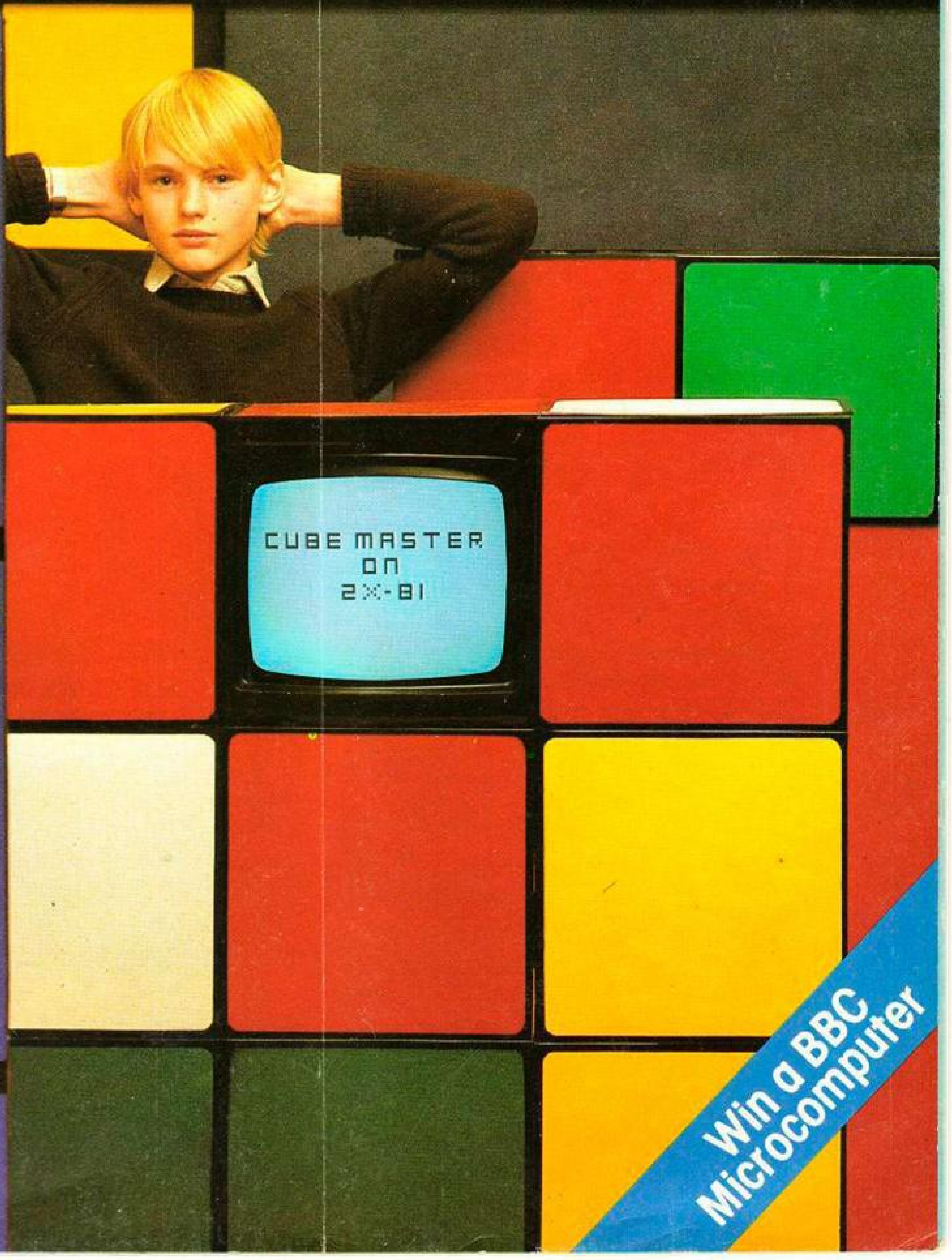
60p

YOUR COMPUTER

FEBRUARY 1982

Vol.2 No.2



Solve the cube on the ZX-81

BBC Basic

ZX cassettes tested

Pet dominoes

Animated graphics

Win a BBC Microcomputer

ingenious!

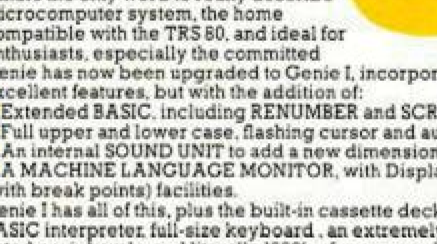
...that's the only word to really describe the superb Genie microcomputer system, the home computer which is compatible with the TRS 80, and ideal for all micro-enthusiasts, especially the committed hobbyist.

Genie has now been upgraded to Genie I, incorporating all of the original, excellent features, but with the addition of:

- Extended BASIC, including RENUMBER and SCREEN PRINT.
- Full upper and lower case, flashing cursor and auto-repeat on all keys.
- An internal SOUND UNIT to add a new dimension to your own programs.
- A MACHINE LANGUAGE MONITOR, with Display, modify, enter and execute (with break points) facilities.

Genie I has all of this, plus the built-in cassette deck, 16K RAM, 12k ROM with BASIC interpreter, full-size keyboard, an extremely wide range of new and up-dated peripherals, and literally 1000's of pre-recorded programmes available. Yet, almost unbelievably, the price of Genie I is even lower than that of the original Genie.

Ingenious for business

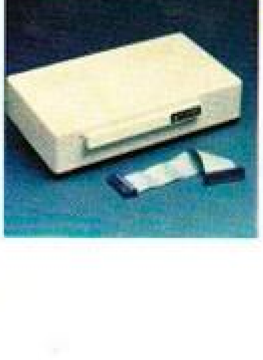


The Genie II is a major breakthrough for small business computers. Harnessing all the advantages of Genie I, including low price, Genie II adapts perfectly to commercial functions with the following features:

- Numeric keyboard
- Four usable, definable function keys.
- Extension to BASIC
- Basic business commands
- Fully expandable with the same peripherals

New!...12" Monitor

Now, a choice of 2 monitors giving a clear easy to read image. The updated EG101 has a new green phospher tube.



New!...Expander

An updated Expansion Box (EG 3014) is a major feature of the new Genie I system, and unleashes all its possibilities, allowing for up to 4 disk drives with optional double density. It connects to a printer, or RS232 interface or S100 cards. There is 16k RAM fitted and it has a new low price!

New!...Printer

The EG 602 printer can be connected to the Genie either through the expander or directly into the computer using the Parallel Printer Interface. It is a compact unit, with an 80 column, 5 x 7 matrix print-out, operating quietly and efficiently at 30 characters per second.

Disk Drive

As well as the obvious advantage of mass storage, the addition of the disk system to the Genie means much faster access to other languages and full random access file handling. Up to 4 of these 40 track drives can be used on a system.

SPECIAL TECHNICAL GENIE HOT – LINE ON 0629 4995

for all your technical advice and service back-up on any aspect of the Genie system direct from the experts.

For full details and demonstration of Genie I, Genie II or advice on any aspect of the system, either call in to your local dealer, or write directly to the sole importers at the address below.

LOWE electronics

Chesterfield Road, Matlock, Derbyshire DE4 5LE.
Telephone: 0629 4995. Telex: 377482 Lowlec G.

YOUR COMPUTER

Cover photograph of Simon Lane by Stephen Oliver. Set design by Ellen Butler.

Editor
DUNCAN SCOT
Assistant Editor
BRENDON GORE
Staff Writer
BILL BENNETT
Production Editor
TOBY WOLPE
Production Assistant
JOHN LIEBMANN
Editorial Secretary
LYNN COWLING
Editorial: 01-661 3144
Advertisement Manager
DAVID LAKE 01-661 3021
Assistant Advertisement Manager
PHILIP KIRBY 01-661 3127
Advertisement Executives
KEN WALFORD 01-661 3139
BILL ARDLEY 01-661 3127
Midlands Office
DAVID HARVETT 021-356 4838
Northern Office
RON SOUTHALL 061-872 8861
Advertisement Secretary
MANDY MORLEY
Publishing Director
CHRIS HIPWELL

Your Computer, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS.
Typesetting by In-Step Ltd, London EC1.
Printed by Riverside Press Ltd, Whitstable, Kent.
Subscriptions: U.K. £8 for 12 issues.

Published by IPC Electrical-Electronic Press Ltd, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS. Tel: 01-661 3500 Telex/grams: 892084 BIPRESG.

EDITORIAL

TEACHERS must be very disappointed by the BBC's Computer Programme series, which began transmissions to schools on January 11. First, most of the schools which had been promised priority delivery of a BBC Microcomputer in time for the start of the series are still waiting for their computers to be made — largely because of reliability problems with some of the chips. Secondly, those schools which were supplied in time for the start of the series would have found that they do not need it. The BBC Microcomputer might well be the best value for money to have hit the market for some time but it simply does not seem very relevant to the Computer Programme, which treats computing in general and abstract terms. The BBC's argument is that it is unreasonable to expect all the viewers to follow the series through from start to finish and that there is, therefore, no point in trying to teach the viewers much about programming other than to Load and to Run programs from cassette.

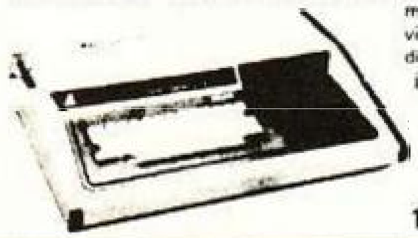
The BBC series is still interesting and will no doubt help the complete novice to understand microcomputers and the kind of programs that one can expect them to run. Those teachers who were planning to let their pupils view the entire series during the school day might well, however, have some doubts about spending so much of their valuable time on such a general overview of the subject. We are sure that most would have welcomed a more detailed and informative look and an attempt to teach some of the specifics of the subject. Why is it that television is terrified of going into detail?

The problems that the BBC has had in making its series raise a more general point about the problems that the media have always had in dealing with science, especially in programmes which are supposed to be educational. With the exception of BBC Radio's science programmes, the emphasis has always been on being entertaining to the exclusion of detail. We suspect that one of the reasons that the BBC abandoned its attempts to teach us how to program the BBC Microcomputer is that no way could be found of fitting hard detail into the received view of how a sleek and glossy science program should look and be presented. Television science programs are always produced and presented by generalists who do not have a science background. These generalists then turn to scientists to advise them on the content; accepting and rejecting that advice is the prerogative of the programme makers. We would prefer an approach in which the programmes were produced and presented by scientists who could accept and reject the programme-making advice of the generalists.

We would like to hear from those of you who already have a BBC Microcomputer — how have you been using it with the series? We would also like to look at programs you have written, especially, but not exclusively, those with a slant towards education. ■

THE Video Genie EG3000 Series

WITH *16K user RAM plus extended 12K Microsoft BASIC in ROM *Fully TRS-80 Level II software compatible *Huge range of software already available *Self contained, PSU, UHF modulator, and cassette *Simply plugs into video monitor or UHF TV *Full expansion to disks, and printer *Absolutely complete just fit into mains plug.

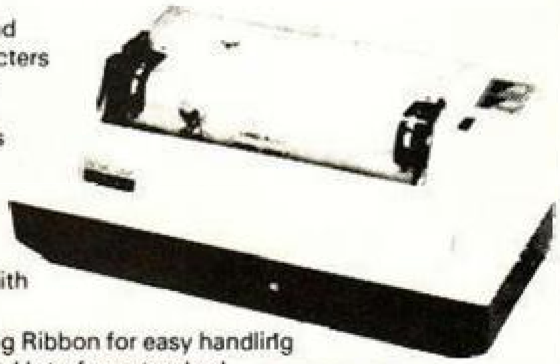


16K **£299** + VAT

GP80

by seikosha

The most compact 80 column impact graphic-dot printer available - at a very compact price



Graphics, Normal and Double- Width Characters can be printed on the same line. Pine Feed Tractor is equipped as standard.
Two Line-Feed Commands (1/6 and 1/9 inch)
Print on Plain Paper with two copies
Continuous Self-Inking Ribbon for easy handling
Centronics type Parallel Interface standard
Wide Range of Optional Interface Boards
Self-Test Programme standard

£225 + VAT

Acorn Atom

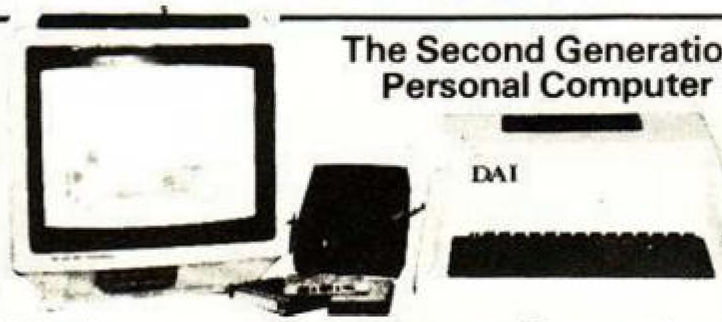
' Unique in concept — the home computer that grows as you do!

Special features include
*FULL SIZED KEYBOARD
*ASSEMBLER AND BASIC
*TOP QUALITY MOULDED CASE
*HIGH RESOLUTION COLOUR GRAPHICS

Simple to build, simple to operate. A powerful, full facility computer with all the features you would expect.
Just connect the assembled computer to any domestic TV and power source and you are ready to begin.
Full-sized QWERTY keyboard 6502 Microprocessor Rugged injection-moulded case 2K RAM 8K HYPER-ROM 23 integrated circuits and sockets Audio cassette interface UHF TV output Full assembly instructions

£120 + VAT

The Second Generation Personal Computer



Highest performance · lowest price

*48K (8080A)
*16 Colours or shades of Grey
*Multiple High Resolution Graphics Modes (64 x 71, 129 x 159, 255 x 335)
*Character mode (60 x 24)
*Split Screen Modes
*Full ASCII Upper & Lower Character Set
*Unique graphical — Sound Commands for Smooth Music, random frequencies & enveloped sound!
*RS232 I/F

DAI Personal Computer at only **£595** + VAT

Paper Tiger 460

The Paper Tiger 460 sets new standards by incorporating many features previously only available on units costing much more.

Features like a specially developed nine wire 'staggered column' head which overlaps the dots of each matrix character with just one pass of the printhead giving a dense, high quality print image without reducing the units 160 c.p.s. print speed.

It also offers a bi-directional logic seeking device to enhance its print optimisation characteristics and wide range of 'print versatility' features such as mono or proportional spacing, automatic justification, programmable horizontal and vertical tabbing, and 'fine' positioning for word processing applications.

£795 + VAT

Paper Tiger 560

The Paper Tiger 560 is the first printer which bridges the gap between convenitonal matrix and 'daisy wheel' types offering quality printing at a relatively low price.
Full 'width' 132 column printing at 160 c.p.s., a unique nine wire 'staggered' print head, bi-directional printing, an inbuilt tractor feed and a host of selectable features set it apart from ordinary matrix printers.
Plus for even greater versatility a full dot plot graphics facility if supplied which includes a 2K buffer.

£995 - VAT

SUPERBRAIN

SuperBrain's CP/M operating system boasts an overwhelming amount of available software in BASIC, FORTRAN, COBOL, and APL. Whatever your application. . . General Ledger, Accounts Receivable, Payroll, Inventory or Word Processing, SuperBrain is tops in its class.

320K £1850 700K £2400
1.5Mb £2750

COMMODORE COMPUTERS

PET 8K	£415
PET 16K	£525
PET 32K	£650
PET 8032	£895

DISK DRIVES

4040	£695

PRINTERS

EPSON TX80B (inc. I/F & cable)	£299
EPSON MX80T	£395
ANADEX DP8000	£495
ANADEX DP9500	£895
ANADEX DP9501	£995

VIDEO MONITORS

10" BLACK & WHITE	£85
10" GREEN SCREEN	£95

Aculab

Floppy Tape
The tape that behaves like a disk.

Connects directly to TRS-80 level 2 keyboard. Operating and file handling software in ROM 8 commands add 12 powerful functions to level 2 BASIC. No buttons, switches or volume controls. Full control of all functions from keyboard or program. Maintains directory with up to 32 files on each tape.

for all TRS 80 & Video Genie owners

TRS80 version: £165
Video Genie: £170

**Please add £10 Securicor delivery charge to all computers etc.
Plus 15% VAT on all prices.**

Books & bits

Books — Manuals — Diskettes — ribbons — Paper — chips (2114 x 2 1K) £4 pair.
RS232 to Centronics interfaces £40 etc. etc.
A variety of second-hand computer equipment usually available, spares, repairs and service.

MicroStyle

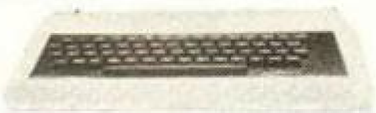
**29 Belvedere, Lansdown Road, Bath BA1 5HR
Telephone: (0225) 334659**

We are only 30 minutes from CENTRAL LONDON

and only a few minutes from Your Postbox

Why not try our new Mail Order Service?

ACORN ATOM



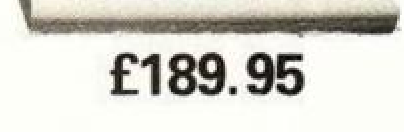
assembled from £155.25

BBC MICROCOMPUTER

Phone for details and availability

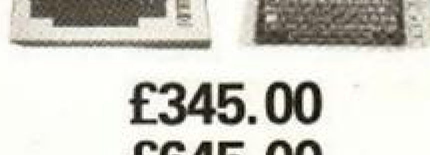
SOFTWARE & BOOKS AVAILABLE

VIC-20



£189.95

ATARI 400 & 800



£345.00
£645.00

STOP PRESS: Bug Byte, ZX81 + Atom Software now in stock

For further information please send large s.a.e.

Computers for All

BARCLAYCARD VISA

72 North Street, Romford, Essex. Tel: 0708 60725

MicroValue

MONEY SAVERS

with the exclusive MicroValue 12 Months Warranty*

New CP/M System based on Gemini Multiboard

Gemini Microcomputers is launching at Which Computer? Show a new system based around its successful MultiBoard range, and MicroValue dealers will have first deliveries in February. It will have:

HARDWARE

- Twin Z80A CP/M System
- 64K Dynamic RAM
- 700K Disk Storage (Formatted)
- 80 x 25 Screen Format
- Inverse Video
- Prog. Character Generator
- 160 x 75 Pixel Graphics
- Centronics Parallel I/O
- RS232 I/O
- Light pen interface
- 59-Key ASCII Keyboard

SOFTWARE

- Full 64K CP/M 2.2 with screen edit facility
- Comal-80 structured BASIC
- GEM-ZAP Assembler/Editor
- GEM-PEN Text editor
- GEM-DE BUG debugging software

MicroValue price

£1,450 +VAT

MicroValue Exclusive

80 × 25 Video for Nascom

Nascom owners can now have a professional 80 × 25 Video display by using the Gemini G812 Intelligent Video Card with onboard Z80A. This card does not occupy system memory space and provides over 50 user controllable functions including prog character set, fully compatible with Gemini G805 and G815/809 Disk Systems. Built and tested.

£140 +VAT

MicroValue Exclusive

NEW

I/O Board for Nascom & Gemini Multiboard Systems

Quantum I/O

The new Quantum Micros I/O board takes the unique approach to the problems of interfacing your Nascom or Gemini Multiboard to external devices. This 80 Bus and Nasbus compatible card is supplied fully built, populated and tested and includes three Z80 PIOs, a CTC and a Real Time Clock with battery backup. In addition, a range of "daughter" boards that attach straight to the I/O board are under development catering for a wide variety of interfacing requirements.

Quantum I/O board MicroValue price – **£140** + VAT

Prototyping daughter board MicroValue price – **£20** + VAT

IEEE-488

The EV Computers' IEEE-488 card is an 80 Bus and Nasbus compatible card designed to fully implement all IEEE-488 interface functions. This built and tested card gives the user a very cost effective and versatile method of controlling any equipment fitted with a standard IEEE-488 or GPIB interface.

MicroValue introductory price

£140 + VAT

MicroValue Exclusive

New Software for Nascom Systems

POLYDOS 1 A disk operating system for use with Nascom 1 or 2 and Gemini G805 Disk Systems. An incomparable and extremely well presented DOS that includes an editor and assembler and adds disk commands to the Nascom BASIC. MicroValue price **£90 + VAT**

MATHSPAK Double precision maths package on tape. MicroValue price **£13 + VAT**

MATHSPAK Handler Used in conjunction with MATHSPAK. MicroValue price – **£9.95 + VAT**

Command Extender For use with MATHSPAK it extends BASIC's reserve word list. MicroValue price **£9.95 + VAT**

Logic Soft Relocater An integrated assembler and disassembler package which allows disassembly and reassembly from anywhere on the memory map. MicroValue price **£13 + VAT**

SAVE MORE MONEY

Standard Firmware for Nascom at Reduced prices

NASPEN	RRP £30 + VAT	**MicroValue price £20 + VAT**
Nas-Sys 3	RRP £25 + VAT	**MicroValue price £20 + VAT**
NasDis – D-Bug (EPROM)	RRP £50 + VAT	**MicroValue price £30 + VAT**
NasDis – D-Bug (TAPE)	RRP £40 + VAT	**MicroValue price £20 + VAT**
Imprint	RRP £30 + VAT	**MicroValue price £20 + VAT**
Bits & PCs Prog. Aid	£28 + VAT	**MicroValue price £20 + VAT**

*MicroValue Warranty

All products, except kits and Nascom Imp, sold by MicroValue dealers are supplied with 12 months warranty and will be replaced or repaired by any dealer (even if you didn't buy it from him) in the group in the event of faulty manufacture.

YOUR LOCAL MICROVALUE DEALER

All the products on these two pages are available while stocks last from the MicroValue dealers listed on right. (Mail order enquiries should telephone for delivery dates and post and packing costs.) Access and Barclaycard welcome.

BITS & PC'S
4 Westgate, Wetherby, W. Yorks.
Tel: (0937) 63774.

ELECTROVALUE LTD.
700 Burnage Lane, Burnage, Manchester M19 1NA.
Tel: (061) 431 4866.

28 St Judes, Englefield Green, Egham, Surrey TW20 0HB.
Tel: (0784) 33603. Tlx: 264475.

SKYTRONICS,
2 North Road, The Park, Nottingham.
Tel: (0602) 45053/45215

TARGET ELECTRONICS
16 Cherry Lane, Bristol BS1 3NG.
Tel: (0272) 421196.

INTERFACE COMPONENTS LTD.
Oakfield Corner, Sycamore Road, Amersham, Bucks.
Tel: (02403) 22307. Tlx: 837788.

HENRY'S RADIO
404 Edgware Road, London W2.
Tel: (01) 402 6822.
Tlx: 262284 (quote ref: 1400).

LEEDS COMPUTER CENTRE,
62 The Balcony, Merrion Centre, Leeds.
Tel: (0532) 458877

ATOM CASSETTES

Eric Deeson, in his review of Acorn Atom Cassettes, *Your Computer* November 1981, states that he was unable to load all four of the cassettes which we submitted. There is a wide disparity among tape recorders and the success in loading one's own tape is always much greater than with those recorded on other people's equipment. Despite this, the percentage of tapes returned to us is no higher than three percent — and we have never failed to provide an adequate replacement, which incidentally we despatch, first-class post, by return.

If this was meant to be a serious review of software, why did Deeson not persist until he could make fair comparisons — albeit with some comments on his loading problems? One very odd suggestion was that the cassette loading system might be more troubled by machine code than Basic. To put the record straight, Astrobirds and Invader Force are in machine code but Music Box and Histats are predominantly written in Basic.

Regarding documentation and instructions, these are provided where necessary. Instructions programs are usually put after the main program since once these are mastered, it saves one having to search the tape for the main program each time.

Deeson mentioned that Acornsoft offered 17 program cassettes. He did not state that the Program Power catalogue sent to him contained 33. Also mentioned was the fact that another supplier offered chips and tapes. Our catalogue states that we also do this.

Finally, I would like to point out that more than 90 percent of our programs are written by independent Atom owners and that, to provide a varied selection of high-quality programs, we have reviewed and turned down many more programs than we have accepted.

R G Simpson,
Program Power,
Leeds.

LOADING TIP

In the November issue, P R Ainsworth wrote a letter on how to overcome loading problems on the ZX-81. Even using these techniques I was having very limited success — that is, until I accidentally hit on something which has meant I can load all my programs first time, every time.

When loading programs, disconnect the microphone lead from the tape recorder. Then instead of letting it hang there, hold the jack plug tightly between forefinger and thumb. Any distortion should disappear from the screen and programs will load normally.

It works for my system and I checked with a colleague who reported it a success. There is probably a technical explanation but even in the absence of one, it should help some of those who have loading problems.

Steve Clarke,
Ipswich,
Suffolk.

SINCLAIR MOANS...

How right D B Orpin is regarding the apparent lack of back-up or even interest that Sinclair Research seems to show to one after buying a ZX-81 — Your Letters, December issue.

Three months ago when I was having trouble loading a cassette; I telephoned and was told that there was a leaflet regarding this and I was assured that a copy would be sent to me.

More than a month ago I sent a stamped, addressed envelope with a request for a circuit diagram. I have had no response on either count.

B G Taylor,
Lincoln.

...AND PRAISE

I feel I must counterbalance the no doubt justified criticism of Sinclair Research — "Sinclair Critic" November 1981 — with my own experience. I received my ZX-81 package on time, the query I had with loading from tape was answered very promptly and, as I had suspected, proved to be due to "finger trouble" on my part.

It seems that earlier criticism may have been justified, but that now Sinclair Research appears to have got its act together. I can only wish this remarkable British company even more success.

D J Adamson,
Nottingham.

EXCHANGE OFFER

In my article in the December 1981 issue on generating sound effects on the Sharp MZ-80K, I mentioned that I would be pleased to send a copy of my effects demonstration program to anyone who sent me a cassette and stamped, addressed envelope. The response to this has been very good — more than 150 letters so far and more with every post.

My offer is still open but I can only reply to those letters containing a stamped, addressed envelope, a cassette and, ideally, a swap program.

Bob Edwards,
Huyton,
Liverpool.

CHESS STRATEGY

I read John White's interesting article in the December issue on chess end-games, and after some research, felt that I should update some of the information given on how chess programs work compared with the way in which people play chess.

I have noticed that ever since computer chess programs have been written, they have all relied on a certain level of insight into the game — that is, looking ahead as many moves as possible. It is assumed then, the more more insight that the program has, the better the computer will play chess. The fact that a relatively good chess player can easily beat the most complex chess programs shows this not to be the case at all.

The reason is that the human player, whether it be novice or chess master, does not assess a chessboard by looking ahead by any more than three or four moves. A chess master rarely looks any more moves ahead than a novice chess player.

The factor that makes the master far better than the novice is the level of thought about the game. The master does not consider the strategically bad moves. His thought level is way above that of a novice.

Computer chess programs, all of which rely on looking ahead, have not been taught to think on the higher level; the strategy has been to use brute-force look-ahead, hoping to crush all types of opposition. This has not worked, so surely it would be better to evolve a program that works on this higher level, ignoring bad moves using data based on the strategic position at the time. Writing more and more complex programs to see further and further into the game overlooks the fact that there is little to be gained by increasing the computer's strategic insight.

The prompter of these conclusions was a Dutch psychologist, Adriaan de Groot. In the 1940s he made studies on how chess novices and masters perceive the same type of chess situation. There is obviously a higher level of description of a chessboard than just K-K2 or R-Q7, and de Groot's findings imply that masters perceive the distribution of pieces on a chessboard in strategic chunks. The master somehow builds up a complicated mental picture of the game.

De Groot proved this by showing the speed with which a master could reproduce a situation on a chessboard, compared with the chess novice's slow reconstruction, after both had been shown a chess situation for five seconds. The most important thing de Groot noted, however, was that where the novice chooses the positions of individual pieces badly, strategically altering the whole game, the master tends to choose the positions of whole groups of pieces badly.

So taking this even further, de Groot presented novice and master with chessboards on which were a number of randomly-placed pieces. Not surprisingly the master did no better than the novice when attempting to reconstruct the positions.

T M Horner,
Portsmouth,
Hampshire.

ZX-81 QUIRK

Computers do not make mistakes? My ZX-81 does — I asked it to evaluate INT (10*X) in four ways:

- FOR X = 0 TO 1 STEP 0.1
- FOR X = 1 TO 0 STEP −0.1
- FOR X = 0 TO 1 STEP 0.05
- FOR X = 1 TO 0 STEP −0.05

Of the four sets of results, no two agree throughout. Furthermore, the first and third sets do not even complete the For-Next loop: they do not evaluate the expression for X = 1. Only the second set of results is completely correct. The program illustrates the INT (10*X) results.

```
  2 REM Y=INT(10*X)
  4 REM BY IAN COPESTAKE 81 NO 05
 10 PRINT TAB 9;"STEP  0.1-0.1 .05-.05"
 20 FOR X=0 TO 1.01 STEP .05
 30 PRINT "X=";X;TAB 10;"Y="
 40 NEXT X
 50 LET T=16
 60 FOR X=0 TO 1 STEP 0.1
 70 GOSUB 200
 80 LET T=20
 90 FOR X=1 TO 0 STEP -0.1
100 GOSUB 200
110 LET T=25
120 FOR X=0 TO 1 STEP .05
130 GOSUB 200
140 LET T=29
150 FOR X=1 TO 0 STEP -.05
160 GOSUB 200
170 STOP
200 LET Y=INT(10*X)
210 PRINT AT X*20+1,T;Y
220 NEXT X
230 RETURN
```

Another ZX-81 quirk, possibly related, concerns the Plot statement. Co-ordinates are rounded to the nearest integer, and 1.5 becomes 2, 2.5 becomes 3, etc. Yet 0.5 becomes 0. Does anyone have an explanation?

Ian Copestake,
Woking,
Surrey.

THAT UNJUST LEVY

Why should I, or other innocent members of the general public, be penalised for offences committed by persons unknown and pay the proposed levy on the sale of blank cassette tapes? If the British Phonographic Institute knows of copyright infringements why does it not take action against the person or persons concerned?

The BPI should pursue a course of raising the standard of records produced by its members and press for a return to the days when a long-playing record was long playing.

S Halstead,
Huddersfield,
West Yorkshire.

ZX packages now from ICL

As SALES of the ZX-81 microcomputer broke the 250,000 mark, ICL announced a new range of ZX-81 software. Six cassettes are available, each consisting of a mixture of games, educational and business programs. Four of the cassettes are suitable for the 16K and two for the 1K machines. W H Smith has already ordered more than 100,000 cassettes.

ICL is also holding discussions with Sinclair Research on the development of a low-cost integrated terminal/digital telephone workstation, using Sinclair's flat-tube technology and Sinclair Basic. Designed as a future peripheral to ICL's DNX-2000 digital PABX, the workstation is provisionally scheduled for production in 1983.

With production of the ZX-81 currently running at 50,000 a month, more than 60 per cent for export, Sinclair has installed more microcomputers worldwide than any other personal computer manufacturer.

Clive Sinclair, chairman of Sinclair Research, said: "Together these figures and new developments completed a very successful year for us. With so many units in the field, supported by a large number of new projects using Sinclair Basic, we believe our language now merits serious consideration by the industry as the 'standard', if confusion, particularly in the educational field, is to be avoided in the future".

Atom to gain new power

A NEW PROGRAM for the Acorn Atom, which gives it some of the facilities of the BBC Microcomputer, such as the ability to mix high-resolution graphics and text anywhere on the screen, has been developed by Computer Concepts.

One of the limitations of the normal Atom is that it only displays 16 lines of text with 32 characters per line. Yet Computer Concepts' Softscreen enables the Atom to display 24 lines of text with up to 42 characters per line.

The Softscreen program uses graphics mode 4 to display the text, though it can also use modes 1, 2 and 3. The lower the resolution, the larger the text. The program is in machine code, occupies 2.75K and is located from 2900 to 3400. As it uses graphics mode 4, the program requires a 6K graphics RAM.

Softscreen also gives the Atom the ability to define text windows. Once the window size has been defined, all text sent to the screen automatically appears in the text window.

The Softscreen program is available on cassette for £11.40 from Computer Concepts, 16 Wayside, Chipperfield, Hertfordshire, WD4 9JJ.

More software broadcasts planned

THE ENORMOUS response to the first computer program to go out over the air — transmitted by BBC Television — has encouraged *Tomorrow's World* to consider running a follow-up experiment using a longer routine. The first trial was restricted to 20 seconds, as the bleeps were of interest only to those people with computers. Now *Tomorrow's World* is looking at the possibility of transmitting another program at night, after the normal TV programmes have closed down.

Despite a hiccup in the studio which prevented the *Tomorrow's World* team from running the first program on their own computer, the experiment was very successful. Trevor Taylor, who produced the program for *Tomorrow's World*, tells *Your Computer* that more than 2,000 people from as far north as the Shetlands, and as far south as France, had written to say they had succeeded in recording and running the program. Two six-year-olds were among those who responded to the program, while one man who replied claimed to have seen the first ever TV broadcast half a century ago.

The program, consisting of a 20-second burst of bleeps, was designed to be picked up by cassette recorders, either direct from the TV ear socket or by placing the cassette microphone close to the speaker.

The ZX-81 listing broadcast.

```
 10 PRINT "ENTER NAME"
 20 INPUT N$
 30 CLS
 40 PRINT "WELL DONE", N$
 50 PRINT "IT WORKED"
 90 PRINT "MORE? Y/N"
100 INPUT Q$
110 IF Q$<>"Y" THEN GOTO 30
120 CLS
140 PRINT "TOMORROWS WORLD"
160 PRINT "WITH"
180 PRINT "MICHAEL, JUDITH"
220 PRINT "KIERAN, SU"
260 PRINT "AND" (IN GRAPHICS)
280 PRINT N$
320 STOP
```

The Tomorrow's World *team.*

Divided into two versions, one for the Apple and one for the ZX-81, the program was preceded by a low tone to enable the viewer to find the correct volume setting.

When the Apple program was loaded into the computer it asked the viewer's name and then printed the following message on the TV screen: "Welcome . . . to *Tomorrow's World* test broadcast".

Next, the viewer was asked if he wanted to continue. If he answered "Yes", the program scrolled the *Tomorrow's World* credits, listing the names of the presenters, and finished by adding "With special thanks to . . .".

The ZX-81 program followed a similar format, but used fewer words.

One of the unexpected results of the first program was the number of people who used video recorders to tape the show and then succeeded in dubbing off the audio to run the program. P O'Brien of Caernarfon, Wales, used this method to send a copy of the ZX-81 listing to *Your Computer*.

Educational program service on Prestel

PRESTEL'S FIRST educational telesoftware service enables subscribers to use the telephone to call down computer programs stored on Prestel and record them on a microcomputer. The programs can then be used in the normal way.

Geoffrey Hubbard, director of the Council for Educational Technology which launched the scheme, said: "CET is always on the lookout for educational applications of the new technologies. This is a beautiful example of such an application, but its success will depend on there being good materials available to serve useful educational purposes. That's why this CET project is starting in the schools".

A group of five schools and institutions are taking part in a two-year trial to evaluate the telesoftware service, and CET is negotiating with another 15. As the software is only available for the Research Machines 380-Z microcomputer, possession of this machine was one of the criteria used in selecting the first group. Other equipment, such as the software, Modem and barrier cable, was financed by CET.

Details of the telesoftware service and the available programs are stored on Prestel page 2114, and can also be obtained from the Council for Educational Technology, 3 Devonshire Street, London W1N 2BA. Telephone: 01-580 7553.

Local software for BBC Micro

DATRON MICRO Centre, Sheffield, is to support the BBC Microcomputer and computer-literacy programmes by acting as a referral centre, despite the BBC's refusal to allow a dealer discount structure.

Ian Dunkley, Datron's managing director and chairman of the Computer Retailers' Association, said: "While I still have misgivings over the methods of marketing, which I believe may reduce the amount of support available, I feel that we owe it to our wide base of educational users to provide local support".

Sinclair users in Pet scheme

IN A BID to boost sales of the Pet microcomputer, Commodore has started a trade-in scheme. Owners of the Sinclair ZX-80 and ZX-81 will be able to trade-in their machines for a £50 discount on a new Pet. The scheme, which is running until March 31, only applies to the Pet and is not available on Commodore's other computers.

Commodore is still uncertain what to do with the collected ZX-80 and ZX-81s. Keith Hall said they would like to donate them to charity, "providing we obtain a reasonable number of machines in satisfactory condition". Hall also said it was too early to say how the scheme was doing.

Clive Sinclair, head of Sinclair Research, said the scheme was "flattering, but I don't think it will do us any harm. We aren't in direct competition".

Sinclair ZX-80 and ZX-81 owners who wish to take advantage of the Pet scheme should contact Commodore Information Centre, Baker Street, High Wycombe, Buckinghamshire. Telephone: Slough (75) 79292.

Schools target of 50 routines

EDUCARE, a software house specialising in education, has released a book of 50 educational programs for the ZX-81. Designed for primary-level education, the programs are written in Basic and will fit the unexpanded 1K machine.

Many of the programs are in the form of games, such as Mastermind and Simon-spell. Other programs include Graph-plotter, Histogram, Times-tables, Conversation, Clock and Money. With an average length of 25 lines, the programs are easy to load and run.

Educare's 50 1K programs for primary education is available from Educare, 139a Sloane Street, London SW1 9AY, and costs £4.95.

Bio-computers: 'by 1983'

A PROTOTYPE biological computer could be in operation by 1983. Dr James McAlear, president of the U.S. bio-technology company EMV Associates, forecast that his company would be making some pilot biological chips by the end of next year.

The pilot bio-chips would be produced by an electron beam that deposited microscopic circuit designs in gold, silver or lead on to protein, said Dr McAlear. The bio-chips would operate in three dimensions and have 10,000 times more capacity than the present silicon chip. Taking the process one stage further, the bio-chip could be used to make a self-repairing biological computer, he claimed.

British scientists are sceptical, however, that a biological computer could become reality so quickly. Research director of GEC Derek Roberts, said it was "absolute nonsense" to talk of such a computer being built and tested by 1983, while Professor John Barker of Warwick University said there were still a number of problems which had yet to be solved.

The U.S., however, is already looking beyond the bio-computer. EMV Associates has been awarded a $30,000 grant from the U.S. National Science Foundation to make a bio-chip that could be linked to the human nervous system. The living computer could be here sooner than we think.

Flawed chips create delays

FAULTY microchips have seriously delayed production of the BBC Microcomputer. Acorn, the company which designed the computer for the BBC, originally expected to have 12,000 micros on the market by the beginning of 1982. Instead, Acorn had only produced 300 micros in full working order.

The root of Acorn's troubles lies in the uncommitted-logic array (ULA) chip designed to control the computer's display screen. The Ferranti-manufactured ULA chip has not been running as fast as it should, due to problems in the doping process which creates conductive paths in the silicon.

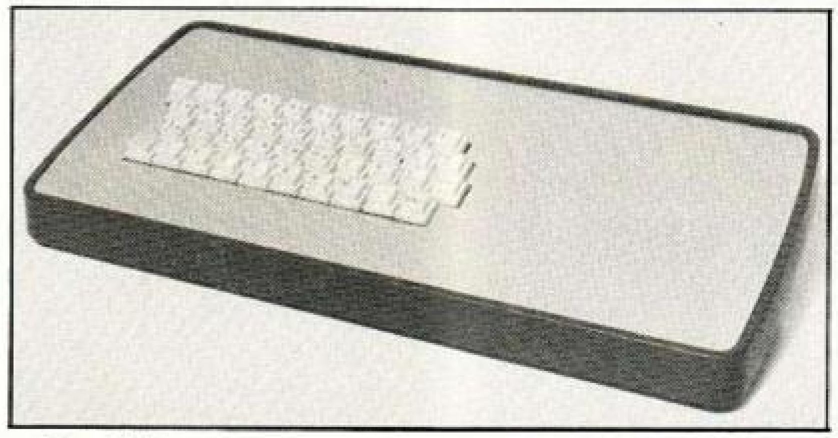
The BBC TV programme on computing has been put back to February 14, in the hope that more micros will be available. However, transmissions to schools went ahead as planned — the first programme was shown on January 11.

No-solder Protos has fitting role as hard-wearing ZX-81 keyboard

A HEAVY-DUTY keyboard and enclosure has been developed for the ZX-81 by Protos Computer Systems. The 40-key Sinclair-coded board uses mechanical contact key switches and a printed-circuit board. A flexible connector joins the Protos board to the ZX-81 via push-fit sockets.

To fit the Protos keyboard, the ZX-81 is first removed from its ABS case. It is then mounted inside the Protos enclosure using four Philips-type screws. No soldering is required and all electrical connections are provided by either plugs or sockets.

A side port on the Protos enclosure allows access to the edge board connector while tape in/out, power and UHF connections are made through the back.



Other ZX-81 peripherals planned for introduction during 1982 include buffered input/outputs, power supply, RAM expansion and keyboard sound generator.

The ZX-81 keyboard and enclosure costs £64.95 including VAT — post and packaging is £2.50 extra — and is available from Protos Computer Systems, Frome Computing, 20 Ashtree Road, Frome, Somerset, BA11 2AS.

Interface that offers control

RAMPORT, A 16K memory expansion card, enables Sinclair ZX-81 users to interface their microcomputers to any electrical equipment. Manufactured by Componedex Ltd, RAMport's analogue and digital input/output ports allows the ZX-81 to control anything from the central heating to the points on a model train set.

A spokesman for Componedex said: "The possibilities of RAMport are enormous".

RAMport is available through mail order for £49.95 from Componedex Ltd, PO Box 33, Leighton Buzzard, Bedfordshire, LU7 7UK.

The champions of micro chess

THE GRANDMASTERS chess tournament held at the Brighton Conference Centre featured SciSys world champion chess computer the Chess Champion Mark V. Winner of the 1981 world microcomputer chess championship, the Mark V is capable of playing 12 games of chess simultaneously.

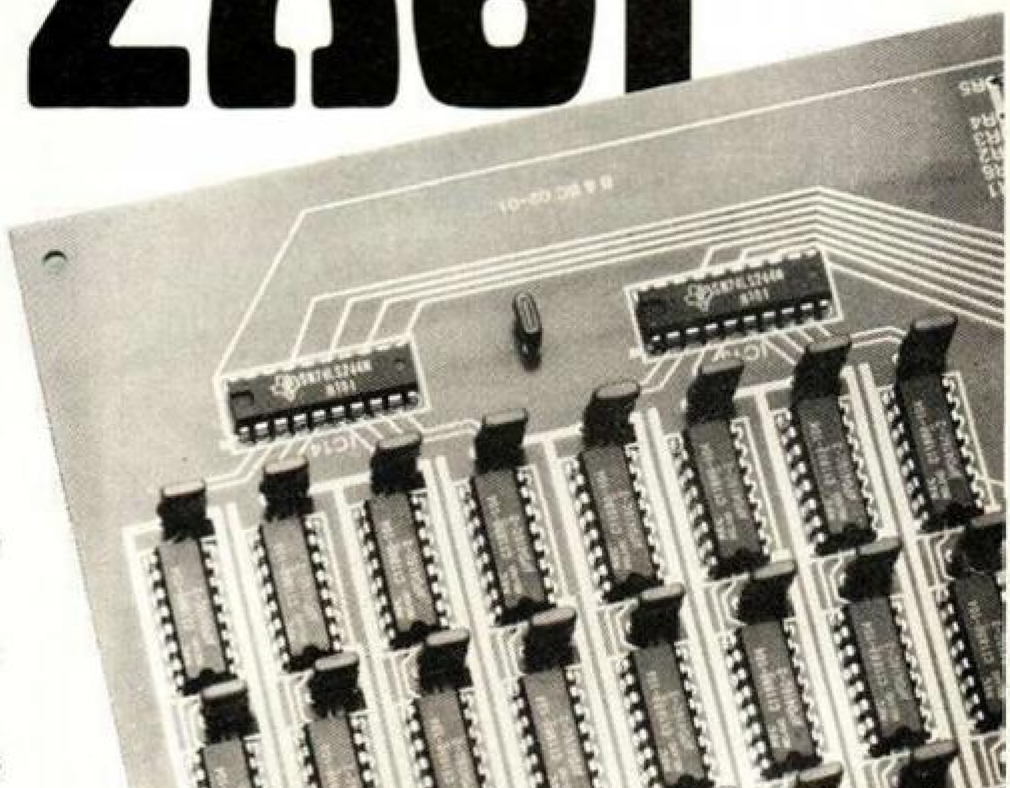
In a subsidiary competition, 20 local schoolchildren took on the Mark V — the top six children won chess computers of their own.

Another of SciSys chess computers, Intelligent Chess, was used to inform chess fans of the progress of the main competitors. The main tournament, which was won by Murray Chandler, included two of Britain's leading young players, Nigel Short 16, the world's youngest International Master, and Stuart Conquest 14, the current Under-16 World Champion.

Caernarfon-based Arfon Microelectronics Ltd has developed a seven-cartridge, fully-integrated expansion system for the Vic-20. All parts are housed in an aluminium shell while an optional lid provides a base for the TV. The original power supply is replaced with a more powerful unit built around a toroidal transformer which reduces any power surge problems. A 24V rail and socket will power the Vic-20 printer being developed by Arfon. All the ports are accessible and the modulator is housed at the back of the expansion board. Approved by Commodore, the expansion system is expected to retail for about £85 plus VAT. Memory cartridges are now available in 3K with two EPROM sockets, 8K or 16K packs which will fit both the Vic-20 and the expansion board. Arfon Microelectronics Ltd can be contacted on 0286-5005.

MEMOTECH

48K memory extension for the ZX81



The MEMOTECH memory extension board will allow the ZX81 to run 48K BASIC programs which may include up to 16K of assembly code.
The unit contains a genuine 48K of user transparent RAM, and accepts such BASIC commands as:
10 DIM A(9000).
A range of I/O Port boards and A/D, D/A convertors is available. The unit is compatible with the ZX Printer, and RS232 interface will be available soon.
The MEMOTECH memory has a fully buffered control-data-address bus with PCB 40 way header plug.
The ZX81 sits on a custom built case which contains the **MEMOTECH** memory and a power supply which not only powers the **MEMOTECH** memory, but also the ZX81.
All Leads are provided. The **MEMOTECH** memory extension board costs: £109.00 + VAT in kit form, £129.00 + VAT assembled. 15% Educational user discounts are available.

Please make cheques payable to:

MEMOTECH

(Sales Dept.) 103, Walton Street, Oxford. OX2 6EB.

Computer Club is here to encourage you to start your own local computer club or, if one already exists, to join it and become involved. Each month we will devote the page to new ideas from local clubs. We would like to hear of anything which has made a club a success, or of any projects or programs you are developing.

TUG of loyalties for Tangerine users

Tangerine Users' Group has over 1,000 members and manufactures its own software. Brendon Gore reports from its Bournemouth headquarters.

Of relations with Tangerine, Users' Group managing director Bob Green says, "TUG supports the system, not the manufacturers".

ERIC, THE Tangerine Users' Group (TUG) mascot, has a considerable following and even stars in his own cartoon strip. Eric, and his girlfriend Silicon Val, are an important part of the Group. Eric shows that computing can be fun, says TUG managing director Bob Green, and reflects the relaxed attitude that TUG members have towards their hobby. Eric may look like a 20-year-old integrated circuit, but he has a personality all of his own.

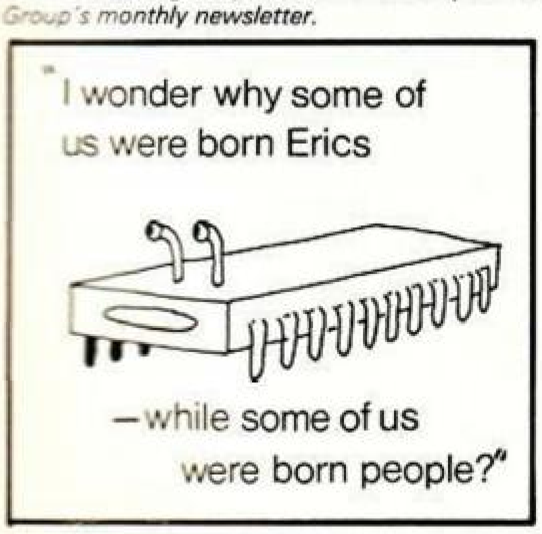
Based in Bournemouth, TUG has been in existence for 18 months. Bob Green, who helped found the group, formed it into a limited company in October 1980. It now has more than 1,000 members, 85 percent in the U.K. A year's subscription costs £12.50.

"TUG was generated by the members' need to know", says Bob Green. "They needed to share other people's experiences and ideas".

With a membership which has more than doubled in the past year, TUG appears to be satisfying that need. A newsletter is distributed once a month to each member. It contains programs, letters, news, Eric's cartoon and advertisements for TUG's own hardware and software. Approximately half the newsletter is compiled by Bob Green while the other half is contributed by members.

TUG is one of the few, if not the only, user group to manufacture and market its own hardware. Products range from the EPROM programmer to a high-definition programmable graphic module designed to run on both Microtan 1 and 2 systems. TUG also makes a mass EPROM storage board, which went on show at Breadboard '81. The board caters for 24/48K of EPROM, 24K in 2K EPROMs and 48K in 4K EPROMs. A 32/64K board is under consideration. The 24/48K board costs £53.67 fully assembled, but TUG members qualify for a 25 per cent discount.

Eric features regularly in the cartoon strip of the Group's monthly newsletter.



The three most popular games programs marketed by TUG are Asteroids, Dive Bomber and Shuttle Lander. Dive Bomber consists of three advanced jump-jets with computer-controlled anti-laser missiles. The aim of the game is to bomb the enemy's laser defences, manoeuvring the jets to avoid incoming fire. To make the game more interesting, the jets are allowed only limited supplies of fuel and missiles.

However, not everyone has been pleased with TUG's venture into business. Tangerine has set up a rival organisation around a bi-monthly magazine, *Tansoft Gazette*, edited by Paul Kausman. The first edition was distributed free to more than 3,000 Tangerine computer owners in November. Future copies can be obtained by paying £15 for a year's subscription.

Relations between TUG and Tangerine are somewhat strained at the moment. "We support the system, not the manufacturers", says Bob Green. He accuses Tangerine of failing to supply users' needs and says TUG has just tried to fill the gap. "After two years, the Microtan still has no floppy discs".

Paul Kausman admits that Tangerine has neglected the Microtan in favour of the Tantel Adaptor, but says this is now being rectified. "The next product we are releasing is a floppy-disc package", he says. "It will include a Forth language specially adapted and extended for the Microtan". The package should soon be available.

Bob Green believes user groups must become more professional if they are to remain independent. "User groups must generate a more progressive attitude towards the systems they support, otherwise the future will be in the hands of the commercial industry", he says. "They will have to design and manufacture their own add-ons to the system".

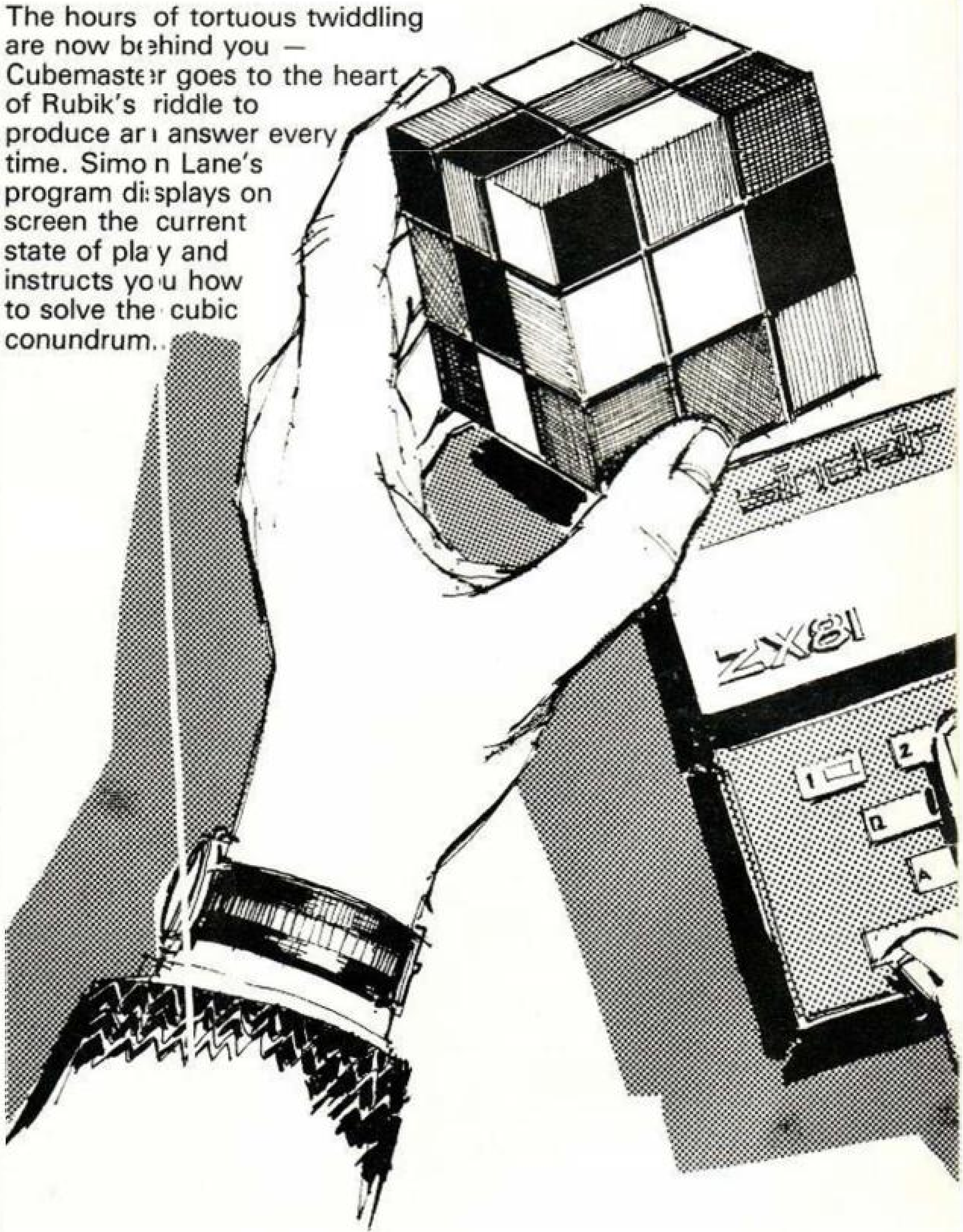
As a step in this direction, TUG is looking for new premises as well as manufacturing a line in sweat shirts. New hardware and software products are ready for production. "We have 21 products waiting to hit the streets", says Green. Yet, wary of saturating the market, TUG is releasing its products at a controlled rate.

TUG's future plans include utilising Prestel to the maximum, but only when it has developed to the point where it can cope with their demands. "Prestel has a few years to go before it becomes a viable proposition", says Bob Green. ■

SOFTWARE

ZX-81 CUBEMASTER

The hours of tortuous twiddling are now behind you — Cubemaster goes to the heart of Rubik's riddle to produce an answer every time. Simon Lane's program displays on screen the current state of play and instructs you how to solve the cubic conundrum.



THE FORMULA for solving the cube which I have used in this program is relatively simple compared with certain techniques, and solves the Rubik's problem in three steps. First, it completes the top side, then the next layer down, and finally the bottom side.

For convenience, the computer always chooses the red side as the top side, which means the orange side is on the bottom. It would be possible to complete any side first. The first task is to put the top-side edges in their correct positions and when the top-side corners are also correct, the side is complete. For the middle layer all that needs to be done is to put the middle edges in their correct positions. Although this seems simple, it can be very time-consuming as the position on the top side must not be destroyed.

The bottom side is the most difficult to complete, since now the positions on the top side and the middle layer must be preserved. For that reason, this part of the formula takes the most time to execute, and has to be reduced to a number of simpler steps:

- Placing the bottom-side corners in their correct positions.
- Turning them so that they are facing the correct way.
- Adjusting the bottom-side edges so that they are all orange side down.
- Placing them in their correct positions.

In the program itself, the formula appears as a large number of conditional statements in the form: "If a certain position Then make

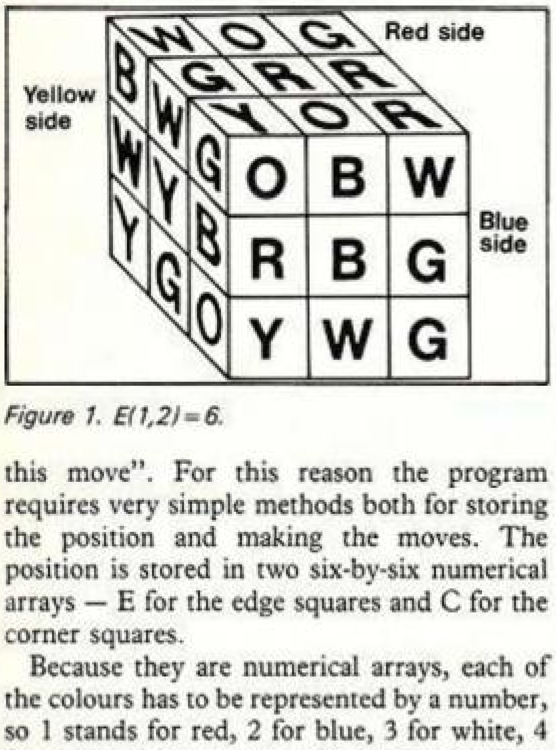


Figure 1. E(1,2) = 6.

this move". For this reason the program requires very simple methods both for storing the position and making the moves. The position is stored in two six-by-six numerical arrays — E for the edge squares and C for the corner squares.

Because they are numerical arrays, each of the colours has to be represented by a number, so 1 stands for red, 2 for blue, 3 for white, 4 for green, 5 for yellow and 6 for orange. Each face is identified by the colour of the square in its centre. So, for example, the side with a red square in its centre is side number 1.

Each edge square is identified by two co-ordinates. The first co-ordinate represents the face it is on, and the second the face it is next to. For example, if E (1,2) = 6, then that would mean that the edge square on the red — which was next to the blue face — was coloured orange — see figure 1.

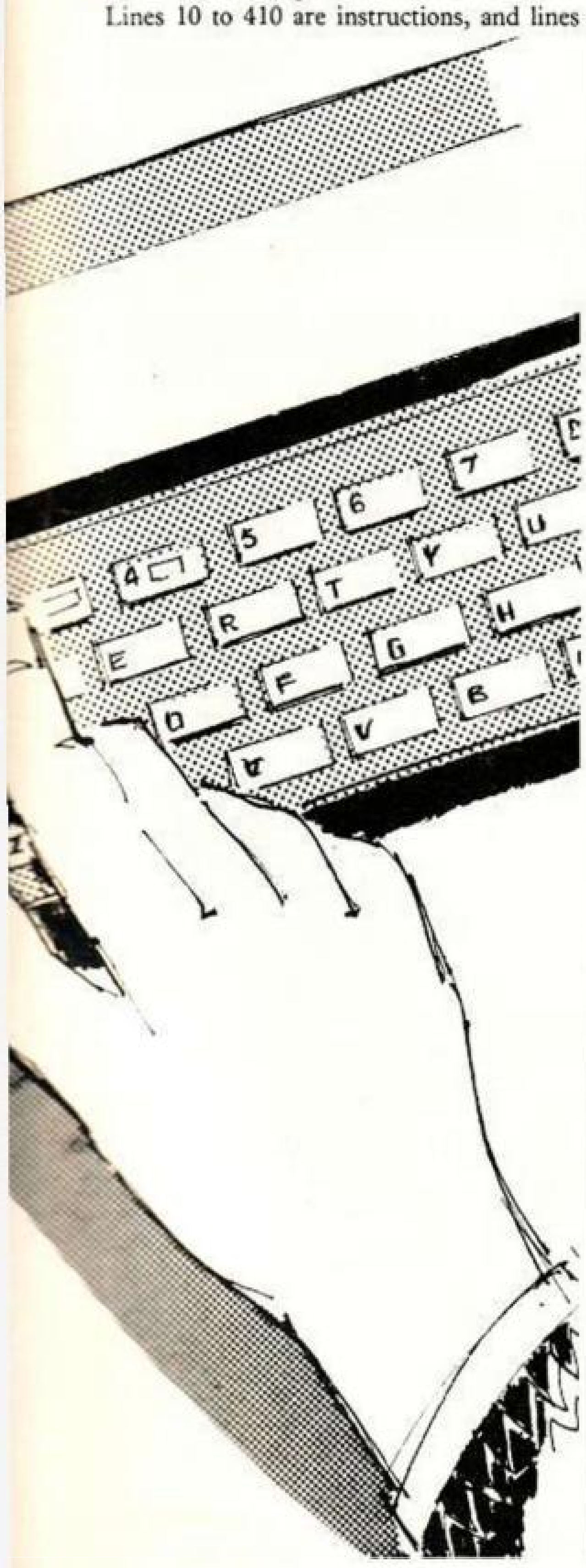
The corner squares are identified using the same system, except that the co-ordinates represent the position of the edge square to the left of — one position anticlockwise — the corner square. For example, if C (2,6) = 6, then that would mean that the corner square on the blue face — which was to the right of the edge square next to the orange face — was coloured orange — see figure 2.

Since moves must be made in many different parts of the program, it is obvious that you need a subroutine to tell the human what move to make, and to change the internal representation of the cube every times a move has been made. I have incorporated this in the program as subroutine 2000, and the move is transferred to it in the string variable X$.

I have used a string variable to save space, since it allows the whole move to be defined in one line; if two numerical variables had been used, one for the side and one to indicate clockwise or anticlockwise, two lines would be needed. Another subroutine is also required to execute a series of moves at the same time. This appears in the program as subroutine 3120, and the moves are also transferred to it in a string variable, Z$, for the same reasons as before. The subroutine divides this string into the individual moves, and executes them by placing them one by one into X$ and calling subroutine 2000.

To ensure that each move is as simple as possible for the person trying to solve the cube, only 12 different moves are used. They turn each of the sides either clockwise or anticlockwise through 90°.

Lines 10 to 410 are instructions, and lines 520 to 950 cover the initialisation of variables, and entry of the position. This is where the E and C arrays are set up. Also the variables R,B,W,G,Y and O are given the values 1 to 6 respectively. This was designed to make the program clearer and to save the user remembering which colour was represented by which number — whenever the number of a colour occurred, it could be replaced by one of these variables.



In the position-entering part of the program, line 620 onwards, subroutine 1000 is called to display a picture of four sides of the cube and then the colours of the eight squares around the centre square are entered. Lines 1000 to 1140 contain the subroutine to draw the cube. Lines 1005 and 1045 are contained as a subroutine because they are used later by subroutine 2000.

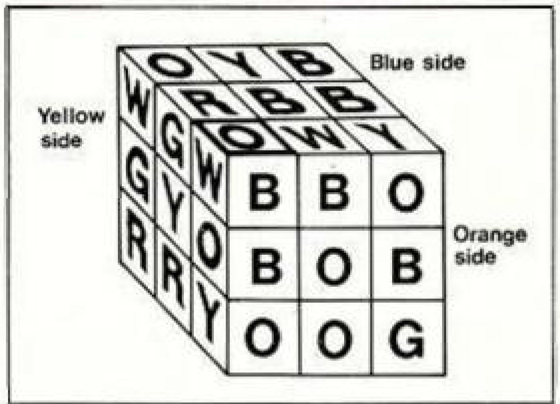


Figure 2. C(2,6)=6.

Here, the variables Top, Lef, Bot and Bac are set up for each side. I could have used nested loops to replace lines 1050 onwards, which would have saved a considerable amount of space, but I avoided them as they would have slowed down the subroutine considerably. Lines 1150 to 1160 hold the subroutine to print a colour.

The colour stored in the variable Colour is printed. Note the peculiar use of And in the Print statements. This is one of the useful idiosyncrasies of the ZX-81. Another example is the use of Val in the previous subroutine. These features increase the speed of the program and save space, but they make conversion for any other dialects of Basic very difficult. Lines 1300 to 1600 contain the subroutine to check the position. Lines 1310 to 1440 are the instructions for using this part of the program.

Inkey$ and Val are used in conjunction with subroutine 1000 to display diagrams of the sides chosen by the user. Lines 2000 to 2290 contain the subroutine at the heart of the program — it is called for the execution of every move made. Lines 2002 to 2006 are necessary because of the fact that calls are frequently made to the subroutine when it is not necessary for a move to be made. The variable X$ is set to the null string in the line 2004 to avoid a move being executed twice by mistake.

Lines 2020 and 2075 transfer the information stored in X$ into two variables, Sid, which has a value of 1 to 6 and represents the side to be turned, and X$ is changed to A or C to represent whether the move should be anticlockwise or clockwise.

Lines 2090 to 2114 are required while the program is trying to complete the top side. If a move to the top side is made, the values of the front, right, back and left sides will change. This is best understood if you consider turning the top side as turning the rest of the cube in the opposite direction. The colours of the front, right, back and left sides will then change accordingly.

Lines 2120 to 2215 control the printing of the move. Print Tab 31; statements are used after Scroll statements to fill the display with spaces. If they were omitted, CLS would be slowed down considerably, since it would have to lengthen the display file again, moving up the part of the memory where the variables are stored.

Lines 2218 to 2290 execute the move in memory — that is, they change the internal representation of the cube. Subroutine 1005 is used to set the variables Top, Bot, Lef, Bac and Rig. Then the painstaking process of moving each square begins.

Initialisation of variables is catered for in lines 2400 to 2500. This is the start of the formula. The variable Move, which counts the number of moves made, is set to Zero. X$ is set to the null string in case it is not set to anything else before subroutine 2000 is called for the first time. TS and TC, the top side and the top colour, are set to 1 since they remain red throughout the program, and US and UC are set to 6, orange, in the same way. Finally the front side and the front colour — *FS* and FC — are set to 2, blue, as the computer will start on the top edge next to the blue edge.

Lines 2505 to 2900 deal with the top-side edges. Lines 2510 to 2570 test to see if any of the top-side edges are already in their correct positions, and ED, which counts the number of edge squares done, is set up. Subroutine

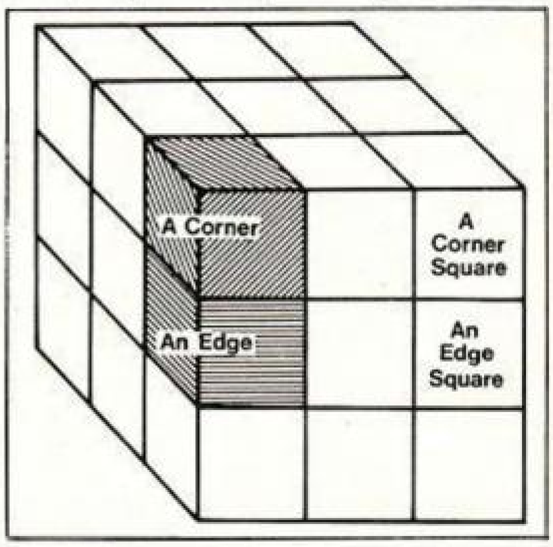


Figure 3.

2589 is used several times during the program, as it works out the left, back and right sides and colours from the front side and the front colour. Lines 2640 to 2850 deal with the actual moves involved, and the rest of this section controls looping back to the beginning of the section until all the top-side edges have been done.

Middle-layer edges are calculated in lines 3370 to 3680. At the beginning of this section two variables are set up, R$ and L$. They are strings of moves which are used extensively in this section in conjunction with subroutines 3120. M1, the number of middle-layer edges in the correct position, is set up, and if all the edges are correctly placed, a jump is made to line 3690. Otherwise, the edges are put in

(continued on next page)

(continued from previous page)

their correct places by lines 3140 to 3680.

Lines 3690 to 4260 deal with bottom-side corners. This section both puts the corners in their correct positions and turns them round the right way. It does this mainly by using the strings of moves A$ and B$.

Bottom-side edges are covered in lines 4270 to 5150. Four strings of moves are used here, A$, B$, R$ and L$, which are, as before, set up at the beginning of the section. It completes the cube in two stages. First, it puts all the bottom-side edges so that they are orange down, and then it puts them in their correct positions. This second stage is relatively short — it only goes from line 5000 to 5150, but it takes the majority of the moves in this section.

By lines 8000 to 8090, the cube should be finished. If it is not, it is not because of computer error, but human error. This program can solve the cube from any position which can be reached without dismantling the cube. First you are asked to type Newline so that the screen will not be cleared before you have done the last move. It then gives you the chance of trying the program again by typing Newline. Otherwise the program has finished.

The shortage of memory on the ZX-81 has forced me to omit the full range of error-checking routines which could be provided. If you accidentally enter the wrong colours when setting up the cube, the program will crash and you will have to start again. The only other problem you might face is that a number of the pirate cubes on the market have their colours on different sides.

To use this program on the 8K RAM ZX-80 requires a few minor alterations:

```
240 PAUSE 4E4
241 POKE 16437, 255
242 IF INKEY$<>"C" THEN GOTO 240
410 PAUSE 4E4
411 POKE 16437, 255
412 IF INKEY$<>"S" THEN GOTO 410
795 PAUSE 4E4
797 POKE 16437, 255
    DELETE 800
805 IF X$ = "" THEN GOTO 795
1450 PAUSE 4E4
1451 POKE 16437, 255
1452 LET K$ = INKEY$
2125 PAUSE 4E4
2126 POKE 16437, 255
2140 IF K$<>"H" AND K$<>CHR$ 118
     THEN GOTO 2125
8010 PAUSE 4E4
8011 POKE 16437, 255
8012 IF INKEY$<>CHR$ 118 THEN GOTO 8010
8075 PAUSE 4E4
8076 POKE 16437, 255
8090 GOTO 8075
```

Conversion for any other machine would be very difficult, because of the extensive use of Val throughout the program since on the ZX-81, Val may be used to evaluate variables as well as numbers.

Variables used in the first half of the program — up to line 1600.

E (6,6)	An array holding the edge squares.
C (6,6)	An array holding the corner squares.
I, J	Loop control variables.
C$	String containing the initials of the colours used in cube-drawing subroutine 1000.
R	Red, set to 1.
B	Blue, set to 2.
W	White, set to 3.
G	Green, set to 4.
Y	Yellow, set to 5.
O	Orange, set to 6.
S	The side being entered.
COLOUR	The colour to be printed by subroutine 1150.
D	Line number for Print At Expression.
A	Column number for Print At expression.
SID	The side next to the square being entered.
X$	The key being typed, used during the position entering section.
C	The number of the square being entered.
TOP	Top side
LEF	Left side
RIG	Right side
BOT	Bottom side
BAC	Back side
K$	The key being typed in subroutine 1300, the position-checking subroutine.

Variables — used in the second half of the program — line 2000 onwards

X$	The move to be made by subroutine 2000. Used in subroutine 2000 to show whether the move is clockwise or anti-clockwise.
SID	The side to be turned in subroutine 2000.
XX	Dummy variable used to hold the temporary value of colours during the execution of a move.
FS	Front side.
LS	Left side.
BS	Back side.
RS	Right side.
TS	Top side.
US	Under side.
FC	Front colour.
LC	Left colour.
BC	Back colour.
RC	Right colour.
TC	Top colour.
UC	Under colour.
MOVE	The number of moves executed so far.
K$	The key being typed in subroutine 2000.
I,J,K,L	Loop control variables.
ED	The number of top-side edges done so far.
Z$	The string of moves to be executed by subroutine 3120.
CO	The number of top-side corners done so far.
R$, L$	Strings of moves used to position the middle edges.
M1	Number of middle edges done so far.
A$, B$	Strings of moves used to position the bottom-side corners.
S$	Representation of two sides in the form of a string.
S1, S2	The numerical equivalents of the sides stored in S$.
X	Flag to indicate whether or not to loop back to line 3982 at line 3963.
R$, L$	Strings of moves used to position the bottom-side corners.
FL	The number of corner squares or edge squares on the bottom-side which are orange.
A$, B$, R$, L$.	Strings of moves used to position the bottom-side edges.

Table 1. The program variables.

```
1   REM CUBE MASTER
2   REM C SIMON LANE OCT 1981
3   REM REQUIRES 16K RAM ZX81
5   SLOW
6   RAND
10  REM INSTRUCTIONS
15  CLS
20  PRINT TAB 10; "CUBEMASTER"
30  PRINT TAB 10; "          "
40  PRINT
50  PRINT " THIS PROGRAM CONSISTS OF TWO "
60  PRINT "PARTS, ONE TO ENTER THE POSITION"
70  PRINT "AND ONE TO ACTUALLY "" DO THE"
80  PRINT "CUBE""."
90  PRINT "WHILE YOU ARE ENTERING THE"
100 PRINT "POSITION YOU WILL HAVE A DIAGRAM"
110 PRINT "TO SHOW YOU WHICH COLOUR TO"
120 PRINT "ENTER. TYPE IN THE FIRST LETTER"
130 PRINT "(R,B,W,G,Y OR O) OF THE COLOUR"
140 PRINT "IN THE POSITION OF THE ""?"", AND"
150 PRINT "TYPE ""0"" TO BACKSPACE."
160 PRINT "IN THE SECOND PART OF THE"
170 PRINT "PROGRAM YOU WILL BE TOLD TO MOVE"
180 PRINT "ONE OF THE SIX SIDES EITHER"
190 PRINT "CLOCKWISE OR ANTICLOCKWISE. ALL"
200 PRINT "TURNS ARE THROUGH NINETY"
210 PRINT "DEGREES."
220 PRINT
230 PRINT " TYPE ""C"" TO CONTINUE."
240 IF INKEY# <> "C" THEN GOTO 240
245 CLS
250 PRINT TAB 10; "CUBE MASTER"
260 PRINT TAB 10; "           "
270 PRINT
280 PRINT " AFTER YOU HAVE MADE THE MOVE"
290 PRINT " WAIT UNTIL THE TYPE "NEW LINE""
300 PRINT " MESSAGE APPEARS. YOU MAY THEN"
310 PRINT "TYPE ""NEW LINE"" TO GET THE NEXT"
320 PRINT "MOVE, OR TYPE ""H"" TO SEE THE"
330 PRINT "CURRENT POSITION"
340 PRINT " NB - ALL REFERENCES TO SIDES"
350 PRINT "MEAN THE SIDE WITH THE STATED"
360 PRINT "COLOUR IN THE CENTRE E.G. THE"
370 PRINT "RED SIDE IS THE SIDE WITH THE"
380 PRINT "RED SQUARE IN ITS CENTRE."
390 PRINT
400 PRINT "TYPE ""S"" TO START."
410 IF INKEY#<>"S" THEN GOTO 410
520 REM ENTERING POSITION
530 DIM E(6,6)
540 DIM C(6,6)
542 FOR I=1 TO 6
544 FOR J=1 TO 6
545 LET E(I,J)=7
546 LET C(I,J)=7
547 NEXT J
548 NEXT I
550 LET C#="RBWGYO"
560 LET R=1
570 LET B=2
580 LET W=3
590 LET G=4
600 LET Y=5
610 LET O=6
620 FOR S=1 TO 6
625 CLS
630 GOSUB 1000
640 PRINT
650 PRINT " HOLD THE CUBE WITH THE ";
660 LET COLOUR=S
670 GOSUB 1150
675 PRINT
```

```
680 PRINT "SIDE TOWARDS YOU, AND THE ";
690 LET COLOUR=TOP
700 GOSUB 1150
705 PRINT
710 PRINT "SIDE FACING UPWARDS"
770 FOR C=1 TO 8
772 LET D=VAL "33455543"(C)
774 LET A=VAL "45554333"(C)
780 LET SID=VAL "TOP TOP RIG RIG BOT BOT LEF LEF"
(C*4-3 TO C*4-1)
790 PRINT AT D,A;"?"
795 LET X$=INKEY$
797 IF X$<>"" THEN GOTO 815
800 PRINT AT D,A;" "
802 LET X$=INKEY$
805 IF X$="" THEN GOTO 790
820 IF X$="R" OR X$="B" OR X$="W" OR X$="G" OR X$="Y"
OR X$="O" THEN GOTO 860
830 IF X$<> "0" OR C=1 THEN GOTO 800
835 PRINT AT D,A;" "
840 LET C=C-1
845 GOTO 772
860 PRINT AT D,A;C$(VAL X$)
870 IF C/2<> INT (C/2) THEN GOTO 900
880 LET C(S,SID)=VAL X$
890 GOTO 910
900 LET E(S,SID)=VAL X$
910 NEXT C
920 NEXT S
930 CLS
940 PRINT "PLEASE WAIT."
950 GOTO 2400
1000 REM SUB TO DRAW CUBE
1001 GOSUB 1005
1002 GOTO 1050
1005 LET TOP=VAL "GRRRRB" (S)
1010 LET RIG=VAL "WWGYBW" (S)
1020 LET LEF=VAL "YYBWGY" (S)
1030 LET BOT=VAL "BOOOOG" (S)
1040 LET BAC=VAL "OGYBWR" (S)
1045 RETURN
1050 PRINT "   ";C$(C(TOP,LEF));C$(E(TOP,BAC));C$(C(TOP,BAC))
1060 PRINT "   ";C$(E(TOP,LEF));C$(TOP);C$(E(TOP,RIG))
1070 PRINT "   ";C$(C(TOP,S));C$(E(TOP,S));C$(C(TOP,RIG))
1080 PRINT C$(C(LEF,BAC));C$(E(LEF,TOP));C$(C(LEF,TOP));C$(C(S,LEF));
C$(E(S,TOP));C$(C(S,TOP));C$(C(RIG,S));C$(E(RIG,TOP));C$(C(RIG,TOP))
1090 PRINT C$(E(LEF,BAC));C$(LEF);C$(E(LEF,S));C$(E(S,LEF));C$(S);
C$(E(S,RIG));C$(E(RIG,S));C$(RIG);C$(E(RIG,BAC))
1100 PRINT C$(C(LEF,BOT));C$(E(LEF,BOT));C$(C(LEF,S));C$(C(S,BOT));
C$(E(S,BOT));C$(C(S,RIG));C$(C(RIG,BOT));C$(E(RIG,BOT));C$(C(RIG,BAC))
1110 PRINT"   ";C$(C(BOT,LEF));C$(E(BOT,S));C$(C(BOT,S))
1120 PRINT"   ";C$(E(BOT,LEF));C$(BOT);C$(E(BOT,RIG))
1130 PRINT"   ";C$(C(BOT,BAC));C$(E(BOT,BAC));C$(C(BOT,RIG))
1140 RETURN
1150 REM SUB TO PRINT COLOUR
1155 PRINT "RED" AND COLOUR=1;"BLUE" AND COLOUR=2;"WHITE" AND
COLOUR = 3; "GREEN" AND COLOUR=4; "YELLOW" AND COLOUR=5; "ORANGE"
AND COLOUR = 6;
1160 RETURN
1300 REM SUB TO CHECK POSITION
1305 CLS
1310 PRINT " CUBE-CHECK -"
1320 PRINT
1330 PRINT "TO SEE A SIDE, TYPE THE INITIAL"
1340 PRINT "LETTER OF THE COLOUR OF THE SIDE"
1350 PRINT "YOU WISH TO SEE, FOR EXAMPLE, TO"
1360 PRINT "SEE THE RED SIDE, TYPE R. THEN"
1370 PRINT "TO SEE ANOTHER SIDE, JUST TYPE"
1380 PRINT " ANOTHER LETTER"
1390 PRINT " IF YOU ARE SATISFIED THAT THE"
1400 PRINT " POSITION IS CORRECT, AND YOU"
1410 PRINT " WANT TO CARRY ON, TYPE ""NEW"
1420 PRINT " LINE"", BUT IF YOU WANT TO START"
1430 PRINT " AGAIN, TYPE ""0"", PRESS THE"
1440 PRINT " RELEVANT KEY NOW."
1450 LET K$=INKEY$
1460 IF K$="" THEN GOTO 1450
1470 IF K$=CHR$ 118 THEN RETURN

1480 IF K$="0" THEN RUN
1490 IF K$<>"R" AND K$<>"B" AND K$<>"W" AND K$<>"G" AND K$<>"Y" AND
K$<>"O" THENGOTO 1450
1500 LET S=VAL K$
1510 CLS
1520 GOSUB 1000
1530 PRINT
1540 LET COLOUR=S
1550 GOSUB 1150
1560 PRINT " SIDE FACING YOU, ";
1570 LET COLOUR=TOP
1580 GOSUB 1150
1585 PRINT
1590 PRINT "SIDE FACING UPWARDS."
1600 GOTO 1450
2000 REM MOVE MAKING SUB
2002 IF X$<>"" THEN GOTO 2010
2004 LET X$=""
2006 RETURN
2020 LET SID=VAL X$(1 TO 2)
2075 LET X$=X$(4)
2085 IF SID<>TS THEN GOTO 2120
2090 IF X$="A" THEN GOTO 2110
2100 LET XX=FS
2101 LET FS=LS
2102 LET LS=BS
2103 LET BS=RS
2104 LET RS=XX
2105 GOTO 2120
2110 LET XX=FS
2111 LET FS=RS
2112 LET RS=BS
2113 LET BS=LS
2114 LET LS=XX
2120 LET MOVE=MOVE+1
2121 IF MOVE=1 THEN GOTO 2160
2122 PRINT "TYPE "NEW LINE" "
2130 LET K$=INKEY$
2140 IF K$<>"N" AND K$<>CHR$ 118 THEN GOTO 2130
2150 IF K$="N" THEN GOSUB 1300
2160 SCROLL
2165 PRINT TAB 31;
2170 SCROLL
2175 PRINT TAB 31;AT 21,0;"MOVE";MOVE;". TURN THE ";
2180 LET COLOUR=SID
2185 GOSUB 1150
2190 PRINT " SIDE"
2195 SCROLL
2200 IF X$="A" THEN GOTO 2215
2205 PRINT TAB 31;AT 21,0;"CLOCKWISE.      ";
2210 GOTO 2218
2215 PRINT TAB 31;AT 21,0;"ANTICLOCKWISE.  ";
2218 LET S=SID
2219 GOSUB 1005
2220 IF X$="A" THEN GOTO 2250
2221 LET XX=E(S,TOP)
2222 LET E(S,TOP)=E(S,LEF)
2223 LET E(S,LEF)=E(S,BOT)
2224 LET E(S,BOT)=E(S,RIG)
2225 LET E(S,RIG)=XX
2226 LET XX=E(TOP,S)
2227 LET E(TOP,S)=E(LEF,S)
2228 LET E(LEF,S)=E(BOT,S)
2229 LET E(BOT,S)=E(RIG,S)
2230 LET E(RIG,S)=XX
2231 LET XX=C(S,TOP)
2232 LET C(S,TOP)=C(S,LEF)
2233 LET C(S,LEF)=C(S,BOT)
2234 LET C(S,BOT)=C(S,RIG)
2235 LET C(S,RIG)=XX
2236 LET XX=C(TOP,S)
2237 LET C(TOP,S)=C(LEF,S)
2238 LET C(LEF,S)=C(BOT,S)
2239 LET C(BOT,S)=C(RIG,S)
2240 LET C(RIG,S)=XX
2241 LET XX=C(TOP,RIG)
2242 LET C(TOP,RIG)=C(LEF,TOP)
2243 LET C(LEF,TOP)=C(BOT,LEF)
2244 LET C(BOT,LEF)=C(RIG,BOT)
2245 LET C(RIG,BOT)=XX
2246 GOTO 2280
2250 LET XX=E(S,TOP)
2251 LET E(S,TOP)=E(S,RIG)
2252 LET E(S,RIG)=E(S,BOT)
2253 LET E(S,BOT)=E(S,LEF)
2254 LET E(S,LEF)=XX
2255 LET XX=E(TOP,S)
2256 LET E(TOP,S)=E(RIG,S)
2257 LET E(RIG,S)=E(BOT,S)
2258 LET E(BOT,S)=E(LEF,S)
2259 LET E(LEF,S)=XX
2260 LET XX=C(S,TOP)
2261 LET C(S,TOP)=C(S,RIG)
2262 LET C(S,RIG)=C(S,BOT)
2263 LET C(S,BOT)=C(S,LEF)
2264 LET C(S,LEF)=XX
2265 LET XX=C(TOP,S)
2266 LET C(TOP,S)=C(RIG,S)
2267 LET C(RIG,S)=C(BOT,S)
2268 LET C(BOT,S)=C(LEF,S)
2269 LET C(LEF,S)=XX
2270 LET XX=C(TOP,RIG)
2271 LET C(TOP,RIG)=C(RIG,BOT)
2272 LET C(RIG,BOT)=C(BOT,LEF)
2273 LET C(BOT,LEF)=C(LEF,TOP)
2274 LET C(LEF,TOP)=XX
2280 LET X$=""
2290 RETURN
2400 REM START OF FORMULA
2410 LET MOVE=0
2430 LET X$=""
2450 LET TS=1
2460 LET TC=1
2470 LET US=6
2480 LET UC=6
2490 LET FS=2
2500 LET FC=2
2505 REM TOP EDGES
2510 FOR I=2 TO 5
2520 IF E(TS,I)=TC THEN GOTO 2550
2530 NEXT I
2540 GOTO 2570
2550 LET FS=I
2560 LET FC=E(FS,TS)
2570 LET ED=0
2580 GOSUB 2589
2581 GOTO 2640
2589 LET LC=VAL " 5234 "(FC)
2590 LET LS=VAL " 5234 "(FS)
2600 LET RC=VAL " 3452 "(FC)
2610 LET RS=VAL " 3452 "(FS)
2620 LET BC=VAL " 4523 "(FC)
2630 LET BS=VAL " 4523 "(FS)
2635 RETURN
2640 IF E(FS,TS)=FC AND E(TS,FS)=TC THEN GOTO 2860
2650 IF E(BS,TS)=TC AND E(TS,BS)=FC OR E(BS,TS)=FC AND E(TS,BS)=TC THEN
LET X$="BS A"
2660 GOSUB 2000
2670 IF E(FS,US)=TC AND E(US,FS)=FC OR E(BS,US)=TC AND E(US,BS)=FC OR
E(BS,US)=FC AND E(US,BS)=TC THEN LET X$="US A"
2680 IF E(BS,RS)=FC AND E(RS,BS)=TC THEN LET X$="TS A"
2685 IF E(BS,LS)=FC AND E(LS,BS)=TC THEN LET X$="TS C"
2690 IF E(RS,TS)=TC AND E(TS,RS)=FC OR E(RS,TS)=FC AND E(TS,RS)=TC THEN
LET X$="RS A"
2695 IF E(FS,TS)=TC AND E(TS,FS)=FC THEN LET X$="FS C"
2700 IF E(LS,TS)=TC AND E(TS,LS)=FC OR E(LS,TS)=FC AND E(TS,LS)=TC THEN
LET X$="LS C"
2710 GOSUB 2000
2720 IF E(US,RS)=TC AND E(RS,US)=FC THEN LET X$="US A"
2730 IF E(US,LS)=TC AND E(LS,US)=FC THEN LET X$="US C"

2740 IF E(US,RS)=FC AND E(RS,US)=TC THEN LET X$="FS C"
2750 IF E(US,LS)=FC AND E(LS,US)=TC THEN LET X$="LS A"
2760 IF E(FS,RS)=TC AND E(RS,FS)=FC OR E(BS,RS)=TC AND
E(RS,BS) = FC THEN LET X$="TS A"
2770 IF E(FS,LS)=TC AND E(LS,FS)=FC OR E(BS,LS)=TC AND E
(LS,BS)= FC THEN LET X$="TS C"
2780 GOSUB 2000
2790 IF E(FS,US)=FC AND E(US,FS)=TC THEN LET X$="FS A"
2800 GOSUB 2000
2810 IF E(FS,RS)=FC AND E(RS,FS)=TC THEN LET X$="FS A"
2820 IF E(FS,LS)=FC AND E(LS,FS)=TC THEN LET X$="FS C"
2830 GOSUB 2000
2840 IF E(BS,LS)=TC AND E(LS,BS)=LC THEN LET X$="LS C"
2842 GOSUB 2000
2845 IF E(BS,RS)=TC AND E(RS,BS)=RC THEN LET X$="RS A"
2850 GOSUB 2000
2860 LET ED=ED+1
2870 IF ED=4 THEN GOTO 2910
```

(continued on page 21)

the vic centre

Adda Computers Ltd., a major supplier of computer systems to industry and business, have opened the Vic Centre in West London. Here you can see, discuss and buy everything to do with the new VIC 20 personal computer—in person or by mail. Hardware, software, technical advice and information is available from an experienced staff. Even if you already own a VIC 20, get on our mailing list to know about new developments. Remember—everything has the backing of Adda's reputation, and there's a full 12-month warranty on all hardware. The Vic Centre is easy to reach—Just off the A40, close to North Acton tube station.

Not just a computer but a whole expandable system

AT ONLY £189·95 inc.VAT. Special cassette deck £44·95 inc.VAT.

The VIC-20 is a fully fledged, easy-to-use computer. It's the core of a great expandable system.

- ★ EXPANDABLE MEMORY—UP TO 32K, USING PLUG-IN MODULES
- ★ DISK DRIVE/CASSETTE—FOR EXTERNAL STORAGE.
- ★ PRINTER—80 COLUMN, 30 CHARACTERS-PER-SECOND

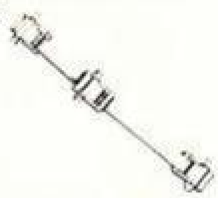
First time users can operate it immediately with plug-in program cartridges, and using your own colour T.V. to get up to 24 colours on screen, four different sound tones and even write your own programs in BASIC. The VIC-20 lets you build a system as needs and budget dictate, so that your VIC-20 can be more than just a personal computer.

VIC-MEMORY 3K £44.85
Small size—low cost memory expansion. Plugs into Vic and reproduces memory-port. Can be used with other expansions gives a total of 6k user static ramon Vic.
FEATURE This board allows Vic to move Basic to begin at 1024 ($0400) as in Pet, and enables the use of HIGH RESOLUTION COLOUR GRAPHICS

VIC-LIGHT PEN £28.75
This high quality light pen works in both normal and Hi-Res modes on the Vic allowing simple interaction with the Vic without keyboard entry.
Easy to program and easy to use.
e.g. Menu selection. Non-keyboard entry. Teaching Games.
FEATURE touch sensitive "Enter" contacts to eliminate accidental entry.

VIC-RS232 INTERFACE £56.35
Fully implemented (true levels) RS232C-V24
BI-DIRECTIONAL INTERFACE
Allows Vic to work as Mainframe Terminal Drive a Qume Daisywheel or a Paper Tape Punch etc. etc.
FEATURE This unit contains master power supply which supports Vic's own supply when carrying Memory Expansions, Cassette Drives, Light Pens, Printers etc.

VIC JOYSTICK Single £14.95
Hand-Held joystick units for games use available in Pair or Single configuration. N.B. (2 Singles will not work as a pair unless modified)



VIC-Games Port Adaptor Cable £19.85
A two into one adaptor for use with both joysticks and light pens. A must for those who require full control of games with graphics.
FEATURE low-cost High quality. Robust.

VIC-20 DOT MATRIX PRINTER Tractor feed, 80 character-per-line, 30 characters-per-second.	£229.95
VIC SINGLE FLOPPY DISK DRIVE	£ P.O.A.
3K RAM CARTRIDGE	£29.95
8K RAM CARTRIDGE	£44.95
16K RAM CARTRIDGE	£74.95
PROGRAMMERS AID CARTRIDGE —an extension of BASIC to aid programming and de-bugging.	£34.95
MACHINE CODE MONITOR CARTRIDGE —includes assembler and dis-assembler capabilities.	£34.95
SUPER EXPANDER HIGH RESOLUT. CARTRIDGE —permits use of high resolution graphics.	£34.95

VIC software Each of these tapes £14.95

Codebreaker/Codemaker
You play the VIC or the VIC plays you in this computerised version of Mastermind.

VIC Seawolf, VIC Trap and Bounce-out
3 fun games, a submarine shoot out, a beat the VIC and an old favourite pub game.Good games with different skill levels.

Monster Maze and Maths Hurdler
A fun game with good colour and sound and a mental arithmetic learning game. Highly rated by everyone we have shown it to. Harder than you think.

Canyon Fighter, Tunesmith, Star Wolf at £5.95 each

VIC GAMES CARTRIDGES at £19.95 each

Fruit Machine, Lander, Road Rally, Alien, Avengers, Poker

TERMS AND CONDITIONS: All goods sold subject to Adda terms and conditions of sale. Full details available on request, but include: 7 day money back guarantee. Adda 12 month hardware warranty. Please allow 21 days for delivery. Allow 7 days for personal cheques to be cleared. **Quoted prices are inclusive of VAT.**

Goods Required ____________ Price ____

Add £2.00 p. & p. for orders under £50.00 Total £ ______

Name: ______________________

Address: ______________________

SHOP ADDRESS: Adda Home Computers Ltd. 154, Victoria Road, Acton, London, W3. Tel 01-992 9904.
OPEN: 10 am—6 pm (Tuesday—Friday), 10 am—5 pm (Saturday).

YRC4

MAIL ORDER to: Adda Computers Limited, FREEPOST, London, W13 0BR or telephone your order (24 hours a day) to 01-992 9904 quoting your BARCLAYCARD OR ACCESS number.

* I enclose a cheque, made payable to Adda Computers Limited for
£ ....
* Please charge my Barclay/Access account. My account number is
....
* Please add my name to your mailing list
* Delete as applicable
Date: ....

adda

(continued from page 19)

```
2880 LET FC=VAL " 3452 " (FC)
2890 LET FS=VAL " 3452 " (FS)
2895 GOSUB 2589
2900 GOTO 2560
2910 REM TOP CORNERS
2915 LET Z$=""
2920 LET CO=0
2930 IF FC=VAL " 4523 " (FS) THEN LET X$="TS C"
2940 GOSUB 2000
2950 IF FC=VAL " 3452 " (FS) THEN LET X$="TS A"
2960 IF FC=VAL " 5234 " (FS) THEN LET X$="TS C"
2970 GOSUB 2000
2980 IF C(FS,TS) =TC AND C(RS,FS)=FC THEN LET Z$="RS A US C
RS C"
2990 IF C(FS,TS)=FC AND C(TS,RS)=TC THEN GOTO 3320
3000 IF C(TS,RS)=FC AND C(RS,FS)=TC THEN LET Z$="FS C US A FS A"
3010 IF C(RS,TS)=TC AND C(BS,RS)=FC THEN LET Z$="RS C US A US A RS A"
3020 IF C(TS,BS)=TC AND C(RS,TS)=FC THEN LET Z$="BS A US A BS C"
3030 IF C(TS,BS)=FC AND C(BS,RS)=TC THEN LET Z$="FS C BS A US A FS A BS
C"
3040 IF C(LS,BS)=FC AND C(BS,TS)=TC THEN LET Z$="BS C RS A US C US C RS
C BS A"
3050 IF C(TS,LS)=TC AND C(BS,TS)=FC THEN LET Z$="LS A RS A US C US C
RS C LS C"
3060 IF C(TS,LS)=FC AND C(LS,BS)=TC THEN LET Z$="LS A FS C US C US C
FS A LS C"
3070 IF C(LS,TS)=TC AND C(FS,LS)=FC THEN LET Z$="RS A LS C US C LS A
RS C"
3080 IF C(TS,FS)=TC AND C(LS,TS)=FS THEN LET Z$="LS C US C LS A"
3090 IF C(FS,LS)=TC AND C(TS,FS)=FC THEN LET Z$="FS A US C US C FS C"
3100 GOSUB 3120
3110 GOTO 3170
3120 FOR K=1 TO LEN Z$ STEP 5
3130 LET X$=Z$(K TO K+3)
3140 GOSUB 2000
3150 NEXT K
3155 LET Z$=""
3160 RETURN
3170 IF C(US,RS)=TC AND C(BS,US)=FC THEN LET X$="US C"
3180 IF C(US,LS)=TC AND C(FS,US)=FC THEN LET X$="US A"
3190 GOSUB 2000
3200 IF C(US,BS)=TC AND C(LS,US)=FC THEN LET Z$="FS C US C FS A"
3210 IF C(US,FS)=TC AND C(RS,US)=FC THEN LET Z$="RS A US C RS C"
3220 GOSUB 3120
3230 IF C(FS,RS)=TC AND C(US,FS)=FC THEN LET Z$="FS C US C FS A"
3240 IF C(RS,US)=TC AND C(FS,RS)=FC THEN LET Z$="RS A US A RS C"
3250 IF C(RS,BS)=TC AND C(US,RS)=FC THEN LET Z$="US A FS C US C FS A"
3260 IF C(BS,US)=TC AND C(RS,BS)=FC THEN LET Z$="FS C US A FS A"
3270 IF C(BS,LS)=TC AND C(US,BS)=FC THEN LET Z$="RS A US A US A RS C"
3280 IF C(LS,US)=TC AND C(BS,LS)=FC THEN LET Z$="FS C US C US C FS A"
3290 IF C(LS,FS)=TC AND C(US,LS)=FC THEN LET Z$="RS A US C RS C"
3300 IF C(FS,US)=TC AND C(LS,FS)=FC THEN LET Z$="US C RS A US A RS C"
3310 GOSUB 3120
3320 LET CO=CO+1
3330 IF CO=4 THEN GOTO 3370
3340 LET FC=VAL " 3452 "(FC)
3350 LET FS=VAL " 3452 "(FS)
3355 GOSUB 2589
3360 GOTO 2980
3370 REM MIDDLE EDGES
3380 LET R$="FS C US A FS A US A RS A US C RS C"
3390 LET L$="FS A US C FS C US C LS C US A LS A"
3400 LET MI=(E(W,G)=W AND E(G,W)=G)+(E(G,Y)=G AND E(Y,G)=Y)+(E(Y,B)=Y
AND E(B,Y)=B)+(E(B,W)=B AND E(W,B)=W)
3405 IF MI=4 THEN GOTO 3690
3410 FOR K=2 TO 5
3420 IF E(US,K)<>UC AND E(K,US)<>UC THEN GOTO 3450
3430 NEXT K
3440 GOTO 3590
3450 LET FC=E(US,K)
3460 LET FS=FC
3470 GOSUB 2589
3480 IF K=FS THEN LET X$="US C"
3485 IF K=FS THEN LET K=RS
3490 GOSUB 2000
3500 IF K=RS THEN LET X$="US C"
3510 IF K=LS THEN LET X$="US A"
3520 GOSUB 2000
3530 IF E(BS,US)=RC THEN LET Z$=R$
3540 IF E(BS,US)=LC THEN LET Z$=L$
3550 GOSUB 3120
3560 LET MI=MI+1
3570 IF MI=4 THEN GOTO 3690
3580 GOTO 3410
3590 FOR K=2 TO 5
3600 LET FC=K
3610 LET FS=K
3620 GOSUB 2589
3630 IF E(FS,RS)<>FC OR E(RS,FS)<>RC THEN GOTO 3660
3640 NEXT K
3650 STOP
3660 LET Z$=R$
3670 GOSUB 3120
3680 GOTO 3410
3690 REM BOTTOM CORNERS
3740 LET A$="RS A US A RS C FS C US A FS A RS A US C RS C"
3750 LET B$="RS A US A RS C FS C US A US A FS A RS A US C RS C"
3760 FOR K=2 TO 5
3770 LET FS=K
3780 LET FC=K
3785 GOSUB 2589
3790 FOR L=1 TO 4
3800 LET S$="LSFSFSRSRSBSBSLS"(L*4-3 TO L*4)
3810 GOSUB 3850
3820 IF X=1 THEN GOTO 3900
3830 NEXT L
3840 STOP
3850 LET S1=VAL S$(1 TO 2)
3860 LET S2=VAL S$(3 TO 4)
3870 LET X = 0
3880 IF C(LS,FS)=S1 AND C(FS,US)=S2 OR C(FS,US)=S1 AND C(US,LS)=S2
OR C(US,LS)=S1 AND C(LS,FS)=S2 THEN LET X=1
3890 RETURN
3891 LET S1=VAL S$ (1 TO 2)
3892 LET S2=VAL S$ (3 TO 4)
3893 LET X=0
3894 IF C(FS,RS)=S1 AND C(RS,US)=S2 OR C(RS,US)=S1 AND C(US,FS)=S2
OR C(US,FS)=S1 AND C(FS,RS)=S2 THEN LET X=1
3895 RETURN
3900 LET S$="FSRSRSBSBSLSLSFS" (L*4-3 TO L*4)
3910 GOSUB 3891
3920 IF X=1 THEN GOTO 3959
3930 NEXT K
3940 LET Z$=B$
3950 GOTO 3970
3959 LET FS=VAL " 3452 " (FS)
3960 LET FC=FS
3961 GOSUB 2589
3962 GOSUB 3851
3963 IF X=1 THEN GOTO 3982
3964 LET FS=VAL " 5234 " (FS)
3965 LET FC=FS
3966 GOSUB 2589
3969 LET Z$=A$
3970 GOSUB 3120
3982 LET R$="RS A US A RS C US A RS A US C US C RS C"
3984 LET L$="LS C US C LS A US C LS C US A US A LS A"
3990 LET FL=(C(US,FS)=UC)+(C(US,RS)=UC)+(C(US,BS)=UC)+(C(US,LS)=UC)
3995 IF FL=4 THEN GOTO 4220
3997 IF FL=1 THEN GOTO 4100
4000 LET FS=VAL " 3452 " (FS)
4010 LET FC=FS
4020 GOSUB 2589
4030 IF C(FS,US)=UC AND C(FS,RS)=UC AND C(US,BS)=UC AND C(US,RS)=
UC THEN LET Z$=R$
4040 IF C(FS,US)=UC AND C(US,FS)=UC AND C(BS,LS)=UC AND C(US,RS)=UC
THEN LET Z$=R$
4050 IF C(LS,FS)=UC AND C(RS,US)=UC AND C(LS,US)=UC AND C(RS,BS)=UC
THEN LET Z$=R$
4060 IF C(LS,FS)=UC AND C(FS,RS)=UC AND C(LS,US)=UC AND C(BS,US)=UC
THEN LET Z$=R$
4070 IF C(FS,US)=UC AND C(US,FS)=UC AND C(US,BS)=UC AND C(RS,BS)=UC
THEN LET Z$=R$
4080 GOSUB 3120
4090 GOTO 3990
4100 FOR K=2 TO  5
4110 LET FC=K
4120 LET FS=K
4130 GOSUB 2589
4140 IF C(US,LS)=UC AND C(FS,RS)=UC THEN GOTO 4180
4150 IF C(US,FS)=UC AND C(FS,US)=UC THEN GOTO 4200
4160 NEXT K
4170 STOP
4180 LET Z$=R$
4190 GOTO 4210
4200 LET Z$=L$
4210 GOSUB 3120
4220 IF C(FS,RS) =BC THEN LET X$="US C"
4230 GOSUB 2000
4240 IF C(FS,RS) =RC THEN LET X$="US C"
4250 IF C(FS,RS) =LC THEN LET X$="US A"
4260 GOSUB 2000
4270 REM BOTTOM EDGES
4280 LET A$="LS A RS C FS C LS C RS A US A US A LS A RS C FS C L S C RS A"
4290 LET B$="LS A RS C FS A LS C RS A US C US C LS A RS C FS A L S C RS A"
4300 LET R$="RS A US A RS C US C LS C US C LS A US A US A RS A U S C RS C
US C LS C US A LS A"
4310 LET L$="LS C US C LS A US A RS A US A RS C US C US C LS C U S A LS A
US A RS A US C RS C"
4320 LET FL=(E(0, B)=0)+(E(0,W)=0)+(E(0,G)=0)+(E(0,Y)=0)
4330 IF FL=4 THEN GOTO 5000
4340 IF FL=0 THEN GOTO 4600
4350 IF (E(0,B)=0 OR E(0,G)=0) AND (E(0,Y)=0 OR E(0,W)=0) THEN GOTO 4440
4360 FOR K=2 TO 5
4370 IF E(K,US)=K  THEN GOTO 4390
4380 NEXT K
4385 LET K=2
4390 LET FC=K
4400 LET FS=K
4410 GOSUB 2589
4420 LET Z$=R$
4430 GOSUB 3120
4440 LET CO=2
4445 LET CO=CO-1
4450 FOR K=2 TO 5
4460 IF E(K,0)<>0  THEN GOTO 4480
4470 NEXT K
4475 GOTO 4445
4480 IF E(K,0)<>K  AND CO THEN GOTO 4470
4490 LET FC=K
4500 LET FS=K
4510 GOSUB 2589
4520 LET Z$=A$
4530 IF E(US,RS)=U C THEN LET Z$=B$
4540 GOSUB 3120
4550 GOTO 5000
4600 FOR K=2 TO 5
4610 IF E(0,K)=K THEN GOTO 4630
4620 NEXT K
4622 LET K=2
4630 LET Z$=A$
4640 LET FS=K
4650 LET FC=K
4660 GOSUB 2589
4670 IF E(US,LS)=BC  THEN LET Z$=B$
4680 GOSUB 3120
4690 GOTO 4440
5000 LET FL=(E(B,0) =B)+(E(W,0)=W)+(E(G,0)=G)+(E(Y,0)=Y)
5005 IF FL=4 THEN GOTO 8000
5009 FOR K=2 TO 5
5010 IF E(K,0)=K THEN GOTO 5100
5020 NEXT K
5030 LET Z$=R$
5040 GOSUB 3120
5050 GOTO 5000
5100 LET FS=K
5110 LET FC=K
5120 GOSUB 2589
5130 LET Z$=L$
5140 IF E(LS,US)=RC  THEN LET Z$=R$
5150 GOSUB 3120
8000 REM CUBE DONE
8005 PRINT "TYPE "NEW LINE""
8010 IF INKEY$<> CHR$ 118 THEN GOTO 8010
8020 CLS
8030 PRINT " YOU HAVE DONE THE RUBIC CUBE "
8040 PRINT " IF IT IS NOT DONE, THEN YOU HAVE"
8050 PRINT " MADE A MISTAKE. TO TRY AGAIN"
8060 PRINT " TYPE "" NEW LINE"", OTHERWISE TYPE"
8070 PRINT """SPACE"" "
8080 IF INKEY$=CHR$ 118 THEN RUN
8090 GOTO 8080
```

ZX81 owners

have you seen

The Cambridge Collection

A book of

30 PROGRAMS

For Only **£4.95**

NO MEMORY EXPANSION NEEDED

Each program has been designed to fit into 1K of RAM

TEACH YOURSELF PROGRAMMING

Comprehensive explanations of each listing will teach you many techniques of **ZX81** programming.

HOURS OF AMUSEMENT

With titles such as FORTRESS, BALLOON, and ODD MAN OUT, you could easily become a **ZX81** addict. Plus, entirely new implementations of well-known favourites; LUNAR LANDING, MASTER CODE, ORBITAL INVADERS, and many others.

CASSETTE AVAILABLE TOO!

If you order the book you can also buy the programs on a quality cassette for only £4.95 extra.

Please send me:

copies of the book at **£4.95** each

copies of the book and cassette at **£9.90** pair

Please send your orders with cheques/PO's to:
Richard Francis, Dept.YC A/S
22 Foxhollow, Barhill,
Cambridge, CB3 8EP.

---

io systems ltd.

A/D BOARD FOR NASCOM

- 8 input channels
- 30 miscrosec conversion
- Over voltage protection
- Prototyping area
- 8 bit resolution
- Sample and hold
- Full flat/interrupt control
- NASBUS compatible

Price **£135** + 15% VAT (post free)

GRAPHICS BOARD FOR NASCOM

- 384(H) × 256(V) high resolution graphics display
- Fully bit mapped
- Full software control
- Mixed text and graphics
- NASCOM 2 or 4MHz NASCOM 1
- Graphics software supplied

Price **£55** + 15% VAT (post free)

EPROM PROGRAMMER

- Programs 3 rail: 2708/2716

Single rail: 2508/2758, 2516/2716, 2532/2732

- Software supplied for Read/Program/Verify
- Can be used with other machines with 2 parallel ports

Price **£63** + 15% (post free)

DUNCAN

- Fast real time interpreter/control language for NASCOM 1 or 2 (please specify)

Price **£12** + 15% VAT (post free)

MEMORIES

- 4116-150ns **95p** each + 15% VAT (min order 8)
- 64K-200ns **£10** each + 15% VAT

MONITORS

- BMC 12" green phosphor — 18MHz

Price **£175** × 15% VAT (carriage paid)

6 Laleham Avenue, Mill Hill,
London NW7 3HL
Tel: 01-959 0106

Microgame Simulations

FREE! tape directory with each tape

ZX81 • Video Genie • TRS80/LII

Battle of Britain As strategic commander of nine British fighter squadrons you must track and intercept the devious enemy bomber squadrons before they reach London. Micro acts as a flight control and communications centre for patrols, missions, intelligence reports etc. Sketch map provided. May also be played on any map of S.E. England with standard grid.

Asset Stripper Compete against your computer controlled arch-rival "KO Investments", capitalizing companies on the stock market and bidding for takeovers to gain control of lucrative assets. Can your micro really outwit you?

Kingdom of Nam As ruler of Nam you must control its economy; allocating labour; building cities, factories and ships; importing and exporting; negotiating pay claims and fighting the looming threats of inflation, strikes, starvation, overpopulation and revolution. How long can you stay in power?

High Stakes Buy, sell, train and bet on racehorses. An opportunity to make money shrewdly and spend it recklessly. Scrolling racing commentary. One or two players.

Party Bran Tub Lots of programs — including one of the best message scrollers around. Play your computer at noughts and crosses and analyse each game afterwards if you wish! Guessing games — words/numbers/rhymes etc. (many <1K)

Also **Invaders** (VG/TRS) and **Wordgram**

All programs in BASIC (for 16K) on quality cassettes.
Send £4.50 for 1/£8 any 2/£11.50:3/£15:4/£18.50:5/£22:6

Please tell us which micro and send cheque/PO to:
73 The Broadway, Grantchester, Cambridge CB3 9NQ

Metal cased 9"

CROFTON MONITOR

10 MHZ Bandwidth
P4 Standard
£59.50 + VAT (£68.42)
plus carriage £3.00
For P31 (green) high resolution tube add £12.50 + VAT (£14.38)

NEW• Plastic cased 12" PRINCE MONITOR

High resolution
24 MHZ Bandwidth
P31 (green) Standard
for only
£78.00 + VAT
(£89.70) plus
carriage £3.00

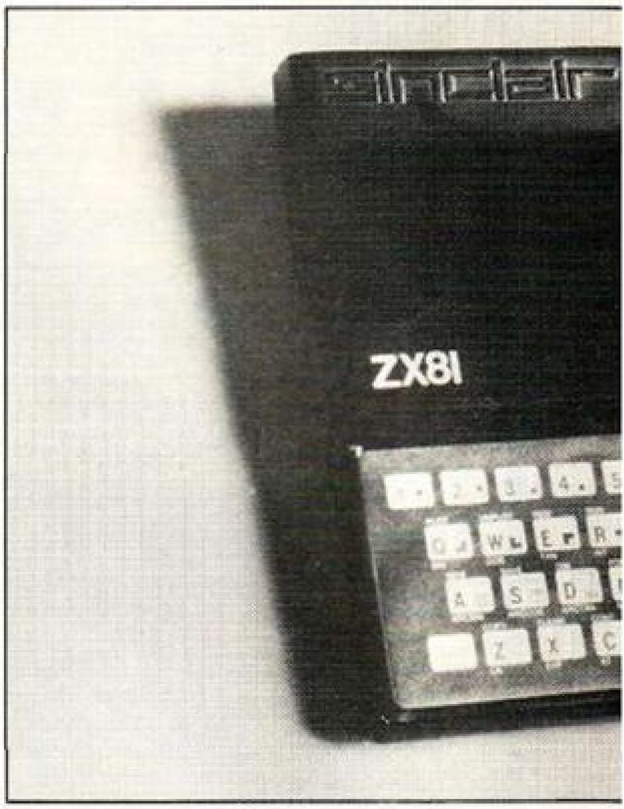
ALL MAJOR CREDIT CARDS ACCEPTED — Small surcharge

CROFTON ELECTRONICS LTD
35 Grosvenor Road, Twickenham, Middx TW1 4AD
01-891 1923/1513

SURVEY

ZX CASSETTES

Carried along in the wake of worldwide ZX-81 sales that have just cruised past the 250,000 marker, software for the Sinclair machine has been forced to change dramatically in quality and volume since the last *Your Computer* survey in October 1981. Eric Deeson tests the latest releases.



THE BUOYANCY of the ZX-81 market has meant that the program manufacturers have found little time to concoct anything new for the considerable following still commanded by the 4K ZX-80. More surprisingly, perhaps, Sinclair Research has released no new packages for either micro.

Let us start by looking at the new games. Their two most glaring failings are a poor choice of keys to use for movement and firing, and no automatic restart when you have lost or won.

In his marvellous book, *Mastering machine code on the ZX-81,* Tony Baker tells us how to program so that whole sectors of the keyboard can act as a single key. Such an approach has yet to be used in commercial ZX-81 software — a shame since it must be better than the finger-contorting demanded by some products.

Non-restarting games — like any non-auto-run program — require the player to have some computing knowledge. If he does not, he can too easily be faced by a disconcerting listing, or might even wipe a line by mistake. Program users should need to know nothing of programming.

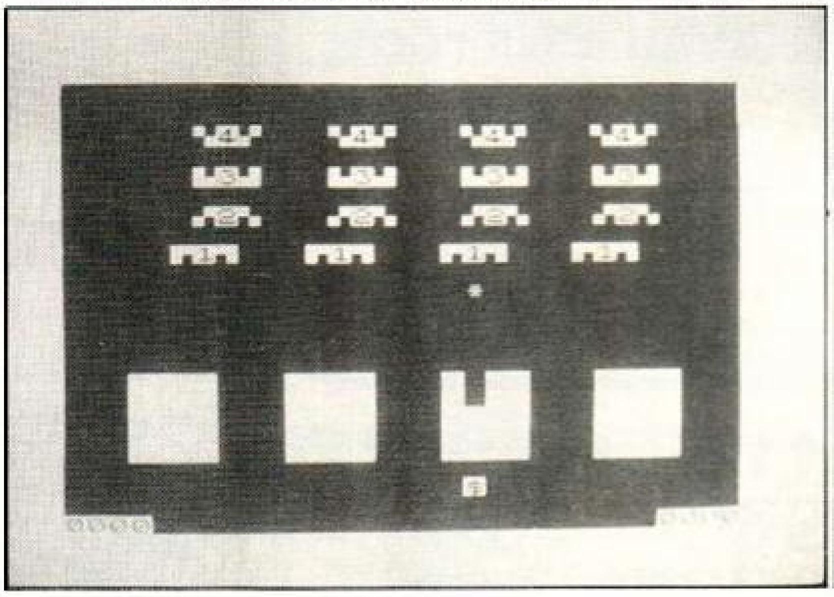
Orwin's Cassette One, £3.80, has been widely advertised as the cassette you should buy first. It contains a miscellany of 11 old and new 1K games, mostly in machine code which means they are faster and more spectacular. The documentation is adequate, and loading provided no problem. The games are not bad given that they are for 1K and provide plenty of variety. They are good value and include several arcade-style programs, including a version of Space Invaders, a maze, I Ching, a lander, Mastermind and an impressive hangman.

Tasteless offerings

Greye's Gamestape 1 offers the Orwin cassette good competition. It costs £2.95 but contains only 10 programs which on the whole are not so good as the Orwin games. Bumper, Lander, Simon, Mastermind, Asteroids, Hangman — guillotine to be precise. There is also a brilliant etch-a-sketch with eight directions and Copy, and an attractive pattern-generator.

I hesitate to mention the only other collection of 1K games, Adult Games — eight from the aptly-named Can of Worms for £3. Accompanied by good notes on the cassette-case slip and thought-provoking audio commentary, they are candid and must suffice for readers with bad taste until something more revolting appears. They feature acne, vasectomy, Hitler, Reagan and Royal Flush.

However, I find Micro-Gen's New York, £4.90, really offensive. I object to conflict games — why should we exterminate all those



pleasant extra-terrestrials, submarines and cars? This program beats them all — you have engine failure over the Big Apple and to avoid hitting the skyscrapers, you must bomb them out of your path. Fortunately, the review copy was a pre-production draft and would not load.

Micro-Gen redeems itself, however, with its excellent Chess for £9.90. I did not think I was too bad at the game, but this program beat me on the lowest of the six levels in just 18 moves. If you are an aficionado, you may dislike its use of grid co-ordinates rather than conventional symbols for moves — but it has a satisfactory board, handles all legal moves and rejects in a flash attempts to cheat. The only shortcoming was that it thinks and plays in Fast mode, so you have to check carefully to see what it has done.

Video Software, cited in the October survey for its several extremely good "serious" programs, now offers games, still maintaining high quality of programming and presentation — useful documentation, audio commentary and good program instructions. All these programs sell at £3.95.

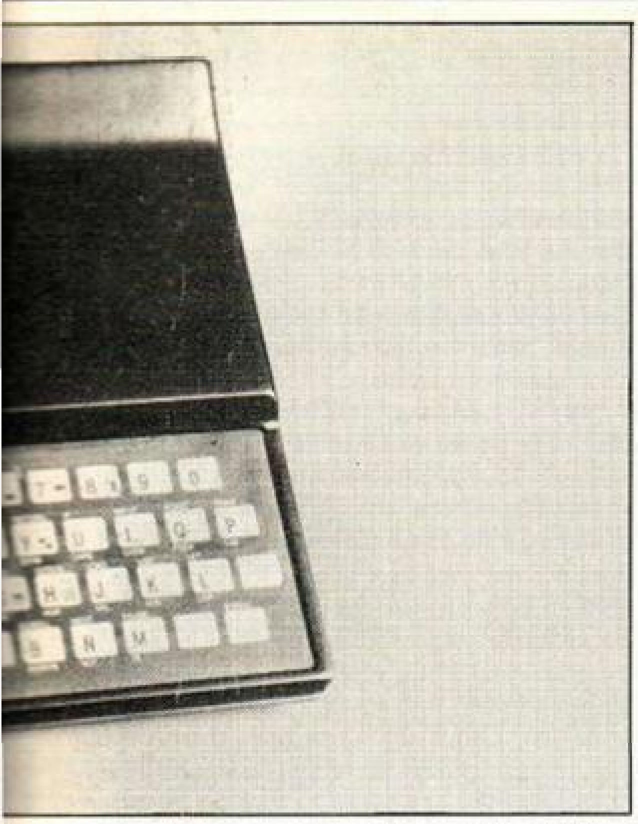
Force Field is the nearest to an arcade game. It is not very exciting, but is nonetheless addictive. You defend your city from a series of bombs by activating the "force-field" at just the right moments.

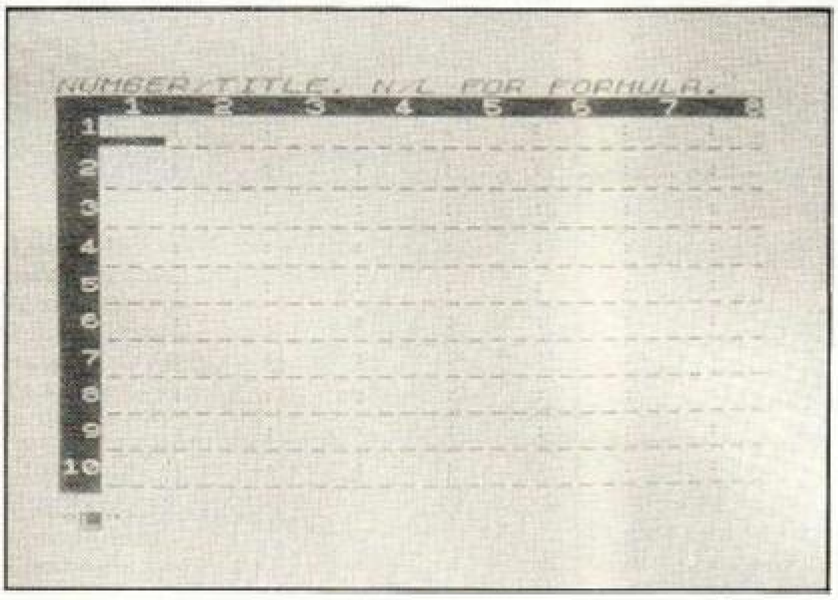
Space Race is, claims Video Software, a success at parties — especially those where guests are keen on building space stations. Football and Test Match are both serious simulations of lengthy sporting occasions. In each, you set up teams, conditions, attributes, run the programs and interrupt as much as you want to investigate situations of your invention.

Stock Market involves interaction with up to four players — so make sure your RAM packs are firmly in place. You have six commodities, news flashes, fluctuating prices and buy or sell at each turn. A clever, but perhaps not sufficiently flexible, computerised board game.

Macronics is well known in the ZX software market with a good range of generally excellent programs. It offers four 16K games. First is its long-standing — but still number one — version of Intruders at £6, or £5 for the 1K version. By Atari standards, it is slow and lacks graphics quality — yet if you are an addict, this version will keep you happy for a long time.

Nightmare Park, £3.75, is a good maze game — novel to the point of impenetrability. It is a sort of adventure where you cannot avoid your fate. StarTrek is a Basic version of the standard game. The documentation and screen messages are inadequate for a newcomer like me. Finally we have Dragon Maze, £6.95 — a complex,





full-screen, invisible maze containing an ever-nearing dragon. You are shown the maze at the beginning and the easiest way to win is to use Copy-Contd.

Hilderbay is relatively new to the ZX scene — but proudly proclaims that the company's experience is with mainframe computers and emphasises its professional approach. There are two games to look at — the very impressive Gold and the make-weight Pick A Word. The two are supplied on one cassette for £8. Gold is a version of adventure set in the Yukon. It is beautifully designed and definitely addictive for those with that kind of mind. Technically, I was much impressed by the ultra-slow rate of printing, giving an entirely appropriate Teletype appearance.

Educational drills

Pick A Word may be a make-weight but it, too, is novel and clever. This neat word-game transforms the ZX-81 into a very sharp opponent. The only failing is that the diabolically well-designed word lists are too few in number for a long game.

Next, we have Space Battle from Green. This game is a relatively slow and monotonous shadow of an arcade game. It is written in Basic and includes simple instructions.

Defender from Quicksilva is the fastest and most sophisticated moving-graphics program I have seen for the ZX-81, and even claims to produce sound effects using the Quicksilva sound board. It is the ZX-81 game on which I have spent more precious time than any other — £5.50 with full screen display.

There are two commercial educational software suppliers now in the ZX-81 market. AVC Software supplies a range of drill programs in 16K. The Hangperson range is based on a hangman routine far more advanced than any I have seen — even if it is in Sinclair low-resolultion graphics. Each program calls words or phrases at random from a pool of 50 on the theme in question. A graphic or verbal clue appears in a box and the game commences. This approach is currently embodied in programs for primary school geography and English, and secondary school physics and biology.

Also at £3, AVC has an impressive graphics program called Angle for helping to teach the use of a protractor; a versatile graph-plotting program; and Tables Count-down, a variable-level, tables-testing program where each correct answer moves the user closer to the launch of a "rocket". Display format and simple language are very important in teaching software; clearly, both have received a good deal of attention in these programs.

The Hodthorpe Collection from Stan Spencer is a little like Hangperson in the sense that a good range of programs surround the same neat skeleton. The Collection is actually a thin book of 60 pages in which the six programs are fully described and listed. We deal with it here because there is also a cassette containing programs.

Adapted from Nascom 2 software, the skeleton allows the user to enter his own test material. The tests can then be administered, marked and recorded, all without the teacher's intervention. The idea is very good though not entirely novel. These six variations are concerned with flash-cards, picture-based quizzes, multiple-choice tests, two kinds of arithmetic testing and primary school language work.

Let us now look at the various new ZX-81 programs in the data-handling, administrative and financial categories. Again, they can be relevant to the owner who may wish to consider computerising his recipes, budgets or work records.

In the data-handling class, we have two files routines and two versions of the very popular commercial program VisiCalc. Multifile from Bug-Byte is a most versatile package with a good range of clear, menu-driven options. The files are not fixed, but user-definable.

While Multifile is in Basic, Database at £10 from Campbell Software is in machine code. Each record is entered into a string whose length is not fixed. Clear menus and sub-menus then allow suitable entries and data-handling. The fields have the fixed titles of Name, Address, Interest code and Commentary.

One of the two VisiCalc versions is from Video Software. Called Video-Plan, £7.95, it is rather stolid and perhaps a touch slow. All the same, it is beautifully documented and laid out, and very easy and foolproof in use. It is supplied with sample data and a thoughtful audio commentary.

Financial programs

The other version is Computacalc from Silicon Tricks. It has more spark than Video Software's version, but is still to be fully debugged and documented. Both programs do the job well once you have developed a feel for them — the approach gives businessmen, and others, a very powerful office tool.

Hilderbay has developed several serious commercial programs for the ZX-81. Finance is the least exciting — but also the least expensive at £8. The cassette contains three standard programs — Loan, Mortgage, and VAT. Loan requires three of four parameters of compound interest — sum, interest, number of payments, amount of payments, and gives you the fourth.

Mortgage deals with building society and bank loans and points out unheaded tables of data. It LPrints, too, though the literature does not say so. This program's main failing is its poor error-trapping.

VAT calculates from input data tables of price, total, tax and rate which is easily variable.

Stock control, at £25, is really rather expensive. It is well documented but uninspiring. The cassette has two versions, one allowing many lines to be handled with little detail, the other the reverse.

Critical-path analysis, £15, on the other hand, is cheap, lively and very effective. Just the job for, say, a self-employed builder or the computing and business departments of schools and colleges. You need to do a good deal of preparation, of course, but this program does the donkey-work of finding and displaying the critical path.

Finally, from this supplier, we have Payroll — £25 and £2 for the excellent manual. With its help, the small-business user can handle the pay records of up to 30 employees. If you have more, you are not small, and can afford more than 16K memory.

PAYE, from Stroud, Litt and Co, £2.95, is really for the individual trying to make sense of his payslip rather than the employer trying to make payslips in the first place. Various 1K and 16K programs are on this tape and you select one from them according to your ZX-81's memory and your tax liabilities.

To close this section we should mention Hewson's Stats which is a reasonable cassette for 1K and £2.95. There are three straightforward, but nonetheless useful, programs here — Chi-square; Graph plot; Statistics.

(continued on next page)

(continued from previous page)

Let us now turn to the utilities programs, and two aids to graphics production. Multigraphics, £3.50 from Bridge Software, is remarkable value. The firm also has a version of Invaders with a good reputation. Multigraphics is a massive product — considerably more than 10K — which means that its usefulness is rather restricted unless you have extra-large memory. The initial menu, however, offers nine main options — print current display on screen; LPrint it; Save it; CLS; inverse CLS; draw on screen; print with standard characters; large print; upper and lower case with the same options; and jumbo text — upper and lower case, and three sub-symbol options. It is most impressive, and has clear instructions.

Equally impressive, but in a very different direction, is the £5 High-Resolution Graphics from Macronics. This has been widely advertised with a picture of the Prince and Princess of Wales. That design is supplied for demonstration and it is brilliant. HRG allows you to print pixel by pixel on a 192-by-192 grid. The routine is tedious, but for special applications very worthwhile. For general applications it is limited in the time needed for you to program it, in the difficulty of mixing normal Print with it, in its being Print rather than Plot and the fact that it is not responsive to Copy. The instructions are reasonably detailed, however, and with a little effort you could enter it into a menu-driven set of subroutines.

Macronics also gives us Scroll, £3.95. Any lines of text entered scroll up the screen in large size, continuously but not altogether smoothly. The scroll speed is variable. Straightforward in use, this program, too, has much potential.

ZXAS is a powerful ZX-81 assembler from Bug-Byte which occupies 5K. This superb product accepts the standard Zilog mnemonics with decimal or hexadecimal values.

Life, a well-designed and valid version, and Music, with 1K one-octave and 16K five-octave versions, are both from Macronics and are written in machine code. Music outputs your symphonies to cassette and, less acceptably, to TV sound.

Finally, there are two cassettes developed from books. These cassettes save you the trouble of keying programs, but the problem is that the book tends to be written without the cassette in mind. As a result, though you need the book to explain what the program is, you cannot guarantee that the book will provide clear user instructions.

This is particularly true with the cassette that camp-follows Randle Hurley's *The Sinclair ZX-81*, Macmillan. That cassette, £11.44 from Globe Book Services, is very hard to work with because the book, although good, does not help the user at all.

Another such cassette is Phipps' ZX-81 pocket-book tape. This contains 22 1K programs and 12 for users with more memory; almost all the programs are from the book, but none is adequately explained there.

I remain concerned about documentation standards. We now have an impressive range of good ZX-81 software on the market. Too much of it is spoiled by the lack of a few lines of carefully-written guidance notes. With so many potential customers, software suppliers understandably think of cutting corners — but they will suffer in the long term if they do. ■

Supplier	Program name	Description	Assessment A	B	C	D	E	F	G	H
	Administration									
10	Stats	Three simple programs	3	4	3	3	3	3	2	1
11	Critical Path*	Finds path	4	5	4	4	5	3	—	4
	Stock Control*	Standard — two	4	5	4	3	4	3	—	3
	Data-Handling									
1	Videoplan*	VisiCalc version	4	4	4	4	4	4	3	4
4	Multifile*	Definable filing	3	2	3	3	4	4	1	2
5	Database*	File-handling	5	4	5	3	3	4	—	3
17	Computacalc*	VisiCalc version	2	5	4	3	4	4	4	4
	Education									
2	Electricity Hangperson*	CSE test game	4	5	5	5	4	4	5	5
	Angle*	Drill	4	5	4	4	4	4	4	5
	Graph*	Versatile plotting	5	5	4	4	4	4	3	4
	Tables Count-Down*	Test game	5	5	4	5	4	4	4	5
18	Hodthorpe Collection*	Various — six	4	5	3	2	2	2	1	4
	Finance									
11	Finance*	Three simple programs	2	5	3	3	4	3	—	3
	Payroll*	Up to 30 staff	5	3	3	4	4	4	—	4
19	PAYE (*)	Tax calculations	1	3	3	3	2	3	—	2
	Games									
1	Force Field*	Defend city	5	4	5	4	4	4	4	4
	Space Race*	Spectators only	5	5	4	—	2	4	4	4
	Football*	Simulate league	5	4	5	4	—	4	2	3
	Test Match*	Simulation	5	5	5	3	—	4	2	4
	Stock Market*	Board game	5	5	5	5	4	4	4	4
6	Adult Games	Various	4	5	2	2	—	3	3	5
8	Space Battle*	Arcade style	3	5	3	3	2	3	2	2
9	Games 1	Various — 10	3	5	3	4	3	3	2	2
11	Gold*	Adventure	—	4	4	4	4	4	—	4
	Pick A Word*	Word-game	—	4	4	3	3	3	—	4
12	Intruders (*)	Arcade	4	5	4	5	4	4	4	2
	Nightmare Park*	Maze adventure	3	5	5	5	4	4	3	4
	Star Trek*	Standard	1	5	2	1	3	4	2	2
	Dragon Maze*	Good maze	1	5	5	3	3	4	4	3
13	Chess*	Six levels	4	5	4	3	4	4	4	3
14	Cassette One	Various — 11	3	5	2	2	3	4	4	3
16	Defender*	Arcade	3	3	5	4	4	4	5	4
	Miscellaneous									
7	Sinclair ZX-81*	Various — 10	1	4	2	2	4	4	—	4
12	Music (*)	One or five octaves	3	5	3	3	3	4	—	3
	Life*	Standard	3	5	3	4	4	4	—	2
15	Pocket-Book*	Various — 34	2	4	3	3	4	4	3	4
	Data-Handling									
3	Multigraphics*	Large menus	4	3	4	4	5	4	4	5
4	ZXAS*	Assembler	3	3	—	3	5	4	—	5
12	High-Resolution*	192 by 192	3	5	—	2	5	5	—	5
	Scroll*	For display	2	5	3	5	4	3	—	4

Notes: **Supplier**; numbers refer to suppliers' list. **Program name**; an asterisk shows 16K is needed: an asterisk in brackets means that, in addition to the 1K program, a 16K version is available. **Assessment** on a 0-5 scale: A, documentation or instructions; B, ease of loading; C, format, or screen layout; D, ease of use by target; E, functional value; F, programming quality; G, quality of graphics if any; H, novelty.

Suppliers and addresses

1. **Video Software** Stone Lane, Kinver, Stourbridge, West Midlands: data-handling, training, games.
2. **AVC Software** PO Box, Harborne, Birmingham , 17: education.
3. **Bridge Software** 36, Fernwood, Marple Bridge, Stockport, Cheshire: graphics, games, statistics.
4. **Bug-Byte** 98-100, The Albany, Old Hall Street, Liverpool, 3: games, utilities, data-handling.
5. **Campbell Systems** 15, Rous Road, Buckhurst Hill, Essex: data-handling.
6. **Can of Worms** 65a, Osborne Road, Portsmouth: games.
7. **Globe Book Services** Canada Road, Byfleet, Surrey: cassette of the book.
8. **Green** 144, Pampisford Road, Purley, Surrey: games.
9. **Greye** 16, Park Street, Bath, Avon: games.
10. **Hewson** 7, Grahame Close, Blewsbury, Oxfordshire: games, statistics.
11. **Hilderbay** 8, Parkway, Regent's Park, London NW1: games, finance, commerce.
12. **Macronics** 26, Spiers Close, Knowle, West Midlands: games, utilities.
13. **Micro-Gen** 24, Agar Crescent, Bracknell, Berkshire: games, including chess.
14. **Orwin** 26, Brownlow Road, London NW10: games.
15. **Phipps** 3, Downs Avenue, Epsom, Surrey: cassette of the book.
16. **Quicksilva** 95, Upper Brownhill Road, Maybush, Southampton: games, utilities.
17. **Silicon Tricks** 2, Chichester Rents, Chancery Lane, London WC2: data-handling.
18. **Spencer** The Sycamores, Queens Road, Hodthorpe, Nottinghamshire: education.
19. **Stroud, Litt and Co.** 85, Jamestown Road, London NW1: finance.

ZX81 SOFTWARE FROM VIDEO SOFTWARE LTD 1K & 16K

16K SOFTWARE

VIDEO-PLAN (ZX81 only). Performs the functions of an analysis book. Arithmetic functions include addition, subtraction, multiplication. Printer optional.

VIDEO-AD. Rotating display of 16 pages of advertising material. Set-up your own pages and change them as and when required.

VIDEO-GRAPH. Planning and design aid. Create pictures/charts/graphs and store within the program. Save on cassette. Combine pictures like an 'identikit'.

VIDEO-VIEW. Do it yourself teletext. Create pages of data. Store them within the program. Save on cassette. View on demand.

VIDEO-MAP. (ZX81 only). Educational game based on maps. Navigate your plane to its destination. Bomb the target and return to base.

FORCE-FIELD. (ZX81 only). Animated bombardment game. You control the force-field which protects your city against hostile UFOs.

SPACE-RACE. (ZX81 only). Party game for eight players. Rockets race to build stations in space. Winners gradually take over losers until only one winner remains.

FOOTBALL-LEAGUE. Realistic simulation of an entire season. Every match played and results shown with progressive league table. You give teams ratings for skill, effort, etc.

TEST-MATCH. Realistic simulation of a test match series. Dynamic scoreboard. Give each player your own ratings for skills, technique, etc.

STOCK-MARKET. (ZX81 only). An exciting game of skill and judgement. Buy and sell stocks and shares as prices change in response to world events.

VIDEO-SKETCH. (ZX81 only). Move the cursor to any part of the screen. Draw or rub-out as you move it. Mix in text or graphics. Save picture in memory. Save picture sequence on cassette.

1K PARTY TRICKS (ZX81 only) NEW! NEW!

If you don't have a 16K RAM this set of programs is for you. Ten separate programs — some games, some more serious. All completely original, all ten programs included in the price.

SHOOT	Take the penalty and watch the goalie try to save.
SKETCH	Draw an almost full screen picture and save on cassette.
NAME THE DAY	Give the date, the ZX81 names the day of the week.
TRAIN	For the very young who would like to drive a train.
ONGER-WONGER	Watch the ZX81 draw its own pictures and yours.
WEATHER	An endless variety of completely inaccurate weather forecasts.
UFO	Shoot down the UFO before he gets you.
WHO SHOT JR	An intriguing test of your powers of detection.
FIELD-GUN	Can you hit the target.
FOLLOW THAT	Follow the path traced by the ZX81.

NOTE: These programs are not suitable for ZX80.

*** OUR NEW SOFTWARE SCHEME ***

SUPPORTED SOFTWARE. This is software written for the ZX81 by named authors and approved and marketed by ourselves. The main criterion for selection is that the quality of the program matches our existing products. These programs are fully supported by ourselves. Watch out for some very interesting products in the near future.

NEWS FOR USERS

Those of you who have already voted with your cheque books may be interested in our 'top ten'. These are the best sellers in November:

1. VIDEO-MAP
2. VIDEO-PLAN
3. VIDEO-VIEW
4. VIDEO-AD
5. VIDEO-GRAPH
6. FORCE-FIELD
7. STOCK-MARKET
8. PARTY TRICKS No. 1
9. FOOTBALL-LEAGUE
10. TEST-MATCH

Surprisingly some of the programs which we rate most highly are well down the list.

We expect our 1K programs to top this list soon. They are worth buying even if you have 16K.

PRICE LIST

ORDER CODE	*Manual*	*Std.*	*Lux*
SKETCH 81	—	7.95	9.95
PLAN 81	—	7.95	9.95
AD 81	—	7.95	9.95
GRAPH 81	—	5.95	7.95
VIEW 81	—	5.95	7.95
MAP 81	—	5.95	7.95
FORCE	—	3.95	5.95
SPACE	—	3.95	5.95
FOOT	—	3.95	5.95
TEST	—	3.95	5.95
STOCK	—	3.95	5.95
PARTY 1	1.95	4.95	6.95
PACK 1 (Plan + Ad)	—	—	17.95
PACK 2 (View + Graph)	—	—	13.95
GAMESET (Map + 5 Games)	—	—	19.95

Prices include VAT.
Mail order customers add £1.00 per order. (50p for manual only). Allow 28 days for delivery.

Std = cassette & manual. Lux = de luxe boxed sets.

VISIT US AT THE ZX MICROFAIR

Stone Lane,
Kinver,
Stourbridge,
West Midlands,
DY7 6EQ,
England.

Tel: Kinver 2462.
Std. 038-483-2462.

VIDEO SOFTWARE LIMITED

Directors: P. Smith, B.A., D. J. Smith (Director and Secretary). Registered Number: 1085829 England. V.A.T. Reg. No. 278 0605 51

Personal callers welcome during office hours.
Send s.a.e. for further details.

ORDER FORM

Code	M/S/L	Qty.	Price	Total
P. & P.				

See above for codes

Your cheque no. ....

NAME ....
ADDRESS ....
....
....

SCREENING BBC FINE DISPLAY OF

Jeremy Ruston runs through the BBC Micro's spectrum of colour and graphics facilities. His Basic routines provide the ground knowledge you need to write your own programs.

THE BBC MICROCOMPUTER has seven statements and functions associated with graphics. Here are three sample programs to demonstrate each command in use. The BBC has a total of eight screen modes, but graphics may be displayed in only five of them. Here are the five modes with their individual resolutions and colours:

Mode 0: 640 by 256; two colours
Mode 1: 320 by 256; four colours
Mode 2: 160 by 256; 16 colours
Mode 4: 320 by 256; two colours
Mode 5: 160 by 256; four colours

Regardless of the mode selected, the screen is defined to be 1,280 units wide and 1,024 units deep. This means that, unlike the Atom, you can write a program written for one mode on the screen and transfer it to another with very little modification. The colours available are shown in table 1. As you can see, to claim that the BBC Micro has 16 colours is somewhat of an exaggeration.

In the four- and two-colour modes one can alter the four colours to be anything in table 1, by executing VDU 19 (colour number), (actual colour number), 0,0,0. So, for example, VDU 19,2,6,0,0,0 will set colour two to be cyan, which is the sixth colour in the table list. Spectacular effects can be achieved by the use of the VDU 19 command when a program is running.

> Black, red, green, yellow, blue, magenta (blue-red), cyan (blue-green), white, flashing black-white, flashing red-cyan, flashing green-magenta, flashing yellow-blue, flashing blue-yellow, flashing magenta-green, flashing cyan-red, flashing white-black.

Table 1.

One graphics command not used in our selection of programs is Point which returns the colour of any point on the screen — program 1. This program draws those well-known string patterns by the simple expedient of having two balls bouncing around the screen, and joining them together with lines.

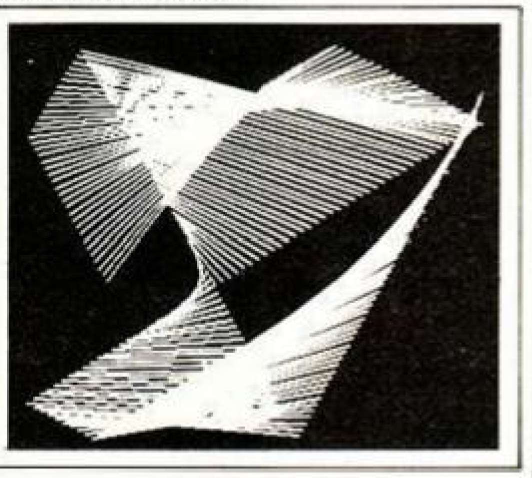
The command Mode 0 puts the machine in graphics mode 0. RND(1280) generates a random number in the range 1 to 1,280. Move moves the invisible graphics cursor to a specified position, from where the next line will be drawn using Draw.

After the program has drawn 150 lines on the screen, it waits for a key to be pressed, line 190, and then clears the screen and starts again. Repeat-Until is the same as the Atom's Do-Until construction — that is, it executes the code between the words Repeat and Until so that the condition after the Until statement is satisifed. As the condition in this case is never satisfied, it repeats for ever — see program 2.

Program 2 is executed in mode 2, the 16-colour mode. It changes the colour of the lines drawn whenever the pattern bounces off the edge of the screen by the use of the GCol statement. The GCol statement is followed by two numbers, the second of which specifies the current plotting colour; the first stimulates whether this colour is to be used as it stands or whether the commands And, Or, ExOr should be used on it or whether it should be even inverted with the colours on the screen at the time. One normally uses GCol 0,X to allow plotting in the colour you want, but the earlier number can be used to generate some spectacular effects.

Notice how few modifications need to be made to convert the program to colour because

A sample graphics routine, and right, the graphics which can be achieved.



Program 1.

```
10 REM "STRING ART BLACK AND WHITE"
20 REM (C) RUSTON 1981
30 REPEAT
40 MODE 0
50 X=RND(1280):Y=RND(1280)
60 L=RND(1280):M=RND(1024)
70 U=20-RND(40):V=20-RND(40)
80 P=20-RND(40):Q=20-RND(40)
90 FOR K=1 TO 150
100 MOVE X,Y
110 DRAW L,M
120 IF X+U>1279 OR X+U<0 U=-U
130 IF Y+V>1023 OR Y+V<0 V=-V
140 IF L+P>1279 OR L+P<0 P=-P
150 IF M+Q>1023 OR M+Q<0 Q=-Q
160 X=X+U:Y=Y+V
170 L=L+P:M=M+Q
180 NEXT K
190 A$=GET$
200 UNTIL FALSE
```

Program 3.

```
10 REM "TRIANGLES"
20 REM COPYRIGHT (C) RUSTON 1981
30 MODE 0
40 A=RND(1280):B=RND(1024)
50 C=RND(1280):D=RND(1024)
60 E=RND(1280):F=RND(1024)
70 U=20-RND(40):V=20-RND(40)
80 W=20-RND(40):X=20-RND(40)
90 Y=20-RND(40):Z=20-RND(40)
100 MOVE A,B:DRAW C,D:DRAW E,F
110 DRAW A,B
120 IF A+U>1279 OR A+U<0 THEN U=-U
130 IF B+V>1023 OR B+V<0 THEN V=-V
140 IF C+W>1279 OR C+W<0 THEN W=-W
150 IF D+X>1023 OR D+X<0 THEN X=-X
160 IF E+Y>1279 OR E+Y<0 THEN Y=-Y
170 IF F+Z>1023 OR F+Z<0 THEN Z=-Z
180 IF RND(10)>7 THEN VDU 19,1,RND(6)+1,0,0,0
190 PLOT 7,C,D:PLOT 7,E,F
200 PLOT 7,A,B
210 A=A+U:B=B+V:C=C+W:D=D+X:E=E+Y:F=F+Z
220 GOTO 100
```

MICRO'S COLOUR

Program 2.

```
10 REM "COLOUR STRING ART"
20 REM (C) RUSTON 1981
30 REPEAT
40 MODE 2
50 X=RND(1280):Y=RND(1280)
60 L=RND(1280):M=RND(1024)
70 U=20-RND(40):V=20-RND(40)
80 P=20-RND(40):Q=20-RND(40)
90 FOR K=1 TO 150
100 MOVE X,Y
110 DRAW L,M
120 IF X+U>1279 OR X+U<0 U=-U:GCOL 0,RND(6)+1
130 IF Y+V>1023 OR Y+V<0 V=-V:GCOL 0,RND(6)+1
140 IF L+P>1279 OR L+P<0 P=-P:GCOL 0,RND(6)+1
150 IF M+Q>1023 OR M+Q<0 Q=-Q:GCOL 0,RND(6)+1
160 X=X+U:Y=Y+V
170 L=L+P:M=M+Q
180 NEXT K
190 A$=GET$
200 UNTIL FALSE
```

of the way the axes are marked — see program 3. This program in effect draws three bouncing balls, all joined up by lines. Each triangle is erased as the next one is drawn. When run it looks as if the triangle is turning in three dimensions.

The program is run in mode 0, but as only one triangle is ever on the screen at a time, you can create the illusion of having 16 colours at 640-by-256 resolution by using the VDU 19,X,X,X,X to reset the current plotting colour to be any of the 16 possible colours. In fact I have only used the first eight colours, and have ensured that black is not used, giving seven in all.

Plot K,X,Y plots a point at X,Y in a manner determined by the value of K. K has the following effects linked to the following values:

0 Move relative to last point visited.
1 Draw line relative in the current graphics colour.
2 Draw line relative in the current logical inverse colour.
3 Draw line relative in the current graphics background colour.
4 Move to absolute position.
5 Draw line absolute in current graphics foreground colour.
6 Draw line absolute in current graphics logical inverse colour.
7 Draw line inverse in current graphics background colour.

For values 8-15 the last point in the line is omitted.
For values 16-23 a dotted line is drawn.
For values 24-31 a dotted line with the last point omitted is drawn.
For values 64-71 only a single point is plotted.
For values 80-87 a triangle is drawn, between X,Y and the last two points visited.

With the triangle graphics, one could alter the triangle program to plot filled triangles. The triangle commands can also let you draw most other polygons by drawing two triangles next to each other. ■

INTERVIEW

MICRO MARKET INSIDER

Kerr Borland, managing director of Arfon Microelectronics Ltd, is one of the key figures in the U.K. microcomputer industry. Together with John Marshall, he was the driving force behind the Nascom 1 — at one time the most popular single-board computer in Britain.

BORN IN EDINBURGH on March 5, 1944, Kerr Borland was educated at Berkhamsted school. After leaving school he did a number of odd jobs before joining Pan Books as a salesman.

"It was an exciting time. We were launching the James Bond series and books such as *Goldfinger* were selling as many as 60,000 copies a day. Unfortunately, it did not pay very well and, as I wanted to get married, I decided to do something else".

In 1964 Kerr Borland joined Sumlock, a leading calculator firm. There he learned the marketing skills which were to stand him in good stead when he launched his assault on the British microcomputer market. He also came into contact with the component side of the business which awakened his interest in microcomputers: "Micros were the coming thing in components".

In 1977 he was appointed U.K. managing director of North American Semi. In conjunction with John Marshall, he approached several people with the idea of a British microcomputer which could be built for less than £200. Despite being told it could not be done, Marshall and Borland persevered with the project. Eventually they contacted Chris Shelton of Shelton Instruments and commissioned him to put their ideas into practice.

The Nascom 1 was officially launched in January 1978, and immediately confounded the critics who thought it was doomed to failure. On the first day of the launch, Nascom received 300 orders for the Nascom 1, and the company was soon deluged in a further 7,000 enquiries.

"Nascom won because it was a kit and because it was very advanced", says Borland. "While other people were taking a steadier role, we took all the Z-80s and the 4118s and put them where nobody else would ever think of using them".

Nascom 1's success was also due to the price advantage it enjoyed over its competitors. The Nascom 1 cost less than £200 while its closest rivals, the Pet, Apple and TRS-80, were up to £400 more expensive.

Yet, while this price advantage boosted sales of the Nascom 1, it also cut Nascom's profit margins to the bone. Inadequate profit margins, increased competition and mounting research and development costs resulted in cash-flow problems.

"Nascom's greatest problem was that it was always too far ahead of itself", says Borland. "We experienced this incredible lag where we could never make enough money to finance the orders we had".

In April 1979, the company launched an upgraded Nascom 2. Based on the Z-80A processor, the Nascom 2 included 20K of on-board addressable memory consisting of 2K monitor, 1K video RAM, 1K user RAM, 8K Microsoft Basic and 8K static RAM. The new machine cost £295 plus VAT.

However, the Nascom 2 was entering a very different market to the Nascom 1. By this time Nascom's competitors had been forced to lower their prices, making the Nascom 2 just one of a number of microcomputers on the market in that price range.

The Nascom 2 was a technical success, but the company was suffering from increasing cash-flow difficulties. Finally, an official receiver was appointed and eventually Nascom was taken over by Lucas.

Kerr Borland has a great regard for the Lucas management, but he

'The BBC should have based the micro on the ZX-81'

does not think they will find the computing market easy to deal with: "I think that Nascom's new management has a great problem. Lucas is a superbly set-up company, but it is use to operating in an entirely different field. I feel tht Lucas has found itself in a retail market which it didn't really foresee.

"Nascom taught me a vast amount about the industry and its problems. To try and do it all yourself was really a very risky thing to do in an area where you have multi-billion dollar companies".

Kerr Borland left Nascom in 1979 to set up Product Launch, a promotional marketing company. Product Launch was designed to help form

– KERR BORLAND

companies for those people who needed specialist marketing skills but who did not want to be tied to long-term contracts.

"I left Nascom in 1979 to do something I had always wanted to do — run a marketing company. When I was with Nascom, I had noticed this huge market gap for people who wanted to start their own companies. You could never find really competent marketing people to help you over the initial hump if you only wanted them for three months. The marketing specialists wanted at least a year's contract".

Yet, although Product Launch proved a success, the lure of computers was too strong to be ignored. In 1980, Kerr Borland set up Specialist Micro Designs (SMD), a company to be subcontracted to design microcomputers.

"Specialist Micro Designs really grew from the friendships I had at the time. Also, when Nascom decided to call it a day, one or two of the engineers who were personal friends of mine asked to join SMD. It is now going very well and doing some extensive contracts for a number of clients".

At the end of 1980, Kerr Borland talked to some of the staff at SMD and suggested forming a company to manufacture peripherals for all the main microcomputers — the Apple, Pet, Tandy and Nascom. The result was Arfon Microcomputers Ltd.

"Arfon served two purposes", he says, "It put me where I wanted to go and it freed my SMD design programme. In a design company there are always goes and no-goes if you have more than one engineer".

Wanting to leave London and its attendant staffing problems, Kerr Borland visited a number of sites before plumping for Caernarfon, North Wales: "We looked at South Wales, but it seemed to me there was a danger of falling into a Silicon Valley-type trap where people go from one job to another looking for an extra £500".

Mid-Wales also failed to measure up to Arfon's requirements, but North Wales provided a number of ready-built factories which were eminently suitable: "We met with a very good reception in North Wales and were helped a great deal by the Welsh Development Agency who gave us a factory far bigger than we needed or wanted. But the factory, which was split into two, had the advantage of leaving us with space for expansion".

Arfon's range of products includes self-contained speech boards and light pens. The speech boards use a National Digitalker chip together with two 64K ROMs to generate speech. The first ROM set has a vocabulary of 256 words and sub-sounds, while additional sets of ROM allow the user to expand the vocabulary to his own design.

The light pens use a high-speed photodiode and work directly with the normally illuminated pixels. They can be used for editing, moving displayed data blocks and X Y plotting. The speech boards and the light pens are boxed with their own power supplies.

"By December we had probably sold around 500 speech boards, of one kind or another, and about 200 or 300 light pens", says Borland, "although we only really started going in August or September. The speech boards were successful because they hit a market area that interested all users. They also presented the hobbyist with a challenge, because he could use the digitised words to build up his own vocabulary".

Arfon's latest venture is a self-contained seven-cartridge expansion system for the Vic-20. The expansion system has its own power supply built around a toroidal transformer. All the ports are accessible and an optional lid provides a base for the TV.

'Nascom won because it was very advanced'

"It has proved to be an enormous success", he says. "In four weeks we sold nearly 2,000 units — mainly in the U.K.".

Future plans include the production of a whole range of cartridges such as the RS-232 and a printer designed specially for the Vic-20. In addition, Arfon plans to manufacture a disc controller, user port and software cartridges.

Looking back at the microcomputer boom of the late 1970s, Kerr Borland picked out two men as having played crucial roles — Kit Spencer and Chris Cary: "The most dominant man in Europe, without a doubt, was Kit Spencer. In between 1978 and 1980 he must have gained more than 50 percent of the sales market for Pet".

Kit Spencer, who masterminded the Pet's penetration of the U.K. microcomputer market, was appointed Commodore's European marketing director last year.

"On the trader market the great character was Chris Cary", says Borland. "People would complain about his discounting, but you have to remember that he would place orders and accept deliveries that added up to more than all the other distributors put together.

"It was like a gold-rush. Cary had them all discounting — people who would never have dreamt of dropping their prices were forced to discount just to survive. Some of the things they were giving away were unbelievable. They would be making £25 on the product and giving away £21 power supplies to sell it".

On a more current note, Kerr Borland is enthusiastic about the BBC computer-literacy series, but he has two reservations about the suitability of the BBC Microcomputer designed to accompany the programme.

"The BBC micro is a great piece of equipment", he says, "but it is entirely different to the original concept which was really the ZX-81. Personally, I would have preferred to see them do something based on the ZX-81, because then the hundreds of thousands of people watching the BBC programme could have afforded to buy a £71 micro. If you have a £200 or £300 micro, far fewer people will gain from the programme.

"The BBC Micro was designed by committee — I know because I was on it at one point. It reached the stage where there were 24 people sitting around a table and you can't design a product like that.

"Although the micro is a great idea, and very good in its own way, it is noticeable that in some areas the design is 24-man strong". ■

CHESS BOOK

The offered pawn in the Queen's Gambit may seem like easy pickings, but it will spell defeat time and time again for your micro if it is not primed to recognise this well-known snare. John White shows you how to program your machine for this and other book openings.

MANY COMMERCIAL games of strategy employ book-opening libraries in the early moves of the game. These are moves the manufacturer has pre-selected as particularly suitable for responding to moves by the opponent.

The advantage is that the book-opening move can be made almost at once, allowing more time for the program to consider its other moves. Secondly, it enables the program to make moves which long experience has taught are the best. Thirdly, it may enable the program to avoid opening traps.

Book openings are found in a variety of different games, but the most important example is that of chess, where literally hundreds of openings are known. A good example of an opening trap is the famous Queen's Gambit opening. After the moves d2-d4, d7-d5 and c2-c4, acceptance of the offered pawn, followed by a grim determination to cling on to it, nearly always spells disaster against accurate play by white. Yet it takes a deep search to see this, and all chess computers fall into the trap if they are not pre-programmed with a book opening which avoids it.

Have you ever wondered how book openings are added to a program of strategy? If you have, you probably assumed, as I did, that the book-opening library is constructed on the general principles of "If he does that, that I do this or this or this".

Let us consider what is required of a book-opening library. First, very few game positions are symmetrical in the sense that a book-opening library could be applied to play from both ends of the board. You need two libraries if you want to be able to play from both ends. For reasons of space, most commercial programs make do with a library which operates from one end of the board only.

Secondly, the program should be able to distinguish between several possible opponent responses which lead to different variations of the same opening, e.g., the Rauzer and Dragon variations of the Sicilian defence which diverge only at move 5 in the main lines.

Table 1.

Reference Number	Opening
0	Random selection by computer
1	Sicilian — Rauzer
2	Sicilian — Dragon
3	Ruy Lopez — Open
4	Giuoco Piano
5	King's Gambit
6	French — Winawer
7	Queen's Gambit Declined — Orthodox
8	Nimzo-Indian — Rubinstein
9	King's Indian — Classical
10	English — Symmetrical

Thirdly, it is desirable that the player should be able to select his own opening; failing this, an opening can be randomly selected by the program. Fourthly, the library should be able to prompt its opponent as to his continuation in the book.

The third and fourth features are unusual in games computers — again, because of the high cost of the extra memory needed.

Fifth, when a library is exhausted, or the opponent diverges from the library, the program must return to its own evaluation function. I soon found, by simple experiment, that construction of a simple tree of openings is tremendously wasteful of memory, and greatly retards the running of the program. The greater the tree, particularly for a deep opening of, say, eight moves, the slower the program runs. This becomes noticeable even at machine-code speeds. Yet commercial opening libraries give instant responses. Clearly, this was not the answer.

In the program presented here, all the machine moves are stored initially as four-figure strings in the two-dimensional array D$(MO,BO), where BO is the reference number of the opening and MO is the move number. The opponent responses are stored in the array C$(MO+1,BO) — MO+1 since one further move has been counted by the time the response is evaluated.

The principle is that the machine first chooses its opening number, BO, either randomly or by the opponent selecting it. It is then displayed. If no selection has been made, the response is matched against all similar responses and then the previous machine move, BQ, and the previous opponent move, BR$, are also checked to ensure that the matched moves have been derived from similar positions.

Duping the program

In theory, one could fool the program by arriving at the same consecutive three moves, on the same turns, from a different position. Yet this would be unlikely to arise by chance, and can be eliminated by careful construction of the library.

In this way, it is possible to switch from one opening to another, provided that both derive from the same original position. If the opponent has selected a book opening, then only the correct response for that opening is accepted. If no match is found at all, the counter BO is set to 0 and the program will henceforward ignore the library.

The advantages of this method is that, once

OPENINGS

the number of different openings has been fixed, the program always takes the same length of time to find each new move. The subroutine in lines 910-1170 contains the entire method.

The openings themselves are stored by reading a whole series of Data statements. The four-figure strings correspond to the array co-ordinates of the pieces on the board. For illustration, I have set up the data tables for 10 chess openings on a chessboard. The bottom-left square is labelled 1,1; the top-left is 1,8; the bottom-right is 8,1 and the top-right is 8,8. Thus a Data statement 7866 means: move from square 7,8 to square 6,6. This corresponds to a knight move from g8 to f6.

Data conversion

Castling is stored only as a king move. Extra routines recognise this and make the appropriate rook move. Various string-handling statements convert the raw numeric data into the usual algebraic notation.

Most of the rest of the program shows the results. The pieces of the board are represented most simply by ASCII codes corresponding to upper-case letters for black or lower-case for white. This is found in lines 150-210, and may need alteration according to your computer.

To maintain a static display — that is, one without scrolling — some cursor control is necessary, and I have used the standard codes:

[CLS] = CLEAR SCREEN
[HOME] = CURSOR HOME
[nCU] = CURSOR UP n times
[nCD] = CURSOR DOWN n times
[nSPC] = n SPACES

Equally, you could use screen Poking.

USR(62) sounds a beep on my Sharp MZ-80K and can be adapted or ignored. Other Sharp users will require an extended Basic for string inequalities and the logical operators And and Or. In addition, the ASCII codes given in the Data statements do not fit the Sharp, which has a non-standard set for lower-case letters.

When you Run the program, which takes about 5K, there will be a stage-wait while the moves are stored. You will then be asked if you wish to go first. Replying "Y" immediately makes the library inaccessible — this was put in to mimic the normal play from one end of the board only. After answering "N", you will be asked to select an opening — choose from the selection in table 1. For a random selection, type in 0.

The computer will now make its first move. You may then make your move by entering algebraic notation with commas, e.g., from

(continued on page 35)

```
100 REM** BOOK OPENING **
110 PRINT"[CLS]"
120 PRINT" BOOK OPENING by J.F.White."
130 PRINT" Storing positions."
140 DIMA (8,8),C$(10,10),D$(10,10),X(4),Y(4),X$(4),Y$(4)
150 FORI=1TO8
160 READA (I,8):A (I,1)= A (I,8)+ 32
170 A (I,7)=80:A (I,2)=112
180 NEXT
190 REM** SETTING UP BOARD **
200 REM** DATA IN 210 ARE ASCII VALUES FOR R,N,B,Q,K,B,N,R
210 DATA82,78,66,81,75,66,78,82
220 REM ** THE FOLLOWING DATA CAN BE INPUT FROM A DATA TAPE **
230 Z = 10: REM** Z = NO. OF OPENINGS STORED.
240 FORJ=1TOZ : FOR I= 1 TO 8
250 READ D$(I , J):READ C$(I +1,J )
260 NEXT:NEXT
270 DATA5254,3735,7163,4746,4244,3544,6344,7866,2133,2836,3175,3847,4142
280 DATA1838,5131,2644
290 DATA5254,3735,7163,4746,4244,3544,6344,7866,2133,7776,3153,6877,6152
300 DATA2836,5171,5878
310 DATA5254,5755,7163,2836,6125,1716,2514,7866,5171,6857,6151,2725,1423
320 DATA4746,3233,5878
330 DATA5254,5755,7163,2836,6134,6835,3233,7866,4244,5544,3344,3524,3142
340 DATA2442,2142,4745
350 DATA5254,5755,6264,5564,7163,7775,6134,4746,5171,8786,4244,6877,3233
360 DATA2836,7273,7574
370 DATA5254,5756,4244,4745,2133,6824,5455,3735,1213,2433,2233,7857,4174
380 DATA3544,7477,8878
390 DATA4244,4745,3234,5756,2133,7866,3175,6857,7163,5878,5253,2847,1131
400 DATA3736,6143,4534
410 DATA4244,7866,3234,5756,2133,6824,5253,2726,6143,3827,7163,2433,2233
420 DATA4745,3113,2847
430 DATA4244,7866,3234,7776,2133,6877,5254,4746,6152,5878,7163,5755,5171
440 DATA2847,4445,4735
450 DATA3234,3735,2133,2836,7273,7776,6172,6877,7163,7866,5171,5878,4244
460 DATA3544,6344,2644
470 REM ** END OF DATA **
480 INPUT"DO YOU WANT TO GO FIRST(Y\N)? ";A$
490 IFA$="Y"THENPRINT"[CLS]":GOTO600
500 INPUT"CHOOSE YOUR OPENING ";B$
510 BO=1:PRINT"[CLS]"
520 MO=MO+1
530 IFBO<>0THENGOSUB 910:IFBO<>0THEN600
540 REM ** MAIN PROGRAM HERE **
550 REM**
560 REM** NUMEROUS LINES **
570 PRINT"[CU]MAIN PROGRAM":USR(62):USR(62)
580 X$(1)="0":X$(2)="0":X$(3)="0":X$(4)="0"
590 REM**
600 PRINT"[HOME] FROM   ";X$(1) ;" ,";X$(2);"   TO   ";X$(3) ;" ,";X$(4)
610 USR(62)
620 IF A(X(3),X(4))= 107 THEN R=1:R1 = X(3): GOSUB 880
630 REM ** CASTLING
640 PRINT:PRINT
650 FORJ=8TO1STEP-1:FORI= 1 TO 8
660 PRINTTAB(5*I-5);
670 IFA (I,J)=0 THENPRINT".";:GOTO690
680 PRINT CHR$(A (I,J));
690 NEXT:PRINT J:PRINT: NEXT
700 PRINT:PRINT"A    B    C    D    E    F    G    H"
710 PRINT"[HOME][22 CD]";
720 INPUT"YOUR MOVE.  FROM ?  " ;A1$,B1$  :USR(62)
730 IFA1$="Q"THEN PRINT"[HOME][24 CD]";:GOTO750
740 GOTO790
750 PRINT"BOOK OPENING REQUIRES FROM ";Y$(1);
760 PRINT" ,";Y$(2);" TO ";Y$(3);" ,";Y$(4);
770 FORI=1TO3000:NEXTI:PRINT"[HOME][24 CD]";
780 PRINT"[38 SPC]";:GOTO710
790 INPUT"               TO  ?  ";A2$,B2$ :USR(62)
800 REM** ERROR CHECKING HERE **
810 A1=VAL(CHR$(ASC(A1$)-16))
820 B1= VAL(B1$): B2= VAL(B2$)
```

(listing continued on page 35)

(continued from page 33)

d,7 to d,5, or enter q,1 when the computer will tell you what you ought to do.

When the library is exhausted, or if you enter a move not recognised by the program, the warning "Main Program" appears. The machine now enters a continuous loop, accepting only your responses without reply, and should be terminated with Break, CTRL/c etc., as appropriate.

The Data statements could easily be replaced by Data tapes or discs and it would be very easy to build up a library of tapes entitled Nimzowitsch, Ruy Lopez, and so on, as a graphic means of learning book openings.

The opening move.

```
FROM   E .2   TO   E .4

R   N   B   Q   K   B   N   R 8
P   P   P   P   P   P   P   P 7
.   .   .   .   .   .   .   . 6
.   .   .   .   .   .   .   . 5
.   .   .   .   p   .   .   . 4
.   .   .   .   .   .   .   . 3
p   p   p   p   .   p   p   p 2
r   n   b   q   k   b   n   r 1

A   B   C   D   E   F   G   H

YOUR MOVE.  FROM ? D.1

BOOK OPENING REQUIRES FROM E .7 TO E .5
```

This could be far more fun than learning by rote from a book.

I wrote this program to find a way of storing book openings. I still do not know if this is the method used by commercial manufacturers — I suspect a yet simpler way may exist — but I hope that it will save some readers from reinventing the wheel. ■

(listing continued from page 33)

```
830 A2=VAL(CHR$(ASC(A2$)-16)): IF A(A1,B1)<90 OR A(A1,B1)>115THEN710
840 A1$=STR$(A1 ):A2$=STR$(A2 )
850 A(A2,B2)=A(A1,B1):A(A1,B1)=0
860 IF A(A2,B2) = 75 THEN R=8: R1= A2:GOSUB 880: REM ** CASTLING
870 GOTO520
880 IF R1 = 3 THEN A(4,R) = A(1,R): A(1,R) = 0
890 IF R1 = 7 THEN A(6,R) = A(8,R): A(8,R) = 0
900 RETURN
910 BO$="":BP$=A1$+B1$+A2$+B2$
920 IFVAL(B$) >0 THENBO=VAL(B$):IFBP$ = C$(MO,BO)THEN1010
930 FORI=1TO Z:REM ** Z = NO. OF BOOK OPENINGS
940 IF(BP$= C$(MO,I))AND(BO$= D$(MO-1,I)) THEN 960
950 GOTO970
960 IF BR$ = C$(MO-1,I)THENBO$=BO$ + STR$(I)
970 NEXTI
980 IF BO$= "" THEN BO=0:RETURN
990 BS = INT(LEN(BO$)*RND(1) + 1)
1000 BO = VAL(MID$(BO$,BS,1))
1010 IF VAL(D$(MO,BO))=0 THEN BO=0 :RETURN
1020 X(1)  = VAL(LEFT$(D$(MO,BO),1))
1030 X(2)  = VAL(MID$(D$(MO,BO).2,1))
1040 X(3)  = VAL(MID$(D$(MO,BO),3,1))
1050 X(4)  = VAL(RIGHT$(D$(MO,BO),1))
1060 BQ$=STR$(X(1))+STR$(X(2))+STR$(X(3))+STR$(X(4)):BR$=BP$
1070 A(X(3),X(4))=A(X(1),X(2)):A(X(1),X(2))=0
1080 Y$(1) =  LEFT$(C$(MO+1,BO),1)
1090 Y$(2) =  MID$(C$(MO+1,BO),2,1)
1100 Y$(3) =  MID$(C$(MO+1,BO),3,1)
1110 Y$(4) =  RIGHT$(C$(MO+1,BO),1)
1120 FOR K=1TO3 STEP2
1130 X$(K) = CHR$(ASC(STR$(X(K)))+16)
1140 Y$(K) = CHR$(ASC(Y$(K))+16)
1150 X$(K+1) = STR$(X(K+1))
1160 NEXT
1170 RETURN
```

GAMES

DOMINO DUELLING

David Smith delves behind the simple facade of dominoes to reveal an intriguing micro game.

IN A GAME OF dominoes between two players, there are four types of domino, depending on temporary ownership. Type 1 are those belonging to one player; type 2, those belonging to the other player; type 3, those in the chain of dominoes on the table; and type 0, those dominoes not in use.

As each domino can be thought of as two-dimensional, dominoes are referred to in the program by an array, D(X,Y), dimensioned in line 1020. Any member of the array can then be assigned a value 1,2,3 or 0 according to the type of domino.

Initially and at each new deal, all D(X,Y) are set at zero. During the deal — see lines 1570 to 1660 — nine dominoes are chosen at random, and set at 1 to represent allocation to the player. Similarly, another nine dominoes are selected at random and set at 2 to represent allocation to the computer. As soon as any domino is played, it is set at 3, not 0, to distinguish the fact that both players now know the position of this domino.

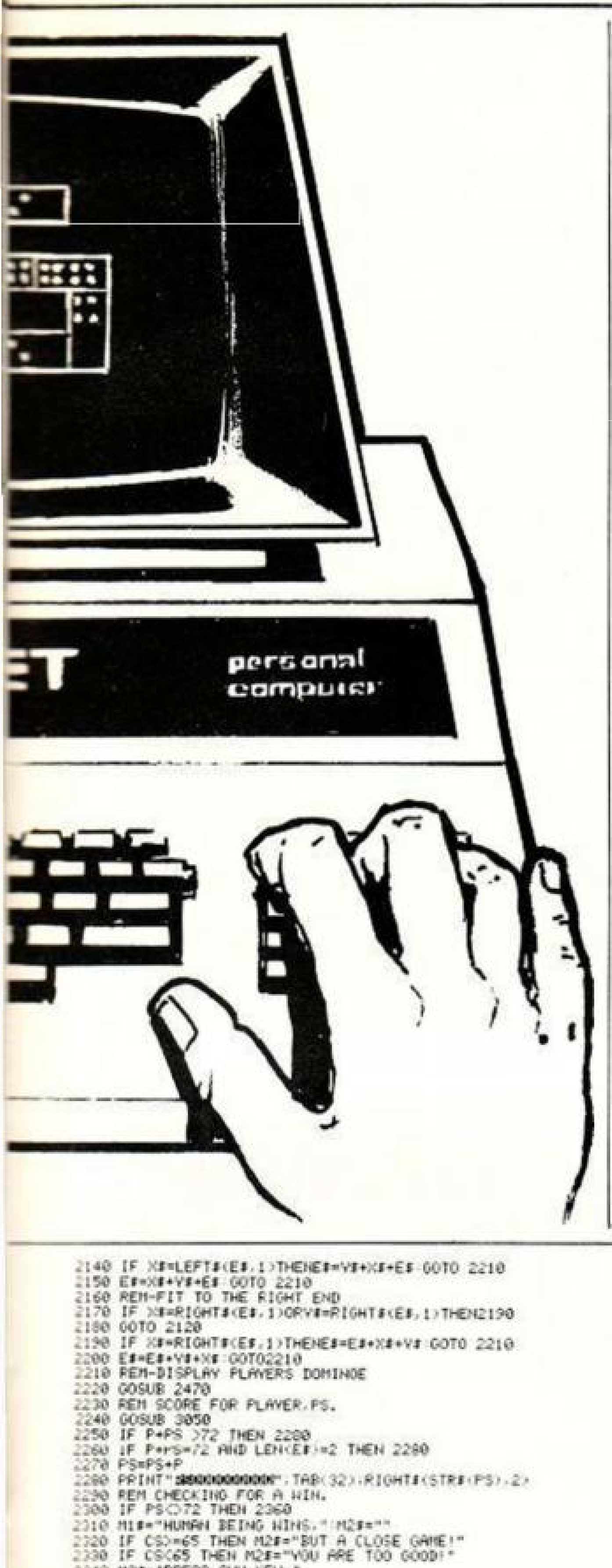
The chain of dominoes which forms during a game has a natural relationship with string variables, so the chain is represented by E$ to which it is easy to add dominoes to the left — see line 2140, for example — or to the right — line 2190. At each new deal the board is renewed simply by E$=" ", and special allowances have to be made for the first domino after a deal where no end-match is required — see line 2070.

Let us suppose that the random selector in line 1660 has chosen the player — as opposed to the computer — to play first. It seems only reasonable to tell him which dominoes he has been allocated, and this is done in lines 1630 onwards. A menu of choices follows from line 1700 to 1760, before selection of the domino to be played.

Various checks have to be made to verify the player's choice is legal; is it his domino to start with? Does it fit? Is he genuinely unable to play? Since you must play whenever possible, the computer must check your position when you say you cannot — see lines 1770 to 1850. The information obtained if you are unable to play is used in lines 1870 to 1910.

The computer strategy is in three parts which overlap within the program and are difficult to follow unless you have grasped the principles. If the computer is to play the first

```
1000 REM-DOMINOES COPYRIGHT D.N.SMITH.
1010 REM DOCUMENTED VERSION.
1020 DIM D(6,6)
1030 REM LOADING THE MACHINE CODE TO WHITE THE SCREEN QUICKLY.
1040 DATA 169,128,133,95,169,0,133,94,162,4,160,0,169,160,145,94,200,208,249
1050 DATA 230,95,202,208,244,96
1060 FORI=826 TO 850:READ A:POKEI,A:NEXT
1070 REM-INSTRUCTIONS IF REQUIRED.
1080 PRINT"J":SYS826:DE=INT(100*RND(2))
1090 PRINT"Q DOMINOES.BY D.N.SMITH."
1100 PRINT"QQ DO YOU REQUIRE INSTRUCTIONS,■YQ OR ■NQ?"
1110 GOSUB 2390:IF A$="N"THEN1350
1120 PRINT"QQ INSTRUCTIONS:-"
1130 PRINT"QQ THIS IS PLAYED LIKE ORDINARY DOMINOES"
1140 PRINT"Q EXCEPT THAT POINTS MAY BE SCORED."
1150 PRINT"QQ YOU PLAY THE COMPUTER,AND WIN IF YOU"
1160 PRINT"Q GET EXACTLY 72 POINTS BEFORE THE"
1170 PRINT"Q COMPUTER DOES."
1180 PRINT"QQ TO SCORE POINTS THE END DOMINOES MUST"
1190 PRINT"Q TOTAL A MULTIPLE OF 3 OR 5."
1200 PRINT"QQ EACH MULTIPLE SCORES 1 POINT."
1210 PRINT"QQQ PRESS ANY KEY AND I WILL SHOW YOU SOME"
1220 PRINT"Q EXAMPLES OF SCORING POINTS.":GOSUB 2390
1230 PRINT"JQQQQ":SYS826
1240 L=INT(7*RND(2)):R=INT(7*RND(2))
1250 F$=RIGHT$(STR$(L),1)+"65"+RIGHT$(STR$(R),1):GOSUB3070:IF F=0 THEN 1240
1260 X=L:Y=6:S=32815:GOSUB2600:X=6:Y=2:S=S+3:GOSUB 2600
1270 X=2:Y=5:S=S+3:GOSUB 2600:X=5:Y=R:S=S+3:GOSUB2600
1280 PRINT"Q REMEMBERING THAT ANY 'DOUBLES' COUNT
1290 PRINT"Q FOR DOUBLE,THE ENDS TOTAL FOR THE":PRINT"Q ABOVE BOARD=";T;"."
1300 PRINT"QQQ     ";RIGHT$(STR$(T),2);" MAKES ";M3;"MULTIPLES OF 3
1310 PRINT"Q          AND ";M5;"MULTIPLES OF 5"
1320 PRINT"QQ   TOTAL POINTS SCORED=";M3;"+";M5;"=";M3+M5
1330 PRINT"QQQ  FOR A FURTHER EXAMPLE PRESS ■FQ KEY.
1340 PRINT"QQ     TO PLAY THE GAME PRESS ■PQ KEY.":GOSUB2390:IFA$<>"P"THEN 1230
1350 REM-SETTING OUT SCREEN.
1360 PRINT"J":SYS826
1370 PRINT"QQQQQQ";TAB(5);"    COMPUTER DOMINOE GAME    "
1380 PRINT"QQ      ┌STATE-OF-PLAY-MESSAGE┐ ┌SCORES┐"
1390 PRINT"Q       |                     || |"
1400 PRINT"Q       | DEALING.            ||YOU=   |"
1410 PRINT"Q       |                     ||ME =   |"
1420 PRINT"Q       |                     ||WIN=72|"
1430 PRINT"Q       └─────────────────────┘└──────┘"
1440 PRINT"Q       ┌YOUR-DOMINOES───────────────┐"
1450 FORI=1 TO 6:PRINT"Q       |                          |":NEXT
1460 PRINT"Q       └────────────────────────────┘"
1470 PRINT"QQQQQQQQQQQQQ";TAB(32)RIGHT$(STR$(CS),2)
1480 PRINT"QQQQQQQQQQQQQ";TAB(32);RIGHT$(STR$(PS),2)
1490 REM RENEW DOMINOES.
1500 S$="32819328143280933089332893348933689336963370133706"
1510 S$=S$+"33711337163372133726334463324633204632846328393283432829"
1520 REM  RELEASING DOMINOES FROM PREVIOUS OWNERSHIP AND RENEWING THE BOARD,E$.
1530 E$="":CD=9:C=0
1540 FOR I=0 TO 6
1550 FOR J=I TO 6:D(I,J)=0:NEXTJ
1560 NEXT
1570 REM-DEALING THE DOMINOES.
1580 FOR I=1 TO 18
1590 Y=INT(7*RND(2)):X=INT((Y+1)*RND(2)):IF D(X,Y)<>0 THEN 1590
1600 IF INT(I/2)=I/2 THEN D(X,Y)=1:          GOTO 1620
1610 D(X,Y)=2
1620 NEXTI
1630 REM PRINTING THE DOMINOES
1640 GOSUB 4200
1650 REM WHOSE TURN TO START?
1660 DE=DE+1:IFINT(DE/2)=DE/2 THEN 3190
1670 REM-PLAYERS TURN STARTS HERE.
1680 REM PLAYERS FIRST DECISION.
1690 IF E$="" THEN 1930
1700 M1$="YOUR CHOICE OF PLAY"
1710 M2$="1=LEFT       2=RIGHT"
1720 M3$="3=CANNOT   R=RESIGN"
1730 GOSUB 2430:GOSUB 2390:C$=A$
1740 IF C$="R"THEN 4150
1750 C=VAL(C$):IF C<1 OR C>3 THEN 1700
1760 IF C<>3 THEN 1930
1770 REM CHECK THAT PLAYER CANNOT GO.
1780 M1$="OK.":M2$="":GOSUB2430
1790 L=VAL(LEFT$(E$,1)):R=VAL(RIGHT$(E$,1))
1800 FOR I=0 TO 6
1810 FOR J=I TO 6
1820 IF D(I,J)<>1 THEN 1850
1830 IF I=L OR J=L THEN 1920
1840 IF I=R OR J=R THEN 1920
1850 NEXT J:NEXT I
1860 FOR I=0 TO 6
1870 IF D(I,L)=0 THEN D(I,L)=3
1880 IF D(L,I)=0 THEN D(L,I)=3
1890 IF D(I,R)=0 THEN D(I,R)=3
1900 IF D(R,I)=0 THEN D(R,I)=3
1910 NEXT:GOTO 3170
1920 M1$="NO CHEATING!":M2$="YOU CAN PLAY!":GOSUB 2430:GOTO 2130
1930 REM-PLAYERS CHOICE OF DOMINOE.
1940 M1$="DOMINOE CHOSEN=?":M2$="":GOSUB 2430
1950 GOSUB 2390:X$=A$
1960 IF X$="0"THEN1980
1970 IF VAL(X$)<1 OR VAL(X$)>6THEN1950
1980 M2$=X$:GOSUB 2450
1990 GOSUB 2410:Y$=A$:IFY$="0"THEN 2010
2000 IF VAL(Y$)<1 OR VAL(Y$)>6THEN1990
2010 M2$=M2$+Y$:GOSUB2450
2020 IF X$>Y$ THEN A$=X$:X$=Y$:Y$=A$
2030 X=VAL(X$):Y=VAL(Y$)
2040 IF D(X,Y)=1 THEN2060
2050 M1$="NOT YOURS.TRY AGAIN!":GOSUB2430:GOTO2130
2060 M1$="OK.":M2$="":GOSUB 2430
2070 IFE$=""THEN E$=X$+Y$:GOTO 2210
2080 REM FIT PLAYERS DOMINOE.
2090 IF C=2 THEN 2160
2100 REM FIT TO THE LEFT END
2110 IF X$=LEFT$(E$,1) OR Y$=LEFT$(E$,1)THEN 2140
2120 M1$="IT DOES NOT FIT.":M2$="TRY AGAIN!":GOSUB2430
2130 FOR I=1 TO 2000:NEXT:GOTO1670
```



domino of a deal, a selection is made from a predetermined list of alternatives in lines 3190 to 3300 which, incidentally, appear in the program in reverse order of actual priority. If the game is in full progress, a copy of the chain of dominoes E$ is made into F$ in line 3060. In this way, various alternative dominoes can be fitted to F$ and scored without affecting E$. The highest-scoring computer-owned domino is then selected, stored in E1, E2 and E3 in line 3570, awaiting possible use later.

The final part of the computer strategy is in lines 3500 to 3570, where depending on the state of play indicated by the players score PS, either the high-scoring or the defensive domino is selected — sometimes they are the same. The domino is displayed, ownership transferred, and play continues as indicated in the flowchart. The 5,3 domino shown in figure 1 has been made from the Pet graphics shown in figure 2.

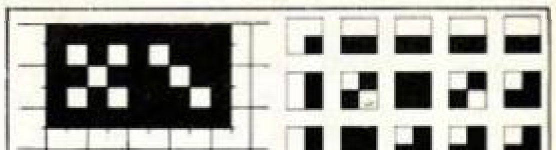
The string S$ is simply a list of five-figure screen Poke numbes, each of which is used as a reference point for the rest of the domino to relate to. The value of S determines whether the domino is in the top row, left row, right row or bottom row, and is used to select the appropriate subroutine to Poke the domino image on to the screen.

The main problems in adapting the program lie in the display system. The machine-code screen white-out in lines 1030 to 1060 will need replacement either by machine code or a Basic program to white the screen. The entire display system would need revision as it is largely machine-based, dependent on the Pet graphics and screen location numbers.

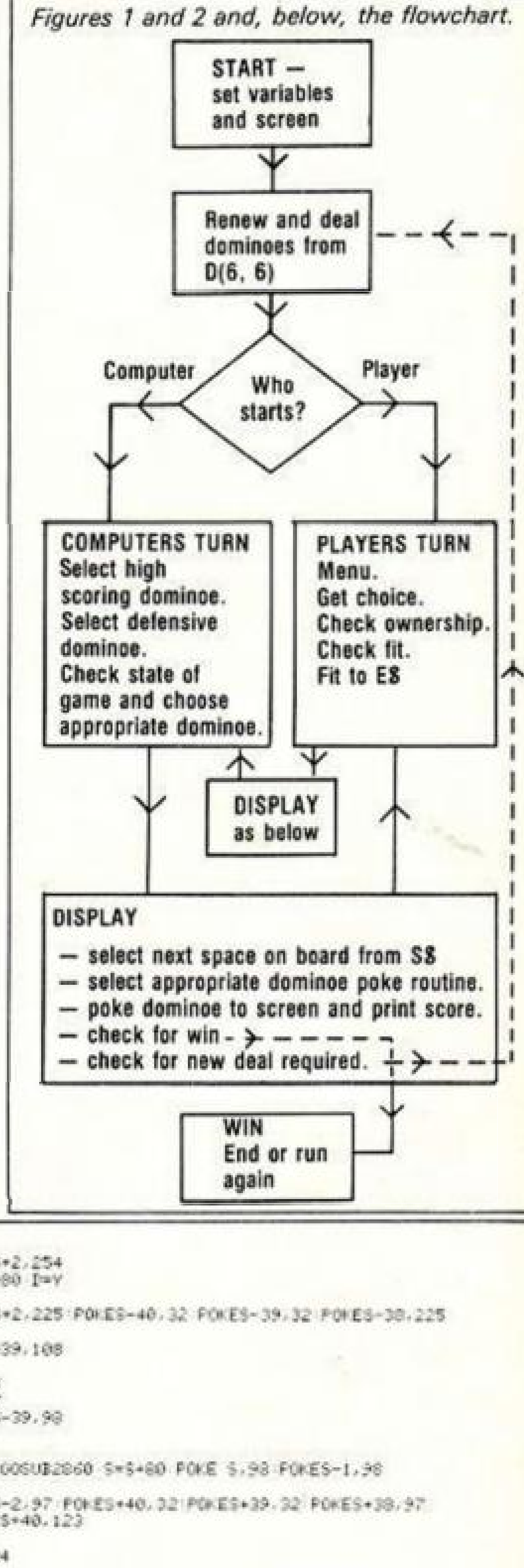
A simple version of display would be to delete lines 2470 to 3040 and replace them with

```
2470 PRINT E$:RETURN
```

and rewrite line 4250 so that the player's dominoes were printed instead of Poked.



Figures 1 and 2 and, below, the flowchart.



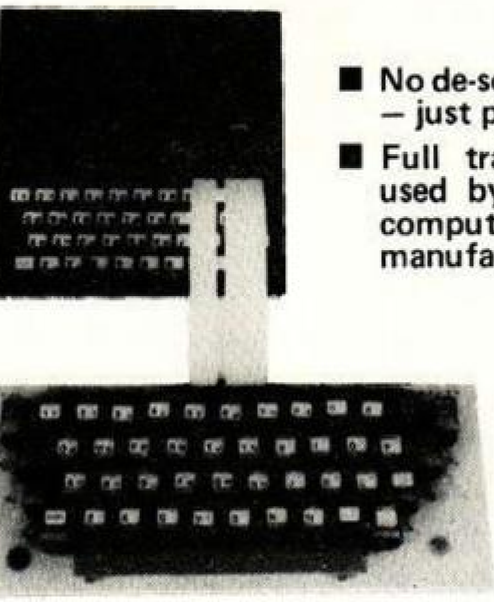

```
2140 IF X$=LEFT$(E$,1)THENE$=V$+X$+E$:GOTO 2210
2150 E$=X$+V$+E$:GOTO 2210
2160 REM-FIT TO THE RIGHT END
2170 IF X$=RIGHT$(E$,1)ORV$=RIGHT$(E$,1)THEN2190
2180 GOTO 2120
2190 IF X$=RIGHT$(E$,1)THENE$=E$+X$+V$:GOTO 2210
2200 E$=E$+V$+X$:GOTO2210
2210 REM-DISPLAY PLAYERS DOMINOE
2220 GOSUB 2470
2230 REM SCORE FOR PLAYER.PS.
2240 GOSUB 3050
2250 IF P+PS >72 THEN 2280
2260 IF P+PS=72 AND LEN(E$)=2 THEN 2280
2270 PS=PS+P
2280 PRINT"▒▒▒▒▒▒▒▒▒▒▒";TAB(32);RIGHT$(STR$(PS),2)
2290 REM CHECKING FOR A WIN.
2300 IF PS<>72 THEN 2360
2310 M1$="HUMAN BEING WINS.":M2$=""
2320 IF CS>=65 THEN M2$="BUT A CLOSE GAME!"
2330 IF CS<65 THEN M2$="YOU ARE TOO GOOD!"
2340 M3$="PRESS ANY KEY."
2350 GOSUB2430:GOSUB 2390:RUN 1070
2360 REM-USING UP THE PLAYERS DOMINOE.
2370 D(X,Y)=3:D(Y,X)=3:GOSUB 4200:IF CP=0 THEN 3970
2380 GOTO 3170
2390 REM-GET SUBROUTINE.
2400 POKE 158,0
2410 GETA$:IFA$=""THEN2410
2420 RETURN
2430 REM-MESSAGE SUBROUTINE.
2440 PRINT"▒▒▒▒▒▒▒▒▒▒▒";TAB(6);"▒                    ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";M1$
2450 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒";TAB(6);"▒                    ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";M2$
2460 PRINTTAB(6);"▒              ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";M3$:M3$="":RETURN
2470 REM-DISPLAY DOMINOE S/ROUTINE.
2480 REM SELECT NEXT SPACE FOR DOMINOE
2490 IFLEN(E$)=2 THEN S=32824:GOTO2600
2500 IF C<>1 THEN 2530
2510 S=VAL(LEFT$(S$,5)):S$=MID$(S$,6)
2520 X=VAL(LEFT$(E$,1)):Y=VAL(MID$(E$,2,1))
2530 IF C<>2 THEN 2560
2540 S=VAL(RIGHT$(S$,5)):S$=LEFT$(S$,LEN(S$)-5)
2550 X=VAL(MID$(E$,LEN(E$)-1,1)):Y=VAL(RIGHT$(E$,1))
2560 IF S<32840 THEN 2600
2570 IF S>33693 THEN 2940
2580 IF S=33089 OR S=33289 OR S=33489 OR S=33689 THEN 2710
2590 GOTO 2830
2600 REM-TOP ROW DOMINOE
2610 D=X:GOSUB2630:D=Y:S=S+2:GOSUB2630
2620 POKE S+2,225:POKE S+42,225:POKE S+82,254:RETURN
2630 POKES,32:POKES+1,32:POKES+40,32:POKES+41,32:POKES+80,98:POKES+81,98
2640 IFD>1THENPOKES,108
2650 IFD>3THENPOKES+1,108:POKES+40,108
2660 IFD=1THENPOKES+41,126
2670 IFD=2ORD=4THENPOKES+41,108
2680 IFD=3ORD=5THENPOKES+41,127
2690 IFD=6THENPOKES+40,225:POKES+41,225
2700 RETURN
2710 REM-LEFT SIDE DOMINOE
2720 POKES,98:POKES+1,98:POKES+2,254
2730 S=S-40:D=X:GOSUB2750:S=S-80:D=Y
2740 GOSUB2750:RETURN
2750 POKES,32:POKES+1,32:POKES+2,225:POKES-40,32:POKES-39,32:POKES-38,225
2760 IFD>1THENPOKES-40,108
2770 IFD>3THENPOKES,108:POKES-39,108
2780 IFD=1THENPOKES+1,126
2790 IFD=2ORD=4THENPOKES+1,108
2800 IFD=3ORD=5THENPOKES+1,127
2810 IFD=6THENPOKES+1,98:POKES-39,98
2820 RETURN
2830 REM-RIGHT SIDE DOMINOE
2840 D=X:GOSUB2860:D=Y:S=S+80:GOSUB2860:S=S+80:POKE S,98:POKES-1,98
2850 POKES-2,252:RETURN
2860 POKES,32:POKES-1,32:POKES-2,97:POKES+40,32:POKES+39,32:POKES+38,97
2870 IFD>1THENPOKES-1,123:POKES+40,123
2880 IFD>3THENPOKES,123
2890 IFD=1ORD=3THENPOKES+39,124
2900 IFD=4THENPOKES+39,123
2910 IFD=5THENPOKES+39,255
2920 IFD=6THENPOKES+39,98:POKES-1,98
2930 RETURN
2940 REM-BOTTOM ROW DOMINOE
2950 POKES,225:POKES-40,225:POKES-80,251:S=S-1:D=X:GOSUB2970
2960 S=S-2:D=Y:GOSUB2970:RETURN
2970 POKES,32:POKES-40,32:POKES-80,226: POKES-1,32:POKES-41,32:POKES-81,226
2980 IFD>3THENPOKES,124:POKES-41,124
2990 IFD>1THENPOKES-1,124
3000 IFD=1THENPOKES-40,123
3010 IFD=2ORD=4THENPOKES-40,124
3020 IFD=3ORD=6ORD=5THENPOKES-40,255
3030 IFD=6THENPOKES-40,225:POKES-41,225
3040 RETURN
3050 REM-SCORE SUBROUTINE.
3060 F$=E$
3070 M3=0:M5=0
3080 A=VAL(LEFT$(F$,1))
3090 B=VAL(RIGHT$(F$,1))
3100 IF LEN(F$)=2 THEN 3130
3110 IF RIGHT$(F$,1)= MID$(F$,LEN(F$)-1,1) THEN B=2*B
3120 IF LEFT$(F$,1)=MID$(F$,2,1)THEN A=2*A
3130 T=A+B
3140 IF(T/3=INT(T/3))THEN M3=T/3
3150 IF(T/5=INT(T/5))THEN M5=T/5
3160 P=M3+M5:RETURN
3170 REM-COMPUTERS TURN STARTS HERE.
3180 M1$="COMPUTING REPLY.":M2$="":GOSUB 2430:IFE$<>""THEN 3310
3190 REM-SELECT FIRST DOMINOE
3200 FOR I=0 TO 6:FORJ=ITO6
3210 IF D(I,J)=2 THENX=I:Y=J:GOTO3230
3220 NEXTJ:NEXTI
3230 IF D(0,6)=2 THEN X=0:Y=6
3240 IF D(3,3)=2 THEN X=3:Y=3
3250 IF D(4,6)=2 THEN X=4:Y=6
3260 IF D(1,5)=2 THEN X=1:Y=5
3270 IF D(2,4)=2 THEN X=2:Y=4
```

(continued on next page)

(continued from previous page)

```
3280 IF D(4,5)=2 THEN X=4:Y=5
3290 IF D(6,6)=2 AND D(3,6)=2 THEN X=6:      Y=6
3300 C=1:GOTO 3930
3310 REM-COMPUTERS STRATEGY.
3320 REM NULLIFY PREVIOUS CHOICES
3330 CO=0:MAX=0:F$=E$:L=VAL(LEFT$(E$,1)):R=VAL(RIGHT$(E$,1)):PM=50
3340 E1=7:E2=7:E3=7:E4=7:E5=7:E6=7:E7=7:E8=7:E9=7
3350 REM TEST EACH COMPUTER DOMINOE FOR FIT TO CHAIN OF DOMINOES.
3360 FOR I=6 TO 0 STEP-1
3370 FOR J=6 TO I STEP-1
3380 IF D(I,J)<>2 THEN 3490
3390 IF I<>L AND J<>L AND I<>R AND J<>R THEN 3490
3400 CO=CO+1:J$=RIGHT$(STR$(J),1):I$=RIGHT$(STR$(I),1)
3410 IFI=LTHENC=1:F$=J$+I$+F$:GOTO3440
3420 IFJ=LTHENC=1:F$=I$+J$+F$:GOTO3440
3430 GOTO 3450
3440 GOSUB 3580
3450 IFI=RTHENC=2:F$=F$+I$+J$:GOTO3480
3460 IFJ=RTHENC=2:F$=F$+J$+I$:GOTO3480
3470 GOTO 3490
3480 GOSUB 3580
3490 NEXTJ:NEXTI
3500 REM SELECT SUITABLE DOMINOE
3510 IF CO<>0 THEN 3550
3520 M1$="COMPUTER CANNOT GO.":M2$="":  GOSUB 2430:FOR I=1 TO 2000:NEXT
3530 IF C=3 THEN M1$="NO PLAY POSSIBLE.":GOTO 3980
3540 GOTO1670
3550 IF E4=7 THEN 3570
3560 IF PS=64 OR PS=66 OR PS>67 OR MAX=0 THEN C=E7:X=E8:Y=E9:GOTO3930
3570 C=E1:X=E2:Y=E3:GOTO3930
3580 REM-CALCULATING SUBROUTINE.
3590 GOSUB 3070
3600 IF P+CS=72 THEN X=I:Y=J:GOTO3930
3610 IF P+CS>72 THEN P=0
3620 IF P<MAX THEN 3640
3630 MAX=P:E1=C:E2=I:E3=J
3640 IF PS=64 OR PS=66 OR PS>67 OR MAX=0 THEN 3660
3650 GOTO 3810
3660 G$=F$:PQ=0
3670 FOR K=0 TO 6
3680 FOR Z=K TO 6
3690 IF D(K,Z)=3 OR D(K,Z)=2 THEN 3790
3700 K$=RIGHT$(STR$(K),1):Z$=RIGHT$(STR$(Z),1)
3710 IF K$=LEFT$(G$,1) THEN G$=Z$+K$+G$         GOTO3740
3720 IF Z$=LEFT$(G$,1)THEN G$=K$+Z$+G$:        GOTO3740
3730 GOTO3750
3740 GOSUB 3820:IF P>=PQ THEN PQ=P:E4=C:E5=I:E6=J
3750 IF K$=RIGHT$(G$,1)THENG$=G$+K$+Z$:        GOTO3780
3760 IF Z$=RIGHT$(G$,1)THEN G$=G$+Z$+K$         GOTO3780
3770 GOTO 3790
3780 GOSUB 3820:IF P>=PQ THEN PQ=P:E4=C:E5=I:E6=J
3790 NEXTZ:NEXTK
3800 IF PQ<=PM THEN E7=E4:E8=E5:E9=E6:PM=PQ
3810 F$=E$:RETURN
3820 REM-SCORE G$
3830 M3=0:M5=0
3840 A=VAL(LEFT$(G$,1))
3850 B=VAL(RIGHT$(G$,1))
3860 IF LEN(G$)=2 THEN 3890
3870 IF RIGHT$(G$,1)= MID$(G$,LEN(G$)-1,1) THEN B=2*B
3880 IF LEFT$(G$,1)=MID$(G$,2,1)THEN A=2*A
3890 T=A+B:IF(T/3=INT(T/3))THEN M3=T/3
3900 IF(T/5=INT(T/5))THEN M5=T/5
3910 P=M3+M5:IF PS+P>72 THEN P=0
3920 G$=F$:RETURN
3930 REM-COMPUTER PLAYING ITS CHOICE
3940 M1$="COMPUTER PLAYING":M2$=STR$(X)+STR$(Y):GOSUB 2430
3950 D(X,Y)=3:D(Y,X)=3:CD=CD-1
3960 IF CD<>0THEN3990
3970 M1$="OUT OF DOMINOES"
3980 M2$="DEAL AGAIN.":GOSUB 2430:FOR I=1 TO 2000:NEXT:GOTO 1350
3990 X$=RIGHT$(STR$(X),1)
4000 Y$=RIGHT$(STR$(Y),1)
4010 IF C=2 THEN 4040
4020 IF X$=LEFT$(E$,1)THENE$=Y$+X$+E$:GOTO 4060
4030 E$=X$+Y$+E$:GOTO 4060
4040 IF X$=RIGHT$(E$,1)THENE$=E$+X$+Y$:       GOTO 4060
4050 E$=E$+Y$+X$
4060 GOSUB 2470
4070 REM SCORE FOR COMPUTER.
4080 GOSUB 3050
4090 IF CS+P>72 THEN 4190
4100 IF P+CS=72 AND LEN(E$)=2 THEN4120
4110 CS=CS+P
4120 PRINT"[illegible]";TAB(32)RIGHT$(STR$(CS),2)
4130 REM CHECK FOR WIN.
4140 IF CS<>72THEN 4190
4150 M1$="COMPUTER WINS!":M2$=""
4160 IF PS<66 THEN M2$="GET SOME PRACTICE."
4170 IF PS>65 THEN M2$="YOU PLAYED WELL!"
4180 GOTO 2340
4190 D(X,Y)=3:D(Y,X)=3:GOTO 1670
4200 REM-PRINT DOMINOES SUBROUTINE.
4210 CP=0:PRINT"[illegible]"
4220 FOR I=1 TO 5:PRINTTAB(6);"[illegible]                    ":NEXT
4230 FOR I=0 TO 6
4240 FOR J=I TO 6
4250 IF D(I,J)=1 THEN CP=CP+1:X=I:Y=J:S=33373+3*CP:GOSUB2830
4260 NEXTJ:NEXTI
4270 RETURN
4280 REM LIST OF VARIABLES.
4290 REM D(X,Y)=DOMINOE ARRAY.
4300 REM X>Y NOT ALLOWED.
4310 REM D(X,Y)=0 IF NOT DEALT.
4320 REM D(X,Y)=1 IF DEALT TO PLAYER.
4330 REM D(X,Y)=2 IF COMPUTERS.
4340 REM D(X,Y)=3 ONCE PLAYED.
4350 REM X,Y OR X$,Y$ =A DOMINOE.
4360 REM CD=NO. OF DOMINOES IN HAND.
4370 REM CP=COUNT OF PLAYER DOMINOES
4380 REM CO=NO. OF DOMINOES THAT FIT.
4390 REM E$=THE CHAIN OF DOMINOES.
4400 REM F$,G$=COPIES OF THE CHAIN.
4410 REM C=1 FOR LEFT,C=2 FOR RIGHT.
4420 REM C=3=CANNOT PLAY,C=0=DEFAULT.
4430 REM L=VALUE OF LEFT OF CHAIN.
4440 REM R=VALUE OF RIGHT OF CHAIN.
4450 REM A=POINTS FOR LEFT OF CHAIN.
4460 REM B=POINTS FOR RIGHT OF CHAIN.
4470 REM T=A+B=ENDS TOTAL.
4480 REM M3=MULTIPLES OF THREE IN T.
4490 REM M5=MULTIPLES OF FIVE IN T.
4500 REM P=M3+M5=SCORE FOR DOMINOE.
4510 REM CS=COMPUTER SCORE TOTAL.
4520 REM PS=PLAYER SCORE TOTAL.
4530 REM S$=LIST OF SCREEN LOCATIONS.
4540 REM S=NEXT SCREEN LOCATION.
4550 REM I,J,D=DUMMY VARIABLES.
4560 REM DE=COUNT OF DEALS TO DATE.
4570 REM MAX=MAX COMPUTER SCORE.
4580 REM PM=MIN PLAYER SCORE.
4590 REM PQ=MAX LIKELY PLAYER SCORE.
4600 REM I$,J$=POSS. COMPUTER DOMINOE.
4610 REM Z$,K$=POSS. PLAYER DOMINOE.
4620 REM M1$ - M3$=MESSAGES TO SCREEN.
4630 REM E1 - E9=DOMINOES SELECTED
```

ADD A PROFESSIONAL KEYBOARD TO YOUR ZX 81



- No de-soldering necessary — just plug in.
- Full travel keyboard as used by international computer and terminal manufacturers.

£28·95
including VAT post& packing

"All-you-need" easy to assemble kit comprises:

One piece 47 key keyboard module fitted but not soldered to double-sided printed circuit board; connectors; simple plug-in flexible cables; screws, mounting feet and legends. Two-part keycaps with buttons pre-assembled on keyboard and clear protective clip-on caps for fitting after legends have been positioned. Comprehensive assembly instructions supplied with each kit. Allow 28 days for delivery.

Please send ...... keyboard kit(s) @ £28.95 each, cheque or postal order made payable to:- YC Feb

DK **COMPUTER KEYBOARDS (dept.)**
Glendale Park. Fernbank Road, Ascot, Berkshire.
Phone-a-Leaflet 03447 4731
Please ensure your name and address are clearly stated

THE

BUFFER

MICRO SHOP

(NEXT TO STREATHAM STATION)

NEW SOFTWARE SHOP EXCLUSIVELY FOR

ZX81

PROGRAMS, GAMES, "ADD-ONS"

MOST OF THE MAIL ORDER ITEMS ADVERTISED IN THIS MAGAZINE AVAILABLE OVER THE COUNTER

LOADING PROBLEMS? TRY OUR INTERFACE
BUSINESS & TECHNICAL DATA HANDLING PROGS;
PROPER KEYBOARDS; CONSOLES; VDUs

**The BUFFER Micro Shop,
374a Streatham High Road,
London SW16
Tel: 01-274 6674**

CAMBRIDGE LEARNING

SELF-INSTRUCTION COURSES

A PRACTICAL DIGITAL ELECTRONIC KIT FOR ☆ LESS THAN £20 ☆

SUITABLE FOR BEGINNERS

Learn the wonders of digital electronics and see how quickly you are designing your own circuits. The kit contains: *seven LS TTL integrated circuits, breadboard, LEDs, and all the DIL switches, resistors, capacitors, and other components to build interesting digital circuits;* **plus** a very clear and thoroughly tested instruction manual (also available separately). All this comes in a pocket size plastic wallet for only £19-90p inc VAT and p&p. This course is for true beginners:

- needs no soldering iron.
- asks plenty of questions, but never leaves you stuck and helpless.
- teaches you about fault-finding, improvisation, and subsystem checking.
- the only extra you need is a 4½V battery (Ever Ready 1289, or similar), or a stabilised 5V power supply.

Using the same breadboard you may construct literally millions of different circuits.

This course teaches boolean logic, gating, R-S and J-K flipflops, shift registers, ripple counters, and half-adders. Look out for our supplementary kits which will demonstrate advanced arithmetic circuits, opto-electronics, 7-segment displays etc.

Other self-instruction courses from Cambridge Learning Ltd include:

COMPUTER PROGRAMMING IN BASIC	£10.50
DIGITAL COMPUTER LOGIC AND ELECTRONICS	£ 8.50
DESIGN OF DIGITAL SYSTEMS	£14.00

Please send for full details (see coupon below).

GUARANTEE No risk to you. If you are not completely satisfied, your money will be refunded upon return of the item in good condition within 28 days of receipt.

**CAMBRIDGE LEARNING LIMITED, UNIT 82, RIVERMILL SITE, FREEPOST, ST. IVES, CAMBS, PE17 4BR, ENGLAND.
TELEPHONE: ST. IVES (0480) 67446. VAT No. 313026022.**

All prices include worldwide postage (airmail is extra - please ask for prepayment invoice). Giro A/c No 2789159.

Please allow 28 days for delivery in UK

Please send me:

......SUPERKIT(S) @ £19.90

......Free details of your other self-instruction courses.

I enclose a *cheque/PO payable to Cambridge Learning Ltd for £......... (*delete where applicable)

Please charge my:

*Access / American Express / Barclaycard / Diners Club Eurocard / Visa / Mastercharge / Trustcard

Expiry Date............ Credit Card No ....

Signature...

Telephone orders from card holders accepted on 0480 67446

Overseas customers (including Eire) should send a bank draft in sterling drawn on a London bank, or quote credit card number.

Name..

Address...

....

....

Cambridge Learning Limited, Unit 82, Rivermill Site, FREEPOST, St. Ives, Huntingdon, Cambs, PE17 4BR, England. (Registered in England No. 1328762).

STAY ONE MOVE AHEAD IN STRATEGY PROGRAM

Boris Allan describes the logical processes which go into the construction of a game of strategy.

THIS IS A truthful description of the thought processes of one person — me — and how that person planned a program. The program itself is immaterial to this account of the plan of action — that is, the algorithm — and will be considered separately next month.

To write a program which allows two people to play noughts and crosses, or tic-tac-toe, using the computer as an erasable board with a winning-line check, is not difficult; to write a simple program is slightly more difficult; and to write an intelligent program, where the user plays against the program, is more difficult still.

There are many programs to play noughts and crosses, but few of them play an intelligent game against the user. Clearly, routine questions and operations are easily solved and coded, but problem-solving procedures are an altogether more difficult proposition.

In deciding to write a program to play an intelligent game of noughts and crosses, my first problem was which of three approaches I should take:

- I could modify an existing procedure.
- I could write a learning program.
- I could write an already-intelligent program.

It does not take much thought to realise that, once the basics are known, it is impossible to

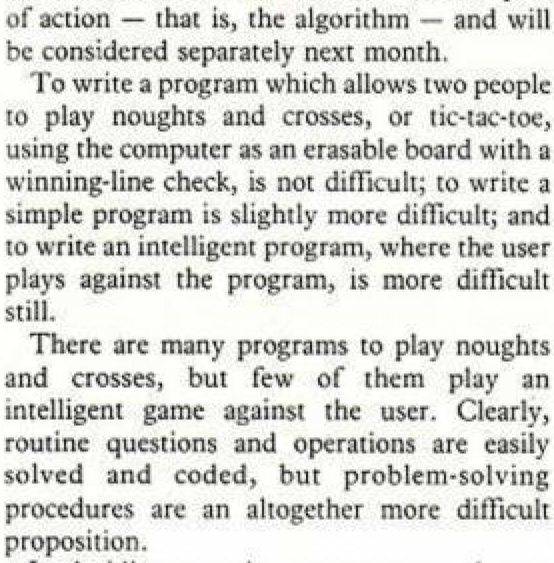


Figure 3.

lose at noughts and crosses, and against most people it is impossible to win. So, whereas all games between intelligent players are drawn, most intelligent noughts and crosses programs are fallible, so I ignored the first option. For the second, if it were possible to imagine trying the game for the first time, how would a person go about learning and preparing to play? If the game were chess, most novices learn by playing the game, and gaining from experience: this is the idea behind the learning program.

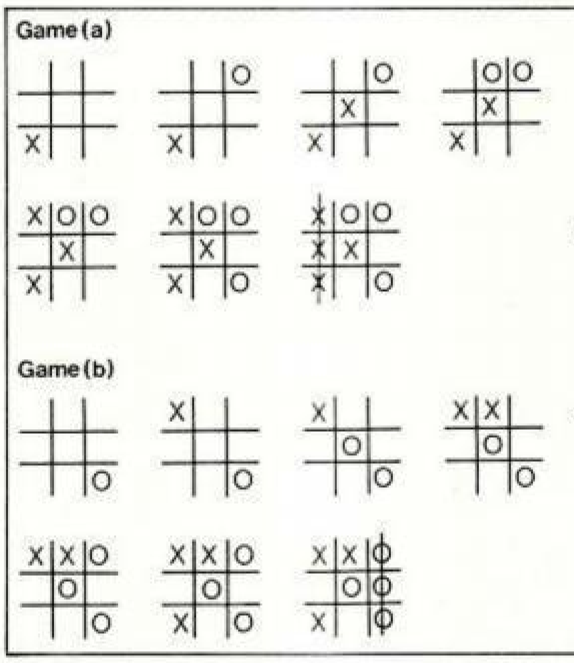


Figure 1.

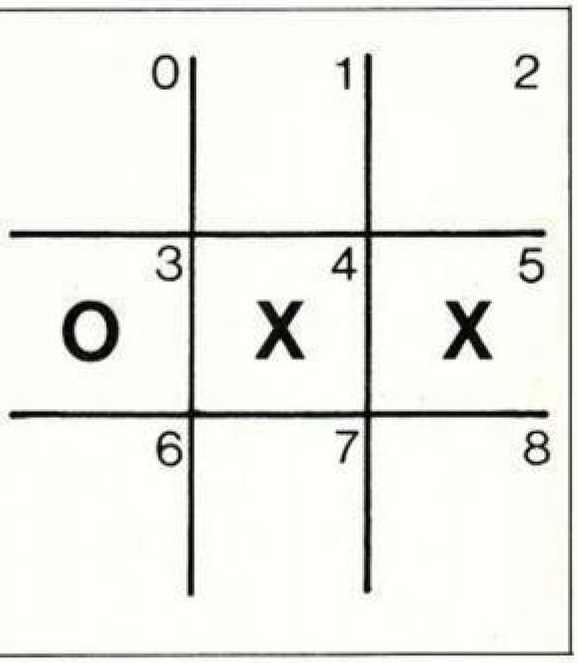


Figure 2.

Some chess novices study the game of chess beforehand to learn plans of action and general strategy. This is the idea behind the already intelligent program, and most chess-playing programs are of the already-intelligent type. Noughts and crosses is so simple that an already-intelligent program should never lose, but chess is so complex that total intelligence is not possible.

Apart from a few rules — such as, try to place three in a line and if your opponent has two in a line, block the line — a learning program will be designed to move randomly, keeping track of moves made. If a sequence leads to a win for either side, the program will try to copy that sequence in any future game. When the sequence is broken by the opponent, the program will try not to follow a losing sequence. This is what a human novice would try to do, given any intelligence.

This method occupies a great deal of storage, and humans must, indeed, be marvellous to cope with all this complexity. The way humans cope is by simplifying and looking for patterns.

In figure 1, the two games follow the same pattern: in game 1b if O and X are exchanged, and the board reflected about a vertical axis, the games are identical. There are only three different opening moves in noughts and crosses: a corner, the centre, and mid-side. After a corner move, there are five different moves; there are only two moves to follow a centre move; and a possible five moves follow a mid-side move.

This pattern analysis is performed readily, and mostly subconsciously, by the normal human. Yet a program would need a set of explicit rules — of rotation, and of reflection — to follow the pattern-making. By making the learning program already intelligent about pattern-making, it assumes less extreme proportions as far as storage is concerned. It does, however, become more complex in terms of program instructions. All learning programs need some resident intelligence and so, for these reasons, I turned to the already-intelligent program.

There are three levels to most games:

- The move — for example, make an entry in a vacant square.
- The tactic — for example, if there are two in a line, fill the empty square.
- The strategy — for example, where to move so as to maximise the possibility of a future win.

Strategy — always start from the centre or corner — sets the basis on which tactics — having two sets of two in a row — can be used, to produce the move. In noughts and crosses the move one makes is dependent on two tactical considerations, and one strategical consideration — the computer plays X:

- if X has two in a line, complete line — tactic — else
- if O has two in a line, block line — tactic — else
- decide on the best square to fill, to maximise potential tactical advantage, and to minimise O's potential tactical advantage — strategy.

It seemed an easy program to plan — apart from the third consideration which happens to be the most important consideration.

Playing through many games, trying traps to see how they worked, I began to see that I knew they were traps because I looked moves ahead. My first strategical plan — the third condition — read:

- Find the square with the greatest number of openings for X, and if there is more than one such square, fill it with the greatest number of openings for O.

I later discovered that this was wrong but at least it introduced an important idea — an opening. In the game shown in figure 2 the openings are as follows:

(continued on next page)

(continued from previous page)

Cell	.	X	O
0	.	1	2
1	.	1	1
2	.	0	0 Cell filled
3	.	0	0 Cell filled
4	.	0	0 Cell filled
5	.	0	0 Cell filled
6	.	1	2
7	.	2	0
8	.	2	0

If O is to move, then cell 0 is his winning position. Not only is it possible to have two lines for O from cell 0, but also each line contains another 0. If X is to move, cells 7 and 8 have two potential openings. If he chooses cell 7, the game is drawn. The strategy can be seen to be defective, but the idea of openings seems useful. What was omitted from that strategy was a look-ahead facility.

At this point I changed my vocabulary to incorporate a new concept, that of a potential square. The emphasis changed from the line to the square. A potential square was an empty square for which, if the square was filled, there would be three in a line. In figure 2, with X having moved to cell 8, the openings are:

Cell	.	X	O
0	.	1p	2
1	.	1	1
2	.	0	0
3	.	0	0
4	.	0	0
5	.	0	0
6	.	1	1
7	.	2	0
8	.	0	0

For cell 0, the important cell, there are two openings for O; and as this cell is also a

Total	Cells filled 0	1	2
+2	–	–	psO, opO
+1	–	opO	–
0	opX,op-		–
-1	–	opX	–
-2	–	–	psX, opX

Note "X" is −1 "O" is +1

Table 1.

potential square for X — shown by the "p" — then O must go to cell 0 or lose.

O, by moving to cell 0, has two openings which produce two potential squares. The third condition now becomes — X is the computer:

- Find the square with the greatest number of openings for X, but any resulting potential square for X must not have two openings for O.

So, the already-intelligent program uses strategical notions of openings and of potential squares to decide on where to move when not forced by tactical considerations.

A strategy is a pre-defined decision procedure which should give an answer, or decision, about every situation and should not rely on random moves. The strategy for intelligent noughts and crosses is, using our new vocabulary:

- if potential square for X then fill, else
- if potential square for O then fill, else
- find square for which openings for X are greatest. If a resulting potential square for X is created for which there are two openings for O, ignore it and repeat; otherwise fill.

The strategy assumes we have: a check of openings for X, and for O; a check on potential squares for X and for O; a record of the current board positions; and checks for the state of affairs, one move ahead.

There seem to be two key procedures: checking openings after every move, and finding potential squares after every move: Input, output, and a record of board positions are simple to arrange. We do not need a win check, because to fill one's own potential square is by definition a win.

Once a square has been used, there are no openings and the square can be skipped. If R represents row, and C represents column — both going from 0 to 2 — then row R and column C are always checked. If R + C = 2 then the forward diagonal — 2,0 to 0,2, see figure 3 — is checked. If R = C then the backward diagonal is checked. For any line through an empty square, the line can either:

1. have two of a kind, or
2. have one of each kind, or
3. have one of one kind, or
4. have no entries.

Option 1 is an opening for one player, plus a potential square. Option 2 is a dead line with no openings or potential squares. If option 3 holds, there is an opening for one player and option 3 is an example of openings for both players. Table 1 expresses this in a more formal way: "op" stands for opening; "ps" stands for potential square; and "—" indicates either a dead line, or an impossible combination.

The three cells on any line can be copied into a three-element array V(0), V(1), V(2) and the decision table, table 1, can be emulated by some routine such as the following written in some strange language.

```
cells = abs(V(0)) + abs(V(1)) + abs(V(2))
total = V(0) + V(1) + V(2)
opX = 0; opO = 0; psX = 1; psO = 1
IF abs(total) GREATERHAN 0 GOTO LABEL1
IF cells GREATERTHAN 0 THEN EXIT
opX = 1; opO = 1; EXIT
LABEL1: IF total LESSTHAN 0 GOTO LABEL2
IF cells EQUALS 2 THEN psO = −1
opO = 1; EXIT
LABEL2: IF cells EQUALS 2 THEN psX = −1
opX = 1; EXIT
```

At the end of this routine, an opening will be coded as 1 — else 0 — and a potential square will be coded −1 — else +1. If the cell number of the square for which the test is being made is square, and the array which holds the state of play for X is state X — with state 0 for O — then state X has the number of openings for each square, and the information about whether the square is a potential square, coded by making the number negative. Writing

```
state X(square) = (abs(state X(square)) + op X)*ps X
```

gives us this information.

If an entry is made in a new square, a check is needed to see if any potential squares have been created. The check will be similar to that which have used already, but since the maximum count could now be three, the co-ordinates of any potential squares are needed to check against the opponent's openings.

If a line is again stored in the array V, and if X made the last move, then one of these elements must have the value −1, and so the minimum total for the line will be −2 — i.e., −1, −1, 0. The maximum total will be +1, i.e., −1, +1, +1 — see table 2. The only important entry in table 2 is that where the

Total	Cells filled 1	2	3
+1	–	–	–
0	–	–	–
-1	opX	–	–
-2	–	psX	–

Note "X" is −1 "O" is +1

Table 2.

total equals −2. If this is the case, we check to see if the blank square has two openings for O.

To answer the question of how to plan an intelligent noughts and crosses program I have started from the bottom up, with the "move". The "move" is the essence, and only by seeing how moves are combined from the bottom upwards can we begin to evolve a strategy, and tactics.

A program is more than just a set of ideas about how to perform manipulations — a communication has to be made between the program and the user, and we need means of input and of output. In the planning of the strategy, to start at the top — the program — and to go from the top down to the bottom — the move — would be less than useful. It makes to go from the idea of the program down to its actual set of statements.

The outline sequence of control in intelligent noughts and crosses can be shown as:

```
1     INITIALISATION
1.5   LOOP up to nine times
2     IF O, INPUT, CHECK WIN
3     IF X, DECIDE MOVE, CHECK WIN
4     DISPLAY
5     IF WIN FLAGGED exit to 6.1
5.5   ENDLOOP
6     DRAW and end
6.1   WIN and end
```

and 2 can be expanded as

```
2.1   RECALCULATE STATE X and STATE O
2.2   INPUT CO-ORDINATE, IF NOT LEGAL
      REPEAT
2.3   STORE CO-ORDINATE
2.4   IF SQUARE PSO, FLAG WIN FOR O
```

so can 3

```
3.1   SAME AS 2.1
3.2   IF PSX, THEN CHOOSE, ENTER CO-
      ORDINATE, FLAG WIN FOR "X", ELSE
3.3   IF PSO, THEN CHOOSE, ENTER CO-
      ORDINATE, ELSE
3.4   CHOOSE SQUARE, ENTER CO-
      ORDINATE
```

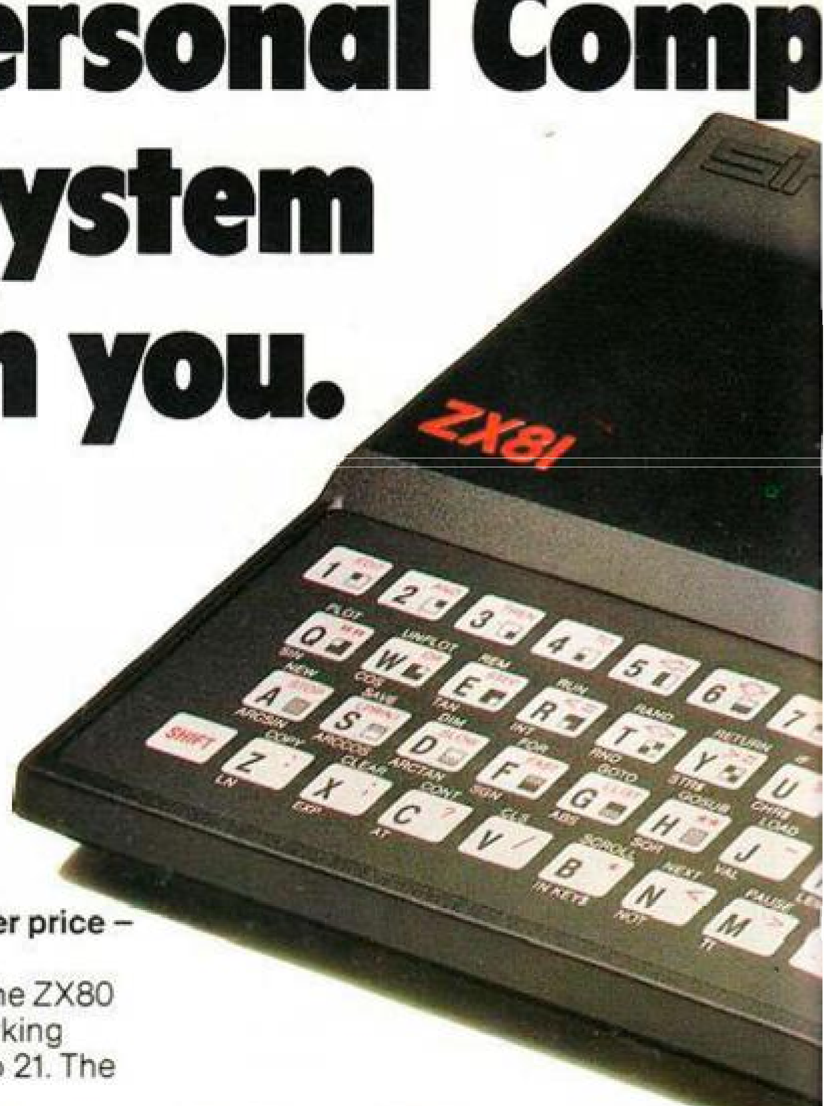
Section 3.4 with expansion becomes

```
3.4.1 FIND SQUARE FOR WHICH OPX IS
      GREATEST
3.4.2 FOR MOVE AHEAD, IF OPX HAS TWO
      OPSs THEN SET CHOSEN SQUARE TO
      ZERO OPXs, GOTO 3.4.1, ELSE
3.4.3 ENTER CO-ORDINATE
```

The program is to be written in Atom Basic, but could easily be for other micros. ■

sinclair ZX81 PERSONAL COMPUTER

Sinclair ZX81 Personal Comp[uter] the heart of a system that grows with you.



1980 saw a genuine breakthrough – the Sinclair ZX80, world's first complete personal computer for under £100. Not surprisingly, over 50,000 were sold.

In March 1981, the Sinclair lead increased dramatically. For just £69.95 the Sinclair ZX81 offers even more advanced facilities at an even lower price. Initially, even we were surprised by the demand – over 50,000 in the first 3 months!

Today, the Sinclair ZX81 is the heart of a computer system. You can add 16-times more memory with the ZX RAM pack. The ZX Printer offers an unbeatable combination of performance and price. And the ZX Software library is growing every day.

Lower price: higher capability
With the ZX81, it's still very simple to teach yourself computing, but the ZX81 packs even greater working capability than the ZX80.

It uses the same micro-processor, but incorporates a new, more powerful 8K BASIC ROM – the 'trained intelligence' of the computer. This chip works in decimals, handles logs and trig, allows you to plot graphs, and builds up animated displays.

And the ZX81 incorporates other operation refinements – the facility to load and save named programs on cassette, for example, *and* to drive the new ZX Printer.

New BASIC manual

Every ZX81 comes with a comprehensive, specially-written manual – a complete course in BASIC programming, from first principles to complex programs.

Kit: £49.95

Higher specification, lower price – how's it done?
Quite simply, by design. The ZX80 reduced the chips in a working computer from 40 or so, to 21. The ZX81 reduces the 21 to 4!

The secret lies in a totally new master chip. Designed by Sinclair and custom-built in Britain, this unique chip replaces 18 chips from the ZX80!

New, improved specification
- Z80A micro-processor – new faster version of the famous Z80 chip, widely recognised as the best ever made.
- Unique 'one-touch' key word entry: the ZX81 eliminates a great deal of tiresome typing. Key words (RUN, LIST, PRINT, etc.) have their own single-key entry.
- Unique syntax-check and report codes identify programming errors immediately.
- Full range of mathematical and scientific functions accurate to eight decimal places.
- Graph-drawing and animated-display facilities.
- Multi-dimensional string and numerical arrays.
- Up to 26 FOR/NEXT loops.
- Randomise function – useful for games as well as serious applications.
- Cassette LOAD and SAVE with named programs.
- 1K-byte RAM expandable to 16K bytes with Sinclair RAM pack.
- Able to drive the new Sinclair printer.
- Advanced 4-chip design: micro-processor, ROM, RAM, plus master chip – unique, custom-built chip replacing 18 ZX80 chips.

Built: £69.95

Kit or built – it's up to you!
You'll be surprised how easy the ZX81 kit is to build: just four chips to assemble (plus, of course the other discrete components) – a few hours' work with a fine-tipped soldering iron. And you may already have a suitable mains adaptor – 600 mA at 9 V DC nominal unregulated (supplied with built version).

Kit and built versions come complete with all leads to connect to your TV (colour or black and white) and cassette recorder.

uter-

Available now- the ZX Printer for only £49.95

Designed exclusively for use with the ZX81 (and ZX80 with 8K BASIC ROM), the printer offers full alphanumerics *and* highly sophisticated graphics.

A special feature is COPY, which prints out exactly what is on the whole TV screen without the need for further intructions.

At last you can have a hard copy of your program listings – particularly useful when writing or editing programs.

And of course you can print out your results for permanent records or sending to a friend.

Printing speed is 50 characters per second, with 32 characters per line and 9 lines per vertical inch.

The ZX Printer connects to the rear of your computer – using a stackable connector so you *can* plug in a RAM pack as well. A roll of paper (65 ft long x 4 in wide) is supplied, along with full instructions.

16K-byte RAM pack for massive add-on memory.

Designed as a complete module to fit your Sinclair ZX80 or ZX81, the RAM pack simply plugs into the existing expansion port at the rear of the computer to multiply your data/program storage by 16!

Use it for long and complex programs or as a personal database. Yet it costs as little as half the price of competitive additional memory.

With the RAM pack, you can also run some of the more sophisticated ZX Software – the Business & Household management systems for example.

sinclair ZX81

6 Kings Parade, Cambridge, Cambs., CB2 1SN.
Tel: (0276) 66104 & 21282.

How to order your ZX81

BY PHONE – Access, Barclaycard or Trustcard holders can call 01-200 0200 for personal attention 24 hours a day, every day.
BY FREEPOST – use the no-stamp-needed coupon below. You can pay by cheque, postal order, Access, Barclaycard or Trustcard.
EITHER WAY – please allow up to 28 days for delivery. And there's a 14-day money-back option. We want you to be satisfied beyond doubt – and we have no doubt that you will be.

To: Sinclair Research, FREEPOST, Camberley, Surrey, GU15 3BR. **Order**

Qty	Item	Code	Item price £	Total £
	Sinclair ZX81 Personal Computer kit(s). Price includes ZX81 BASIC manual, excludes mains adaptor.	12	49.95	
	Ready-assembled Sinclair ZX81 Personal Computer(s). Price includes ZX81 BASIC manual and mains adaptor.	11	69.95	
	Mains Adaptor(s) (600 mA at 9 V DC nominal unregulated).	10	8.95	
	16K-BYTE RAM pack.	18	49.95	
	Sinclair ZX Printer.	27	49.95	
	8K BASIC ROM to fit ZX80.	17	19.95	
	Post and Packing.			2.95

☐ Please tick if you require a VAT receipt TOTAL £______

*I enclose a cheque/postal order payable to Sinclair Research Ltd, for £______

*Please charge to my Access/Barclaycard/Trustcard account no.

*Please delete/complete as applicable.

Please print.

Name: Mr/Mrs/Miss

Address:

FREEPOST – no stamp needed. YOC 02

How the ZX81 compares with other personal computers

SYSTEM IDENTIFICATION		ZX81	ZX80	ACORN ATOM	APPLE II PLUS	PET 2001	TRS 80 LEVEL I	TRS 80 LEVEL II
ROM		8K	4K	8K	8K	14K	4K	12K
GUIDE PRICE	Basic unit – inc. VAT	£70	£100	£175	£630	£435	£290	£375
	Unit plus 16K RAM (*12K RAM)	£120	£150	£285*	£630	£530	£360	£375
COMMANDS	LIST, LOAD, NEW, RUN, SAVE	●	●	●	●	●	●	●
STATEMENTS	PRINT, INPUT, LET, GOTO, GOSUB/RETURN, FOR/NEXT IF/THEN	●	●	●	●	●	●	●
	STEP	●		●	●	●	●	●
	TAB	●			●	●	●	●
ARITHMETIC	ABS, RND	●	●	●	●	●	●	●
FUNCTIONS	INT	●			●	●	●	●
	ATN, COS, EXP, LOG, SGN, SIN, SQR, TAN	●			●	●		●
	ARCSIN, ARCOS	●						
STRING	CHR$	●	●		●	●		●
FUNCTIONS	LEN	●		●	●	●		●
	ASC(CODE), STR$, VAL, INKEY$	●				●		●
NUMBERS	FLOATING PT $\pm 10^{\pm 38}$	●			●	●	●	●
	INTEGERS		●	●	●	●		●
NUMERIC	A-Z			●			●	
VARIABLES	AA-ZØ				●	●		●
	An-Zn, n = any alphanumeric string	●	●					
STRING	A$ & B$						●	
VARIABLES	A$ to Z$	●	●	●				
	An$ to Zn$ n = any alphanumeric character				●	●		●
NUMERIC	SINGLE DIMENSIONAL		●	●			●	
ARRAYS	MULTI DIMENSIONAL	●			●	●		●
DISPLAY	ROWS	24	24	16	24	25	16	16
	COLUMNS	32	32	32	40	40	64	64
	LOW RES GRAPHICS (<7000 pixels)	●	●	●	●	●	●	●
	HI RES GRAPHICS (>40000 pixels)			●	●			
SPECIAL	USR (CALL, LINK)	●	●	●	●	●		●
FEATURES	PEEK, POKE (OR EQUIV)	●	●	●	●	●		●

Sinclair software on cassette.

The unprecedented popularity of the ZX Series of Sinclair Personal Computers has generated a large volume of programs written by users.

Sinclair has undertaken to publish the most elegant of these on pre-recorded cassettes. Each program is carefully vetted for interest and quality, and then grouped with others to form single-subject cassettes.

Software currently available includes games, junior education, and business/household management systems. You'll receive a Sinclair ZX Software catalogue with your ZX81 – or see our separate advertisement in this magazine.

The ultimate course in ZX81 BASIC programming.

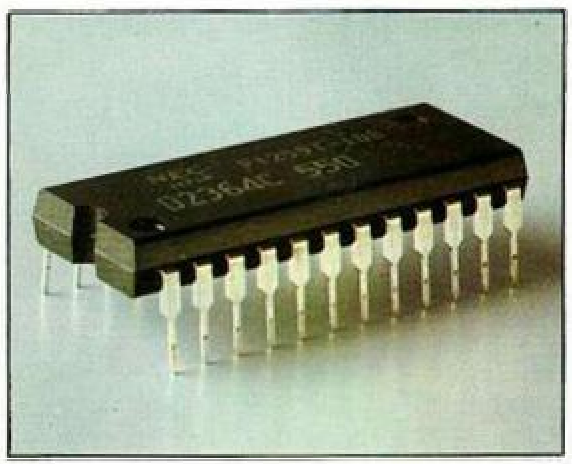
Some people prefer to learn their programming from books. For them, the ZX81 BASIC manual is ideal.

But many have expressed a preference to learn *on* the machine, *through* the machine. Hence the new cassette-based ZX81 Learning Lab.

The package comprises a 160-page manual and 8 cassettes. 20 programs, each demonstrating a particular aspect of ZX81 programming, are spread over 6 of the cassettes. The other two are blank practice cassettes.

Full details with your Sinclair ZX81.

If you own a Sinclair ZX80...



The new 8K BASIC ROM used in the Sinclair ZX81 is available to ZX80 owners as a drop-in replacement chip. (Complete with new keyboard template and operating manual.)

With the exception of animated graphics, all the advanced features of the ZX81 are now available on your ZX80 – including the ability to drive the Sinclair ZX Printer.

sinclair
ZX81

6 Kings Parade, Cambridge, Cambs., CB2 1SN.
Tel: (0276) 66104 & 21282.

MICROCOMPUTERS AT LASKYS

10 Shops Nationwide

WORLD BEATING MICRO'S

Birmingham

19/21 Corporation Street, Birmingham, B2 4LP.
Tel: 021-632 6303.
Manager: Peter Stallard. 300 yards from Bullring Centre.

Bristol

16/20 Penn Street, Bristol, BS1 3AN. Tel: 0272 20421.
Between Holiday Inn and C & A.

Chester

The Forum, Northgate Street, Chester, CH1 2BZ.
Tel: 0244 317667.
Manager: Jeremy Ashcroft. Next to the Town Hall.

Edinburgh

4 St. James Centre, Edinburgh, EH1 3SR. Tel: 031-556 6217.
Manager: Colin Draper.
East end of Prices Street, St. James Centre.

Preston

1/4 Guildhall Arcade, Preston, PR1 1HR.
Tel: 0772 59264.
Manager: Jim Comisky. Directly under Guild Hall.

Manchester

12/14 St. Mary's Gate, Market Street, Manchester, M1 1PX.
Tel: 061-832 6087.
Manager: Lesly Jacobs. Corner of Deansgate.

Glasgow

22/24 West Nile Street, Glasgow, G7 2PF. Tel: 041-226 3349.
Manager: David Livingstone.
Between Buchannan Street and Central Station.

Sheffield

58 Leopold Street, Sheffield, S1 2GZ. Tel: 0742 750971.
Manager: Justin Rowles. Top of the Moor, opposite Town Hall.

Liverpool

33 Dale Street, Liverpool, L2 2HF. Tel: 051-236 2828.
Manager: Mark Butler.
Between the Town Hall and Magistrates Courts.

London

42 Tottenham Court Road, London, W1 9RD.
Tel: 01-636 0845. Manager: Vass Demosthenis.

Official Orders over £50 are welcome with normal 30 days credit extended to bona-fide commercial and government organisations.

Laskys, the retail division of the Ladbroke Group of Companies

MICROCOMPUTERS AT LASKYS

NEW
Now in stock the powerful Apple III and the exciting new Osborne!

Apple II

The worlds best selling micro offers unmatched flexibility. At our special prices.

Apple II + 48K
Nett: 675.00 Vat: 101.25 Total: 776.25

Disk Drive Plus Controller (3.3)
Nett: 375.00 Vat: 56.25 Total: 431.25

Disk Drive
Nett: 295.00 Vat: 44.25 Total: 339.25

Deservedly popular the MZ 80K comes with 48K of RAM, built in screen and cassette. – superb value.

MZ 80K Computer 48K
Nett: 347.00 Vat: 52.00 Total: 399.00

Atari 400

Ideal for the home with excellent games and education packages. This computer features colour and sound and is built tough to last.

400 16K Computer
Nett: 300.00 Vat: 45.00 Total: 345.00

410 Tape Recorder
Nett: 43.48 Vat: 6.52 Total: 50.00

Atari 800

The big brother of the 400 shares all the star features and has much more as well.

800 16K Computer
Nett: 560.87 Vat: 84.13 Total: 645.00

810 Disk Drive
Nett: 300.00 Vat: 45.00 Total: 345.00

822 Thermal Printer
Nett: 230.43 Vat: 34.57 Total: 265.00

Hewlett Packard HP 85

Scientific and technical professionals favour the HP 85, they are being joined by increasing numbers of business professionals.

HP 85 Computer
Nett: 2146.95 Vat: 322.04 Total: 2468.99

16K Memory Module
Nett: 194.88 Vat: 29.23 Total: 224.11

ROM Drawer
Nett: 29.73 Vat: 4.46 Total: 34.19

CRA

THE SOUND OF VIC

BY NICK HAMPSHIRE

THE VIC'S capabilities for sound effects and music-generation are controlled by five registers in the 6561. Four of the registers are associated with sound generation, the fifth controls the volume of the sound output. Each of the four sound-generation registers has an associated oscillator and the register contents determines the frequency of the oscillator output. The frequency is determined by varying the pulse width; the output from all four oscillators is a symmetrical square wave.

These outputs are combined to give the audio input to the TV display, where the sound is generated via the TV speaker. One of the four audio oscillators acts as a variable-frequency noise source and the other three generate a simple tone. The five control registers used are:

- **Audio oscillator 1** — control register 11, location 36874. Bits 0 to 6 control the frequency, bit 7 turns the oscillator on or off. The value 128 put in this register will produce the lowest frequency sound of any of the three audio oscillators.
- **Audio oscillator 2** — control register 12, location 36875. Bits 0 to 6 control the frequency, bit 7 turns the oscillator on or off. The base frequency for this oscillator is between that for audio oscillators 2 and 3.
- **Audio oscillator 3** — control register 13, location 36876. Bits 0 to 6 control the frequency, bit 7 turns the oscillator on or off. This has the highest base frequency of the three oscillators.
- **Noise generator** — control register 14, location 36877. Bits 0 to 6 control the base frequency of the noise generator, bit 7 turns it on or off. This is a pseudo white-noise generator, giving a random sequence of pulses with a frequency determined by the contents of the control register.
- **Volume control** — control register 15, location 36878. The volume of the composite audio signal produced when one or more of the four audio oscillators is turned on is controlled by bits 0 to 3.

Program 1 — Breaking waves.

```
 10 POKE36877,180
 20 FORL=1TO10
 30 D=INT(RND(1)*5)*50+50
 40 FORM=3TO15
 50 POKE36878,M
 60 FORN=1TOD
 70 NEXTN
 80 NEXTM
 90 FORM=15TO3STEP-1
100 POKE36878,M
110 FORN=1TOD
120 NEXTN
130 NEXTM
140 NEXTL
150 POKE36878,0
160 POKE36877,0
200 GOTO10
```

Program 2 — Galloping horse.

```
  5 FORX=15TO0STEP-1
  6 FORZ=1TO4
  7 A=60
 10 POKE36878,X
 20 POKE36876,230
 30 POKE36876,0
 40 FORQ=1TOA:NEXTQ
 50 POKE36876,230
 60 POKE36876,0
 70 FORQ=1TOA:NEXTQ
110 POKE36878,INT(X/2)
120 POKE36876,230
130 POKE36876,0
140 FORQ=1TOA:NEXTQ
150 POKE36876,230
160 POKE36876,0
170 FORQ=1TO4*A:NEXTQ
180 NEXTZ
190 NEXTX
```

Program 3 — Ringing telephone.

```
 10 POKE36878,15
 20 FORL=1TO5
 30 FORM=1TO50
 40 POKE36876,230
 50 FORN=1TO5
 60 NEXTN
 70 POKE36876,0
 80 NEXTM
 90 FORM=1TO3000
100 NEXTM
110 NEXTL
120 POKE36878,0
```

Program 4 — Grandfather clock.

```
  5 A=700
 10 POKE36878,15
 20 POKE36876,230
 30 POKE36876,0
 40 FORQ=1TOA:NEXTQ
 50 POKE36876,236
 60 POKE36876,0
 70 FORQ=1TOA:NEXTQ
 80 GOTO20
```

Each of the audio oscillators is capable of generating 128 frequencies and each oscillator is different, thus oscillator 1 can be described as a "base" sound generator, oscillator 2 as a "tenor" and 3 as a "soprano". The combined audio output has one of 16 volume levels.

The four sound generators can be used to create a wide range of sound effects for use in games programs; they can also be used to play music. Writing routines to create sound effects is simply a matter of experimentation. Try to analyse the required sound and then re-create it using a combination of the four audio oscillators and the volume control. This is demonstrated in some of the programs 1 to 5.

Using the audio generators on the 6561 to play music requires some thought, otherwise the result will sound very abrasive and not at all satisfactory. The first problem is that the square-wave output from the audio oscillators produces a rather unpleasant set of harmonics which gives the note a rough quality.

Only external electronics can change the shape of the wave-form, but by using two audio oscillators to produce the same note of frequencies an octave or two apart a more pleasing sound is produced.

The second problem is to generate the correct attack and decay for the instrument; this is done by changing the amplitude of the output during the generation of each note. These two ideas are illustrated in program 6 which plays scales and the sound resembles a piano.

The sound locations must be Poked with numbers between 128 and 255. The frequency rises as the number, with the exception of 255 which is a low frequency. Each tone location produces one voice. A zero in any byte will turn off that voice. The decimal codes given in table 1 produce an approximation of three octaves of the even-tempered musical scale. The scale is relative, not absolute concert pitch. This table lists the musical note and its respective Poke location.

To play a musical score requires a note table. This table contains each note in the score in the form of the value to be placed into the audio oscillator register and the duration of that note. ■

Table 1. Poke locations of musical notes.

Musical Note	Poke	Musical Note	Poke	Musical Note	Poke
C	128	C#	195	D	227
C#	134	D	198	D#	228
D	141	D#	201	E	230
D#	147	E	204	F	231
E	153	F	207	F#	232
F	159	F#	210	G	234
F#	164	G	213	G#	235
G	170	G#	215	A	236
G#	174	A	217	A#	237
A	179	A#	219	B	238
A#	183	B	221	C	239
B	187	C	223	C#	240
C	191	C#	225		

Program 5.

```
  5 REM *MAKES A SOUND LIKE THE SINGING
  6 REM *OF BIRDS
  7 REM
 10 POKE36878,15
 20 FORL=1TO20
 30 FORM=254TO240+INT(RND(1)*10)STEP-1
 40 POKE36876,M
 50 NEXTM
 60 POKE36876,0
 70 FOR M=0TOINT(RND(1)*100)+120
 80 NEXTM
 90 NEXTL
100 GOTO10
```

Program 6.

```
  1 REM *PLAYS A REPEATING OCTAVE SCALE
  2 REM *THE SOUND OF EACH NOTE DECAYS AND
  3 REM *THUS SOUNDS MORE LIKE A PIANO THAN
  4 REM *AN ELECTRIC ORGAN
  5 READA:IFA=100THEN150
 10 POKE36874,A
 20 POKE36875,A
 30 FORQ=15TO0STEP-1
 40 POKE36878,Q
 50 FORX=1TO50:NEXTX
 60 NEXTQ
100 GOTO5
150 RESTORE:GOTO5
200 DATA223,227,230,231,234,236,238,239
210 DATA239,238,236,234,231,230,227,223,100
```

GRAPHICS SMOOTH

Producing realistic graphics on the ZX-81 poses few programming problems. Yet if you then have to make those graphics move, you soon find yourself trapped in loops of undreamt-of complexity. John Watson's techniques for animation use machine-code subroutines, are easy to handle and produce far more polished results than their Basic counterparts.

THE ZX-81, like many other microcomputers, can print a wide range of graphics symbols as well as the conventional characters. Yet if you want to draw graphics quickly or to move complicated illustrations around the screen, you will find the task hampered by the slowness of Basic.

Anyone who has constructed a picture on the ZX-81 using more than a few characters will have noticed the way that Basic builds the picture from the top downwards. If the screen is cleared, the picture also disappears from the top downwards.

The overall effect is rather like an invisible hand rapidly piling up and then demolishing building bricks. Space Invaders seem to build and re-build themselves in different positions on the screen rather than move about as they do in the arcade games.

It is for this reason that most games involving animated graphics are written in machine code. Anyone who has written extensive machine-code programs will know what a marathon chore it can be — even using an assembler. As an alternative to writing the whole program in machine code, I decided to write a short, general-purpose machine-code routine for producing drawings on the screen. Using the subroutine, anyone can write a Basic program with animated graphics.

Figure 4. The Basic program begins at line 20. The odd appearance of line 10 is caused by the ROM interpreting the machine-code subroutine as best it can.

```
10 REM COPY ? █RND GOSUB ?▄RND
E£RND; FAST GOSUB ?▌RND.<AT ?U
▙RND RETURN  C▘/▖Y /▘ ,▘?I▘█COS
 <? FAST, A ?▒ FAST AT LPRINT
</ THEN -▄3▄▘▛▛▛1▛▀▛▀█1
▛▛▛█▘█2█ ▘2▛▘▘▀▜
  20 PRINT AT 21,0;"-----------
----------------------"
  30 POKE 16515,3
  40 POKE 16516,0
  50 RAND USR 16519
  60 FOR T=1 TO 35
  70 FOR D=1 TO 12
  80 NEXT D
  90 POKE 16514,0
 100 LET E=34
 105 IF T<18 THEN LET E=1
 110 IF T>26 THEN LET E=33
 120 LET A=(PEEK 16515+256*PEEK
16516)+E
 130 LET B=INT (A/256)
 140 LET C=A-INT (A/256)*256
 150 RAND USR 16519
 160 POKE 16514,255
 170 POKE 16516,B
 180 POKE 16515,C
 190 RAND USR 16519
 200 NEXT T
 300 FOR D=1 TO 200
 310 NEXT D
 320 POKE 16514,0
 330 RAND USR 16519
 340 GOTO 20
```

The ZX-81 fitted with the 16K RAM pack produces a screen which is effectively memory-mapped. Mark a sheet of squared paper with 33 squares wide by 24 squares deep. This corresponds to the printing positions on the screen. Each line on the screen has 32 characters, plus an invisible end-of-line character at the right-hand end of each line. Screen location number 1 is in the top left-hand corner, and the bottom right-hand location is 792.

The only slightly unusual feature of the Sinclair memory-mapped screen is that it moves around in memory. However, that is not too much of a problem since you can locate the beginning of the display file by looking in memory locations 16396/7. This location actually gives the address of the byte immediately before the first screen position. The first screen position can therefore be found with:

(PEEK 16396 + 256*PEEK 16397) + 1

and the last screen position with:

(PEEK 16396 + 256*PEEK 16397) + 791

One word of warning: if you Poke the end-of-line character, you will crash the program; if you want to avoid doing this, have the Basic check the screen location to make sure it is not evenly divisible by 33. Also ensure that it is not 0 — Poking the character before the first position in the screen will also cause a crash.

The only way to print graphics characters rapidly is with machine code. Rather than go through the immense task of writing a machine-code program for every graphics game, a subroutine can be used. This machine-code subroutine consists of only 56 bytes, and the subroutine and its data lines are housed in a 10-Rem statement.

Figure 1 shows how the subroutine in Z-80 mnemonics is tucked into Rem statements ready for treatment by the Bug-Byte ZXAS assembler. Alternatively, figure 2 gives a hexadecimal dump of the program. Note that the program itself finishes at 40B9 — everything from 40BA onwards is data.

The program begins with five control bytes —4082 hexadecimal equals 16,514 in decimal. If location 16514 contains a number other than zero, the program will draw a picture from the data. If 16514 contains zero, the program erases the drawing using the same data.

Locations 16515 and 16516 are Poked with a number in the range 1 to 79 which is the screen position for the start of the drawing. Locations 16517 and 16518 contain the address of the start of the data for the drawing. These features enable you to put in several groups of data, and draw or erase at any position on the screen. Even complex

ANIMATION FOR GAMES

```
100 REM <
101 REM LD DE.(16515);LD HL.(16396);ADD HL.DE;PUSH HL
102 REM LD DE.(16517);LD A.(DE);INC DE;POP BC
103 REM LD H.A;:L1LD A.(16514);CP 0;JR Z.L2;JR L3;:L2LD
A.0;JRL4
104 REM :L3LD A. (DE);:L4LD (BC).A;LD A.H;LD L.1;SUB L
105 REM RET Z;INC DE;LD H.A;PUSH HL;LD A.(DE);LD H.0;LD
L.A;ADD HL.BC;PUSH HL;POP BC;POP HL;INC DE;JR L1
106 REM >
```

Figure 1. The machine-code subroutine, listed as assembly-language mnemonics.

drawings appear and disappear from the screen instantly.

To generate, say, a Space Invader, Poke the data location, the position on the screen, and the draw/erase control byte. Then call the subroutine with USR. For the demonstration program, it looks like this:

```
100 POKE 16514,255    { Selects "draw", as opposed to "erase"
110 POKE 16515,03     { Start drawing at the
120 POKE 16516,0      { third position on the top line
130 POKE 16517,186    { Data starts at 40BA —
140 POKE 16518,64     { 64 decimal = 40 hexadecimal
150 RAND USR 16519    { 186 decimal = BA hexadecimal
```

Using this, the picture appears: to erase it again, you need only change one location:

```
200 POKE 16514,0      Select "erase"
210 RAND USR 16519    All other parameters
                      are the same, so call
                      the subroutine
```

You will see that it vanishes again. To redraw it one step to the right:

```
300 POKE 16514,255    Select "draw"
310 POKE 16515,04     Start drawing at the
                      fourth position on the
                      top line
320 RAND USR 16519
```

The data for the machine-code program starts at 40BA hexadecimal which is 16,570 decimal. The data determines how the drawing is formed — for an explanation we need to return to the 33-by-24 grid. Start by drawing the picture you want. You can use any printable characters or graphics symbols. Count the number of characters to be printed. This, in hexadecimal, is the first byte of the data. The maximum number of characters in a drawing is 128, or 80 hexadecimal — enough for the most ambitious lunar lander.

```
1 REM HEX DUMP OF THE QUICK
DRAW PROGRAM. THE DATA STARTS
AT 40BA.

4082:FF 0E 00 BA 40 ED 5B 83
408A:40 2A 0C 40 19 E5 ED 5B
4092:85 40 1A 13 C1 67 3A 82
409A:40 FE 00 28 02 18 04 3E
40A2:00 18 01 1A 02 7C 2E 01
40AA:95 C8 13 67 E5 1A 26 00
40B2:6F 09 E5 C1 E1 13 18 DE
40BA:16 83 1F 83 01 80 01 80
40C2:01 80 01 83 1D 80 01 0A
40CA:01 80 01 0A 01 80 1D 80
40D2:01 80 01 80 01 80 01 80
40DA:1E 05 02 85 1E 07 01 01
40E2:02 02 01 84
```

Figure 2. Hex dump and data, to be entered using the loader program in figure 3.

Start with the top line of the drawing and take the rightmost character. This is the starting position for the drawing, and is the point referred to in the control location in the main subroutine at 16515 and 16516. The second byte of data is the hexadecimal code for that character — the codes are listed on pages 181-187 of the Sinclair ZX-81 manual.

The third byte of data is the displacement for the second character to be printed. All the displacements are positive. Thus a displacement of one means that the next character to be printed will be the one immediately to the right of the first. A displacement of two means that the next print position is two places to the right. Since the displacements are mapped 33 characters to the line, a displacement of 33 — 21 hexadecimal — would mean that the next print position is immediately under the first one. To put a horizontal black bar on the screen, consisting of four inverse spaces, the data line used would be:

04,80,01,80,01,80,01,80

The first byte records that there are four characters. The second byte is the code for inverse space, 80 hexadecimal, which is what is to be printed at the first position; next is the displacement to the right, one place; then the next character, also 80 hexadecimal, and so on.

A vertical bar consisting of just three inverse spaces would be:

03,80,21,80,21,80

Practise writing the data for a few single illustrations before progressing to anything too complicated.

The machine-code subroutine and the data are contained in a Rem statement which is the first line in the program. The first step is to enter a dummy Rem statement:

10 REM XXXXXXX

with the number of Xs, or any other character you like, equal to the number of bytes in the program — 56 — plus data. For the demonstration program, that is 100 bytes.

The program in figure 3 is used to load the hexadecimal code into the Rem statement. Copy the codes shown in the hexadecimal dump — figure 2. When all the code is entered, type "S" to stop the loading and then List to see what has happened to the program. If all has gone well, the Rem statement will now look very peculiar indeed. Now that the code is entered, you can delete the hexadecimal loader program — all the lines in the listing except line 10 — or you can leave it where it is for future use.

The Basic program shown in figure 4 provides a demonstration of the use of the subroutine. When you run this program it will produce a picture. In it, the alien appears in the top-left corner of the screen, cruises horizontally to the right for a while, descends at an angle, and finally lands vertically in the bottom right-hand corner.

The graphic images pop into place almost instantly. The program makes the images move by the usual technique of erasing an image and then replacing it slightly further along the screen; erasing that one, and

```
500 LET M=16514
505 REM TO LOAD M/C "DRAW"
PROGRAM DATA ONLY, SUBSTITUTE
:500 LET M=16570
510 SCROLL
520 PRINT M,
524 INPUT H$
525 PRINT H$
530 IF H$="S" THEN STOP
550 POKE M, (16*(CODE H$(1)-
28)+(CODE H$(2)-28))
560 LET M=M+1
570 GOTO 510
```

Figure 3. Hex loader program.

replacing it with another, and so on. There is still a slight flicker of the image, caused by the delay between erasing one picture and drawing the next.

This delay occurs while Basic is changing the control values. The flicker can be minimised by doing as little as possible in Basic between "erase" and "redraw". This program shows the capabilities of the subroutine, but to produce graphics which move perfectly smoothly, one of two alternative techniques can be used.

The simplest technique is to surround the image of the moving object with a halo of spaces. There is no reason, of course, why the

(continued on page 55)

(continued from page 53)
subroutine should not be used to print spaces as well as characters. Provided you then move the drawing only one step at a time in any direction, it will automatically erase all traces of its previous incarnation as it goes. Data to produce this kind of animation is given in figure 5. It is 84 bytes long, which is the main problem with this technique. The dummy Rem statement will need to be 140 bytes long. Use the same Basic demonstration program but make the following changes:

```
30 POKE 16515,1
90 POKE 16514,255
105 IF T<17 THEN LET E=1
```

and delete line 190.

The second technique is to use different data blocks to erase and redraw only the changed parts of a moving drawing. The simplest example is a moving rectangle. If it moves vertically, you need only erase the trailing edge and redraw a new front edge. The sides and the middle can be left alone, and the overall effect is a much smoother movement than you would achieve by erasing and redrawing the whole object. Multiple data blocks can also be used to create the various stages of an explosion, for example.

You must be methodical in producing the data for the drawing, but this is a small price to pay for the improvement in the program. Finally, watch out for accidental Poking of the forbidden end-of-line characters in the display file. If you want to see the effect of doing this, change line 110 of the Basic program to read:

```
110 IF T> 30 THEN LET E=33
```

and watch the program crash. ■

```
1 REM    HEX DUMP OF REVISED
DATA FOR SELF-ERASING MOVING
DRAWING. THE DATA STARTS AT
40BA AND OCCUPIES 84 BYTES.

40BA:2A 00 01 00 01 00 01 00
40C2:01 00 01 00 1C 00 01 00
40CA:01 00 01 83 01 00 01 00
40D2:1C 00 01 83 01 80 01 80
40DA:01 80 01 83 1C 00 01 80
40E2:01 0A 01 80 01 0A 01 80
40EA:1C 00 01 80 01 80 01 80
40F2:01 80 01 80 1C 00 01 00
40FA:01 05 01 00 01 85 01 00
4102:1C 00 01 07 01 01 01 00
410A:01 02 01 84
```

Figure 5. Revised data for improving the smoothness of animation.

PROJECT

ELECTRONIC SCANNING

This month, the role of the micro in electronic scanning falls under John Dawson's blistering gaze.

MOST ROBOTS IN science fiction stories can see. There is good sense behind the fiction — the ability to acquire information simply by looking at a subject is an invaluable sense. Human beings see using a parallel bus in which hundreds of thousands of neurons carry information simultaneously from the retina, the back surface of the eye, down the optic nerve to a certain part of the brain. There the incoming information is processed to produce stereoscopic colour images which can be used as a basis for making decisions.

Many electro-mechanical and electronic systems have been devised for transmitting a picture from one place to another using a single channel for the information flow. In the 1930s, we very nearly had an electro-mechanical television system in the U.K. and photographs have been sent by transmitting an electrical signal from one place to another by press agencies and the police for many years. The Germans developed the Hellschreiber for transmitting a facsimile document by radio. Satellite pictures, both of weather conditions on the Earth and from the Solar System are transmitted electronically.

In all these applications, serial transmission is used to carry the information contained in the picture one part at a time down a single bus. Then it recreates the original picture by storing each part in its correct order for simultaneous presentation to the view at the other end.

Figure 1 illustrates the basic principle of picture transmission used by every system from domestic colour television to the simplest electro-mechanical document fascimile systems. The picture is scanned starting in one corner and moving in an orderly sequence until each element has been covered.

Image enhancement

As the picture is scanned the intensity of light is measured. The light may be in any part of the spectrum from the ultra-violet region through to the very long infra-red part of the spectrum used for thermography in hospitals. It is transmitted either in an analogue form to the receiving device for re-creation as a picture, or as a series of digital numbers.

The signal may be processed on its route to the receiving device. For example, malignant tumours tend to have a richer blood supply than normal tissue and the increased flow of hot blood from the core of the body tends to create a local warmth in the malignant tissue.

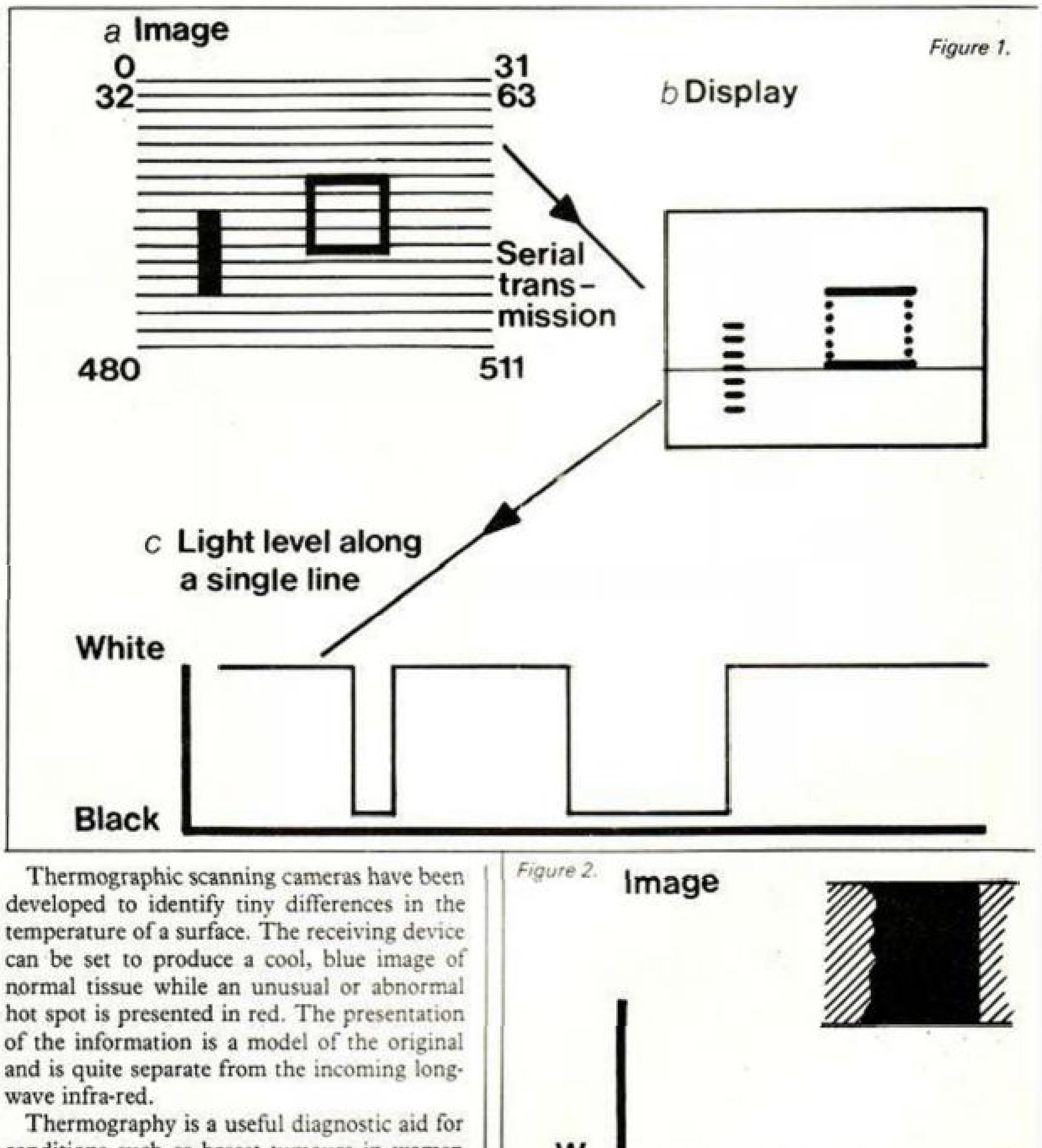


Figure 1.

Thermographic scanning cameras have been developed to identify tiny differences in the temperature of a surface. The receiving device can be set to produce a cool, blue image of normal tissue while an unusual or abnormal hot spot is presented in red. The presentation of the information is a model of the original and is quite separate from the incoming long-wave infra-red.

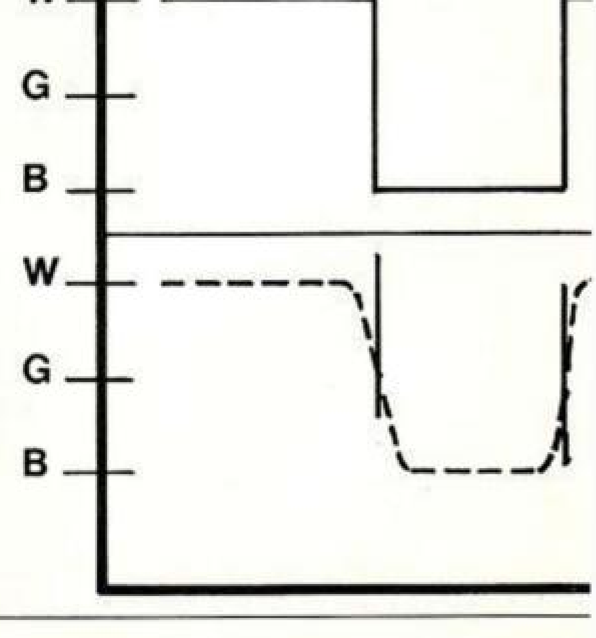
Thermography is a useful diagnostic aid for conditions such as breast tumours in women and for the analysis of heat flows in industry.

If the information picked up by the sensory device is turned into numbers using an analogue-to-digital converter, it becomes possible to process the information in a computer to improve the contrast of a picture, to remove interference or noise in the image, and to search for objects of a particular size or luminance.

Figure 2 is a simple illustration of how something like a Schmitt trigger in software can be used to increase the apparent resolution of a picture by digital processing. The fuzzy image at the top has a gradual transition from white, through darkening shades of grey to black. The intensity of the light reflected from that image is shown as the dotted line in the diagram.

If that information is captured by a

Figure 2.



WITH MICROS

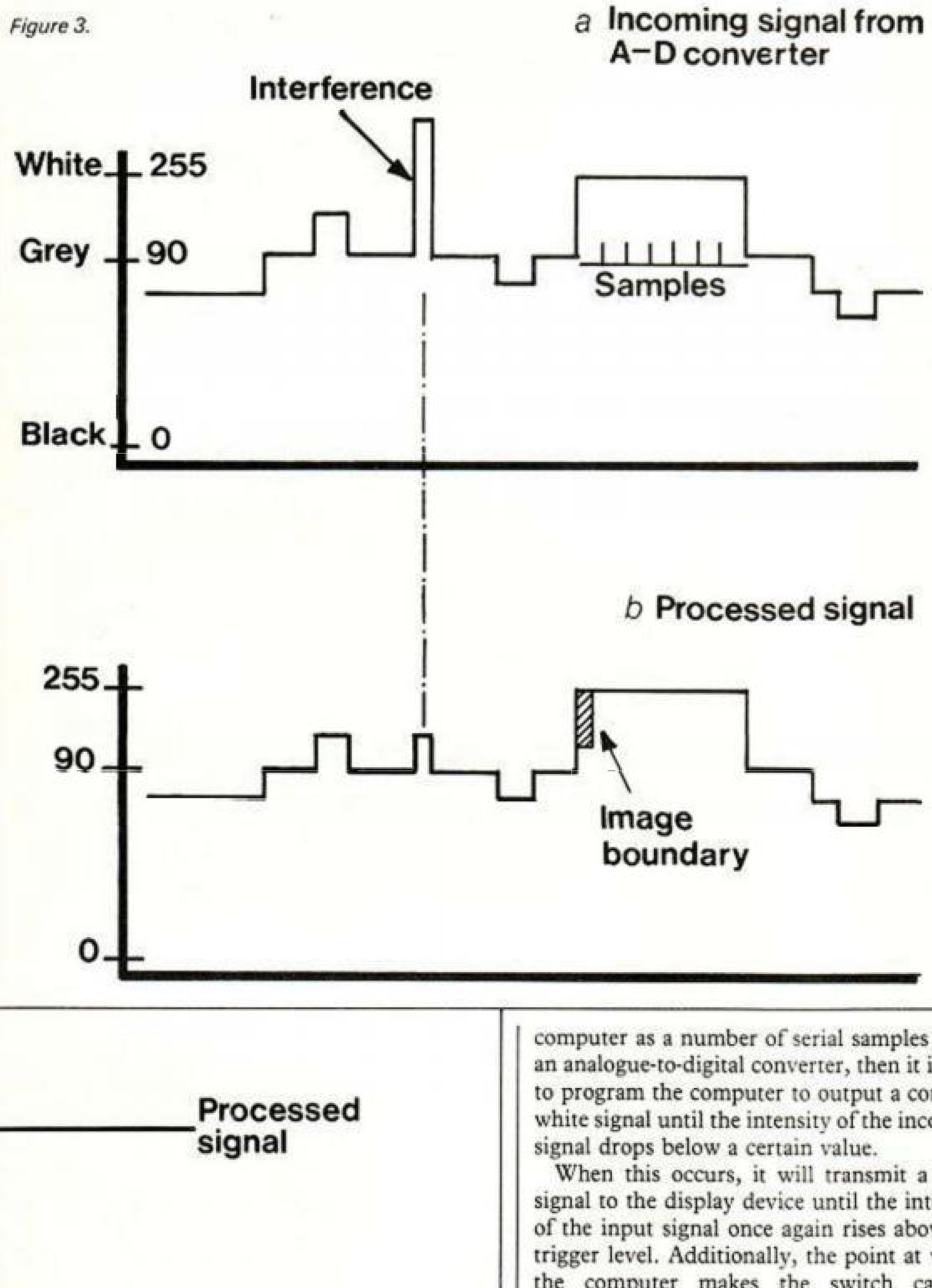
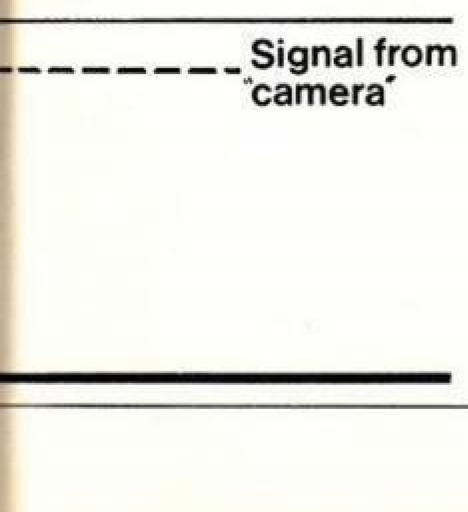


Figure 3.



computer as a number of serial samples using an analogue-to-digital converter, then it is easy to program the computer to output a constant white signal until the intensity of the incoming signal drops below a certain value.

When this occurs, it will transmit a black signal to the display device until the intensity of the input signal once again rises above the trigger level. Additionally, the point at which the computer makes the switch can be adjusted from the program or from the keyboard to highlight certain tone levels.

If you have used a portable television or if you live in a bad-reception area, you have probably noticed interference on the screen. Interference on the old VHF 405-line service caused white speckling on the picture. The new 625-line transmissions use negative-going modulation of the video signal so that a burst of energy caused by interference will tend to produce a black rather than a white spot. The interference consists of very short pulses of an unusually high amplitude — figure 3a.

To discriminate against either interference of this kind or small objects in the field of view of a scanning camera is comparatively easy using a computer. If the computer compares each sample from the analogue-to-digital converter with the previous sample, it can refuse to allow the size of the sample to change by more than a certain amount in one step.

For example, if the first sample has an amplitude of 90 or mid-grey on our arbitrary scale — figure 3b — then the computer may be programmed so that the next sample may change by only plus or minus 10 — that is, either 80 or 100.

The fact that the second sample is measured by the analogue-to-digital converter as having an intensity of 250 is ignored by the computer unless the third sample is also 250 +/- 10 in which case the two samples are accepted as a true part of the image being scanned.

The process can, of course, be extended, so that the computer will maintain the black level until five or 10 or 12 samples have consistently shown a higher white level. If information is stored in an array in the computer, it is possible to perform processing which will restore retrospectively the original boundary of the object from the first white-level sample.

Scanning the object

In other words, if the width of the object along the scanning line is insufficient to satisfy the limits programmed within the computer, it is rejected; otherwise, it will be presented in its correct topographical position. By comparing corresponding areas of adjacent lines it is possible to repeat the process in two dimensions. It should also become comparatively easy to identify lines in the image which are straight or which vary according to a pattern stored in the computer.

If you want to scan a picture, you will need, first of all, a lens to focus the image at which you are looking. Baird used a rotating disc with a spiral set of holes to break up the image into serial samples. The position is very simple: either you move the photocell which will convert the light intensity into an electrical signal, or you keep the photocell stationary and move the image.

Figure 4 is an outline illustration of a scanning camera which moves the image formed by the lens over a stationary photocell. The mirror has to move in two dimensions in this design and it may be easier to move the mirror in one plane and the photocell in the other.

For example, if the mirror is rotated up and down to provide the vertical component of the scan, the photocell could be mounted on an arm and swung horizontally from one side of the image formed by the lens to the other to give the other component.

The radio-control servos described in the August/September 1981 article are well suited to this application provided that the 270° rotation of the servo-output lever can be reduced to whatever field of view you require for the system. This could be done using cams to press on the mirror and the photocell arm.

The mathematics for calculating the performance limits for acquiring and building up a simple image are relatively straightforward.

(continued on next page)

(continued from previous page)

Suppose, for example, that you decide to display the picture gathered from the electronic-scanning camera on the VDU that you use with your Microtan computer. The Microtan screen consists of 16 lines each containing 32 elements, a total of 512 elements.

Suppose that you would like to acquire one complete picture or frame of information each second, then the maximum frequency to be handled by the system occurs when alternative picture elements are light and dark. One cycle consists of the time between two identical points on a wave-form which consists in this case of two elements. In other words, the maximum frequency at which the system is required to work is:

$512/2 = 206$Hz or cycles per second

If the image scanned by the camera has broader lines in which more than one element of the display will be set to either a white level or a black level, then the frequency of the signal being processed by the system is lower.

If you wrote the software to present an image using the 64-by-64 graphics on the Microtan VDU, the screen would contain 4,096 elements. A picture of the highest resolution of which the system was capable, displayed in one second, would require the system to handle a frequency of:

$4,096/2 = 2,048$ Hz

In either case, the speed at which the mirror must be moved in its horizontal and vertical axes remains tied to the time in which a picture is to be acquired. If you allow one second for the mirror to complete all its movements, then each horizontal line requires the mirror to move through the angle of field in which you will acquire 32 samples in 62.5ms. if the mirror can be brought back instantly to the beginning of the next line. This, of course, is impossible owing to the mechanical inertia of the system.

Cogent details

Probably the simplest method of moving the mirror in two dimensions is to scan the first line from left to right, set the vertical rotation of the mirror for the second line, and then scan that line from right to left putting the data from that scanned line on to the computer VDU in reverse order.

At the end of the second line, the mirror will then be in the correct lateral position for the start of the third line. A realistic estimate using radio-control servos might be:

16×0.5 seconds	lateral scan
15×0.05 seconds	vertical movement of mirror
1×0.5 seconds	restore mirror to start position

total 9.30 seconds

I think the most fascinating point to be learnt from the idea of moving a mirror to bring an image across a detector is that it should be possible to alter the focal length of the system by changing the speed with which the samples are taken.

Suppose that the computer gathers 100 samples in the course of scanning one line and suppose that during the time that it takes to gather 100 samples the mirror is moved through a right angle. The picture displayed on an oscilloscope or TV screen will show a field of view of 90°.

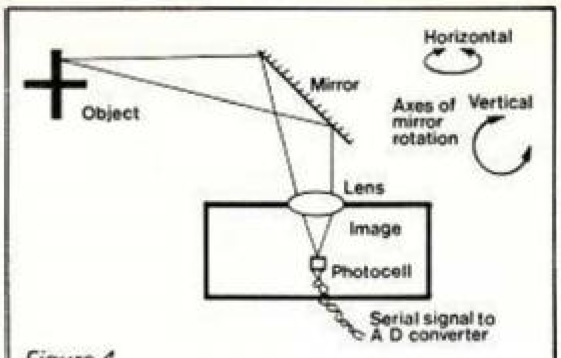


Figure 4.

If the rate at which the computer acquires its samples is increased so that 100 samples are gathered while the mirror moves only 10°, then the image presented to the viewer will cover a field of view of 10°. So, by altering the time in which the samples are gathered, the focal length of the system has been changed from a wide-angle view to a telephoto picture.

The software I have listed this month contains the main command loop for Cogent and the routine which will step through a source program you have written, interpreting valid characters and diverting to sections of machine code to perform individual instructions. Cogent, while still simple in outline, is becoming more complex and difficult to describe except as a whole. The subroutines I shall describe at the start of Cogent may not be essential as a whole for your own purposes.

Suzi transfers values from a table in the Cogent program to establish known values in TXST, TXST+1, TXEnd, and TXEnd+1. ClIn, the clock-initialisation routines, and ClS, clear the VDU screen. Suvia is a subroutine call into the ClIn routine which bypasses the instructions which zero the clock counters going directly to the section of the subroutine that sets up the VIA timer.

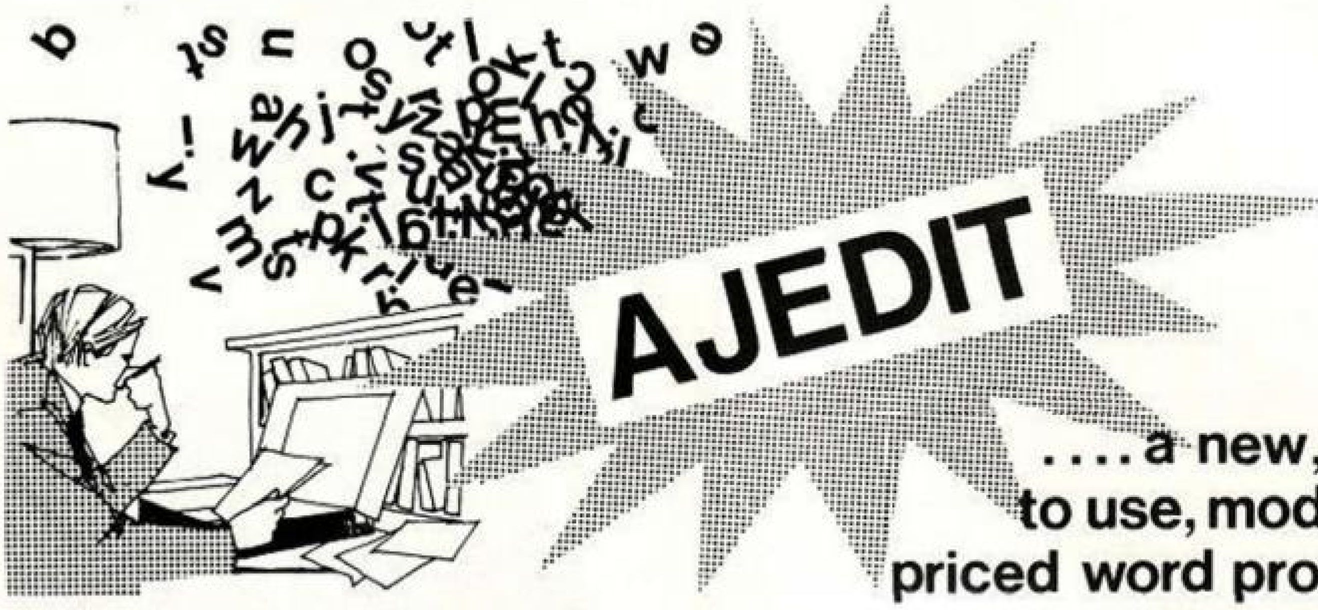
FLASC is a flag which is loaded with FF hex and used later on in the command loop. GChar takes a character from the keyboard and can be replaced with JSR $FDFA. Cubot puts the cursor in a known position on the VDU bottom line and the following lines, 813 to 820, display the main title.

After finding a character, it is compared for a match with one of the letters in the command loop and if a coincidence is found, the program will jump to one of the Cogent functions. If no match is found, the instructions at line 854 check to see whether this is the first time through the command loop and, if so, the present character is converted from lower-case ASCII to upper-case ASCII and the program tries again by jumping to SRT1. If no match is found on the second pass through the loop, the program goes through ERR1. ■

The main command loop for Cogent.

```
0789 :                             4478
0790 :  START OF COGENT            4478
0791 :                             4478
0792 START  CLD                    4478 D8
0793 :  SET UP ZERO PAGE          4479
0794        JSR SUZI               4479 20 F4 45
0795 :  INITIALISE CLOCK -         447C
0796 :  AFTER COLD START           447C
0797        JSR CLIN               447C 20 0A 40
0798 :  WARM START                447F
0799 :  RESTORE STACK POINTER     447F
0800 WARST  LDX £$FF               447F A2 FF
0801        TXS                    4481 9A
0802 :  CLEAR SCREEN AND SET      4482
0803 :  UP CLOCK VIA              4482
0804        JSR CLS                4482 20 E0 42
0805        JSR SUVIA              4485 20 18 40
0806 :  SET CAPS FLAG TO          4488
0807 :  NON-ZERO                  4488
0808        STX FLASC              4488 86 9C
0809        NOP                    448A EA
0810 :  PUT CURSOR AT VDU         448B
0811 :  BOTTOM LINE               448B
0812        JSR CUBOT              448B 20 40 47
0813 :  SET UP PROGRAM TITLE      448E
0814        LDA TMS1               448E AD C9 42
0815        STA MS1                4491 85 84
0816        LDA TMS1+1             4493 AD CA 42
0817        STA MS1+1              4496 85 85
0818        LDY NULL               4498 AC 05 40
0819 :  DISPLAY PROGRAM TITLE     449B
0820        JSR DMES1              449B 20 F6 42
0821 :  GET CHARACTER             449E
0822        JSR GCHAR              449E 20 4E 44
0823        LDA ICHAR              44A1 A5 01
0824 SRT1   CMP £'W                44A3 C9 57
0825        BNE TED                44A5 D0 03
0826        JMP WRITE              44A7 4C A0 45
0827 TED    CMP £'E                44AA C9 45
0828        BNE TST                44AC D0 03
0829        JMP EDIT               44AE 4C 2C 46
0830 TST    CMP £'S                44B1 C9 53
0831        BNE TRE                44B3 D0 03
0832        JMP STORE              44B5 4C 52 47
0833 TRE    CMP £'R                44B8 C9 52
0834        BNE TPR                44BA D0 03
0835        JMP READ               44BC 4C F0 47
0836 TPR    CMP £'P                44BF C9 50
0837        BNE TFD                44C1 D0 03
0838        JMP PRINT              44C3 4C F3 44
0839 TFD    CMP £'T                44C6 C9 54
0840        BNE TEX                44C8 D0 03
0841        JMP TIME               44CA 4C 6D 42
0842 TEX    CMP £'X                44CD C9 58
0843        BNE TAPP               44CF D0 03
0844        JMP EXEC               44D1 4C F9 44
0845 TAPP   CMP £'A                44D4 C9 41
0846        BNE TQUIT              44D6 D0 03
0847        JMP APPEND             44D8 4C 72 45
0848 TQUIT  CMP £'Q                44DB C9 51
0849        BNE TLC                44DD D0 04
0850        CLI                    44DF 58
0851        JMP TANBUG             44E0 4C 00 FC
0852 :  CHECK - WAS CHAR. A       44E3
0853 :  CAPITAL?                  44E3
0854 TLC    LDY FLASC              44E3 A4 9C
0855        CPY £$FF               44E5 C0 FF
0856        BNE ERR1               44E7 D0 07
0857 :                             44E9
0858 :                             44E9
0859        AND £$DF               44E9 29 DF
0860        STA FLASC              44EB 85 9C
0861 :      TRY AGAIN              44ED
0862        JMP SRT1               44ED 4C A3 44
0863 :  CHAR. NOT RECOGNISED      44F0
0864 :  GET ANOTHER               44F0
0865 ERR1   JMP WARST              44F0 4C 7F 44
0866 :                             44F3
0872 :                             44F9
0873 :                             44F9
0874 :  EXECUTE PROGRAM           44F9
0875 :                             44F9
0876 EXEC   LDA TXST               44F9 A5 B0
0877        STA PROG               44FB 85 9D
0878        LDA TXST+1             44FD A5 B1
0879        STA PROGH              44FF 85 9E
0880 EX1    LDY NULL               4501 AC 05 40
0881        LDA (PROG),Y           4504 B1 9D
0882        CMP SPACE              4506 CD 07 40
0883        BNE EX5                4509 D0 08
0884        INC PROG               450B E6 9D
0885        BNE EX3                450D D0 02
0886        INC PROGH              450F E6 9E
0887 EX3    BNE EX1                4511 D0 EE
0888 EX5    CMP NULL               4513 CD 05 40
0889        BNE EX6                4516 D0 03
0890        JMP WARST              4518 4C 7F 44
0891 EX6    STA SD                 451B 85 80
0892        SEC                    451D 38
0893        SBC £$40               451E E9 40
0894        BCC EXERR              4520 90 10
0895        ASL A                  4522 0A
0896        TAY                    4523 A8
0897        LDA INSREF,Y           4524 B9 52 45
0898        STA INSVEC             4527 85 54
0899        INY                    4529 C8
0900        LDA INSREF,Y           452A B9 52 45
0901        STA INSVEC+1           452D 85 55
0902        JMP (INSVEC)           452F 6C 54 00
0903 EXERR  LDA SD                 4532 A5 80
0904        JSR OPCHR              4534 20 75 FE
0905 NEXINS INC PROG               4537 E6 9D
0906        BNE NX1                4539 D0 02
0907        INC PROGH              453B E6 9E
0908 NX1    LDY NULL               453D AC 05 40
0909        LDA (PROG),Y           4540 B1 9D
0910        CMP NULL               4542 CD 05 40
0911        BNE NX2                4545 D0 03
0912        JMP WARST              4547 4C 7F 44
0913 NX2    CMP SPACE              454A CD 07 40
0914        BNE EXERR              454D D0 E3
0915        JMP EX1                454F 4C 01 45
0916 :                             4552
0917 :                             4552
0918 :  INSTRUCTION REFERENCE     4552
0919 :  TABLE                     4552
0920 :                             4552
0921 INSREF WOR HALT               4552 60 45
0922        WOR HALT               4554 60 45
0923        WOR HALT               4556 60 45
0924        WOR HALT               4558 60 45
0925        WOR TIME               455A 6D 42
0926        WOR HALT               455C 60 45
0927        WOR HALT               455E 60 45
0928 :                             4560
0929 :                             4560
0930 :  HALT INSTRUCTION          4560
0931 :                             4560
0932 HALT   TYA                    4560 98
0933        ADC £$40               4561 69 40
0934        JSR OPCHR              4563 20 75 FE
0935 HLT1   JSR GCHAR              4566 20 4E 44
0936        LDA ICHAR              4569 A5 01
0937        CMP £'C                456B C9 43
0938        BNE HLT1               456D D0 F7
0939        JMP NEXINS             456F 4C 37 45
0940 :                             4572
```

INNOVATIVE
TRS 80-GENIE SOFTWARE
from the professionals



.... a new, simple to use, moderately priced word processor

The introduction of a brand new word processor is a major event and AJEDIT is without doubt a major program. There are, however, quite a few Word Processors around and most of them are extremely good ones - why, therefore, another? The question is even more pertinent when it is known that we specifically commissioned the writing of it from an author of the status of Denville Longhurst of Enhanced Basic fame. The answer is that user feedback shows that a large number of customers do not need or want word processor programs which require a quantity of training before use. Scripsit, for instance, is an excellent program, but is complex to use; it even comes with a training course on tape. If one operator is dedicated to using the word processor then it makes sense to have her trained, and the more complex the program (so long as the complexity is accompanied by more and bigger functions) the better.

AJEDIT has been written for the user who needs a word processor intermittently, say three or four times a week. Its prime design criteria was ease of use - and just as importantly - ease of recollection of its commands. Take, for instance, the text editing commands - they are as close to the Basic Edit commands as possible, so that the user will remember them: To insert type I, to delete D, to take out three letters type 3D and so on.

Furthermore, AJEDIT has benefited from being written after a number of other word processors. The deficiencies in its predecessors are corrected in AJEDIT. For instance, any control characters can be outputted so that full advantage can be taken of the features of the particular printer being used. Disk directory access is available from within AJEDIT as is the killing of files on the disk. The FREE command and a number of other DOS commands can be carried out from within the program with a return to AJEDIT - with its text intact.

AJEDIT contains close to one hundred commands covering most word processor requirements. Dedicated printer commands for the Epson MX series and the Centronics 737 are included - again for ease of use of these two popular printers.

One of the big features of AJEDIT is the ability to "mail-merge". The facility is available whereby two special files are created, one containing names and addresses and a salutation, the other a standard letter or form. AJEDIT will call the address and salutation from one file and the letter from the other and thereby compile personalised letters. The salutation may be repeated in the body of the letter.

AJEDIT needs 48K and one disk minimum and is suitable for the TRS-80 Models I and III and the Video Genie Models I and II.

AJEDIT **£49.95**
Inclusive of V.A.T. and P. & P.

MOLIMERX LTD
A J HARDING (MOLIMERX)

1 BUCKHURST ROAD, TOWN HALL SQUARE, BEXHILL-ON-SEA, EAST SUSSEX.

TEL: [0424] 220391/223636 **TELEX 86736 SOTEX G**

AMERICAN EXPRESS Cards Welcome

BARCLAYCARD VISA

Buy it with Access

TRS-80 & VIDEO GENIE SOFTWARE CATALOGUE £1.00 **[refundable]** plus £1 postage.

RESPONSE FRAME

Do you have a problem? Your manual is incomprehensible or you just cannot get the hang of that programming trick you tried — whatever it is, Tim Hartnell will do his best to answer your queries. Please include only one question per letter and mark them "Response Frame".

KEYBOARD FAULT

■Over a period of about seven months, my Acorn Atom has developed a fault which produces double entries when any of the keys are pressed. This fault seems to be the result of poor-quality contacts on the keyboard — a problem which seems to be occurring increasingly frequently. Acorn's solution was to send me several new key contacts to replace the old ones, but I feel this is no solution to the poor quality of the keyboard. Could you direct me and others in possibly the same situation to any company which stocks a suitable replacement keyboard? I am quite prepared to make alterations to the printed-circuit board and re-case my machine.

Andrew Taylor,
Headingley, Leeds.

YOU ARE not alone — the office Acorn Atom tends to double-strike the letter s, but has not yet shown a tendency to do so with other letters. We understand from Acorn that the Atoms being despatched at present have a much better quality keyboard, based on the one for the BBC Microcomputer. That, however, does not help you. We cannot recommend a specific keyboard from a specific supplier, but suggest any coded keyboard is likely to prove suitable. Hobbyist suppliers such as Maplin or Henry's Radio have suitable keyboards in stock. We suggest that before you go for this radical solution, you replace the keys supplied by Acorn, and see if a change in your typing touch does not diminish or completely remove the double-strike problem.

PET LANGUAGE

■I am looking for a computer and it must cost, at most, £200. It must also have more than 1K memory, and have the same Basic as we are learning at school — we use it on the Commodore Pet. It must also be able to load programs from cassette. If there is such a computer, I would be very grateful if you could tell me what it is and how much it costs.

N Jones,
Swansea.

THE SIMPLE answer to your needs is the Commodore Vic-20, which uses Pet Basic, and loads and saves reliably from the Commodore cassette recorder. However, it will not meet your cost criterion. It costs around £185 to £195 for the computer, with an extra £35 to £40 for the special cassette recorder which it demands. However, it may well be worth paying the extra to buy a computer which meets your other needs. Otherwise, you may like to look at the ZX-81, which except for certain programming demands — such as the need for the word Let — will work reasonably well on the Basic you are learning. It costs £69.95. The Acorn Atom, £120 to £150, is a splendid computer for the price, but suffers from using a very non-standard version of Basic.

QUIETER SHARP

■Is it possible to control the sound volume of the Sharp MZ-80K by means of software? If so, can you please help and if not, can you show me any other method before the computer is smashed by the neighbours?

Hassoud Amire,
Geneva, Switzerland.

WE WERE ALSO appalled by the volume level of the MZ-80K, and at first attempted to solve the problem by placing a book against the speaker. There is, however, a volume control in the MZ-80K, although you need to unscrew four bolts at the base to lift off the cover. You will have no trouble locating this control once you remove the lid.

BOOKS ON BASIC

■I have seen a number of advertisements in *Your Computer* for books which purport to teach a beginner to program. I do not have a computer as yet but would like to learn to program — if such a thing is possible — before I go wild and spend a few hundred pounds on a computer myself. Can I learn to program without a computer, and if I can, which books would help me?

Arnold Gattworth,
Colwyn Bay, Clwyd.

YES, YOU CAN learn to program without a computer, but to do so is as unsatisfying as "learning" to play a musical instrument by just reading a correspondence course on how to play the piano. However, you can certainly gain a head start in programming, so when you finally decide to buy a machine, you will find it easier to make progress. It is difficult to recommend specific books — there are so many good books on Basic programming. A few which you may find useful are: *Basic from the ground up* by David E Simon, Hayden; *Making Basic work for you* by Claude J De Rossi, Reston; *Beginning Basic* by Paul M Chirlian, Dilithium Press. One book which we found particularly useful was *Game playing with Basic* by Donald D Spencer which, although it does not attempt to teach Basic from first principles, is capable of doing so if you work methodically. Once you have the basics under your belt you may wish to look at *The Basic cookbook,* by Ken Tracton, published by Radio Shack and available from most Tandy dealers.

ZX-81 PROBLEMS

■Can you help with two problems on my ZX-81? Having experienced great delays and frustrations over replacing a faulty RAM, I am reluctant to attempt to approach Sinclair again. I have the vanishing memory/vanishing program phenomenon. In my case, this is a result of editing in the presence of a multi-dimensional array. The defect of the Gosub stack is not confined to bad programming. With a databank type of program, where a new version has to be saved from time to time, surely the Gosub stack accumulates every time, and will eventually wreck the program. I am having to keep away from Gosubs at present because of this. Am I right?

K H Sargent,
Byfleet, Surrey.

YOU SOUND AS if you have a crazy ZX-81, or else you are simply misinterpreting what it is doing. We have never encountered instabilities due to size of arrays. The most likely explanation for the memory dropout is mains fluctuations which can be solved by putting a 9V battery in parallel across the power-supply lines. Another possible cause is dirt or moisture on the contacts. Clean the contacts very gently with fine emery paper, then either put the RAM pack on and leave it there for ever, making sure it does not move when you press a key, or — from time to time — spray the contacts at the back with the kind of moisture-removing spray sold for hard-to-start cars. The ZX-81 appears to do its own house-cleaning on the address which is left on the stack. Unlike many other computers, you can jump out of loops for ever without the loop count clogging the works. Why would you have a subroutine which is not followed by a Return? If you need a subroutine of this form, use a Goto instead, with a variable assigned just before the Goto which is, in effect, the return address. In this way, the Goto can end with another Goto taking you back to the line after your original Goto.

TEACHER'S ADVICE

■I am 12 years old, interested in computers and have been saving for some months to buy my own machine. I had intended to buy a Sinclair ZX-81, which seems very popular. When I told my teacher of this, he explained that the ZX-81 keyboard shows signs of wear after some months of constant use and suggested that I consider buying another make of computer. Unfortunately, other machines are well beyond my resources. I would be most grateful if you could advise me as to whether the ZX-81 has this keyboard weakness.

Huw Howell,
Port Talbot, West Glamorgan.

YOUR TEACHER, we respectfully suggest, does not know what he is talking about. Any keyboard weakness has never been mentioned in mail to the National ZX-80 and ZX-81 Users' Club, and we assure you that if this were a weakness, we would have known about it by now. You should buy whichever computer you want, and can afford, because any computer is better than none at all, and any computer will help you to become computer literate.

CHOICE IS YOURS

■I am considering buying a BBC Microcomputer which I understand, uses a standard form of Basic. Should I start with a BBC machine, or perhaps buy something cheaper first, and then progress to the BBC computer in due course?

Martin Wellwing,
Hampshire.

APART FROM the fact that demand appears to be outstripping supply, which could mean a considerable wait before you receive your machine, there is no reason — assuming you can afford it — for not buying a BBC or any other machine you desire. The BBC Microcomputer has a very flexible Basic, very close to standard Microsoft, which you will find easy to transfer to other machines of your choice. There are also a number of features it has inherited from the Atom — the Atom and the BBC machine are both produced by Acorn Computer of Cambridge. The features that the two machines have in common include the use of P. for Print, ? for Peek and Poke — where the context defines which is which — and abbreviations such as L. for List and GOS for Gosub.

DOUBLE-SIZE RAM

■I own a ZX-81 with a 16K RAM. The pack is unreliable and too often causes the program to crash for no reason. So I am looking for some alternatives. One of the advertisements in *Your Computer* claimed an expansion module from 16K up to 128K. However, to my knowledge, for the Z-80 CPU, the maximum addressable capacity should only be 64K. How can it be 128K?

C C Fung,
Roath, Cardiff.

THESE DEVICES use a paged approach, where the memory available is switched by commands from the computer through the output port. It is switched on to different sections of external memory. ■

Fingertips is our regular calculator column covering calculator news, programming hints and examples of unusual applications. The column is written and compiled by calculator enthusiast David Pringle who is glad to hear of any of your ideas. *Your Computer* pays £6 for each of your contributions published.

BENJAMIN FRANKLIN once said that the only certain things in life were death and taxes. Correspondent Wilfred Ashworth has decided that it is about time his TI-59 removed the grind from at least one of the two.

A programmable calculator is ideal for solving tricky repetitive mathematical equations, he writes, but it is not always easy to find really useful everyday jobs for it to do. This income-tax program, however, is especially interesting because it not only enables a continuing check to be made on the amount of tax currently owed, but is also an excellent exercise in the use of decision functions and in accommodating a complex calculation within the range of a hand-held machine.

After appropriate deductions have been made, the amount of tax to be paid is, of course, dependent on the level to which the total income rises in a series of non-equal bands of income, and on the nature of the source of income. Any tax office can supply a leaflet which outlines the personal allowances, bands of income chargeable at basic and higher rates, and investment income surcharge, so there is no need to list them here.

At first glance it would seem logical to begin at zero income and to calculate tax as the income rises above each band limit. In practice, a quicker and shorter program can be created by starting at the top and testing whether the upper figure of each band in turn can be subtracted from the taxable income and leave a positive result.

When it does, the tax on what remains is calculated at the appropriate level and the amount of tax which has accrued from all lower bands is added to the result. The calculation then proceeds by a GTO step to a subroutine which determines whether the investment income is greater than the allowed £5,500. If so, the surcharge is calculated and added to the tax due.

The 318 steps in the program are too numerous to fit into the TI-58C without re-partitioning to 399.09, but they can be easily accommodated on the TI-59. Fortunately, the re-partitioning leaves the TI-58 with 10 memories — exactly the number required. The stores are allocated as:

- 00 Gross income
- 01 Taxable income
- 02 Investment income
- 03 Temporary store used throughout the calculations
- 04 Highest tax rate applied
- 05 Tax owed
- 06 Investment income surcharge
- 07 Tax from last transaction
- 08 Tax credits on Building Society loans, dividends, etc.
- 09 Total current tax

User-defined labels are allocated to the various kinds of income as follows:

- A Salary, pension or other earned income
- B Dividends or Building Society interest
- C National Savings Bank interest
- D Bank interest
- E To clear all stores and reset the calculator for new data.

For example:

- If a sum is keyed into the display and key A is pressed, it is added to stores 00 and 01.
- If a sum is keyed into the display and key B is pressed, then the sum is grossed-up by the factor 10/7. The gross sum is added to stores 00, 01 and 02 and the increment by which the interest is increased is added to store 08 as a tax credit.
- If a sum is keyed into the display and key C is pressed, it is added only into store 00; but if it is greater than £70 — the tax-free sum — the amount in excess is added to the Stores 01 and 02.

Before any tax calculation is carried out, personal and other allowances must be summed and the result is made negative and added to store 01. Thus until allowances have been wiped from store 01, no taxable income — a positive amount — is shown there.

The program is given as a series of steps some of which require the use of several keystrokes. For example, "GTO A" would need the keystrokes "GTO", "2nd" and "A", but this convention makes the purpose of each step much clearer than a plain listing of the keystrokes. Various keys are used as labels in the testing routines, e.g.,

x^2, 1/x, 2nd $\Sigma +$

and the choice of these is arbitrary.

You will recognise that the large numbers appearing at intervals represent the levels of tax bands or tax accrued up to that stage. You will also see that the tax owing in store 05 consists of total tax less the tax credits allowed which appear in store 08.

The program can be used not only at a year's end to carry out a complete tax calculation but also throughout the period to find tax due at any time. The TI-58C retains the program and data; with the TI-59 these can be transferred to magnetic cards.

Should, for example, a dividend be received it can be keyed in, B pressed, followed by RST, R/S and not only will the stores show the new tax, but the figure in store 07 will show the amount of tax added by the inclusion of the single extra dividend.

It is thus possible for anyone using the program to set aside cash in a separate, and interest-bearing, account as each dividend or payment of interest is received, and not be caught out by a surprise demand for money that has already been spent.

I had little idea what a can of worms I was opening with the crossed-ladder problem in the November 1981 issue, writes David Pringle. Not only was every single one of the prodigious number of entries correct, but obviously a good deal of thought went into each solution. Hence the surplus of sardonic comments: "In fact, over the years, I must have spent more time looking for the 'final solution' than I care to remember" or: "I have lived with it since the early fifties and have foisted it on the managers of three large companies, with the loss of many thousands of paid and unpaid man-hours".

If you remember, our crossed-ladder problem is simply stated: Consider two ladders of length 20 and 30ft. facing in opposite directions between two parallel walls. The base of each ladder lies at the base of either wall while the top of each rests on the opposite wall to its base. If the intersection of the ladders is 10ft. off the ground, then how far apart are the walls?

Many thanks to John Snell of Hertfordshire who spent most of his reply discussing the second root of the ladder equation. This corresponds to the larger ladder lying under the ground, so we can discard that root as slightly unphysical. He, like almost everyone else, used the Newton-Raphson method of root finding. This is one of the fastest analytic methods of rooting so I am going to risk national anarchy and disorder by announcing my eventual winner.

John Greenwood of London has a blatant advantage with his HP-34C's Solve key which will find the root of most conveniently stated $f(X)=0$ problems. Still, his solution was the first received and the shortest. He attacks the problem in the following way.

For a given estimate of d he calculates

$$f(d) = a + b - d$$

where a and b equal tan $\sin^{-1}(d/3)$ and tan $\sin^{-1}(d/2)$ respectively. The correct d has been found if $f(d)=0$

(continued on page 65)

Income-tax program.

```
RCL 1
-27,750
=
STO 3
INV2NDX=>T
X^2
RCL 3
X.6
STO 4
+11,525
=
STO 9
GTO A'
2NDLBLX^2
CP
RCL 1
-22,250
=
STO 3
INV2NDX=<T
√X
RCL 3
X.55
STO 4
+8,500
=
STO 9
GTO A'
2NDLBL√X
CP
RCL 1
-16,750
=
STO 3
INV2NDX=>T
1/X
RCL 3
X.5
STO 4
+5,750
=
STO 9
GTO A'
2NDLBL1/X
CP
RCL 1
-13,250
=
STO 3
INV2NDX=>T
Y^X
RCL 3
X.45
STO 4
+4,175
=
STO 9
GTO A'
2NDLBLY^X
CP
RCL 1
-11,250
=
STO 3
INV2NDX=>T
EE
RCL 3
X.4
STO 4
+3,375
=
STO 9
GTO A'
2NDLBLEE
CP
RCL 1
X.3
STO 4
=
STO 9
GTO A'
2NDLBL2NDA'
CP
5,500
X⇆T
RCL 2
2NDX=>T
IN^X
RCL 9
-RCL 8
=
-RCL 5
=
STO 7
SUM 5
R/S
2NDLBLIN^X
RCL 2
-5,500
=
X.15
=
STO 6
SUM 9
RCL 9
-RCL 8
=
-RCL 5
=
STO 7
SUM 5
R/S
2NDLBLA
SUM 0
SUM 1
R/S
2NDLBLB
STO 3
X10/7
=
SUM 0
SUM 1
SUM 2
-RCL 3
=
SUM 8
R/S
2NDLBLC
STO 3
SUM 0
CP
70
X⇆T
RCL 3
2NDX=>T
2NDΣ+
R/S
2NDLBL2NDΣ+
RCL 3
-70
=
SUM 1
SUM 2
R/S
2NDLBLD
SUM 0
SUM 1
SUM 2
R/S
2NDLBLE
2NDCM
R/S
```

CHROMASONIC electronics

48 JUNCTION ROAD, ARCHWAY LONDON N19 5RD 100 yds FROM ARCHWAY STATION & 9 BUS ROUTES
TELEPHONE: 01-263 9493/01-263 9495 TELEX: 22568.

YOUR SOUNDEST CONNECTION IN THE WORLD OF COMPUTERS

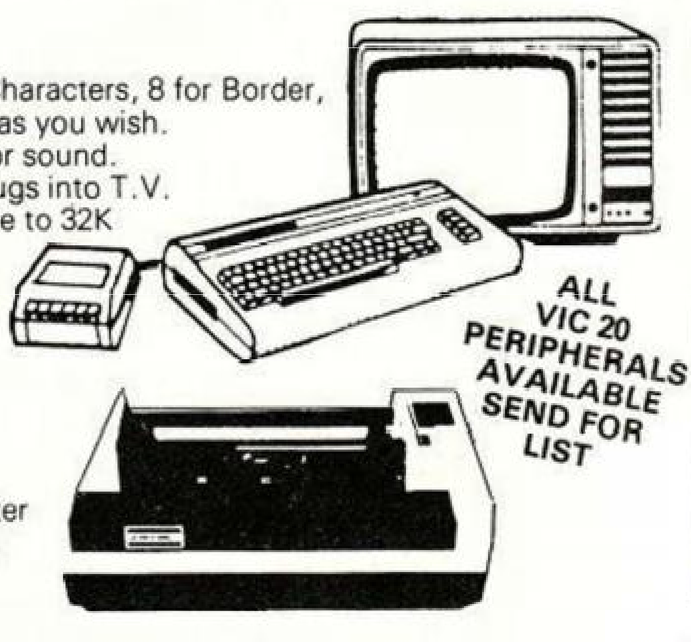
VIC 20

- 24 Colours, 8 for Characters, 8 for Border, 16 for screen mixed as you wish.
- 3 tone Generator for sound.
- Uses Pet Basic
- Plugs into T.V.
- Memory expandable to 32K
- VIC complete with T.V. Modulator and Power Supply

ONLY £165.00

VIC Cassette Deck
ONLY £38.00

Available soon
VIC Dot Matrix Printer
80 Column, 30 CPS,
Tractor Feed
ONLY £199.95



ALL VIC 20 PERIPHERALS AVAILABLE SEND FOR LIST

UK101

DOWN IN PRICE

UK101 Kit **inc 8K memory**	**£125**
Ready Built **inc 8K memory**	**£175**
Complete in case	**£199**
4K Expansion 8x2114	**£10**
Parallel Printer Interface	**£24.50**
Cases	**£19.95**
Chromasonics Sound Kit	**£24.50**
Colour Kit	**£69.95**

NEW NEW NEW

32K Dynamic Memory Board only	**£89.95**
P.I.O. and Eprom Programmer Kit only	**£24.50**

APPLE II PLUS

AUTOSTART "EUROPLUS"

48K Apple Computer	**649.00**
Disc Drive with Controller	**349.00**
Disc Drive without Controller	**299.00**
Colour Card	**69.00**
Silentype Printer	**199.00**
Graphics Tablet	**425.00**
TV Modulator	**14.00**

A range of Apple Accessories and Software are available

PET

4016 16K RAM	445.00
4032 32K RAM	569.00
8032 32K RAM	755.00
8096 96K RAM	960.00
4040 Disk Drive	585.00
8050 Disk Drive	755.00
4022 Printer 80 Col	357.00
8024 Printer 132 Col	975.00
8026 Printer/Typewriter	835.00
8027 Daisy Wheel	735.00

A range of Pet Accessories and Software are available

PRINTERS

INTERFACES AND CABLES FOR APPLE II, PET, TRS80, RS232, UK101, SHARP SUPERBOARD — ALL AVAILABLE.

EPSON MX80 £359
Dot-matrix printer with Pet graphics interface. Centronics parallel and serial. Pet and Apple compatible. True bidirectional, 80 cps.

EP80 MX82 £389
As MX80 plus high Resolution Graphics

EPSON MX80 FT/1 £399
Dual single sheet friction and tractor feed, 9 wire head, true descenders.

EPSON-MX80 FT/2 £440
An FT/1 with high resolution graphics.

SEIKOSHA GP80A £199
Dot matrix 5 x 7, 80 columns, 30 cps graphics, double width characters.

JUST PHONE FOR FURTHER DETAILS

VIDEO GENIE

£299 EG3003

Utilises Z80, 12K level II Basic, Integral Cassette Deck, UHF O/P, 16K RAM, all TRS80 features. Simply plugs into monitor or UHF TV. With V.U. Meter.

PARALLEL PRINTER INTERFACE INC CABLE	£38.00
CHROMASONICS **PROGRAMMABLE** SOUND KIT	£24.50
SOUND KIT (FITTING EXTRA)	£8.20
LOWER CASE KIT (FITTING EXTRA)	£29.80
COLOUR KIT (FITTING EXTRA)	£36.00
EXPANSION BOX INC 16K RAM	£199
16K/32K RAM BOARD	£94/ £129
NEW GENIE II NOW AVAILABLE	£310

MONITORS

9" Green Screen	£99.00	HITACHI PROFESSIONAL	
12" Green Screen	£125.00	9"	£99.95
12" BMC Green Screen	£159.00	12"	£149.00

TANTEL

PRESTEL BY TANTEL

COMMUNICATIONS AT YOUR FINGERTIPS FOR BUSINESS & HOME. UP TO DATE INFO

180,000 pages of information on Travel, News, Investment, Holidays, Hotels Etc., Etc.

£159

TANTEL IS POST OFFICE APPROVED. SEND FOR DETAILS. DEMONSTRATION AVAILABLE AT OUR SHOWROOM.

ALL ITEMS CARRY A ONE YEAR GUARANTEE

BARCLAYCARD VISA

Please add VAT 15% to all prices. Postage on computers, printers and cassette decks charged at cost, all other items P&P 30. Place your order using your Access or Barclaycard (Min. tel. order £10). Export enquiries welcome. Official orders welcome.

(continued from page 63)

for then the ladders intersect at the required height, which is 1ft. Note that he solves the 10 scale problem and multiplies the final answer — see figure 1.

```
01 1 or any estimate of D/10
02 ENTER
03 SOLVE 1
04 1
05 0
06 X
07 RTN
08 LBL 1
09 3
10 GSB 2
11 2
12 GSB 2
13 —
14 +
15 RTN
16 LBL 2
17 ÷
18 SIN⁻¹
19 TAN
20 R↑
```

To run, simply press the R/S key. The current f(d) estimate is calculated by subroutine 1 and passed on to the Solve function to iterate to the next value.

The underlying Solve microprogram uses five personal memory registers for holding three sample arguments of the function, say, a, b, and c, and the two previously calculated trial values f(a) and f(b). Value c is derived from a and b so we must know their relationship.

Suppose a and b lie close to a root x=e of the equation f(x)=0. Then a secant, or straight line, that cuts the graph of f at the points

$(x=a, y=f(a))$ and $(x=b, y=f(b))$

must cut the x-axis at the point

$(x=c, y=0)$

given by:

$$c = b - (b-a)(f(b))/f(b)-f(a))$$

Provided the graph of f is smooth and a relatively simple root is being searched, then c is a much better approximation than either a or b — see figure 2. Hence a is discarded and the next point, d, is derived from b and c.

This is, not surprisingly, called secant iteration and although not the fastest root finder, it is one of the most generally applicable and manageable for the microprogram of a calculator. The version inside the HP-34C is slightly more complex in that it judges whether a root is to be forthcoming at all and will actually register those values for which the gradient of the function disappears.

If you found the ladder problem a trifle easy, here is something slightly more testing for those dark winter nights. Consider the same set-up but with the base of each ladder 1ft. away from each wall. How far apart are the walls if the ladders are 15ft. and 16ft. long and their point of intersection is 6ft. above the ground? What if the 15ft. ladder is 2ft. from its wall?

Figure 1.

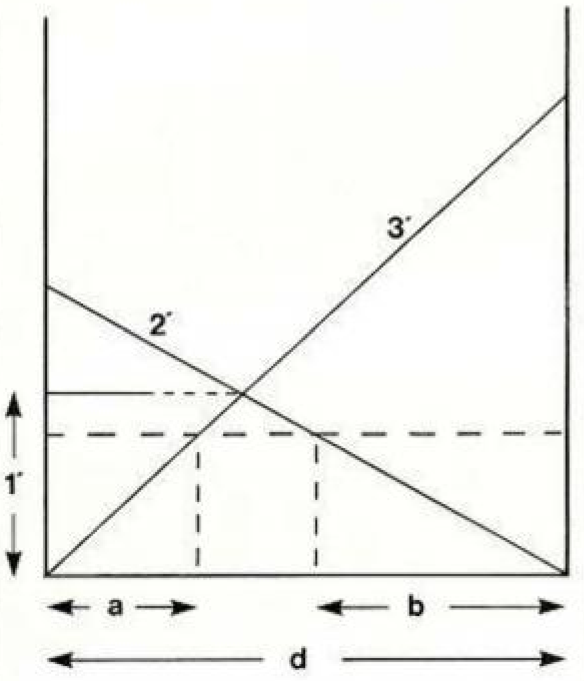
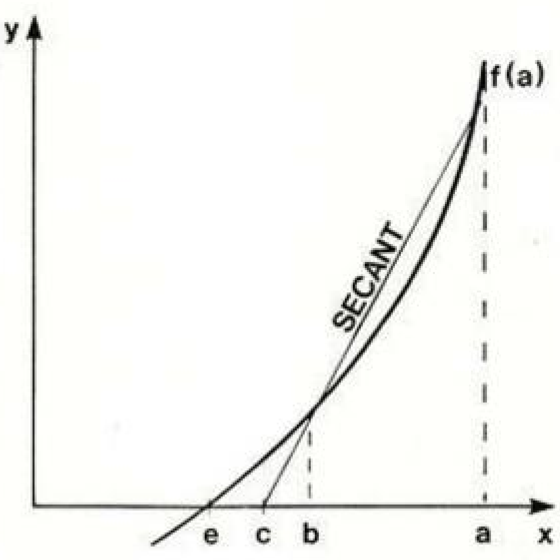


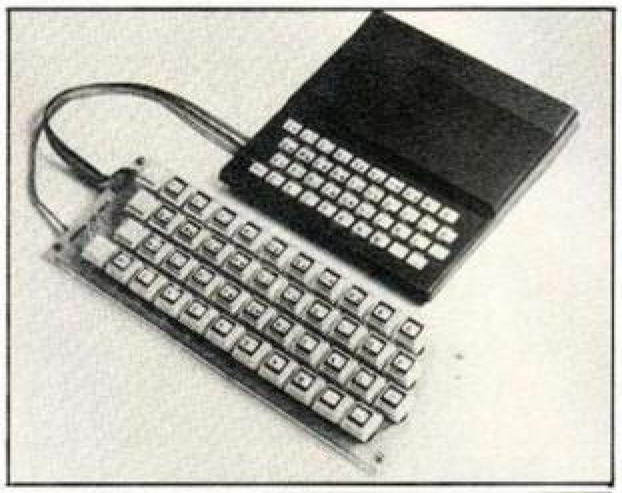
Figure 2.



ZX 80/81 HARDWARE/SOFTWARE

ZX KEYBOARD

A full size keyboard for the 80/81. The keyboard has all the 80/81 functions on the keys, and will greatly increase your programming speed. It is fitted with push type keys as in larger computers.
The keyboard has been specially designed for the Sinclair computer and is supplied ready-built. It also has facilities for 4 extra buttons which could be used for on/off switch, reset, etc. **£27.95**



4K GRAPHICS ROM

The dK Graphic module is our latest ZX81 accessory. This module, unlike most other accessories fits neatly inside your computer under the keyboard. The module comes ready built, fully tested and complete with a 4K graphic ROM. This will give you 448 extra pre-programmed graphics, your normal graphic set contains 64. This means that you now have 512 graphics and with there inverse 1024. This now turns the 81 into a very powerful computer, with a graphic set rarely found on larger more expensive machines. In the ROM are lower case letters, bombs, bullets, rockets, tanks, a complete set of invaders graphics and that only accounts for about 50 of them, there are still about 400 left (that may give you an idea as to the scope of the new ROM). However, the module does not finish there; it also has a spare holder on the board which will accept a further 4K of ROM/RAM. IT NEEDS NO EXTRA POWER AND WORKS FROM YOUR NORMAL POWER SUPPLY. **£27.95**

RAM 80/81

16K RAM
Massive add-on memory for 80/81.
16K KIT-A-KIT VERSION
of a 16K Ram. Full instructions included. All memory expansions plug into the user port at the rear of the computer. 16K RAM **£42.95** 16K KIT **£32.95**
2K & 4K RAM
Static Ram memory expansion for the 80/81. They both work with onboard Ram i.e. 4K plus onboard = 5K. This is the cheapest small memory expansion available anywhere. 2K RAM **£15.95**. 4K RAM **£22.95**

16K 81 SOFTWARE

As seen at the ZX Microfair.
DEFLEX This totally new and very addictive game, which was highly acclaimed at the Microfair, uses fast moving graphics to provide a challenge requiring not only quick reaction, but also clever thinking. One and two player versions on same cassette. £3.95
3D/3D LABYRINTH You have all seen 3D Labyrinth games, but this goes one stage beyond; you must manoeuvre within a cubic maze and contend with corridors which may go left/right/up/down. Full size 3D graphical representation. £3.95.
CENTIPEDE. This is the first implementation of the popular arcade game on any micro anywhere. Never mind your invaders, etc., this is positively shining, the speed at which this runs makes ZX invaders look like a game of simple snap. £4.95.
Please add £1 p&p for all hardware, Software p&p free. Specify ZX80/81 on order.
ALL OUR PRODUCTS ARE COVERED BY A MONEY BACK GUARANTEE

dK'tronics

23 Sussex Road, Gorleston,
Great Yarmouth, Norfolk.
Telephone: Yarmouth (0493) 602453

SOFTWARE FILE

Software File gives you the opportunity to have your programs, ideas or discoveries published. We will accept contributions for any personal computer and will group programs for like machines together in the file. Please double-check your listings before sending them. Mark your letter clearly for *Your Computer*. We will pay £6 for each contribution published.

Noughts and crosses

Tony Poulter, Meopham, Kent.

ZX-81

SO FAR AS I know, this is the first unbeatable noughts and crosses in Basic for the 1K ZX-81. An unbeatable program may not be thought an interesting game but from a programming point of view, it is better.

Also, it has to be very economic. So it avoids numbers as far as possible. Go To code is used. Line 33 uses Not S for zero, and line 44 uses SGN S for 1 and INT PI for 3.

The program displays a very small screen in lines 24 to 26. On each move the choice is available either to press one of the characters on the screen to enter a human's move there, or to press zero causing the computer to move. Thus the program can be used five ways:

- As a notepad for two human players. Pressing zero is never used.
- As a game against the computer, pressing the player's move and zero alternately.
- Let the computer start by pressing zero and human moves alternately.
- Press zero on every move — the computer plays itself.
- Enter the first few moves as the player's moves and then press zero for the computer to make best move no matter what peculiar situation you have set up.

When you press zero, the screen goes blank while the computer thinks. The maximum time until the display returns is 12 seconds. As the program does not stop anywhere, I need to explain what facilities there are for dealing with the end of the game. First, when the computer is waiting for a key to be pressed, the possible keys are 1, 2, 3, 5, 6, 7, 9, A, B, zero, full stop, and break.

Break is common to all programs. Any other key will have no effect.

Any key in the range 1 to B will have no effect if that place has already been taken on the screen by X or O. Pressing zero will cause the computer to think, then return with no effect if all the places have been taken on the screen. There is no effect either if many places have been taken and a draw is inevitable and also if a completed line is present on the screen. You escape from these, and any other situation you do not like, by pressing full stop, which prints a new starting position.

If anyone wishes to amend the program they should note that the constant P, which is the address of the Newline character starting the display file less 28, has to be altered if the length of the program is altered.

```
22 LET P=17115
23 CLS
24 PRINT "123"
25 PRINT "567"
26 PRINT "9AB"
27 LET S=CODE "X"
28 SLOW
29 LET C=CODE INKEY$
30 IF C=CODE "." THEN RUN
31 IF C<>CODE "0" THEN GOTO CODE "Z"
32 FAST
33 LET D=NOT S
34 LET E=D
35 FOR Q=CODE"1" TO CODE"B"
36 IF PEEK(P+Q)<>Q THEN GOTO CODE"Y"
37 LET F=NOT S
38 LET G=F
39 LET H=F
40 FOR N=F TO CODE "+" STEP INT PI
41 LET I=NOT S
42 LET J=I
43 LET K=I
44 FOR M=SGN S TO INT PI
45 LET L=PEEK (P+CODE "1235679AB15926A37B16B963"(N+M))
46 IF L=S THEN LET J=J+Q
47 IF L=113-S THEN LET K=K+Q
48 IF L=Q THEN LET I=Q
49 NEXT M
50 IF J=INT PI*Q OR K=INT PI*Q THEN GOTO C
51 IF NOT I OR J AND K THEN GOTO CODE "S"
52 IF J+K>Q THEN LET H=Q+J
53 LET G=G+J
54 LET F=F+K
55 LET H=H+SGN S
56 NEXT N
57 IF F>Q OR G>Q THEN LET H=H+PI
58 IF F>Q THEN LET E=E+S
59 IF E=S+S AND H=PI+INT PI AND D=H THEN LET C=CODE "2"
60 IF H>D THEN LET C=Q
61 IF H>D THEN LET D=H
62 NEXT Q
63 IF PEEK (P+C)<>C THEN GOTO CODE "0"
64 POKE P+C,S
65 LET S=113-S
66 GOTO CODE "0"
```

The dictator

Martin Bishop, Warrington, Cheshire.

ZX-81

YOU ARE A dictator in a small city for a period of 10 years. Each year you can buy and sell land, sow your land with corn, and feed your populace. If you do not feed your people properly — 10 bags of corn per person per year — some will starve and if too many die, the survivors may rebel against you.

Each acre shown requires one bag of corn. The harvest from the land is your only income, unless you speculate with land, buying and selling at different prices. Beware, though, of the rats. They eat corn but not land. After 10 years you will be given a report and score.

```
  10-120 SETS UP VARIABLES
1000-1100 DISPLAYS CURRENT STATUS
1110-1230 INPUT "ACRES TO BE SOLD"
1240-1360 INPUT "ACRES TO BE BOUGHT"
1370-1560 INPUT "ACRES TO BE SOWN"
1570-1680 INPUT "BAGS TO FEED POPULATION"
2000-2170 CALCULATES STATUS
2200-2280 REBELLION ROUTINE
2400-2580 REPORT AFTER 10 YEARS
SUBROUTINES
9000-9050 INDICATES TOO LITTLE LAND TO DO AS ASKED
9100-9130 ERASES A LINE INDICATED BY QQ
9200-9270 CHECKS SYNTAX OF INPUTS AND CONVERTS TO A NUMBER
9300-9310 STATUS UPDATE
9400-9410 INDICATES TOO LITTLE CORN TO DO AS ASKED
9500-9510 PRINTS A LINE
ZX81 DICTATOR LISTING 16K
  10 RAND 0
  20 LET P=100
  30 LET Y=1
  40 LET SP=0
  42 LET TSP=0
  50 LET NP=10
  60 LET C=3000
  70 LET A=1000
  80 LET H=5
  90 LET L=INT(RND*5)+10
 100 LET R=1000
 110 LET AP=0
 120 LET SC=0
1000 PRINT AT 0,10;"DICTATOR"
1010 GOSUB 9500
1020 PRINT "POP. OF CITY IN YEAR ";Y;" IS ";P
1030 PRINT NP;" PEOPLE CAME TO THE CITY"
1040 PRINT SP;" CITIZENS STARVED"
1050 GOSUB 9300
1060 PRINT "CORN YIELDED ";H;" BAGS PER ACRE"
1070 PRINT "LAND COSTS ";L;" BAGS PER ACRE"
1080 PRINT "RATS ATE ";R;" BAGS OF CORN"
1090 GOSUB 9500
1100 PRINT AT 17,0;"--------------------------------"
1110 PRINT AT 12,0;"HOW MANY ACRES TO BE SOLD?"
1120 INPUT A$
1130 GOSUB 9200
1140 IF Z=1 THEN GOTO 1170
1150 IF AS<=A THEN GOTO 1200
1160 GOSUB 9000
1170 LET QQ=12
1180 GOSUB 9100
1190 GOTO 1110
1200 LET A=A-AS
1210 LET C=C+AS*L
1220 PRINT AT 12,28;AS
1230 GOSUB 9300
1240 PRINT AT 13,0;" ""   ""   ""   ""  BOUGHT?"
1250 INPUT A$
1260 GOSUB 9200
1270 IF Z=1 THEN GOTO 1300
1280 IF AS*L<=C THEN GOTO 1330
1290 GOSUB 9400
1300 LET QQ=13
1310 GOSUB 9100
1320 GOTO 1240
1330 LET A=A+AS
1340 LET C=C-AS*L
```

(continued on next page)

(continued from previous page)

```
1350 PRINT AT 13,28;AS
1360 GOSUB 9300
1370 PRINT AT 14,0;"HOW MANY ACRES TO BE SOWN?"
1380 INPUT A$
1390 GOSUB 9200
1400 IF Z=1 THEN GOTO 1430
1410 IF AS=A THEN GOTO 1460
1420 GOSUB 9000
1430 LET QQ=14
1440 GOSUB 9100
1450 GOTO 1370
1460 IF AS<=C THEN GOTO 1490
1470 GOSUB 9400
1480 GOTO 1430
1490 IF AS<=P*10 THEN GOTO 1530
1500 PRINT AT 19,0;"YOU ONLY HAVE ";P;" WORKERS"
1510 GOSUB 9010
1520 GOTO 1430
1530 LET C=C-AS
1540 PRINT AT 14,28;AS
1550 GOSU% 9300
1560 LET AP=AS
1570 PRINT AT 15,0;" ""   "" BAGS TO FEED POP.?"
1580 INPUT A$
1590 GOSUB 9200
1600 IF Z=1 THEN GOTC 1630
1610 IF AS<=C THEN GOTO 1660
1620 GOSUB 9400
1630 LET QQ=15
1640 GOSUB 9100
1650 GOTO 1570
1660 LET C=C-AS
1670 PRINT AT 15,28;AS
1680 GOSUB 9300
2000 LET SP=0
2005 IF P*10=AS THEN GOTO 2050
2010 LET SP=P-INT(AS/10)
2020 LET P=P-SP
2030 LET TSP=TSP+SP
2040 IF SP>P*(RND*5+10)/10 THEN GOTO 2200
2050 LET Y=Y+1
2060 IF Y=11 THEN GOTO 2400
2070 LET H=1+INT(5*RND)
2080 LET C=C+AP*H
2090 LET R=0
2100 IF C>=10000 THEN LET R=C-10000
2110 LET R=R+INT(C*ABS(RND-.5))
2120 LET C=C-R
2130 LET NP=INT(RND*30)
2140 LET P=P+NP
2150 LET L=INT(RND*5)+10
2160 CLS
2170 GOTO 1000
2200 CLS
2210 PRINT "YOUR POPULACE HAS REBELLED DUE
            TO YOUR TOTAL LACK OF REGARD
            FOR THEIR NOURISHMENT"
2215 GOSUB 9500
2220 PRINT "IN YOUR SHORT PERIOD OF
            MANAGEMENT YOU SUCCESSFULLY
            STARVED ";TSP;" PEOPLE"
2225 GOSUB 9500
2230 PRINT "THE LEADER OF THE REBELS HAS
            GIVEN YOU THE CHOICE TO REFORM
            AND HAVE ANOTHER 10 YEARS OR
            FACE EXECUTION"
2235 GOSUB 9500
2240 PRINT "PRESS ""R"" FOR REFORM AND ANY
            OTHER KEY FOR EXECUTION"
2250 IF INKEY$="" THEN GOTO 2250
2260 IF INKEY$="R" THEN RUN
2270  CLS
2280 STOP
2400 CLS
2410 PRINT AT 0,8;"10 YEAR REPORT"
2420 GOSUB 9500
2430 PRINT "IN 10 YEARS OF DICTATORSHIP -"
2440 PRINT "  ";TSP;" PEOPLE STARVED TO DEATH"
2450 IF A<1000 THEN PRINT "  YOU SOLD OFF ";(1000-A);" ACRES OF LAND"
2460 IF A>=1000 THEN PRINT "  YOU ACQUIRED ";A-1000;" ACRES OF LAND"
2470 GOSUB 9500
2480 LET SC=ABS(INT(100*((150-TSP)/150)*(A/1500)*(P/150)))
2490 PRINT "ON A SCALE FROM 1 TO 100 YOUR
            PERFORMANCE RATES A SCORE OF ";SC
2500 GOSUB 9500
2510 IF SC<20 THEN PRINT "YOU ARE AN INCREDIBLE FAILURE"
2520 IF SC>=20 AND SC<40 THEN PRINT "WHAT A LOAD OF RUBBISH"
2530 IF SC>=40 AND SC<60 THEN PRINT "HI, MUSSOLINI"
2540 IF SC>=60 THEN PRINT "SO YOU THINK YOU ARE GOD, DO YOU"
2550 IF TSP>150 THEN PRINT "HITLER WAS A WET COMPARED TO YOU"
2560 GOSUB 9500
2570 PRINT "TYPE ""R"" FOR ANOTHER RUN OF 10
            YEARS OR ANY OTHER KEY TO FINISH"
2580 GOTO 2250
9000 PRINT AT 19,0;"YOU ONLY HAVE ";A;" ACRES OF LAND"
9010 FOR I=0 TO 100
9020 NEXT I
9030 LET QQ=19
9040 GOSUB 9100
9050 RETURN
9100 FOR I=0 TO 30 STEP 2
9110 PRINT AT QQ,I;"■";AT QQ,I;" ";AT QQ,I+1;"■";AT QQ,I+1;" "
9120 NEXT I
9130 RETURN
9200 IF A$= "" THEN GOTO 9206
9202 LET AS=0
9204 GOTO 9240
9206 FOR I=1 TO LEN A$
9210 IF CODE A$(I)<28 OR CODE A$(I)>37 THEN GOTO 9260
9220 NEXT I
9230 LET AS=VAL A$
9240 LET Z=0
9250 RETURN
9260 LET Z=1
9270 RETURN
9300 PRINT AT 0,5;"YOU HAVE ";C;" BAGS OF CORN   ";TAB 5;
              "AND ";A;" ACRES OF LAND   "
9310 RETURN
9400 PRINT AT 19,0;"YOU ONLY HAVE ";C;" BAGS OF CORN"
9410 GOTO 9010
9500 PRINT "--------------------------------"
9510 RETURN
VARIABLES
  A = NUMBER OF ACRES OF LAND OWNED
 AP = NUMBER OF ACRES PLANTED WITH CORN
 AS = NUMERICAL FORM OF INPUT
 A$ = STRING INPUT
  C = NUMBER OF BAGS OF CORN OWNED
  H = BAGS OF CORN HARVESTED PER ACRE
  I = FOR/NEXT LOOP VARIABLE
  L = PRICE OF LAND IN BAGS OF CORN PER ACRE
 NP = NUMBER OF PEOPLE WHO CAME TO CITY
  P = POPULATION OF CITY
 QQ = LINE TO BE ERASED
  R = BAGS OF CORN EATEN BY RATS
 SP = NUMBER OF PEOPLE STARVED IN ONE YEAR
TSP = TOTAL NUMBER OF STARVED PEOPLE
 SC = SCORE
  Y = YEAR
  Z = INPUT CHECK (1=INVALID ENTRY 0=VALID ENTRY)
  1 RAND 0
  2 LET P=100
  3 LET C=3000
  4 LET A=1000
  5 LET Y=0
  6 LET L=8+INT(RND*7)
  7 LET Y=Y+1
  8 GOSUB 200
  9 PRINT "SELL?"
 10 INPUT S
 15 IF S>A THEN GOTO 8
 20 LET A=A-S
 25 LET C=C+S*L
 30 GOSUB 200
 35 PRINT "BUY?"
 40 INPUT S
 45 IF S*L>C THEN GOTO 30
 50 LET A=A+S
 55 LET C=C-S*L
 60 GOSUB 200
 65 PRINT "SOW?"
 70 INPUT W
 75 IF W>A OR W>C OR W>P*10 THEN GOTO 60
 80 LET C=C-W
 85 GOSUB 200
 90 PRINT "FEED?"
 95 INPUT S
100 IF S>C THEN GOTO 85
110 LET C=INT((C-S+W*RND*5)*(RND/2+.5))
115 LET P=INT(P-(P-S/10)*(S=P*10)+RND*30)
120 IF Y=10 THEN GOTO 400
125 GOTO 6
200 CLS
210 PRINT "Y=";Y;" P=";P;" L=";L
220 PRINT "A=";A;" C=";C
230 RETURN
400 CLS
410 PRINT "SCORE=";INT(A*P/10000)
SCREEN DISPLAY
Y=1
P=100
L=10
A=1000
C=3000
SELL?
Y=YEAR NUMBER (1-10)
P=POPULATION
L=PRICE OF LAND (IN BAGS PER ACRE)
A=ACRES OF LAND OWNED
C=NO. OF BAGS OF CORN OWNED
```

Mechanical music

Andrew Turner,
Rugby, Warwickshire.

ATOM

MY MACHINE-CODE program, Music, is capable of accepting a string of musical notes, say, CDEFG, and converting them into sounds of the correct frequency played over the Atom loudspeaker.

As it stands, the program will accept the characters A to G, up-arrow, full stop, hash and space. All other characters are ignored. The letters A to G are used to represent the notes starting from A below middle C. If a hash character is put after a letter, the sharp of that note will automatically be played.

The up-arrow and full-stop characters are used to change between the two octaves available. On entering the routine, notes are assumed to be in the lower octave until an up-arrow character is discovered.

Once in the upper octave, all notes are played as the higher notes until a full stop is encountered, which takes it down again to the lower octave. If, for example, two up-arrows are encountered, the second up-arrow will be ignored. If a space is encountered, then a rest of the correct duration will be played. Finally, notes such as E flat must be converted into the correct sharp, in this case D sharp. Also if E sharp or B sharp is encountered, F and C will be played respectively.

The program takes 168 bytes of memory plus the length of the string of notes to be played. My favourite place to assemble the program is between # 2800 and # 2900 — that is, 255 bytes, leaving enough space after the program for a number of notes to be played.

However, this memory space may not be convenient for you if you have a floating-point ROM fitted since it is used to store the values of the floating-point variables.

The notes can be set to any duration before the program, but the program can only play one note length throughout the piece. The tempo is set by the subroutine at line 700, which takes only about 0.3 seconds to execute. This subroutine is dependant on the variable D which must be set between 256*1 for a very fast tune, to 256* # 2F for the longest duration. It must also be divisable by 256.

A quick demonstration is included at the end of the program showing some of the many things it can do.

Once the program is assembled, save it on tape with

*SAVE "MUSIC" 2800 2900

Save all 255 bytes so that the note and tempo vectors will be ready to use. When reloaded, put the string of notes to be played at # 28A8, by:

$# 28A8 = "CDEFG ↑ ABC"

for example, and Link # 2800. If you want to change the vectors, you have to use a Basic subroutine at line 700 as described.

```
10 REM**MUSIC**
20 REM**BY ANDREW TURNER**
30 DIM VV(9)
40 P.$21
50 FOR W=1 TO 2
60 P=#2800
70 [ :VV0
80 LDY@0;STY#83
90 :VV9 LDA #28A8,Y
100 LDX@0;STX#84
110 CMP@#5E;BNE VV1
120 LDX#83;BNE VV1
130 LDX@#E;STX#83
140 :VV1 CMP@#2E;BNE VV2
150 LDX#83;BEQ VV2
160 LDX@0;STX#83
170 :VV2 CMP@13;BEQ VV3
180 CMP@32;BNE VV4
190 LDX@27;STX#84;LDA@#41
200 VV4 CMP@#41;BMI VV5
210 CMP@#48;BPL VV5
220 CLC;SBC@#40;ASL A
230 CLC;ADC#83;TAX
240 LDA #28A9,Y;CMP@#23
250 BNE VV6
260 INX;INY
270 :VV6 STY#81
280 LDA #2870,X;LDY #288C,X
290 TAX;LDA#B002
300 STX#80
310 :VV7 LDX#80
315 :VV8 DEX
320 NOP;NOP;NOP
330 BNE VV8
340 LDX#84;EOR@4
350 STA#B002,X
360 DEY;BNE VV7
370 LDY#81
380 :VV5 INY;BNE VV9
390 :VV3 RTS
400 ];NEXT W;PRINT $6
405 REM*NOTE FREQUENCY TABLES*
410 V=#2870
420 !V=#ACB7BFCE
430 V!4=#909AA0AC
440 V!8=#78808088
450 V!12=#5F656B71
460 V!16=#5055555A
470 V!20=#4044484C
480 V!24=#34383C40
490 REM*DEMONSTRATION*
500 CLEAR0;PRINT$30
510 $#28A8="AA#BCC#DD#EFF#GG#↑AA#BCC#DD#EFF#GG#"
515 D=256*#2F;GOSUB 700
520 PRINT"A DEMONSTRATION."''
530 PRINT"I'LL PLAY MY WHOLE RANGE OF"'
540 PRINT"NOTES, WHICH IS:"''
550 PRINT $#28A8''
560 PRINT"FIRST SLOWLY..."''
570 FOR W=1 TO 90;WAIT;NEXT
580 LINK #2800
590 PRINT"AND NOW GO FAST..."'''
600 D=256*#2;GOSUB 700
610 FOR W=1 TO 40;WAIT;NEXT
620 FOR W=1 TO 20;LINK#2800;NEXT
630 PRINT"FINALLY, ON A PATRIOTIC NOTE..."
640 $#28A8="↑BBB B C C B C BA  .GF#F#F#F#      ↑DDD"
650 $(#28A8+LEN(#28A8))=" CCC B.G↑CAD C BBB AAA. GGGGG "
660 D=256*#2A;GOSUB 700
670 FOR W=1 TO 60;WAIT;NEXT
680 LINK #2800
690 END
700 REM*SUB TO CHANGE TEMPO*
710 FOR M=0 TO 27;M?#288C=D/(M?#2870);NEXT;RETURN
```

Dare devil

Mark Andrews, Glasgow.

ZX-81

DARE DEVIL is a game I invented to use with the 16K ZX-81 microcomputer. You are in control of a parachutist who is slowly drifting down between two parallel skyscrapers. You can manoeuvre it left or right with the two cursor keys.

To land the parachutist like this would be easy, but obstacles block its path. The hazards to be overcome include flagpoles and a strong breeze which blows more strongly after the first successful landing.

```
1 LET U=0.5
2 REM "DD"
5 CLS
10 PRINT AT 5,2;"ZX81  DARE DEVIL."
11 PRINT AT 10,0;"TRY TO MANOEUVER YOUR"
12 PRINT AT 11,0;"PARACHUTIST DOWN BETWEEN THE"
13 PRINT AT 12,0;"SKY-SCRAPERS, BUT BEWARE OF"
14 PRINT AT 13,0;"WIND, AND THE FLAG-POLES. ALSO"
15 PRINT AT 14,0;"AS YOU GET LOWER, THE PASSAGE"
16 PRINT AT 15,0;"BETWEEN THE BUILDINGS GETS"
17 PRINT AT 16,0;"NARROWER. YOU HAVE ONE LIFE."
18 PRINT AT 21,0;"PRESS N/L TO START."
20 INPUT A$
21 CLS
22 PRINT AT 21,0;"WHAT IS YOUR NAME?"
30 INPUT B$
35 CLS
40 FOR E=0 TO 21
41 PRINT AT E,7;"GOOD LUCK ";B$
42 NEXT E
43 PAUSE 200
50 CLS
80 LET A=10
81 LET X=0
90 LET W=INT(RND*2)
91 LET Y=INT(RND*17)
92 FOR T=19 TO 0 STEP -1
93 LET S=INT(RND*2)
94 IF S=0 THEN LET A=A+U
95 IF S=1 THEN LET A=A-U
96 IF X>2 THEN LET Z=1
97 IF X<=7 THEN PRINT AT 21,0;"          "
98 IF X<=7 THEN PRINT AT 21,26;"      "
99 IF X>7 AND T=6 THEN GOTO 500
100 IF X>5 THEN PRINT AT 21,20;"   "
101 IF X>5 THEN PRINT AT 21,9;"   "
102 IF X>2 THEN PRINT AT 21,6;"   "
103 IF X>2 THEN PRINT AT 21,23;"   "
104 IF X>7 THEN PRINT AT 21,0;"  32 SPACES  "
105 PRINT AT 6,A-1;"  O  "
106 PRINT AT 7,A-1;"     "
107 PRINT AT 8,A-1;"     "
108 PRINT AT 5,A-1;"     "
109 PRINT AT 4,A-1;"     "
110 PRINT AT 3,A-1;"     "
115 LET Q=INT(RND*5)
116 IF Q=2 AND X<6 THEN GOSUB 300
117 IF W=0 AND X<=7 THEN GOSUB 400
118 IF W=1 AND X<=7 THEN GOSUB 600
120 PRINT AT 2,A-1;"     "
125 IF X<=7 AND W=0 AND A<=15 AND T=9 THEN GOTO 1000
126 IF X<=7 AND W=1 AND A>=14 AND T=9 THEN GOTO 1000
127 IF X<=2 AND A<=6 OR X<=2 AND A+2>=26 THEN GOTO 1000
128 IF X>2 AND A<=8 OR X>2 AND A+2>=23 THEN GOTO 1000
129 IF X>5 AND A<=11 OR X>5 AND A+2>=20 THEN GOTO 1000
130 IF INKEY$ ="8" THEN LET A=A+1
140 IF INKEY$ ="5" THEN LET A=A-1
190 SCROLL
200 NEXT T
205 LET X=X+1
```

(continued on page 71)

(continued from page 69)

```
 220 GOTO 90
 300 LET P=INT(RND*25)
 310 PRINT AT 21,P;"     "
 320 PRINT AT 20,P;"     "
 330 PRINT AT 19,P;"     "
 340 RETURN
 400 PRINT AT T,6;"             "
 401 PRINT AT T+1,13;"    "
 402 PRINT AT T+2,14;"V"
 410 RETURN
 500 PRINT AT 17,5;"WELL DONE"
 510 PRINT AT 18,5;"NOW TRY WITH"
 511 PRINT AT 19,5;"A STRONGER WIND."
 520 PAUSE 300
 530 LET U=U+0.5
 540 CLS
 550 GOTO 80
 600 PRINT AT T,16;"            "
 610 PRINT AT T+1,16;"    "
 620 PRINT AT T+2,17;"V"
 630 RETURN
1010 PRINT AT 2,8;"GOOD-BYE ";B$
1100 FOR E=7 TO 19
1111 PRINT AT E,A+3;"AGG"
1200 PRINT AT E,A;"    "
1250 PRINT AT E+1,A;"    "
1260 PRINT AT E+2,A;"    "
1270 PRINT AT E-1,A;"    "
1275 NEXT E
1280 FOR E=0 TO 21
1285 PRINT AT E,0;"   32 SPACES   "
1300 IF E>10 THEN PRINT AT E,0;"   32 GREY CHARS.   "
1350 NEXT E
1360 FOR E=6 TO 10
1365 PRINT AT E,15;" "
1370 IF E=8 THEN PRINT AT E,13;"     "
1380 NEXT E
1390 PRINT AT 0,0;"YOUR PARACHUTIST IS DEAD"
1400 PRINT AT 1,0;"BUT IF YOU WOULD LIKE TO "
1450 PRINT AT 2,0;"TRY AGAIN, TYPE "YES"."
1500 INPUT X$
1501 CLS
1550 IF X$="YES" THEN GOTO 1
1560 STOP
```

Chi-squared

Gordon Millington, Guildford, Surrey.

GENIE

IN THE SOCIAL sciences it is often necessary to deal with data in the form of frequencies. One compares the number of times a particular event occurs in a group under study with the frequency of the same event in a control group.

We might, for instance, have obtained the following relationships between smokers, X, and non-smokers, not X, developing lung cancer, Y, or not, not Y.

	Smokers	Non-smokers
Cancer	230	78
No cancer	465	652

The program first asks for these figures to be input — lines 10 to 40: L=230, M=465, N=78, and P=652. It then calculates and prints out the comparative table of the numbers observed and expected — lines 120 to 160 — and finally computes the statistic chi-squared according to the corrected formula for small samples.

With one degree of freedom in a two-by-two table such as ours, chi-squared = 3.481 is significant at the five-per-cent level.

```
  3 CLS
  5 PRINT"CHI SQUARED 2x2-1 DF":PRINT
 10 INPUT"X & Y";L
 20 INPUT"X & NOT Y";M
 30 INPUT"NOT X & Y";N
 40 INPUT"NOT X & NOT Y";P
 50 DIM E(3)
 60 A=L+M:B=N+P:C=L+N
 70 D=M+P:F=C+D
 80 E(0)=A*(C/F)
 90 E(1)=A*(D/F)
100 E(2)=B*(C/F)
110 E(3)=B*(D/F)
115 CLS
120 PRINT"OBSERVED","EXPECTED"
130 PRINT L,E(0)
140 PRINT M,E(1)
150 PRINT N,E(2)
160 PRINT P,E(3)
170 IF L>E(0)THEN G=(L-E(0)-.5)[2/E(0)ELSE G=(E(0)-L-.5)[2/E(0)
180 IF M>E(1)THEN H=(M-E(1)-.5)[2/E(1)ELSE H=(E(1)-M-.5)[2/E(1)
190 IF N>E(2)THEN J=(N-E(2)-.5)[2/E(2)ELSE J=(E(2)-N-.5)[2/E(2)
200 IF P>E(3)THEN K=(P-E(3)-.5)[2/E(3)ELSE K=(E(3)-P-.5)[2/E(3)
210 CHI=G+H+J+K
220 PRINT:PRINT
230 PRINT"CHI SQUARED =";CHI;"WITH
240 PRINT"1 DF AND 1 TAIL"
250 PRINT:PRINT:END
```

Learning fun

C M Robinson, Slough, Berkshire.

ZX-81

TIMES IS A 1K program designed to provide practice drill and test children on knowledge and use of multiplication tables. It may be set to any table up to 21, via the input at line 2. Random questions are set followed by five possible answers.

If the correct answer is displayed, the child must respond by touching the corresponding key, 1 to 5, within the permitted time. A correct answer credits the child with one point and an incorrect answer debits the total by one.

Line 70 ensures the child can recheck the display as long as required before continuing. A final score is given out of a maximum possible of 10. To save memory space, others may find the technique employed here in lines 170 and 240, of saving numbers as characters in a string useful.

The second program sets 10 questions of the form

12 X ? = 108

which have to be answered by touching the number key. I find lines 110 to 140 particularly useful as protection against inadvertently touching the wrong key.

Thinking time is penalised by a descending score as is an incorrect answer. Up to three incorrect answers are allowed. The correct answer is automatically displayed and held until the operator commands the game to continue by touching the Newline key.

A reward appears for a good score. The degree of difficulty can be adjusted by amending line 80, for example, to become:

LET X=INT (RND*100)+1

```
  1 PRINT "WHICH TABLE?"
  2 INPUT B
  5 RAND
 10 LET S=0
 20 LET J=0
 30 LET V=0
 35 LET K=0
 40 LET Z=100
 50 PRINT ,"SCORE ";S
 60 IF J=10 THEN STOP
 70 INPUT Z$
 80 CLS
 90 LET A$=""
100 LET A=INT(RND*13)
120 PRINT A;" X ";B;
130 FOR N=12 TO 28 STEP 4
140 LET X=B*INT(RND*13)
```

(continued on page 73)

ELECTRONIC GAMES

COLOUR CARTRIDGE T.V. GAME

SEMI-PROGRAMMABLE T.V. GAME
• 4 Cartridges • Mains Adaptor
Normal Price £73
NOW REDUCED TO: £39.50 inc VAT

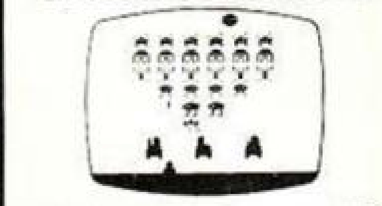
DATABASE T.V. GAME

FULLY PROGRAMMABLE CARTRIDGE T.V. GAME
14 Cartridges available
Normal Price £87.86
NOW REDUCED TO: £59 inc VAT

ATARI T.V. GAME

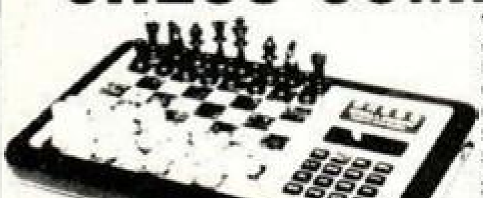
The most popular T.V. Game on the market with a range of over 40 cartridges including SPACE INVADERS with over 112 games on one cartridge. **£95.45** inc VAT

SPACE INVADERS



Hand-held Invaders Games available **£19.95**
• Invaders Cartridges available to fit *ATARI/RADOFIN/ACETRONIC/PHILIPS G7000*
• Cartridges also available for *MATTEL/TELENG/ROWTRON/DATABASE/INTERTON*

CHESS COMPUTERS



MANY UNITS ARE COVERED BY THE EXCLUSIVE SILICA SHOP 2 YEAR GUARANTEE

We carry a range of over 15 different Chess computers:

Electronic Chess	£29.95
Chess Traveller	£39.95
Chess Challenger 7	£79.00
Sensory 8	£119.00
Sensory Voice	£259.00

SPECIAL OFFERS:
VOICE CHESS CHALLENGER
Normal Price £245 **NOW £135.00**
SARGON 2.5/BORIS 2.5
Normal Price £273.70 **NOW £199.95**
All prices include V.A.T.

TELETEXT

ADD-ON ADAPTOR £199 inc VAT

THE RADOFIN TELETEXT ADD-ON ADAPTOR

Plug the adaptor into the aerial socket of your colour T.V. and receive the CEEFAX and ORACLE television information services.

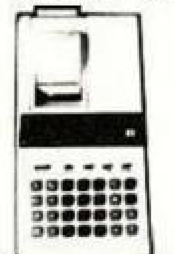
THIS NEW MODEL INCORPORATES:
* Double height character facility
* True PAL Colour
* Meets latest BBC & IBA broadcast specifications
* Push button channel change
* Unnecessary to remove the unit to watch normal TV programmes
* Gold plated circuit board for reliability
* New SUPERIMPOSE News Flash facility

SPEAK & SPELL

Normal Price £49.95
NOW REDUCED TO: £39.50 inc VAT

Teach your child to spell properly with this unique learning aid. Fully automatic features and scoring. Additional word modules available to extend the range of words.

ADDING MACHINE
OLYMPIA HHP 1010



Normal Price £57.21
NOW REDUCED TO: £34 inc VAT

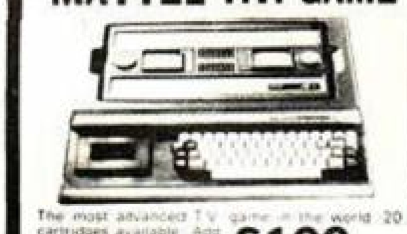
Uses ordinary paper!
No need to buy expensive thermal paper!
Fast add listing PRINTER CALCULATOR 2 lines per second. 10 digit capacity. Uses normal adding machine rolls. Battery or mains operated.
Size 9¼"x4½"x2¾"
(Mains adaptor extra)

24 TUNE ELECTRONIC DOOR BELL

Normal Price £19.70
NOW REDUCED TO: £12.70 inc VAT

Plays 24 different tunes with separate speed control and volume control. Select the most appropriate tune for your visitor, with appropriate tunes for different times of the year!

MATTEL T.V. GAME



The most advanced T.V. game in the world. 20 cartridges available. Add on KEYBOARD coming soon to convert the MATTEL to a home computer with 16K RAM, fully expandable and programmable in Microsoft Basic. Other accessories will be available later in the year. **£199.95** inc VAT

HAND HELD GAMES
EARTH INVADERS

These invaders are a breed of creature hitherto unknown to man. They cannot be killed by traditional methods — they must be buried. The battle is conducted in a maze where squads of aliens chase home troops. The only way of eliminating them is by digging holes and burying them. **£23.95** inc VAT

HAND HELD GAMES
GALAXY 1000

The 2nd generation Galaxy Invader. The invaders have re-grouped and have a seemingly endless supply of spacecraft whilst the player's arsenal is limited to just 250 missiles to be launched from 3 missile stations. You have to prevent the invaders landing or from destroying your home defences. **£19.95** inc VAT

THE OLYMPIA — POST OFFICE APPROVED
TELEPHONE ANSWERING MACHINE
WITH REMOTE CALL-IN BLEEPER

This telephone answering machine is manufactured by Olympia Business Machines, one of the largest Office Equipment manufacturers in the U.K. It is fully POST OFFICE APPROVED and will answer and record messages for 24 hours a day. With your remote call-in bleeper you can receive these messages by telephone wherever you are in the world. The remote call-in bleeper activates the Answer/Record Unit, which will at your command repeat messages, keep or erase them, and is activated from anywhere in the world, or on your return to your home or office. The machine can also be used for message referral, if you have an urgent appointment, but are expecting an important call, simply record the 'phone number' and location where you can be reached. With optional extra bleepers (£13 each) this facility can be extended to colleagues and members of the family. Using a C90 standard cassette you can record as many as 45 messages. The announcement can be up to 16 seconds long and the incoming message up to 30 seconds long.

The machine is easy to install and comes with full instructions. It is easily wired to your junction box with the spade connectors provided or alternatively a jack plug can be provided to plug into a jack socket. Most important, of course, is the fact that it is fully POST OFFICE APPROVED.
The price of £135 (inc. VAT) includes the machine, an extra-light remote call-in Bleeper, the microphone message tape, A.C. mains adaptor. The unit is 9¼"x6"x2½" and is fully guaranteed for 12 months. The telephone can be placed directly on the unit — no additional desk space is required.

£135 inc VAT

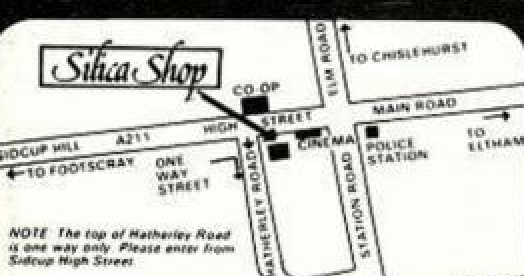
PRESTEL VIEWDATA

The ACE TELCOM VDX1000 Prestel Viewdata adaptor simply plugs into the aerial socket of your television and enables you to receive the Prestel Viewdata service in colour or black & white.

Features —
— Simplified controls for quick, easy operation
— Special graphics feature for high resolution
— State-of-the art microprocessor controller
— Standard remote telephone keypad with Prestel keys * #
— Auto dialler incorporated for easy Prestel acquisition
— True PAL colour encoder using reliable IC — chroma filter and dela line incorporated for minimum picture interference/maximum fidelity
— Includes convenient TV — Prestel switchbox
— Easily connected to standard home or office telephone lines
— Fully Post Office approved

SPECIAL PRICE **£228.85** inc VAT

FOR FREE BROCHURES – TEL: 01-301 1111



NOTE The top of Hatherley Road is one way only. Please enter from Sidcup High Street.

For free illustrated brochure and reviews on our range of electronic games please telephone 01-301 1111. Free delivery service available. To order by telephone please quote your name, address and ACCESS/BARCLAYCARD number and leave the rest to us. Post and packing Free of Charge. Express 48hr delivery service available.
* CALLERS WELCOME — Demonstrations daily at our Sidcup shop, open from 9am-6pm Monday-Saturday (Early Closing Thursday 1pm — Late Opening Friday 8pm)
* 2 YEAR GUARANTEE — All goods are covered by a full year's guarantee and many are further covered by our exclusive Silica Shop 2 year Guarantee
* MONEY BACK UNDERTAKING — If you are unsatisfied with your purchase and return it within 7 days we will give you a full refund
* AFTER SALES SERVICE — Available on all machines out of guarantee
* COMPETITIVE PRICES — We are never knowingly undersold
* HELPFUL ADVICE — Available on the suitability of each machine
* CREDIT FACILITIES — Full credit facilities available over 12, 24 or 36 months at competitive rates of interest
* PART EXCHANGE SCHEME — available on second hand machines
* CREDIT CARDS WELCOME — Access, Barclaycard, Diners Club, American Express

SILICA SHOP LIMITED DEPT YC I–82
1-4 The Mews, Hatherley Road, Sidcup, Kent DA14 4DX
Telephone: 01-301 1111 or 01-309 1111

SILICA SHOP

(continued from page 71)

```
150 IF X=A*B THEN GOSUB 300
160 PRINT TAB N;X;
170 LET A$=A$+CHR$ X
180 NEXT N
185 LET J=J+K
190 FOR T=1 TO 20
200 IF INKEY$<>"" THEN LET V=VAL INKEY$
210 IF V>5 THEN LET V=0
220 NEXT T
230 IF V=0 THEN GOTO 270
240 IF CODE A$(V)=A*B THEN GOTO 350
250 LET S=S-1
260 IF V<>0 THEN PRINT ,"WRONG"
270 IF Z<>100 THEN PRINT ,"YOU MISSED ";Z
280 GOTO 30
300 LET K=1
310 LET Z=X
320 RETURN
350 LET S=S+1
360 PRINT ,"CORRECT"
370 GOTO 30
 10 RAND
 20 PRINT "PRESS NEWLINE WHEN YOU ARE READY"
 30 INPUT A$
 40 LET S=1000
 50 FOR A=1 TO 10
 60 LET E=0
 70 CLS
 80 LET X=INT (RND*12) +1
 90 LET N=INT (RND *10)
100 PRINT X;" X ? = ";X*N,
110 LET B=CODE INKEY$
120 IF B>27 AND B<38 THEN GOTO 150
130 LET S=S-1
140 GOTO 110
150 LET B=B-28
160 PRINT B
170 IF B=N THEN GOTO 260
180 LET S=S-10
190 PRINT "WRONG","SCORE= ";S
200 LET E=E+1
210 IF E<3 THEN GOTO 100
220 PRINT "ANSWER IS ";N
230 INPUT A$
240 NEXT A
250 GOTO 300
260 LET S=S+10
270 PRINT "CORRECT","SCORE= ";S
280 PAUSE 50
290 NEXT A
310 IF S>800 THEN PRINT ,,"**GOOD**",
320 PRINT,,,,"GAME OVER"
330 INPUT A$
335 CLS
340 RUN
```

The 24-line screen

Timothy Gilbert, Barry, South Glamorgan.

ZX-81

As an avid ZX-81 user I have discovered a few useful tricks while experimenting. One of these is a 24-line screen. The screen is normally 22 lines by 32 columns, with another two lines at the bottom for input. Many probably know how to Poke characters on to these lower lines, but would it not be better to Print or Print At on them? Here is how to.

The system variable at 16418 is the number of lines in the lower half of the screen. Although you are told it crashes the system if Poked, it does not in fact do so. By Poking in the value — the numbers on the screen are reversed in this variable — we are able to use the full 24 screen lines.

On these lines we can Print At — the bottom line is now 23 — and Print. This is especially useful when adapting programs from systems with 24 screen lines and in producing a very large chess-type board. Do not try using Input or Scroll — you will crash the system.

```
1000 LET RV=PEEK 16418
1010 POKE 16418,2
1020 INPUT R$
1030 POKE 16418,RV
1040 LET RV=0
1090 RETURN
```

As an extension to this idea, if you print what you require and then Poke *16418* with the number of lines you wish to protect, when you use Scroll only the top half of the screen will move leaving your printed text untouched.

If you want to input something, Poke 16418,2 but ensure there is nothing important on the bottom two lines as they will be cleared. I usually print my prompts on the normal screen only and protect them with a Poke. This leaves the bottom two lines clear for my Input subroutine, to obtain data without disturbing my prompts but still leaving them protected on Return.

Through the maze

R Pincott, Mansfield, Notts

ATOM

By changing a few lines in the three-dimensional maze in the *Acorn Atom Magic Book* you can turn the program into a quick-reaction, real-time, high-speed game. Hit the wall and you are dead. Use navigator for next left and right turns and do not press too soon or too late or you will hit the wall. Try to find your way through the maze at high speed. The C key is for left and B key for right.

On the end of my Maze game is a shortened version of Space Battle — level 6 with one alien — so when I have eased through the maze, I have to shoot down an alien. The more memory you use, the smaller your maze size will become. You exit via Line 1175.

1175 p L=X; M=Y; IF L>N; IF O=M; G.5000
5000 SPACE BATTLE

```
1120 ▨
1122 F.C=1 TO 5;?#B000=?#B000 &#F0+C
1123 C?#7F=?#B001 &8:N.C
1124 Z=CH"F"
1125 IF?#83=0;Z=CH"C"
1126 IF?#84=0;Z=CH"R"
  TAKE OUT LINE 1127
  TAKE OUT LINE 1128
  TAKE OUT LINE 1129
1152 P.$7$12,"YOUR'E DEAD NEXT";RUN
```

Equations solved

A Jones, Leeds.

ZX-81

As this program contains no Peeks or Pokes, it could run on many other machines in addition to the 1K ZX-81 for which it was written. It solves simultaneous equations of a standard found in GCE O-level and CSE mathematics exam papers. Not only can it be used to do homework, but it can be used to solve general logic puzzles. For example:

Fred buys three apples and two bananas which cost him 4p. Then George buys seven apples and three bananas which cost him 11p. If X equals apples and Y equals bananas, find how much one apple costs and how much one banana costs.

$$3X + 2Y = 4$$
$$7X + 3Y = 11$$

(continued on next page)

(continued from previous page)

First the title is printed. Lines 20 to 130 are involved with the inputting of the equations and lines 150 to 170 print the two equations. Lines 180 to 190 change variables B and C to negative numbers and then they are applied to the formula in line 200 and 210.

The lines 220 to 240 print the appropriate values of X and Y. The remainder of the program then asks you whether you wish for another equation to be solved.

If we consider the Fred and George example the two equations were:

$$3X + 2Y = 4$$
$$7X + 3Y = 11$$

The program first asks to input the X value. Here you would type 3 and Newline. Then it instantly asks for the Y value. You type 2 and a Newline. Then, when it asks for the answer, type 4. It then continues to question you on the second equation. After this, the screen clears and the two equations are printed followed by the two numbers that X and Y represent — the first is X.

```
 10 PRINT "SIMULTANEOUS EQUATIONS"
 20 PRINT AT 11,11;"INPUT X VALUE"
 30 INPUT A
 40 PRINT AT 11,11;"INPUT Y VALUE"
 50 INPUT B
 60 PRINT AT 11,11;"INPUT THE ANSWER"
 70 INPUT E
 80 PRINT AT 11,11;"INPUT OTHER X VALUE"
 90 INPUT C
100 PRINT AT 11,11;"INPUT OTHER Y VALUE"
110 INPUT D
120 PRINT AT 11,11;"INPUT OTHER ANSWER"
130 INPUT F
140 CLS
150 PRINT" THE ANSWER **********************************"
160 PRINT AT 5,5;A;"X+";B;"Y=";E
170 PRINT AT 6,5;C;"X+";D;"Y=";F
180 LET S=B*-1
190 LET R=C*-1
200 LET X=((D*E)+(S*F))/((D*A)-(S*R))
210 LET Y=((R*E)+(A*F))/((D*A)-(S*R))
220 PRINT "X= ";X
230 PRINT
240 PRINT "Y= ";Y
250 PRINT "ANOTHER? (Y,YES:N,NO)"
260 INPUT L$
270 IF L$="N" THEN STOP
280 GOTO 10
```

Telescope modelling

B Spencer,
Chatham, Kent.

ZX-80

TELESCOPE MODELLING should be of general interest, and of special interest to those who own, or intend to buy or build an astronomical telescope.

When run the program will display:

DATA OR CALC

If data is input, the screen will display two rows of figures:

LOWEST MAGNITUDE
11 15 22 33 38 45
MAGNIFICATION
60 80 120 180 200 400

These figures are related vertically — that is, a magnification of 60 will resolve stars down to a magnitude of 11. The table of data is useful as a means of reference. When selecting eyepieces in the calculation section.

Key Newline for the calculation section of the program which allows the telescope optics to be configured.

OBJECT LENS DIA

means input the desired value — say, 200 for a 200mm. lens.

FOCAL RATIO OF OBJECT

means input the ratio — 10 for an f/10 lens.

The screen will clear and then display the values input plus the focal length of a mirror which will have been computed for an eyepiece focal length from which the magnification of the telescope can be calculated. The following display then appears:

OBJECT DIA = 200mm.
FOCAL RATIO = F/10
FOCAL LENGTH = 2,000mm.
FOCAL LENGTH EYEPIECE

Input a value for the eyepiece. For example, 25mm. input 25. The screen will clear and the display is produced.

OBJECT DIA = 200mm.
FOCAL RATIO = F/10
FOCAL LENGTH = 2,000mm.
FOCAL LENGTH EYEPIECE 25mm.
MAGNIFICATION IS 80
FOCAL LENGTH EYEPIECE

The display now shows all the data input plus the focal length and magnifications which have been computed by the program. The magnification is calculated to the first decimal place. Finally, the user is asked for another eyepiece value. The program can cope with up to four eyepiece values and gives the resulting magnification. After this the display will again ask:

***** DATA OR CALC *****

Should a value of eyepiece focal length be input which would prove to be beyond the effective magnification of the system — say, too low a value — then the program will tell you that it is too small.

```
 10 PRINT "DATA OR CALC"
 20 INPUT A$
 30 IF A$="CALC" THEN GOTO 120
 40 CLS
 50 PRINT "LOWEST MAGNITUDE"
 60 PRINT
 70 PRINT "11    15    22    38    45"
 80 PRINT "MAGNIFICATION"
 90 PRINT
100 PRINT "60    80    120    180    200    400"
110 INPUT A$
120 CLS
130 PRINT "OBJECTIVE LENS DIA"
140 INPUT C
150 PRINT "FOCAL RATIO OF OBJECT"
160 INPUT D
170 CLS
180 LET E=C*D
190 PRINT "OBJECT DIA = ";C;" MM"
200 PRINT "FOCAL RATIO = F/";D;" "
210 PRINT "FOCAL LENGTH = ";E;" MM"
220 LET N=1
230 PRINT "FOCAL LENGTH EYEPIECE"
240 INPUT Y
250 PRINT Y;" MM"
260 LET Z=E/Y
270 LET L1=E-Z*Y
280 LET K1=10*L1/Y
290 IF Z>C*2 THEN GOTO 340
300 PRINT "MAGNIFICATION IS      ";Z;" ";K1
310 LET N=N+1
320 IF N=4 THEN GOTO 360
330 GOTO 230
340 PRINT Y;" MM TOO SMALL"
350 GOTO 230
360 PRINT "*****DATA OR CALC*****"
370 INPUT A$
380 IF A$="CALC" THEN GOTO 120
390 GOTO 40
```

Calculated risk

Loll Holt,
Worsley, Manchester.

ZX-81

WHEN SOMEONE referred to my ZX-81 as a "glorified calculator", I began to wonder if it would be really possible to make it perform like one. The resulting program is really a basis for future development. You enter your calculation in the usual form, e.g.:

3*SIN (PI/4)+3

followed by Newline. The result is displayed at the top of the screen as on a calculator display. If you then enter another expression, the first result, stored as E, is forgotten while the new value is displayed. If, however, you now enter:

+SQR 32

the value of this — root 32 — is added, or whichever operation you want, to the displayed number and the new result is printed. If you just press Newline, the value at the top of the screen is stored in memory as a variable M. Thus the memory can be recalled by:

M (Newline)

and also used in calculations:

SQR M − 4

It should not be difficult to expand this program, introducing such functions as more memories, percentages, a Clear function and constant.

Incidentally, the only ways out of this program are Stop or Break, both of which are bad programming but easier and quicker than checking to see if the user has input some special code meaning "stop".

In Golf, you play a nine-hole course, and the length of each hole is random. To make a stroke, hold down any key for a certain time: the longer you hold it down the further the ball goes. Beware — the maximum distance is 200yd.

When you are within 30yd. of the pin the computer putts for you: two putts from more than 5yd., one from less than 5yd. unless, of course, you chip into the hole. After some practice you should be able to complete the course at level par.

My final program is a mystery for you to solve. Type it in and find out what, if any, its purpose is. When you have discovered what it does, try working out how it does it. As a clue, it would take nearly 20 years to fill a line.

```
  5 RAND
 10 LET T=0
 15 LET H=1
 20 LET S=0
 25 LET L=INT(RND*250)+250
 30 LET P=VAL(STR$ L)(1)+1
 35 CLS
 40 PRINT "GOLF" ,, "SCORE" ,,,, "HOLE" ,, "LENGTH" ,, "PAR" ,,
"STROKES"
 45 IF H>9 THEN GOTO 120
 50 PRINT AT 1,7; T; "-"; AT 3,6; H; AT 4,8; L; "--"; AT 5,5; P AT 6,9;
S
 55 IF L<30 THEN GOTO 105
 60 IF INKEY$="" THEN GOTO 60
 65 FORI=0 TO 250
 70 IF INKEY$="" THEN GOTO 90
 75 NEXT I
 80 LET I=100
 85 GOTO 70
 90 LET S=S+1
 95 LET L=ABS(L-I)
100 GOTO 50
105 LET T=T-P+S+(L>5)+(L<>0)
110 LET H=H+1
115 GOTO 20
120 PRINT AT 1,7; T
125 STOP
```

```
CALCULATOR
 5 PRINT TAB 10; "CALCULATOR"
10 LET E=0
15 PRINT AT 2,0; "(14 SPACES)"; AT 2,0; E
20 INPUT E$
25 IF E$="" THEN GOTO 50
30 LET C=CODE E$
35 IF C>20 AND C<25 OR C=216 THEN LET E$=STR$ E+E$
40 LET E=VAL E$
45 GOTO 15
50 LET M=E
55 GOTO 15
```

```
MYSTERY PROGRAM
1 PRINT "(32 INVERSE SPACES)"
2 LET A=PEEK 16396+256*PEEK 16397+31
3 LET B=PEEK A=157
4 POKE A, 157-B
5 LET A=A-1
6 GOTO 2+B
```

Formula for success

Michael Dunn,
Hebburn, Tyne and Wear.

ZX-81

THIS SHORT routine runs on the 1K basic machine and calculates empirical chemical formulae from experimental data.

First, you are asked to input the number of elements in the compound. Then, in response to the prompt "Type data", you must type the symbol, relative atomic mass and percentage of the element present in the compound, from each element in turn.

The computer will then calculate the simplest formula from these inputs and print it out, with the ratios of atoms in brackets correct to two decimal places. The program can handle up to 10 elements. Press Newline to process another set of data.

The data is stored in three arrays A$(), A(), and B() and is input and processed via two loops. Line 105 prints out the empirical formula rounding down to two decimal places.

```
  0 REM   M.DUNN 1981
  5 CLS
 10 PRINT "HOW MANY ELEMENTS?"
 15 INPUT A
 20 PRINT "TYPE DATA"
 25 DIM A$(A,2)
 30 DIM A(A)
 35 DIM B(A)
 40 PRINT ,,"ELEMENT";TAB 10;"R
.A.M.";TAB 18;"P.C."
 45 LET C=100
 50 FOR N=1 TO A
 55 INPUT A$(N)
 60 INPUT A(N)
 65 INPUT B(N)
 70 PRINT A$(N);TAB 10;A(N);TAB
18;B(N)
 75 LET B(N)=B(N)/A(N)
 80 IF B(N)<C THEN LET C=B(N)
 85 NEXT N
 90 FOR N=1 TO A
 95 PRINT A$(N);"(";INT (B(N)/C
*100)/100;")";
100 NEXT N
105 INPUT B$
110 IF B$="" THEN GOTO 5
```

Clanger dropper

G Stephen,
Aberdeen.

ATOM

I WOULD LIKE to offer some corrections to the explanatory text of Zero Dropper in Software File, December 1981 and a correction to the program itself. In the text, the variable used was %N and not ZN, and in the explanation of the string format, it is not an equals sign but a decimal point. Line 10040 should read:

10040 I = 550

and the last statement in line 10030 may be dropped.

Doodlebug

Luc Fountain,
New Ash Green, Kent.

ATOM

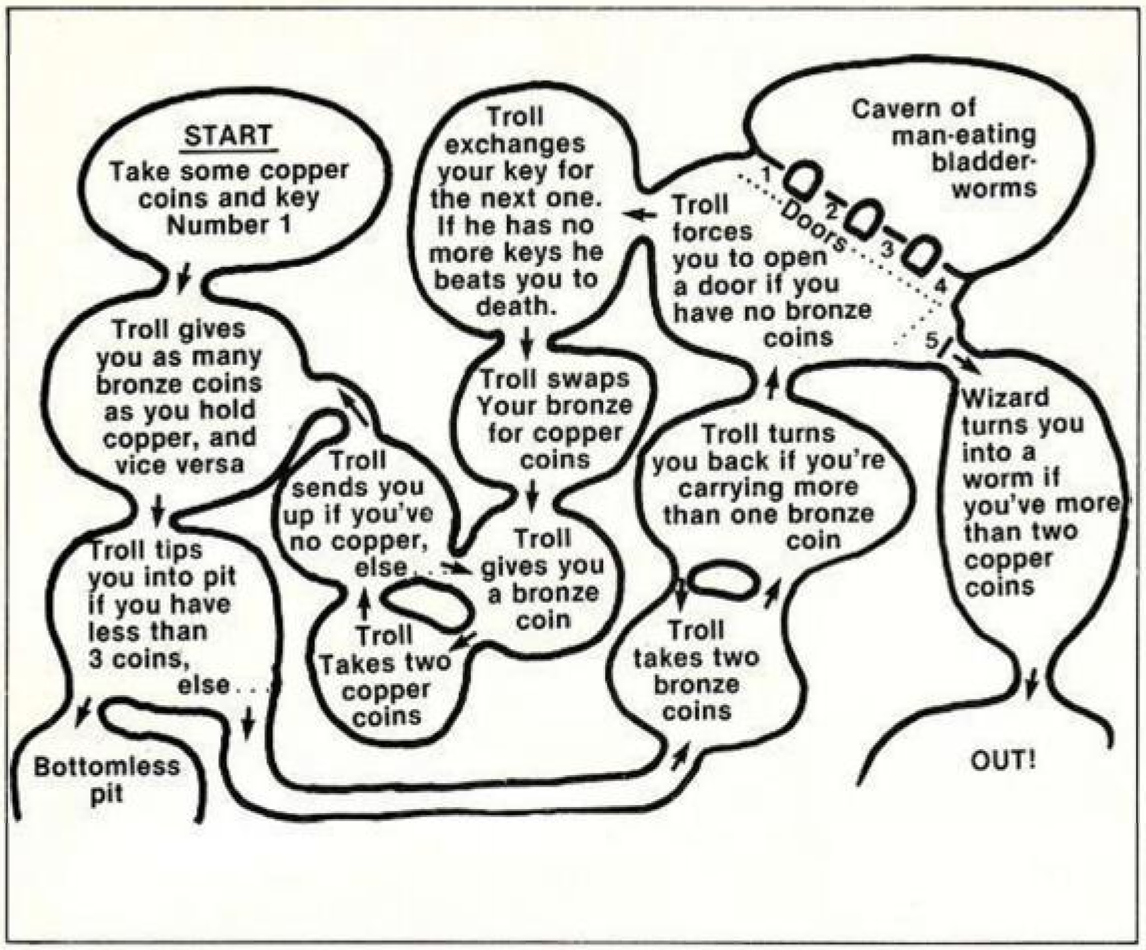
DOODLEBUG IS A brief and easy program for the Atom and occupies less than 6.5K. A line tracks across the screen, starting in the middle and rising. It can be steered diagonally up or down, using the CTRL, Shift and repeat keys. For example, to go diagonally down and left, press CTRL and repeat simultaneously.

The designs that appear can be frozen by pressing ESC, or alternatively by allowing the cursor to reach the top of the screen. To restart, press any letter or number key.

```
0 REM DOODLEBUG
1 REM BY LUC FOUNTAIN
2 REM
3 CLEAR 0
6 B=32
8 F.A=25TO50;IF?#B001<
129;B=B+1
10 IF?#B001&#40=0;B=B-1
14 IF?#B002&#40=0;A=A-2
16 MOVEB,A; DRAWB,A
18 WAIT;WAIT;WAIT;WAIT
19 N.A
20 LINK#FFE3;G.3
```

■

COMPETITION CORNER



Troll plague

BY ANTHONY ROBERTS

THERE ARE five exits from Ben-Heri's troll-infested cave system. Each is barred by a great iron door, opened by one of five keys. Only the fifth door leads to freedom and you need to have the correct key in order to escape. Ben-Heri gives you the first key and offers you some copper coins. All the trolls in the caves have their instructions and all passages are one-way.

How many copper coins should you take? Here is the cave map with the trolls' instructions.

A £15 book token will be awarded to the first correct solution drawn from the competition bag. All entries must be at the *Your Computer* offices by the last working day in February. The name of the winner, the solution, and a competition report will be published in the April issue of *Your Computer*.

If you want to set a competition for Competition Corner, remember that the simplest solution should be calculable by a short program rather than by any other form of reckoning.

Competition reports: solutions to ZX Printer crossword and the Christmas quiz

AN OVERWHELMING NUMBER of entries were received for the December ZX Printer crossword competition. More than 600 solutions were sent in and nearly all of them were correct. Picking a winner was a difficult task, but eventually we plumped for Roger Colyer of 37 West View Road, St Albans, Hertfordshire, AL3 5JX, for his "I will use the hard copy from my ZX Printer to make Prints Charming from an ugly prog". A ZX Printer is on its way.

Similar entries on this theme included W McQuarrie's "ensure that some day my Prints will come" and W Baker's "aluminate my roll and make my mark as the character Prints Charring". Other entries which caught the eye included T Collins — he must be a Tiswas fan — with "llist mmy pprograms. OO.KK". D Owen with "print Diana Dors in 2-D" and M Yates with "record and analyse the total vocabulary of my parrot".

Clive Sinclair proved to be a popular choice for readers wishing to let off steam. D Gawthorpe's entry "get something in writing from Sinclair at long last" summed up a number of people's feelings, as did Julian Stradling's "tie up the head of Sinclair customer service department". Pride of place, however, must go to Nick Willder's "generate those awful puns that win *Your Computer* crossword competitions". Awful puns? Shame on you.

No-one sent in a complete correct entry for the Christmas competition. Perhaps you found it too hard. For those who attempted the competition, the answers are as follows:

Round 1.
1. Bell Telephone announced it in 1948.
2. It was patented in 1893, six years after its invention, by Leon Bokee.
3. 1948; it was called the Selective Sequence Electronic Calculator.
4. 1950; it cost £250,000 in those days.
5. Blaise Pascal.
6. Electronic Numerical Integrator and Calculator, built 1945/6.

Round 2.
Doc, Dopey, Grumpy, Happy, Sleepy, Sneezy, Bashful.

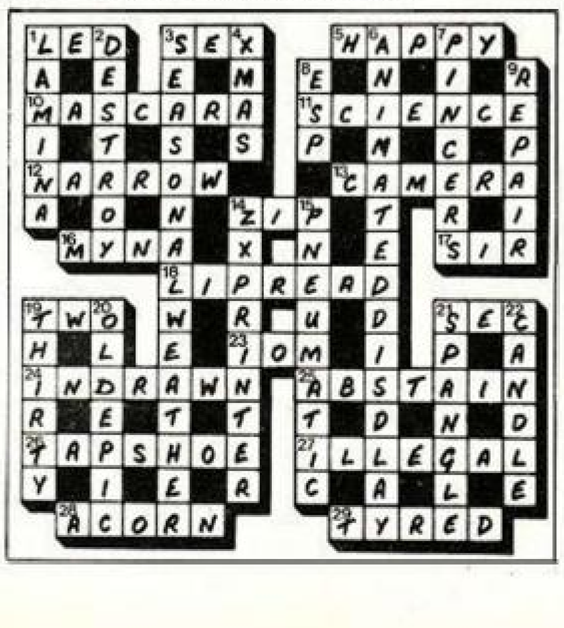
Round 3.
1. 60; n^2-4.
2. 3; numbers 1 to 10 in alphabetical order.
3. SE5EN; numbers five, six, seven with roman numerals substituted.
4. 2016; the next years commencing on a Friday.
5. 91; $\sum_{1}^{n}(x^2)$.

Round 4.
The relationship between all the machines, wires and screens is:

"CAT" PUCE	"ANT" MAUVE	"DOG" NAVY	"BAT" OCHRE
EBONY WIRE	HAZEL WIRE	GOLD WIRE	FAWN WIRE
KINGFISHER SCREEN	INDIGO SCREEN	JADE SCREEN	LILAC SCREEN

Hence, the kingfisher screen goes with the puce machine.

The December crossword solution.



Round 5.
1. "Invisible maniac" — Calder-Marshall, 1964.
2. Einstein.
3. Arnold Wesker.
4. The word is half Greek and half Latin.

Round 6.
1. "To be or not to be".
2. Singing in the rain.

Round 7.
It is a decimal "chop". It divides Y into X giving an integer answer Z and remainder A: X/Y = ZY + A.

Round 8.
The fifteen moves are:
S-G-F-Y-C-G-S-F-Y-C-S-F-Y-G-C-5 or
5-4-3-2-1-3-4-2-1-3-2-1-3-4-2-5

Round 9.
8208 and 9474 only.

Round 10.
1. Store Guide.
2. *Your Computer.*
3. The Acorn Atom.
4. Response Frame.
5. Commodore Pet.

The Exhibition Which Works For You

MICROSYSTEMS'82

WEST CENTRE HOTEL, LONDON FEBRUARY 24-26, 1982

Over 6300 quality visitors attended the 1981 show — providing the correct balance of users and specifiers of your products and services.
The formula is right — you can make MICROSYSTEMS '82 work for you by reserving your stand space NOW.

Find out how exhibiting at MICROSYSTEMS '82 can work for you by completing and returning the coupon now, to:
Exhibition Manager, MICROSYSTEMS '82, IPC Exhibitions Ltd., Surrey House, 1 Throwley Way, Sutton, Surrey SM1 4QQ.

MICROSYSTEMS '82 is sponsored by Computer Weekly, Systems International, Practical Computing, Your Computer, Computer Talk, Office Systems, Data Processing and Microprocessors and Microsystems and organised by IPC Exhibitions Ltd.

Please send details of exhibiting at MICROSYSTEMS '82, to:

Name ____________________

Position in company ____________________

Company ____________________

Address ____________________

____________________ Tel. No. ____________________

ZX-80 & ZX-81

THE FINEST MACHINE CODE FAST MOVING GRAPHICS ARCADE GAMES AVAILABLE

QS-DEFENDERS

UP/DOWN: THRUST/FIRE CONTROLS. FULL SCREEN DISPLAY. 84 fast moving chrs. 10 missiles. Attack waves. Moving surface. Generally considered to be the best arcade type game written for the ZX COMPUTERS. REQUIRES 3K RAM, 8K ROM

QS-ASTEROIDS

LEFT/RIGHT: THRUST/FIRE CONTROLS. 2 sizes of ASTEROIDS. Wrap-around screen. Full mobility of ship. Bonus ship. QUICKSILVA'S latest arcade game. As good as QS-DEFENDER. REQUIRES 4K RAM; 8K ROM + SLOW MODE

Both programs feature on screen scoring and have software to drive QS SOUND BD.

They are recorded twice on High quality cassettes and have FULL COLOUR cassette inserts of original paintings by 'STEINAR LUND'

HARDWARE

QS MOTHER BOARD and QS CONNECTOR
The heart of any expansion system. Features on board 5V regulator + two expansion sockets to take add on boards. Can be used in two ways.
1) ZX COMPUTER-CONNECTOR-ANY SINGLE ADD-ON (but no (extra RAM PACK)
2) ZX COMPUTER-CONNECTOR-MOTHER BD. (for two extra bds)
— ANY RAM PACK

QS 3K RAM BOARD (using 2114 I.C.s)
A 3K static Ram bd to fit ZX-80/81. Combines with original 1K to give 4K

QS SOUND BOARD (using AY-3-8910)
A 3 channel sound effects and music board easily programmed from BASIC

QS CHRS BOARD
A programmable Graphics generator giving user control of 128 different chrs. Comes with Demo cassette of M/C routines for easy use. Works with ZX PRINTER (Demo cassette available separately) to make use of Printers graphics possibilities).

NEW HARDWARE

QS HI-RES BOARD (available end of Jan). 256 × 192 PIXELS. SOFTWARE SELECT. 6K ON BOARD RAM. MIXED TEXT & GRAPHICS. RESIDENT HI-RES SOFTWARE IN ROM.

COMMANDS . . . MOVE x,y; PLOT x,y; DRAW x,y; PRINT X$; COPY; WHITE; BLACK; CLEAR
A MASTERPIECE of ZX design. Resident software in ROM provide extremely fast Hi-res facilities. No tedious cassette loading. No loss of valuable Ram space. Just instant HI-RES GRAPHICS. Screen may COPIED TO the printer.

All products fully guaranteed. Fully inclusive prices are as follows:
QS DEFENDER £5.50/QS ASTEROIDS £5.50/QS CHRS DEMO (separately) £3.50
QS MOTHER BD. £12.00/QS CONNECTOR £4.00/QS SOUND BD. £26.00
QS CHRS BD. £26.00/QS 3K RAM £18.00/QS 16K RAM £35.00/QS HI-RES £85.00
Cheques should be made payable to "Quicksilva" and sent to the following address

QUICKSILVA, 95 UPPER BROWNHILL ROAD, MAYBUSH, SOUTHAMPTON, HANTS.

Send S.A.E. for Catalogue and data sheets to above address.
COME AND SEE OUR FULL RANGE AT THE ZX MICROFAIR ON 30TH JAN 1982

S. ELECTRONICS ZX81 BUSINESS START UP PROGRAM

- Calculate how to maximise cash flow without overtrading.
- Program on cassette. Comprehensive manual describes how to adapt the program to a wide variety of applications.
- Business game based on the program included on the cassette.
- No memory expansion needed. Runs in the standard ZX81.
- Blank cassette on which to develop your own program included in the pack.

Send cheque/PO for £7 payable to:
S. ELECTRONICS
Box 321, YOUR COMPUTER,
Quadrant House, The Quadrant,
Sutton, Surrey SM2 5AS.

THE ZX81 POCKET BOOK

Also suitable for ZX80 with 8K ROM

- ADVENTURE
  - City of Alzan
  - Create your own
- TUNNELS & TROLLS
  BUCKET CATCHING
- JAWS
  BECOME AN ARTIST
- PRO-AM GOLF
  PUTTER
  ETCH-A-SKETCH
  FRUIT MACHINE
  With hold
  DIGITAL CLOCK
  DICE ROLLING
  plus 20 others

110 PAGE BOOK **5.95**

CASSETTE & BOOK **10.00**

★ require 16K RAM

PLUS Hints & tips on programming
Reference Sections

The ZX80 Pocket Book still available, prices as above.

ATOM BUSINESS

Suitable for the expanded Atom with floating point ROM

Each section contains:
- Management Summary
- Operating Instructions
- Program notes
- Source listing

SALES GRAPH
SALES RECORDS
NOMINAL LEDGER
QUEUEING SIMULATION
EXPENSE CLAIMS

ADDLIST
LABEL PRODUCTION
LEASE OR BUY DCF
METRIC CONVERSION
STANDARD DEVIATION
BUDGET FACTORING

110 Page Book **6.95**

Cassette & Book **15.57**

PHIPPS ASSOCIATES

3 DOWNS AVENUE EPSOM Surrey KT18 5HG
Tel: Epsom (03727) 21215 quoting your credit card reference. 24 Hour phone service. Air Mail delivery in Europe add 50p, air mail elsewhere add £1.50 per book. Prices shown above include UK postage and VAT on cassettes

Fuller FD System for ZX80/81

The Fuller FD System is not a ZX80 81 'add on' but uses the micro board at the heart of a more powerfull system, allowing the user to expand with the system or stop with a standard keyboard & case and using existing Sinclair 16K RAM pack etc, either way you only need a screwdriver to assemble the built items, the FD Keyboard simply plugs into the ZX81 which is screwed in position in the FD case, ZX80 installation requires soldering to the keyboard, we will carry out this work at a fixed charge of £10. SAE will bring you details of our products. Have your ZX81 Kit built by us FREE when you buy The FD Keyboard, Case and Power Supply at £42.95 + £2.25 p&p

- FD Keyboard Kit £18.95
- FD Keyboard Built £24.95
- FD 16K RAM Board £39.95
- Above items + 80p p&p
- FD Keyboard/Motherboard add £15.95 to Keyboard price
- FD Case £11.75 + £1.25 p&p

To **Fuller Micro Systems**
Sandfield Park East
Liverpool L12 9HP
For orders and enquiries tel. 051-236 6109
tick ZX80 ☐ 81 ☐

Please send me
I enclose a cheque/po for
Name
Address

HILDERBAY LTD PROFESSIONAL SOFTWARE

If you don't believe that the Sinclair ZX81 can do real work, come and see us at the ZX Microfair (30 January 1982, Central Hall Westminster, London SW1).

New product: a **RELIABLE** 16K RAM pack for the ZX81 £42

ZX81 + 16K Software:

* **Payroll** for up to 30 employees. Meets all regulations. £25
* **Stock Control.** Fast, big, and versatile. £25
* **Budget.** £15
* **Critical Path Analysis.** Up to 500 activities. £15
* **VAT + MORTGAGE + LOAN.** £8
* **GOLD:** an adventure game. Try it if you have a week to spare With another game on same tape. £8
* **Memotech** 48K memory module for ZX81.

More details in our January advert, or from us.
Prices include VAT, postage, and packing. COD orders £2 extra.

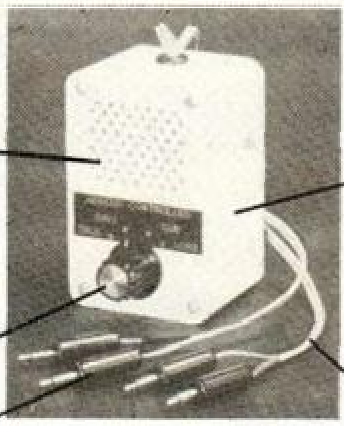
Tape hint: always rewind cassette, keeping their delicate recording surface protected inside the cassette housing.

HILDERBAY LTD (YCF) 8/10 Parkway, Regents Park, London NW1 7AA. Tel: 01-485 1059. Telex 22870

LOAD AND SAVE PROGRAMS FASTER AND MORE RELIABLY WITH THE

ABACUS CONTROLLER

A MUST FOR CASSETTE BASED MICROCOMPUTER SYSTEMS — ZX80/81, ACORN ATOM and APPLE



•BUILT IN MICROPHONE/ SPEAKER: Allows you to name programs before saving and cue up programs accurately ready for loading

•SINGLE SWITCH OPERATION: Load and save programs simply by selecting position on switch. Eliminates hum caused by leaving both jack plugs in cassette recorder.

•PLUG IN CONNECTIONS: Supplied ready to use, just plug into computer and cassette recorder — no need to remove plugs again. *Connections supplied to suit your computer.*

•SMART COMPACT CASE: White A.B.S. housing measuring 100 × 65 × 50mm.

•SIMPLE EASY TO FOLLOW INSTRUCTIONS: Connect up the controller in seconds — learn to use in minutes.

•70 CENTIMETRES OF CABLE: Allows you to site cassette recorder conveniently.

Send Cheque or P.O. for £12.00 including P&P to:

ABACUS ELECTRONICS
186, St Helens Avenue
Swansea, West Glam.
Tel: (0792) 50282

SILVERSOFT

personal computer software

16K/ZX81 SOFTWARE "STARTREK" AND NOW "SUPER-WUMPUS"

"16K STAR TREK": Full Graphics, 4-levels of play, 8 × 8 Galaxy, Starbases, Phasors, Photon Torpedos, Klingons, Romulans, Thomans, etc.

"16K SUPER-WUMPUS": Changing Caves, Super Bats, Earthquakes, Pits, Magic Arrows, Traps, etc.

"GAMES PACK 1": Starwars, Hammurabi, Graphic Lander, Mastermind, Minefield.

"GAMES PACK 2": 3D-Oxo, Pontoon, Android-Nim, Flight Simulator.

Prices: £4.95 inc. each cassette
£8.95 any two.

Programs wanted — excellent royalties paid.

Send large s.a.e. for details of our expanding range of software. Mail Order only.

Please quote Ref: YCJ1

SILVERSOFT
40 Empress Avenue
Ilford, Essex
01-518 0877

ZX81

MACHINE CODE DE-BUG/ MONITOR

- **ELIMINATE MACHINE CODE PROBLEMS** with "ZX-MC" — a new software debug/monitor for 16K ZX81.
- **ENTER, RUN & DEBUG** your machine code programmes independently of Basic commands. "ZX-MC" resides in RAM and leaves you 12½K of RAM to work with.
- **SAVE & LOAD AT DOUBLE SPEED** any specified area of RAM. At last you are freed from storing your machine code in Arrays or REM lines.
- **REGISTERS DISPLAY & BREAKPOINTS** to make debugging easier — PLUS many more useful commands.
- **A MUST FOR BEGINNERS & ADVANCED USERS** — concentrate on your machine code programs, not on how and where to store them.
- ZX-MC is supplied on a high quality cassette with a 32 page operating manual, for only
- SAE for further details.

PICTURESQUE **£6.50** *fully inclusive (C.W.O.)*

6 Corkscrew Hill, West Wickham, Kent BR4 9BB

PROGRAMS FOR ZX81

CASSETTE ONE

I had your Invaders/React cassette . . . I was delighted with this first cassette."

P. Rubython, London

"I have been intending to write to you for some days to say how much I enjoy the games on "Cassette One" which you supplied me with earlier this month. Please let . . . into the secret of your first time load every time!"

E.H., London

CASSETTE ONE SIDE ONE 1K MACHINE CODE PROGRAMS React, Invaders, Phantom aliens, Maze of death, Planet lander, Bug splat, Bouncing letters

CASSETTE ONE SIDE ONE 1K BASIC PROGRAMS I Ching, Mastermind, Basic hangman, Robots

CASSETTE ONE SIDE ONE has large screen versions of Invaders and Maze of Death, ready for when you get 16K.

All on CASSETTE ONE for **£3.80.**

CASSETTE TWO

Just out, ten 16K Basic games on one cassette.

OTHELLO	CRASH
AWARI	ROULETTE
LASER BASES	PONTOON
WORD MASTERMIND	PENNYSHOOT
RECTANGLES	GUN COMMAND

All on CASSETTE TWO for **£5.**

Recorded on quality cassettes, sent by first class post from:

Michael Orwin, 26 Brownlow Rd., Willesden, London NW10 9QL.

ADVERTISEMENT INDEX

Make the most of your Sinclair ZX Computer...

Sinclair ZX software on cassette.

£3.95 per cassette.

The unprecedented popularity of the ZX Series of Sinclair Personal Computers has generated a large volume of programs written by users.

Sinclair has undertaken to publish the most elegant of these on pre-recorded cassettes. Each program is carefully vetted for interest and quality, and then grouped with other programs to form a single-subject cassette.

Each cassette costs £3.95 (including VAT and p&p) and comes complete with full instructions.

Although primarily designed for the Sinclair ZX81, many of the cassettes are suitable for running on a Sinclair ZX80 – if fitted with a replacement 8K BASIC ROM.

Some of the more elaborate programs can be run only on a Sinclair ZX Personal Computer augmented by a 16K-byte add-on RAM pack.

This RAM pack and the replacement ROM are described below. And the description of each cassette makes it clear what hardware is required.

8K BASIC ROM

The 8K BASIC ROM used in the ZX81 is available to ZX80 owners as a drop-in replacement chip. With the exception of animated graphics, all the advanced features of the ZX81 are now available on a ZX80 – including the ability to run much of the Sinclair ZX Software.

The ROM chip comes with a new keyboard template, which can be overlaid on the existing keyboard in minutes, and a new operating manual.

16K-BYTE RAM pack

The 16K-byte RAM pack provides 16-times more memory in one complete module. Compatible with the ZX81 *and* the ZX80, it can be used for program storage or as a database.

The RAM pack simply plugs into the existing expansion port on the rear of a Sinclair ZX Personal Computer.

Cassette 1 – Games

For ZX81 (and ZX80 with 8K BASIC ROM)

ORBIT – your space craft's mission is to pick up a very valuable cargo that's in orbit around a star.

SNIPER – you're surrounded by 40 of the enemy. How quickly can you spot and shoot them when they appear?

METEORS – your starship is cruising through space when you meet a meteor storm. How long can you dodge the deadly danger?

LIFE – J. H. Conway's 'Game of Life' has achieved tremendous popularity in the computing world. Study the life, death and evolution patterns of cells.

WOLFPACK – your naval destroyer is on a submarine hunt. The depth charges are armed, but must be fired with precision.

GOLF – what's your handicap? It's a tricky course but you control the strength of your shots.

Cassette 2 – Junior Education: 7-11-year-olds

For ZX81 with 16K RAM pack

CRASH – simple addition – with the added attraction of a car crash if you get it wrong.

MULTIPLY – long multiplication with five levels of difficulty. If the answer's wrong – the solution is explained.

TRAIN – multiplication tests against the computer. The winner's train reaches the station first.

FRACTIONS – fractions explained at three levels of difficulty. A ten-question test completes the program.

ADDSUB – addition and subtraction with three levels of difficulty. Again, wrong answers are followed by an explanation.

DIVISION – with five levels of difficulty. Mistakes are explained graphically, and a running score is displayed.

SPELLING – up to 500 words over five levels of difficulty. You can even change the words yourself.

Cassette 3 – Business and Household

For ZX81 (and ZX80 with 8K BASIC ROM) with 16K RAM pack

TELEPHONE – set up your own computerised telephone directory and address book. Changes, additions and deletions of up to 50 entries are easy.

NOTE PAD – a powerful, easy-to-run system for storing and retrieving everyday information. Use it as a diary, a catalogue, a reminder system, or a directory.

BANK ACCOUNT – a sophisticated financial recording system with comprehensive documentation. Use it at home to keep track of 'where the money goes,' and at work for expenses, departmental budgets, etc.

Cassette 4 – Games

For ZX81 (and ZX80 with 8K BASIC ROM) and 16K RAM pack

LUNAR LANDING – bring the lunar module down from orbit to a soft landing. You control attitude and orbital direction – but watch the fuel gauge! The screen displays your flight status – digitally and graphically.

TWENTYONE – a dice version of Blackjack.

COMBAT – you're on a suicide space mission. You have only 12 missiles but the aliens have unlimited strength. Can you take 12 of them with you?

SUBSTRIKE – on patrol, your frigate detects a pack of 10 enemy subs. Can you depth-charge them before they torpedo you?

CODEBREAKER – the computer thinks of a 4-digit number which you have to guess in up to 10 tries. The logical approach is best!

MAYDAY – in answer to a distress call, you've narrowed down the search area to 343 cubic kilometers of deep space. Can you find the astronaut before his life-support system fails in 10 hours time?

Cassette 5 – Junior Education: 9-11-year-olds

For ZX81 (and ZX80 with 8K BASIC ROM)

MATHS – tests arithmetic with three levels of difficulty, and gives your score out of 10.

BALANCE – tests understanding of levers/fulcrum theory with a series of graphic examples.

VOLUMES – 'yes' or 'no' answers from the computer to a series of cube volume calculations.

AVERAGES – what's the average height of your class? The average shoe size of your family? The average pocket money of your friends? The computer plots a bar chart, and distinguishes MEAN from MEDIAN.

BASES – convert from decimal (base 10) to other bases of your choice in the range 2 to 9.

TEMP – Volumes, temperatures – and their combinations.

How to order

Simply use the order form below, and either enclose a cheque or give us the number of your Access, Barclaycard or Trustcard account. Please allow 28 days for delivery. 14-day money-back option.

sinclair ZX SOFTWARE

Sinclair Research Ltd, 6 Kings Parade, Cambridge, Cambs., CB2 1SN. Tel: 0276 66104.

To: Sinclair Research, FREEPOST, Camberley, Surrey, GU15 3BR. Please print
Please send me the items I have indicated below.

Qty	Code	Item	Item price	Total
	21	Cassette 1 – Games	£3.95	
	22	Cassette 2 – Junior Education	£3.95	
	23	Cassette 3 – Business and Household	£3.95	
	24	Cassette 4 – Games	£3.95	
	25	Cassette 5 – Junior Education	£3.95	
	17	*8K BASIC ROM for ZX80	£19.95	
	18	*16K RAM pack for ZX81 and ZX80	£49.95	
		*Post and packing (if applicable)	£2.95	
			Total £	

*Please add £2.95 to total order value **only** if ordering ROM and/or RAM.

I enclose a cheque/PO to Sinclair Research Ltd for £_____

Please charge my Access*/Barclaycard/Trustcard no.

**Please delete as applicable.*

Name: Mr/Mrs/Miss

Address:

YOC 02

VIC 20

See it here, buy it now

AT YOUR LOCAL BRANCH TODAY!

Birmingham Byteshop Computerland 94/96 Hurst St, B5 4TD **Tel:** 021 622 7149 **London** Byteshop Computerland 324 Euston Road London W1 **Tel:** 01-387 0505 **Nottingham** Byteshop Computerland 92A Upper Parliament St NG1 6LF **Tel:** 0602 40576 **Manchester** Byteshop Computerland 11 Gateway House Piccadilly Station Approach **Tel:** 061 236 4737 **Glasgow** Byteshop Computerland Magnet House 61 Waterloo St, G2 7BP **Tel:** 041 221 7409

A member of the Comart Group of Companies.