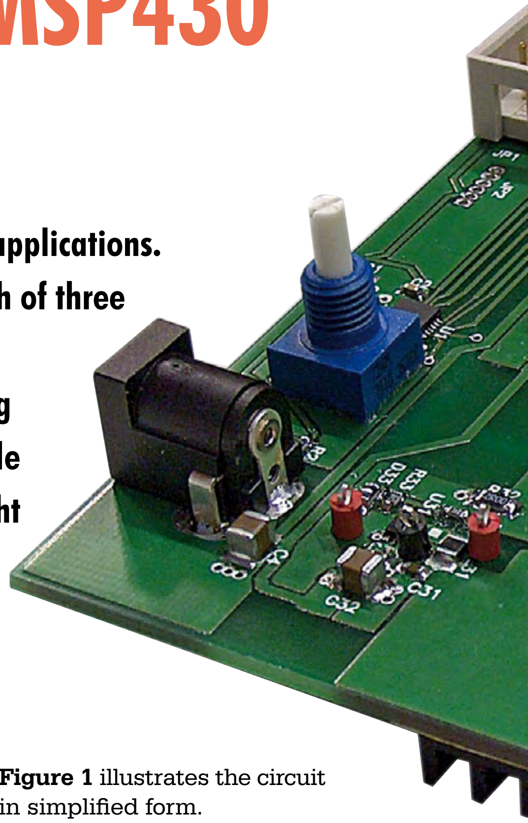
# RGB LED Mood Light

## Ambience courtesy of an MSP430

Dirk Gehrke and Christian Hernitschek



**High-brightness LEDs are being used more and more widely in lighting applications. Here we present a simple 'mood light' using just a few components. Each of three LEDs is provided with a constant current using a switching regulator, and brightness control is performed by an MSP430 microcontroller generating three PWM signals. The printed circuit board could be fitted inside a table lamp with a frosted glass envelope or could be used with an LED spotlight for indirect lighting.**

Whatever their power, LEDs are now normally driven using a constant current source. This is because their light output, measured in lumen (lm), is proportional to the current flow.
All LED manufacturers therefore specify parameters such as light output (sometimes expressed as optical efficiency), viewing angle and wavelength as functions of forward current $I_F$ rather than of forward voltage $V_F$ as might be expected. We therefore use suitable constant current regulators in our circuit.

### Constant current for high-brightness LEDs

The majority of switching regulators on the market are configured as constant voltage sources rather than as constant current sources. A small and easy-to-understand modification to the circuit is required to convert a constant voltage regulator to constant current operation. Instead of the voltage divider normally used to set the output voltage we use a current sense resistor, across which the voltage drop is regulated. **Figure 1** illustrates the circuit in simplified form.

### Dimming LEDs

There are essentially two ways to dim LEDs. The first, and simplest, way is analogue control, where the current flowing through the LED is controlled directly: a lower current gives a lower brightness. Unfortunately there are two severe disadvantages to this method. First, the brightness of the LED is not exactly proportional to the current,

**Caution! High-brightness LEDs!**

***Never look directly into the LEDs!***
Very bright LEDs are not just uncomfortable to look at; they can actually be dangerous to the eyes as they can damage the retina. We therefore recommend operating the board with the LEDs pointing at a white wall to give indirect illumination.

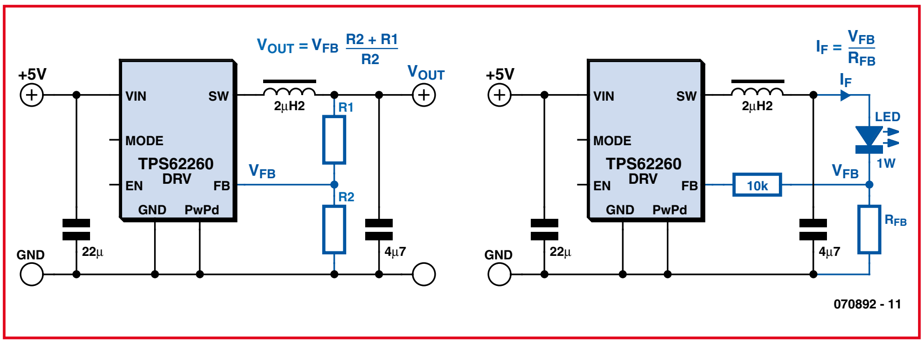



Figure 1. A switching regulator can be configured as a voltage source or as a current source.

ting

and second, the wavelength (and hence colour) of the emitted light shifts as the current is varied away from the nominal value for the LED in question. These two phenomena are seldom desirable.

The slightly more complicated control method uses a constant current source configured to deliver the nominal operating current for the LED. An additional circuit can then quickly switch the LED on and off with a given mark-space ratio, on average emitting less light. This is perceived as a reduction in light intensity. By adjusting the mark-space ratio we can easily adjust the perceived brightness of the LED. This method is known as pulsewidth modulation (or PWM).

## Dimming using PWM

We will look at ways of implementing PWM control using the TPS62260 as an example. The TPS62260 is a synchronous step-down converter with integrated switching element, operating at a typical clock frequency of 2.25 MHz. In the circuit of **Figure 2** we show in black the possibility of connecting the PWM signal directly to the EN (enable) pin. The whole switching regulator circuit is started up and shut down in sympathy with the PWM signal. Experiments in our laboratory indicate that in this configuration we can use a PWM frequency of up to 100 Hz. The advantage of this arrangement is its simplicity: no additional components are required. It is also the most energy-efficient implementation, as the switching regulator draws very little quiescent current when disabled. Its disadvantage is that the reaction of the LED to a high level on the enable pin is delayed. This is because the switching regulator has a 'soft-start' function: when the device is enabled the output current is gradually ramped up until it reaches the nominal LED current. In some applications this ramp can be problematic as the wavelength of the light emitted by the LED varies as the current builds up from its minimum value to the normal operating level. For example, in a DLP projec-

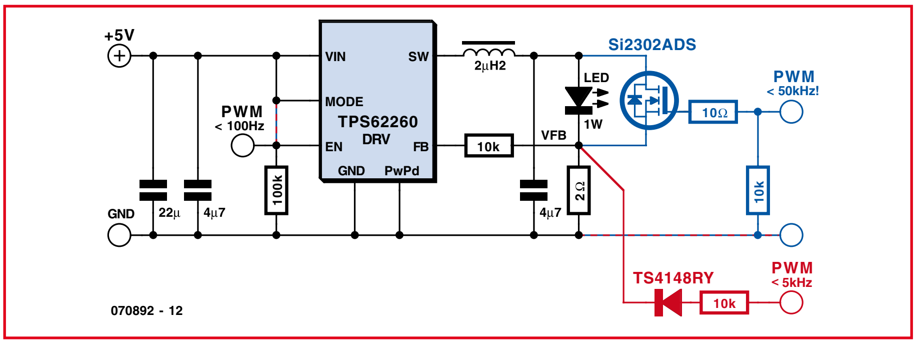



**Figure 2.** Three ways to implement the dimmer function.

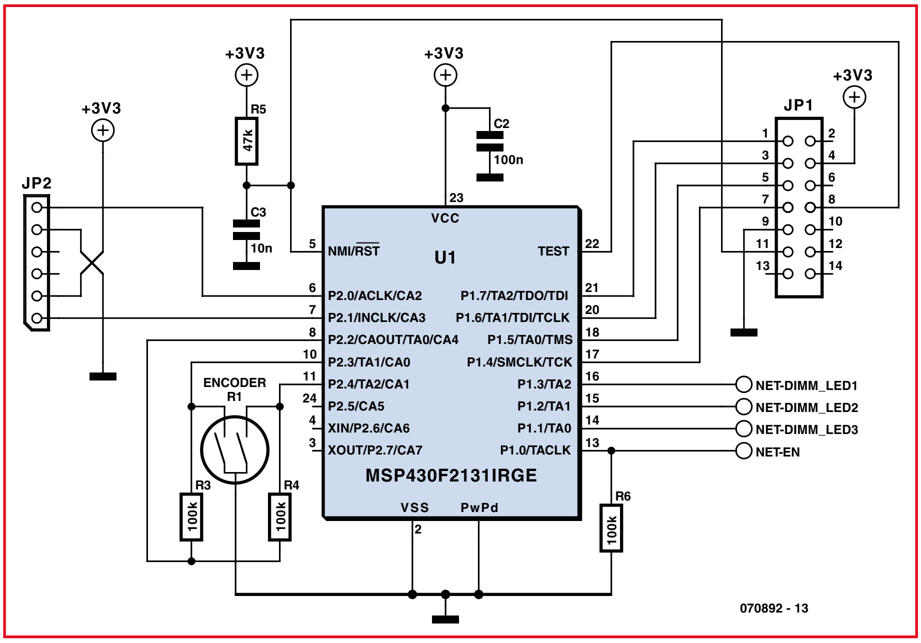



**Figure 3.** Control part of the circuit, based on an MSP430 microcontroller, with JTAG connection (JP1), eZ430 connector (JP2) and rotary encoder (R1).

tor or in the LED backlight for an LCD television panel, such variation might not be acceptable. For this demonstration project, however, the effect is not noticeable to the eye.

In the second variant (shown in red in Figure 2) the PWM signal is coupled into the error amplifier input of the TPS62260 via a small-signal diode. In this circuit a positive voltage in excess of 600 mV applied to the control input will over-drive the error amplifier input and thus switch off the LED. Since this circuit does not use the enable input it does not suffer from the start-up delays associated with the soft-start function of the regulator, and the LED is switched on and off very rapidly. The shift in output wavelength due to the current ramp mentioned above is therefore negligibly small in this configuration. Furthermore, we found in the laboratory that the PWM frequency could be raised to 5 kHz.

The third possibility is shown in blue in Figure 2. Here the PWM signal is used to control a MOSFET wired across the LED. The MOSFET shorts out the LED and allows it to be switched on and off even more rapidly. The regulator is operating in constant current mode, and this current will flow either through the LED or through the MOSFET. Disadvantages of this approach include the additional cost of the MOSFET and poor energy efficiency: up to 180 mW of power can be dissipated continuously in the 2 Ω current sense resistor. Its advantage is the high switching frequency: in experiments we saw successful operation of the TPS62260 in this con-

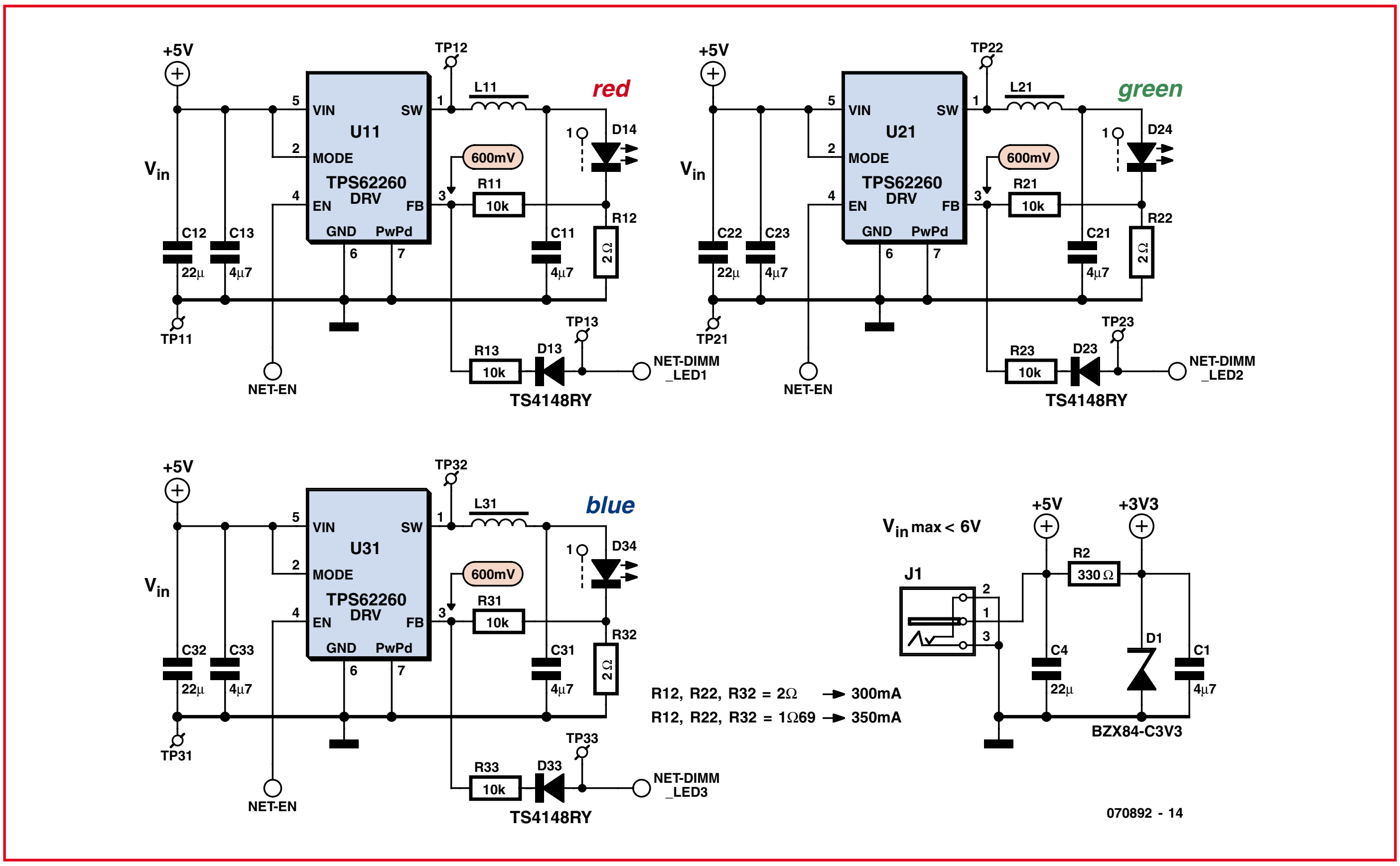



**Figure 4.** Circuit section consisting of three switching regulators configured as constant current sources and a 3.3 V stabilised supply built using discrete components.

## A brief history of LEDs

The story of the light emitting diode (or LED), or 'luminescence diode', starts in earnest in 1962 when General Electric (GE) began to manufacture and sell red LEDs on the commercial market. Customers had to be satisfied with a rather poor light output: an LED of that time had an output of just 0.1 lm/W, corresponding to an efficiency of barely 0.1 %. The material used was a mixed crystal comprising gallium arsenide and gallium phosphide. Since then the market for LEDs has changed out of all recognition, with many other manufacturers making advances in LED technology, and light output available for a given current has increased steadily.

Efficiency, as well as electrical and thermal robustness, needed to increase still further before LEDs could be practically used as light sources in any significant quantity. At the same time, prices have fallen rapidly, opening up entirely new application areas. Such has been the result of forty years of research and development.

Today it is possible to buy very bright LEDs at moderate prices which run at respectable levels of efficiency: for example, the Golden Dragon range from OSRAM Opto (formerly Infineon), the Rebel LED range from Lumileds (Philips Semiconductors) and the X-Lamp range from Cree. There are of course many other manufacturers of high-brightness LEDs, although for reasons of time we were unable to evaluate them all for this article.

The light output of LEDs available today has risen to 20 lm/W, and some examples manage as much as 40 lm/W. These values correspond to efficiencies of 5 % and 10 % respectively, rather higher than the efficiency of a standard commercially-available incandescent bulb, which offers in the region of 10 lm/W, or an efficiency of around 2 %. High-brightness LEDs are already more efficient than halogen bulbs (around 25 lm/W) and will probably also soon be outperforming energy-efficient bulbs (around 60 lm/W).

figuration with PWM frequencies as high as 50 kHz.

### The practical circuit

At the heart of the circuit (**Figure 3** and **Figure 4**) is an MSP430F2131 microcontroller. This is programmed to operate as a triple PWM generator and to read values from the rotary encoder (R1). The encoder value is used to index a look-up table containing mark-space ratio values for each of the red, green and blue LEDs. The corresponding PWM signals are then made available on output pins TA0, TA1 and TA2 at a frequency of approximately 122 Hz. This is high enough to ensure that the LEDs do not appear to flicker, as the eye smoothes out the individual pulses of light to an average perceived intensity value.

For a practical implementation we chose the PWM control method shown in red in Figure 2, which gives a good compromise between circuit complexity and performance. Each LED, red (D14), green (D24) and blue (D34) is supplied with a constant current from a separate TPS62260 DC/DC converter. The 2 Ω resistor sets the nominal current flowing through the LED at 300 mA. Higher currents (up to 1 A) can be obtained using a TPS62290, the TPS62260's 'big brother', which comes in the same package style.

## Clock generation

MSP430 microcontrollers have a choice of integrated clock sources. In software, the MSP430 can switch between an external crystal-based oscillator and an internal RC-oscillator. To keep circuit costs down we have dispensed with external components and used the internal calibrated RC oscillator. 'Calibrated' means that the calibration parameters, stored in the MSP430's 'information memory', simply need to be copied into the relevant control registers in the clock generator module. Using these calibration parameters gives an overall accuracy for the RC oscillator of ±2.5 % over the temperature range from 0 °C to 85 °C. The RC oscillator frequency lies between 7.8 MHz and 8.2 MHz: this frequency is used as the CPU clock frequency and to drive the counter in the Timer_A module.

## Implementation of triple PWM

The Timer_A module in the MSP430 consists of a counter block and a range of capture and compare blocks. The frequency of the generated PWM signals is determined by the rate at which the counter overflows. Since the Timer_A counter is 16 bits long, the PWM frequency is given by $f_{PWM} = f_{IN}/2^{16}$ = 8 MHz/65536 = 122.07 Hz, where $f_{IN}$ is the frequency of the clock input to Timer_A.

$$f_{PWM} = \frac{f_{input}}{2^{16}} = \frac{8\,MHz}{65536} = 122.07\,Hz$$

$f_{PWM}$ : *PWM signal frequency*

$f_{input}$ : *Timer_A frequency – input clock*

If we repeat this calculation using the minimum and maximum frequencies given above, we obtain the maximum deviation of the PWM signal frequency from its nominal value. We find that 119 Hz < $f_{PWM}$ < 125 Hz.

$$119\,Hz < f_{PWM} < 125\,Hz$$

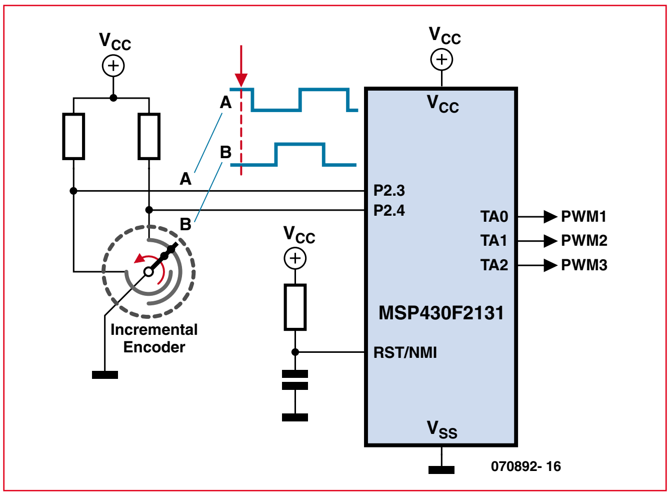
Generation of the PWM signals themselves is carried out by the 'output units' which form a part of each capture and compare block. In the MSP430F2131 the Timer_A module has a total of three capture and compare blocks and therefore three output units. Each capture and compare block consists of a digital comparator which compares the current value in the counter with a value specified independently for each block (TACCR0, TACCR1, and TACCR2). If the values match then the comparator output triggers the output unit, setting the corresponding PWM output to a '1'. The PWM outputs are reset in software. The overflow of the 16-bit counter causes an interrupt; the interrupt service routine sets all the PWM outputs to zero in turn.

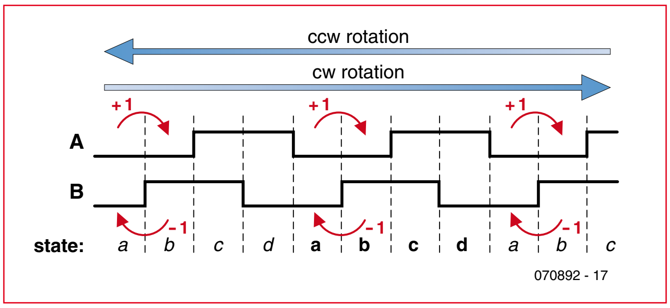
Using software to reset the PWM outputs puts a limit on the available range of mark-space ratios. Execution of the Timer_A interrupt service routine takes approximately 100 cycles, and so the three colour table arrays may only contain values in the range from 100 to 65535.

## Rotary encoder

The PWM mark-space ratios can be set manually using a rotary encoder (or 'shaft encoder'), which is a device similar in appearance to a potentiometer. However, instead of containing a resistive track, it employs two contacts which open and close as the shaft is turned in a 2-bit Gray code pattern. The internal construction of the encoder is very simple. A wiper with two contacts sweeps over two conducting rings, insulated from one another. An insulating material covers the rings in a pattern such that as the wiper turns it operates as a switch, producing the two-bit Gray code on the output pins.

The upper figure shows in outline how the encoder is connected to the microcontroller, and the lower figure shows the output signals when the shaft is turned steadily in either direction.




Using the two signals A and B we can detect when the shaft is turned, as well as in which direction. In the timing diagram four states, a, b, c and d, are shown. These states repeat continuously as the shaft is turned. If the MSP430 software detects a change from state a to state b, it knows that the colour table pointer LEDptr needs to be incremented. Conversely, a change from state b to state a causes the pointer to be decremented.

If the encoder oscillates between states a and b the pointer will be alternately incremented and decremented. This can give rise to a flickering of the LEDs as the settings change to and fro. For this reason (as well as to reduce the effective resolution of the encoder to a more comfortable value for the user) the pointer LEDptr is divided by four before it is used to access the colour table arrays.

Finally, a note on the wiring of the rotary encoder. In the circuit diagram of Figure 3 the pull-up resistors for each contact are connected to pin 8 of the MSP430 (P.2.2) rather than to VCC. This is not a mistake: P2.2 is taken high in software and is therefore at 3.3 V, the same voltage as VCC. Of course, the pull-up resistors could be connected directly to VCC (3.3 V), freeing up P2.2 for other purposes.

## COMPONENTS LIST

**Resistors**
(SMD 0603 unless otherwise stated)
R2 = 330Ω
R3,R4,R6 = 100kΩ
R5 = 47kΩ
R11,R13,R21,R23,R31,R33 = 10kΩ
R12,R22,R32 = 2Ω (SMD 1206)

**Capacitors**
C1,C11,C13,C21,C23,C31,C33 = 4µF7 6.3V; X5R, (SMD 0603)
C2 = 100nF (SMD 0603)
C3 = 10nF (SMD 0603)
C4,C12,C22,C32 = 22µF (SMD1210)

**Semiconductors**
D1 = BZX84-C3V3 (SMD SOT23)
D13,D23,D33 = TS4148 RY (SMD 0805)
D14 = 1W LED, Golden Dragon, red (Osram)*
D23 = 1W LED, Golden Dragon, green (Osram)*
D33 = = 1W LED, Golden Dragon, blue (Osram)*
U1 = MSP430F2131RGB (TI)
U11,U21,U31 = TPS62260DRV SMD SON-6 (TI)

**Inductors**
L11,L21,L31 = 2µH2, 1.1 A, 110 mΩ, SMD 2x2.5 mm (MIPSA2520D2R2, FDK)

**Miscellaneous**
R1 = rotary encoder, Bourns 3315-001
JP1 = 14-way boxheader
JP2 = 6-way connector (Samtec TMS-106-XX-X-S-RA)
TP11,TP12,TP13,TP21,TP22,TP23,TP31,TP32 ,TP33 = test pin, e.g. Keystone 5001
Heatsink, Fischer SK 477 100
Heat conducting self-adhesive tape, Fischer WLFT 404 R25
PCB, order code 070892-2**

***LED alternatives:**
Lumiled REBEL LED using PCB 070892-1**
CREE XLAMP LED using PCB 070892-3**

** Artwork download and PCB ordering at www.elektor.com

## Colour table

The colour look-up table takes the form of an array stored in the MSP430. The array is arranged so that it can at any time be extended with additional pulse width modulation values for the red, green and blue LEDs. Whenever the rotary encoder is turned new red, green and blue values are read from the array and used to generate the three PWM output signals. Currently 252 values are stored, which can be changed if desired. A decimal value of 100 switches the LED off, and a value of 65535 produces a mark-space ratio of 100 %.

When the 5 V supply is applied the MSP430 goes into a demonstration mode where the values stored in the array are read and output in sequence in an infinite loop. As soon as the rotary encoder is turned the sequence stops and a particular fixed colour value can be selected.

The PWM signal is coupled in using a small-signal diode (D13, D23 and D33). When the PWM signal is high it overrides the normal error signal input of the corresponding switching regulator, which has a threshold voltage level of 600 mV. This means that a high level on the PWM signal forces the LED to extinguish. When the PWM signal subsequently goes low the regulator starts up again and the LED lights.

The whole circuit is powered from a regulated 5 V 1 A DC mains adaptor. A simple voltage stabiliser built using a resistor and a Zener diode reduces the 5 V level to 3.3 V for the MSP430 microcontroller.

The circuit can be built on the printed circuit board shown in **Figure 5**. There are three versions of the circuit board differing only in the footprint and connection arrangement of the LEDs. This allows various types of LED to be used. The LED options available are listed in the parts list.

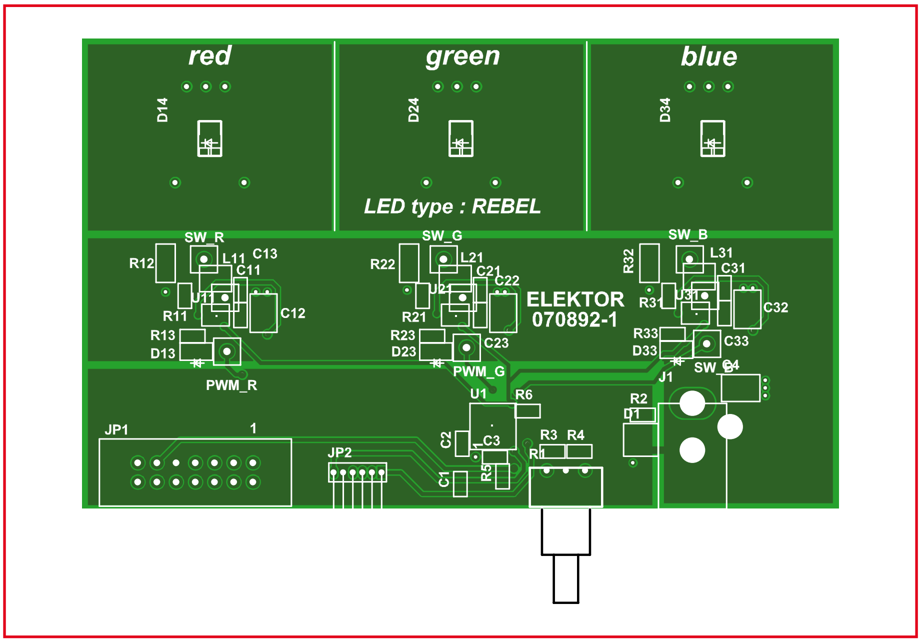
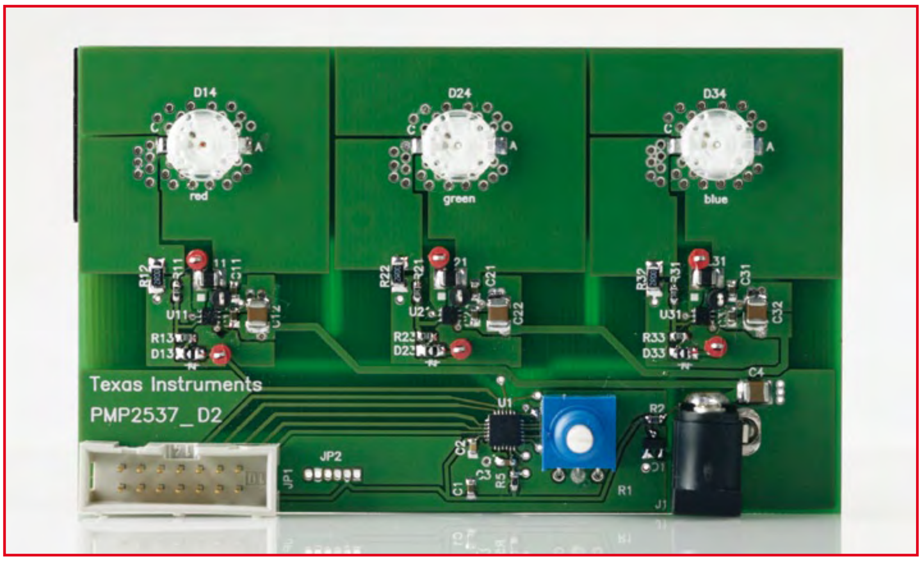
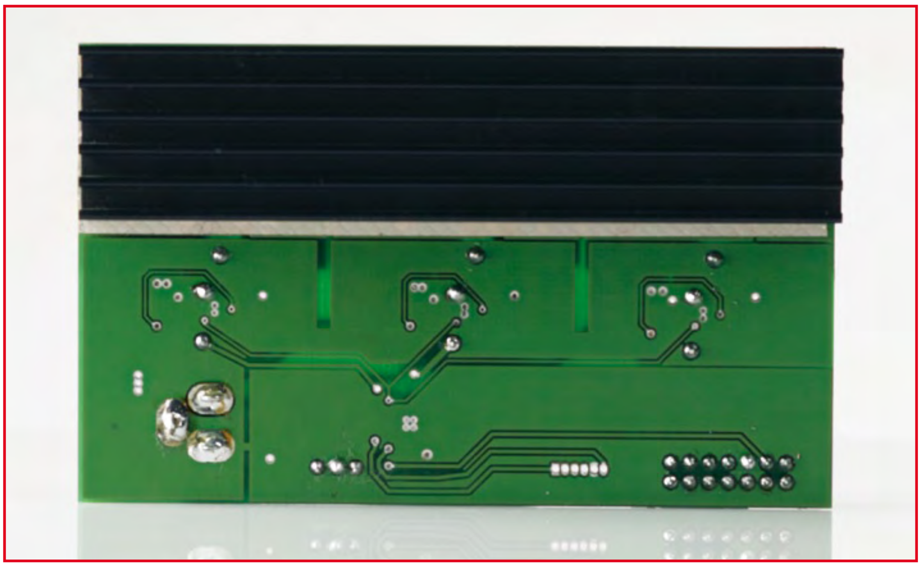



**Figure 5.** Printed circuit board for building the circuits of Figure 3 and Figure 4. Three variants are available for download, supporting different types of LED.



## Heat map

Operating temperature is an important parameter in the performance of a high-power LED. It strongly affects operating life, forward voltage, output wavelength and even the brightness of the device. The higher the operating temperature of the LED, the shorter will be its expected lifetime. For this reason the dimensions of our experimental printed circuit board have been chosen to allow a type SK477100 heatsink (made by Fischer Elektronik) to be fixed to the reverse of the board using double-sided adhesive thermal transfer material. Running at full power, this reduces the temperature of the LEDs from 61 °C (without heatsink) to 54 °C (with heatsink). The heatsink also helps to spread the dissipation of heat over the area of the printed circuit board.

To make an example thermal image we populated the board with LEDs from Cree. **Figure 6** shows the results vividly, illustrating the temperature of the LEDs without heatsink (on the left) and with heatsink (on the right).

## Software

The source code for the MSP430 software for this application is available for download from the Elektor website. The code begins by including the

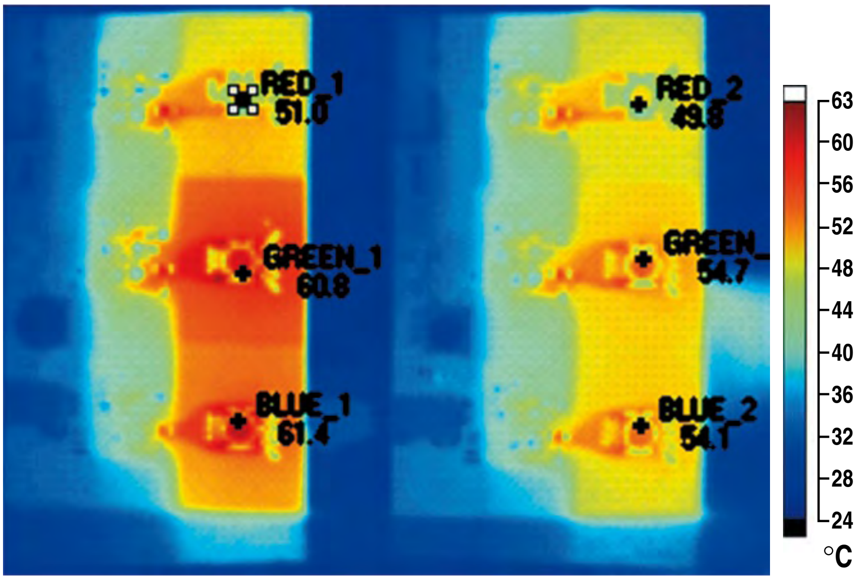



**Figure 6.** Thermal image of the circuit board, populated with LEDs from Cree. Left: without heatsink; right: with attached heatsink.

'MSP430F21x2.h' header file, which contains definitions of all the control register names and of the control bits available in the MSP430. Next the length of the colour table is defined. Notice here that the value of 'LED_TabLength' is actually set to four times the length of the table. Then follows the colour table itself, with a separate array for each individual LED. Pointer 'LEDptr' is used to read the relevant PWM mark-space ratio settings for each of the three outputs from the individual colour table arrays: see also the text box 'Colour table'.

The microcontroller is initialised at the beginning of the function 'main()'. The watchdog timer is disabled, the calibration values for the adjustable system clock are loaded, the Timer_A module is configured and the multiplexed inputs and outputs are suitably initialised. The main loop consists of two 'while' blocks. In the first 'while' block the colour table pointer LEDptr is incremented, which results in a continuous change in the PWM mark-space ratios and thus in the generated colour. The overall timing of these colour changes is governed using two nested 'for' loops. The first 'while' loop runs until the rotary encoder reports a change on one of its outputs. The second 'while' block, written as an infinite loop, then takes control: it increments or decrements the colour table pointer according to the direction in which the rotary encoder is turned.

## A bright future

The printed circuit board allows additional functionality to be implemented. For example, there is a socket for a Texas Instruments eZ430-RF2500 radio module. The eZ430-RF2500 kit is supplied with two radio modules. One of these can be fitted with a rotary encoder (using the test pins on the microcontroller in the radio module), creating a radio link to the LED board.

The circuit board described here is primarily intended for experimentation and evaluation purposes. Since the MSP430 source code is made available, it is possible to modify it for a range of other projects. The switching regulators can also find use in other applications: have fun!

(070892-I)

## About the authors

**Dipl. Ing. (FH) Dirk Gehrke**
Dirk Gehrke was born in Münster in Germany and studied communications technology at Dortmund University of Applied Sciences and Arts. He started working for Texas Instruments in 1998 as a Field Application Engineer (FAE) in Britain, France and the United States. From 2000 he worked in Freising, Germany, as an FAE for power management products, and in January 2006 he became Business Development Manager for analogue products in EMEA (Europe, the Middle East and Africa). Contact: http://www.ti.com/europe/csc.

**Dipl. Ing. (FH) Christian Hernitscheck**
Christian Hernitscheck studied electronic engineering at the Landshut University of Applied Sciences in Germany, specialising in microelectronics. Since 1998 he has worked as an FAE for Texas Instruments covering the whole of Europe, focussing on the MSP430 microcontroller product line. Contact: http://www.ti.com/europe/csc.