ISSN 0265-2919

R 80p

# THE HOME COMPUTER COURSE 3

MASTERING YOUR HOME COMPUTER IN 24 WEEKS

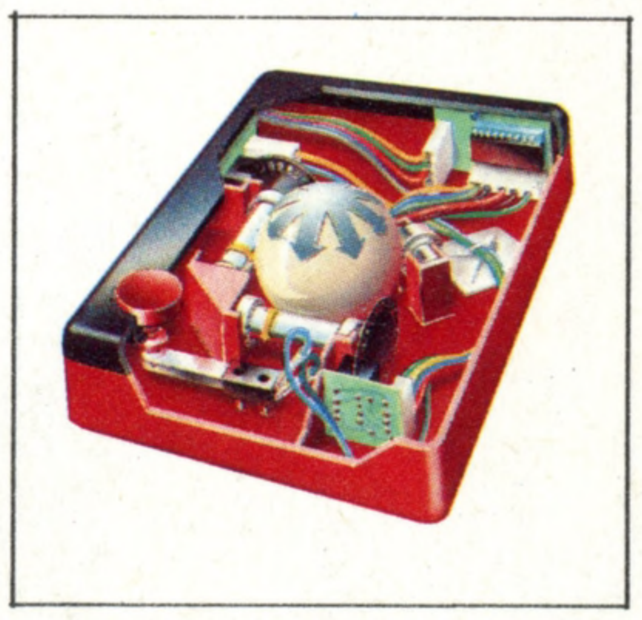

8 PAGE GLOSSARY SUPPLEMENT

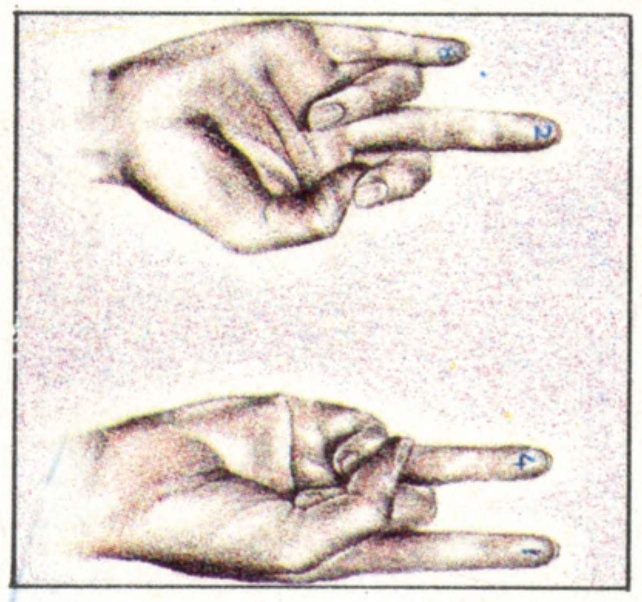
An ORBIS Publication

IR £1 Aus $1.95 NZ $2.25 SA R1.95 Sing $4.50 USA & Can $1.95

# CONTENTS



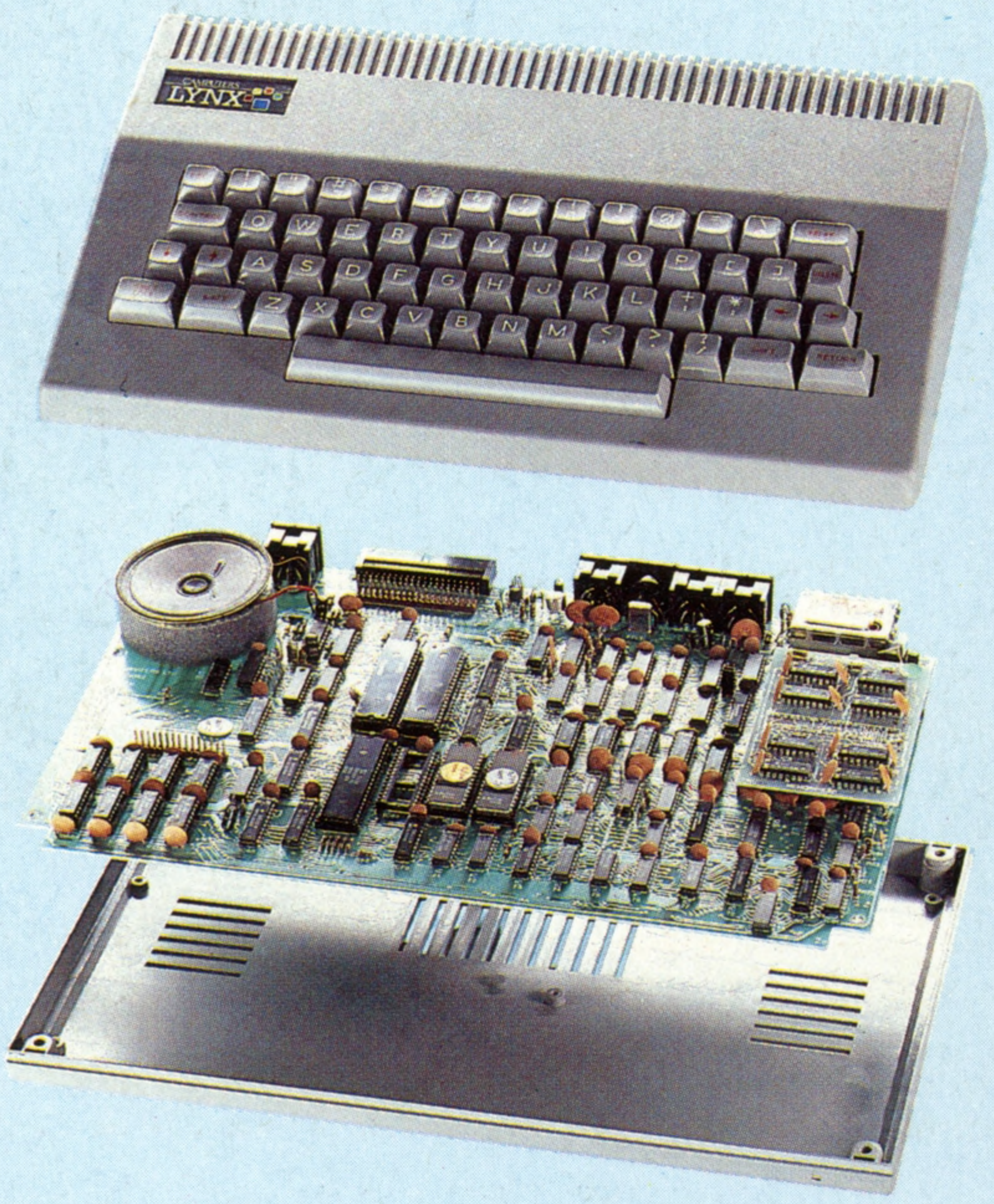






## Next Week

- We review the versatile and inexpensive Lynx, an all-British computer with disk drive capability
- We explain how word processing software turns a computer into a business aid
- Next week we continue our BASIC course with a look at 'programs within programs'







**Editor** Gareth Jefferson; **Art Editors** David Whelan, Jimmy Egerton, **Production Editor** Catherine Cardwell; **Illustrations Editor** Claudia Zeff; **Designer** Hazel Bennington; **Art Assistants** Steve Broadhurst, Julie-Ann Chambers, Dawn Coleman; **Sub Editor** Teresa Bushell, **Researcher** Melanie Davis; **Consultant Editor** David Tebbutt; **Contributing Consultants** Garry Marshall, Dick Pountain; **Contributors** Tim Heath, Peter Jackson, Ray Hammond, Chris Cunningham, Ron Smith, Ian Scales; **Group Art Director** Perry Neville; **Managing Editor** Bruce Sawford; **Managing Director** Stephen England; **Editorial Director** Brian Innes; **Project Development;** Peter Brooksmith; **Executive Editor** Chris Cooper; **Production Co-ordinator** Ian Paton; **Circulation Director** David Breed; **Marketing Director** Michael Joyce. **Designed and produced by Bunch Partworks Ltd. Published by Orbis Publishing Ltd.**



**Typesetting Bunch Typesetters. Reproduction by Mullis Morgan Ltd. Printed in Great Britain by Artisan Press Ltd., Leicester.**

**HOME COMPUTER COURSE** – Price UK 80p IR £1.00 AUS $1.95 NZ $2.25 SA R1.95 SINGAPORE $4.50 USA and CANADA $1.95

**How to obtain your copies of HOME COMPUTER COURSE** – Copies are obtainable by placing a regular order at your newsagent.

**Back Numbers UK and Eire** – Back numbers are obtainable from your newsagent or from HOME COMPUTER COURSE. Back numbers, Orbis Publishing Limited, 20/22 Bedfordbury, LONDON WC2N 4BT at cover price. AUSTRALIA: Back numbers are obtainable from HOME COMPUTER COURSE. Back numbers, Gordon & Gotch (Aus) Ltd, 114 William Street, PO Box 767G Melbourne, Vic 3001. SOUTH AFRICA, NEW ZEALAND, EUROPE & MALTA: Back numbers are available at cover price from your newsagent. In case of difficulty write to the address in your country given for binders. South African readers should add sales tax.

**How to obtain binders for HOME COMPUTER COURSE** – UK and Eire: Details of how to obtain your binders (and of our special offer) are in issues 4, 5, 6, and 8. EUROPE: Write with remittance of £5.00 per binder (incl. p&p) payable to Orbis Publishing Limited, 20/22 Bedfordbury, LONDON WC2N 4BT. MALTA: Binders are obtainable through your local newsagent price £3.95. In case of difficulty write to HOME COMPUTER COURSE BINDERS, Miller (Malta) Ltd, M.A. Vassalli Street, Valletta, Malta. AUSTRALIA: For details of how to obtain your binders see inserts in early issues or write to HOME COMPUTER COURSE BINDERS, First Post Pty Ltd, 23 Chandos Street, St. Leonards, NSW 2065. The binders supplied are those illustrated in the magazine. NEW ZEALAND: Binders are available through your local newsagent or from HOME COMPUTER COURSE BINDERS, Gordon & Gotch (NZ) Ltd, PO Box 1595, Wellington. SOUTH AFRICA: Binders are available through any branch of Central Newsagency. In case of difficulty write to HOME COMPUTER COURSE BINDERS, Intermag, PO Box 57394, Springfield 2137.

**Note** – Binders and back numbers are obtainable subject to availability of stocks. Whilst every attempt is made to keep the price of the issues and binders constant, the publishers reserve the right to increase the stated prices at any time when circumstances dictate. Binders depicted in this publication are those produced for the UK market only and may not necessarily be identical to binders produced for sale outside the UK. Binders and issues may be subject to import duty and/or local taxes, which are not included in the above prices unless stated.

# Counter Revolution

**Computers On The Farm**

You may be surprised to discover that the computer is equally at home on the farm as in the office.

An enterprising software engineer has designed a program called 'Optimiser' that minimises the cost of pig feed. A typical pig farm uses 400 tons of feed a month but the mixture of cereals changes daily as the piglets are fattened for market. Using the computer, the farmer gives the daily needs of the pigs and the nutrient value of the available cereals in terms of protein, energy and vitamins. The program then determines the most economical mixture.

With cereal prices jumping seasonally from £135 to £200 a ton, this software has helped farmers calculate their feed mixes more efficiently. A few 'human' pig problems remain. The animals like a regular diet and will turn their noses up at any new meal that is drastically changed from the last. The program, though, has had worldwide success and is sold from Thailand to Mexico

## The computer can save small businesses time and effort — and so boost their profits

The computer was born in military and university laboratories. The early machines were built to calculate how shells would fly when fired from a battleship in stormy sea and for forecasting the weather in mid ocean.

However, it didn't take long for the commercial applications of computers to be recognised. Initially only the largest of companies could afford the capital outlay for a business system. But the micro revolution of the late 1970's has at last made the power of the computer available to the small business.

How can a spin-off from military scientists help, say, the local newsagent? The computer can accept and store large amounts of data. It can rearrange the information that it is given haphazardly into useful forms. In the newspaper shop there are large stocks of pens and paper, sweets and chocolates, and of course a wide range of daily newspapers and periodicals.

Sales figures are entered into the newsagent's computer. A program is used that checks stock levels. Whenever the reserves of a particular item fall below the 'threshold' number a message is given reminding the newsagent to reorder. A standard reordering form can be called up from the memory. The newsagent then adds in the particular details of that item, and the computer prints out the form.

But the computer does not stop there. The time-consuming tasks of working out wages, assessing VAT, and doing the annual accounts all fall within the realm of the computer.

Computers are not merely convenient. Men could never have landed on the moon without computers. There are problems in rocket design and navigation that can be solved only by computer. The power of the computer opens new worlds to humanity which were closed before. But to the owner of a small business who may wonder what computers can do that can't be done manually, the simple answer is — they make money. Activities that formerly required such organising and administration that they consumed as much money as they earned now, with the advent of the microcomputer, become profitable.

The newsagent's original occupation was the distribution of newspapers. But the small profit margin on each paper forced him to diversify into other goods. Now the computer can bring profitability back into newspaper delivery by keeping records of the daily requirements of each household. Customers may now change to different newspapers or magazines at weekends and from day to day, without throwing the newsagent's system into confusion.

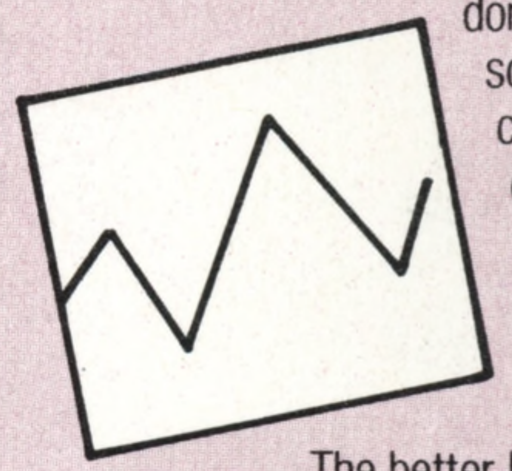
First the correct number of each newspaper is ordered from the publisher. Early in the morning the machine prints out the delivery requirements, dividing the households into networks of streets. Every delivery boy or girl receives his or her own personal list. They each leave the shop clutching a list and a precisely arranged bundle of papers.

The computer lists the houses in the order to be

## The Small Business System

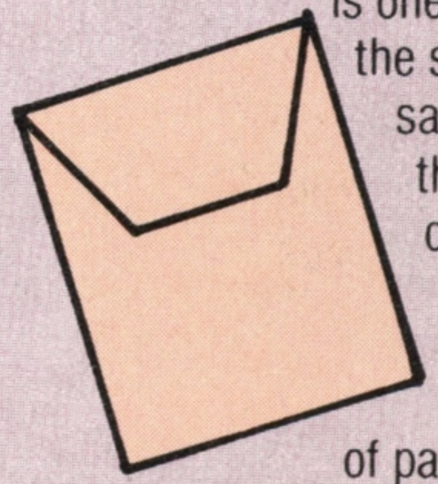
**A computer system suitable for a small business will need to be a high quality machine capable of withstanding hours of everyday use. A good keyboard is essential and disk drives will also be needed to 'run' the business software. The NCR Decision Mate illustrated here costs around £1,500 and is a typical system. The only other item needed is a printer costing £300 to £400**

**Accounting**



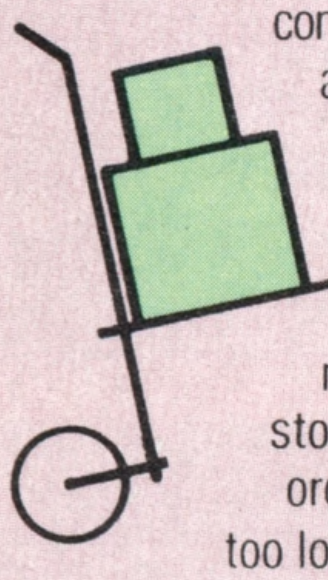
All of the company's accounts can be done using special accounting software. Sales ledgers, cash books and standard double-entry bookkeeping can all be automated to produce yearly, quarterly or monthly accounts in a form acceptable to accountants or auditors. The better kinds of accounting software comprise 'modules' that work together so that new data supplied for one section of the software (the sales ledger, for example) automatically update all the other parts of the accounting system. The advantage for the businessman translates into fewer man-hours spent in routine bookkeeping, more accurate accounts and higher profitability at the end of the day

**Payroll**



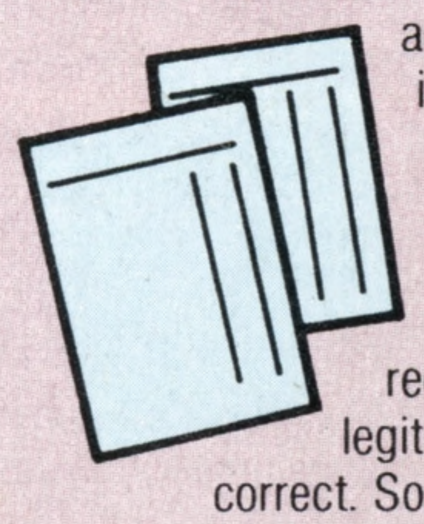
A payroll program is one that computes the salary and prints the salary slips for all the members of the company. Payroll programs can be|configured to take into account hours worked, basic rates of pay, standard deductions including tax and social security, overtime and so on. Such programs usually work by asking the computer operator a series of simple questions: 'How many employees are on the payroll?', 'Are salaries paid monthly or weekly?' etc. These can be answered by typing the information in on the computer keyboard

**Stock Control**

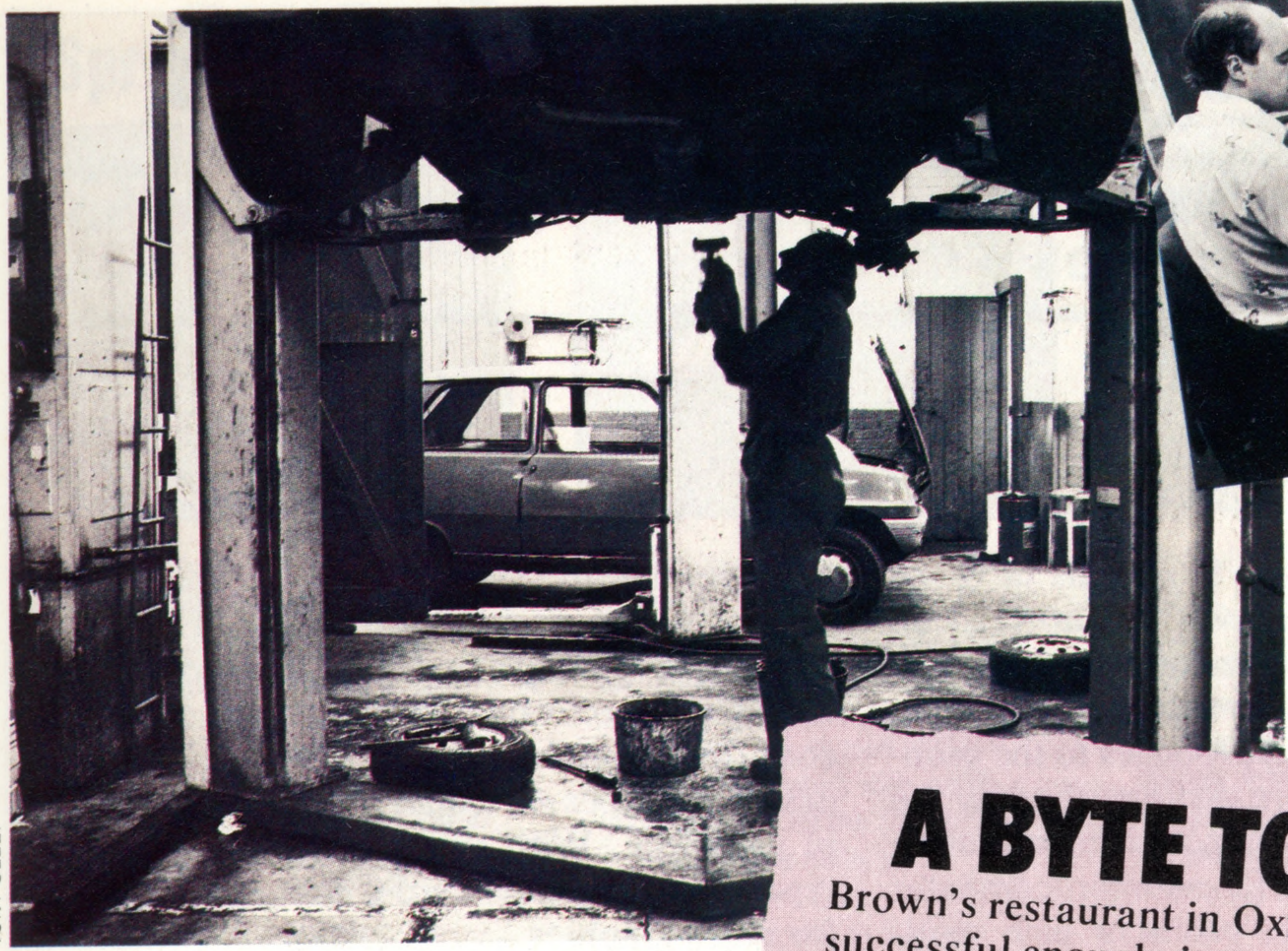


A stock control program gives the computer operator access to information about how the company's stock is being depleted. If one of the big sellers is Peanut Treets and the operator types in (in response to a question on the screen) the number of packets left on the shelf, the stock control program can issue re-stocking orders as soon as stock levels fall too low. Some stock control programs can take input from a bar code reader (see page 40)| when items are sold, so that both the sales price can be indicated on the cash register and the inventory can be updated simultaneously

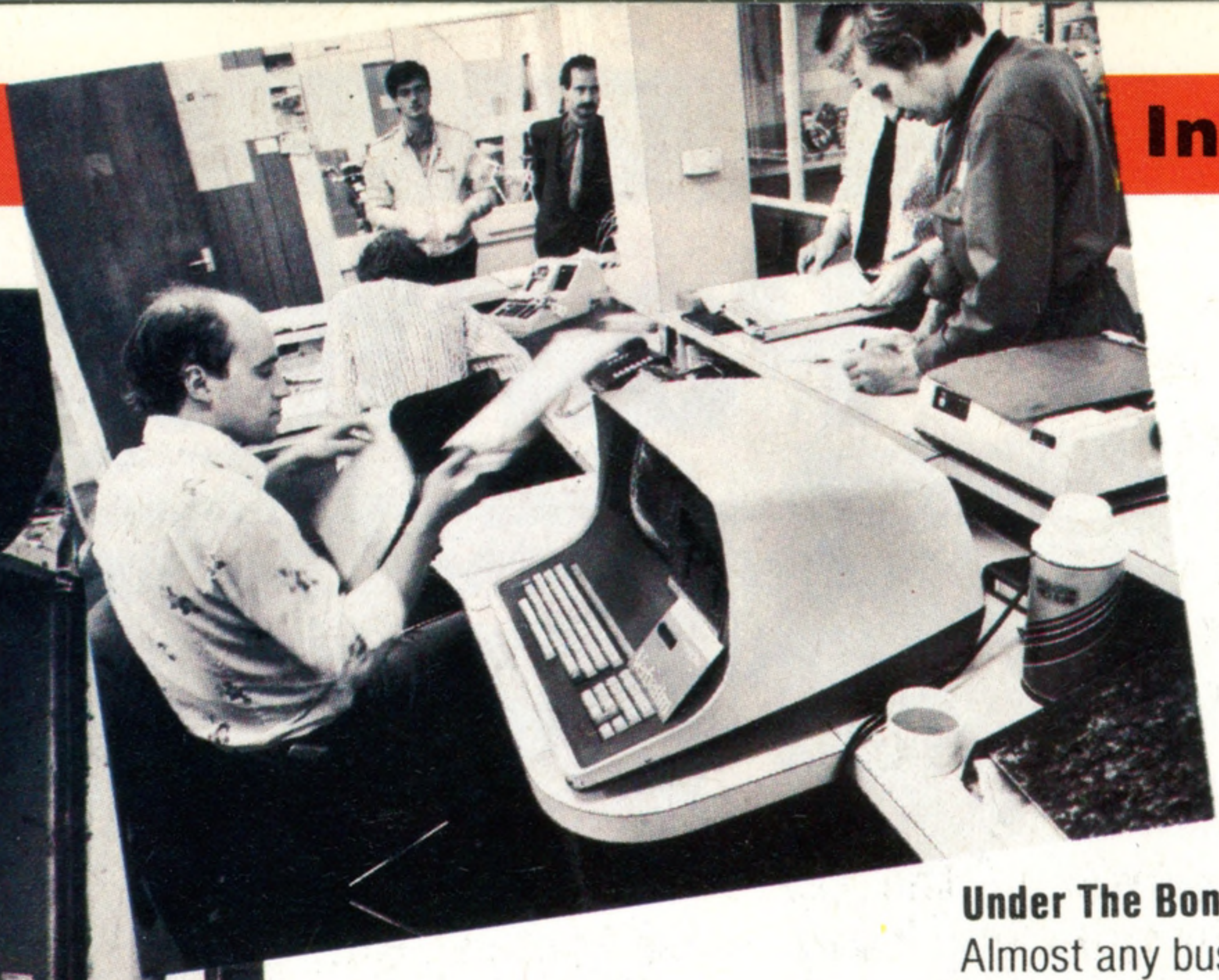
**Invoicing**



An invoicing program generates printed invoices with less trouble and more accuracy than a typist. When an invoicing program is 'run', the computer operator is asked questions and enters the appropriate answers on the keyboard. The program looks up records to check that the invoice is legitimate and that the details are correct. Some invoicing programs are able to cross-reference with the stock control program to keep all the books in balance



TONY SLEEP



**Under The Bonnet**
Almost any business can benefit from a computer. When your car goes in for its MOT, the computer displays the list of checks that have to be made. The mechanic works his way down the check list, entering each spare part as it is used and the time taken for the job. Once every safety check has been made the computer will calculate the bill and invoice the customer for parts and labour.

Computers take care of the routine chores and leave you free for more creative work

## A SLICE FOR ALL

One day at Gilson's Bakery in Colnbrook, a computer moved in. Nothing sinister, though. The owner realised that the ready mix of standing orders: 20 rolls here, 400 loaves there, could best be dealt with by a computer.

The new machine also took over weekly invoices, payroll, and calculations of delivery patterns, recipe costings and discounts.

Gilson's played safe with a small order of hardware consisting of PET 8096s, 8050 floppy disk drives, and Mannesman Tally printers.

The tough part was entering 350 standing orders on the files. But once this chore was finished, the micro could work unattended over the weekend.

## A BYTE TO EAT

Brown's restaurant in Oxford seemed successful enough; it was usually full of diners. But its management were worried about how much overheads, and wages were eating into the takings.

A computer provided some answers. By recording and despatching orders, and providing accurate bills, it minimised errors. Closer monitoring of customer demand also led to a profitable adjustment in opening hours.

The company wanted to maintain a bonus scheme, and that required an analysis of staff productivity, so Brown's invested in an Apple II. They also bought software including a payroll package, that saved two days a week in accounting, and VisiCalc for stock control and recipe costing.

## FRINGE BENEFITS

At the smart Schumi hair salon in London, a personal computer asks some personal questions. The client is quizzed on hair condition, colour treatment, and perming.

The hairdresser keys in the responses, and works to precise instructions from a computer printout specifying a product, and how to use it. The client also takes away a personal hair-care program.

The diagnostic programs are simple to operate and take a few minutes to run. Schumi leases the system, based on a 10 Kbyte DMS 80F micro, from Goldwell, a cosmetics supplier.

Heinz Schumi views the system as a good training aid. In his opinion, it enables the hairdresser to colour hair to a high standard.

followed on the delivery route and states what has to be put through each letterbox. At the end of the month the computer calculates every householder's bill and prints out an accurate invoice. Profitability comes back, through the computer, to a highly labour-intensive business.

Many other business programs are available in software packages. Word processing programs are among the first to appeal to businessmen. All the corrections on a word processing program can be painlessly carried out on the screen. When they have been completed, a perfect copy is printed over and over again. Although repetition would be tiresome for the businessman, the computer never tires.

Word processing is one area in which the computer excels. But as technology advances more and more people will have computers available at work or home. Direct communication between computers will eventually replace the posting of invoices and the ordering of stocks by letter. The business computer will be as indispensable as the telephone is today.

# Painting By Numbers

## Three fundamental types of computer graphics and how you can create images with them

A painting is built up from thousands of brush strokes applied to a bare canvas. But how can an artist create an image using a computer?

Three main systems on microcomputers are used for graphics. These approaches to the problem of graphical design are distinguished by the degree of control that they offer over the resolution or grain of the final image. Every microcomputer uses either block graphics, pixels or high resolution graphics.

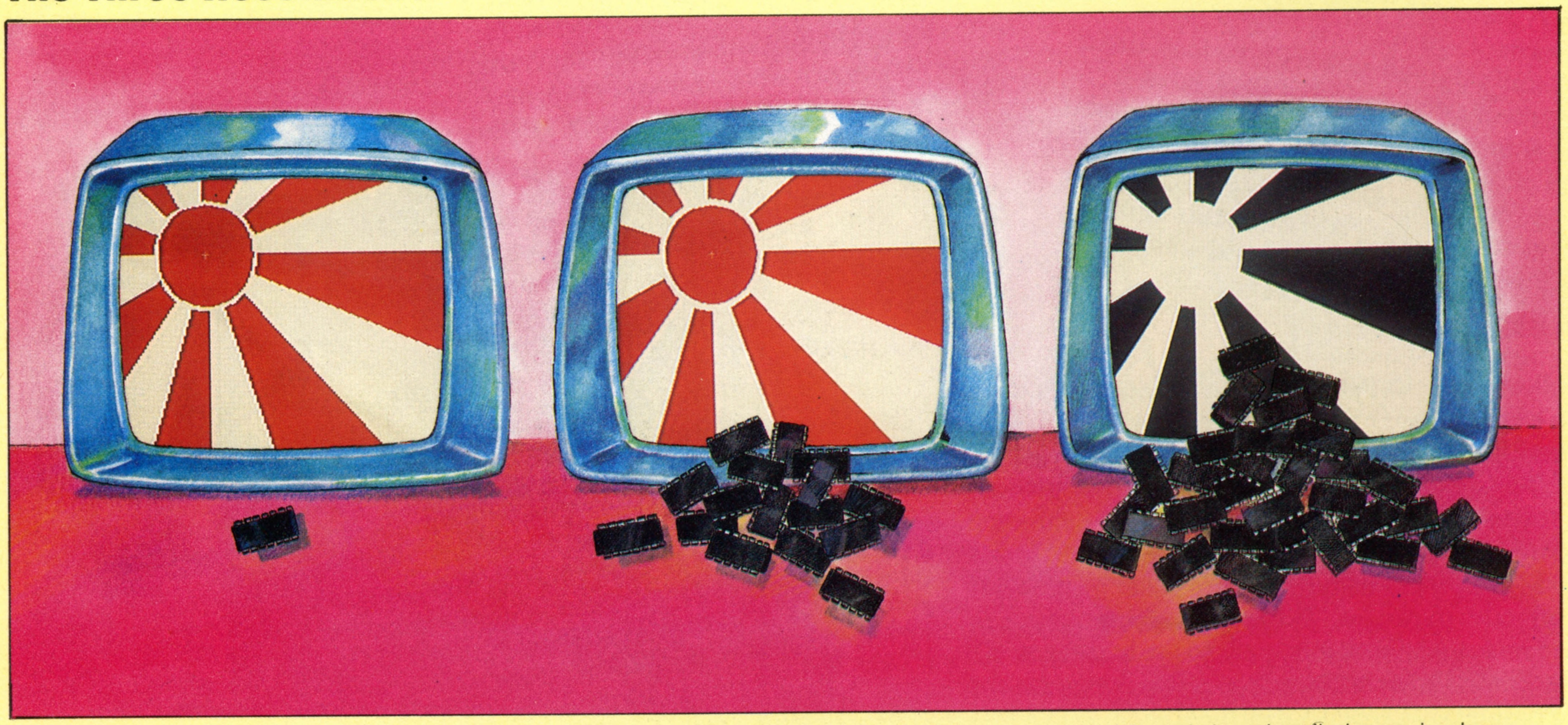
In high resolution graphics, the artist aims at controlling the individual phosphor dots on the television monitor. The only limitation is the memory size of the computer. This determines how closely the computer's screen memory models the television screen. With a 32K computer every phosphor dot has a counterpart in the computer's model of the screen.

In block graphics, what the artist loses in control over the individual dots that make up the screen, he gains in convenience. Elementary shapes are already constructed and are available through the software to build an image. They are controlled directly from the keyboard, and the predesigned shapes are usually displayed on the front of each key. With the touch of a shift key, a typewriter-style keyboard is turned into a palette for block graphics.

Each shape is formed within a small matrix of dots, eight rows by eight columns. Some microcomputers even offer the facility to define your own block characters. A minor program is used to define the new character and add it to the computer's range.

Pixels fall between block graphics and high resolution graphics. They give the artist control over a picture cell (hence the name 'pixel') which contains more than one individual phosphor dot

### The Three Resolutions



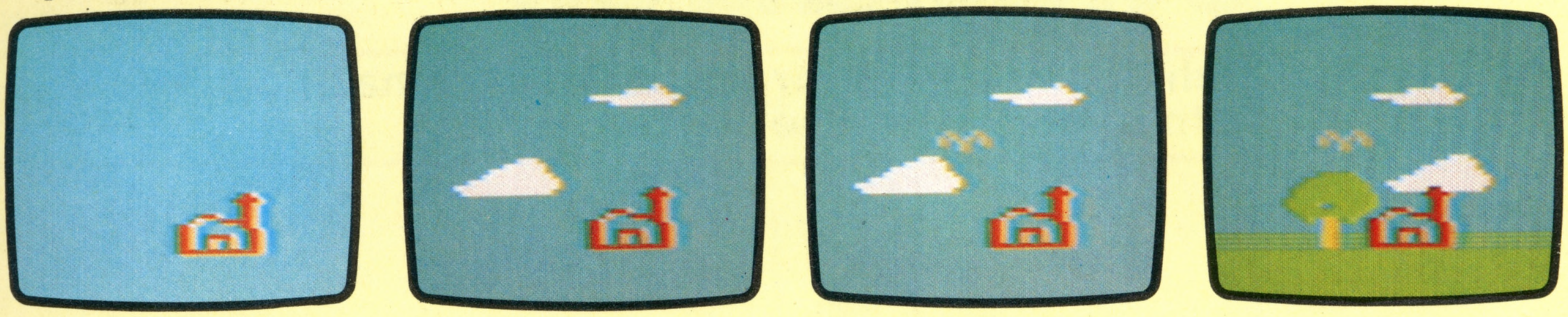
In low resolution block graphics, as little as 1 Kbyte of memory may be enough to store all the details of what should appear on the screen. If there are just 40 blocks in each row and only 24 rows down the screen, the total number of blocks is 960, just under a kilobyte (a kilobyte is actually equal to 1,024). A kilobyte of memory for the screen can therefore store up to eight bits of information (255 different combinations) for each of the blocks on the screen

In medium resolution, the detail shown on the screen is higher so more memory is needed. Eight kilobytes of RAM could allow 65,536 dots of light on the screen to be either on or off. If the display is in colour, some of this memory will be needed to specify the colour of each dot, leaving less memory for the number of dots. If we wish the display to be in eight colours, three bits will be needed to specify the colour of each dot

A high resolution colour display requires large quantities of memory. High resolution displays sometimes have as many as 640 dots on each line and 240 lines down the screen. This gives a total of 268,800 dots. If only one colour is to be displayed, 33,600 bytes (268,800 ÷ 8) would be adequate. Each bit would be either a zero or a one, and this corresponds to each dot on the screen being either off or on

JO LAWRENCE/IAN McKINNELL

## Sprite Graphics



Sprites is the name given to a way of making more realistic and easier to use pictures on your computer system. Originally developed by Texas Instruments, sprite graphics are now available on several home computers, including Texas Instruments TI99/04A, the Commodore 64, Atari Computers and the Sord M5.

In conventional graphics, pictures are built up on a single screen, just as you would paint a picture on a sheet of paper. With sprites the computer artist has several 'planes' or layers, each of which can have its own pictures. On some computers with sprite graphics, such as the Sord M5, there can be as many as 32 separate planes.

The easiest way to think of these planes is as sheets of clear plastic. If the sheet 'nearest' to the viewer has a picture of a tree, while the sheet behind it has a picture of a cloud, the cloud will be seen to pass behind the tree as if drifting across the sky.

By putting different elements of the picture on separate 'planes', convincing three-dimensional effects can be created.

Sprite graphics systems have several other advantages too. The character 'drawn' on any one of the planes is called, in computer jargon, an 'object'. Having created an object (a bird, for instance), the computer programmer can forget the details of how the picture was built up. If he wants it to move across the screen, he can specify a speed and a direction.

but less than an eight by eight block of dots. Each pixel can be called up individually and located in the desired position on the screen.

In high resolution graphics two BASIC commands are used for line drawing. MOVE identifies the beginning of a line and DRAW makes the line appear. Each end of a line is identified by a pair of numbers representing the row and column on the screen where the end falls. Rows are usually numbered in order from the top of the screen to the bottom, while columns are numbered from left to right. So the dot in row zero and column zero falls in the top left-hand corner of the screen.

The program below can be run on the Lynx and the BBC Micro. Remember to set your micro into graphics mode. Type this program in exactly as it appears. It can be run on any computer with high resolution graphics by replacing MOVE and DRAW with equivalent commands (see the 'Basic Flavours' box).

```
10 MOVE 100,50
20 DRAW 92,95
30 DRAW 57,125
40 DRAW 100,110
50 DRAW 143,125
60 DRAW 108,95
70 DRAW 100,50
```

When you RUN this program a 'Trinacria' shape will appear on the screen. This shape resembles the island of Sicily. (It is so called because Trinacria is the old Latin name for the island). The program scheme is suitable for drawing any shape consisting of a network of connected lines. The numbers after the DRAW command indicate the row and column position of the point to which the shape is to be extended. It is possible to draw anything on the screen using just these two commands. The only limitation is the resolution of the computer screen and your perseverance, since even a curve must be represented by dots. And MOVE and DRAW give you access to these dots.

The next program gives you an image of a cone by plotting a pattern of circles. As it is written this program will run on the Sinclair Spectrum. Other computers, including the Dragon and the Oric, also possess the CIRCLE command (see the 'Basic Flavours' box).

```
10 FOR K = 2 TO 40
20 CIRCLE 40 + K, 40 + K,K
30 NEXT K
```

These two programs illustrate how a computer can generate graphic displays through commands acting on numbers. However, a digital approach will always fall short in attempting to imitate a continuous flowing process. For this reason there are special hardware devices for designing images. These can be attached to microcomputers and free the artist from the need to enter the thousands of numbers needed for fine images. The digitiser is one such device. The artist draws his design with a special 'pen' and the digitiser translates the movement of the pen into the row and column numbers that the computer accepts.

## Basic Flavours


To run on the BBC Micro the program must be preceded by:
5 MODE 0


To run on the Oric the program must be preceded by:
5 HIRES
and line 20 must be replaced by:
20 CURSET 40 + K, 40 + K, 0
25 CIRCLE K, 1
To run on the Dragon the program must be preceded by:
5 PMODE 4, 1 : SCREEN 1,1 : COLOR 0,5 : PCLS
and line 20 must be replaced by:
20 CIRCLE (40 + K, 40 + K), K

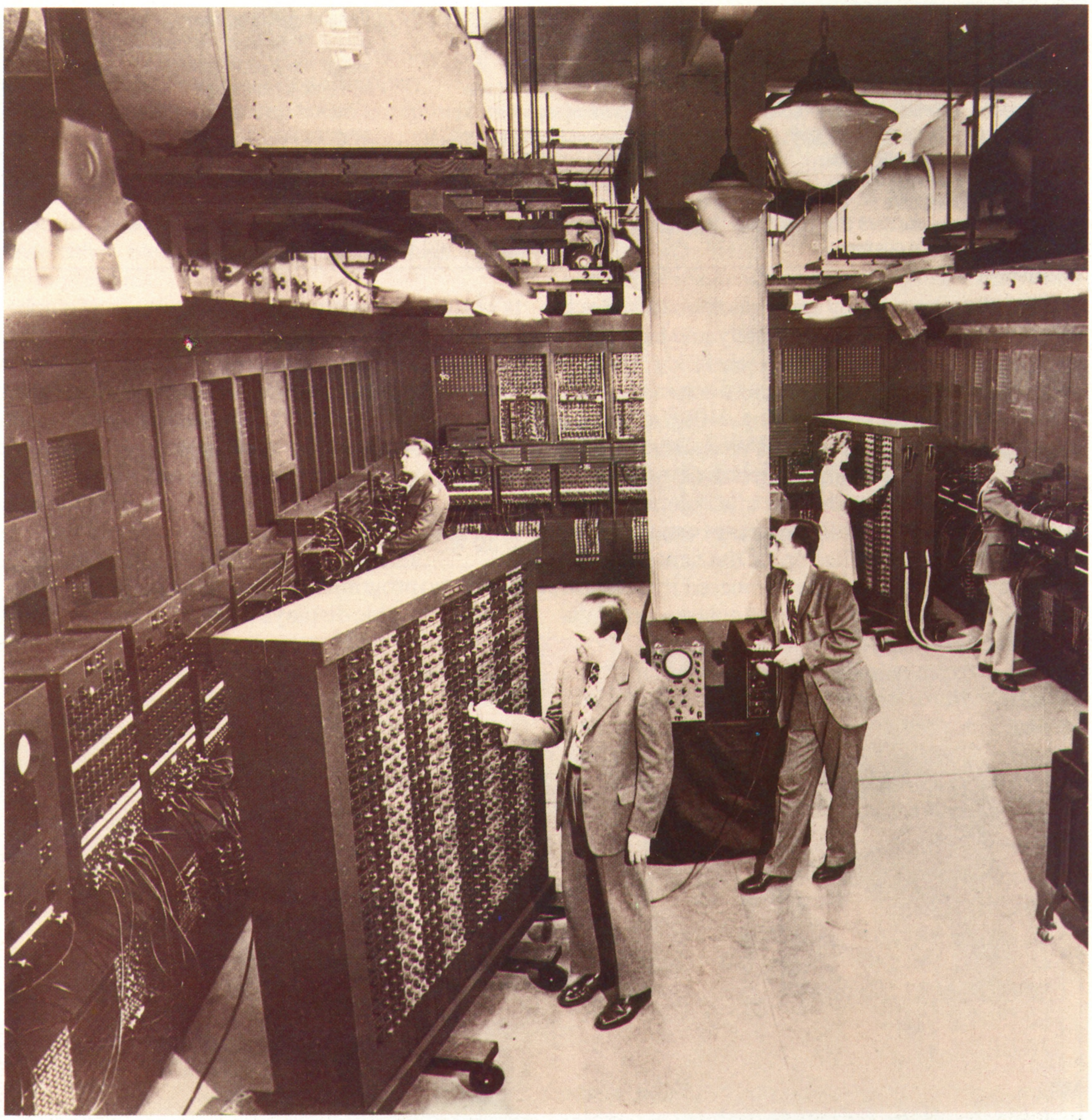
# Switching On

## Behind the computer revolution lie amazing advances in 'miniature engineering'

**The First Valve Computer**
In 1943, at the height of the Second World War, a colonel in the US army requested a calculating machine for the artillery. The challenge was taken up by the University of Pennsylvania whose invention was presented in 1946, having taken 7,237 man-hours to perfect.

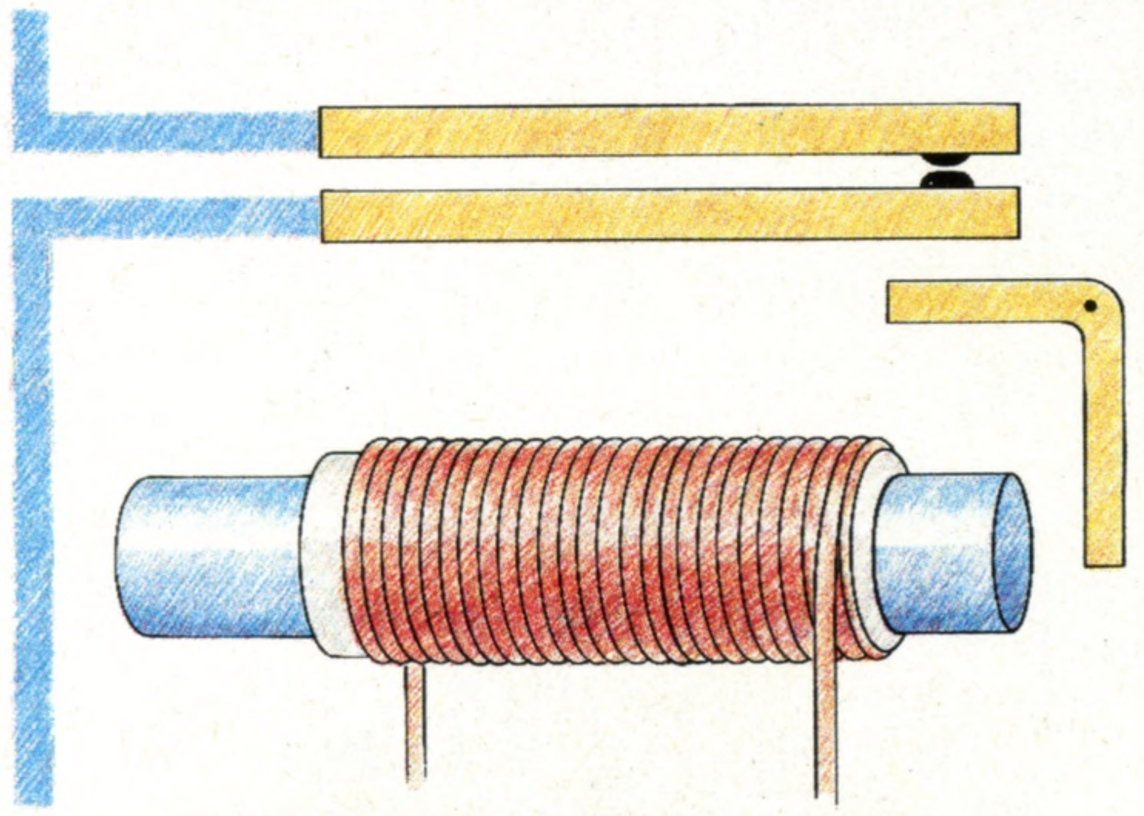
The machine was given the name ENIAC (Electrical Numerical Integrator and Calculator) and was the earliest valve computer. ENIAC used 18,000 valves, 1,500 relays and emitted the heat equivalent of 200 kilowatt fires. This enormous construction was housed in a room 9m by 30m.
Memory and reliability were the early problems. ENIAC could only store 20 10-digit numbers and all the programming had to be done by rearranging the wiring. In 1952 over 19,000 valves had to be replaced because the machine could only run for about two minutes before valves started to burn out.

ENIAC's life was short-lived and it went into retirement in 1952



© THE SCIENCE MUSEUM

**Relay Switch**
When current is passed through the coil that surrounds the iron rod, a magnetic force is created. This force attracts the bent strip of iron which is pivoted at the right angle bend. The strip turns on its pivot and presses the two contacts together thus closing the switch



TONY LODGE

The modern computer contains millions of small electronic switches. These are fundamental to the design of computers; without them the technological revolution that has occurred since the Second World War could not have taken place.

In 1938, an electrical engineer, Claude Shannon, demonstrated that logical operations could be performed using electrical switching circuits. Since it was apparent that the operation of a computer consisted of a sequence of logical operations, the search was now on to make an electronic switch.

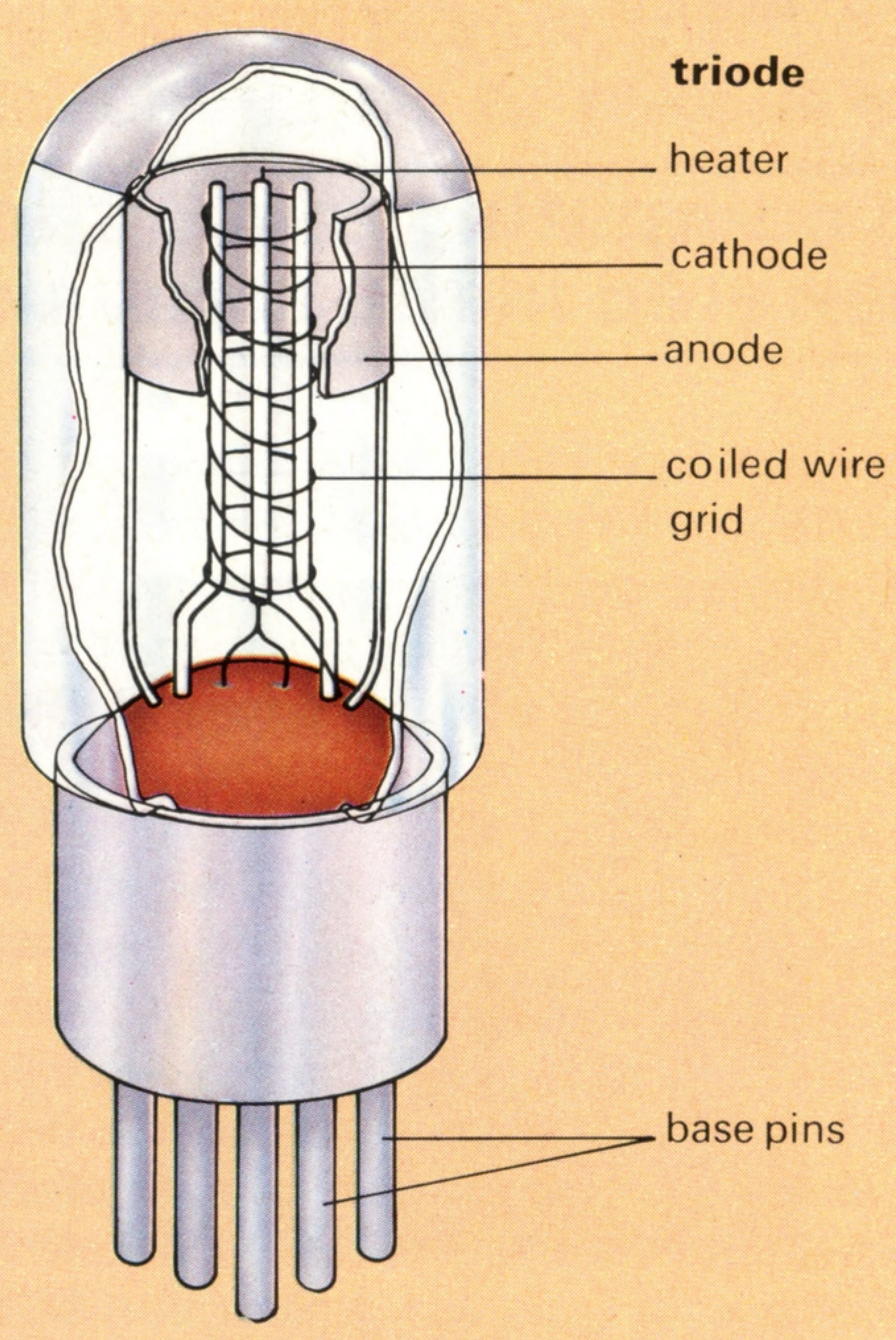
The first attempt resulted in the 'relay'. This switch was successfully used in pioneering

## Valve

The device illustrated is a triode — a valve containing three electrodes, which are housed in a glass tube. The cathode (negative terminal) and anode (positive terminal) are separated by a coiled wire 'grid'. When the cathode is heated, it emits electrons, which are negatively charged and are attracted to the anode. The emission of electrons is promoted by a special chemical coating on the cathode.

The grid does not interfere with the flow of electrons unless a negative voltage is applied to it. It then repels the electrons and prevents them from passing through to the anode. The valve can thus be used as an electronic switch, which is turned off by applying a small negative voltage to the grid. 'First generation' digital computers contained thousands of valves used as switches in this way

In addition to the three fundamental components (cathode, anode and grid), most valves contain a number of additional elements to improve performance. The principle of operation, however, remains unchanged




COURTESY OF MULLARD LTD

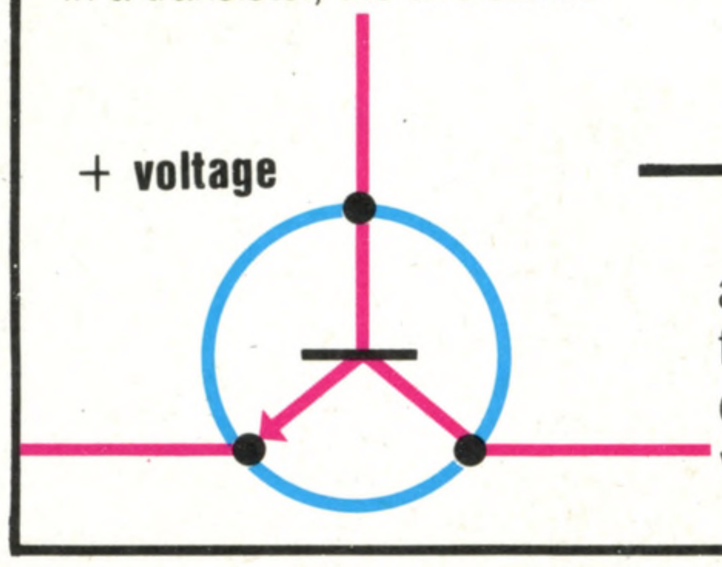
computers, but its design limited the growth of computers and their power. The relay was not wholly electrical in its operation and the mechanical components led to frequent breakdowns, unreliability and slow operation.

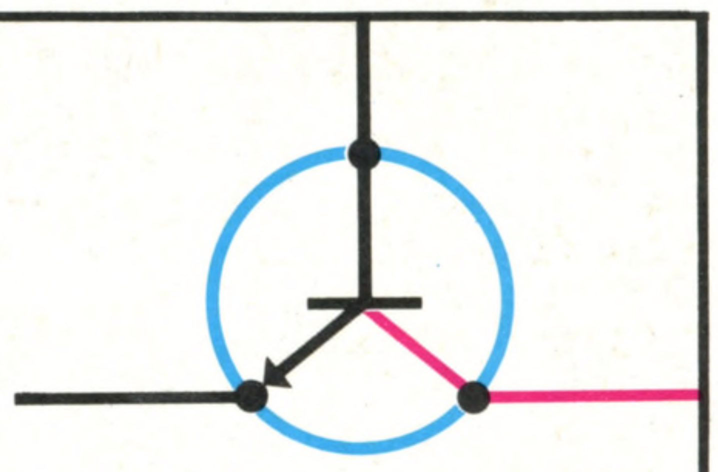
The first generation of working computers was characterised by the use of the valve in the essential role of the switch. These were completely electronic in operation and consequently faster. However they used large amounts of electricity (this was expensive and caused problems with generated heat) and were very bulky and not yet wholly reliable.

The invention of the transistor heralded a new generation of computers. Transistors are theoretically similar to the valve in operation, but superior in performance, smaller, and cheaper to manufacture. These advances took the computer out of the universities and military establishments and into the commercial world.

Today's computers still use transistors as switches, but the transistors are no longer discrete, separate items. On a silicon chip the size of a fingernail, there can be as many as a quarter of a million transistors, each one too small to be seen by the naked eye. Tiny though they are, each one is still a switch. By packing the thousands of switches needed to make a computer work onto a small chip of silicon, further dramatic savings in cost have become possible. The most expensive and powerful computers from the 1950's that filled a whole laboratory have been reduced onto a single chip, the powerhouse of today's micro. Computers are now small enough and cheap enough for almost anyone to own.

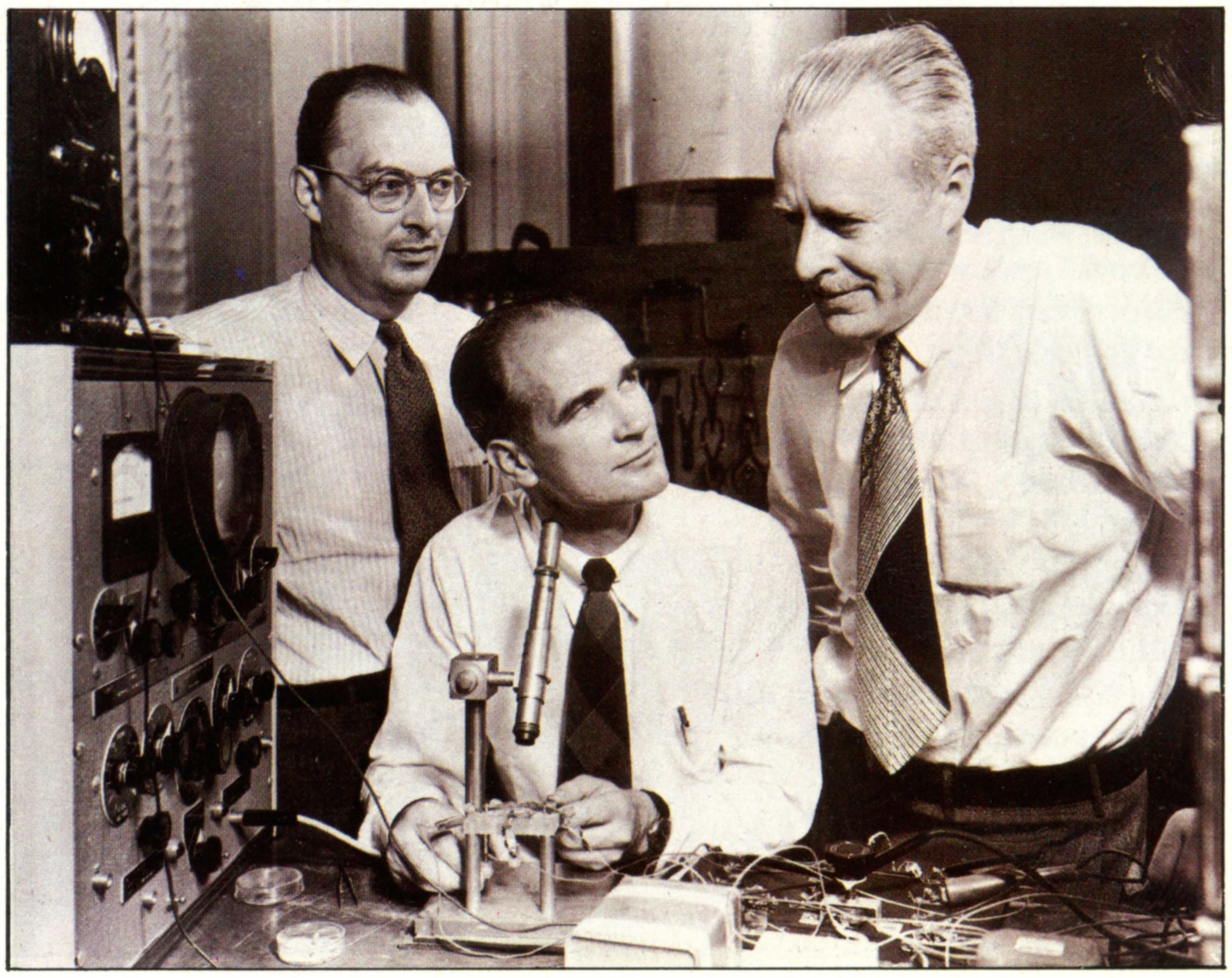
**The Transistor Switch**
A switch is either on or off. In a transistor, the two states



are represented by a current flowing or not flowing. It is controlled by applying a voltage to the control wire

**The Inventors Of The Transistor**
The 1956 Nobel Prize was awarded to the team whose research led to the invention of the transistor in 1947. Pictured here at the Bell Telephone laboratories (left to right): Dr John Bardeen, Dr William Shockley and Dr Walter Brattain



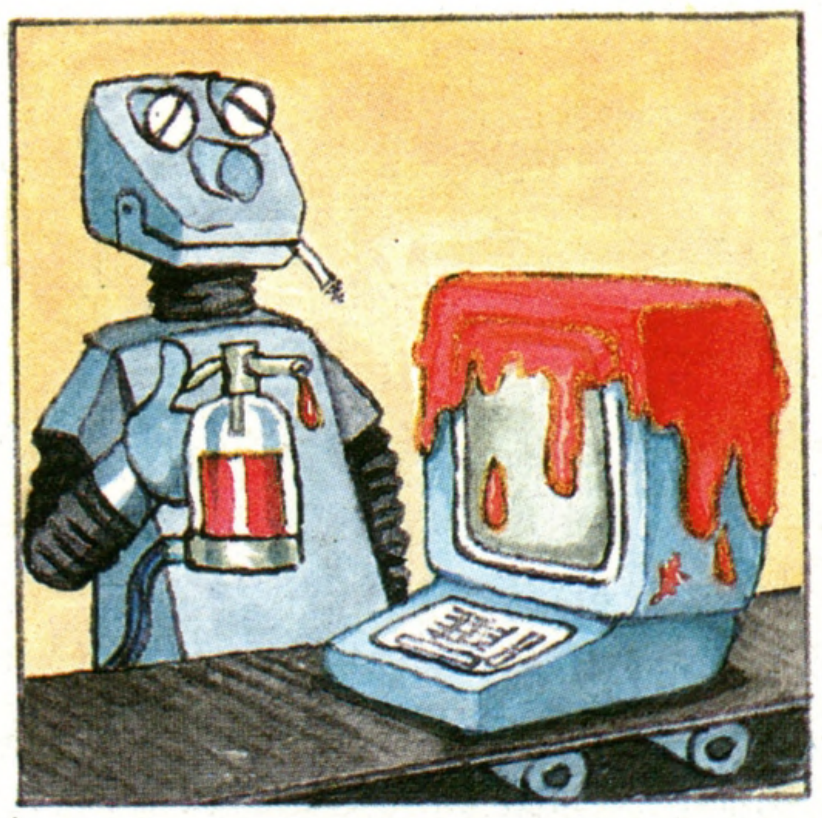
© ASSOCIATED PRESS LTD

# Questions And Answers

## Questions about computing that often spring to mind but are rarely answered in the manuals and magazines

**Could computers have emotions?**
Computers do not and could not have emotions at the moment. The really interesting question is: Why not? The computers of today are not intelligent — they cannot think for themselves. How long it will take to produce thinking computers is not known, but probably it will be within the lifetimes of people living today. Some researchers hold the view that creative thought processes are inseparable from emotions. Computers that can think will, according to this view, be computers with emotions.

**What is the difference between a computer and a robot?**
Robots are mechanical extensions of computers; the arms and eyes do what the 'brain' of the computer tells them. The robots helping to build today's cars and stereo systems all incorporate microcomputers, but they are still fairly 'dumb'. Confronted with an unexpected situation, they simply don't know what to do. The robots of tomorrow will incorporate more sophisticated computers and robots with limited intelligence are just around the corner.

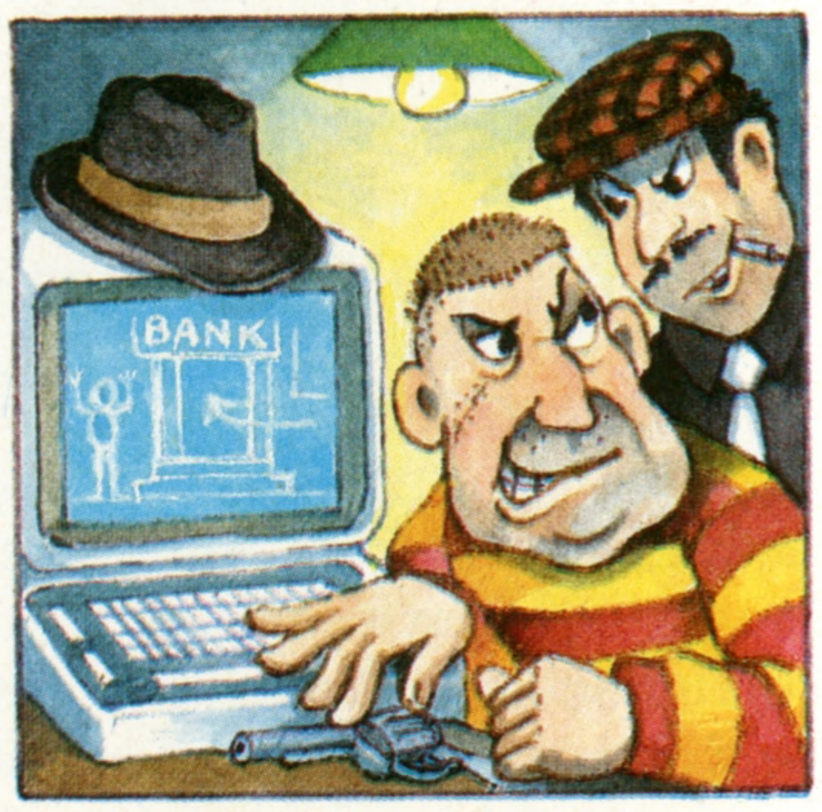


**Why is some software so expensive? Many games programs cost only a few pounds, but business programs often cost hundreds.**
Writing large programs, especially thoroughly tested business software, takes teams of highly paid programmers months or years of work. To recoup the huge financial investment and to make a profit, software companies have to sell their products at prices guaranteed to cover costs. A computer game may sell hundreds or even thousands of copies, so a retail price of a few pounds may ensure a profit. If the potential market is strictly limited, the retail price will have to be far higher. Many programs are highly specialised; a printer's estimating package (allowing estimates for printing jobs to be made quickly and accurately) has a potential market limited to the total number of printers in the country. An investment of hundreds of thousands of pounds will have to be recouped whether sales are measured in tens or in thousands.

**They say the silicon chip will throw millions out of work. How could a microcomputer make me redundant?**
The long-term social effects of the microcomputer are hard to predict, but what seems very clear is that we are witnessing the start of the second industrial revolution. Computers, particularly miniaturised and low-cost microcomputers, when linked to mechanical robots, can easily be adapted to replace expensive manual labour. Even skilled jobs are not safe. Bookkeeping and accounting can now be handled by computer programs, and newspaper typesetters' jobs are threatened now that journalists' word processors can be directly linked to electronic typesetting equipment. Computers can do complex arithmetical processes so quickly, and robots can perform complex mechanical operations so well, that fewer workers are needed to get a job done.

**Can computers be used to rob a bank or start a Third World War?**
Since computers can communicate with each other using ordinary telephone lines, it would be possible in theory to tap in to a bank's central computer and issue orders to transfer funds to your account. In practice things are not so simple. The banks use advanced data protection methods to ensure there is no unauthorised access to confidential information. The techniques used involve secret methods of encoding all the information. These codes are almost impossible to break and in many cases are not even available to the bank employees. One of the codes used for highly confidential information is so difficult to crack it has been estimated that the world's most powerful computer would take billions of years to do it.

Breaking into a military




DAVID HIGHAM

computer system would be even more difficult. Military computers generally do not use public telephone lines for this very reason. The microwave and satellite links used are not readily accessible to ordinary people — even dedicated computer buffs. Even if one were able to intercept a microwave link carrying computer information, the problem of cracking the codes would still remain.

# Commodore 64

## Commodore's latest home computer offers lively 'sprite' moving graphics, and uses your TV or hi-fi speakers to generate high-quality sound

The Commodore 64 is really the first of a new generation of home computers equally suitable for playing games, or assisting in the running of a small business.

Sixty-four Kbytes of memory as standard is enough to cope with sophisticated graphic displays, or business programs such as spreadsheets, word processors, and databases. Some compatability with the Commodore VIC-20 and PET business systems, has increased the range of available software further.

The 64's range of interfaces means that it will operate with most types of peripherals including the VIC disk drive and printers. A complete system, consisting of the computer, disk drive and a printer can thus be purchased for around £700.

Two of the 64's strongest features are: sprite graphics (see page 45) and full music synthesis. Sprites are visual objects created on the screen using high resolution graphics, which can then be moved around using simple commands — ideal for space invaders, aircraft, explosions etc. Such effects are possible without sprite graphics, but require far more programming. On the 64, sprites can be made to move, grow in size, shrink or change colour, or can be made to pass in front of or behind other sprites or stationary graphic objects such as background scenery — giving depth to a picture. It is even possible to detect when two sprites have collided — the cue for an explosion!

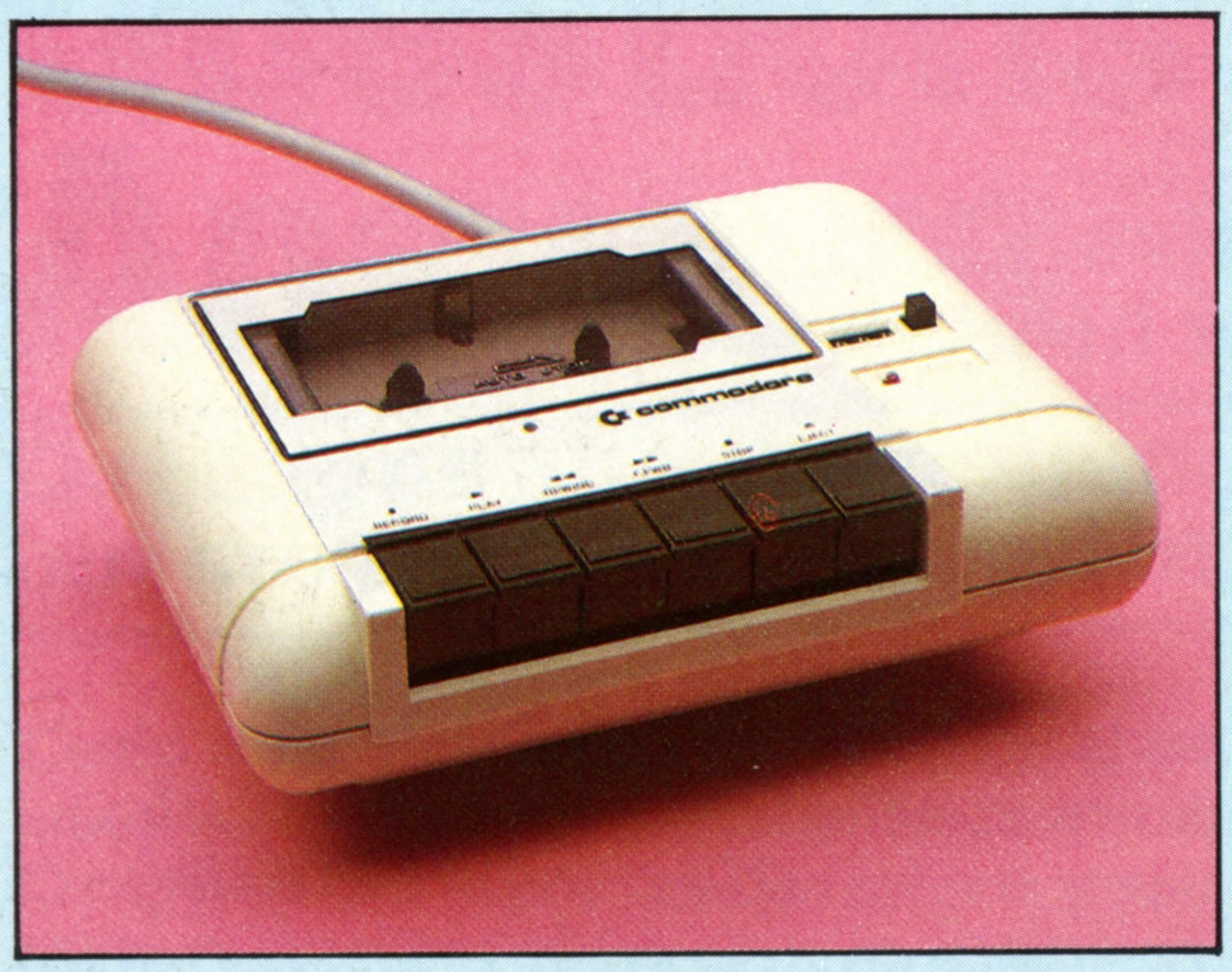
The sound synthesis is equally sophisticated, by contrast with the simple 'beeps' and 'squawks' of cheaper machines. In addition to having three separate voices (allowing chords and harmonies instead of just simple notes), the 64 permits full control over the various parameters that govern the sound or timbre of the note being played. In other words, the 64 can simulate a whole variety of musical instruments, and more abstract noises.

The weakness of the 64 is its BASIC language — which is virtually the same version as Commodore were using on their very first computers. Instead of a nice range of user-friendly commands to take advantage of the 64's otherwise superb features, most sophisticated operations require the unfriendly POKE command. Fortunately it is now possible to buy cartridge add-ons (such as Simon's BASIC —£50) to rectify this deficiency.


### Commodore 64 Keyboard

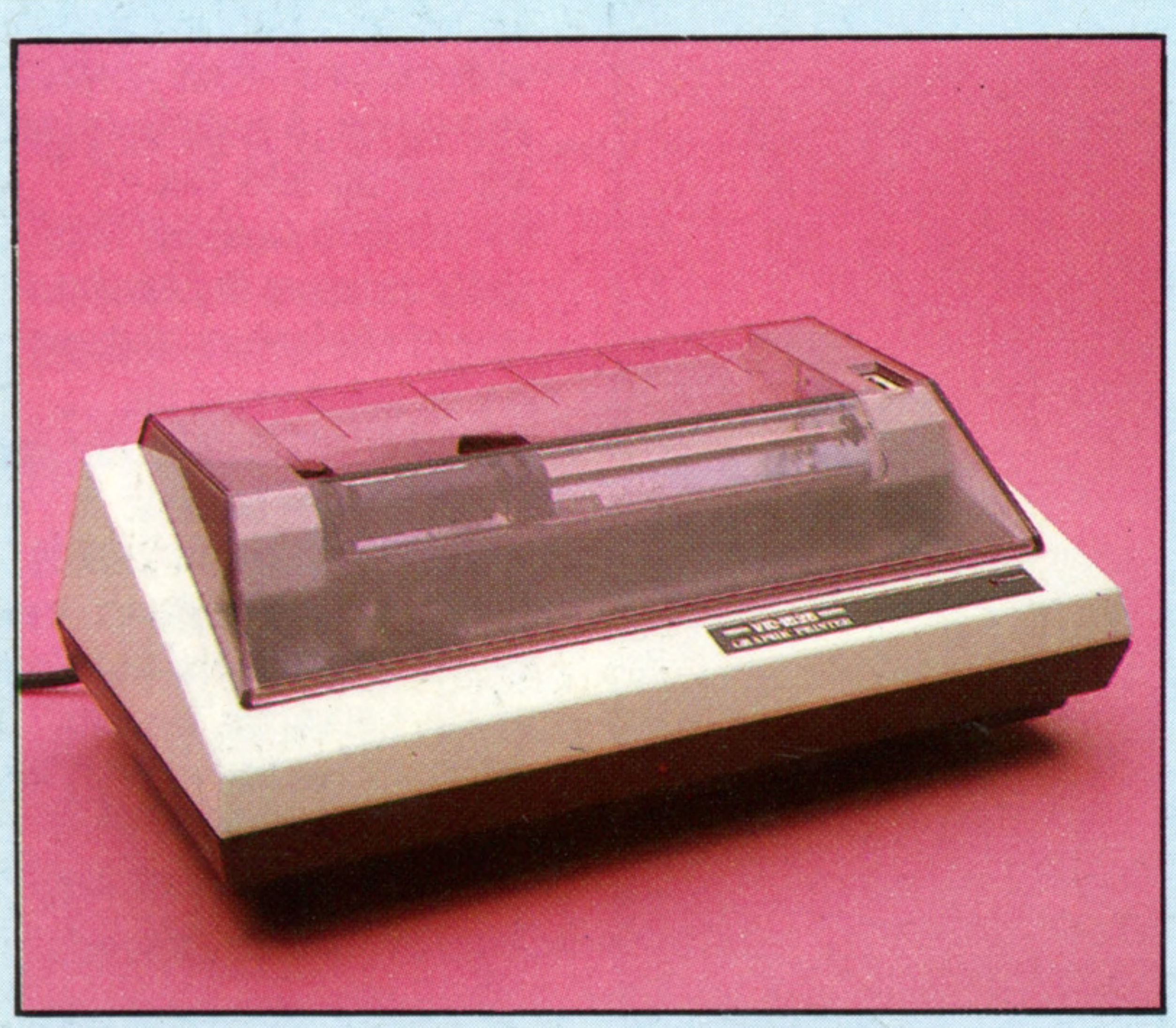
**The Commodore 64 has an excellent keyboard with 'sculptured' keys (contoured for easy typing). Besides the normal characters there is a wide range of block graphic characters. A multi-purpose key marked with the Commodore company logo shifts between the various sets of characters. Colours are changed using the CONTROL (CTRL) key and one of the top row of keys.**

**The four function keys to the right of the keyboard can be assigned special functions within a program, thereby providing shorthand entry of special commands.**

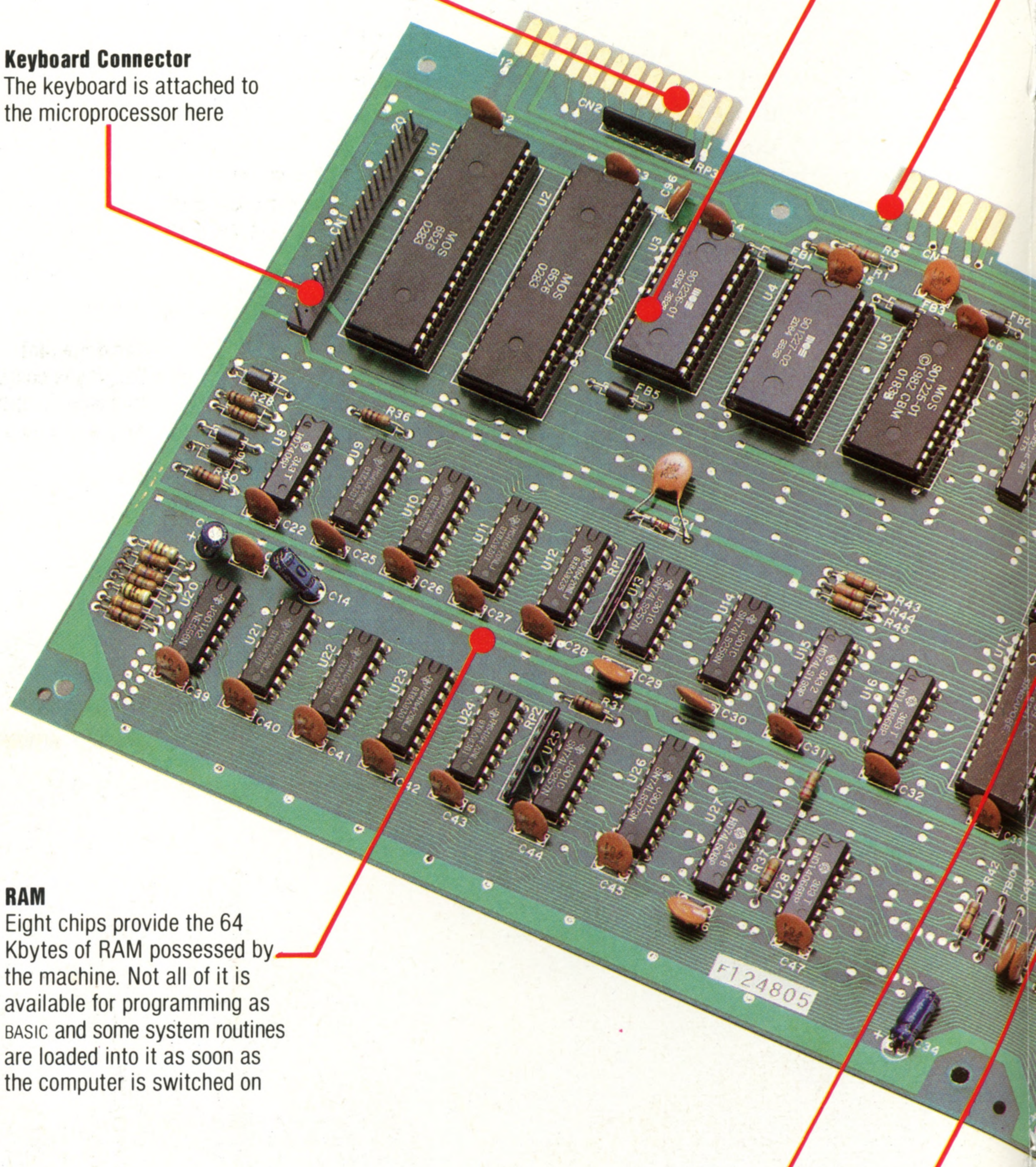


**1530 C2N Cassette Deck**
This is Commodore's own standard and reliable cassette unit and is designed to plug directly into the 64's cassette port. It draws its power from the computer which can also turn the cassette motor on and off


**1540 Disk Drive**
This single disk drive connects to the 64 via the serial port. One 5¼-inch disk provides 170 Kbytes of storage and the maximum time taken to access any stored information is 2 seconds



**1525 Printer**
This dot matrix printer can print all the characters that the 64 can display, including graphics characters. It is connected via the serial port, and prints at 30 characters per second




**User Port**
Using a 24-pin socket, a variety of devices can be attached to the computer. Various pins can be set as input or output lines

**Keyboard Connector**
The keyboard is attached to the microprocessor here

**RAM**
Eight chips provide the 64 Kbytes of RAM possessed by the machine. Not all of it is available for programming as BASIC and some system routines are loaded into it as soon as the computer is switched on

**Microprocessor**
The 6510 is the main processor. It controls all the operations of the computer as well as handling a good deal of the input/output

**Sound Chip**
The 6581 Sound Interface Device generates all the sounds that the 64 can produce

**BASIC ROMs**
These three chips contain the computer's BASIC language

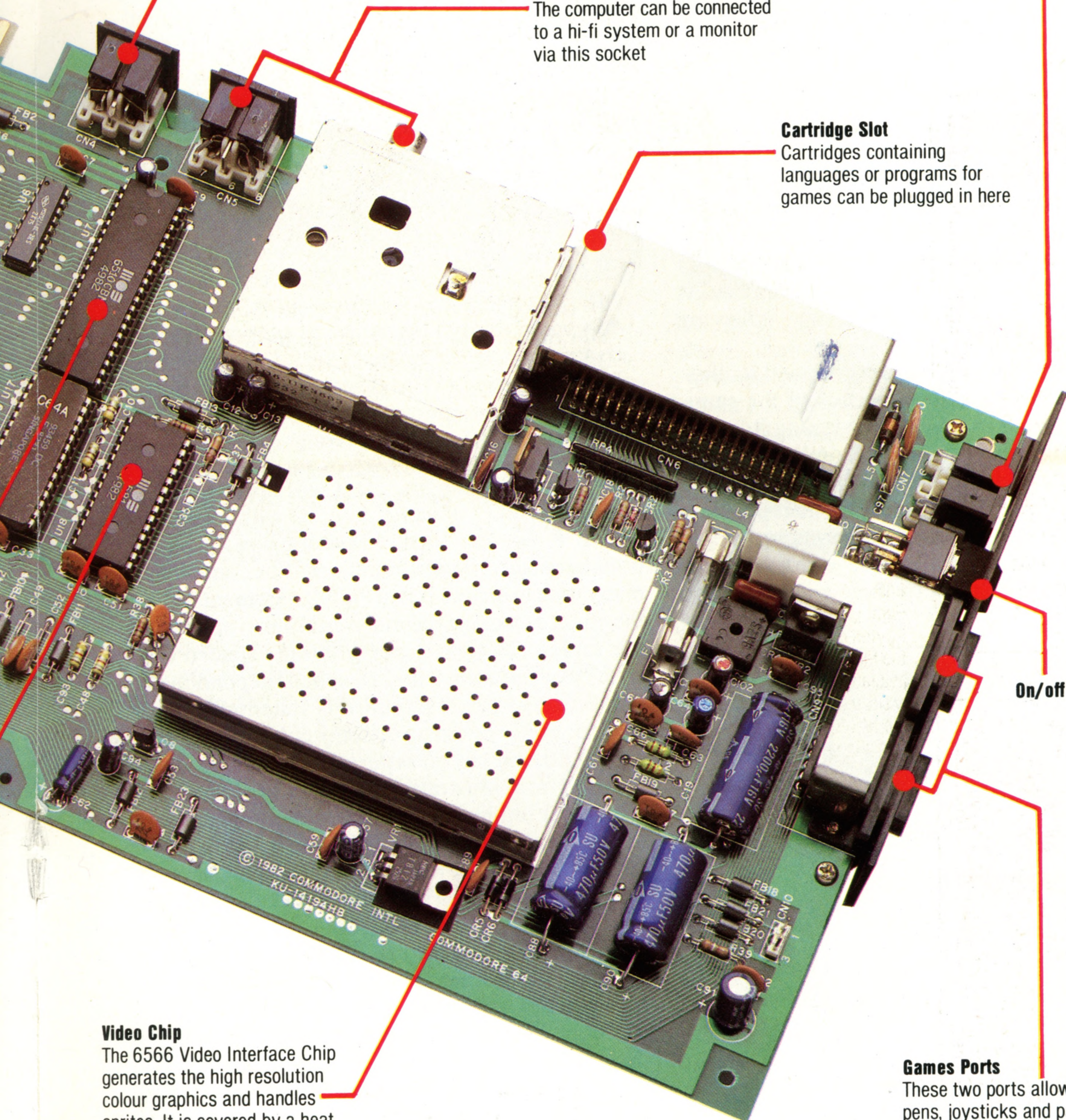
**Power Socket**
This is where the power supply unit is connected to the computer

**Cassette Port**
A special port for connecting Commodore's standard cassette deck to the computer. It will not accept other makes of recorder without special modifications

**Serial Port**
This is a socket for 'serial' communication with additional devices. In serial communication binary digits are sent one after the other

**Audio/Video connector**
The computer can be connected to a hi-fi system or a monitor via this socket

**Cartridge Slot**
Cartridges containing languages or programs for games can be plugged in here



**On/off switch**

**Video Chip**
The 6566 Video Interface Chip generates the high resolution colour graphics and handles sprites. It is covered by a heat protection unit

**Games Ports**
These two ports allow light pens, joysticks and paddles to be plugged into the computer

# COMMODORE 64

**PRICE**
£229

**SIZE**
404 x 216 x 75mm

**WEIGHT**
1820g

**CLOCK SPEED**
1 MHz

**MEMORY**
A total of 64 Kbytes. 20 Kbytes of ROM supply the operating system and BASIC. A maximum of 54 Kbytes of RAM is available to the user if the BASIC interpreter is not used

**VIDEO DISPLAY**
25 rows with 40 character positions. High resolution with 320 x 200 dots. 16 colours

**INTERFACES**
Cassette port, TV connector, cartridge slot, monitor connector, RS232 interface, user port

**LANGUAGE SUPPLIED**
BASIC

**OTHER LANGUAGES AVAILABLE**
FORTH, COMAL, PILOT, LOGO, UCSD, PASCAL, and other versions of BASIC

**COMES WITH**
Power supply unit, aerial lead, manual

**KEYBOARD**
Typewriter-style keys, QWERTY layout with 62 keys and four function keys

**DOCUMENTATION:**
The User's Guide is of the low standard associated with Commodore's manuals. The machine, its operation and language are described but not at a level that is entirely appropriate for beginners. It does not provide a comprehensive guide to the addresses of the special registers. This is particularly unfortunate since the sound and graphics of the 64 are programmed using these special registers.

For all but the complete beginner it might be better to ignore the User's Guide and acquire a copy of 'Commodore 64 Programmer's Reference Guide', published by Commodore Business Machines Inc. and Howard W Sams at £14.95

# To The Point

## Why you have to pay attention to every detail of punctuation when you write a computer program

You may well have noticed, in the program listing in the first part of Basic Programming (page 20) that there is a semi-colon at the end of line 50. The function of this punctuation mark in BASIC was not explained at the time, but is nevertheless very important. It is used in almost all versions of BASIC to concatenate printed sections ('concatenate' means 'join together'). Lines 50 and 60 on page 20 were:

```
50 PRINT "I THINK THE NUMBER YOU TYPED
   WAS ";
60 PRINT A
```

Line 50 printed the words inside the double quote marks. Line 60 printed the value of the variable A. Putting in the semi-colon caused the value of variable A to be printed directly after the words within quotes in line 50. If no semi-colon had been used, it would have been printed on the line following the words.

The program below has been designed to illustrate some of the useful properties of the semi-colon as it is used in BASIC. Try typing it in and running it. From now on, we will omit the <CR> reminder at the end of each line to indicate that you should press the RETURN key. This next program allows you to enter a range of temperatures in Centigrade (also known as Celsius) and have them converted automatically to their equivalents in Fahrenheit.

```
10  REM  PROGRAM TO CONVERT DEGREES C TO F
20  PRINT "ENTER THE LOWEST TEMPERATURE"
30  INPUT L
40  PRINT "ENTER THE HIGHEST TEMPERATURE"
50  INPUT H
60  FOR X = L TO H
70  LET F = X * 9 / 5 + 32
80  PRINT X;" IN CENTIGRADE IS ";F;"
    IN FAHRENHEIT"
90  NEXT X
100 END
```

Enter this program, LIST it to check that it has been entered correctly, and then RUN it. First you will be asked to enter the lowest temperature. Try typing in −5. Then you will be asked to enter the highest temperature. Try typing in 10. The program will convert all temperatures at one degree intervals from −5 to 10 degrees Centigrade to their Fahrenheit equivalents. You should get a 'printout' on the screen looking something like:

```
-6  IN CENTIGRADE IS  21.2  IN FAHRENHEIT
-5  IN CENTIGRADE IS  23  IN FAHRENHEIT
-4  IN CENTIGRADE IS  24.8  IN FAHRENHEIT
```

Notice that the columns are not very even because of the decimal points, but that each value in Centigrade is printed with its equivalent in Fahrenheit on a single line. After you have run the program a few times, re-type line 80 just as it is, but substitute commas wherever we have printed semi-colons. RUN the program again. As you can see, the printout becomes a complete mess.

To see why this happened, let's try a very simple program to compare the effect of commas compared with the effect of semi-colons. Type NEW<CR>. Then enter:

```
10 REM COMPARE ; WITH ,
20 PRINT "THIS LINE USES SEMI-COLONS"
30 PRINT "H";"E";"L";"P"
40 PRINT "THIS LINE USES COMMAS"
50 PRINT "H", "E", "L", "P"
60 END
```

When BASIC prints line 30 it will appear on the screen as HELP, whereas line 50 will appear as H E L P. See the 'Basic Flavours' box for variations between different machines. The comma has many uses in BASIC, but in PRINT statements it has the effect of making the individual items appear on the screen (or on a paper printout) spaced out, usually by between 8 and 16 spaces depending on the version of BASIC. If the PRINT statement is used without either commas or semi-colons, the items will be printed out on separate lines.

Apart from illustrating BASIC's use of the semi-colon, our temperature conversion program also revises several statements covered in the first two parts of the Basic Programming course. Lines 30 and 50 set variables L and H to the values for the lowest and highest temperatures we want to convert. Line 60 is the first part of a FOR-NEXT loop. It seems to differ from the FOR-NEXT loop we have encountered so far by using letters instead of numbers. In fact, there is no difference. The letters we are using here, L and H, are variables with numeric values corresponding to the values typed in at the INPUT L and INPUT H stage of the program. If, as suggested earlier, you entered −5 and 10, the statement FOR X = L TO H is therefore equivalent to FOR X = −5 TO 10.

Line 80 in effect says: PRINT the value of X (which starts at the lowest temperature and increments by 1 each time up to the highest temperature) followed directly on the same line (that's why we used the semi-colon) by the words in quotes, followed directly again (another semi-colon) by the value of F. If you look carefully at F, you will see that it is the current value of the Centigrade temperature, converted into Fahrenheit by multiplying it by nine, dividing it by five and then adding 32. The NEXT X line ensures that we go through the conversions until the upper limit in the FOR-NEXT loop has been reached.

Before going on to look at a more sophisticated variation on the PRINT statement, it is worth taking a second look at line 70 in our temperature conversion program:

```
70 LET F = X * 9 / 5 + 32
```

This line assigns a value to the variable F (which stands for Fahrenheit). The program first takes the value of X (the temperature in Centigrade), multiplies it by 9, divides that by 5 and then adds 32. The way this formula would be presented in an ordinary arithmetic book is $F = C \times 9 \div 5 + 32$. BASIC uses * for multiplication, / for division, + for addition and − for subtraction.

In ordinary arithmetic, and in BASIC too, the order in which arithmetical operations are carried out is important. Multiplication always has top priority, followed by division, followed by addition, followed by subtraction. If parts of an arithmetic expression are enclosed in round brackets, they must be evaluated first. If you want an addition to be performed first, before a multiplication, the addition part must be enclosed in brackets. For example, if you wanted to know how much money you had in your current account plus your savings account in dollars, you might express it in part of a program like this:

```
D = (C + S) * 1.5
```

If your current account has £600 (C) and your savings account has £1,300 (S) and there are 1.5 dollars to the pound, you will want to add the pounds first (C + S) and then multiply by 1.5 to convert to dollars. Without the brackets, the value of your savings account would first be multiplied by 1.5 and then the value of your current account would be added to the result — not what you wanted at all! Always be sure to check that the arithmetic parts will be calculated in the right order.

## Print Using

In order to look at a final refinement to our temperature conversion program, try typing it in again and RUNing it. Enter, say, −10 as the value for the lowest temperature and 10 as the value for the highest. As we have already seen, the printout on the screen is very ragged. This is because of the semi-colons used in line 80 in order to concatenate (run together) all the parts being printed, instead of printing them on separate lines. Which is fine, except that the space taken up by the figures — both the Centigrade and the Fahrenheit ones — varies. This has the effect of pulling the columns out of alignment and making the printout look untidy.

Almost all versions of BASIC have a special PRINT feature called PRINT USING. It allows the appearance of the printed numbers or words to be 'formatted' or tidied up. If you want to print the value of X and that value is known in advance to range from, say, −99 to 99, the figures can be printed out correctly aligned by using PRINT USING "###";X. The three 'hash' signs allow up to three digits, or two digits preceded by a minus sign, to be printed. If more than three digits are entered, they will not be printed out correctly. If, however, only two digits are entered (or only one) they will be positioned correctly. If decimal points are required, they can be included in the appropriate position within the hash signs. For example, the statement can take the form PRINT USING "###.##";X. Use one 'hash' sign for each digit. All the decimal points will line up automatically.

Modify the original program by changing line 80 and adding lines 82, 84 and 86:

```
80 PRINT USING "###";X;
82 PRINT " IN CENTIGRADE IS ";
84 PRINT USING "###.##";F;
86 PRINT " IN FAHRENHEIT"
```

LIST the program again and then RUN it. All the columns should now be lined up perfectly.

We will find out how to 'save' programs, so that they do not need to be re-typed every time, in the next instalment of the course.

## Exercises

■ Try entering a 'lowest temperature' of −1000. Why doesn't the program work this time? How would you modify the PRINT USING statement in line 80 to make it work?

■ Alter line 84 so that only whole numbers (no decimal fractions) are printed.

■ Write a program to convert a range of figures in pounds to dollars, using an exchange rate of $1.50 to the £.

## Basic Flavours

**PRINT USING**

This facility is not available on the Commodore 64, Oric, Spectrum, ZX81 or BBC Micro. However, the BBC can limit the number of decimal places to be printed, and this is achieved by using the following instruction:
@%=131594

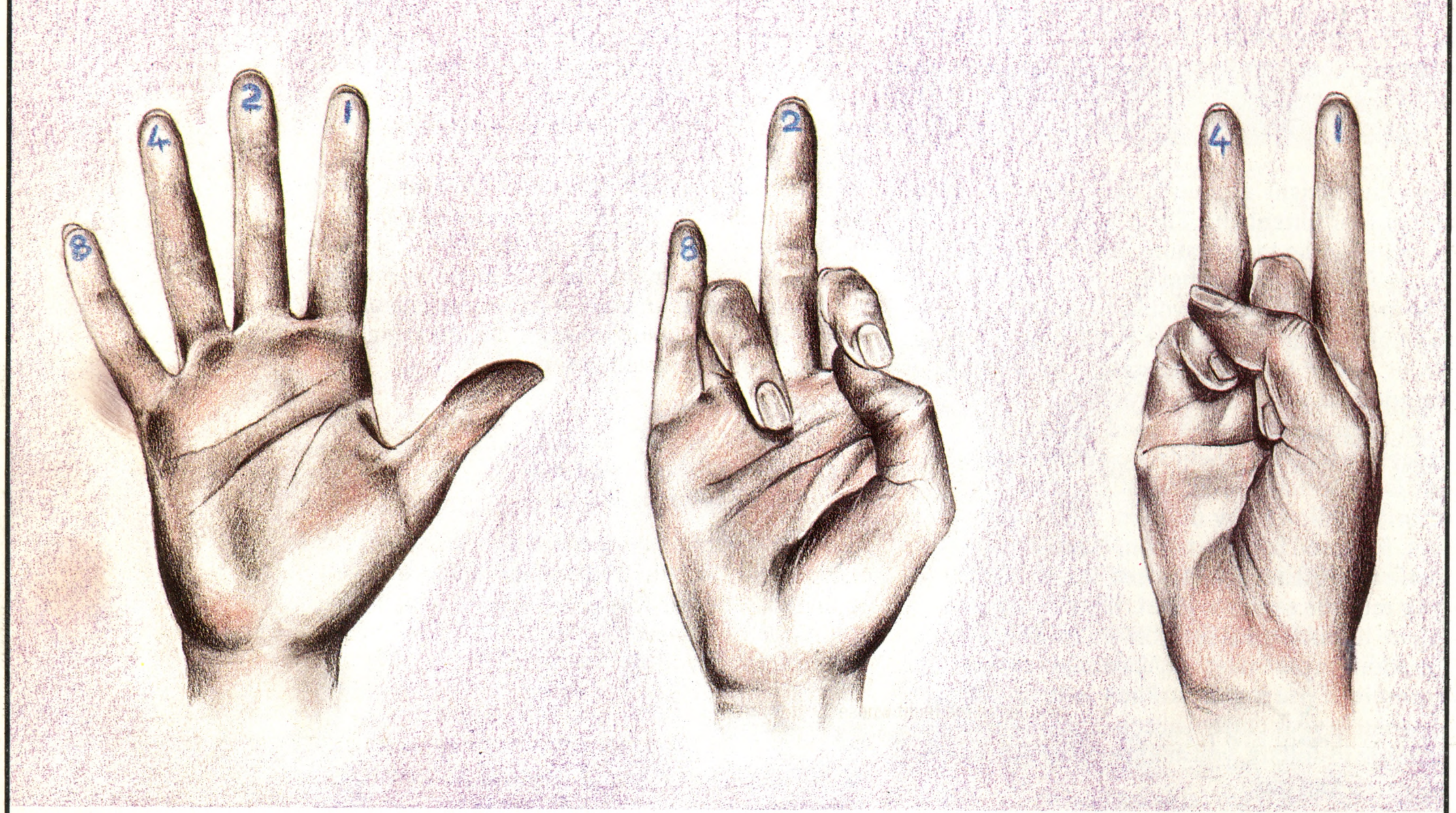
**COMMA**

The use of a comma between print fields will separate the items to be printed by causing the second one to begin at the next mid-line position. The spacing that this gives rise to will vary according to the lengths of the print fields and the number of characters in one line of the particular computer's display

# When 1 And 1 Is 10

## Computers achieve their prodigies of calculation with just two digits — 0 and 1

### Converting To Binary



The easiest way to convert small binary numbers to their decimal equivalent is to imagine writing the 'place value' of each binary column on the fingers of the right hand. As long as the binary number is not more than four digits long, all you have to do is to hold up the appropriate finger if the corresponding binary digit is a 1 and to hold the finger down if it is 0.

Hold up the appropriate fingers for 1010 and you get an 8 and a 2, which add together to give the decimal number 10. The third illustration shows how to decode 0101 — it gives a 4 and a 1, which comes to 5 in decimal form. Try using the method to compute the decimal equivalent of 1110 and 0110.

The method can be extended using both hands to figure out longer binary numbers. To do this, the fingers of the left hand (palm facing you) will need to be labelled 16, 32, 64 and 128, with the 16 on the little finger and the 128 on the index finger

ELAINE KEENAN

Most people take our system of using numbers so much for granted that it never occurs to them that any other system is possible.

The Romans thought up a system for representing numbers, using letters of the alphabet. X stood for 10, L stood for 50, C stood for 100, D stood for 500 and so on. The Roman system worked well enough as a way of recording simple numbers. It did not lend itself, however, to computations. Even additions in Roman numerals are difficult, for one reason: there is no concept of 'place value'. The position of a numeral in a Roman number tells us nothing about how much it is 'worth'.

Look at the two numbers 506 and 56. The only apparent difference is the zero in the middle. Its role is that in the number 506 it tells us that there are no 'tens', only five 'hundreds' and six 'ones'.

Every 'column' or position in a conventional number has a 'value' associated with it. The column on the right of the number is the 'ones' column, the next one (moving to the left) is the 'tens' column, the next one is the 'hundreds' column and so on. The digit used in any 'column' merely signifies how many of that column's value are involved.

You may be wondering what all this has to do with computers and the binary system. Computers are electronic machines, which can easily deal with numbers by using voltage levels. Five volts represent a one and zero volts represent a zero. As we learnt in Bits and Bytes (page 28), ones and zeros are perfectly adequate to represent any number, however big.

Using the familiar decimal system based on 10 (also known as the denary system) the number 506 is a concise way of representing the equivalent of five hundred and six knots in a string or five hundred and six notches on a stick. In binary arithmetic, the same number is represented as a clumsy 111111010.

Because the system used here is 'binary', the place value of each digit in each column is different. Instead of increasing in value in powers of 10, the columns go up in powers of 2.

The column on the right is still the 'ones' column, but because there are only two symbols (0 and 1), we run out of digits as soon as we add 1. In the decimal system, we only run out of symbols when we get to 9; the next column uses a digit which says: we have run out of symbols — we don't have anything for numbers bigger than 9 — so we'll use the 'tens' column and use a 1 to indicate that we now have one 'lot' of ten.

The binary system works in exactly the same way. Instead of grouping in tens and writing 10 for ten, binary groups in twos, so the binary digits 10 represent the decimal number 2.

Showing the number five hundred and six written in decimal and in binary illustrates the essential similarity clearly:

| Hundreds | Tens | Ones | |
|---|---|---|---|
| 5 | 0 | 6 | |
| = 5x100 + | 0x10 + | 6x1 | (= 506) |

| 256s | 128s | 64s | 32s | 16s | 8s | 4s | 2s | 1s |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |

= 1x256 + 1x128 + 1x64 + 1x32 + 1x16
+ 1x8 + 0x4 + 1x2 + 0x1 (= 506)

The rules of arithmetic in the binary system are exactly the same as the familiar rules of the decimal system — the only difference is that we run out of counting symbols after 1 instead of after 9. Let's try some additions to prove it. Decimal equivalents are printed in brackets.

| | |
|---|---|
| (3) | 11 |
| +(5) | +101 |
| (8) | 1000 |

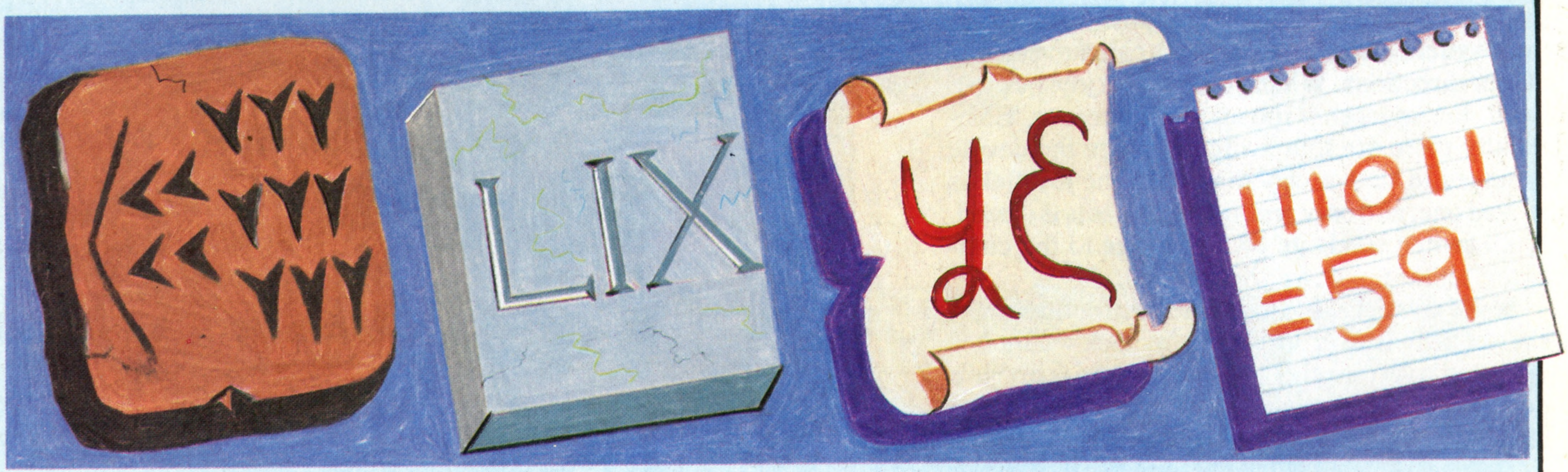
(1 +1 = 0 carry 1)
(1(carried) + 1 = 0 carry 1)
(1(carried) + 1 = 0 carry 1)
(1(carried) + 0 = 1)

In binary, as we have just seen, adding 1 to 1 means we have run out of symbols as only zeros and ones are allowed. So we say 'one and one equals nought carry one' (just as in decimal adding 1 to 9 means we have run out of symbols — there are no symbols larger than 9 — so again we say 'nine and one equals nought carry one'). Here's another addition, worked out for you, with two more to try for yourself.

| | | | | | |
|---|---|---|---|---|---|
| (4) | 100 | (7) | 111 | (3) | 11 |
| +(6) | +110 | +(2) | + 10 | +(12) | +1100 |
| (10) | 1010 | (?) | ? | (?) | ? |

By now you will have noticed that binary numbers are much longer than their decimal equivalent. See if you can add 11010110 to 1101101 — remember to keep the rightmost columns lined up just as you would when adding a longer decimal number to a shorter one!

## The History Of Numbers




**Babylonian** **Roman** **Hindu** **Binary**

The ancient Babylonians had an advanced number system, based on 60 rather than 10. Their representation of the number 59 in Babylonian 'cuneiform' script is shown above. The use of 60 as a number base had many advantages and there are still traces of their system in use today. There are 60 seconds in a minute, 60 minutes in an hour and six times 60 degrees in a circle — all vestiges of a mathematical system perfected 4000 years ago.

The Roman system was a considerable step backwards. Letters of the alphabet were used to represent numbers, but the position of each Roman numeral gave no indication of its value, making even simple arithmetic almost impossible.

The Hindus used nine signs for the numbers 1 to 9 and later added a sign to represent zero. Their vital contribution was 'place value' — the idea that a digit's position in a number determines how much that digit is 'worth'. Thus the 3 in 30 is 'worth' three tens. The Hindu system was adopted by the Arabs and gradually spread to Europe. One of the leading Arab mathematicians was called Al Khowarizmi. The Latinised pronunciation of his name gave us the mathematical term algorithm and his book 'Al-jabr wa'l Mugabalah' is remembered in the word algebra.

Computers use the binary system because numbers of any size can be represented using only ones and zeros

ANDY LESLIE

# Fast Reactors

## Some add-ons that will help the computer games enthusiast to speed up the action

IAN McKINNELL

In a computer game you might have to pilot a spaceship through enemy lines and fire your missiles to destroy a target. The joystick transfers the control of the spaceship from a finicky typewriter keyboard into your own hands. It is modelled on the original pilot's joystick found in aeroplanes.

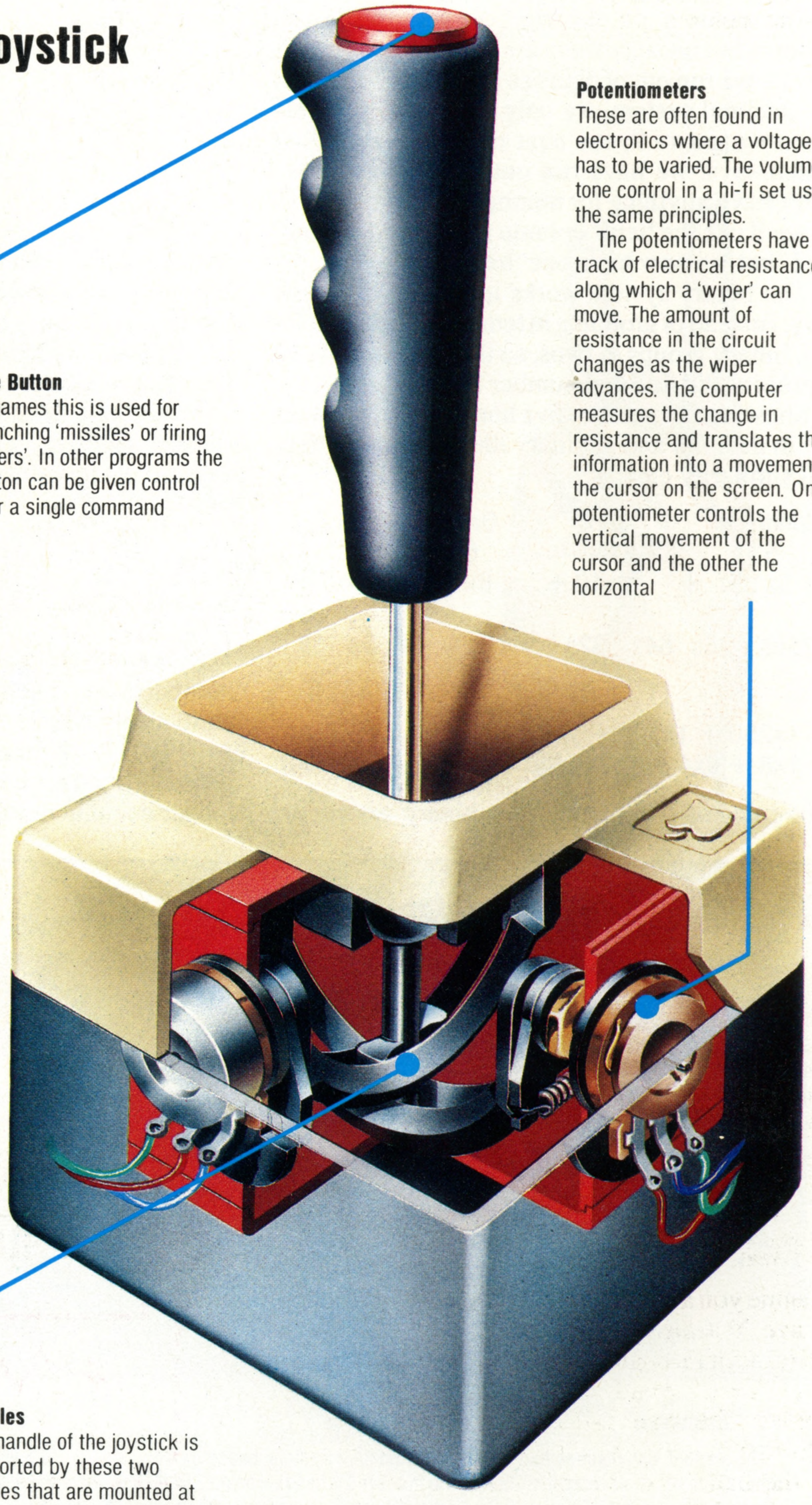
The joystick plugs into the back of the microcomputer and is most often used in arcade-style games. The spaceship, or whatever object the joystick controls, moves in the same direction as the joystick. Usually the joystick can move in any of four directions. When you ease the joystick forward, the spaceship moves up the screen. Electrically there are four switches inside the device arranged in such a way that when the joystick is moved one, and only one, of the contacts is closed. Each switch sends its own message to the computer: either up, down, right or left.

Some joysticks also have a button for firing missiles. The button is beside the joystick where it is operated with the other hand. Or, in the pistol grip design of joystick, missiles are released by squeezing the thumb-trigger.

The cheaper microcomputers, notably the Sinclair ZX81 and the Spectrum, don't always have joystick facilities. You either have to type in the desired directions of motion, using the allocated keys or else purchase a joystick interface.

The interface is an adaptor that allows a joystick to be connected to the computer. Some independent companies have produced interfaces for these machines, but even with such a device the game's programs have to be written to include joystick control as well as keyboard control.

### Joystick



DAVID WEEKS

**Fire Button**
In games this is used for launching 'missiles' or firing 'lasers'. In other programs the button can be given control over a single command

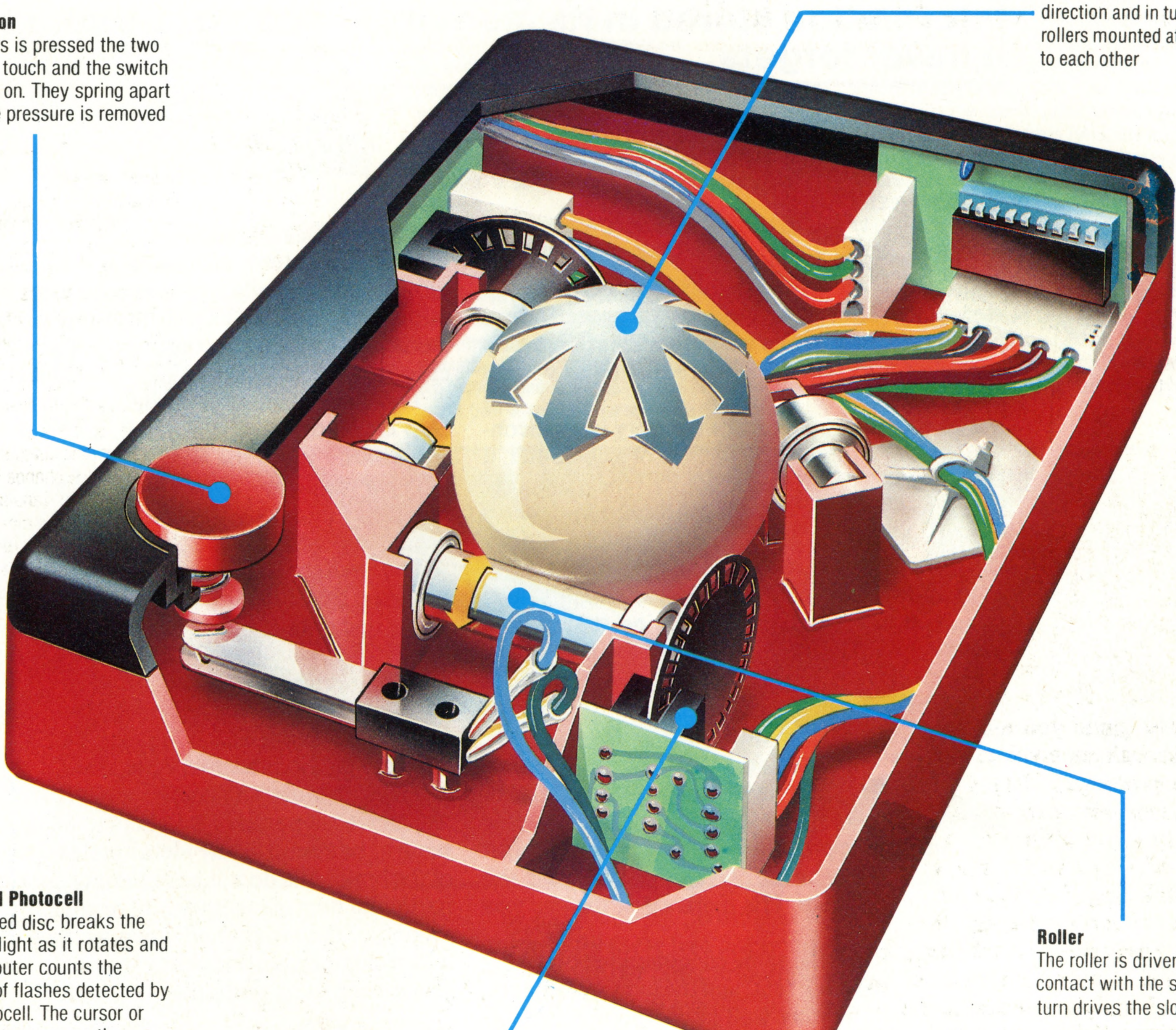
**Potentiometers**
These are often found in electronics where a voltage has to be varied. The volume or tone control in a hi-fi set uses the same principles.

The potentiometers have a track of electrical resistance along which a 'wiper' can move. The amount of resistance in the circuit changes as the wiper advances. The computer measures the change in resistance and translates this information into a movement of the cursor on the screen. One potentiometer controls the vertical movement of the cursor and the other the horizontal

**Cradles**
The handle of the joystick is supported by these two cradles that are mounted at right angles. They are linked to the potentiometers. When the joystick handle is moved, the 'wipers' on the potentiometers slide along and change the electrical resistance

**Roller Ball**
The ball can be rotated in any direction and in turn drives two rollers mounted at right angles to each other

**Fire Button**
When this is pressed the two contacts touch and the switch is turned on. They spring apart when the pressure is removed



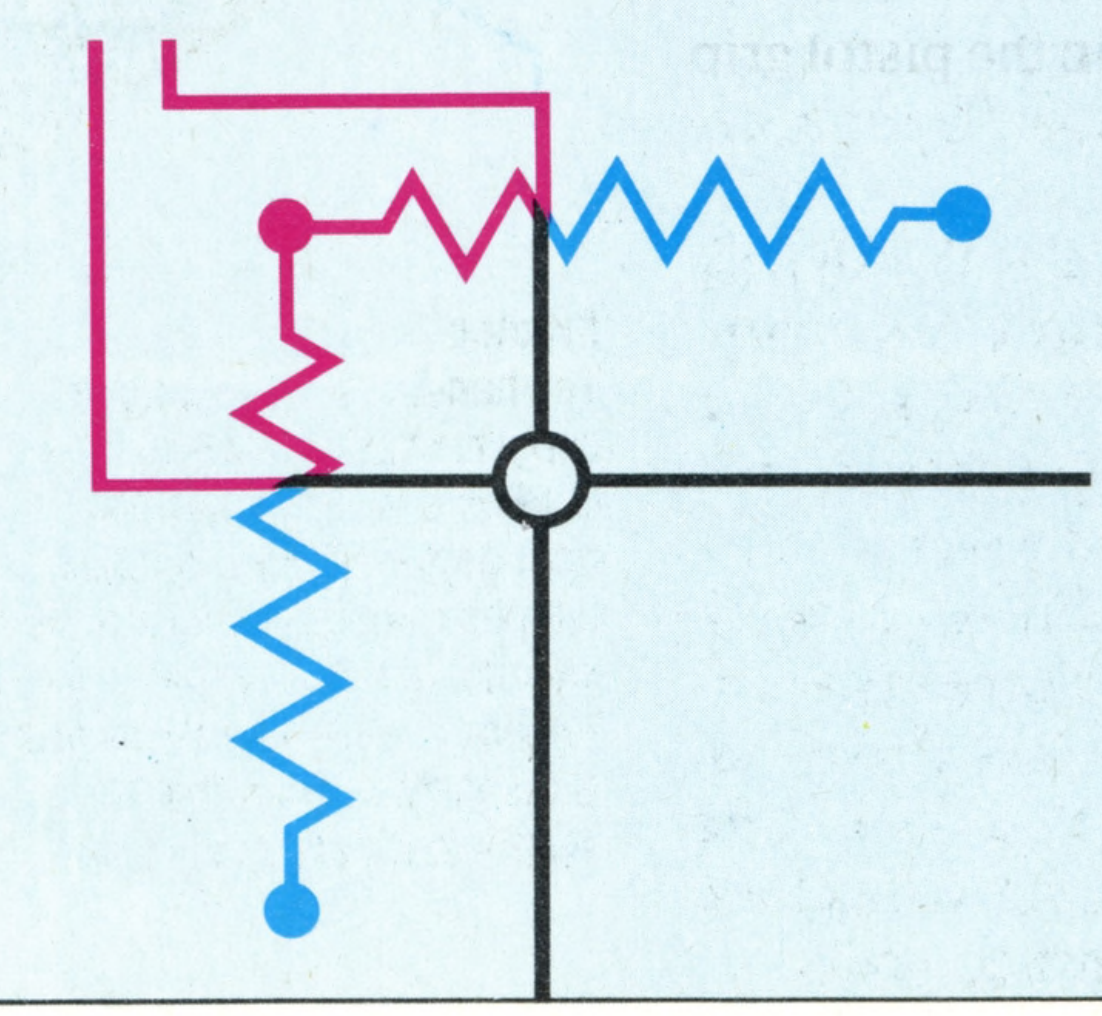
**Light And Photocell**
The slotted disc breaks the beam of light as it rotates and the computer counts the number of flashes detected by the photocell. The cursor or object moves across the screen in proportion to this number. To control both the vertical and horizontal movement on the screen there are two sets of disc and photocell

**Roller**
The roller is driven through contact with the sphere and in turn drives the slotted disc

## Track Ball

Imagine you are guiding the screen cursor through a maze. You have to be able to advance the cursor and direct it through the passages as they twist and turn. The trackball is designed for this type of problem. The trackball uses a sphere the size of a billiard ball that you roll in the palm of the hand. As the ball rolls, the object moves in the same direction giving you complete and immediate control. Inside the device are two wheels set at right angles, which rub against the ball. As you roll the ball in the palm of the hand one wheel picks up the vertical part of the motion and the other the horizontal. The computer unites the two signals to recreate the movement.

## Potentiometer Circuit



The lever of the joystick is connected to two variable resistors (called potentiometers) The mechanical linkage moves the contact point (wiper) along either or both of the resistance tracks (represented as zig-zag lines). The position of the joystick therefore sets the electrical resistance of the two potentiometers. The computer checks the voltages and calculates the joystick's position. The computer then converts this information into positional changes on the screen

# Total Recall

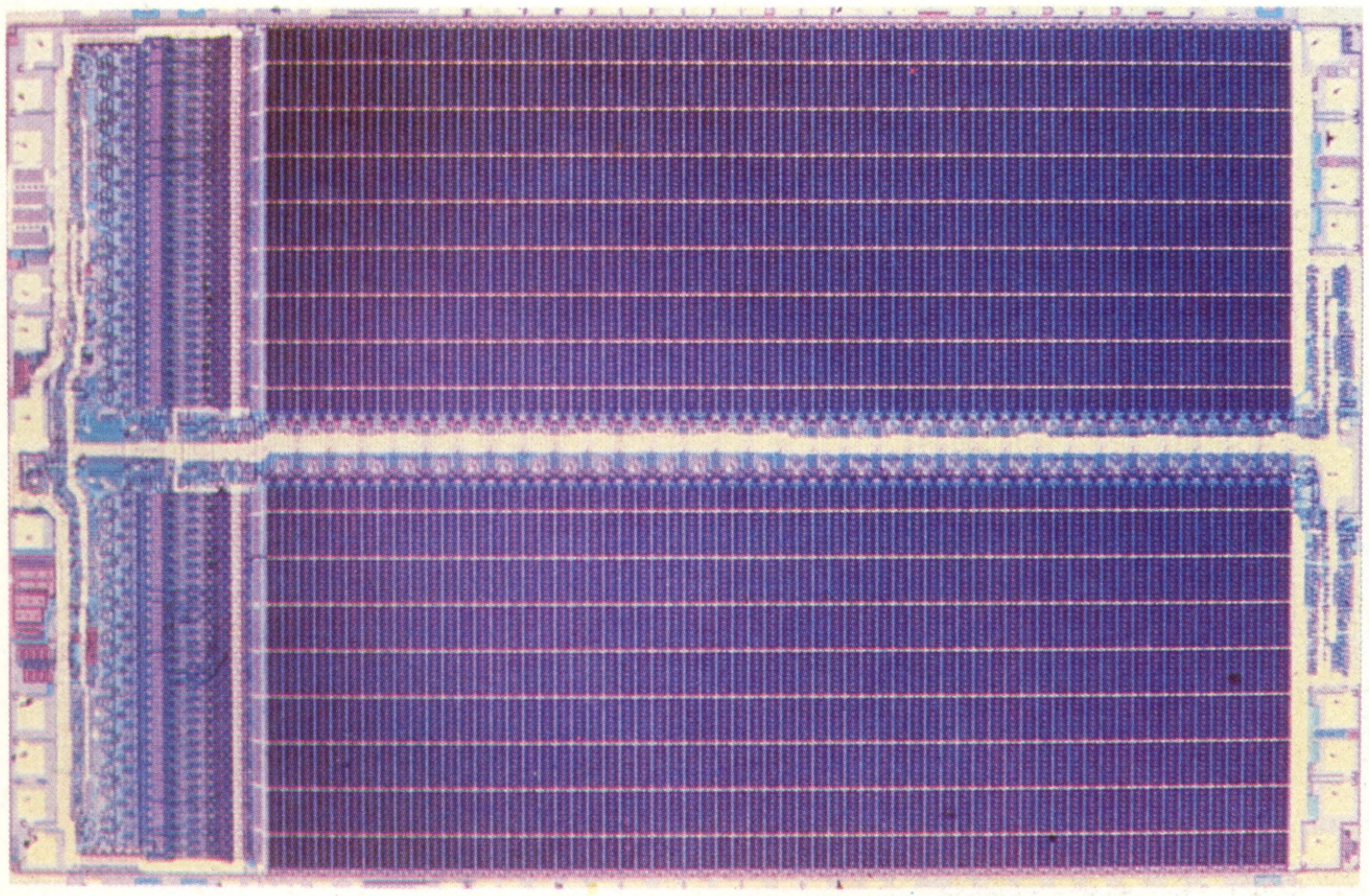
## How the computer keeps careful track of all the information stored in its memory — and makes sure that it never forgets

**RAM Memory**

The RAM chip (below) is one of the major recent advances in computer technology. RAM (Random Access Memory) is one of the varieties of completely electronic memory, a category that also includes ROM (Read Only Memory). Cassette tapes and magnetic floppy disks are examples of the other main kind, electro-magnetic memory.

RAM memory is fabricated from silicon using a photographic process and chemical etching to create thousands of tiny transistors. Each 'bit' of memory requires at least one transistor in a storage cell circuit.

The time taken to 'write' a single bit into any one of the 16,384 storage cells is about 200 nanoseconds — a five-millionth of a second



COURTESY OF MOSTEK LTD

In human terms, memory is the storehouse of the mind, the place where details of experience are stored for later use. And in computer terms 'memory' means pretty much the same thing, only a computer's memory is more limited in what it can do.

For a human being, poor memory is an inconvenience or an embarrassment. For a computer, it is disastrous. Without memory, the computer would have nothing to work on and nothing to tell it what to do, since it also uses its memory to store the programs that drive it.

In both cases, the word 'memory' implies two things; storage and recall. Storing information without being able to get it out again is not very useful, and trying to recall information that hasn't been stored is obviously futile.

The two kinds of memory are similar in another way too. Human memory appears to be of two general types, short-term and long-term. A man crossing the road, for example, will remember to wait until the approaching car has passed by. But when he is safely on the other side of the road, the vehicle is forgotten. His memory of the car was short-term.

However, had the same car contained two masked men in the back seat, and the man's wife in the driver's seat, he may well have remembered the whole incident, including the type and colour of the car, and possibly its registration number! This was a long-term memory.

Stretching a point a bit, computers have short-term and long-term memories as well. The long-term or 'non-volatile' type contains programs and information that the user wants to keep; these are stored as magnetic recordings on the surface of cassette tape, floppy disks, or ROM packs.

The short-term or 'volatile' type is the RAM chip inside the computer itself, and is only used temporarily while the computer is working. The moment the power supply is removed, even for a fraction of a second, all the contents in the memory disappear instantly.

The analogy with human memory is not exact, however. For the computer to work, the right programs and data need to be transferred from long-term storage into short-term storage so that the computer can have instant access to them. And the *way* data are stored and recalled to and from a computer memory is completely different too from the methods human beings use.

The way human memory works is still a mystery, since memories of a particular incident do not seem to be stored in any identifiable tiny segment of the brain. We do not have to figure out where a particular item of memory is to recall it to the foreground of the mind. And when we have finished with a memory, we do not have to worry about putting it back in its particular slot in the brain.

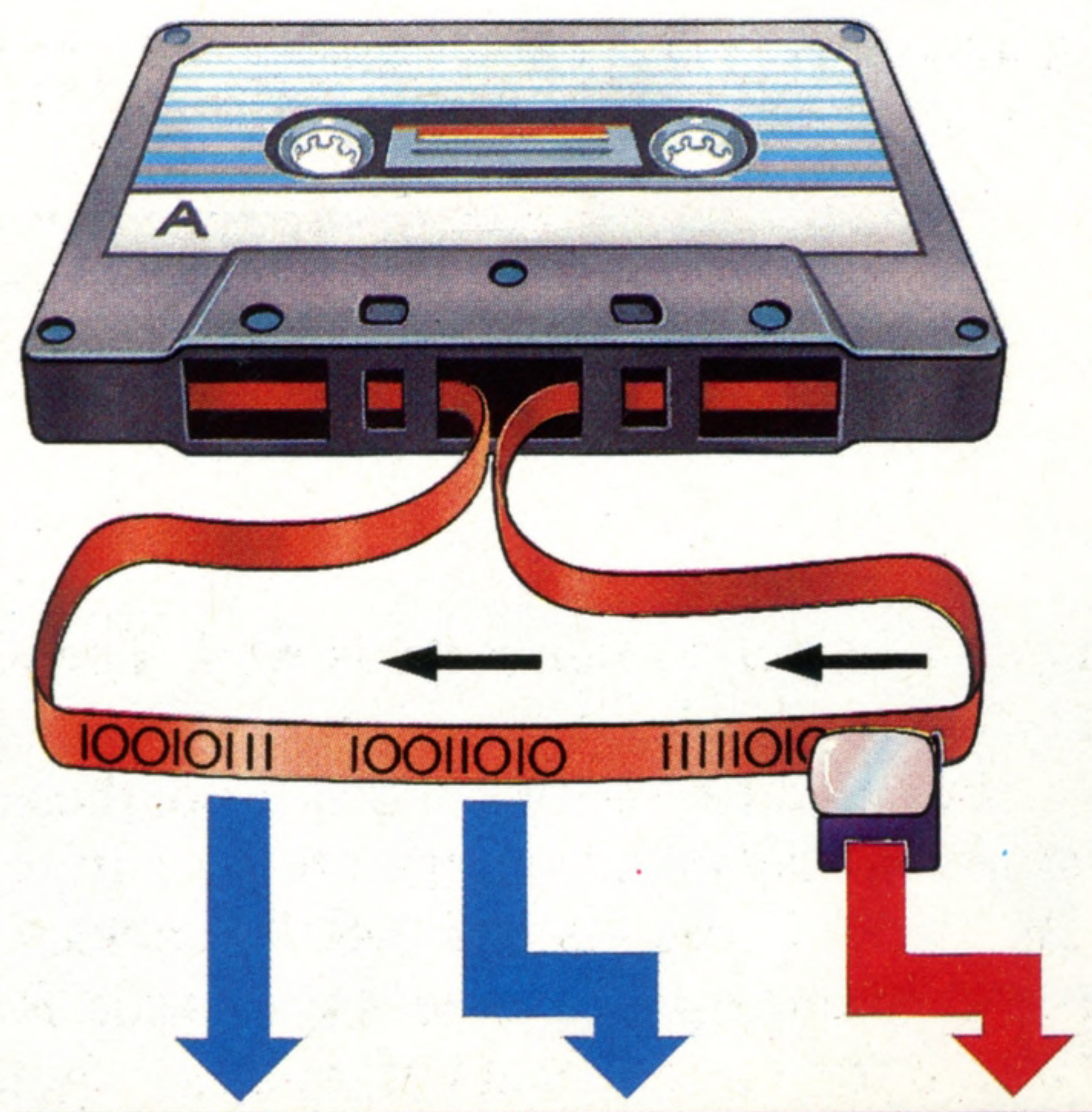
## Organised Chaos

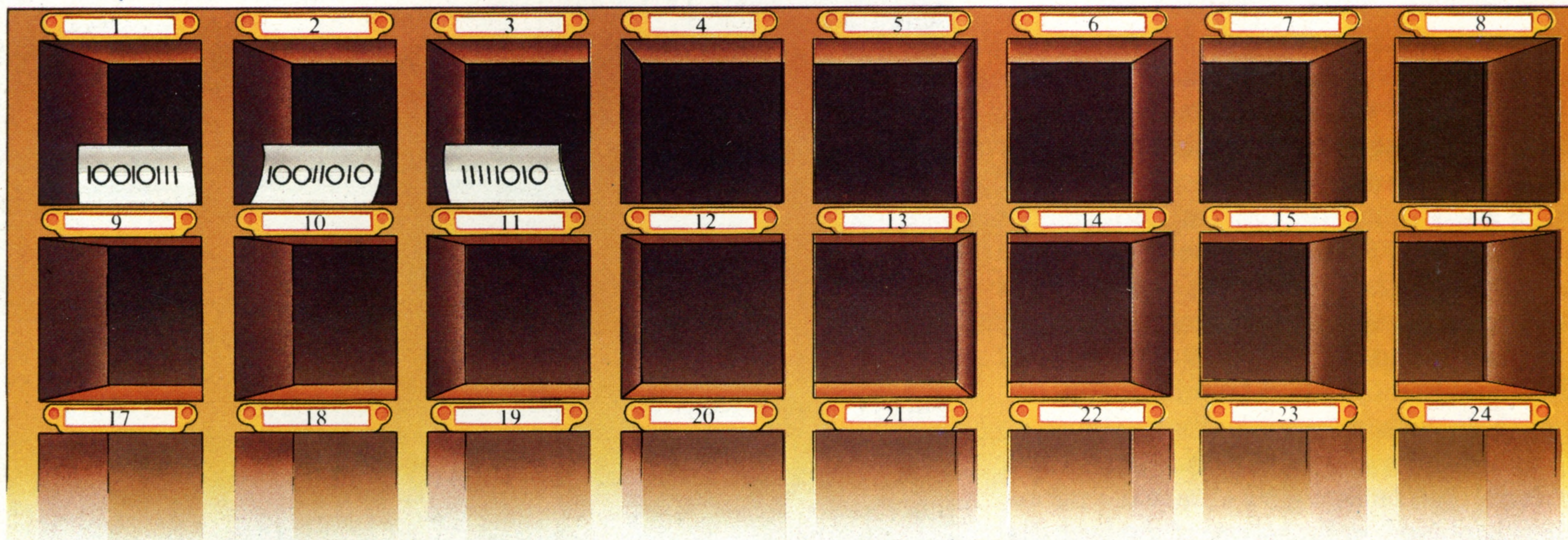
In computer memory, it is the *location* of each item that is vital. The computer has to be able to find a particular byte of information, whether it is part of a program or part of the program's data. The computer also needs to keep a 'note' of where it puts the information.

Human memory seems more like a box crammed full of information, but not organised. The pieces of information are just stuffed in, apparently at random, tangling up with each other and being shoved around inside the brain as more and more images and experience are crowded in. Somehow or other, the brain can make sense of this and pull out what it needs, when it needs it.

Computer memory is more like a giant rack of pigeonholes, each hole completely separate from the rest. Everything is very orderly; each pigeonhole has a number (called its 'address') and contains just one byte, no more and no less. The computer finds information by pigeonhole number, not by what is stored in that pigeonhole.

## Memory




TONY LODGE

Programs recorded on tape are stored 'sequentially', with every bit from each byte recorded one after the other. When the tape is replayed, the computer 'reads' each bit but stores them in groups of eight (bytes) in each memory cell. The first byte on the tape is placed in the first available memory cell, the second byte in the next one and so on. When the computer needs to 'run' the program, all it needs to know is the 'starting address'. The computer transfers the contents of each memory cell into the CPU in sequence and these bytes cause it to 'execute' or perform the actions required by the program.

Part of the computer's memory is occupied by 'housekeeping' programs responsible for fundamental aspects of its operation — checking which keys have been pressed, displaying characters on the screen and so on. Such 'built-in' software may also include the BASIC programming language. These internal programs take up space in the memory and leave less for the storage of the user's own or commercial software. Some BASIC versions, for example, are stored in 16 Kbytes of memory. If the computer is supplied with 64 Kbytes of memory, only 48 Kbytes will be left for other programs. When a program is loaded from cassette tape, the first available (empty) memory location will clearly not be the first location in RAM. It is one of the duties of the housekeeping software to know and remember where the first memory location available to the user is. After the program has been loaded into the computer's RAM, the housekeeping software says, in effect, 'start by looking at memory location x and then continue by examining each successive memory location, entering the contents of that location into the CPU and doing what it says'. The original order in which the program was entered on the keyboard by the programmer is the same as the order recorded on the tape. When the program is transferred from the tape to the computer's memory, it is put into the memory cells in the same order. To the computer, the effect is the same as if the program had just been typed in on the keyboard

Computers have no intelligence, and cannot organise their memories for themselves. The only reason a computer can store anything is because someone has put it into the right pigeonhole in the right order, and at the right time. How does that happen in a typical home computer system?

When you turn your home computer on, a message usually appears on the screen to tell you it is working. In most cases, it also informs you that you can start writing a program. This message, and the facilities that let you start programming, are stored in part of the computer's internal memory; they need to be stored in the long-term memory (usually in a Read Only Memory or ROM chip — see page 9).

This portion of the computer's memory contains programs that check if keys have been pressed, 'print' letters on the screen, and perform other essential 'housekeeping' jobs. It also contains a special program that translates commands usually written in BASIC into the much simpler binary language of ones and zeros understood by the computer.

When the home computer is switched on, the message on the screen often says 'x bytes free', where 'x' is something like 15,797 or another such strange number. What this tells you is the number of pigeonholes in the computer's memory that are free for you to use. Hitting keys on the keyboard starts to fill these free pigeonholes up — and here we come to the other important thing about computer memory, the *order* in which information is stored.

Pressing a key on the keyboard sends one byte (representing the letter pressed — see page 3) to the memory for storage. Hitting the 'k' key, for example, puts the letter 'k' into a pigeonhole in memory in binary form.

But which pigeonhole does that 'k' go into? It goes into the first free slot in the computer's short-term memory. If you think of a block of empty pigeonholes hung on a wall, the 'k' would go in one in the top left-hand corner.

Hit another key, say 'e', and the appropriate pattern of bits goes into the second empty hole, to the right of the 'k'. Hit a third, a 'y', and it goes into the third hole next to the 'e'. Looking at our block of pigeonholes, the codes for the word 'key' appear on the top row.

The computer has an internal 'counter' to assess which pigeonhole it has reached; it knows where to start because the built-in 'housekeeping' program tells it where the free area of memory starts. As each letter is stored, the counter is increased by one to nominate the next pigeonhole for the next letter typed.

# Cash On Command

## How you get reliable 24-hour service from the computerised cash machine at your bank

Bank customers now have access to cash whenever, and almost wherever, they want it thanks to the microcomputer. All that is needed is a small plastic 'cash card' containing coded information about the customer's account. When funds run low and instant cash is required, all the customer needs to do is go to the nearest branch of the bank that has a cash dispensing machine.

These cash dispensers are really microcomputers in disguise. To obtain money, the customer inserts his card in the slot on the machine. The card contains a black magnetic stripe similar to the magnetic material used on cassette tape. When the card is inserted, a magnetic reader checks the numbers encoded on the stripe. These numbers show that the card is of the right kind (not an imitation) and also tell the microcomputer inside the cash dispenser the customer's secret PIN (Personal Identification Number).

The cash dispenser then asks the customer to identify himself by typing in his PIN. If the number entered is correct, he is given a choice of options, which are displayed on the screen of the cash dispenser. The options usually include a choice of withdrawing cash or making an enquiry about the balance in the account.

If the customer wants cash, he presses the appropriate button and then types in the amount of money needed. At this stage, the details of what goes on inside the system differ. A code on the magnetic stripe tells the cash dispenser if it needs to contact the bank's central computer or if the request for money can be dealt with locally.

If the request can be handled locally, the microcomputer in the cash dispenser checks further information encoded on the magnetic stripe. This tells it what the cash limit for withdrawals is and how much money has been drawn in the last week. If the limit is £100 per week and £60 has already been withdrawn, it will allow up to £40 to be dispensed. A prompt on the screen asks the customer how much he wishes to withdraw, usually with a message like PLEASE ENTER AMOUNT OF MONEY REQUIRED. THE TOTAL MUST BE A MULTIPLE OF £5.

If the customer is making a statement enquiry, or if there is no cash limit on withdrawals, the cash dispenser goes 'on line' to the bank's main computer. This means that a connection is made over special telephone lines so that the dispenser's microcomputer can communicate with the main computer to check the amount of money left in the account.


Assuming enough money is in the account, the main computer will instruct the microcomputer in the dispenser that the money may be dispensed. A note-counting machine in the dispenser counts out the bank-notes and pushes them out through a separate slot. If the dispenser is working 'on line', the central computer is informed of the amount withdrawn and the customer's account is updated accordingly.

The cash dispenser was introduced to provide banking facilities out-of-hours. Day or night the customer can obtain cash, request a new cheque book, check the balance in the account or order a detailed statement. The customer is given a card — like a cheque or credit card — and is allocated a personal number

**Yellow Ticket Trouble**

Magnetic cards can even be found on the London Underground. A few years ago a comprehensive system was devised to allow passengers in and out of automatic gates by use of a yellow magnetic ticket. Unfortunately the exit machines rejected tickets so often that irate passengers forced the system of exit gates to be abandoned


JULIE-ANN CHAMBERS

# Micro Update

After the great Christmas home computer bonanza, prices of micros in your High Street shops continue to follow the downward trend signalled by the specialist retailers. So to keep you up to date on how to get what you want at the best possible price, here are the latest changes in price and specification together with details of some more machines that have only very recently come on to the market.

There has been a price change on the **Atari 800,** which has gone down by £20, from £300 to £280. This is probably due to the arrival of its successor, the 800XL (see panel).

Both **Commodore** machines have been the subject of vigorous price cutting: the **Commodore 64** is down £99 to a more realistic £200, and the **VIC 20** is down to just £100. This, in fact, is an even better deal than it seems, as many retail outlets are selling VIC 20 starter kits, which include the cassette unit and other bits and pieces.

**Dragon,** instead of cutting the cost of their machine, have introduced a 64K version, which is currently selling at £225. The old 32K machine is holding its price, but can be found in some shops for as little as £165.

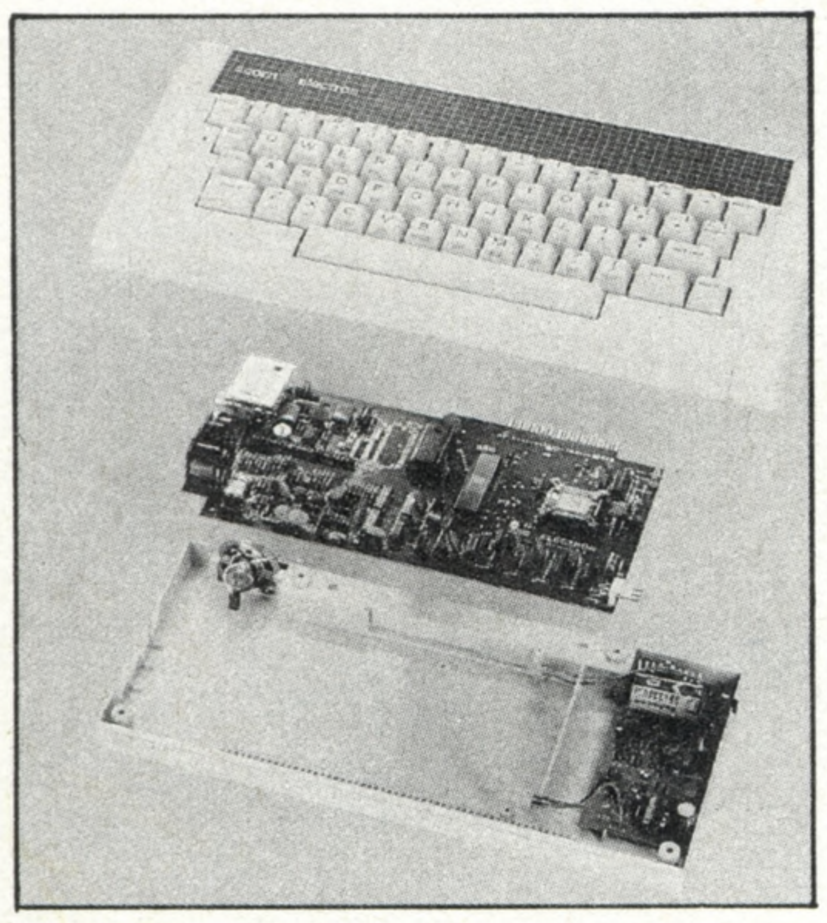
One machine to disappear completely from the scene is the **Jupiter Ace.** This is perhaps not surprising as it uses FORTH, which is of almost no interest to the beginner, instead of BASIC — and also because of its shoddy manufacture.

**Oric** have dropped the price of their 16K machine from £130 to £100, while the 48K version has fallen by £29 to a new low price of £140.

The cheap and very popular **Sinclair ZX81** is likely to remain with us for at least another year, and has also been subject to a small price cut. It now sells for just £40, which ensures its place as the cheapest micro on the market.

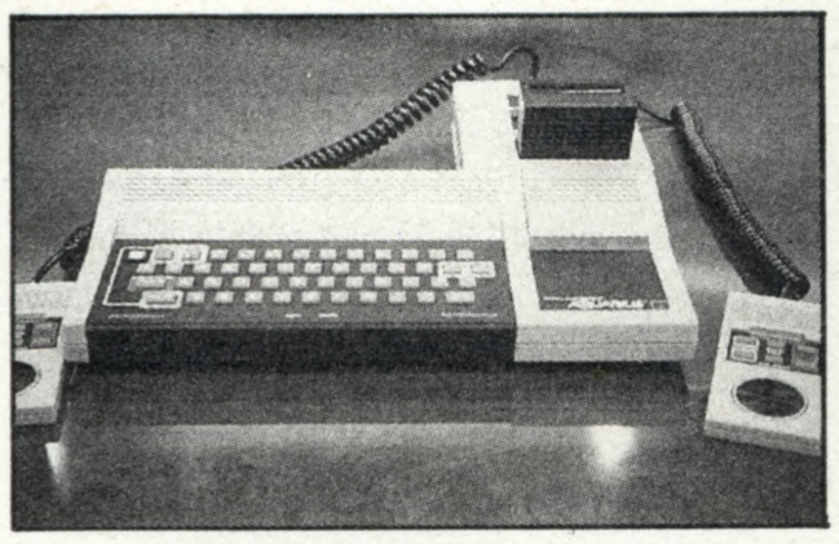
**Sord** have decided to maintain the price of their M5, but have increased the memory size to 20K.

Finally, **Texas Instruments** have dropped the price of their **TI99/4A** by a further £50, and it is now selling at £100. But beware, this machine will disappear from the shops any time now.



## ACORN ELECTRON

**TYPICAL PRICE:** £199.
**STANDARD MEMORY:** 32K.
**EXPANDABLE TO:** Non-expandable.
**CONNECTS TO:** Colour TV monitor and cassette deck.
**ADVANTAGES:** Cheaper than the BBC Micro, though it has a comparable specification, and it still has a full 56-key keyboard.
**DISADVANTAGES:** Much slower than its big brother.
**SUMMARY:** The Electron's value is enhanced by the fact that there is a certain degree of software compatibility with the BBC Micro, although the difference in speed may detract from this.

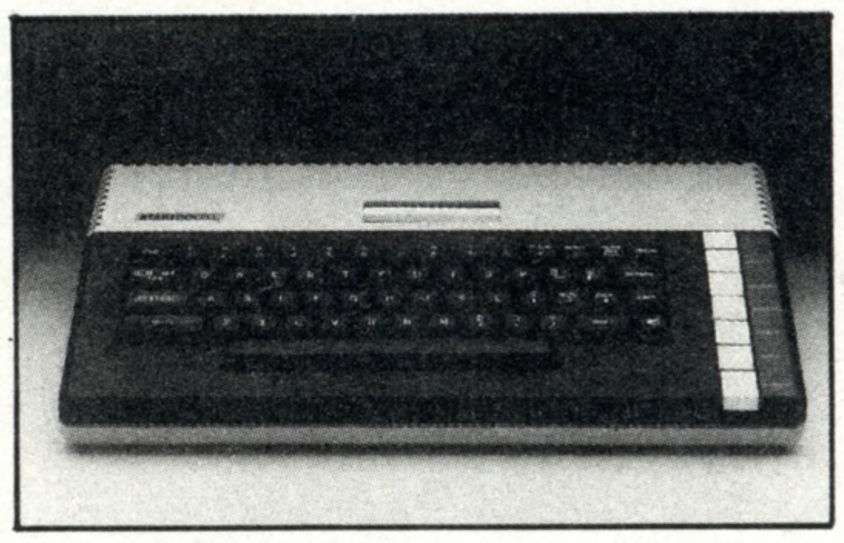


## AQUARIUS

**TYPICAL PRICE:** £50.
**STANDARD MEMORY:** 4K.
**EXPANDABLE TO:** 8K (+£20), 20K (+£30).
**CONNECTS TO:** Colour TV, RS232 expansion unit with two joypads.
**ADVANTAGES:** Cheap machine, but the add-ons are expensive.
**DISADVANTAGES:** Bad keyboard, small memory and little software available.
**SUMMARY:** UK's cheapest colour computer.

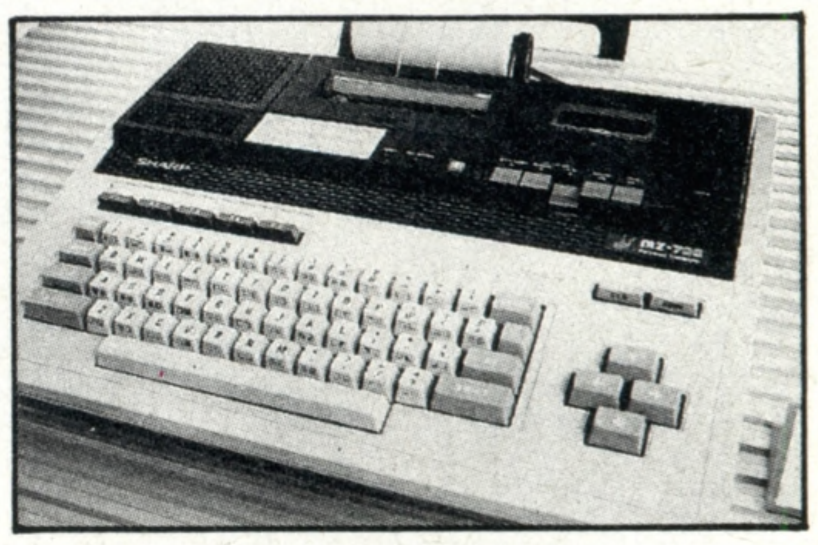
## ATARI 600XL

**TYPICAL PRICE:** £160.
**STANDARD MEMORY:** 16K.
**EXPANDABLE TO:** 64K+£90.
**CONNECTS TO:** Colour TV monitor, up to 4x88K floppy disks (£299 each), two joysticks, printer and cassette deck.
**ADVANTAGES:** Optional Centronics and RS232 module for £135.
**DISADVANTAGES:** Software, as usual, is expensive for the Atari.
**SUMMARY:** The 600XL replaces the Atari 400, and is compatible with both hardware add-ons and software from its predecessor.



## ATARI 800XL

**TYPICAL PRICE:** £250.
**STANDARD MEMORY:** 64K.
**EXPANDABLE TO:** Non-expandable.
**CONNECTS TO:** Colour TV monitor, up to 4x127K floppy disks (£299 each), printer and cassette deck.
**ADVANTAGES:** Very good keyboard. Greater expandability than its predecessor.
**DISADVANTAGES:** Software, as usual, is expensive for the Atari.
**SUMMARY:** Currently top end of the Atari range.



## SHARP MZ-700

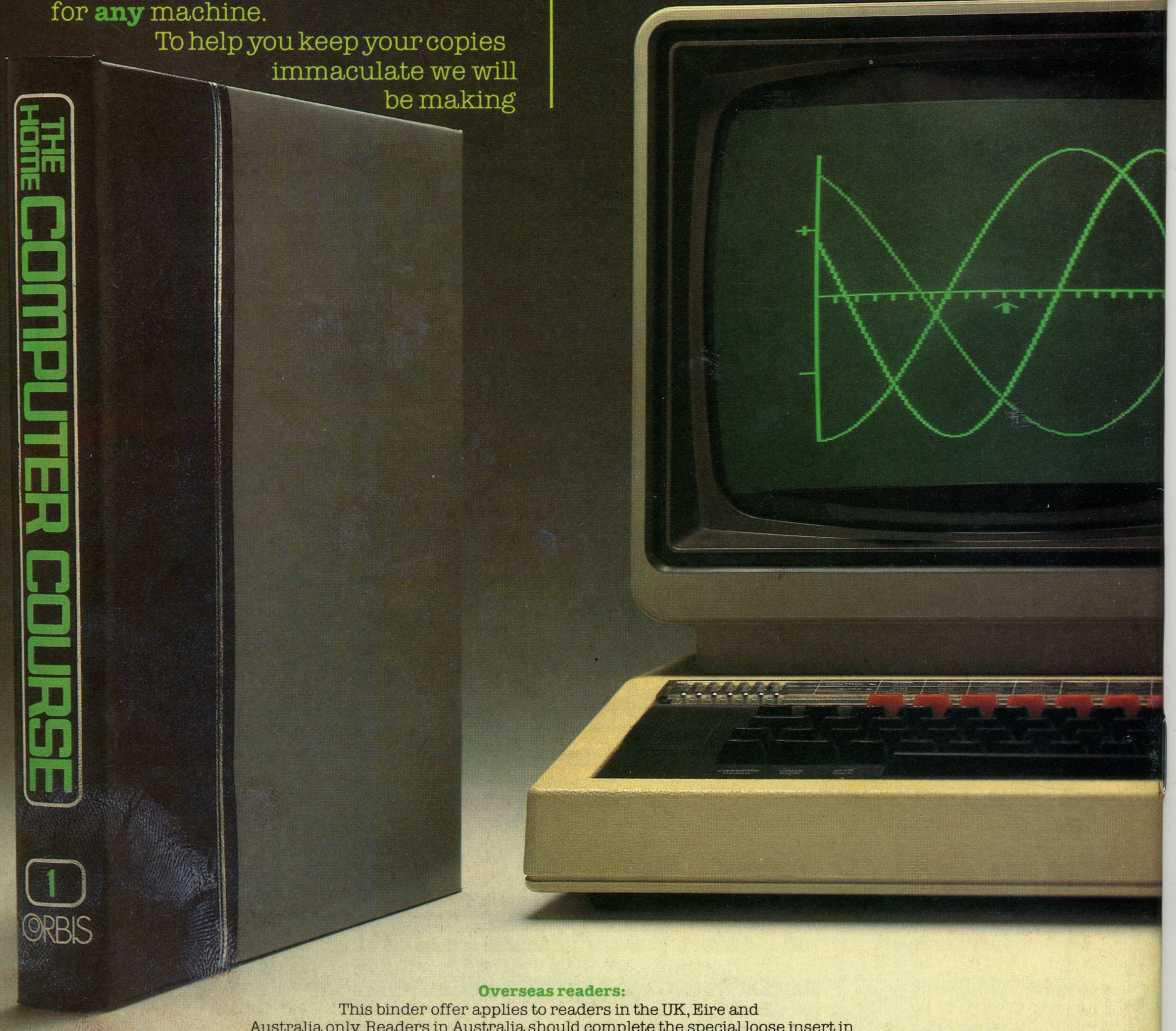
**TYPICAL PRICE:** £250.
**STANDARD MEMORY:** 64K.
**EXPANDABLE TO:** Non-expandable.
**CONNECTS TO:** Colour TV, two joysticks and printer.
**ADVANTAGES:** Optional built-in cassette recorder and printer.
**DISADVANTAGES:** The BASIC is not held in ROM, and must therefore be loaded from tape.
**SUMMARY:** Serious hobbyist machine.

# THE HOME COMPUTER COURSE BINDER.

# A BASIC NECESSITY.

**The Home Computer Course** grows week-by-week into the first ever step-by-step guide to computing for **any** machine.

To help you keep your copies immaculate we will be making a very special binder offer with an order form in a future issue. Be sure not to miss it.




**Overseas readers:**
This binder offer applies to readers in the UK, Eire and Australia only. Readers in Australia should complete the special loose insert in Issue 1 and see additional binder information on the inside front cover. Readers in New Zealand and South Africa and some other countries can obtain their binders **now.** For details please see inside the front cover.

Binders may be subject to import duty and/or local tax.