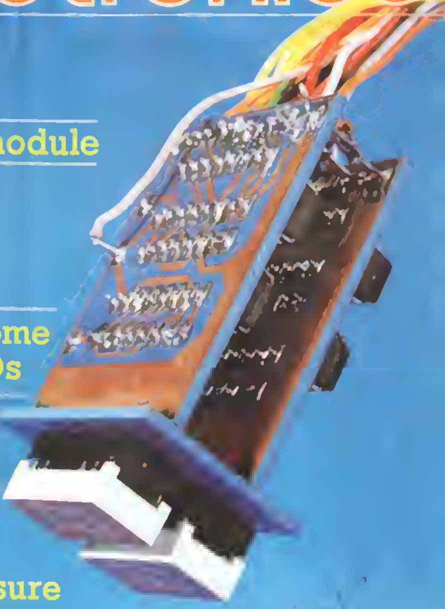
119

April 1985

elektor INDIA electronics

Rs. 7.50



Versatile module

counts up and down; presets and latches; drives any 7-segment display – LED, LCD, common cathode or common anode

throwing some light on LEDs

KITT scanner

the secret behind those robot eyes why they look as if they're looking

Sound pressure meter

BASIC (PART 4)

elektor contents

Volume 3-Number 4

EDITOR: SURENDRA IYER
PUBLISHER C. R. CHANDARANA
PRODUCTION C. N. MITHAGARI

ADVERTISING & SUBSCRIPTIONS
elektor electronics pvt ltd.
Chotani Building
52 C, Proctor Road Grant Road (E)
Bombay-400 007

DISTRIBUTORS: INDIA BOOK HOUSE

PRINTED AT:
TRUPTI OFFSET
103 Vasan Udyog Bhavan,
Off Tulsi Pipe Road,
Lower Parel, BOMBAY-400 013

Elektor India is published monthly under agreement with Elektuur B V Holland. August/September is a double issue.

SUBSCRIPTION
INLAND
1 Yr Rs 75/- 2 Yrs Rs. 140/ 3 Yrs Rs 200/-
FOREIGN
One year Only
Surface mail Rs 125/- Air mail Rs 210/-

INTERNATIONAL EDITIONS EDITOR P HOLMES

Dutch edition	Elektuur b v, 6190 AB Beek (L), the Netherlands
German edition	Elektor Verlag GmbH, 5133 Gangelt, W Germany
English edition	Elektor Publishers Ltd, Elektor House, 10 Longport, Canterbury CT1 1PE Kent U K
French edition	Elektor Sarl, Le Seau, 59270 Bailleul, France
Italian edition	Elektor, 20092 Cinisello B., Italy
Spanish edition	Elektor Av. Alfonso XIII 141 Madrid 16
Greek edition	Elektor, Karaiskaki 14 Voula Athens
Turkish edition	Elektor Yayincilik ve Ticaret A S , Karakoy Istanbul
Portuguese edition	Ferreira and Bento Lda, R.D. Estefania 32, 1000 Lisboa Protugal

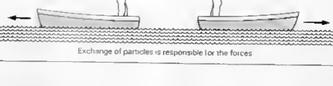
Grand Unification: for and against

During the past few years two important events in fundamental particle physics have provided evidence on the one hand for and on the other against grand unified theories, known as GUTs for short, which seek to explain all the fundamental particles of matter and all the forces of nature in the same mathematical terms. The first event is really a non-event: it is the failure of physicists so far to detect proton decay. The second is the production and identification of particles called W and Z bosons, findings which strengthen the case for GUTs perhaps more than it is weakened by the failure to find protons decaying.

Protons, the positively-charged particles in atomic nuclei, certainly have very long lives. They last for billions of years before they decay, losing mass as energy and changing into other particles. In fact it is still uncertain whether protons do decay. Scientists in India, Europe and America have set up experiments deep underground with sensitive detectors to search for proton decay. The reason for siting the experiments underground, in a disused gold-mine in India, in the Mont Blanc tunnel in Europe and in a salt mine in North America, is that it prevents the detectors being confused by the arrival of highly energetic particles in cosmic rays from outer space. No event generally accepted by physicists as being due to proton decay has yet been detected by any of the scientific teams.
The failure to detect proton decay is important to grand unification theory. There are several reasons for wishing to observe it, among them the fact that it probably happened a lot in the first fraction of a second after the 'big bang' in which the universe began. Observing proton decay would provide a replay of one of the more important events of that all-important earliest epoch.
Equally important is the evidence that proton decay would provide for symmetry and unity in nature. If protons decay, then the particles of which protons themselves are made up, namely quarks, have to be transformed into others called leptons, the generic name for particles such as electrons. Quarks and leptons are now believed to be the only two truly fundamental types of particles, and groupings of various types of quarks on the one hand and leptons on the other are beginning to look very similar. If proton decay shows that quarks can be transformed into leptons, then the unification of all forms of matter is in sight. For proton decay to mean that quarks and leptons are interchangeable, however, implies an average lifetime for protons before they decay of about 10^{31} years. The more the observed lifetime of the proton exceeds this span, the less likely it is that quarks and leptons are interchangeable and the further the unification of matter which depends upon it recedes.
So, the failure of physicists to agree that they have found decaying protons in any experiment so far, and the consequent implication that the lifetime of the proton is longer than had been thought (statistics show that they should have found several decaying by now if the lifetime was as predicted) does not encourage GUTs. But while the physicists involved have been taking one step back from GUTs, other groups may be said to have taken two steps forward with their identification of the W and Z bosons.
To understand why this is so important, you need first to understand how far unification of the forces of nature has proceeded. Matter is held together by four forces: gravitation, electromagnetism, and the strong and weak nuclear forces. The great physicist James Clark Maxwell unified the hitherto separate electrical and magnetic forces into one in the 19th century. Just over ten years ago particle physicists Abdus Salam in the UK and Steven Weinburg and Sheldon Glashow in the USA constructed a theory which unified the electromagnetic and weak nuclear forces. To prove it meant that particles, the so-called W and Z bosons which the theory predicted must exist, had to be found.
The electromagnetic force between particles is carried by other particles, photons. If, as Salam, Weinberg and Glashow predicted, weak and electromagnetic forces are really just two manifestations of the same thing, the weak nuclear force should also be mediated by particles, the W and Z bosons.
W bosons must be very heavy and so operate over very short distances. It requires very high energies to create W bosons that are not locked up inside atomic nuclei and that fly about freely enough and for long enough to be observed. Such energies were not available at the time when Salam, Weinberg and Glashow put forward their theory. But the energies needed to liberate detectable W and Z bosons have now been produced at the European Laboratory for Particle Physics, CERN, near Geneve.
The experiments involved colliding a beam of protons circulating inside the Super Proton Synchroton (SPS) with a beam of anti-protons circulating in the opposite direction. Each proton is made of three quarks and each anti-proton of three anti-quarks. When quark and anti-quark collide they annihilate each other and release enough energy to create particles as heavy as the W and Z bosons.
On 20th January 1983 CERN announced that, among some 10^9 collisions of particles observed in their proton-anti-proton colliding experiments, they had seen the characteristic signature of W bosons in just five. These observations were made by one of the two teams of scientists working with the SPS

The forces in nature

Type	Intensity of forces (decreasing order)	Binding particle (field quantum)	Occurs in
Strong nuclear force	~ 1	Gluons (no mass)	Atomic nucleus
Electromagnetic force	$\sim \frac{1}{1000}$	Photon (no mass)	Atomic shell
Weak nuclear force	$\sim \frac{1}{100\,000}$	Bosons Z^0, W^+, W (heavy)	Radioactive beta disintegration
Gravitation	$\sim 10^{-39}$	Graviton?	Heavenly bodies



Exchange of particles is responsible for the forces proton-anti-proton collidar, the team known as UA1. Soon aftar, four more such events were reported by the second team, UA2. Some 80 W bosons have now been seen and very recantly five Z bosons have baan identified.

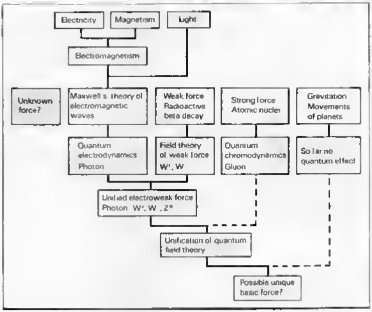
The remarkable discovery rewarded six yeafs of intense activity at CERN and on the part of the British participators in the CERN experiment, from tha Rutherford-Appleton laboratory near Oxford, Queen Mary Collaga in the Univarsity of London, and Birmingham University.

Altogether, 120 international physicists, including 22 from the UK are involved in UA1. Their experiment uses a chamber that records the tracks of ell tha charged particles from tha collisions of protons and anti-protons. Surrounding it are two arrays of counters to measure the anergias of all the particles produced by the collision. One array measures the energies of laptons, the other the energias of strongly-interacting nucleer particles such as protons and neutrons.

This second array, called e Hadronic calorimeter, was built by the three UK scientific groups. It consists of seven thousand sheets of scintillators, detectors that scintillate whan struck by the particles, with instrumentation to record and measure the energy of the impects. The UK ecientists aleo produced an electronic processor to measure the energy of impacts and to decide electronically whether or not they should ba recordad. This vital pert of the apperetus enables the axparimanters to select, from the thousands of events produced evary second by the colliding beams, just those that merit study.

The W boson is so heavy, about 80 timas the mass of a proton, that it is created in a stata of rest out of the energy of quark-anti-quark collision. It ie unstable and decays almost immediately to lighter particles. One way in which it decays, which can ba detected, is to form two particles — an electron end an anti-neutrino.

Only the alactron is discerned when this happens, because nautrinos cannot be datactad by conventional maans. So what should ba discarnable in this moda of decay is a high-energy electron track going off in one direction with, most unusually, nothing complementing it in the other direction. What has really happanad is that while the high-energy electron has gone off in one direction carrying a momentum of 40 GeV (giga-alectronvolts), half the energy of the W boson it came from, an



anti-neutrino has gone off, undetectably, in exactly tha opposite direction with a balancing momentum of 40 GeV. In the UA1 group's first expariments 800 000 events ware recorded on magnetic tape; computer selection picked out 40 of them, and careful scanning of those produced just five that showed exactly the right characteristics, with high-energy alactrons going off in one direction end a large emount of momentum clearly missing in the opposite direction.

Discovery of the W particla was followed in May by four events to do with Z particles. Tha Z particla is eimilar to the W but carries no charge and is slightly heavier. It is expected to be produced ten times less frequantly by collisions between protons and anti-protons than is the W particle, which explains why the Z tcok longer to find, although its signature is equally distinctive.

Most physicists agree that the identification of the W boson cemants the bond between the electromagnatic and weak nuclear forces. The naxt step in grand unification is to try to demonstrate that the strong nuclear force and the weak/electromagnetic forces are mediated by particles in the same way. A theory of tha action of the strong force involving its mediation by particles, called gluons, already exists and it may be possible to confirm their existence experimentally. So far, however, physicists have made little progress towards assimilating the fourth force, gravitation, into a grand unified scheme of things.

Staniforth Webb (938 S)

The roll-up, fold-up battery

Rechargeable batteries of tha future, big enough to drive a car, and yat small enough for ordinary home use, are likely to be completely solid and capable of being rolled up like sticking tape or made in flat sheets and folded

This is the prospect held out by a new project just launched by scientists at the UK Atomic Energy be mada from lithium and vanadium oxide instead of the lead and nickel-iron of conventional batterias. Authority's Harwell research laboratory. Their work to find the right materials and technology to enable solid batterias to be offered in any shape or size is being sponsored by battery users, manufacturers, and materials specialists.

According to a UKAEA spokasman, the project will provide the basic technology for the manufacture of all-solid batteries. These are likaly to Harwell's applied electrochemistry centre has alraady shown that its new type of battery cells do work. Using foil-thin strips for the components, a complete cell need be only 0.22 mm thick, and this is likely to be reduced further still

Harwell has been working on solid batteries for some time. With help from Denmark, the EEC, and several universities, is has been studying the possible use of lithium batteries to drive cars (LPS)

K.I.T.T. scanner

we let you in on the secret behind one of TV's special effects of the moment

Science fiction writers are hampered only by their own imaginations, which generally means that they are very little hampered indeed. Movie-makers, on the other hand, are often forced to take reality into consideration and do the best possible job with the tools that are available. Most of them, fortunately, succeed in making their films sufficiently 'unearthly' with special effects. Special effects also serve to enhance more ordinary films or TV shows, one of the most popular at the moment being a computer that thinks it is a car. Inspired by certain science fiction movies and our automotive friend we came up with a special effect unit of our own.

Movie makers and TV show producers feel justified in doing almost anything to draw a big audience or improve their ratings. One of the more popular of these 'special effects' shows at the moment has a fully-computerised car as its hero. This car (we will refrain from telling you that it is called K.I.T.T. as that would be advertising) has a series of lights running across its bonnet to simulate a 'scanner'. We all know, of course, that it is nothing of the sort but that does not detract from the effect.

Many special effects are very simple when the 'trick' is known. The 'KITT scanner', for instance, is simply a row of lights flashing in sequence one after another. As the photograph shows, this does not involve anything very complicated. Much of the circuit is repeated eight times. But before we get ahead of ourselves let us have a look at the actual circuit diagram.

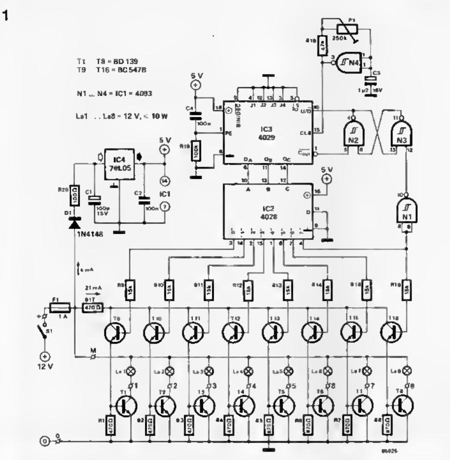
The circuit

The operation of the circuit is easy to follow. When the power is applied the switch-on reset circuit, consisting of C4 and R19, takes pin 1 (Preset Enable) of counter IC3 high for a short time. The data presented to the parallel inputs, I1...I4, is loaded into the counter. All four of these inputs are grounded so IC3 is reset to zero. This has the effect that the circuit always starts from the same condition, except for the state of flip-flop N2/N3.

As long as the circuit is powered the oscillator based on N4 provides a clock signal for the 4029, with the frequency being preset by means of P1. The count in IC3 is incremented by each clock pulse and is continually output via QA, QB and QC. Each of these outputs is connected to a corresponding input in IC2. This binary information is decoded by the 4028 so one of its outputs is continually high. The 4029 always starts from a count of zero so this means that the first output of IC2 to be high will be pin 3 ('0'). Each successive clock pulse from N4 then causes the next of the 4028's outputs to go high, and the previous one to return low, of course. When output '7' of IC2 goes high this signal is fed via N1 to flip-flop N2/N3 causing it to toggle. As a result pin 10 of IC3 is taken low so the 4029 starts to count down. When it reaches zero the $\overline{CO}$ output causes N2/N3 to flip again so IC3 starts counting up.

Each of IC2's outputs ('0'...'7') is connected to the exact same sort of circuit. When an output goes high the appropriate switching transistor, T9...T16, causes the corresponding driver transistor, T1...T8, to conduct so one of the lamps lights. The result is that each of La1...La8 lights, one after the other, first in rising sequence, then falling, then rising again,

1



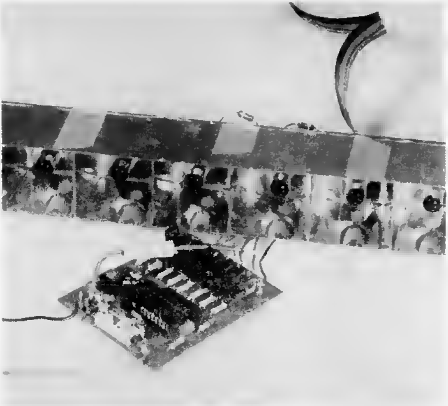


Figure 1. The circuit diagram and the photograph attest to the simplicity of the circuit. The electronics itself uses very little power, most of it being consumed by the eight lamps. The transistors will be overloaded if the lamps used have a higher power rating than the one stated in the drawing.

Parts list

Resistors:
R1 ... R8,R17 = 470 Ω
R9 ... R16 = 15 k
R18 = 47 k
R19 = 100 k
R20 = 100 Ω
P1 = 250 k preset

Capacitors:
C1 = 100 μ/25 V
C2,C4 = 100 n
C3 = 2.2 μ/16 V

Semiconductors:
D1 = 1N4148
T1 ... T8 = BD 139
T9 ... T16 = BC 547B
IC1 = 4093
IC2 = 4028
IC3 = 4029
IC4 = 78L05

Miscellaneous:
F1 = fuse, 1 A slow blow, with PCB mounting fuse holder
La1 ... La8 = light bulb, 12 V/10 W max
S1 = single-pole single throw switch

2

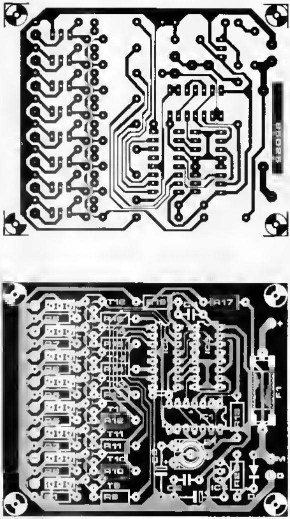


Figure 2. The scanner can be assembled on this printed circuit board (EPS No. 85025). Do not forget to insert the short wire link between IC1 and IC2

and so on. The speed at which this occurs is determined by the position of P1.
The circuit requires a power source of 12 V d.c., such as a car battery, and its current consumption will be about 25 mA. The regulated 5 V needed for IC1...IC3 is provided by IC4. Incidentally, the switching transistors (T9...T16) need only a single common collector resistor as only one of them conducts at any one time.

Construction and calibration

Assembling the circuit for the KITT scanner is simply a matter of soldering all the components onto the printed circuit board whose design is shown in figure 2. The mechanical section is no more difficult. The photograph shows how our prototype was put together. Each lamp is mounted in its own section in the reflector. The reflector is made of a number of pieces of tin soldered together. The wall between two lamps is not vertical; it consists of two pieces of tin soldered together in a 'V' shape with the free ends soldered to the floor of the reflector. Do not make these walls finish flush with the top of the reflector as we have done. The lamps will seem to run into each other better if the walls only extend about 2/3 to 3/4 of this height. A sheet of perspex was fitted in front of the reflector and this was covered with red tape.
The only variable component in the scanner is preset P1. This is 'calibrated' by adjusting it until the lamps flash at a speed that you find pleasing.

A final point...

...about using the circuit. As any policeman will tell you, not just anybody is legally permitted to fit flashing lights to their car, even if they are not blue. That means that the 'scanner' as shown on the TV show is illegal in most parts of the world. Elektor readers are imaginative, however, and we are sure you can come up with an even more innovative use for this KITT scanner.

This photograph of two AXLs forming a stereo amplifier shows how the printed circuit boards, the MOSFETs, the aluminium bracket, and the heat sink are connected together.

AXL amplifier

class A, B or AB

The AXL amplifier described here is intended for operation in class A, AB, or B. Its design specification stipulated that it should be reasonably compact, reliable, robust, and relatively inexpensive to build. It is suitable for use as a power amplifier for electrostatic headphones, in an active loudspeaker system, or in a small hi-fi installation.

The classification of an amplifier depends on the portion of the input current cycle during which output current flows. In class A amplifiers, output current flows over the whole of the input current cycle. These amplifiers have low distortion and low efficiency. In class B amplifiers the output current is cut off at zero input signal, so that a half-wave rectified output is produced. Such amplifiers are very efficient but suffer from cross-over distortion. In class AB amplifiers the output current flows for more than half but less than the whole of the input cycle. At low input-signal levels class AB amplifiers tend to operate in class A, and at high input-signal levels as class B amplifiers.
Power amplifiers commonly work in push-pull, that is, they use two matched devices in such a way that they operate with a 180° phase difference. The output circuits combine the separate outputs in phase. When complementary transistors are used in the two halves, no phase shift is required in the inputs. If both halves of the stage are active simultaneously, they

Technical data

Input sensitivity	790 mV_{rms} for 25 W into 8 Ω 700 mV_{rms} for 40 W into 4 Ω
Input impedance	5 kΩ
Power gain	25 dB
Output power	15 W into 8 Ω } class A – full drive 7 W into 4 Ω } quiescent current: 1 A 2 W into 8 Ω } class A – reduced drive 1 W into 4 Ω } quiescent current: 350 mA 25 W into 8 Ω } class AB – full drive 40 W into 4 Ω } 50 W into 8 Ω } class B 70 W into 4 Ω } quiescent current: 100 mA
Dissipation (no signal condition)	65 W – class A with quiescent current of 1 A and supply voltage of ± 32 V 23 W – class A with quiescent current of 350 mA and supply voltage of ± 32 V 9 W – class B with quiescent current of 100 mA and supply voltage of 2 × 45 .. 50 V
Frequency response	13 Hz ... 65 kHz at −3 dB class A } 600 Ω source 20 Hz ... 20 kHz at −3 dB class B } impedance
Harmonic distortion (primarily 2nd harmonic)	≤ 0.02 per cent in frequency range 20 Hz ... 20 kHz
Damping factor	100 (at 1 W output at 100 Hz)

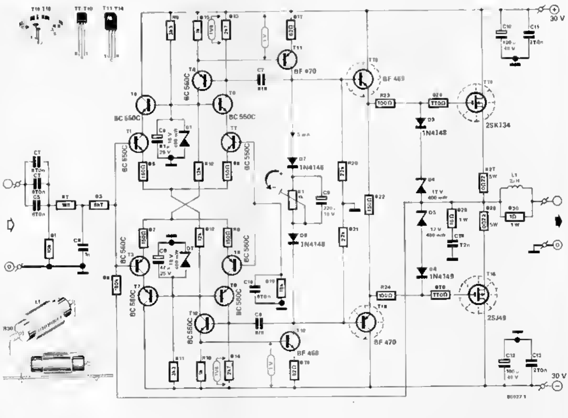
provide equal contributions to the output current; this is the case in class A operation. In class B operation only one half of the stage is active at any one time, and this depends on the polarity of the output current.
A predetermined mode of operation, A, B, or AB, is effected by suitably adjusting the quiescent current (that is, the current under no signal conditions) through the output stage. The quiescent current flows through both halves of the output stage. Each change in the current with respect to the quiescent current in each half of the stage contributes to the output current. In class A operation, the quiescent current is so high, and the output current so low, that both halves of the output stage are on all the time. In class B operation, the quiescent current is, in theory at least, zero. In class AB operation, the quiescent current is set to a level which is appreciably higher than in class B, but much lower than in class A
Because of the heavy demands on the power supply and cooling, a class A amplifier is considerably dearer per watt output power than a class B amplifier. But, since the reproduction in class A is better than in class B, it seems logical to opt for a compromise: that is, class AB! This becomes even more attractive when you realize that during the reproduction of both music and speech full output is required during very short periods only. With a well-chosen quiescent current, the amplifier therefore sometimes works in class A (low inputs) and sometimes in class AB (high inputs). The consequent increase in distortion as compared with that in class A operation is measurable, but not audible
As to the question of rated power output, both the Crescendo (*Elektor* (UK) December 1982) and the Mini Crescendo (*Elektor* — June 1984) appear to meet a need, at least according to many readers' letters. But bearing in mind the design specification mentioned before, we modelled the AXL amplifier on the Mini Crescendo, resulting in a symmetrical circuit with two complementary MOSFETs in the output stage. Both as regards costs and dimensions, the case, the power supply, and the heat sinks are comparable to those in the Mini Crescendo.

Circuit

As most amplifiers, the AXL may be split into an input stage, voltage amplifiers, and an output stage. As shown in figure 1, the input stage consists of a dual symmetrical differential amplifier. The two transistors normally constituting a differential amplifier are formed by cascodes T1-T5, T2-T6, and T3-T7, T4-T8 respectively. A cascode is a super-transistor in which there is only

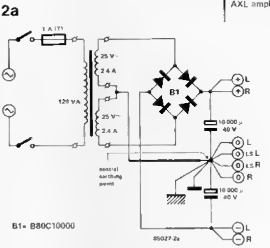
Figure 1. The circuit diagram of the AXL amplifier.

1



negligible feedback from collector to base. Furthermore, the collector of such a transistor is an almost ideal current source.
The output voltages of the differential amplifiers are present across resistors R13 and R14 from where they are applied to driver stages T11 and T12 via emitter followers T9 and T10. Note that the collectors of the emitter followers are conveniently connected to zener diodes D1 and D2 which are required to ensure proper balance between the two sections of the dual input circuit.
In contrast to the two Crescendos, drivers T11 and T12 are not connected in a cascode circuit, because the output stage here is voltage-controlled via complementary emitter follower T13+T14. This dual stage can draw a sufficiently high current via R22. This arrangement obviates the need of using the input capacitance of the MOSFETs for frequency compensation. This compensation is now obtained via Miller capacitors C7 and C8, which in essence are connected between base and collector of T11 and T12 respectively. There is, therefore, a deliberate feedback from output to input of the drivers, and the aim of a cascode circuit is precisely to prevent such feedback. Current amplification in this arrangement is low, and this is the reason that emitter followers T9 and T10 have been added.
The collectors of the drivers are interconnected via network P1-C9-D7-D8, which serves to adjust the quiescent current to the required level. The diodes provide temperature compensation for the current set by P1; they derive their temperature essentially from the heat sinks of T13 and T14. The stability with temperature of the quiescent current is not of paramount importance in view of the excellent thermo-electronic properties of the MOSFETs.
The parallel combination of R20 and R21 forms the load of the driver stages. The values of these resistors have been chosen such that on the one hand the voltage amplification of the drivers is reasonably high, and on the other that the contribution of these resistors (via the current amplification mechanism of T13+T14) to the gate control impedances of T15 and T16 is negligible (that is, with respect to R23+R25 and R24+R26 respectively).
As already mentioned, the output stages of the AXL amplifier are voltage controlled, because that gives an even better linearity than current drive. It also keeps the output impedance, without feedback, lower. The improved linearity and lower output impedance result in very good overall performance with a low feedback factor. And that is desirable, because feedback is and remains a necessary evil. Diodes D3...D6 provide simple, but efficient current limiting of the MOSFETs. Network R29/C14 improves the stability under no-load conditions. Resistors R27 and R28 act as stabilizers of the direct current setting of the output stages. Network

2a



b

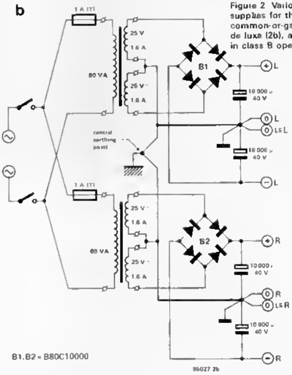
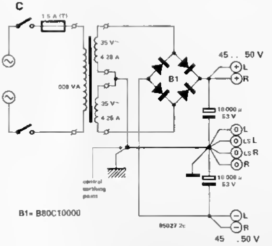


Figure 2. Various power supplies for the AXL: common-or-garden (2a), de luxe (2b), and for use in class B operation (2c).

c



3

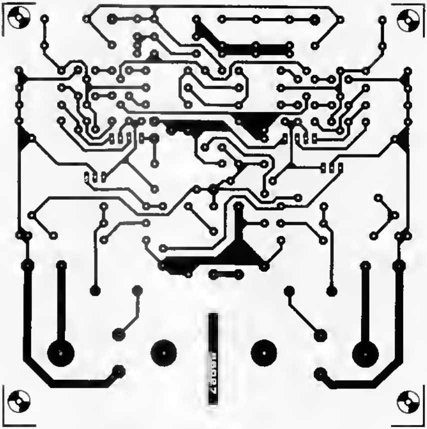
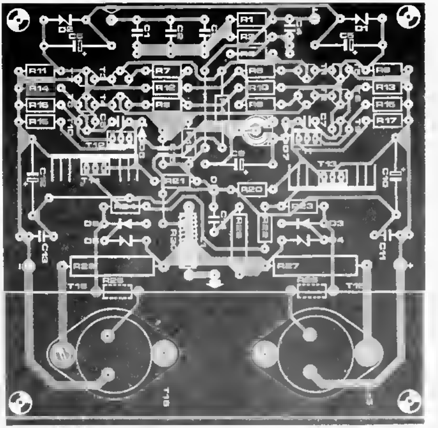


Figure 3. The printed circuit board of the AXL amplifier.

L1/R30 reduces to some extent the capacitive load at the negative-feedback take-off point. The feedback is applied to the input stages via R4. Capacitors C10 . . . C13 provide decoupling of the supply lines.
The parallel combination C1-C2-C3, in conjunction with R1, provides a filter for d.c. and very low frequency signals. Filter R2/C4 prevents signals above about 60 kHz from reaching the input stages.

Construction

The AXL amplifier is constructed along similar lines as the two Crescendos, and it may therefore be useful to reread the two relevant articles. Note that the output transistors are mounted on the printed circuit board: thermal coupling with the heat sink is effected via a right-angled aluminium bracket as shown in the photograph on page 3-27. This arrangement obviates any critical wiring and results in a very compact construction.
As regards the power supply, figure 2 gives you a choice of three. Figure 2a shows one that is common to the left-hand and right-hand channel; figure 2b gives a design for separate supply to the two channels; and figure 2c is intended for use when the AXL amplifier is operated in class B.
The circuit of figure 2a is a single-transformer design. The large-value smoothing capacitors are necessary to keep the ripple voltage on the supply lines low; with smaller capacitances this voltage might easily become unacceptably large in view of the high quiescent current. The ripple voltage does not so much affect the audio signals as reduce the dynamic range.
Note that there are two earth returns per channel: one to the pcb, and one to the loudspeaker. The central earthing point should be the only connection to the amplifier case. This means that the phono (or jack) sockets must be mounted insulated from the case. The connections between these sockets and the pcb should be made in screened cable with the screen connected as appropriate at both ends of the short cable.
The design in figure 2b provides separate supplies for the left-hand and right-hand channels, which are normally only found in very expensive amplifiers. The arrangement ensures that there is guaranteed no interaction between the two channels via the supply lines. The great advantage of using this power supply is that a stereo amplifier can be built from two absolutely symmetrical mono amplifiers which only have the mains switch in common!



If it is required to operate the AXL permanently in class B, higher supply voltages are needed. A suitable power supply is shown in figure 2c. Note that the rating of capacitors C10 and C12 in the amplifier should also be increased to 64 V.

Construction of the amplifier on the printed circuit board is straightforward; note, however, that diodes D7 and D8 should be mounted vertically.

The mounting of the MOSFETs, the aluminium bracket, the heat sink, and all other practical constructional details have been described in the previous crescendo articles (*Elektor* (UK) — December 1982 and June 1984) and is further illustrated in the photograph on page 4-23.

Before the amplifier can be taken into use, it is necessary to check and, if necessary, to correct the off-set direct voltage at the amplifier output, and to set the quiescent current.

Ideally, the direct voltage at the output should be zero, but in practice a value of not more than ± 50 mV is perfectly acceptable. First, measure the direct voltage under no-load and no-drive conditions. If it is negative, T2/T6 and T3/T7 should be made to conduct harder, and T1/T5 and T4/T8 less so. This may be done by reducing R6 and R7 by a certain amount, and increasing R5 and R8 by the same amount. The total values of R5 + R6 and R7 + R8 therefore remain unchanged. For instance

R6 = R7 = 120 Ω; R5 = R8 = 180 Ω.

If the direct voltage has risen to less than —50 mV, no further action is required; if not, the resistance values should be changed further, e.g.,

R6 = R7 = 100 Ω; R5 = R8 = 220 Ω.

If the direct output voltage is too high and positive, R6 and R7 should be increased, and R5 and R8 reduced, in a similar way to that described for negative values.

The quiescent current is measured by connecting a d.c. milliammeter in the positive or negative supply line, or by a d.c. millivoltmeter across R27 or R28 (about 25 mV per 100 mA). The quiescent current may be set with P1 between 100 mA and 1 A. The lower value pertains to class B operation, the higher to class A. We have found that a value of 350 mA gives the best compromise between performance and dissipation, but the final choice is, of course, yours!

Parts list (each channel)

Resistors:
R1 = 10 k
R2 = 1k8
R3 = 8k2
R4 = 180 k
R5,R6,R7,R8,R22 = 150 Ω
R9,R11 = 3k3
R10,R12 = 12 k
R13,R14 = 2k7
R15,R16 = 1 k
R17,R18 = 82 Ω
R19 = 18 k
R20,R21 = 22 k
R23,R24 = 100 Ω
R25,R26 = 220 Ω (mount on trackside of pcb)
R27,R28 = 0Ω22 5 W
R29 = 10 Ω/1 W carbon
R30 = 1 Ω/1 W carbon
P1 = 1 k preset (turn fully anti-clockwise before mounting)

Capacitors:
C1,C2,C3,C15 = 820 n (preferably MKM metallized plastic poly carbonate)
C4 = 1 n polystyrene
C5,C6 = 47 µ/25 V
C7,C8 = 47 p polystyrene
C9 = 220 µ 10 V
C10,C12 = 100 µ (rated voltage > single supply voltage)
C11,C13 = 220 n
C14 = 22 n

Semiconductors:
T1,T2,T5,T6,T10 = BC 550C
T3,T4,T7,T8,T9 = BC 560C
T11,T14 = BF 470
T12,T13 = BF 469
T15 = 2SK134 (Hitachi MOSFET)
T16 = 2SJ49 (Hitachi MOSFET)
D1,D2 = zener 15 V/400 mW
D3,D6,D7,D8 = 1N4148 (mount D7 and D8 vertically)
D4,D5 = zener 12 V/400 mW

Miscellaneous:
L1 = about 2 µH, 20 turns in 2 layers of 1 mm dia enamelled copper wire (SWG19) on R30, see detail in figure 1
heat sink for T15 + T16, minimum height 100 mm; e.g. SK85, 0.6°C/W
aluminium bracket, right-angled, minimum dimensions 125 mm long, 6 mm thick, each side 60 mm wide
two heat sinks for T13 and T14, 8.5°C/W, e.g. SK09
mounting and insulating hardware and silicon grease substitute for the transistors to be cooled

Determining the pH value of an aqueous solution is one of the more important measurements in inorganic chemistry. Any connection with electronics seems remote, and yet chemists have made use of electronics in pH measurements for years. They do this with a special sensor which enables the degree of acidity or alkalinity to be displayed analogously or digitally. Until recently these sensors were prohibitively expensive for hobbyists, but as prices have been coming down, we decided to design a pH meter which will be particularly appreciated by aquarium owners.

pH meter

acidity/alkalinity measurement by LCD

As most electronic hobbyists are no chemists, we will keep 'chemistry' to an absolute minimum.

Each aqueous solution has a certain measure of acidity or alkalinity, which is dependent on the concentration of hydrogen-ions in it. The higher the concentration, the higher the acidity, and the lower the pH. When the concentration is low (very few hydrogen-ions), the pH is high and the solution is alkaline. A pH below 7 indicates acidity and a pH in excess of 7 indicates alkalinity.

The pH is defined as the logarithm of the reciprocal of the hydrogen-ion concentration, (H^+), i.e. $pH = \log_{10} 1/(H^+)$

Table 1 gives the pH scale with corresponding numbers of grams of hydrogen-ions per litre of solution, the relative strength of the solution, and typical examples.

A neutral value does not correspond to a concentration of zero ions, but to one which lies at the division between acidity and alkalinity: that is a pH of 7. The concentration is also dependent to some extent on the temperature of the solution. Depending on the nature of the solution, the pH/temperature relation is either direct or inverse. Measurement of the pH is normally related to a temperature of 25°C.

There are two methods for determining the number of H^+ ions in an aqueous solution: colorimetry and electrometry.

In the first, an acid-base indicator is used, which has a different colour in acid or base solutions. The colour change is due to a marked difference in colour between the undissociated and ionic forms. Such indicators are accurate only to about 30 per cent.

The electrometric method is based on comparing the voltage measured by a sensor and a reference potential. A detailed description of this sensor is given later in this article.

The output potential of the sensor changes by about 59 mV per pH unit; this is a reasonable value which may be measured direct with a d.c. voltmeter. Because of the temperature-dependent behaviour of the pH sensor, a temperature sensor was thought to be no luxury. Our pH meter therefore includes a pH sensor and a temperature sensor, with the temperature correction for the pH sensor being made automatically. Moreover, the temperature can be displayed independently

Table 1.

pH		hydrogen ions in g/l	relative strength	typical example
0	acid	1.0	10 000 000	5% hydrochloric acid sulphuric acid
1	↑	0 1	1 000 000	gastric juice
2		0.01	100 000	lemon juice, vinegar
3		0 001	10 000	fruit juice, wine
4		0.0001	1 000	beer
5		0.000 01	100	black coffee
6		0 000 001	10	mineral water, rain water
7	neutral	0.000 000 1	0	purified water, fresh milk
8		0.000 000 01	10	solution of washing soda
9		0.000 000 001	100	borax solution
10		0 000 000 000 1	1 000	soapy water
11	↓	0 000 000 000 01	10 000	film developer
12	alkaline	0 000 000 000 001	100 000	ammonia solution
13		0.000 000 000 000 1	1 000 000	lime water
14		0 000 000 000 000 01	10 000 000	10% caustic soda solution

Electronic part

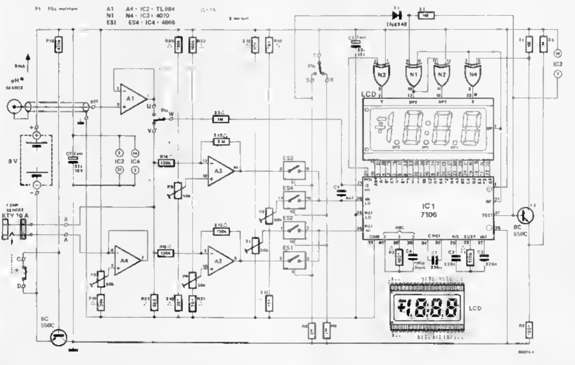
The circuit of the pH meter uses a special voltmeter IC and is therefore quite straightforward as figure 1 shows. This chip, IC1, contains a dual-slope analogue/digital converter and a complete LCD drive stage.

Capacitor C2 is a memory for the auto-zero function in the IC. Capacitor C3 is an integrator which is charged via R1. Reference capacitor C1 is also part of the dual-slope integrator. The battery is connected to the IC (pins 1 and 26) via switching tran-

sistor T1. This transistor is controlled by a micro-switch in the stereo socket for the temperature sensor; it conducts only if the plug of that sensor has been inserted into the socket. This makes an on/off switch superfluous.
The POL(arity) output, pin 20, switches on the minus sign on the display via gate N3 when the input signal is negative. The TEST output, pin 32, arranges a low-battery signal on the display if the battery voltage drops below 7.8 V.
The LCD display is controlled via outputs A1...G1, A2...G2, and A3...G3. The decimal point is set according to the chosen function by N1 and N2.
The reference voltage is connected to REF HI and REF LO (pins 36 and 35 respectively), while the potential difference from the pH sensor is applied to IN LO and IN HI (pins 30 and 31 respectively).
When the reference voltage is suited to the quantity to be measured (temperature or pH) and the starting voltage of the measuring range (IN LO) is preset appropriately, the display will give a direct reading of the temperature or the pH value.
Input REF LO is connected to COM (pin 32), which is not the earth connection of the IC, but provides a stabilized potential which is about 3 V below the 9 V supply voltage. The reference voltage for temperature measurements is set simply by R13/P1; that for pH determination is provided by voltage divider R22/P4/R21 and opamp A2. Switching between the two reference voltages is carried out by electronic switches ES1...ES4. The starting voltage for measuring temperature and pH is preset by R19/P2/R17 and R20/P3/R16/A3 respectively.
With switch S1 in position R/V, the temperature sensor is connected between A (1.69 V reference voltage provided by opamp A4) and B. In this way, the sensor forms a temperature-dependent voltage divider with R23. At 0°C the resistance of the sensor is about 1650 Ω and the potential at B (with respect to earth) is then around 1.3 V. The level at IN LO is set to the same value by P2, so that the display reads 00.0 at 0°C. At higher temperatures, the resistance of the sensor decreases, the voltage at B rises, and the display will then read a positive value. Similarly, at temperatures below 0°C, the display will indicate a negative value. The reference voltage is preset with P1.
Opamp A1 at the pH input provides the required high input impedance of 10^{12} Ω. It would be possible to connect the pH input direct to the relevant inputs of IC1 (also 10^{12} Ω) but switch S1 would then have to be of very high quality to provide such a high isolation resistance between its contacts. As there was a spare opamp available in IC2 anyway, we opted for using that and a cheap switch. The output voltage of the pH sensor drifts about 200 μV/°C. This means that the reference voltage must drift 200 μV/°C in the opposite direction to compensate for the

Figure 1. The circuit diagram of the pH meter is based on a special IC which processes the voltages provided by a pH sensor and a temperature sensor; the results are shown on a 3½ digit LCD display.

1



sensor drift. This correction is provided automatically by opamp A2. The inverting input of this opamp is therefore connected to the temperature sensor via a resistor. The ratio R11:R12 determines the drift per °C.
A separate setting by P3 arranges for a suitable level at the IN LO input. At pH = 7, the pH sensor provides a voltage of around 0 V, but it is preferable and much more convenient for the display to read 7.00. This is achieved by voltage divider R20/P3/R16 and A3, which combine to provide a voltage of 413 mV (7 × 59 mV) at the IN LO input. This setting is also temperature dependent: compensation is provided automatically by R14/R15.

Construction

The printed circuit shown in figure 2 contains relatively few components, which makes construction a fairly simple matter. However, the choice of those components is pretty important. Many resistors (those marked with an asterisk in the parts list and a triangle in the circuit diagram) MUST BE metal film types. This is not so much because of high tolerance, but rather to

The pH sensor

In pH measurements use is made of the potential difference that exists between a metal electrode and the electrolyte into which it is immersed. Such a potential difference also exists between two different electrolytes. Electrolytes may be acids, bases, or salts.
In electrometrical pH measurements a galvanic chain is used which consists of two electrodes that are electrically connected by one or more electrolytes. Externally, the two electrodes are interconnected by a measuring device of very high internal resistance. As this device draws virtually no current, the chemical constitution of the electrolyte is not affected.
Several galvanic voltages are produced in the chain which cannot be measured independently. In the sensor used here, there are therefore a reference electrode and a measuring electrode: figure 4 shows a sketch of the construction.
As the name implies, the reference electrode provides a fixed galvanic voltage between it and the electrolyte to be measured. It consists of a rod of silver compound which is surrounded by a buffer solution, potassium chloride, KCl. The potassium chloride is connected with the electrolyte via a diaphragm which ensures minimal liquid transfer and very low electrical resistance. In the sensor used here, the diaphragm is made of a porous ceramic.
The measuring electrode consists of a silver rod which is bonded to a glass membrane and surrounded by a potassium chloride solution. A potential difference will arise across the membrane which is dependent upon the difference in acidity/alkalinity between the buffer solution inside the sensor and the electrolyte into which the sensor is immersed. The potential difference is probably caused by an exchange of sodium and hydrogen ions between the glass and the solutions.
The potential difference between the two electrodes is directly proportional to the difference in pH of the buffer solution and the electrolyte. All other galvanic voltages cancel one another. Because of the high transfer resistance of the measuring electrode, and to prevent chemical changes in the solutions, the measuring device interconnecting the two electrodes externally must have a very high impedance input — of the order of 10^{12} Ω.

4

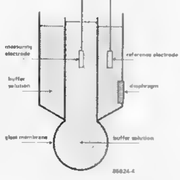


Figure 4. Construction of the pH sensor used for this article.

Using the sensor

It should be evident that because of the glass membrane the sensor should be used with care.
The sensor used contains a maintenance-free buffer solution. In other words, the solution around the two electrodes cannot be replenished. To prevent drying out of the solution, the sensor should therefore always be kept in a potassium chloride solution when it is not in use.
Some golden rules for using the sensor are:

- Never leave the sensor unguarded. A protective cap is supplied which should always be placed over the membrane side. This cap is filled with potassium chloride and may need replenishing once in a while with a KCl solution of 3 mol/litre (this is available as a standard solution).
- Never touch the glass membrane — not even with a cloth, because this will almost certainly destroy the electrode.
- Before every measurement rinse the sensor thoroughly in distilled water.

ensure good stability. All presets MUST BE 10-turn cermet types. IC1 MUST BE a type 7106 and not a type 7106R, because that has all connecting pins reversed w.r.t. the 7106.

After all components have been mounted on the component side, the display should be fitted at the track side. Use a 40-way IC socket which has been cut lengthways as holder. Note that some types of display do not have an arrow for battery indication, but LOW BAT or some other sign. In such cases it may be that the battery indication is terminated in another than pin 38.

Then, place the printed circuit in the box, but do not yet secure it. Just see where the packing density at the edge of the board is least and on that basis determine the location for switch S1 and the BNC socket at the underside of the box. Screen the BNC socket with a piece of tin soldered in position.

File some slots in the box for the presets so that these can be operated even when the box is closed.

Locate the stereo socket for the temperature sensor beside the battery compartment.

The wiring between the BNC socket and the board is fairly critical: use double-



It should not be done like that



But like this

Never use tap water!

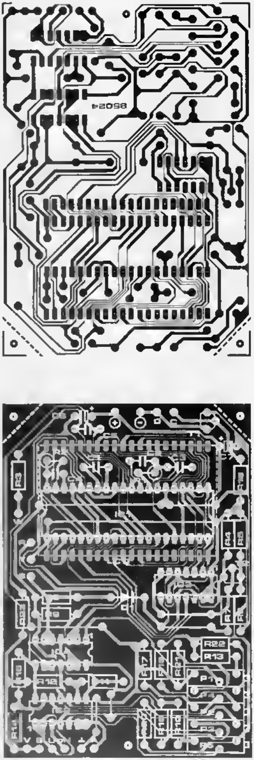
■ Before every measurement, take the temperature of the electrolyte to be measured. This is necessary because of the temperature-dependent behaviour of the sensor. The pH meter automatically compensates for temperature differences w.r.t. 25°C.

■ Some electrolytes may discolour the glass membrane or the diaphragm. The suppliers of the sensor have available a variety of cleaning liquids for specific cases.

■ At the extremes of the measuring range (around 0...2 and 12...14) a small metering error occurs which cannot be corrected. Over the remainder of the range, an accuracy of two per cent is attainable provided the calibration has been carried out correctly.

Finally, a few words about the life of the sensor. Filled with a gel as used here, it has a life of 1...3 years, depending on the number and type of measurements. The great advantage of this type of sensor lies in its ease of use: it only needs to be immersed in the electrolyte to be measured. There are, however, also separate measuring and reference electrodes available (which therefore make replacement of the buffer solutions possible), but these are much more cumbersome to use, although they have an appreciably longer life. They can also be given a new lease of life by being treated with special aqueous solutions. Against that, they are also considerably more expensive, so that in practice most hobbyists will invariably opt for the sensor used in this article.

2



Parts list

Resistors:
R1 = 270 k*
R2 = 100 k*
R3 = 1 M*
R4,R5,R7 R9 = 1 M
R6 = 180 k
R10,R16 = 39 k*
R11,R12,R14 = 130 k*
R13,R15 = 91 k*
R17 = 47 k*
R18 = 470 k*
R19,R21 = 75 k*
R20 = 150 k*
R22 = 180 k*
R23 = 6k8*
P1 P5 = 50 k ten-turn preset
* = 1% metal film

Capacitors:
C1 . C3 = 220 n
C4 = 100 p polystyrene
C5 = 4µ7/16 V
C6,C7 = 22 µ/16 V tantalum

Semiconductors:
D1 = 1N4148
T1,T2 = BC 550 C
IC1 = 7106
IC2 = TL084
IC3 = 4070
IC4 = 4066

Miscellaneous
S1 = double-pole changeover switch
1 off pH sensor type U455; order no. 104553001 — BNC termination — (*Life Science Lab, Serum Road, Luton LU3 2RA, Bedfordshire. Telephone. 0582 597676*)
1 off temperature sensor, type KTY-10-6 (old code KTY 10-A); KTY-81-210; KTY-81 220
1 off 3½ digit LCD display
1 off stereo 3.5 mm socket with built-in switch
1 off BNC socket
1 off 9 V power pack battery
1 off Verobox 65 2996H handheld box with switch cut-off, dimensions 145 x 80 x 36 mm
PCB 85024

Figure 2. The printed circuit board of the pH meter. The LCD display is mounted at the track side. DO NOT use a 7106R for IC1 — only a 7106!

3

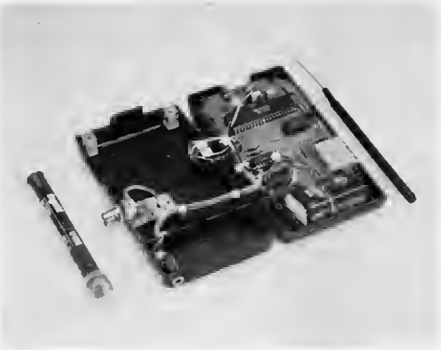


Figure 3. This photograph shows clearly how the meter is assembled in the case.

screened teflon cable and keep it as short as possible. This is vital because of the high isolation resistance required. All other connections may simply be made from flexible equipment wire.
If the pH sensor has not yet been filled with a BNC plug, do so now. Fitting should be done with great care: the cable is of a special type with a very high isolation resistance.
The temperature sensor should be fitted with a length of standard single screened cable and then inserted into an empty ballpen holder which is subsequently filled with two-component araldite (an example of this may be seen in the photo of figure 3). The cable should then be fitted with a 3.5 mm stereo jack: screen to the housing and the conductor to the foremost section. The rear section is used for switching the supply voltage on and off.

Calibration

Buffer solutions with a pH of 4, 7, and 9 respectively are required for calibrating the meter; they are normally available from the suppliers of the sensor.
Set switch S1 to position 'temp' (V/R in figure 1). Place the temperature sensor in a mixture of water and crushed ice (stir well!), wait a few minutes and then adjust P2 to obtain a reading of exactly 1.69 V d.c. between the output of opamp A4 (pin 7) and earth. The display should then read 00.0. Next, place the sensor, together with a clinical thermometer, in a large bowl of water at about 37°C. Again wait a few minutes and then adjust P1 so that the display shows the same value as the thermometer.
Next, set S1 to position 'pH' (U/S in figure 1). Remove the protective cap from the pH sensor and thoroughly rinse the sensor in distilled water. Place the pH sensor and the temperature sensor in the buffer solution with a pH of 7, which should be at 25°C. Wait a few minutes and then adjust P3 to give a reading on the display of 7.00. Remove both sensors from the solution, rinse them thoroughly in distilled water, and then place them in the buffer solution with a pH of 4, which again should be at 25°C. Wait a few minutes, and then adjust P4 to give a display reading of 4.00. Remove both sensors from the solution, rinse them thoroughly in distilled water, and then place them in the buffer solution with a pH of 9. The display should then read 9.00; if not, slightly adjust P4, and repeat the pH 4 test. Finally, remove the sensors from the solution, rinse them thoroughly in distilled water, and then place them again in the solution with a pH of 7. The display should then read 7.00; if not, carefully repeat the above calibration.
The meter is now ready for use. It is advisable to calibrate it at regular intervals because of the ageing of the pH sensor. Calibration is also recommended before measurement when the meter has not been used for some time.
This concludes the description of the pH meter, but for those of you who are interested there follows a detailed account of the pH sensor.

speech for microcomputers

phoneme synthesizer with Centronics input

Speech processing with personal computers is still a very costly and restricted affair. Fairly simple methods using your own spoken text need large memories and even then the results are modest. Industrial, and therefore much more expensive, computers don't do all that much better, although they sound better. The phoneme generator suggested in this article is relatively inexpensive and can be operated with a medium-sized memory.

The proposed phoneme generator is based on the type SPO 256 speech processor IC. We set out to produce a comprehensive construction plan, complete with pcb and application notes. But practical work with the circuit soon showed (as we should have known) that it's not as simple as all that: a pity, but on the other hand the SPO 256 is by far the cheapest IC of its kind on the market. If you want quick results ("Hurray, my computer can talk!"), you'll have to think again. This article is rather for those who are interested in experimenting with speech. It can teach you a lot about the structure of spoken language and the programming of a phoneme synthesizer. A little experience in the art of phoneme arranging can give much enjoyment in coming to grips with the principle of speech deliverance.

A phoneme (Greek 'speech sound') is a basic sound unit in a language. E.g. in English 't' and 'd' are separate phonemes; 'c' and 'k' may be the same or different (cider, caulk, kirk); 'c' and 's' may be the same or different (sirup; cork, civet); 'ph' and 'f' are the same

Phonemes and sounds

From the definition of a phoneme it is evident that any word in a language can be broken down into a number of phonemes. The word 'man', for instance, consists of the phonemes 'm', 'a', and 'n'. But here again, difficulties arise, particularly in English, for there are twenty-six basic letters in the English alphabet, but over forty basic sounds. The English language gets around this problem by using the same letter or letters for different sounds, as in the many ways in which the *ough* combination can be pronounced; and it gives the same sound all sorts of different meanings and spellings: the same vowel<i> appears in *sit, women, village, busy* and *enough,* for instance. Then there are *pear, pair,* and *pare; site, sight,* and *cite; so, sew,* and *sow; caught* and *court; father* and *farther.* German has the same in, for instance, *sein* and *sein* ('to be' and 'his'); *Wetter* and *Wetter* ('punter' and 'weather'); and French in *père, pair, paire* ('father', 'peer', 'pair'); *sûr* and *sur* ('sure' and 'on'), and *sot* and *seau* ('foolish' and 'bucket').

If a word can be broken down into phonemes, it should be possible to build up words from phonemes and it is this consideration, of course, that lead to the concept of speech production by microprocessor. From the above examples it is evident that speech production by microprocessor is immeasurably easier than speech recognition. We must bear in mind, however, that if a computer utters the word *rain*, it could actually mean *reign*; which of the two can only be assessed by the context in which it is used. The human brain can cope; but then it has a memory besides which even the most powerful computer memories pale into absolute insignificance. You will also see the enormous problems still to be resolved before we can hope to produce a computer that can differentiate between *taut* and *taught* or between *Mona* and *moaner*. None the less, in theory at least, if the memory of a computer is loaded with the forty-odd phonemes of the English language, it should be able to produce all the words contained in our language — given a suitable speech processor, of course.

Table 1. Correlation between phoneme code, allophone, phoneme duration, and respresentative sound.

phoneme			representative sound (bold letters)
decimal code	allophone	duration (ms)	
00		10	pause
01		20	pause
02		50	pause
03		100	pause
04		200	pause
05	OY	290	boy
06	AY	170	five
07	EH	50	left
09	KK3	80	count
09	PP	150	peak
10	JH	100	jump
11	NN1	170	none
12	IH	50	it
13	TT2	100	to
14	RR1	130	right
15	AX	50	trouble
16	MM	180	magnet
17	TT1	80	part
18	DH1	140	they
19	IY	170	see
20	EY	200	stay
21	DD1	50	card
22	UW1	80	computer
23	AO	70	long
24	AA	60	hot
25	YY2	130	yard
26	AE	80	man
27	HH1	90	he
28	BB1	40	trouble
29	TH	130	thin
30	UH	70	push pull
31	UW2	170	food
32	AW	250	south
33	DD2	250	do
34	GG3	120	jig
35	VV	130	very
36	GG1	80	go
37	SH	120	shift
38	ZH	130	measure
39	RR2	80	bring
40	FF	110	for
41	KK2	140	skip
42	KK1	120	ask
43	ZZ	150	zero
44	NG	200	talking
45	LL	80	look
46	WW	140	wire
47	XR	250	dear
48	WH	150	where
49	YY1	90	yes
50	CH	150	chip
51	ER1	110	counter
52	ER2	210	turn
53	OW	170	slow
54	DH2	180	lathe
55	SS	80	stop
56	NN2	140	no
57	HH2	130	hertz
58	OR	240	store
59	AR	200	arm
80	YR	250	clear
61	GG2	80	glue
62	EL	140	angle
63	BB2	60	bit

An allophone (Greek, 'other sound') is one of the variant sounds forming a phoneme.

The SPO256 as phoneme synthesizer

The prototype of the SPO256 introduced some years ago was not really a phoneme sythesizer, but rather a speech card shrunk onto a chip with a word store in ROM. The later version, the SPO256-AL2, is, however, since its internal ROM con-

Table 2. Example of a simple sentence, the corresponding phonemes, and the relevant BASIC program.

hello	27-07-45-53-02
this	18-12-55-04
is	12-55-04
the	18-19-04
elektor	19-45-07-08-13-58-04
speech	55-09-19-50-04
card	08-59-21-02

```
10 for K = 1 0 35 step 1
20 restore 80
30 read I
40 LPrint CHR$(I)
50 next K: end
60 data 27, 07, 45, 53, 02
70 data 18, 12, 55, 04
80 data 12, 55, 04
90 data 18, 19, 04
100 data 19, 45, 07, 08, 13, 58, 04
110 data 55, 09, 19, 50, 04
120 data 08, 59, 21, 02
```

1

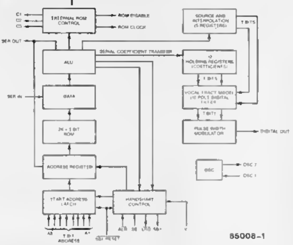


Figure 1. Block diagram of the single-chip NMOS speech processor type SPO256-AL2. The 2 K ROM of this new version contains information for the generation of sixty-four different phonemes

2

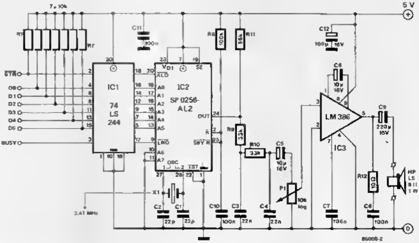


Figure 2. Circuit diagram of the complete experimental phoneme synthesizer.

tains data for phonemes instead of for a certain number of words. This is also a very economical chip, because it is used in vast quantities in many industrial applications — with a different ROM content. Integrated circuits developed specially for use as phoneme generators, e.g. the SC01 produced by Votrax, are considerably more expensive, although to our ears this IC produces a clearer, albeit American rather than English, sound. This American influence is also evident in the SPO256-AL2 which stores fifty-nine phonemes, listed in table 1, although English linguistics and phonetics recognize only forty-odd phonemes in the **Received Standard**. This is a pronunciation of English which gives little or no clue to the speaker's regional affiliations. The synthesis of words from phonemes is comparable to a jigsaw puzzle. Particularly in the beginning it seems almost impossible to get the correct phoneme, but after a while your ear will become attuned and word formation is then feasible.
The block diagram of the SPO256-AL2 is given in figure 1. The data for the phonemes are stored in a 2 K ROM. The synthesizer, which consists of a twelve-pole digital filter and a five-register generator, is controlled by addressed data. The phoneme data determine on the one hand which raw sound material is required from the generator, and on the other with which filter coefficient this sound must be processed. The digital, filtered signal is converted externally into a pulse-width modulated analogue signal.

Example of application

All necessary hardware is shown schematically in figure 2. The input of the small circuit can be connected to any standard Centronics interface. As only the phoneme data are transferred, the total data flow is very small. On average, eight bytes are sufficient for one second's speech. The phonemes (speech signal) generated by IC2 are available at pin 24 as a digital signal. A small external low-pass filter converts this into an acceptable analogue signal. The required sound volume is provided by IC3, a simple audio amplifier of the well-known type LM 386. Programming is relatively simple: all you have to do is to write in the appropriate phoneme code from the list in table 1. These data are then transferred to the print interface by LPrint. As an example, the sentence 'this is the elektor speech card' and the relevant phonemes are given in the upper half of table 2, while the corresponding BASIC program is listed in the lower half.
This is only a small example, but we hope that you will soon progress to bigger things! Have fun!

3

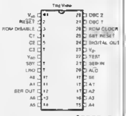


Figure 3. The pin configuration of the SPO256-AL2.

Literature
talk to computers by H P Baumann, *Elektor*, May 1981, p. 5-17
talking board, *Elektor*, December 1981, p 12-04

BASIC (PART 4)

In the first three parts of this series we heve discussed the possibilities of BASIC in generel, es e programming language for computers. In this final part, some espects of 'Extended BASIC' will be described. The possibilities offered by NIBL (the BASIC dialect used for the BASIC microcomputer described in Elektor, May 1979) will elso be dealt with.
First, however, some general programming tips ere in order — as well as tips on 'debugging' programs.

Programming tips

Programs will normally be required to perform a calculation, control a system, or for some similar task. The first thing that must be done — before even thinking about the program itself — is to define the problem carefully. For a calculation, for instance, it is important to know what values the input data may have, whether or not they can be positive, negative or zero, etc. If some form of system control is required, it is important to know in what order various actions must be undertaken — and what can go wrong! Problems when running a program are often the result of incomplete or inaccurate definition of the task to be performed. Once the task is known, the next step is to draw up a flow chart. This provides a clear overview of the basic structure of the program; possible simplifications, improvements or modifications are often immediately apparent. Sometimes it will be discovered that the program can be simplified considerably by slightly modifying the definition of the task. For instance, it may be useful to add a 'call for help to the (human) operator' if a rare exception occurs, instead of laboriously writing a whole section of program to enable the computer to solve that particular problem on its own. For that matter, one should not expect miracles of the computer: any programming venture is doomed to failure unless the problem is fully understood (or the task fully described) before starting to develop the program.
When it comes to the program itself, it is a good idea to start with an 'initialisation' procedure: all variables are given an initial value (usually 0 or 1), by means of the LET statement. Setting variables to zero may seem unnecessary, since it is often done automatically when the computer is switched on. However, this is not always the case and, furthermore, the variables may well have been assigned a new value in the course of a preceding program. In general, 'initialisation' is advisable. If several variables are used in a program, it is a good idea to keep a record of the variables already used and their meaning. A suitable system was given in Part 2 (figure 2).
Only reasonably experienced programmers should attempt to tackle a long and complicated program. Very often, long programs can be 'split up' into several short sections. Each of these can be tested separately, and when they are all running properly they can be 'glued together', producing the complete final program.

Debugging

Once the program has been written, it is time for the first trial run. At this point, the general validity of the Law of Cussedness becomes apparent: it is rare indeed for a program to run properly first time. The next step, therefore, is debugging.
Possible errors can be divided into two categories: those that can be detected by the computer itself ('procedure errors') and those that only become apparent from incorrect execution of the program ('execution errors').
Procedure errors are usually discovered by the interpreter, when it attempts to translate the instructions into machine language. In some way, the instructions don't conform with the rules of BASIC. These are 'silly' mistakes, usually — otherwise the computer wouldn't find them! — like typing PRANT instead of PRINT, or A = C (B + Q) instead of A = C * (B + Q). Both of these examples are so-called 'syntax errors', as discussed in Part 2. The computer will often indicate the program line as well as the type of error ('SNTX ERROR AT 40'); this, of course, is a great help.
A practical example of computer-aided error correction can be obtained by deliberately inserting mistakes in one of the program examples given in Part 3:

BASIC (PART 4)

```
> 10 REM CALCULATION IF A!
> 20 LET N = 1
> 30 PRINT "ENTER A"
> 40 ILPUT A
> 50 IF A <= 1 THEN 90
> 60 LET N = N + X
> 70 NEXT X
> 80 PRINT A; "I ="; N
> 90 ENO
> RUN
ENTER A

SNTX ERROR AT 40
> 40 INPUT A
> RUN
ENTER A
? 3

NEXT ERROR AT 70
> 55 FOR X = 1 TO A
> RUN
ENTER A
? 3
3

CHAR ERROR AT 80
> 80 PRINT A, "I =", N
> RUN
ENTER A
? 3
3! = 7

8RK AT 90
>
```

After the first RUN command, the interpreter started to translate and axecute the program. At line 40, it found the first error (ILPUT instead of INPUT) and printed a warning. After correcting this error, we tried again: 'RUN'.

Everything now goes smoothly until line 70: NEXT what? Thara is no praceding FOR statement! This is added at line 55, followed (with bated breath?) by RUN. No luck. There's something wrong in line 80. A 'character error'? After studying the description of the PRINT statement as it applies to NIBL, we suddenly realise that the semi-colon is only used at tha end of the complete statament (to suppress the Carriage Return and Line Feed). In NIBL, different sections within the PRINT statement are separated by comma's! (Note that in 'normal' BASIC this would result in a print-out in 'zones'.).

Now, at last, the program runs right through to the end. The 'procedure errors' have been found . . . but the final result is wrong! It's time to take a closer look at the program, as stored in the computer memory:

```
> LIST
10 REM CALCULATION IF A!
20 LET N = 1
30 PRINT "ENTER A"
40 INPUT A
50 IF A <= 1 THEN 90
55 FOR X = 1 TO A
60 LET N = N + X
70 NEXT X
80 PRINT A, "I =", N
90 ENO

> 10 REM CALCULATION OF A!
> 60 LET N = N * X
> RUN
ENTER A
? 3
3! = 6

8RK AT 90
>
```

BASIC (PART 4)

At last! We have locatad the two remaining errors:
a) a spelling mistake in the REM statamant — this has no effect on the execution of the program, but it looks better if it's corrected;
b) in line 60, we had entered N + X instead of N • X.

In complicated programs, debugging is ept to be a more time-consuming process. Some additional PRINT statements can be a greet help — printing out intermediate results, so that the point at which the error occurs can be located more rapidly. For instance, in the example given above an additional '65 PRINT N,X' would have helped to locate the fault. Once the program is running properly, the 'redundant' PRINT statements can be deleted. Another trick is to run the program several times, with different values for the input variablas. This often gives a good indication of the type of error that is occurring.
Overflow and underflow can also cause problems: the result of some intermediate calculetion is too large or too small. In some cases, this can be solved by a minor modification in the calculation. For instance, the statement

LET X = (C/(A-B) + B) * (A - B)

will result in an overflow for A = B, even though the calculation itself is mathematicelly valid. The computer will ettempt to divide C by zero — it hasn't got the sense to multiply by A-B first! This problem can be avoided by rewriting the statement as follows:

LET X = C + B * (A - B).

Sometimes it is useful to stop the computer at some point in the program. One way to do this is to add an INPUT statement: the computer will stop at this statement, print a '?', and continua with the program as soon as a number is entered. In some BASIC dialects a special statement exists for this: STOP . . .

Finally, e list of error indications as known in NIBL should prove useful:

indication	meaning
AREA ERROR	the program memory is 'full'
CHAR ERROR	charactar error in a statement
DIVO ERROR	division by zero
ENO" ERROR	no quotation marks at the and of a 'string'
FOR ERROR	FOR without NEXT
NEST ERROR	too many loops within loops
NEXT ERROR	NEXT without FOR
NOGO ERROR	the line number specified in a GOTO or GOSUB statement does not exist
RTRN ERROR	RETURN without GOSUB
SNTX ERROR	syntax error ('bad lenguage')
STMT ERROR	statement used incorrectly
UNTL ERROR	UNTIL without OO
VALU ERROR	'wrong number' (too large or incorrect format)

If the error indications refer to a progrem line, the line number is specified. 'SNTX ERROR AT 30', for instance.

Extended BASIC

Extended BASIC — also known as 'advanced BASIC' — is a mora flexible dielect of BASIC. Some of the more important additional fecilities can now be explained. It is interesting to note that some of these are also known in NIBL.

Arrays

Use of errays can sometimes simplify (scientific) calculations. An example of a one-dimensionel array is: A(1), A(2), A(3), A(4), A(5). In this case, the array consists of five variebles. These are referred to as the 'elements' of the array. An element is represented by one letter (referring to the variable), followed by a number in brackets. Only numbers between 1 and 10 are normally permitted, elthough in some BASIC dialects 0 can also be used. Alternatively, it is often permissible to include a variable or an expression in brackets instead of e number: A(X) or A(2 + 3). Use of arrays can be illustrated as follows:

```
10 REM EXAMPLE OF ARRAY
15 LET N = 0
20 FOR X = 1 TO B
30 INPUT A(X)
40 N = N + A(X)
50 NEXT X
60 PRINT N
70 ENO
```

In this program, the elements A(1) to A(B) of the array are entered and addad consecutively.

An array can be extended to more than one 'dimension'. A two-dimensional array, for instance, could be as follows:

A(1,1)	A(1,2)	A(1,3)	← row
A(2,1)	A(2,2)	A(2,3)	
	↑ column		

BASIC (PART 4)

This array contains two rows and three columns. It is dealt with in the same way as a one-dimensional array.
If more than 10 elements are required in a one-dimensional array or more than 10 x 10 in a two-dimensional array, a DIM statement can be used to extend the range. For instance,

DIM A (50), B(20,20)

reserves 50 memory locations for the elements of the one-dimensional array 'A' and 20 x 20 (=400) locations for 'B'. A DIM statement is normally included at the beginning of the program.

User-defined functions

It is often the case that a particular program section is required several times in the course of one program. One solution is to include it as a subroutine, as described earlier. A further possibility is to define it as a 'function'. An example:

```
> 10 REM USER DEFINED FUNCTION
> 20 DEF FNA(X,Y) = (X + Y) + (X * Y)
> 30 INPUT A, B
> 40 INPUT "FNA ="; FNA (A,B)
> 50 END
> RUN
? 3,4
FNA = 19
BRK AT 50
```

In this example, a so-called *one-line function* is defined in line 20. After the key-word 'DEF' (for defenition) the name of the function is specified. This always consists of FN followed by one or (in some BASIC dialects) more than one letter(s). After the name of the function, FNA in this case, 'dummy variables' are given in brackets. Finally, after the '=' sign, the necessary formula is defined using these dummies.
When this function is specified in the program (in this example this occurs in the PRINT statement on line 40) the following steps are performed. The values of the specified variables (A and B) are assigned to the dummy variables, so X = A = 3 and Y = B = 4 (line 30 has already been executed!). Then the calculation is performed, as specified in line 20:

FNA(A,B) = FNA(3,4) = (3 + 4) + (3 * 4) = 19

User-defined functions can also run over several program lines. Not surprisingly, these are referred to as *multi-line functions.* An example:

```
10 DEF FNB(X,Y)
20 A= X + Y
30 B= X * Y
40 LET FNB= A + B
50 FNEND
60 PRINT FNB (3,4)
70 END
```

As before, the key-word DEF is followed by the name of the function and the dummy variables (FNB(X,Y)). Then a calculation is specified, using these dummy variables, and a final value (the result of the calculation) is assigned to the function. The multi-line function is concluded by 'FNEND'.

Library functions

Some standard functions ('SIN(X)', for instance) were introduced in Part 3. Extended BASIC dialects usually include some further functions, referred to as 'library functions'. An extensive survey of all the possibilities would be rather pointless, especially since these functions vary from one BASIC dialect to another.
One example, however, is worth mentioning: it is common to most advanced BASIC dialects and it is also known in NIBL. The 'RND' function generates a random number in the range from 0.000 000 001 to 0.999 999 999. This function can be used to generate a random number between 1 and 6, for instance, as follows:

```
10 REM A= RND
20 LET A= RND
30 LET B= 6 * A + 1
40 LET C= INT B
50 PRINT C
60 END
```

BASIC (PART 4)

In line 20, A is assigned a random value between 0 and 1. Then B becomes equal to a random number between 1 and 7, and C is the 'integer' of this. C therefore becomes a random whole number between 1 and 6. In this example, the various steps are spread over several program lines, for clarity. In practice, lines 20 to 40 would normally be run together:

20 LET C= INT(6 * RND + 1)

As mentioned earlier, this function is also defined in NIBL. It is entered in a slightly different way:
RND *(initial value, final value)*
This generates a random whole number (NIBL doesn't recognise decimal fractions!) between the values specified. In NIBL, the example given above corresponds to a very simple program:

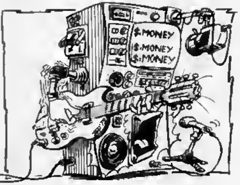
```
10 REM RND IN NIBL
20 C= RND(1, 6)
30 PRINT C
40 END
```

The TAB statement
The TAB statement is an extension of the PRINT statement. It is used to specify the position on a line in the print-out where a number or text should start. An example:

```
10 REM EXAMPLE OF TAB
20 INPUT A, B, C
30 PRINT A;
40 PRINT TAB(15), B;
50 PRINT TAB(30), C
60 END
```

The TAB statement in line 40 specifies that the first digit of 'B' must be printed in the 15th position on the line. Similarly, the first digit of 'C' will be printed in the 30th position. It should be noted that the TAB statement will not cause the printer to 'back space'; if it has already passed the specified position the TAB statement will be ignored. For instance, if the TAB positions in the example given above are reduced to 5 and 10, respectively, and ever larger numbers are printed, the result might be as follows:

```
1    2     3
111  222   333
111112222233333
111111122222223333333
```



Strings
Computers can not only manipulate numbers: texts and (random) groups or 'strings' of characters can also be dealt with. A BASIC dialect with extensive 'string-handling' capabilities can, for example, print out a list of names in alphabetical order.
Variables can be used for string-handling. To distinguish them from numerical variables, 'string variables' are followed by some special symbol – '$', for instance. String variables (also referred to as alphanumeric variables) therefore consist of a letter followed by a symbol: A$, B$, C$, etc. The length of a string is usually limited to 15 characters, including spaces. 'JOHN BULL' consists of nine characters, not eight! The following program is an example of string-handling:

BASIC (PART 4)

```
> 10 REM STRING HANDLING
> 20 READ A$, A, B$
> 30 PRINT A$, B$, A
> 40 DATA "ELEKTOR,", 50, "ISSUE"
> 50 END
> RUN
ELEKTOR, ISSUE 50
BRK AT 50
>
```

In line 20, the alphanumeric variable A$ is assigned the 'value' ELEKTOR, (specified in line 40), A becomes 50 and B$ becomes ISSUE. In line 30, these 'values' are printed in the specified order. Some further examples may serve to illustrate tha possibilities of string-handling:

LET A$ = B$

LET A$ = "TOM" (note the quotation marks!)

IF A$ = B$ THEN . . . .

IF A$ <= B$ THEN . . . .

The final example may seem rather surprising. How can the computer decide whether TOM is larger or smeller than TIM? The thing to realise is that all characters — numbers, letters and other symbols — are stored in the computer memory in ASCII (American Standard Code for Information Interchange). This is e binary code, so each character can also be considered as a (binary or decimal) number:

A = 01000001 (ASCII) = 65 (decimal)

B = 01000010 (ASCII) = 66 (decimal)

etc.

The complete code is listed in Elektor 43, November 1978 (p. 110).

Apparently, A is less than B! This information can be used to list a random collection of words and names in alphabetical order.

The (limited) string-handling capabilities in NIBL differ from the principles outlined above. Since use is made of the fact that memory locations can be addressed directly, it is better to come back to this in the section devoted to 'NIBL statements'

PEEK and POKE

The PEEK and POKE statements are normally limited to BASIC dialects developed for microprocessors.

The POKE statement is used to store a 'byte' (group of binary bits) in a memory location. Similarly, the PEEK statement is used to retrieve a byte from a memory location. The way in which the memory location should be addressed depends on the microprocessor system in question. NIBL, for instance, is designed for the SC/MP system. It uses the @ symbol to address the memory, making for extremely simple and clear PEEK and POKE statements This is described in greater detail under 'NIBL statements'.

Logic operators

A 'logic operation' is a well-defined manipulation of binary numbers. A full description is outside the scope of this series: the principles are explained in detail in 'Digibook'.

Most BASIC dialects (including NIBL) recognise the logic operators OR, AND and NOT. All other operators can be formed by combining these three. A few examples:

IF (X = 1 AND Y = 1) THEN . . . . .

If both conditions are met (X = 1 and Y = 1), the operation or jump specified after THEN is executed.

Y = NOT X

Y is assigned the inverse value of X. Bear in mind that logic operators refer to binary numbers (usually 16-bit numbers in binary systems). To taka a 4-bit example, for clarity: the inverse of 3 (= 0011) is 12 (= 1100).

The order in which the operations are performed in a calculation (see Part 2, 'Arithmetic') is as follows:

OR has the same 'priority' as + and −;

AND has the same priority as * and /;

NOT has the highest priority.

Time-sharing

In time-sharing systems, several users are connected to the same computer (often via telephone lines). Usually, a fairly straightforward BASIC dialect is used but the commands can vary considerably. Full details will always be contained in tha 'time-sharing manual'. One significant difference in many of these systems is that the symbol for 'raising to the nth power' (↑) is replaced by two multiplication symbols: '**'. For instance, 'three squared' (3^2) is entered as 3**2, not as 3↑2.

NIBL statements and capabilities

The Elektor BASIC microcomputer uses NIBL. For this reason, the Elektor BASIC course is rounded off with some further explanation of this particular dialect of Tiny BASIC. Some of the possibilities are related directly to the SC/MP microprocessor; if these are to be used, a general understanding of the SC/MP is therefore required. For this, readers are referred to the series 'Experimenting with the SC/MP' (Elektor, November 1977 . . . March 1978); it is the intention to publish these articles in book form, with some additions, later this year.

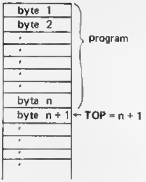
BASIC (PART 4)

The MOD function

This is really a 'library function' that can be used to extend NIBL's number-handling capabilities to include fractions. In a sort of a way, that is . . . MOD (X, Y) calculates the absolute value of the remainder after the division X/Y. A few examples:
$14/3 = 4\frac{2}{3}$. The remainder is 2, the absolute value of 2 is 2, so MOD(14,3) equals 2.
$-25/7 = -3\frac{4}{7}$. The remainder is -4, the absolute value of -4 is 4, so MOD(-25,7) equals 4.
In this way, the statement Y = MOD(3,4) makas the variable Y equal to 3.

The TOP function

The TOP function calls for the address of the first unused memory byta in the current memory. This is aqual to the 'top' of the unused memory area:



This function is useful when looking for a 'vacant slot' for immediate addrassing — in other words, when the operator wants to store data at a specific point in the memory

Pseudo-variables

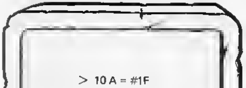
Two so-called pseudo-variables are known in NI8L: PAGE (discussed in Part 2) and STAT. 8oth can be included on either side of an 'equals' sign:
'LET Y = PAGE' makes Y equal to the 'page number';
'LET PAGE = Y' makes the page number equal to Y.
The pseudo-variable STAT refers to the status register in the SC/MP. It can be used to request a print-out of the content of this register:
PRINT STAT
On the other hand, it can be used for 'presetting' the status register to a desired value:
STAT = 15
As usual, the 'value' after the equals sign can be a number, a variable or an expression. This is first converted into a single binary number and then stored in the status register; if necessary, the 'interrupt enable' bit is first cleared.
Obviously, only one byte (8 bits) can be stored in the status register. The low-order byte (least significant bits) is used; the 'high' byte is ignored. The carry and overflow bits will of course be modified as required in the course of the program, so there is little point in presetting them.
The main advantage of STAT is that it gives direct access to I/O lines on the SC/MP chip. It can be used to scan the 'sense' inputs and set the 'flag' outputs.

Hexedecimal numbers

Only whole numbers are recognised in NI8L. So far, we have assumed that thasa must be decimal numbers. NI8L can, however, also work with hexadecimal numbers.

decimal system	haxadacimal system
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	8
12	C
13	D
14	E
15	F
16	10
17	11
18	12
etc.	etc.

A haxadecimal number is preceded by the '#' sign. For example:



BASIC (PART 4)

The indirect operator

In NIBL, the PEEK and POKE statements are replaced by the 'indirect operator' @. The @ symbol is followed by an address (given as e number, a variable or an expression); it refers to the contents at that address. In this way, the contents et eddress 515 cen be called up and assigned to a variable, for instence:

LET X = @ 515

If the contents of memory location 515 were 25, X is now equal to 25.

By placing the indirect operator before the 'equals' sign, data can be stored in memory:

@ 515 = 31

stores the valua 31 at mamory address 515. In the same wey, data can be copied from one memory location to another. Copying '515' into '530', for instance:

@ 530 = @ 515

Although arrays are not known as such in NIBL, the indirect operator can be used to obtain the same result. Provided the numbers in the arrey ere all positive whole numbers between 0 and 255 (8 bits!), any element in an M x N array can be defined with a little care. To take a 4 x 5 array (M = 4, N = 5) as an example:

0,0	0,1	0,2	0,3	0,4	
1,0	1,1	1,2	1,3	1,4	←Row I
2,0	2,1	2,2	2,3	2,4	
3,0	3,1	3,2	3,3	3,4	

M Rows

Column J

N Columns

Assuming that this array is to be stored in memory from address A (and providad there is enough room 'below' A to fit the whole array!), the elements will be stored at the following addresses:

adress	element
A	0,0
A + 1	0,1
A + 2	0,2
A + 3	0,3
A + 4	0,4
A + 5	1,0
A + 6	1,1
A + 7	1,2
A + 8	1,3
etc.	etc.

Element (1,2), for instance, is stored at address A + 1 x 5 + 2 = A + 7. In general, the position of any element (I,J) in an array (M,N) can be specified es @ (A + I∗N + J). Admittedly, this is more clumsy than specifying A(I,J) – but it does work.

String handling

Tha string-handling capabilities in NIBL are rather limited, but 'a little gumption (on the part of the programmer) goes a long way'. Note that the string-handling methods in NIBL are completely different from those described earlier!

The statement INPUT $ *address* can be used for entering a string. For example:

80 INPUT $ 6150

When this statement appears in e program, the computer responds by printing a question mark – as with a normal INPUT statement. A string (consisting of a series of characters) can now be entered, followed by CR (carriege return). This string of characters is stored in consecutive memory locations, starting at the specifiad address (6150 in the exempla given above). No quotation marks are required (these would simply be stored as part of the string). Note that the final CR is also stored at the end of the string.

It is also possible to specify an address in hexadecimal: INPUT $#180A.

Another way to store a string in memory is to key in:

$*address*= THIS IS A STRING

Having stored a string, further manipulations, checks etc. can be carried out by means of the indirect operator @. It is also possible to transfer it from one series of mamory locations to another:

$*destination address* = $*present location*

This causes all characters in the original string (starting at the specified 'present location') to be copied one at a time into the memory locations starting at the specified 'destination address'. The CR is also copied, and recognised as the end of the string. Note that tha series of eddresses used as 'destination' should on no account overlap with the 'source addresses'. This can have disastrous results, even to the point whare the whole current page of memory is erased!

QUESTIONS

1. What is the difference between a procedure error and an execution error?
2. Is a DIM statement always required when using arrays?
3. What is the difference between the user-defined functions FNA(X,Y) and FNB(X,Y) on page 32?
4. Does a TAB statement always produce a print-out at the desired position on the line?
5. What error indication will be produced in NIBL if the following lines are entered?
   10 A = B ; D = 1
   20 C = D/(A-B)
6. How is a string variable represented?
7. What is the decimal value of the hexadecimal number 1B?

BASIC (PART 4)

GLOSSARY

Alphanumeric variable
See *string variable*

Bug
Error in program.

Byte
Binary code consisting of several bits (usually 8).

Dummy variable
Variable specified in a user-defined function.

Execution error
Error that occurs when a program is running.

Expression
(Mathematical) operation.

Hexadecimal
Number system with base 16. The digits run from O to F.

Indirect operator
The symbol @, used in NIBL when addressing memory locations.

Initialisation
The specification of initial conditions (initial values of variables etc.).

Library functions
Additional functions in some Extended BASIC dialects.

Logic operator
Logic operators, such as AND, OR and NOT, are used for binary logic operations.

Procedure error
Errors that are detected by the compiler or interpreter: mistakes in the BASIC language used.

String
A group of characters (letters, numbers, etc.).

String variable
Also referred to as *'alphanumeric variable'*, this is a variable to which a *string* can be assigned instead of a (numerical) value.

User-defined function
A function defined by the programmer.

ANSWERS TO QUESTIONS IN PART 3

1. It is permissible to enter more information in a data block than required. The redundant information remains unused. Storing insufficient data will result in an error indication.
2. The REM statement is used to add 'reminders' that will prove useful at a later date, when a program listing is requested.
3. If one runs short on memory space, the number of REM statements may have to be reduced.
4. In general, the use of 'jump' statements will have little or no effect on the execution time, they merely reduce the amount of memory space required. If a compiler is used, the initial translation may require less time.
5. If negative steps are specified in the FOR . . . TO . . . STEP . . . statement, the final value should be less than the initial value.
6. Subroutines simplify programming and use less memory space.
7. Jumping out of a FOR . . . NEXT . . . or DO . . . UNTIL . . . loop almost invariably causes problems, since the computer will still be looking for another NEXT statement or waiting for the UNTIL requirement to be fulfilled.

ANSWERS TO QUESTIONS IN PART 4.

1. Procedure errors are those where mistakes are made in the BASIC language used (keying errors and the like). These errors are usually detected by the computer, unlike execution errors: the latter refer to cases where a program is keyed in that *could* be correct, but isn't what the programmer intended. The program will run, but it will come up with the wrong answers.
2. A DIM statement is only required for arrays larger than 10 or 10 x 10 (one- or two-dimensional, respectively).
3. Both user-defined functions will perform the same calculation. The only difference is the number of program lines – and thus the amount of memory space required.
4. A TAB statement will produce a print-out at the desired position, unless this has already been passed. In that case, the TAB statement will be ignored.
5. In line 10, A is made equal to B. Therefore, in line 20 D is divided by zero. In NIBL, this results in the error indication DIVO.
6. A string variable is represented by a letter followed by a dollar sign: A$.
7. The hexadecimal number 1B corresponds to the decimal number 27. This can be found by extending the table given on page 35.

BASIC (PART 4)

Summary of symbols, statements and commands

Symbols

10 *statement*	A number at the beginning of a program line indicates that the following *statement* is part of the program.
:	Colon, used as separation between statements, if more than one statement is to be printed on the same line.
E	Symbol used in scientific notation. The number following the E defines the number of places over which the decimal point must be shifted.
+	addition — mathematical operations
—	subtraction — mathematical operations
*	multiplication — mathematical operations
/	division — mathematical operations
↑	involution — mathematical operations
**	Symbol often used in time-sharing systems instead of ↑.
=	equals — comparison symbols
< > or > <	is not equal to — comparison symbols
>	greater than — comparison symbols
<	smaller than — comparison symbols
> = or = >	greater than or equal to — comparison symbols
< = or = <	smaller than or equal to — comparison symbols
AND, OR, NOT	Logic operators
A .. Z, A1 ... Z9	'names' of variables
$	Symbol used in a string variable. For example, A$ = ELEKTOR
A(3)	Element in a one-dimensional array A.
B(3,5)	Element in a two-dimensional array B
#, >, :	These so-called 'prompt' symbols can be printed by the computer at the beginning of a line.

Special keys

←	Backspace. This key is used when correcting keying errors.
BREAK	A key on the terminal that is used to stop the program.
CR	Carriage Return (return to beginning of line in display).
LF	Line feed (move to next line in display). This is normally carried out in conjunction with **CR** (carriage return).

Input/output statements

INPUT *variable(s)*	This statement causes the computer to request keyboard entry of value(s) that must be assigned to the specified *variable(s)*.
READ *variable(s)* READ *variable(s)* • DATA *data, data* ...	The *variable(s)* listed after the READ statement(s) are assigned the *data* value(s) given after the DATA statement
RESTORE	This statement causes the data block to be re-used from the beginning
PRINT *"TEXT"*	This statement causes the *text* included in quotation marks to be printed out
PRINT *2A + 3*	The expression (e.g *2A + 3*) following 'PRINT' is carried out and the result is printed.
PRINT ... ; ... ,	The semi-colon can be used to separate groups of symbols and/or expressions to be printed. A semi-colon at the end of the statement causes a following PRINT statement to be carried out on the same line
PRINT ... , ...	By using commas, the print-out can be divided into 'zones'.
TAB*(position)*	This causes a print-out at the specified *position*.
PEEK*(decimal address)*	This is a request for a print-out of the decimal value stored in the *decimal address* specified.
POKE*(decimal address and value)*	This stores the *decimal value* in the specified *address*.
REM *text*	The specified *text* appears in a listing, but has no effect on the program.

Various statements

LET	Statement, by means of which a value can be assigned to a variable.
RND	Library function, generates a random number.
DEF FNA(X,Y) = *expression*	One-line function. FNA is the function name; X and Y are the dummy variables used in the *expression*.
DEF FNA(X,Y) • • • • LET FNA = *expression* FNEND	Multi-line function. The statements between DEF FNA(X,Y) and FNEND describe the function to be calculated; the *expression* assigns the calculated values to the correct dummy variables
DIM A(50) DIM B(20, 30)	These statements reserve additional memory space for large arrays.
END	This statement indicates the end of a program.

BASIC (PART 4)

Jump/loop/subroutine statements

GOTO *line number*	This causes a jump to the specified *line number*.
IF *comp.* . . THEN *line number* . . . THEN *statement* . . GOTO *line number*	If the result of the *comparison* after IF is true, the computer 'jumps' to the specified *line number*; otherwise the program is continued on the next line. In NIBL, a *statement* can be given instead of a line number; if a jump to a line number is required 'GOTO' must be used instead of 'THEN'.
FOR . . . TO . . . STEP • • • • • • • • • • • • NEXT . . .	A 'running variable' is assigned an 'initial value', both as specified after FOR (e.g. FOR *A=1*). The statements between FOR and NEXT (the 'FOR-NEXT block') are then carried out; the running variable is increased by the specified step (e.g. STEP*15*), after which the FOR-NEXT block is repeated; and so on until the 'final value' specified after TO (e.g. TO*90*) is reached or exceeded. If no STEP is specified, the step is automatically taken as +1.
DO • • UNTIL *comp.*	This 'loop' is known in NIBL. The statements between DO and UNTIL are repeated until the *comparison* specified after UNTIL becomes 'true'.
GOSUB *line number*	This causes a jump to the subroutine that starts at the specified *line number*.
RETURN	Last statement in a subroutine: it causes a jump back to the main program.

Commands

CLEAR	This command can be given before re-running a program.
RUN	This command causes the computer to execute the program.
STOP	This stops the program execution; the program is continued when the CR key is operated.
LIST	This command initiates a print-out of the program.
LIST n	The program is printed out from line n.
List n,m	The program is printed out from line n to line m.
SCRATCH, DELETE, PURGE, NEW	These commands cause the program memory to be erased.

Special NIBL statements and symbols

#	This symbol indicates that the following number is in hexadecimal
@	Symbol for indirect operator.
$	Symbol used in a string variable. Note that in NIBL this preceeds the variable. For example: $A = ELEKTOR.
PRINT . . . . :	The comma is used to separate groups of symbols and/or expressions to be printed. A semi-colon at the end of a PRINT statement will result in the following PRINT statement being carried out on the same line
NEW *n*	This command (in NIBL) erases page *n* in the memory, preparatory to storing a new program.
PAGE *n*	This command (in NIBL) causes the computer to jump to page *n*, readying the computer to store (or modify) a program there
MOD *(X,Y)*	A library function that calculates the absolute value of the remainder after a division *(X/Y)*.
STAT	Pseudo-variable, used for reading or modifying the contents of the status register in the SC/MP.
TOP	Library function, requesting the decimal value of the first unused memory location in the current memory
LINK *(address)*	The program is continued in machine language, from the *address* indicated. The address must be given as a decimal number.

Error indications as known in NIBL

AREA ERROR	the program memory is 'full'
CHAR ERROR	character error in a statement
DIVO ERROR	division by zero
END" ERROR	no quotation marks at the end of a 'string'
FOR ERROR	FOR without NEXT
NEST ERROR	too many loops within loops
NEXT ERROR	NEXT without FOR
NOGO ERROR	the line number specified in a GOTO or GOSUB statement does not exist
RTRN ERROR	RETURN without GOSUB
SNTX ERROR	syntax error ('bad language')
STMT ERROR	statement used incorrectly
UNTL ERROR	UNTIL without DO
VALU ERROR	'wrong number' (too large or incorrect format)

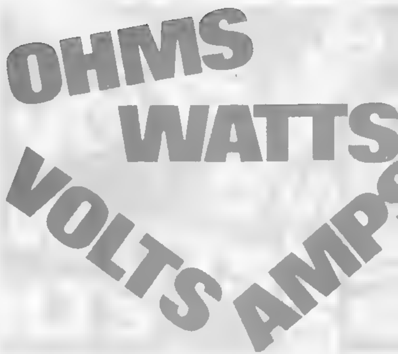
BASIC (PART 4)

output power nomogram

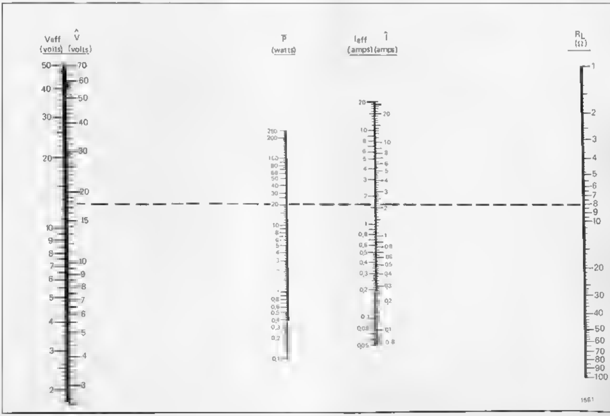
This nomogram has been prepared by the editors in response to regular requests from readers. When the required output power and the loudspeaker impedance are known, the nomogram can be used to find the associated voltage and current. It can actually be used as soon as any two of the variables are known—to find the remaining set.

$\overline{P}$ is the continuous (sine wave) power
R_L is the impedance of the loudspeaker
V_{eff} is the effective (RMS) output voltage
$\hat{V}$ is the peak value of the output voltage swing
I_{eff} and $\hat{I}$ are the effective and peak values of the current swing

The power supply must deliver at least $2\hat{V} + 4$ volts (measured to the lowest edge of any ripple waveform). For a stereo amplifier, it must be rated for at least I_{eff}.
"Music power"—depending on the power supply and the output stage heat sink—can be anything from 1 to 20 x $\overline{P}$ ..!



Example (see dashed line):
For 20 watts into 8 ohms we find $\hat{V}$ = 18 volts and I_{eff} = 1.6 amps. So the power supply must be rated to deliver 2 x 18 + 4 = 40 volts at 1.6 amps.



When Sinclair introduced a new Spectrum last autumn, they at last acknowledged that the poor keyboard on the old Spectrum was allowing the competition to take over slices of what had been a safe Sinclair market for some time. It is gratifying to see that they have finally decided to right this abomination.
The new machine is type-coded Spectrum+ and is a much more usar-friendly unit that can stand comparison with any other computer in its prica class. Its faint resemblance with its bigger brother, the QL, gives it an impression of solidity. Inside it has remained very much tha sama. And why not? Apart from its miserable keyboard, the old Spectrum, with its wide-ranging BASIC and its really useful graphics, was one of the best-value-for-money home computers around.

new keyboard for Spectrum . . .

. makes programming easier

The noteworthy aspects of the new keyboard are not just the much easier to operate keys, but also a large number of single-key operations, which on the old Spectrum were only available by pressing more than one key at the same time. This makes programming and particularly editing a lot easier.
If you have an old Spectrum and would like the facilities of the Spectrum+, without buying the new version, the circuit suggested here is for you.

The new keyboard

The layout of the new keyboard is shown in figure 1. All keys of the old keyboard have been retained, and new ones have been added as follows.
Top row. TRUE VIDEO; INVERT VIDEO; BREAK
Second row: DELETE; GRAPHIC
Third row: EXTEND MODE; EDIT
Fourth row: FULL STOP
Bottom row: SEMICOLON; DOUBLE QUOTATION MARKS; four CURSOR keys; COMMA
Other novelties are that the CAPS SHIFT and SYMBOL SHIFT have been duplicated at the left-hand and right-hand side, and that the SPACE bar has been centred and made wider.

Operation

How Sinclair has solved the problem of the additional keys is of little interest here, as we had designed the present circuit before the Spectrum+ had been announced.
Figure 2 shows what the Spectrum looks like when the keyboard cover and the conductive silicone rubber sheet directly underneath it have been removed. The circuit diagram of the 8 by 5 scanner matrix that has become visible is shown in figure 3. Each switch shown in the diagram represents a key contact. The numbering is arbitrary and has no special significance. This matrix is also present i our proposed new keyboard, with the difference, however, that the touch-keys have been replaced by full-size keys: this gives a feel that is similar to that of a typewriter and entry speed is, therefore, much higher and much more reliable. The second CAPS SHIFT and SYMBOL SHIFT keys have simply been wired in parallel with the original ones.
The old keys that need no additional or new electronic circuitry are called 'A' keys in figure 4.
The key contacts in figure 3 are controlled without any problem by CMOS switches. These switches are at the heart of our design: any of the additional keys operates two or more CMOS switches simultaneously. If we now consider the 'B' keys in figure 3, we see that all functions controlled by contacts S41...S61 have this in common: the CAPS SHIFT key must be pressed at the same time as another key. It is therefore necessary that each key contact is connected to a CMOS switch

1

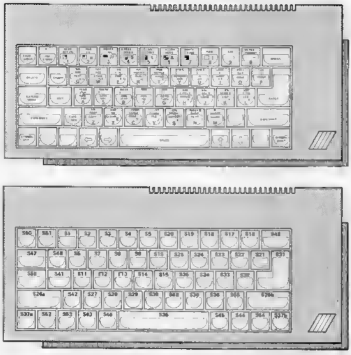


Figure 1. Layout of the proposed new Spectrum keyboard, with all the old keys retained in their original position. The newly added keys do not represent new instructions or functions, but enable single-key operation where before more than one key had to be pressed. Note the enlarged ENTER key, the duplication of the CAPS SHIFT and SYMBOL SHIFT keys, and the much better location of the SPACE bar.

2

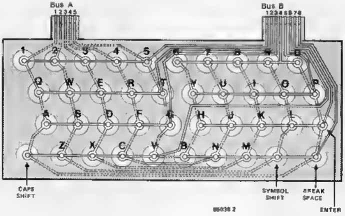


Figure 2. Once the lid and conductive silicone rubber sheet have been lifted off the keyboard, the scanner matrix becomes visible. Connections A1...A5 and B1...B8 are identical to those on the circuit diagram. The new keyboard is connected in exactly the same way as the old one: this is best done with ribbon cable which is soldered at the underside of the computer board in place of the obsolescent socket.

(ES) that is connected in parallel with the CAPS SHIFT contact on the 8 by 5 matrix, and to a switch that is connected in parallel with the relevant original key contact. For example, to carry out the operation 'edit', it is necessary to actuate the '1' and the CAPS SHIFT keys. The former is effected by ES2, and the latter by ES1 (which interconnects the A1 and B1 bus lines). All CMOS switches, except those that operate the CAPS SHIFT and SYMBOL SHIFT keys, are controlled via delay networks. This is necessary to ensure that at all times the CAPS SHIFT or SYMBOL SHIFT contact is operated first. The diodes are decoupling devices and should not be omitted, otherwise erroneous operation may ensue.
Keys S52...S55, the 'C' keys, operate similary to S41...S51, except that here the

new keyboard for Spectrum.

3

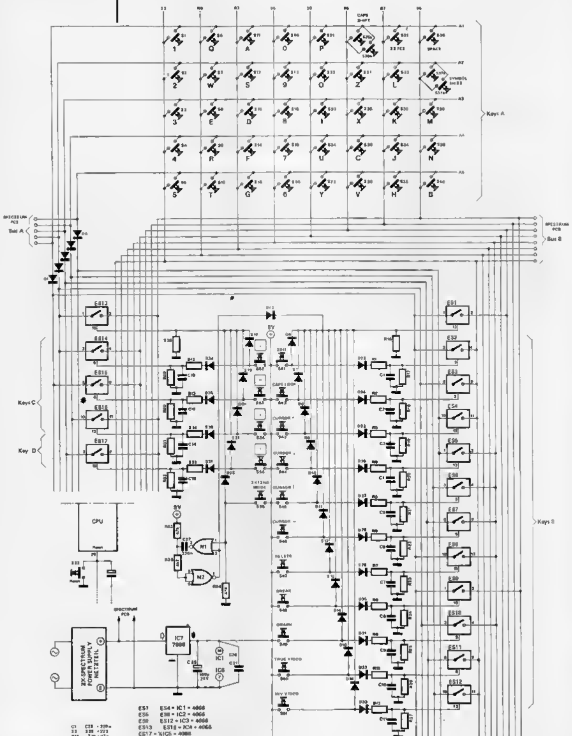


Figure 3. The circuit diagram of the new keyboard. The CMOS switches and associated components may be mounted on a eurocard. S57 is a RESET switch.

4

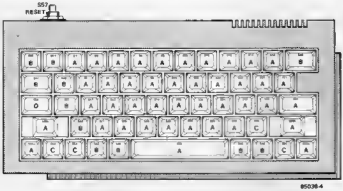


Figure 4. Identifying the single key switches by A. . .D makes it clear that from a purely electronics point the keyboard may be considered in four parts: (1) keys A, which are unaltered; (2) keys B, which control functions for which on the old keyboard the CAPS SHIFT key had to be operated at the same time; (3) keys C, which control functions for which on the old keyboard the SYMBOL SHIFT key had to be pressed at the same time; (4) key D, which actuates either the lower case (red) functions or the upper case (green) ones

SYMBOL SHIFT key instead of the CAPS SHIFT key is actuated (by ES13). With the help of figures 1 and 3 it is possible to realize any function you want by using further keys and CMOS switches. The proposed circuit follows the keyboard of the Spectrum+, however, although many of you may find the VIDEO keys rather an unnecessary luxury which are better replaced by COLON (:) and SOLIDUS (/) respectively.

Extend mode key

The EXTEND MODE key, the 'D' key, is somewhat different from those described sofar. To jog your memory: if you want to use instructions printed in green (old keyboard), it is necessary to press the CAPS SHIFT and SYMBOL SHIFT keys simultaneously, release them, and then press the appropriate command key. If you want to use red instructions, the situation is even worse: you then have to press the SYMBOL SHIFT key, holds this down, and then press the instruction key. These problems are resolved by the EXTEND MODE key, S56. This switch is connected to ES1 (CAPS SHIFT) via monostable N1/N2, and to ES13 (SYMBOL SHIFT) via D22. When S56 is held down when the instruction key is pressed, the monostable formed by N1 and N2 opens the CAPS SHIFT key contacts after a very short time: the 'lower case' (red) instructions are then available. If, however, S56 is pressed and then released before the instruction key is pressed, the upper case (green) instructions become available.

Power supply

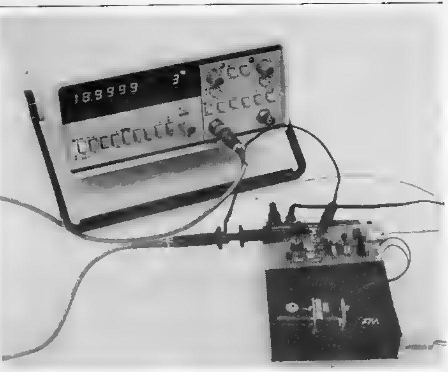
The keyboard may be operated from the Spectrum power supply. It is, however, recommended to use an 8-volt regulator, because a level of 5 V is pretty close to the lower limit of the CMOS ICs. It is further recommended to solder a capacitor of about 220 nF between the supply pins of each IC. The circuit has been working faultlessly in our laborstories for some months, so that we cannot foresee any problems.

Construction

As keytops with the original Sinclair inscriptions do not appear to be commercially available, it is best to use keyboard switches with transparent keytops. These tops snap on to the switch and consist of two parts. The lower part may be engraved, marked with Letraset, or a piece of printed card may be placed on it, so that when the transparent top part is snapped on, the key-top appears to have a printed legend. Character set transparencies are available from many retailers. The keyboard may be constructed on a printed circuit board (we do not offer one with this project, so it will have to be made by yourself) or on veroboard. Dimensions are about 400 × 150 mm, so unless your retailer stocks this size board, you may have to make one from two. Because of the pressures exerted on a keyboard, mechanical strength of the boards is, of course, vital.

Finally. . .

. . a tip for all d.i.y. programmers. In many games, it is possible to interrupt with the aid of the cursor keys. On the old Spectrum keyboard, the cursor arrows were located above the figures 5. . .8, and it was, therefore, simple to interrupt the game or other program by writing these figures in conjunction with the INKEY $-function. For instance,
IF INKEY $ = "8" THEN. . .
Because the cursor keys on the new keyboard are separate, and have no symbol, but only a function character, it is necessary to use the **code** of the cursor function if you want to control the computer with the cursor keys. This can be ascertained from the handbook. The corresponding instruction will now be something like
IF CODE INKEY $ = 10 THEN. . .



a simple test-instrument calibration aid

19 kHz precision calibrator

Test instruments are essential for any serious laboratory, be it 'professional' or a hobbyist's workshop. Bad test equipment, on the other hand, can be worse than no equipment at all as it gives wrong impressions that are likely to be taken as 'truths'. Accuracy is always a point of doubt with home-made test instruments and for this reason we were very careful to give a detailed test procedure for our recently-published frequency counter. In hindsight, however, it occurred to us that one point could be considered as an example of 'Catch 22': a good frequency counter was needed to calibrate the crystal oscillator. To remove this difficulty we came up with a simple, but accurate, circuit.

A test instrument must be reliable and accurate and, in general, the more you are prepared to pay the more of these two qualities you can expect. Home-made equipment is somewhat of an unknown quantity in this respect and must in some cases be referenced to other proven test instruments. This is the case with the frequency counter circuit, whose crystal oscillator must be calibrated with reference to a frequency meter that is known to be accurate. Fortunately, there is a way out of this 'Catch 22' situation, and all that is needed is the small circuit shown here and a cheap FM transistor radio.

19 kHz from the radio

The frequency of 19 kHz was not simply chosen at random for this circuit. This is, in fact, the frequency of the pilot tone transmitted in FM radio signals. In the case of good quality or hi-fi tuners, however, this frequency is very effectively suppressed after the stereo decoder so they are unsuitable for our purposes.
The loudspeaker or earphone output of a mono FM radio is connected directly to the input of our circuit. Band-pass filter R1/C1/L1/C2 removes all unnecessary parts of the signal and this is then amplified by T1 and fed into pin 3 (INPUT) of IC1.
The 567 (IC1) is a phase locked loop (PLL) used as a tone decoder. The functions of each of its pins are indicated in the diagram of figure 1. The PLL is tuned to a particular frequency by means of the external timing components connected to pins 5 and 6. The values indicated set IC1

1

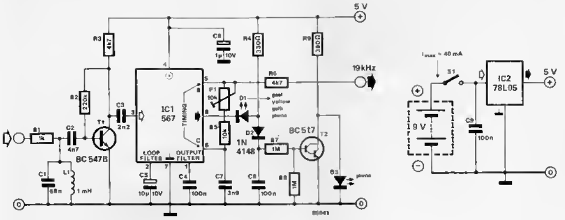


Figure 1. With a cheap radio connected to its input this circuit is a very effective 19 kHz calibrator. Depending on the actual radio used it may be possible to draw the power supply of 9 V d.c. from the radio's batteries or power source. The circuit's current consumption is about 40 mA with both LEDs lit. If the radio cannot supply it the 9 V can be provided by separate batteries.
A different value of inductor can be used for L1 but the capacitance of C1 will then have to be re calculated from the formula $C1 = 1/4\pi^2f^2L$, where f = 19 kHz, of course.

to a frequency of 19 kHz, of course. The band-width of this 'lock frequency' is determined by C5, while C4 serves to suppress the effects at the output of spurious input signals outside the lock range.

When IC1 detects an input signal within the frequency range selected it locks onto it. The 567's output then goes low so the yellow LED lights. If the PLL is not properly locked onto the input signal, or if the input signal does not remain within the lock range for a while, capacitor C8 is partly charged so the green LED cannot turn on. As soon as the PLL is properly locked onto the 19 kHz signal T2 is turned off and the two LEDs light together. The signal output from pin 5 of IC1 is then a square wave with a frequency of exactly 19 kHz.

Construction, calibration and use

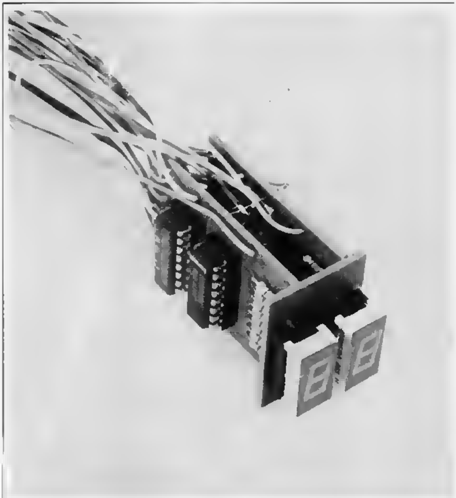
The circuit should be constructed carefully bearing a few points in mind. Keep all wiring as short as possible and make sure to use thick cable for the ground and positive voltage supply lines. It is also very important to connect capacitor C6 as close as possible across IC1's V_{CC} and ground pins (4 and 7).

Connect the input of the circuit to the loudspeaker (or earphone) output of an FM mono radio and tune into a strong transmitter broadcasting in stereo. Carefully set preset P1 so that the green LED just lights at the input signal's minimum amplitude.

To use the circuit increase the radio's volume control slightly and tune into a different transmitter. The yellow LED lights when IC1 locks. The green LED will only light if there is not too much noise on the signal. Increase the volume setting or/and tune to a different transmitter until the green LED lights continuously. If the green LED lights when the radio is tuned to a number of different transmitters the circuit is working correctly and the output signal then has a frequency of 19 kHz.

There are a few final points regarding using this circuit. First of all, it is advisable to tune the radio to a 'quiet' transmitter such as a classical music program. There is less chance of noise or interference on such a signal than would be the case with a contemporary music program. The second point concerns the accuracy of the 19 kHz pilot tone. The EBU (European Broadcasting Union) norm states that this must be 19 kHz ± 2 Hz. This is quite a tight tolerance but in most cases the signal will be even closer to 19 kHz. The measurements we made showed that the signal was, in fact, accurate to within ± 0.001%.

This test circuit was designed as an aid to setting up the microprocessor-controlled frequency counter described in the February 1985 issue of *Elektor* but it can also be used to test and calibrate other laboratory equipment. In the case of the frequency meter simply let it warm up first and then measure the 19 kHz with the A input. Trim the preset in the oscillator until the display shows '19.0000 KHZ'. By the same token this circuit need not only be used for calibration purposes. It could be used in any application where a very accurate 19 kHz signal is required but because a radio has to be connected to its input it cannot be considered as a permanent fixture.



A de Kock

Counter circuits have long been a tradition in Elektor. It started with one of the very first issues way back in 1975 and has continued ever since. In spite of this we still get regular requests for counters to do this and counters to do that. To satisfy all these, we have developed a counter circuit that can:

- **count upwards and downwards;**
- **be used with a variety of displays: LED, LCD, FD, and others;**
- **store the counter contents;**
- **preset the counter position.**

versatile counter circuit

upwards or downwards, for LED or LCD, for common-cathode or common-anode

The circuit diagram in figure 1 shows nothing really surprising: decoder, IC1; counter, IC2; and seven-segment LD1. The surprises are contained within the ICs! IC2 is a synchronous BCD upwards and downwards counter whose content may be preset. The presetting function is asynchronous.

BCD counter ICs generally contain four bistables and a number of gates with which the required function can be arranged. Asynchronous operation means that one of the bistables toggles when the clock at its input changes. In other words, each bistable is clocked by its predecessor. Operation is synchronous when the output level of a bistable changes when the output level of the preceding bistable goes logic high and a fresh clock pulse arrives at its input. This pulse is provided simultaneously to the clock inputs of all the bistables. With this arrangement the result does not have to wait for the clock to be provided to the

1

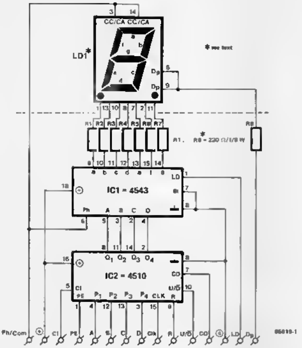


Figure 1. A counter element consists of a counter IC (synchronous BCD upwards and downwards counter with asynchronous preset), a decoder IC (BCD-to-seven-segment decoder with latch and display driver), and the LED display.

last in a long line of bistables.
For instance, in an eight-stage counter operating in the asynchronous mode, 32 bistables have to be clocked before the result is indicated. In synchronous mode, the result is known immediately.
There are therefore seven terminals needed: two for the supply voltage, U_b (3...18 V), four for the outputs of the bistables (Q1...Q4), and one for the clock input (Clk) which is internally connected in parallel to all bistables. Then there is an input for signal U/$\overline{D}$, which gives the command to count upwards or downwards. And, of course, there is a reset (R) input.
Preselection of the counter position is carried out via inputs P1...P4. The lowest value bit accords with P1 (and subsequently with output Q1!). Preselection is evaluated when input PE is logic 1, independent of the clock signal; this mode of operation is therefore asynchronous.
The two remaining terminals of the counter IC are CI (carry in) and CO (carry out). It is these terminals that make the circuit of figure 1 into a proper functional element, for they make the connection between the previous and the following counter elements. The counter elements can thus be connected in cascade by connecting the CO of the previous element to the CI of the next.
The other IC, a BCD-to-seven-segment decoder with latch and display driver, is similar. A glance at the pin-out shows that seven outputs are available for display segments a...g. Then it has four inputs for the BCD information, A...D, and two terminals for the supply voltage of 3...18 V.
The interesting pins here are Ph, BI, and LD. Pin LD is normally logic high; when it

2

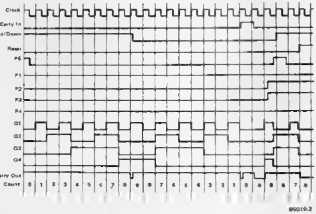


Figure 2. Operation of the counter IC is more easily understood with the aid of this timing diagram.

3

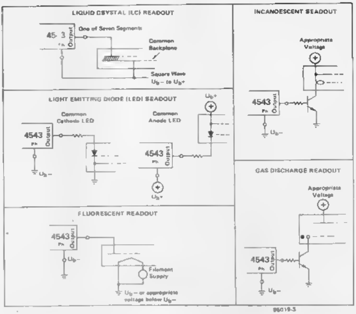


Figure 3. Connections to various display readouts. When a liquid crystal display is used, the decimal point should be connected via an XOR gate, the inputs of the gate to Ph and Dp, and the output to Dp

Table 1
R = reset
CI = carry input
CO = carry output
PL = parallel load
U/D = up/down
Clk = clock
PE = pulse enable
Dp = decimal point
Ph/Com = common anode/ common cathode
LD = latch disable

Parts list

Resistors:
R1 ... R8 = 220 Ω/1/8 W

Semiconductors:
LD1 = MAN4410A gr, MAN4610A o; MAN4910A r; MAN4810A y (all General Instrument) for Siemens or Hewlett Packard types see text
IC1 = MC14543B (Motorola)
IC2 = MC14510B (Motorola)
Printed circuit board 85019

4

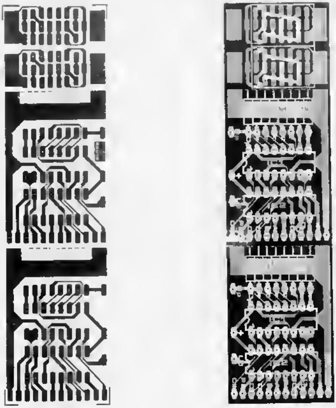


Figure 4. The printed circuit board can house two counter elements; it should be cut into two, with one part containing the display section, or into three if only one element (and display) is required.

goes low, the information at the BCD inputs is stored in the IC, and the memory content is subsequently fed to pins a...g. Pin BI is normally logic low. When it goes high, all pins a...g are low. The segment outputs are also logic 0 when a number greater than 9 (in BCD code) is present at the input pins.

The junction of pin Ph is best seen with reference to figure 3, which shows the various connections to the display readouts. We have opted for the LED

5

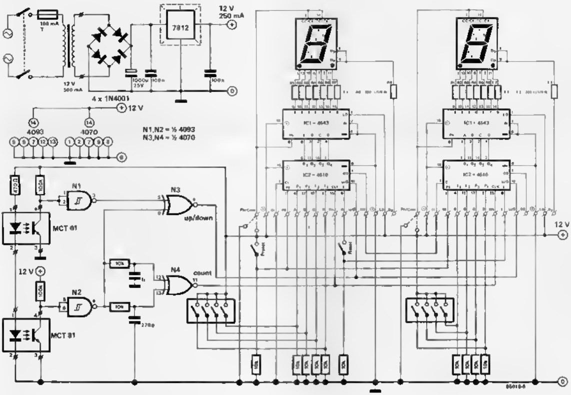


Figure 5. Example of an adaptation of the *revolution counter* described in the September 1981 issue of *Elektor*. Note that only two of the six required counter elements are shown.

display, because this is not only the most economical but is also the most suitable for use with this particular IC.

Construction

A printed circuit for two counter elements is shown in figure 4. The board should be cut into two, with one part containing the display, or into three if only one element (and the display) is required.

The boards are put together as shown in the photograph: that containing IC1 and IC2 is at right angles to the display board. The earth planes of the boards must be soldered together; this makes it imperative that the boards are cut absolutely straight. Additional stability is provided by resistors R1 . . . R8 being soldered to two boards!

Most terminals are located at the short edge of the board(s); only Dp, +, and LD are at the long side. This was arranged so that when several boards are in cascade, they can be placed side by side on a prototyping (vero) board. To ensure sufficient stability, the terminals at the long side, and possibly also the Clk terminal, must be soldered to the vero board. Our prototype which was assembled in this manner proved more than adequately stable. Do not forget to connect the CO terminal on one board to the CI terminal on the next.

A final word about the LED display. Since IC1 can provide a segment current of only 10 mA, it is advisable to use the General Instrument type given in the parts list. Siemens and Hewlett Packard types draw rather more current, about 15. .25 mA, for the same light intensity. When these types are used, it is therefore advisable to buffer each of the segment outputs as, for instance, in figure 3 (incandescent or gas discharge readout).

Take care to solder the Ph/Com terminal with correct polarity.

G. Fossan

Many electronic components may only be fitted into a circuit in one way: the polarity must be correct in other words. Diodes, electrolytic capacitors, ICs (to name but a few) are marked to show what is their correct polarity but transistors do not have any such indication. Knowing the type of the transistor in question it is, of course, a simple matter to look at the data sheet and find out which pins correspond to emitter, collector and base. If you do not have the data sheet to hand, however, this makes matters somewhat more difficult.

transistor unitester

a universal all-in-one transistor connection tester

Two things are of vital importance when a transistor is to be used in a circuit, namely which pins correspond to emitter, collector and base and whether it is an NPN or PNP transistor. The transistor's data sheet gives this information but the chances are that you will not have the appropriate data sheet when you need it. A collection of data sheets would seem to be the answer but a much better solution is to make a transistor connection tester like the one shown here.

A switching transistor

The most striking thing about the unitester circuit shown in figure 1 is its simplicity. The component under test, T_T, is used as a switching transistor. The base current is varied with P1 until the transistor switches on and causes two of the LEDs to light. Which LEDs light (D1/D2 or D3/D4) depends on whether T_T is NPN or PNP. (This is defined by the position of S1.) The intensity of the LEDs at a particular position of P1 gives an indication of the transistor's current gain.

The test procedure

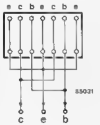
In our prototype we fed the 'b', 'c' and 'e' terminals from the circuit to an IC socket, as the diagram indicates. We found that this simplified using the circuit as all possible connection orders are catered for. To try a different bce layout the transistor's pins are each moved one hole further. A transistor is tested as follows:

- Plug the pins of T_T into the IC socket in any order but making sure that each fits into a different hole — one into 'b', one into 'c' and one into 'e'. Turn P1 completely around and back, then switch S1 over. If one of these two operations causes two of the LEDs to light simultaneously the pin stuck into the 'b' position is the transistor's base. In this case both LEDs will light when P1 is at one end of its travel and both will be off at the other end of the travel. Any other indication on the LEDs indicates an incorrect connection so T_T's pins should be changed around until the right indication is found. If none of the possible combinations gives the correct indication either the transistor is faulty or the component in question is not a transistor.
- Having found the base, the emitter and collector must now be determined. The base current is set with the potentiometer so that moving P1 slightly gives a clearly visible change in the light intensity of the LEDs. The LEDs are set to 'medium' brightness and the collector and emitter connections are then swapped. If the LEDs burn more brightly than before the 'c' and 'e' connections are now correct. If, on the other hand, the LEDs become dimmer the previous arrangement was correct.

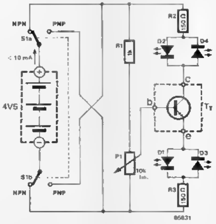
Final points

The circuit is easily constructed on a piece of Veroboard and can be connected to a suitable d.c. power source (batteries will be sufficient). The voltage supply should be about 4.5 V, but must never be greater than 6 V. There is no actual need to use an IC socket for the 'b', 'c' and 'e' connections but this does simplify matters. The operation of the unitester can be verified by taking a transistor whose connections are known. Select PNP or NPN as appropriate and plug the transistor correctly into the T_T socket. When P1 is moved fully around two of the LEDs will light or go out, depending on whether they were on or off. ◼

Figure 1. The transistor to be tested is fitted to the 'T_T' position. Its polarity (b, c and e), correct operation and gender (PNP or NPN) are determined with P1 and S1.



1



COS/MOS digital ICs

COS/MOS is a development of bipolar IC technology and an offspring of the MOS (Metal Oxide Semiconductor).
It started with the MOSFET being developed from the universally known junction FET (Field Effect Transistor). The former distinguish themselves from the latter by their isolated gate. The result of this gate isolation is a particularly high gate resistance. A drawback is that a static charge can build up on such a gate when the transistor is not connected in a circuit. This charge usually causes the immediate destruction of a MOSFET because the extremely thin isolating layer breaks down. So the handling of MOSFETs calls for special precautions. This also applies to COS/MOS ICs in which MOSFETs are integrated.

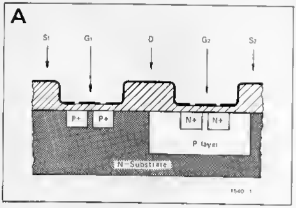
The integration is such that P+ and N- channel transistors are used alternately. Furthermore the switching circuits are integrated symmetrically. The latter two characteristics form the basis for the term COS (COmplementary Symmetry). Thus COS/MOS can be briefly described as complementary symmetrical MOSFET integration. A simple example of a COS/MOS IC construction is given in figure A. Here the dark-shaded area represents the n- (polarized) substrate. The diagonally-hatched area is the metal oxide film on which the electrical contacts are made. These contacts are drawn in deep black. Below the isolating layer at the electrical contact interruptions are the p- and n- layers. The layers are so integrated that the result is a complementary MOSFET pair as shown in figure B. Corresponding to the labelling of figure A, we have the following labelling in figure B 'S' for sources,, 'G' for gates and 'D' for common drain.

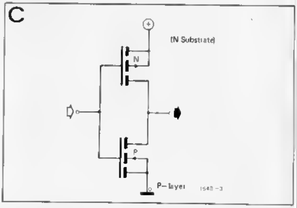
As can be seen from figure A the integration of an N- channel MOSFET is of a simpler construction than a P- channel. The latter requires an extra p- layer separating the substrate from the two n- layers which lie between the drain and G2 (= gate 2) and the junction between G2 and S2 (= source 2), respectively.

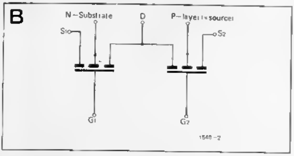
Of course, the integration of even the simplest COS/MOS IC is slightly more complex than figure B suggests. Even a common 2-input NAND gate consists of no less than four integrated MOSFETs.

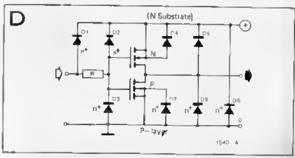
Like MOSFETs, every COS/MOS IC must be handled with due care because the inputs (gates) are isolated with respect to the rest of the integrated circuit. Normally the input impedance of a gate is 10^{12} Ω. As a result a static charge can easily build up if such an IC is kept in a plastic box, for instance. The human body too, is often statically charged. Touching the inputs with a finger can be sufficient to destroy the COS/MOS IC. Therefore the ICs are packed in a kind of expanded plastic containing a highly conductive substance. The connecting pins of the IC are pressed into the expanded plastic.
To give the inputs some measure of protection, manufacturers often provide COS/MOS IC inputs with an inbuilt protection circuit. These circuits are not shown in the circuit diagrams of the ICs.

Figure C is an example of an input circuit of a COS/MOS inverter. As can be seen in this figure, the circuit consists of a P- and an N- channel MOSFET. In reality the input circuit is as shown in figure D. Here we see that each gate input protection circuit comprises one resistor and three diodes. The diodes D_4 to D_8 are usually formed in the diffusion process. The gate input protection, however, is added as an extra (a resistor of about 500 Ω plus three diodes).
In figure D the diode D_3 has a breakdown voltage of about 25 V. The breakdown voltage of the diodes D_1 and D_2 is about 50 V. ◼









throwing some light on LEDs

Light-emitting diode (LED) lamps are replacing incandescent filament lamps in a variety of indicator applications, as they offer improved reliability and performance at a comparabla prica. A bewildering variety of LEDs of different shapes, sizes, colours and prices is now available, and the amateur constructor may find it difficult to choose a device for e particular project, especially if the parts list simply says that a 'LED' should be used, with no indication of type. This article aims to dispel soma of the mystary surrounding LEDs, so that the constructor can choose tha most suitable type for his requirements, and calculate the operating conditions.

Light emitting diodes were first made in 1954, when it was discovered that a point-contact diode made with gallium phosphide (GaP) as the base material emitted red light when forward biased. Although it was realised that this material offered the prospect of making a commercial solid-state light source, the physics of light emission from semiconductors was poorly understood, the technology to make the material was difficult, involving high temperatures and pressures, and it was some time before commercial devices appeared. Early LEDs were packaged in metal TO-18 type transistor housings, with a glass or plastic end window or lens, and costs were initially very high; furthermore, one could have any colour, provided it was red. Efficiency (i.e. light output for a given power input) was also very low.

When the phenomenon of semiconductor light emission was better understood, it was realised that the red emission of early GaP diodes was due to zinc and oxygen impurities in the GaP material. LEDs made with purer GaP produce a green light. Various exotic semiconductor materials for LEDs have now been developed, but the most common compound used is gallium arsenide phosphide (GaAsP). The advantage of this material is that the colour of light emitted can be varied by altering the proportions of arsenic and phosphorus in the material, from infra-red radiation, obtained with pure GaAs, to green radiation, obtained with pure GaP. At present there is no commercially available LED that emits blue light.

The most popular colour for LEDs is still red, using GaAsP material with the formula $GaAs_{0.6}P_{0.4}$ (i.e. the ratio As:P is 6:4). LEDs using this material are easiest (and hence cheapest) to produce, and have the highest efficiency. Green LEDs are the least efficient, but this disadvantage is offset to some extent by the fact that the human eye is more sensitive to green light than to red light.

LEDs are now commonly available in four colours; red, orange, yellow and green. An important factor to be considered when choosing the colour of a LED is the proposed application. For example, red is conventionally used for warning lights, but green and yellow may be aesthetically more pleasing for other purposes.

Cost is always an important consideration. Green and yellow LEDs may be up to twice as expensive as red LEDs, as well as being less efficient. This inefficiency is not necessarily a disadvantage, provided low-current (e.g. battery) operation is not required. For comparable light output from a green LED it may be necessary to run it at twice the current of a red LED, but if a mains power supply is available this is no great problem, provided the ratings of the LED are not exceeded.

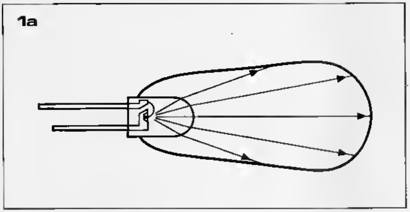
In general, it is true to say that, in terms of efficiency, 'yer gets what yer pays for' with LEDs. The high-efficiency, 'state-of-the-art' devices now appearing on the market are considerably more costly than the less efficient second generation devices that are commonly available to the amateur contructor, since the technology required to make high-efficiency LEDs is considerably more difficult, and development costs still have to be recouped.

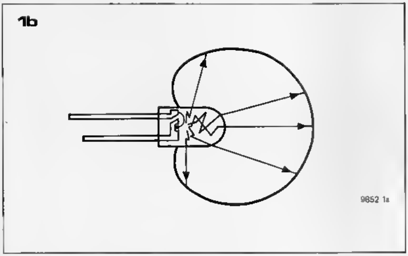
1

Packaging

The high cost of the early LEDs was partly due to the expensive metal-can package, which is still used for some military and industrial devices. Modern consumer LEDs utilise a much cheaper form of encapsulation, the semiconductor wafer and its leadouts simply being encapsulated in a moulded epoxy resin housing. A typical selection of modern, epoxy-encapsulated LEDs is shown in photo 1.

Although the diode junction is essentially a point source of radiation, the encapsulation can have a profound effect upon the radiation pattern of the LED. For example, it the epoxy encapsulation is transparent then the LED functions as a point source, with the emitted light being confined to a relatively small angle, as shown in figure 1a. If the epoxy material is translucent, then the light produced by the LED is diffused over a much wider





angle, as shown in figure 1b. For a given light output from the LED chip, the point source LED will appear brighter, when viewed on axis, than the diffuse LED. However, off axis the brightness of the point source LED falls off rapidly, while the diffuse LED provides even illumination over a much wider viewing angle.

The shape of the encapsulation also has a marked effect on the radiation pattern, since it acts as a lens. For example, a LED in a cylindrical encapsulation with a domed end produces a radiation pattern as shown in figure 2a, whereas one with a parabolic cross-section produces the radiation pattern in figure 2b. It is apparent that the radiation pattern of figure 2b would produce much more even illumination of a plane surface placed at right-angles to the axis of the LED.

As well as being transparent or translucent, the LED encapsulation may be either clear or coloured. Of course, a coloured encapsulation does not influence the colour of light emitted by the LED, this is determined by the semiconductor material. If a coloured encapsulant is used it must be the same colour as the light emitted by the LED, otherwise the light output will be seriously attenuated.

Special packages

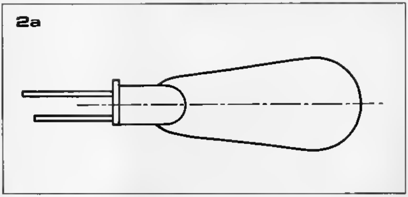
Most commonly available LEDs have a circular cross-section, for the simple reason that, for panel mounting purposes, round holes are easiest to

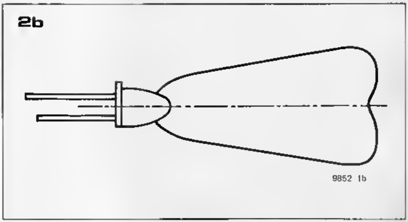
Figure 1a. A point source LED produces a fairly narrow beam of light.

Figure 1b. A diffuse LED produces a much more even radiation pattern, and has a wider viewing angle.

Figures 2a and 2b. The LED encapsulation acts as a lens, the shape of which has a marked effect on the radiation pattern.

Photo 1. A typical selection of commonly available LEDs.





drill. However, with the demand for types of LED display other than single panel lamps (e.g. bar graph type displays), different types of package have appeared. Photo 2 shows a LED which has a flat rectangular cross-section with a rounded top. The dimensions of this type of LED (2.5 x 5 mm cross-section) allow it to be stacked on a standard 2.54 mm (0.1") pitch, to form arrays for such applications as audio level meters.
Another interesting shape is shown in photo 3. This type of LED has a transparent plastic case fitted with a flat diffuser screen, which makes it particularly suitable for backlighting of legends. In fact, press-on lettering or transfers can be applied direct to the diffuser screen.
Integrated LED arrays, housed in dual-in-line packages, are also becoming quite popular. Such an array of 10 LEDs is shown in photo 4.

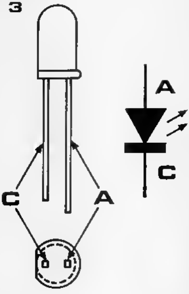
Electrical characteristics of LEDs

Electrically, LEDs behave like normal semiconductor diodes, which is not surprising, since they consists of a single PN junction. However, the forward voltage drop of LEDs is considerably greater than that of, say, a silicon diode. Furthermore, this forward voltage drop is not the same for all LEDs: it depends on the type and colour. Earlier types of LED had forward voltages varying from around 1.6 V for red, to around 2.4 V for green. However, modern high-efficiency LEDs tend to have forward voltages around the 2 V mark, irrespective of colour.
As with normal diodes, the forward resistance of LEDs is very low, which means that once the forward voltage is exceeded the current through it will increase very rapidly for only a very small increase in voltage. This makes it essential to use an external, series, current-limiting resistor if the LED is to be connected to a voltage source. For DC operation, the required series resistor is found from the equation:

$$R = \frac{U_s - U_f}{I}\text{, where}$$

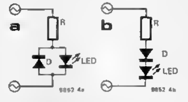
U_s = supply voltage
U_f = LED forward voltage
I = required current

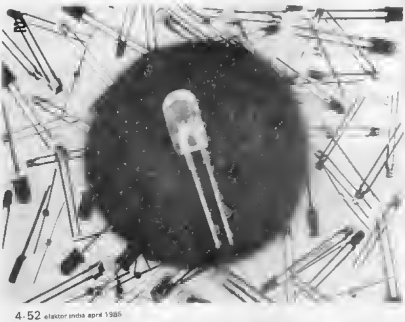
3



9852 2

4

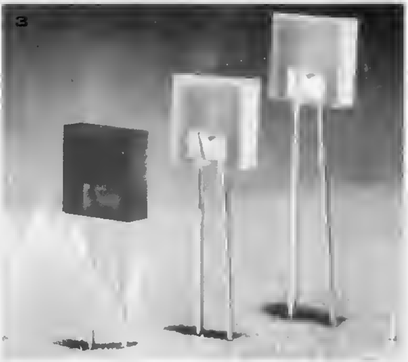




If data on a LED is unobtainable (e.g. unmarked, untested types) then as a rule of thumb, most LEDs will withstand a forward current of up to 40 mA (many will withstand more and only a few types will withstand less). Using 2 V as a value for the forward voltage drop will also not be far out.
However, if a LED is to be used with a low supply voltage then extra care must be taken not to operate the LED too near its maximum current, since a small variation in the supply voltage could lead to a large increase in current.
Care should also always be taken to connect LEDs the correct way round, since they have a very low reverse-breakdown voltage (typically 4 V) and are easily destroyed by excessive reverse voltages. For this reason great care should always be taken when trying to identify the leadouts of an unknown LED. A 3 V supply with a 150 Ω series resistor should be fairly safe. However, most manufacturers identify the leadouts of LEDs in one of two ways. The cathode, which is connected to the more negative supply voltage, has a shorter leadout than the anode (which is connected to the more positive supply voltage), or else the LED package has a flat side next to the cathode leadout (this only applies to circular cross-section LEDs). These identification marks are shown in figure 3.

AC operation

LEDs can be used to replace low-voltage incandescent lamps where only an AC supply voltage is available. The LED conducts only on one half cycle of the AC waveforms and is reverse biased on the other half cycle. The LED must therefore be protected from excessive reverse voltages. This can be done by connecting a diode in reverse parallel with the LED, as shown in figure 4a. The diode conducts on the negative half-cycle of the waveform and this limits the reverse bias on the LED to the diode forward voltage drop.
Another method is to connect a diode with a high breakdown voltage (greater than peak supply) in series with the LED, as shown in figure 4b.
The first method has the advantage that the diode need not have a high reverse breakdown voltage, since it is protected by the LED. However, it has the disadvantage that current flows through the series resistor during the whole cycle, so the resistor dissipates twice as much power as in the second circuit, where the resistor conducts only on positive half-cycles of the waveform.
In either case, when calculating the resistor value it is important to remember that the LED is conducting for only half the time, so the average LED current will be only half that expected from the calculated resistor value. To allow for this the approximate required resistor value is obtained from the equation:


3

$$R = \frac{U_{RMS} - U_f}{2I}, \text{ where}$$

U_{RMS} = AC supply voltage
U_f = forward voltage of diode(s)
I = required average current
The protecting diode must have a current rating greater than I.

Lifetime of LEDs

Early LEDs had problems with copper contaminants poisoning the diode junction, which caused a reduction in brightness after only a few hundred operating hours. Modern LEDs, however, if properly treated, should have an operating life of at least 100,000 hours, and possibly up to 1,000,000 hours (defined as the time taken for the light output to fall to 50%).

For the constructor, ensuring that a LED has a long life starts with careful handling of the device. The leads of a LED should never be bent closer than about 2 mm from the encapsulation; pliers should always be used to relieve the strain, otherwise the package could be damaged, resulting at best in the ingress of moisture, and at worst in complete disintegration of the package. When soldering LEDs the junction temperature should never be allowed to exceed 125°C, so a heat shunt should be used on the leads.

LEDs should not be operated at excessive temperatures. A LED operating at a temperature of 75°C produces only half the light output that it does at 25°C and also has a shorter life. The rule as far as the constructor is concerned is thus to keep LEDs away from hot spots in equipment, and not to operate them too near their maximum current rating.

Conclusion

To sum up, the choice of a LED for a particular application should be based on several criteria. For general indicator lamp applications in mains powered equipment, most LEDs are adequate, and the choice can be made on the basis of cost and the required colour. If a narrow viewing angle is acceptable, then a point-source LED will give greater apparent brightness (within its viewing angle) than a comparable diffuse LED. If high light output and/or low power consumption are prime considerations, then it is worth considering a high-efficiency LED from a reputable manufacturer, though this will inevitably be more expensive

For special applications, such as bar-graph type displays, interesting possibilities are offered by the integrated LED arrays and the new shapes of LED packages now available.

Readers wishing to pursue the subject further are recommended to read the

'Optoelectronics Applications Handbook' from Hewlett-Packard.

4


Figure 3. The leadouts of a LED may be identified by a shorter lead for the cathode, or a flat on the package next to the cathode.

Figures 4a and 4b. Two methods of connecting a LED for AC operation.

Photo 2. LEDs are also available in a rectangular package with a half-round end, specifically for use in arrays.

Photo 3. These LEDs are fitted with a flat diffuser screen, and are ideal for back illumination of legends.

Photo 4. An array of 10 LEDs housed in a dual-in-line package.

measure from 50 to 110 dBA

sound pressure meter

Anybody can tell whether they are in relatively quiet or noisy surroundings. At least you'd think so. Although . . . sometimes you wonder. Human hearing is subjective: what some people consider 'pleasant background music', others would class as 'an abominable row'. For a more objective assessment of the actual sound level, some kind of meter is required.
However, since we are mainly interested in sound as it relates to us, the measurement must also take the average frequency response of our ears into account. The meter described here measures in dBA, over the whole range from normal conversation up to loud disco music.

Sound . . rapid vibrations, travelling through the air, is always present — even if we don't always realise it. However, those who have ever spent some time in a completely sound-proof room will know the difference between 'no sound' and normal background levels.
Sounds can be quite pleasant — music, for instance — or decidedly unpleasant, like a car horn going off unexpectedly just behind you. The difference is not only the type of sound, but also the level. Above a certain level, sounds tend to get annoying. At even higher levels, it actually hurts your ears — and permanent damage may well occur.

This is cause for some concern, nowadays The extremely high levels that are pumped into disco's may give a nice 'high' sensation at the time However, if your ears are ringing when you step outside after a few minutes, be warned! Prolonged exposure to this kind of abuse can (and often does) cause *permanent* damage to your hearing And after all, we all hope that our ears will last a lifetime
Before describing the sound pressure meter itself, let's take a closer look at our own built-in meter: our ears. What can they measure?
We can only hear sound within a certain frequency range — broadly speaking, between 20 Hz and 20 kHz. There is some controversy about the actual limits, but that's not so important in this context Whether the upper limit is 20 kHz, 10 kHz or only 7 kHz is partly a question of age, and below 20 Hz sound may possibly be 'felt' — but it is not really 'heard'. However, who said electronics was an accurate science? When designing a sound pressure meter, 'somewhere between 20 Hz and 20 kHz' is a sufficiently accurate definition for the limits.
For sound to be audible, it must not only be within the correct frequency range. Loudness is also important, and the minimum level that we can hear varies with frequency. Our ears are most sensitive in the 500 Hz to 5 kHz range, as shown in figure 1. For a 100 Hz and a 1 kHz tone to 'appear' equally loud to us, the former must actually be at a much higher level than the latter — certainly at low levels.
This is all clearly shown in the plots given in figure 1. The lower dotted line is the hearing threshold: sounds below this level are inaudible. From the scale at the left it can be seen that this corresponds to 0 dB at 1 kHz (no coincidence, that), and to 40 dB at 50 Hz Quite a difference! The higher lines all correspond to equal (apparent) loudness, as a function of frequency. The highest line is marked 'threshold of pain' This is rather misleading, unfortunately: it suggests that everything is perfectly all right up to this level. Not so! Prolonged exposure to much lower levels (30 minutes at 100 dB, for instance) can already lead to permanent damage The only point about the actual threshold is that it really hurts, and damage is likely within a very short time indeed
A lot more could be said about these plots, but there are several good books on the subject. Theory is one thing, but there is nothing like practical examples. In figure 2, several well-known sounds are plotted on a sound level scale. This is calibrated in dBA, as in common practice. But what *is* a 'dBA', exactly?
If we want to measure sound levels as they relate to human hearing, we must obviously 'weigh up' the results to match the characteristics shown in figure 1 An 'objective' sound level of 60 dB at 100 Hz, say, must give the same 'loudness' result as 50 dB at 1 kHz. Obviously, it would take some doing to build a circuit that accurately follows all plots at all levels Fortunately, there is no need for that kind of accuracy, and according to international standard a single fixed frequency compensation can be used. This is the so-called A-weighting curve, shown in figure 3 Sounds picked up by a microphone are passed through a filter with this response, and the level is measured behind the filter The result is expressed in dBA

Measuring sound in dBA

By now we've got a reasonable idea of what we need to measure sound pressure in a useful way Obviously, since we want to measure sound, we will need a microphone with a reasonably flat response. Some kind of capacitor microphone would be ideal.

1

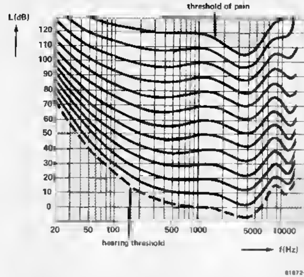


Figure 1. This graph illustrates the degree of sensitivity of human hearing. The lines of equal loudness, isophones, indicate at what volume a given frequency must be for it to sound as loud as a 1000 Hz tone.

2

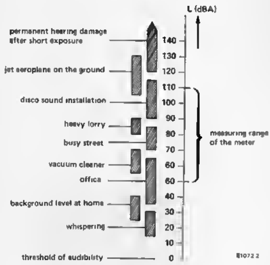


Figure 2. Examples of loudness values expressed in dBA.

3

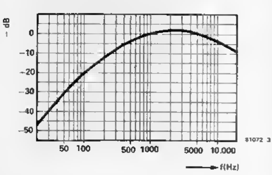


Figure 3. The characteristics of the A weighted curve.

Then, a microphone preamplifier of course — you can hardly expect to drive a pointer instrument from the microphone output! This preamp must be followed by the A weighting filter mentioned above, the output from the filter is fed to an AC measuring circuit, that indicates the level in dB.

The circuit described here will measure in the 50 dBA to 110 dBA range. It is apparent, from a brief look at figure 2, that this is quite adequate for normal use. Below this level, you're in the background noise. And above it? You shouldn't be there in the first place!

Within the range, you can compare the output level from two loudspeaker systems: measure your neighbour's car as he goes past, to find out whether he really needs a new exhaust pipe; or compare the noise produced by a jet aircraft overhead to that of your own little model aeroplane.

The circuit

The complete circuit is shown in figure 4. A good choice for measuring microphone is the Philips electret type, LBC 1055/00. Basically, this is a capacitor microphone without the need for a special high-voltage supply. It has an FET buffer stage built in, so that its output is at quite a low impedance. Its frequency response is virtually flat from 100 Hz to 14 kHz, and it doesn't run into overload until the level exceeds 134 dB . . .

The FET in the microphone needs a positive supply, and this is derived via R8 and C3. The actual microphone signal is amplified by T1 and T2. The gain of this stage is approximately x20 — determined by the ratio between R7 and R3. Both the input impedance (determined by R1) and the gain are chosen to suit this type of microphone. If some other type is to be used, some modifications may be required here.

The amplifier signal is passed through an emitter follower (T3) to the A-weighting filter, consisting of R10 . . R12 and C5 . . . C7. This filter gives a reasonable approximation of the desired frequency response shown in figure 3.

The final stage is the actual meter circuit. IC1, together with the diode bridge, a 1 mA moving-coil pointer instrument and assorted feedback resistors, makes a very good AC voltmeter. Diode D1 is included to protect the meter itself from overload. The desired measuring range is selected by means of S1. Effectively, the *voltage* across the divider chain (R14 . . R18) is proportional to the *current* through the meter, and when the feedback is taken off from a lower point in the chain this will correspond to a lower input voltage required for full scale deflection.

The actual meter used is a relatively 'sluggish' (heavily damped) 1 mA type — as used for tuning indication, for

4

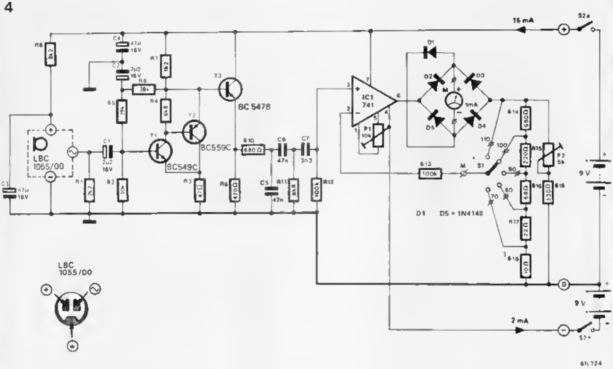


Figure 4. The sound presure meter circuit consist of a microphone, an amplifier, a filter, and an AC voltmeter with range switch.

5

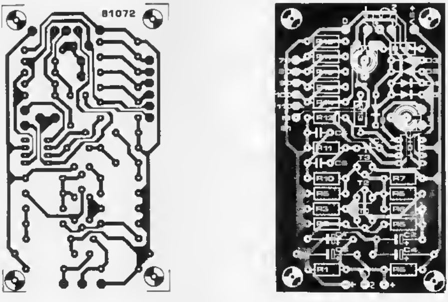


Figure 5 The printed circuit board with components overlay for the sound meter.

6

Figure 6. A scale in dB's for the moving coil meter. The lower scale (from 0 to 1) is the original mA scale.

7

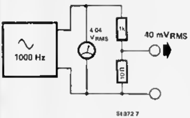


Figure 7. This auxiliary circuit is used to calibrate the meter.

Parts list

Resistors

R1 = 2k2
R2 = 10 k
R3 = 47 Ω
R4,R11 = 6k8
R5,R6 = 39 k
R7 = 1k2
R8 = 8k2
R9 = 470 Ω
R10,R14 = 680 Ω
R12,R13 = 100 k
R15 = 220 Ω
R16 = 68 Ω
R17 = 22 Ω
R18 = 10 Ω
R19 = 330 Ω
P1 = 10 k preset
P2 = 5 k preset

Capacitors:

C1,C2 = 2μ2/16 V
C3,C4 = 47 μ/16 V
C5,C6 = 47 n
C7 = 3n3

Semiconductors

T1 = BC 549C
T2 = BC 559C
T3 = BC 547B
D1 ... D5 = 1N4148
IC1 = 741

Miscellaneous:

Electret microphone LBC 1055/00 (Philips)
M = 1 mA moving coil meter.
S1 = single-pole five-way switch
S2 = double-pole on/off switch

instance. A more sensitive instrument can also be used, provided a suitable shunt resistor is included in parallel, to bring the total sensitivity to 1 mA f.s.d. A suitable scale is shown in figure 6.

There should be no problems with the construction; a printed circuit board layout is given in figure 5. The connections to the microphone are included in figure 4.

Calibration

There are two calibration points in the circuit: P1 is used to compensate the offset of IC1 and P2 calibrates the actual meter.

The first step is the offset compensation. Put in simple terms with no input signal present, the meter should read zero! The adjustment procedure is as follows. Disconnect the microphone (otherwise it may be damaged!), short R1 and switch S1 to the most sensitive range (70 dB f.s.d.). Set P2 to the centre position, and adjust P1 until the meter just rests at 0.

Now to calibrate the meter. This is more awkward. The best way is to calibrate it against a reference sound source, or by comparing the reading with that of a properly calibrated sound pressure meter. However, we assume that relatively few of our readers will have access to this kind of equipment.

There is another way — less accurate, but good enough for most applications. Manufacturers specify the output from their microphones at some reference level. For the LBC 1055/00, it can be calculated from the manufacturer's data that the output at 110 dB should be 40 mV (RMS). This is rather a low value to set accurately at the output of a tone generator but using two resistors, as shown in figure 7, will solve that problem. The microphone remains disconnected for the time being, instead, the output from the test circuit given in figure 7 is connected across R1.

With the output from the tone generator set to 4.04 V at 1 kHz, we now have the desired 40 mV reference input to the meter circuit. Switch S1 is turned up to the 110 dB range, and P2 is adjusted until the meter reads 0 dB.

One final word, regarding the power supply. We deliberately opted for batteries, so that the unit is portable. A mains supply would be possible, but it's rather clumsy. With the low current consumption involved, batteries will last quite long enough! ◼

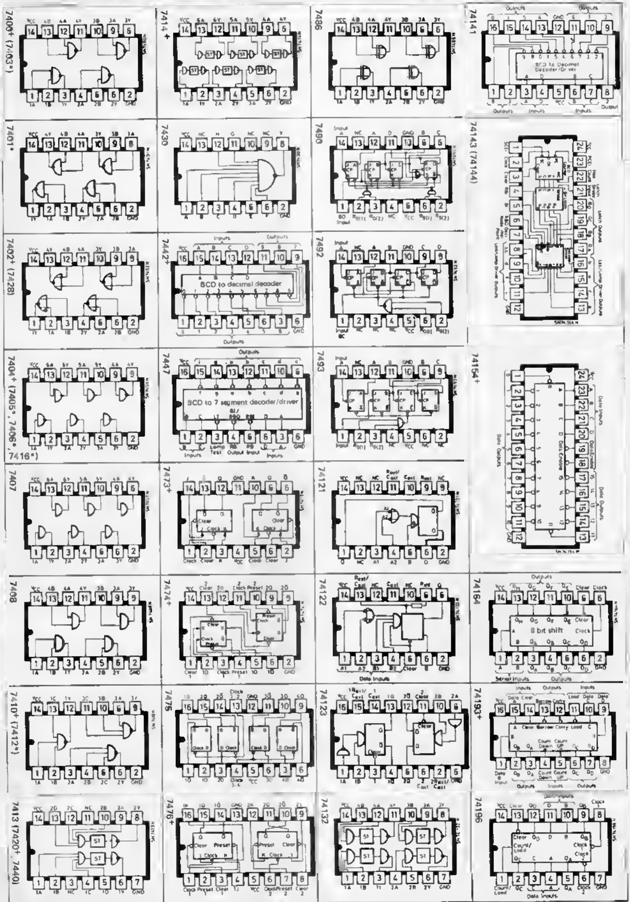
soldering aluminium

H.M. Wolber

Constructors use aluminium a great deal because it is relatively inexpensive and very easy to work with. However, soldering it causes problems, for using an ordinary soldering iron and solder just does not work. The solder does not seem to feel particularly attracted to aluminium. Nonetheless, Mr. Wolber has come up with a solution which allows 'normal' (60/40) solder to be soldered on aluminium.

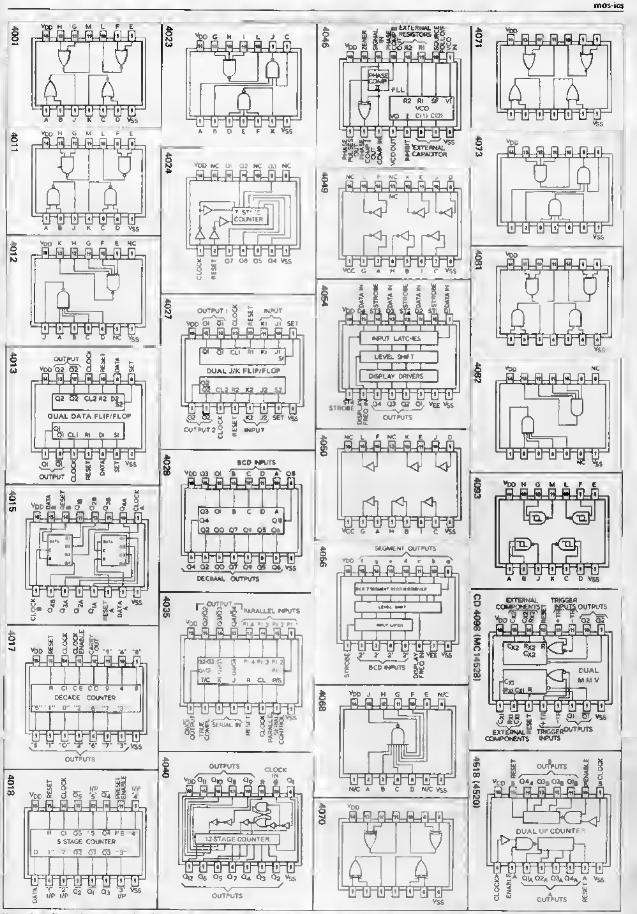
The problem with aluminium is that it oxidises quickly. In other words, aluminium is always covered in a layer of oxide. As you know, this is a very good insulator and prevents solder from even touching the aluminium. However hard you try to scrape the oxide layer off, oxidation occurs so rapidly that a new layer will appear as soon as the 'old one' is scraped off.

To prevent oxidation, apply grease or oil to the solder point. Then scrape off the layer of oxide with a sharp object underneath the oil. The oil prevents oxygen from reaching the clean aluminium. Now drip hot flux onto the area with a soldering iron, causing the oil to evaporate and the flux to cover the area. The area can now be soldered. To ensure a solid connection make sure plenty of heat is produced by using a soldering iron of at least 100 W. If necessary, the area can be held over a gas flame to remove all traces of oil before soldering. ◼



* open collector

+ Pin-compatible CMOS equivalents available from Teledyne Semiconductor and National Semiconductor



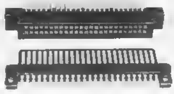
Note: A prefix to the type number denotes the manufacturer,
e.g. **CD** 4001 (RCA), **MC** 14001 (Motorola), **N** 4001 (Signetics), **SCL** 4001 (Solid State Scientific), **SIL** 4001 (Siltek).

market

TWO PART EDGE CONNECTOR

O/E/N have introduced edge connector as per HE 901 specification. It can be used as direct/indirect edge connector with flexibility in matching circuit requirements. Contacts available are from 4 to 98, with 3 A rating. Terminal styles available are wire wrap, solder pins and solder eyelets.

When the edge connector is used as a two part system, it eliminates the need of gold plated fingers on PCB.

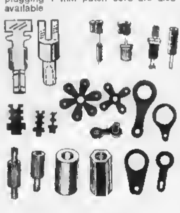


For further information, write to:
O/E/N Connectors Ltd
Vyttila, Cochin 682 019

BRASS TERMINALS

Jain Electronics offer a wide range of brass terminals for use in electronic products. Use of these terminals results in neat and clean wiring and simplifies servicing and testing of the product.

The terminals are suitable for mounting of components or can serve as monitoring points during assembly and testing. Terminals suitable for plugging 1 mm patch cord are also available.



For further information, write to
Jain Electronics
F 3, Nand Dham Industrial Estate
Marol, Andheri (East)
Bombay 400 059

FUNCTION GENERATOR

Rashmi Electronics have developed a new function generator F-20.

The function generator is available as assembled unit or in kit form. It has following features: Frequency range 1 Hz to 100 KHz continuously variable in 5 decades. Output amplitude 0 to 10 V pp continuously variable in 3 decades. Output Impedance 50 Ohms. Output waveform: Sine, Square and Triangle.

For further information, write to
Rashmi Electronics
2-15-34, Kadrabad,
Polas Lane Corner, Jalna 431 203

DIGITAL PANEL METER

Jivan Electro Instruments manufacture a general purpose DPM for OEM applications. It has three and half digit LED display and requires a panel cut out of 30 x 70 mm. The circuit uses CMOS IC and superior grade components for high stability and good noise immunity. Readings are claimed to be free from temperature drift. Supply voltage required is 5 V DC, and the measurement range is 200 mV/2 V DC.

For further information, write to:
Jivan Electro Instruments
394 GIDC Estate
Makarpura, Baroda 390 010

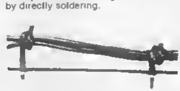
R.F. DEVICES

Acrian Inc. offer a wide range of R.F. Transistors which can be used in products with power ratings as high as 1000 watts and frequencies upto 4.2 GHz. These devices manufactured using the latest ISOFET technology are suitable for use in military, communications, avionics and other similar applications.

For further information, write to:
Tecmark Electronics Pvt. Ltd
44/2, Sassoon Road,
Pune 411 001

FIXING CLIP

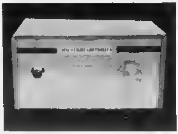
Suresh Electrics and Electronics have introduced a new type of fixing clip for use where fixing the clip using screws is not possible. This is a completely insulated binding fixative clip. This is mainly suitable for fixing onto the PCB by directly soldering.



For further information, write to:
Suresh Electrics and Electricals
P.B. No 9141, 3 B Camac Street,
Calcutta 700 016

DISCO LIGHT CONTROLLER

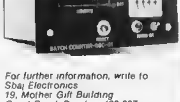
Elektro World's disco light controller is capable of handling the loads upto 500 watts. The electric bulbs controlled by this unit will flash to the tune of music. The level of sensitivity is adjustable through the sensitivity control provided on the front panel of the unit. The controller operates directly from 230 V mains supply and weighs only 500 grams.



For further information, write to:
Elektro World
48/B Harpal Nagar
Moradabad-244 001

DIGITAL BATCH COUNTER

Sbaj Electronics have introduced their new digital batch counter Model SA-01 A. A versatile digital counter specially designed for use in a variety of industrial applications with flexibility. The instrument is compact and rugged and is claimed to be highly accurate. The counting can be done through optical or mechanical transducers, or any other means of obtaining electrical pulses.



For further information, write to
Sbaj Electronics
19, Mother Gift Building
Grant Road, Bombay 400 007

QUICK PCB LAB

Elektromin have marketed a new miniature laboratory set up for making prototype PCBs in the R & D laboratory. It is claimed that an engineer can make his own PCB from layout stage to finished Tin plated PCB in a matter of few hours.

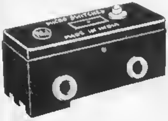
The set up is available in two models-Junior and Senior, and contains all the necessary equipment, fixtures, processing aids and consumables.

Installation and commissioning is carried out by the manufacturer's representative at the users premises.

For further information, write to:
Elektromin
3, Electronic Estate
Satara Road, Pune 411 009

MICROSWITCH

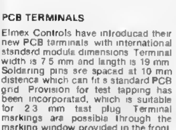
Raj microswitch type RMS-1 is a single pole snap acting micro-switch with one NO and NC contact. The moving contact are of pure silver and the operation mechanism is of beryllium copper trident spring. The repeat accuracy is claimed to be exceptional and the mechanical life, very long



For further information, write to:
Raj Industries,
118, Guru Nanak Udyog Bhavan,
L B Shastri Marg Bhandup,
Bombay 400 078

PCB TERMINALS

Elmex Controls have introduced their new PCB terminals with international standard module dimensions Terminal width is 7 5 mm and length is 19 mm Soldering pins are spaced at 10 mm distance which can fit a standard PCB grid Provision for test tapping has been incorporated, which is suitable for 2 3 mm test plug Terminal markings are possible through the marking window provided in the front. The terminal is rated for 500 V, 15 A



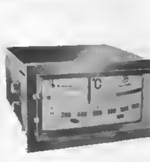
For further information, write to

Elmex Controls Pvt Ltd
12 GIDC Estate, Makarpura Road
Baroda 390 010

TEMPERATURE CONTROLLER

Chowdhary Instrumentation have designed high accuracy, direct deflection type temperature indicators/controllers A set point arm is provided which can be set at any position externally. When the indicating pointer reaches the set point, a switching operation is initiated by an inductive pick up. Two lamps are provided on the dial to indicate mains ON/OFF and control ON/OFF

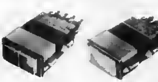
Accuracy is claimed to be ±1% Relay rating is 4 Amps at 230 V AC. The controller is available in various temperature ranges, depending on the thermocouple used



For further information, write to:
Chowdhary Instrumentation
110, Model Basti,
New Delhi 110 005

LIGHTED PUSH BUTTONS

Efficient Engineers have developed a new series of lighted push buttons and indicators with rectangular bezel of 48 x 24 mm. Two independent lamp circuits and one or two pole self cleaning and snap acting microswitches in momentory or maintained action are offered. Any legend can be engraved on the front facia. These push buttons are suitable for process control instrumentation, supervisory remote control systems, data acquisition, control systems, sequencing logic controls, hierarchy controls and turn key instrumentation



For further information, write to:
Sai Electronics
Thakore Estate,
Kurla Kirol Road,
Vidyavihar (West)
Bombay 400 086

8 AMPERE RELAYS

Jost's Engineering have announced the availability of the new 8 Amp relay family 'CS' from ITT France. The relays are available to actual users in the following versions: Time delay, Bistable, Pulse, Flasher, Passage and Instantaneous. With 4 C/O or 2 C/O contacts.

The 'CS' family confirms to the international standards of CEI, VDE and NFC

For further information, write to:
Jost's Engineering Co. Ltd.
60, Sir Phirozeshah Mehta Road,
Bombay 400 001

PORTABLE THERMOMETER

Arun Electronics Pvt Ltd have introduced a pocketsize portable thermometer with LCD display The instrument works on 9 V battery and consumes very low power Ambient temperature compensation is automatically provided. The instrument can either be used with Iron-Constantan (0 to 600° C) or with Chromel-Alumel (0 to 1000°C) thermocouples.

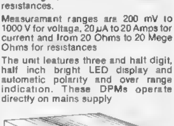
For further information, write to:
Arun Electronics Pvt Ltd
B 125, Ansa Industrial Estate
Saki Vihar Road,
Bombay 400 072

DIGITAL PANEL METERS

Omega offer a large range of digital panel meters which indicate AC and DC voltages, AC and DC currents and resistances.

Measurement ranges are 200 mV to 1000 V for voltage, 20 µA to 20 Amps for current and from 20 Ohms to 20 Mega Ohms for resistances

The unit features three and half digit, half inch bright LED display and automatic polarity and over range indication. These DPMs operate directly on mains supply



For further information, write to:
Omega Electronics
36, Hathi Babu Ka Bagh
Jaipur 302 006

classified ads

advertisers index

missing link

musical doorbell

(Aug/Sept 1984 Page 8.80)

This doorbell is very sensitive and come on when an electrical distrubance is caused eg when a light is switched on or off

This can be rectified by reducing the values of R4, R5, R6, to 1K

digital band-pass filter

(Aug/Sept 1984, page 8-42)

Resistor R19 should be connected in series with additional capacitor of 1 µF between pin 13 of IC1 (— input of A4) and earth this will raise the Q factor to its required value the times stated at the output waveforms of the two monostables are the periods of the input pulses and not of the output signals

flash meter

(October 1984, page 10-30)

There are some stubborn cases defying our previous solution to problems caused by leakage currents If you are experiencing these, remove pin 3 of IC6 from its socket, unsolder all connections to this point on the pcb and re make them 'in the air' direct to pin 3 Use short lengths of insulated wire The other side of DIL switches S5 S8 and the corresponding terminals of C8 C11 should also be unsoldered from the pcb and remade 'in the air' with short lengths of insulated wire

R N No 39881/83

MH/BYW - 228
LIC No 91

Printer & Publisher - C. R. Chandarana, 2 Kaumen, 14th A Road, Khar, Bombay-400 052 & Printed at Trupti Offset, 103, Vasan Udyog Bhavan, Off Tulsi Pipe Road, Lower Parel, Bombay 400 013