February 1996
£ 2·65

ELEKTOR ELECTRONICS

THE INTERNATIONAL ELECTRONICS MAGAZINE

More circuits from our Design Competition

AUDIO & HI-FI • COMPUTERS & MICROPROCESSORS • DESIGN IDEAS • RADIO, TELEVISION & COMMUNICATIONS • SCIENCE & TECHNOLOGY • TEST & MEASUREMENT

SIMM TESTER

FM receiver in SMT

I²C interface for Centronics port

Measuring capacitance

9 770268 451050 02

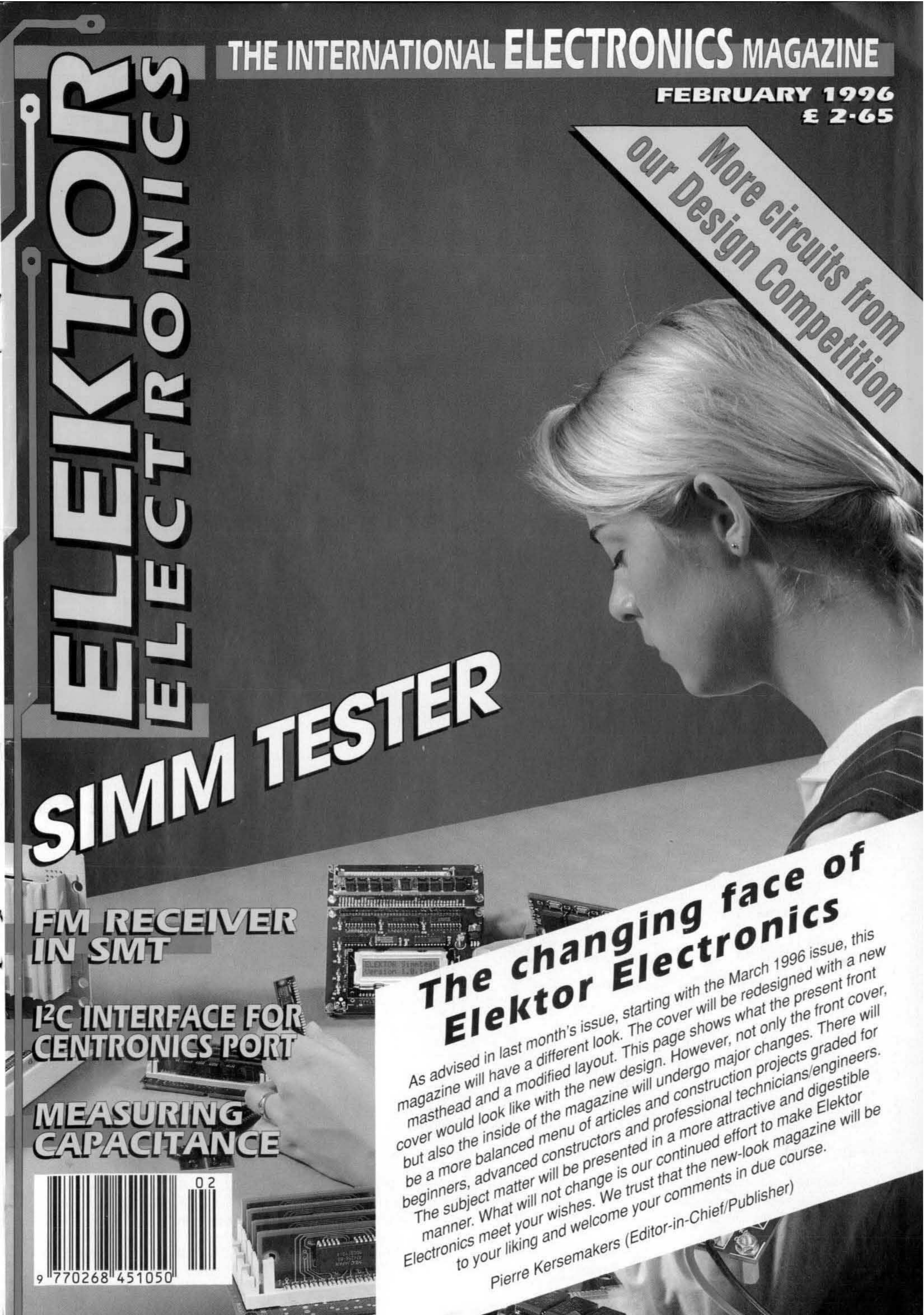
ELEKTOR ELECTRONICS

THE INTERNATIONAL ELECTRONICS MAGAZINE

FEBRUARY 1996

£ 2·65

More circuits from our Design Competition



SIMM TESTER

FM RECEIVER IN SMT

I²C INTERFACE FOR CENTRONICS PORT

MEASURING CAPACITANCE

02

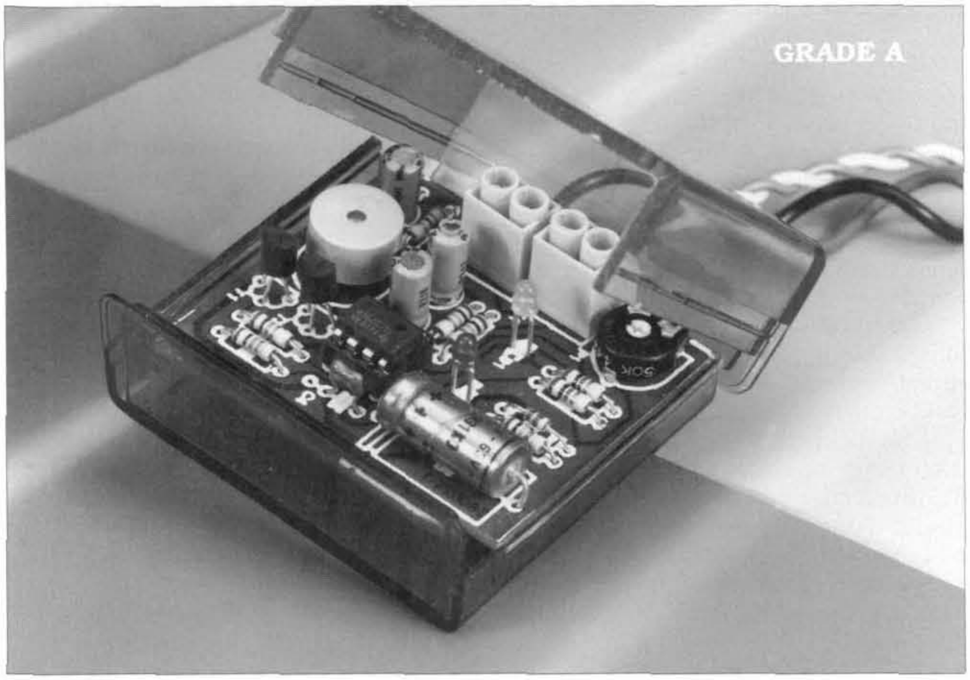
9 770268 451050

The changing face of Elektor Electronics

As advised in last month's issue, starting with the March 1996 issue, this magazine will have a different look. The cover will be redesigned with a new masthead and a modified layout. This page shows what the present front cover would look like with the new design. However, not only the front cover, but also the inside of the magazine will undergo major changes. There will be a more balanced menu of articles and construction projects graded for beginners, advanced constructors and professional technicians/engineers. The subject matter will be presented in a more attractive and digestible manner. What will not change is our continued effort to make Elektor Electronics meet your wishes. We trust that the new-look magazine will be to your liking and welcome your comments in due course.

Pierre Kersemakers (Editor-in-Chief/Publisher)

ICY ROADS WARNING



Winter landscapes may be beautiful, but winter weather can be treacherous for motorists: the difference between wet and icy roads is not easy to see, particularly in the dark. The circuit described measures the temperature just above the road surface and gives a warning when this gets close to freezing point. A useful instrument for any car.

Design by K. Walraven

As early as September (particularly in northern Europe and Canada) there is a likelihood of icy roads, especially late at night or early in the morning. These may occur in certain locations only: when you leave home, all looks well until 10 miles further on you take a bend at your usual speed and find the car's wheels drifting: a potentially dangerous situation. The risk of it arising is lessened by the circuit described. The outside temperature is monitored by a sensor close to the ground and a warning given to the driver when the temperature drops, or is, close to zero.

A safety margin is built in, since it is clearly impossible to monitor the actual road temperature, which may be below freezing point, whereas the air just above it is still above zero. This is achieved by mounting the sensor as low as possible, for instance, inside one of the bumpers, and setting the border line between 'safe' and 'unsafe' at, say 3 °C (38 ° F). When the temperature is above that level, a green LED lights, if it is below it, a red LED lights. Optionally, an audible warning may be given in the latter case.

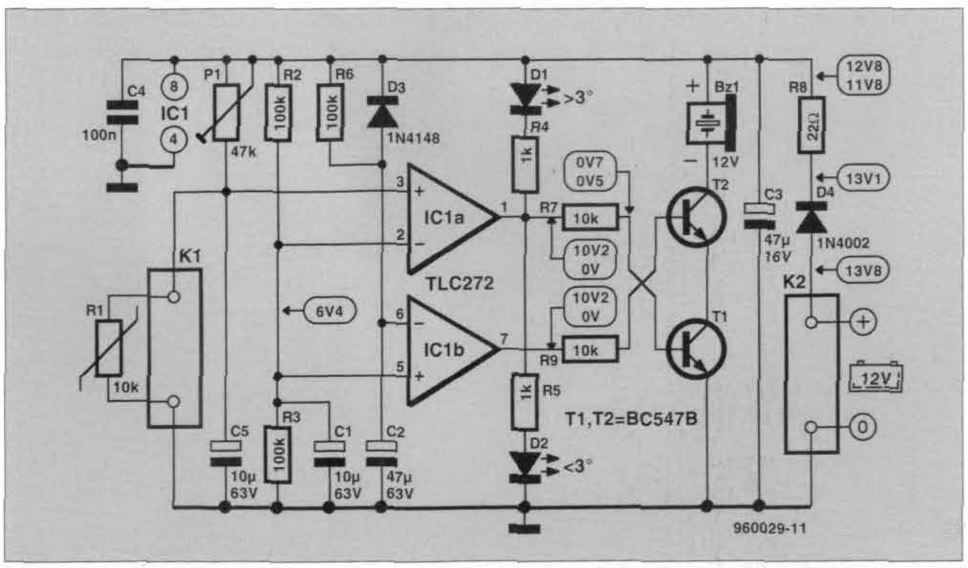


Fig. 1. The circuit of the icy roads warning unit is an example of design simplicity. The temperature is formed by a resistor with negative temperature coefficient and the display consists of two LEDs.

Circuit description

From the circuit diagram in **Fig. 1**, it is seen that the design is straightforward. The temperature sensor is formed by R_1, a resistor with negative temperature coefficient (NTC), which is connected to K_1.

The sensor is part of as bridge circuit; the other parts are P_1, R_2, and R_3. The benefit of a bridge circuit is that both branches are affected equally by voltage variations, which makes their combined effect negligible. The junctions of the bridge are linked to the inputs of comparator IC_{1a}. Capacitors C_1 and C_5 suppress any spurious voltages.

A resistor with NTC has the property that its resistance rises when the ambient temperature drops. When the temperature falls below a set value, the resistance of R_1 will exceed that of P_1. In that case, the potential at the non-inverting (+) input of IC_{1a} becomes higher than that at the inverting (–) input. The voltage at the –input is set to half the supply voltage, U_B, by R_2-R_3. The output of the comparator is logic low at (relatively) high temperatures, causing the green LED to light, and logic high at (relatively) low temperatures, resulting in the red LED lighting.

The indicator LEDs are low-current types that light brightly at currents as low as 1 mA. This is right, because the output current of the op amp is limited to about that level. It can provide a current higher than 1 mA, but this is at the expense of the accuracy of the output signals.

The audible warning signal is generated by the (optional) remainder of the circuit. When the output of IC_{1a} becomes logic high (low temperature), transistor T_1 is switched on via R_7, and this energizes the buzzer. This is,

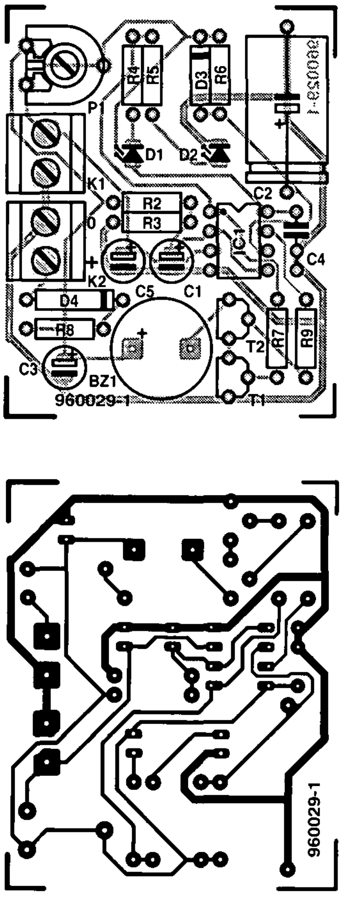


Fig. 2. The printed-circuit for the icy roads warning unit is available through the Readers Services (p. 70).

however, not actuated until T_2 is also switched on. This cannot happen until five seconds after the supply has been switched on: the time taken for C_2 to be charged via R_6. When the potential across C_2 has risen to half the supply voltage, the output level of IC_{1b} will change from logic high to logic low. This causes T_2, which until then conducts, to be cut off, so that the buzzer cannot be actuated. Thus, in case of the temperature being below 3 °C, the buzzer will sound for five seconds after the supply has been switched on. This time period may be lengthened by increasing the value of R_6 and C_2, or shortened by reducing the component values.

Diode D_4 is a protection against polarity reversal of the battery voltage connected to K_2.

Resistor R_8 and capacitor C_3 provide additional smoothing of the supply voltage, which is beneficial to the reliable operation of the unit.

Construction

The unit is best constructed on the printed-circuit board shown in **Fig. 2**. The board is small, yet provides ample space for the components. The terminal blocks are for connecting the sensor and the supply voltage. The IC should be fitted in a suitable socket.

Make sure that the diodes, LEDs and electrolytic capacitors are connected with correct polarity, which is also important in case of the (optional) buzzer. In the latter case, the + terminal is usually indicated on the case.

When the board has been completed, connect it to a supply of 13.8 V (nominal level of a car battery), and check the various voltages indicated in Fig. 1 with a (digital) multimeter. Note that the voltage at junction R_8-C_3 should be about 12.8 V in quiescent operation; it drops to about 11.8 V when the buzzer sounds.

It is important that the potential at pins 2 and 5 of IC_1 is as close as possible to half the supply voltage.

The current drain from the supply is a useful indication as to whether the circuit operates correctly or not. In quiescent operation, it should be about 10 mA and rises to about 50 mA when the buzzer sounds.

The correct operation of the circuit can be checked without R_1. When K_1 is open, the input impedance is infinitely high. The circuit 'sees' this as a low temperature, so that the red LED should light. If the buzzer is used, this should sound when the supply is switched on. If then the terminals of K_1 are short-circuited, the green LED should light.

Calibration and use

Before the unit can be built into a suitable case, P_1 must be set for the desired temperature. To that end, connect a 12 V supply to K_2 and link R_1 to K_1 via a good length of flexible 2-way cable.

Adjust the thermostat in a fridge to rather lower than usual (4–8 °C) and, after an hour so, check that the temperature is about 3 °C. When this is so, put the resistor into the fridge, close the door over the thin cable and let let the sensor acclimatize for about five minutes. Then adjust P_1 until the red LED just lights. This completes the calibration.

The unit can then be built into a suitable case, which is preferably, but not necessarily, transparent as shown in the introductory photograph. A few holes need to be drilled for the supply cable and the cable from the sensor. Mount the unit in an easily seen position in the car with the aid of one or two pieces of double-sided adhesive tape. The supply is taken from a point after the ignition key.

Mount the sensor, preferably on a small piece of aluminium or tinplate, in or behind the front bumper or spoiler, where it is exposed to the out-

Fig. 3. In spite of the compact size of the board, there is ample space for all components.

side air and does not get warmed up by the engine. This position is, of course, also conveniently close to the road surface.

It is best to twist the two wires of the cable linking the sensor to the indicator unit or, even better, to use screened two-core audio cable. The screen should be connected to the negative supply line at K_2.

If the buzzer is too loud for comfort, stick some adhesive tape across the hole at the top.

Parts list

Resistors:
R_1 = 5–20 kΩ with NTC (see text)
R_2, R_3, R_6 = 100 kΩ
R_4, R_5 = 1 kΩ
R_7, R_9 = 10 kΩ
R_8 = 22 Ω
P_1 = 47 kΩ preset

Capacitors:
C_1, C_5 = 10 µF, 63 V, radial
C_2 = 47 µF, 63 V
C_3 = 47 µF, 16 V, radial
C_4 = 100 nF

Semiconductors:
D_1 = LED, green, low current
D_2 = LED, red, low current
D_3 = 1N4148
D_4 = 1N4002
T_1, T_2 = BC547B

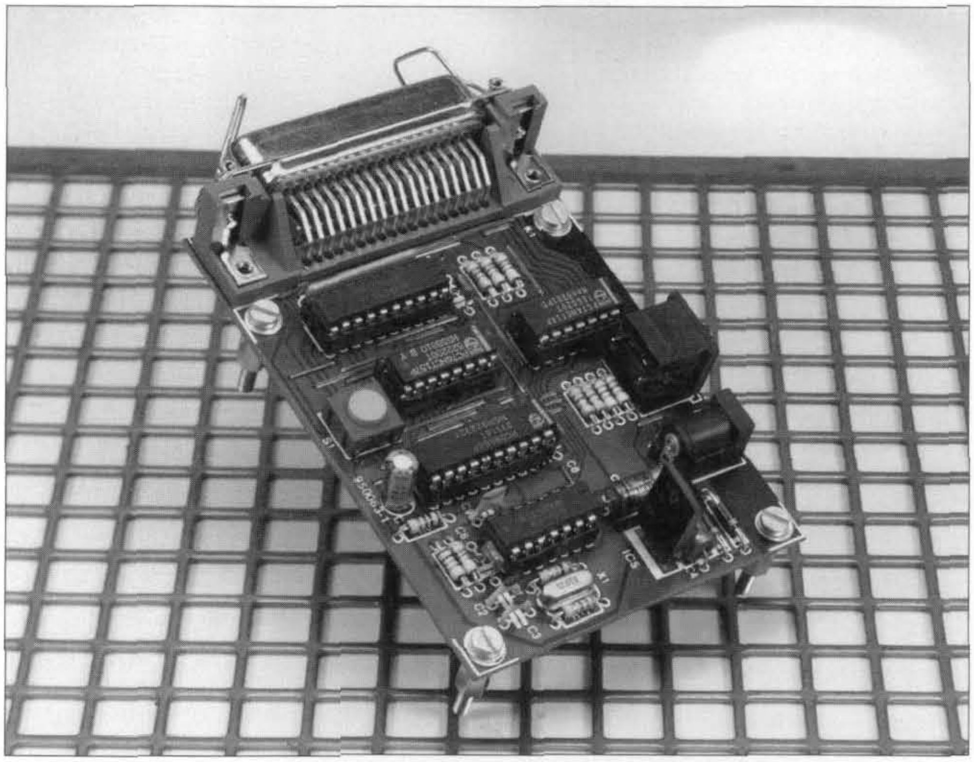
Integrated circuits:
IC_1 = TLC272

Miscellaneous:
K_1, K_2 = 2-way terminal block, pitch 5 mm
Bz_1 = buzzer, 12 V d.c.
Case = to personal requirements
PCB order no. 960029 (see p. 70)

[960029]

I²C INTERFACE FOR CENTRONICS PORT

Many circuits, extensions and modules for the Inter-IC-Communications (I²C) bus have appeared in this magazine over the past few years. This two-wire bus designed by Philips allows ICs as well as complete modules such as TV tuners to communicate with each other. The principle of the bus also allows I²C circuits to communicate with a PC. Thanks to the circuit presented here, it is no longer necessary to insert a special interface card in the PC. The interface function is now taken over by the easily accessible Centronics (parallel printer) port. That allows even portable computers to make use of the I²C bus.



Design by H. Zängerl

I²C represents a communication system which can greatly simplify the design of complex electronic circuits. Why? Because the I²C bus allows data and/or commands to be exchanged between ICs via two or three lines. Considering the increasing use of microcontrollers, the I²C bus offers a great opportunity to reduce the complexity of printed circuit boards, mainly because the ICs need to have fewer pins. An additional boon is that the same ICs become cheaper. It is therefore not surprising to see many I²C ICs and sub-circuits in today's video recorders, CD-i players and colour TVs.

Although there are computer manufacturers who design an I²C interface into their PCs (such as Acorn), normally an extra interface is required to give a PC access to the I²C bus. In an article published in early 1992 (Ref. 1) we described a simple and relatively inexpensive PC insertion card, together with a device driver. That combination allowed any PC or compatible to be used for the control of a host of I²C devices and circuits which were subsequently published. The system allowed higher programming languages such as BASIC and Pascal to be used to access I/O ports, memories, switches, clocks, converters and many more devices, via a three-wire bus.

The large number of reactions on this publication gives us every reason to assume that we hit upon a very popular subject. None the less, the interface card had one distinct disadvantage: it was an insertion card, and could only be used if the computer had a free ISA slot. Is there a simpler way to get on to the I²C bus? Yes, we think so, and you can read about it here.

The interface proposed in this article has the same properties as the insertion card mentioned above. It is also compatible as regards software, so that all existing programs may be used, and it is readily connected to any PC sporting a free Centronics port. Another interesting advantage is the low price, because the circuit board for the present design is much simpler to produce than the insertion card.

The approach

The circuit diagram of the interface is given in **Fig. 1**. The actual translation from parallel data into a serial data stream is performed by IC4, a type PCF8584. This is a pin-compatible, improved, version of the PCD8584 which we used before. The IC receives the information to be translated at its eight data inputs, DB0-DB7. Between these inputs and the PC port sits a type 74HCT245 buffer. To make sure that the information on the databus is always properly defined (as regards logic levels, that is), all eight lines have 4.7-kΩ pull-up resistors.

The software which belongs with this project arranges the information traffic in the following order: if the computer wants to write data to IC_4, the databits first appear on the printer port. Next, a $R/\overline{W}$ signal is generated via the line marked $\overline{STROBE}$. That signal enables IC_2 ($\overline{G}$ input pulled low), so that the data is conveyed, and IC_2 is switched to 'read' mode. A pulse appears on the AUTOFEED line which is converted into the $\overline{CS}$ signal for IC_4 by a network consisting of IC_{1c}, IC_{1d}, IC_{6a}, IC_{6b} and C_9-R_{13}.

The circuit which supplies the $\overline{CE}$ pulse is pretty complex because the

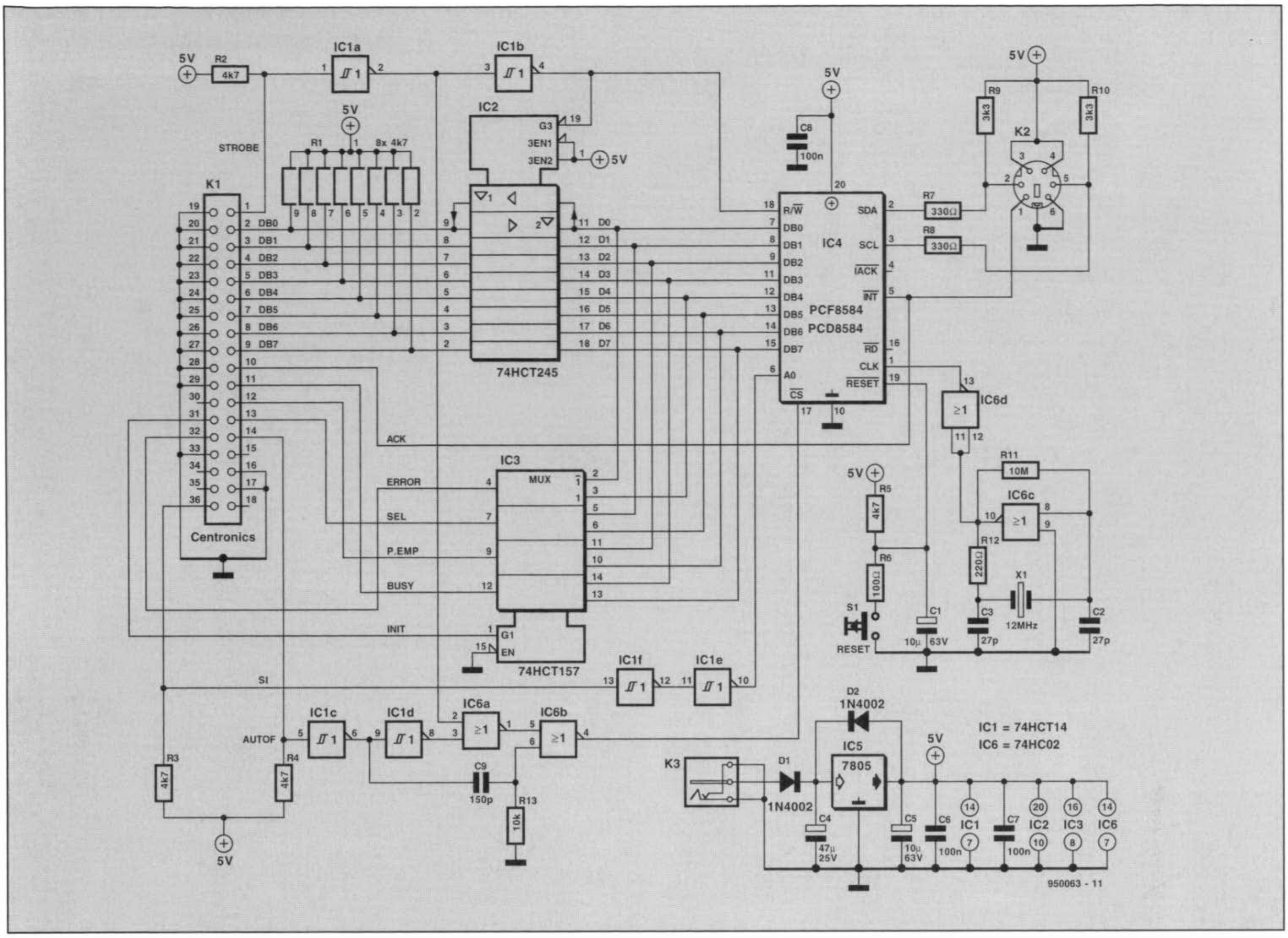


Fig. 1. Circuit diagram of the I²C interface card for connection to the PC's parallel printer port.

R/$\overline{W}$ signal usually arrives before the $\overline{CE}$ signal. The PCF8584 then operates in so-called 68000 mode, when it does not allow the $\overline{CE}$ signal to remain active longer than one I²C clock pulse. When a slow computer is used in combination with a slow Centronics port, the active time of the $\overline{CE}$ signal easily amounts to a couple of microseconds. The information on the I²C bus is then corrupted. Hence, the hardware limits the length of the $\overline{CE}$ signal to about 1 µs with write operations. With read operations, however, the $\overline{CE}$ line must remain active until the computer has copied all data. That is why IC_{1d}, IC_{6a}, IC_{6d} and C_9-R_{13} are used to shorten the $\overline{CE}$ signal at the command of the R/$\overline{W}$ signal while the PC writes, and leaves CE untouched while the PC reads.

Reading in two chunks

Since the printer port does not have a bidirectional structure, it is not possible to copy 8-bit data directly from the I²C interface to the PC. A trick, however, does allow data to be conveyed in this direction at reasonable speed. The levels of the control lines ERROR, SELECT, PAPER EMPTY and BUSY may be read by the PC, and obviously allow four bits to be copied to the PC in one go. Consequently, the full word, consisting of eight databits, may be conveyed in two chunks of four bits. This is done via multiplexer IC_3, on two clock pulses, under the control of the signal on the INIT line. The R/$\overline{W}$ signal (generated via the STROBE line) ensures that the outputs of IC_2 are at high impedance while data is being read.

The SI line of the printer port is used to drive address line A0 of IC_4. Via this line it is possible to gain access to the control registers of the PCF8584. The internal architecture and the functions of the registers in the I²C bus controller are described in the inset. An interrupt signal generated by any of the ICs on the I²C bus is passed directly to IC_4, and appears at pin 10 of the Centronics port (Acknowledge). Because the interrupt facility was never used in any of the I²C projects published so far, the software does not provide an interrupt handler. The hardware does, however, support this option.

MAIN SPECIFICATIONS

Function:	**I²C interface**
Connection:	**via Centronics port**
Software:	**for MS-DOS PCs**
Base address:	**278_H or 378_H**
Device driver:	**Supplied on disk with board**
Output:	**6-pin mini-DIN**
Power supply:	**mains adaptor, current 30-40 mA**
I²C bus clock:	**selectable, 1.5 kHz, 11 kHz, 45 kHz or 90 kHz**

From theory to practice

Well you are down to cutting, bending and soldering at this stage. The copper track layout and component overlay of the single-sided circuit board are given in **Fig. 2**. The advantages of a single-sided board are its low price and the possibility to etch it yourself (the board is however available ready-made

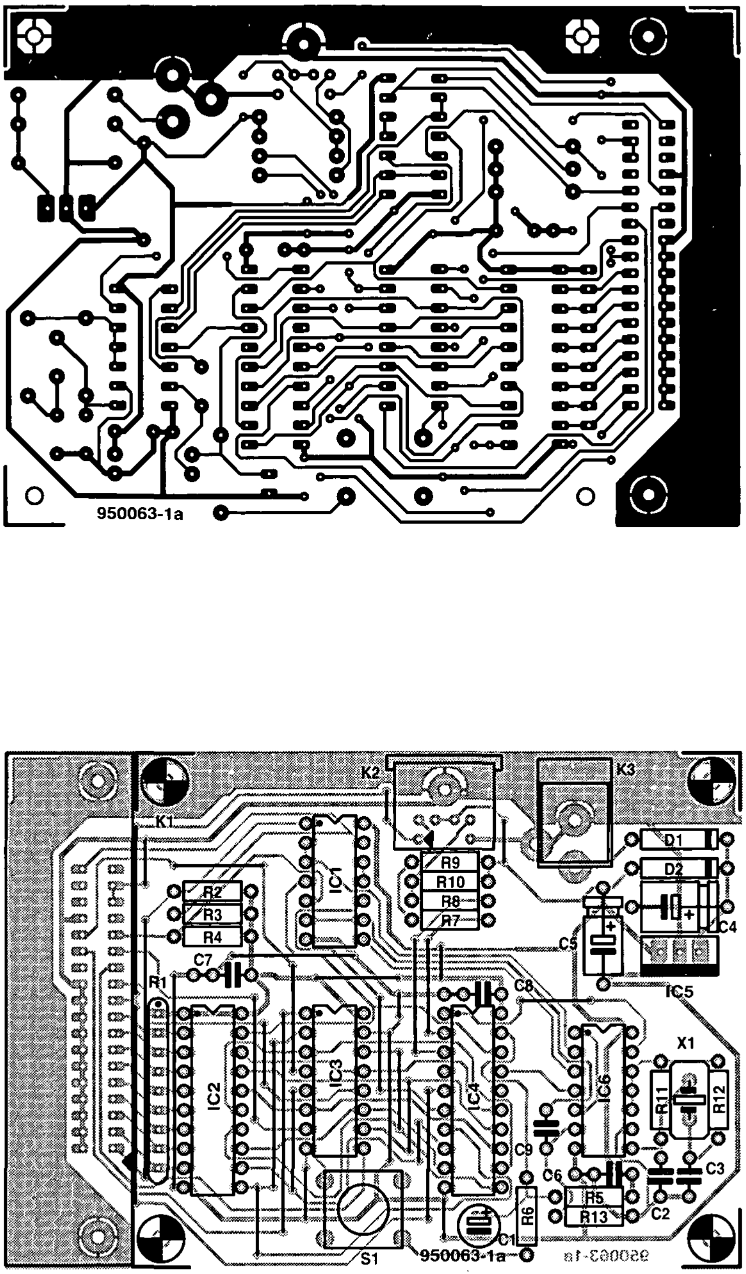


Fig. 2. Copper track layout and component mounting plan of the single-sided circuit board designed for the interface. Start by fitting the thirty wire links!

COMPONENTS LIST

Resistors:
R1 = 8-way SIL-array 4kΩ7
R2-R5 = 4kΩ7
R6 = 100Ω
R7,R8 = 330Ω
R9,R10 = 3kΩ3
R11 = 10MΩ
R12 = 220Ω
R13 = 10kΩ

Capacitors:
C1 = 10µF 63V radial
C2,C3 = 27pF
C4 = 47µF 25V
C5 = 10µF 63V
C6,C7,C8 = 100nF
C9 = 150pF

Semiconductors:
D1,D2 = 1N4002
IC1 = 74HCT14
IC2 = 74HCT245
IC3 = 74HCT157
IC4 = PCD8584 or PCF8584
IC5 = 7805
IC6 = 74HC02 (do not use HCT-version)

Miscellaneous:
K1 = Centronics connector, PCB mount, angled.
K2 = 6-way mini-DIN-socket, PCB mount, angled.
K3 = socket for mains adaptor.
S1 = presskey, make contact, CTL3.
X1 = quartz crystal 12 MHz/30 pF.
PCB and control software on disk, order code 950063-C, see page 70.
Software also available separately, order code 946202-1, see page 70.

Kits and special components for this project available from C-I Electronics, P.O. Box 22089, NL-6360-AB, Nuth, The Netherlands. Fax: (+31) 45 5241877.

through our Readers Services, see page 70). The disadvantage is, well, thirty wire links! Start by fitting these. It may be tedious, but at least you are sure that they are not overlooked later. Next, fit the IC sockets and the passive parts on to board. Mount all parts that make up the power supply, and then check the presence of the supply voltage at the proper points on the IC sockets (refer to the circuit diagram to find these points). The supply voltage is provided by a 9-12 V d.c mains adaptor which must be capable of supplying at least 100 mA.

If you have an oscilloscope, use it to check that the oscillator (IC_{6c}) works. If something is amiss here, use your multimeter in continuity tester mode to check out all connections on the board. For this test you should use the IC pins as measuring points and references, rather than the copper tracks on the board. This will enable you to trace a faulty solder joint quite easily. Do not, however, push the test probes too firmly into the IC socket contacts, because they may lose their spring loading, causing bad IC pin contacts later.

If everything appears to be in order, the ICs may be fitted, and the interface is ready for use. All that is required at this point is the right software and that, fortunately, is available ready-programmed on disk through our Readers Services.

One driver does it all

In principle there are two ways of developing the software which is to work in conjunction with the present circuit. The first alternative is to write a couple of routines which are to be integrated into a larger application program. That approach is simple, easily implemented in a program, and, generally, fast too. The disadvantage is, however, the difficulty you may run into while implementing these routines

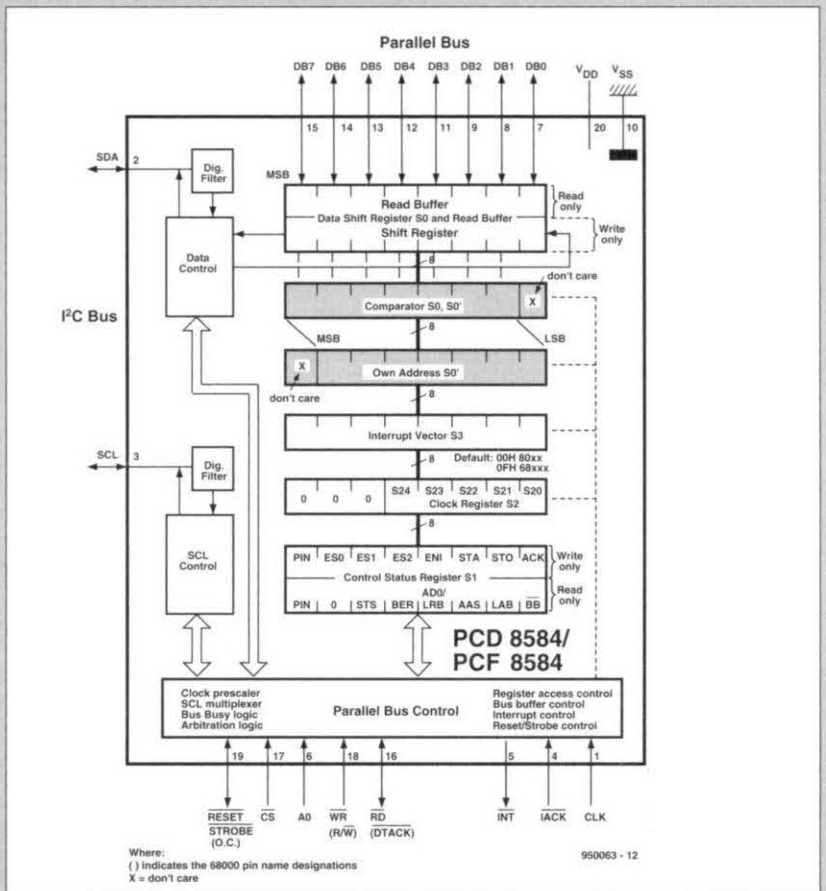
PCF8584 MULTI-PURPOSE I²C BUS CONTROLLER

The PCF8584 is an improved version of the PCD8584, a powerful multi-purpose I²C bus controller which acts as an interface between an 8-bit parallel port on a microcontroller and the serial I²C bus. The improvements over the PCD8584 mainly entail support for the so-called long-distance mode, which, by the way, is not used in the present application. For the rest, these two ICs are fully compatible.
The PCD/PCF8584 supports reading and writing of bytes via the I²C bus, and is remarkably easy to implement in systems based on different types of microcontroller, including the 8048/8051, 80xx processors and Motorola's 6800. The block diagram of the IC indicates a number of functional blocks.

S0, bus buffer. This is the block between the parallel computer bus and the internal shift register. This register is two-fold. The read buffer may only be read, while the shift register may only be written to.

S0', own address. In a multi-master system, this register holds the address to which the IC responds. This address may not be '00' because then the IC goes into a passive monitor mode. That mode is irrelevant in the present application, however, because the PCF8584 is the only master device on the I²C bus. None the less, programmers should ensure that a value (but not 00) is written to the IC at power-up.

S1, control/status register. This register also has a two-fold structure: the status section is read-only, and the control section is write-only. This register is accessible when A0 is logic high. When A0 is low, one of the other registers is addressed. Which one depends on bits ES0, ES1 and ES2 contained in S1:control. The selection is also dependent on the serial interface being switched on or off. This is accomplished by bit ES0: 0 means off, 1 means on. With the serial interface switched off, registers S0', S2 and S3 may be read and written using ES1 and ES2, while S1 may only be written to. With the interface switched on, registers S0, S1 and S3 may be read and written using ES1 and ES2. Note that S0 is the data register. Consequently, databytes can only be conveyed if the serial interface is switched on.



Block diagram of the PCD/PCF8584 from Philips Semiconductors. This IC converts parallel data on a microprocessor bus into the serial format needed for the three-wire I²C bus.

S2, clock register. The clock pulses on the SCL line are derived from the signal at the CLK input. Bits S20 and S21 in register S2 allow the programmer to select one of four clock frequencies: 1.5 kHz, 11 kHz, 45 kHz or 90 kHz. The other bits, S22, S23 and S24 enable the clock input to be matched to one of five possible quartz crystal frequencies: 3 MHz, 4.43 MHz, 6 MHz, 8 MHz or 12 MHz. The 12-MHz option is selected as the default at power-on.

S3, interrupt vector. When the controller is used on interrupt basis, it is capable of putting an interrupt vector on to the PC bus. This address is copied on to the bus as soon as the $\overline{IACK}$ line goes low, and ENI in the S1 register was set beforehand. This option is not used in the circuit proposed here, and not discussed further. In fact, register S3 is of no significance in the present Centronics-based interface.

into higher level programming languages. Also, a small change in the hardware then often means that the software becomes useless.

The most 'transparent' approach is based on the use of a so-called device driver, which gives software access to hardware in a defined manner. The power of that approach is already apparent from the fact that the software originally developed for the insertion card (see above) may also be used in combination with the present Centronics version. The only condition is that you replace the old device driver with the new one.

The diskette supplied for this project contains three directories. The directory I2CP_BP contains programs which have all code needed to communicate, i.e., they do not use a device driver. These programs should be used with a Turbo Pascal 5.5 compiler (or higher). Additional information, where required or useful, may be found in the source code files (*.pas). The directory I2CP_TP6 contains the same information, but for Turbo Pascal 6.0 or higher.

Finally, the directory I²CP_DOS

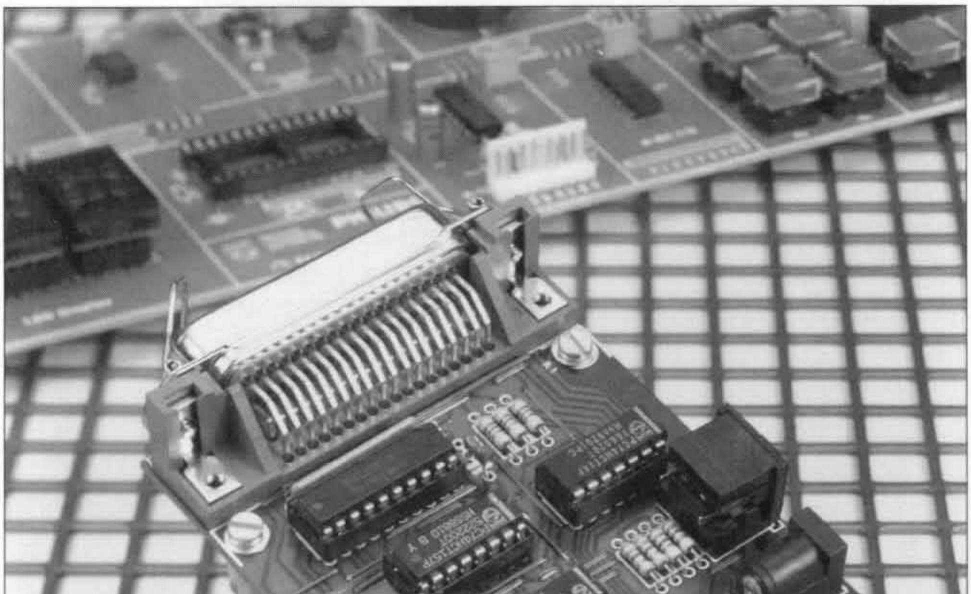


Fig. 3. The completed board is connected directly to any MS-DOS PC. All I²C bus lines are available on a 6-way mini-DIN socket.

contains the previously mentioned device driver. To activate it, include the following statement in your CONFIG.SYS file:

```
DEVICE = I2CDRIVP.SYS b:xxx c:y
```

where xxx indicates the base address of the printer port you wish to use. For LPT1: that will be 378_H, for LPT2:, 278_H. The parameter 'y' is used to set the SCL clock frequency, as follows:

0 = 90 kHz
1 = 45 kHz
2 = 11 kHz
3 = 1.5 kHz

For example, the setting

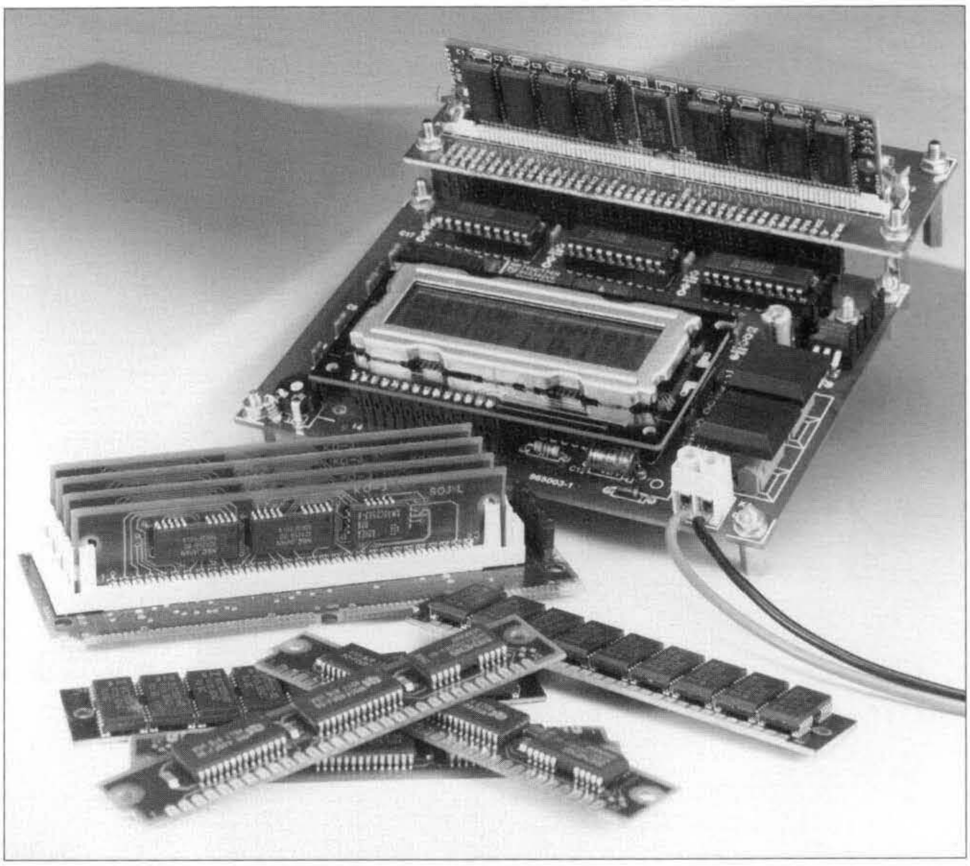
```
DEVICE = I2CDRIVP.SYS b:378 c:0
```

installs a driver which makes use of the interface connected to LPT1, and sets a bus frequency of 90 kHz. The software us ready for use after a restart of the PC.

The driver works flawlessly with all I²C software published so far in *Elektor Electronics*. So, all existing I²C projects may be hooked up to the present interface without any modification.

(950063)

SIMM TESTER



Today's computers invariably require vast amounts of memory, with 16 Mbytes of DRAM generally accepted as a kind of minimum for Windows 95 to run smoothly on, say, a multimedia Pentium machine. Having lots of memory available in your PC is great, but what do you do when it does not work all of a sudden? To remain stuck with a 'memory error' during a boot-up sequence is a frustrating and disheartening experience, and finding the faulty RAM chip is pretty difficult. Up to now, because we present a tester that runs a thorough check on most types of SIMM in use today.

Design by A. Rietjens

MEMORY in a modern PC consists of dynamic RAMs (with the exception of the cache section). These RAMs are fitted on 30-pin or 72-pin plug-in modules called SIMMs. The 30-pin version is going a bit rusty already, and has a digital width of 8 bits (or 9 bits if parity is used). The newer 72-pin PS/2 SIMMs have a width of 32 bits (or 36 bits if parity is used). Depending on its exact type, a PC motherboard may contain one or more SIMMs, usually fitted in special sockets on the board. Processors with a bus width of 32 bits always have a multiple of four SIMMs installed on the motherboard. The mighty Pentium processor with its bus width of 64 bits requires a minimum of two PS/2 type SIMMs. The fact that a computer contains several SIMMs makes troubleshooting the memory section rather difficult.

It also happens frequently these days that upgrading to a faster, larger PC involves migrating the useful bits salvaged from an older computer. In that case, a SIMM tester is a very useful test device for any computer hardware specialist who wants to find out which SIMMs are okay, and which can be scrapped.

It should be noted that the present tester can not be used to measure the famous access time ('speed') of SIMMs under test, because that requires specialized test equipment. The SIMM tester is fine for a general test, however, on many different types of SIMM. By the way, the nominal speed of DRAMs is usually printed on the chips in the form of a type suffix.

Before explaining the operation of the SIMM tester in detail, let's first have a look at the structure of those memory chips that everyone seems to be after these days.

Dynamic RAMs

Dynamic RAMs are marked by their tremendous memory density. Present-day technology allows memory capacities of up to 64 Mbit ($2^{24}\times 4$) to be achieved. Lots of address lines (well, 24) are required to be able to address such a vast amount of information. To keep the number of pins on ICs within limits, address lines are usually multiplexed, for instance, in 2×12-bit format for a 24-bit address width.

The general structure of a dynamic RAM is shown in **Fig. 1**. The memory is addressed via a matrix with an equal number of rows and columns. Consequently, each added address line

QUICK SPECIFICATION

SIMMs that can be tested:

256k	×8, ×9, ×32, ×36
512k	×32, ×36
1M	×8, ×9, ×32, ×36
2M	×32, ×36
4M	×8, ×9, ×32, ×36
8M	×32, ×36
16M	×8, ×9, ×32, ×36
32M	×32, ×36

Can not be tested:
EDO RAMs and 3-V SIMMs

Supported refresh methods:
- RAS-only
- CAS-before-RAS

Other features:
- automatic SIMM type detection
- manual selection of SIMM type
- extensive test data on LCD
- two test algorithms
- simple two-key operation
- fully buffered test connectors
- RAS/CAS read and write cycle

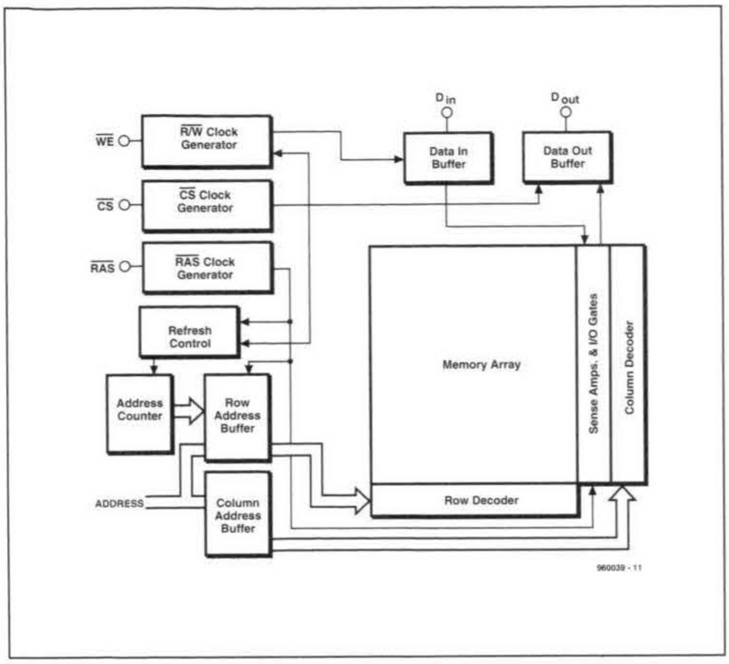


Fig. 1. **Block diagram of a dynamic RAM. The memory block is addressed via rows and columns. A separate section arranges the refresh addressing of the memory cells.**

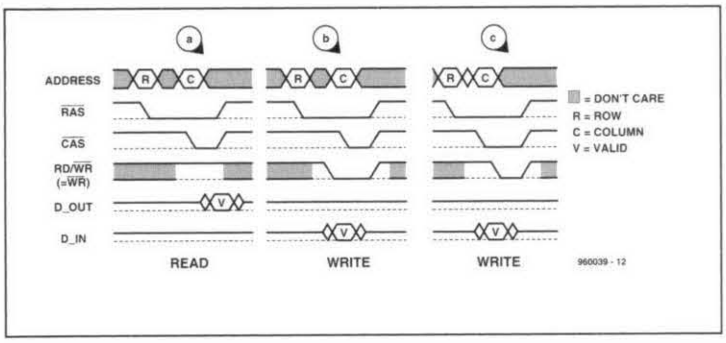


Fig. 2. **This timing diagram elucidates the addressing of a dynamic RAM.**

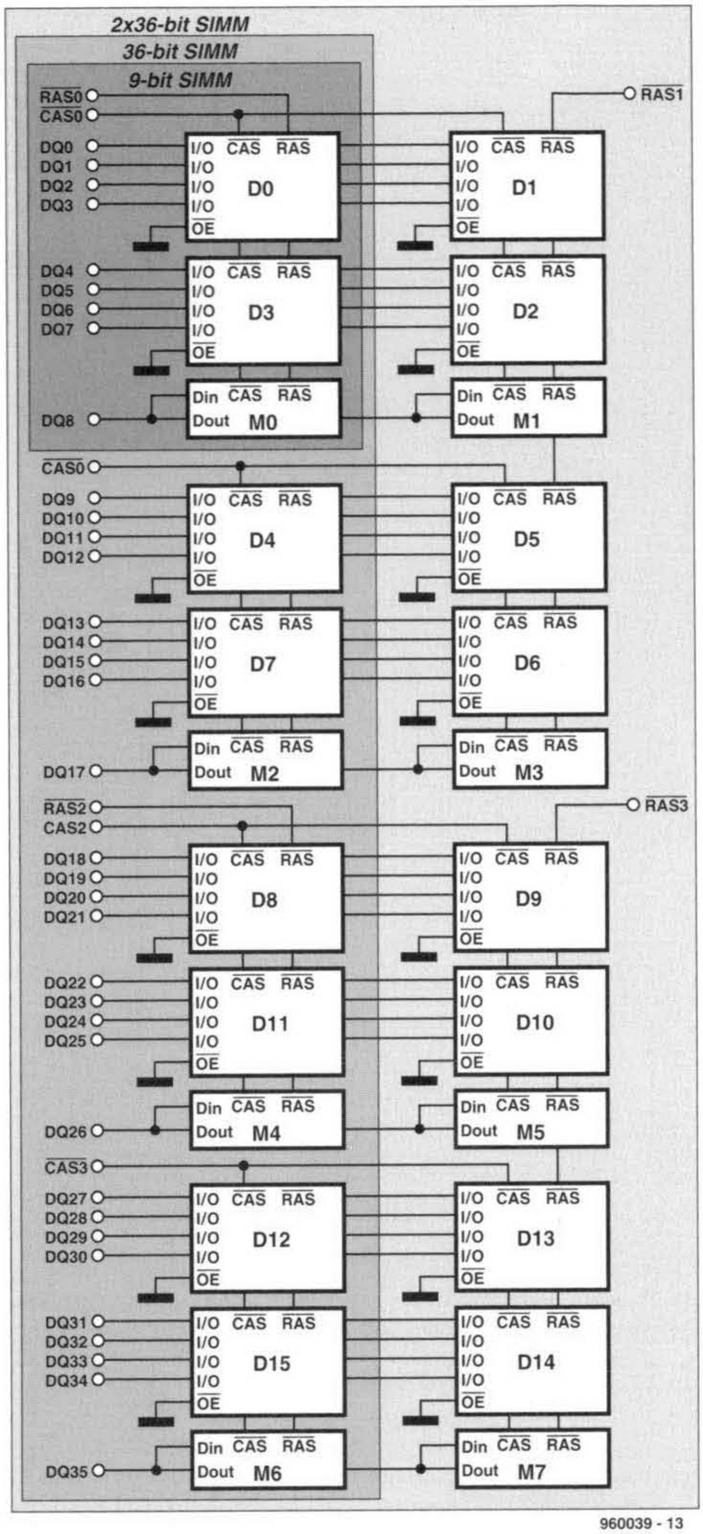


Fig. 3. **Schematic structure of various SIMMs in use today. For clarity's sake, address lines are not shown.**

quadruples the total amount of memory. Next, row and column addresses may be conveyed on the falling edge with the aid of the $\overline{RAS}$ (Row Address Select) and $\overline{CAS}$ (Column Address Select) signals. In combination with a read or write signal, these two active-low signals determine the access to a certain memory location (**Fig. 2**). Data is written on the falling edge of the $\overline{CAS}$ or $\overline{WR}$ signal (the exact moment is determined by the last active signal: $\overline{CAS}$ in Fig. 2, and $\overline{WR}$ in Fig. 3).

SIMMs and PS/2 SIMMs

A SIMM is a small printed circuit board which contains a number of dynamic RAMs. These RAMs together allow an 8-bit or 32-bit wide data bus to be implemented. If the parity bit is used, the width becomes 9 bits or 36 bits respectively. The dynamic RAMs themselves may be types with a width of one or four bits. So, a SIMM of the '1M×9' type may come as a PCB with nine or three chips on it.

On 8-bit (or 9-bit) SIMMs, the $\overline{RAS}$ and $\overline{CAS}$ lines of the DRAMs are connected in parallel, and bonded out as two control lines.

With PS/2-type SIMMs (i.e., SIMMs with 72 pins), parallel busing is done per byte, to which the parity bit is added (if applicable). In this arrangement, each byte has a separate CAS line (CAS0 through CAS3), while RAS is applied for every two bytes (in the order RAS0, RAS2, RAS1 and RAS3). If RAS1 and RAS3 are used, two DRAMs are connected in parallel. This is indicated by DRAMs fitted at both sides of the SIMM board. The schematic structure of such a board is shown in **Fig. 3**. Note that the address lines have been left out for the sake of clarity.

The reliability of today's memory chips is such that there is almost no point anymore in using parity. Hence, some types of PS/2 SIMMs have a parity simulator which replaces the actual parity chip. Unfortunately, the parity function of these PS/2 SIMMs can not be checked with the present SIMM tester. That is because the parity bit is multiplexed with databit D7. So, although these bits may be read individually, they have to be written to at the same time, i.e., with the same data.

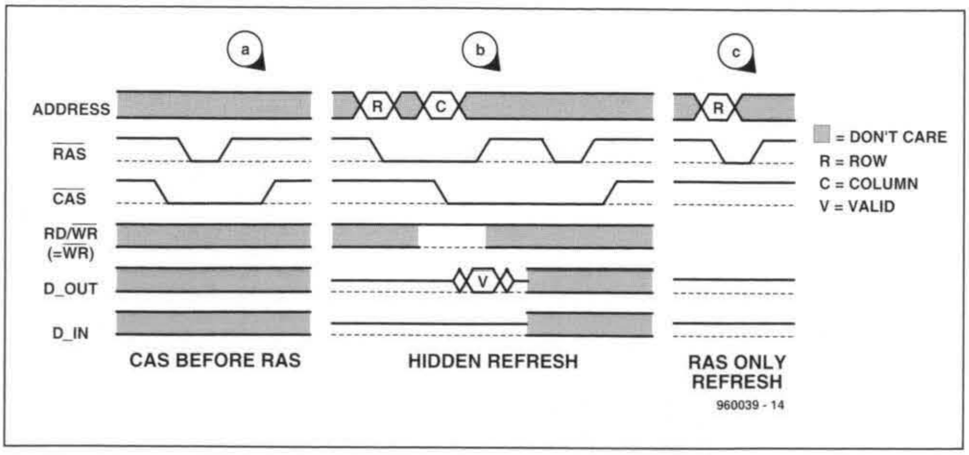


Fig. 4. Currently used DRAM refresh methods.

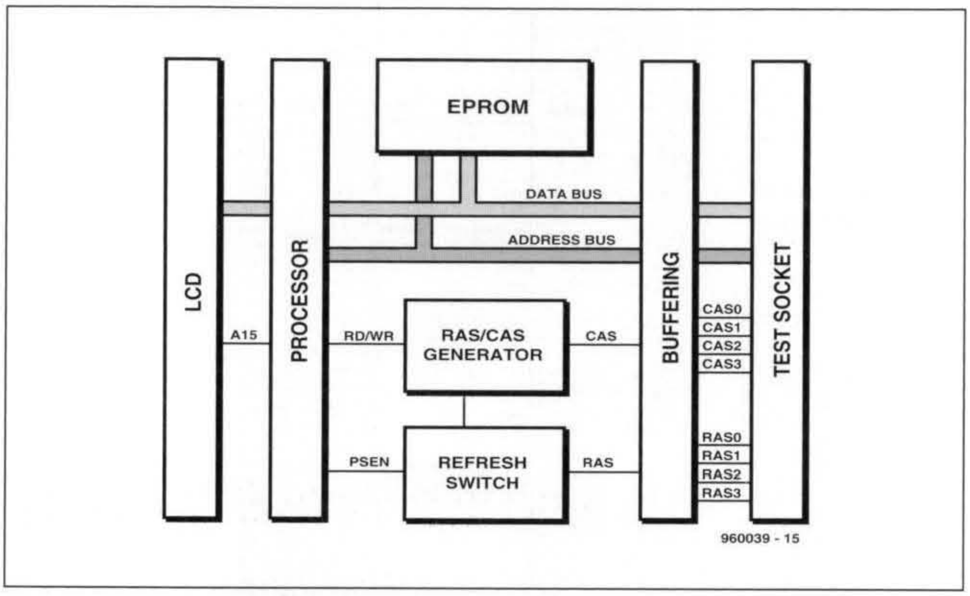


Fig. 5. Block diagram of the SIMM tester. The circuit is capable of checking the operation of each memory cell in a SIMM or PS/2 SIMM.

The refresh story

In contrast with static RAMs which are capable of retaining data by themselves after a write operation, the memory cells of dynamic RAMs have to be refreshed at certain intervals to prevent them losing stored information. Depending on the exact type, DRAMs require a certain minimum number of refresh cycles per unit of time. The refresh rate may vary from a modest 256 cycles per 4 ms right up to 4,096 cycles per 64 ms. Basically, three methods are available to generate a single refresh cycle:

CAS-before-RAS

This type of refresh makes use of a refresh counter built into the RAM. The counter is incremented after each CAS-before-RAS command, and performs the refresh automatically. As illustrated in **Fig. 4**, this command consists of a RAS pulse embedded in a CAS pulse. Without a special controller or additional hardware, such a combined signal is difficult to generate efficiently by software only.

Hidden refresh

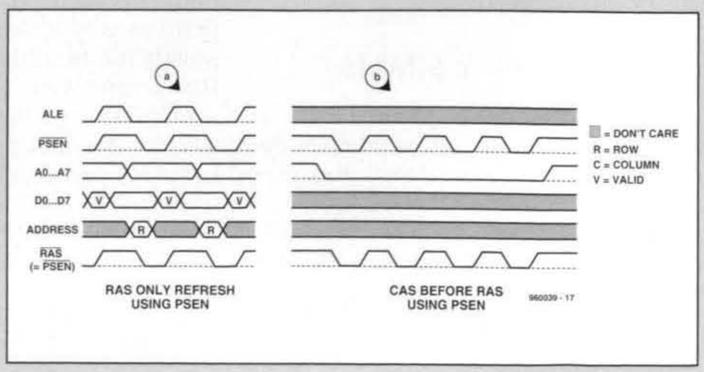
Hidden refresh is like RAS-before-CAS in as far as an extra RAS pulse is concerned which is appended to a normal read or write cycle. Here, too, use is made of the internal refresh counter. This type of refresh is only possible if the RAM is continuously addressed. Hidden refresh is now a thing of the past, and the newer generation of RAMs no longer supports this technique.

REFRESH METHODS USED BY THE SIMM TESTER

The SIMM tester is capable of generating the 'CAS-before-RAS' as well as the 'RAS-only' type of refresh cycle. To make sure that the time spent on DRAM refreshing remains within limits, a dedicated piece of hardware has been added: IC5. This multiplexer passes either the RAS selection for IC4a, or connects all RAS lines to the controller's PSEN signal. This switching depends on the level on address line A13 (P2.5), which is (significantly) labelled 'RFRSH'. Any time instructions are executed from an address range where A13 is at '1', PSEN is linked to all RAS lines. By loading 'NOP' (no operation) instructions in this area, a number of RAS pulses may be fed to the DRAM, along with the corresponding address.

In this way, we ensure that all required addresses arrive within the desired time with 'RAS-only' refresh. Since the system does not know (initially) what type of SIMM is fitted in the test socket, it has to cater for all refresh rates between 256 per 4 ms to 4,096 per 64 ms. That is why the refresh is divided into 16 cycles of 256 addresses. During each cycle, 256 adresses (A0-A7) are refreshed, while the higher-order address section (A8-A11) is incremented during each next cycle. As regards software, each cycle consists of 255 NOPs, followed by an RET (return) instruction to give exactly 256 adresses. These refresh routines therefore occupy exactly 4 Kbytes in the EPROM memory. The refresh is initiated every 3.5 ms (instead of 4 ms) by means of a timer interrupt, leaving a sufficiently large noise margin. Its very simplicity also makes the refresh relatively fast.

Address lines A12 and A2 are used to build a CAS-before-RAS cycle. Here, NOPs are not necessary, and a single RET is sufficient. The CAS line is pulled low when A12 and A2 are logic high. Next, the execution of the RET instruction generates four PSEN pulses, which are sufficient for an equal number of refresh cycles. Considering the software overhead for this method, more time is needed to realize 256 cycles. With this refresh method, a test will take about 30% longer.



RAS-only refresh

During a RAS pulse, a row address is also indicated as a refresh address, and copied into the refresh memory on the falling edge of the RAS pulse (Fig. 4c).

No complex hardware

The block diagram shown in **Fig. 5** proves that the general structure of a powerful SIMM tester need not be too complex. The processor and the EPROM are part and parcel of a standard 8032 configuration which also controls the LCD. With the aid of some additional electronics (mainly buffers) we are able to run a thorough check on SIMMs.

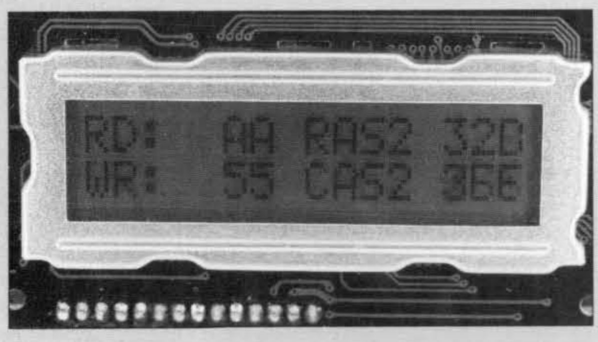
The practical realization of the circuit is shown in **Fig. 6**. The processor section of the circuit should be familiar by now, consisting of little more

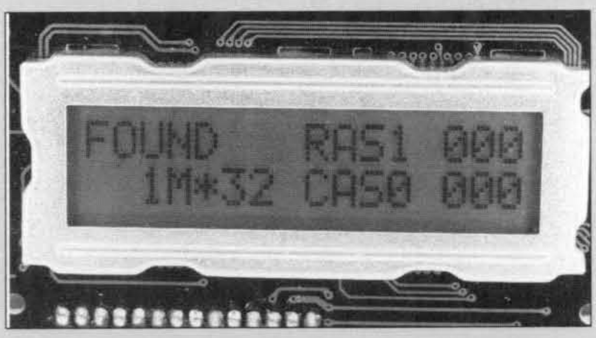
GENERATING THE RAS AND CAS SIGNALS

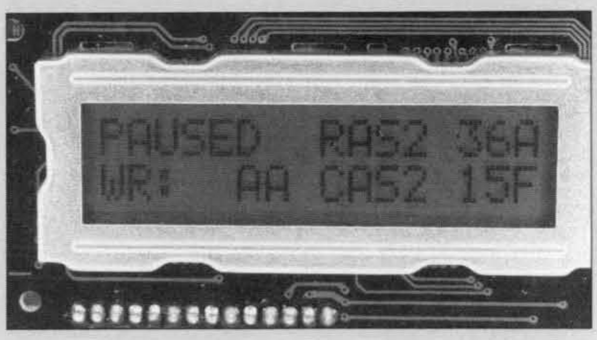
Before any read or write instruction to the SIMM, a row address ($\overline{RAS}$) and a column address ($\overline{CAS}$) needs to be written (in that order). The $\overline{RAS}$ signal is started by reading an external address (IC7a high) of which address line A15 is low (IC7b low). Inverter IC8d ensures that the $\overline{RAS}$ signal is actuated on the falling edge of the $\overline{RD}$ or $\overline{WR}$ signal (if input D, pin 2 of IC6a, is high), so that the lower address section (A0-A11) is conveyed to the SIMM as the row address. At the same time, switches IC9c and IC9d are opened, so that the $\overline{CAS}$ line remains inactive.

Next, switch IC9c is opened, so that a $\overline{CAS}$ pulse is generated with the next read or write instruction. This $\overline{CAS}$ pulse supplies the column address, and reads or writes to/from the selected memory location. When pin 12 of IC6b is logic high, the $\overline{RAS}$ line is also de-activated at the end of the $\overline{CAS}$ pulse. The assembly code listing below gives an example of a 'read' command.

```
WR TO RAM: MOV   WRITEDATA,A ;SAVE DATA FOR DISPLAY
           CLR   EA          ;DISABLE ALL INTERRUPTS, I.E. NO REFRESH ALLOWED
           SETB  D7BIT       ;WRITE D7 INTO D7 AND PARITY BIT
           SETB  PARITYBIT
           CLR   CASBIT      ;DISABLE CAS PULSING
           MOV   DPH,ROW_H   ;SET ROW ADDRESS
           MOV   DPL,ROW_L
           MOVX  A,@DPTR     ;DUMMY READ TO OUTPUT ROW ADDRESS
           SETB  CASBIT      ;ALLOW CAS PULSE FOR NEXT EXTERNAL ADDRESS
           MOV   DPH,COL_H   ;SET COLUMN ADDRESS
           MOV   DPL,COL_L
           MOV   A,WRITEDATA ;LOAD A WITH DATA TO WRITE INTO RAM
           MOVX  @DPTR,A     ;WRITE CONTENTS OF A TO RAM
           CLR   D7BIT
           CLR   PARITYBIT
           CLR   CASBIT      ;DISABLE CAS PULSING
           SETB  EA
           RET
```







than the components already indicated in the block diagram, 'glued together' by wiring, of course. The $\overline{RAS}$ pulses are generated with the aid of IC_{7a}, IC_{7b}, IC_{8d}, IC_{8e} and IC_{6a}. IC_{9c}, IC_{9d} and IC_6 generate the $\overline{CAS}$ pulses. Two NAND gates and an inverter (IC_{7a}, IC_{7d} and IC_{8a}) ensure that the LC display may be used in the external address range (A15='1', via port P2.7). Address lines A0 and A1 then determine the selection of registers in the LCD controller, and also select between reading and writing.

Because PS/2 SIMMs have shared $\overline{RAS}$ and $\overline{CAS}$ inputs, IC_{4a} and IC_{4b} are needed to perform the actual addressing of individual $\overline{RAS}$ and $\overline{CAS}$ lines. IC_5 enables all $\overline{RAS}$ lines to be selected in one go during a refresh. That simplifies the software implementation of the 'RAS-only' refresh system considerably.

IC_{11}, IC_{12} and IC_{13} buffer all input and output lines on the SIMM test sockets (K2 for 30-pin SIMMs, and K3 for 72-pin SIMMs). The buffering prevents the test system from crashing when a faulty SIMM is under test. Because the 8032 has a databus width of only 8 bits, bit 7 is multiplexed with the parity bit (IC_{9a} and IC_{9b}).

Finally, push-buttons S1 and S2 ensure easy operation of the tester.

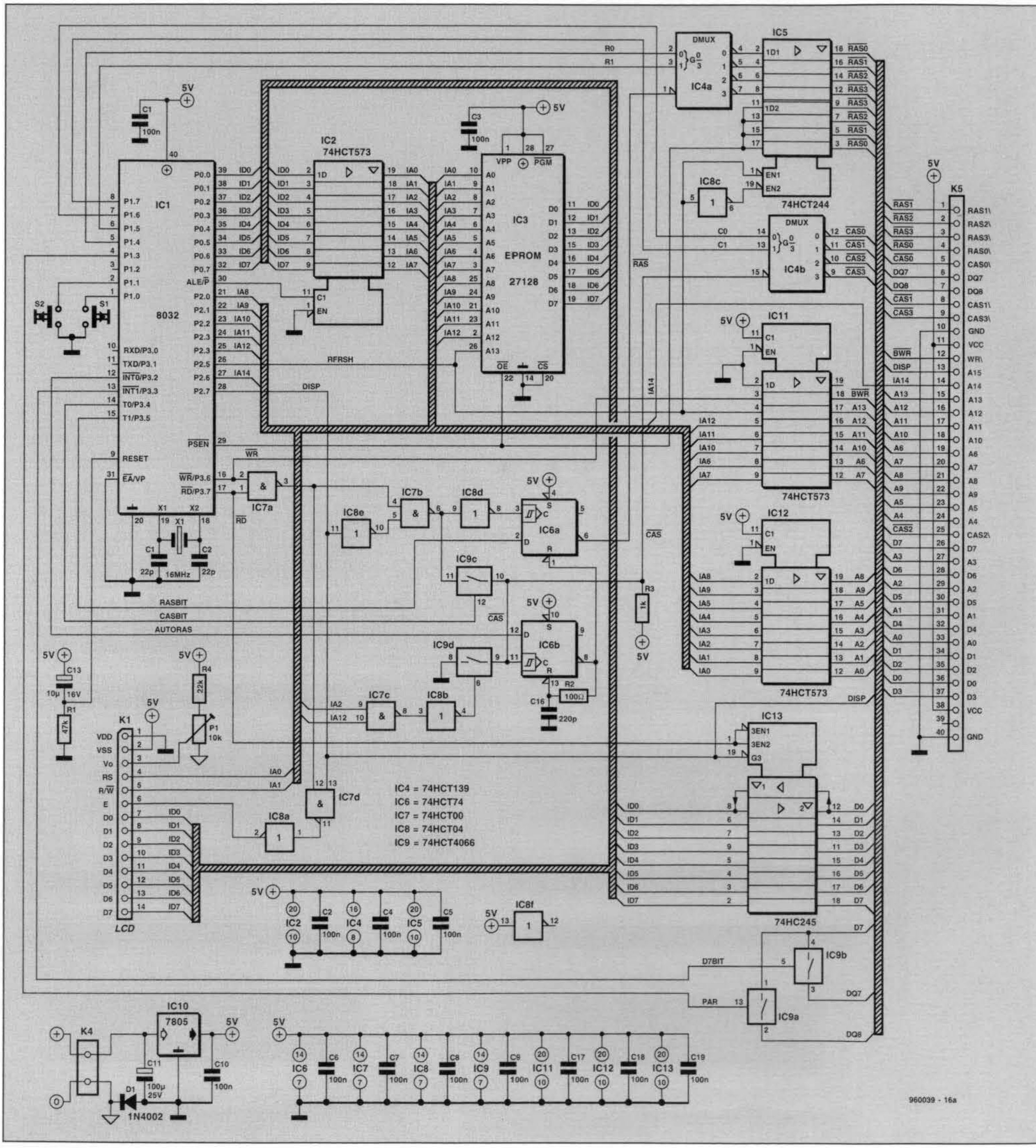


Fig. 6. From theory to practice. The structure of the hardware shown here bears great similarity with that of the block diagram.

The SIMM tester is accommodated on two printed circuit boards. That explains the presence of connectors K5, K6 and K7 which convey all relevant signals between the two boards.

The tester is powered by a type 7805 5-V regulator which provides a stable supply voltage. Diode D1 serves as a polarity reversal protection. It also provides a negative reference voltage for contrast adjustment of the LCD.

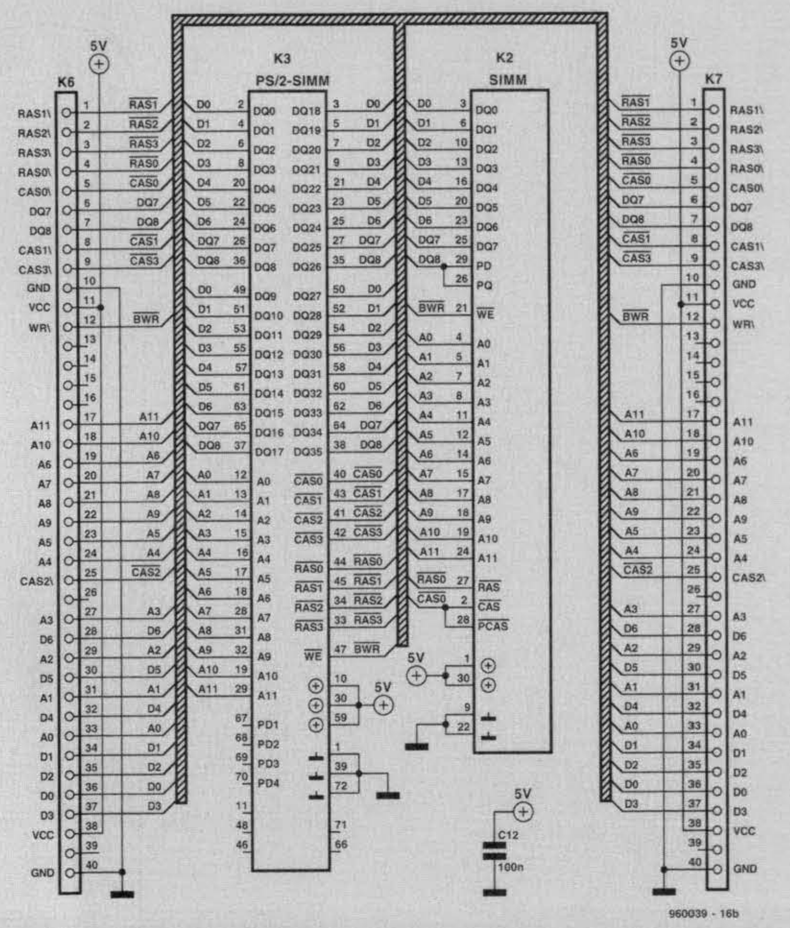
Construction and test

The printed circuit board for this project is double-sided and pretty compact (see **Fig. 7**). The board is available ready-made through our Readers Services (see page 70). Anyone capable of soldering accurately should be able to produce a working copy of the SIMM tester. It is cut in two sections so that the section with the SIMM sockets may be mounted above the main board. Cutting is easy with the aid of the fraised groove in the ready-made board.

Start by building up the main board, taking care to observe the mounting direction of the ICs. Note the different orientation of IC_1 and IC_2 on the board! Depending on the enclosure used, S_1 and S_2 are either PCB-mount or panel-mount types.

Once all parts are fitted on the board, connect only the LCD. Do not insert the ICs in their sockets yet, with the obvious exception of IC_{10}. Apply power (from a 12 V/150 mA mains adaptor), and check that one text line of the LCD lights up. If you see nothing, adjust P1. Switch off, fit IC_1, IC_2, IC_3, IC_7 and IC_8, and switch on again. The welcome screen should appear on the LCD, and it should be possible to select all functions by pressing the two push-buttons. Switch off again, and fit IC_4, IC_5, IC_6 and IC_7. The circuit should start again without problems. If everything is all right so far, buffers IC_{11}, IC_{12} and IC_{13} may be fitted.

The main board and the socket board are connected via a header (K_5) and mating sockets (K_6 or K_7) which come in strips of 36 pins. The remaining pieces are used to attach the LCD (K_1) in a similar way. The plug-in constructions allow the socket board and the LCD module to be taken off easily. If you have just a little more space available, you may mount the socket on K_6 as illustrated in **Fig. 8**. If K_7 is used, the unit may be made even more compact. If the connector is secured to the front panel of the case, it may be handy to mount capacitor C_{12} at the solder side of the board.



The SIMM socket board is shown as a separate circuit here.

About the software

The software burned into the EPROM allows (PS/2) SIMMs with capacities between 256k×8 up to 32M×36 to be tested. The tester is operated via push-buttons S_1 and S_2. Depending on the progress of the test procedure, push-button S_2 acts as a START, PAUSE or STOP control. S1, on the other hand, is used as the SELECT or CONTINUE control. Before testing a SIMM, the circuit should be switched off. Next, carefully insert and secure the SIMM in the relevant socket. Switch on again. Go to the main menu of the start program by pressing one of the keys after the start message has appeared. Four options are offered: AUTO, MANUAL, RAS ONLY and FAST. The function which is selected after S_2 is pressed flashes on the display. You step through the menu by pressing S_1. During all tests, the display indicates the currently tested address. That address is determined by address lines A0 through A11 (max.) (i.e., $0FFF_H$ with 16-MB SIMMs), and

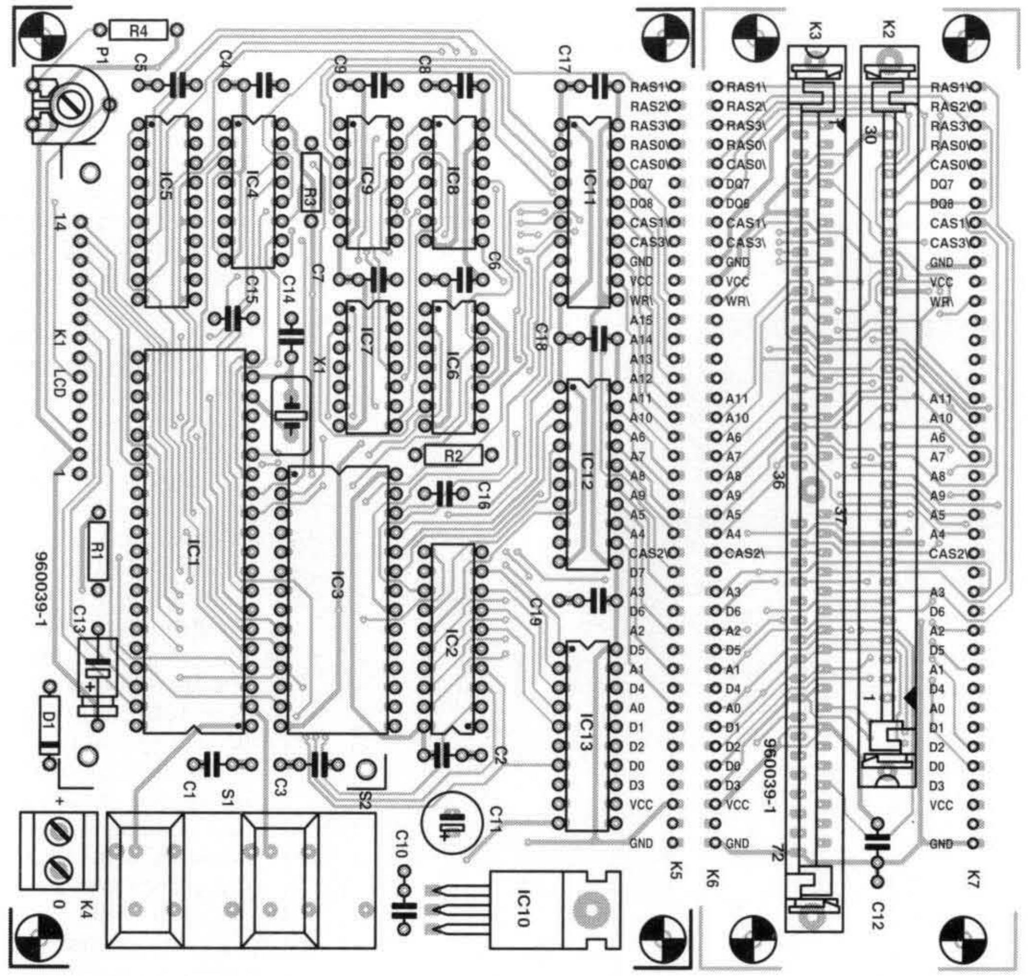


Fig. 7. Track layouts and component mounting plan of the double-sided through-plated board (available ready-made through the Readers Services).

the $\overline{RAS}$ and $\overline{CAS}$ lines. The display also indicates the data being written and read back. As soon as errors are encountered, the display shows a message telling you which memory location is faulty. Below is a brief summary of the function of each of the menu options offered by the SIMM tester.

AUTO

When AUTO is selected, the software automatically detects the type of SIMM fitted in the test socket. The detection process is based on a shortened test algorithm. The result appears on the display. Next, the actual test may be started by pressing S_2.

MANUAL

After choosing this option, S_1 may be used to select the type of SIMM you wish to test. This option is useful if the 'auto' function fails to establish the right type. If that is the case, by the way, it is almost certain that the relevant SIMM has a serious fault. Start

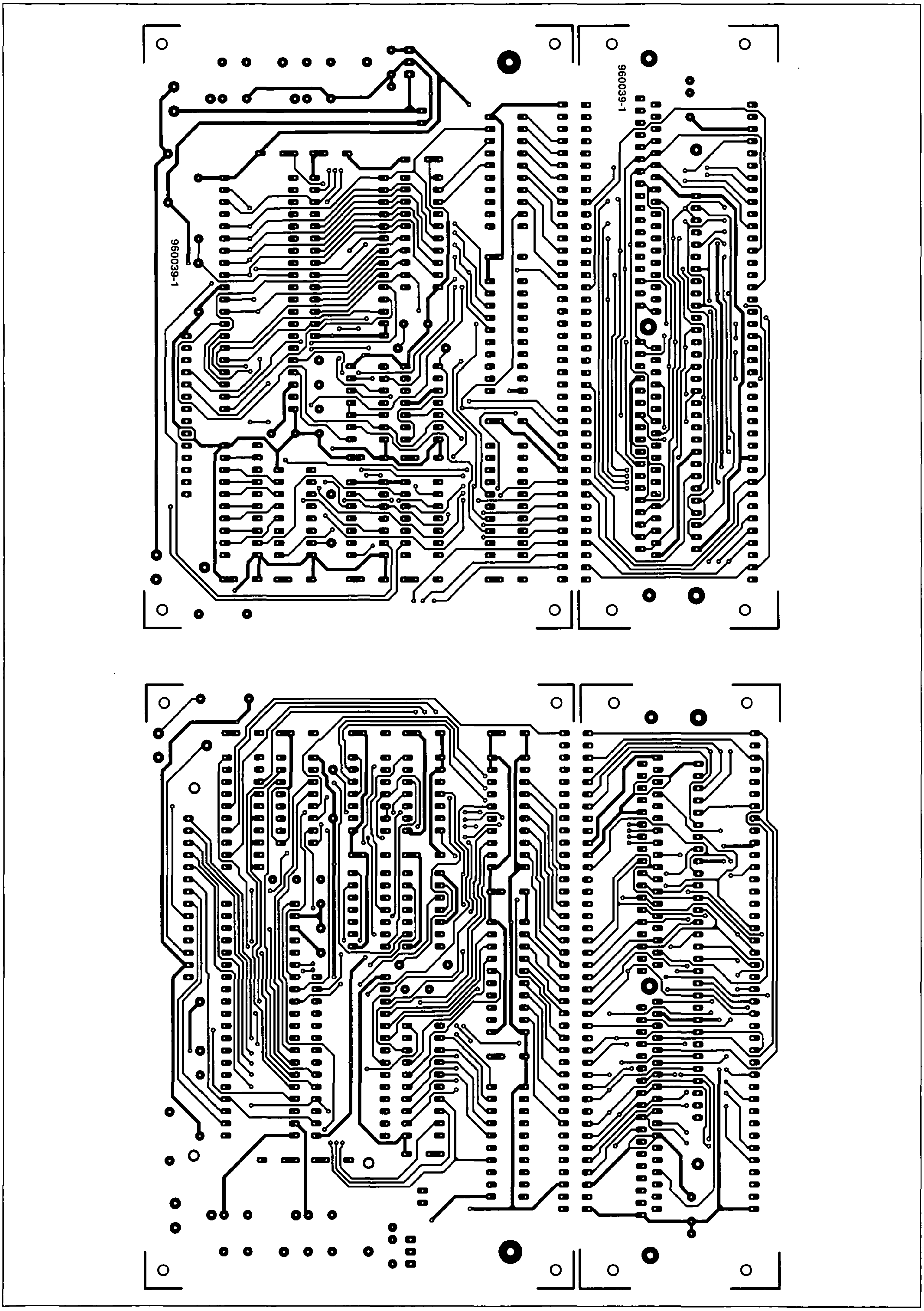
COMPONENTS LIST

Resistors:
R1 = 47kΩ
R2 = 100Ω
R3 = 1kΩ
R4 = 22kΩ
P1 = 10kΩ preset

Capacitors:
C1-C10;C12;C17;C18;C19 = 100nF
C11 = 100μF 25V radial
C13 = 10μF 16V
C14,C15 = 22pF
C16 = 220pF

Semiconductors:
D1 = 1N4002
IC1 = 8032 (16MHz, 40-pin DIL)
IC2;IC11;IC12 = 74HCT573
IC3 = 27128 (order code 966503-I)
IC4 = 74HCT139
IC5 = 74HCT244
IC6 = 74HCT74
IC7 = 74HCT00
IC8 = 74HCT04
IC9 = 74HCT4066
IC10 = 7805
IC13 = 74HCT245

Miscellaneous:
K1 = 16-character LC display, e.g. sharp LM16A21.
K2 = 30-pin SIMM socket or ZIF socket.
K3 = 72-pin PS/2 SIMM socket or ZIF socket.
K4 = 2-way PCB terminal block, pitch 5mm.
K5 = 40-pin single-row pinheader.
K6 or K7 = 40-pin single-row socket (see text).
S1,S2 = Dataswitch presskey (with wide cap).
X1 = 16MHz quartz crystal, low-profile.
14-pin header.
14-pin socket.
PCB and control software in EPROM: set order code 960039-C (see page 70).
Programmed EPROM also available separately, order code 966503-1 (see page 70).


960039-1
960039-1

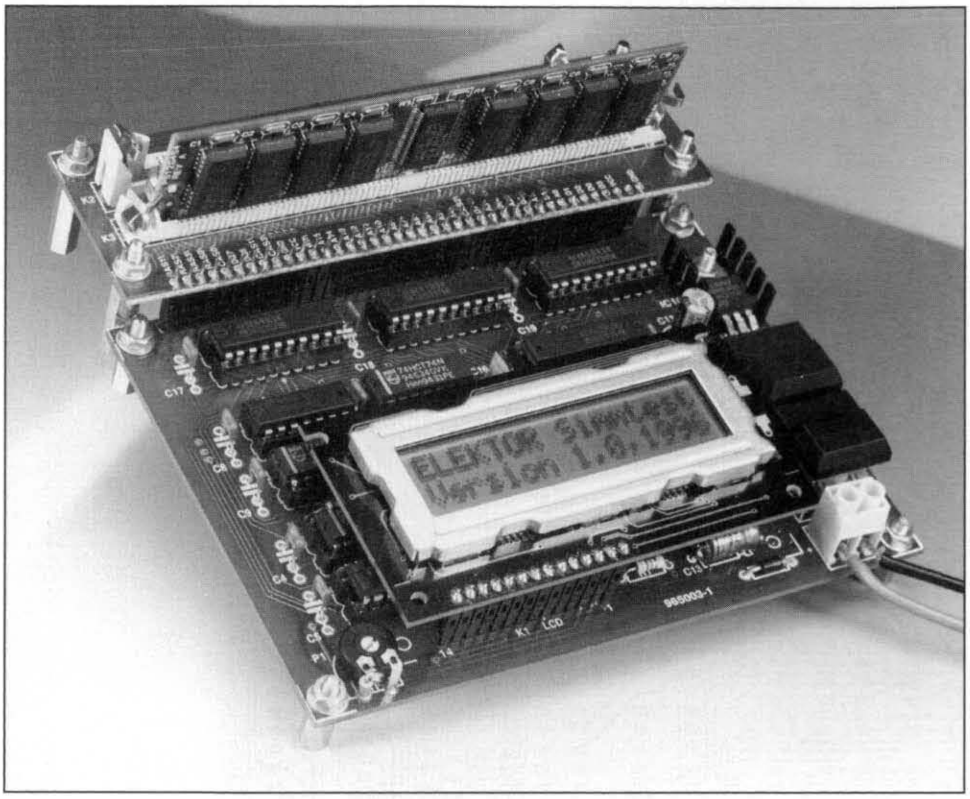


Fig. 8. The SIMM socket board may be attached to the main board in two ways. Here, it is fitted as far as possible to the outside.

the test by pressing S_2.

RAS ONLY versus CAS<RAS

This menu option allows you to choose between two refresh methods: 'RAS-only' and 'CAS-before-RAS'. Remember that the latter method takes about six times longer than 'RAS-only'. Consequently, the whole test will take much longer.

Fast versus Marching

This menu option enables you to select the actual test method. The 'fast' method runs three times through the entire memory. During the first cycle, all memory locations are filled with the value '$0AA_H$'. During the next cycle (which counts backwards), the contents of each location is compared with '$0AA_H$', and subsequently filled with its complement, '055_H'. Once the test has arrived at address 0, the third part of the test follows: testing for '055_H', and then filling with '$0AA_H$' again. In this way, all memory location are tested for their ability to accept and retain ones and zeroes. Mirrored areas, caused by address line errors on the SIMM, will be spotted because the data belonging with the addressed has been changed already. That, incidentally, is the reason for reading and writing test data in two directions.

The 'marching' test method (the name is used by Hitachi and other DRAM makers) goes one step further: the entire memory is first cleared by filling it with zeroes. Next, it is filled with ones, bit-by-bit, where the system checks beforehand for the presence of the zero in each location.

The operation of the two test methods is illustrated in the flowcharts shown in **Fig. 9**.

Testing, ready while-U-wait

Having built and tested the circuit as described above, you are ready to run a thorough check on your first SIMM. Make sure the tester is off, and insert the SIMM into the relevant test socket. Switch on. If you use the AUTO option, the SIMM identification and testing procedures are run automatically. The progress of the test procedure may be followed on the LCD screen. If the message 'RAM OK' appears, it is certain that the SIMM is fully functional. As already mentioned, there is a snag: the SIMM tester 'just' checks all memory locations in the SIMM, it **does not** check the access time claimed by the manufacturer. In most cases, however, the access time printed on the device (for instance, '-6' for 60 ns) may be taken in good faith.

If a SIMM gives errors in a PC, but checks out okay in the tester, you may need to add wait states. That is achieved fairly easily by reviewing the PC's BIOS settings, which can be opened (usually) by pressing the DEL key after a cold start.

(960036)

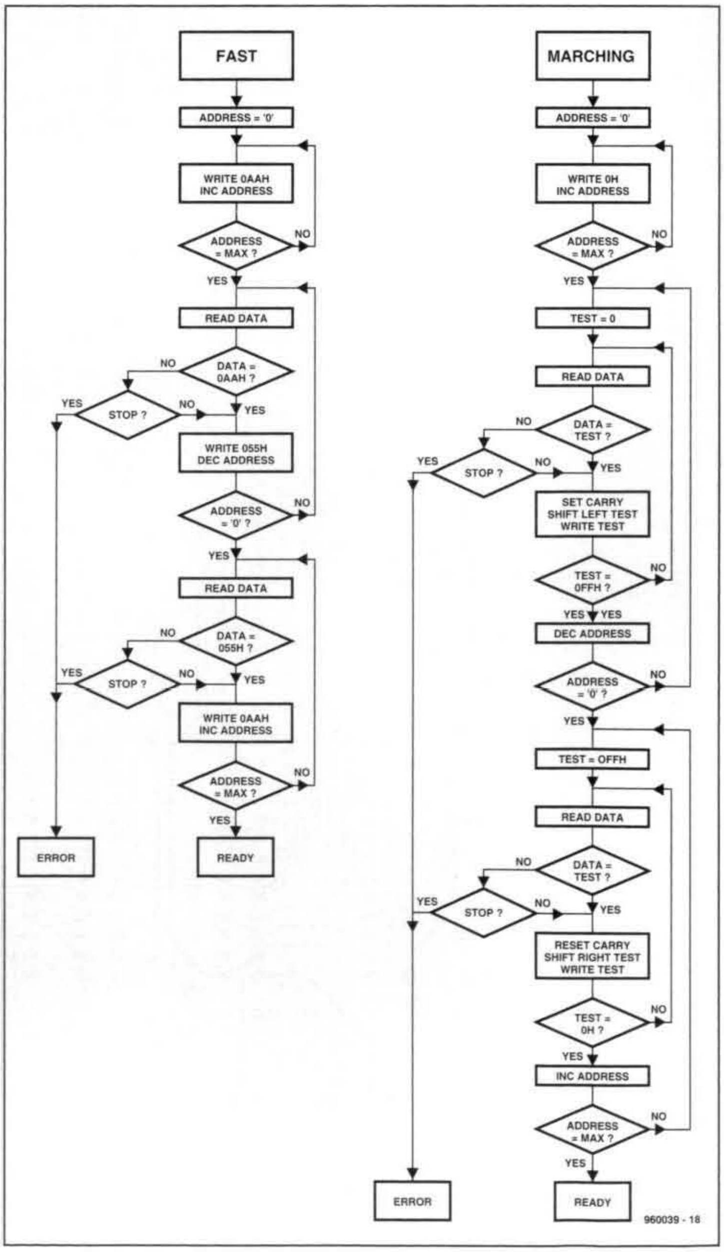


Fig. 9. Flow diagrams of two advanced DRAM test methods, 'fast' and 'marching', which are used by the circuit.

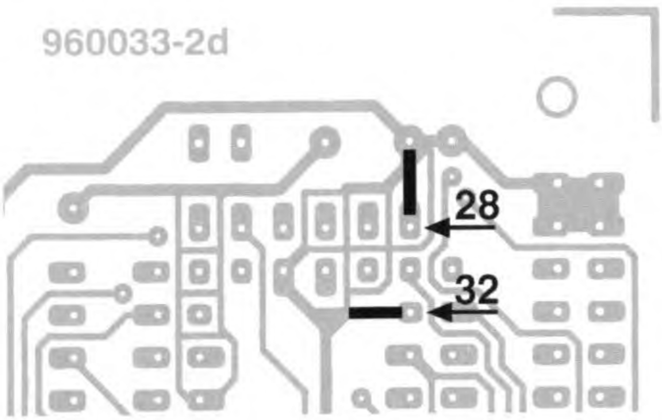
CORRECTIONS & UPDATES

64-channel logic analyser
(May 1996, p. 35-43, 960033)

Constructors using the ready-made printed circuit boards for this project should note that capacitors C25, C35 and C45 were not included in the circuit diagrams, PCB layouts and parts lists as printed in the magazine. These capacitors afford additional supply decoupling, and should have a value of 10µF, 16V.

Channels 48 through 64 (probe D) are not available because IC40 can not be selected. The problem is simple to solve by connecting pin 28 ($\overline{CS0}$) and 32 ($\overline{CS1}$) of IC40 to ground (see drawing). The circuit diagram on page 38 should be corrected likewise.

Finally, on the main board, copper tracks run very close to the board mounting hole near pins 49/50 of connector K4. Care should be taken not to cause short-circuits here by PCB spacers or screws.



Matchbox BASIC computer as data logger
(September 1996, p. 18-21, 960065)

Owing to a conversion mistake in the electronic page layout process, all underscore characters (_) have disappeared from the listing on page 19. Readers wishing to obtain a free copy of the corrected listing (on paper) may apply to our Customer Services department in Dorchester.

SIMM tester
(February 1996, p. 18-26, 960039)

If fast SIMMs are tested, a bus conflict may arise, causing a latch-up situation and an incorrect message stating that the SIMM is faulty. This may happen because buffer IC13 uses the $\overline{RD}$ signal to reverse its direction, while the SIMM does so using the $\overline{WR}$ signal.
This problem may be solved as follows:

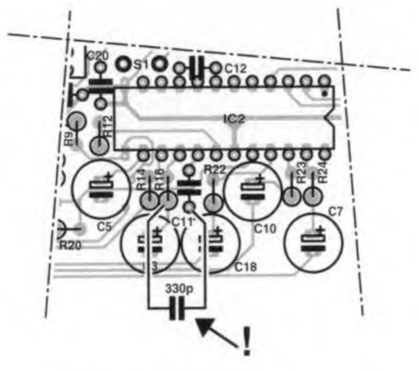
a. Disconnect pin 1 of IC13, and connect it to pin 12 of IC8 (a non-used inverter).
b. Disconnect pin 13 of IC8, and connect it to pin 2 of IC7 (WR).
c. Disconnect pin 19 of IC13, and connect it to pin 2 of IC8.

Keyboard swap for PCs
(June 1996, p. 40-43, 950126)

Because resistors R5 and R6 may form a too large load for IC1, the PC may not receive anything although the LEDs indicate that one of the keyboards is active. This problem may be solved by increasing the value of R5 and R6 to 1kΩ. If the LED intensity is reduced too much, high-efficiency LEDs should be used.

Video test chart generator
(October 1996, p. 24-29, 960076)

The S-VHS output may oscillate. This problem may be solved by fitting a 330-pF ceramic capacitor between junction R16/C3 and the ground connection of C11 (at the underside of the board, see drawing).



The value of capacitor C14 has to be increased from 100nF to 470nF.

The modulator case has to be soldered to the ground plane of the PCB. This may be achieved by fitting solder pins near the corners of the modulator (drill additional holes), or by removing the protective lacquer in these locations, and solder the modulator case directly to the ground plane.

U2402B battery charger
(April 1996, p. 10-15, 950120)

In the circuit diagram on page 12, the switch identified as S2a (near R22) should be S2c.

Oscilloscope prescaler
(November 1995, p. 28-34, 950115)

A number of readers have reported timing problems with the RAMs used in the circuit. For these RAMs, a short period appears to be necessary between the 'address stable' and 'write enable low' instants.

Two solutions are available:

1. Use the type GM76C28A-10 from Goldstar in position IC13. This RAM chip was also used in our prototype.
2. Modify the PCB as follows:
   - desolder the socket for IC9;
   - cut the connection between pins 1 and 2 of IC9 at the component side of the board;
   - fit a new IC socket;
   - connect a short isolated wire between pin 2 of IC9, and pin 10 of IC6.

The latter solution causes a delay of 238 ns on the WE line, enabling the circuit to work with RAMs having a specification other than $t_{as}=0$ also.

FM RECEIVER IN SMT

Present-day integrated components allow a complete FM stereo receiver to be built from just a few ICs. The next step is to use SMT (surface mount technology) ICs, and you have your matchbox-size personal stereo.

Design by L. Lemmens

THE present radio is based on the TDA7088T SMT IC from Philips Semiconductors. This is the successor of the famous TDA7000 which some of you may remember from earlier publications in this magazine. The TDA7088T now features stereo reception!

The block diagram of the TDA7088T is shown in **Fig. 1**. All the functions of a classic superheterodyne receiver are easily found: mixer, oscillator, IF amplifier and demodulator. In addition to these standard functions, the chip offers a number of extra features.

Pins 11 and 12 form the balanced inputs of the mixer. With the aid of the oscillator (VCO), the received signal is mixed down to an intermediate frequency of 73 kHz. The mixer's output signal is applied to an active filter consisting of a 3-pole low-pass and a single high-pass. Next, the filtered 73-kHz signal is given a boost by the IF amplifier/limiter before it is demodulated by a quadrature demodulator. The phase shift required for the demodulation process is supplied by an all-pass network. So far, everything looks pretty normal, apart from the electronic filters, of course. Now for the 'special' bits. As illustrated in Fig. 1, the output of the demodulator is fed back to the oscillator via a loop filter. This creates a so-called FFL (frequency-feedback loop) which reduces the frequency deviation of the received signal by a factor of five.

The demodulated signal is also applied to a 'mute' circuit via an inverter. This attenuator suppresses all products that do not look like a transmitter signal, and is actuated via two detectors. The first responds to signs of mistuning in the demodulated signal, while the other ensures that the mute is switched on if the signal is too weak. The latter information is taken from a point right behind the block marked 'IF limiter'.

The audio signal processed by the mute function leaves the IC via pin 2.

Search tuning

Another specialty of the TDA7088T is its easy-going tuning system. This is basically a push-button operated station finder with AFC. The relevant block is shown in the left-hand bottom corner in Fig. 1, and its operation certainly merits a short description.

When the 'run' push-button connected to pin 15 is pressed, a positive pulse is applied to the 'set' input of the 'search tuning' circuit. The 0.1-µF capacitor connected to pin 16 is then charged. The slowly rising voltage across this capacitor is fed to the variable-capacitance diode ('varicap') in the VCO. This voltage changes the receiver tuning. As soon as a station is received, the charging of the above mentioned capacitor is stopped at the command of the two detectors in the mute circuit. Next, the AFC (automatic frequency control) is actuated, which ensures that the tuning voltage is held at the current value. This value does not change until the 'run' button is pressed again. An internally created threshold prevents the capacitor voltage from exceeding about 1.8 V, a value which is well below the minimum allowed supply voltage. Consequently, the varicap used should have a capacitance ratio that enables the entire VHF FM broadcast band to be covered with a tuning voltage range as small as 0 to 1.8 V.

When the 'reset' button is pressed, the 0.1-µF capacitor is discharged, and a new station search is performed, starting at the low end of the band.

A matchbox-size FM receiver

The circuit diagram in **Fig. 2** illustrates the simplicity of a complete FM receiver based on the TDA7088T. The number of external parts is very low indeed. With some mechanical skill, it should be possible to fit the receiver into a matchbox, complete with two button cells as a power supply. Let's examine the circuit diagram.

The input tuned circuit of the receiver is connected between pins 11 and 12 of IC_1, and consists of L_1, C_1

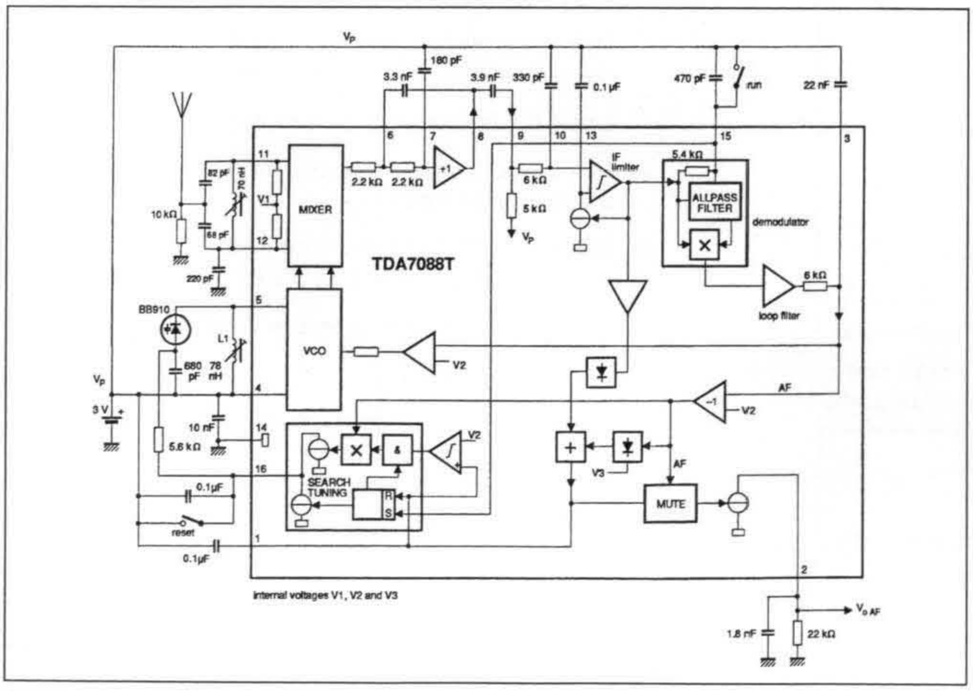


Fig. 1. Block diagram of the TDA7088T integrated FM receiver from Philips Semiconductors.

and C_2. This tuned circuit is fairly 'broad', so that it does not have to be re-tuned for each station in the band. The antenna is connected to the junction of C_1 and C_2, which is actually a capacitive tap on the tuned circuit. Resistor R_1 serves to shunt static electricity to the ground rail. The oscillator tuned circuit is found between pins 4 and 5. It consists of L_2, D_1 and C_{14}. The component values differ very little from those of the input circuit, but that is not surprising because the input frequency and the oscillator frequency are only 73 kHz apart.

Varicap D_1 is provided with a tuning voltage supplied by pin 16, via resistor R_2. The operation of the automatic station seek function is controlled by the 'run' and 'reset' buttons, S_1 and S_2. The external capacitors of the IF filter are connected to pins 6 though 10.

That completes the description of the actual receiver in the mono version. The demodulated AF signal is available at pin 2 of the TDA7088T, and may be fed to any small amplifier capable of driving headphones. If you want to keep the size of the receiver to an absolute minimum, use the one-transistor driver shown in Fig. 2. This amplifier is capable of driving a pair of Walkman-type headphones with an impedance of 32 Ω (connect the loudspeaker elements in parallel).

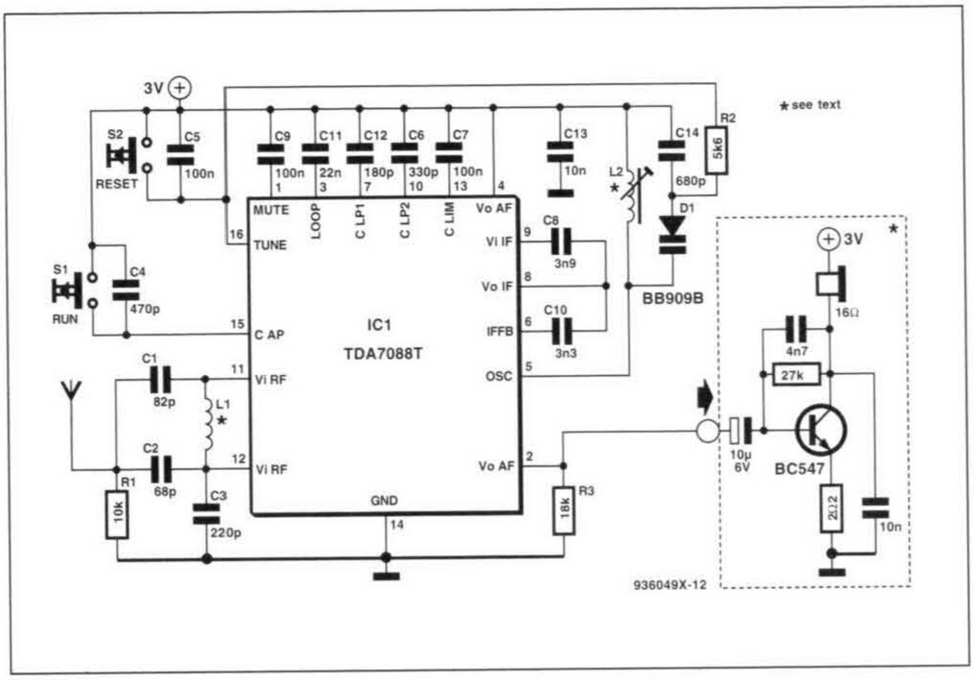


Fig. 2. Circuit diagram of a miniature, mono, FM receiver based on the TDA7088T.

The stereo version

The circuit diagram in **Fig. 3** shows a battery-powered, pocket-size, personal FM stereo radio for use with headphones.

Even a cursory look at the circuit diagram brings out the likeness with the circuit shown in Fig. 2. The only difference is that the simple AF stage is replaced by a dedicated FM stereo decoder and a stereo output amplifier. Because both components are highly integrated, they can not be said to make the receiver more complex.

IC_3 is the stereo decoder. The miniature IC package contains a fairly complex circuit which is not discussed here. Suffice it to say that IC_3 turns the L+R and L–R components of the stereo MPX (multiplex) signal into 'left' and 'right' audio signals. The auxiliary carrier required for this conversion is generated with the aid of an internal oscillator, which is synchronized by the pilot carrier, and adjusted by preset P_1. Those of you who wish to learn more about MPX FM stereo signals are advised to read the article 'FM Stereo Signal Generator' in the May 1993 issue of *Elektor Electronics*.

The decoded L and R signals at the outputs of IC_3 (pins 5 and 6) are fed to stereo volume control, P_2, near IC_2. This tiny component houses two complete audio amplifiers capable of sup-

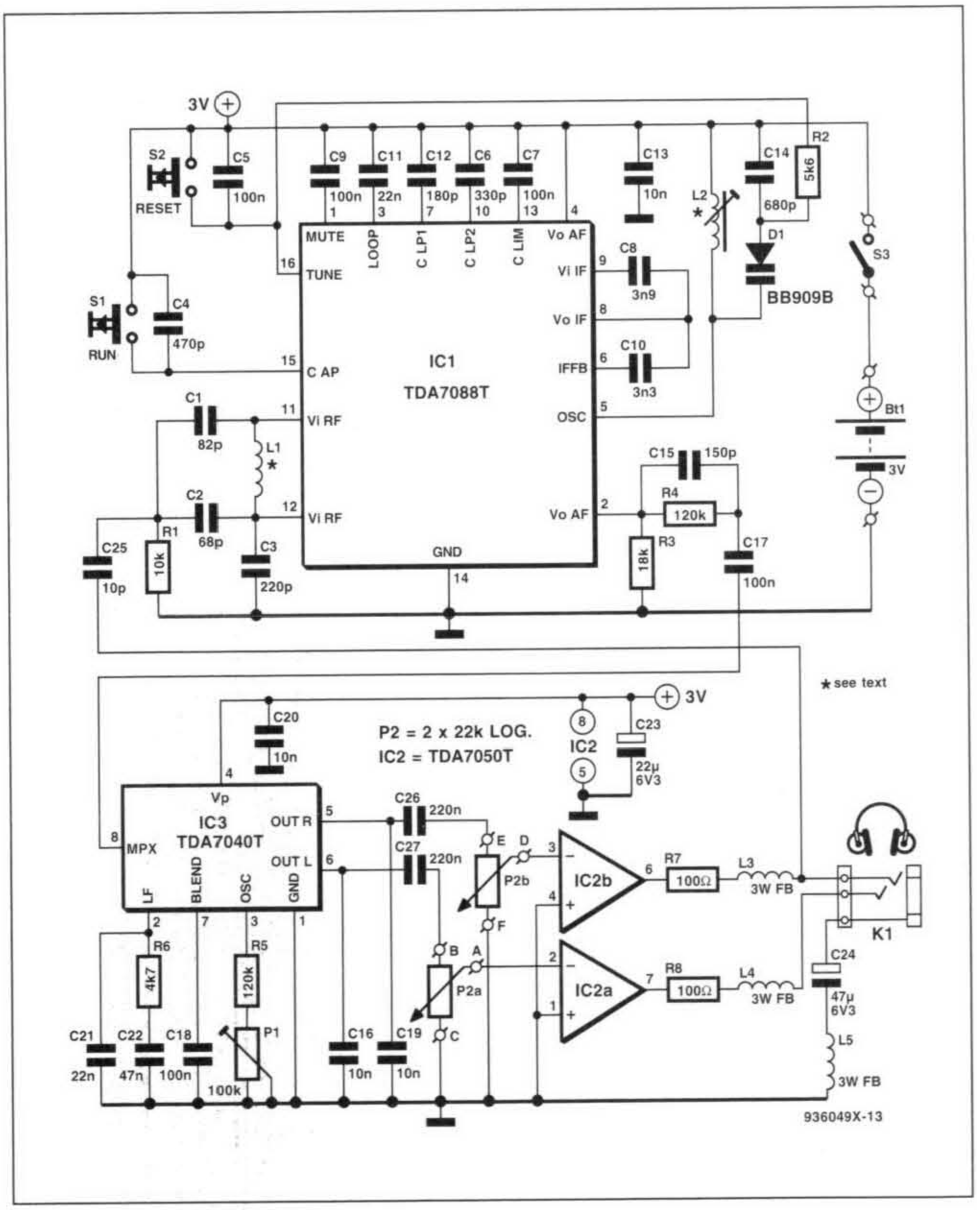


Fig. 3. Circuit diagram of the actual FM receiver, as it is implemented on a printed circuit board. This version of the receiver features a stereo decoder and a stereo output amplifier.

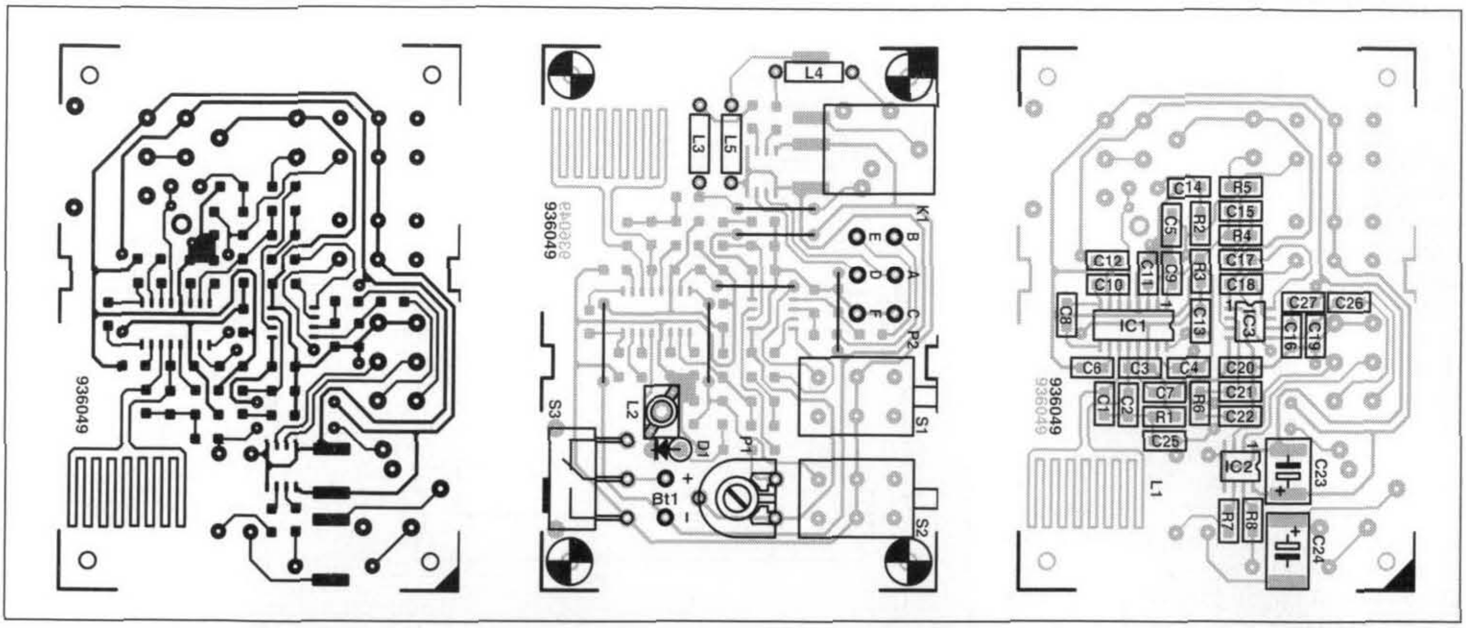


Fig. 4. Track layout and component mounting plans. Note that components are fitted at both sides of the PCB: the SMT devices go to the solder side, and the 'regular-size' parts, to the top side. Don't forget those wire links!

COMPONENTS LIST

Resistors: (SMDs)
R1 = 10kΩ
R2 = 5kΩ6
R3 = 18kΩ
R4,R5 = 120kΩ
R6 = 4kΩ7
R7,R8 = 100Ω
P1 = 100kΩ preset
P2 = 22kΩ stereo, log. potentiometer

Capacitors (SMDs):
C1 = 82pF
C2 = 68pF
C3 = 220pF
C4 = 470pF
C5,C7,C9,C17,C18 = 100nF
C6 = 330pF
C8 = 3nF9
C10 = 3nF3
C11,C21 = 22nF
C12 = 180pF
C13,C16,C19,C20 = 10nF
C14 = 680pF
C15 = 150pF
C22 = 47nF
C23 = 22μF 6V3
C24 = 47μF 6V3
C25 = 10pF
C26,C27 = 220nF

Semiconductors:
D1 = BB909B
IC2 = TDA7088T (SMD)
IC3 = TDA7040T (SMD)

Inductors:
L1 = copper track on PC board
L2 = E514HNE150014S14 (Toko)
L3,L4,L5 = 3 turns 0.2mm dia. e.c.w. through ferrite bead

Miscellaneous:
K1 = stereo jack socket, 3mm, for PCB mounting.
S1,S2 = presskey, single pole, Siemens type BO2AMAP-2.
S3 = slide switch, single pole, angled pins, PCB mount.
Bt1 = 3-V battery (2 penlights).
Case: Heddic type 222.
Printed circuit board, order code 934049, see page 70.

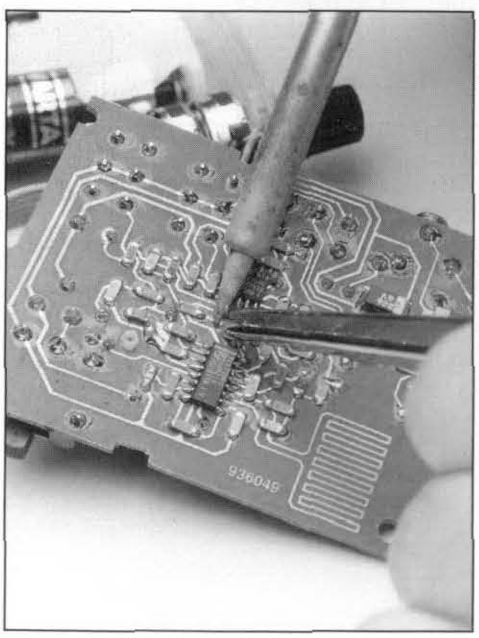


Fig. 5. Soldering the SMT parts is not half as difficult as it is often said to be. In fact, all you need is accuracy, a cleaned up work desk, good eyesight, a steady hand, and a low-power solder iron!

plying enough power to a pair of headphones. There is really not much to say about the TDA7050T except that the IC works virtually without external parts. Electrolytic capacitor C_{24} and resistors R_7 and R_8 are the only ones used here. The headphones are connected to 'jack' socket K1.

Inductors L_3, L_4 and L_5 allow the headphones cable to act as an antenna. One of the contacts on K_1 is connected to the antenna input of the TDA7088T. The chokes (which consist of ferrite beads with a few turns of wire through them) ensure that the RF signal is not short-circuited by IC_2, or lost in the ground rail via C_{24}. The high reactance (to RF) of the inductors forces the antenna signal to travel to the input of IC_1 via C_{25}.

Construction

The receiver is built almost entirely from surface-mount technology (SMT) parts, which are also called SMDs (surface-mounted devices) in some catalogues. The layout of the single-sided printed circuit board for the receiver is given in **Fig. 4**. As you can see, there are two component mounting plans, one for the SMT parts at the copper side, and one for the regular-size parts at the component side. An important detail of the circuit board is that inductor L_1 is a so-called stripline, i.e., it is formed by a thin copper track on the board.

Because L_2 is available ready-made, there are no home-made inductors in the circuit. Only chokes L_3, L_4 and L_5 need to be wound, but that is extremely simple. Just a few turns of enamelled copper wire through a ferrite bead, that's all.

Populating the component side of the board is not expected to cause problems since only regular-sized parts are involved. Not so with the other side, however, which holds the tiny SMT parts. To begin with, make sure you leave the SMT resistors and capacitors in their bags. If they get mixed up, you will have a big problem finding out the values which may not

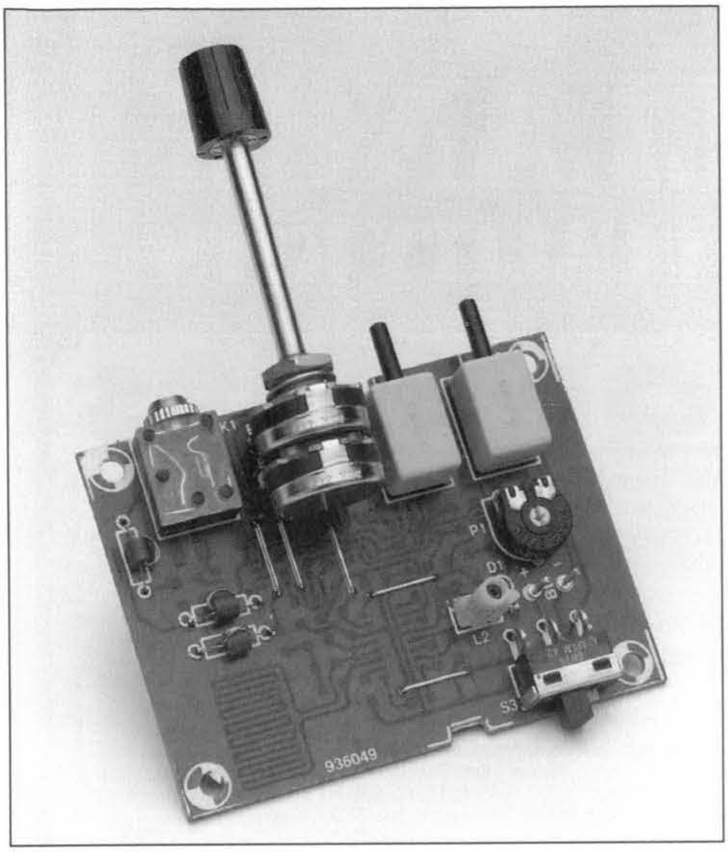


Fig. 6. The completed board should look like this when viewed from the top side...

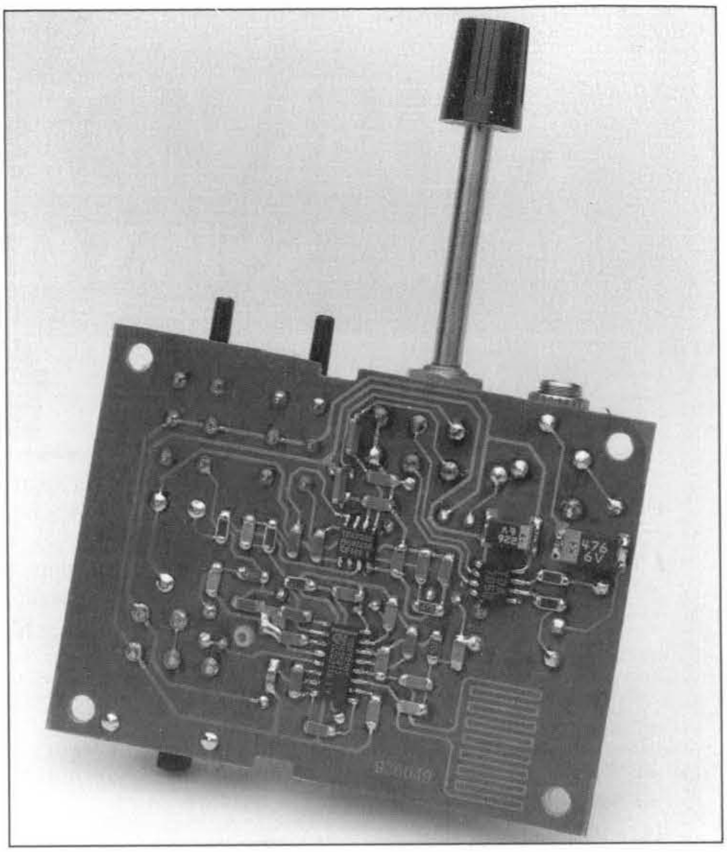


Fig. 7. and like this when viewed from the solder side.

be printed on the devices. With SMT electrolytic capacitors, the white band or dash indicates the positive terminal.

Use a low-power (max. 15 W) solder iron with a fine tip, and pre-tin the end caps on the passive SMT parts, holding them in tweezers. Use tin sparsely. Next, use the tweezers to press the part in place on the board, and solder one side. Check the position, correct if necessary, and them solder the other side. The ICs are mounted similarly. First, pre-tin two diagonally located corner pins of the device, as well as the relevant solder spots on the board. Press the IC on the board and align its position. Solder one corner pin. Check the position of the device again, and if necessary align the pins with the copper pads. Then carefully secure each pin, making sure you do not use too much solder tin. The photographs in **Figs. 6 and 7** tell you what the PCB should look like if everything is mounted correctly.

Finding a suitable, attractive, case for the receiver should not be a problem thanks to the compactness of the board. The prototype was built into a transparent case from Heddic. Although all control elements (switches, output socket and potentiometer) are, in principle, fitted on the board, they may also be mounted externally and connected to the board by short wires. It all depends on the enclosure you wish to use.

Adjustment

Adjustment is simple, and requires no special tools. There are only two adjustment points, L_2 and P_1. Start with the inductor. Remember, each action on the 'run' key causes the receiver to tune to a higher frequency, while pressing 'reset' re-tunes it to the start of the band. Initially, turn the core in L_2 about half-way into the former. Switch on the receiver. After pressing 'run' a few times, a couple of FM stations should be received. If not, run a thorough check on your construction work, and correct any errors. Assuming that the receiver does work, you press the 'reset' button, and then adjust the core in L_2 (with a plastic trimming tool) until you receive a station which you know transmits at a frequency between 87 and 88 MHz. This adjustment sets the start of the band.

The adjustment of P_1 is even more simple. Tune to a station which is certain to transmit in stereo, and adjust the preset until you actually hear stereo sound. That may not be so easy with all stations, and sometimes you may only notice an increase in the audio noise level. Repeat this adjustment with a few other stations. Happy listening! (936049X)

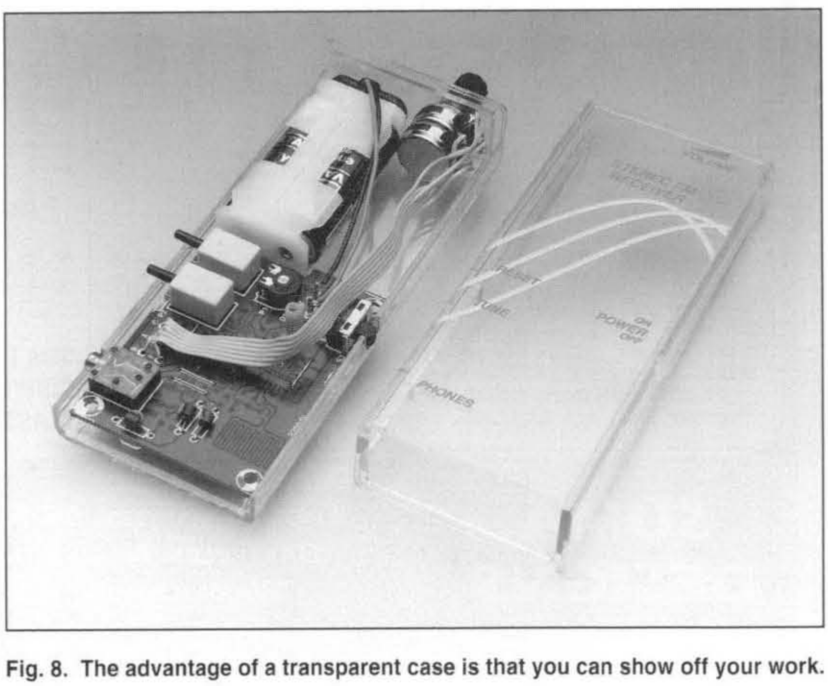
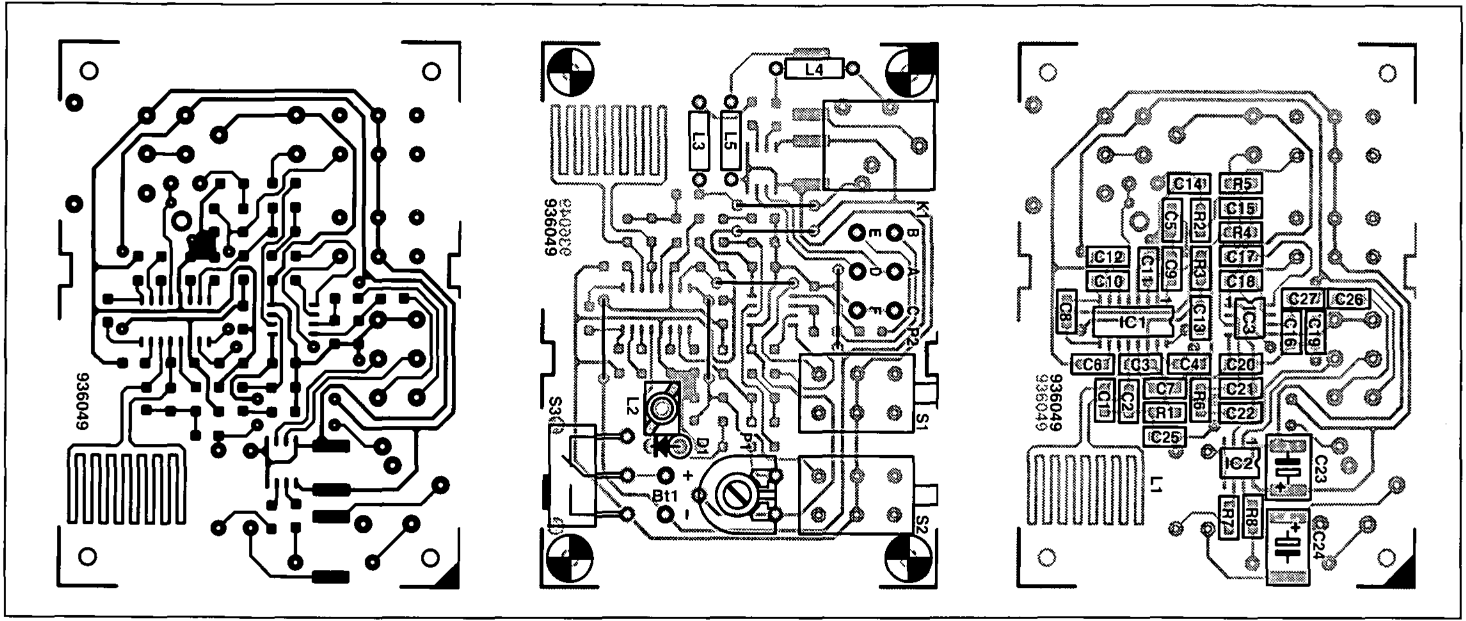
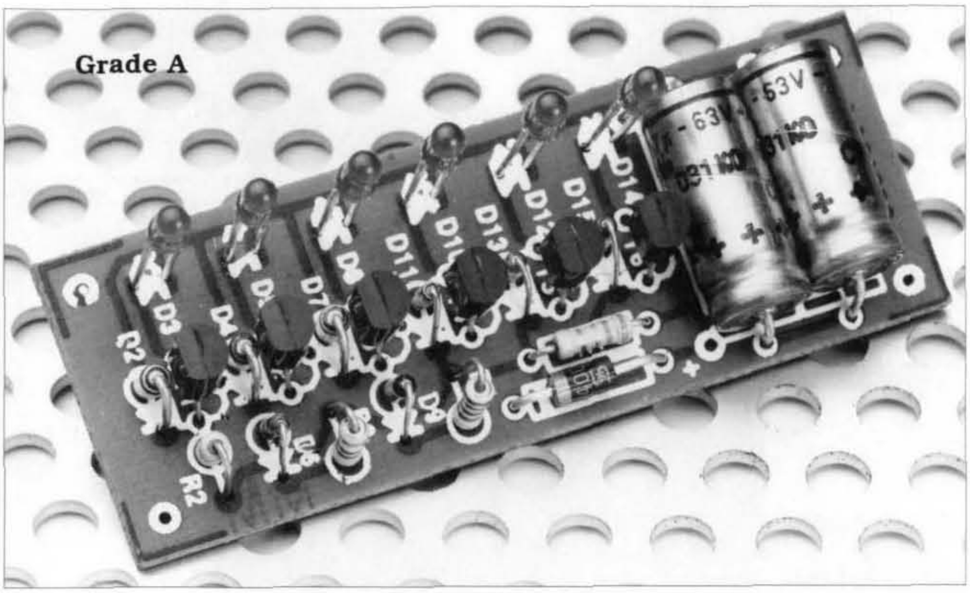


Fig. 8. The advantage of a transparent case is that you can show off your work.


936049
L4
L3
L5
K1
B
E
A
D
C
F
P2
S1
S2
S3
L2
D1
P1
Bt1
+
-
C14
R5
C15
C5
R2
R4
C12
C11
C9
R3
C17
C10
C18
C8
IC1
C13
IC3
C27
C26
C16
C19
C6
C3
C4
C20
C1
C2
C7
R6
C21
R1
C22
C25
IC2
C23
L1
R7
R8
C24

PASSIVE VU METER



Design by T. Giesberts

If you want to fit your loudspeaker enclosure(s) with a drive indicator, it is best to use a unit that does not need a power supply. The September 1995 issue of this magazine (p. 75) described an AF POWER INDICATOR for PA (public address) loudspeakers, that is, for fairly high power outputs. This article describes a passive VU unit for smaller amplifiers and loudnesses—see **Fig. 1**. It consists of a rectifier and six (may be fewer or more) identical stages that each comprises a current source, a zener diode and a light emitting diode, LED. The current sources are built from JFETs with interconnected gate-source terminals—here, Type BF256A. The saturation current, I_{DSS}, of these devices with a drain-source voltage, U_{DS} = 15 V is about 5 mA. This current is not exactly constant, but is perfectly all right for driving a low-current LED and will not exceed the permissible value of 7 mA. Networks R_2-D_2, R_3-D_4, and R_4-D_7 are protection circuits; they prevent the drain voltage of the relevant JFET rising above 30 V, which normally destroys the transistor.

The rectifying circuit is formed by D_1 and capacitors C_1, C_2. Resistor R_1 limits the peak current to about 1.5 A at a source voltage, U_S, of 50 V. Since it is in series with C_1 and C_2, and thus in parallel with the amplifier output, it has no effect on the level of the input voltage. The peak output voltage of the rectifier is applied directly to the single LED stages. The potential across C_1 and C_2 is not exactly equal to the instantaneous peak voltage, but, because of time constant R_1-C_1-C_2, is a good average of it. Consequently, the unit indicates briefly the instantaneous peak voltage, and then the mean of it.

The coworking of the three parts of a stage is easily understood by considering the following. If the rectified and smoothed voltage rises a few volts over the level set by the zener diode, the current source comes into action and causes the LED to light. Since the (input) voltage to the meter is directly proportional to the amplifier output and the (assumed constant) impedance of the loudspeaker, the indicated threshold level (in watts) can be converted into a zener voltage:

$$P = U_{RMS}/R = (U_s/\sqrt{2})^2/R = U_s^2/2R$$

$$\therefore \quad U_s = U_{ZENER} = \sqrt{2PR} - U_{LED}$$

where U_{LED} is the starting voltage of the LED (and the voltage drop across the current source), which is equal to 2 V. Thus, for an indication of 100 W into an 8 Ω loudspeaker, the zener voltage is

$$U_{ZENER} = \sqrt{(100 \times 2 \times 8)} - 2 = 38 \text{ V}.$$

The zener to be used should have the next lower rating in the table (36 V), so that it lights brightly when the output is 100 W. In this way, the stages may be designed more or less to individual requirements.

In the most sensitive stage, T_1-D_2-D_3, the zener diode is, strictly speaking, superfluous since the indicated power is determined entirely by the threshold values of D_1, T_1 and D_3.

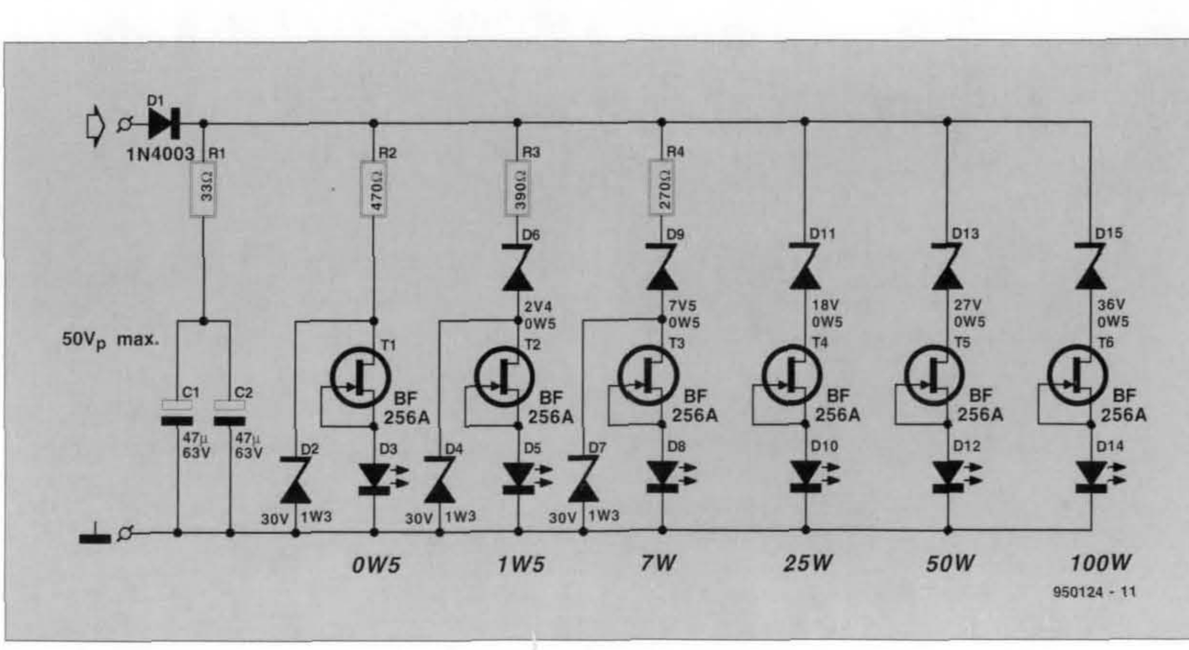


Fig. 1. The circuit of the passive VU meter is based on six identical stages.

The input current to the circuit has a peak level of U_{in}/R_1 = 50/33 = 1.5 A. With a constant 1 kHz signal and an output level of 150 W into 8 Ω, the current drops to 280 mA. However, if this signal is pulsed with a duty factor of 1:99, the current rises to 1.3 A owing to the then low average potential across capacitor C_1. It is noteworthy that the circuit is not truly passive, because it draws its energy from the audio signal. This means, by the way, that there is a (very) slight rise in the distortion!

The circuit may be built quickly and without any undue difficulties on the printed-circuit board shown in **Fig. 2**. The finished board is best fitted in a small plastic enclosure, which is then fitted on to the loudspeaker box(es).

Standard zener diode voltages (V)

1.0*	10	100
	11	110
	12	120
1.4*	13	130
1.5*	15	150
	16	160
	18	180
2.0*	20	200
	22	
2.4	24	
2.7	27	
3.0	30	
3.3	33	
3.6	36	
3.9	39	
4.3	43	
4.7	47	
5.1	51	
5.6	56	
6.2	62	
6.8	68	
7.5	75	
8.2	82	
9.1	91	

* rare

Parts list

Resistors:
R_1 = 33 Ω
R_2 = 470 Ω
R_3 = 390 Ω
R_4 = 270 Ω

Capacitors:
C_1, C_2 = 47 μF, 63 V

Semiconductors:
D_1 = 1N4003
D_2, D_4, D_7 = zener diode 30 V, 1.3 W
D_3, D_5, D_8, D_{10}, D_{12}, D_{14} = low-current LED
D_6 = zener diode 2.4 V, 500 mW
D_9 = zener diode 7.5 V, 500 mW
D_{11} = zener diode 18 V, 500 mW
D_{13} = zener diode 27 V, 500 mW
D_{15} = zener diode 36 V, 500 mW
T_1–T_6 = BF256A

Miscellaneous:
PCB Order no. 950124

[950124]

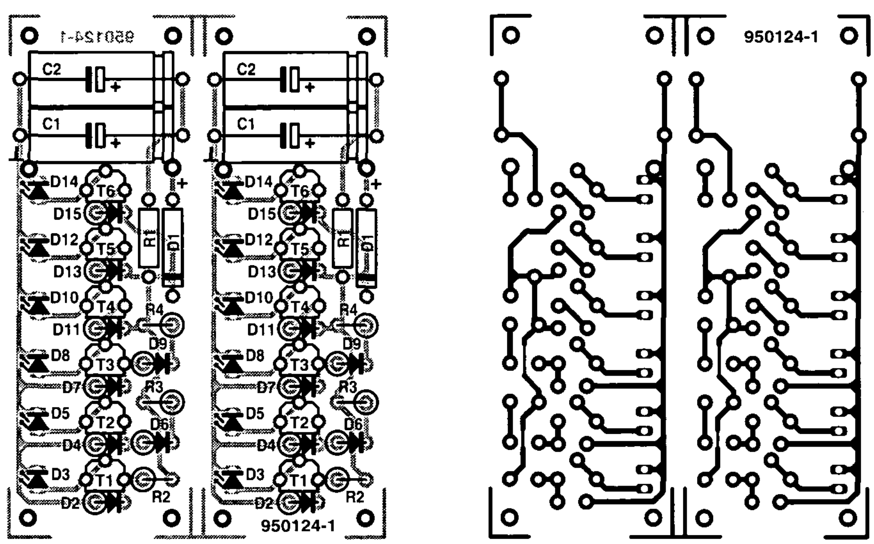
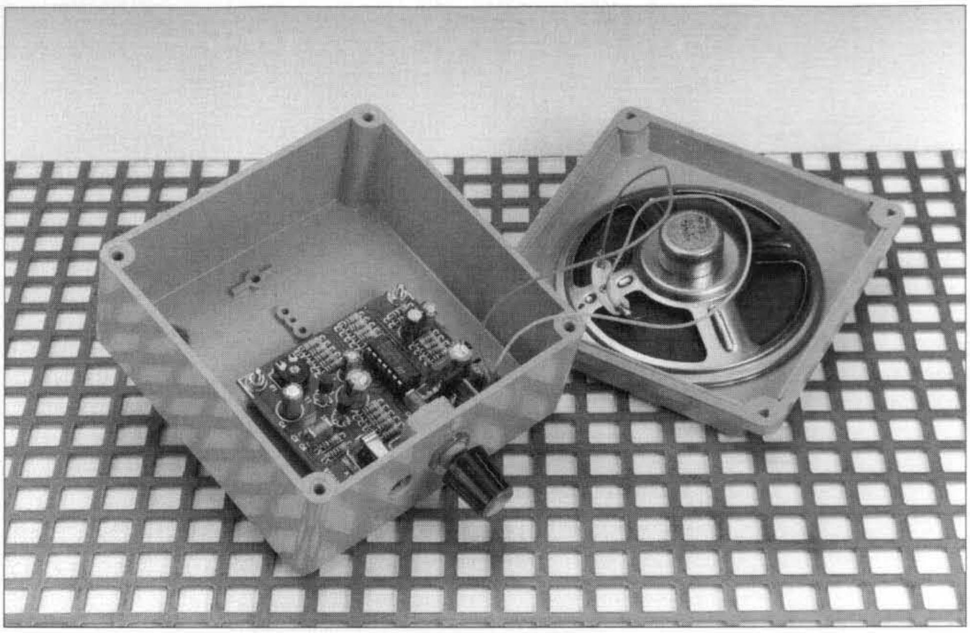


Fig. 2. The printed-circuit board for the passive VU meter must be cut into two before any assembly work is begun.

WAVE SOUND GENERATOR

The noise of small rippling waves and a gentle breeze has a calming effect which may be exploited by insomniacs and others who find it difficult to reach total relaxation in these hectic times. Whatever the reason for this stress-reducing effect, we present a low-cost, simple to build circuit which can be adjusted to produce wave and wind noises.



Design by K. Walraven

FRANKLY, we are unable to explain with scientific certitude why the sound of the sea has a calming effect on people, and can be a great help for insomniacs wishing to steer clear of sleeping pills and other medication. Although the inset in this article provides a plausible explanation based on a few simple facts from practical psychoacoustics research, it is no more than an attempt at explaining a neurological phenomenon whose complexity is such that we can not even scratch the surface in this article.

Apart from helping you to relax and fall asleep more easily, the wave sound generator may also be used as a background noise generator for amateur movies, sound dubbing, etc.

How it works

The heart of the circuit is a run-of-the-mill transistor which is purposely connected the wrong way around to make it behave like a zener diode. The zener effect has been known for years to produce quite a bit of noise, which rises, in principle, with the zener current. Normally, this noise is undesirable, and therefore decoupled with a large bypass capacitor. Here, however, everything is done to get as much noise as possible from the collector-emitter junction of T_1.

Let's first look at what happens to the noise voltage generated by T_1. The signal is capacitor-coupled to the base of T_2, which acts as an amplifier whose gain is adjustable with pot P_2. Then follows another amplifier, this time an opamp, IC_{1a}. Its operating point is set to about half the supply voltage by potential divider R_{18}-R_{19}. The noise signal is applied to the +input of the opamp via coupling capacitor C_8. Although this capacitor has a small value for audio signals, that is not a problem since low-frequency components are almost absent because of the design of the first amplifier stage, and the use of a small loudspeaker. The loudspeaker is driven by a simple, discrete, class-B power amplifier based on inexpensive transistors type BD139/BD140. The amplifier has feedback via resistor R_{22}. A low-power (approx. 1 W) general-purpose loudspeaker is connect to the output of the wave sound generator.

The circuit operates off a 12-V regulated supply rail created by IC_2. Input power is obtained from a mains adaptor set to an (actual) output voltage of about 15 V d.c. and connected to socket K_1 on the board.

Because sea noise is never a steady level, something has to be done to modulate the noise source if we want a realistic output sound. The zener current sent through T_1 is caused by a voltage at the junction of R_{16}, C_4, P_1, R_5, R_{10} and R_{15}. Here, a so-called complex envelope waveform is available which mimics the irregular sound produced by larger and smaller waves. In fact, the comparison with waves could not be more appropriate, because the circuit contains three waveform generators. The three waveform generators are built around three opamps contained in IC_1. The generators are almost identical except for some of their component values, and the different take-off point on the lower generator. The two top oscillators, IC_{1b} and IC_{1c}, generate a rectangular wave signal which is converted into a sawtooth waveform by capacitor C_4. The sawtooth mimics the swell and decay of a wave. A fairly random complex waveform is developed because the three waveform generators operate asynchronously and at different frequencies (look at the values of the feedback resistors, R_4, R_9 and R_{14}, and those of the capacitors, C_1, C_2 and C_3). The lower oscillator supplies a fairly low frequency, at which C_4 no longer functions very well as an integrator. This problem is solved by taking the modulation signal from the – input of the opamp, where a sawtooth is already present. The effect produced by IC_{1d} is that of large waves as heard on ocean shores. Note, however, that no attempt has been made at imitating some of the very deep sounds that may be produced by the sea[1]. The same goes for higher-pitch, extremely complex, sounds heard in, for example, storms. Remember, this is simple circuit, not a complex sound generator.

The generator output waveforms are summed by R_5, R_{10} and R_{15} to give the previously mentioned complex envelope waveform which serves to modulate the zener's noise voltage. A preset potentiometer, P_1, is provided to enable the best noise production to be achieved, and to enable you to adjust for slightly different sound effect settings.

Test while you build

Although the circuit is likely to work spot-on if it is carefully assembled according to the parts list and the component mounting plan (**Fig. 2**) it may be worthwhile, particularly for students and beginners, to follow the

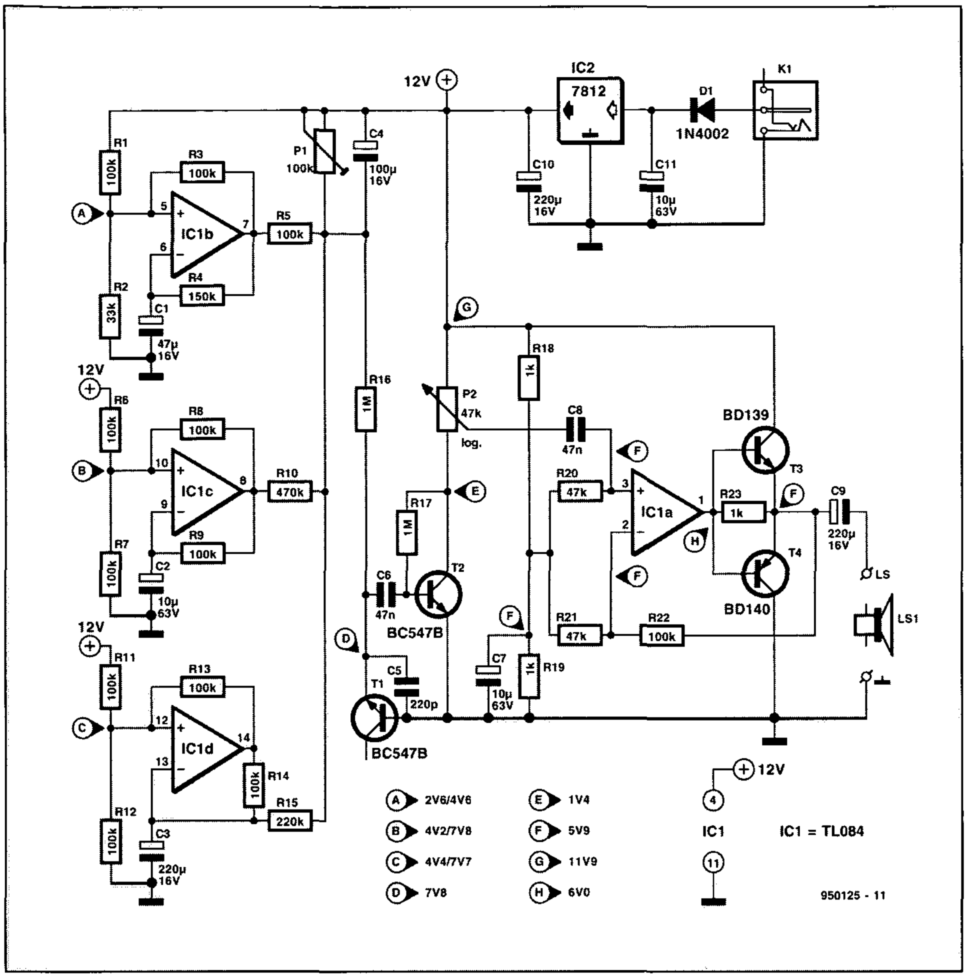


Fig. 1. Although the wave noise generator consists of commonly available and inexpensive parts, it still produces a fairly natural sounding imitation of sea waves and wind.

step-by-step assembly guidance given below.

Start by switching off the soldering iron, and concentrate on the mechanical construction of the case. Select a plastic case which will easily hold the board and, if desired, the loudspeaker. Drill holes in one of the side panels for the volume control (P_2) and the mans adaptor socket (K_1). Next, drill a number of small holes in the top panel for the loudspeaker.

Having finished the mechanical work you may fire up your iron again, and get out the parts. Start by fitting the parts that make up the power supply and the audio power amplifier, in other word, everything to the right of the dashed line in the circuit diagram. Do not forget the wire link on the board! Note that diode D_1 is soldered below the spindle of the potentiometer. It is not necessary to connect the loudspeaker yet. Fit the TL084 in its socket, taking care to observe the orientation (look at the notch in the device, and the symbol drawn on the component overlay). Switch the mains adaptor to an output voltage of 9 V to start with, and connect it to the wave sound generator. Measure the adaptor output voltage again, and if necessary step it up to make sure that you have at least 15 V at the input of IC_2. Check for the presence of about 6 V at junction R_{18}-R_{19}. If you can not find 6 V here, it is likely that you have made a mistake with the values of R_{18} and/or R_{19}, or C_7 is fitted the wrong way around. The 6-V potential should also be measured at pin 3 of the IC and at its output, pin 1. If not, there is a fault

COMPONENTS LIST

Resistors:
R1,R3,R5,R6-R9,R11-R14,R22 = 100kΩ
R2 = 33kΩ
R4 = 150kΩ
R10 = 470kΩ
R15 = 220kΩ
R16,R17 = 1MΩ
R18,R19,R23 = 1kΩ
R20,R21 = 47kΩ
P1 = 100kΩ preset H
P2 = 47kΩ log. potentiometer

Capacitors:
C1 = 47µF 16V radial
C2,C7,C11 = 10µF 63V radial
C3,C9,C10 = 220µF 16V radial
C4 = 100µF 16V radial
C5 = 220pF
C6,C8 = 47nF

Semiconductors:
D1 = 1N4002
T1,T2 = BC547B
T3 = BD139
T4 = BD140
IC1 = TL084
IC2 = 7812

Miscellaneous:
K1 = Mains adaptor socket, PCB mount.
LS1 = Loudspeaker, 8Ω/1W.
K2,K3 = Solder pins.
Suitable case.

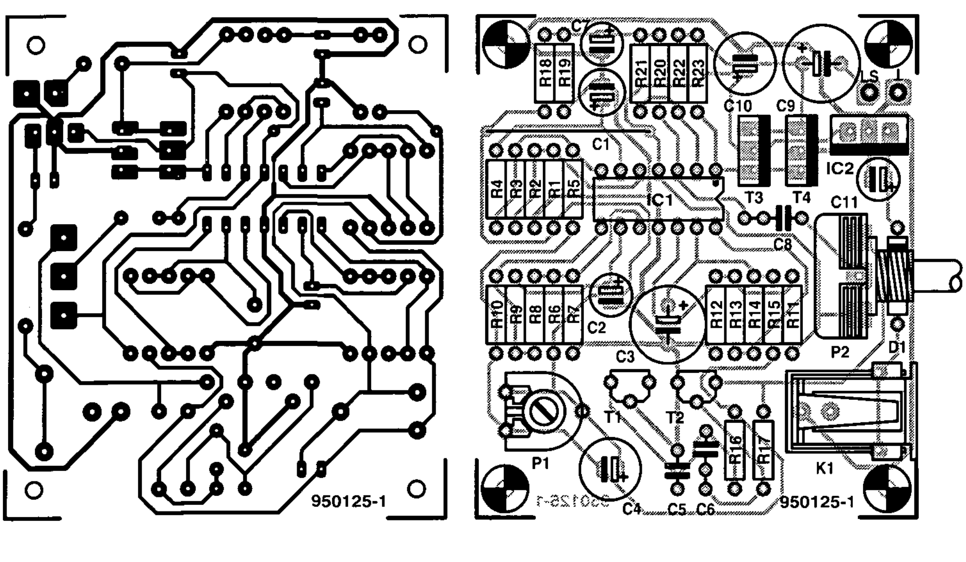


Fig. 2. PCB track layout and component mounting plan (board not available ready-made through the Readers Services).

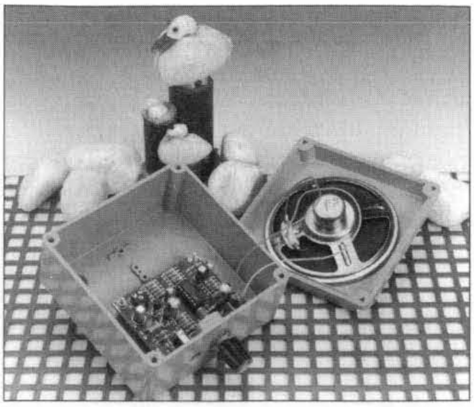


Fig. 3. Interior view of the built-up prototype. Seagulls and pebbles courtesy of your technical editor.

in the transistor output circuit (T_3-T_4). Check that the right transistor is in the right place (apart from their type number the BD139 and BD140 look very similar!). Okay so far? Then hook up the loudspeaker. Touching C_8 should produce a little hum. Current consumption at this stage should be about 17 mA.

Proceed by building up the centre section of the circuit. Set P_1 and P_2 to the centre of their travel. Power up, you should hear brisk noise. Measure the voltage across T_1, this should be between 7 V and 12 V. Check that the volume is controlled by P_2.

Okay, and now on with the wave generators. These may be built in one go, but is it more fun, and also more instructive didactic, etc., if you fit them one by one because then you are able to hear the effect of each individual waveform component on the output noise. You will find that P_1 has to be re-adjusted every time you add a generator. That is because R_{10} and R_{15} affect the bias on T_1, and so the output noise level. The best setting of P_1 is that which causes the generator to produce noise at all times, i.e., also in the almost silent periods of the 'waves'.

Finally, the current consumption of the circuit will be between 25 mA and 100 mA depending on the output volume.

Postscript

For various reasons this circuit should not be used by epileptic patients, or sufferers from neurological disorders. In case of doubt consult your GP. And also for all electronically inclined, if at all possible reduce your stress by natural rather than electronic therapy! If you are fortunate enough to live by the seaside, there is nothing to match the wholesome effect of a coastline walk with moderate wind in the early evening. Batteries not included.

(950125)

HOW CAN NOISE HAVE A RELAXING EFFECT?

The relaxing effect of the present circuit has probably little to do with unconscious association with tropical beaches, holidays, etc. Pink (i.e., audio-band filtered) noise contains, in principle, all frequencies that can be perceived by the human ear. Psychoacoustic research seems to indicate that human speech has a high degree of redundancy. This can be proved by cutting small fragments from utterances, and replacing these fragments with pink noise bursts. In most cases, test persons are perfectly capable of understanding what was being said originally despite heavy substitution of fairly large sound chunks (morphemes, syllables) by noise bursts. This lead to the assumption that these chunks are 'recovered' from noise by the brain, on the basis of expectation patterns. A good example of this happening in practice is the astounding ability of some radio amateurs to extract messages from what sounds like a lot of interference to the outsider's ear. Psychoacoustic experiments in which sound chunks were replaced by 'blanks' gave significantly worse recognition rates. From a point of view of electronics, that is not surprising, because noise is, in principle, all frequencies together. Taking this a step further, it may be assumed that feeding just pink noise, with some modulation (for variety), to the hearing system causes a part of the brain to 'switch off' because of its inability to recover sounds that make sense (not necessarily speech) by applying the expectation patterns. Total silence, on the other hand, hardly ever has a relaxing effect. In fact, some of us find it claustrophobic, and most of us, oppressing. Finally, it must be mentioned that a side effect of sea noise is the totally random character of the wave sounds. This randomness could also account for the relaxing effect, because it makes no sense for the hearing system to predict or structure sound fragments (which is normally very well possible, and indeed, essential if you listen to speech or music).

[1] Fragment from *The Dry Salvages, Four Quartets*, by T.S. Eliot (1941).

The sea howl
And the sea yelp, are different voices
Often together heard: the whine in the rigging,
The menace and caress of wave that breaks on water,
The distant rote in the granite teeth,
And the wailing warning from the approaching headland
Are all sea voices, and the heaving groaner
Rounded homewards, and the seagull:
...

MEASURING CAPACITANCE

By Dr K C Rohwer

Many circuits intended for measuring capacitance depend on an often difficult to obtain and expensive reference capacitor of precisely known capacitance. If you have a frequency meter that can measure pulse duration, it is possible to carry out accurate capacitance measurements without the need of a reference capacitor

When a capacitor of unknown value, C_X, is charged by a constant current, I_0, the potential across it, U_X, varies linearly with time, t:

$$U_X = I_0/C_X t \qquad [1]$$

To reach a certain potential, the time must be proportional to the capacitance. When voltage and current are known, the capacitance can be computed on the basis of the measured time.

A suitable network is shown in **Fig. 1**, in which a constant-current source, providing 1 μA, charges C_X. At the same time, generators A_1 and A_2 provide a pulse that starts when U_X exceeds ascertain value and ends when U_X has risen by exactly 1 V. The pulse duration can be measured accurately with a frequency meter; according to Eq. 1, the time lapse is 1 ms nF^{-1}.

The accuracy of the capacitance measurement thus depends on that of the current, voltage and time measurements, which in modern instruments is high.

The measurement can be simplified even further. The circuit in Fig. 1 is similar to that of a Type 555 IC., and such a device can, therefore, be used as basis of the circuit. It is even possible to do without the constant-current source if you don't mind some arithmetic. The potential across C_X follows an exponential curve as shown in **Fig. 2**.. From this:

$$U_X(t) = U_B(1-\exp(-t/RC_X)), \qquad [2]$$

where R is the charging resistor between C_X and $+U_B$. Typical change-over voltage levels are $U_B/3$ and $2U_B/3$, which give pulse widths t_1 and t_2, so that

$$U_X(t_1) = U_B/3, \qquad [3]$$

and

$$U_X(t_2) = 2U_B/3. \qquad [4]$$

Substituting these equations in [2] and rearranging gives:

$$\exp(-t_1RC_X) = 2t_1/3RC_X = -\ln(2/3), \qquad [5]$$

and

$$\exp(-t_2RC_X) = 1t_1/3RC_X = -\ln(1/3). \qquad [6]$$

Arranging the pulse durations so that $t = t_1 - t_2$, gives:

$$t/RC_X = \ln(2/3) - \ln(1/3) = \ln 2 \qquad [7]$$

From this and the stated ratio of t/C_X:

$$R = t/(C_X \cdot \ln 2), \qquad [8]$$

which is independent of U_B. For a pulse duration of 1 ms nF^{-1}:

$$R = 1.4427\ \text{M}\Omega.$$

Practical circuit

Figure 3 shows a circuit that makes use of these considerations. It provides an periodic output signal, whose pulse duration in ms corresponds to the capacitance of C_X in nF. Resistance R is formed by R_1, R_2 and R_3. Resistor R_3 determines the discharge time of C_X and thus the pulse width.

The circuit is suitable for measuring capacitances between 4.7 nF and 1 μF; the lower value is determined by the type of frequency meter used. The upper limit can be as high as wanted, but it may take a while before the result can be read.

After C_X has been connected to the test terminals, it takes $2\Delta t$ before the capacitance has been charged to $2U_B/3$. The first indicated value must be ignored, because the output pulse did not start at $U_X = U_B/3$, but at $t = 0$. The first usable measurement takes another Δt : 1 s μF^{-1}.

A drawback with the measurement of large capacitances is the degrading

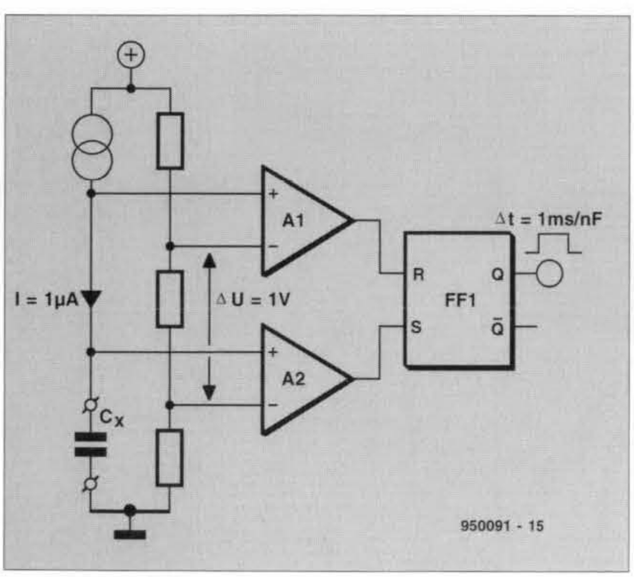


Fig. 1. Principle of capacitance measurement by pulse duration measurement.

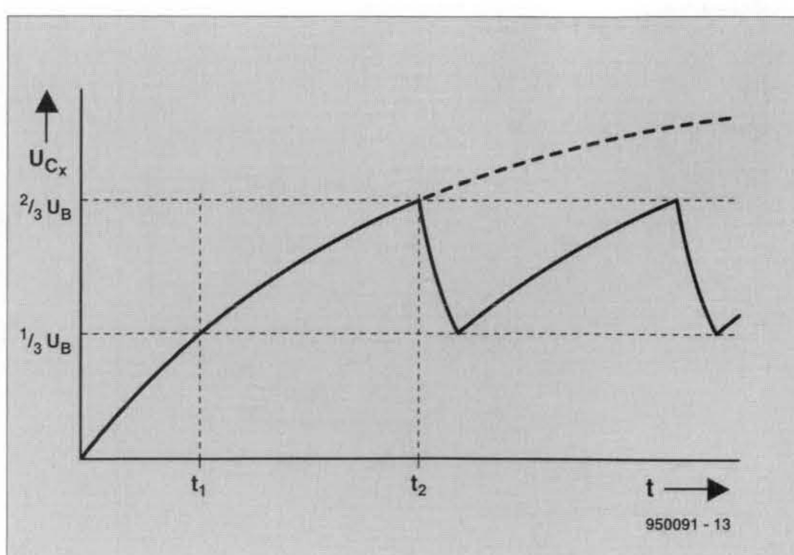


Fig. 2. Voltage across a capacitance vs time when the capacitance is being charged by a constant current via a resistance.

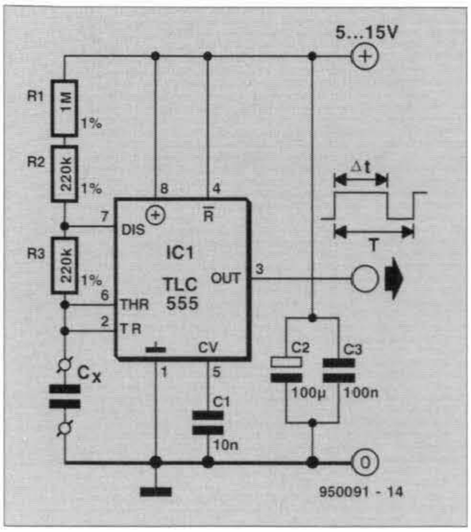


Fig. 3. A simple circuit, based on a timer IC, for accurately measuring capacitance.

through leakages currents in electrolytic capacitors. These should, therefore, be charged with larger current, so that the leakage current becomes relatively smaller. The charging current can be increased a thousandfold, but the values of the three resistors should then be changed to $R_1 = 1\ k\Omega$, and $R_2 = R_3 = 220\ \Omega$ in order to retain $\Delta t/C_X = 1\ s\ \mu F^{-1}$.

To avoid the input currents of the IC affecting the measurements, the CMOS version of the device should be used.

When a standard 555 is used, the best accuracy that can be achieved is 2% (owing to the tolerances of the voltage divider in the IC and resistors R_1–R_3, this cannot be bettered). Normally, owing to the temperature coefficients of the various components, the accuracy will be rather worse

Greater accuracy

Greater accuracy cannot really be achieved with the circuit in Fig. 3. If, however, calibration is acceptable (for which a $4^1/_2$ digit digital multimeter is required), the circuit in **Fig. 4** should be used.

When small capacitances are measured with the circuit in Fig. 3, not only does the pulse width Δt get shorter, but also the period T. Because of this, it becomes more and more difficult to read the (flickering) values indicated by the frequency meter.

With some frequency meters, there is the added difficulty that the internal timing becomes unreliable with (very) short pulse spacings. This is, to some extent, alleviated by network R_{12}-C_7. This combination makes it impossible for bistable N_2-N_3 to be retriggered by A_2 before its time constant has elapsed. The indicated values, even with small capacitances, are then shown with larger spacings. The output-pulse duration is lengthened by the time required to recharge C_X to $U_B/3$. Gate N_4 ensures that the output signal remains correct. To be sure, the gate reverses the polarity of the output signal, so that in this circuit the pulse spacing represents the value of capacitance. It is, of course, possible to connect an inverter in series with N_4 to revert to the pulse width representing the capacitance.

Gate N_4 is also of benefit when larger capacitances are being measured. In that case, however, the first indicated measurement result is correct, since the output is then low as long as U_X is between the two voltage thresholds.

Network P_7-P_8-C_6 serves to compensate (negate) the short pulses at the output caused by inevitable parasitic capacitances, which are present even when the test terminals are open. In the prototype, they had a duration of about 42 µs in the most sensitive range (1 ms nF^{-1}), which is equivalent to 42 pF. The compensating network delays the trailing edge of the output pulses of A_2 by this time. That the leading edges are also delayed does not matter.

In the 1 ms μF^{-1} measurement range, the delay causes a parasitic capacitance of up to 42 nF. Although this value is not normally reached in practice, C_1 ensures that it is. This makes the circuit not only more accu-

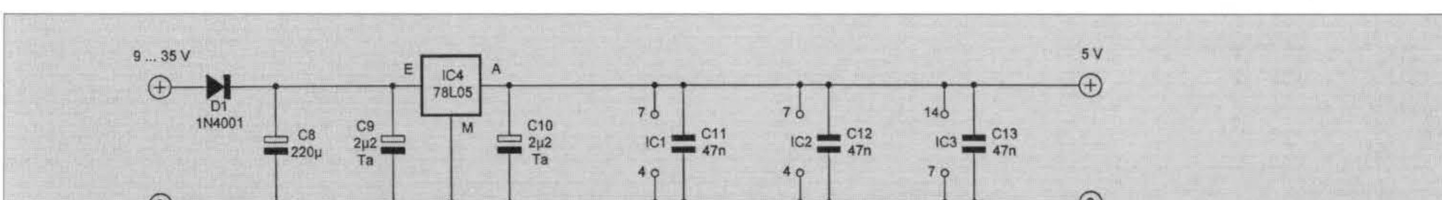


Fig. 4. Circuit for more accurate capacitance measurements; it is, however, more expensive than that in Fig. 3.

Error calculation

The error in converting pulse period to capacitance can be calculated from the measured values of the time-determining resistors and the error of the digital multimeter used. It should be assumed that the desired values at the inputs of the operational amplifiers are not exactly $U_B/3$ and $2U_B/3$. Any offset voltages of the op amps remaining after calibration will be ignored.

If the three voltages measured under 'Calibration' are designated U_1, U_2, and U_3, then

$$U_1 + U_2 + U_3 = U_B. \quad [9]$$

Then, from Eq. [3] and [4]:

$$U_X(t_1) = U_1, \quad [10]$$

and

$$U_X(t_2) = U_1 + U_2. \quad [11]$$

Substituting these with the aid of Eq. 2 into Eq. 5 and Eq. 6 gives

$$\exp(-t_1/RC_X) = 1-U_1/U_B = t_1/RC_X$$
$$= -\ln(1-U_1/U_B = \ln\{U_B/(U_2+U_3)\}, \quad [12]$$

and

$$\exp(-t_2/RC_X) = 1-(U_1+U_2)/U_B = t_2/RC_X$$
$$= -\ln\{1-(U_1+U_2)/U_B\} = \ln(U_B/U_3). \quad [13]$$

If the pulse period, $t = t_2-t_1$, Eq. 7 becomes

$$\Delta t/RC_X = \ln(U_B/U_3) - \ln(\{U_B/(U_2+U_3)\} = \ln\{(U_2+U_3)/U_3\}. \quad [14]$$

The result is independent of U_1, so that this need not be rechecked during calibration. The value in which we are interested is the variation of $\Delta t/C_X$, that is, $\delta(\Delta t/C_X)$, related to δU_2, δU_3, and δR. The complete differential equation of $\Delta t/C_X$ is:

$$d(\Delta t/C_X) = dR \cdot \ln(U_2/U_3+1) + dU_2 \cdot R/(U_2+U_3)$$
$$+ dU_3 \cdot (-RU_2)/\{(U_2+U_3)U_3\}. \quad [15]$$

To calculate the error, the terms at the righthand side of Eq. [15] should be considered on their own (worst case), so that in the last term the sign changes. Thus, and taking Eq. 14 also into account, the relative variation of $\Delta t/C_X$ is:

$$\delta(\Delta t/C_X)/(\Delta t/C_X) = \delta\Delta t/\Delta t$$
$$= \delta R/R + 1/\ln\{(U_2+U_3)U_3\} [\delta U_2/(U_2+U_3)$$
$$+ \delta U_3 \cdot U_2/\{U_3(U_2+U_3)\}]. \quad [16]$$

The desired voltage levels are:

$$U_1 = U_2 = U_3 = U_B/3. \quad [17]$$

Since these are measured with the same instrument in the same range,

$$\delta U_1 = \delta U_2 = \delta U_3. \quad [18]$$

Therefore,

$$\delta\Delta t/\Delta t = \delta R + \delta U/(U \ln 2). \quad [19]$$

This equation gives the worst case; the probable error is almost always smaller:

$$\delta\Delta t/\Delta t = \{(\delta R/R)^2 + ½(\delta U/U)^2 \cdot 1/(\ln 2)^2\}^{1/2} \quad [20]$$

Assume that a 3½-digit DMM is used. The specification of this instrument states the following errors:
$\delta R/R = 0.5\% + 1$ digit (2 MΩ range);
$\delta U/U = 0.5\% + 1$ digit (2 V range).

Using these figures in Eq. 19 gives

$$\delta R/R = 0.005 + 10^3/1.443\times10^6 = 0.0057;$$
$$\delta U/U = 0.005 + 10^{-3}/1.67 = 0.0056.$$

From these figures, the accuracy of the calibrated capacitance meter is:

$$\delta\Delta t/\Delta t = 0.0057 + 0.0056/0.69 = 1.4\%; \quad \text{[worst case]}$$

or,

$$\delta\Delta t/\Delta t = 0.81\%. \quad \text{(probable error)}$$

The specification of a 4½-digit DMM shows that

$\delta\Delta R/R = 0.15\% + 3$ digits (2 MΩ range);
$\delta U/U = 0.05\% + 3$ digits (2 V range).

As before,:

$$\delta R/R = 0.0015 + 300/1.4427\times10^6 = 0.0017;$$
$$\delta U/U = 0.0005 + 300\times10^{-6}/1.67 = 0.00068.$$

From these figures, the accuracy of the calibrated capacitance meter is:

$$\delta\Delta t/\Delta t = 0.0017 + 0.00068/0.69 = 0.27\% \quad \text{[worst case]}$$

or

$$\delta\Delta t/\Delta t = 0.18\% \quad \text{[probable error]}$$

In practice, it should be borne in mind that the frequency meter has an additional error, normally 1 digit for the final displayed cipher.

rate, but also more user-friendly.

Construction

The supply voltage was chosen at 5 V. This level is dictated by the calibration voltage: P_3 and P_4 should be set so that the supply voltage is three-parted exactly. A voltage is measured accurately when it lies just below a range limit of the measuring instrument. Owing to digit errors, this is also the case with digital multimeters. The best measuring range is then 2 V, so that the maximum supply voltage should be just below 6 V. Because of this, the voltage regulator is a Type 7805. The voltage source is a 9 V battery (dry or rechargeable), or mains voltage adaptor. In case of battery supply, protection diode D_1 should be replaced by a supply on/off switch.

Most of the resistors should be metallized film types with a tolerance of 1% or better. The preset potentiometers should be multiturn cermet types. Switch S_1, connecting block K_1 and potentiometer P_7 are, of course, best mounted on the front panel.

The Type TL081 operational amplifiers are fairly fast in spite of the low supply voltage (and they are less expensive than low-voltage types

CA3140 and TLC251).

Do not solder C_1 into place until after the calibration.

The test terminals are made from a 10-way SIL-IC socket of which contacts 1-5 and 6-10 are interlinked. This makes it possible for different sizes of capacitor to be measured.

The output socket is a standard BNC type.

Points A–F, as well as 0 V and 5 V, should be soldered on to solder pins to make them accessible for testing.

Calibration

The circuit has seven calibrating points. A short cable terminated into small sockets that fit over solder pins is required.

Set P_7 and P_8 to minimum resistance, and all other potentiometers to the centre of their travel.

With S_1 in position 1 ms/nF, adjust P_1 to give a total resistance for $R_1+R_2+R_3+P_1$ of 1.4427 MΩ.

Change over S_1 and adjust P_2 to give a total resistance for $R_4+R_5+R_6+P_2$ of 1.4427 kΩ.

After switching on the supply voltage, measure with a digital voltmeter set to 2.0 V the voltages at B w.r.t. ground, at B w.r.t. A, and at A w.r.t. U_B and add them. This gives a more accurate level of U_B than measuring it directly Divide the value so found by 3, which gives the desired level of the potentials at the inputs of the op amps. Adjust P_3 to give this level at A w.r.t. U_B, and then adjust P_4 to give this same level at B w.r.t. A.

Link A to C (A_1+ to A_1–) and connect the digital multimeter (2.0 V range) between D (output of A_1) and earth. Adjust P_5 to exactly $U_B/2$.

Link B to C (A_2+ to A_2–) and connect the digital multimeter (2.0 V range) between E (output of A_2) and earth. Adjust P_6 to exactly $U_B/2$.

Connect a frequency meter set to the time measurement mode to the output of the circuit. With the test terminals open, and S_1 in the 1 ms/nF position, the meter should show a pulse spacing of some tens of microseconds. This represents on the one hand parasitic capacitances (pF) and on the other, the value of C_1.

Set P_7 to the centre of its travel and adjust P_8 until the frequency meter reads zero. Next, solder a capacitor in the C_1 position whose value is nearest to the average measured value. This need not be exact, since P_7 provides a compensation of about ±10 μs. To this end, every time the supply is switched on or the measuring range is changed, P_7 must be set to minimum resistance and then adjusted until the frequency meter reads zero. If the preset is turned too far, it becomes impossible to read small values in either range of S_1.

Make sure that any capacitor to be measured is fully discharged before it is connected to the test terminals. This is particularly important in the case of electrolytic capacitors whose polarity should also be taken into account during measurements. Although no electrolytic capacitor will be damaged by the 3.3 V test voltage, the measurement will result in a wrong value if the polarity is incorrect.

[960004]

MICROCONTROLLER SWITCHING CLOCK RTC56

2nd Prize (G)

Design by H. Schaefer

This Competition entry describes a microcontroller-driven switching clock (housekeeper) with a 2x16-character LC display, input keys and two relay outputs. The clock has a built-in power supply and is contained in a compact plastic box (see photograph). The clock is capable of memorizing a program of up to 56 switching instants, which may be distributed over a full week. The programming memory may also be subdivided into multiple switching sequences.

THE actual clock is not implemented in software, but by a dedicated integrated circuit called a real-time clock (RTC). This quartz-controlled RTC IC sports a century calendar, automatic summer/winter time adjustment and an alarm function. It also features battery backup by two Lithium cells so that the clock continues to operate, and the switching program is retained, when the mains voltage disappears. The clock is controlled by a microprocesscr type 8751H, which contains about 2,200 bytes of EPROM code.

The clock has four modes of operation: in SET mode, it can be adjusted to synchronize the switching program. In RUN mode, the clock displays the time using the format 'hh:mm:ss', plus the day of the week, the date, the number of the valid switching program step, and the state of the two relays. In PROG mode, you may enter and change the switching times. In CHECK mode, finally, you can view the contents of the program memory quickly and easily without affecting the program.

The switching clock is operated with three push-buttons, according to a simple set of rules.

Circuit description

The circuit diagram (**Fig. 1**) shows a bare-bones design which consists essentially of IC1 and IC2. The heart of the circuit is an 8751H microcontroller, which is a member of Intel's MCS-51 family. Note the absence of the usual address latch. Normally, it is necessary to access external RAM or EPROM. Here, however, these external parts are not used, and the program is contained in the 8751's internal 4-kByte EPROM. Also on the chip is an 128-byte RAM in the form of a register bank which also allows bit access. The internal EPROM and RAM obviate the need for any external memory components. Another advantage is that Port 0 and Port 2 are freed up for other purposes. All in all, the 8751 offers exactly the number of

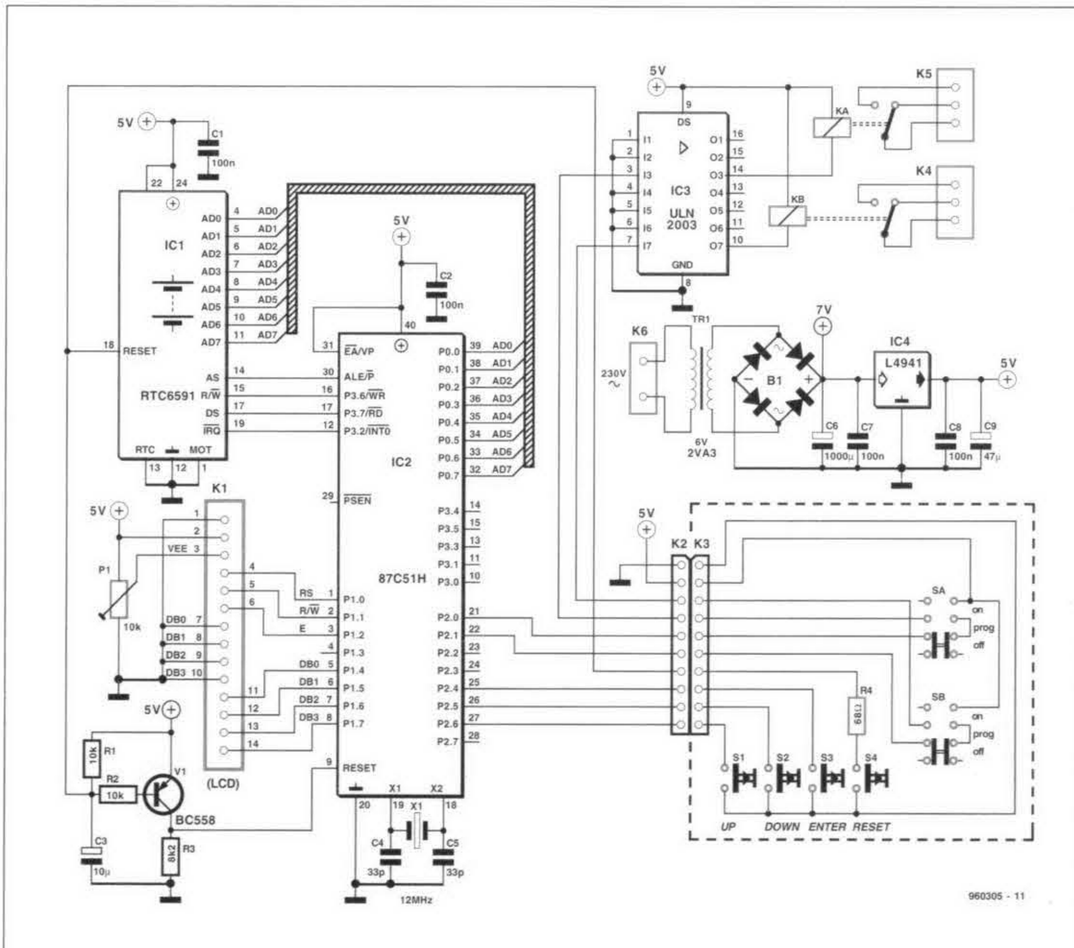


Fig. 1. Circuit diagram of the RTC56 microcontroller-driven switching clock.

ports actually required by the circuit: Port 1 is linked to the LC display, while Port 0 handles the data exchange with the real-time clock contained in IC1. Port 2, finally, serves to read the presskeys, and control the relays.

The LCD normally requires 14 drive lines. However, because it is operated in 4-bit mode here, interface lines DB0 through DB3 are not required, so that one port is sufficient.

The actual clock is contained in IC1, and has an internal RAM of 128 bytes. The memory map of the clock is given in **Fig. 2**. The range from 0EH to 7FH is free. Here, this range is loaded with the switching instants to be programmed. Every alarm data set consists of two bytes:

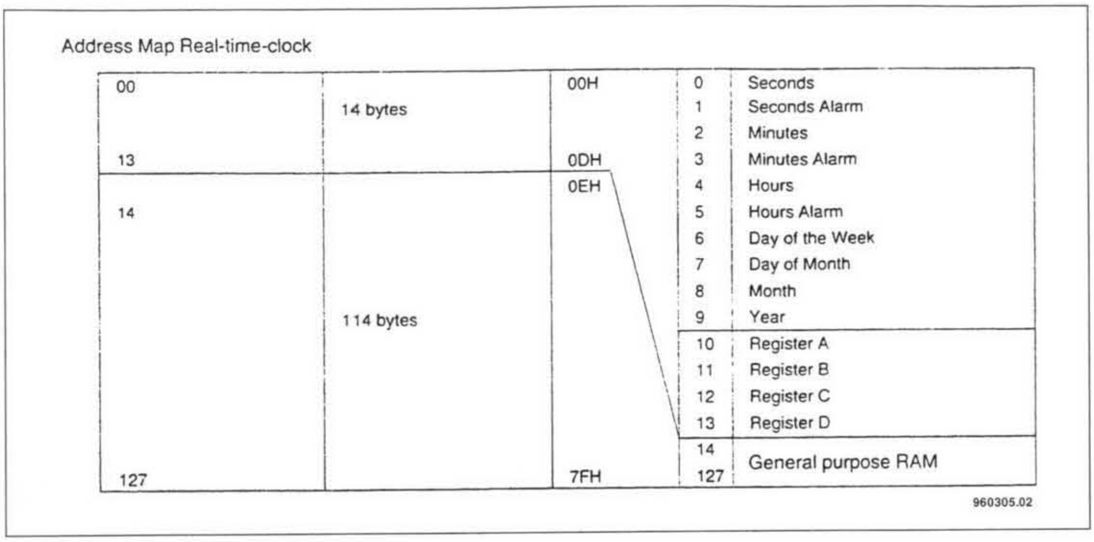


Fig. 2. Memory map of the real-time clock module.

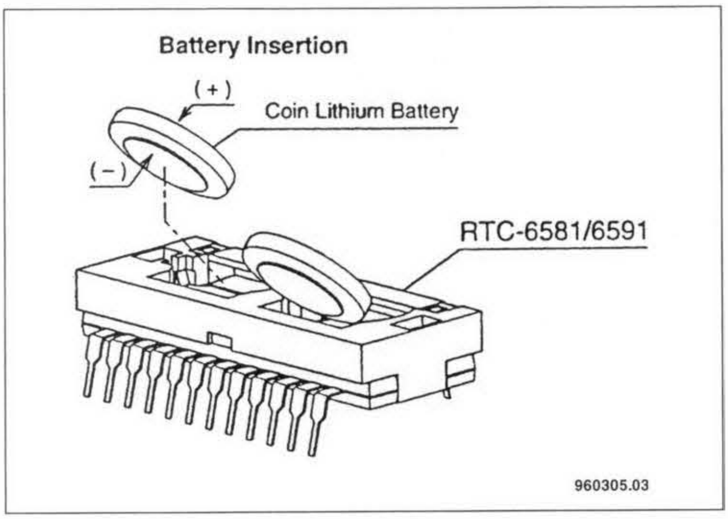


Fig. 3. The RTC chip has a piggy-back battery holder for two Lithium button cells.

1. Alarm byte wwwhhhhhh
   Day of the week/hours
2. Alarm byte abmmmmm
   Relay/minutes

Each alarm data set has a number which is shown on the LCD. This number is the address of the first alarm byte minus 14. In this way, 56 alarm instants may be programmed. The last two bytes in the memory have special functions. The current alarm number is found at address 7FH. It is the number of the alarm contained in the alarm registers of the RTC at that particular time.

At 7EH there is also an alarm number. This is the one which is current in the PROG and CHECK modes.

The RTC IC has a piggy-back battery holder for two Lithium button cells type BR125, which guarantee program retention and clock operation for more than ¾ of a year after the regular supply voltage has disappeared. IC1 also has an Intel Bus interface, so that the controller is able to control the clock using a 'MOVX' command, just as an external RAM.

Port 2 drives the relays, and reads the presskeys which serve to program and operate the clock. All control elements are gathered on a keyboard PCB, which is stacked on to the main board. Slide switches SA and SB allow the relays to be switched on and off independently of Port 2. This has proved to be a very useful feature in the practical use

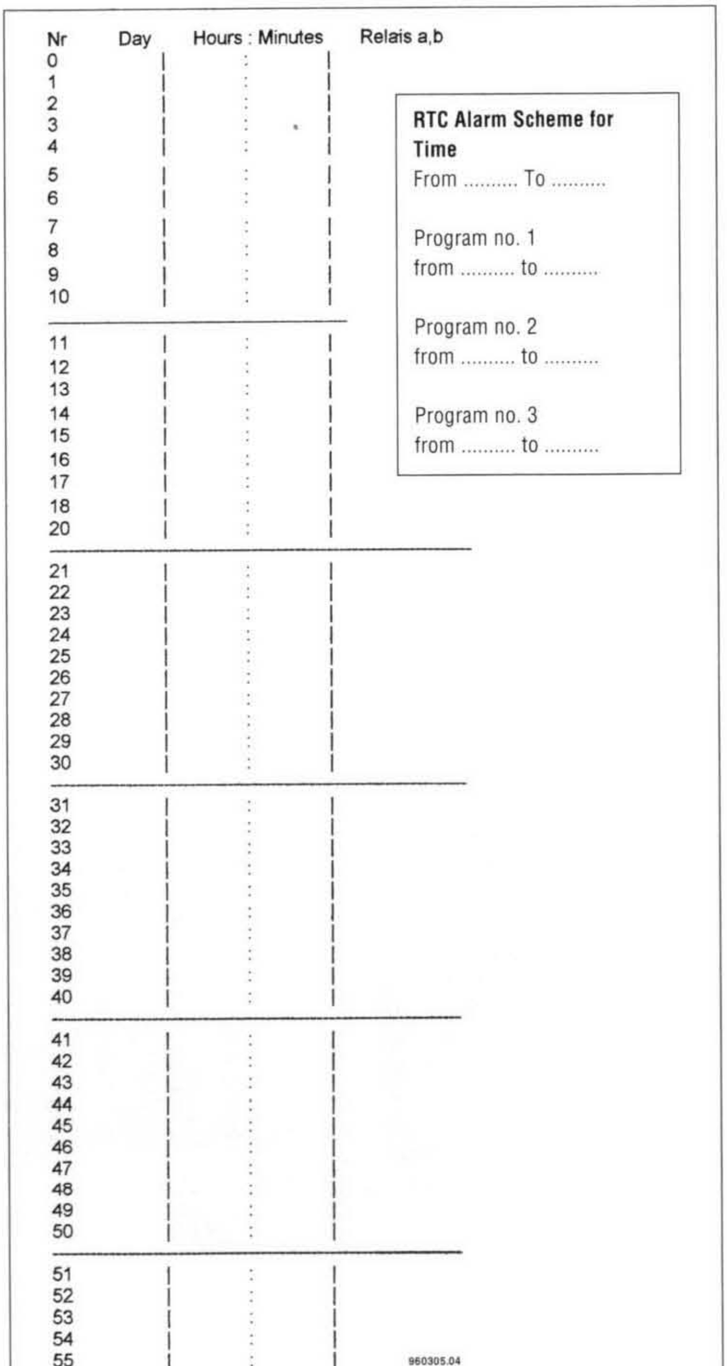


Fig. 4. Before you start programming, make a list of dates and times pertaining to the individual alarm instants. Assign a consecutive number to each alarm instant.

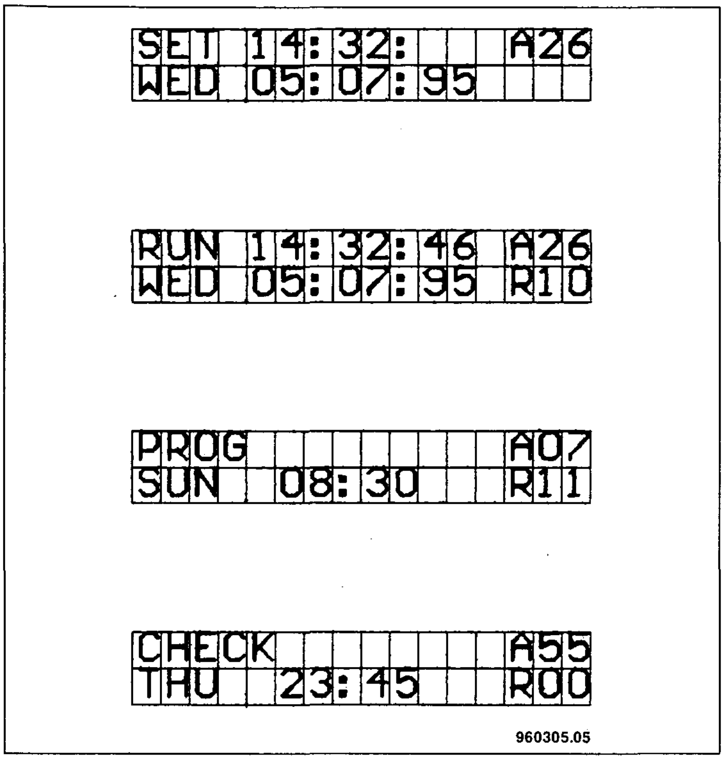


Fig. 5. Display readouts in each of the four modes.

of the clock. The relay coils are driven via IC3, a Darlington transistor array.

Components R1 and C3 provide a RESET when the clock is switched on. Because the controller reset has a different active level than the clock reset, the RESET signal is inverted by transistor V1. The RESET presskey enables the circuit to be re-initialized at any time.

The power supply section is conventional. The circuit requires a current of about 320 mA. To keep the transformer as small as possible, a 6-V type was chosen. Because the transformer is not fully loaded, it supplies about 7 V. A low-drop voltage regulator then supplies a stable 5 V rail.

Operating the RTC56

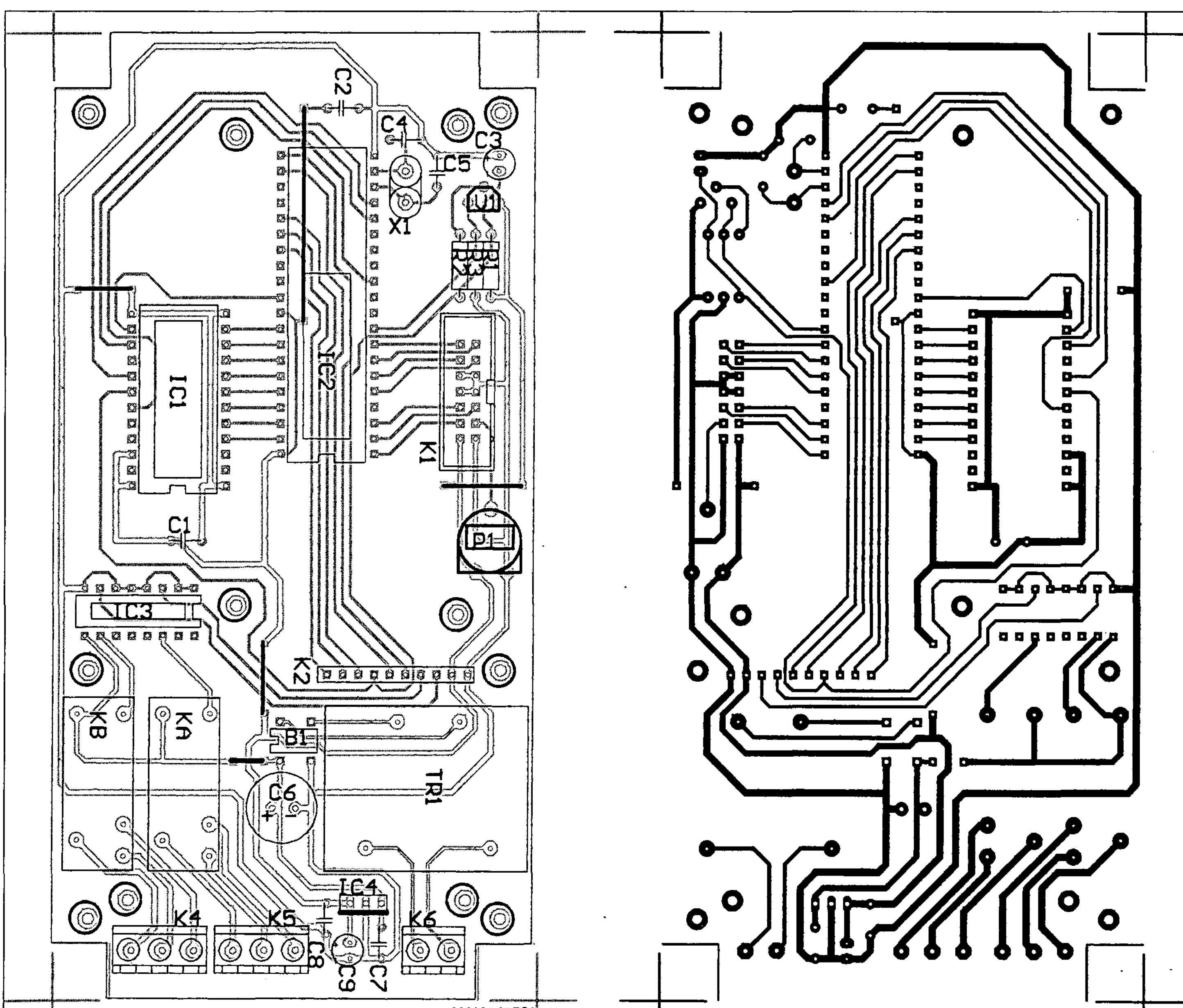
Clear memory

When the RTC56 is first used, the clock registers must be cleared. Keep the UP and ENTER buttons pressed, and then press the RESET button. Release it again, but keep the other two pressed until 'SET' appears on the display. This clearing does not affect the program memory.

Adjusting the values

The cursor on the LCD is an underscore, (_). It tells you what can be changed by pressing the yellow UP key, or the green DOWN key. The cursor may be moved to the right by pressing the red ENTER key. At the end of the

Fig. 6. Track layout and component mounting plan of the single-sided printed circuit board.



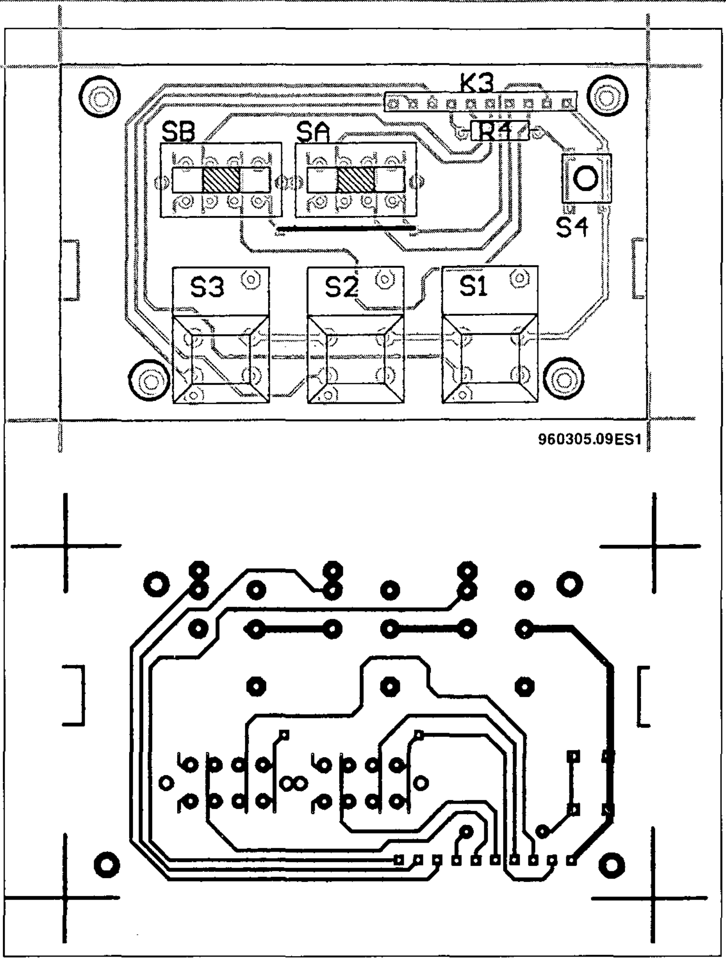


Fig. 7. Track layout and component mounting plan of the keyboard PCB.

COMPONENTS LIST

Resistors:
R1;R2 = 10kΩ
R3 = 8kΩ2
R4 = 68Ω
P1 = 10kΩ preset

Capacitors:
C1;C2;C7;C8 = 100nF
C3 = 10µF 16V
C4;C5 = 33pF
C6 = 1000µF 25V
C9 = 47µF 16V

Semiconductors:
V1 = BC558
IC1 = RTC 6591
IC2 = 8751 (programmed)
IC3 = ULN2003
IC4 = L4941 (5V low-drop)

Miscellaneous:
X1 = 12MHz crystal, HC18
Tr1 = 6V 2.3VA mains transformer
S1;S2;S3 = Digitast presskey, 1 c/o
S4 = miniature presskey
KA;KB = relay, 6V/80Ω, 1 c/o
SA;SB = 3-position slide switch
K1 = 14-way boxheader
K2 = PCB connector, 10-way, socket
K3 = PCB connector, 10-way, header
K4;K5 = 3-way PCB terminal block
K6 = 2-way PCB terminal block
LCD display, 2x16 characters, e.g., LTN211R or LM16255
ABS case, 160x82x52mm

top line you wrap around automatically to the start of the second line. The same for the last position of the second line, which wraps around to the start of the top line again.

Modes of operation

SET

When the RTC56 is first switched on, or after a reset, the clock enters the SET mode after a few seconds. This mode is used to adjust the clock. Press ENTER to enter this mode. First the cursor is at the tens of hours positions on the display. Adjust to the desired value by pressing UP or DOWN. Press ENTER to go to the hours indication, and adjust the clock similarly. After adjusting the minutes indication, you arrive at the alarm number next to A. Here you enter a number (0 to 55) which indicates the number of the current switching times data set from the alarm program. Such a data set is henceforth called 'alarm'. Entering the alarm number causes the program to be synchronized with the clock.

Pressing the ENTER key takes you to the start of the second line, to the day of the week. Press UP/DOWN to set the current day of the week. The remaining cursor positions are reserved for the date. This is entered in the format dd:mm:yy, for example, 09:01:96 for 9 January 1996. After entering the year number, ENTER takes you back to the start of the SET mode, and the clock is enabled again internally. Press DOWN to move from SET to RUN mode.

RUN

In this mode, the clock shows the time with an hours, minutes and seconds indication (hh:mm:ss). Also shown are the current alarm number and the state of the two relays. A '1' means that the relevant relay is energized. A '0' means that the relay is off.

The clock also indicates the day of the week and the date. As long as the cursor is below the word 'RUN', the clock runs internally, but the display indication is decoupled. Use the ENTER key to move the cursor in front of the 'R' to make the display 'tick' along with the clock.

Press ENTER if you want to leave the RUN mode. The cursor moves to the start of the first display line. Pressing UP takes you to SET mode, while DOWN takes you to PROG mode.

PROG

This mode enables you to program the switching times used by the clock. The clock is capable of storing 56 switching instants for the control of the relays. These instants may be distributed throughout the week. It is also possible to enter multiple switching programs.

A switching program is performed step-by-step as a cycle, so it is must be coherent. If a program is located somewhere in the middle of the memory between 00 and 55, a dummy alarm with day of the week 'xxx' must be programmed before the first alarm and after the last alarm. If a program ends at location 55, the dummy alarm is not necessary. The same applies when a program starts at location 00, which should normally be the case.

Before you start programming the clock, make a list of data which applies to the individual alarm instants you require. An alarm data set consists of an alarm number, day of the week, switching time hh:mm (no seconds), and relay states. The programmed relay states remain valid until the programmed alarm time is reached.

After entering an alarm data set, you are returned to the start of PROG mode. Press ENTER to enter the next alarm set. Repeat this sequence until you have entered the entire program. If you want to view the program data, change from PROG to CHECK mode by pressing the DOWN button.

CHECK

This mode allows you to get a quick overview of the program memory contents, without running the risk of destroying the program. Pressing ENTER takes you into CHECK mode. The cursor is at the tens position of the alarm number. Use the UP/DOWN keys to change the number in steps of 10. You will see the associated alarm data immediately. When you attempt to exceed the memory range, the indication changes to 00 or 55.

Press ENTER to move to the units (0-9) of the alarm number, and use the UP/DOWN controls to select the desired number in steps of 1. Pressing ENTER again takes you back from the alarm number units to the start of the CHECK mode. Next, pressing UP or DOWN allows you to change the mode of operation. (960305)

Note:

The control software to be programmed into the 8751H controller is available on disk through our Readers Services (see page 70). This software has not been tested by *Elektor Electronics*.

HYBRID HEADPHONES AMPLIFIER

2nd Prize (NL)

Design by W.A. van Pelt

If you are after great sound at an affordable price, go out and buy a pair of high-quality headphones. Next, you will need a dedicated headphones amplifier. The one described here combines transistor and valve technology. The combination results in an interesting kind of technological 'synergy', or mutual reinforcement of the strong points of transistors and valves. A further remarkable aspect of the amplifier is that it has no output transformer.

Input stage

The remarkable thing about the input stage is the anode load of the ECC83. This load is a current source built around transistors T1 and T2. The output impedance of the transistor circuit is about 2 MΩ. Among the advantages of the current source is the excellent rejection of hum and noise in the anode circuit. Also, the valve gain rises to almost the maximum achievable value, μ (which equals about 100 for the ECC83). In other words, the gain of the input stage is determined by one valve parameter only: μ, which happens to be one of the most constant parameters, and so helps to improve the linearity. Although the cathode voltage remains constant because of the constant anode current, capacitors C1 and C2 are still present. These caps serve to eliminate the noise produced by R2, and drop the output impedance of the input stage from 120 kΩ to about 30 kΩ. Operating two ECC83 halves in parallel also lowers the output impedance, although to a lesser extent.

Power stage

The power output stage is biased such that sufficient current and voltage are available to make a pair of 600-Ω headphones 'deafening'. The value of C5 is such that the LF roll-off frequency is still about 2 Hz if 32-Ω headphones are used. Together with the 1-Hz roll-off point caused by coupling capacitor C3, the phase shift at better audible frequencies is negligible. Because the screen voltage has a much larger effect on the anode current than the anode voltage itself, it is held constant by a separate filter. Without this filter, the anode voltage would require much heavier filtering.

Filament current

Each valve has its own filament current source. Thanks to these current sources, it is impossible for high current surges to occur when the filaments are 'cold'. The gradual, controlled, heating up helps to lengthen the lifetime of the valves. The pots are adjusted such that the filament voltages are exactly 6.3 V.

Construction

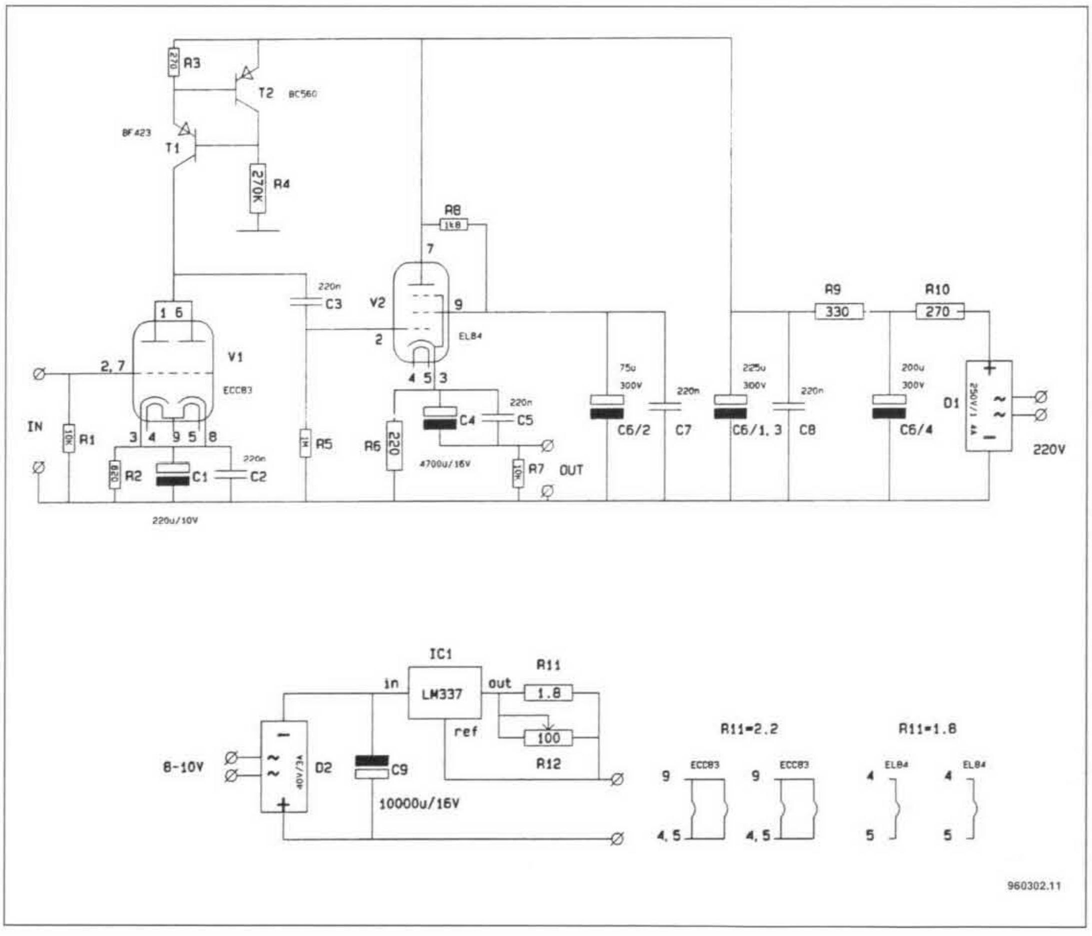
The motherboard is double-sided. The filament current flows at the component side, and arrives from the supply board via a length of flatcable. The LM337s may be mounted without isolation and, if necessary, all on a common piece of metal. Resistor R1 may be omitted if a volume pot is used at the input. To enable the constructor to build the amplifier into the case of his/her liking, no provision is made on the PCB for input or output sockets.

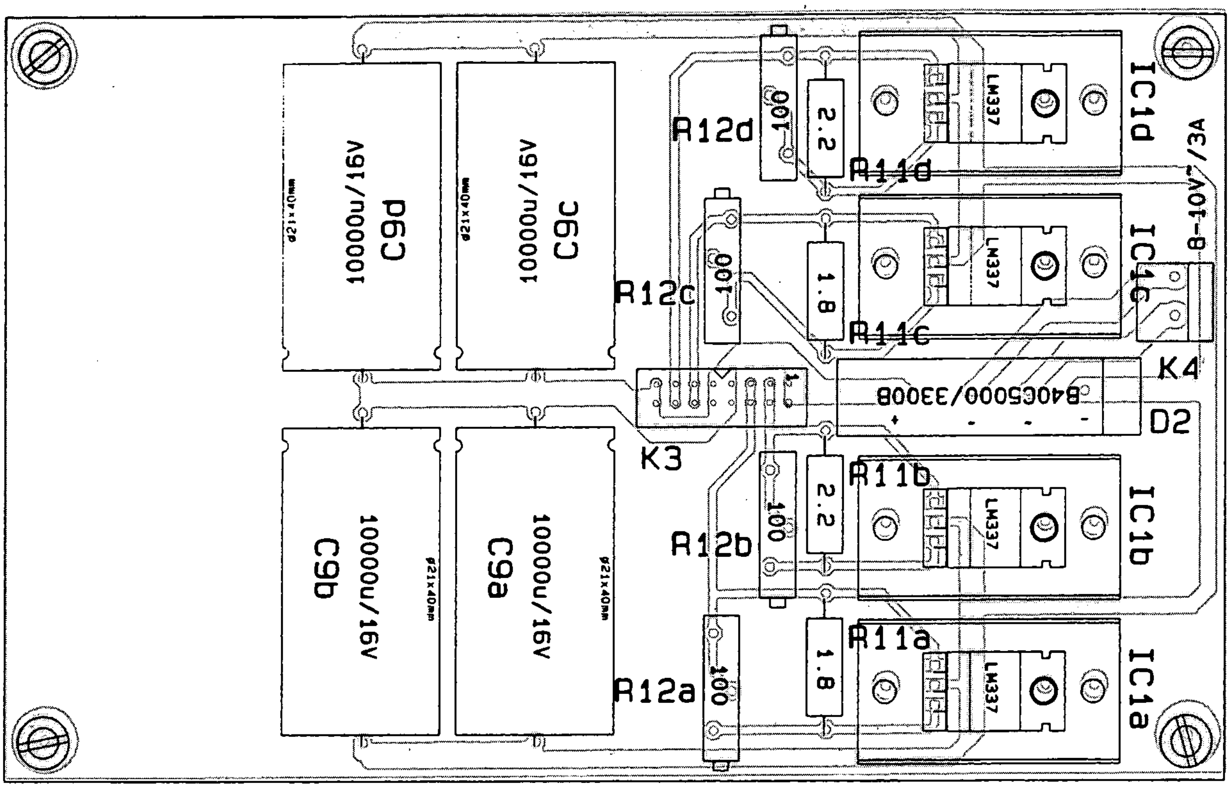
A lot of board space is used by the 'FKP' capacitors which guarantee superb pulse response. Use high-quality components: metal film resistors and valve sockets with good pin clamps.

The amplifier will also work with, for example, an ECC81 or ECC90 instead of an ECC83. These substitutes will reduce the gain somewhat, but this will still be high enough. Other ECC types may also be possible, as long as the anode voltage remains between 50 V and 250 V. If necessary, small corrections may be made by changing R1 or R3.

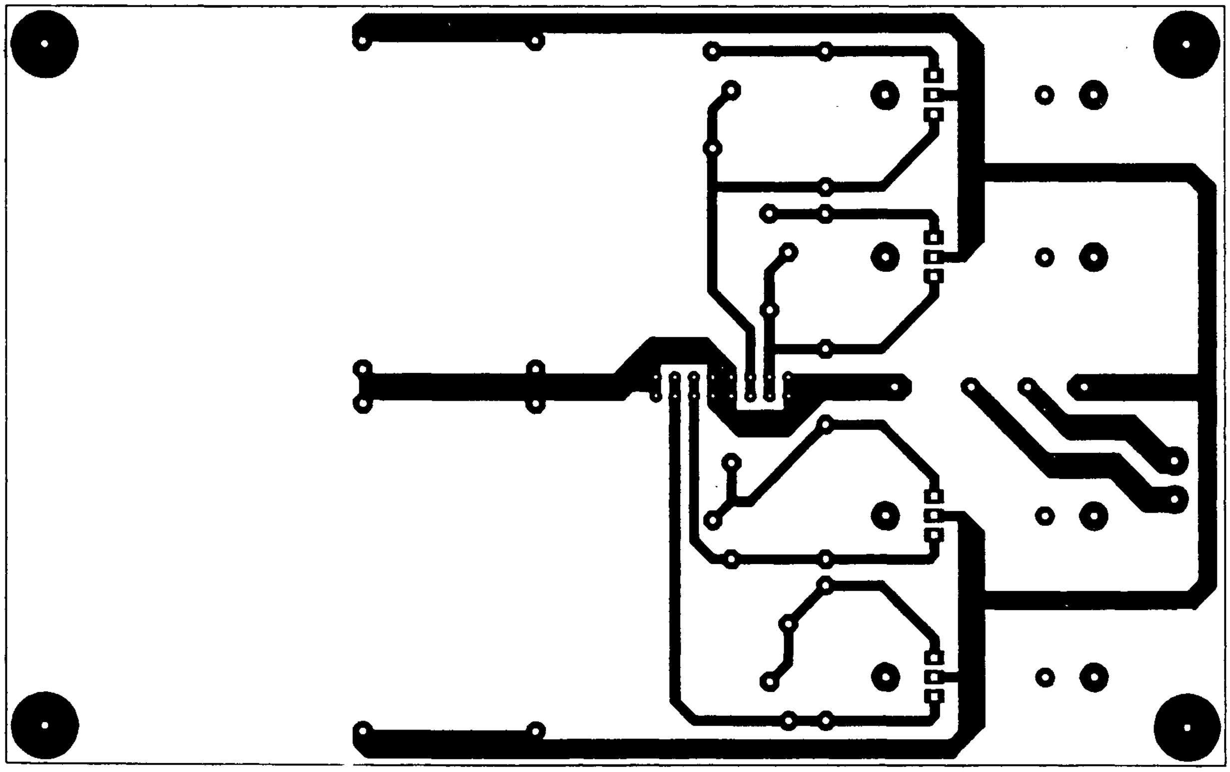
(960302)

Editor's note: The mains transformer for the high-voltage supply is not shown in the circuit diagram, nor is it included in the parts list. This component is, however, needed for the amplifier.





960302.ES1



COMPONENTS LIST
(for each channel)

Resistors:
R1 = 10kΩ
R2 = 820Ω
R3 = 270Ω
R4 = 270kΩ 0.5W
R5 = 1MΩ
R6 = 220Ω 0.5W
R7 = 10kΩ
R8 = 1kΩ8
R9 = 330Ω 2W
R10 = 270Ω 2W

Capacitors:
C1 = 220μF 10V
C2;C3;C5;C7;C8 = 220nF 400V FKP (Wima)
C4 = 4700μF 16V radial
C6/1,3 = 200μF+25μF 300V (NSF)
C6/2 = 75μF 300V (NSF)
C6/4 = 200μF 300V (NSF)

Semiconductors:
T1 = BF423
T2 = BC560
D1 = B250C1500/1000 600V

Valves:
V1 = ECC83
V2 = EL84

Connectors:
K1 = 16-pin boxheader
K2 = PCB terminal block, pitch 5mm

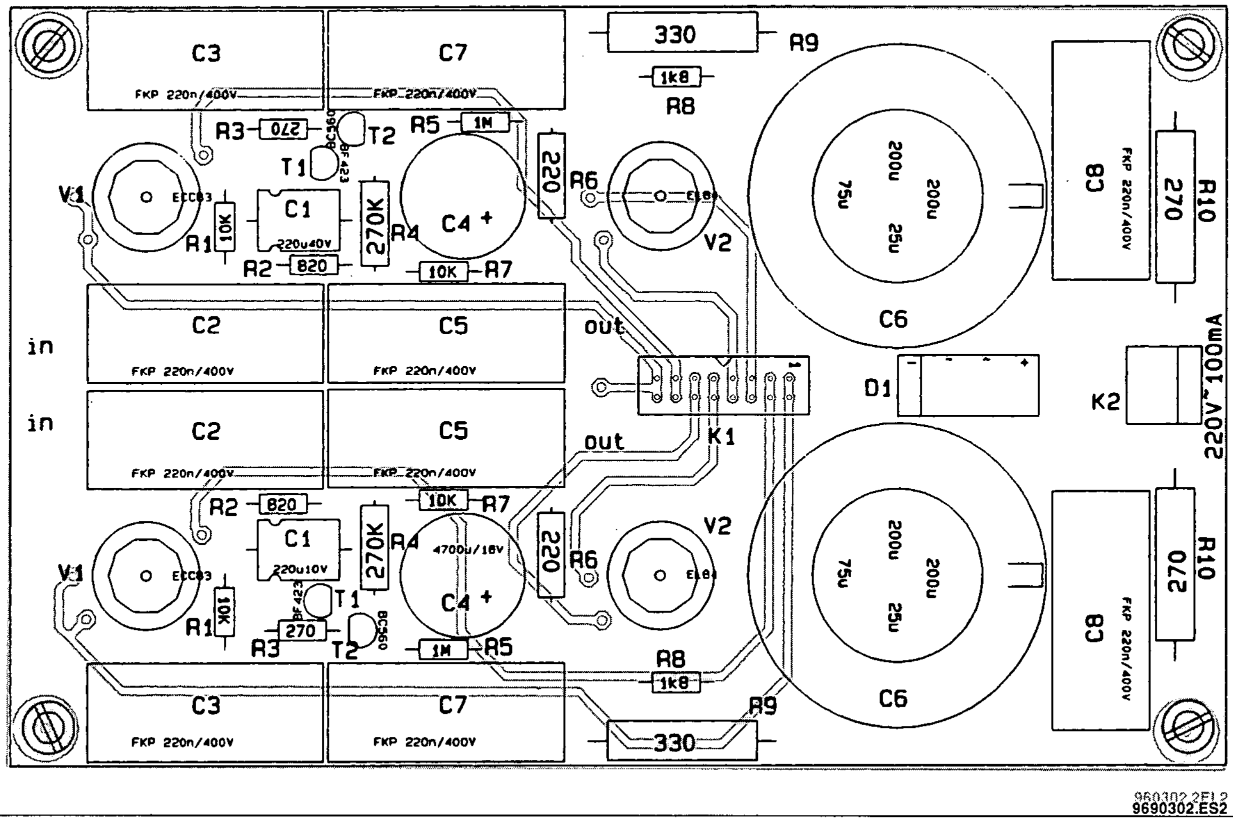
POWER SUPPLY
(per section)

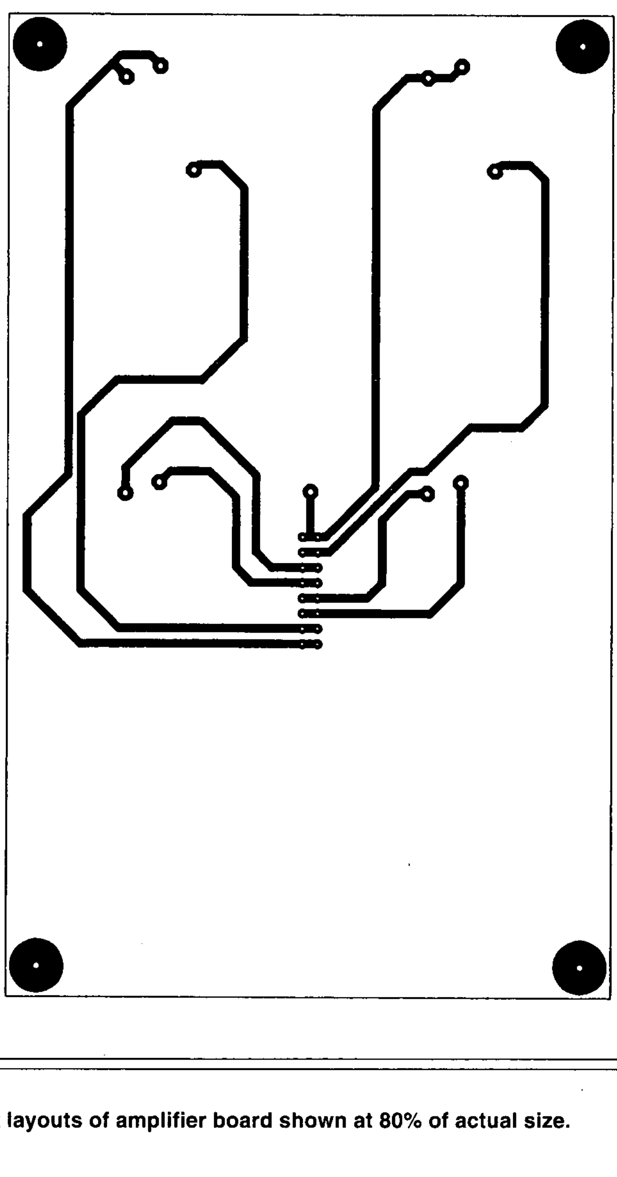
Resistors:
R11a = 1Ω8 2W
R11b = 2Ω2 2W
R12a;R12b = 100Ω preset (Bourns 3006P or Beckman 90P)

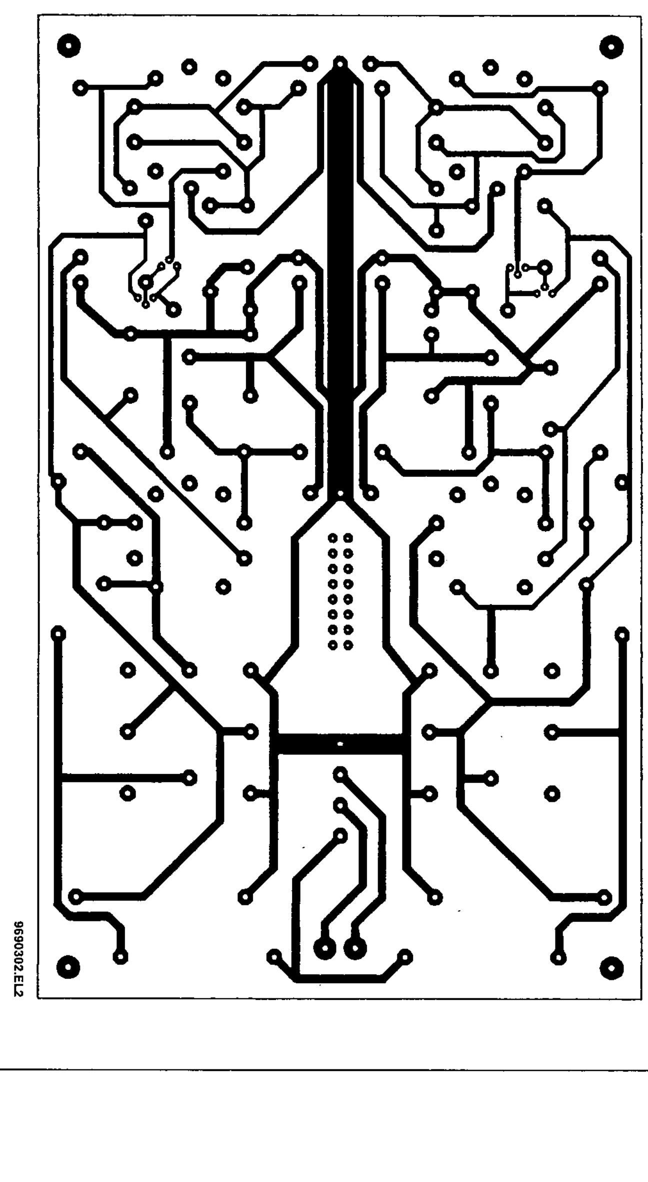
Capacitors:
C9a;C9b = 10,000μF 16V

Semiconductors:
IC1a;IC1b = LM337
D2 = B40C5000/3300 40V

Connectors:
K3 = 16-pin boxheader
K4 = PCB terminal block, raster 5mm







Track layouts of amplifier board shown at 80% of actual size.

PC-CONTROLLED BATTERY TESTER

3rd Prize (G)

Design by R. Mohrlock

This battery tester is capable of measuring the capacity of batteries up to 17 V at a maximum discharge current of 1 A. Remarkably, the tester is controlled by a PC.

After entering all data, including the nominal discharging current and the lowest battery voltage, the battery is discharged until the 'deep discharge' voltage is reached. The process is stopped automatically to prevent damage caused by deep discharging.

During the entire process, the PC display shows the battery voltage, the discharging current, the currently absorbed power and the capacity. Also indicated is the energy supplied by the battery. This is a remarkable feature of the tester, because measuring residual battery energy requires integrating power over time, and that means complex circuitry in conventional discharging units.

How it works

The heart of the circuit is a 12-bit A-D converter with an internal reference and an 8-channel multiplexer. The ADC is used to measure the battery voltage, but also as a DAC (yes!) to control the discharging current. The input multiplexer of the MAX186 is configured with the aid of an 8-bit dataword supplied by the PC.

Signals CLOCK and DATA IN are generated by the PC software via status lines on the serial interface (RS232). The rectified signals (D1 and D2) are filtered (C6), limited to about 5 V (D3), filtered once more (C5) and, finally, applied to the A-D converter. Signals CLOCK and DATA IN are applied via resistors R2 and R6.

The current consumption of the MAX186 is so small at 2 mA that a separate power supply is really unnecessary. Capacitors C1 and C2 serve to filter the reference voltage which is generated internally by the MAX186.

The battery voltage, the gate voltage of T1 and the drive voltage of the current sink (C4) are measured one after another by the control software. The current sink is essentially formed by U3 and T1. Resistor R5 drops a voltage which is proportional to the discharging current. This voltage is compared to the scaled-down (R4/R5) control voltage (on C4) by U3. U3 then controls the gate voltage of T1 such that a constant discharging current is established.

Essential in the operation of the circuit is the way in which a drive voltage appears on C4. At the far end of the control byte that configures the MAX186 is a bit which may be set to 0 or 1 without changing the function of the ADC. When the

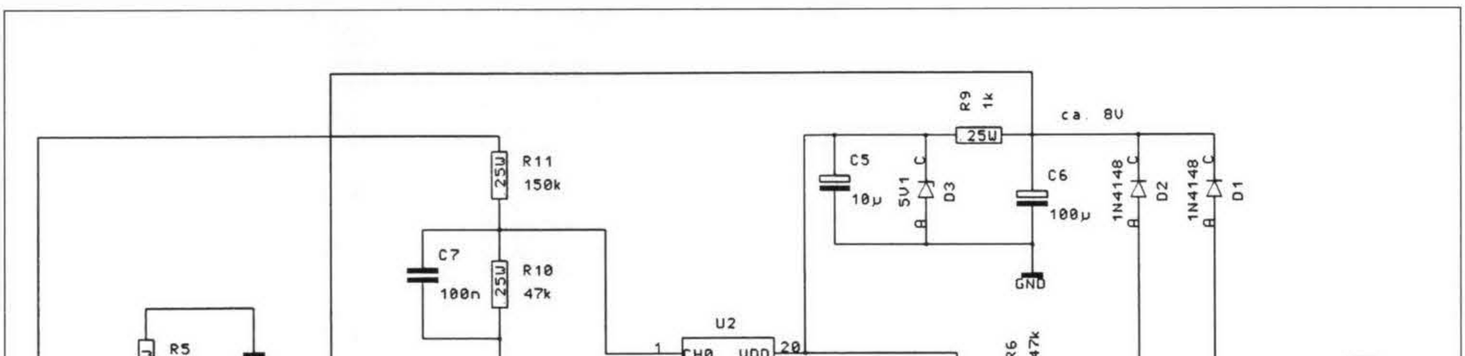


Fig. 1. Circuit diagram of the intelligent battery tester — a PC driven 12-bit analogue-to-digital converter for the voltage measurement which doubles as a digital-to-analogue converter for the control of a current sink.

control byte is read, the strobe output, pin 16, generates a 'high' pulse. This signal enables bistable U1A to pick out the last bit in the control word, and use it to charge capacitor C4 via R1. The PC software then compares the voltage on C4 (which depends on the desired discharging current) with the voltage actually measured by the ADC. Depending on the outcome of the comparison, the software sets or resets the current sink control bit. Because this bit can be set and reset several thousand times per second on modern PCs, a smooth control voltage is obtained across C4.

The main purpose of monitoring the gate voltage is to enable the circuit to detect errors. If the fuse blows (for example, after a mistake with the battery polarity), the gate voltage rises, and that can be detected by the software.

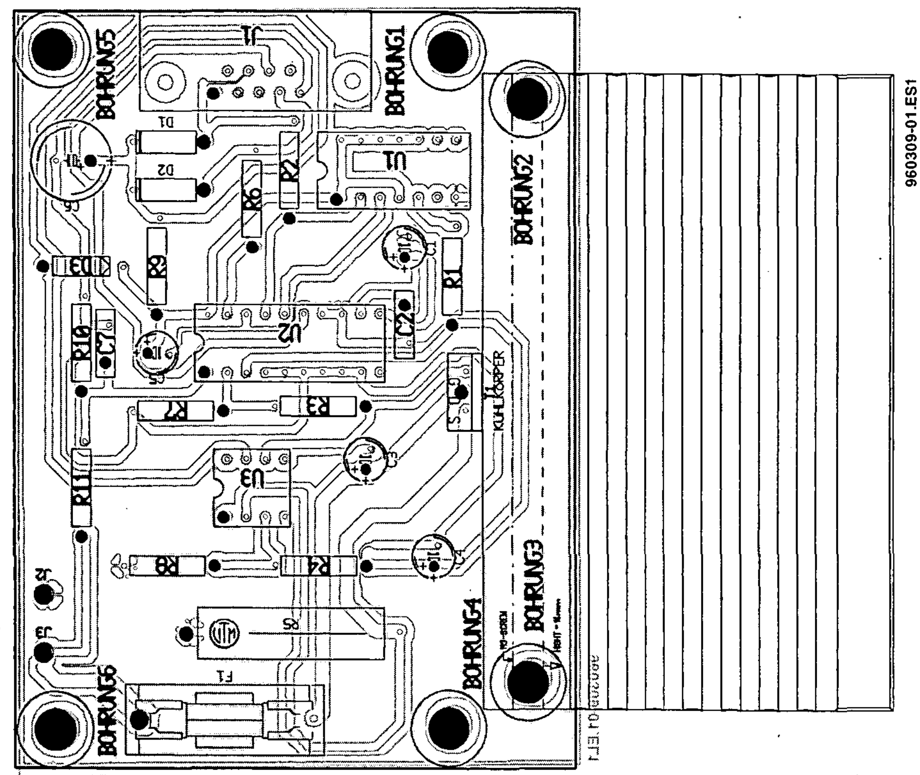
A measurement accuracy which is a few classes better than most conventional equipment may be achieved by using an opamp with a low off-set voltage is position U3, a special measurement resistor in position R5, and a separate supply in lieu of diodes D1 and D2. The stated ('regular') components, however, will do a good job, too, because they afford a measurement accuracy better than 1%.

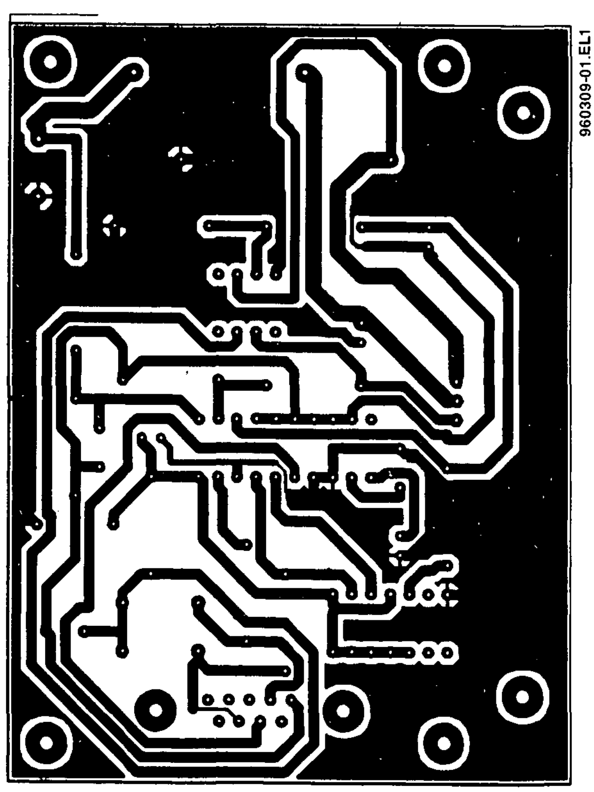
Construction

With the exception of the MAX186, the components used are uncritical and should be generally available.

The circuit is simple to build on the single-sided printed circuit board. Some extra care should be taken to make sure that solder pads are not accidentally connected to the large ground plane. Depending on your requirements as regards accuracy, the resistors should be close-tolerance metal film types.

The unit is connected to the PC via a standard 9-way RS232 cable (fully wired, pin-to-pin) purchased from a computer shop. To make sure you do not lose too much voltage, use heavy-duty cables to connect the tester to the battery. The circuit is designed such that individual cells may be measured also. At a current of 1 A and a cell voltage of 1 V, however, a voltage drop of 100 mV across the cables means an error of 10%.





COMPONENTS LIST

Resistors:
R11;R8 = 10kΩ
R2;R6;R10 = 47kΩ
R3 = 200kΩ
R4;R7 = 100kΩ
R5 = 0Ω33 5W
R9 = 1kΩ
R11 = 150kΩ

Capacitors:
C1;C3;C4;C5 = 10µF 35V
C2;C7 = 100nF 63V
C6 = 100µF 16V

Semiconductors:
D1;D2 = 1N4848
D3 = zener, 5V1, 0.5W
T1 = BUZ11
U1 = 74HC74
U2 = MAX186DCPP or MAX192 (cheaper but less accurate)
U3 = CA3140E

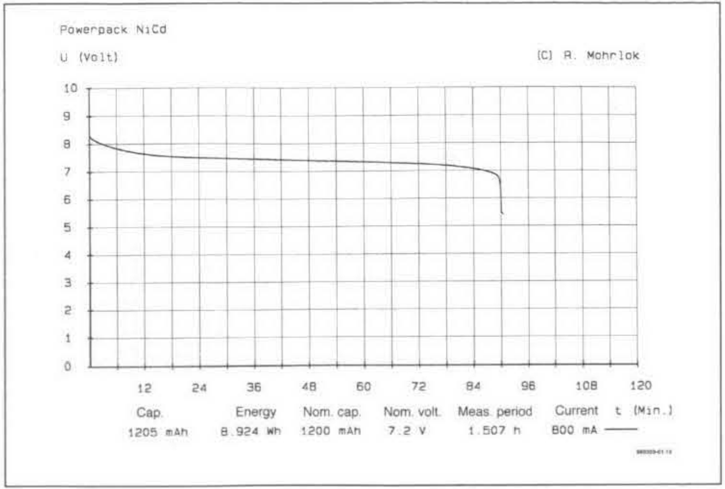
Miscellaneous:
J1 = 9-pin PCB mount sub-D connector
J2;J3 = solder pin 1mm
F1 = fuse 1A plus holder
Heatsink SK96/84 (Fischer)

Fig. 2. PCB track layout and component mounting plan.

Software

The control software was written in BASIC so it should run on any PC. The appeal of the whole story is the ability to make one's own application on the basis of a short program. With speed in mind, the serial interface driver is programmed 'straightforward' instead of as a loop. No attempt should be made to change the driver because it is essential for the correct operation of the current sink.

If you are interested in experimenting with the software, do note that the result of any ADC interrogation is not available until one cycle



after the call to the relevant subroutine. Consequently, the result of an A-D conversion always lags the associated control byte by one ADC call.

(960309)

Fig. 3. Example of a measurement protocol obtained from the tester in combination with a PC.

Note: The software mentioned in this article is available on disk (see page 70). This software has not been tested by *Elektor Electronics*.

PWM SIGNAL GENERATOR

3rd Prize (F)

Design by Fabrice Baudoin

This circuit, aimed at electronic and power electrical engineers, is a test and simulation instrument for ambitious experiments. In combination with a PC, the pulse-width modulation (PWM) signal generator produces rapid command sequences for modulators, switch-mode power supplies, or other PWM amplifiers.

The present design demonstrates once more that simple and useful projects can be realized without a mass of components.

The hardware

In the circuit diagram shown in **Fig. 1**, three type IL112 opto-isolators provide full electrical isolation on the three channels, while resistors R1, R3 and R5 limit the current through the diodes in the opto-isolators. The electrically isolated signals are recovered by inverting amplifiers which serve to give sufficient drive capacity to the modulated signal. The green LED, D1, indicates the presence of signals on the three channels, while a red LED, D2, indicates that the circuit is busy computing the relevant signals. The push-to-make buttons, BP1, B2 and BP3, enable you to set the frequency of the modulated signal. BP1 acts as the 'fine' control, and BP2 as the 'coarse' control. When push-button BP3 is pressed together with BP1 or BP2, the frequency increases or decreases respectively (up/down control).

The numbers shown with the different labels in the diagram refer to pin numbers on the 25-way sub-D connector which links the circuit to the PC's Centronics port.

The current consumption of the circuit being very small, the power supply consists of no more than a 9-V battery and a type 7805 5-V regulator.

Construction and component selection

The circuit is built on a printed circuit board of

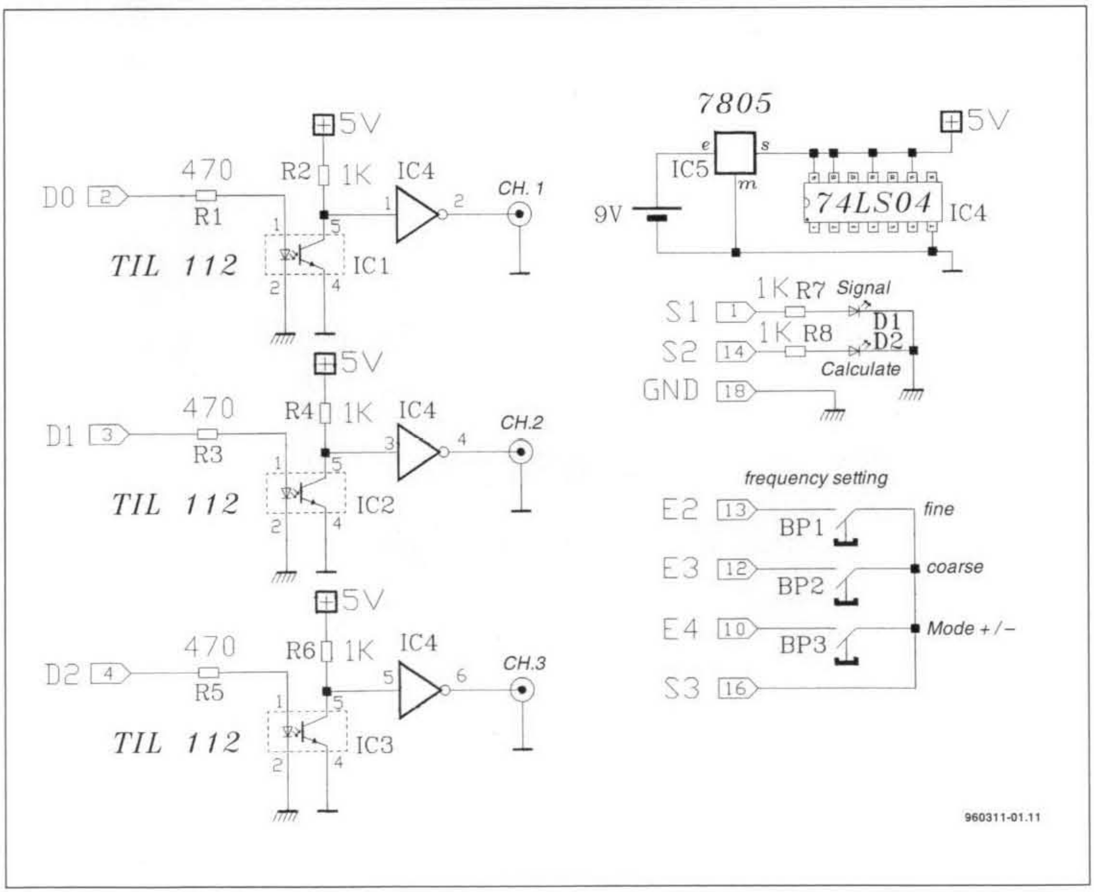


Fig. 1. The PWM Signal Generator hardware is of an amazing simplicity.

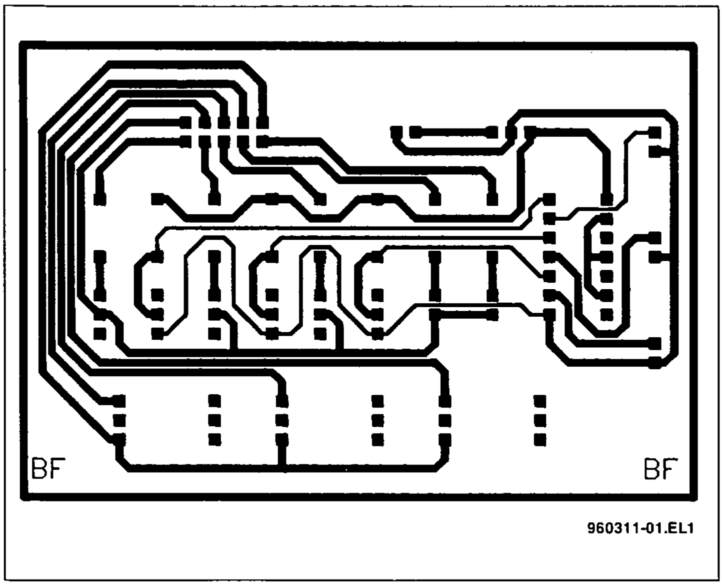


Fig. 2. PCB track layout.

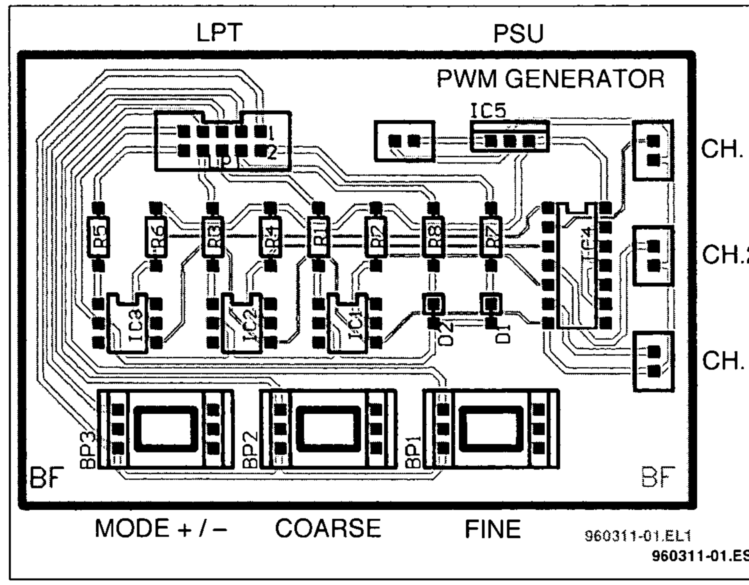


Fig. 3. Component mounting plan.

which the artwork (designed by the author) is shown in **Figs. 2** and **3**. The signal generator is relatively simple to construct. The circuit board has no wire links. Special or programmed parts are not required.

The opto-isolators and the TTL circuit can be picked up at low prices. The cable to the Centronics port may be salvaged from an older PC with a hard disk controller card. The same for the push-buttons, which may be removed from a scrapped keyboard. In keeping with the Competition rules, the design has no more than 30 components.

The cable between the generator board and the PC's Centronics port consists of a short length of flatcable. The circuit diagram indicates pin 1 of the LPT socket. The blue (or red) wire in the flatcable should go to that pin. **Table 1** shows all information required to actually make the cable.

10-way boxheader	IBM sub-D 25-pin	Function
1	16	(Initialize)
2	1	$\overline{\text{STR}}$
3	10	$\overline{\text{ACK}}$
4	14	$\overline{\text{AUTO-FEED}}$
5	12	Paper out
6	4	(Data Bit 2)
7	13	(SEL)
8	3	(Data Bit 1)
9	18	(GND)
10	2	(Data Bit 0)

Applications

Applications of the PWM signal generator will be found mainly in the 'power control' field, more specifically, the setting up of:
- wave sources (no-break PC supplies);
- 3-phase wave sources (simulating a 3-phase source is possible);
- motor controllers (asynchronous motors driven by a PWM generator);
- d.c. motor controllers (acceleration under the control of the PWM signal generator);
- sound generators for audio amplifiers (try it, it's great fun);
- sequences to be loaded into EPROM (for microcontroller systems).

The software

It goes without saying that is it difficult to get a circuit as simple as the one shown here to function without an efficient piece of software.
The software written for this project is relatively complex, and capable of performing several functions. Because of the limited space available for this article, the description below has been kept as concise as possible. At the end of this article is an example showing the practical use of the generator.

Control software: the options

The software that goes with this project offers quite a few possibilities. Essentially, you are looking at DOS software which runs at impeccably, and at good speed, even in a Windows 95 window. Here are the functions recognized by the control software:
- editing signal files ('fichiers.sig')
- building signal files dot-by-dot;
- various existing functions;
- utilize non-linear functions (threshold, saturation, exp. function)
- production of 3-phase and 2-phase systems with phase(s) as parameter(s);
- conversion of signal files to Excel format for reporting purposes;
- conversion of signal files to hexadecimal (EPROM) format;
- controlling the phase of channels 2 and 3 (channel 1 is a reference);
- digital frequency control;
- computing the frequency spectrum of PWM signals or signals to be modulated;
- simulation of first or second-order pass-band, low-pass or high-pass filters;
- recovering a filtered signal;
- using parameters for the spectrum analyser (up to 250 harmonics, display type, etc.);
- and more, for you to explore. (960311)

COMPONENTS LIST

Resistors:
R1;R3;R5 = 470Ω
R2;R4;R6;R7;R8 = 1kΩ

Semiconductors:
D1 = LED, flat, green
D2 = LED, round, red
IC1;IC2;IC3 = TIL112
IC4 = 7805

Miscellaneous:
Clip-on connector for PP3 battery.
10-way boxheader (if necessary with eject levers)
10-way flatcable; one side has a 10-way IDC socket, the other, a 25-way sub-D plug.

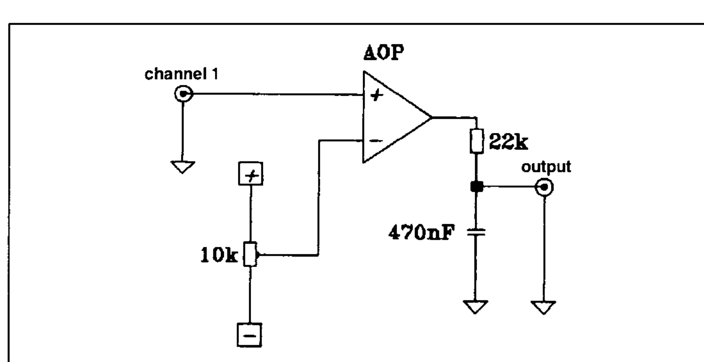


Fig. 4. Example showing the practical use of the PWM Signal Generator.

Note: the software for this project is available on disk through our Readers Services (see page 70). Note that this software has not been tested by *Elektor Electronics*.

PRACTICAL USE

Lets' assume that you wish to test a circuit as shown in **Fig. 4**. The diagram shows an opamp-based amplifier whose inverting input accepts a variable voltage supplied by a potentiometer or an adjustable resistor.

The transfer function, H, of the RC network may be expressed as

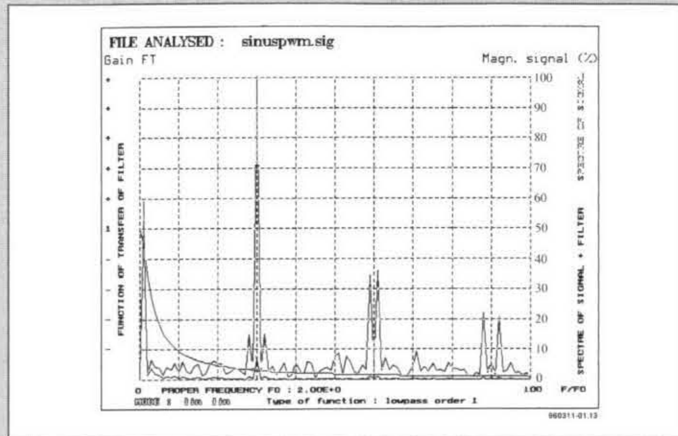
$$H = 1/(1+j^*x)$$

where $x = \omega/\tau$, and τ=RC=0.1s.

We wish to synthesize a sinewave. Launch the program by typing **GO**. Next, select option 1 to generate a 'classic' signal, then select another '1' for sinewave. The level of the modulator is 120% of the carrier, and the frequency ratio is 30. Visualize the output signal by adjusting the PWM signal frequency with the aid of the FINE and COARSE presskeys. The MODE (+/-) presskey allows you to decrease or increase the frequency.

If a certain system requires a PWM signal to be adjusted, it can be run using the **simulation mode**. Return to the main menu, and select option 3 to launch the **spectrum analyzer** function. Load the file 'sinuspwm' (without its extension 'sig'). The software will respond by running a calculation. Activate option 2 to obtain a **display** of the signal spectrum. At this point, you may choose a **filter type**: press option 4, then 1 to get a first-order low-pass filter with a **chopping frequency of 2** (value of 'x'). Re-display the spectrum **2**, this will look like the left-hand screendump in **Fig. 5**. Here you see the **filter transfer function**, the **signal spectrum**, and the **new signal spectrum after filtering**. By selecting option 5 you may **reconstruct the signal** (with the filter action).

In this way, you are able to generate all kinds of signals with the aid of the editor. You may also assign parameters to the generator's channels, de-phase the channels, and use the spectrum analyzer. The rest is for you to discover.



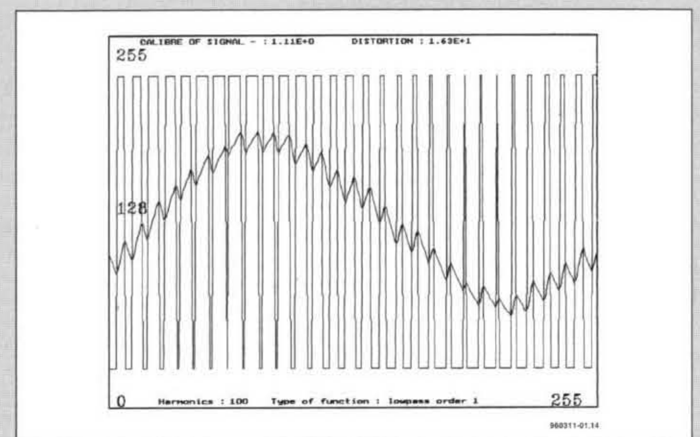


Fig. 5. These two program screens (captured from the PC display with the aid of a printer) should give you some idea of the capacity of the software that belongs with this project. The lower screendump shows another example of a signal.

INTELLIGENT MOTOR CONTROL FOR R/C MODELS

4rd Prize (G)

Design by A. Voggeneder and A. Nader

This miniaturized motor controller uses pulse-width modulation (PWM) for loss-free speed control of d.c. motors up to about 40 amps. The extremely small size and low weight of the controller will appeal to builders of radio-controlled model airplanes, cars and boats, in which space is always at a premium.

The circuit is, however, equally suitable for use in model cars simply by swapping the mode of operation to 'bidirectional' (jumper 4). Motor reversing is achieved with the aid of a 2-pole relay. To make the model drive in reverse, the relay is energized to swap the motor supply polarity. The relay is driven only if no current flows. That is achieved by electronic control to prevent the contacts from burning out or being welded because of the high motor currents.

Another advantage of this type of drive is that the relay contacts need not be rated at the motor's switching current. Instead, far lower, continuous, currents apply. Also, the motor controller is easily linked to a new remote (radio) control. By contrast, conventional motor controllers used to require complex adjustments. The present controller only requires the upper and lower joystick position to be programmed, as well as the 'dead zone'. All relevant data is automatically stored in an internal EEPROM. The motor controller also features a temperature monitor for either the motor or the MOSFET power stage. This protection is actuated at a temperature of about 120 °C, and responds by disabling the output. The circuit is re-enabled when the measured temperature has dropped to about 80 °C. The over-temperature protection may be omitted simply by not connecting the NTC. An on-board low-drop voltage regulator allows the motor controller to take over the power supply of the receiver and the servos, if desired. If you do not need this function, the motor controller may be powered via the receiver. In that case, you may also omit the low-drop voltage regulator.

Hardware: a PIC does it all

At the heart of the circuit shown in **Fig. 1** is a type 16C84 PIC from Microchip Technology. This 18-pin IC boasts the following functions:

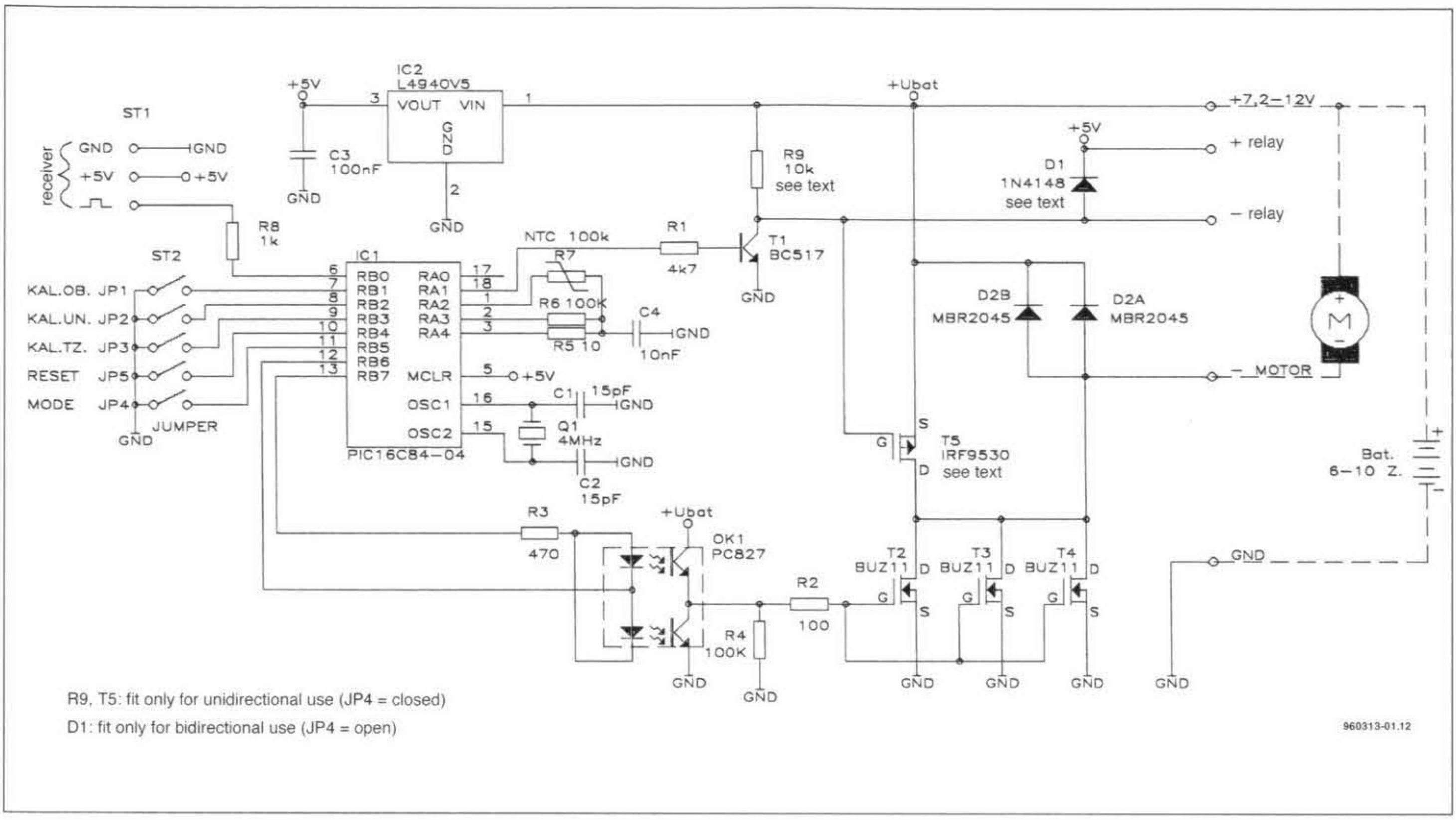


Fig. 1. Circuit diagram of the intelligent motor controller for R/C models.

- 1K × 14 EEPROM;
- 64-byte user EEPROM;
- 36-byte RAM;
- 8-bit timer with 7-bit prescaler;
- 13 programmable I/O pins;
- external interrupt input;
- integrated watchdog and reset timer;
- 4 MHz max. clock speed;
- wide supply voltage range (3.5 to 6V);
- 2 mA current consumption.

The controller measures the output pulses supplied by the radio receiver, and uses this information to calculate the required pulse/pulse ratio of the motor drive signal. The length of the receiver's output pulses is typically between 1 ms (joystick up) and 2 ms (joystick down). The length is about 1.5 ms with the joystick in the centre position. All pulses have a frequency of about 40 Hz. Practical experience indicates that the pulsewidth of the R/C signal may vary up to 500 µs, especially with older R/C transmitters. Some adjustment is therefore unavoidable.

The BUZ11 MOSFET is driven via a dual optocoupler. For this the microcontroller generates an alternating voltage via RB6 and RB7. This voltage alternately switches on one of the optocouplers. One of these pulls the gate of the BUZ11 directly to the positive battery voltage, while the other pulls the gate to ground. This 'rough' drive method guarantees the shortest possible switching time (MOSFETs have a relatively high gate capacitance), as well as a very low drain-source resistance (0.04 Ω, thanks to the high and low gate voltages).

Resistor R2 acts as a current limiter, while R4 pulls the gates to ground during a reset, when the controller outputs are at high impedance, and the two optocouplers are switched off.

The motor direction relay is connected to the relevant points in the circuit. The relay contacts are wired as usual. To suppress motor noise, in particular when reversing, fit a 470-nF 250-V rated MKT (polycarbonate) capacitor across the motor terminals.

The PIC measures the temperature by comparing a fixed 100-kΩ resistor with an NTC of the same nominal value. First, the PIC pulls port line RA3 to +5 V, and then measures the time for C4 to be charged to +2.5 V via R6. Next, C4 is discharged, and the NTC takes over the charging. The ratio of the two charging times allows the PIC to compute the actual resistance of the NTC. When the value drops below a certain threshold, the output is disabled. When another threshold is exceeded, it is enabled again. That creates the necessary amount of hysteresis. R5 limits the capacitor discharging current.

The controller is clocked by a 4-MHz quartz crystal (Q1) which is loaded by two small capacitors (C1 and C2). It is recommended to fit a miniature crystal. If you use a ceramic resonator instead of a crystal, C1 and C2 have to be increased to 33 pF.

Building the controller

The motor controller is built on a printed circuit board of which the artwork is shown in **Fig. 2**. The PIC controller should be mounted in an IC socket. The TO-220 case semiconductors are arranged in a neat row in the high-current area of the board. To prevent them burning out because of the high motor currents, the wide tracks between the MOSFETs and the PCB terminals must be tinned. It is even better to strengthen them with 1-mm dia. silver-plated wire.

The way the board is populated depends to some extent on the application. Model airplanes require unidirectional operation. Boats and cars, on the other hand, need bidirectional operation.

If the motor controller is mounted in a motorized sailboat, a motor brake is usually required to prevent the propeller from turning in the water current behind the model. If a braking function is required, fit the relevant P-MOSFET (T5) and resistor R9, and omit flyback diode D1. Transistor T5 short-circuits the motor as soon as this is switched off, and so brakes the propeller. Because this transistor is driven by the same output as the motor reversing relay, it is essential for JP4 to be fitted, because else T5 is destroyed when the motor is reversed. The upshot is that the motor brake function may only be used in unidirectional mode.

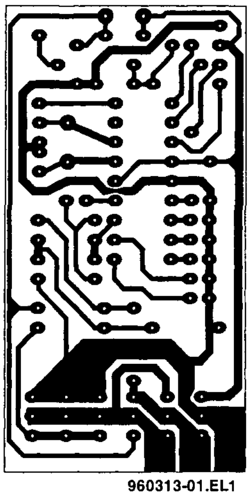


Fig. 2. The motor controller is built on this small printed circuit board. The copper tracks that carry high currents are best tinned or strengthened with silver-plated wire.

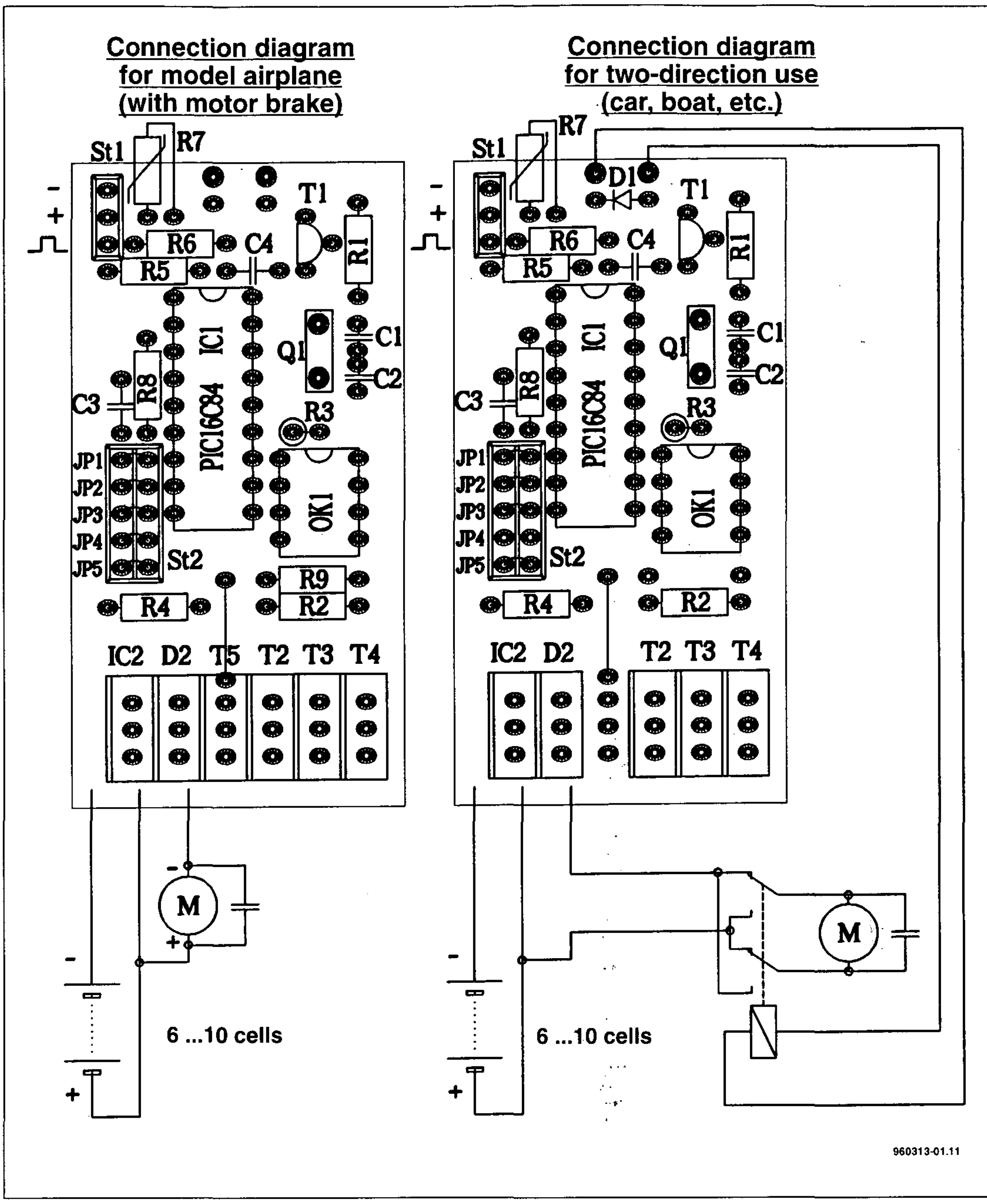


Fig. 3. The board has to be configured for unidirectional or bidirectional operation of the d.c. motor. Also note the important function of jumper 4.

COMPONENTS LIST

Resistors:
R1 = 4kΩ7
R2 = 100Ω
R3 = 470Ω
R4;R6 = 100kΩ
R5 = 10Ω
R7 = NTC 100kΩ
R8 = 1kΩ
R9 = 10kΩ

Capacitors:
C1;C2 = 15pF ceramic
C3 = 100nF ceramic
C4 = 10nF ceramic

Semiconductors:
D1 = 1N4148
D2a;D2b = MBR2045 (dual Schottky diode)
T1 = BC517
T2;T3;T4 = BUZ11 (Siemens)
T5 = IRF9530 (International Rectifier)
Ok1 = PC827 optocoupler
IC1 = PIC16C84
IC2 = L4940V5

Miscellaneous:
Q1 = 4MHz quartz crystal
Re1 = 6-V relay, 2 c/o
ST1 = 3-pin pinheader
ST2 = 10-way boxheader

To adjust the circuit, move the joystick on the R/C transmitter to the upper (front) position, and then briefly fit jumper P1 (for about 1 s). The other extreme of the joystick is adjusted similarly with jumper 2. To set the dead zone, the joystick is pulled to the desired position from 'zero', and jumper 3 is briefly fitted. This procedure enables the controller to compute the current calibration values, which are subsequently stored in the on-chip EEPROM. Jumper 4, as already mentioned, selects between unidirectional mode (jumper fitted) and bidirectional mode (jumper not fitted).

Jumper 5 allows the configuration to be overwritten by standard values. This is achieved by fitting the jumper, switching the supply voltage off and on, and removing the jumper again. PIC ports RB1 through RB4 are actually programmed for connection to push-buttons. Because the actual switching function is only rarely required in this application, and because of the limited space available on the board, jumpers are used instead of switches.

(960313)

Note: the control software to be programmed into the PIC16C84 is available on disk through the Readers Services (see page 70). This software has not been tested by *Elektor Electronics*.

'GREEN POWER' FOR PCs

6th Prize (G)

Design by Joachim Kircher

Most of today's 486 and Pentium PCs offer a 'Green Power' function which serves to switch off external components automatically when the computer has not been used for some time. All equipment is restored to full power again the instant you move the mouse or press a key on the keyboard. This wonderful Design Competition entry tells you how a Green Power function can be installed on older computers.

As you probably know, the monitor is the component with the highest current consumption of all PC peripherals. The circuit presented here saves power and prevents screen burn-in problems by switching off the monitor when no keyboard or mouse activity is detected during a predetermined period (adjustable between 1 and about 20 minutes). A simple switch allows you to override the Green Power controller at any time. The circuit has the following additional features:

- monitor supply automatically restored as soon as a key is pressed, or the mouse is moved;
- powered by the PC, no external supply required;
- monitor power-on delay to prevent current surges when the PC and the monitor are switched on simultaneously.

About the circuit

To avoid all compatibility problems, the Green Power function was implemented as a pure hardware solution. To enable the controller to monitor keyboard and mouse activity, the required signals are 'tapped' from the keyboard and mouse interfaces. That is achieved with the aid of cable insertion adaptors. The supply voltage, +5 V, is stolen from the keyboard adaptor on pin 5; ground from pin 4 and clock from pin 1. Whenever a key a pressed, the CLK output of the keyboard triggers the negative trigger input of monostable IC1A. The bistable responds by generating a 'high' signal of about two seconds at its output. The transmit data supplied by the mouse are also 'stolen' via the adaptor, and arrive at the circuit via pin 2 of the 9-way plug. Moving the mouse causes the voltage level to jump from –5 V (or –12 V) to +5 V (or +12 V). Diode D3 ensures that only positive levels arrive at the circuit. R5 and R4 create a TTL level which triggers the positive input of monostable IC1B. The output of the monostable supplies an active high level of about 2 seconds.

Diodes D1 and D2 form an OR function. The 'high' levels ensure the discharging of the timing capacitor, C4, via R6, T1 and R10. The latter determines the actual discharging current. The two monostables are necessary because C4 is relatively large at 1,000 µF, and can not be fully discharged by a keyboard or mouse pulse.

Timer IC2 (a 7555 = CMOS version of the 555) and transistor T1 form a retriggerable monostable. T2 acts as a driver for a power relay that switches the monitor on and off. Assuming that the two monostables IC1A/IC1b are not re-triggered by the mouse or the keyboard within their monotime, T2 is switched off, and C4 is charged via R8 and po-

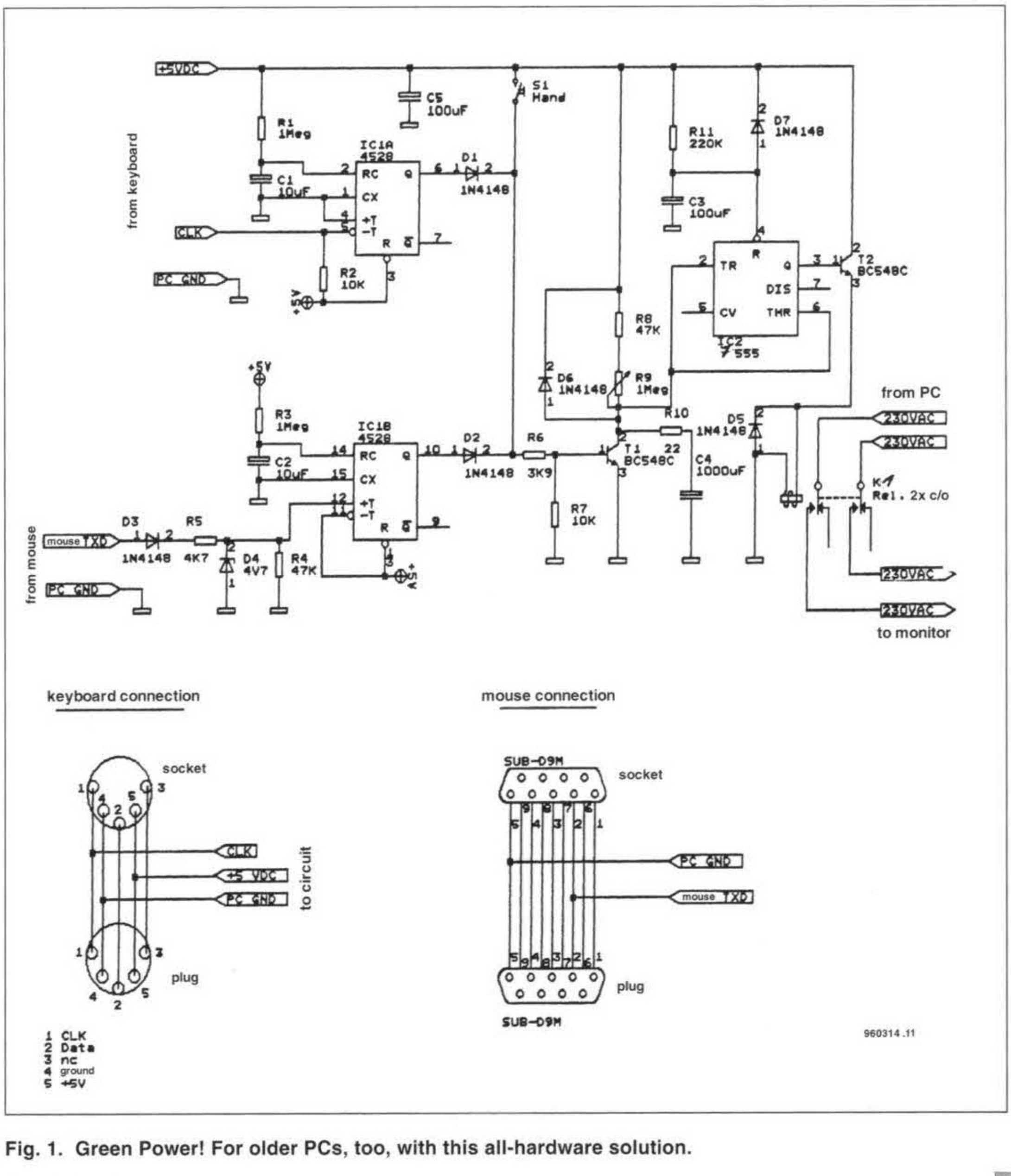


Fig. 1. Green Power! For older PCs, too, with this all-hardware solution.

tentiometer R9. The switch-off delay may be set using R9. Once the switching threshold at IC2 is exceeded, the relay switches off the monitor. When the circuit is switched off, diode D6 ensures rapid discharging of C4. D5 acts as a back-e.m.f. suppressor for the protection of T2. R11 and C3 together provide a power-on delay of about two seconds after the PC is switched on. This prevents the domestic fuse from being blown (or set) by the current surge caused by the PC and the monitor. Here, too, D7 acts as a discharging diode. When S1 is pressed, the Green Power control is disabled. The monitor then remains on all the time. The power-on delay function is, however, retained. The monitor switch-off delay is adjustable between 1 and 20 minutes by setting R9. If the maximum time is not long enough, you may use a pot with a higher value.

Construction hints

Thanks to the relatively small number of components, a simple, single-sided circuit board may be used. Remarkably, there are not even wire links. For the sake of safety, the power relay should be fitted at some distance from the control circuit. That also prevents problems with relays of different sizes and using different pin layouts. No special parts are used in this circuit. All capacitors are fitted upright, which means that you should ask for 'radial' or 'single-ended' types when purchasing these devices.

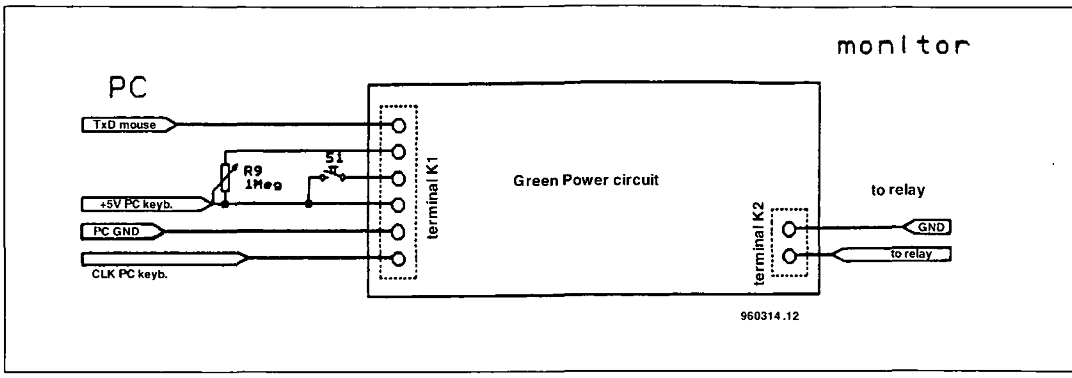


Fig. 2. Board connection diagram.

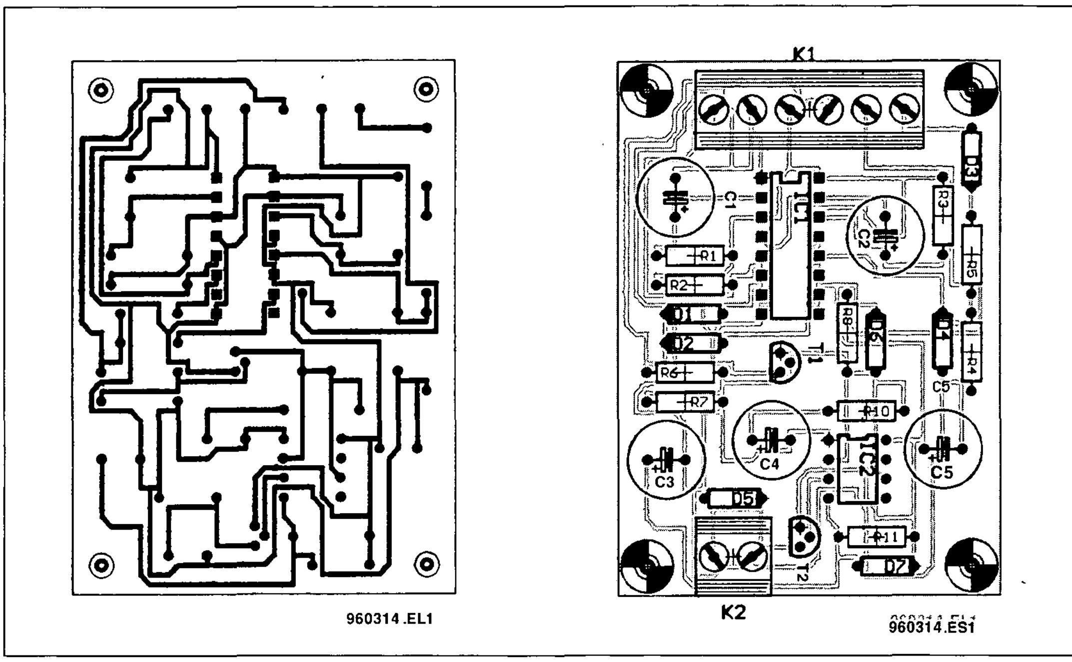


Fig. 3. Copper track layout and component mounting plan.

COMPONENTS LIST

Resistors:
R1;R3 = 1MΩ
R2;R7 = 10kΩ
R4;R8 = 47kΩ
R5 = 4kΩ7
R6 = 3kΩ9
R9 = 1MΩ potentiometer
R10 = 22Ω
R11 = 220kΩ

Capacitors:
C1;C2 = 10μF 16V radial
C3;C5 = 100μF 16V radial
C4 = 1000μF 16V

Semiconductors:
IC1 = 4528
IC2 = 7555
D1;D2;D3;D5;D6;D7 = 1N4148
D4 = zener diode 4V7
T1;T2 = BC548C

Miscellaneous:
S1 = rocker switch 1 c/o
K1 = relay, 2 c/o, 5V coil, contact ratings: 230 V min., 2-3A as required by monitor
One 5-pin DIN plug (keyboard adaptor)
One 5-pin DIN cable socket (keyboard adaptor)
One sub-D 9-way plug (mouse adaptor)
One sub-D 9-way socket (mouse adaptor)

At the end of this second and last 16-page non-stop collection of winning circuits from the International Design Competition 1995 we once more extend our thanks and appreciation to all of you who have actively participated by sending in a design. We also extend our gratitude to the sponsors/advertisers for the magnificent prizes they allowed us to award to the winners.

The thirteen designs which we have had the pleasure of showing to you on these pages over the last three months are only a small selection of a vast number of entries received from all over the world. Many of the entries, although they were not among the published winners, proved that electronics is all about ingenuity, originality, fun, and making common components do things which are definitely not in the standard applications book.

Meanwhile, all winners have been advised, and the prizes have been awarded. That's not the end of the story, however, because some of the better circuits which just did not make it to print in these two 16-page supplements will be prepared for publication in a future issue of *Elektor Electronics*.

HANDS-ON PLC PROGRAMMING (PART 2)

In this second and final instalment of the PLC programming course we deepen our knowledge of the Micro PLC. First we discuss the individual commands that make up the instruction set. Then follow two examples to illustrate the practical application of the system.

PART 2 (FINAL): THE MICRO PLC INSTRUCTION SET AND SOME EXAMPLES

Software by J. Joostens

Let's start by recapitulating the main features of the Micro PLC system. All inputs are electrically isolated, and have a range of –8 V to +16 V (typical: 0 to 12 V_{dc}). The inputs are numbered 0 through 5. The outputs are numbered 6 through 11, and have an open-collector structure. Each of them is capable of switching currents of up to 0.5 A, and voltages of up to 50 V_{dc}. Furthermore, the Micro PLC has available six bit memories (locations 12-17), and a program memory with a size of exactly 48 bytes (locations 16-63). In practice, this offers enough room for a program consisting of 30 lines. An integrated counter is available with a count range of 0-250, along with a programmable delay capable of generating delays from 0.1 s to 25 s in steps of 0.1 s.

The Micro PLC is programmed via a serial (RS232) link with a PC running the Micro PLC software (available on disk). The Micro PLC has two modes: 'program' and 'run'. When in program mode (e.g., after a reset pulse), the user is allowed to copy a program into the programming memory. Interestingly, this mode also allows the Micro PLC to be used as an external I/O device for the PC. In 'run' mode, the Micro PLC executes the program available in its memory.

The Micro PLC indicates its status via two LEDs, a green one and a red one. Their functions are as follows:

red LED	green LED	status
on	off	programming mode
off	on	run mode
off	flashes	mains outage in run mode
flashes	off	invalid instruction in memory

When the mains voltage disappears, the PLC automatically switches over to a back-up supply. The battery contains enough energy to cover a period of three to four hours. When the mains voltage suddenly disappears while a program is being executed, all program output is halted, and all outputs are switched off. Next, the green LED starts to flash. The instant the mains voltage reappears, the green LED stops flashing, and the system starts to execute the program again. If the mains voltage disappears while the PLC is in programming mode, you may simply continue programming. In other words, the PLC may also be programmed while it is disconnected from the mains.

Instruction overview

As already mentioned, PLC programming bears great resemblance to machine code programming. Each line of instruction code consists of three fields: line number, opcode and operand. Labels are not used. Consequently, all branch instructions ('jumps') relate to line numbers. When PLC programs are printed on paper, comment may be added. Just as with code for microcontrollers and microprocessors, opcodes may be represented by mnemonics. To illustrate the general structure, an example of a program line:

```
0023  XOR  04
```

A slightly different approach

To save memory space, the Micro PLC uses memory locations instead of line numbers as on conventional PLCs. Because of this, instructions with an operand take up only two memory locations, and an instruction without an operand, only one. In practice, the 48 available memory locations allow 30-odd program lines to be stored. The jump instruction jumps straight to the stated memory location, and not to a line. This will become apparent from the instruction descriptions given below.

I/O instructions

STH	n	Start High	Opcode =1
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = n			
Level at input, output or aux. memory (n) is fetched into accumulator.			

STL	n	Start Low	Opcode = 2
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = !n			
Level at input, output or aux. memory (n) is fetched, inverted and copied into accumulator.			

OUT	n	Output Accumulator	Opcode = 9
n = number of output (6-11) or aux. memory (11-17)			
Accumulator: not changed			
Execution: n = A			
Level in accumulator written to an output or an aux. memory (n).			

SEO	n	Set Output	Opcode = 10
n = number of output (6-11) or aux. memory (11-17)			
Accumulator: not changed			
Execution: n = 1			
Level of output or aux. memory (n) is made high.			

REO	n	Reset Output	Opcode = 11
n = number of output (6-11) or aux. memory (11-17)			
Accumulator: not changed			
Execution: n = 0			
Level of output or aux. memory (n) is made low.			

CPO	n	Complement Output	Opcode = 12
n = number of output (6-11) or aux. memory (11-17)			
Accumulator: not changed			
Execution: n = !n			
Level of output or aux. memory (n) is inverted.			

WIH		Wait If High	Opcode = 21
n = number of input (0-5)			
Accumulator: not changed			
Wait as long as input n is high.			

WIL		Wait If Low	Opcode = 22
n = number of input (0-5)			
Accumulator: not changed			
Wait as long as input (n) is low.			

WTO	nn	Write To Outputs	Opcode = 23
nn = value between 0 and 63			
Accumulator: not changed			
Execution: outputs 6-11 = nn			
Value (nn) is written to outputs in binary form.			

Logic instructions

ANH	n	AND High	Opcode = 3
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = A&n			
Performs a logic AND function between the level contained in the accumulator and the level at the input, output or aux. memory (n). Result returned to accumulator.			

ANL	n	AND Low	Opcode = 4
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = A&!n			
Performs a logic AND function between the level contained in the accumulator and the inverted level at the input, output or aux. memory (n). Result returned to accumulator.			

ORH	n	OR High	Opcode = 5
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = A+n			
Performs a logic OR function between the level contained in the accumulator and the level at the input, output or aux. memory (n). Result returned to accumulator.			

ORL	n	OR Low	Opcode = 6
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = A+!n			
Performs a logic OR function between the level contained in the accumulator and the inverted level at the input, output or aux. memory (n). Result returned to accumulator.			

XOR	n	Exclusive OR	Opcode = 7
n = number of input (0-5), output (6-11) or aux. memory (11-17)			
Accumulator: changed			
Execution: A = (!A+n)+(A&!n)			
Performs a logic XOR function between the level contained in the accumulator and the level at the input, output or aux. memory (n). Result returned to accumulator.			

CPA		Complement accum.	Opcode = 8
Accumulator: changed			
Execution: A = !A			
Inverts the level contained in the accumulator.			

SEA		Set accum.	Opcode = 24
Accumulator: changed			
Execution: A = 1			
Copies a high level into the accumulator.			

REA		Reset accum.	Opcode = 25
Accumulator: changed			
Execution: A = 0			
Copies a low level into the accumulator.			

Jump instructions

JMP	nn	Unconditional Jump	Opcode = 18
nn = value between 16 and 63			
Accumulator: not changed			
Jumps unconditionally to the specified memory location (nn).			

JIO	nn	Jump If One	Opcode = 19
nn = value between 16 and 63			
Accumulator: not changed			
Jumps to memory location nn if accumulator is at 1.			

JIZ	nn	Jump if Zero	Opcode = 20
nn = value between 16 and 63			
Accumulator: not changed			
Jumps to memory location nn if accumulator is at 0.			

Timer instructions

DLY	nn	Delay	Opcode = 13
nn = value between 1 and 250			
Accumulator:not changed			
Wait nn × 0.1 s			

Counter instructions

ICR	nn	Init Counter	Opcode = 14
nn = value between 0 and 250			
Accumulator: not changed			
Loads value (nn) into counter. Note that the counter is, in principle, capable of counting to 255. Do not use values higher than 250, however, because serial data above 250 are treated as commands by the PLC when in programming mode (see description of use as an I/O card).			

INC		Increment Counter	Opcode = 15
Accumulator: not changed			
Execution: counter = counter+1			
Increments the counter by 1.			

DEC		Dcrement Counter	Opcode = 16
Accumulator: not changed			
Execution: counter = counter–1			
Decrements the counter by 1.			

CCR	nn	Compare Counter	Opcode = 17
nn = value between 0 and 250			
Accumulator: changed			
If the value in the accumulator matches nn, accumulator is made high. If not, the accumulator is made 0.			

Control instructions

NOP		No operation	Opcode = 00
Accumulator: not changed			
Does nothing			

RPM		Return to Program Mode	Opcode = 26
Takes the Micro PLC back from 'run' mode to 'programming' mode.			

VER		Software version	Opcode = 27
Accumulator: not changed			
Sends the system software version number from the Micro PLC to the PC via the serial interface.			

Practical applications

Now that you have an overview of the available instructions, it may be useful to discuss a practical application. To be able to try out the system properly, it is recommended to tie all inputs to +12 V (if necessary via a switch). All outputs have LEDs on the board, so you have an instant indication of their status. The LEDs also enable you to follow the operation of the Micro PLC.

Staircase light control

The purpose of the first project to be discussed is to demonstrate the operation of the timer and counter functions offered by the Micro PLC. The use of the I/O and jump instructions is also discussed.

The function of the program is as follows: when the button connected to input 4 is pressed briefly, output 11 must be actuated for 2 minutes.

The description of the instructions indicates that the maximum delay that can be created with the aid of the DLY instruction is 25 seconds. A delay of 120 s (two minutes) may be achieved by repeating a delay of 20 s six times. This is done with the aid of the Micro PLC's counter. The program is shown in **Fig. 1**, complete with comment.

Location	Mnemonic	Operand	Comment
16	WTO	0	Reset all outputs
18	WIL	4	Wait for button pressed on input 4
20	WIH	4	an released again
22	SEO	11	Actuate output 11
24	ICR	6	Put value 6 in counter
26	DLY	200	Wait 20 seconds
28	DEC	-	Decrement counter by 1
29	CCR	0	Counter equal to 0?
31	JIZ	26	If not, jump to 26
33	REO	11	Turn off output 11
35	JMP	18	Jump back to start

Fig. 1. Listing of the staircase control program.

Window shutter control

The second example is a control system for a motor-operated roller shutter or sun blind. Use is made of the logic operators offered by the Micro PLC. The program discussed may also be used for garage door openers or electrically operated doors, gates and fences. The electrical circuit diagram and the required hardware may be found in **Fig. 2**. The starting point is the system in the 'off' state. The shutter is then down, so that the end contact, S_5, is actuated. The make contact of S_5 is closed, and lamp H_2 is on. The break contact of S_5 ensures that drive K_2 (shutter down) is off. When S_2 is pressed (shutter up), K_1 (shutter-up drive) is actuated. Hold contact K_1 then ensures that the motor remains powered when S_2 is released. As soon as the shutter reaches the top position, S_4 is actuated. That causes the hold current for K_1 to be interrupted, and the motor stops. The make contact of S_4 then switches on lamp H_1. When button S_3 is pressed (shutter down), K_2 is actuated again, and the shutter is lowered until the end switch is actuated. The user is able to stop the lowering or raising by pressing S_1 (stop). Provision is made for an emergency stop. This consists of a bracket at the underside of the shutter (S_6). As soon as this bracket touches an object while the shutter is lowered, the drive is immediately switched off, and the shutter stops.

For reasons of security, push buttons S_1, S_3, S_5 and particularly S_6 should be break contacts. The break contacts of K_1 and K_2 ensure that these drives are never energized at the same time, which would cause a short-circuit between phases L_1 and L_3. Converting all switching functions shown in the diagram into logic functions results in the following relations:

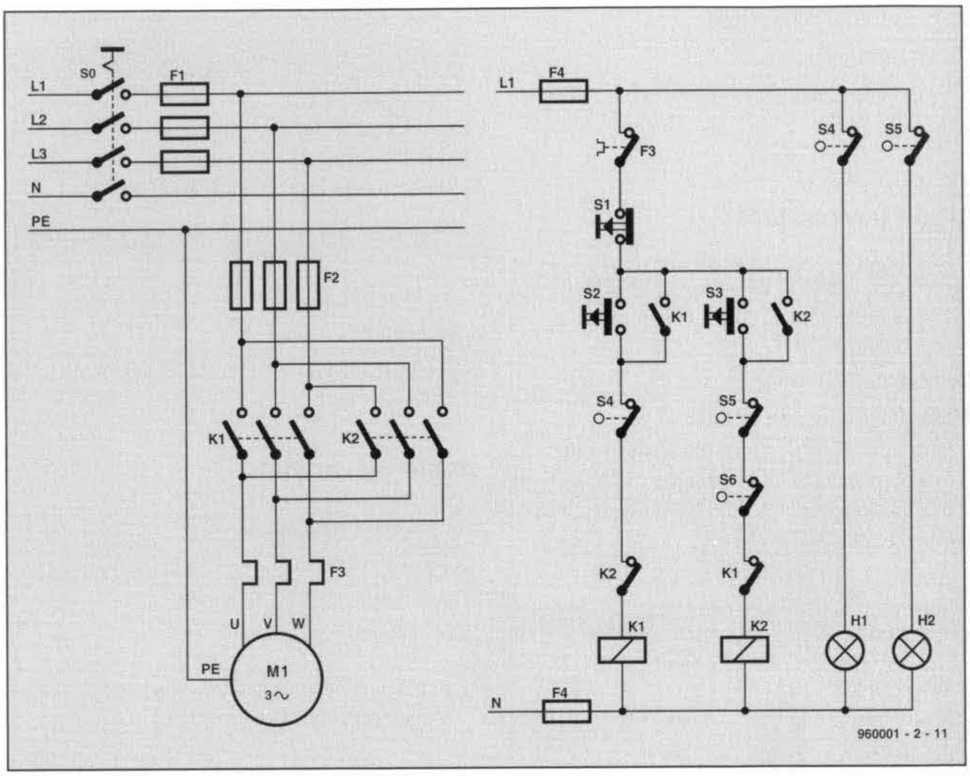


Fig. 2. PLC-controlled window shutter hardware.

K_1 = (!S_1 & !S_4 & !K_2) & (S_2 + K_1)
K_2 = (!S_1 & !S_5 & !S_6 & !K_1) & (S_3 + K_2)
H_1 = S_4
H_2 = S_5

These equations are incorporated into the PLC program. The inputs and outputs are wired as follows:

Input 0:	Up	(S_2)
Input 1:	Down	(S_3)
Input 2:	Stop	(S_1)
Input 3:	Emergency Stop	(S_6)
Input 4:	End Stop Top	(S_4)
Input 5:	End Stop Bottom	(S_5)
Output 6:	Drive Up	(K_1)
Output 7:	Drive Down	(K_2)
Output 8:	Lamp Top	(H_1)
Output 9:	Lamp Bottom	(H_2)

The resulting circuit diagram may be found in **Fig. 3**. The equations set up earlier already took the electrical type of the contact into account: 'make' or 'break'. With the Micro PLC, the actual levels at the inputs should be observed if you want to set up the equations. So, the following equations apply to outputs 6, 7, 8 and 9 of the Micro PLC:

Output 6 = (2 & 4 &!7) & (0 + 6)
Output 7 = (2 & 5 & 3 & !6) & (1 + 7)
Output 8 = !4
Output 9 = !5

Using these equations to write a program results in the code sequence shown in **Fig. 4**.

Programming the Micro PLC

The actual programming of the Micro PLC is considerably simplified if you make use of the auxiliary program 'microplc.exe' found on disk 956016-1 (available through the Readers Services, see page 70). Start this pro-

gram with the serial port used for the Micro PLC as a parameter. If you have the Micro PLC connected to COM2:, you type
microplc.exe -com2 <enter>

A screendump of the program is shown in **Fig. 5**. All locations are initially filled with '26', the code for 'Return to Program Mode'. Reading a program into memory is simple: select the option 'Load Buffer with File' and choose, for example, the file 'loop.plc'. It is also possible to enter a program. After starting the program, select 'Edit Buffer Contents', and go to the 'Mnemonics' option. Enter the program with the aid of mnemonics, and save it to the hard disk with the aid of the option 'Save Buffer to File'. Next, select 'Program MicroPLC', and then 'Download & Autostart'. At that point, you have to reset the Micro PLC. Once the data have been received, the red LED goes out, and the green one lights. The program is being executed. The switches on the Micro PLC may be used to check if the program does what you want it to do.

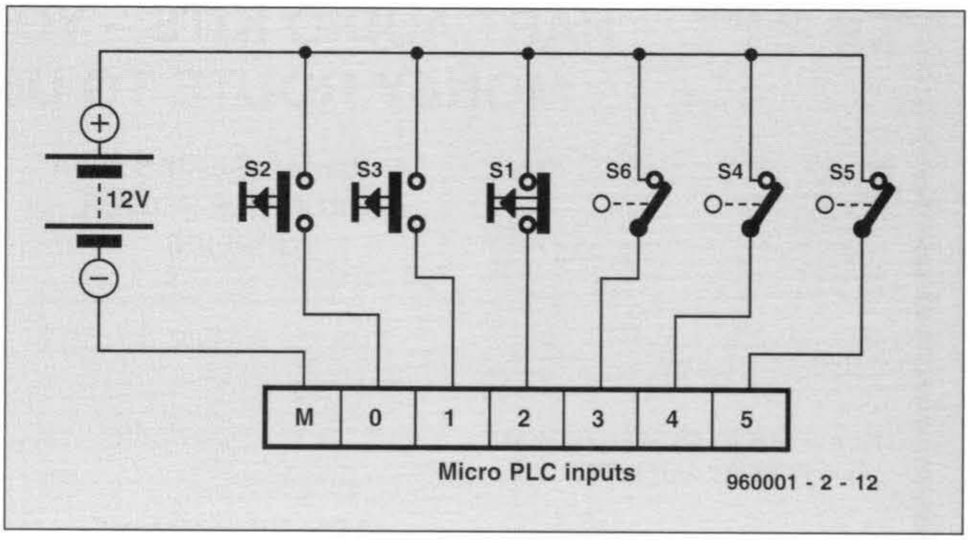


Fig. 3. Motor and switch connections for the shutter control.

Location	Mnemonic	Operand	Comment
16	WTO	0	Reset all outputs
18	STH	0	Read input 0 (S2) into accum.
20	ORH	6	OR function with output 6 (K1)
22	ANH	2	AND function with input 2 (S1)
24	ANH	4	AND function with input 4 (S4)
26	ANL	7	AND function with ![output 7 (K2)]
28	OUT	6	Write accum. to output 6 (K1)
30	STH	1	Read input 1 (S3) into accum.
32	ORH	7	OR function with output 7 (K2)
34	ANH	2	AND function with input 2 (S1)
36	ANH	3	AND function with input 3 (S6)
38	ANH	5	AND function with input 5 (S5)
40	ANL	6	AND function with ![output 6 (K1)]
42	OUT	7	Write accum. to output 7 (k2)
44	STL	4	Read ![input 4 (S4)] into accum.
46	OUT	8	Write accum. to output 8 (H1)
48	STL	5	Read ![input 5 (S5)] into accum.
50	OUT	9	Write accum. to output 9 (H2)
52	JMP	18	Repeat program

Fig. 4. Listing of the window shutter control program.

Alternative application: an intelligent I/O card

Besides its obvious function as a Programmable Logic Controller with a limited instruction set, the Micro PLC offers the functionality of an intelligent I/O card for PCs. As long as the Micro PLC is in 'programming' mode, it considers all characters with an ASCII value smaller than 250 as data, while characters in the range 251 through 255 are interpreted as commands. The function of these commands is as follows:

Code	Application
251	return value of inputs as characters (0 - 63)
252	write next character in binary to outputs
252	return value of outputs as characters (0 - 63)
254	return software version number
255	switch Micro PLC to run mode

To simplify the use as an I/O module, the diskette offers the Turbo Pascal program 'PLC.TPU'. This program contains a series of procedures and functions which are useful to the programmer who develops his/her own software. Here are short descriptions of the available Turbo Pascal procedures.

Procedure: setcom (comadr : integer);
Initialize the serial port with the base address 'comadr' for the communication with the Micro PLC.
The following procedures/functions may only be used after the procedure 'setcom' has been called.

Function readinputs : byte;
Reads the inputs of the Micro PLC, and returns the value as a binary number between 0 and 63.

Procedure output (number : byte);
Puts the value of 'number' (between 0 and 63, binary) on the outputs of the Micro PLC.

Function readback : byte;
Reads and returns the values of the outputs.

The short demonstration program 'plctest.exe' on the diskette shows you some of the possibilities of the Micro PLC when used as an intelligent I/O card. The program is written in Turbo Pascal, and makes copious use of the unit 'plc.tpu'.

Like 'microplc.exe', the program 'plctest.exe' must be started with the number of the communication port

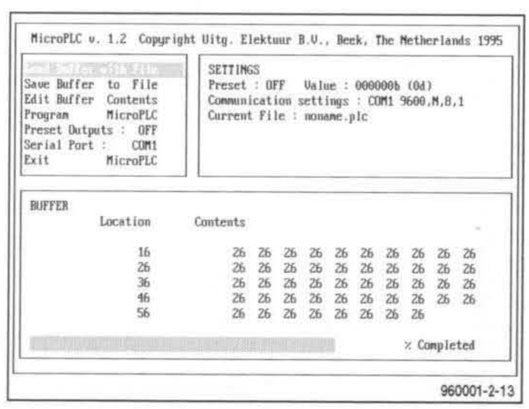


Fig. 5. The programming and downloading utility, microplc.exe, in action on a PC.

(com_x) as a parameter. For example:
plctest -com2 <return>

The program drives the Micro PLC outputs like an incrementing binary counter. Apart from this function, the inputs are read continuously, and their states are displayed on the PC screen. (960001-