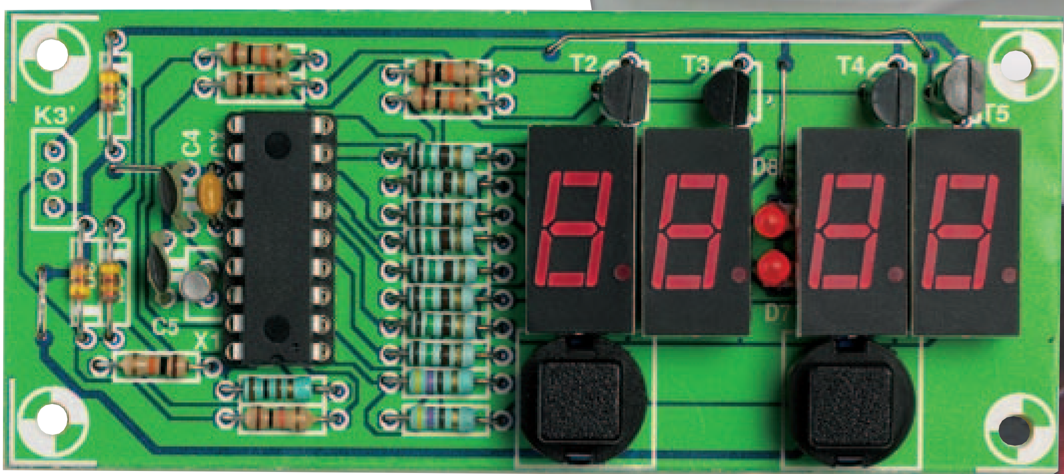
Timer Switch for Washing Machine

Switching on command

Goswin Visschers



Many households have a utility supply with a separate day/night tariff, where the electricity board delivers cheaper power during the night and weekends. It can be economical to use, for example, the washing machine at that time. That is, as long as you're actually there to turn it on at the right time, because modern electronically driven washing machines cannot be delayed with a simple timer switch. But with a handful of components you can build yourself a circuit than can actually do this!

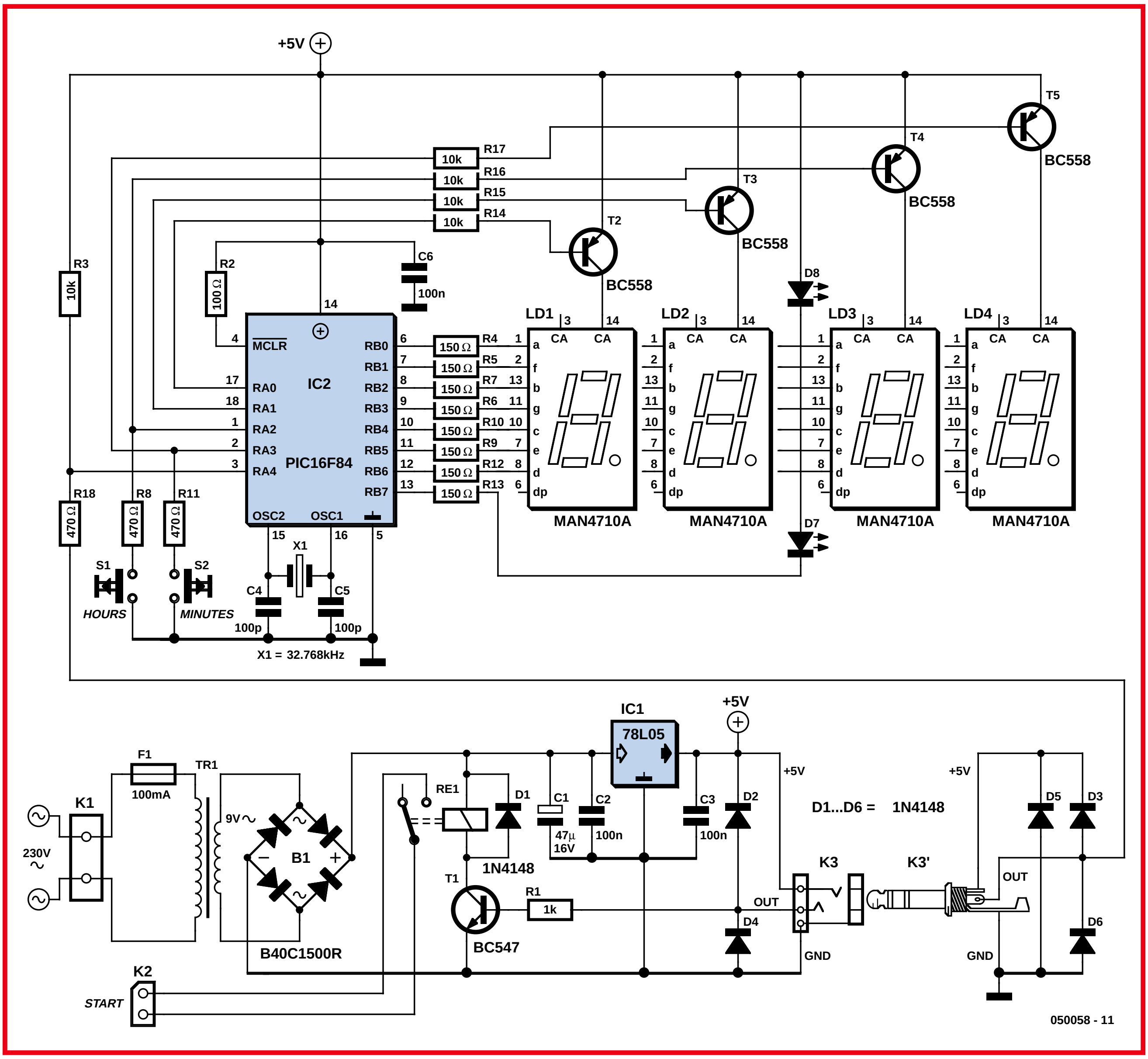


Figure 1. The schematic for the timer switch consists of two parts, each of which has its own PCB.

This electronic timer switch starts the washing machine by bridging the pushbutton for the start button for a few seconds using a relay. This does, however, require a small change to the washing machine. It is necessary to solder a wire to each side of the start pushbutton on the washing machine control PCB and to connect these wires to the relay contacts.

The timer switch is built on two circuit boards. The power supply PCB with the relay will go inside the washing machine. The PCB with the microcontroller is mounted in a separate enclosure and is attached to the outside of the washing machine.

The connection between the two PCBs (switch signal and power supply) is achieved with a 3.5-mm stereo jack plug and mating socket.

The design

Figure 1 shows the schematic for the complete timer switch. The top section is the control part with a PIC16F84 and four seven-segment displays. The bottom section is the power supply and switching part that will be built into the washing machine.

Power supply PCB

The power supply PCB is a straightforward design. A small mains transformer provides an AC voltage of about 9 V. The direct voltage resulting from rectification and filtering is regulated by IC1 into a supply voltage of 5 V for the entire circuit.

The relay, whose contacts bridge the start button, is energised with the aid of switching transistor T1. The purpose of diodes D2 and D4 is to protect the base of T1 when the jack plug is either connected or disconnected.

If the washing machine has an on/off-switch instead of a start button, then this switch can be bridged with a bigger relay instead (however, take note of the maximum power rating of the washing machine).

This also requires a small change to the software: after the washing machine has been switched on by the timer switch, the relay needs to remain energised until the clock is reset.

Control PCB

In this part of the timer switch, the objective was to use as few electronic parts as feasible and integrate as much functionality as possible into the microcontroller.

Diodes D3, D5 and D6 have the same purpose as D2 and D4, that is, protection for the ports when the jack plug is connected or disconnected.

X1 is a 32.768 kHz quartz crystal, the same type as those used in watches. This frequency is a power of two, which makes it easy to divide it down to a half second (half the time period of the colon).

The pins from the microcontroller are used as follows. Pins RB0 through to RB6 are used to drive the seven segments of the displays in multiplexed fashion. Resistors R4 to R13 limit the current through the LED-segments. Pin RB7 is for the colon (LEDs D7 and D8). Pins RA0 through RA3, via driver transistors T2 to T5, generate the multiplex drive signals for the displays. In addition, RA2 and RA3 are read once per cycle of the 'main loop' to test the state of the two pushbuttons S1 and S2.

In case you would like to modify the software yourself, it is possible to use RA0 and RA1 in the same manner as RA2 and RA3 and allow for two more pushbuttons.

RA4 is used to switch the relay via T1. Since this is an open-collector output, the purpose of R3 is to supply the drive current for T1.

Construction and Installation

Figure 2 shows the PCB layout that has been designed for this timer switch. Here, the two parts are joined together, but normally the two PCBs will be mounted in different locations. The assembly of the PCBs is mostly routine, but don't forget the three wire links on the control PCB.

The construction and installation of this circuit has already been mentioned at the start of this article. You can choose to keep the two parts completely separate. That is, the power supply PCB inside the washing machine and the control PCB outside. The front of the washing machine could be fitted with a jack socket for the interconnection with the control PCB. The control enclosure can then be fastened on the washing machine somewhere (for example with magnetic stickers).

It is also possible to fit the entire circuit inside the washing machine, but this is a much more challenging task because of the available space and the openings that are required for the front

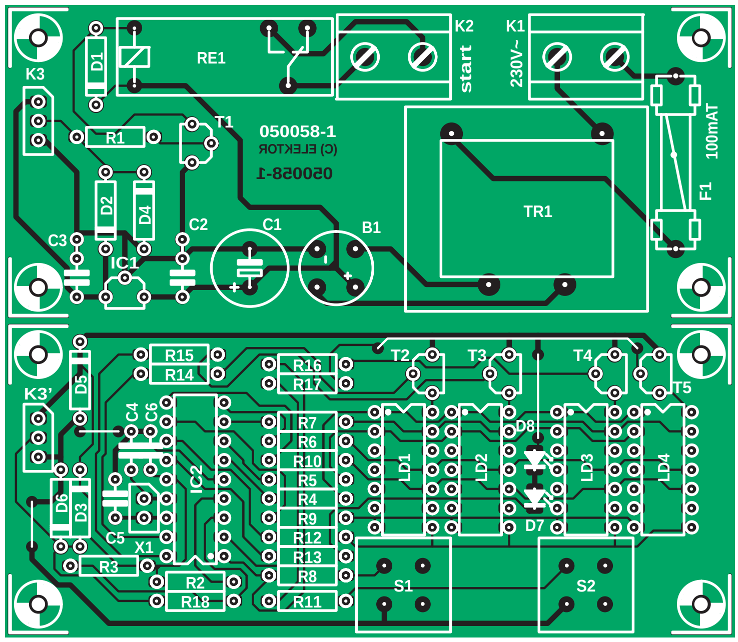


Figure 2. The two PCBs are shown here joined together, but in practice each one will be mounted in a different location.

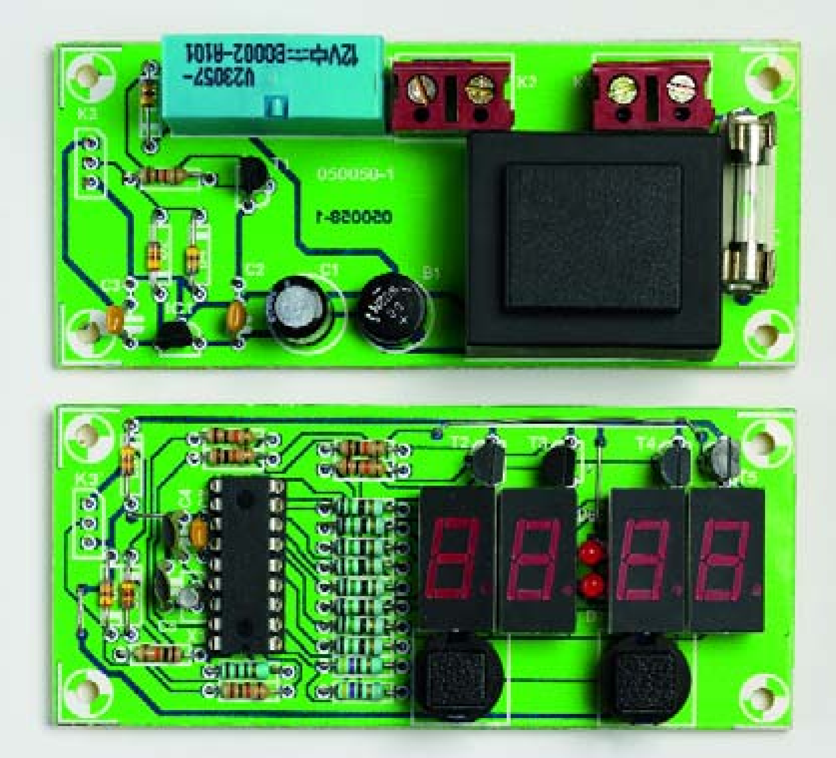


Figure 3. The prototype shown here differs slightly from the final printed circuit boards.

COMPONENTS LIST

Resistors:
R1,R18 = 1kΩ
R2 = 100Ω
R4-R7,R9,R10,R12,R13 = 150Ω
R8,R11 = 470Ω
R3,R14-R17 = 10kΩ

Capacitors:
C1 = 47µF 16V radial
C2,C3,C6 = 100nF
C4,C5 = 100pF

Semiconductors:
B1 = B40C1500R bridge rectifier (40V piv, 1.5A)
D1-D6 = 1N4148
D7,D8 = LED, red, 3mm diam.
IC1 = 78L05
IC2 = PIC16F84, programmed, order code **050058-41**
LD1-LD4 = 7-segment display, e.g. Fairchild type MAN4710A
T1 = BC547B
T2-T5 = BC558

Miscellaneous:
F1 = fuse, 100mAT (slow) with PCB mount holder
K1,K2 = 2-way PCB terminal block, lead pitch 7.5mm
K3,K3' = 3.5mm stereo jack plug with socket
R1 = 12V relay (e.g. Siemens V23057-12V)
S1,S2 = pushbutton (e.g., ITT D6-R-90)
Tr1 = mains transformer, 230V primary, secondary 9V @ 1VA (e.g. Block VB1,0/1/9)
X1 = 32.768kHz quartz crystal
PCB, order code **050058-1**
Disk, PIC source and hex code, order code **050058-11**

panel. This is why the author chose for the solution with the jack socket.

Operation

The operation is extremely easy. After the circuit is switched on, four dashes are shown on the display ('——').
With one pushbutton (S2) you can increase the minutes in steps of 10 minutes, with the other pushbutton (S1), the hours in steps of 1 hour. Once the buttons have not been pressed for 1 second, the timer will begin to count down automatically.
During the countdown, the time remaining is visible on the displays.

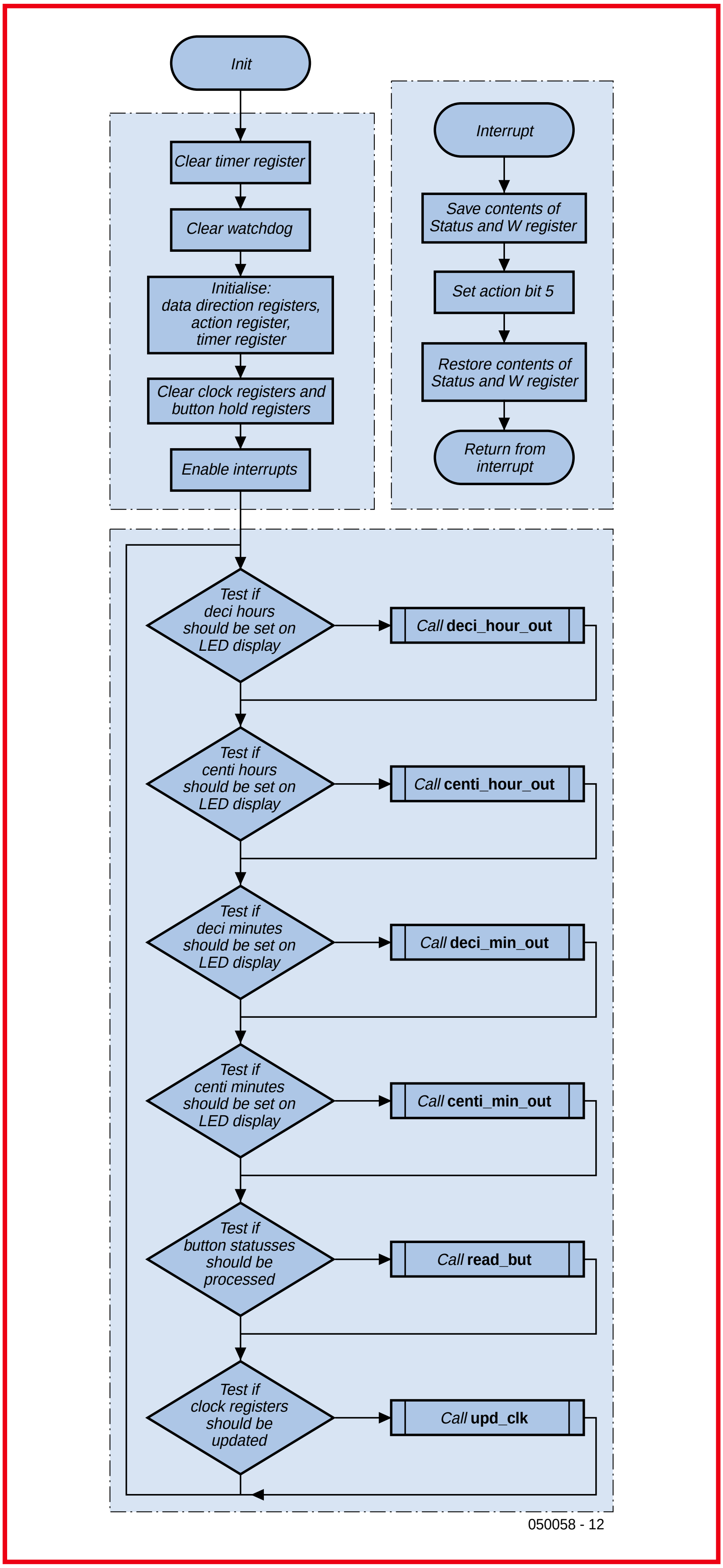


Figure 4. This flowchart shows the design of the main program.

When the time is up, the relay is energised and the display is turned off.
After the relay is turned off again, the four dashes reappear on the display.
If both pushbuttons are pressed at the same time, the timer switch will be reset and the four dashes will appear on the display.

Software

The code in the microcontroller has been designed to be as general-purpose as possible, so that this circuit can be used for other applications by changing a few constants.
The code comprises two files: *clock.asm* that contains the machine code and *clock.inc*, which contains the constants and register definitions. These files are available free of charge from the *Elektor Electronics* website (and on floppy disk, order code **050058-11,** for those without access to the Internet).
The heart of the machine code file naturally consists of the main loop (lines 105 to 118).
The bits in the 'action_reg' register are tested in the 'main loop' in order to determine the next action (see comments in the code).
The seven-segment displays are driven in turn, after which the pushbuttons are tested. Then, depending on whether an interrupt has taken place, the time is updated.
If the output has to remain energised permanently, instead of momentarily, the 'goto acti_time' command has to be replaced with 'return'.

A few constants can be changed:

active_time
Duration that the output is energised in half seconds. (not relevant if the output has to stay on permanently.)

hold_but_cyc
Number of cycles to wait before a button can be operated again.

Zero_led...Err_led
Bit values that display a number on the seven-segment display.
In the event the seven-segment displays are connected differently, the numbers can be displayed correctly via Port B by suitably changing these values.

dub_dot_bit
Bit value which represents the location of the colon.
In this case the value is such that the two LEDs for the colon are driven via Port RB7.

setm_but and *seth_but*
Port numbers for Port A to which the minutes and hours pushbuttons are connected.

For further clarification, **Figure 4** shows a flow chart of the main program. The charts for the various program parts referred to may be downloaded as a PDF file from the *Elektor Electronics* website.
You can find more details in the source code file supplied to us by the author.

(050058-1)

CONSTRUCTION GUIDELINES

Elektor Electronics (Publishing) does not provide **parts and components other than** PCBs, fornt panel foils and software on diskette or IC (not necessarily for all projects). Components are usually available form a number of retailers – see the adverts in the magazine.

Large and small values of components are indicated by means of one of the following prefixes :

E (exa) = 10^{18}	a (atto) = 10^{-18}
P (peta) = 10^{15}	f (femto) = 10^{-15}
T (tera) = 10^{12}	p (pico) = 10^{-12}
G (giga) = 10^{9}	n (nano) = 10^{-9}
M (mega) = 10^{6}	μ (micro) = 10^{-6}
k (kilo) = 10^{3}	m (milli) = 10^{-3}
h (hecto) = 10^{2}	c (centi) = 10^{-2}
da (deca) = 10^{1}	d (deci) = 10^{-1}

In some circuit diagrams, to avoid confusion, but contrary to IEC and BS recommandations, the value of components is given by substituting the relevant prefix for the decimal point. For example,
3k9 = 3.9 kΩ   4μ7 = 4.7 μF

Unless otherwise indicated, the tolerance of resistors is ±5% and their rating is ⅓–½ watt. The working voltage of capacitors is ≥ 50 V.

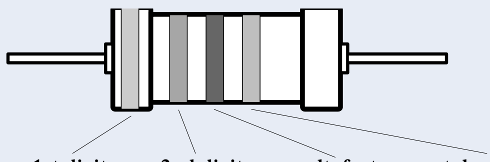
In **populating a PCB**, always start with the smallest passive components, that is, wire bridges, resistors and small capacitors; and then IC sockets, relays, electrolytic and other large capacitors, and connectors. Vulnerable semiconductors and ICs should be done last.

Soldering. Use a 15–30 W soldering iron with a fine tip and tin with a resin core (60/40) Insert the terminals of components in the board, bend them slightly, cut them short, and solder: wait 1–2 seconds for the tin to flow smoothly and remove the iron. Do not overheat, particularly when soldering ICs and semiconductors. Unsoldering is best done with a suction iron or special unsoldering braid.

Faultfinding. If the circuit does not work, carefully compare the populated board with the published component layout and parts list. Are all the components in the correct position? Has correct polarity been observed? Have the powerlines been reversed? Are all solder joints sound? Have any wire bridges been forgotten?
If voltage levels have been given on the circuit diagram, do those measured on the board match them – note that deviations up to ±10% from the specified values are acceptable.

Possible corrections to published projects are published from time to time in this magazine. Also, the readers letters column often contains useful comments/additions to the published projects.

The value of a resistor is indicated by a **colour code** as follows.



color	1st digit	2nd digit	mult. factor	tolerance
black	–	0	–	–
brown	1	1	×10^{1}	±1%
red	2	2	×10^{2}	±2%
orange	3	3	×10^{3}	–
yellow	4	4	×10^{4}	–
green	5	5	×10^{5}	±0,5%
blue	6	6	×10^{6}	–
violet	7	7	–	–
grey	8	8	–	–
white	9	9	–	–
gold	–	–	×10^{-1}	±5%
silver	–	–	×10^{-2}	±10%
none	–	–	–	±20%

Examples:
brown-red-brown-gold = 120 Ω, 5%
yellow-violet-orange-gold = 47 kΩ, 5%