2010 issue of *EPE*. This design uses the high voltage generated by the vehicle's magneto to charge a capacitor. That charge is then dumped into the spark plugs via the ignition coil when triggered.

available at RB4 (pin 10). This produces a 5V square-wave to drive a suitable tachometer via a 2.2kΩ resistor. Note, however, that an impulse tachometer will usually be connected to the ignition coil instead.

In order to correctly process the trigger signal, IC1 monitors three separate voltages. The first is the battery voltage, at the AN6 input (pin 13). The battery voltage is first divided by 3.13 by the 100kΩ and 47kΩ resistors and filtered by a 10µF capacitor. The resulting voltage is then converted to a digital value using the micro's internal ADC and this is used to adjust the dwell time with low battery voltages.

The dwell and debounce periods are set using trimpots VR1 and VR2, each connected across the 5V supply. VR1 (dwell) is monitored by input AN1 (pin 18), while VR2 (debounce) is monitored by input AN2 (pin 1).

The dwell is adjustable from 129µs to 26ms and is set by monitoring the voltage at TP1. However, this voltage is not linearly proportional to the dwell period, to allow finer resolution for shorter dwell periods. The relationship between the two is shown in a graph to be published next month.

By contrast, the debounce period can be set anywhere from 0-5ms. This is done by monitoring the voltage at TP2, with 1V on AN2 equivalent to 1ms (ie, the relationship is linear).

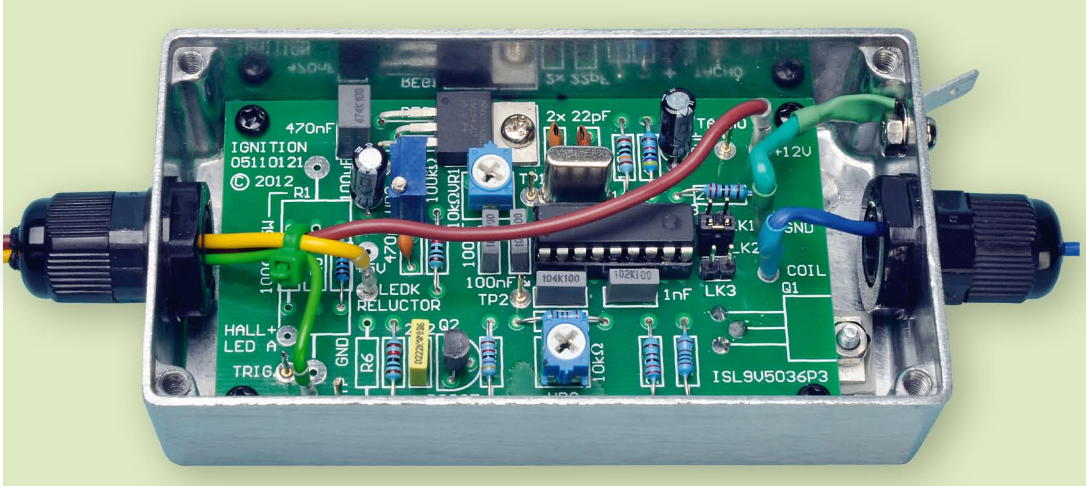
Links LK1-LK3 are used to select the various operational modes (see Table on Fig.3). These links connect to the RB5, RB6 and RB2 inputs (pins 11, 12 and 8) respectively. Internal pull-up resistors are enabled by IC1, so these inputs are held high with no jumper fitted. If a link is fitted, its corresponding input is pulled to 0V.

The default setting is with all jumpers out, for normal operation. The invert link (LK3) is fitted if the trigger sense needs inverting, while LK1 is fitted to enable the 'follow' mode (this mode is used with very noisy points, as explained earlier).

The spark test mode, selected when LK2 is fitted, causes the unit to charge and fire the coil at a rapid rate, regardless of the state of the trigger input. This allows a coil (or the module itself) to be tested without installing the unit in a vehicle. In this mode, trimpot VR1 is set to a fully anti-clockwise setting and then wound clockwise to give the best visual spark. VR2 can be used to set the spark rate, with a range of 15-75Hz (clockwise for increased frequency).

Bits and pieces

IC1 operates with a 4MHz crystal to ensure accurate debounce and dwell



All the parts for the *High-Energy Ignition Module* go on a single PCB that fits inside a small metal diecast case (reluctor pick-up version shown). The full construction and installation details will be in Part 2 next month.

Parts List: High-Energy Ignition

1 PCB, available from the *EPE PCB Service,* code 05110121, 89mm × 53mm
1 diecast aluminium case, 111mm × 60mm × 30mm
2 cable glands to suit 3-6mm cable
1 transistor insulating bush
2 TO-220 3kV silicone insulating washers
1 4MHz HC-49 crystal (X1)
1 18-pin DIL IC socket
3 2-way pin headers, 2.54mm pitch
3 shorting links for headers
1 solder lug
1 crimp eyelet
4 6.3mm tapped nylon standoffs
8 M3 × 5mm screws
3 M3 × 10mm screws
3 M3 nuts
2 M3 star washers
9 PC stakes
1 2m length of red automotive wire
1 2m length of black automotive wire
1 2m length of green automotive wire
1 2m length of white automotive wire

Semiconductors
1 PIC16F88-E/P microcontroller programmed with 0511012A.hex (IC1)
1 ISL9V5036P3 ignition IGBT (Q1) (X-On; x-on.com.au, or eBay.co.uk)
1 LM2940CT-5 low drop out 5V regulator (REG1)

Capacitors
1 100µF 16V PC electrolytic
1 10µF 16V PC electrolytic
1 470nF MKT
3 100nF MKT
1 1nF MKT
2 22pF ceramic

Resistors (0.25W 1%)
1 100kΩ 2 2.2kΩ
1 47kΩ 2 1kΩ
2 10kΩ mini horizontal trimpots (VR1,VR2)

Miscellaneous
Angle brackets for mounting, automotive connectors, self-tapping screws, heatshrink tubing

Points version
1 100Ω 5W resistor (R1)

Reluctor version
1 BC337 NPN transistor (Q2)
1 2.2nF MKT capacitor
1 470pF ceramic capacitor
1 100kΩ top adjust multi-turn trimpot (VR3)
1 47kΩ 0.25W 1% resistor
1 10kΩ 0.25W 1% resistor
1 10kΩ 0.25W 1% resistor (R4)
1 1kΩ 0.25W 1% resistor (R3)
2 PC stakes

Hall Effect/Lumenition Module
1 1kΩ 0.25W 1% resistor (R3)
1 100Ω 0.25W 1% resistor (R2)
2 PC stakes

Optical Pick-up
1 optical pick-up (Piranha or Crane)
1 22kΩ 0.25W 1% resistor (R3 or R6)
1 120Ω 0.25W 1% resistor (R4 or R5)
2 PC stakes

settings, regardless of temperature. We recommend using the extended version of IC1 (ie, the PIC16F88-E/P) which will operate reliably up to 125°C, compared to 85°C for the industrial version (PIC16F88-I/P).

IC1 is powered from a regulated 5V supply. This is derived using REG1, an LM2940CT-5 low-dropout regulator designed specifically for automotive use. It features both transient overvoltage and input polarity protection and it provides a regulated 5V output even if its input voltage drops as low as 5.5V, eg, when starting the engine in cold weather with a partially flat battery.

REG1 has a 470nF bypass capacitor at its input and a 100µF filter capacitor at its output, both of which are required for stable operation. The input capacitor is non-polarised so that it will not be damaged if the supply polarity is inadvertently reversed.

Trigger input options

Fig.4 shows the various trigger input circuit options. We'll look at each of these in turn:

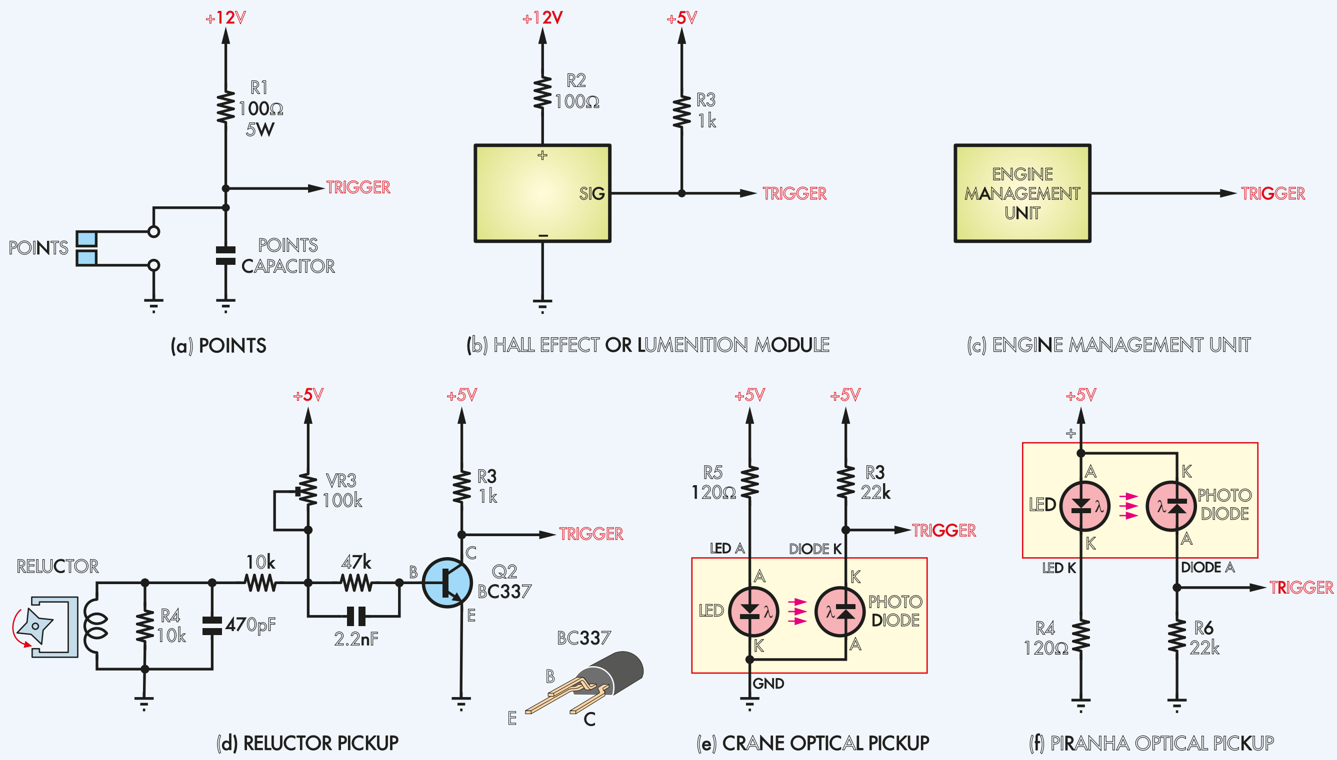


Fig.4: the six input trigger circuits: (a) points triggering; (b) Hall effect (and Lumenition) triggering; (c) triggering from an engine management module; (d) reluctor pickup; (e) Crane optical pickup; and (f) Piranha optical pickup. These different trigger options are all catered for on the PCB and it's simply a matter of installing the relevant parts.

• **Points:** Fig.4(a) shows the points input circuit. This simply comprises a 100Ω 5W resistor (R1) which connects between the top of the points and the 12V supply (the points capacitor is already present in the vehicle). This 100Ω resistor acts as a pull-up for the trigger input and it also provides a 'wetting current' to ensure that the contacts remain clean.

The points are connected between the trigger input and ground. As a result, the trigger input is pulled low each time the points close and high (via the 100Ω resistor) each time they open.

• **Hall Effect:** Fig.4(b) is for a Hall effect or Lumenition (optical trigger) sensor module. This module is powered via a 100Ω resistor (R2) from the 12V rail, to limit the current into an internal clamping diode. A 1kΩ resistor (R3) on the output is also included, to pull up the output to 5V when the internal open-collector transistor inside the sensor module is off. Conversely, the trigger output falls to nearly 0V when that transistor is on.

• **ECU:** the circuit for a vehicle with an engine management computer is shown in Fig.4(c). It's very simple – the 5V output signal from the computer simply connects to the trigger input of the ignition module.

• **Reluctor:** the reluctor input circuitry is shown in Fig.4(d). In operation, the output from the reluctor produces an AC signal, switching transistor Q2 on and off. Initially, with no reluctor output voltage, transistor Q2 is switched on via current through trimpot VR3 and the 47kΩ resistor to its base. The actual voltage applied to Q2's base depends on VR3, the two 10kΩ resistors (one across the reluctor coil) and the internal resistance of the reluctor itself.

Trimpot VR3 allows the circuit to be adjusted to suit a wide range of reluctor resistance values. In practice, VR3 is adjusted so that Q2 is just switched on when there is no signal from the reluctor. When the reluctor signal goes positive, Q2 remains switched on. Conversely, when the signal swings negative, Q2 switches off.

The signal output is taken from Q2's collector and this provides the trigger signal for the ignition module.

Resistor R4 provides the necessary load for the reluctor, and the parallel 470pF capacitor shunts very high frequency signals to ground. The 2.2nF capacitor across the 47kΩ base resistor speeds up Q2's switch-on and switch-off times.

• **Optical:** finally, Figs.4(e) and 4(f) respectively show the Crane and Piranha optical trigger pick-up circuits. The Crane trigger has a common-ground connection, while the Piranha has a common positive, but apart from that, they operate in similar fashion.

For the Crane trigger, resistor R5 limits its internal LED current from the 5V supply, while R3 pulls up the photodiode output. Similarly, for the Piranha system, R4 is the LED current-limiting resistor, while R6 pulls down the photodiode output.

All the different trigger options shown on Fig.4 are catered for on the ignition module's PCB. It's just a matter of installing the relevant parts (more on this next month).

What's coming

That's it for now. Next month, we will go through the construction, set-up and installation of the *High-Energy Electronic Ignition System.*

Mysterious ways

TechnoTalk

Mark Nelson

Solar cells are a bit of a misnomer. They function best in bright sunlight (hence the 'solar' tag) but they also operate under cloudy skies, albeit less efficiently. New research indicates they also work in completely different – and unexpected – ways, as Mark Nelson explains.

IT'S not only *the almighty* who works in mysterious ways. It transpires that some types of solar cell have secondary modes of operation that might increase their utility if the technology is developed. Two fascinating approaches are under investigation. One looks at making the most of the usually wasted non-visible radiation that falls on a solar cell. The other works by tweaking a cells' properties using vibration. Intrigued? – then read on!

Sunshine – not just sunlight!

The first is a practical means of making solar cells use thermal radiation (rather than light) from the sun to generate power. Generally, thermal radiation from the sun has no effect on most silicon solar cells. However, wavelength up-converters can be used to transform infrared radiation into usable light, and now scientists from the Fraunhofer Institute for Solar Energy Systems (ISE) (Germany), the University of Bern (Switzerland) and Heriot-Watt University in Edinburgh (Scotland) are able to harness this 'free' energy.

As everyone 'senses', even if they don't technically 'know', there is more to solar radiation than visible light. We suffer sunburn from unseen ultraviolet radiation, while the heat we feel on our skin in hot weather results from infrared radiation. Just as we see only the visible sunlight, solar cells also react only to a portion of solar radiation, limiting their potential efficiency. What the research team has achieved for the first time is to make a portion of this 'wasted' radiation usable with the assistance of a practical 'up-converter'. Although the technology that can transform infrared into usable light has been known since the 1960s, it has only been investigated for solar cells since 1996. 'We have been able to adapt both the solar cells and the up-converter so as to obtain the biggest improvement in efficiency so far,' enthuses Stefan Fischer, a scientist at ISE. The potential is great: theoretically, up-converters could increase the portion of solar radiation converted into electrical power by 40 per cent.

A ladder for light particles

Clever stuff, but exactly how does the up-converter manage to use infrared radiation with solar cells? As solar radiation falls on a cell, it absorbs the visible and near-infrared light. The infrared portion is not absorbed, however, and goes right through them. On the rear side of the cell the radiation runs into the up-converter – essentially a microcrystalline powder made of sodium yttrium fluoride embedded in a polymer or 'plastic glue'. Part of the yttrium has been replaced by the scientists with the element erbium, which is active in the optical range and responsible in the end for the up-conversion.

As the light falls on this up-converter, it excites the erbium ions. That means they are raised to a higher energy state. You can imagine this reaction like climbing up a ladder: an electron in the ion uses the energy of the light particle to climb up the first step of the ladder. A second light particle enables the electron to climb to the second step, and so on. An ion that has been excited in this manner can 'fall down' from the highest step or state. In doing so, it emits light with an energy equal to all of the light particles that have helped the electron to climb on up. The up-converter collects, so to speak, the energy of several of these infrared-powered particles and transfers it to a single one. This has so much energy that the solar cell can 'see' it and can use its energy.

Hit me with your rhythm rod

Our second fascinating approach to improving solar cells leads from the sublime to the seemingly ridiculous. But only seemingly – according to a report in *New Scientist* magazine, some solar cells convert sunlight into electricity more efficiently when loud music is played to them – as long as it is pop rather than classical. And according to the National Physical Laboratory (NPL), the high frequencies and pitch found in pop and rock music cause vibrations that enhance energy generation in solar cells containing a cluster of 'nanorods', leading to a 40 per cent increase in efficiency of the solar cells. The study has implications for improving energy generation from sunlight, particularly for the development of new, lower-cost, printed solar cells.

Dr Steve Dunn, Reader in Nanoscale Materials from Queen Mary's School of Engineering and Materials Science, said: 'At the moment, we are working very closely with the NPL, who are helping us with measuring, calibration and the traceability of our output from our devices.' Paul Weaver, Principal Research Scientist in NPL's Functional Materials Group added: 'Anything that can improve the conversion efficiency of this type of solar cell is clearly useful. It's interesting that exploiting combinations of properties in these complex, multifunctional materials can produce unexpected benefits.'

Mu-zinc power

This intriguing phenomenon will not assist the researchers working on silicon solar cells, however, as the scientists at Queen Mary University and Imperial College in London who discovered the beat music effect are working on an alternative material – zinc oxide. This is significantly cheaper than silicon and more adaptable in its application, having the ability to be made into thin, flexible films. This is a major snag, however, and as *New Scientist* reported, its efficiency is just 1.2 per cent, between a tenth and a twentieth of comparable silicon cells. This is why the researchers investigated an unusual property of zinc oxide: its ability to generate an electric field in response to mechanical stress (such as the vibrations produced by sound). Quoted in the online scientific publication *Advanced Materials*, Dr Dunn stated: 'We tried our initial tests with various types of music, including pop, rock and classical. Rock and pop were the most effective, perhaps because they have a wider range of frequencies. Using a signal generator to produce precisely measured sounds similar to ambient noise we saw a 50 per cent increase in efficiency, rising from 1.2 per cent without sound to 1.8 per cent with sound.'

Ever missed an important call because you and your mobile were separated? Here's the *EPE* solution!

Mobile Phone LOUD RINGER!

by John Clarke

You know the scene: you're working outside and the mobile is inside. Or maybe you've left it in the work vehicle while you're at a job. Either way, you pick up the phone and all you see is 'missed calls'. Rats!

Sure, they could have left a message (but many people don't like doing that). Either way, you now have to return the call (at your cost!) and the odds are it's someone trying to flog you something you don't want, someone who wanted you to do a job but has gone elsewhere in the meantime, someone seeking a donation to a worthy cause, or even a wrong number.

Whatever it is, it's an inconvenience. And an expense. Wouldn't it be nice if you could hear the phone ring 'cos it's now **REALLY LOUD?** Yes it would be!

We can't change the ring volume on your phone itself, but we can help you out by adding a mobile phone ring 'extender'.

This simple device picks up the vibration of the mobile phone ringing (and pretty well all mobile phones have this feature so you can 'feel' the phone ringing in a noisy environment) and uses that vibration to trigger a loud piezo siren that you can position wherever you like.

The vibrating alert on a mobile phone typically produces a 150-180Hz 'buzz'. It's produced by a small electric motor running at around 10,000rpm that rotates an eccentric (or off balance) mass on its shaft.

The *Mobile Phone Ring Extender* (let's call it the *MPRE* for short) is housed in a small case. The idea is that when you don't want the phone in your pocket or even close by, you place it on top of the *MPRE* case.

The *MPRE* then provides a (very!) loud phone ring alert that allows you to get to the phone and hopefully answer it before it stops ringing.

How it works

Fig.1 shows the general arrangement. A piezo transducer is the sensor that detects the vibration from the phone. The vibration from the phone is transferred to the case lid and that movement is further transferred to the piezo element via an actuator made from a short length of nylon or polycarbonate M4 thread, cut from a screw.

The signal from the piezo transducer is amplified and converted to a DC voltage. This DC voltage is monitored using a comparator that compares the voltage against a

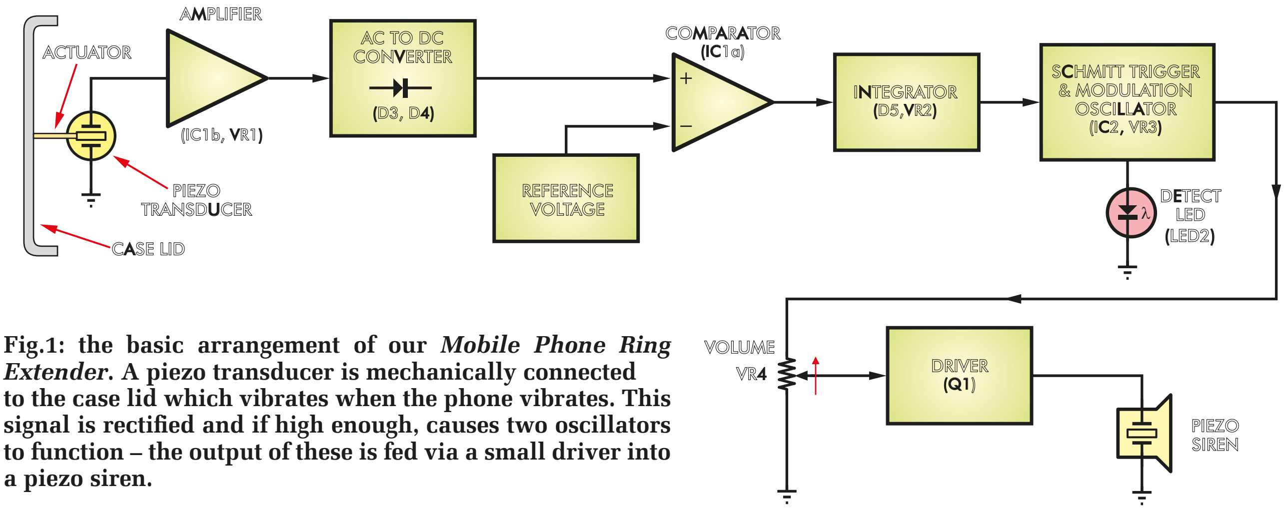


Fig.1: the basic arrangement of our *Mobile Phone Ring Extender*. A piezo transducer is mechanically connected to the case lid which vibrates when the phone vibrates. This signal is rectified and if high enough, causes two oscillators to function – the output of these is fed via a small driver into a piezo siren.

reference voltage. With no signal, the DC voltage is below the reference and the comparator output is low (near to the ground supply). With vibration detected, the DC voltage rises above the reference voltage and the comparator output goes high (towards the positive supply).

The following circuitry forms a delay circuit whereby the high level is integrated over time. This integrator is included so that brief vibrations – such as the phone bouncing to footsteps – are not sufficiently long enough to be detected by the following Schmitt trigger (IC2c).

Only longer periods of vibration that really do mean there's an incoming call are detected. The integrator delay is adjustable with VR2 so that the time period can be set correctly; that is, not too short to detect the placing of the phone onto the *MPRE* case, but not too long so as to significantly delay the detection to an incoming call.

The Schmitt trigger is a part of the modulation oscillator and starts oscillation with sufficient signal from the integrator. The detect LED driven from IC2b visually indicates the detection of an incoming call. IC2b in turn drives IC2d and IC2a respectively, and allows the modulation oscillator to switch the tone on and off. Modulation rate is adjustable using VR3.

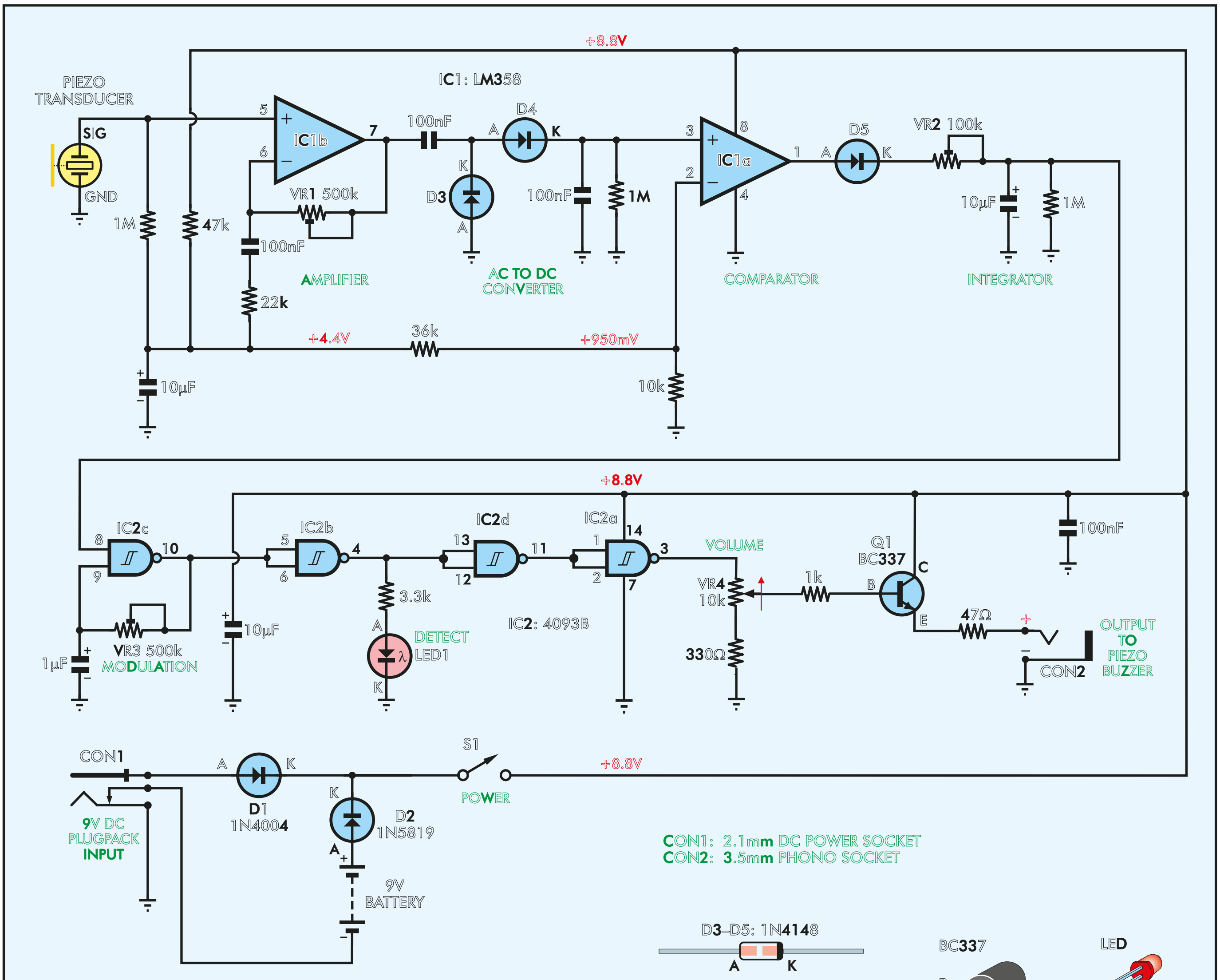
Output drive from IC2a is adjustable with VR4 and the wiper voltage is buffered with voltage follower Q1 to drive the external piezo siren.

Circuit

Circuitry for the *MPRE* is mainly based on just two ICs, a dual op amp (IC1) and a quad Schmitt trigger NAND gate (IC2). Fig.2 shows the full circuit.

IC1b is the piezo transducer amplifier. This is biased at 4.4V using a voltage divider comprising a 47kΩ, 36kΩ

Here's how it works: simply place your mobile phone on top of the *Mobile Phone Ring Extender* (with vibrate turned on). The unit detects the vibration from incoming calls and sounds the piezo siren at right. The siren shown is one of several options; more information is in the text.



MOBILE PHONE RING EXTENDER

Fig.2: the circuit is based on two low-cost ICs, a dual op amp (IC1) and a quad Schmitt trigger NAND (IC2). It can be operated from either a 9V battery or, for longer term operation, a 9V DC plugpack.

and 10kΩ resistor string connected across the 8.8V supply. Pin 5 is held at this 4.4V via the 1MΩ resistor that provides a high impedance loading for the piezo transducer.

Amplifier gain is set by the 22kΩ resistor connecting to the 4.4V reference and VR1's setting. Low frequency roll off is at 72Hz, due to the 100nF capacitor in series with the 22kΩ resistor.

The amplifier mainly amplifies signal above 72Hz and does not amplify a DC signal. Gain is variable from 1-23.7, with VR1 set between zero ohms and 500kΩ respectively.

Output from IC1b is rectified using diodes D3 and D4. The 100nF coupling capacitor at IC1b's output only allows AC signals to pass and the signal is clamped at about 0.7V below 0V by diode D3. A positive signal passes through diode D4 and is filtered with a 100nF capacitor. The 1MΩ resistor discharges the capacitor over a 100ms period.

IC1a is the comparator that monitors the filtered DC voltage at the non-inverting input (pin 3) and compares this against the 950mV reference voltage at the inverting input (pin 2). The 950mV is derived from the same voltage divider that produced the 4.4V, only from the lower tapping across the 10kΩ resistor. With the pin 3 input lower than 950mV, IC1a's output is low, at near to 0V. When the pin 3 voltage rises above 950mV, then IC1a's output goes high, at close to the 8.8V supply.

This high output from IC1a charges the 10µF capacitor via diode D5 and the resistance set by VR2. When IC1a's output is low, the 10µF capacitor discharges via the 1MΩ resistor.

When the 10µF capacitor is discharged, the pin 8 input to IC2c is low and the output at pin 10 remains high, because the NAND gate output will only go low when both the pin 8 and pin 9 inputs are high. The 1µF capacitor at the second input at pin 9 is charged to a high level via VR3. IC2b inverts this high to a low output at pin 4 and so IC2d's output also remains with its output (pin 11) high. IC2a inverts this high so that its pin 3 output sits low and there is no drive to the piezo siren.

Upon detection of a vibration signal (and the IC1a output going high), the resulting high at pin 8 of IC2c's input allows the modulation oscillator to run. IC2c's output goes

Here's what it looks like fully assembled, with the end-on view above showing the minimal controls. We used a 'remote control' case because it already has provision for an internal 9V battery (battery and output sockets are on the side of the case in purpose-cut holes).

low and discharges the 1µF capacitor via VR3, whereupon IC2c's output goes high again to recharge the 1µF capacitor via VR3. This cycle repeats.

The input threshold for the NAND gate includes hysteresis that is internally provided within IC2.

Each time IC2c's output is low, IC2b's output is high and this also drives the detect LED via a 3.3kΩ resistor. The LED switches on and off at the modulation rate. IC2d buffers the modulation signal from IC2b, while IC2a inverts the logic level again and applies the signal across the volume potentiometer (VR4). The output at the wiper drives the base of the emitter follower Q1. When the base voltage goes high, the emitter of Q1 supplies power to the piezo siren via CON2. A 330Ω resistor is included in series with the volume control (VR4) to reduce the 'dead area' at the full anticlockwise pot travel where there is no volume. The wiper voltage for VR4 needs to go above about 0.6V for Q1 to switch on.

The 330Ω resistor sets the wiper voltage at 268mV when IC2d's output is high and the wiper is set fully anticlockwise. Without this resistor, the wiper voltage would be 0V and would require more clockwise travel before sound is heard from the siren.

Power for the *MPRE* can be either from a 9V battery or a DC supply such as from a 9V plugpack. The 9V battery supply is via Schottky diode D2, providing reverse polarity connection protection for the circuit with minimal voltage drop. Whenever power is connected via the DC socket, the battery is automatically disconnected. Both supplies are isolated from each other by the diodes.

With the 0.2V drop across diode D2, the rail voltage with a fresh battery is very close to 8.8V, as shown on the circuit diagram. It is usually a little higher from a plugpack because even though branded '9V', the output from these can be (and usually is) anywhere up to about 12V at low currents. 'At rest' current consumption is about 6mA.

Construction

All components for the *MPRE* are mounted on a PCB measuring 63.5mm × 86mm, coded 12110121 which is

Why detect the vibration?

Why have we gone to the trouble of detecting the vibration of the mobile phone? Why not simply detect the ring of the phone, say via a microphone, and use that to trigger the siren?

The reason is pretty simple: there are so many ring tones, so many tunes and so many variations on a theme in mobile phones that it was difficult to create a 'one size fits all' detector; one that would work with everything.

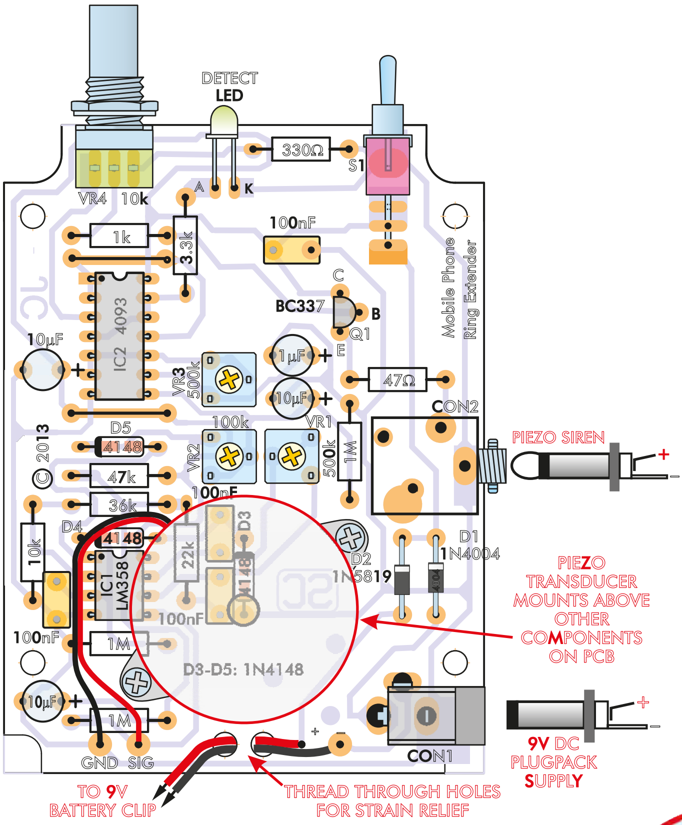
And there was a second problem: how sensitive do you make it, so that it reliably triggers with a ring but doesn't false trigger when the dog barks?

Just about everyone has their vibration alert turned on all the time – even if the ringer itself is turned off (and that was another problem!). So the vibration detector was the way to go – reliable, works with all phones, etc.

Believe it or not, we tried yet another method of detection based on RF. You know how the phone's handshaking (beep beep, beepity beep beep) gets into everything?

We thought this would be a great way to detect an incoming call, even though mobile phone transmitter power varies significantly with distance to the cell tower. But again, we couldn't make it reliable with all brands of phone – and it even had more problems detecting 3G calls than it did 2G. 4G? Don't know – none of us has a 4G phone!

So, again, we decided detecting the vibration was the best option!



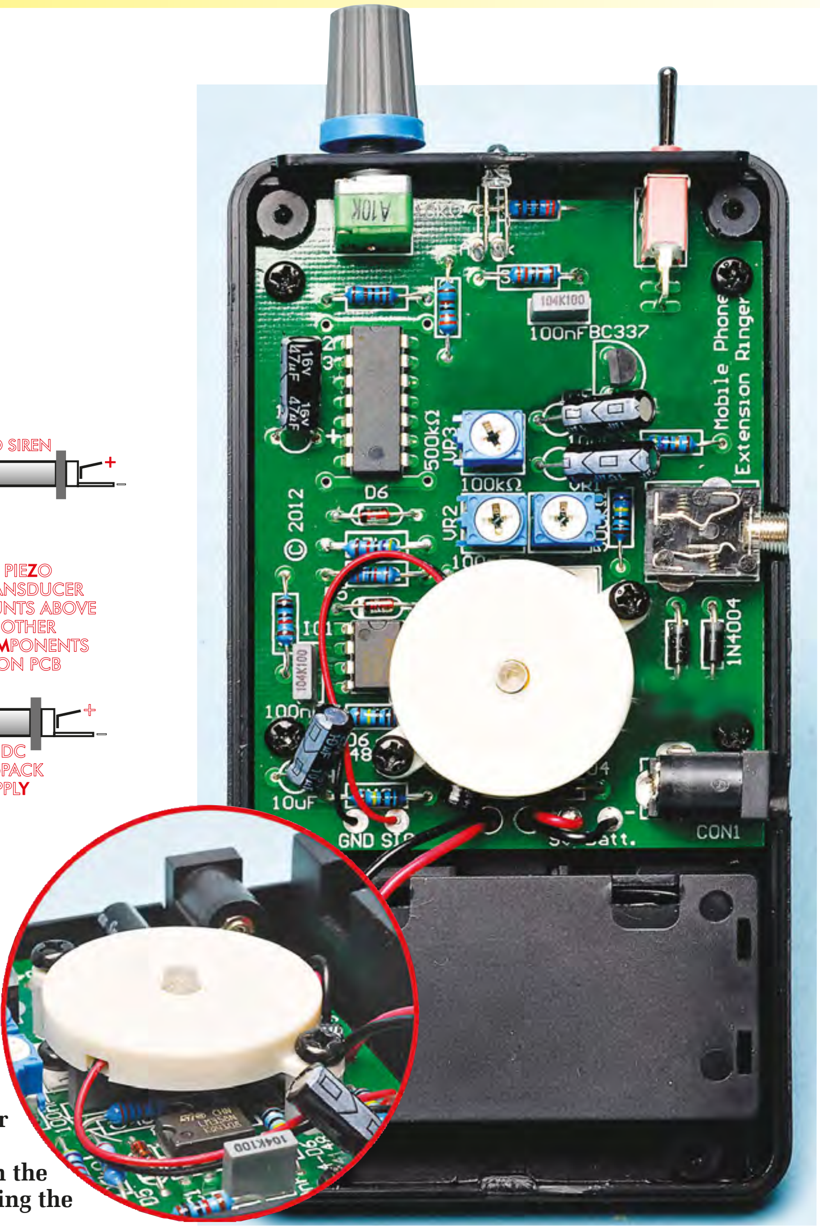


Fig.3 (above): the same-size PCB component overlay, with an early prototype photo alongside (some components have been changed since the photo was taken). While the electrolytic capacitors are shown above in traditional position, the photo at right shows that these are all 'laid over' to give room for the case lid to fit. Also, the two links shown above will only be required if the PCB is single-sided.
Inset at right is a close-up of the piezo transducer with a short length of 4mm nylon screw thread, used to provide a mechanical connection between the transducer and case lid. It simply relies on touching the lid; it is not glued in.

available from the *EPE PCB Service*. The PCB and components are housed in a 'remote control' case measuring 135mm × 70mm × 24mm. This case is used because it also houses the 9V battery. A panel label attaches to the front face of the case.

The PCB is designed to mount onto the integral mounting bushes within the box. Make sure the corner edges of the PCB are shaped to the correct outline so they fit into the box. They can be filed to shape if necessary using the PCB outline shape as a guide.

It is rare to find faults with modern PCBs, but it's worth checking the PCB for breaks or shorts between tracks or pads. Repair if necessary.

Check the hole sizes for the PCB mounting holes – they should be 3mm in diameter.

Assembly can now begin. Start by inserting the resistors and use the resistor colour code table to help in reading the resistor values. A digital multimeter can also be used to measure each value – in fact, it's a good idea because many colour bands on resistors look very similar.

If you use a PCB purchased from *EPE*, it will be double-sided, but other PCBs may be single sided and require two wire links (above and below IC2) to be soldered in.

The diodes can now be installed and being polarised, must be mounted with the orientation as shown. IC1 should be mounted directly on the PCB since there is insufficient room (height wise) for a socket once the piezo transducer is installed. IC2 can be mounted on a socket if desired.

When installing the ICs, take care to orient them correctly with the notch (or circle marking pin 1) as shown on the overlay.

Capacitors can be mounted next. The electrolytic types must be oriented with the shown polarity and far enough above the board (~6mm or so) so that they can be bent over to lie parallel, or near-parallel, to the surface. Otherwise you will not be able to put the lid on. This simply means you need to have sufficient capacitor lead length to allow each part to bend over.

Trimpots VR1-VR3 can be mounted next. They're not all the same – VR2 is 100kΩ (code 104) while the others are 500kΩ (code 504). Transistor Q1 is next to install.

CON1 (DC input socket) and CON2 (3.5mm output to siren) should be mounted right down on the PCB surface.

Potentiometer (VR4) and the PCB-mounted switch (S1) can also be fitted.

LED1 and its series 3.3kΩ resistor can be regarded as optional if you don't require a visual indication of detected ring (leaving them out will also save a little bit of power if running from a battery).

If you do fit them, mount LED1 horizontally but at a height of 6mm above the PCB. Bend its leads at 90°, 7mm back from the LED body, making sure the anode lead is to the left.

Panel holes

The side of the base of the case needs to be filed using a rat-tailed file to allow connections to both CON 1 and CON2 through the side the case. Position the PCB in the base of the case with the PCB mounting holes aligned with the mounting pillars. Mark out the socket positions and file to shape.

A similar shape is required on the lid and its position is found by placing the lid onto the base of the case (with the PCB removed) and filing out the lid half for a circular hole.

Before securing the PCB in place, drill out the small front edge panel for the potentiometer, switch and LED (if used).

Nuts for the potentiometer and 3.5mm socket are not required. The potentiometer shaft is fitted with a knob after the front edge panel is placed over the shaft, switch S1 and the LED.

Wiring

Follow the wiring diagram to make the connections from the piezo transducer to the PCB. The battery clip lead is inserted first from within the battery compartment before being looped through the strain relief holes and attaching to the PCB. Make sure the polarity is correct – the red battery clip lead as the positive lead.

Piezo transducer

The piezo transducer is raised above the PCB using two 6.3mm tapped standoffs. M3 screws secure the standoffs from the bottom and the piezo is secured with two more M3 screws into the standoffs. Note that the mounting holes in the piezo transducer mounting lugs will need to be carefully enlarged with a 3mm (or 1/8-inch) drill bit.

The wires are attached to the 'Sig' and 'GND' inputs on the PCB.

An M4 nylon screw, which provides the mechanical connection between the vibrating case lid and the piezo transducer, is cut so that you have a 4mm-long length of thread. The head of the screw is not used. This length is inserted into the centre hole of the piezo transducer.

The length of this screw thread is important. Too short and it will not make contact with the lid. Too long and the lid will not fit the case without excessive bowing.

The PCB is secured to the base of the case using four M3 × 6mm screws that screw into the integral mounting bushes in the box.

Label

The panel label for this project can be downloaded from the *EPE* website. When downloaded, you can print on to paper, sticky-backed photo paper or plastic film.

When using clear plastic film (overhead projector film) you can print the label as a mirror image so that the ink is behind the film when placed on the panel. Once the ink is dry, cut the label to size.

Glue the panel to the lid of the case with silicone sealant, contact adhesive or similar glue. Where you use the clear film, a contrasting silicone colour can be used such as white or grey to show up the printing on a black panel.

Clear silicone can be used for non-white panels since the panel will provide the contrast against the printed label.

	No.	Value	4-Band Code (5%)	5-Band Code (1%)
❑	3	1MΩ	brown black green gold	brown black black yellow brown
❑	1	47kΩ	yellow violet orange gold	yellow violet black red brown
❑	1	36kΩ	orange blue orange gold	orange blue black red brown
❑	1	22kΩ	red red orange gold	red red black red brown
❑	1	10kΩ	brown black orange gold	brown black black red brown
❑	1	3.3kΩ	orange orange red gold	orange orange black brown brown
❑	1	1kΩ	brown black red gold	brown black black brown brown
❑	1	330Ω	orange orange brown gold	orange orange black black brown
❑	1	47Ω	yellow violet black gold	yellow violet black gold brown

Parts List
Mobile Phone Ring Extender

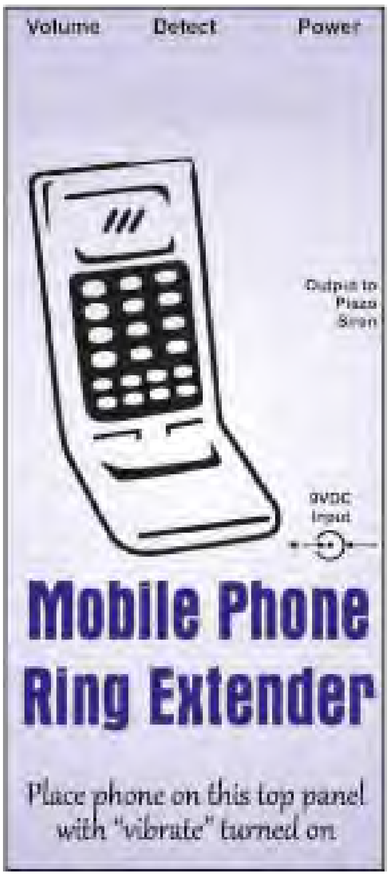
1 PCB available from the *EPE PCB Service,* code 12110121, 63.5mm × 86mm
1 panel label 113mm × 50mm
1 remote control case 135mm × 70mm × 24mm
1 high output piezo siren
1 piezo transducer
1 PCB mount SPDT switch (S1)
1 PCB mount DC socket (CON1)
1 3.5mm stereo PCB mount socket (CON2)
1 3.5mm mono plug [to connect siren]
1 knob to suit VR4
1 9V battery, with clip
8 M3 × 5mm screws
2 6.3mm-long M3 tapped nylon spacers
1 M4 polycarbonate or nylon screw (cut for a 4mm thread section without the head)
4 PC stakes (optional at wiring points)
Suitable length polarised figure-8 cable if siren is to be remotely mounted

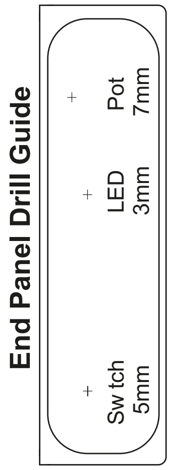
Semiconductors
1 LM358 dual op amp (IC1)
1 4093 CMOS quad Schmitt NAND gate (IC2)
1 1N4004 1A diode (D1)
1 1N5819 Schottky diode (D2)
3 1N4148 switching diodes (D3-D5)
1 BC337 NPN transistor (Q1)
1 3mm high intensity LED (LED1)*

Capacitors
3 10µF 16V PC electrolytic
1 1µF 16V PC electrolytic
4 100nF MKT polyester (code: 104)

Resistors (0.25W 1%)
3 1MΩ 1 47kΩ 1 36kΩ
1 22kΩ 1 10kΩ 1 3.3kΩ*
1 1kΩ 1 330Ω 1 47Ω
1 100kΩ horizontal mount trimpot (code 104) (VR2)
2 500kΩ horizontal mount trimpots (code 504) (VR1,VR3)
1 10k log 9mm potentiometer (Jaycar RP-8610 or equivalent) (VR5)

* optional (see text)





Figs. 4 and 5: front panel artwork for the *Mobile Phone Ring Extender*, and below, the template for drilling the end (top) panel holes. These images can also be downloaded from the *EPE* website

Piezo siren

The piezo siren is pretty loud – you might find it loud enough to mount close to the *MPRE* via the short length of cable it comes with.

Or you might prefer to mount the siren elsewhere, where it can be more easily heard – outside, for example. In this case, you'll need to connect a suitable length of figure-8 polarised cable to the siren and in either case, you'll need to solder on a 3.5mm mono plug so that it can plug into the *MPRE*. The siren is polarised – the red (+) wire goes to the centre pin on the 3.5mm plug.

If you do decide to mount the siren outside, you will need to fit it so it's protected from the elements – under an eave, for example.

Within reason, there is no limit (say to a standard suburban house boundary) to the length of wire between the piezo siren and *MPRE*.

Testing

Initially, don't connect the siren – it will deafen you at close range!

When you switch on, using a 9V battery, there should be around 8.8V between pins 4 and 8 of IC1 and between pins 7 and 14 of IC2. Pin 2 of IC1 should be about 950mV above 0V (the GND terminal). Pin 5 of IC1 should be about 4.4V above 0V.

Note that these voltages might differ a little from these values depending on the supply voltage. With a plugpack supply they will almost certainly be higher, but still should be in much the same ratio.

Now set VR1, VR2 and VR3 to mid position and attach the piezo siren. You should be able to trigger the *MPRE* into operation by repetitively tapping the case. That should introduce sufficient vibration to be detected by the piezo transducer and you should hear the alert sound and see the detect LED flash.

You can adjust VR3 and VR4 for the desired sound, with VR3 adjusting the rate of switching the tone on and off. Clockwise will increase the frequency.

Final testing is done with a mobile telephone (set to vibrate). Place the phone on top of the *MPRE* and make a call to the phone using another phone.

Again, the *MPRE* should begin flashing the detect LED and the siren should sound. If neither happens, adjust VR1 further clockwise for more sensitivity.

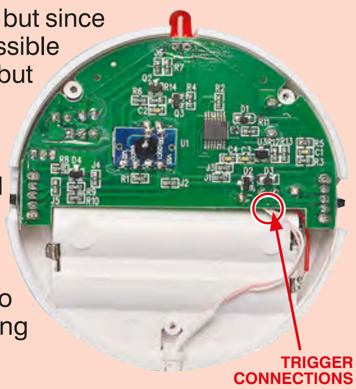
VR2 should be adjusted so that the *MPRE* does not sound unless there is an incoming call.

It should not detect a single tap on the case with your finger nail. VR2's adjustment is usually at mid-setting, but may need to be set more anticlockwise to ensure that an incoming call is detected without too much delay or more clockwise to prevent detection of single tapping on the case.

Reproduced by arrangement
with SILICON CHIP
magazine 2014.
www.siliconchip.com.au

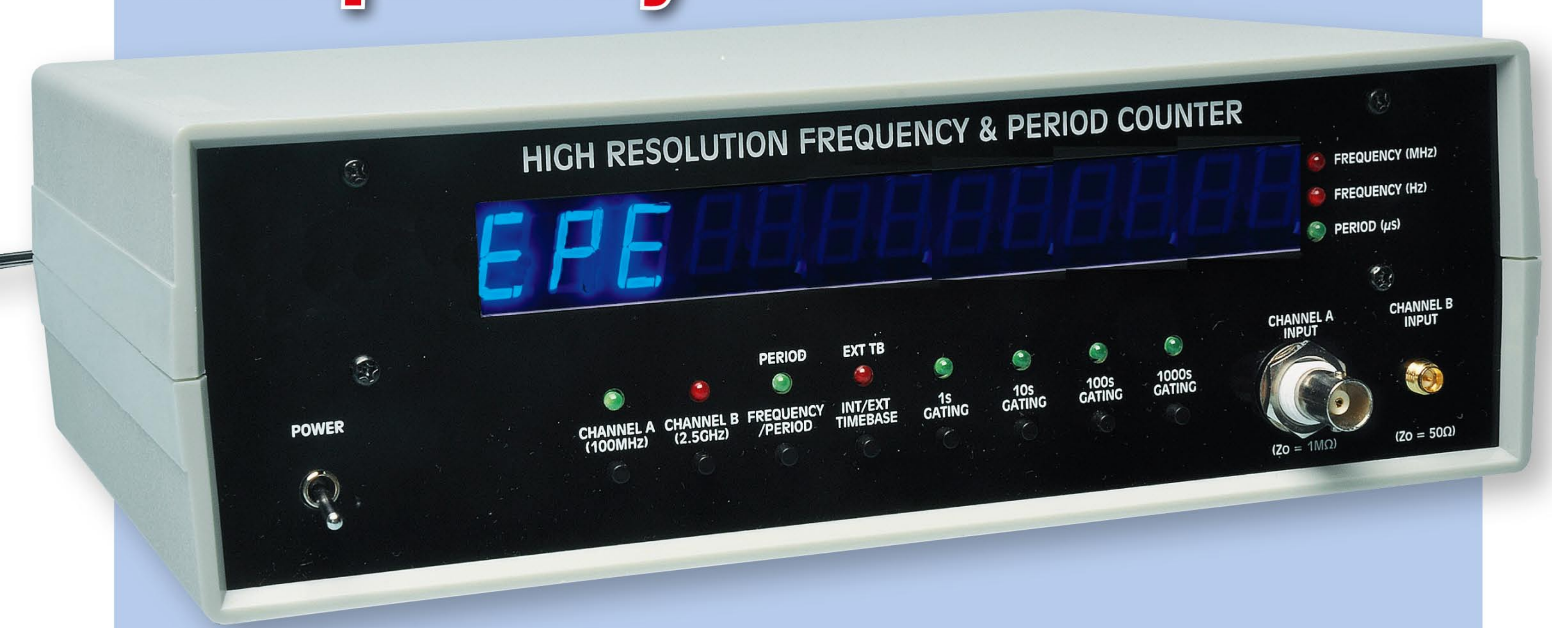
Another 'siren'?

Two piezo sirens have been used for this project, but since building the prototype, we've come across a possible alternative – albeit not made for the purpose but nevertheless we believe would be quite suitable. Jaycar Electronics offers a 'Water Leakage Alarm' (Cat LA-5163) – cheaper than either of the piezo sirens. We aren't particularly interested in detecting water, but we were interested in the alarm side. On opening the case, we found it very easy to connect to the two solder pads which trigger the alarm. The photo shows the two pads to connect to (clip off or unsolder the existing water sensor wires).



Part 2: By JIM ROWE

Building the . . .
2.5GHz 12-digit frequency counter



Last month, we explained how our new 12-digit 2.5GHz counter works and gave the full circuit details. This month, we describe the construction and detail the simple setting up procedure.

MOST OF THE CONSTRUCTION work involves building the two PCB assemblies used in the counter. Before building these boards, however, note that we've made a few changes to the circuit published last month in order to achieve optimum performance. This involves a few component value changes, plus the addition of two extra components.

The details of these changes are given in the panel titled 'Circuit Changes' (towards the end of this article). *This panel should be read carefully before beginning construction.*

As mentioned in Part 1, most of the parts are fitted on a 225 × 175mm main PCB which mounts in the lower half of the case. The rest of the components are for the multiplexed display and these go on a second 200 × 50mm display PCB mounted vertically behind the front panel. The two boards are linked by a short 20-way ribbon cable fitted with IDC connectors.

Display PCB

Start the assembly by building the display PCB – see Fig.6. This carries the 7-segment displays, the LEDs, IC25 and IC26 and all the resistors and capacitors on the front. The other side carries 23 SMD MOSFETs (Q8-Q30) plus connector CON6, which is the header for the IDC cable.

The 23 SMD MOSFETs on the rear of the PCB should be installed first. Note that you must use 2N7002 N-channel devices for Q8-Q22 and NX2301P P-channel MOSFETs for Q23-Q30. Don't mix them up, or you'll get some very strange results!

You'll need a soldering iron with a fine tip, a pair of tweezers and preferably a magnifying lamp for this job. Start by applying some solder to one of the PCB pads and then, using the tweezers, slide the device into place while heating the solder on the pad. That done, check that the part is correctly positioned. If not, reheat the solder and gently nudge it into place.

The remaining two leads can then be soldered. If you get a solder bridge, simply use desoldering braid (or solder wick) to clean it up. A dab of no-clean flux paste on the bridge beforehand will make it disappear a lot more quickly and easily.

Repeat this procedure for all 23 SMD MOSFETs, taking care to fit the correct type at each location. Once these are in place, fit CON6 then turn the PCB

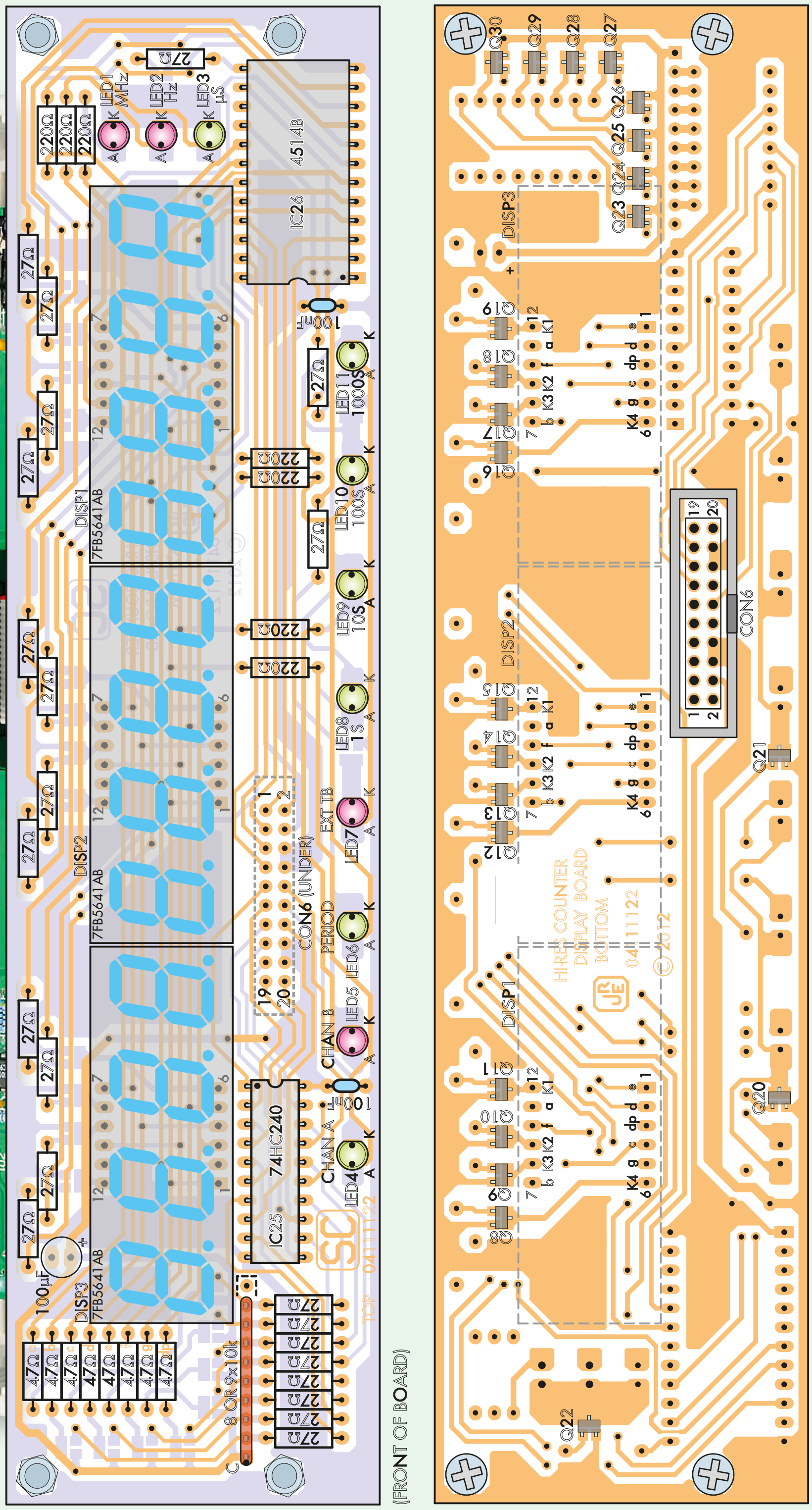


Fig.6: follow these parts layout diagrams to build the display PCB, starting with the SMD transistors (Q8-Q30) and IDC connector CON6 on the back of the board. Take care with the orientation of the LED readouts and note that IC25 and IC26 face in opposite directions.

over and solder its pins to the pads on the front. Make sure that CON6 sits flush against the PCB before soldering.

The next step is to install the resistors on the front of the PCB, followed by the 8 x 10kΩ SIL resistor array. Be sure to fit the latter with its common (C) pin towards the lefthand edge of the PCB. The two 100nF MMC capacitors and the 100µF electrolytic can then go in, followed by the three 4-digit LED displays, LEDs 1-11 and ICs 25 and 26. Take care with the orientation of these devices (note: the displays must be oriented with the decimal points at lower right).

The LEDs must be mounted with their bodies 10mm above the PCB, so that they will later protrude through their corresponding holes in the front panel. This can be done by sliding a 10mm-high strip of cardboard between the leads when soldering each device into position.

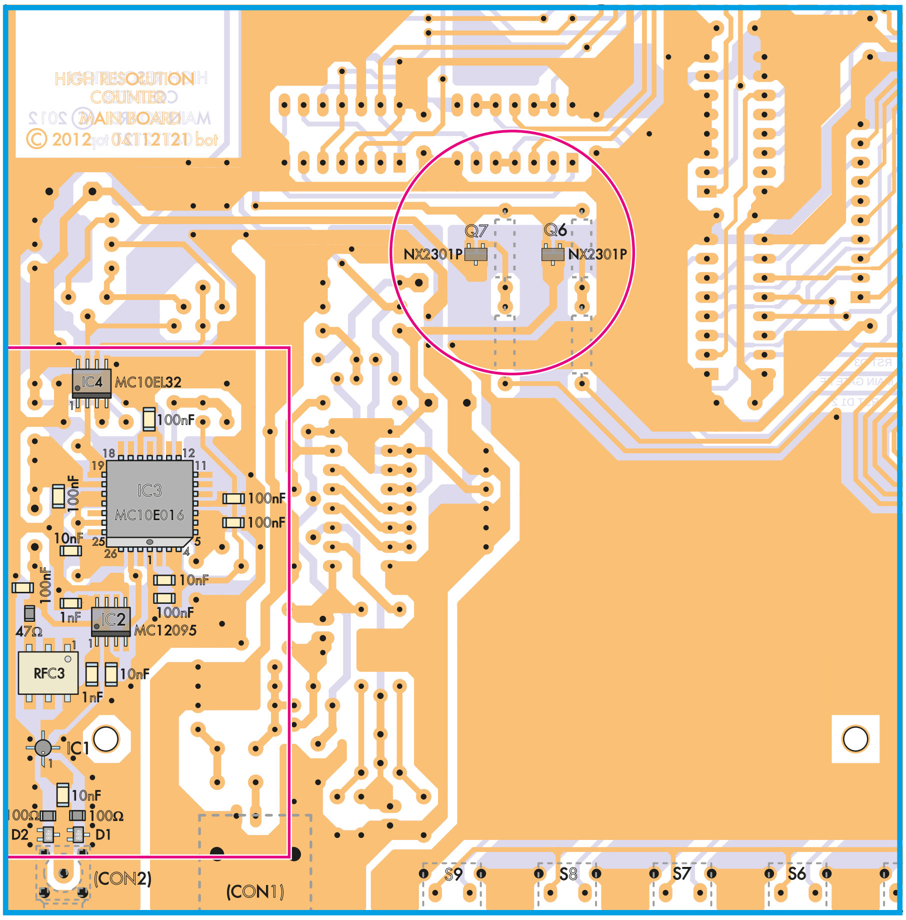
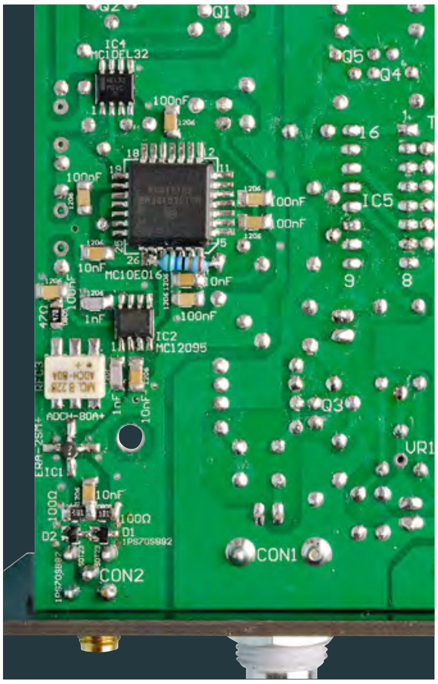


Fig.7: follow this layout diagram to mount the SMD parts on the rear of the main PCB (ie, the parts inside the red box and the red circle). Take care to ensure that the transistors, ICs and RFC3 are correctly oriented and note the bevelled end on one of IC1's leads, indicating pin 1.



The view shows the SMD parts inside the red box of Fig.7. Ignore the resistor just below IC3; this is a modification to our prototype and the final PCB now has this on the top side.

When all of these components are in place, the display PCB can be placed aside while you work on the main PCB.

Main PCB assembly

Like the display PCB, the main PCB has all of the SMD components mounted on one side (ie, the underside) and all of the leaded components on the top side. Figs.7 and 8 show the assembly details.

As before, it's best to install the SMD parts on the underside first. Most of these SMDs are located inside the red rectangle at lower left. The only exceptions are transistors Q6 and Q7, which are located inside the red circle nearer the centre of the board – see Fig.7.

Take particular care when installing IC1, IC2, IC3 and IC4 to ensure that they are correctly oriented. In each case, pin 1 is identified by an adjacent 'dot' in the plastic body of the device. IC3 also has a bevelled corner, while IC1 has a bevelled end on one lead.

Once again, you'll need a soldering iron with a very fine tip, preferably one that is earthed to avoid electrostatic damage when you are soldering in the active components (IC1-IC4, Q6 and Q7 and D1 and D2). The iron should also be temperature controlled, so you can make joints quickly without risking heat damage to the component.

The soldering procedure is much the same as for the SMD MOSFETs on the display PCB. In each case, you can hold the device in place using tweezers (or even a wooden toothpick) while you make the first solder joint, after which the SMD should stay in position while you solder the remaining leads. However, if the device has eight or more leads, then it's best to solder two diagonally opposite leads initially. The device can then be checked for correct alignment before the remaining leads are soldered.

Don't worry too much if you accidentally bridge two or more of the leads of IC2, IC3 or IC4 together when you're soldering them in place. In fact, this is almost inevitable. As before, it's simply a matter of using the tip of the iron to push some fine desoldering braid against the bridged leads. The braid will 'suck up' the excess solder forming the bridge, while still leaving the leads soldered to the PCB pads underneath.

By the way, don't forget to install RFC3, which is also an SMD component. It's fitted in the same way as the SMD ICs and must also be correctly oriented (ie, pin 1 at top right).

When you have soldered all of the SMDs in place (including Q6 and Q7), the PCB can be flipped over and the leaded components installed. Fig.8 shows the details.

Begin by installing the low-profile parts first, such as the resistors, diodes and Zener diodes. The 10 PC stakes can then be installed at the test points (TP).

Table 1: Capacitor Codes

Value	μF Value	IEC Code	EIA Code
470nF	0.47μF	470n	474
100nF	0.1μF	100n	104
10nF	0.01μF	10n	103
1nF	0.001μF	1n	102
47pF	NA	47p	47
39pF	NA	39p	39
27pF	NA	27p	27

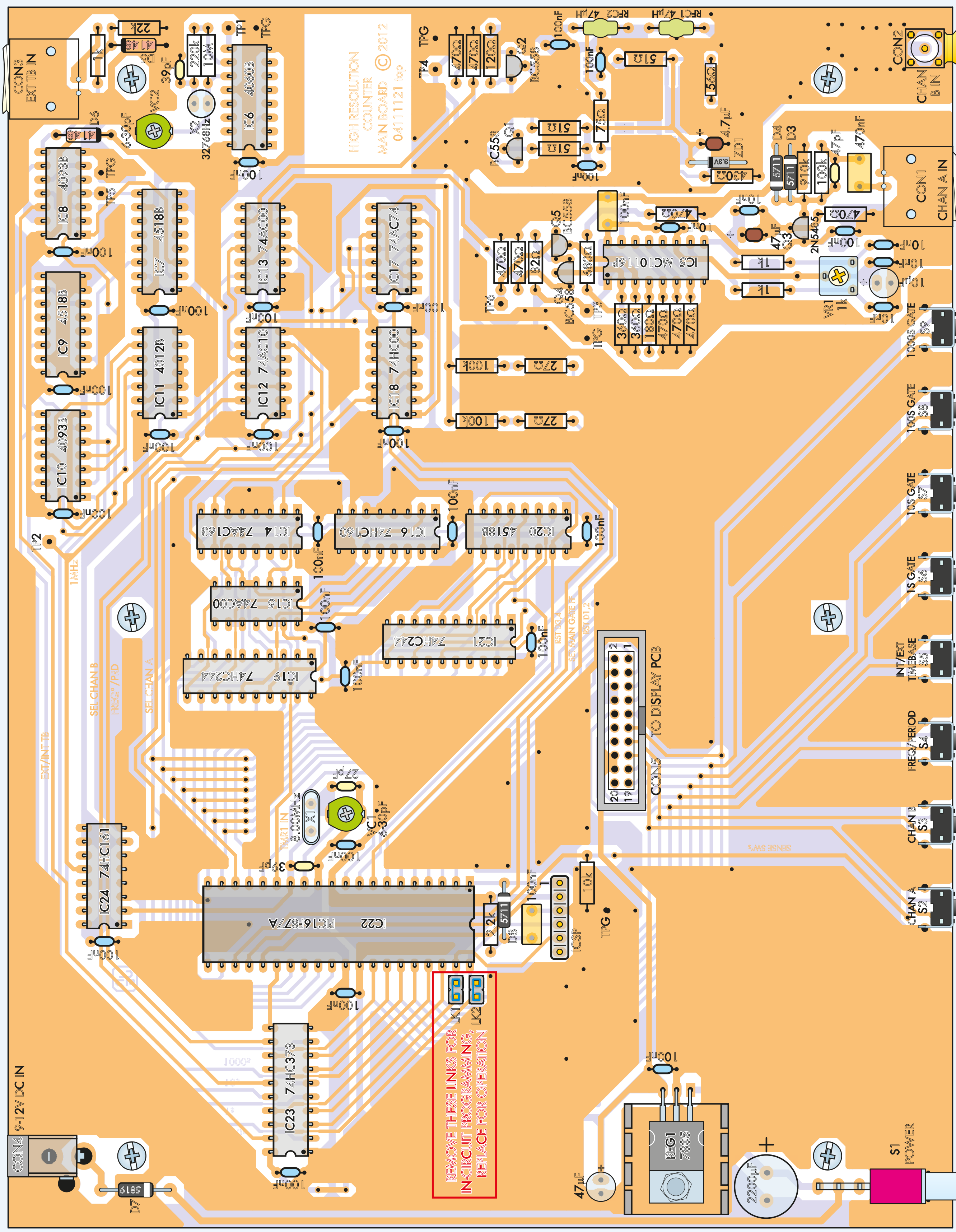


Fig.8: the parts on the top side of the main PCB should be installed only after the SMD parts have been installed on the underside. Be sure to fit the correct IC at each location and make sure that all polarised parts are correctly oriented.

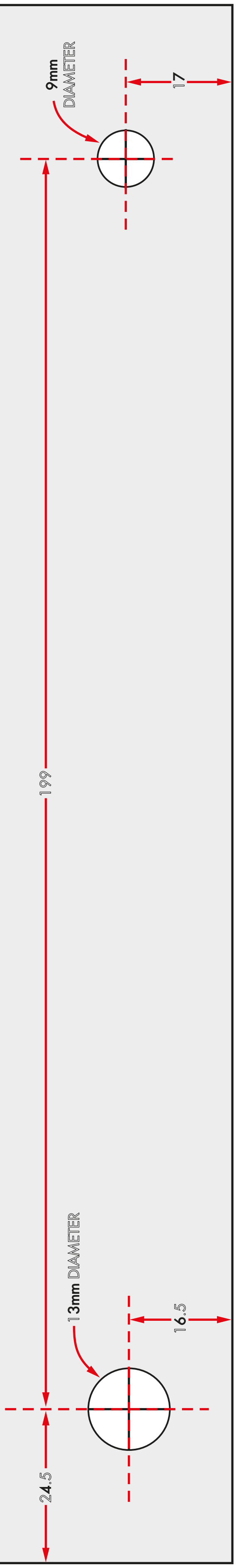


Fig.9: this drilling template for the rear panel can either be copied or downloaded in PDF format from the *EPE* website and printed out. Start each hole using a small pilot drill, then carefully enlarge it to the correct size using a tapered reamer.

TP2 is required for calibration, while the remaining test points will make it much easier to troubleshoot the unit later on if necessary.

Trimpot VR1 can now go in (bottom right), followed by RF chokes RFC1 and RFC2 and then the non-polarised capacitors (including trimcaps VC1 and VC2). Follow with the polarised electrolytic and tantalum capacitors, taking care to fit them with the correct polarity.

Next, install connectors CON1-CON4, then pushbutton switches S2-S9, power switch S1, the 6-pin ICSP header and the two 2-pin headers for LK1 and LK2. CON5, the 20-pin DIL connector, can then be fitted.

Crystals X1 and X2 are next. X1 is located just to the right of IC22 (the PIC micro), while X2 is located just below VC2 at upper right. They can be fitted either way around.

Once, they're in, install transistors Q1-Q5 and regulator REG1 (7805). The latter is mounted horizontally on the top of the PCB, with a 19mm-square finned heatsink underneath to help dissipate heat. To install REG1, first bend its leads down through 90° 6mm from its body, then secure both it and the heatsink to the PCB using an M3 × 6mm machine screw and nut. Do the nut up tightly before soldering the regulator's leads to the PCB.

The main PCB assembly can now be completed by installing all the DIL ICs, beginning with IC5 at lower right and then working through them in roughly numerical order until you reach IC24 at upper left. Be sure to select the correct IC for each position and make sure it is oriented correctly and sits all the way down on the PCB before soldering its pins.

By the way, apart from the PIC micro, it's NOT a good idea to use DIL sockets for any of the ICs, in the interests of performance and long-term reliability. Instead, they should all be soldered directly to the PCB pads, like the SMD components underneath. Just take the usual precautions to avoid damage from overheating or electrostatic discharge. Use an earthed iron tip and always solder each IC's Vss and Vdd pins (usually 7 and 14, or 8 and 16, or 10 and 20 etc) first.

As stated, the exception is the PIC micro and this should be installed in a 40-pin IC socket. That way, if you ever have to change it, it will be easy to remove.

Once all the ICs are in place, the next step is to fit links LK1 and LK2 provided your PIC has already been programmed (eg, if purchased from the Silicon Chip partshop). If not, you will first have to program it via the ICSP connector (ie, with LK1 and LK2 out).

The easiest way to do this is to use a Microchip PICkit3 programmer, driven from one of your PC's USB ports and using Microchip's MPLAB IDE (v8.85 or later) to control the programming operation. The firmware file required, 0411112A.hex, is available for free download from the *EPE* website.

Once you have programmed the PIC, be sure to fit the jumper shunts to LK1 and LK2, so that the PIC will be able to control the counter properly when it's fully assembled and powered up.

Front and rear panels

A standard plastic instrument case measuring 256mm × 189mm × 83mm is used to house the assembled PCBs. A separate PCB (code 04112123) is used for the front panel. This is available from the *EPE PCB Service* and is supplied with all holes drilled (including the display cutout) and screened lettering for the labels. This not only saves you from having to accurately drill the 26 holes and make the display cut-out yourself, but also gives a professional finish.

By contrast, the rear panel has just two holes and these can be drilled using the template shown in Fig.9. This template can either be copied direct from the magazine or it can be downloaded

This is the view inside the completed 2.5GHz frequency counter, from the rear. All the parts fit on the main and display PCBs and these are linked together by a short IDC cable (see Figs.10 and 11). Power comes from a 9-12V DC 1A plugpack supply.

in PDF format from the *EPE* website and printed out. Use a small pilot drill to start the holes, then carefully ream them to size using a tapered reamer.

Once these holes have been made, download and print out the rear-panel label from the *EPE* website. This can then be laminated and attached to the rear panel using silicone adhesive, after which the holes can be cut out using a sharp hobby knife.

The next step is to fit a 160 × 30mm piece of 1mm-thick clear acrylic sheet over the front-panel display cut-out. This should be fitted from the rear and can be held in place by applying a thin smear of silicone sealant around the outside edge before attaching it to the panel.

The front and rear panels can now be attached to the main PCB. That's done by first removing the mounting nuts from CON1 and CON3. Each panel is then fitted in place and the connector mounting nuts re-fitted. The rear panel simply fits over CON3, while the front panel not only fits over CON1 but also over CON2 and switches S1-S9.

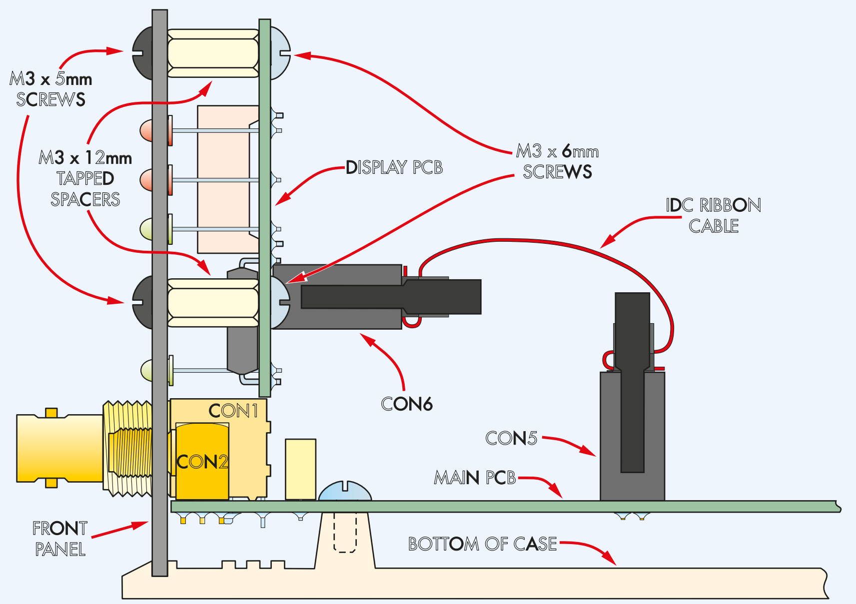


Fig.10: the mounting details for the two PCBs. The display PCB is attached to the front panel using M3 × 12mm tapped spacers and eight M3 screws (note the different lengths for the screws at each end of the spacers). This PCB assembly is then connected to the main PCB via a 20-way IDC cable (see Fig.11 for details).

Parts List – Frequency Counter

1 front panel PCB, available from the *EPE PCB Service,* coded 04111123
1 plastic instrument case, 256mm ×189mm × 83mm
1 9-12V DC 1A plugpack
1 95mm length of 20-way IDC ribbon cable
2 20-way IDC line sockets
6 4G × 6mm self-tapping screws
4 rubber self-adhesive feet
1 160 × 30mm clear acrylic sheet (1mm thick)

Main PCB

1 PCB, available from the *EPE PCB Service,* coded 04111121, size 225mm × 175mm
1 8.00MHz crystal, HC-49 (X1)
1 32.768kHz crystal, DT38 case (X2)
2 47µH RF chokes, axial type (RFC1,RFC2)
1 wideband SHF choke, SMD type (Mini-Circuits ADCH-80A+) (RFC3)*
1 40-pin 0.6-inch DIL IC socket
1 small finned TO-220 heatsink, 19mm square
1 SPDT mini toggle switch, 90° PCB-mount (S1)
8 SPST micro tactile switches, 90° PCB-mount with 6mm or longer actuators (S2-S9)
2 BNC sockets, PCB-mount (CON1, CON3)
1 SMA reverse PCB socket, 90° (CON2) F
1 2.5mm concentric DC socket, PCB-mount (CON4)
1 6-pin SIL header (ICSP connector)
1 20-way IDC locking header (CON5)
2 2-pin SIL headers
2 2-way jumper shunts
10 1mm-diameter PCB stakes
1 M3 × 6mm machine screw
1 M3 nut
1 1kΩ mini horizontal trimpot (VR1)

Semiconductors

1 ERA-2SM+ wideband monolithic amplifier (IC1)*
1 MC12095 ECL microwave frequency divider, SOIC (IC2) LD
1 MC10E016 ECL synchronous mod-16 UHF counter, PLCC (IC3) LD
1 MC10EL32 ECL UHF flip-flop, SOIC (IC4) F
1 MC10116P ECL triple line receiver (IC5) LD
1 4060B CMOS counter (IC6)
3 4518B CMOS dual BCD counters (IC7,IC9,IC20)
2 4093B CMOS quad Schmitt NANDs (IC8,IC10)
1 4012B CMOS dual 4-input NAND (IC11)
1 74AC10 high-speed CMOS triple 3-input NAND (IC12) F
2 74AC00 high-speed CMOS quad 2-input NANDs (IC13, IC15) F
1 74AC163 high-speed CMOS synchronous mod-16 counter (IC14)
1 74HC160 CMOS synchronous BCD decade counter (IC16)
1 74AC74 high speed dual D-type flip-flop (IC17) F
1 74HC00 CMOS quad 2-input NAND (IC18)
2 74HC244 CMOS octal buffer/line drivers (IC19,IC21)
1 PIC16F877A MCU programmed with 0411112A.hex (IC22)
1 74HC373 CMOS octal latch (IC23)
1 74HC161 CMOS synchronous mod-16 counter (IC24)
1 7805 +5V regulator (REG1)
4 BC558 PNP transistors (Q1,Q2, Q4,Q5)
1 2N5485 VHF junction FET (Q3) LD
2 NX2301P P-channel MOSFETs (Q6,Q7) F
1 3.3V 1W Zener diode (ZD1)
2 1PS70SB82 very high speed Schottky diodes (D1,D2) F
3 1N5711 Schottky diodes (D3,D4, D8) F
2 1N4148 signal diodes (D5,D6)
1 1N5819 1A Schottky diode (D7)

Capacitors

1 2200µF 25V RB electrolytic
1 47µF 16V tantalum
1 47µF 16V RB electrolytic
1 10µF 16V RB electrolytic
1 4.7µF 25V tantalum
1 470nF MKT
2 100nF MKT
6 100nF X7R MMC 1206 SMD F
25 100nF through-hole MMC
4 10nF X7R MMC 1206 SMD F
5 10nF through hole MMC
2 1nF C0G MMC 1206 SMD F
2 47pF NP0 disc ceramic
1 39pF NP0 disc ceramic
1 27pF NP0 disc ceramic
2 6-30pF mylar mini trimcaps (VC1,VC2)

Resistors (0.25W, 1%)

1 10MΩ	2 360Ω
1 910kΩ	1 180Ω
1 220kΩ	1 120Ω
3 100kΩ	2 100Ω 0805 SMD F
1 22kΩ	1 82Ω
1 10kΩ	1 75Ω
1 2.2kΩ	1 56Ω
3 1kΩ	3 51Ω
1 680Ω	1 47Ω 0805 SMD F
9 470Ω	2 27Ω
1 430Ω	

Display PCB

1 PCB, available from the *EPE PCB Service,* coded 04111122, size 200mm × 50mm
1 20-way IDC locking header (CON6)
4 M3 × 12mm tapped Nylon spacers
4 M3 × 6mm machine screws
4 M3 × 5mm machine screws

Semiconductors

3 7FB5641AB quad 7-segment blue LED displays (DISP1, DISP2,DISP3) (Futurlec)*
1 74HC240 CMOS octal buffer/line driver (IC25)
1 4514B CMOS 1-16 latching decoder (IC26)
15 2N7002 N-channel MOSFETs (Q8-Q22) F
8 NX2301P P-channel MOSFETs (Q23-Q30) F
4 3mm red LEDs (LED1, LED2, LED5, LED7)
7 3mm green LEDs (LED3, LED4, LED6, LEDs8-11)

Capacitors

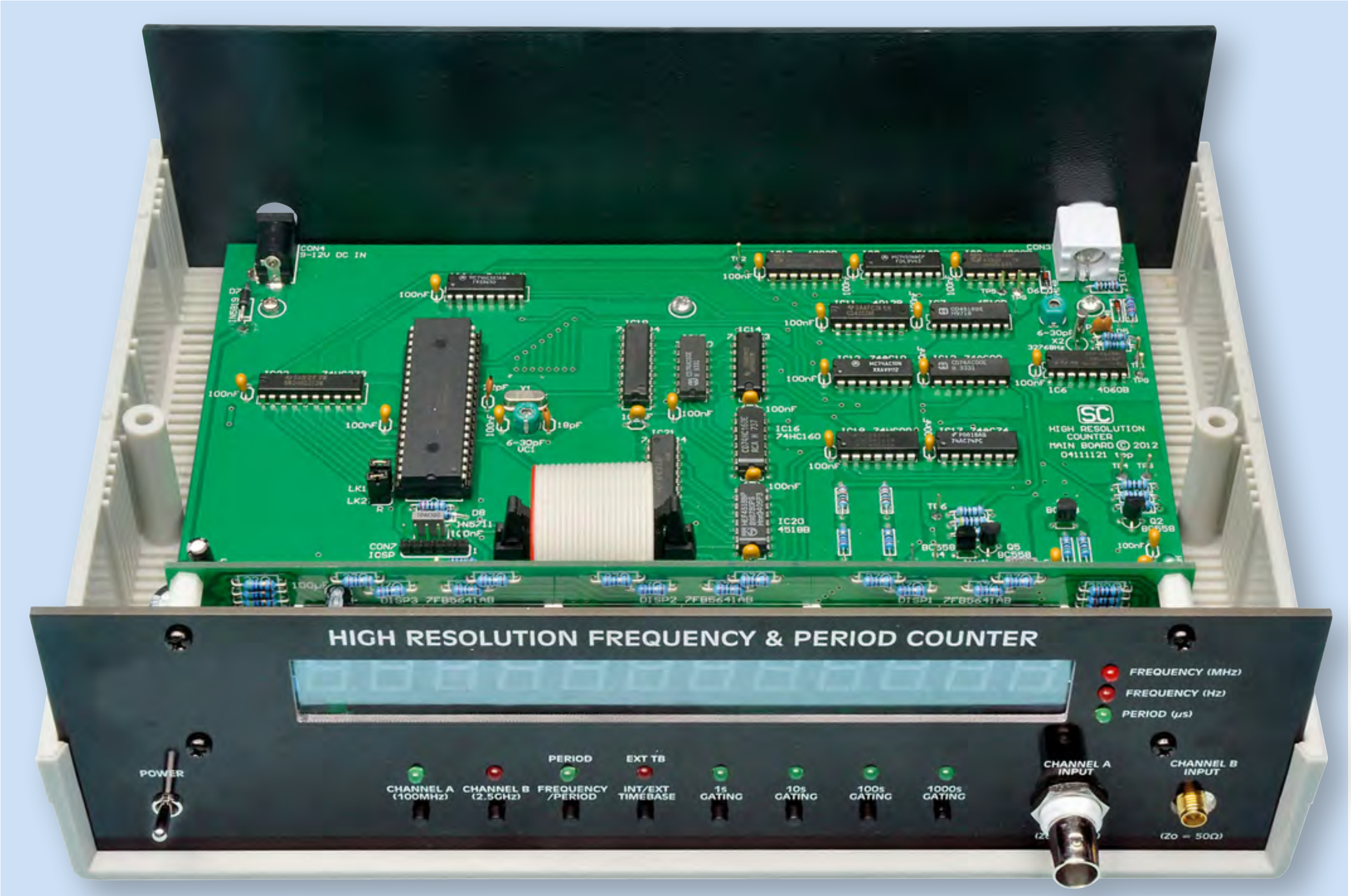
1 100µF 16V RB electrolytic
2 100nF through-hole MMC

Resistors (0.25W, 1%)

7 220Ω 8 47Ω 23 27Ω
1 10kΩ SIL resistor array, 8× or 9× F

The programmed PIC microcontroller and parts marked with a red asterisk (*) are available from the SILICON CHIP Partshop: www.siliconchip.com.au/shop. Parts marked with an F are available from Farnell. Parts marked LD are available from: www.littlediode.com

Note: Do ensure all parts are available before commencing project



Another view of the completed frequency counter, this time from the front. The main PCB assembly is attached to the base of the case using six self-tapping screws that go into integral moulded pillars.

Do the mounting nuts up finger-tight to hold things together, then lower the entire assembly into the bottom half of the case, with the panels slipping down inside the vertical moulded channels. Once it's in place, the assembly can be fastened down using six 4G × 6mm self-tapping screws which go into integral pillars moulded into the case bottom with the connector nuts firmly tightened.

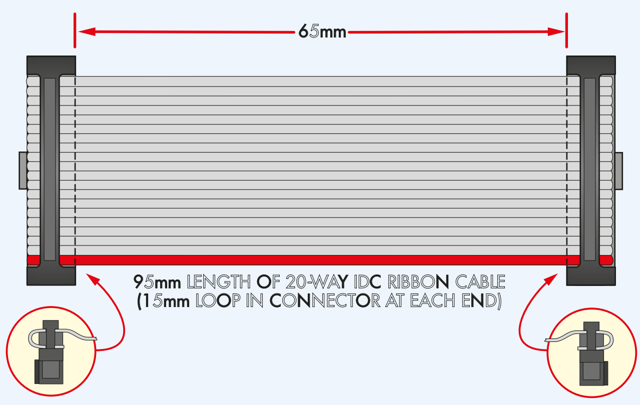


Fig.11: follow this diagram to make up the 20-way IDC cable. The cable is then used to link the main and display PCBs, as shown in Fig.10.

Mounting the display PCB

With the main assembly in place, you're now ready to mount the display PCB. This is attached to the rear of the front panel using four M3 × 12mm tapped spacers and eight machine screws – see Fig.10. Note that M3 × 5mm screws are used to secure the spacers to the front panel, while M3 × 6mm screws are used to secure the display PCB to these spacers.

The counter assembly can now be completed by making up and fitting the short IDC ribbon cable which links the two PCBs via CON5 and CON6.

Fig.11 shows how the IDC cable is made up. All you need is a 95mm length of 20-way flat ribbon cable plus the two 20-way IDC connectors. A small bench vice can be used to clamp the connectors in position if you don't have an IDC crimping tool.

Don't forget to fit the locking bar to each connector after crimping. The completed IDC cable is then used to link the two PCBs, as shown in Fig.10.

Set-up and adjustment

With the cable in place, the counter can now be powered up and adjusted.

Begin by connecting a 9-12V 1A DC plugpack supply to CON4 and then switch on using power-switch S1. You should immediately be greeted with the message 'SILICON CHIP' on the 7-segment LED displays. In addition, the following indicator LEDs should light: LED4 (CHANNEL A),

Circuit changes

Since publication of Part 1 last month, we've changed a few component values and added two extra components, in order to achieve the highest possible performance from the counter. The details are as follows:

1) The capacitor in parallel with the 100kΩ resistor in the channel A input divider has been increased from 22pF to 47pF.
2) The 100µF electrolytic capacitor in parallel with the 10nF capacitor at Q3's source has been changed to a 47µF tantalum.
3) The resistors connecting pins 2 and 3 of IC5a to ground have been changed from 470Ω to 360Ω.
4) The resistor between pins 15 and 4 of IC5 has been changed from 1kΩ to 680Ω.
5) A 100nF MKT capacitor has been added as a supply bypass between pins 1 and 16 of IC5 and ground.
6) A 56Ω resistor has been added between pin 4 of IC2 (and pin 27 of IC3) and the +3V (V_L) supply line of the channel B prescaler. This is not shown in the photo of the prototype, but is included in the final version of the PCB.
7) To make it easier for the PIC's clock oscillator to be adjusted to exactly 8.000MHz with almost any crystal, the NP0 ceramic capacitor between pin 14 of IC22 and ground has been increased from 22pF to 39pF. Similarly, the capacitor in parallel with trimcap VC1 has been increased from 18pF to 27pF.
8) Tests with a number of 32.768kHz crystals in the internal timebase oscillator (IC6) have shown that, in some cases, it may be necessary to increase the fixed NPO ceramic capacitor from 39pF to 47pF, in order to achieve calibration.

LED8 (1s Gating) and LED2 (Frequency Hz).

The 7-segment displays should now show the initial message for a second or so, but then change to display just '0.' on the extreme righthand digit, indicating that there is currently no input to Channel A of the counter.

If the display shows something other than '0.', this simply means that trimpot VR1 needs adjustment. In that case, use a small screwdriver to tweak VR1 in one direction or the other, until you get a zero display. Normally, this will be with VR1 set to about its midway position.

Next, check the DC voltage at the output (rearmost) pin of REG1. It should be very close to +5.00V. You should also find this voltage at pins 1 and 16 of IC5, at TP3 and also pins 11 and 32 of PIC micro IC22.

Frequency calibration

Finally, you need to make two more adjustments involving trimcaps VC1 and VC2, which are used to calibrate the counter's internal timebases. VC1 adjusts the PIC micro's main 8MHz clock frequency from which the '1µs' period counting timebase pulses are derived, while VC2 adjusts the frequency of the internal 1Hz timebase.

To make these adjustments, you'll need a 1pps (ie, 1Hz pulses) signal from an accurate source like the *GPS-Based Frequency Reference* (*EPE*, April-May 2009). Here's the procedure:

Switch the counter into its Period mode by pressing S4 and holding it down for half a second or so until LED6 lights. Then connect the accurate 1Hz signal from your GPS source to the counter's channel A input. This should give you a display of close to '1000000', which is the period of the 1Hz signal in microseconds. If the reading is slightly above or below this figure, carefully adjust VC1 until the reading is '1000000'.

The 1MHz 'period clock' that you've just calibrated is now used to adjust VC2, to calibrate the counter's internal timebase. To do this, press S4 again and hold it down until LED6 goes out, showing that the counter has switched back to frequency mode.

Next, disconnect the GPS 1Hz signal from the Channel A input and instead connect a short coax test lead with insulated clips at its free end. Then connect the 'live' clip of this cable to TP2 at the rear centre of the counter's main board (ignore the 'earthy' clip).

After making sure that LED7 is off (indicating that the counter is using its internal timebase), you should again get a reading that's close to '1000000', corresponding to the 1MHz 'period measurement' clock. If the reading you get is slightly above or below this correct figure, carefully adjust VC2 until the reading does become '1000000'.

This is using the default gating time of 1s, by the way. If you wish, you can switch to the 10s gating time by pressing S7 until LED9 lights. This will allow you to adjust VC2 until you get a reading of '1000000.0', which is about as accurate as it's possible to adjust the internal timebase.

Your counter will now be correctly set up and calibrated. All that remains is to fasten the top half of the case in position and the counter is ready for you to use.

Finally, if you don't have access to the *GPS-Based Frequency Reference*, just set VC1 and VC2 to mid-range for the time being. We plan to describe a low-cost GPS-based 1pps reference in the near future, so this can be used for calibration at a later date.

You'll need the *GPS-Based Frequency Reference* (*EPE*, April-May 2009) to accurately calibrate the unit (or some other accurate 1pps signal source). We will describe a low-cost GPS 1pps reference in the near future.

STORE YOUR BACK ISSUES ON CD-ROMS

FIVE YEAR DVD/CD-ROMs NOW AVAILABLE

ONLY £16.45 (£29.95 FOR 5 YEAR) each including VAT and p&p

VOL 1: **BACK ISSUES** – January 1999 to June 1999
Plus some bonus material from Nov and Dec 1998

VOL 2: **BACK ISSUES** – July 1999 to December 1999

VOL 3: **BACK ISSUES** – January 2000 to June 2000

VOL 4: **BACK ISSUES** – July 2000 to December 2000

VOL 5: **BACK ISSUES** – January 2001 to June 2001

VOL 6: **BACK ISSUES** – July 2001 to December 2001

VOL 7: **BACK ISSUES** – January 2002 to June 2002

VOL 8: **BACK ISSUES** – July 2002 to December 2002

VOL 9: **BACK ISSUES** – January 2003 to June 2003

VOL 10: **BACK ISSUES** – July 2003 to December 2003

VOL 11: **BACK ISSUES** – January 2004 to June 2004

VOL 12: **BACK ISSUES** – July 2004 to December 2004

VOL 13: **BACK ISSUES** – January 2005 to June 2005

VOL 14: **BACK ISSUES** – July 2005 to December 2005

VOL 15: **BACK ISSUES** – January 2006 to June 2006

VOL 16: **BACK ISSUES** – July 2006 to December 2006

VOL 17: **BACK ISSUES** – January 2007 to June 2007

VOL 18: **BACK ISSUES** – July 2007 to December 2007

VOL 19: **BACK ISSUES** – January 2008 to June 2008

VOL 20: **BACK ISSUES** – July 2008 to December 2008

VOL 21: **BACK ISSUES** – January 2009 to June 2009

VOL 22: **BACK ISSUES** – July 2009 to December 2009

VOL 23: **BACK ISSUES** – January 2010 to June 2010

VOL 24: **BACK ISSUES** – July 2010 to December 2010

VOL 25: **BACK ISSUES** – January 2011 to June 2011

VOL 26: **BACK ISSUES** – July 2011 to December 2011

VOL 27: **BACK ISSUES** – January 2012 to June 2012

VOL 28: **BACK ISSUES** – July 2012 to December 2012

VOL 29: **BACK ISSUES** – January 2013 to June 2013

FIVE YEAR CD/DVD-ROMs

No.1 – Jan '03 to Dec '07
No.2 – Jan '04 to Dec '08
No.3 – Jan '05 to Dec '09
No.4 – Jan '06 to Dec '10
No.5 – Jan '07 to Dec '11
No.6 – Jan '08 to Dec '12

NOTE: These DVD/CD-ROMs are suitable for use on any PC with a DVD/CD-ROM drive. They require Adobe Acrobat Reader (available free from the Internet – **www.adobe.com/acrobat**)

WHAT IS INCLUDED

All volumes include the *EPE Online* content of every listed issue. **Please note that we are unable to answer technical queries or provide data on articles that are more than five years old. Please also ensure that all components are still available before commencing construction of a project from a back issue.**

A great way to buy *EPE* Back Issues – our DVD/CD-ROMs contain back issues from our *EPE* Online website. Note: no free gifts are included.

Order on-line from
www.epemag.com (go to the UK store)
or by phone, fax, email or post

BACK ISSUES DVD/CD-ROM ORDER FORM

Please send me the following Back Issue DVD/CD-ROMs.

Volume Numbers:. . . . .

. . . .

Price £16.45 each, £29.95 for Five Year DVD/CD-ROMs – includes postage to anywhere in the world.

Name . . . .

Address . . . .

. . . .

. . . .

Post Code. . . . .

☐ I enclose cheque/P.O./bank draft to the value of £ . . . .

☐ please charge my Visa/Mastercard/Maestro £ . . . .

Card No. . . . .

Card Security Code . . . . (The last 3 digits on or just under the signature strip)

Valid From. . . . . Expiry Date . . . .

Maestro Issue No.. . . . .

Maestro MasterCard VISA

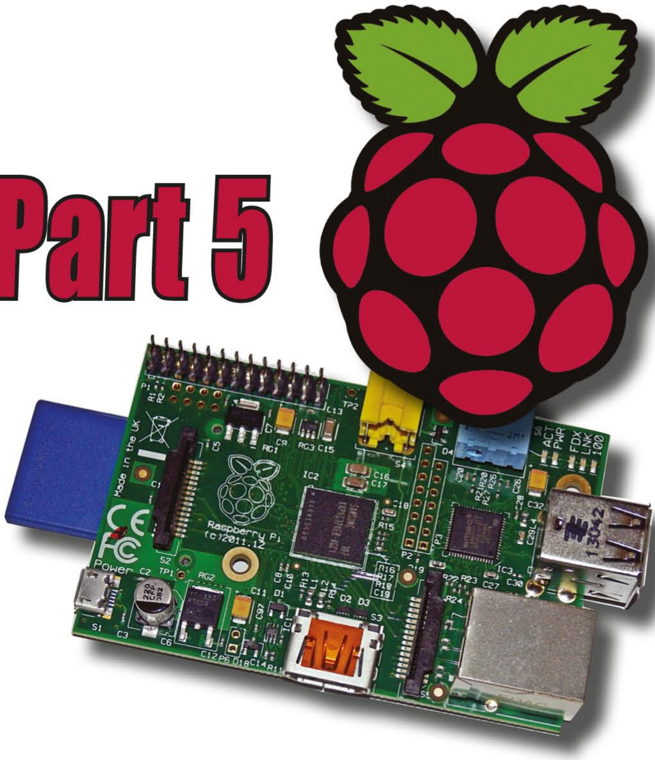
SEND TO: **Everyday Practical Electronics, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU.**

Tel: 01202 880299. Fax: 01202 843233.
E-mail: orders@epemag.wimborne.co.uk

Payments must be by card or in £ Sterling – cheque or bank draft drawn on a UK bank.
Normally posted within seven days of receipt of order.

Teach-In 2014 Raspberry Pi – Part 5

by Mike and Richard Tooley



Welcome to *Teach-In 2014* with Raspberry Pi. This exciting series has been designed for electronics enthusiasts wanting to get to grips with the immensely popular Raspberry Pi, as well as computer buffs eager to explore hardware and interfacing. So, whether you are considering what to do with your Pi, or maybe have an idea for a project but don't know how to turn it into reality, our *Teach-In* series will provide you with a one-stop source of ideas and practical information.

The Raspberry Pi offers you a remarkably effective platform for developing a huge variety of projects; from operating a few lights to remotely controlling a robotic vehicle through the Internet. *Teach-In 2014* is based around a series of practical exercises with plenty of information for you to customise each project to meet your own requirements.

The Raspberry Pi is no mean performer; it can offer you very similar performance to that which you might expect from a larger and much more expensive computer system, so don't be fooled by the relatively small price tag. By shopping around you can build a very effective computer system based on a Raspberry Pi for less than £100. However, if you are looking for something more modest and just want to take advantage of the Raspberry Pi as a single-board computer for a particular control application then you can be up and running for a very reasonable outlay.

This series will teach you about:

- **Programming** – introducing you to the powerful Python programming language and allowing you to develop your programming skills
- **Hardware** – learning about the components and circuits that are used to interface microcomputers to the real world
- **Computers** – letting you get to grips with computer hardware and software and helping you understand how they work together
- **Communications** – showing you how to connect your Raspberry Pi to a network and control a remote device using Wi-Fi and the Internet.

So, what's coming up? Regular features of Teach-In 2014 with Raspberry Pi will include:

- **Pi Project** – the main topic for each part will be a project that explores a particular use or application of the Raspberry Pi in the real word. Projects will include shopping for your Pi, set up, environmental monitoring, data logging, automation and remote control.
- **Pi Class** – each of our Pi Projects will be linked to one or more specific learning aims. Examples will include methods of representing and handling data, serial versus parallel data transmission and architecture of a microprocessor system.
- **Python Quickstart** – a short feature devoted to specific programming topics, such as data types and structures, processing user input, creating graphical dialogues and buttons and importing Python modules. We will help you get up and running with Python in the shortest time!
- **Pi World** – this is where we take a look at a wide range of Raspberry Pi accessories, including breadboards, prototype cards, bus extenders and Wi-Fi adapters. We will also help you build your Raspberry Pi bookshelf with a selection of recommended books and other publications.
- **Home baking –** suggested follow-up and extension activities such as 'check this out', a simple quiz, things to try and websites to visit.
- **Special features –** an occasional 'special feature'. For example, how to laser cut your own mounting plate – with additional downloadable resources such as templates and diagrams.

What will I need?

To get the best out of our series you will, of course, need access to a Raspberry Pi. If you don't already have one, don't worry – we will be explaining what you need and why you need it (we will also be showing you how you can emulate a Raspberry Pi using a Windows PC).

This month

In this month's *Teach-In 2014* we will be showing how the eight-channel analogue-to-digital converter (ADC) described last month can be used in sensing and measurement applications including light level, resistance and temperature. We will also be introducing you to techniques used for data logging that will enable you to save your data in the form of a list of comma separated values that can be imported into standard spreadsheet software. In *Pi World* we will be explaining some of the potentially confusing differences that exist between different versions of Python. To get you started, *Python Quickstart* describes a quick and efficient method of saving Python objects so that they can be recovered intact at a later time.

Python Quickstart

Last month, we looked at ways of using the `tkinter` package and giving your Python programs a more attractive graphical user interface (GUI). This month we will be introducing you to a simple way of saving your data so that you can recover it at some later time. This method (known as 'pickling' in the Python world) is much quicker and easier to use than the more conventional file handling methods.

Getting pickled

Python provides you with a simple and very effective way of storing an object in a file so that you can get it back at some later time. This is referred to as 'pickling', and it applies to any plain Python object such as a numeric variable, a line of text or even a complete list of data items. In order to use pickling you need first to import the pickle module, as follows:

```
import pickle
```

Once you have imported the pickle module you will need to provide a name for the file in which your pickled data will be stored. For example:

```
stockfile = 'stock.dat'
```

Now let's assume that you need to store a stock list of transistors:

```
stocklist = ['2N5401', 'MPSA42',
'BC548', 'BC557', 'J112']
```

To write the current data to the file we need to first open the file and then dump the stock list into it:

```
file = open(stockfile, 'wb')
pickle.dump(stocklist, file)
```

Finally, we need to close the file using the following:

```
file.close()
```

To recover the data at some later stage we just need to open the file (using the same filename) and then load the data back into our Python object. For example:

```
file = open(stockfile, 'rb')
stocklist = pickle.load(file)
```

The `wb` and `rb` (standing for 'write binary' and 'read binary' respectively) used when opening the file needs some comment. The `open()` function takes a filename and path as input and returns a file object (named `file` in the example above). The first parameter inside the bracket is the path to the file in which the data is to be saved, while the second parameter is the file mode which can be either (`r`), write (`w`), or append (`a`). If no mode parameter is passed, the mode defaults to read. To open binary files the letter `b` is appended to the mode parameter, and so we have `wb` and `rb` for write and read respectively.

Now for a couple of complete examples for you to try. We've kept things simple at first, with only one item (a short text string) to be pickled:

```
# pickletest1.py

# import the pickle module
import pickle

# item to be pickled
loc = 'Locator IO91sa'

# open the file for writing in binary mode
file = open('locdata.dat', 'wb')

# pickle the item and dump it in the file
pickle.dump(loc, file)

# and then close the file
file.close()
```

If you run the code above you will find that you have created a new file named `locdata.dat` with some data in it. To recover the data you can use the following:

```
# unpickletest.py

# import the pickle module
import pickle

# open the file for reading in binary mode
file = open('locdata.dat', 'rb')

# unpickle the item into a new object
loc = pickle.load(file)

# and print the item that we've just read
print(loc)

# then close the file
file.close()
```

Now for a more complex example. Let's suppose that you are using your Pi as part of a weather station that monitors temperature data along with wind speed and direction. The code fragment that follows would allow you to save the current weather station data to a file so that it can be recovered at some later time:

```
# pickle_example.py

# import the pickle module
import pickle
import time
import datetime

# datestamp = datetime.datetime.today()
new_data = [str(datetime.datetime.now()), 'Temp', 12, 'Wind',
15, 'NE']

# open the file for writing in binary mode
file = open('wxdata.dat', 'wb')

# pickle the weather data and dump it in the file
pickle.dump(new_data, file)

# and then close the file
file.close()
```

To recover the stored data you can use the following code:

```
# unpickle_example.py

# open the file for reading in binary mode
file = open('wxdata.dat', 'rb')

# unpickle the weather data into a new object
old_data = pickle.load(file)

# to print recovered weather data
print(old_data)

# before closing the file
file.close()
```

Finally, it's important to note that the pickle module is not secure against erroneous or malicious data. Because of this constraint, it's important to restrict the use of `unpickle` to known and trusted sources of data. Later, in *Pi Project* we will be using an alternative (and somewhat more conventional) method of capturing and storing data in the form of a list of comma separated values (CSV) files.

In this month's *Pi Class* we will introduce some simple techniques for counting and timing with the Raspberry Pi. All that you need is access to the Raspberry Pi's GPIO port and a handful of components.

Counting

Counting events, such as opening doors or passing through barriers, is a fairly common requirement that is easily satisfied with a Raspberry Pi and minimal external hardware. In fact, if your sensors produce digital outputs that are compatible with 3.3V logic there is no need for any complex interface circuitry and the connections can be made directly to the Raspberry Pi's GPIO port.

Fig.5.1 shows how a simple normally open switch can be connected to the GPIO. This switch can be virtually any single-pole miniature push-button or could be a low-cost proximity reed switch (see Fig.5.2). A simple Python module for counting switch closures is shown below.

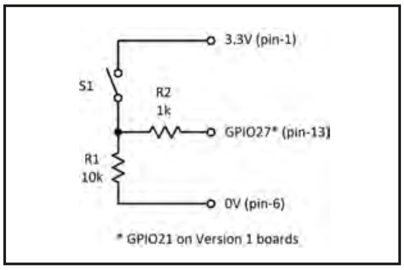


Fig.5.1. A simple switch interface for counting applications

```
import time
import RPi.GPIO as GPIO

# Configure GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(13, GPIO.IN)    # Input

# Tell user to press ENTER to start
print("Press ENTER to start counting ....")
#Initialise the count
count = 0

while True:
    print("Count = ", count)
    # Wait for the input to go high
    input_state = False
    while input_state = False:
        # Read input status
        input_state = GPIO.input(13)

    # Input has gone high so increment the count
    count = count + 1
    # Wait for the input to go low
    input_state = True
    while input_state = True:
        # Read input status
        input_state = GPIO.input(13)
     # Input has gone low so go back to the start
of the loop
```

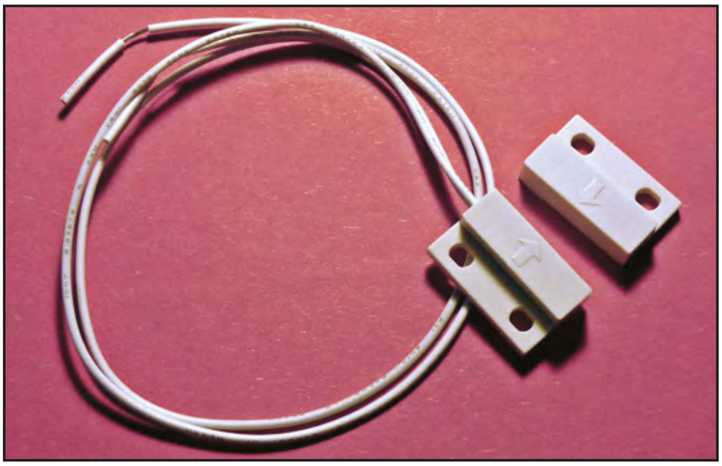


Fig.5.2. A low-cost proximity reed switch

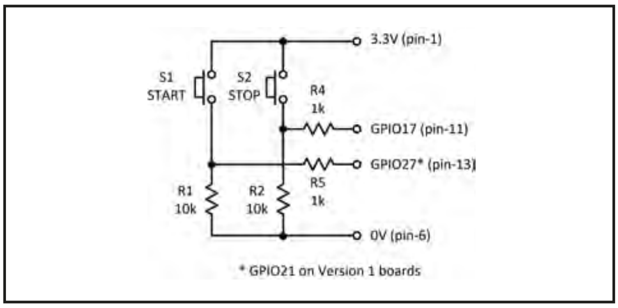


Fig.5.3. Simple interface for timing applications

Timing

The simple interface described earlier can be easily extended to work as a simple timer. All we need to do is add a second switch, as shown in Fig.5.3. The first switch, S1, starts the timing and the added second switch, S2, stops it, as shown in Fig.5.4. The time for each switch closure is taken from the Raspberry Pi's internal clock and the elapsed time is calculated from the difference between the two, as shown in the Python module below.

```
import time
import RPi.GPIO as GPIO

# Configure GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)  # STOP switch
GPIO.setup(13, GPIO.IN)   # START switch

# Tell user to press the start switch
print("Press START to start timing ....")

# Wait for switch press to start
switch_state = False
while switch_state = False:
    # Read switch status
    switch_state = GPIO.input(13)

#  START  switch  has  been  pressed  so  start
timing
start = time.clock()

# Tell user to press the stop switch
print("Press STOP to start timing ....")
switch_state = False
while switch_state = False:
    # Read switch status
    switch_state = GPIO.input(11)

# STOP switch has been pressed so calculate
the elapsed time
elapsed = (time.clock() - start)

# Finally print the elapsed time
print("Elapsed time = ", elapsed)
```

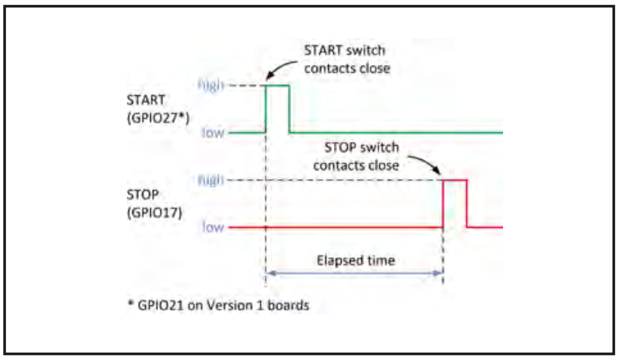


Fig.5.4. Timing diagram for the timer interface

Python problems

Until you become proficient with coding in Python you may find that your programs will not run the way that you intended. Indeed, in many cases they may simply refuse to run at all. This situation can often be avoided by carefully planning your code and building it from modules that you've already used and tested in other applications. Testing is usually more successful if it is carried out on an ongoing basis. As each new user-defined function and module is written it should be thoroughly tested as part of the ongoing development process. However, you may still encounter problems, even with a few lines of the most innocent looking code.

In the authors' experience, the ten most commonly encountered problems when developing Python code (listed in no particular order) are:

1. Failing to indent blocks of code correctly. After a block-opening statement, the next line should be indented (conventionally by four space characters or, when using the Python Shell, by one tab character).
2. Forgetting to add a colon at the end of a line that uses `for`, `if`, `while`, etc. This is a very common mistake made by first time Python programmers, and it will invariably result in a syntax error (see below).
3. Failing to ensure that all of your top-level code starts in the first column. Python uses indentation as a delimiter for a block of code and so the interpreter will get confused if it finds that your top-level code does not start in the extreme left-hand column.
4. Failing to be consistent with indenting at multiple levels. Any nested code blocks (ie, code blocks that occur within other code blocks) need to be nested to the correct level. This means paying particular care to the white space that occurs before each line of code and ensuring that the correct number of spaces are inserted.
5. Syntax errors – these are often encountered particularly where you are at the early stages of getting to grips with the language. Syntax errors are caused when the Python interpreter is unable to correctly parse a line of code. At 'run time', the Python parser will invariably repeat the offending line along with an arrow that indicates the point in the line where the error has occurred. For example, failing to include a full colon at the end of a `while` statement.
6. Exceptions – a line of code may sometimes parse correctly (ie, appear syntactically correct) but still contain an error that will prevent its execution. In such an event you are likely to be presented with a Traceback message that provides you with information on what has happened. There are several different types of exception, including `ZeroDivisionError`, `NameError` and `TypeError`. In a future Teach-In we will focus on methods of handling certain types of exception, making your code more robust and thus better able to deal with unexpected events.
7. Failing to import the library modules referenced by your code and also attempting to execute code that makes use of incompatible library modules (ie, those written for earlier versions of Python). This will usually result in a `NameError` exception (see above).
8. Forgetting to initialise variables. In Python, it is important to assign a value to a variable before you attempt to use it within an expression. This problem often arises if you fail to initialise a counter to 0 before testing or incrementing it.
9. Function calls. When calling a function (whether in-built or user-defined) you need to remember to use opening and closing brackets immediately after the function name, even if you are not passing any parameters). For example, use `password()` rather than just `password` when calling a user-defined function with the same name that requires no parameters.
10. Failing to ensure the correct order of operations within Python expressions. For example, `w = 1 + 2 / 3 * 4`. Brackets should be used to perform operations with the correct order of precedence. Note that each of the following produce different results:

```
x = (1 + 2) / 3 * 4
y = 1 + (2 / 3) * 4
z = (1 + 2 / 3) * 4
```

Finally, it's worth noting that Python's integrated development environment (IDLE) makes good use of colour, which is automatically added when it recognises comments, definitions, keywords, and strings. You might want to keep an eye on the following when you next enter some code:

Item	**Colour**
Comments	Red
Definitions	Blue
Keywords	Orange
Strings	Green

Pi Project

Last month we described the design and construction of an eight-channel analogue input port. We also showed how the port can be used to perform simple voltage measurements. The eight-channel analogue-to-digital converter (ADC) can also be used to interface a variety of transducers and sensing devices that base their operation on a change in resistance, so in this month's *Pi Project* we shall continue the analogue theme with a variety of other sensing and measurement applications, including light level, resistance and temperature.

Sensing light level

Light levels can be easily sensed using a light-dependent resistor (LDR). In order to obtain a small voltage that we can feed to one of the available ADC channels we need to supply the transducer with a small current (typically less than 1mA) and then use the voltage dropped across the LDR as an input, as shown in Fig.5.5. The resistance of the LDR (in this case a commonly available NORP12 type) will typically fall from several tens of thousands of ohms to only a few hundred ohms as the light level increases from semi-darkness to bright sunlight. The corresponding voltage (developed across the LDR) will fall as the light level increases and this change can be fed to any available analogue input. A simple environmental monitoring station might need to sense the ambient light level at regular intervals and make this data available for remote interrogation via a network (see last month's *Teach-In*) or via the Internet (see next month's *Teach-In*).

Since we are not going to attempt to make absolute measurements of light level in this simple application we will concentrate on merely assigning a simple text message to each of five bands of light level, ranging from total darkness to bright light. For this purpose (and bearing in mind the need to accommodate logarithmic changes in light intensity) we will adopt the somewhat arbitrary ranges shown in Table 5.1. The corresponding range of values returned from the `readadc` function (with RV1 set to mid-position) is also shown in Table 5.1. We will use these values in our Python code (see below) when assessing the light level at any particular time in order to return a text string that describes the particular light condition.

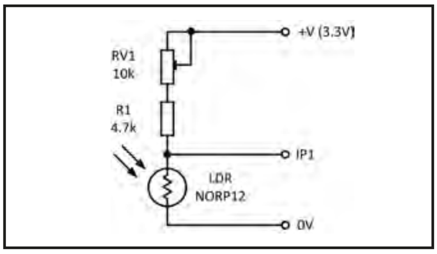


Fig.5.5. Using a light-dependent resistor with the eight-channel analogue input board

Table 5.1: Light level ranges

Light condition	Approximate light level range (lux)	Approximate range of values returned from the readadc function
Dark (night)	Less than 5	>895
Twilight	5 to 50	512 to 895
Very dull	50 to 500	134 to 511
Overcast	500 to 5,000	30 to 133
Bright	More than 5,000	<30

```
#!/usr/bin/env python
import time
import os
import RPi.GPIO as GPIO
from gpiospiadc import *

GPIO.setmode(GPIO.BOARD)
DEBUG = 1

# Define pins on the GPIO connector
SPICLK = 23
SPIMISO = 21
SPIMOSI = 19
SPICS = 24

# Set up the SPI
GPIO.setup(SPIMOSI, GPIO.OUT)
GPIO.setup(SPIMISO, GPIO.IN)
GPIO.setup(SPICLK, GPIO.OUT)
GPIO.setup(SPICS, GPIO.OUT)

sensor_adc = 0

# Function to determine current light condition
def light(sensor_value):
    if sensor_value >= 896:
        condition = 'Dark'
    elif sensor_value >= 512:
        condition = 'Twilight'
    elif sensor_value >= 134:
        condition = 'Overcast'
    elif sensor_value >= 30:
        condition = 'Dull'
    else:
        condition = 'Bright'
return condition

# Main loop starts here
while True:
    sensor_value = readadc(sensor_adc, SPICLK,
SPIMOSI, SPIMISO, SPICS)
    print(light(sensor_value))
    time.sleep(1)
```

Resistance measurement

Now let's move on to look at a slightly more complex application in the form of a digital ohmmeter. In this application we are not just looking at a range of resistance values, but also we need to be able to produce a reasonably accurate indication of the resistance of an unknown component. In order to do this we need to apply a known value of current (and keep this constant) to the unknown component. If we can do this we can use the voltage drop that appears across the unknown component to find its resistance value. This principle is illustrated in Fig.5.6.

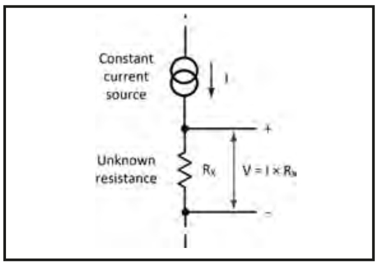


Fig.5.6. Constant-current source for measuring resistance

A simple but effective constant-current source can be built using just two transistors, as shown in Fig.5.7. The circuit delivers a nominal current of 1mA (adjustable by means of RV1). To calibrate the digital ohmmeter a resistor with an accurately known resistance (eg, 1kΩ ±1%) should be connected to the input terminals and RV1 should be adjusted for a resistance indication of exactly 1000Ω.

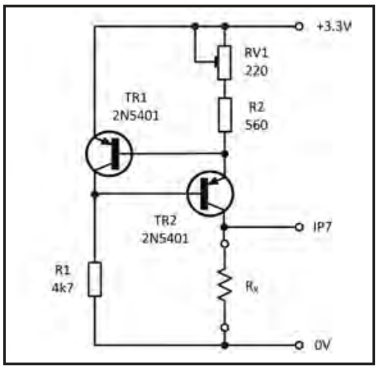


Fig.5.7. Digital ohmmeter circuit

The range of resistance that can be measured reasonably accurately with this arrangement extends from less than 100Ω to around 2.5kΩ. Outside this range there will be a need to make a corresponding change to the value of the constant current. For example, a current source of 10mA would be required to extend the measuring range from around 10Ω to 250Ω.

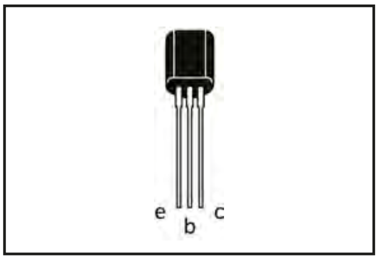


Fig.5.8. Pin connections: TR1,2

Fig.5.9. Digital ohmmeter display

The Python code for a simple ohmmeter is shown below. This follows along the same lines as the digital voltmeter application that we described last month, and once again uses the `tkinter` library module as a means of providing a friendly graphical user interface.

At the start of the code we need to import the required library modules (in this case we need access to five of them, including `tkinter`). Next, we've defined a root window (every GUI program must have one of these) and given it a title (`kOHM`). This makes it look a bit more impressive than just the default, `tk`. Within the root window, and so that our display is easily readable, we've increased the font size to 36 and used Arial font (in black) against a green background (as shown in Fig.5.9).

In the main loop of the program we've used a user-defined function (UDF) called `measure()`. This reads the value returned from Channel 0 (IP1) of the ADC and then converts it to a corresponding resistance (`sensor_resistance`). It then converts the sensor voltage into a format that will produce a sensible display using the `meter.config()` function. The entire loop is repeated every 200ms until the user closes the display window (at which point the program terminates). The code is shown below:

```
#!/usr/bin/env python
from tkinter import *
import time
import os
import RPi.GPIO as GPIO
from gpiospiadc import *

root = Tk()
root.title("kOHM")
resistance1 = ''
meter = Label(root, font=('arial', 36, 'bold'),
bg='yellow')
meter.pack(fill=BOTH, expand=1)
```

```
GPIO.setmode(GPIO.BOARD)
DEBUG = 1

# Define pins on the GPIO connector
SPICLK = 23
SPIMISO = 21
SPIMOSI = 19
SPICS = 24

# Set up the SPI
GPIO.setup(SPIMOSI, GPIO.OUT)
GPIO.setup(SPIMISO, GPIO.IN)
GPIO.setup(SPICLK, GPIO.OUT)
GPIO.setup(SPICS, GPIO.OUT)

sensor_adc = 7

def measure():
    global resistance1
     # get the current resistance from the SPI
device
    sensor_value = readadc(sensor_adc, SPICLK,
SPIMOSI, SPIMISO, SPICS)
   sensor_resistance = (sensor_value * 3.3)/1024
    resistance2 = sensor_resistance
    sv = '%.3f' % sensor_resistance
    # if the resistance has changed, update it
    if resistance2 != resistance1:
        resistance1 = resistance2
        meter.config(text=sv)
    # update the display every 200ms
    meter.after(200, measure)

measure()
root.mainloop()
```

Temperature measurement

Now let's look at how we can very easily interface a temperature sensor to the eight channel analogue input port. This will allow you to sense temperatures with a reasonable degree of accuracy and then either display the current temperature or save it to a file for later analysis.

The TMP35 chip that we will be using for temperature sensing is one of a family of devices that together are capable of covering a temperature range from –40°C to +125°C with an accuracy better than ±2°C and typically ±1°C at +25°C. Usefully, each of the sensors in the range produces an output voltage that is linearly proportional to the Celsius (centigrade) temperature, as illustrated by Fig.5.10. All three devices are available in low cost 3-lead TO-92, 8-lead SOIC-N, and 5-lead SOT-23 surface-mount packages and they are thus ideal for building into miniature probes.

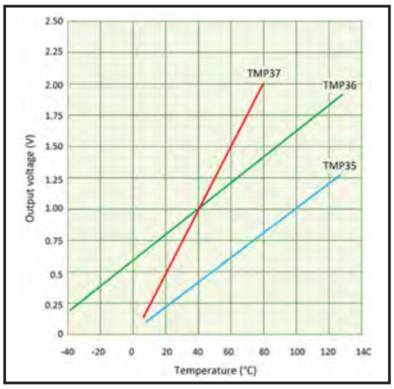


Fig.5.10. Characteristic graphs for the TMP35, TMP36 and TMP37 temperature sensing chips

The TMP35 is intended for single-supply operation from 2.7 V to 5.5 V maximum, and it is thus eminently suitable for operation from the Pi's +3.3V supply. To avoid the risks associated with self-heating, the chip requires only a very small supply current (well below 50µA). In addition, a shut-down function is provided that limits the residual supply current to less than 0.5µA.

The TMP35 provides an output of 250mV at 25°C and is suitable for sensing temperatures in the range +10°C to +125°C. Other devices in the series are specified from –40°C to +125°C (TMP36) and +5°C to +100°C (TMP37). The TMP36 provides an output voltage of 750mV at +25°C whilst the TMP37 produces 500mV at the same temperature (see Fig.5.10). Just in case you might need to use a different sensor we've summarised the characteristics of all three of these temperature sensors in Table 5.2.

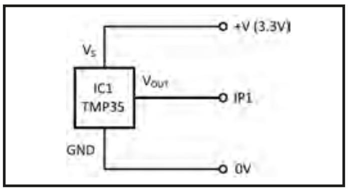


Fig.5.11. Temperature measurement interface

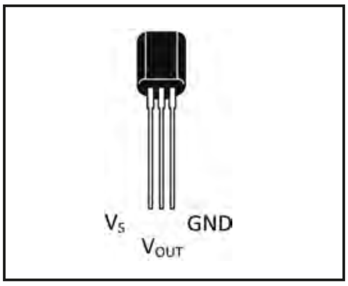


Fig.5.12. Pin connections for the TMP35, TMP36 and TMP37 temperature sensors

The interface to the eight-channel analogue port (see Fig.5.11) is extremely simple and no other components are required apart from the temperature sensor. The pin connections for TMP35, TMP36 and TMP37 temperature sensors are shown in Fig.5.12.

```
#!/usr/bin/env python
import time
import os
import RPi.GPIO as GPIO
from gpiospiadc import *

GPIO.setmode(GPIO.BOARD)
DEBUG = 1

# Define pins on the GPIO connector
SPICLK = 23
SPIMISO = 21
SPIMOSI = 19
SPICS = 24

# Set up the SPI
GPIO.setup(SPIMOSI, GPIO.OUT)
GPIO.setup(SPIMISO, GPIO.IN)
GPIO.setup(SPICLK, GPIO.OUT)
GPIO.setup(SPICS, GPIO.OUT)

sensor_adc = 0

while True:
    sensor_value = readadc(sensor_adc, SPICLK,
SPIMOSI, SPIMISO, SPICS)
    sensor_voltage = (sensor_value * 3.3)/1024
   sensor_temperature = (sensor_voltage - 0.5)
* 100
    print("%4d/1023 => %5.3f V => %4.1f deg.C"
%  (sensor_value,  sensor_voltage,  sensor_
temperature))
    time.sleep(0.5)
```

Table 5.2: Temperature sensor specifications

Device	TMP35	TMP36	TMP37
Useful temperature range	+10°C to +125°C	–40°C to +125°C	+5°C to +100°C
Output voltage at 25°C	250mV	750mV	500mV
Temperature characteristic	10mV/°C	10mV/°C	20mV/°C

Data logging

Finally, here's a complete data logging application in which we capture and store temperature data and save it in a comma-separated (CSV) file for later analysis. We've provided prompts to request a filename from the user (and then added a .CSV extension) as well as the number of samples to take and the interval between the samples (entered in seconds). So, for example, to monitor the temperature every minute for one hour would require 60 samples with an interval of 60 seconds between samples. Note that in order to eliminate false and/or rapidly changing readings the program waits for two identical sensor values to be returned before appending them to the data in the CSV file. Fig.5.13 shows some typical captured data displayed in graphical form using a popular spreadsheet program.

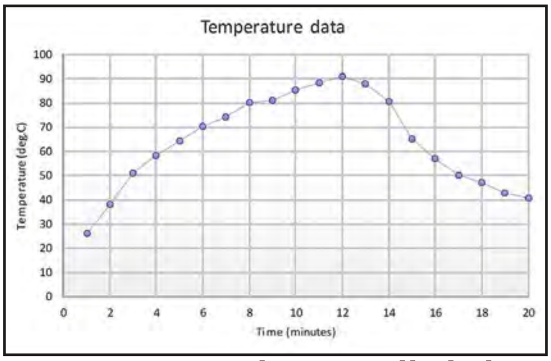


Fig.5.13. Temperature data captured by the data logging application and presented in graphical form using a spreadsheet

```
#!/usr/bin/env python
import time
import os
import csv
import RPi.GPIO as GPIO
from gpiospiadc import *

GPIO.setmode(GPIO.BOARD)
DEBUG = 1

# Define pins on the GPIO
connector
SPICLK = 23
SPIMISO = 21
SPIMOSI = 19
SPICS = 24

# Set up the SPI
GPIO.setup(SPIMOSI, GPIO.
OUT)
GPIO.setup(SPIMISO, GPIO.IN)
GPIO.setup(SPICLK, GPIO.OUT)
GPIO.setup(SPICS, GPIO.OUT)

sensor_adc = 0

filename = input('Filename to save to: ')
filename = '/home/pi/Desktop/' + filename + '.csv'
samples = input('Number of samples: ')
interval = input('Interval between samples (seconds): ')
void = input('Press enter to start collecting data ...')
print('Thanks - now collecting data ....')
count = 1
while (count < (int(samples) + 1)):
    print('Sample No. ' + str(count))
    sensor_value = 0
    sensor_value1 = 1
    while sensor_value1 != sensor_value:
          sensor_value = readadc(sensor_adc, SPICLK, SPIMOSI,
SPIMISO, SPICS)
          sensor_value1 = readadc(sensor_adc, SPICLK, SPIMOSI,
SPIMISO, SPICS)
    sensor_voltage = (sensor_value * 3.3)/1024
    sensor_temperature = (sensor_voltage - 0.5) * 100
    sensor_data = round(sensor_temperature, 1)
    with open(filename, 'a', newline='') as f:
        writer = csv.writer(f)
        writer.writerow([count] + [sensor_data])
    time.sleep(int(interval))
    count = count + 1
print('Finished - ' + str(count - 1) + ' samples collected')
```

Lastly, don't forget that you will need to run all of the code listed above with user privileges. So, for example, if your Python3 program code is saved as `temperature.py` in a folder called `Work_files` on your desktop the command to execute the program from the terminal program (eg, LXTerminal) would be:

```
sudo python3 /home/pi/Desktop/Work_files/temperature.py
```

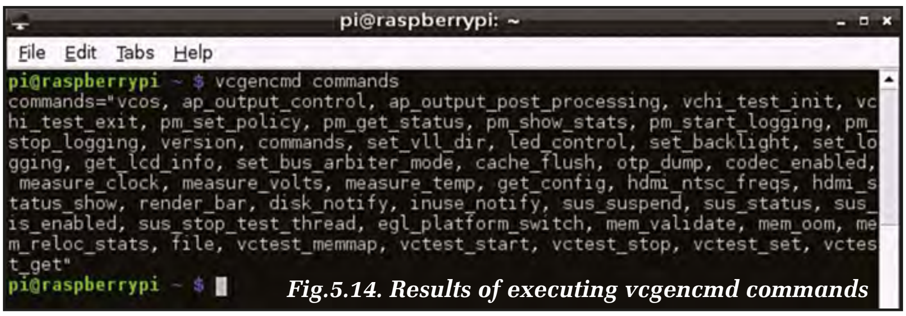


Fig.5.14. Results of executing vcgencmd commands

Using firmware commands to monitor Raspberry Pi

The Raspberry Pi's chip designers have thought to provide you with a means of monitoring some of the more critical aspects of your Raspberry Pi using a variety of commands that are present in the Pi's firmware. To display a list of the currently available commands (they may change with future firmware versions) you simply need to enter the following directly from the command line or from within LXTerminal:

```
vcgencmd commands
```

This will display a long list of commands. Since many of these commands might appear a little obscure at first sight, we will just describe some of the most useful here.

Checking the Pi's core temperature

It can often be useful to have some idea of what temperature the processor core of your Raspberry Pi is running at. This information can be particularly important if you have your Raspberry Pi mounted in a small enclosure or when the immediate vicinity is not adequately ventilated. The necessary command (entered from the command line or from within LXTerminal) is simply:

```
vcgend measure_temp
```

This will typically display a message of the form: `temp=35.8'C`

Monitoring the Raspberry Pi's internal supply voltages

The Raspberry Pi has several internal supply voltage rails (see Fig.1.8 of *Teach-In 2014*) and each one must remain within specification for the Pi to operate correctly. The following commands check these voltage rails:

To check the processor's core voltage:
`vcgend measure_volts`

This will typically display a message of the form: `volt=1.20V`

To check the SDRAM controller voltage:
`vcgend measure_volts sdram_c`

This will typically display a message of the form: `volt=1.20V`

To check the SDRAM I/O voltage:
`vcgend measure_volts sdram_i`

This will typically display a message of the form: `volt=1.20V`

To check the SDRAM physical voltage:
`vcgend measure_volts sdram_p`

This will typically display a message of the form: `volt=1.23V`

Which version of Python?

As some of our readers have already discovered, life can become problematic when moving from earlier versions of Python to the latest version (Python 3.x). The reason for this is simply that Python 3 has been conceived as intentionally backwards incompatible with earlier versions. This, when combined with numerous other changes, makes it very difficult to do a straightforward 'port' of your code from one version to the other.

We've based this series on Python 3 for several reasons, notably:

1. Support for earlier versions of the language is likely to be discontinued and future development of the standard Python libraries is likely to focus on Python 3.x.
2. The latest version of Python 3 is friendlier than previous versions and thus more suitable for beginners.
3. A number of problems and anomalies associated with earlier versions have now been resolved and various aspects of the language have been 'tidied up'.
4. Python 3.x is becoming widely available and included in the latest Linux distributions.

Having said that, readers need to be aware that Python version 2.7 is still in common use and so you will often see Python code that will not run as intended (or not run at all!) in the most recent version. The solution, of course, is to 'port' the code to the latest Python version but, as mentioned previously, this can often be fraught with difficulty. Here are a few pointers that will help you get up to speed with this task:

The print() function

The print statement present in earlier version (ie, Python version 2.7 and before) has now been replaced by the `print()` function and the syntax of the old print statement has been replaced with arguments that can be passed into the print function. This is a major change and will always require attention. Here's an example:

```
# Python 2.7
print "Fetching data – please
wait"
```

```
# Python 3.3
print("Fetching data – please
wait")
```

At the end of printing, both of these code fragments will produce a newline character. To continue printing on the same line we would need to use:

```
# Python 2.7
print "Fetching data – please
wait",
```

```
# Python 3.3
print("Fetching data – please wait", end = "")
```

Thankfully there's nothing very earth shattering here.

The input() function

In earlier versions of Python, keyboard input was obtained from the `raw_input()` function. In Python 3.x the `raw_input()` function is replaced by the `input()` function. Here are some examples:

```
# Python 2.7
password = raw_input()
```

```
# Python 3.3
password = input()
```

To combine the keyboard input with a prompt message you would use:

```
# Python 2.7
password = raw_input("Enter your password: ")
```

```
# Python 3.3
password = input("Enter your password: ")
```

String formatting

Python 3.x offers a new `format()` method that can be used to format strings. In previous versions (and also in Python 3) you can use `%s` inside a string and then follow the string with a list of values, as follows:

```
# Python 2.7 and 3.3
message = "User name: %s Password: %s" % ("EPE", "everyday13")
```

If we were to subsequently print the message (using either `print` or `print()` as appropriate to the Python version) it would appear as:

```
User name: EPE Password: everyday13
```

The new `format()` method in Python 3.x uses a list of arguments that can be more easily associated with the position in a string, as in the following example:

```
# Python 3.3
message = "User name: {0} Password: {1}".format("EPE", "everyday13")
```

The numbers inside the curly brackets give the position in the list. So, if, for example we had reversed their order like this:

```
# Python 3.3
message = "User name: {1} Password: {0}".format("EPE", "everyday13")
```

The message, when printed, would appear as:

```
User name: everyday13 Password: EPE
```

Finally, here's another example:

```
# Python 3.3
message = "Direction: {0} {1} {2} {3}".format("N", "E", "S", "W")
```

The message, when printed, would appear as:

```
Direction: N E S W
```

If we needed to travel around the points of the compass in the opposite direction (ie, anticlockwise instead of clockwise) we would need to use:

```
# Python 3.3
message = "Direction: {0} {3} {2} {1}".format("N", "E", "S", "W")
```

Which would result in the following message:

```
Direction: N W S E
```

Other differences

There are many other differences (some of which are quite subtle) between Python 2.x and Python 3.x. These include the return value obtained by the `range()` function and the result of performing a division with the / operator. For further information we suggest that you go to: **http://docs.python.org/3/whatsnew/3.0.html**

Keeping it cool

During normal operation in an environment where the ambient temperature is between 20°C and 25°C, and depending on the load on the processor, you can expect the processor chip on the Raspberry Pi to run at a temperature of between 42°C and 50°C. This is well within the chip's maximum operational specification of 85°C and no additional cooling is necessary.

With the Pi fitted into a small enclosure with limited ventilation, and depending on processor load, you can expect the processor temperature to rise to between 48°C and 55°C. Once again, this is well within the maximum ratings for the chip and it should not cause the Pi to lock-up or fail prematurely.

That said, there are a number of situations in which the build-up of heat may cause problems and these include operating the Raspberry Pi for long periods in a small enclosure where there is no free air flow, where the ambient temperature exceeds 35°C, and where the processor is being overclocked (imposing a significantly increased load on it). In such cases it is well worth taking some action to reduce the temperature by using some additional cooling.

Additional cooling can take the form of small heat sinks (either aluminium or copper) which can be attached to the three principal chips (ie, the voltage regulator, LAN controller, and processor). Typical heat sink dimensions (length, width and height) are as follows:

Voltage regulator	9 × 9 × 5mm
LAN controller	9 × 9 × 5mm
Processor	14 × 14 × 6mm

Heat sinks are available in kits specifically for the Raspberry Pi and are usually supplied with thermal tape attached at the bottom. Note that it is important to check that your Raspberry Pi will still fit in its enclosure when the heat sinks are fitted or when an additional I/O daughter board (such as the Humble Pi that we've been using in this series) is fitted over the top of the Raspberry Pi's board.

An alternative to fitting heat sinks (and one that can be successfully used when a daughter board is fitted) is that of using a small 5V DC fan to provide additional cooling. A typical fan would have the following specification:

Dimensions	20 × 20 × 10mm
Operating voltage	5V DC
Power rating	0.2W
Speed	7000 RPM
Airflow	0.86 CFM

Finally, here are a few tips that will help you combat the effects of high operating temperatures:

1. Ensure plenty of free air flow around your Pi
2. Never locate the Pi close to heat-producing objects such as radiators, heaters, and other electronic equipment
3. Always keep the Pi out of direct sunlight (particularly when fitted in an enclosure)
4. Consider fitting a heat sink kit or a small fan (see earlier) whenever you expect the ambient temperature to rise above 30°C
5. Don't attempt to overclock your Pi.

In next month's Teach-In with Raspberry Pi

In next month's *Teach-In 2014* we will be introducing the Pi's i2C interface. Our *Pi Project* deals with the construction of a digital-to-analogue (DAC) interface and *Perfect Python* looks at error logging. Held over from this month, our *Home Baking* feature will show you how you can use the Internet to remote control your Raspberry Pi anywhere in the world.

Correction...

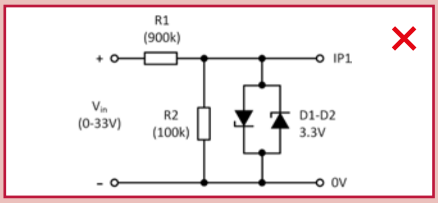


Fig.4.13. Potential divider and input protection for simple digital voltmeter

We need to inform you that there is an error in Fig.4.13 from *Teach-In Part 4* (Jan' 14 issue of *EPE*). The circuit diagram showed two diodes, D1 and D2. There should in fact only be one diode (D2) and this is shown in the corrected circuit diagram on the right, renamed as ZD.

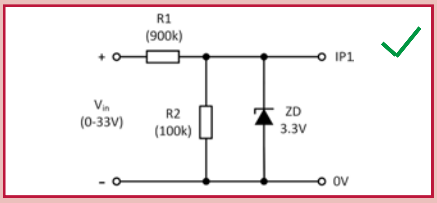


Fig.4.13. Potential divider and input protection for simple digital voltmeter


RasWIK
Wireless Inventor Kit
for the Raspberry Pi ™
Contains
88
Parts
• Starter examples require no soldering at all
• Plug in wires and breadboard to make building easy and fast
• A pre installed and configured Raspberry Pi OS makes using your Pi a breeze
• Examples you build, can be mixed with our out the box wireless devices...how cool!
• "It provides possibly the simplest platform for experimenting with wireless sensor networks I've ever seen."
(Gareth Halfacre, CustomPC, Issue 121, Oct 2013)
• Made in the UK
£49.99
www.ciseco.co.uk
Raspberry Pi is a trademark of the Raspberry Pi Foundation
CISECO
Powering creative minds

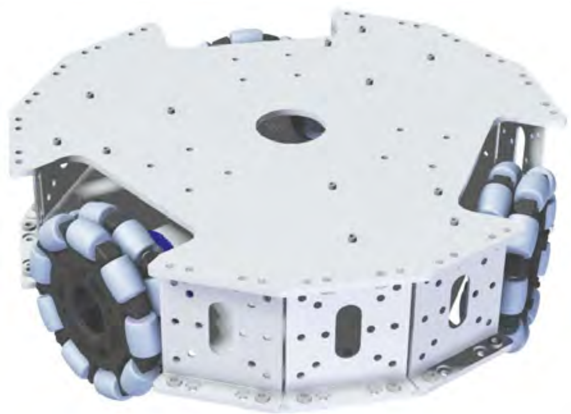

UK designed, UK made, with pride.
Tel. 01298 70012
www.peakelec.co.uk
sales@peakelec.co.uk
Atlas House, 2 Kiln Lane
Harpur Hill Business Park
Buxton, Derbyshire
SK17 9JL, UK
Follow us on twitter for tips, tricks and news.
@peakatlas
For insured UK delivery: Please add £3.00 inc VAT to the whole order. Check online or give us a call for overseas pricing.
PEAK®
electronic design ltd
LCR45
LCR and Impedance Meter
with Auto and Manual modes
Brand new product!
Introducing a new powerful LCR meter that identifies your passive components (Inductors, Capacitors and Resistors) but also measures complex impedance, impedance with phase and admittance!
Auto and Manual test modes to allow you to specify the test frequency or component type.
- Continuous fluid measurements.
- Improved measurement resolution (<0.2μH, <0.2pF)
LCR and impedance meter
LCR45
on - menu
enter - off
warning: avoid charge/voltages
Component Summary
Complex Impedance
£99.95
£83.29+VAT
DCA75
The all new
DCA Pro
"A very capable analyser"
- Detailed review in RadCom magazine (March 2013)
Exciting new generation of semiconductor identifier and analyser. The DCA Pro features a new graphics display showing you detailed component schematics. Built-in USB offers amazing PC based features too such as curve tracing and detailed analysis in Excel. PC software supplied on a USB Flash Drive. Includes Alkaline AAA battery and comprehensive user guide.
Now Shipping
£115.95
£96.62+VAT
UTP05
- Identify and fault-find your network cabling.
- Comprehensive connection analysis.
- Detailed fault descriptions.
- Full colour wiring charts.
- Supplied with everything pictured here.
£77.95
£64.96+VAT
It's only possible to show summary specifications here. Please ask if you'd like detailed data. Further information is also available on our website. Product price refunded if you're not happy.

Max's Cool Beans

By Max The Magnificent

Designing my Arduino robot

Last month, I discussed my plan to construct an Arduino-controlled robot. What kicked me into gear was investing in a Kickstarter project for a low-cost machine-vision sensor (**http://kck.st/1bGfhdn**) that can detect and identify objects. Obviously, there's no point in having this capability unless you can use this information to control something – hence my robot.



3D Model of Max's Magnificent Robot (Base Platform)

Initially, I thought about purchasing an off-the-shelf robot platform, but everything I saw on the Internet was either incredibly boring or mind-bogglingly expensive, so I decided to build my own from the ground up. The current state of play exists as a virtual 3D model (shown above).

Hardware – wheels, motors and platform

I've opted for a 3-wheeled platform in which three geared motors and associated wheels are mounted at 120° to each other. The reason this works is that I'm using 4-inch diameter omni-directional wheels from VexRobitics.com (**http://bit.ly/Hhesya**). These little beauties have a dual set of rollers mounted around the periphery of the wheel, thereby allowing the wheel to move in any direction.

To power my wheels, I've opted for three Pololu 12V, 50:1 DC gear motors with encoders (**http://bit.ly/1baaMWj**). These have a free-run speed of 200RMP at 12V, drawing 400mA. Meanwhile, the integrated encoder generates 3,200 counts per revolution. The reason for the integrated encoders is to provide one form of feedback about the robot's actual location.

The platform is being built by my chum Eugene 'Willie' Richards. Willie is a mechanical design expert who specialises in working with sheet metal, so he has all sorts of useful nuggets of knowledge like the tensile strengths of different materials, how much material is required to accommodate bends, and all sorts of other 'stuff'. Even better, Willie has access to precision metal fabrication and machining tools at a facility that's located just around the corner from my office.

In the case of my robot, Willie has decided to employ just two elements to minimise cost and effort – large (12-inch diameter) flat plates and vertical risers. In addition to separating the plates, the risers can also be used to support a variety of sensors. Note that the illustration shown here shows only one 'slice of the cake', but the platform is being designed in such a way that I can add more layers in the form of additional riser/plate combos.

Did you hear, see, smell that?

One thing about having a robot platform is that it's tempting me to play with all sorts of cool things, especially sensors. In fact, I'm planning on making this a sensor-laden platform. In addition to my machine vision sensor, I'm going to have ultrasonic sensors, infrared sensors, a three-axis accelerometer, a three-axis gyroscope, and a three-axis magnetometer, not to mention temperature, humidity, and atmospheric pressure sensors. Also, since we're on this topic, I plan on having Hall effect sensors monitoring the current being used by my motors (although their free-running current is 400mA, their stall current is 5A, which could damage my batteries).

Each of these sensors presents its own challenges. Take the ultrasonic ranging sensors, for example. Some robots might have one such sensor mounted on the front of the robot to detect if it was about to run into anything. But my robot doesn't have a 'front', it can move in any direction. So, one option would be to have a single ultrasonic sensor mounted on the top of the robot with a servo mechanism – spinning it round like the radar on a ship. However, I plan on locating my machine vision sensor on the top, so another option – the one I plan on using – is to mount nine ultrasonic sensors around the base of the robot – one for each of the risers.

Now, call me 'old fashioned', but given a choice, I prefer to spend as little of my hard-earned money as possible. The HC-SR04 ultrasonic sensor from **Amazon.com** (**http://amzn.to/1eJB5Ye**) boasts a detection distance from 2 to 450cm and costs only $2.40. This is very attractive, but there is a fly in the soup. The problem is that this sensor requires an external microcontroller to initiate the 'ping' and to then count how much time passes until the sensor detects the response (the reflection from the nearest object). If I were to use my main controller to handle nine ultrasonic sensors, it wouldn't have any time free to do anything else. The trick is to off-load the main processor by having a small, dedicated processor controlling the sensor. My chum Duane Benson pointed me at the SRF08 ultrasonic range finder module (**http://bit.ly/1c3qxPR**) that has its own on-board microcontroller. This means that all the main controller has to do is instruct the sensor to take a reading, and then come back later and ask for the result.

The problem is that SRF08 modules cost around $55 each, which makes this an expensive option if you intend to use nine of them. Fortunately, Duane has designed a little circuit board populated with a PIC microcontroller and a handful of other parts. Duane's board, which costs only $3 fully populated, makes a $2.40 HC-SR04 sensor behave exactly like a $55 SRF08 module.

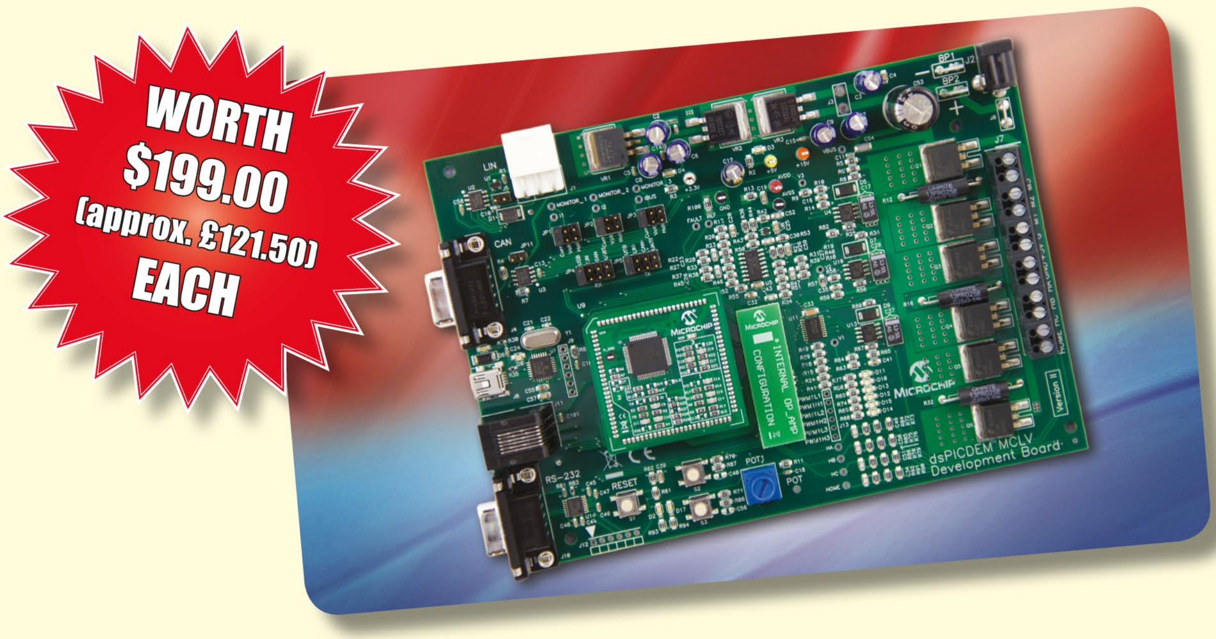
Next is power, so now I'm off to research battery technologies. Until next time, have a good one!

EPE EXCLUSIVE

Win a dsPICDEM MCLV-2 Development Board

EVERYDAY PRACTICAL ELECTRONICS is offering its readers the chance to win a dsPICDEM MCLV-2 Development Board. This board provides a cost-effective method of evaluating and developing sensored or sensorless brushless DC (BLDC) and permanent magnet synchronous motor (PMSM) control applications. It supports Microchip's 28-pin SOIC and 100-pin plug-in modules (PIM) with dsPIC33E or dsPIC33F digital signal controllers, as well as the use of the internal, on-chip op amps found on certain dsPIC devices or the external op amps found on the MCLV-2 board. A dsPIC33EP256MC506 internal op amp PIM (MA330031) is included. The board is capable of controlling motors rated up to 48V and 15A, with multiple communication channels such as USB, CAN, LIN and RS-232.

The dsPICDEM MCLV-2 Development Board is designed to control a BLDC motor or PMSM in sensor or sensorless operation. This flexible and cost-effective board can be configured in different ways for use with Microchip's specialised motor control digital signal controllers. It supports the dsPIC33E or dsPIC33F motor control device family, and offers a mounting option to connect either a 28-pin SOIC device or a generic 100-pin PIM.



HOW TO ENTER

For your chance to win a dsPICDEM MCLV-2 Development Board, please visit: **www.microchip-comps.com/epe-mclv** and enter your details in the entry form.

CLOSING DATE

The closing date for this offer is 28 February 2014

PIC n' Mix

Mike Hibbett

Our periodic column for PIC programming enlightenment

Software design and organisation

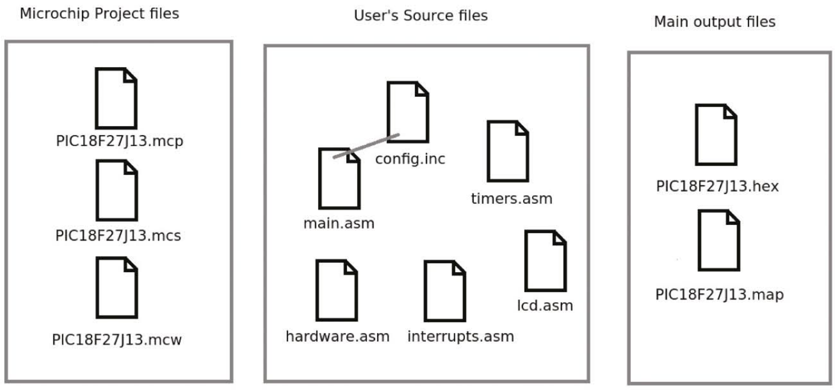


Fig.1 Template files organisation

LAST month, we completed the hardware for our PIC development board, transferring it from a prototyping board to its own home-made custom PCB. So we have our hardware, but what about software? What do we need?

We would like an 'easy-to-re-use' template of source files that can get the hardware up and running, doing something, *anything*, to give an indication that the hardware and software build tools are all working together correctly.

The goal of the template software is not only to accelerate the development of a project; it is also to provide the means of 'abstracting' the complexities of the hardware away, leaving the programmer to focus on the important stuff – implementing the functionality, what the project is actually supposed to do. Not to struggle with getting low-level interrupt routines to operate correctly or identify the correct set of CONFIG bits.

One of the reasons why we struggle at this point is that, in general, we do not spend much time developing these low-level functions. We normally spend a short time getting them working and then move on to the project itself. Consequently, the headaches and final solution to making the low-level 'stuff' work gets forgotten. Without a template for this 'stuff', we just repeat the same headaches time and time again!

So a template is great – it abstracts the hardware-specific features away, allowing you to move on to concentrating on your actual application, quickly. You can gain confidence with operational code, and then slowly tinker with the hardware features as you progress. When you make a change and the hardware stops working, you can be confident that it was your last change that was at fault, not the entire board.

Template code

Once again, let's start by defining some objectives, this time for our template code:

1) Easy to re-use on different projects.
2) Flash the on-board LED when built unmodified 'out of the box'.
3) Provide initialisation code for the main on-chip peripherals.
4) Provide some utility functions for other peripherals, such as a common LCD.

The first point needs a little clarification – what does 'easy to re-use' mean? In this case, we define it to mean how functionality can be turned on or off without affecting other functions, and how well each feature is described (in the code comments.)

We are keeping this reconfigurability to a minimum to avoid making our solution too complex; sometimes the configuration can be more complex than the problem it is trying to solve. Keeping that in mind, we provide configuration of the following:

- CPU CONFIG bits
- Interrupts (at least timer)
- ADC
- PWM
- High-speed crystal oscillator
- Watch crystal oscillator
- Internal RC oscillator
- 2 × 16 LCD module

Development environment

Part of the design of the template software includes choosing a development environment – why? because the library functions that a particular development environment may supply are usually different and incompatible with other environments. Changing your project code to support another environment (or 'porting' as it is known) can be difficult, time consuming and ultimately unrewarding, as it provides no 'new' functionality. So the best policy is to choose an environment and stick with it.

Some development environments, such as commercial ones from Microchip and MikroElektronika, provide extensive code libraries for commonly used functions, whereas other tools such as gpasm are very basic.

Unsurprisingly, we have settled on the Microchip tools because they are still very good, easy to use and free – a commercial version is available, but all that adds is some code generation optimisation that isn't really necessary anyway.

So let's go ahead and start playing with some code. The only software tool you require is MPLAB 8, which can be downloaded for free from the Microchip website. Be sure to download MPLAB 8 and not MPLAB-X; MPLAB-X is the new development environment from Microchip and there are still some teething troubles with it, so we will avoid using MPLAB-X until it has become more mature.

The source code for our development board can be downloaded from the magazine's website, in the file template.zip. Remember, this is a 'template' for software built around our development board, and specifically the PIC18F27J13 processor. This is not a generic template for any PIC processor project (although it should be simple enough to modify for other processors.)

Source code layout

The source code is organised as shown in Fig.1. The .mcp .mcs and .mcw files are the MPLAB project files, which specify things such as what source files are included in the project, what windows were open and what dialog settings were applied in MPLAB when last run. You do not need to concern yourself with the content of these files; MPLAB takes care of them. The remaining files are our actual source files, and are where we will make changes.

config.inc – This 'include' file contains the list of processor config bit settings, which define amongst other

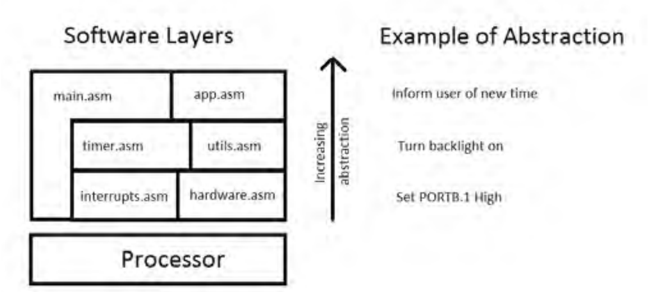


Fig.2 Program abstraction, showing how the source files build up on one another

things, what oscillator configuration we are using. Config bit settings are a common problem area; getting them wrong results in a non-working processor, which can be hard to debug.

hardware.asm – This file provides routines to configure the default state of the I/O port pins. Most of our pins are left as general purpose I/O, so there isn't much to do in this file.

timers.asm – This file provides the setup code for the timer subsystem, providing a number of general-purpose delay functions. Timers are difficult to set up and are generally always required (try flashing an LED without some kind of timer) so it's important to have some example implementation available.

interrupts.asm – The interrupt subsystem is one of the most complex within the PIC processor, requiring a number of different topics to be understood. This file provides a basic, working interrupt system as a 'starting point' for other interrupt handlers.

lcd.asm – This file provides example code for driving an external standard 2×16 character LCD. Although not essential to the operation of the development board, an LCD is such a common addition to a PIC project that it is included here for convenience. Unlike 'normal' LCD drivers, however, this code has a separate section for defining which pins are used to interface to the LCD, as no assumption can be made about whether an LCD display is in use or not, or how it is connected.

main.asm – The startup code goes in here; the code that is executed as soon as power is applied to the processor. This is typically called to initialise the hardware, and then a 'main loop' which provides the body of your project's code.

Ideally, main.asm will only hold the *calls* to our application's code, not the full source code. You should keep the source code in one or more separate source files, which will be included in the MPLAB project (as will be mentioned next month.) This way, main.asm will contain the 'high level' logic of your application, and will therefore, hopefully, be somewhat easier to read and understand. We refer to this as 'abstracting' the software away from the complexities of the underlying hardware. Fig.2 shows how this is done, using our kitchen timer as an example.

Layering the software implementation makes it easier to think about the problem you are trying to solve, thinking in what is called the 'application domain'. Imagine trying to write a payroll program if you had to think about whether the addition of two 8-bit numbers resulted in a carry or not!

Creating a project

Now we have the template software defined, let's go ahead and create a project with it. We are going to create a simple kitchen timer application that will allow the user to select a time, in minutes, after which a buzzer will sound. The time will be displayed on a two-line LCD.

This is the sequence we should follow:

1) Copy template files to new directory.
2) Build the code, download it to the board and check that the LED flashes. (This proves we haven't broken anything, which would be very difficult to do!)
3) Customise the code, testing on a prototyping board. Adding new source files where sensible.
4) Copy the code to a controller board fitted to real hardware.
5) Final test.

Example project – kitchen timer

The kitchen timer uses an LCD, two buttons and a small buzzer for sound output. We will use the processor's PWM module to provide the audible tone signal, which will require us to customise the lower level hardware module in the template. Fig.3 shows the circuit (drawing the development board as a self contained 'black box' module) and Fig.4. shows the prototype hardware layout. Notice that we are using a single button, even though there are two in the design – we simply moved it between the two connection points during testing.

This is to be a battery-powered device, so the focus will be on low power operation. The LCD will be switched off automatically when not in use or counting down the time.

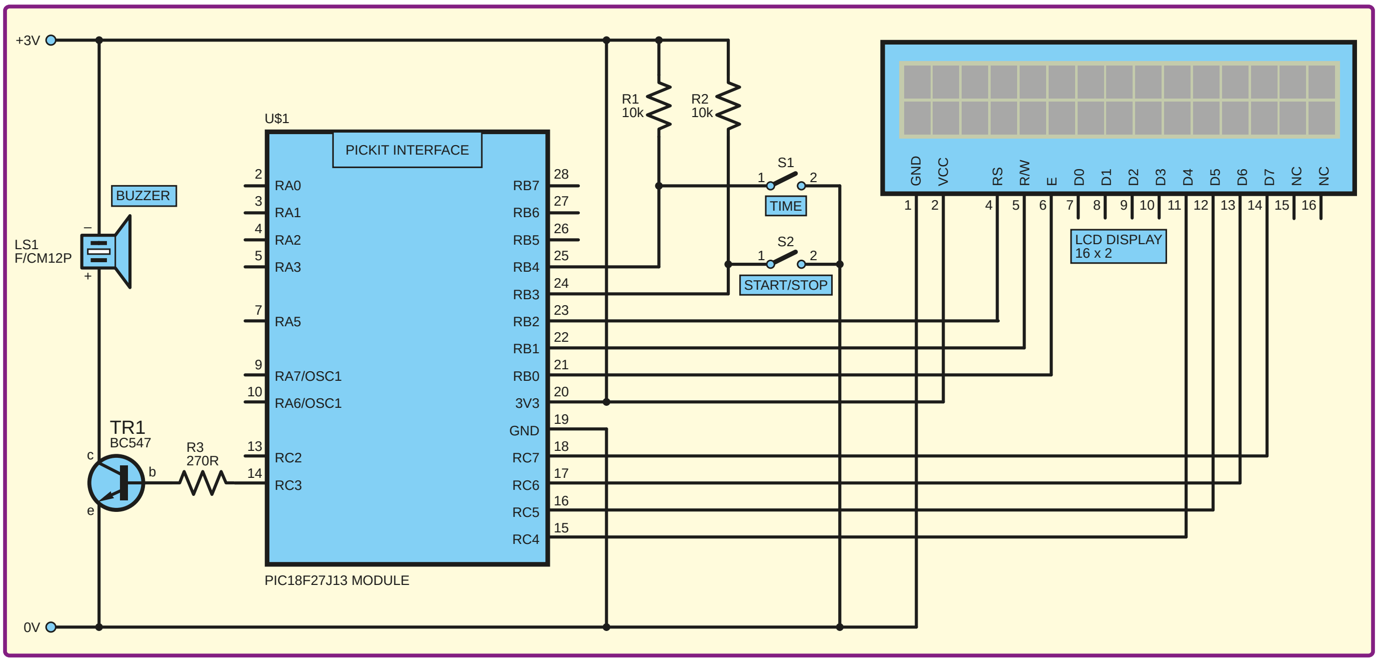


Fig.3 Kitchen timer circuit

The buzzer will sound, and continue until a button is pressed. Any keypress will 'wake up' the display; one button will cycle through different timer periods, and the other button will start the timer. Any button press will stop the timer if it is currently running. The display will time out after one minute of inactivity (when the timer is not running.)

Next month

We will complete the software for the kitchen timer next month, and move it away from the prototype board and into real use. We will also start our move away from assembly language programming and into the higher level programming language, C, while maintaining the same hardware platform. We show how using a higher level language simplifies the task of writing software, and makes it easy to expand on our simple kitchen timer design.

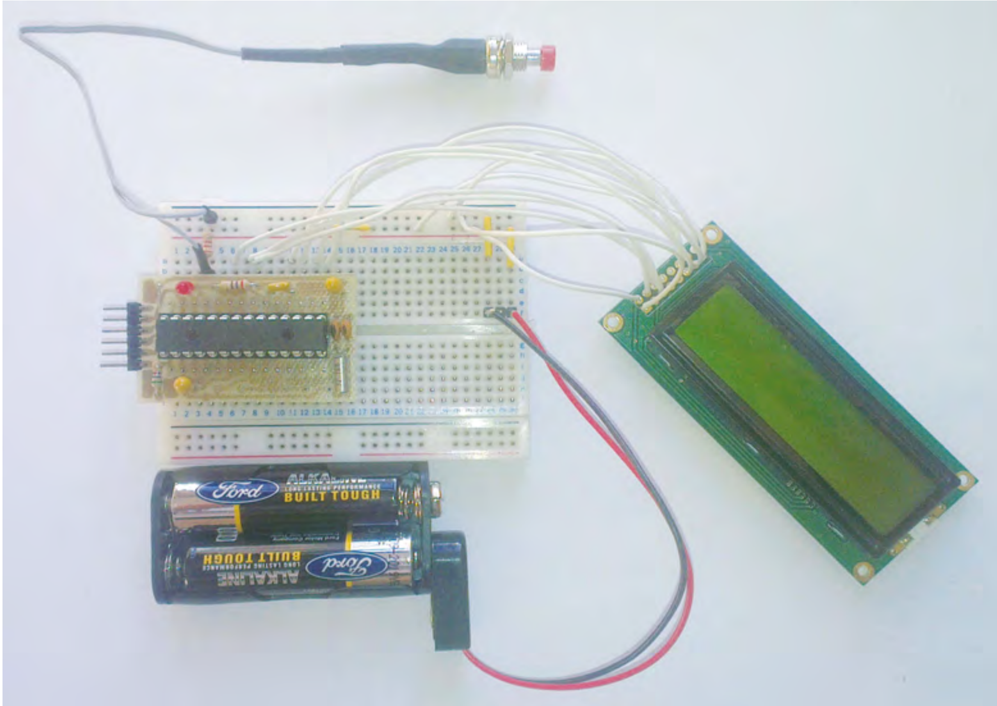


Fig.4 Kitchen timer development using a breadboard


THE No 1 UK MAGAZINE FOR ELECTRONICS TECHNOLOGY & COMPUTER PROJECTS
EPE
EVERYDAY PRACTICAL ELECTRONICS
www.epemag.com
USB MULTI-INSTRUMENT PROJECT
PC controlled
2-channel digital scope
Spectrum analyser
DMM and frequency counter
Audio function generator
CLASSiC-D AMPLIFIER
CONSTRUCTING OUR HIGH-POWER HIGH-PERFORMANCE AMP
WIN A MICROCHIP REMOTE CONTROL DEMO BOARD WITH ZENA WIRELESS ADAPTOR
Teach-In 2014 Raspberry Pi – Part 3
EPE's comprehensive guide to Raspberry Pi
DEC 2013 £4.40
PLUS: NET WORK, READOUT, CIRCUIT SURGERY, TECHNO TALK, PIC N' MIX & INTERFACE

By Robert Penfold

Raspberry Pi GPIO and CSI

IN an earlier *Interface* article about the Raspberry Pi computer I mentioned that some of the commands associated with the add-on for the GPIO port did not seem to work correctly. In particular, simultaneously setting a GPIO line as an output and an initial output state did not seem to work. I had similar problems using the commands associated with interrupts. The cause of the problem and its solution are both quite simple.

Behind the times

If you download and install the latest version of the Raspian operating system it would seem reasonable to assume that you have the latest version of the operating system and the various extras that come with it. However, things do not operate in such a simple fashion with a complex piece of software such as Raspian. Individual parts of the system are updated much more frequently than the full installation. This means that a new installation of Raspian might not be sufficiently up-to-date to handle the latest additions to the GPIO software add-on.

The solution to this problem is to update the individual parts of the operating system and the additional software packages that are supplied with it. This is easy provided you can get your Raspberry Pi connected to the Internet. It will be necessary to issue instructions from a command line, but Raspian will probably boot into the GUI rather than a text screen with a command line. However, a command line can be obtained by running the LXE terminal program. First use this command to make sure that the latest version of the application installation program is installed:

```
sudo apt-get update
```

The operating system and its add-ons can then be downloaded and installed using this command:

```
sudo apt-get dist-upgrade
```

The download could be quite large, and was well over 200 megabytes when I updated the operating system of my Raspberry Pi. With a broadband connection it will probably take two or three minutes to download the file containing the update, but it could then take the best part of an hour for all the changes to be made. Anyway, at the end of the process you will have a fully up-to-date Raspian operating system that should be able to run all the latest commands in the GPIO module.

Interrupting

The basics of using the GPIO lines as inputs has been covered previously, but this coverage did not include using GPIO inputs to generate interrupts. Various interrupt options are available from a recent version of the GPIO add-on software, but for most purposes it is only the basic ones that are needed. These simply wait until a certain type of change is detected on the monitored input, and then a certain action is performed by the program. In other words, a hold-off is provided until a suitable trigger signal is detected on the appropriate GPIO line.

Of course, it is not essential to use interrupts in order to achieve a hold-off of this type. The same thing can be achieved using a software loop that repeatedly reads an input until it goes to a certain logic level, and then exits the loop. The problem with this method is that it tends to use a lot of computing power to 'mark time' and achieve nothing more than a simple hold-off.

Using interrupts is a much more efficient method. This system is reliant on extra hardware in the microprocessor that enables external devices to indicate that they require the attention of the computer. The important point here is that the microprocessor does not use software to repeatedly read the interrupt inputs as in the polling method. Interrupts are largely handled by hardware.

The computer goes about its normal tasks and ignores the interrupt inputs until one is activated. Hardware in the microprocessor then detects that an interrupt has been generated, and breaks the microprocessor out of its normal routine. The interrupt is then serviced, after which the computer continues where it left off. The interrupt method ensures that the amount of processing power used to service peripheral devices is kept to a minimum.

The GPIO software add-on makes it easy to use interrupts with GPIO inputs. When used with GPIO interrupts the program provides a hold-off until an appropriate transition is detected on the specified input line. An important point to note here is that it is a transition that triggers the interrupt, and not the input going to a particular logic state. In other words, it is the input going from one logic state to another that generates an interrupt. Three types of transition can be detected (Table 1).

Table 1

Transition type	Action on input
RISING	A transition from low to high
FALLING	A transition from high to low
BOTH	A transition from low to high or high to low

In order to make the program wait for a high-to-low transition on pin 8, this instruction would be used:

```
GPIO.wait_for_edge(8, GPIO.RISING)
```

These instructions are the equivalents for respectively detecting falling edges or both types:

```
GPIO.wait_for_edge(channel, GPIO.
FALLING)
GPIO.wait_for_edge(channel, GPIO.
BOTH)
```

It should be noted that these instructions require the RPi.GPIO package to be version 0.5.1 or later. Using earlier versions you will get an error message to the effect that Python does not understand the `wait_for_edge` part of the instruction. The short program in Listing 1 can be used to test an edge triggered instruction.

After the usual preamble, pin 8 is set as an input using an internal pull-

Listing 1

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(8, GPIO.IN, pull_up_down=GPIO.
PUD_UP)
GPIO.wait_for_edge(8, GPIO.BOTH)
GPIO.cleanup()
print ("Finished")
```

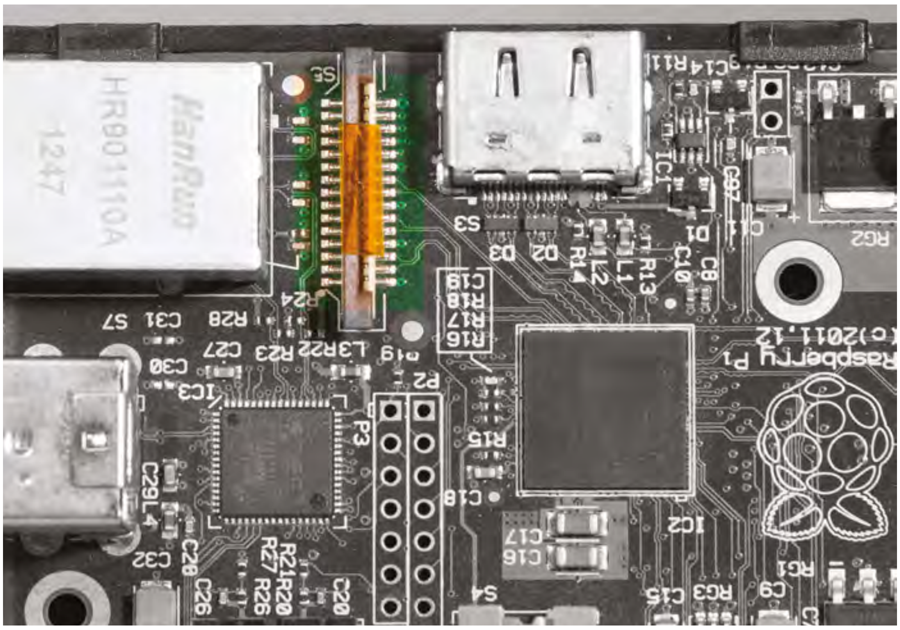


Fig.1. The CSI port is situated between the Ethernet and HDMI connectors. It is a form of edge connector that accepts the stiffened end of the cable that is supplied ready fitted to the camera

up resistor. The next line sets pin 8 as an edge-triggered input that will respond to a rising or falling edge, and it halts the program until a suitable edge is detected. In order to set a pin as an edge-triggered input it must first be configured as an input and then set for edge-triggered operation, as in this example. It is not good enough to just set it for edge-triggered operation without setting it up as an input first. The last line of the program prints 'Finished' on the screen, but this will only happen once a transition on pin 8 has been detected. Connecting pin 8 to ground via a resistor of a few hundred ohms will generate a transition and end the program.

Normally it is possible to break out of a Python program that is in an indefinite loop, or has simply not reached the end yet, by using either Control-C or Control-Z from the keyboard. Neither of these key combinations will fully end the program if it is waiting for a transition on a GPIO input. The program will end only when a suitable transition is detected on the input. It will then end completely and provide the usual on-screen message explaining where in the program it has stopped and why. If you use an edge triggered input it is important to ensure that the program will not get stuck waiting for an edge-triggered interrupt that will never happen. Of course, as a last resort you can simply close Python if a program should get stuck in this way.

CSI

CSI in a Raspberry Pi context has nothing to do with crime scenes; it stands for Camera Serial Interface. The CSI is situated between the Ethernet and HDMI connectors (Fig.1), and it perhaps looks more like a SIL component of some kind rather than a normal computer port. It is primarily intended for operation with a little camera board (Fig.2) that can handle up to 5-megapixel still images, plus a few video formats. The camera connects to the Raspberry Pi board via a miniature ribbon cable that is supplied with the camera, and should already be connected to it. Like the electronics in the Raspberry Pi board itself, I think it is safe to say that the camera is based on technology that was produced for use in mobile telephones. Whatever the origins, its performance is very good for such a tiny add-on that is also reasonably inexpensive.

The CSI port is not enabled by default, which is presumably to prevent it from unnecessarily hogging the computer's resources. Anyway, in order to use the camera it is necessary to have an up-to-date version of the Raspian operating system installed, and the CSI port must be enabled via the configuration utility. An update is required if the configuration program does not include an option for switching on the CSI port.

The camera board is a popular add-on, and it sold out almost overnight when it was first introduced. It certainly offers some interesting possibilities, and it adds plenty of play value to a Raspberry Pi system. There is no software module available to enable the camera to be directly controlled via a programming language such as Python. However, there is a way around this in the form of a free command line utility that gives excellent control over the camera. It enables video clips or still pictures to be taken, and it also gives some control over things such as white balance, picture resolution and contrast.

Most programming languages, including Python, can be used to issue commands to the operating system. The program that controls the camera can be called from within a Python program, so the camera can actually be used under software control via this slightly indirect route. Using this method the camera can therefore be triggered automatically via suitable hardware on the GPIO port. Many of the sensors featured in previous Interface articles can be easily modified for this purpose. I have not tried it, but files generated by the camera can be loaded into a Python program. For those suitably skilled at Python programming it should then be possible to implement things such as movement detection and basic pattern recognition.

Connecting the camera

As supplied, you get the camera with attached cable, and an anti-static bag to protect the electronics of the camera. The normal anti-static precautions are needed when handling the camera. In normal Open Source fashion, there is no documentation other than a few statutory warnings. However, installing and using the camera is reasonably straightforward.

The simple connector used for the CSI is not polarised, and due care has to be taken to ensure that the camera is connected to the port with the ribbon cable fitted the right way around. Fortunately, the connectors are single-sided, and there is little risk of damage occurring if you should get it wrong. Anyway, the cable should be fitted with the plain side towards the Ethernet connector and the side having the metal tracks facing towards the HDMI port (Fig.3).

This operation is a bit fiddly, and there will probably be a thin plastic flap covering the port. This has to be bent to one side to give proper access to the connector, and it might simply break away from the connector. The top of the connector is darker in colour and has a tiny tab at each end. This top section can be raised slight-

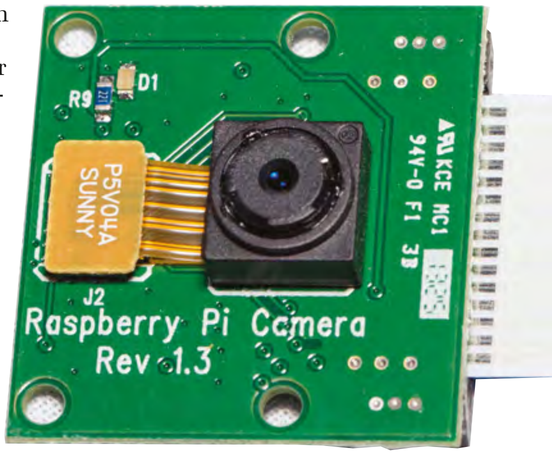


Fig.2. This shows the lens side of the camera board, which is tiny. In fact, it is not much bigger than a normal-sized postage stamp

Fig.3. The camera connected to the Raspberry Pi. The cable must be connected to the Raspberry Pi with the plain side towards the Ethernet port, and the silver connectors facing towards the HDMI connector

ly by pulling on these tabs, and this opens the connector (Fig.4). It should then be possible to push the cable fully down into the connector on the Raspberry Pi board. Push the cable straight down holding it as near to the end as possible so that any severe flexing is avoided. With the cable in place, push the top section of the connector back down so that it is closed and the cable is locked in place. Finally, check for a plastic covering over the lens. If present, this should obviously be removed before you start using the camera!

There are two command line instructions that can be used to control the camera. These are `raspivid` and `raspistill`, which respectively use the camera to take video clips and still pictures. There is actually a third command called `raspistillyuv`, which seems to be for capturing still images as raw data from the camera, and is rather more specialised than the other two commands. It is probably best to use `raspistill` when initially testing the camera. The `raspistill` command can be followed by various options that govern the way in which pictures are taken, but most of these are optional. This is the basic command for taking a picture:

```
raspistill -o picture.jpg
```

This will take a picture in jpeg format and store it on the SD card using 'picrture.jpg' as the filename. The `-o` part instigates the output option, and it must be followed by the filename to be used for the image file. The `–t` (timeout) option is a useful one, and this sets a delay between issuing the `raspistill` command and a picture actually being taken. During this delay the output from the camera is displayed on the monitor, and this gives the user an opportunity to frame the picture accurately. A default delay of five seconds is used unless a delay time is specified in the command.

Next time we will move on to automatic triggering of the camera.

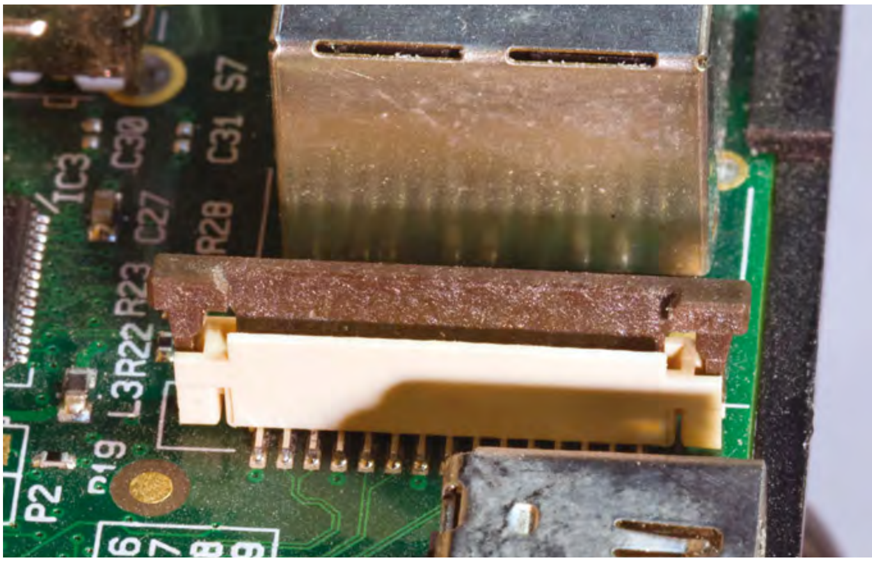


Fig.4. Open the CSI connector by raising the top section slightly. The cable should then fit in quite easily, and the top of the connector can then be pressed down in order to lock the cable in place

EPE PIC RESOURCES CD-ROM V2

ONLY £14.75
INCLUDING VAT and P&P

Version 2 includes the EPE PIC Tutorial V2 series of Supplements (EPE April, May, June 2003)

The CD-ROM contains the following Tutorial-related software and texts:

- EPE PIC Tutorial V2 complete series of articles plus demonstration software, John Becker, April, May, June '03
- PIC Toolkit Mk3 (TK3 hardware construction details), John Becker, Oct '01
- PIC Toolkit TK3 for Windows (software details), John Becker, Nov '01

Plus these useful texts to help you get the most out of your PIC programming:

- How to Use Intelligent LCDs, Julyan Ilett, Feb/Mar '97
- PIC16F87x Microcontrollers (Review), John Becker, April '99
- PIC16F87x Mini Tutorial, John Becker, Oct '99
- Using PICs and Keypads, John Becker, Jan '01
- How to Use Graphics LCDs with PICs, John Becker, Feb '01
- PIC16F87x Extended Memory (how to use it), John Becker, June '01
- PIC to Printer Interfacing (dot-matrix), John Becker, July '01
- PIC Magick Musick (use of 40kHz transducers), John Becker, Jan '02
- Programming PIC Interrupts, Malcolm Wiles, Mar/Apr '02
- Using the PIC's PCLATH Command, John Waller, July '02
- EPE StyloPIC (precision tuning musical notes), John Becker, July '02
- Using Square Roots with PICs, Peter Hemsley, Aug '02
- Using TK3 with Windows XP and 2000, Mark Jones, Oct '02
- PIC Macros and Computed GOTOs, Malcolm Wiles, Jan '03
- Asynchronous Serial Communications (RS-232), John Waller, unpublished
- Using I2C Facilities in the PIC16F877, John Waller, unpublished
- Using Serial EEPROMs, Gary Moulton, unpublished
- Additional text for EPE PIC Tutorial V2, John Becker, unpublished

NOTE: The PDF files on this CD-ROM are suitable to use on any PC with a CD-ROM drive. They require Adobe Acrobat Reader – included on the CD-ROM

EPE PIC RESOURCES V2 CD-ROM ORDER FORM

Please send me .... (quantity)
EPE PIC RESOURCES V2 CD-ROM

Price £14.75 each – includes postage to anywhere in the world.

Name ....

Address ....

....

....

.... Post Code ....

☐ I enclose cheque/P.O./bank draft to the value of £ ....

☐ please charge my Visa/Mastercard/Maestro £ ....

Card No. ....

Card Security Code ....
(The last 3 digits on or just under the signature strip)

Valid From .... Expiry Date ....

Maestro Issue No. ....

SEND TO:
Everyday Practical Electronics,
Wimborne Publishing Ltd.,
113 Lynwood Drive, Merley, Wimborne,
Dorset BH21 1UU.
Tel: 01202 880299. Fax: 01202 843233.
Email: orders@epemag.wimborne.co.uk

Payments must be by card or in £ Sterling – cheque or bank draft drawn on a UK bank.

Normally posted within seven days of receipt of order. Send a copy of this form, or order by letter if you do not wish to cut your issue.

Order on-line from
www.epemag.com
or by Phone, Fax, Email or Post.

BECOME A PIC WIZARD WITH THE HELP OF EPE!

READOUT

Matt Pulzer addresses some of the general points readers have raised. Have you anything interesting to say?

Drop us a line!

All letters quoted here have previously been replied to directly

Email: editorial@wimborne.co.uk

★ LETTER OF THE MONTH ★

Rotary encoders – mechanical origins

Dear editor

Ian Bell's recent pieces on rotary encoders were most enjyable. I attach a photo of a secondary-surveillance radar encoder from my aircraft museum. Of 1960s vintage, Moore & Reed crammed two mechanical-contact rotating discs into the encoder. It has to resolve 100ft increments from zero to perhaps 40000ft (giving flight level, as displayed on the controller's radar screen).

This encoder was incorporated into a Smiths servo altimeter, as fitted to the Hawker Siddeley Trident. An air data computer (a wondrous-looking electro-mechanical analogue contraption) sensed the static pressure. This produced a signal from a synchronous data transmitter (synchro), itself a rotating electromechanical device and nothing to do with serial digital data transmission, despite the suggestive name.

The altimeter has a synchro receiver that compares the incoming demand with present position and produces an error signal, fed to a servo amplifier and servomotor in order to turn the display.

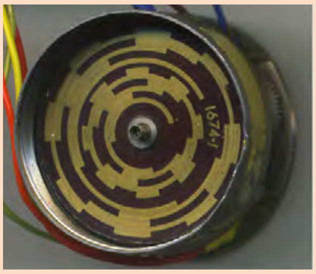
Mechanically slaved to the display is the shaft encoder, as in the photo. This produces an output corresponding to flight level (FL) altitude in 100ft increments when the altimeter is set to standard pressure (1013.25hPa). The FL output then supplies the secondary surveillance radar transponder, by which means the air traffic controller can see the aircraft's flight level displayed on the radar screen.

The altimeter reads to 99950ft and so requires at least 1000 distinct codes (not that airliners fly anywhere near that height), which means at least 10 binary bits. The usual code is known as Gillham, after the inventor who extended a version of Gray code for this purpose. In the photos, assuming the usual scheme adopted by Moore & Reed, the fast disc rotates once every 8000ft and the slow disc is geared down 1:16 from this. Unfortunately, these items aren't readily available, so I haven't another one to dismantle for more experiments. I suspect that there is some further internal pre-encoding before the sequence reaches the outside world, because the device also contains some diodes.

I've decided not to try to reverse-engineer the code that comes out of 13 wires, it's a task more suited to Bletchley Park! The first question would be to know when the shaft is at the home or zero altitude position; perhaps one of the wires signals this special case.

My description sounds like a system invented by Heath Robinson! Unfortunately, it's real and little changed in principle to this day, except that data between the instruments is now more likely to be digital than from a synchro.

Godfrey Manning, Edgware



Matt Pulzer replies:

*Fascinating stuff Godfrey! Thank you very much for your letter and explanation. You may well be aware of the very talented (and magnificently eccentric) Tatiana van Vark (**www.tatjavanvark.nl**) who delights in all things mechanical. If you hunt around on her website you will find the navigation and bombing system from a Vulcan that she has beautifully restored. Take a few minutes to look at: **http://youtu.be/w1ODwgX_KkQ**. She actually made many of the machines on show there.*

Theremin conundrum

Dear editor

I'm writing with regard to a constructional article that appeared in *EPE*, March 2011, written by John Clarke – *New, Improved Theremin*. I built this unit, but came across an apparent inconsistency in the article, which has prevented the successful completion of the project. Perhaps Mr Clarke could be made aware of this and clarify where necessary – thus enabling me to achieve proper operation of the circuit.

My questions arise when attempting to follow the 'Setting Up' procedure starting on page 20, where it says: 'Using a suitable alignment tool, wind the ferrite slug in T2 clockwise until there is a resistance in movement. Then count the number of turns to wind it out anticlockwise completely.' No problem there, and as they say – 'Righty tighty, lefty loosie'.

But then in the next section, 'Volume Alignment', it goes on to say: 'Wind the slug for T3 anticlockwise and then OUT again, counting the number of turns. That done, set T3 about 30% of turns anticlockwise.'

What's confusing here is the use of the phrase 'Out again', which doesn't make sense, thereby obfuscating the actual meaning of '30% of the turns anticlockwise'. Now I'm wondering if I'm using the right IF transformer and the correct one to use has a reverse thread in it or something. I tried setting the device to both of the possible settings (30% anticlockwise and 30% clockwise) to little avail. I suspect that either I'm missing something, or there's another error in the article that I haven't detected.

I looked in the forums on your website and couldn't find any significant mention of this project anywhere. Can someone please help me with this?

RT Bennett, by email

John Clarke replies:

Regarding the Volume Alignment. The statements can be clarified as such:

Wind the slug for T3 anticlockwise until there is a resistance in movement. Then wind the slug for T3 clockwise and count the number of turns made until there is a resistance in movement again. That done, set T3 to about about 30% of the total number of turns anticlockwise from the full clockwise position. So, for example, if there are 10-turns of the slug from fully anticlockwise to fully clockwise, set the slug some 3-turns anticlockwise from the fully clockwise position.

Software options

Dear editor

As a subscriber to *EPE* I was interested to read your editorial in the August issue. It would seem your magazine is locked into production software running on Windows. Adobe and Microsoft do like their 'pound of flesh'! So, why not consider OpenSource software as an option? Think of how much you would save on production costs – for example:

- Linux as the operating system
- Gimp for photo manipulation
- Inkscape or Eagle for drawing circuits
- Scribus for desktop publishing – you can export directly into PDF format from this program.

I notice the magazine is very Windows oriented, and would like to see more content using OpenSource options. I suspect there are plenty of readers who already run Linux as their main operating system, and would appreciate you including it in some of your articles.

I have been using Linux for very many years, and can usually find a program to do what I want without too much difficulty. I will be interested to hear your comments.

On a personal level, I like the new MicroChip software being based on Netbeans, with which I am very familiar, since I use it for Java and PHP programming. It's handy being cross platform.

The Raspberry Pi is great fun, isn't it? There is so much you can do with it. I enjoy helping my son with a Pi-based project for his work. There is life in the 'Old Dog' yet!

Francis Greaves, West Cork, Ireland

Matt Pulzer replies:

There are a number of reasons we – and just about any publisher you ask – do not go down the OpenSource route.

First, PDFs for industrial printers are not the straightforward files we all know and love. There are very many settings that need to be right every single time. Adobe does have a very good track record in supporting the industry and if we were to have any problems and told a printer that our software was Scribus they would just laugh at us – harsh, but true. The publishing industry is based around a few (expensive) programs like InDesign and Quark, and since these work it would be a brave production manager who opted for anything else. When you are a running a publishing house 'safe, expensive but reliable' trumps 'cheap but unknown' every time.

Second, the CS suite is heavily integrated. Illustrator files can be simply dropped into InDesign, no ifs, no buts.
Third, working across multiple programs that work in very similar ways is an important plus point.

Fourth, I actually use a Mac system, not a Windows one and I can send across my CS files to Windows-based CS applications at Wimborne safe in the knowledge that the files will work – every time.

Fifth, I actually like Adobe products and I know how to use them. I just don't like Adobe's commercial practices and I don't want to learn a whole new set of apps.

It is true, EPE *is very Windows oriented, which is simply a reflection of the world we live in. I'm sure there are plenty of Linux users (and Mac users) out there who read* EPE*. But the numbers are dwarfed by Windows users. I'm no fan of Windows, but our readership overwhelmingly uses it.*

I appreciate to some extent this is a chicken and egg problem – Windows remains top dog because everyone sticks to it, including us. But we are here to provide good content to readers, not to support an OS revolution!

That said, despite our current interest in Raspberry Pi, we are mostly a PIC-oriented publication, and MicroChip support all three main OS platforms with their software tools. So, with a little extra hard work, many of our projects and articles are open to Linux users/programmers. I hope that answers your interesting questions.

Weighing light

Dear editor

I am sorry to say that Max is not so 'magnificent' when he states in the July 2013 issue of *EPE* that the photon is massless.

It is well known in cosmology circles that the humble photon is bent around gravity wells in the gravity lensing effect. If it is affected by gravity then surely it has mass, albeit very small.

Maurice King, by email

Max the Magnificent replies:

I am delighted to be able to reassure you that my magnificence remains intact. Whether or not photons have mass and the way in which they are affected by gravity caused scientists to scratch their heads for many years. Eventually, Albert Einstein figured it out with his theory of General Relativity*. Gravitational fields change the shape of space-time. Strong gravitational fields – such as those caused by stars, galaxies, and black holes – really change the shape of space-time.*

Photons have momentum, but not mass. The reason photons bend around gravity wells in the gravitational lensing effect is not that they are affected by gravity per se*; instead, they are responding to the curvature in the space-time continuum. If photons did have mass, they would speed up as they approached a gravity well and slow down as they receded from one, but they don't – they just keep on travelling at the speed of light.*

Quite apart from anything else, the fact that photons do travel at the speed of light means that they must be massless, otherwise they could never move at the speed of light in the first place. This is because the energy required to move anything with mass approaches infinity as that mass approaches the speed of light.

EPE Mini Lab

Dear editor

I hope you can help me. I am looking for a PCB for the *EPE Mini Lab* from 1993. Help me please!

Carelo, by email

Alan Winstanley replies:

Thanks for your enquiry. Both the Mini Lab *and* Micro Lab *were developed for* Teach-In 93 *by a firm of electronics engineers who co-wrote the series with me. The* Mini Lab *PCB foil pattern was never published due to copyright limitations.*

A major benefit was that we could exert quality control over the PCB and silk-screen print, and with sales exceeding 1,000 units we did not have a single reader problem caused by poor construction or errors. This was important because we were dealing with youngsters and beginners in electronics.

I will shortly revisit the Mini Lab *and* Micro Lab *for my own website, as I still have the original prototypes. However, some devices, including the ZN414Z radio chip and ICL8038 signal generator are no longer easily available.*

I am sorry if this causes any disappointment, but I really appreciated your interest in the Teach-In Mini Lab.

Circuit Surgery

Regular Clinic

By Ian Bell

Synchronous and asynchronous counters

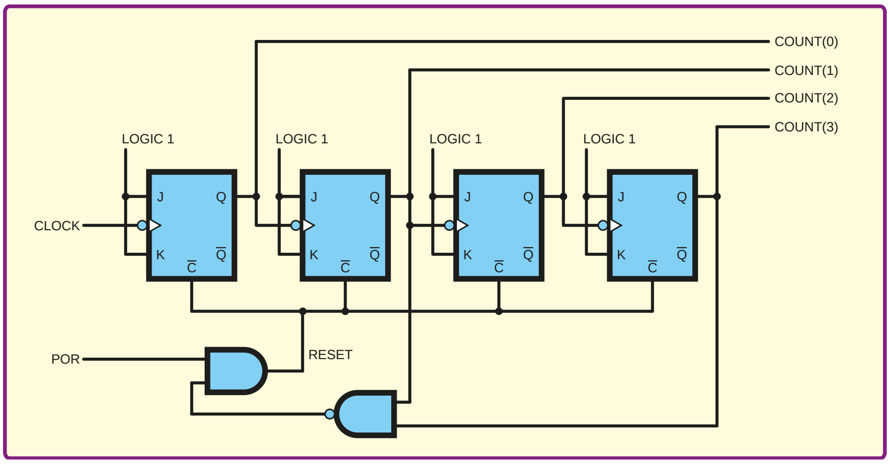


Fig.1. Four bit, asynchronous, decade ripple counter

LAST month, we looked at some issues which arose from the simulation of a binary counter. A question was posted by ***perro*** on the *EPE Chat Zone* asking if the unwanted behaviour of the circuit was due to a problem with the *Circuit Wizard* simulation he had used, or was something to do with the circuit itself. ***Perro's*** circuit was an asynchronous ripple counter similar to the one presented in *Teach-In 2011* Part 6 (Fig.6.37) (*EPE* April 2011), but with a change of component which may caused it to behave differently from the original. The circuit we looked at last month was asynchronous and this month we will look at a synchronous version We will also discuss the way both types of circuit function in a bit more detail. First, though, a quick recap of the discussion from last month.

The version of ***perro's*** circuit we discussed, which excludes the seven segment display from his simulation, is shown in Fig.1. It uses a NAND gate to detect when the counter's output value reaches ten (binary 1010) and 'immediately' resets the counter to 0000. The reset is not actually immediate because of the response time of the gates and flip-flops, but would not be detectable by a human looking at a numerical display. The result is (or should be) that the counter counts from zero (0000) to nine (1001) and so can provide one digit of a decimal number. ***Perro's*** circuit failed to do this, with the counter going to 0100, not 0000 after reaching nine.

Logic race

Last month, we discussed the simulation of this circuit and the reason for the wrong result, which related to the time delay inherent in all the gates in the circuit, including those inside the flip-flops. In this circuit we have the possibility of a race condition, which can occur when a signal can take two (or more) paths through a circuit from A to B. In a race, the behaviour of the circuit depends on the order in which the effect of the change at A reaches B via multiple paths.

In this case, the reset, triggered in part by the 1 on the count(1) output (our point A), causes two things to happen inside the count(2) flip-flop, at its internal output latch (our point B). It causes itself to be removed because the counter no longer outputs the value detected by the NAND gate. Also, the count change from 1010 towards 0000 creates an active clock edge (1 to 0 change) on the count(2) flip-flop. If the reset is removed from this flip-flop's internal latch before the clock edge gets through to the same latch, the clock edge will cause the flip-flop to toggle. Its output will change to 1 rather than remaining at 0 as required.

Last month we demonstrated how changing the relative time delays of gates in the circuit could determine whether this problem occurred or not

A nasty implication of this type of problem, particularly for commercial designers, is that individual copies of the same circuit may behave differently. Furthermore, because gate delay depends on supply voltage and temperature, the circuit may behave differently under different operating conditions. Advanced logic design tools can help identify such timing ambiguities, but it is best to avoid the possibility by design if possible.

Other problems

In addition to this timing uncertainty, the circuit in Fig.1 has other potential problems. Last month we also looked at how the output can go through intermediate output values as changes ripple through the circuit from one flip-flop to the next (see Fig.2). Although this happens too fast for a human to detect it could cause false triggering of another circuit.

The point of the reset in the circuit in Fig.1 is to create a base-ten-oriented digital counter from a fundamentally binary or hexadecimal (four-bit

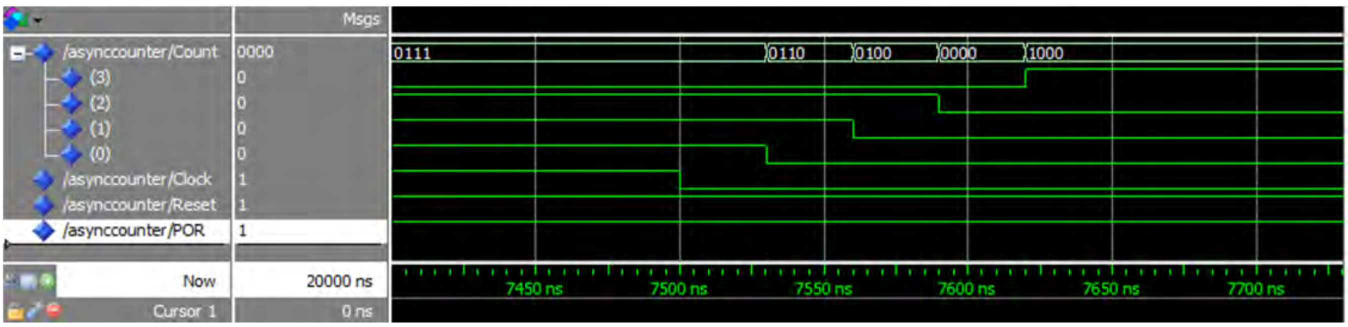


Fig.2. Details of waveform for counter in Fig.1 as it increments from 0111 to 1000. Note the intermediate values

binary) circuit. Such circuits are useful when the count value needs to be displayed to people directly because there is no need for additional circuitry to decode a binary value into decimal first. However, reseting a counter at a specific value is not restricted to this application and this general approach is used widely in digital circuit design. In some cases, the number at which the counter resets is programmable, rather than being hardwired to a specific number, as it is in Fig.1. It is, therefore, useful to be able to do this in a reliable way, so in this article we will look at some other circuits for counters, specifically those with synchronous rather than asynchronous operation.

A counter effectively performs the action of adding 1 to the number it currently holds. If we think about this in terms of long addition, then, if the least significant bit ('1s' digit) is 0 it will become 1 after the addition of 1. If the least significant digit is 1 then adding 1 will cause a carry of 1 to the next digit (the '2s') and the '1s' digit in the result will be 0 (1 + 1 in binary is 10, which is '0 and carry the 1' in long addition terms). The first bit will therefore always toggle (change 0 to 1 or 1 to 0) when the counter counts. The flip-flops in the circuit in Fig.1 are configured to always toggle when they are clocked (J=K=1), so this requirement is met for count(0).

If the first bit starts at 1 the resulting carry will add 1 to the second digit, so it will toggle, carrying a 1 to the third digit. This process will continue through all the digits until a 0 is reached. The digit which was 0 will toggle to 1, but changes will stop there because there is no carry from this digit to the next. A carry from a digit occurs with a 1 to 0 change on that digit. Thus a negative edge on one digit means the next digit must toggle. This is how the circuit of Fig.1 is structured – a negative edge on one flip-flip clocks the next device, causing it to toggle. The 'rippling' of a ripple counter corresponds to the propagation of the carry in a long addition in which we add 1 to the counter's current value. Each carry step takes a finite time, which accounts for the effect seen in Fig.2.

The synchronous route

The circuit is Fig.1 is described as asynchronous because the flip-flops do not all clock together – they are not synchronised. An alternative approach is to clock all of the flip-flops together, which is referred to as a synchronous circuit. If we think about a counter in this way, with reference back to a discussion on long addition, then at the occurrence of each clock the circuit has to determine whether or not a given digit (flip-flop) toggles. Each digit must either toggle or not toggle when clocked. This is different from the circuit in Fig.1, where all the flip-flops are hard-wired to always toggle when clocked. To build a synchronous counter we therefore need a flip-flop with a toggle/don't toggle control input, often called toggle/hold (T/H) or toggle enable (TE) input.

The symbol for a (negative-edge-triggered) toggle/hold flip-flop is shown in Fig.3. This is perhaps not one of the well-known classic flip-flop types (D, JK and RS) but it is an important building block in digital circuits. We can use JK flip-flops with J and K connected together, or a D-type with an XOR gate, to obtain a toggle/hold function (see Fig.4). Other toggle/hold flip-flops circuits are possible.

For the synchronous counter to work correctly all the flip-flops' TE inputs must be simultaneously at the correct value before the clock occurs. Rather than letting the addition carry ripple from bit to bit, in a similar manner to a person laboriously performing a long addition on paper, all the digits are updated together. This is really quite straightforward and follows the same argument as above. The first digit always toggles. The second digit toggles if the first digit is one due to the carry. The third digit toggles if the first two digits are one as the carry propagates through both. Fig.5 shows this pattern visually in a 3-bit binary sequence. In general, a digit toggles whenever all lesser significant digits are 1.

The circuit in Fig.6 is a 3-bit synchronous binary counter based on the arguments above. The first bit always toggles, so the first flip-flop has its toggle/hold wired to logic 1. The second bit (Q1) toggles if the first bit (Q0) is 1, so we connect Q0 to the Toggle Enable control of the second flip-flop. The third bit (Q2) toggles when both Q0 AND Q1 are 1, so we use an AND gate to obtain this function, connecting its output to the TE control of the third flip-flop. This circuit is synchronous because all the flip-flops are clocked together.

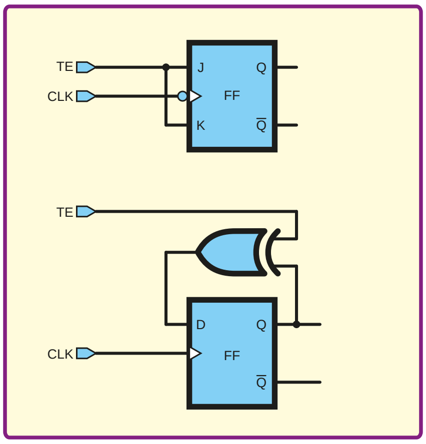


Fig.4. Forming a toggle/hold flip-flop from a JK or D-type flip-flop

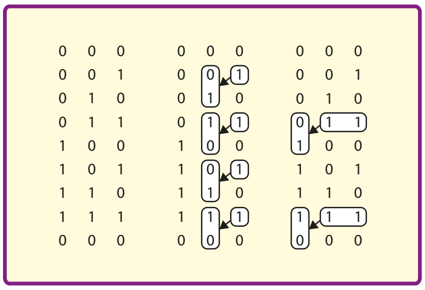


Fig.5. Patterns in binary counting. These relate to the carries which occur when adding 1 to the current count value in binary addition

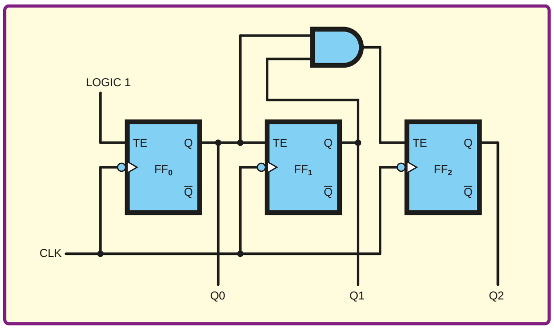


Fig.6. 3-bit synchronous counter using toggle/hold flip-flops

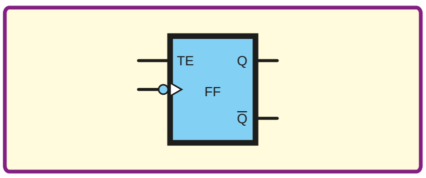


Fig.3. Toggle/hold (toggle enable) flip-flop symbol

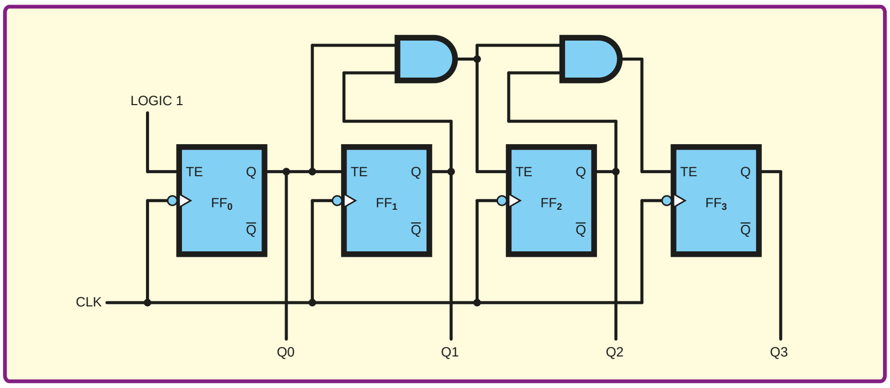


Fig.7. 4-bit synchronous counter using toggle/hold flip-flops

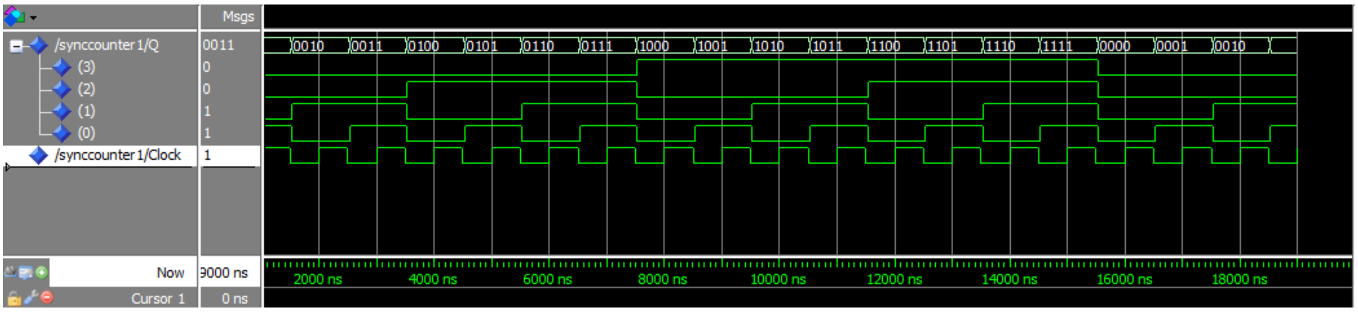


Fig.8. Simulation of the circuit in Fig.7

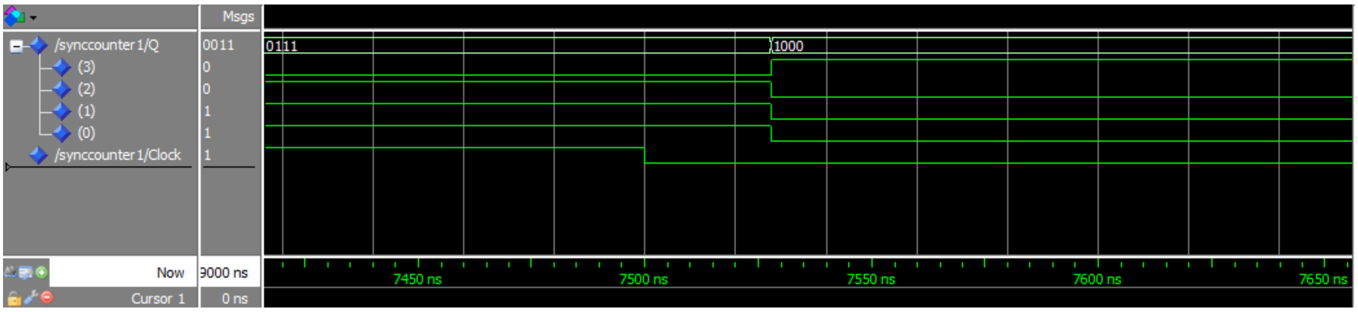


Fig.9. Zoom in on the 0111 to 1000 change from Fig.7. Compare with Fig.2

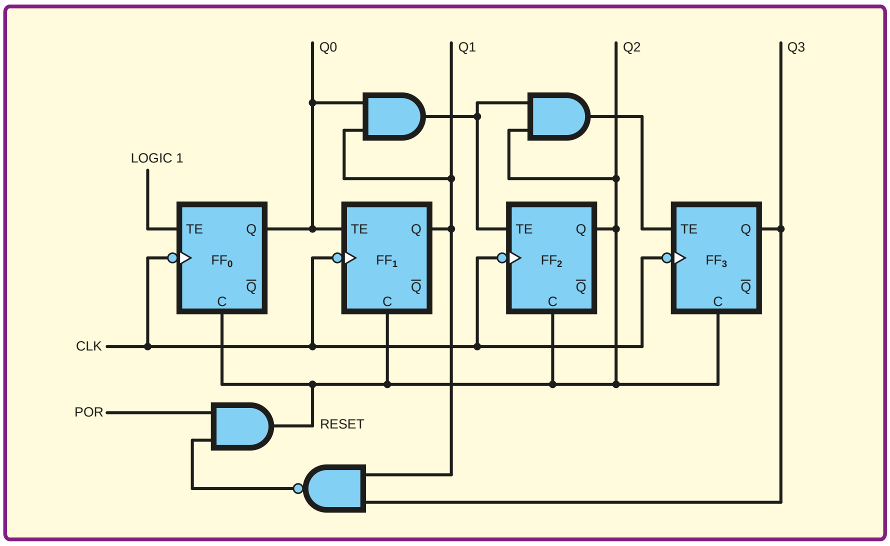


Fig.10. A 4 bit synchronous decade counter using toggle/hold flip-flops with asynchronous reset

The structure of the synchronous counter follows a pattern, so it easy to extend the circuit in Fig.6 to add more bits. Fig.7 shows a 4-bit version. Adding more bits is simply a matter of including more copies of the TE flip-flop and AND gate stage for all but the last and first bits.

Fig.8 shows a simulation of the circuit in Fig.7. Fig.9 shows a zoom-in to the 0111 to 1000 change, a similar time point to Fig.2. Unlike the circuit in Fig.1, no intermediate output values are produced. The circuit in Fig.7 does not suffer from the output ripple effect inherent in the circuit in Fig.1. A synchronous counter with many bits will update its outputs as quickly as one with just a few bits. However, extending the asynchronous structure in Fig.1 makes the counter take progressively longer to settle its outputs.

Limits

There are limits to the synchronous counter however. The outputs of all the AND gates must have settled to the correct value before the next active clock edge. This determines the maximum clock frequency, but as long as this requirement is met the update process is hidden from a user (or other circuitry) looking at the counter's outputs.

It is straightforward to make a synchronous decade counter using the same approach as Fig.1 – this circuit is shown in Fig.10. It uses toggle/hold flip-flops with asynchronous clear inputs. The use of asynchronous resets means that the circuit is not really fully synchronous – the reset is not under the control of the clock. Fig.11 and Fig.12 show a simulation of the circuit in Fig.10 zoomed in to the point where the reset occurs. The output does not change cleanly from nine (1001) to zero (0000) because the circuit must output ten (1010) for long enough for the reset to take effect.

It is possible to design flip-flops with synchronous reset (clear) inputs, rather than the asynchronous resets

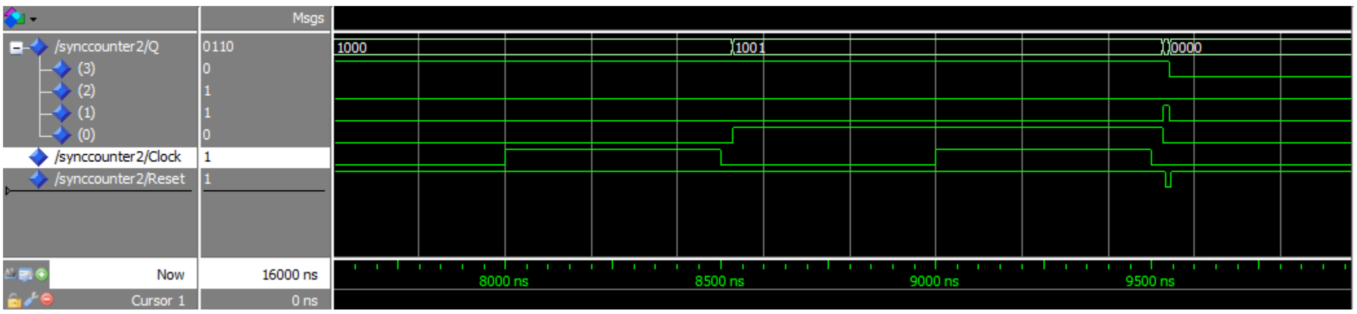


Fig.11. Simulation of the circuit in Fig.10 zoomed in on the 1001 to 0000 change due to the reset from the NAND gate

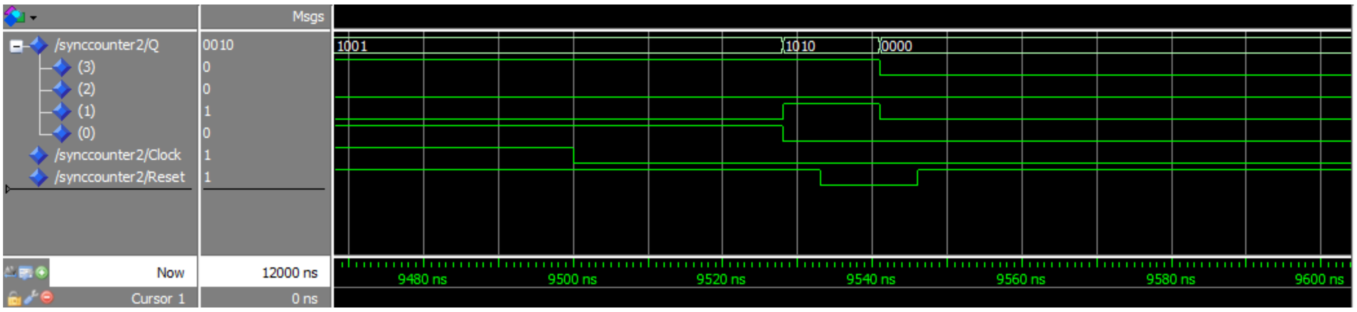


Fig.12. A closer zoom on the reset action from Fig.11

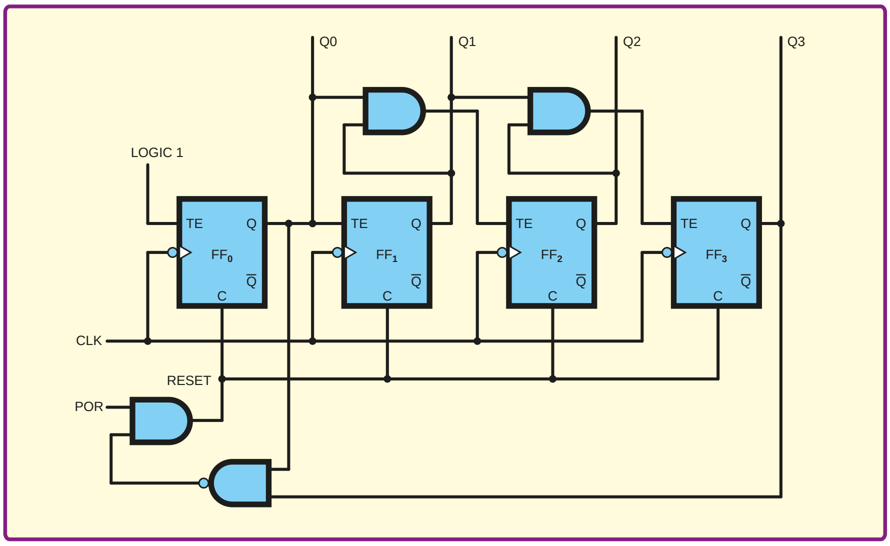


Fig.13. 4-bit synchronous decade counter using toggle/hold flip-flops with synchronous reset

most common in 4000 and 74 series devices. These flip-flops do not reset immediately when the reset is applied, but also require an active clock edge. To build a decade counter with synchronous reset flip-flops we need to detect nine (1001) to apply the reset rather than ten, this is shown in the circuit in Fig.13. Simulation waveforms of the rest as the count goes from nine to zero are shown in Fig.14.

When the fully synchronous counter reaches nine the NAND gate output goes high producing an active clear input into each of the flip-flops. However, unlike in the previous circuits, they do not reset almost immediately. The counter remains outputting value nine with the reset applied until the next active clock edge. At this point the flip-flops' internal logic overrides the toggle/hold action and sets their stored value to 0 – the counter resets to zero shortly after the clock edge and does not temporarily output the value ten. In this circuit the reset pulse lasts for a complete clock cycle and does not have the glitch-like characteristic of the circuits in Fig.1 and Fig.10. The counter changes from nine to zero cleanly, without ever outputting ten.

A fully synchronous design has the advantage of predictable and reliable circuit behaviour and is therefore widely used. The power consumption may be higher than an asynchronous version of the same circuit, because all of the flips-flops clock all the time, rather than just when they are need to change state. However, designers of large digital circuits have developed a variety of techniques for making synchronous circuits power efficient.

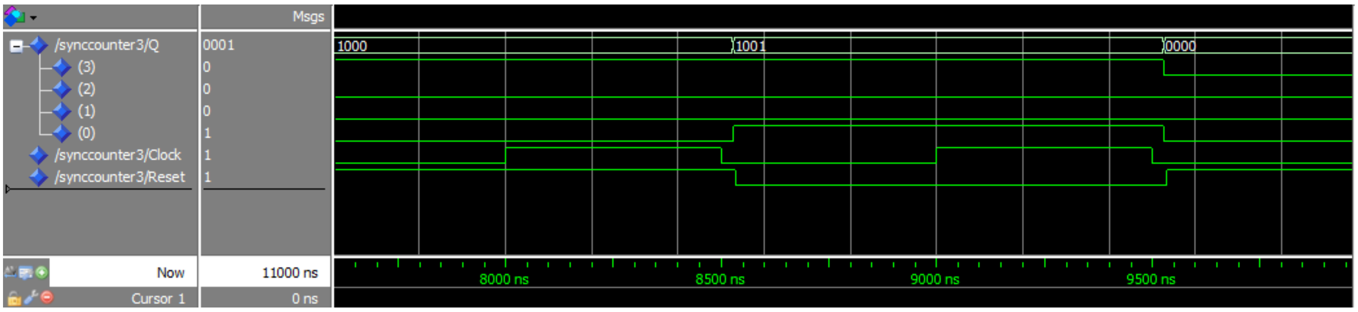


Fig.14. Simulation of the circuit in Fig.12 zoomed in on the 1001 to 0000 change due to the reset from the NAND gate. Compare with Fig.11 and Fig.12

EPE IS PLEASED TO BE ABLE TO OFFER YOU THESE

ELECTRONICS CD-ROMS

NEW OUT NOW

FLOWCODE 6

Now you can develop complex electronic and electromechanical systems with ease

Flowcode is one of the world's most advanced environments for electronic and electromechanical system development. Engineers use Flowcode to develop systems for control and measurement based on microcontrollers, on rugged industrial interfaces or on Windows compable personal computers

FREE VERSION
The free edition of Flowcode allows users to sample the fully featured product for 30days, after which it will revert to a more limited form and will lose its ability to compile to micro-controllers. It will continue to provide most of its functionality with Matrix hardware devices like the Matrix Forumula Flowcode robot, MIAC and ECIO.

CHIP PACK ONLY - £79.00
Purchasing a Chip Pack only will provide the ability to compile to hardware. Chip Pack versions are suitable for users who only require the basic Flowcode components. The Chip Pack version is equivalent to the Flow-code v5 Student/Home version. Multiple chip packs can be purchased to allow Flowcode to compile to more hardware targets.

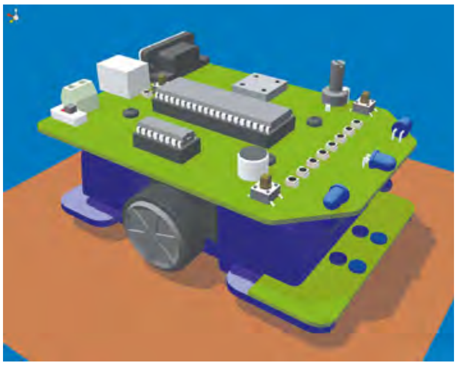
PROFESSIONAL UPGRADE PACK - £220.00
Adding the Professional bolt-on pack to the Flowcode licence will unlock all the functions and components available in Flowcode. It will also be licensed for commercial use.

ACADEMIC VERSIONS
Special arrangements can be made for Academic versions of Flowcode. Academic multi-user versions (10 user, 50 user) are also available. Only educational schools, colleges and universities are able to apply to purchase an Academic version of Flowcode. Please contact us for more information.

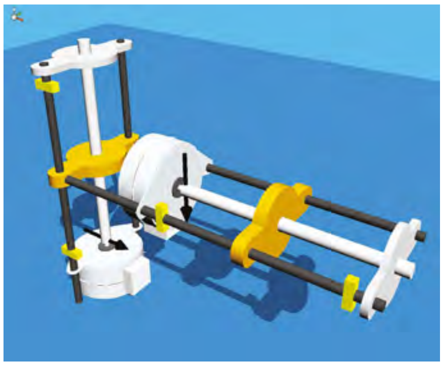
FLOWCODE 5 TO FLOWCODE 6 UPGRADE
Upgrades from Flowcode 5 to Flowcode 6 will be given a discount of 50% to the appropriate licence prices in Flowcode v6.

FLOWCODE 4 TO FLOWCODE 6 UPGRADE
Upgrades from Flowcode 4 to Flowcode 6 will given a discount of 30% to the appropriate licence prices in Flowcode v6 (pay 70% of the price).

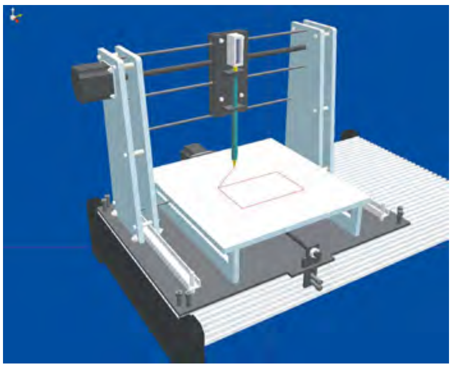
New simulation, 3D and test features allow you to design a greater range of systems including:



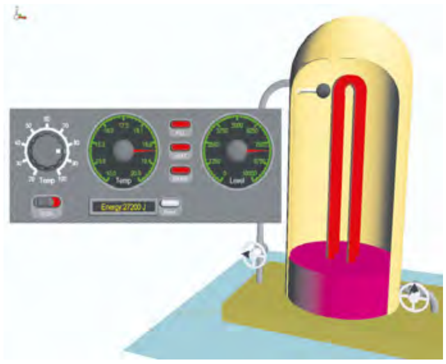
...robotic systems design...



...electromechanical system design...



...machine building...



...pumping and heating systems

PRICES Prices for each of the items above are:

(Order form on third page)

(UK and EU customers add VAT to 'plus VAT' prices)

Flowcode 6

PICMICRO/AVR/DSPIC/PIC24 & ARM Download only

Chip Pack (Hobbyist/Student)	**£94.80 inc. VAT**
Pro-pack (Schools/HE/FE/Industry)	**£299.00 plus VAT**
Flowkit	**£64.90 plus VAT**

Please note: **Due to popular demand, Flowcode PICmicro, AVR, DSPIC, PIC24 & ARM V6 are now available as a download. Please include your email address and a username (of your choice) on your order. A unique download code will then be emailed to you. If you require the CDROM as a back-up then please add an extra £14 to the above price.**

PICmicro TUTORIALS AND PROGRAMMING

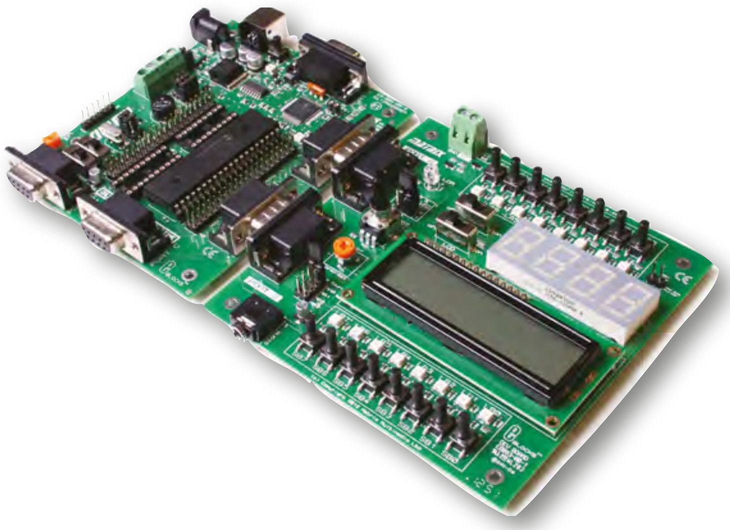
HARDWARE

PICmicro Multiprogrammer Board and Development Board

Suitable for use with the three software packages listed below

This flexible PICmicro microcontroller programmer board and combination board allows students and professional engineers to learn how to program PICmicro microcontrollers as well as program a range of 8, 18, 28 and 40 pin devices from the 12, 16 and 18 series PICmicro ranges. For experienced programmers all programming software is included in the PPP utility that comes with the multiprogrammer board. For those who want to learn, choose one or all of the packages below to use with the hardware.

- Makes it easier to develop PICmicro projects
- Supports low cost Flash-programmable PICmicro devices
- Fully featured integrated displays – 16 individual LEDs, quad 7-segment display and alphanumeric LCD display
- Supports PICmicro microcontrollers with A/D converters
- Fully protected expansion bus for project work
- USB programmable
- Compatible with the E-blocks range of accessories



£167 including VAT and postage, supplied with USB cable and programming software

SOFTWARE

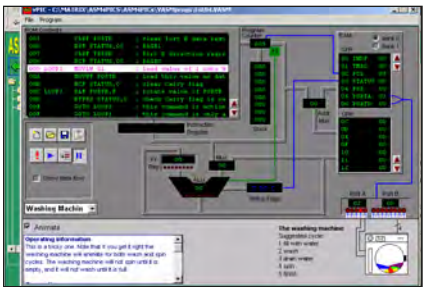
ASSEMBLY FOR PICmicro V4

(Formerly PICtutor)

Assembly for PICmicro microcontrollers V3.0 (previously known as PICtutor) by John Becker contains a complete course in programming the PIC16F84 PICmicro microcontroller from Arizona Microchip. It starts with fundamental concepts and extends up to complex programs including watchdog timers, interrupts and sleep modes.

The CD makes use of the latest simulation techniques which provide a superb tool for learning: the Virtual PICmicro microcontroller, this is a simulation tool that allows users to write and execute MPASM assembler code for the PIC16F84 microcontroller on-screen. Using this you can actually see what happens inside the PICmicro MCU as each instruction is executed, which enhances understanding.

● Comprehensive instruction through 45 tutorial sections ● Includes Vlab, a Virtual PICmicro microcontroller: a fully functioning simulator ● Tests, exercises and projects covering a wide range of PICmicro MCU applications ● Includes MPLAB assembler ● Visual representation of a PICmicro showing architecture and functions ● Expert system for code entry helps first time users ● Shows data flow and fetch execute cycle and has challenges (washing machine, lift, crossroads etc.) ● Imports MPASM files.

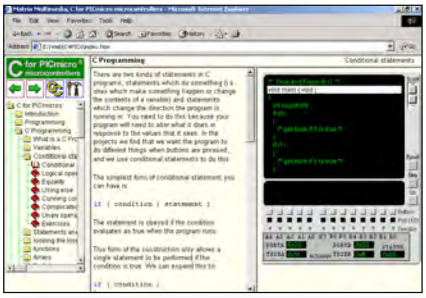


'C' FOR 16 Series PICmicro Version 4

The C for PICmicro microcontrollers CD-ROM is designed for students and professionals who need to learn how to program embedded microcontrollers in C. The CD-ROM contains a course as well as all the software tools needed to create Hex code for a wide range of PICmicro devices – including a full C compiler for a wide range of PICmicro devices.

Although the course focuses on the use of the PICmicro microcontrollers, this CD-ROM will provide a good grounding in C programming for any microcontroller.

● Complete course in C as well as C programming for PICmicro microcontrollers ● Highly interactive course ● Virtual C PICmicro improves understanding ● Includes a C compiler for a wide range of PICmicro devices ● Includes full Integrated Development Environment ● Includes MPLAB software ● Compatible with most PICmicro programmers ● Includes a compiler for all the PICmicro devices.



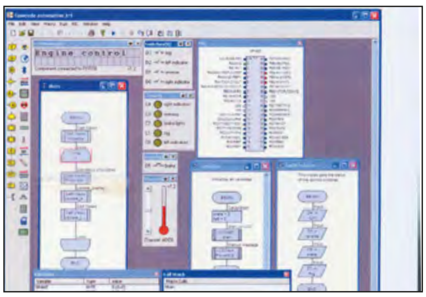
Minimum system requirements for these items: Pentium PC running, 2000, ME, XP; CD-ROM drive; 64MB RAM; 10MB hard disk space.
Flowcode will run on XP or later operating systems

FLOWCODE FOR PICmicro V6 (see opposite page)

Flowcode is a very high level language programming system based on flowcharts. Flowcode allows you to design and simulate complex systems in a matter of minutes. A powerful language that uses macros to facilitate the control of devices like 7-segment displays, motor controllers and LCDs. The use of macros allows you to control these devices without getting bogged down in understanding the programming. When used in conjunction with the Version 3 development board this provides a seamless solution that allows you to program chips in minutes.

- Requires no programming experience
- Allows complex PICmicro applications to be designed quickly
- Uses international standard flow chart symbols
- Full on-screen simulation allows debugging and speeds up the development process.
- Facilitates learning via a full suite of demonstration tutorials
- Produces ASM code for a range of 18, 28 and 40-pin devices
- 16-bit arithmetic strings and string manipulation
- Pulse width modulation
- I2C.

Features include panel creator, in circuit debug, virtual networks, C code customisation, floating point and new components. The Hobbyist/Student version is limited to 4K of code (8K on 18F devices)



PRICES

Prices for each of the CD-ROMs above are:

See previous page for Flowcode Hobbyist/Student prices

(Order form on next page)

(UK and EU customers add VAT to 'plus VAT' prices)

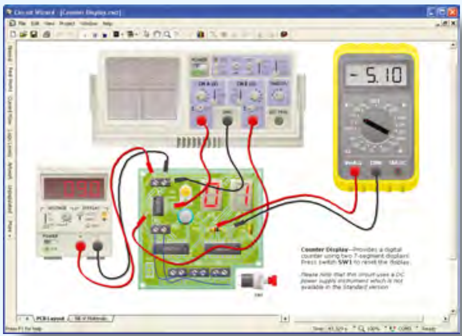
Hobbyist/Student	£58.80	inc VAT
Professional (Schools/HE/FE/Industry)	£150	plus VAT
Professional 10 user (Network Licence)	£499	plus VAT
Site Licence	£999	plus VAT
Flowcode 10 user (Network Licence)	POA	plus VAT
Flowcode Site Licence	POA	plus VAT

CIRCUIT WIZARD

Circuit Wizard is a revolutionary software system that combines circuit design, PCB design, simulation and CAD/CAM manufacture in one complete package.
Two versions are available, Standard or Professional.

By integrating the entire design process, Circuit Wizard provides you with all the tools necessary to produce an electronics project from start to finish – even including on-screen testing of the PCB prior to construction!

** Circuit diagram design with component library (500 components Standard,1500 components Professional) * Virtual instruments (4 Standard, 7 professional) * On-screen animation * Interactive circuit diagram simulation * True analogue/digital simulation * Simulation of component destruction * PCB Layout * Interactive PCB layout simulation * Automatic PCB routing * Gerber export * Multi-level zoom (25% to 1000%) * Multiple undo and redo * Copy and paste to other software * Multiple document support*



This software can be used with the *Jump Start* and *Teach-In 2011* series (and the *Teach-In 4* book).

Standard **£61.25** inc. VAT Professional **£91.90** inc. VAT

GCSE ELECTRONICS

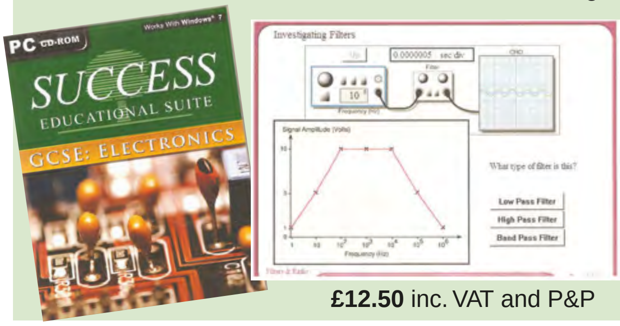
Suitable for any student who is serious about studying and who wants to achieve the best grade possible. Each program's clear, patient and structured delivery will aid understanding of electronics and assist in developing a confident approach to answering GCSE questions. The CD-ROM will be invaluable to anyone studying electronics, not just GCSE students.

**** Contains comprehensive teaching material to cover the National Curriculum syllabus * Regular exercises reinforce the teaching points * Retains student interest with high quality animation and graphics * Stimulates learning through interactive exercises * Provides sample examination ques-tions with model solutions * Authored by practising teachers * Covers all UK examination board syllabuses * Caters for all levels of ability * Useful for self-tuition and revision***

SUBJECTS COVERED

Electric Circuits – Logic Gates – Capacitors & Inductors – Relays – Transistors – Electric Transducers – Operational Amplifiers – Radio Circuits – Test Instruments

Over 100 different sections under the above headings



£12.50 inc. VAT and P&P

Minimum system requirements for these CD-ROMs: Pentium PC, CD-ROM drive, 32MB RAM, 10MB hard disk space. Windows 2000/ME/XP, mouse, sound card, web browser.

Please send me: CD-ROM ORDER FORM

☐ Assembly for PICmicro V4
☐ 'C' for 16 Series PICmicro V4

Version required:
☐ Hobbyist/Student
☐ Professional
☐ Professional 10 user
☐ Site licence

☐ Flowcode for PICmicro V6 (DOWNLOAD ONLY)
☐ Flowcode for AVR V6 (DOWNLOAD ONLY)
☐ Flowcode for ARM V6 (DOWNLOAD ONLY)
☐ Flowcode for dsPIC V6 (DOWNLOAD ONLY)
☐ Flowcode for PIC24 V6 (DOWNLOAD ONLY)
☐ Flowkit

Version required:
☐ Chip pack (Hobbyist/Student)
☐ Pro pack (Schools/HE/FE/Ind.)

Email: . . . .

Username: . . . .

Note: The software on each version is the same, only the licence for use varies.

☐ PICmicro Multiprogrammer Board and Development Board V4 (hardware)

☐ Circuit Wizard – Standard
☐ Circuit Wizard – Professional
☐ GCSE Electronics

Full name: . . . .

Address: . . . .

. . . . Post code: . . . . Tel. No: . . . .

Signature: . . . .

☐ I enclose cheque/PO in £ sterling payable to WIMBORNE PUBLISHING LTD for £ . . . .
☐ Please charge my Visa/Mastercard/Maestro: £ . . . .
Valid From: . . . . Card expiry date: . . . .
Card No: . . . . Maestro Issue No. . . . .
Card Security Code . . . . (The last 3 digits on or just under the signature strip)

ORDERING
ALL PRICES INCLUDE UK POSTAGE

Student/Single User/Standard/Hobbyist Version price includes postage to most countries in the world
EU residents outside the UK add £5 for airmail postage per order

Professional, Multiple User and Site License Versions – overseas readers add £5 to the basic price of each order for airmail postage (**do not add VAT** unless you live in an EU (European Union) country, then add VAT at 20% or provide your official VAT registration number).

Send your order to:
Direct Book Service
Wimborne Publishing Ltd
113 Lynwood Drive, Merley,
Wimborne, Dorset BH21 1UU

To order by phone ring

01202 880299. Fax: 01202 843233

Goods are normally sent within seven days

E-mail: orders@wimborne.co.uk

Online shop:

www.epemag.com

NET WORK

by Alan Winstanley

A Near-Field experience

I HAD an amusing experience in a lighting shop recently, when the store clerk tried to snatch my debit card from my hand and poke it into the swipe terminal, but I gripped my card firmly and we arm-wrestled over the chip-and-pin machine. Then I waved my card over the machine's LCD display and a 'beep' was emitted. To the clerk's amazement, the transaction had been completed without needing a PIN number. Then I talked the clerk through the procedure to print a duplicate receipt. I had to do the same at my local Post Office.

Near-field communication

Contactless transactions like these are becoming a convenient way to pay for small purchases (typically £20 or $30 maximum) without needing to punch in a PIN number. Payment cards use an RF system to transmit encrypted data to the terminal wirelessly. Another contactless technology is near field communications (NFC), an ultra short-range RF protocol which is finding its way into more mobile devices, including smartphones. NFC is not new and Nokia, part of the NFC founding consortium, introduced the first NFC-based mobile phone eight years ago. The owner's payment card details were configured on the phone's SIM card using a 'ticketing application' to pay for, say, London Underground rail tickets. TV programmes at the time marvelled at how Finns could use a cola vending machine or a car park, paying 'contactlessly' with their Nokia mobile phone.

Many current smartphones incorporate NFC, but the bi-directional wireless technology is a glaring omission from Apple's current iPhone line-up. Thanks to contactless debit and credit cards that are plain and simple to use, there's a slight feeling that NFC mobile phones might already have missed the contactless payment boat, but Barclaycard (Quick Tap) and Mastercard (Cash on Tap) are rolling out NFC SIMcard-based payments on some mobile networks.

Following an OTA software update from HTC, I was intrigued to find a new 'N' icon shining in the notification area of my HTC One phone. The mystery icon indicated that the NFC sensor was running and so, with mobile phone contactless payments ruled out for now, I decided to see what else NFC could offer. A quick trip to Maplin and the Google Play Store revealed some interesting applications.

RapidNFC offers a wide range of NFC consumables

Putting NFC to use

If your smartphone (eg, Samsung, HTC, Sony Experia and others) is equipped with an NFC sensor (eg, the HTC One sensor is situated around the camera lens, and it works through the aluminium body) then it can also be used to control mobile phone apps, change your phone's settings or open websites on the phone's web browser. Readers may be interested in the use of NFC tags and labels which are readily available online from **RapidNFC.com** who claim to be Europe's largest supplier of NFC consumables.

A typical NFC label has a spiral antenna and a 137-byte memory chip that can be scanned wirelessly. These can be bought in small packs from eBay, Maplin (A65LR) or RapidNFC and there's the prospect of recycling NFC badges from exhibitions or hotels as well. NFC labels are laminated and are intended to stick onto non-metallic surfaces, but shielded 'on-metal' labels are also available. By scanning the label chip with an NFC reader, the NFC tag can open another app, or configure your phone to suit the circumstances. Typical ranges are a few centimeters at most and your phone must be 'awake' to scan NFC tags.

I tested some ideas, including turning on the flashlight app of my phone – the label could be stuck by the bedside for instance, and tapped to turn on the phone's light automatically; another label would turn it off again. Or tap a tag to set your phone to night-time mode. An NFC label could be stuck to a car dashboard to configure hands-free in-car usage, start the sat nav and turn on Bluetooth; another label or tag could be used on a bicycle or hiker's backpack to set up the speakerphone and turn up the volume. Tap a red NFC label (for example) and you could dial an emergency number. You can also interact with 'smart posters' in the same way you scan a QR code.

Other uses include embedding them in rigid plastic discs, wristbands, tickets or business cards –

Small starter packs of NFC plastic adhesive labels are available allow your guest to tap them to open a business website URL or send a text string or portfolio web address, or fix them onto equipment to act as a contactless ID tag. NFC labels offer all the benefits of QR codes with none of the problems caused by poor print or damaged graphics, but QR codes and modern smartphones work together well over greater distances and still have their uses. There are plenty of NFC apps around, and some Android apps to check out include NFC ReTag and the free NFC Tools app, which can identify, write, read, copy or lock (read-only) tags which may be worth a try. Windows phone users can download a choice of free NFC apps from **windowsphone.com**. Even though NFC payments have some way to go, you can still harness NFC technology today.

The NFC ReTag app for Android allows tags to be configured and scanned. Others are available from Google Play Store

You've been farmed!

In earlier years, I used to work in the retail sector and one day I found myself at the head office of a national car accessories chain, watching sales of our products clocking up on their buyer's screen in real time. Data was fed from the cash registers and we got a feel for what sold and what risked becoming a duff product.

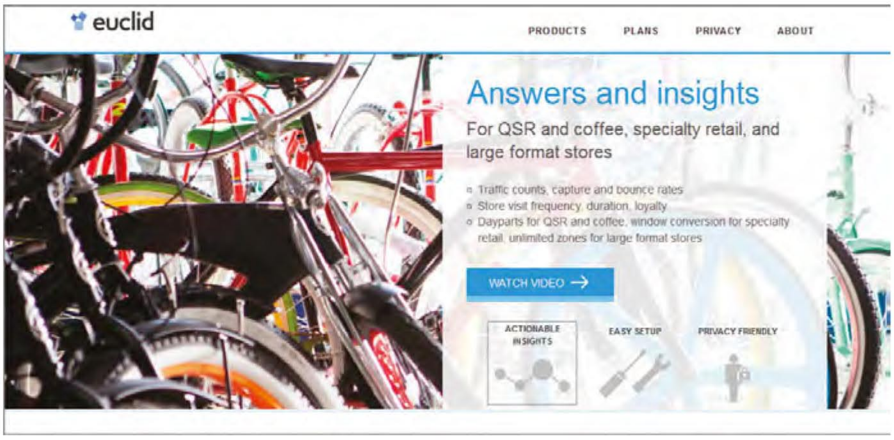
One advantage website owners have over traditional bricks and mortar stores is the wealth and diversity of visitor statistics available to them. Software makes it easy to analyse web server logs, click-through ratios, page hits, search engine referrer terms, exit and entry pages, abandoned shopping carts (yes, those are monitored too), conversion rates and much more, and this allows website marketers to record, refine and control the flow of visitors to their online presence down to a granular level. Google's analytical tools offer enough raw data to satisfy an army of number-crunching statisticians. Website designers can also determine the 'hot spots' on a webpage as prime areas for placing adverts.

Furthermore, websites can change their shop window in minutes, with new campaigns, products and offers being launched electronically. Online advertisers can experiment quickly and monitor results in real time, comparing the effects of various changes until they hit on a magic formula. This ability to monitor and respond quickly has made traditional stores look sclerotic in comparison.

It's all about Wi-Fi

Help could now be at hand for store owners who want to track the footfall of customers in their bricks and mortar stores in the same way that website owners track their visitors. Thanks to the popularity of smartphones and a thirst for Wi-Fi, shop retailers can now obtain and interpret all sorts of anonymous customer data without their visitors' knowledge. Euclid Analytics (**http://euclidanalytics.com/**) specialises in tracking real-time shopper behaviour in retail stores and the QSR (quick service restaurant) sectors, and the amount of physical visitor data that can be produced seems impressive.

At the simplest level, Euclid says they collect visitor data via a store Wi-Fi's wireless access points. Larger-scale stores would deploy an array of discrete wireless sensors. How does it work? The customer's smartphone is the key to everything. Euclid works like an in-house radar screen, detecting the Wi-Fi 'pings' that smartphones emit as they search for an available wireless network. Of course, having free Wi-Fi in store will encourage smartphone Wi-Fi to be left on. The phone's unique MAC address is then captured, encrypted and sent securely to the cloud and visitor data can be computed in real time.



Euclid Analytics uses a smartphone's MAC code to track physical shoppers' movements

By checking the customer's MAC address as it moves around the store, herds of shoppers can be tracked in the same way that a website visitor's IP address is logged on a web server. A shopper's physical tracks can therefore be monitored to determine his or her shopping patterns as they move through the 'shopper funnel', and then aggregated data can reveal customer trends by location, time, day and month.

The store can now get the raw visitor data that it craves.

Metrics such as conversion rates can be measured – eg, how successfully a window display entices a customer into entering a particular area and whether they approach the cash register. Bounce rates, equivalent to website 'stickiness', measure how soon they leave. Long-term loyalty can be quantified by checking how frequently an individual visits the store or a department within it, and data can be sorted by days of the week. Data for 'cross shopping' – what other stores did you visit? – and shopper engagement can also be derived. Figures can at last be put on 'shopper retention' and using this raw data store owners can work out seasonal offers, more effective marketing campaigns and even how to manage staff rotas. Euclid's Youtube video **http://www.youtube.com/watch?v=LNckTJZaY_g** briefly shows the scale of data that can be derived. Euclid is keen to emphasise the privacy and data encryption aspects of its software, stating that no personal data is collected, and consumers can opt in or out at the Euclid website or instructions are given in-store to find their MAC addresses. But will today's consumers really care as long as free Wi-Fi is available?

In due course, Apple iPhone users may have fun of their own with the implementation of iBeacons, an emerging system of localised Bluetooth-based transmitters that detects your proximity wirelessly and responds accordingly. In a retail store, for example, a customer's presence could trigger a range of offers, discounts, product suggestions and promotions that are sent directly to a customer's iPhone. It is already under trial in Macy's in the US, so if you already recoil at the sight of so many people wandering aimlessly around while glued to their phones, then things may get worse: see Shopkick's iBeacon video at: **www.youtube.com/watch?v=c3h0eKGfUfI**. A UK writer has also configured a Raspberry Pi to act as an iBeacon, see: **www.theregister.co.uk/2013/11/29/feature_diy_apple_ibeacons/**

That's all for this month's *Net Work*. You can email the author at **alan@epemag.demon.co.uk**.

DIRECT BOOK SERVICE

ELECTRONICS TEACH-IN BUNDLE – SPECIAL BUNDLE PRICE £14 FOR PARTS 1, 2 & 3

ELECTRONICS TEACH-IN 2 CD-ROM
USING PIC MICROCONTROLLERS A PRACTICAL INTRODUCTION

This *Teach-In* series of articles was originally published in *EPE* in 2008 and, following demand from readers, has now been collected together in the *Electronics Teach-In 2* CD-ROM.

The series is aimed at those using PIC microcontrollers for the first time. Each part of the series includes breadboard layouts to aid understanding and a simple programmer project is provided.

Also included are 29 *PIC N' Mix* articles, also republished from *EPE*. These provide a host of practical programming and interfacing information, mainly for those that have already got to grips with using PIC microcontrollers. An extra four part beginners guide to using the C programing language for PIC microcontrollers is also included.

The CD-ROM also contains all of the software for the *Teach-In 2* series and *PIC N' Mix* articles, plus a range of items from Microchip – the manufacturers of the PIC microcontrollers. The material has been compiled by Wimborne Publishing Ltd. with the assistance of Microchip Technology Inc.

CD-ROM **Order code ETI2 CD-ROM** **£9.50**

ELECTRONICS TEACH-IN 3 CD-ROM

The three sections of this CD-ROM cover a very wide range of subjects that will interest everyone involved in electronics, from hobbyists and students to professionals. The first 80-odd pages of Teach-In 3 are dedicated to *Circuit Surgery*, the regular *EPE* clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers the practical aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and indentifying components, are covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by the readers of *EPE*.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge, TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The Teach-In 1 series covers everything from Electric Current through to Microprocessors and Microcontrollers and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

CD-ROM **Order code ETI3 CD-ROM** **£8.50**

CD-ROMs **Order code ETIBUNDLE** Bundle Price **£14.00**

ELECTRONICS TEACH-IN 4

ELECTRONICS TEACH-IN 4 CD-ROM

A Broad-Based Introduction to Electronics.

The Teach-In 4 CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the new Diploma in Engineering, Level 2.

The CD-ROM also contains the full Modern Electronics Manual, worth £29.95. The Manual contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to all those interested in learning about electronics or brushing up on their theory, be they hobbyists, students or professionals.

CD-ROM **Order code ETI4 CD-ROM** **£8.99**

RASPBERRY PI

RASPBERRY Pi MANUAL: A practical guide to the revolutionary small computer
176 pages **Order code H001** **£17.99**

RASPBERRY Pi USER-GUIDE
262 pages **Order code JW001** **£12.99**

PROGRAMMING THE RASPBERRY Pi
192 pages **Order code MGH4** **£10.99**

GETTING STARTED WITH RASPBERRY Pi
164 pages **Order code OR01** **£11.50**

ELECTRONICS TEACH-IN 5

FREE CD-ROM

ELECTRONICS TEACH-IN 5

Jump Start – 15 design and build circuit projects dedicated to newcomers or those following courses in school and colleges. The projects are: Moisture Detector, Quiz Machine, Battery Voltage Checker, Solar-Powered Charger, Versatile Theft Alarm, Spooky Circuits, Frost Alarm, Mini Christmas Lights, iPod Speaker, Logic Probe, DC Motor Controller, Egg Timer, Signal Injector Probe, Simple Radio Receiver, Temperature Alarm.

PLUS:

PIC' N MIX – starting out with PIC Microcontrollers and PRACTICALLY SPEAKING – the techniques of project construction.

FREE CD-ROM – The free CD-ROM is the complete Teach-In 2 book providing a practical introduction to PIC Microprocessors plus MikroElektronika, Microchip and L-Tek PoScope software.

160 Pages **Order code ETI5** **£8.99**

MICROPROCESSORS

INTERFACING PIC MICROCONTROLLERS
Martin Bates
298 pages **Order code NE48** **£30.99**

PROGRAMMING 16-BIT PIC MICROCONTROLLERS IN C – LEARNING TO FLY THE PIC24 Lucio Di Jasio (Application Segments Manager, Microchip, USA)
496 pages +CD-ROM **Order code NE45** **£38.00**

INTRODUCTION TO MICROPROCESSORS AND MICROCONTROLLERS – SECOND EDITION
John Crisp
222 pages **Order code NE31** **£29.99**

THE PIC MICROCONTROLLER YOUR PERSONAL INTRODUCTORY COURSE – THIRD EDITION. John Morton
270 pages **Order code NE36** **£25.00**

PIC IN PRACTICE (2nd Edition)
David W. Smith
308 pages **Order code NE39** **£24.99**

MICROCONTROLLER COOKBOOK
Mike James
240 pages **Order code NE26** **£36.99**

The books listed have been selected by *Everyday Practical Electronics* editorial staff as being of special interest to everyone involved in electronics and computing. They are supplied by mail order direct to your door. Full ordering details are given on the next page.

FOR A FULL DESCRIPTION OF THESE BOOKS AND CD-ROMS SEE THE SHOP ON OUR WEBSITE www.epemag.com

All prices include UK postage

1

PROJECT CONSTRUCTION

IC 555 PROJECTS
E. A. Parr
167 pages **Order code BP44** **£5.49**

ELECTRONIC PROJECT BUILDING FOR BEGINNERS R. A. Penfold
135 pages **Order code BP392** **£5.99**

PRACTICAL FIBRE-OPTIC PROJECTS
R. A. Penfold
132 pages **Order code BP374** **£5.45**

THEORY AND REFERENCE

PRACTICAL ELECTRONICS HANDBOOK – Fifth Edition. Ian Sinclair
440 pages **Order code NE21** **£32.50**

STARTING ELECTRONICS – 4th Edition
Keith Brindley
296 pages **Order code NE100** **£18.99**

ELECTRONIC CIRCUITS – FUNDAMENTALS & APPLICATIONS – Third Edition Mike Tooley
400 pages **Order code TF43** **£25.99**

FUNDAMENTAL ELECTRICAL AND ELECTRONIC PRINCIPLES – Third Edition C R. Robertson
368 pages **Order code TF47** **£21.99**

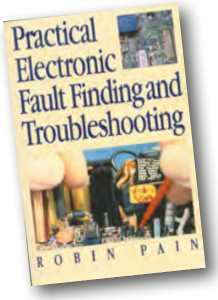
A BEGINNER'S GUIDE TO TTL DIGITAL ICs
R.A. Penfold
142 pages **Order code BP332** **£5.45**

UNDERSTANDING ELECTRONIC CONTROL SYSTEMS
Owen Bishop
228 pages **Order code NE35** **£36.99**

FAULT FINDING AND TEST EQUIPMENT

GETTING THE MOST FROM YOUR MULTIMETER
R. A. Penfold
96 pages | Order code BP239 | £5.49

PRACTICAL ELECTRONIC FAULT FINDING AND TROUBLESHOOTING
Robin Pain
274 pages | Order code NE22 | £41.99



HOW ELECTRONIC THINGS WORK – AND WHAT TO DO WHEN THEY DON'T
Robert Goodman
394 pages | Order code MGH3 | £21.99

OSCILLOSCOPES – FIFTH EDITION
Ian Hickman
288 pages | Order code NE37 | £36.99

HOW TO USE OSCILLOSCOPES AND OTHER TEST EQUIPMENT
R.A. Penfold
104 pages | Order code BP267 | £5.49

AUDIO & VIDEO

VALVE AMPLIFIERS – Second Edition
Morgan Jones
288 pages | Order code NE33 | £40.99

BUILDING VALVE AMPLIFIERS
Morgan Jones
368 pages | Order code NE40 | £29.00

DIGITAL AUDIO RECORDING
Ian Waugh
60 pages | Order code PC121 | £7.95

QUICK GUIDE TO MP3 AND DIGITAL MUSIC
Ian Waugh
60 pages | Order code PC119 | £7.45

MAKING MUSIC WITH YOUR COMPUTER
Stephen Bennett
92 pages | Order code PC120 | £10.95

QUICK GUIDE TO DIGITAL AUDIO RECORDING
Ian Waugh
208 pages | Order code PC121 | £7.95

ELECTRONIC PROJECTS FOR VIDEO ENTHUSIASTS
R.A. Penfold
109 pages | Order code BP356 | £5.45

COMPUTING AND ROBOTICS

NEWNES INTERFACING COMPANION
Tony Fischer-Cripps
295 pages | Order code NE38 | £41.00

HOW TO BUILD A COMPUTER MADE EASY
R.A. Penfold
120 pages | Order code BP707 | £8.49

EASY PC CASE MODDING
R.A. Penfold
192 pages + CDROM | Order code BP542 | £8.99

FREE DOWNLOADS TO PEP-UP AND PROTECT YOUR PC
R.A. Penfold
128 pages | Order code BP722 | £7.99

WINDOWS XP EXPLAINED
N. Kantaris and P.R.M. Oliver
264 pages | Order code BP514 | £7.99

THE INTERNET – TWEAKS, TIPS AND TRICKS
R. A. Penfold
128 pages | Order code BP721 | £7.99

eBAY – TWEAKS, TIPS AND TRICKS
R. A. Penfold
128 pages | Order code BP716 | £7.50

AN INTRODUCTION TO eBAY FOR THE OLDER GENERATION
Cherry Nixon
120 pages | Order code BP709 | £8.49

COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
308 pages | Order code BP601 | £8.99

ANDROIDS, ROBOTS AND ANIMATRONS
Second Edition – John Iovine
224 pages | Order code MGH1 | £16.99

ROBOT BUILDERS COOKBOOK
Owen Bishop
366 pages | Order code NE46 | £26.00

INTRODUCING ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
288 pages + | Order code BP901 | £14.99

MORE ADVANCED ROBOTICS WITH LEGO MINDSTORMS – Robert Penfold
298 pages | Order code BP902 | £14.99

HOW TO FIX YOUR PC PROBLEMS
R. A. Penfold
128 pages | Order code BP705 | £8.49

WINDOWS 7 – TWEAKS, TIPS AND TRICKS
Andrew Edney
120 pages | Order code BP708 | £8.49

GETTING STARTED IN COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
120 pages | Order code BP704 | £8.49

BOOK ORDERING DETAILS

All prices include UK postage. For postage to Europe (air) and the rest of the world (surface) please add £3 per book. Surface mail can take up to 10 weeks to some countries. For the rest of the world airmail add £4 per book. CD-ROM prices include VAT and/or postage to anywhere in the world. Send a PO, cheque, international money order (£ sterling only) made payable to **Direct Book Service** or card details, Visa, Mastercard or Maestro to: **DIRECT BOOK SERVICE, WIMBORNE PUBLISHING LIMITED, 113 LYNWOOD DRIVE, MERLEY, WIMBORNE, DORSET BH21 1UU**.

Books are normally sent within seven days of receipt of order, but please allow 28 days for delivery – more for overseas orders. *Please check price and availability (see latest issue of* Everyday Practical Electronics*) before ordering from old lists.*

For a full description of these books please see the shop on our website.
Tel 01202 880299 Fax 01202 843233. E-mail: dbs@wimborne.co.uk

Order from our online shop at: www.epemag.com

BOOK ORDER FORM

Full name: ....

Address: ....

....

....

.... Post code: .... Telephone No: ....

Signature: ....

☐ I enclose cheque/PO payable to DIRECT BOOK SERVICE for £ ....

☐ Please charge my card £ .... Card expiry date..

Card Number .... Maestro Issue No.............................

Valid From Date .... Card Security Code .... (The last three digits on or just below the signature strip)

Please send book order codes: ....

....

Please continue on separate sheet of paper if necessary

PCB SERVICE

CHECK US OUT ON THE WEB

Basic printed circuit boards for most recent *EPE* constructional projects are available from the *PCB Service*, see list. These are fabricated in glass fibre, and are drilled and roller tinned, but all holes are a standard size. They are not silk-screened, nor do they have solder resist. Double-sided boards are **NOT plated through hole** and will require 'vias' and some components soldering to both sides. * **NOTE: PCBs from the July 2013 issue with eight digit codes** have silk screen overlays and, where applicable, are double-sided, plated through-hole, with solder masks, they are similar to the photos in the relevent project articles.

All prices include VAT and postage and packing. Add £2 per board for airmail outside of Europe. Remittances should be sent to **The PCB Service, Everyday Practical Electronics, Wimborne Publishing Ltd, 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU. Tel: 01202 880299; Fax 01202 843233; Email: orders@epemag.wimborne.co.uk. On-line Shop: www.epemag.com**. Cheques should be crossed and made payable to *Everyday Practical Electronics* **(Payment in £ sterling only)**.

NOTE: While 95% of our boards are held in stock and are dispatched within seven days of receipt of order, please allow a maximum of 28 days for delivery – overseas readers allow extra if ordered by surface mail.

Back numbers or photocopies of articles are available if required – see the Back Issues page for details. WE DO NOT SUPPLY KITS OR COMPONENTS FOR OUR PROJECTS.

PROJECT TITLE	ORDER CODE	COST
JANUARY '13		
Low-Capacitance Adaptor for DMMs	880	£8.55
★ 3-Input Stereo Audio Switcher		
– Main Board	881 } pair	£20.00
– Switch Board	882	
Stereo Compressor – Main Board	883	£12.63
Jump Start – iPod Speaker	884	£8.16
FEBRUARY '13		
10W LED Floodlight	885	£6.75
★ Crystal DAC (double-sided)	886	£18.46
Jump Start – Logic Probe	887	£6.42
MARCH '13		
Lightning Detector	888	£8.75
★ Digital Spirit Level	889	£8.75
★ SemTest – Part 2		
– Main/Lower Board	890	£16.52
– Display/Upper Board	891	£15.55
Interplanetary Voice	892	£8.75
Jump Start – DC Motor Controller	893	£8.55
APRIL '13		
Six-Decade Resistance Substitution Box	894	£10.10
SoftStarter (single-sided)	895	£8.36
Jump Start – Egg Timer	896	£8.36
SemTest HV DC Crowbar	897	£13.61
MAY '13		
Electronic Stethoscope	898	£9.72
PIC/AVR Programming Adaptor Board (d'ble-sided)	899	£23.33
Jump Start – Signal Injector Probe	900	£8.16
JUNE '13		
USB Breakout Box	901	£7.97
Jump Start – Simple Radio Receiver	902	£8.94
Mix-It	903	£11.28
JULY '13		
6-Decade Capacitance Substitution Box	04106121	£18.00
6-Decade Capacitance Substitution Box Panel/Lid	04106122	£18.00
SoftStarter For Power Tools	10107121	£9.00
High-Current Adaptor For Scopes And DMMs	04108121	£18.00
Jump Start – Temperature Alarm	904	£7.97
AUGUST '13		
Driveway Sentry	03107121	£18.00
Build A Vox	01207111	£22.00
Milliohm Meter Adaptor For DMMs	04102101	£18.00
SEPTEMBER '13		
MiniReg Adjustable Regulator	18112111	£6.00
USB Stereo Recording Interface	07106111	£18.00
Digital Sound Effects Module	09109121	£9.00

PROJECT TITLE	ORDER CODE	COST
OCTOBER '13		
High-Temperature Thermometer/Thermostat	21105121	£22.00
LED Musicolour	16110121	£22.00
NOVEMBER '13		
Mains Timer For Fans Or Lights	10108121	£9.10
CLASSiC-D Amplifier (inc. 3 Rockby Capacitors)	01108121	£26.50
– Speaker Protector	01108122	£9.10
DECEMBER '13		
CLASSiC-D Amplifier – Power Supply	01109111	£16.66
USB Instrument Interface	24109121	£26.38
– Front Panel	24109122	£28.54
JANUARY '14		
Champion	1109121/2	£8.88
Simple 1.5A Switching Regulator	18102121	£5.53
FEBRUARY '14		
2.5GHz 12-Digit Frequency Counter – Display PCB	04111122	£12.88
– Main PCB	04111121	£27.13
– Front Panel	04111123	£22.38
High-Energy Electronic Ignition System	05110121	£9.10
Mobile Phone LOUD Ringer	12110121	£9.10

*** See NOTE above left regarding PCBs with eight digit codes ***

Please check price and availability in the latest issue.
A large number of older boards are listed on, and can be ordered from, our website.
Boards can only be supplied on a payment with order basis.

EPE SOFTWARE

★ All software programs for *EPE* Projects marked with a star, and others previously published can be downloaded free from the Library on our website, accessible via our home page at: **www.epemag.com**

PCB MASTERS

PCB masters for boards published from the March '06 issue onwards can also be downloaded from our website (**www.epemag.com**); **go to the 'Library' section.**

EPE PRINTED CIRCUIT BOARD SERVICE

Order Code **Project** **Quantity** **Price**

. . . .

Name . . . .

Address . . . .

. . . .

Tel. No. . . . .

I enclose payment of £ . . . . (cheque/PO in **£ sterling only**) to:

Everyday Practical Electronics

MasterCard VISA Maestro

Card No. . . . .

Valid From . . . . Expiry Date . . . .

Card Security No. . . . . Maestro Issue No. . . . . . .

Signature . . . .

Note: You can also order PCBs by phone, Fax or Email or via the Shop on our website on a secure server:

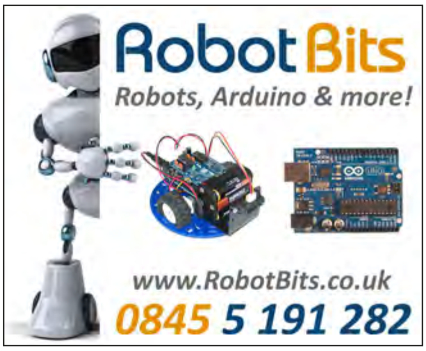
http://www.epemag.com

CLASSIFIED
A D V E R T I S E M E N T S

Everyday Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine, our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-six years.

If you want your advertisements to be seen by the largest readership at the most economical price our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2·5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).
All cheques, postal orders, etc., to be made payable to Everyday Practical Electronics. **VAT must be added**. Advertisements, together with remittance, should be sent to Everyday Practical Electronics Advertisements, 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU. Phone: 01202 880299. Fax: 01202 843233. Email: stewart.kearn@wimborne.co.uk. For rates and information on display and classified advertising please contact our Advertisement Manager, Stewart Kearn as above.



If you would like to advertise on the Classified page then please call Stewart Kearn on:

01202 880299

or email

stewart.kearn@wimborne.co.uk

BTEC ELECTRONICS TECHNICIAN TRAINING

NATIONAL ELECTRONICS
VCE ADVANCED ICT
HNC AND HND ELECTRONICS
FOUNDATION DEGREES
NVQ ENGINEERING AND IT
DESIGN AND TECHNOLOGY

LONDON ELECTRONICS COLLEGE
20 PENYWERN ROAD
EARLS COURT, LONDON SW5 9SU
TEL: (020) 7373 8721
www.lec.org.uk

CANTERBURY WINDINGS
UK manufacturer of toroidal transformers (10VA to 3kVA)
All transformers made to order. No design fees. No minimum order.
www.canterburywindings.co.uk
01227 450810

BOWOOD ELECTRONICS LTD
Suppliers of Electronic Components
Place a secure order on our website or call our sales line
All major credit cards accepted
Web: www bowood-electronics co uk
Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield, Derbyshire S40 2QR Sales: 01246 200222
Send 60p stamp for catalogue

LOOKING FOR
AUDIO & MULTICORE CABLES
Visit Section 1
www.partridgeelectronics.co.uk

MISCELLANEOUS

KITS, TOOLS, COMPONENTS. S.A.E. Catalogue. SIR-KIT ELECTRONICS, 52 Severn Road, Clacton, CO15 3RB, **http://sir-kit.webs.com**

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits. Visit **www.cstech.co.uk**

WANTED – 2061D IC by Nikko 20-Pin and data information. Good reward – contact Mr Ward 01425 475445

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

ADVERTISERS INDEX

ADVERTISEMENT OFFICES:
113 LYNWOOD DRIVE, MERLEY, WIMBORNE,
DORSET BH21 1UU
PHONE: 01202 880299
FAX: 01202 843233
EMAIL: stewart.kearn@wimborne.co.uk
WEB: www.epemag.com
For editorial address and phone numbers see page 7

Next Month

Content may be subject to change

Infrasound Detector

Are wind turbines making you sick? Is building vibration making you nauseous? Or do you just want to measure infrasound in your environment? You don't need to spend thousands to do it properly; just build our low-cost but accurate *Infrasound Detector.*

Model Railway Automatic Points Controller

This fun project uses two IR sensors to detect an approaching train and then automatically switches a set of points to suit the track on which the train is travelling. This avoids the possibility of inadvertent derailments!

High-Energy Electronic Ignition System – Part 2

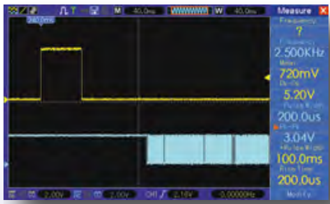
In this issue, we introduced our new *High-Energy Ignition System* and described its operation. Next month, we'll give you the assembly details for six different versions, to suit your car's trigger input.

Extremely accurate GPS 1pps timebase

Designed for last month's *12-Digit Frequency Counter*, this add-on brings measurement accuracy close to that of an atomic clock.

Teach-In 2014: Raspberry Pi – Part 6

In next month's Teach-In 2014 we will be introducing the Pi's I²C interface. Our *Pi Project* deals with the construction of a digital-to-analogue (DAC) interface and *Perfect Python* looks at error logging. Held over from this issue, our *Home Baking* feature will show you how you can use the Internet to remote control your Raspberry Pi anywhere in the world.



MARCH '14 ISSUE ON SALE 6 FEBRUARY 2014

Published on approximately the first Thursday of each month by Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU. Printed in England by Acorn Web Offset Ltd., Normanton, WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions INLAND: £23.50 (6 months); £43.00 (12 months); £79.50 (2 years). EUROPE: airmail service, £27.00 (6 months); £50.00 (12 months); £95.00 (2 years). REST OF THE WORLD: airmail service, £37.00 (6 months); £70.00 (12 months); £135.00 (2 years). Payments payable to "Everyday Practical Electronics", Subs Dept, Wimborne Publishing Ltd. Email: subs@epemag.wimborne.co.uk. EVERYDAY PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Does your design need class-leading 32-bit MCU performance, code density and ADC throughput?

PIC32MZ offers 330 DMIPS and 3.28 CoreMarks™/MHz performance

Microchip's new PIC32MZ 32-bit MCUs achieve class leading performance, combined with ADC throughput of 28 Msps, 30% better code density and up to 2 MB dual-panel Flash with live update and 512 KB RAM.

The PIC32MZ Embedded Connectivity (EC) family of 32-bit MCUs introduces a breakthrough in high-end embedded control with its class-leading performance and code density in addition to new levels of on-chip memory and peripheral integration.

With up to 2 MB of dual-panel Flash and 512 KB of RAM, the PIC32MZ offers 4x more on-chip memory than any other PIC® MCU, with fail-safe operation during live Flash updates. It is also the first PIC MCU to use the enhanced MIPS microAptiv™ core which adds 159 new DSP instructions that enable the execution of DSP algorithms at up to 75% fewer cycles than the PIC32MX families.

Advanced connectivity is supported over Hi-Speed USB, 10/100 Ethernet and two CAN 2.0b modules as well as multiple UART, SPI/I²S, and I²C channels. The optional on-chip crypto engine ensures secure communication with a random number generator and high-throughput data encryption/decryption and authentication.

DEVELOPMENT TOOL SUPPORT:

- Turn-key PIC32MZ EC Starter Kits
- Multimedia Expansion Board II
- PIC32MZ2048EC Plug-in Module for Explorer 16

PIC32MZ Embedded Connectivity Starter Kit (DM320006 or DM320006-C with crypto engine)

For more information, go to: www.microchip.com/get/eupic32mz

MICROCHIP

Microcontrollers • Digital Signal Controllers • Analog • Memory • Wireless