POPULAR Computing WEEKLY

10 June 1982 Vol 1 No 8

30p

BBC Paintbox

Reviews: Mazogs

ZX81 keyboard tone

The Fast One

Defining your own characters

More on Spectrum colour



ZX HARDWARE....

PROFESSIONAL 40 KEY KEYBOARD

- All legends and graphics in 2 colours
- No soldering to ZX81, just plug in
- Proper typewriter keys
- RAM/Printer compatible

Kit **£19.95** Built **£24.95** Case **£10.20**
Repeat key add on TBA

Range of in/out ports, music boards, motherboards, D to A converter boards. Write for catalogue

23 way double sided, gold female edge connector, wirewrap type **£2.95**

Male connector **£1.25**
Ribbon cable **£1.40** per metre

Mastering Machine code book **£5.50**
Programming for real applications **£6.50**
Tape for real applications **£11.25**

HARRIS & LOCKYER ASSOCIATES
(Sole distributors for Redditch Electronics)

Dept PW
33 Pedmore Close
Woodrow South
Redditch
Worcs
Tel: (0527) 24452

Prices include VAT + P&P
Overseas add £1.80 postage
Delivery 3 days for in stock items else allow 28 days
Send SAE for Free Catalogue
Official orders welcome

SIR COMPUTERS

8 bit input/output port for Sinclair Spectrum

4 channel analogue input for Joystick

Please write or phone for details:

38 Danycoed Road
Cyncoed
Cardiff

0222-759 015

ZX80 **ZX81**

JRS SOFTWARE

19 WAYSIDE AVENUE, WORTHING, SUSSEX BN13 3JU. Telephone: WORTHING 65691 (Evenings and Weekends only)

CASSETTE professionally recorded by SOUND NEWS STUDIOS **£4.95**

GAMES PACK — Beat this for value! 5 x 16K programs PLUS 2 x 1K programs. ($9.90)

3-D Battle (M/code-1K) — Fast-moving space battle with continuous count down of energy units left.

City Bomb (M/code-1K) — Destroy the buildings and land your plane. Your fuel has nearly gone and you circle the city lower and lower.

Warp Wars (Basic and M/code-16K) — Features realistic space craft moved by M code for instant response. (Previously sold at Microfair with Sweet Tooth for £4.95.)

Snake (Basic-16K) — A game of thought and skill. Pass through all the marked squares without crossing or doubling back on your path, but watch out for the expanding black blob. (Previously Microfair for £3.95.)

Sweet Tooth (Basic and M/Code-16K) — M code routines used to move your fat face round the screen and gobble the sweets.

PLUS Slalom and Black Holes. (Previously sold together for £4.95.)

An ESSENTIAL addition to your 1K RAM ZX81 (or ZX80 8K ROM)

TOOLKIT (written by PAUL HOLMES). Provides the following additional facilities:
Line renumber — you state starting number and increment value. Search and replace — changes every occurence of a character as you require. Free space — tells you how many free bytes you have left.

SPECIAL GRAPHICS ROUTINES
Hyper graphics mode — graphics never seen on a ZX81 before. Open — instantly sets up as many empty print lines as you require. (1K version only.) Fill — used in conjunction with Open fills your screen instantly with your specified character. Reverse — changes each character on your screen to its inverse video.

TAPE ROUTINE — provides a system **WAIT** condition until a signal is received in the cassette ear jack — many uses. All these routines are written in machine code and together make up only 164 BYTES of your precious RAM, an incredible achievement. The price is incredible, too! **Only £3.95 ($7.90)** for cassette, including FULL instructions and example programmes. ALSO available 16K version **only £4.95 ($9.90)** which includes all the above. PLUS GOTOs and GOSUBs included in-line renumber. Search for and list every line containing specified character. **16K VERSION**

As reviewed in *YOUR COMPUTER* March 1982
16K RAM PACK
£22.95 ($59.95)
Quite simply the best available + FREE 'Alien Attack' (7K-M/code) on a cassette — value **£5.75.**



Uses existing power supply (min. 600m.a.).
Compatible with printer.
No wobble problems.
Gold-plated edge connector for perfect contact with your ZX81. Normally despatched within 10 days of receipt of your order.
AND NOW — 64K RAM PACK
Same quality as the 16K, too, for perfect contact with your ZX81. **£62.95 ($125).**

NEW **GRAPHICS TOOLKIT.** (Another masterpiece by PAUL HOLMES.)
22 exciting MACHINE CODE routines that give you control over your screen as never before! **(ZX81 — 16K RAM ONLY)**

DRAW/UNDRAW draws or deletes your multi-character shape which is defined in a REM statement. You may define as many different shapes as you like and draw or undraw each at will at whichever screen position you choose.
FOREGROUND ON/OFF use this to 'protect' existing characters on your screen. When on new shapes will appear to slide behind and re-emerge from other shapes.
BORDER/UNBORDER Draws a border round the edges of your screen area. Edit lines can be used if required. Your border is protected when foreground is on.
FILL Fills any number of lines you specify, starting at any line you specify, by your chosen character.
REVERSE Converts all characters to their inverse bideo, control as in FILL.
PRINT POSITION CONTROLS
UP, DOWN, LEFT, RIGHT } Alter your next PRINT position in the direction indicated
EDITPRINT Moves next PRINT position to first edit line.

SCROLL facilities
UPSCROLL, DOWNSCROLL, RIGHTSCROLL, LEFTSCROLL } Scroll your screen in the direction indicated
ON SCREEN/OFF SCREEN turns your screen on or off.
BACKGROUND ON/OFF Fills your screen by your specified character.
When foreground is on existing information is unaffected and shapes will appear to pass in front of your background, without deleting it.
SEARCH AND REPLACE will search the screen for every occurence of the character you specify and replace it with your new character.
SQUARE draws a square or rectangle from your specified co-ordinates.
ALL these routines are in machine code for SUPER-FAST response! Simply load GRAPHICS TOOLKIT, which repositions itself at the end of your RAM, and then your own program (or key in a new one). GRAPHICS TOOLKIT uses only 2K of Your RAM and that includes space to load the programmers TOOLKIT described above (16K RAM version).

ALL FOR ONLY £5.95 ($11.90) **(amazing value from JRS)**

This includes a cassette with 2 copies of the program plus a comprehensive instruction booklet with examples.
NOTE: All prices are fully inclusive — send cheque of P.O. to JRS Software at above address

OVERSEAS CUSTOMERS PLEASE NOTE: Payment may be made in Sterling (Money Order available at your bank) or $U.S. (U.S.A. customers only). Prices quoted above are also export prices and include AIRMAIL postage.

POPULAR Computing WEEKLY

The Team

Editor
Duncan Scot

Reporter
David Kelly [01-930 3271]

Sub-editor
Peter Harvey

Designer
Eric Robbie

Editorial Secretary
Fiona McCormick

Advertisement Manager
David Lake [01-839 2846]

Advertisement Executive
Alastair Macintosh [01-930 3840]

Publishing Director
Nick Hampshire

Popular Computing Weekly,
Hobhouse Court, 19 Whitcomb Street,
London WC2
Telephone: **01-839 6835**

Published by Sunshine Publications Ltd.

Typesetting, origination and printing by Chesham Press, Chesham, Bucks

Distributed by S M Distribution
London SW9. 01-274 8611. Telex: 261643

© Sunshine Publications Ltd 1982

Subscriptions
You can have *Popular Computing Weekly* sent to your home: the subscription rate is **£19.95** per year, for addresses in the UK.

How to submit articles
Articles which are submitted for publication should not be more than 1000 words long.

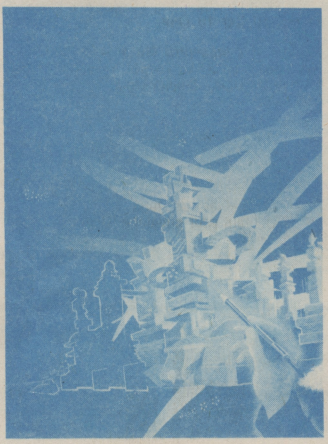
All submissions should be typed and a double space should be left between each line.

Programs should, whenever possible, be computer printed.

At present we cannot guarantee to return every submitted article, so please keep a copy.

Accuracy
Popular Computing Weekly cannot accept any responsibility for any errors in programs we publish, though we will always try our best to make sure programs work.

This Week



Cover illustration by Teoman Irmak

News 5
Prompt delivery for Beeb micros

Club Reports 7
Dave Kelly reports on the Longfield School computer club.

Paintbox 8
Brian Reffin Smith presents some colouring painting programs.

Reviews 10
Mazogs, keyboard audio tone.

Open Forum 13
Seven pages of programs.

Programming 20
Define your own BBC characters.

Spectrum 21
Nick Hampshire continues his look at ZX Spectrum colour.

Hand & mouth 22
Guessing roots and getting your assembler ORGanised

Sound & vision 23
More music for your micro.

Peek & poke 25
Your questions answered.

Competitions 26
Puzzle, crossword.

Editorial

When Clive Sinclair launched the Spectrum, he said, 'Delivery within 28 days.'

That was seven weeks ago.

Every day that passes more people who were counting the days to the arrival of their new computer call us up asking for help.

There is only so much we can do. We can mention here that the promised delivery dates of the Spectrum seem to have slipped.

More normally we are always willing to take up individual complaints from customers against suppliers.

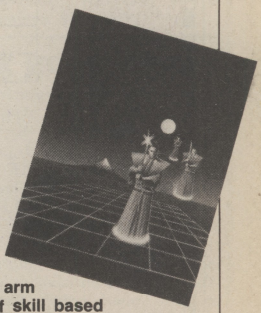
But when it comes to such a rush of complaints the responsibility necessarily has to fall back in the hands of you, the customer.

The answer is to add a clause to your order stating: 'I regard it as binding that the goods are delivered within 28 days.'

If your order and cheque are accepted so is the contract.

If the goods are not delivered, go to your Citizens' Advice Bureau, ask about the Small Claims Court procedure, and make a fuss.

Next Week



Chance your arm in a match of skill based on the ancient Japanese territorial game called GO: we call it Othello

Classified

20 SIMPLE ELECTRONIC PROJECTS FOR THE ZX81 and other computers

Make the most of your Microcomputer with this great book of construction projects. **£6.45.**

Available from:
National ZX Users' Club
44-46 Earls Court Road
London W8 6EJ

Please make your cheque payable to Interface.

SINCLAIR ZX81 plus big byte 16K, £82. Ring Walsingham (032872) 374.

TRS-80 4K LEVEL 1 CPU and monitor. Manuals, software. Perfect for beginner. £195. Tel: Ramsbottom 6254 after 6.30 pm.

ZX81 +16K, books, mags and games tapes, £90. 802-7937.

ZX81 16K RAM, 5 months' old + software, £85 ono. Tel: 025-482 2660.

ZX81 — MC : INPUT — prints on screen, flashing cursor and read/data, £3. M. Whitfield, 66 Bramblebury Road, London SE18.

VIC-20 16K and cassette recorder, over £15 software, worth £310, will accept £270 ono. Tel: 0773-813351.

16K ZX81 with five tapes, 3D Adventure, etc, in good condition with manual, £90 ono. Tel: 0904-797737.

THREE ZX81 PROGRAMS (Biorhythm, Rubic Cube and Tax) on cassette with full instructions. Send £3.25 to 23 Jolliffe Road, West Wittering, Sussex PO20 8ET. Make all cheques payable to S. Hibbert.

ZX81 KITS. Inverse video £2.95; built £3.55; repeat key £3.75; built £4.95. Memory crashes and top line slant cured, £2.95. Tel: 021-444 3034.

ZX81 16K RAM GAME CASSETTES: Get Them Before your Bosses get You! Nasty Invaders £4.95; Solve the Clues if you want to Escape! Very Nasty Mountain £6.95. Giltrole Ltd., Dept CW, PO Box 50, Rugby, Warks CV21 4DH.

ZX81 + 16K RAM PACK and software for £85. Tel: Bungay 3382.

TRS80 LEVEL II lower case mod fitted. Games, tapes, £220. Wickham Market, Suffolk. Tel: 0728 747241.

ZX81 (16K) machine. Sinclair built, £75. Tel: Winchester 65184 after 6 pm.

ZX81, SINCLAIR BUILT (new ROM), 16K RAM, tapes, £79. (04252) 71764.

VIC-20 plus cassette, game cartridge and magazines £180. Tel: 01-251 3769 after 5 pm.

BBC MICRO, model B with 18in colour TV, books, magazines and cassettes including 30 hour basic £550. Tel: 057 283 735.

MICROTAN 65 GRAPHICS TANEX, 8K RAM Xbug PSU £200 ono; 79 key keyboard, data encoding Eprom £30 ono. Tel: Kenilworth 58384.

SHARP PC1211 COMPUTER with printer £120 ono. Phone Barry 01-434 1365.

BBC MODEL B COMPUTER £435; Star matrix printer (Centronics interface) £190. Noyce, 18 George Street, Brighton.

ZX81 KEYBOARD, sounder and reverse display, P.C.B. and instructions £3.50 each or SAE for details to R. Mitchell, 20 Gorse Close, Portslade, Sussex.

SINCLAIR ZX81 plus 16K RAM plus tapes and books, very good condition, £80. Tel: 0252 317217 (evenings).

VIC-20 plus cassette unit, books. Many programs and joysticks, £260 ono. Tel: 01-892 9787.

ZX81 (16K) with books and tapes, very good condition, £85 ono. Tel: 0493 720774.

BBC MICRO: M/C Invaders, Breakout and Dodgems. All run on model A or B using sound and colour. Three programs on one cassette, £4 (inc). From: B. Cridland, 8 Bowness Avenue, Fleetwood, Lancs FY7 8PA.

ZX81 (16K) STORE/SORT PROGRAMME. Store, sort and order hundreds of items of information and amend as required. Tape and instructions, £4.95. J. Blackford, 63 Eltisley Avenue, Cambridge 3.

ZX81 (16K) FREE PROGRAM. Send cassette and SAE to Nick Godwin, 4 Hurkur Crescent, Eyemouth, Berwickshire.

ZX81 (16K) for sale, £80, ZX81 Chess, £4. Tel: 021-773 9521.

ZX81 plus 1.2A PSU and programs, £50. Tel: Maldon (Essex) 57713.

ZX81 (16K). Immaculate condition, seven months unexpired guarantee, £69. Weybridge 48401.

ZX81. One month old, £55. Dr. Jayadev, General Hospital, Aberdare, Mid Glamorgan.

ZX81 (16K) machine. Sinclair built, £75. Tel: Winchester 65184 after 6 pm.

CLASSIFIED ADVERTISING RATES:

Line by line: For private individuals, 20p per word, minimum 10 words.
For companies, traders, and all commercial bodies, 40p per word, minimum 20 words.
Semi-display: £10 per single column centimetre, minimum length 3 cm. (Please supply A/W as PMT. Or supply rough setting instructions.)
Conditions: All copy for Classified section must be pre-paid. Cheques and postal orders should arrive at least two weeks before the publication date.
If you wish to discuss your ad, please ring Alastair Macintosh 01-930 3840.

Here's my classified ad.

(Please write your copy in capital letters on the lines below.)

		Please continue on a separate sheet of paper	

I make this ..... words, at ..... per word so I owe you £.....

Name .....

Address .....

.....

.....

Telephone .....

Please cut out and send this form to: Classified Department, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2

Acorn clears its BBC backlog

Acorn has now cleared its backlog of orders for the BBC micro.

The company has dispatched 22,400 machines, leaving only some 5,000 orders outstanding. Production, at 6400 micros per month, should soon clear this backlog though orders are still being received at the rate of 4,000 per month.

Acorn stress that the production capacity is now well able to cope.

In addition to the much publicised production difficulties, caused by insufficient chip supply from Ferranti and, according to their spokesman, by snow in Wales, Acorn apparently misjudged both the extent and the type of demand for the machines.

The company thought, said the spokesman, that demand for the two machines, Models A and B, would be weighted towards the less expensive Model A in a ratio of about 3 to 1.

But orders favoured the Model B in about the same ratio and Acorn was not geared to manufacture that volume of Model B machines. To solve these difficulties, and in particular to ease production capacity problems of one of the machine's manufacturers, Cleartone in Gwent (the other is ICL), Acorn has appointed a third subcontractor, WNGS, based in Hong Kong.

Although Acorn has tried to produce a wholly British-built micro, WNGS were appointed to assemble some machines for export.

Now, to reduce the backlog of orders in this country, Acorn has been importing some part-assembled micros which are then completed (and the interfaces are added) and then quality-controlled by Cleartone.

The present situation on the availability of the BBC machines is as follows:

Model A — there is no delay and delivery is guaranteed within 21 days.

Model B — an order placed now would be fulfilled by the beginning of August.

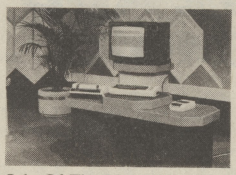
Micro sheets on offer

The BBC Computer Referral Service is offering a range of fact-sheets giving advice to micro owners.

There are five fact-sheets offering information on books, jobs in computing, micro-computers in small businesses, computers in education, and regional and national user groups.

The Referral Service, set up as an information exchange to handle letters arising out of BBC's Computer Programme, is open to all enquirers and already supplies a wide range of user information for any micro.

For any or all the leaflets, or for the address of your local club, contact BBC Computer Referral Service, Broadcasting Support Services, PO Box 7, London W3 6XJ.



Sale Of The Moment? A certain well-known computer finds its way on to ***that*** *programme.*

Acorn at your service

Acorn has chosen Retail Control Systems Ltd, of Middlesex, to provide a nationwide maintenance and service facility for all the Acorn micros, including the BBC Models A and B.

Retail Control Systems will operate in tandem with the 50 or so approved Acorn dealers who already provide a repair and back-up facility.



Plotting new courses for the PET . . . JJ Instrument's PD4.

JJ produce portable plotter for the PET

J J Instruments Ltd has produced a new low-cost X-Y plotter, primarily for use with the PET.

Called the PD4, it's A4 sized and designed to be easily portable. It is a development of the PL4 X-Y recorder.

David Sawyer, the company's product manager, explained, that the machine accepts digital commands via an IEEE-488 interface bus making the plotter compatible, in particular, with the Commodore Pet.

J J Instruments also supplies a software package in ROM form for use with the Pet, eliminating the programming required to command the plotter.

These commands control the pen, raising and lowering it, and provide the plotter with simple routines to, for example, draw lines between absolute or relative positions or draw shapes and characters.

The pen is capable of responding at speeds of up to 600mm s^{-1} and the PD4 plotter can recognise off-scale data, raising the pen, automatically.

The size and cost of the PD4 makes it useful for a wide range of applications in industry, education and commerce, and its dimensions make it particularly suited for use with a micro-computer.

The PD4 plotter plus interface costs £596, and comes complete with an instruction manual giving the setting-up procedures, guidance on its uses and sample programs. The software allowing the use of the PD4 with the PET costs an additional £65.

Both the PD4 plotter and software are available from: J J Instruments Ltd, Brook Avenue, Warsash, Southampton SO3 6HP.

What a Bleep-ing good idea!

If you have ever become infuriated by the touch keys on the ZX81, Fulcrum Products now offers a small unit that is designed to help.

It is called the ZX Bleep. When fitted to the ZX81 the module causes the micro to bleep every time a key is successfully depressed, thus making keying in possible without constant reference to the screen.

The Bleep is completely self-contained and is sufficiently small that it actually fits inside the ZX81 casing. It is easily installed, and no soldering is required.

The unit functions equally well in both the FAST and SLOW modes.

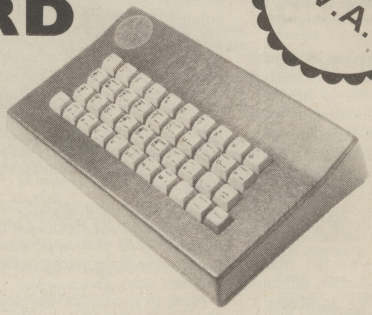
The ZX81 Bleep, together with fitting instructions, is available, within one week, from Fulcrum Products, Hillside, Steep Lane, Findon, West Sussex, price £8.95 (including VAT and postage and packing).

ZX80/1 PRICE WAR!

KAYDE

£37.95 incV.A.T.

ZX KEYBOARD FULLY CASED WITH REPEAT KEY



Fully cased keyboard	£37.95
Uncased keyboard	£27.95
Keyboard case	£10.95

This is a highly proffessional keyboard using executive buttons as found on top Quality Computers. It has a Repeat key and comes complete in its own luxury case. This is a genuine proffessional keyboard and should not be confused with toy keyboards currently available on the market.

16K RAMPACK

ORDER WITH YOUR CREDIT CARD

MASSIVE ADD ON MEMORY

£32.95 incV.A.T.

WHY WAIT TO PAY MORE

FAST IMMEDIATE DELIVERY

Post To
Dept CW3
KAYDE ELECTRONIC SYSTEMS
48/49 Exmouth Road
Great Yarmouth
Norfolk NR30 3DP
Tel 0493 57867

All products include VAT are fully built and tested and come with a complete Money Back Guarantee

PLEASE SEND ME.....	RAM PACK/S	£32.95 each
PLEASE SEND ME.....	CASED KEYBOARD/S	£37.95 each
PLEASE SEND ME.....	UNCASED KEYBOARD/S	£27.95 each
PLEASE SEND ME.....	KEYBOARD CASE/S	£10.95 each

I enclose £.....

NAME .....

ADDRESS .....

.....

.....

Please add £1.50 P&P and make cheques payable to Kayde Electronic Systems.

Club Reports

A lesson for the rest of the country?

***David Kelly** talks to Mike St John about his innovative school computer education club*

A strange ritual begins at Longfield School each Friday lunchtime. From 12 noon a small queue begins to form, grows, and by half past the hour stretches away out of sight. The computer club is about to convene.

Longfield is a comprehensive with 1390 pupils and Mike St John is the head of its Computer Department, one of the most advanced in Europe. Mike introduced me to Neil Hancock, the systems manager, a 15-year-old who is preparing for his O-levels next year. He explained how the system came into being and how it is operated.

In 1976 pupils at the school formed a computer club and, with help from the Parents' Association, the school purchased an Altair 8800 micro computer in the following year.

This computer had been carefully selected to meet the requirements of both the club and the CSE, O- and A-level computer studies curricula.

The Altair was bought in kit-form and members of the club put it together. Since then the computer studies department has never looked back.

When the system was inaugurated on October 27, 1977 there were less than 30 members of the club. Now there are nearly 200 pupils using the system.

This expansion was always planned and Neil explained that it was precisely for that reason that Mike chose the Altair. A multi-user machine was essential and Longfield now has 16 terminals in full-time operation.

The organisation of the department is unusual since it is run, as far as possible, as a commercial installation.

At the start of each term the computer department officers are elected. Neil pointed out some of the other pupils busy at the terminals, among them the software controller, chief software writer, supplies officer and engineers.

In common with many of the students, Neil is writing programs intended for business use in his spare time. At the moment he is developing a data handling package for a building company to enable them to produce work quotations quickly and easily.

I asked Neil what he hoped to do when he left Longfield. He frowned and said, 'I don't really know — something in computers.'

When the lesson in progress ended Mike St John was free to talk and he explained how he came to be in charge of the oldest established secondary school computer department.

When Mike first went to Longfield, from a job as a financial analyst, computer studies departments were unheard of. For a while the school had access to the main-frame computer at Mid-Kent College of Technology. But the only communication with the machine was by post.

'It was a bit like trying to learn to drive using pictures!' said Mike.

A computer in the school was essential but, with an initial budget for the department of only £50, donations from the Parents' Association and local businesses were much needed.

Now, using these gifts and money raised from sales of software produced by the pupils themselves, he has built up an enviable system. In five years the school has the most advanced micro system of any secondary school in Europe and Mike reckons they are the only school in the world to have a hard disk storage facility.

Mike firmly believes in an 'open house' policy. The computer room opens at 8.30 in the morning and closes at 9.30 in the evening. During that time there is rarely a free terminal and, as far as possible, he leaves the running of the system to the pupils.

Continuing his motoring metaphor, Mike explained why the young scholars have to be dragged from their terminals in the late evening:

'It's because the kids are driving,' he says, 'that's why the subject has taken off. Solving their own problems — they get a kick out of that.'

Mike is critical of the way computer studies are usually conducted in schools. As regional organiser of MUSE (Micro Users in Secondary Education) he is in a position to appreciate the general ignorance of computers in secondary education.



Eyes down . . . full house at Longfield

This ignorance, he feels, is shown by the mad scramble of some schools to offer computer education. And this, coupled with a fundamental lack of understanding, has produced generally lamentable examination results.

Over 5600 pupils took the AEB (Aldershot) O-level last year and the overall pass rate was 18 per cent.

Mike St John is justifiably proud that Longfield's results last year showed a 97 per cent pass rate.

As he comments, 'Nobody goes into unemployment from Computer Studies at Longfield.'

When I asked Mike if he minded if I took his photograph for this article he declined.

'It's not me you want to photograph — it's the kids.'

Mike indicated the classroom and terminals: 'This is theirs — not mine.'

We want to hear from you!

Whether you are starting a new club, holding a special meeting, or just changing the venue, we want to hear from you.

Write to David Kelly, Club News, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2 7HF or call him on 01-930 3271.

For your diary

Edinburgh ZX Computer Club meets at 7.30 pm on the second Wednesday of every month at the Claremont Hotel, Claremont Crescent, Edinburgh.

Occasional Saturday workshops and talks are held and there is a club newsletter and Software library.

Annual membership is £5 (£3 for students, children OAPs and the unwaged).

More information from Keith Mitchell, 19 Meadowplace Road, Edinburgh (031 334 8483).

Edinburgh ZX Club is also holding the first ZX fair in Scotland on July 24. The show, featuring over 30 exhibitors, is at Meadowbank Stadium and will be open from 10 am to 6 pm.

Further details from Gordon Hewit, 3 Baberton Mains View, Edinburgh (031 332 1163).

COVER STORY

Paintbox

Brian Reffin Smith explores the world of computer art

Nobody would ever deny that the quality and style of superior television graphics will always place the micro version into the realms of inferiority. But there's no need to have a complex about it.

To get those high quality graphics with crystal image and high-speed processing, you need to spend a fortune — tens of thousands of pounds! And if you could afford that you wouldn't be here, hoping to find ways of improving the graphics capabilities of your micro.

The fact is, all that high technology with its sky-high price tag is quite unnecessary. Computing quickly teaches that its not how you display it on the screen that's important but exactly what it is that you're displaying.

In other words, the message is the medium and *not,* as in the arty jargon of yesteryear, the converse. So all you have to do is adopt the same approach that a trendy illustrator uses when he's preparing a rough.

Then, in order to get his 'message' through, any quickly etched out sketch will do.

When you're using graphics on small systems, it's as well to remember this, the most direct, approach. Indeed, you can make a virtue out of this necessity to get the message through and go on to produce the basis of non-trivial graphics software package.

I have made two such systems, both available shortly. One is called *Jackson,* and runs on that ubiquitous educational and research computer, the 3802 from Research Machines.

Then there are two smaller programs, related to each other, soon to be produced by BBC Publications for the BBC micro.

One is concerned with *Drawing,* the other with *Painting.* These are only approximations to the sort of images that can be produced. They run on the Model A or B machine — so you can see how small and tight they have to be — running in the small amount of memory available in the Model A when graphics are used.

Later there is to be a third, large, general-purpose graphics package to run just on the Model B.

The *Jackson,* running on different principles to the BBC programs, is a visualisation aid. It is being used by a wide range of artists and designers at the Royal College of Art. I wanted it to be of use to, say, a textile designer (with its repeat patterns facility), someone wanting slides to illustrate a talk (information graphics in general), a painter, industrial designer, film-maker and so on.

Because of the possible combinations of memory, disc size and input device (advanced joystick or digitizing pad) there are several versions of *Jackson,* but it is hoped that the Government-funded MEP body will make them generally available in the education area.

The BBC programs are driven directly from the keyboard, using the function and cursor control keys.

Even though, on a micro, colours and numbers of pixcels (plottable points) are limited (there's no such thing as a true high-resolution micro) you can still 'say' a lot on a tv screen with a small computer — enough not to run out of ideas anyway.

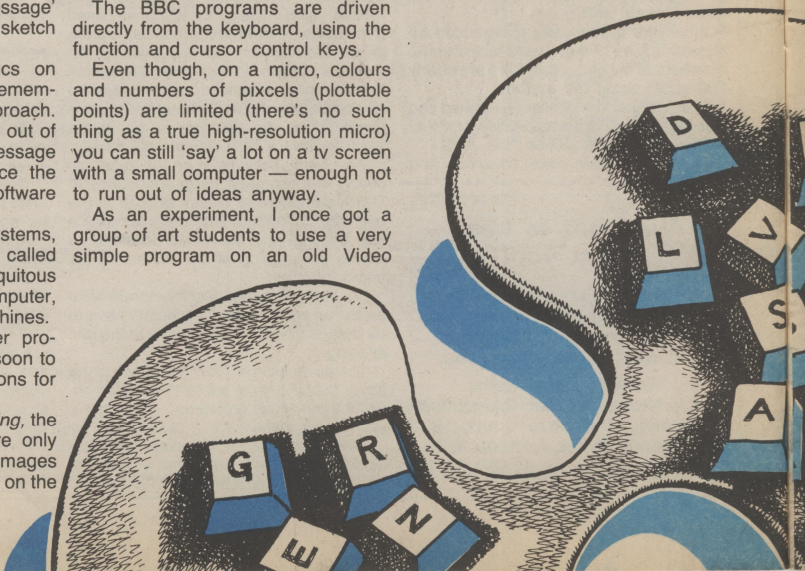
As an experiment, I once got a group of art students to use a very simple program on an old Video Genie, drawing with Ascii characters on the screen.

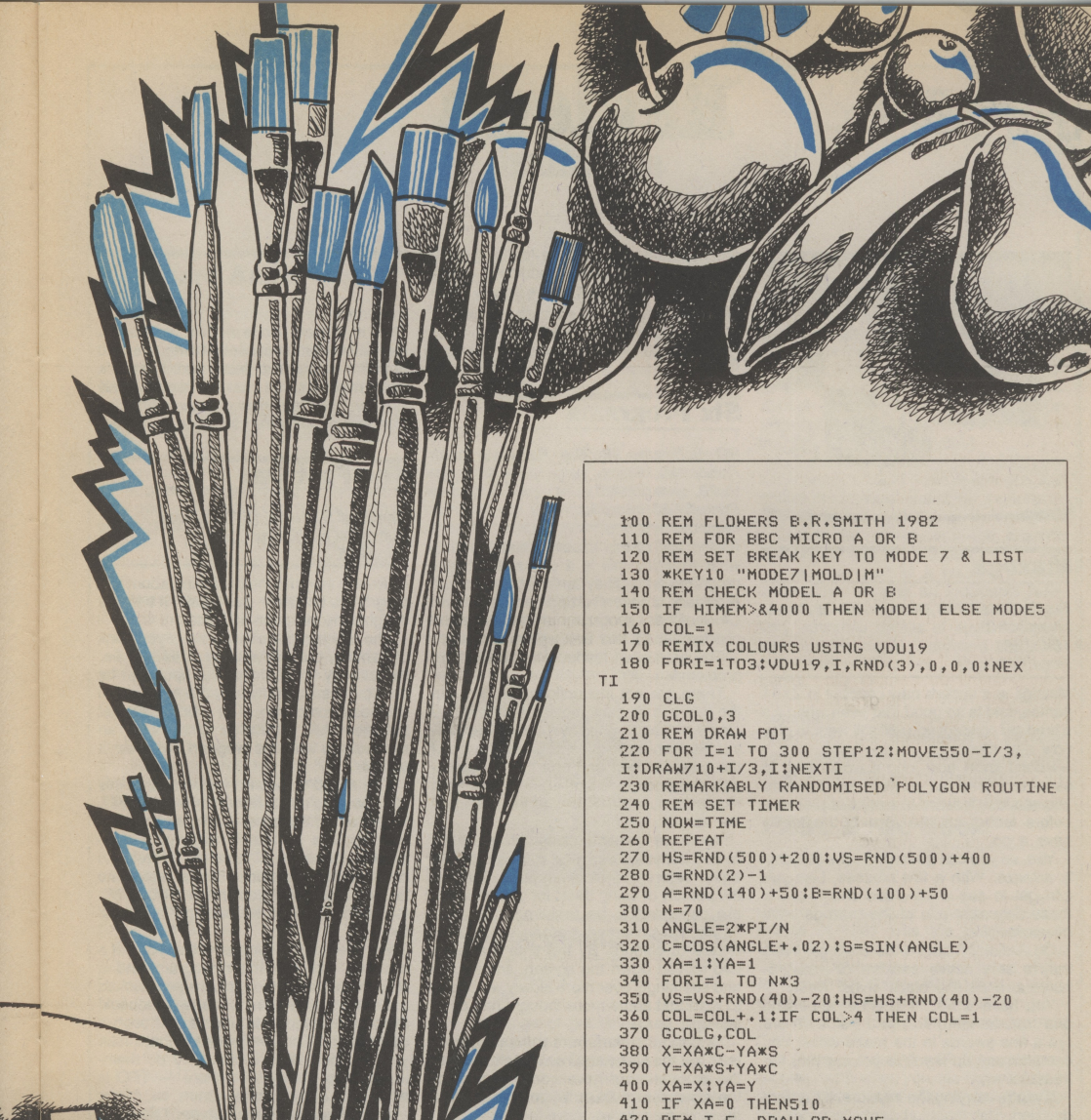
The letters could only appear in 'normal' print positions, unlike the BBC machine where they can be plotted with the same accuracy as graphics, yet even in this constrained, coarse environment, stylistic differences and expressions came through quite strongly.

Although a million pixcels and thousands of colours are ideal, the use of low-cost colour graphics systems has hardly been fully explored. And we are still limited far more by our imaginations than by the technology!

Finally, here is a short program based on some of the ideas used in the BBC programs, which sketches a pot-plant.

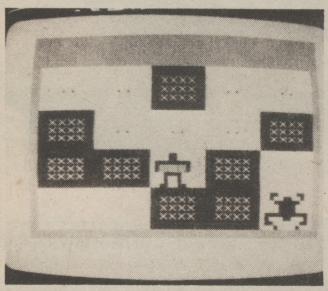
Whilst I would normally avoid such subject matter, I wanted to explore the possibilities of more tentative, less 'definite' graphics with the BBC machine, in this clichéd area. Try it and see if you think it's a step in that direction.





```
  100 REM FLOWERS B.R.SMITH 1982
  110 REM FOR BBC MICRO A OR B
  120 REM SET BREAK KEY TO MODE 7 & LIST
  130 *KEY10 "MODE7|MOLD|M"
  140 REM CHECK MODEL A OR B
  150 IF HIMEM>&4000 THEN MODE1 ELSE MODE5
  160 COL=1
  170 REMIX COLOURS USING VDU19
  180 FORI=1TO3:VDU19,I,RND(3),0,0,0:NEX
TI
  190 CLG
  200 GCOL0,3
  210 REM DRAW POT
  220 FOR I=1 TO 300 STEP12:MOVE550-I/3,
  I:DRAW710+I/3,I:NEXTI
  230 REMARKABLY RANDOMISED POLYGON ROUTINE
  240 REM SET TIMER
  250 NOW=TIME
  260 REPEAT
  270 HS=RND(500)+200:VS=RND(500)+400
  280 G=RND(2)-1
  290 A=RND(140)+50:B=RND(100)+50
  300 N=70
  310 ANGLE=2*PI/N
  320 C=COS(ANGLE+.02):S=SIN(ANGLE)
  330 XA=1:YA=1
  340 FORI=1 TO N*3
  350 VS=VS+RND(40)-20:HS=HS+RND(40)-20
  360 COL=COL+.1:IF COL>4 THEN COL=1
  370 GCOLG,COL
  380 X=XA*C-YA*S
  390 Y=XA*S+YA*C
  400 XA=X:YA=Y
  410 IF XA=0 THEN510
  420 REM I.E. DRAW OR MOVE
  430 IF I>1 THEN P=5 ELSE P=4
  440 PLOTP,A*XA+HS,B*YA+VS
  450 REM GRAPHICS WINDOW AROUND FLOWER
  460 VDU24,A*XA+HS;B*YA+VS;HS+200;VS+100;
  470 IF I=1 THEN 500
  480 REM PLOT 86=INVERSE COLOUR TRIANGLE
  490 PLOT86,HS,VS
  500 DRAWA*XA+HS,B*YA+VS
  510 NEXTI
  520 REMOVE GRAPHICS WINDOW
  530 VDU26
  540 REMEMBER THE PLANT STEMS...
  550 MOVEHS,VS
  560 DRAW530+RND(100),300
  570 REM DISGUISED GOTO!
  580 UNTIL TIME > NOW+24000
  590 PRINT"DONE"
```

Reviews
software



Mazogs

Bug-byte, 100 The Albany, Old Hall Street, Liverpool 3.
16K ZX81
Price £10

Mazogs is a superb new concept in ZX81 games, and is so good that even when it's a tired old concept you will still want to play now and then.

I don't know how long it would take to become a tired old concept, but I'm sure you can look forward to weeks of pleasure before letting anyone else in the family have a go.

Yes, as you can guess from the title, this is a maze. With a few plusses. Like you can get to see a fair-sized chunk of the maze only after one of the mazogs have overcome you.

Oh, I didn't mention the mazogs. Well, they're very active seekers of treasure-hunters, a bit like black elves. Nasty!

You have a 50:50 chance of beating one bare-handed. A sword helps. There are a few swords in the maze walls. You can pick one up if you're not carrying the treasure.

If you're carrying the treasure, it's a hard decision. Still, that big bar of silver may be worth less than your life. I guess.

There are blinking prisoners in the maze walls, too. It's their eyes that blink really. The prisoners can guide you a little way along the route to the treasure, or, when with it, back to safety. Nice prisoners, aren't they?

Anyway, it *is* possible to find the treasure (though it is always at least a hundred moves from start. Can be 400 it says here. Wow). And it *is* possible to get back safely with it.

But it's a big maze, and you can see very little of it in the normal view, and little more in the bird's-eye view, and, like I said, not all of it in the post-mortem.

Next game, of course, it's a different maze anyway.

Bug-byte provide a close-printed (and not well written) four-page leaflet to help you get into this grossly addictive game. How can *I* describe it in the space that's left?

Summary
Great. **KJ**

Six Programs

PR Software, 28 The Fairway, South Ruislip, Middlesex.
Vic20, unexpanded
Price £7.50 inc. p&p.

As the title of this cassette states somewhat explicitly this is a cassette with six programs on it. They are *War, Smashout, Blackjack, Logic, Pickupgame* and *Alarm Clock*. All the programs are for an unexpanded Vic20 and are written in Basic.

The games all loaded and ran without problems.

There are not instructions for *War,* but the game is so slow moving that one had plenty of time to guess what was happening. Red tanks are lined up along the top of the green screen and one has the occasional chance to shoot at the tanks from nine entrenched positions at the bottom of the screen.

You have to wait too long before the fire button becomes active and then too long again before anything happens. Such is the penalty of using Basic for games which are supposed to be real-time, or active.

The red on green was an unfortunate choice of colours. By the time the tanks had moved down the screen, line by line and seemingly ever more slowly we began to wish that some more thought had gone into this game.

Smashout suffers from the same problem. It is the same as *Breakout* except that the programmers have compensated for the slowness of Basic by making the bat three or four times the usual size.

Blackjack is quite straightforward except that there are no instructions. We had to play all the keys to discover that the S key let you stick and the H key would turn another card.

Logic is mastermind with the usual combination of five colours in a hidden sequence.

Pickupgame gives you a pile of stones and you and the computer take it in turns to pick-up a number of stones, each trying to avoid being left with the losing last stone.

The final program, *Alarm Clock,* is an odd choice for a games cassette. You enter the time, set the time for the alarm, and in due course the alarm goes off — a sort of flood warning type siren. Special features include ever-changing background colours and clock strikes every hour.

Summary
This is a poor collection of programs, reminiscent of the cassette being released onto the market two years ago. First time users might find the cassette useful, if only to give them an idea of how much the standard of software has since improved.

The Fast One

Campbell Systems, 15 Rous Road, Buckhurst Hill, Essex.
Price £15.

I'd love to say that the best thing about this program is its title — but I can't. If title is a delight, the program is better.

The title is not meant to make you think of gaming, but of business. This supplier has already a well-deserved reputation for data-processing software, the essence of business computing. *The Fast One* must enhance that reputation since it is an exceedingly versatile program and, wow, it *is* fast.

The speed comes from the fact that this is a great big string-handling routine. Well, it's the string that's big — all the data in all your records go into it. With your commands dealt with almost entirely in machine code, and up to 50K of immediate access storage (if you use a 64K RAM), this becomes an exceedingly powerful tool.

For your £15 (a very fair price) you get a cassette with the main program and a demonstration version, plus a jam-packed but fairly well-written 12-page handbook. Practise is needed to understand such a system, and Campbell reckons that'll take you an hour. Well, I think they do themselves less than justice there — TFO is extremely easy to use.

It is, of course, menu-driven. There are 11 main options — with everything you can think of except one — start again. Each option routine usually has a good number of sub-options on continuous display.

Thus you can define and re-define classification headings, fields and records, obtain screen display or printout of data in a range of forms, and dig up data to your heart's content.

There are, I know, a rapidly-increasing number of business users of the ZX81. This is almost certainly the best data-processing package yet available, particularly in regard to its speed. Sample it if you have data to process — and who hasn't?

Summary
A most impressive program, well documented and satisfyingly versatile. **KJ**

Reviews
hardware

Keyboard Audio Tone

TV Services of Cambridge Ltd., Chesterton Mill, Frenches Road, Cambridge CB4 3NP. Tel: 0223-358366. Price £8.95 assembled, £11.85 factory-fitted.

The main claim to fame for this firm has been that they do all of Sinclair's repair work on ZX80s and ZX81s etc.

They have now decided to put all their experience into producing a product of their own — a printed circuit board with all components soldered into place and only five wires need to be soldered to the printed circuit board of the ZX81 to make it work.

If this part worries you, just send your ZX81 to them (without the power pack and leads) for them to fit it for you.

The board is attached to a miniature loudspeaker and both of these are stuck to the inside of the ZX81s case underneath the keyboard.

If you have something there already such as a character board or extra memory, there is no problem. The KAT board is so thin that it will even fit underneath the ZX81's printed circuit board without obstructing the edge connector or touching any other components.

The KAT gives two tones to reassure the user, one when the key is pressed and another when the key is released. This will work on all keys and in all modes including FAST and SLOW.

The tone produced is quite loud, but if required the level of the tone can be increased. This is not the only use of this little unit, however, it can be programmed to give a beep during a program to simulate an explosion or to stimulate an input from the user.

On INPUT or typing in programs the unit works as normal, but if you are using INKEY$ then a PAUSE of at least five must be used before the INKEY$ to produce the tone.

The KAT will even help you with SAVEing and LOADing as it will beep at the beginning and end of a program, reminding the user that he needs to turn off the tape recorder. This is very useful, especially on long 16K programs as the beep is quite loud enough to be heard several feet away while you are making a cup of tea or something.

If you are fitting this device yourself it might be as well to fit a switch in the red (+5 volt) lead so that it can be switched off if not required.

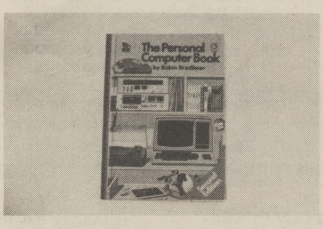
The board is easy enough to fit if you can solder, though I wish that the wires were labelled to say what pins they went to (although it is clear what colours they are it is not clear whether they are connected to power, etc).

The soldering requires a little care as the connections must be very short or they will connect to adjacent pins. There is also a strap on the board marked B and C to adjust the level of the tone, but I would think that the lowest level C was quite loud enough.

Summary

A very useful gadget which makes keyboard entry even on a Sinclair flat keyboard very easy. This firm have obviously put a great deal of thought into what the user requires and the two-tone system makes sure the user gets it right every time.

Another advantage is that they will fit it for you. Most ZX81 users do not like delving into the inside of their machines for fear of damaging some vital component. A very useful device which this reviewer will keep on his machine for a long time. **SA**



Personal Computer Book

By Robin Bradbeer
Price £5.95.

With a change of publisher and a great deal of up-dating this remarkably successful book has now appeared in a second edition, after no more than a couple of years. Of course, that has been a rather active couple of years so the new edition is entirely welcome.

The book comes in three parts, all useful and carefully written. Aimed principally at the person hesitating about dipping his or her toe into the computing torrent, those three parts are not consistent in value. Firstly we have several chapters on what a computer is, what it does, and how it does it. This is associated with an appendix on binary arithmetic.

Now Robin Bradbeer is a lovely chap and one of Britain's experts on home computing. He is, however, a high-level lecturer and *must* come to realise that home computer users, and especially people thinking of joining the club, are not going to need much of this knowledge. Indeed they will be put off if they find it in the opening chapters of a book.

Robin, why not relegate all the technical stuff to a small appendix in the third part, if you can't bear to omit it entirely!

The other two parts are exceedingly useful though, and will remain so long after the novice has become an expert.

The first is a comprehensive description of all the micros on the British home market — well, 58 of them, ranging from Microtan through ZX81, BBC and 480Z to the Ithaca DPSi (at £4,000).

In each case there is a photograph and a few paragraphs of good description, with details of software availability, peripherals and prices (end 1981).

Criteria for choice and the chapters associated with this section.

The other part is a wealth of useful appendices, making an invaluable reference bank for everyone. Here are brief but very adequate details of such things as interface standards, hardware suppliers, clubs and user groups, magazines and books.

Summary

Apart from the unnecessarily technical early material, this is a most valuable book for all — pre-beginners and experts. **ZZ**

ZX81 Basic Book

By Robin Norman
Price £5

One of the saddest Sinclair-associated stories of 1981 was the publication of Robin Norman's *Learning Basic with your Sinclair ZX80* within a week or so of the launch of the ZX81. Well Newnes have done it again. Robin Norman's ZX81 book preceded the Spectrum launch by less than a month.

Is there a moral there? Expect a new Sinclair micro when you see Newnes' book on the current one? Surely not that, but if there *is* a moral, it concerns publishing speeds and the speed of computer development. And it explains why grotty, poorly presented but rapidly provided books do so well.

For Robin Norman's works are indubitably not grotty or poorly presented. And as far as established publishers are concerned, they have both reached the bookshops in record time. Newnes are learning fast, I guess, and like everyone else they're working flat out on a book on the Spectrum, though this time not with Robin Norman at the helm.

Maybe that's a pity. Norman's new book is every bit as excellent for the newcomer as the previous one. He retains his delightfully readable but authoritative style, and is most thorough in developing his reader's understanding on Sinclair Basic. **KJ**

Our classifieds are faster.

Do you want to sell your computer and buy a bigger and better one?

Have you ever thought of trying to make some money out of selling tapes of your own programs?

Whatever it is you want to buy or sell why not use our classified pages?

It has to be better than waiting for up to nine weeks to get into one of the old monthly magazines.

Not only that, but our rates are very reasonable.

For private individuals it only costs 20p per word, with a minimum of 10 words.

We can make it so cheap because we charge companies using the classified columns 40p per word.

The classified pages can be used for semi-display advertising.

The cost for this is £10 per single column centimetre, with a minimum charge of £30.

All copy for the classified pages must be pre-paid. (You'll find a handy form on page 22).

Cheques and postal orders should be made out to *Popular Computing Weekly*. Your advertisement should arrive at least two weeks before the publication date.

If you have any queries regarding Classified or semi-display advertising please call Alastair Macintosh on 01-930 3840

Popular Computing Weekly.

The fast one.

Open Forum

Open Forum is for you to publish your programs and ideas.
It is important that your programs are bug free before you send them in. We cannot test all of them.
Contributions should be sent to: Popular Computing Weekly, *Hobhouse Court, 19 Whitcomb Street, London WC2H 7HF.*

How to contribute

Each week the editor goes through all the programs that you send to Open Forum in order to find the Program of the Week.

The author of that program will qualify for DOUBLE the usual fee we pay for published programs.

(The usual fee is £10.)

Then at the end of the month the four best programs of the week go forward to our amazing Program of the Month contest, for which there is a STAR prize.

This month the star prize is a super ZX printer, worth £59.95!

At the end of the year, all the best Programs of the Month will be entered in the super colossal competition, Program of the Year.

So send in your program today!

Presentation hints

Programs which are most likely to be considered for the Star Prize will be computer printed and accompanied by a cassette.

The program will be well documented, the documentation being typed with a double spacing between each line.

The documentation should start with a general description of the program and then give some detail of how the program has been constructed and of its special features.

Listings taken from a ZX Printer should be cut into convenient lengths and stuck down on to white paper.

Please enclose a stamped, self-addressed envelope.

Never a crossword

on ZX81

This program allows the user to do crosswords while at the same time, having the ability to correct any mistakes.

First the size of the crossword is input then each row is entered with a string of 0s and 1s with a 0 representing a space and a 1 is a block.

After the grid is completed each word is entered in the following way:

first the row letter
second the column letter
third "A" or "D" for across or down
fourth the word.

all the above are entered in one statement, eg ABDSINCLAIR displays the word Sinclair starting at row A and column B with the word going downwards.

The program can be stopped or the display copied at any time, with additions to the program it will be impossible to SAVE an uncompleted crossword on tape (perhaps using an array).

Never a crossword
By David Poole

```
  10 REM "NEVER A CROSSWORD"
  20 REM  BY  D.POOLE APRIL 1982
  30 SLOW
  40 PRINT AT 10,10;"CROSSWORD"
  50 PRINT AT 21,0;"DO YOU NEED
INSTRUCTIONS"
  60 INPUT A$
  70 IF A$<>"Y" THEN GOTO 300
  80 CLS
  90 PRINT "THIS PROGRAM ALLOWS
YOU TO DO"
 100 PRINT "CROSSWORDS WITHOUT T
HE CHANCE OF"
 110 PRINT "MAKING A MISTAKE."
 120 PRINT "FIRST ENTER THE SIZE
 OF THE CROSSWORD. NE
XT"
 140 PRINT "INPUT EACH ROW WHEN
REQUESTED."
 150 PRINT "AFTER SETTING UP THE
GRID EACH"
 160 PRINT "WORD IS ENTERED THUS
:-"
 170 PRINT
 180 PRINT "1ST THE ROW LETTER"
 190 PRINT "2ND THE COLUMN LETTE
R"
 200 PRINT "3RD ""A""=ACROSS OR
 ""D""=DOWN"
 210 PRINT
 220 PRINT "ALL ENTERED IN ONE S
TATEMENT"
 230 PRINT "EG:- ABDSINCLAIR"
 240 PRINT
 250 PRINT "TO STOP PROGRAM TYPE
 'STOP'"
 260 PRINT "TO COPY TO PRINTER T
YPE, COPY"
 270 PRINT
 280 PRINT "PRESS N/L TO CONTINU
E"
 290 INPUT A$
 300 CLS
 310 PRINT AT 21,0;"HOW WIDE IS
THE CROSSWORD"
 320 INPUT U
 330 PRINT AT 21,4;"LONG"
 340 INPUT L
 350 IF U>=27 OR L>=18 THEN GOTO
 300
 360 REM DRAW FRAME
 370 CLS
 380 PRINT TAB 2;
 390 FOR I=38 TO 37+U
 400 PRINT CHR$ I;
 410 NEXT I
 420 PRINT "-2ND"
 430 FOR I=1 TO U+2
 440 PRINT AT 1,I;CHR$ 136;AT L+
2,I;CHR$ 136
 450 NEXT I
 460 FOR I=38 TO 37+L
 470 PRINT AT I-36,0;CHR$ I;CHR$
 136;TAB U+2;CHR$ 136
 480 NEXT I
 490 FOR I=1 TO L
 500 PRINT AT 21,0;"ENTER LINE "
;CHR$ (I+37);" 0=SPACE,1=BLOCK"
 510 INPUT A$
 520 IF LEN A$<>U THEN GOTO 500
 530 FOR J=1 TO LEN A$
 540 PRINT AT I+1,J+1;CHR$ ((COD
E A$-28)*128);
 550 LET A$=A$(2 TO )
 560 NEXT J
 570 NEXT I
 580 PRINT AT 21,0;"PLEASE ENTER
 FIRST CODE AND WORD"
 590 INPUT B$
 600 IF B$="COPY" THEN COPY
 610 IF B$="STOP" THEN STOP
 620 PRINT AT 21,0;"
 "
 630 IF B$(3)="D" THEN GOTO 690
 640 IF B$(3)<>"A" THEN GOTO 590
 650 REM ACROSS
 660 PRINT AT CODE B$-36,CODE B$
(2)-36,B$(4 TO LEN B$)
 670 PRINT AT 21,0;"NEXT PLEASE"
 680 GOTO 590
 690 REM DOWN
 700 FOR I=1 TO LEN B$-3
 710 PRINT AT CODE B$-37+I,CODE
 B$(2)-36;B$(I+3)
 720 NEXT I
 730 GOTO 590
```

Volumes

on ZX81

Volumes will run in an unexpanded ZX81. It will calculate the volume of a shape given the correct data.

When run a list of possible shapes will appear with the data they require underneath. The data is in order of entry.

'W'=width
'L'=length
'H'=height
'R'=radius

You must first enter the first letter of the shape and then the data, eg, for a cone enter **"C"** then radius, then height.

```
 10 PRINT ,,,,"BLOCK-CONE-PYRAMID-TUBE-
    SPHERE",
"-WLH- -- RH- - - - WLH- - - - RH- - - -R"
 20 INPUT I$
 30 CLS
 40 IF I$="B" OR I$="P" THEN INPUT A
 50 INPUT B
 60 IF I$<>"S" THEN INPUT C
 70 PRINT ,,"VOLUME OF ";
 80 IF I$="B" THEN PRINT "BLOCK=";AxBxC
 90 IF I$="C" THEN PRINT "CONE=";.
    3xPIxBxx2xC
100 IF I$="P" THEN PRINT "PYRAMID=";
    AxBxCx.3
110 IF I$="T" THEN PRINT "TUBE="; PIxBxx2xC
120 IF 1$="S" THEN PRINT "SPHERE=
    "1 3xP1xBxx3
130 RUN
```

Mastermind

on BBC Micro

This program, for a Model A or Model B BBC Micro, plays a version of Mastermind, in which the computer selects a random 4-digit number, and you have to find it using your skill and judgement.

The computer gives you clues as to how close you are to finding the number by

Open Forum

Mastermind

By Mark Rogers

```
  10 REM ********************************
  20 REM **   Mastermind BBC (16K)     **
  30 REM ** By Mark D. Rogers 12/5/82 **
  40 REM **   (C) M.D.ROGERS 1982      **
  50 REM ********************************
  60 DIM DIGIT$(4),GUESS$(4)
  70 ON ERROR GOTO 200
  80 MODE 7:VDU 7
  90 VDU 23;8202;0;0;0;
 100 PROCsetup
 110 PROCnumber
 120 TIME=0
 130 goes=0
 140 REPEAT
 150 goes=goes+1
 160 PROCgetguess
 170 PROCcalculate
 180 UNTIL bull=4
 190 PROCresult
 200 MODE 7
 210 PROCagain
 220 IF answer$="Y" OR answer$="y" THEN 80
 230 MODE 7:VDU 7:END
 240 DEF PROCsetup
 250 PRINT''TAB(12)CHR$(130)CHR$(141)"MASTERMIND"'TAB(12)CHR$(130)CHR$(141)"MAS
TERMIND"'TAB(13)CHR$(130)"          "
 260 PRINT''CHR$(131)" Bull  =   Right number in right place"'CHR$(131)" Cow
=   Right number in wrong place"
 270 ENDPROC
 280 DEF PROCnumber
 290 R=RND(9000)+999
 300 R$=STR$(R)
 310 FOR X=1 TO 4
 320 DIGIT$(X)=MID$(R$,X,1)
 330 NEXT X
 340 redo=0
 350 FOR X=1 TO 4:FOR Y=1 TO 4
 360 IF X=Y THEN 380
 370 IF DIGIT$(X)=DIGIT$(Y) THEN redo=1
 380 NEXT Y:NEXT X
 390 IF redo=1 THEN 290
 400 DELAY=INKEY(150)
 410 ENDPROC
 420 DEF PROCgetguess
 430 VDU 23;29194;0;0;0;
 440 PRINT TAB(2,11)CHR$(134)"Enter your four-digit guess....     "
 450 VDU 31,34,11
 460 FOR X=1 TO 4
 470 *FX 15 1
 480 guess$=GET$:PRINT guess$;
 490 GUESS$(X)=guess$
 500 NEXT X
 510 redo=0
 520 FOR X=1 TO 4:FOR Y=1 TO 4
 530 IF X=Y THEN 550
 540 IF GUESS$(X)=GUESS$(Y) OR GUESS$(X)<"0" OR GUESS$(X)>"9" THEN redo=1
 550 NEXT Y:NEXT X
 560 IF redo=1 THEN 440
 570 VDU 23;8202;0;0;0;
 580 ENDPROC
 590 DEF PROCcalculate
 600 bull=0:cow=0
 610 FOR H=1 TO 4
 620 FOR J=1 TO 4
 630 IF GUESS$(J)<>DIGIT$(H) THEN 650
 640 IF H=J THEN bull=bull+1 ELSE cow=cow+1
 650 NEXT J
 660 NEXT H
 670 IF bull=1 THEN B$=" Bull and " ELSE B$=" Bulls and "
 680 IF cow=1 THEN C$=" Cow  " ELSE C$=" Cows "
 690 PRINT TAB(0,12)CHR$(134)bull;B$;cow;C$
 700 T=TIME:REPEAT UNTIL TIME-T>400
 710 ENDPROC
 720 DEF PROCresult
 730 ON goes GOTO 740,750,750,750,760,760,760,760,770,770,770,780,780,780:ELSE
790
 740 PRINT''CHR$(134)"                FLUKE!!":GOTO 800
 750 PRINT''CHR$(134)"        A VERITABLE EINSTEIN!!":GOTO 800
 760 PRINT''CHR$(134)"          GOOD GAME!!!":GOTO 800
 770 PRINT''CHR$(134)"            OK, I SUPPOSE!!!":GOTO 800
 780 PRINT''CHR$(134)"           BIT SLOW, EH?":GOTO 800
 790 PRINT''CHR$(134)"      WHY DID IT TAKE SO LONG?":GOTO 800
 800 IF goes=1 THEN G$=" go" ELSE G$=" goes"
 810 PRINT''CHR$(131)"    It took ";INT(TIME/100);" seconds and ";goes;G$
 820 T=TIME:REPEAT UNTIL TIME-T>1000
 830 ENDPROC
 840 DEF PROCagain
 850 *FX 15 1
 860 VDU 23;8202;0;0;0;
 870 PRINT TAB(11,10)CHR$(131)"Would you like"
 880 PRINT TAB(9,11)CHR$(131)"another go?  (Y/N)"
 890 answer$=GET$
 900 IF answer$<>"Y" AND answer$<>"y" AND answer$<>"N" AND answer$<>"n" THEN 89
0
 910 T=TIME:REPEAT UNTIL TIME-T>150
 920 ENDPROC
```

telling you if you have any bulls or cows. A bull is a correct number in the correct place, and a cow is a correct number in the wrong place.

Thus, if the computer chose '3672' as its number, and you inputted '3826', the computer would say: '1 bull and 2 cows'.

The number chosen by the computer consists of four *different* digits, so that it would not choose for example '3646'. Any illegal entry inputted by the user is rejected by the computer.

Many of the programming techniques used in this program I discovered while working through the BBC's 'Welcome' Tape.

The program runs in the Teletext mode, Mode 7, and colours are used throughout. As with many of the programs on the Welcome tape, the main 'core' of the program is at the beginning, with all the procedure definitions at the end (line 240 onwards).

The computer starts by changing mode, beeping, and turning the cursor off (lines 80 and 90). It then prints the title and the description of bulls and cows (PROCsetup) and chooses its random number (PROCnumber).

It then starts the clock, and goes into a REPEAT UNTIL loop of getting the user's guess, and calculating the number of bulls and cows (PROCgetguess and PROCcalculate).

It repeats this until the user has found the number (when bull=4), and then displays a suitable message, relative to the number of guesses taken (PROCresult).

The user is then asked if he wants to go again (PROCagain), and the game accordingly either ends or starts again.

Music 2

on ZX81

In the second issue I read Peek & poke and saw the article on forming music on the ZX81. I typed this into my 1K ZX81 and found it very surprising. I have made a similar but improved program, called *Music 2.*

Target practice

on BBC Micro

In each run you have 10 attempts to hit the target with a shot from a gun the other side of a barrier. The routine in lines 220-250 plots the parabolic trajectory (you can add lines to bring in air friction and wind if you wish).

For each run, gun and target are placed at random and the barrier has random height and width.

The program employs the BBC feature of:

Open Forum

Music 2
By Lee Hayden

```
 10 GOSUB 100 x (INT (RND x 9)) + 1
 20 RUN
100 FOR N=1 TO RND x 100
110 SLOW
120 FAST
130 NEXT N
140 RETURN
200 FOR N=1 TO RND x 100
210 SLOW
220 FAST
230 NEXT N
240 RETURN
300 FOR N=1 TO RND x 100
310 SLOW
320 FAST
330 NEXT N
340 RETURN
400 FOR N=1 TO RND x 100
410 SLOW
420 FAST
430 NEXT N
440 RETURN
500 FOR N=1 TO RND x 100
510 SLOW
520 FAST
530 NEXT N
540 RETURN
600 FOR N=1 TO RND x 100
610 SLOW
620 FAST
630 NEXT N
640 RETURN
700 FOR N=1 TO RND x 100
710 SLOW
720 FAST
730 NEXT N
740 RETURN
800 FOR N=1 TO RND x 100
810 SLOW
820 FAST
830 NEXT N
840 RETURN
900 FOR N=1 TO RND x 100
910 SLOW
920 FAST
930 NEXT N
940 RETURN
```

P.S. don't forget to turn the volume up!

Target practice
By Eric Deeson

```
 10 VDU 23, 225, 3, 6, 60, 40, 104, 60, 126, 225          (define characters
 20 VDU 23, 226, 36, 90, 156, 90, 60, 155, 24, 60
 30 VDU 23, 227, 24, 36, 66, 153, 153, 66, 36, 127
 40 VDU 23, 228, 28, 42, 85, 170, 127, 170, 85, 225
 50 MODE 5                                                (set up playing area
 60 GCOL 0, 130
 70 GCOL 0, 0
 80 VDU 5
 90 CLG
100 A = RND(4) - 1                                        (set up random sites
110 B = RND(4) + 15
120 C = RND(9) + 1
130 D = 0                                                 (go count
140 REPEAT                                                (go loop
150 D = D + 1
160 PROC G(A,B,C,)                                        (set up characters
170 MOVE 10, 976: INPUT "Angle", A1
180 MOVE 10, 936: INPUT "Speed", E
190 AN = AL*PI/180                                        (convert to radians
200 X = 8
210 REPEAT                                                (trajectory loop
220 X1 = X + 64 * (A + 1)
230 Y = 32 + X * TAN(AN) - X]2/CE 2 * COS(AN)]2)
240 PLOT 69, X1, Y + 32
250 X = X + 16
260 UNTIL X>1280 OR Y< = 32 OR POINT (X1 + 9, Y + 32) = 0 (end trajectory cases
270 PRINT CHRS (226)                                      (detonate
280 FOR F = 1 TO 1E4: NEXT                                (delay
290 CLG
300 UNTIL D = 10 OR ABS (X1 = B * 64)<40                  (end go cases
310 IF D = 10 THEN PRINT TAB (4,10); "USELESS
    ELSE PRINT TAB (5,10); "GOOD SHOT" '*''
    "You got it in ";D;"."
320 FOR F = 1 TO 1E4:NEXT
330 PRINT "HAVE ANOTHER GO...."; GOTO 90                  (restart
340 DEF PROC (A.B.C)
350 LOCAL D
360 PRINT TAB (A, 30); CHRS (225)                         (gun
370 PRINT TAB (B, 30); CHRS (226)                         (target
380 FOR D = 1 TO C: PRINT TAB (5,30 - D + 1);
    STRINGS (C, CHRS (228))*NEXT                          (barrier
390 ENDPROC
```

```
 70 REM COLOURAMA
 80 BB=INT(254*RND(1))+1
 90 POKE 3687BB
100 S=7680: =(128*RND(1))+1
105 P=INT(255*RND(1))+1
110 X=INT(10*RND(1))+1
115 N=INT(44*RND(1))+1
120 FOR I=0 TO 506 STEPN
130 POKE S+30730+X+I,C
140 POKE S+X+I,P
160 NEXT I
180 FOR J=1 TO 50: NEXT J
200 PRINT " "
210 GOTO 80
```

Colourama
By Chris Palmer

Colourama
on Vic-20

The purpose of the program is to demonstrate varied animation across and down the screen using the poke statement. Random numbers are used deliberately for flexibility but these could be changed in order to repeat a specific action or character required. The program itself or any such adaptation could also make a useful subroutine.

The variables used are:
BB = Background/Board Colour.
P = character poke number.
X = point where appears.
N = frequency of the character on the screen.
S = Start of screen memory.
C = Character colour.
I = limits of screen area.
J = speed of action.

◯ definable characters (10-40);
◯ mixed text and graphics (80-170 etc);
◯ random number generation (100-120);
◯ REPEAT ... UNTIL (the two loops);
◯ POINT (260);
◯ procedure rather than sub-routine (160, 340-39;);
◯ STRING$ (380).

If you have any comments about Open Forum or about the kind of programs you'd like to see in this section, please write to us.

The address is:
Popular Computing Weekly
Hobhouse Court,
19 Whitcomb Street,
London WC2

Open Forum

Froggy

on ZX81

The idea of the game is to jump from lily pad to lily pad down the screen and through the gate at the bottom.

But it is not as easy as that, half-way down there is a snail endlessly crossing the screen and the only way to get to the bottom is to jump on his back and then off again where there is a lily pad near.

If you take too long on the snail he will drop you off the edge of the screen.

Also there is a bird who is determined to get you. The only way to escape its clutches is to make jumps from side to side when it gets near.

The game has five lines and gets harder every sheet. On the third and fifth sheet the background colour changes and on the fifth sheet a bonus frog is awarded.

The controls are as follows:

8 right one
5 left one
S right three
A left three
L Leap (RND 3, 5, or 7 down)
J Jump (2 down)
B Back (2 up)

If you are in a jam there is a hyperspace, key H, along the horizontal line of your frog. And if reaching the gate seems impossible there is a New Sheet button (key N) which draws a new pond.

The only problem with the new sheet button is that your frog goes down a level.

If the game seems too hard the bird can easily be removed or the number of frogs (F) increased.

Froggy
By Graham Plowman

```
   1 LET F=5
   2 LET A$="-"
   3 LET W=150
   4 LET L=1
   5 FAST
   6 DIM Q(31,21)
   7 FOR O=1 TO 20
   8 PRINT AT O,0;A$;A$;A$;A$;A$
;A$;A$;A$;A$;A$;A$;A$;A$;A$;A$;A
$;A$;A$;A$;A$;A$;A$;A$;A$;A$;A$;
A$;A$;A$;A$;A$;A$
   9 NEXT O
  11 LET AX=1
  12 LET AY=15
  13 PRINT AT AX,AY,"▒▒"
  14 LET Q(AY,AX)=1
  15 LET Q(AY+1,AX)=1
  40 LET A=1
  50 LET B=15
  70 FOR P=0 TO W
  80 LET AX=INT (RND*20)+1
  90 LET AY=INT (RND*30)+1
 100 IF AX=11 THEN GOTO 80
 110 IF Q(AY,AX)=1 THEN GOTO 80
 120 PRINT AT AX,AY;"▒▒"
 130 LET Q(AY,AX)=1
 135 LET Q(AY+1,AX)=1
 140 NEXT P
 150 FOR O=2 TO 20 STEP 2
 160 PRINT AT O,0;A$;A$;A$;A$;A$
;A$;A$;A$;A$;A$;A$;A$;A$;A$;A$;A
$;A$;A$;A$;A$;A$;A$;A$;A$;A$;A$;
A$;A$;A$;A$;A$;A$
 169 NEXT O
 170 PRINT AT 21,0;"████████████
████████████████████"
 171 LET E=1
 172 LET G=INT (RND*28)+1
 173 PRINT AT 21,G-2;"▌  ▛"
 174 PRINT AT 11,0;"
"
 175 SLOW
 176 LET CY=INT (RND*30)+1
 177 LET CX=19
 180 LET T=300
 190 LET A=1
 200 LET B=15
 210 PRINT AT 0,0;"FROGGY  LEVE
L";L;" LIVES ";F;" BY G.P"
 300 PRINT AT A,B;"█"
 305 LET V=0
 310 LET AO=A
 311 PRINT AT CX,CY;"▣"
 312 LET U=0
 313 LET CMM=CY
 314 LET CXX=CX
 315 IF CX>A THEN LET CX=CX-1
 316 IF CX<A THEN LET CX=CX+1
 317 IF CY<B THEN LET CY=CY+1
 318 IF CY>B THEN LET CY=CY-1
 320 LET BO=B
 330 PRINT AT 11,E;"▄█O"
 340 LET EO=E
 350 LET B=B+(INKEY$="8")-(INKEY
$="5")
 360 IF INKEY$="J" THEN LET A=A+
2
 365 IF INKEY$="H" THEN LET B=IN
T (RND*30)+1
 370 IF INKEY$="B" THEN LET A=A-
2
 375 IF INKEY$="N" THEN GOTO 900
 380 IF INKEY$="A" THEN LET B=B-
3
 390 IF INKEY$="S" THEN LET B=B+
3
 400 IF INKEY$="L" THEN GOSUB 50
0
 405 PRINT AT 11,E;"    "
 410 LET E=E+1
 420 IF E=29 THEN LET E=0
 430 PRINT AT AO,BO;" "
 450 IF Q(B,A)=1 THEN LET V=1
 460 LET T=T-1
 470 IF T<=0 THEN GOTO 600
 480 IF A=11 THEN GOSUB 700
 485 IF A=21 THEN GOTO 800
 487 PRINT AT CXX,CMM;" "
 488 IF A=CX AND B=CY THEN GOTO
600
 490 IF V=1 THEN GOTO 300
 495 GOTO 600
 500 LET R=INT (RND*7)
 510 IF R=1 THEN GOTO 500
 520 IF R=3 THEN GOTO 500
 530 IF R=5 THEN GOTO 500
 540 LET A=A+R
 550 RETURN
 600 LET F=F-1
 610 PRINT AT A,B-1;":X:"
 620 IF F>0 THEN GOTO 176
 630 PRINT AT 21,0;":::::::::::::
SPLASH::::::::::::::"
 635 PRINT AT 0,0;"::::::::::::::S
PLASH::::::::::::::"
 640 IF INKEY$="R" THEN RUN
 650 GOTO 640
 700 IF B=E THEN GOTO 750
 710 IF B=E+1 THEN GOTO 750
 720 IF B=E-1 THEN GOTO 750
 730 IF B=E+2 THEN GOTO 750
 740 GOTO 600
 760 LET B=B+1
 765 LET V=1
 770 IF B=29 THEN LET B=1
 780 RETURN
 800 IF B=G-1 THEN GOTO 850
 810 IF B=G+1 THEN GOTO 850
 820 IF B=G THEN GOTO 850
 830 GOTO 600
 850 PRINT AT A,B;"▣"
 860 PRINT AT 0,0;"██████████████
████████████████"
 870 PAUSE 300
 871 LET L=L+1
 872 IF L=5 THEN LET F=F+1
 873 IF L=3 THEN LET A$="▌"
 874 IF L=5 THEN LET A$="▖"
 875 LET W=W-10
 880 GOTO 5
 900 LET L=L-1
 910 GOTO 5
```

Correlation

on ZX81

Here is a program to analyse straight line graphs. This 16K program will analyse linear lines to give a correlation coefficient and regression analysis. It is of particular value to people working with graphs that have scattered values that might lie on a straight line, eg statisticians, biologists, students etc.

It is written in simple basic using almost entirely LET and PRINT statements. It does have the advantage of being extremely user friendly.

By entering the X and Y values of a succession of data points, a correlation coefficient can be calculated to show how close to a perfect straight line the date is. A regression analysis will give the properties of that line and if needed a list of X and Y values can be given for the perfect straight line.

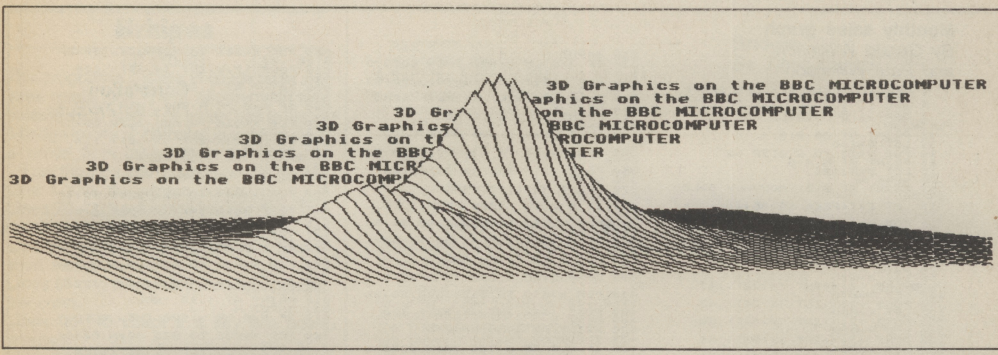
There is almost no limit to the number of values that can be entered.

Run it and see!

Correlation
By Lee Frances Cooper

```
   1 REM L COOPER
   2 REM CORRELATION COEFFICIENT
     AND REGRESSION ANALYSIS OF
LINEAR RELATIONSHIPS
   9 LET T=0
  10 LET X=0
  11 LET Y=0
  12 LET XS=0
  13 LET YS=0
  14 LET XY=0
  15 LET N=0
  16 LET AY=0
  17 LET AX=0
  18 LET XI=0
  19 LET YI=0
  20 PRINT ",,DO YOU NEED INSTRU
CTIONS ?"
  25 PRINT "ENTER Y FOR YES,N FO
R NO"
  35 IF INKEY$="N" THEN GOTO 70
  36 IF INKEY$="Y" THEN GOTO 40
  37 GOTO 35
  40 CLS
  41 PRINT ,,"THIS PROGRAM WILL
ANALYSE LINEAR LINES IF THE X,Y
VALUES OF DATA POINTS IS ENTERED
."
  45 PRINT ,,"IF -0 IS ENTERED F
OR X THEN      ANALYSIS OF THE DA
TA WILL OCCUR"
  50 PRINT ,,,,"PRESS ""Y"" TO C
ONTINUE"
  55 IF INKEY$="Y" THEN GOTO 70
  60 GOTO 55
  70 CLS
  99 CLS
 100 PRINT "CORRELATION COEFFICI
ENT"
 101 PRINT "████████████████████
███",,,"LAST X=";X;"  LAST Y=";Y
 102 PRINT ,,,,"INPUT X "
 110 INPUT XI
 115 IF XI=-0 THEN GOTO 500
 120 PRINT "X=";XI
 125 PRINT ,,"INPUT Y"
 130 INPUT YI
 140 PRINT "Y=";YI
 150 PRINT ,,"ARE THEY CORRECT ?
 Y OR N "
 160 IF INKEY$="Y" THEN GOTO 210
 172 IF INKEY$="N" THEN GOTO 179
 175 GOTO 160
 179 CLS
 180 PRINT "TRY AGAIN",,"ENTER C
ORRECT VALUES PLEASE"
 185 PAUSE 50
 190 GOTO 100
 210 CLS
 220 LET X=XI
 230 LET Y=YI
 240 LET N=N+1
 250 LET AX=AX+X
 260 LET AY=AY+Y
 270 LET XS=XS+X*X
 280 LET YS=YS+Y*Y
 290 LET XY=XY+X*Y
 300 GOTO 100
 500 PRINT ,,"DO YOU WISH TO STA
RT CALCULATION ?",,"Y OR N"
 505 IF INKEY$="Y" THEN GOTO 600
 510 IF INKEY$="N" THEN GOTO 528
 520 GOTO 505
 529 PRINT ""
 530 PRINT "DO YOU WISH TO ",,,,
"START AGAIN (A)",,,,"STOP (S)",
,,," CONTINUE WITH INPUT (C)"
 535 LET T=0
 540 IF INKEY$="S" THEN GOTO 699
 550 IF INKEY$="A" THEN GOTO 0
 560 IF INKEY$="C" THEN GOTO 99
 570 GOTO 540
 600 LET R=0
 610 CLS
 615 LET A=0
 616 LET B=0
 617 LET C=0
 620 LET A=XY-(AX*AY)/N
 625 LET B=XS-(AX*AX)/N
 630 LET C=YS-(AY*AY)/N
 640 LET R=A/(SQR (B*C))
 650 PRINT ,,"CORRELATION COEFFI
CIENT",,," R=";R,,,,"SAMPLE SIZE
 =";N
 660 PRINT "DO YOU WISH TO",,,,"
CONTIUE(C)",,,," FINISH (F)",,,,
"START AGAIN (S)",,,,"REGRESSION
 ANALYSIS (R)"
 665 IF INKEY$="C" THEN GOTO 99
 670 IF INKEY$="S" THEN GOTO 0
 680 IF INKEY$="R" THEN GOTO 799
 690 IF INKEY$="F" THEN GOTO 700
 695 GOTO 670
 700 CLS
 701 PAUSE 100
 703 PRINT AT 10,10;"GOODBYE"
 704 PRINT AT 11,10;"███████"
 705 STOP
 799 CLS
 800 PRINT "ANALYSIS OF Y=MX+C",
"████████████████",,,,"CORREL
ATION COEFFICIENT=";R
 805 PRINT ,,"SAMPLE SIZE=";N
 810 PRINT ,,"M= ";A/B,,"C= ";(A
Y/N)-(A/B)*(AX/N)
 820 PRINT ,,"WOULD YOU LIKE VAL
UES OF X AND Y ?   Y OR N"
 830 IF INKEY$="Y" THEN GOTO 849
 835 IF INKEY$="N" THEN GOTO 700
 840 GOTO 830
 849 CLS
 850 PRINT AT 2,0;"INPUT THE FIR
ST VALUE OF X YOU  WANT"
 860 INPUT IX
 861 PRINT "=";IX
 865 PRINT ,,"INPUT THE LAST VAL
UE OF X YOU   WANT"
 870 INPUT FX
 871 PRINT "=";FX
 875 PRINT ,,"INPUT THE SPACE, O
R INTERVAL     BETWEEN VALUES OF
X YOU WANT"
 880 INPUT I
 885 PRINT "=";I
 890 LET Y=0
 895 CLS
 899 PRINT "X","Y",,,
 900 LET Y=((A/B)*IX)+(AY/N)-(A/
B)*(AX/N)
 920 PRINT IX,Y
 930 LET IX=IX+I
 940 IF IX>FX THEN GOTO 950
 945 GOTO 900
 950 PRINT "ENTER ""B""  TO BEGI
N AGAIN OR ""S"" TO STOP"
 960 IF INKEY$="B" THEN RUN
 965 IF INKEY$="S" THEN GOTO 700
 970 GOTO 960
 980 STOP
```

Open Forum



Hi-res graphics

on BBC Micro

The procedure 'gdump' will print the whole screen in mode 0 or mode 4 on to paper using a printer with high res capability. The actual listing is for a Seikosha GP80 but the following notes should allow transfer to other printers.

VDU1 stops the next character from affecting the display.

VDU8 is the code to indicate graphics information on the way.

The graphics information is sent in seven bits. The eighth most significant bit is always set to 1 to indicate that the info is not a control code, hence line 1140. A border is made by lines 1030, 1060, 1170 and 1200.

The main body of the program is built around the FOR X loop (1090 to 1130), where the procedure works its way up then across the screen, transferring the screen data into seven bit codes (the printer works on a seven by five matrix) (line 1110).

The changes to print mode 0 graphics are simply to take into account that the X axis is twice that in mode 4.

Because the mode 0 X resolution is 640, the procedure prints the screen with a 90 degree shift, ie, it prints down the paper instead of across because the printer can only cope with 475 dots across the page.

Since the procedure will probably be the last thing the program will do, there is no need to declare the variables as local.

Hi-res graphics
By G. L. Jones

```
1000 DEF PROC9dump
1010 REM**MODE4 GRAPHICS DUMP FOR BBC MICRO**
1020 VDU2:VDU1:VDU8
1030 FOR L=0TO257:VDU1:PRINTCHR$(192);:NEXT L
1040 FOR D=0TO1279STEP28
1050 VDU1:VDU10:VDU1
1060 PRINT CHR$(255);
1070 FOR Y=0TO1023STEP4
1080 PSEND=0:INC=1
1090 FOR X=(D+4)TO(D+28)STEP4
1100 PO=POINT(X,Y)
1110 IF PO=1 PSEND=PSEND OR INC
1120 INC=INC*2
1130 NEXT X
1140 PSEND=PSEND OR 128
1150 VDU1:PRINT CHR$(PSEND);
1160 NEXT Y
1170 VDU1:PRINTCHR$(255);
1180 NEXT D
1190 VDU1:VDU10
1200 FORL=0TO257:VDU1:PRINTCHR$(129);:NEXT:VDU1:PRINT
1210 VDU3
1220 ENDPROC
1230 REM**TO CHANGE TO MODE 0 DUMP**
1240 REM**1040 FOR D=0TO1279STEP14
1250 REM**1090 FOR X=(D+1)TO(D+14)STEP2
```

Calculator

on ZX81

I have recently written this program which converts my ZX81 into a calculator.

All the numerical functions are useable. The user needs to type in the question for the computer.

The program has a self-running routine so type GOTO 9998 instead of SAVE.

Calculator
By Simon Erhardt

```
   1 PRINT "
D R
   8 LET B$=""
   9 LET M=0
  10 INPUT A$
  11 CLS
  12 IF A$="" THEN PRINT "TRY PU
TTING IN A NUMBER NEXT TIME....."
"
  13 IF A$="" THEN RUN
  20 LET B$=A$
  30 PRINT A$;"=";VAL A$
  40 PAUSE 500
  50 GOTO 10
  60 LET M=VAL B$
  70 GOTO 40
  80 PRINT M
  90 GOTO 10
9998 SAVE "CALCULATOR"
9999 RUN
```

Open Forum

Monthly sales graph

By Gerald Wilkinson and Paul Wilkinson

```
    1 REM ** MONTHLY SALES GRAPH
    2 REM ** WRITTEN FOR 16K ZX81
    3 REM ** IN MAY 1982
    4 REM ** BY GERALD AND PAUL W
ILKINSON
   11 PRINT AT 20,2;0
   15 LET B=100
   16 FOR N=10 TO 2 STEP -2
   17 PRINT AT N,0;B
   18 LET B=B+100
   19 NEXT N
   17 PRINT AT 0,5;"MONTHLY SALES
GRAPH"
   19 LET Z=1+PEEK 16396+256*PEEK
16397
   20 FOR X=643 TO 667
   30 POKE (Z+X+INT (X/32)),3
   40 NEXT X
   50 FOR X=611 TO 67 STEP -32
   60 POKE (Z+X+INT (X/32)),5
   70 NEXT X
   80 LET X=576
   90 POKE (Z+X+INT (X/32)),47
  100 LET X=578
  110 POKE (Z+X+INT (X/32)),43
  120 LET X=580
  130 POKE (Z+X+INT (X/32)),50
  140 LET X=602
  150 POKE (Z+X+INT (X/32)),38
  160 LET X=604
  170 POKE (Z+X+INT (X/32)),50
  180 LET X=606
  190 POKE (Z+X+INT (X/32)),47
  200 LET X=608
  210 POKE (Z+X+INT (X/32)),47
  220 LET X=590
  230 POKE (Z+X+INT (X/32)),38
  240 LET X=592
  250 POKE (Z+X+INT (X/32)),56
  260 LET X=594
  270 POKE (Z+X+INT (X/32)),52
  280 LET X=596
  290 POKE (Z+X+INT (X/32)),51
  300 LET X=598
  310 POKE (Z+X+INT (X/32)),41
  311 PRINT AT 1,7;"ENTER JAN SAL
ES"
  312 INPUT J
  313 IF J<0 OR J>900 THEN GOTO 3
12
  320 FOR X=612 TO 612-32*J/50+32
STEP -32
  330 POKE (Z+X+INT (X/32)),128
  340 NEXT X
  341 PRINT AT 1,13;"FEB"
  342 INPUT F
  343 IF F<0 OR F>900 THEN GOTO 3
42
  350 FOR X=614 TO 614-32*F/50+32
STEP -32
  360 POKE (Z+X+INT (X/32)),128
  365 NEXT X
  370 PRINT AT 1,13;"MAR"
  371 INPUT M
  372 IF M<0 OR M>900 THEN GOTO 3
72
  380 FOR X=616 TO 616-32*M/50+32
STEP -32
  390 POKE (Z+X+INT (X/32)),128
  395 NEXT X
  400 PRINT AT 1,13;"APR"
  401 INPUT A
  402 IF A<0 OR A>900 THEN GOTO 4
02
  410 FOR X=618 TO 618-32*A/50+32
STEP -32
  420 POKE (Z+X+INT (X/32)),128
  425 NEXT X
  430 PRINT AT 1,13;"MAY"
  431 INPUT M
  432 IF M<0 OR M>900 THEN GOTO 4
32
  440 FOR X=620 TO 620-32*M/50+32
STEP -32
  450 POKE (Z+X+INT (X/32)),128
  455 NEXT X
  460 PRINT AT 1,13;"JUN"
  461 INPUT J
  462 IF J<0 OR J>900 THEN GOTO 4
62
  470 FOR X=622 TO 622-32*J/50+32
STEP -32
  480 POKE (Z+X+INT (X/32)),128
  485 NEXT X
  490 PRINT AT 1,13;"JUL"
  492 INPUT J
  493 IF J<0 OR J>900 THEN GOTO 4
92
  500 FOR X=624 TO 624-32*J/50+32
STEP -32
  510 POKE (Z+X+INT (X/32)),128
  515 NEXT X
  520 PRINT AT 1,13;"AUG"
  521 INPUT A
  522 IF A<0 OR A>900 THEN GOTO 5
22
  530 FOR X=626 TO 626-32*A/50+32
STEP -32
  540 POKE (Z+X+INT (X/32)),128
  545 NEXT X
  550 PRINT AT 1,13;"SEP"
  551 INPUT S
  552 IF S<0 OR S>900 THEN GOTO 5
52
  560 FOR X=628 TO 628-32*S/50+32
STEP -32
  570 POKE (Z+X+INT (X/32)),128
  575 NEXT X
  580 PRINT AT 1,13;"OCT"
  581 INPUT O
  582 IF O<0 OR O>900 THEN GOTO 5
82
  590 FOR X=630 TO 630-32*O/50+32
STEP -32
  600 POKE (Z+X+INT (X/32)),128
  605 NEXT X
  610 PRINT AT 1,13;"NOV"
  611 INPUT N
  612 IF N<0 OR N>900 THEN GOTO 6
12
  620 FOR X=632 TO 632-32*N/50+32
STEP -32
  630 POKE (Z+X+INT (X/32)),128
  635 NEXT X
  640 PRINT AT 1,13;"DEC"
  641 INPUT D
  642 IF D<0 OR D>900 THEN GOTO 6
42
  650 FOR X=634 TO 634-32*D/50+32
STEP -32
  660 POKE (Z+X+INT (X/32)),128
  670 NEXT X
  680 PRINT AT 1,3;"DO YOU WANT A
COPY ? (Y/N)"
  690 INPUT A$
  700 PRINT AT 1,3;"
  710 IF A$="N" THEN STOP
  720 COPY
```

Rock climber

By Robert Shipley

```
   15 LET B=CODE " "
   20 LET Z=CODE "="
   25 LET P=B
   30 LET K=CODE "■"
   35 LET X=Z
   40 LET A=B
   45 FOR N=0 TO CODE "<"
   50 LET C=CODE " "
   55 PRINT AT A,B;"■■■■  ■"
   60 LET A=A+K
   65 LET C=C+RND*PI
   70 IF C>K+K THEN GOTO 85
   75 LET B=B+K
   80 NEXT N
   85 LET B=B-K
   90 NEXT N
   95 PRINT TAB CODE "-";"■■■"
  100 PRINT AT Z,X;"■"
  105 IF Z=CODE " " THEN GOTO 155
  110 LET P=P+K
  115 PRINT AT Z,X;" "
  120 IF INKEY$="5" THEN LET X=X-K
  125 IF INKEY$="8" THEN LET X=X+K
  130 IF INKEY$="7" THEN LET Z=Z-K
  135 PRINT AT Z,X;
  140 IF PEEK (PEEK 16398+256*PEEK 16399)<>128 THEN GOTO 100
  145 PRINT "TRAPPED ON ROCKS"
  150 STOP
  155 PRINT "FOUND HIM ";P
```

Rock climber

on ZX81

A climber is stuck at the top of a cliff with his rope hanging down the side of it. You must make your way from an ambulance to the top of the cliff and take him a medical box.

The program starts by printing a random rock face which you must climb up without getting tangled in the rope or getting caught on the rocks and so becoming stuck yourself.

Use the cursor movement keys to climb the cliff. The faster you are the lower your score.

Program notes:

The graphics in line 55 are 4 graphic space, 2 spaces and a graphic space. In line 95 they are a graphic shifted 6, graphic space, graphic plus and a graphic space. In line 100 the graphic is a graphic plus.

Slide-along

By Clive Carter

```
   20 DIM A(36)
   30 FOR B=1 TO 36
   40 LET A(B)=-1
   50 NEXT B
   60 FOR B=1 TO 4
   70 FOR C=1 TO 4
   80 LET A(6*B+C)=33+4*B+C
   90 NEXT C
  100 NEXT B
  110 LET B=26
  120 LET A(B)=0
  130 PRINT "1-EASY 2-NORMAL 3-HA
RD"
  140 INPUT F
  145 FAST
  150 LET F=200+150*(F=2)+180*(F=
1)
  160 LET C=0
  170 LET D=INT (RND*4)+1
  180 LET E=(D=1)-(D=2)+6*(D=3)-6
*(D=4)
  190 IF A(B+E)<0 THEN GOTO 170
  200 LET A(B)=A(B+E)
  210 LET A(B+E)=0
  220 LET C=C+1
  230 IF C=F THEN GOTO 410
  240 LET B=B+E
  250 GOTO 170
  260 CLS
  270 PRINT "MOVE ";E
  275 PRINT
  280 LET D=0
  290 FOR B=1 TO 4
  310 FOR C=1 TO 4
  320 PRINT AT B,C;CHR$ (A(6*B+C)
)
  330 IF A(6*B+C)=33+4*B+C THEN L
ET D=D+1
  340 NEXT C
  370 NEXT B
  375 PRINT
  380 IF D=15 AND A(28)=0 THEN GO
TO 550
  390 PRINT "LETTER PLEASE?"
  400 RETURN
  410 SLOW
  415 LET E=0
  420 GOSUB 260
  430 INPUT A$
  440 IF A$="9" THEN STOP
  450 FOR B=7 TO 28
  460 IF A(B)=CODE (A$) THEN LET
C=B
  470 IF A(B)=0 THEN LET D=B
  480 NEXT B
  490 LET B=C-D
  500 IF NOT (ABS (B)=1 OR ABS (B
)=6) THEN GOTO 430
  510 LET A(D)=A(C)
  520 LET A(C)=0
  530 LET E=E+1
  540 GOTO 420
  550 PRINT "WELL DONE"
```

Open Forum

Bridges

on Vic-20

This program is a game in which there are three roads, one above another with two holes in each road. The player has to place a bridge in one of the holes in order to stop a 'cross' from falling down the hole.

I got the idea from a small pocket game.

Line 1 sets the screen colour.

Lines 20-42 check which line the cross will come to on the road.

Line 70 prints up the cross and makes a short bleep.

Lines 85 and 87 check if a bridge is under the cross.

Lines 90-140 check if a key has been pressed.

Lines 160-176 set the difficulty (speed) of the cross going across the road.

Line 1000 sets up screen variables.

Lines 1010-1080 set up the screen. the last five lines which contain data are used for the game's melody.

Slide-along

on ZX81

This program substitutes for that mind boggling game that came before the cube. The game was a flat plastic case which was square in shape, and inside of which were a number of smaller squares, usually in a 4 x 4 array.

Each of the smaller squares carried one letter or number, and had two sides tongued and two sides grooved so it could be slid along or up and down. To allow this there was always one square missing. The object of the game was to rearrange the squares in the least number of moves from a messed up position into an orderly fashion.

After you have run the program there is a slight pause before it asks you what level you want to play at.

After typing in the level the computer will go into FAST mode while it works out what the game will look like. It then displays the screen and waits for your move, you move by typing in the letter you want to move. If you want to give in, type '9'.

Program notes:

Lines 20 to 120 start the board off.

Lines 130 to 150 set up the level.

Lines 160 to 250 set up the game.

Lines 260 to 400 set up the board on the screen and any other printing to go on the screen.

Lines 410 and 415 put the computer back into SLOW mode and start up the count for how many moves you have made.

Lines 420 to 550 make up the main body of the program, checking the moves, making the moves and finishing the game off.

Bridges
By Tibor Marosi

```
 1 POKE36879,46
 2 GOSUB4000:S=0
 3 REM***BY T.MAROSI ***
 4 REM*** JAN - 1982 ***
 5 POKE36878,15:TT=99:GOSUB5000
 10 PRINT"□":GOSUB1000
 19 REM**RANDOMISE RUN**
 20 M=INT(3*RND(1)+1):V=0:Q=0
 32 IFM=1THENL=7768
 35 IFM=3THENL=8076
 42 IFM=2THENL=7922
 50 IFV>21THEN20
 70 POKEL+Q,86:POKEL+Q-1,32:POKE36874,155:POKE36874,0
 72 V=V+1
 74 PRINT"□■■■■■■SCORE = ";S
 80 Q=Q+1:QQ=QQ+1
 84 REM*CHECK FOR BLOCK*
 85 IFPEEK(L+Q+21)=32THEN2000
 87 IFPEEK(L+Q+21)=102THENS=S+1:POKE36876,232:POKE36876,0
 89 REM**GET KEY**
 90 GETA$:IFA$=""THEN160
 95 POKEN,32:POKEN+1,32
 100 IFA$="Q"THENN=7797:Z=1
 103 IFA$="Z"THENN=8105:Z=1
 105 IFA$="M"THENN=8111:Z=1
 110 IFA$="A"THENN=7951:Z=1
 120 IFA$="K"THENN=7957:Z=1
 130 IFA$="O"THENN=7803:Z=1
 140 IFZ<>1THEN160
 150 POKEN,102:POKEN+1,102
 159 REM*DIFFICULTY-TT*
 160 IFS=12THENTT=70
 162 IFS=24THENTT=55
 164 IFS=30THENTT=45
 166 IFS=40THENTT=37
 168 IFS=50THENTT=27
 170 IFS=60THENTT=17
 172 IFS=70THENTT=7
 174 IFS=80THENTT=1
 176 IFS=120THEN3000
 180 FORV=1TOTT:NEXT:GOTO60
 999 REM*SET UP SCREEN*
 1000 P=7790:P2=7944:P3=8098
 1010 FORR=0TO6:POKEP+R,160:POKEP2+R,160:POKEP+22+R,160:POKEP2+22+R,160:POKEP3+R
 160
 1015 POKEP3+22+R,160:NEXT
 1020 FORR=9TO12:POKEP+R,160:POKEP2+R,160:POKEP+22+R,160:POKEP2+R+22,160
 1025 POKEP3+R,160:POKEP3+22+R,160:NEXTR
 1030 FORR=15TO21:POKEP+R,160:POKEP2+R,160:POKEP+22+R,160:POKEP2+R+22,160
 1035 POKEP3+R+22,160:POKEP3+R,160:NEXTR
 1040 POKEP-1,86:POKEP2-1,86:POKEP3-1,86
 1050 FORE=38510TO38554:POKEE,3:NEXT
 1060 FORE=38664TO38708:POKEE,3:NEXT
 1070 FORE=38818TO38861:POKEE,3:NEXT
 1080 POKE38517,0:POKE38518,0:POKE38671,0:POKE38672,0:POKE38825,0:POKE38826,0
 1100 RETURN
 2000 POKEL+Q-1,32:POKEL+Q+21,160:FORR=15TO0STEP-.3:POKE36877,188:POKE36878,R:NE
XTR
 2100 POKE36877,0:GOTO3020
 2999 REM**USER'S WON**
 3000 FORR=1TO1000:NEXT
 3005 FORR=129TO244:POKE36875,K:POKE36876,K:NEXT:POKE36876,0:POKE36875,0:S=0
 3010 PRINT"□□□□SUPERB YOURE AN EXPERT AT THIS GAME NOW "
 3015 FORR=1TO2000:NEXTR
 3020 PRINT"□□□□□ANOTHER GAME (Y/N)":S=0
 3030 GETA$:IFA$=""THEN3030
 3035 IFA$="N"THENEND
 3036 IFA$<>"Y"THEN3020
 3040 GOTO5
 3999 REM*INSTRUCTIONS*
 4000 PRINT"□■■      BRIDGES"
 4003 PRINT"▙                "
 4005 PRINT"□THIS IS A GAME WHERE  YOU HAVE TO GET THE × ACROSS THE GAPS."
 4010 PRINT"YOU DO THIS BY FILLINGTHE GAP WITH A BLOCK  ▒▒ ."
 4020 PRINT"TO PLACE THE BLOCK IN THE RIGHT POSSITION   USE THE FOLLOWING KEYS"
 4030 PRINT"   Q - TOP LEFT"
 4040 PRINT"   O - TOP RIGHT"
 4050 PRINT"   A - MIDDLE LEFT"
 4060 PRINT"   K - MIDDLE RIGHT"
 4070 PRINT"   Z - BOTTOM LEFT"
 4080 PRINT"   M - BOTTOM RIGHT"
 4090 PRINT"■WHEN YOU ARE READY    PRESS ANY KEY"
 4100 GETA$:IFA$=""THEN4100
 4110 RETURN
 5000 POKE36878,15-
 5010 READI:IFI=-1THENI=0:I=0:RESTORE:RETURN
 5020 READD:POKE36875,I:POKE36876,I
 5030 FORN=1TOD:NEXT:POKE36875,0:POKE36876,0
 5040 GOTO5010
 5500 DATA217,400,213,400,223,400
 5510 DATA227,200,234,200,230,400
 5520 DATA227,200,234,200,230,400
 5530 DATA223,400,227,400,217,400
 5540 DATA213,500,-1
```

A fast route to impressive Beeb graphics

***Max Phillips** describes how characters can be re-defined on the BBC micro*

User defined characters provide a quick and easy route to fast, impressive graphics. Those complex animated displays, as used in *Space Invaders*, are possible without the need for assembly language sub-routines.

Different shapes may be assigned to any character — for example a lower case 'a' may be re-defined as a space invader or a ball. This facility exists on a number of machines, including the BBC micro.

To define a character for the BBC machine, draw an 8x8 grid and shade in the squares you wish to appear in the current text colour when the character is printed. Each row of eight blocks can be thought of as an eight-bit binary number, a zero defines the unshaded square and one the shaded.

Figure 1 shows an invader character converted to eight such bytes in this way. Then each binary needs to be converted to decimal for use from within your Basic program. This is done by multiplying the first bit on the right by one and adding the next bit multiplied by two, the next by four, the next by eight and so on. This should give you your character defined by eight decimal numbers in the range 0 to 255.

To use your new character, pick which character in the computer's set you wish to re-define. For this you will need to know the Ascii code for that particular character. In your program the statement VDU 23 , A , N1 , N2 , N3 , N4 , N5 , N6 , N7 , N8 will change the character whose code is 'A' to your character, where N1 to N8 are your eight numbers, starting from the top of the list. From this point on, printing that particular character, for example with CHR$(A), will produce your character on the screen.

BBC Basic reserves characters 224 to 255 for your own definitions and you will need to use the PAGE command before you can safely change the other values.

Don't try to alter characters 0 to 31 or character 127. Notice that all characters are defined on this 8x8 grid but will be printed in slightly different proportions depending on which mode you use.

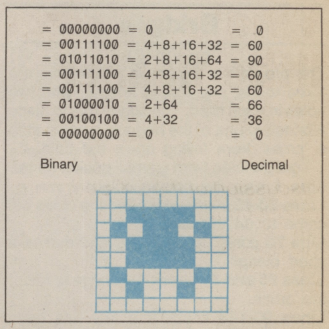


Figure 1: Converting a drawing into a character definition.

Program notes

If the above appears difficult or just time consuming, the program here allows you to define characters using the computer itself. It shows continuously the state of the grid, what the character will look like when it is printed and the eight decimal values needed to define the character.

When you run the program, the grid is shown by a red box with a white cursor inside. Next to this is the column of the numbers needed to define the character and below your character will appear as it is defined next to the words 'character is'.

Using the cursor (arrow) keys you can move the cursor to any point in the grid. Pressing D will colour that square yellow, move the cursor to the next square and automatically update the decimal.

Pressing O will turn a square that is 'on' back to 'off'. Note that you cannot turn 'on' a square that is 'on' or vice versa.

When your character looks how you want it, simply copy down the decimal. Pressing C allows you to start again and Q will quit and return you to MODE 7.

One useful trick in the program is the ★FX 4,1 call which allows the GET function to read the arrow keys. Their normal function is restored when you quit the program with ★FX 4,0.

Just to show what can be done in a few minutes, put your computer into MODE 5. Enter VDU 23,255,0,60,90,60,24,36,66,0 then PRINT STRING$(20,CHR$(255)). Think what could be done using different colours and several such characters!

Printing strings of your new characters is only one simple technique. Other suggestions are to overprint with slightly different versions of the same character, giving the impression of rotation or waving arms or to define a sequence of characters that are identical but shifted one square to the side.

Overprinting these will allow you to slide characters across character positions (you could of course put your characters anywhere by using the graphics cursor in conjunction with text cursor, try VDU 5).

Lastly, don't just stick at single characters, define bigger shapes by designing pairs of characters to be printed side by side. The list and the potential is endless.

```
  10 REM @@@@ Character Definer
  11 REM @@
  12 REM @@ produces 6 decimal bytes
  13 REM @@ needed to define a char.
  14 REM @@ with VDU 23
  15 REM @@
  16 REM @@ For BBC B, should be OK
  17 REM @@ on the A as well.
  18 REM @@
  19 REM @@@@ M.Phillips, MAY 82
  20 REM @@
  21 REM @@ Rights released for
  22 REM @@ non-commercial use only
  23 REM @@
  29 REM @@@@ Initialise
  30 DIM B(8)
  35*FX 4,1
  40 VDU 23,255,255,255,255,255,255,255,255,255
  45 C$=CHR$(255)
  50 REM @@@@ Start or restart
  60 FORI=0 TO 7:B(I)=0:NEXTI
  70 R=0:C=1:X=0:Y=0:OLDCOL=0
  00 REM @@@@ Set up screen
 110 MODE 5:CLS
 120 PRINT" Character Definer":PRINT
 130 PRINT"By M.Phillips  1982"
 140 COLOUR 1:PRINT STRING$(10,C$)
 150 PRINT STRING$(8,C$+STRING$(8," ")+C$+CHR$(13)+CHR$(10))
 160 PRINT TAB(0,13);STRING$(10,C$)
 170 COLOUR3:PRINT:PRINT"Character is "
 180 PRINT:PRINT"Commands :":PRINT:PRINT"C clear  Q quit":PRINT"D dot
     on O dot off":PRINT"or cursor keys"
 190 PRINT TAB(1,5);C$
 200 REM @@@@ Update decimal & reprint the character
 210 FOR I=0 TO 7:PRINT TAB(12,1+5);"= ";B(I);"  ";:NEXTI
 220 VDU 23,254,B(0),B(1),B(2),B(3),B(4),B(5),B(6),B(7)
 230 PRINT TAB(14,15);CHR$(254)
 300 REM @@@@ Process next command
 310 A=GET:REM fetch next command
 320 COLOUR OLDCOL:PRINT TAB(C,R+5);C$;
 330 IF A=139 THEN PROCmoveit(0,-1)
 340 IF A=138 THEN PROCmoveit(0,+1)
 350 IF A=136 THEN PROCmoveit(-1,0)
 360 IF A=137 THEN PROCmoveit(+1,0)
 370 IF A=79 AND FNcol(C,R)=2  THEN B(R)=B(R)-2^(8-C):PRINT TAB(C,R+5)
     ;" ":PROCmoveit(+1,0)
 380 IF A=68 AND FNcol(C,R)=0THEN B(R)=B(R)+2^(8-C):COLOUR 2:PRINT TAB
     (C,R+5);C$;:COLOUR3:PROCmoveit(+1,0)
 390 IF A=67 THEN 60
 400 OLDCOL=FNcol(C,R):COLOUR3:PRINT TAB(C,R+5);C$
 410 IF A<>81 THEN 210
 420 REM @@@@ End the program
 430*FX 4,0
 440 MODE7:END
1000 REM @@@@ Move cursor procedure
1010 DEF PROCmoveit(X,Y)
1020 C=C+X:R=R+Y
1030 IF C=0 THEN C=8:R=R-1
1040 IF C=9 THEN C=1:R=R+1
1050 IF R=-1 THEN R=7
1060 IF R=8 THEN R=0
1070 ENDPROC
2000 REM @@@@ return colour at R,C
2010 DEF FNcol(C,R)=POINT(C*64,1023-(R+5)*32)
```

Spectrum

In this new slot various contributors explore different aspects of the ZX Spectrum.

Who's a bright little flasher then?

***Nick Hampshire** continues his discussion of the ZX Spectrum colour commands.*

The commands used to set up the colour and display attributes were examined in last week's article. Having used these commands to create a colour display there are a series of commands which can be used to control the way particular characters are displayed without actually altering the dot pattern or colours of each character space.

The first three of these commands are BRIGHT, INVERSE and FLASH.

The BRIGHT command will display the background colour of the printed string following the BRIGHT statement with an enhanced brightness. This means that it will stand out in relation to other displayed strings which are used without the BRIGHT command.

The number following the BRIGHT command determines whether it is turned on or off. A 0 and the "bright" is off. A 1 and the "bright" is on. The following is an example of a command using BRIGHT:

```
10 PRINT INK 0; PAPER 7; BRIGHT 1; "this is in bright mode"
20 PRINT INK 0; PAPER 7; BRIGHT 0; "the bright mode is turned off"
```

The INVERSE command simply reverses the foreground and background colours for the characters in the printed string after the INVERSE command. It does this without changing the dot pattern printed on the screen.

To turn the INVERSE command on it should be followed by a 1, and to turn it off it should be followed by a 0. The following is an example of the INVERSE command:

```
10 PRINT INK 0; PAPER 7; INVERSE 1; "characters are inversed"
20 PRINT INK 0; PAPER 7; INVERSE 0; "characters returned to normal"
```

The FLASH command is used to set a following character string to flash on and off between the normal screen display and the inverted display produced by the INVERSE command. The rate of flashing is about three times per second.

This command, like the previous two commands, is very useful in drawing attention to a displayed statement or command. The following is an example of the FLASH command:

```
10 INPUT FLASH 1; INK 1; PAPER 7; "input data"; N
```

The Spectrum has a very useful overprinting command called OVER which allows the creation of new characters by overprinting one or more characters over an existing character. The most obvious use of this command is to add an accent to a character.

Normally when a character is displayed, and another character previously occupies that space then whatever is already written in that character space is obliterated. In the OVER command the existing character is retained and the dots of the new character added.

As with the previous commands following it with a 1 will turn it on and a 0 will turn if off. The following is an example of the OVER command:

```
10 OVER 1
20 PRINT "a"; CHR$8; " "
```

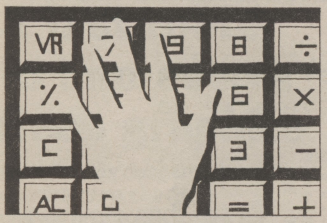
The CHR$8 causes the cursor to back up one character space. All the commands which control the attribute of a character can also be set using the character codes which represent the command; the following commands and codes are identical:

CHR$ 16 — INK command
CHR$ 17 — PAPER COMMAND
CHR$ 18 — FLASH command
CHR$ 19 — BRIGHT command
CHR$ 20 — INVERSE command
CHR$ 21 — OVER command.

```
  10 REM rainbow
 100 LET xo=100: LET yo=20
 105 LET dp=5: LET p1=1: LET p2=
160
 110 FOR r=40 TO 70 STEP 10
 120 READ c
 122 PAPER c
 125 FOR q=1 TO 10
 130 LET dp=dp*3.14159/180
 140 LET p1=p1*3.14159/180
 150 LET p2=p2*3.14159/180
 160 FOR p=p1 TO p2 STEP dp
 170 LET x=r*COS (p)
 180 LET y=r*SIN (p)
 190 LET x=xo+x
 200 LET y=yo+y
 310 PLOT x,y
 500 NEXT p
 505 NEXT q
 510 NEXT r
1000 DATA 2,6,4,1,7
```

```
   1 REM Kinetic art program
   2 REM fills the screen with a
 randomly
   3 REM moving pattern of diffe
rent colour squares
   4 REM just run and be hypnoti
sed
   5 REM by the changing pattern
s.
   6 REM note that spaces are be
ing displayed
   7 REM so the paper colour det
ermines display colour
   8 RANDOMIZE
   9 LET a=10: LET b=20
  10 LET c=INT (RND*9)
  20 LET n=INT (RND*20)
  30 FOR x=0 TO n
  40 LET d=INT (RND*4)
  50 IF d=0 THEN LET a=a+1
  60 IF d=1 THEN LET a=a-1
  70 IF d=2 THEN LET b=b+1
  80 IF d=3 THEN LET b=b-1
  90 IF b>=30 THEN LET b=30
 100 IF b<=1 THEN LET b=1
 110 IF a>=20 THEN LET a=20
 120 IF a<=1 THEN LET a=1
 130 PAPER c
 140 PRINT AT a,b;" "
 150 NEXT x
 160 GO TO 10
```

Hand & mouth



Who's playing the guessing game?

One of the cornerstones of the mathematical armoury necessary to deal with the everyday problems of a physicist, engineer or 'A' Level student is the ability to solve the roots of an equation. This has scope all the way from bridge building to understanding the origin of the universe. Don't let that put you off though, as I can do neither anyway.

Now, by root I mean any solution, X, to the equation f(X)=0. (That's why, incidentally, we call $\sqrt{A}$ a square root, as that is the solution to the squared equation $f(X)=X^2=A=0$).

Nature rarely conspires to give us an exact functional form for the solution to such equations (such as the solution to $aX^2+bX+c=0$; the well known $-b\pm\sqrt{b^2}-4ac$).

If you're out of luck then one of the techniques left at your disposal is the numerical solution. This crudely means making successive guesses towards an answer which, if they are intelligently made, converge towards a solution.

Sounds easy? One root-finding numerical solution which is ideally suited to the modern programmable calculator was discovered over three hundred years ago by Isaac Newton — of falling apple fame. He showed that a good approximation to a zero in f(X) and the difference term $\triangle X$ is given by $\triangle X= f(X)/f'(X)$ where f'(X) is the differential of f(X).

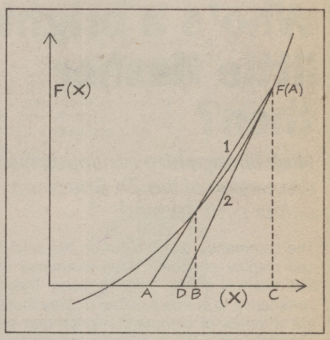
Now what this means to a good programmer is that if we generate a cyclic program to evaluate our new guess as $X-[f(X)/f'(X)]$ where X is the old guess, and repeat the procedure, then the sign of a root is $\triangle X$=(new guess-old guess) tending to zero.

If you are worried about f'(X) remember that it can be approximated by $f(X)-f(X-d)/d$ where d is small, in many cases.

All too easy? Well, I'm afraid that I'd better spill some of the beans. First there is the problem that the calculator only carries internally some 10 to 13 digits (depending on the model) and hence we will introduce what are known as rounding errors owing to our non-exact evaluation of numbers.

For instance no calculator can solve $f(X)=X+(2(X-5))=0$ exactly by the above procedure even though the equation simplifies to $(f(X)=3X-10$!!

Another problem is that with only a finite amount of time available we can only give a finite number of guesses — and we can prove that in full generality we need to put in an infinite number of guesses.



Still, a good iteration program can usually overcome most of the problems — although it may be very complex to do so.

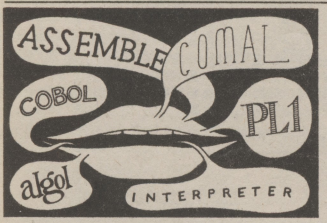
Anyway, there's an offer of £5 for the best program to generate the two roots of the equation $f(X)=EXP(X)-5X+3$ starting with initial guess 2.50; by some iteration other than the aforementioned Newton's Method. Perhaps you might take a hint from the above diagram.

Method One (Secant Iteration) — uses two guesses B and C to generate a third A and so on.

Method Two Newton's Method was, in fact, just the tangent to f(X) at old guess X=A to generate new guess D.

Generally, method one is faster than method two.

John Gowrie



Get yourself ORGanised!

When you have written the source code and stored it carefully, that means twice on different cassettes or discs, you come to the stage of turning all the lines of text, the mnemonics, into machine code that will run on your computer.

Generally you will spend far less time assembling a program than you will in preparation but there are, nevertheless, some important features to watch out for in this part of the assembler's functions.

Will the assembler allow you to use another pseudo-op code called ORG (ORiGin) in more than one place? ORG is used to set address at which the assembler will generate machine code. It is often useful to be able to alter the origin in the middle of the source code.

It is useful to be able to suppress the storage of the object code while the source is being assembled. This function allows you to check for assembly errors without actually producing any machine code. The second benefit is that you can check the size of the final object code program to make sure it will not overwrite your source code — a major disaster!

Does the assembler have good error messages when you are generating object code from source code? ZEN, for the Sharp MZ80K, has nine messages that may be triggered by faults.

How does the assembler link with the printer in your system? The Microtanic Software assembler will only print the source code when it is assembling. The ZEN assembler for the MZ80K computer had no print routine included when it was marketed first, only a space in the program where you could write your own routine.

When you have a working program will the assembler help you to generate machine code in a form suitable to be "blown" into an EPROM? You will need to be able to assemble the code at one address while the jump instructions are set for the eventual ROM address.

My EPROM programmer takes machine code from 0400 Hex. to 13FF Hex. If I want to produce an EPROM that will be placed at C000 Hex. the assembler must use an ORG instruction to start the machine code jump instructions at the right place while the code is actually stored at 0400 Hex.

I have not mentioned other features that you may encounter such as the ability to print the symbol table addresses or to sort them into alphabetical order. Nor have I gone into assembly from tape or disc rather than from the computer RAM.

The speed of assembly varies widely from one assembler to another and if you intend to write a long program, that may be important.

Next week I will write about macro assemblers and the difference one can make to the efficiency of the final object code.

John Dawson

Sound & vision



I look at all the lonely programs

The listing below is a simple program for playing Eleanor Rigby on the Vic-20. I have written it almost entirely with Pokes. I found it easier using Pokes, instead of Data statements, because there are so many repeated themes in the tune.

I listed the program on my school Pet printer. To do this with Vic programs you need to:

1 SYS 1024
2 M 0400 0407
3 0400 00 01 10 00 00 8F 00 00
4 Press Return
5 Press X and Return

For Pets that cannot use machine code type:

POKE 4096,1:POKE 41,16:CLR

Neil Jayakumar

```
5 X=1
10 POKE36878,20
20 S=36876
30 POKES,183:FORT=1TO200:NEXTT:POKES,0
40 POKES,191:FORT=1TO200:NEXTT:POKES,0
45 POKES,199:FORT=1TO200:NEXTT:POKES,0
50 POKES,183:FORT=1TO400:NEXTT:POKES,0
60 POKES,167:FORT=1TO600:NEXTT:POKES,0
70 POKES,183:FORT=1TO200:NEXTT:POKES,0
80 POKES,191:FORT=1TO200:NEXTT:POKES,0
90 POKES,199:FORT=1TO200:NEXTT:POKES,0
100 POKES,207:FORT=1TO400:NEXTT:POKES,0
110 POKES,203:FORT=1TO200:NEXTT:POKES,0
115 POKES,199:FORT=1TO200:NEXTT:POKES,0
120 POKES,203:FORT=1TO400:NEXTT:POKES,0
130 POKES,199:FORT=1TO200:NEXTT:POKES,0
140 POKES,191:FORT=1TO200:NEXTT:POKES,0
150 POKES,199:FORT=1TO400:NEXTT:POKES,0
160 POKES,191:FORT=1TO200:NEXTT:POKES,0
170 POKES,183:FORT=1TO200:NEXTT:POKES,0
180 POKES,191:FORT=1TO1400:NEXTT:POKES,0
181 X=X+1
190 POKES,183:FORT=1TO200:NEXTT:POKES,0
200 POKES,191:FORT=1TO200:NEXTT:POKES,0
210 POKES,199:FORT=1TO200:NEXTT:POKES,0
220 POKES,201:FORT=1TO600:NEXTT:POKES,0
230 POKES,199:FORT=1TO400:NEXTT
235 IFX<>3THEN10
240 POKES,199:FORT=1TO1600:NEXTT:POKES,0
250 POKES,212:FORT=1TO400:NEXTT:POKES,0
260 POKES,212:FORT=1TO200:NEXTT:POKES,0
270 POKES,212:FORT=1TO200:NEXTT:POKES,0
280 POKES,217:FORT=1TO200:NEXTT:POKES,0
290 POKES,219:FORT=1TO400:NEXTT:POKES,0
300 POKES,223:FORT=1TO400:NEXTT:POKES,0
310 POKES,219:FORT=1TO400:NEXTT:POKES,0
320 POKES,217:FORT=1TO400:NEXTT:POKES,0
330 POKES,212:FORT=1TO400:NEXTT:POKES,0
340 POKES,199:FORT=1TO400:NEXTT:POKES,0
350 POKES,183:FORT=1TO2000:NEXTT:POKES,0
360 X=X+1
370 IFX<>5THEN250
380 POKES,191:FORT=1TO1600:NEXTT:POKES,0
390 FORZ=1TO8
395 POKES,199:FORT=1TO400:NEXTT:POKES,0
400 NEXTZ
410 FORZ=1TO8
420 POKES,201:FORT=1TO400:NEXTT:POKES,0
430 NEXTZ
440 POKES,175:FORT=1TO200:NEXTT:POKES,0
450 POKES,187:FORT=1TO200:NEXTT:POKES,0
460 POKES,195:FORT=1TO600:NEXTT:POKES,0
470 POKES,187:FORT=1TO400:NEXTT:POKES,0
480 POKES,195:FORT=1TO400:NEXTT:POKES,0
490 POKES,201:FORT=1TO200:NEXTT:POKES,0
500 POKES,187:FORT=1TO400:NEXTT:POKES,0
510 POKES,175:FORT=1TO400:NEXTT:POKES,0:
    FORT=1TO600:NEXTT
520 POKES,201:FORT=1TO200:NEXTT:POKES,0
530 POKES,215:FORT=1TO400:NEXTT:POKES,0
540 POKES,201:FORT=1TO400:NEXTT:POKES,0
550 POKES,195:FORT=1TO50:NEXTT:POKES,0:
    FORT=1TO150:NEXTT
560 POKES,187:FORT=1TO400:NEXTT:POKES,0
570 POKES,175:FORT=1TO1400:NEXTT:POKES,0
    FORT=1TO400:NEXTT
575 X=10
580 POKES,215:FORT=1TO1200:NEXTT:POKES,0
590 POKES,219:FORT=1TO200:NEXTT:POKES,0
600 POKES,221:FORT=1TO200:NEXTT:POKES,0
610 POKES,225:FORT=1TO400:NEXTT:POKES,0
620 POKES,221:FORT=1TO400:NEXTT:POKES,0
630 POKES,219:FORT=1TO400:NEXTT:POKES,0
640 POKES,215:FORT=1TO400:NEXTT:POKES,0
650 POKES,201:FORT=1TO400:NEXTT:POKES,0
660 POKES,187:FORT=1TO1200:NEXTT:POKES,0
670 X=X+1
680 IFX<>12THEN580
690 FORT=1TO200:NEXTT
700 POKES,175:FORT=1TO200:NEXTT:POKES,0
710 POKES,183:FORT=1TO200:NEXTT:POKES,0
720 POKES,187:FORT=1TO400:NEXTT:POKES,0
730 POKES,195:FORT=1TO200:NEXTT:POKES,0
740 POKES,201:FORT=1TO400:NEXTT:POKES,0
750 POKES,201:FORT=1TO200:NEXTT:POKES,0
760 POKES,207:FORT=1TO200:NEXTT:POKES,0
770 POKES,212:FORT=1TO200:NEXTT:POKES,0
780 POKES,201:FORT=1TO400:NEXTT:POKES,0
790 POKES,191:FORT=1TO600:NEXTT:POKES,0
800 POKES,201:FORT=1TO200:NEXTT:POKES,0
810 POKES,207:FORT=1TO200:NEXTT:POKES,0
815 POKES,212:FORT=1TO200:NEXTT:POKES,0
820 POKES,219:FORT=1TO400:NEXTT:POKES,0
830 POKES,217:FORT=1TO200:NEXTT:POKES,0
840 POKES,217:FORT=1TO400:NEXTT:POKES,0
850 POKES,212:FORT=1TO200:NEXTT:POKES,0
860 POKES,207:FORT=1TO200:NEXTT:POKES,0
870 POKES,212:FORT=1TO400:NEXTT:POKES,0
880 POKES,207:FORT=1TO200:NEXTT:POKES,0
890 POKES,201:FORT=1TO200:NEXTT:POKES,0
900 POKES,207:FORT=1TO2100:NEXTT:POKES,0
910 RUN
```



Keyboard Audio Tone

Audio Keystroke and Program Monitor

* TWO TONE. One to indicate initial press of key, the second tells you when computer is ready to accept next entry (very useful when entering long program listing)
* Indicates start and finish of running programs, and start and end of SAVE and LOAD routines
* Indicates Keyboard entry during INPUT prompts
* Use to indicate progress of running programs, or for sound effects in games programs, by including short PAUSE in listing
* Operates in FAST and SLOW modes, with normal or shifted keys
* Only five simple connections to ZX81 P.C.B.

Supplied ASSEMBLED complete with fitting instructions or factory fitting service available by sending us your ZX81 only (NOT power supply)

£8.95 inc vat ASSEMBLED

£10.95 inc vat + 90p P & P FACTORY FITTED

Cheques and P.Os payable to:
TV SERVICES OF CAMBRIDGE LTD
CHESTERTON MILL
FRENCH'S ROAD
CABRIDGE CB4 3NP
Tel: (0223) 358366

POPULAR Computing WEEKLY

HOBHOUSE COURT, 19 WHITCOMB STREET WC2

WHY NOT TAKE OUT A SUBSCRIPTION?

You can have *Popular Computing Weekly* sent to your home for £19.95 a year. This will cover the cost of the magazine, postage and packing.

Fill in the coupon below and keep yourself up to date week by week.

I would like to take out ..... subscription(s) to *Popular Computing Weekly.*

Name.....

Address.....

.....

.....

Tel No.....

I enclose £..... for ..... subscription(s).

Please make cheques payable to Sunshine Publications Ltd.

Peek & poke

Peek your problems to our address. Ian Beardsmore will poke back an answer.

SINCLAIR: PLEASE LET THE GOOD TIMES ROLL

Phillip Lee of Charmaine Court, Harlington, Middlesex writes:

Q After a two-month delay, Sinclair Research has finally delivered five rolls of printer paper ordered in February. I have been wondering if there is any way to speed up delivery, or is there an alternative supplier? If not I might as well order my next batch now.

A Unfortunately there is not much that you can do other than order your next rolls now. An alternative supplier did come forward last September, but the rolls supplied simply did not print. Since then, as far as I know, no one else has ventured into this section of the ZX market.

KEEP A WELCOME IN THE RAM PACK

Richard Haunton of Hitchens, Petworth, Sussex, writes:

Q I am in a slight fix, I am going away to live in Wales, and I have a Welsh pen friend. I wondered if you could write a program for the ZX81, with 16K RAM Pack, to convert English to Welsh, and Welsh to English.

I would like it to work by feeding in sentences and translating them. I hope that you can help. I know that you can buy computers that can translate eight languages. I only want one.

A In programs such as this the available memory is the point which must be borne in mind. The program here is adapted from an English/Dutch glossary I wrote some time ago. Phrases and sentences consume a lot more memory, but I cannot say how much as I do not know what you actually want to put in.

It is because of space that I have used lines 30 and 40 in the way that I have. If you find that you can lower the value of P in line 40 all to the good.

Remember that spaces must be included in the same place each time. In the lines from 3010 on I have separated the two complete phrases by a comma.

You will probably find that different line lengths need different spacings. This is something you can only do when the lines are on the screen in front of you. Because of the line numbering, the English phrase at 1010 is the translation of the Welsh phrase at 2010 and they are printed at 3010.

This system follows all the way through, and should make it easier to enter the program.

```
  10 REM TRANSLATION
     PROGRAM
  20 REM I. BEARDSMORE
  30 LET Q=1
  40 LET P=9
  50 PRINT "ENGLISH (E) WELSH
     (W) STOP (S)"
  60 INPUT a$
  70 IF A$= "E" THEN GOTO 100
  80 IF A$= "W" THEN GOTO 200
  90 IF A$= "S" THEN STOP
 100 PRINT "INPUT ENGLISH
     PHRASE"
 110 INPUT E$
 120 GOTO 1000
 200 PRINT "INPUT WELSH
     PHRASE"
 210 INPUT W$
 220 GOTO 2000
1000 FAST
1010 IF E$ (Q TO P)= "first nine
     letters and spaces of first English
     phrase" THEN GOTO 3010
1020 IF E$ (Q TO P)= "first nine
     letters and spaces of second
     English phrase" THEN GOTO
     3020
     Input all the English phrases
        in the same way
2000 FAST
2010 IF W$= "first nine letters and
     spaces of first Welsh Phrase"
     THEN GOTO 3010
3020 IF W$= "first nine letters and
     spaces of second Welsh phrase"
     THEN GOTO 3020
     Input all the Welsh phrases
        in the same way
3010 PRINT "first English
     phrase","first Welsh phrase"
3015 GOTO 50
3020 PRINT "second English
     phrase","second Welsh phrase"
3025 GOTO 50
```

IS THERE A HEX ON THIS PROGRAM?

S. A. Chowdhery of Tooting Bec Road, London, writes:

Q I am the owner of a ZX81 and I have just reveived a program for *Space Invaders*. The bulk of this program is written in hexadecimal. I have tried many times to enter it, but after a few bytes the computer gives back the report 4/150. Please could you tell me where I am going wrong?

A I cannot give you any advice about the program as such, because you do not say which *Space Invaders* it is. Anyway in a case like this it would be best to get in touch with whoever you got the program from.

They will have the correct listing to hand, whereas anyone else would probably have to spend many hours working it out.

It does strike me that you have made, what is in fact a common mistake. How many K does the program require? The error code, 4/150 means that you have run out of memory at line 150.

It is probable that if you have a ZX81, without a RAM Pack and the 1K onboard RAM is not enough. If this is the case then one of the several 16K RAM packs that are on the market should take care of this.

HIS PROBLEM IS INDEX-LINKED

R. R. Williams of Menai Bridge, Gwynedd, writes:

Q I would be grateful for your help and comments with regard to the possibility of using the ZX81 for the purpose of indexing.

The project I have in mind is to index from a text as follows:

a indexing personal names with a reference number in alphabetical order;
b indexing place names with a reference number in alphabetical order;
c indexing status and profession with a reference number in alphabetical order;
d indexing of ships names with a reference number;
e indexing of special personal names in alphabetical order;
f indexing of miscellaneous items in alphabetical order.

My purpose is to call up the information, say **a**, and print it. I do not want to computer store the information since the printed material is my final information. I would prefer to be able to print on A4 paper in two columns.

We have an electronic Silver Reed EX55 typewriter with a daisy-wheel capable of 20 cps. Can you also say if a suitable interface can be purchased for this machine to act as a printer for the ZX81?

A You are looking for an indexing program for the ZX81 and surprisingly there seem to be very few published.

I feel your best bet would be to look at *Byting Deeper Into Your ZX81* by Mark Harrison. On page 47 there's an indexing program along with a good, concise description of how it works. The book is published by Sigma Press and is available from Dillons University Bookshop in Mallet Street, London WC1. Price £4.95 plus p&p.

However on reading the second part of your letter, I think you should take a close look at the ZX81, and what it can do. It is a computer, not a word processor. While many micros will have the facilities that you need (including the new ZX Spectrum), I do not think that the ZX81 is among them.

Most obvious drawback is the print-out. Amidst the plethora of add-ons for the ZX81, I have never seen a conventional paper printer. The ZX81 doesn't even have lower case letters (though d'k tronics do have an add-on 4K ROM chip that includes lower case letters) which might overcome this problem.

The problem for you is that the ZX81 *can* do the task you want it to do, but it cannot process the information in the way you want it — on A4 paper.

It *is* a task that seems ideally suited to a micro, and if you want to use one, then perhaps you should look at other systems.

● *Stop agonising over that problem. Write to Ian Beardsmore, Peek & Poke,* Popular Computing Weekly, *Hobhouse Court, 19 Whitcomb Street, London WC2 7HF.*

Competitions

Puzzle No 8

You will need a chess-board, two dice and a supply of counters.

Throw the two dice and note the numbers uppermost. This represents the throw for the game and the dice are not rethrown.

Starting at the bottom left of the chess-board (counting the corner square as number 1) count off the number shown by the first die, moving across the bottom row of the board, from left to right.

When the first number thrown is reached, place a counter on this square, and continue counting, only this time the number on the other die. Once again place a counter to mark the square on which you land.

If you reach the end of the row, then move up on to the second line and continue, this time moving from right to left — rather in the manner of moves in a game of *Snakes and Ladders*.

Then continue up the board by counting again the number of the first die, then the second — always alternating the two numbers. Each time you land put down a counter so you will leave a trail of counters showing the squares on which you landed.

Now, when you arrive at the top left-hand square, without interrupting the sequence of counting, turn and continue down the left-hand file of the board and proceed by zig-zagging up and down the board until you finally reach (or pass over) the top right-hand square.

On this second (up and down) stage, although you continue to place counters on vacant squares, if you land on a square that already has a counter on it, you must take that counter off the board (ie this square becomes vacant). The total number of counters remaining on the board represents the 'score' for that particular throw of the dice.

(a) Which dice give the lowest score?
(b) Which dice give the highest score?
(c) Which two throws produce an identical final arrangement of counters that form an almost perfect "enlarged" chequered pattern?

Solution to Puzzle No 4:

```
  755
 x33
2325
2325
25575
```

Winner of Puzzle No 4
The winner is: R. G. Girvan, Gillott Road, Edgbaston, Birmingham, who receives £10.

Solution to Crossword No 4
Across: 4 Gosub 7 Display 8 Access 10 Cream 12 Edam 13 Experts 14 Thor 15 Sigma 16 Apples 20 Unbound 21 Store.
Down: 1 Microchip 2 Speakers 3 Wax 5 Orc 6 Use 8 Acorn 9 Statement 11 Spies 12 Eskimos 17 Pet 18 Ear 19 One.

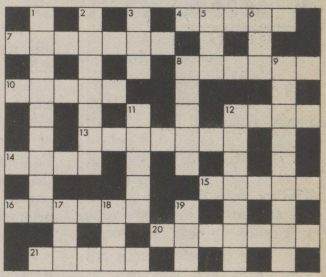
Winner of Crossword No 4
The winner is: A. C. Lewis, St John's Place, Rhoose, Nr Barry, S Glamorgan, who gets £10.

Rules
The winner for the crossword and the winner of the puzzle will be the first name out of the hat (in each case).

Closing date for both the crossword and the puzzle is Monday, 28 June.

Please mark your envelope 'CROSSWORD' or 'PUZZLE'.

Crossword No 8

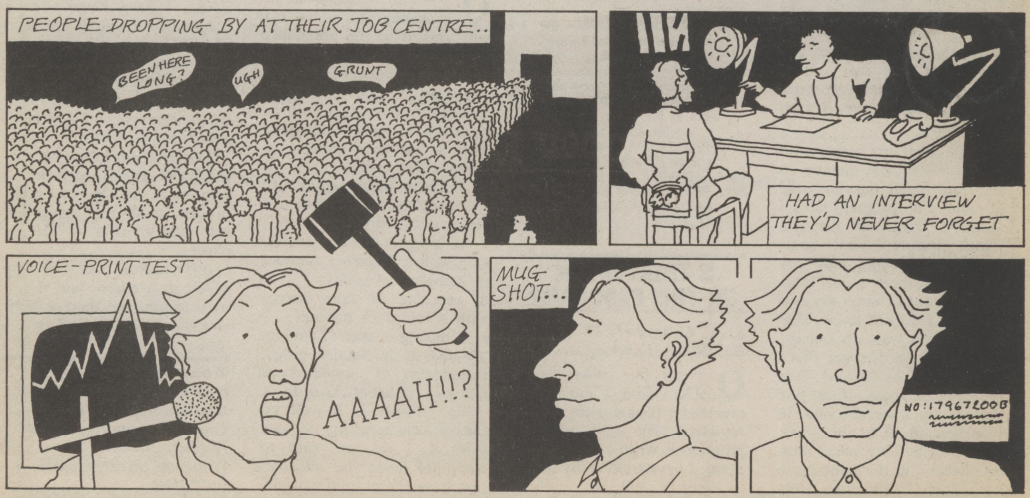


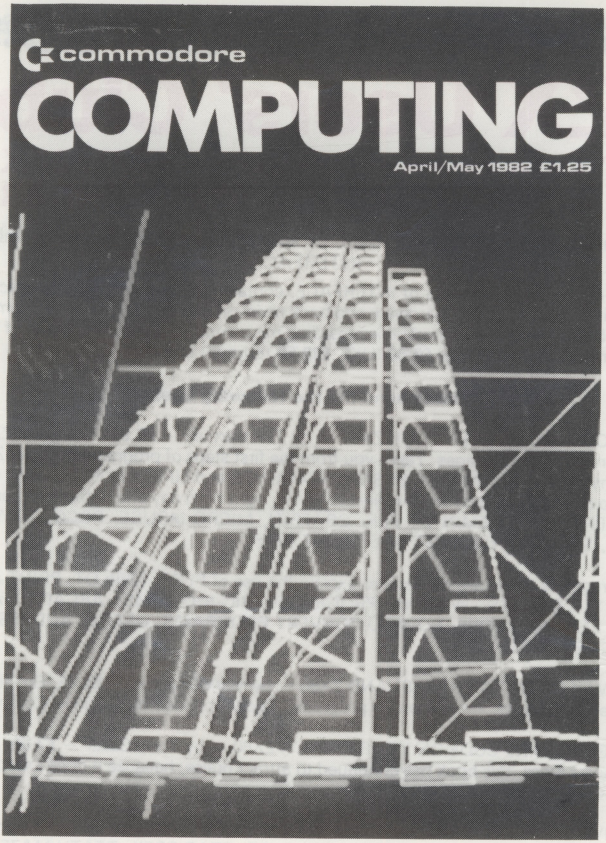
ACROSS
4 Current or not, start with the micromaker (5)
7 Parent with cage containing a king is the subject of a complete computer system (7)
8 Set on a nail, I hear (6)
10 Ape after a thousand million, wide mouthed (5)
12 Total money for an oil pan (4)
13 Level, not odd, not in (4, 3)
14 Turn and strike in the stomach (4)
15 Animal with upper-class people (5)
16 Prises a student, always (6)
20 Digits — the Frenchman's end, the German's half, and the Spanish start (7)
21 Width of a display device (5)

DOWN
1 Citrus micro maker (9)
2 Jumped and missed (7)
3 For example, love of a self-image (3)
5 Stop the upcoming union meeting (3)
6 Drowned valley in safari area (3)
8 A right note on maker of 17D (5)
9 17D's former officer (9)
11 In state of great exasperation (5)
12 Bunches of characters in basic threads (7)
17 Micro No 5997 (3)
18 Sentiment on passing tear (3)
19 Basic statement about a basic function (3)

CITIZEN PAIN

BY DAVID IRELAND and JAMES MACDONALD





An exciting new Commodore peripheral

Own or use a Pet or a Vic?

Fed up with being ignored by all the traditional monthly magazines?

Fed up with listings, which are too simple or simply do not work?

You need *Commodore Computing*, the new monthly magazine. *Commodore Computing* is published by Nick Hampshire, author of *The Pet Revealed, Pet Graphics, A Library Subroutines* and *The Vic Revealed.*

Each issue is packed with advanced advice on how to make the most of your computer, whether you use a Pet of a Vic.

Each issue covers a host of applications – software, hardware, machine code, games, business use – whatever it is you'll find it in *Commodore Computing.*

If you want to learn more about your computer, take out a subscription to *Commodore Computing.*

That's the only way to get it, and get it straight.

Send £12.50 for 1 year's subscription (10 issues) to:

Commodore Computing,
Magsub, Oakfield House, Perrymount Road, Haywards Heath, Sussex RH16 3DH

ZX Users' Club

JOIN YOUR USERS' GROUP — AND MAKE THE MOST OF YOUR MICROCOMPUTER

Join the National ZX80 and ZX81 Users' Club, by subscribing to the official monthly club magazine INTERFACE.

( ) Please send me the next 12 issues of INTERFACE, containing many programs for each machine in each issue, plus hints, tips, software, hardware and book reviews, plus special offers for members. I understand you will be able to help me with problems regarding my computer, and let me know of any local branches of the club in my area. I enclose **£9.50** (UK), **£12.50** (Europe) or **£16.00** (elsewhere).

Please send me the following books:

( ) GETTING ACQUAINTED WITH YOUR ZX81 — by Tim Hartnell — **£5.95**. This great ZX81 book contains over 80 programs in its 128 pages. Takes you from the first steps of programming your ZX81 to quite complex programs such as WORD PROCESSOR, DRAUGHTS and LIFE. You'll find a host of programs to get your ZX81 up and running with worthwhile programs, right from day one. Other programs include SPACE BOY, ROLLER-BALL, CHEMIN DE FER, GRAFFITI, MICRO-MOUSE, POGO, TOWERS OF HANOI, BLOCKOUT, SALVADOR, BANDIT and DODGE CITY.

As well as programs, there are sections to explain the use of PLOT, UNPLOT, PRINT AT, MAKING THE MOST OF 1K, ARRAYS, WRITING PROGRAMS, BIO-RHYTHMS, ARCADE GAMES, RANDOM NUMBERS, PEEK AND POKE, HOW TO CONVERT PROGRAMS, USEFUL ADDRESSES, SPECIFICATIONS, THE NEW ROM.

( ) THE GATEWAY GUIDE TO THE ZX81 AND ZX80 — by Mark Charlton — **£6.45**. Explains ZX BASIC from first principles. 180 pages, more than 70 programs. Recommended by Creative Computing.

( ) MASTERING MACHINE CODE ON YOUR ZX81 OR ZX80 — by Tony Baker — **£7.50**. Warmly welcomed by the computer press, this book has continued to attract praise, because it does exactly what it claims to do in the title.

( ) 49 EXPLOSIVE GAMES FOR THE ZX81 (and 29 for the ZX80) — edited by Tim Hartnell — **£5.95**. Every game you need: DRAUGHTS, GALACTIC INTRUDERS, STAR TREK, DEATH MAZE, 4-IN-A-ROW and an 8K ADVENTURE-type program SMUGGLERS BOLD.

( ) 34 AMAZING GAMES FOR THE 1K ZX81 — by Alastair Gourlay — **£4.95**. All programs dumped from the printer and guaranteed to run. This book is the key to making the most of 1K.

( ) GETTING ACQUAINTED WITH YOUR VIC20 — by Tim Hartnell — **£6.95**. This book is the ideal one for first-time users of the VIC 20, with over 60 programs.

( ) SYMPHONY FOR A MELANCHOLY COMPUTER and other programs for the VIC20 — **£6.95**. A great collection of 24 great games — all dumped direct from the printer — for the VIC20.

( ) GETTING ACQUAINTED WITH YOUR ACORN ATOM — by Trevor Sharples and Tim Hartnell — **£7.95**.

( ) 39 TESTED PROGRAMS FOR THE ACORN ATOM (the best of INTERFACE) — **£6.45**.

( ) PASCAL FOR HUMAN BEINGS — Jeremy Ruston — **£4.95**.

Atom

VIC

INTERFACE,
44-46, Earls Court Road, Department PC, London W8 6EJ.

Please send me the indicated items. I enclose £----------.

Name ----------

Address ----------
