AN ARGUS SPECIALIST PUBLICATION

JULY/AUGUST 1983

NEW

A&B COMPUTING

£1.85

FOR THE USER OF THE BBC MICRO

Over 120 pages of features, programs, news and reviews to help you get the most from your BBC Micro

Hardware Review

New launch we review the Torch disc system

Software Reviews

Games galore, educational aids and useful utilities

Programs

Arcade action and classroom clues on your micro

PLUS: Q&A – Helpful hints and software solutions
Speakers – Sounding out your BBC Micro

Go for £1000 cash software challenge

SIR COMPUTERS LTD
CARDIFF

Agents for Acorn, BBC and TORCH Computers

BBC Microcomputers

Model B **£399.00**
Model B with disc interface **£509.00**
Single 100K disc drive **£249.00**
Dual 2 × 100K disc drive **£389.00**
The disc manual and utilities disc are both included.
Disc interface for the BBC Micro
Kit **£95.00**
Fitted **£110.00**
Upgrade of BBC Model A to B **£90.00**

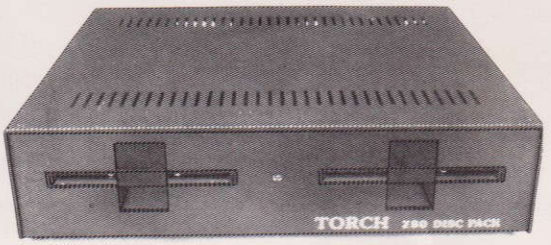
Please telephone for up to date information on Prestel, Teletext, speech synthesis, second processors, etc.

TORCH Computers

Z-80 Disc Pack for the BBC Microcomputer **£897.00**

This unit connects to the BBC Micro in the same way as a normal disc drive, but as well as offering a dual 2 × 400K disc drive for use under BBC BASIC or other languages it provides the option of using the wide range of CP/M software available for business and data processing applications. The firmware supplied with the machine allows switching between BASIC and CPN, a powerful operating system developed from CP/M 2.2.

In addition to the disc pack a second processor is supplied. This is a Z-80A with its own 64K RAM card, communicating with the 6502A in the BBC computer through the 'Tube'. Typically the speed of execution of programs under the twin-processor system is increased by up to 50% compared with a conventional single-processor computer. A third processor, the 16 bit 68000, will shortly be available.



TORCH CF240 ..... **££3200.00**
WORDSTAR WORD PROCESSOR SUPPLIED FREE OF CHARGE

This is an extension of the BBC microcomputer/Torch disc pack system, available in a single unit. The computer contains a BBC-based peripheral processor connected to the main Z-80 computer, a dual 2 × 400K disc drive as described above, a high resolution (80 character) colour monitor and a complete British Telecom approved 1200 baud modem. It is the only microcomputer which has been granted permission for direct connection to the Public Switched Telephone Network both in the U.K. and the United States.

The TORCH can communicate either directly with another TORCH or with virtually any other type of computer via Prestel. Using the Gateway facility of Prestel it is possible for the TORCH to access vast amounts of information stored by private organisations on public database systems. The Mailbox facility of Prestel also allows the use of electronic mail.

TORCH CH240/10 As above but with a 10 MB hard disc drive.
TORCH CH240/21 As above but with a 21 MB hard disc drive

PERIPHERALS

Seikosha GP 100A Printer **£229.00**
Epson SX 80 ..... **£470.00**
CP80 Printer **£389.00**
Kaga 12" RGB Monitor **£280.00**
Sanyo 14" RGB Monitor **£270.00**
High resolution 12" black/green monitor **£85.00**

SOFTWARE

We currently hold in stock programs from the following suppliers. Acornsoft, A & F Software, Bug Byte, Computer Concepts, Data Base Software, Digital Fantasia, Gemini, Golem, IJK Software, Level 9 Software, Molimerx, MP Software, Program Power, Salamander Software, Software for All, Superior Software.

Unfortunately, we are unable to supply software by mail except as part of a larger order. Delivery by Interlink of any of the above items **£10.00**.
Unless otherwise stated all prices include VAT.

SIR COMPUTERS LTD
91 WHITCHURCH ROAD, CARDIFF
TELEPHONE (0222) 21341

Midwich
COMPUTER COMPANY LIMITED

1st choice for BBC microcomputers

BBC COMPUTERS

Model B	£346.95
Model B + Disc Interface	£441.95

Please phone to check delivery

BBC MICRO DISC DRIVES

BBC 31	Single 100K Drive Expandable to 2 x 100K	£199.00
BBC 32	Dual 100K Drives	£330.00
BBC 33	100K Upgrade for BBC 31	£140.00
BBC 34	Dual 400K Drives	£575.00
BBC 35	Utilities Disc (supplied only with BBC 31,32, 34)	£25.00

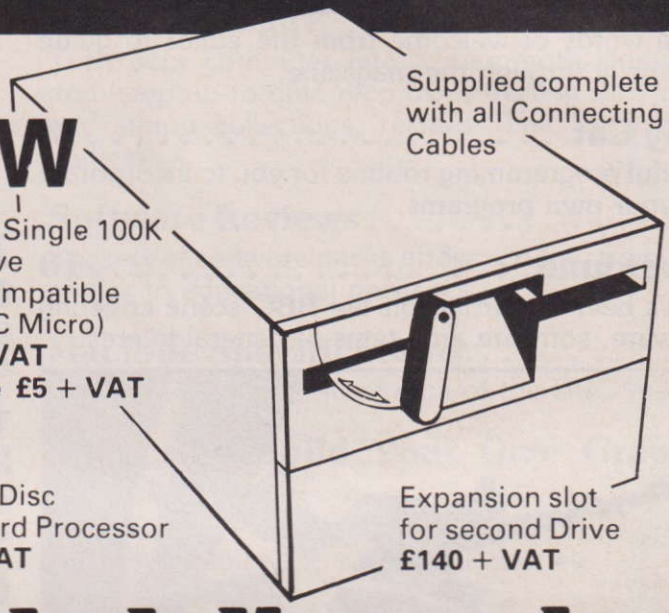
All Drives except BBC 33 supplied cased with Connecting Leads.

PRINTERS

NEC 8023 Printer £320.00
(Carriage £10)

NEW

Shugart Single 100K Disc Drive
(Fully compatible with BBC Micro)
£199 + VAT
Carriage £5 + VAT



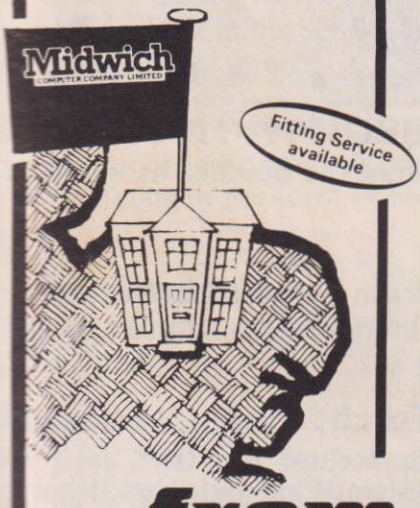
Supplied complete with all Connecting Cables

Utilities Disc with Word Processor
£25 + VAT

Expansion slot for second Drive
£140 + VAT

Fast ex-stock delivery by Securicor...

All prices exclude VAT.



BBC UPGRADE KITS

BBCA2B	Complete A to B Upgrade	£44.75
BBC 1	16K Memory	£18.00
BBC 2	Printer User 1.0 Kit	£ 7.50
BBC 3	Disc Interface Kit only supplied with Disc Drives	£95.00
BBC 4	Analogue Input Kit	£ 6.70
BBC 5	Serial 1.0 RGB Kit	£ 7.30
BBC 6	Bus. Expansion Kit	£ 6.45

All kits are supplied with full fitting instructions.

BBC CONNECTORS

BBC 21	Printer Cable and Amphenol Plug (not assembled)	£13.00
BBC 22	User Port Connector and Cable 36"	£ 2.46
BBC 23	Cassette Lead	£ 3.50
BBC 24	7 Pin Din Plug	£ 0.60
BBC 25	6 Pin Din Plug	£ 0.60
BBC 26	5 Pin Din Plug	£ 0.60
BBC 35	Disc 1.0 Cable 34W IDC to 2 x 34 way Card Edge	£12.00
BBC 36	Disc Power Cable	£ 6.00
BBC 44	Analogue Input Plug & Lever	£ 2.25
BBC 66	1 M Bus Connector + 36" Cable	£ 3.50

BBC ACCESSORIES

BBC 45	Joysticks (per pair)	£11.30
BBC 67	Eprom Programmer (assembled)	£57.95

12 month "no quibble" warranty on all products

ACORNSOFT FOR THE BBC

SBE03	Business Games	£ 8.65
SBE04	Tree of Knowledge	£ 8.65
SBE02	Peeko Computer Inc Manual	£ 8.65
SBE01	Algebrail Manipulation Pk	£ 8.65
SBX01	Creative Graphics Cassette	£ 8.65
SBX02	Graphs & Charts Cassette	£ 8.65
SBB01	Desk Diary Inc Manual	£ 8.65
SBL02	Lisp Cassette	£14.65
SBL01	Forth Cassette	£14.65
SBG01	Philosophers Quest	£ 8.65
SBG07	Sphinx Adventure	£ 8.65
SBG03	Monsters	£ 8.65
SBG04	Snapper	£ 8.65
SBG15	Planetoid	£ 8.65
SBG06	Arcade Action	£10.35
SBG05	Rocket Raid	£ 8.65
SBG13	Meteors	£ 8.65
SBG14	Arcadians	£ 8.65
SBG10	Chess	£ 8.65

ACORNSOFT BOOKS FOR THE BBC MICRO

SBD01	Creative Graphics	£ 7.50
SBD02	Graphs - Charts	£ 7.50
SBD04	Lisp	£ 7.50
SBD03	Forth	£ 7.50

* Please ring for current delivery on Acornsoft products before ordering.

BBC MICRO COMPONENTS

4516 100NS	£ 2.25
6522	£ 3.19
74LS244	£ 0.59
74LS245	£ 0.69
74LS163	£ 0.34
DS3691N	£ 4.50
DS88LS120N	£ 4.50
UPD7002	£ 4.50
8271	£36.00
20 Way Header	£ 1.46
26 Way Header	£ 1.76
34 Way Header	£ 2.06
40 Way Header	£ 2.32
15 Way D Skt	£ 2.15
6 Way Din Skt	£ 0.90
5 Way Din Skt	£ 0.90

BBC SOFTWARE IN ROM.

Wordprocessor "View"	£52.00
1.2 MOS	£10.00

...from East Anglias leading supplier

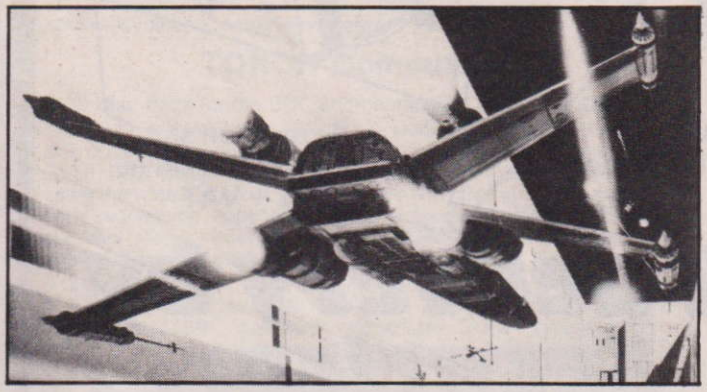
Delivery charges
Computers/Disc Drives £7.50
Components/Software £0.50
Books/Joysticks £1.00

MIDWICH COMPUTER COMPANY LIMITED
RICKINGHALL HOUSE, RICKINGHALL, SUFFOLK IP22 1HH
TELEPHONE (0379) DISS 898751

A&B

Volume One Numbe

Editor: Elspeth Joiner
Deputy Editor: Roger Munford
Consultant Editor: Henry Budgett
Advertising Manager: Miriam Roberts
Managing Editor: Ron Harris
Managing Director: TJ Connell



A&B Computing is constantly on the look-out for well-written articles and programs for publication. If you feel that your efforts meet our standards, please feel free to submit your work to us for consideration.

All submitted material should be typed, double spaced if possible, and any program submitted should be listed, a cassette of the program alone will not be considered. All programs must come complete with a full explanation of the operation and, where relevant, the structure; cassettes of the program should also be included so that screen photographs and printer dumps can be included to illustrate the article. (Cassettes will, of course, be returned in due course).

All submissions will be acknowledged and any published work will be paid for at competitive rates. All work for consideration should be sent to the Editor of A&B Computing at our Charing Cross Road address.

vo July/August 1983

Origination and design by MM Design & Print, 145 Charing Cross Road, London WC2H 0EE

Published by Argus Specialist Publications Ltd, 145 Charing Cross Road, London WC2H 0EE. Telephone: 01-437 1002

A&B Computing is published bi-monthly on the fourth friday of the month. Distributed by: SM Distribution Ltd, 16-18 Trinity Gardens, London SW9 8DX. Telephone: 274 8611. Printed by Henry Garnett Ltd, Rotherham.

Software for the BBC Micro

....ROMs....

Available Soon! SEND FOR DETAILS

Beeb-calc

A ROM based spread sheet program, like Wordwise this firmware is very fast and simple to use — yet is a powerful spread sheet analysis program, considerably better than the original 'calc' program — full floating point maths. Works in 80 or 40 column screen modes, variable column widths. Works with either cassette or disk.
This ROM coupled with Wordwise can turn your micro into an ideal small business machine.
£34.00 + VAT

Debugging Program

2 machine code programs — one in ROM, one on tape. Essential for the machine code programmer. An ideal compliment to the assembler built into the BBC machine. Contains a full machine code monitor allowing examination and alteration of memory, registers, setting of break points and even single stepping through machine code programs.

ROM based £19.00 + VAT
cassette £10.00 + VAT

Disk Doctor

A ROM containing useful disk utility programs. Enables the recovery of any data off the disk including deleted files etc. The full disk editor allows the alteration of any bytes directly on the disk (or in memory), or the loading and saving of any track or sector on the disk. Automatic transfer of programs from tape to disk and visa versa.
Includes a whole host of other useful utilities — string search, function key editing, the ability to format 35, 40 & 80 track disks.
£19.00 + VAT

Printer toolkit ROM

This includes routines for high resolution screen dumps for both the EPSON and NEC printers. Will work in any graphics mode with automatic grey shading of all screen colours. The most useful feature of this program is its 'spooling' capability, this enables data such as a program listings or high res screen dumps to be automatically spooled from your disk to the printer **while** using your BBC machine for running other programs.
£19.00 + VAT

Quantity discounts, starting at five off.

Wordwise

THE WORD PROCESSOR FOR THE BBC MICRO

Wordwise

The word processor for the BBC machine.
This ROM based word processor has received superb reviews.
A powerful and flexible system — it's greatest strength being that it is fast, simple and very easy to use. No other existing word processor (even ROM based, costing twice as much) can compare with the flexibility of this system.
Supplied with full spiral bound manual and cassette containing an example document and free typing tutor program.

£39.00 + £1.50 p&p + VAT

Now available from stock. Quantity discounts, starting at five off.

all ROM based software requires new series 1 Operating Systems. We are now in the position to supply 1.2 ROM's for £5.00 + VAT when you buy any of the above software.

CASH OR ROYALTIES waiting for any interesting software (not just games) for the BBC Micro. The better the program the more we will pay.

Computer Concepts

Dept AB2
16, Wayside, Chipperfield,
Herts, WD4 9JJ. tel (09277) 69727

Buy it with Access

BARCLAYCARD VISA

Menu

INTRO

Having got off to a fine start in our first issue of *A&B Computing*, the second carries on where we left off.

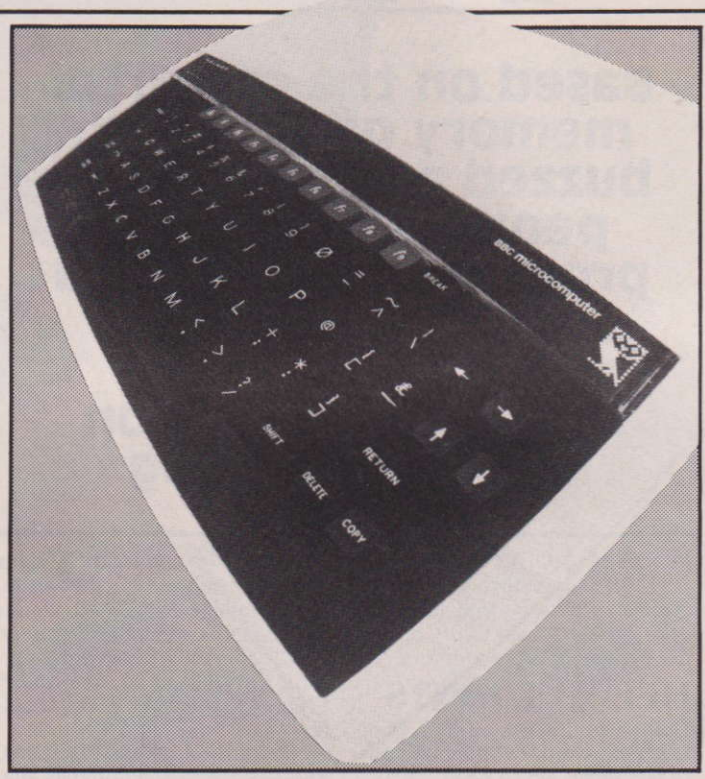
Judging from the letters we received from people who bought the launch issue of *A&B Computing*, it was generally to your liking. Lots of you had welcome suggestions for improving the magazine, which I will bear in mind for future issues of *A&B Computing*. Not everything was perfect for you, but with time, hopefully, it will be.

Inside this issue our main hardware review is the new Torch Computer Disc System which runs with the BBC Microcomputer, as featured on our front cover. Our thanks go to Torch for providing us with the cover illustration. The Torch gets a good going over, right from installation stage through to the performance phase, and is evaluated in terms of its price accordingly. If you are thinking of broadening your computer horizons to include a disc system, don't make a final decision until you have read our review.

Also on the hardware side you'll find a review of Ikon's newly launched micro tape drive, which could provide you with an alternative to a disc driven loading system. Called the Hobbit it literally looks like a 'black box' — and it certainly contains a lot of tricks. But I won't spoil your fun — read all about it in the article itself.

And while we're on the subject of disc drives the BBC's own are further explored this issue.

This feature explains some of the advantages and processes involved in using your disc drives. By the way, in future issues of the magazine we'll include more articles about disc drive systems and how you can get more out of them. At the same time we won't forget that not all BBC Micro owners, want, or can afford, discs so a vast majority of our material will cater for them.



CHALKING UP

We are including several programs in this issue with a strong educational content. First up, there's a program showing you how to put more pie charts and graphs on-screen, both for school and home use. Also, there's a couple of great little programs for younger users of the BBC Micro to learn elementary arithmetic in a fun yet informative way.

SOFTWARE SCENE

If any of you have in your software library Acornsoft's Monsters game then we've got an article which will be of interest to you. The drawback with this particular program, good though it is to play, is that you control the game using the keyboard. Seasoned games players will know that key control is in most circumstances awkward to say the least. Our article overcomes that difficulty by telling you what amendements to make to the original program so that it will run with a pair of joysticks.

COMPETITION

Up for grabs in our summer issue is a mammoth £1,000, kindly put for the taking by Acornsoft. All you have to do to stand a chance of winning this grand prize is write a program which will play a legal game of the ancient strategy game, 'GO'. Full details for entering the competition are given later in the magazine (see contents for the page number) as well as an in-depth description of how to play GO for those who are unfamiliar with the game and a few programming hints. So get up and go to the relevant pages!

We are always on the lookout for good programs and articles for future issues of *A&B Computing*, and where better to look than to our own readers. If, when reading through the magazine, you think you can write programs as well, or better than, our present contributors, then lets hear from you.

All contributions are, of course, paid for at very competitive rates. So, if you've got your eye on a new BBC add-on or you'd just like to supplement your pocket money, get writing! It is vital, though, that all the programs you send to us are totally original, and not 'borrowed' or 'adapted' from other magazines or books.

Any kind of program (business, domestic, educational, or just fun) will be welcomed, but particularly those which use BBC BASIC in clever and efficient ways, or those which employ certain routines which can be used in other programs.

Program listings are vital, along with a clear explanation of how the program is constructed, what it does and what the user can expect to see once the program is RUN (a screen dump is particularly valuable in this respect). When submitting your programs, it is very important to remember to enclose a cassette of the program as well as the listing, as this will allow us to check the program before publication.

Elspeth Joiner,
Editor.

BBC Copycat

J Stacey

Based on the colourful memory game which buzzed and bleeped in people's hands this program provides you with a tantalising challenge as well as a basic education in programming

This is a bright and breezy computer game that tests your potential for remembering a pattern of flashing coloured squares. It's a bit like the popular memory game, Simon, only this computer version deals with colour alone, leaving sound aside.

When first running the program you will be confronted by the title page which gives a brief outline of the game and the instructions. After pressing a key the computer screen displays a large box containing a cross. For a short time you will notice that the four 'windows' will each be filled with a different colour.

The computer takes its turn first by printing up on the screen 'COMP TURN', and flashing the first square. Now the message 'YOUR TURN' appears, and you have to press the cursor key corresponding to the computer's squares. If you get the right answer the computer prints 'correct' on the screen. If you get the wrong answer the end page is shown giving your score and asking if you want to play another game.

This program was written specifically for a BBC Micro Model B but with the following alterations it will run on the Model A. Change MODE 2 to MODE 5 and alter the VDU 19 statements and the program should swing into action on your Model A. All told Copycat uses up about 3K of memory.

HELPFUL HINTS

Generally the program is fairly straightforward, but there are one or two things which need explaining. They are as follows.

In line 30 the '! &FE00 = &10200A' statement turns the cursor off. Variables with '%' signs after them indicate that they are integer variables. *FX4,1 causes the cursor keys to produce an ASCII character rather than perform their usual function and *FX15,1 flushes the keyboard buffer.

When DRAWing, PLOTting, and MOVEing, the numbers after them refer to the X,Y, co-ordinates of the pixels on the screen. They are numbered from 0,0 in the bottom left-hand corner of the screen to 1279, 1024 in the top right-hand corner. PLOT is of the form PLOT 85,X,Y, where the 85 means plot and fill a triangle from the last two points visited and X,Y. CGOL changes the colour used to plot triangles.

The VDU statements in the program change the logical colour 'x' to the actual colour 'y', and it is used to flash the coloured squares on and off. Lines 130 and 410 are exactly the same and they just move the cursor and print the title for the title page, and the end page. It can be replaced by PRINTTAB (X,Y)"title".

CALL &DBBE calls the break routine, which is used to change to MODE 4 when the break key is pressed, or a wrong entry is encountered. This is done in this manner because if the mode was changed as a direct line, the error 'bad mode' would come up.

In line 280 the GET statement returns a number between 1 and 4, and hence the −&87 on the end of it.

LINE DESCRIPTION

10-20	Control error handling routines and traps the break key.
30-70	Main program loop. Call procedures to set variables and display the coloured boxes.
80-100	Procedure to draw box and plot the four 'windows' of colour.
110-180	This procedure defines the characters, displays the instructions and waits until a key is pressed before starting.
190-250	Procedure to light up the colours corresponding to the computer's turn or your key depressions.
260-330	This procedure gets your key depression and lights up the corresponding colour. It also contains statements to check whether or not the input was correct.
340-370	If your answer is correct this procedure prints 'correct' on the screen and produces a sound.
380	Calls the break routine.
400-480	End routine. If you get one answer wrong it plays a harsh note and displays your score. It then resets the variables and asks if you want to play another game or not.

PROGRAM LISTING

```
%L.
   10 ON ERROR GOTO 20
   20 *KEY10 MO.4|MOLD|MGOTO390|M
   30 MODE6:!&FE00=&10200A:PROCINIT:MODE2:!&FE00=&10200A:PROCSETUP:FORA
=1TO1500:NEXT:DIM D 100:Y%=1:S%=0
   40 *FX4,1
   50 FORC=1TO100:D?C=RND(4):NEXT
   60 FORC=1TOY%:PRINTTAB(0,2)"      COMP TURN":PROCL(D?C):NEXT:PRINTTAB
(0,2)STRING$(19," "):FORC=1TOY%:PRINTTAB(0,2)"      YOUR TURN":PROCYOUR:
NEXT:PRINTTAB(0,2)STRING$(19," "):Y%=Y%+1:GOTO60
   70 END
   80 DEFPROCSETUP
   90 MOVE160,128:DRAW160,896:DRAW1120,896:DRAW1120,128:DRAW160,128:MOV
E160,512:DRAW1120,512:MOVE640,128:DRAW640,896:GCOL0,3:MOVE176,880:MOVE6
24,880:PLOT85,176,528:PLOT85,624,528:GCOL0,1:MOVE656,880:MOVE1104,880:P
LOT85,656,528
  100 PLOT85,1104,528:GCOL0,6:MOVE656,496:MOVE1104,496:PLOT85,656,144:P
LOT85,1104,144:GCOL0,5:MOVE176,496:MOVE624,496:PLOT85,176,144:PLOT85,62
4,144:VDU19,3,0,0,0,0,19,1,0,0,0,0,19,6,0,0,0,0,19,5,0,0,0,0:ENDPROC
  110 DEFPROCINIT
  120 VDU23,224,15,24,51,54,54,51,24,15,23,225,240,24,204,12,12,204,24,
240
  130 VDU31,13,2,67,32,79,32,80,32,89,32,67,32,65,32,84,31,13,3,45,32,4
5,32,45,32,45,32,45,32,45,32,45,32,31,11,5,224,225,32,74,46,32,83,116,9
7,99,101,121,32,49,57,56,51,31,0,7
  140 PRINT" This game is like the game of 'Simon', you have to watch t
he sequence played bythe computer and try to copy it."''''''''   USE THE
 CURSOR KEYS FOR YOUR MOVE"
  150 PRINT''''"      ANY KEY....."
  160 *FX15,1
  170 A$=GET$:IF A$="" GOTO170
  180 ENDPROC
  190 DEFPROCL(M)
  200 IFM=1 THEN VDU19,3,3,0,0,0:SOUND1,-15,25,5:GOTO240
  210 IFM=2 THEN VDU19,1,1,0,0,0:SOUND1,-15,50,5:GOTO240
  220 IFM=3 THEN VDU19,6,6,0,0,0:SOUND1,-15,75,5:GOTO240
  230 IFM=4 THEN VDU19,5,5,0,0,0:SOUND1,-15,100,5
  240 FORT=1TO1000:NEXT:VDU19,3,0,0,0,0,19,1,0,0,0,0,19,6,0,0,0,0,19,5,
0,0,0,0
  250 ENDPROC
  260 DEFPROCYOUR
  270 *FX15,1
  280 J=GET-&87:IFJ<1OR J>4 THEN280
  290 PROCL(J)
  300 IF D?C<>J THEN 380
  310 S%=S%+1
  320 IF D?C=J AND Y%=C PROCCORRECT ELSE IF Y%=C ANDD?C<>J THEN C=Y%:GO
TO380
  330 ENDPROC
  340 DEFPROCCORRECT
  350 *FX15,1
  360 PRINTTAB(0,2)"                    ":PRINTTAB(0,2)"      CORRECT":FO
RT=1TO255:SOUND1,-15,T,0:NEXT:FORF=1TO400:NEXT
  370 ENDPROC
  380 CALL&DBBE
  390 CLS
  400 FORT=1TO5:SOUND0,-15,70,10:NEXT
  410 VDU31,13,2,67,32,79,32,80,32,89,32,67,32,65,32,84,31,13,3,45,32,4
5,32,45,32,45,32,45,32,45,32,45,32,31,11,5,224,225,32,74,46,32,83,116,9
7,99,101,121,32,49,57,56,51,31,0,7
  420 PRINT''"      You scored ";S%;" points."
  430 PRINT''''''"     ANOTHER (Y/N).....";
  440 Y%=1:S%=0
  450 A$=GET$:IFA$=""THEN450
  460 IFA$="Y" THEN RUN
  470 CLS:PRINT'"     Goodbye"
  480 END
```

News Ne

ELECTRON TO BEAT SPECTRUM?

Acorn Computer's managing director Chris Curry says Sinclair Research's recent price drops are due to the Electron launch.

The firm's £150 computer will be a direct rival to the ZX Spectrum and by dropping the price of the machine Sinclair must hope that the new launch will increase Spectrum sales. Now the 16K Spectrum is on sale for under £100, and the 48K version costs £129.95. With the Electron due to weigh in with 32K of on-board memory and a price tag of £150 it makes it a very attractive deal. The Electron has the disadvantage of any new product in that it is not a tried and tested product, as is the Spectrum.

Chris Curry said: 'We quite expected Sinclair's prices to drop just prior to the launch of the Electron, because the Spectrum cannot compete with the Electron in the same price bracket.'

He also welcomed the ZX81 price drop because he believes it will broaden the base of the home computer market and will probably make new recruits aware of other products. That means higher sales of other computers, including the BBC models and hopefully the soon to be released Electron. Judging from Curry's comments he has high hopes for the Electron. Let's hope that the machine lives up to them.

TO CATCH A THIEF

With one burglary taking place every three minutes it's not surprising that computer thefts are on the increase particularly as many of them, including the BBC Micro computer can now be moved more easily than a television set.

That's why a north country firm, Rothron Electronics, has come up with a device to protect your computer from the light fingers of a bunch of thieves. Originally the idea was to protect video recorders hence the device's name, the Rothtron Video Alarm. But the manufacturers soon cottoned on to the fact that it can easily be attached to other expensive electronic equipment including computers.

The device comes in a small black box that is attached to the mains lead coming from the computer and plugs into the wall socket. It works on rechargeable batteries and the 13amp plug does not even have to be switched on for it to burst into action if necessary.

If the mains lead is unplugged or cut in order to remove the computer the anti-theft device emits a piercing scream which, I can assure you, would be enough to give an unsuspecting burglar a heart attack on the spot, never mind leave the computer and the building as fast as his legs could carry him.

The price of this device is quite high at £54.50 (plus VAT) but you may think it is worth the extra expense to protect your computer system.

NOT THE 9 O'CLOCK NEWS

Acorn Computer has just come out with a speech chip and aptly had the well-known voice of newsreader Kenneth Kendall printed on its circuitry for good.

Kendall's dulcet tones are the first, believe it or not, British ones to be heard on computer. Up until now the only voices to be reproduced on chip have been the twang of American tones reverberating through your computer's speakers. Acorn Computer's brand new speaking chip has 165 words available for you to use which includes both letters and numbers on the keyboard. There is also access to keyboard related words like stop, start, correct and incorrect.

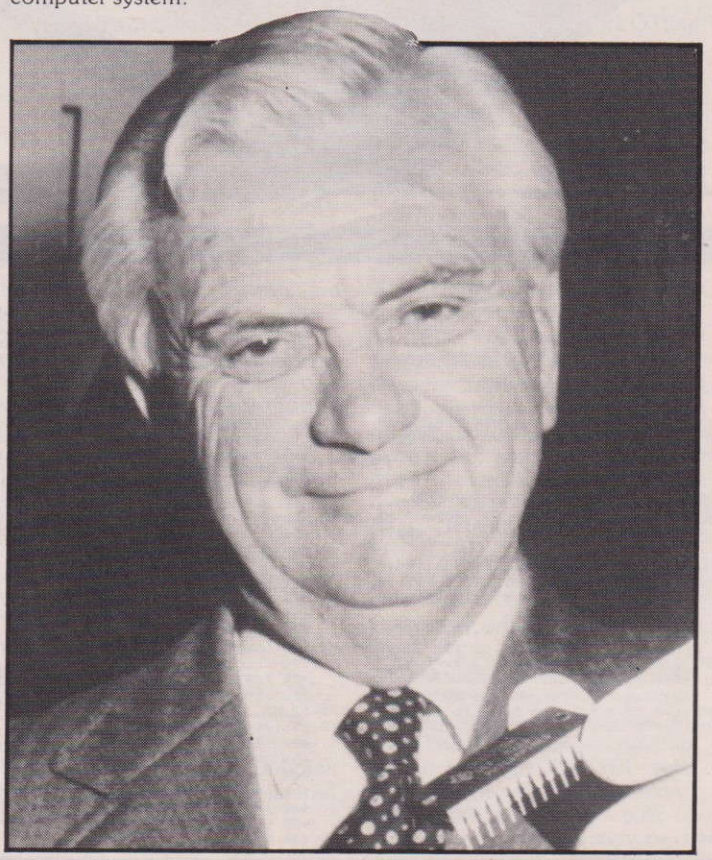
The £55 chip comes in a package from Acorn Computer and comprises the chip itself and a PHROM – which stands for Phrase Read-Only Memory. It's that part which contains the digitised sounds of Kenneth Kendall's voice. One thing to bear in mind, you *must* not fit the device to your BBC micro yourself, it will have to pay a visit to your local dealer.

Another slight inconvenience is that it only runs on the 1.2 operating system. If your computer doesn't already have this ask your dealer to add the necessary items. You also need two sockets to be inserted to the left of the BBC micro's keyboard into which you can plug in program cartridges.

The new chip is said to have a particularly good clarity in comparison to American equivalents. The reason for this clarity has been achieved lies in the fact that two chips are fitted in the upgrade. One is a Texas TMS 5220 Speech Processor and the other is the aforementioned PHROM which is 16K and contains the necessary data for the speech processor to work on.

Acorn Computer reckons that the chip will be particularly useful for those who suffer from dyslexia and the blind. The firm also hopes that school children will benefit from the new development particularly those who have access to their school's machine where the BBC micro has achieved 80% penetration.

If you're interested in fitting your own computer, or your school's machines with a few vocal chords, get in touch with your local dealer for information about inserting the chip and upgrade.



ECONET FOR Z80s

SJ Research of Cambridge have produced an interface card for Z80 based micros which will enable them to link up with the BBC Micro's Econet system.

Econet has already found wide acceptance in schools using the BBC Microcomputer because it allows separate workstations to pass information and files to and from each other. It also enables these workstations to share common resources, such as printers and disc drives.

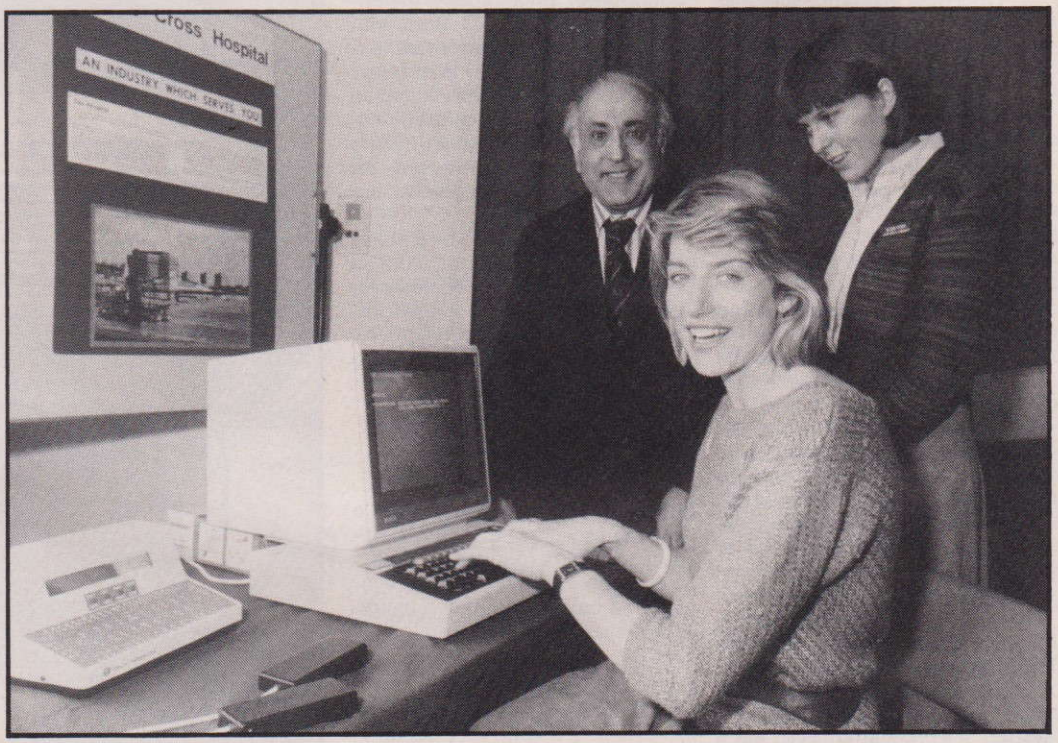
WS News

The interface card has been designed for use with the Research Machines 380Z computer, a machine which is also found in a large number of schools. By using this interface you will be able to tie the 380Z into your Econet network and communicate with the other stations.

Alternatively you can employ the 380Z as a file server. This means that instead of tying up the network when printing out a long program or when doing a complicated disc access, you can send the information to the 380Z and let it carry on with the job on its own without using the network.

The cost of the basic network interface is £153.00 plus VAT. The complete package, which includes an extra 64K of RAM and a full software package to run a sophsticated file server, costs £495.00.

Further information about the interface can be obtained from SJ Research, 108 Mill Road, Cambridge, CB1 2DB.



ARE YOU IN CONTROL?

A number of graphics input devices have been introduced for the BBC Micro which will allow much more control in applications in CAL (Computer Assisted Learning) and CAD (Computer Aided Design). Marketed by Academic Software, these devices are for use with the BBC Micro Model B (or Model A with 32K RAM and analogue input socket).

The CADSTIK is a joystick with alternative double thumbwheel control for precise vertical and horizontal lines. Twin pushbuttons direct the MOVE and PLOT options and the device is mainly driven from a screen-displayed menu; additional options being provided from the function keys.

The CAD—GET, a graphics entry tablet, is a jointed-arm potentiometer with twin pushbutton control. Each tablet is individually calibrated, using computer-based iterative pit-mapping to ensure maximum accuracy of performance.

The menu options for both these devices include Erase, Redraw, Text, Trace, Full-line/Dotted-line, Rectangle, Arc/Circle, Polygon, Fill/Draw, Background and Foreground colour selection. Both devices can operate in MODE 4 or MODE 5 interchangeably and since designs can be saved on cassette or disc, any number of separate design elements can be re-loaded and combined with each other or with 'live' designs, the composite design being saved if required.

The BASIC software, supplied on disc or cassette with the package, can be used with any Operating System (0.1,1.0 or 1.2) and with the disc and Econet interfaces installed. As the program dimensions the storage array according to the free memory available when you power-up your machine, the program may be extended or changed by the user as required.

Both devices are in use in schools, colleges and teacher's centres in West Yorkshire, and are now available for more widespread demand. Prices for the CADSTIK and CAD-GET are £45 and £70 respectively: a left-handed version of the CAD-GET is also available on request, as well as an option to have a thumbwheel control (this latter option will make the price of the CAD-GET £75). As a further incentive, Academic Software are offering discount to educational establishments and for orders of large quantities of their devices.

For further information on these control devices, contact Academic Software, Sourby Farm, Timble, Otley, West Yorkshire LS21 2PW or 'phone Blubberhouse (094-388) 628.

100,000 UP

On Wednesday, 4 May at 10.30am, the one hundred thousandth BBC Micro was presented to its new owners, the Speech Therapy Unit at Charing Cross Hospital in London.

Presented by BBC Breakfast Time's Selina Scott, the micro was gratefully received by the hospital's Chief Speech Therapist, Alison Perry. Amongst others attending this momentous occasion were Hugh Rossi, Minister for Social Security and the Disabled, Bryon Parkin, Managing Director of BBC Enterprises, and Sir Roy Redgrave, Chairman of the Hammersmith and Fulham Health Authority.

The Charing Cross Hospital is the first of five hospitals to start a Communications Aid Centre, scheduled to open in July and jointly funded by the DHSS and the charity for the disabled, RADAR. The new BBC Micro is to play a key part in the new centre and will be used in conjunction with Toucan, a portable, battery-operated voice synthesiser and visual aid for speech-impaired handicapped people. Thus, each patient can use the system programmed with individual words and phrases specifically designed to aid their condition.

INSURE YOUR COMPUTER

Computer systems are very expensive objects and are costly to replace if lost through accident, theft or disaster.

CONTINUED OVER

News Ne

Until now insurance schemes for such equipment has been reasonably hard to come by. But a new protection scheme for personal computers has just been put into action by insurance brokers Graham Brown and Company. The firm claims that this is the first policy of its kind relating specifically to personal computers.

The Personal Computer Insurance Scheme has been drawn up to cover both computers and peripherals in the home, educational establishments and also the office if in temporary use there. The idea behind the scheme is to protect the computer after the manufacturer's guarantee period has run out. Graham Brown estimate that by the middle of this year 50% of computers will be beyond the protection of their manufacturer's guarantee.

This brand new policy provides an 'all risk' clause to cover users/owners against internal breakdown and accidental loss or damage as well as the inclusion of damages which occurs during transit. It certainly sounds like a comprehensive package.

You can take out this insurance policy for a minimum period of one year. The charges are worked out according to the value of your computer system, much in the same way as standard policies covering house contents. To give you an idea of the pricing structure a system worth £100 will cost you £7.50 a year to insure while for a £500 system the price rises to £15 a year.

There are, of course, stipulations involved in the policy. To be eligible for the policy your computer system must be no more than two years old at the time the insurance is taken out. Once your policy has been taken out you can continue it for as long as you require.

Chris Bowen, a director of the insurance firm Graham Brown, said of the new scheme: 'There are almost a million personal computers in use in this country today, of which some 75% have been bought within the last year. Many owners are now losing the benefit of their guarantees. In a rapidly expanding market such as this, with another million computers expected to be sold this year, manufacturers have little incentive to offer extended guarantees. Under most home contents policies, the only protection an owner can expect is against fire and theft.'

If you are in need of a policy of this kind why not contact Graham Brown & Co (Guildford) Ltd, Pannells Court, Guildford, Surrey GU1 4EY, or telephone them on Guildford (0483) 65651.

A MONSTER OF A MICRO

Anyone who has a BBC micro knows that the machine's applications are many and varied but a firm specialising in designing and building animated figures has come up with one use to beat the lot.

N J Farmer Associates have used their BBC micro computer to program the movements of a giant demon beast. Not a real one of course, but a plastic monster. And it's the star attraction at a chamber of Horrors type walk-through down in sunny Great Yarmouth.

Where the computer came in handy during this monster's creation was to do with its movement actions and co-ordination. The finished human-shaped model's resulting movements and lighting are apparently very realistic.

The program for movement and lighting was transferred through a series of coded signals to a spare track on the audio tape loop. On playback the program is decoded and operates the solenoid valves in perfect time with the sound track.

Farmer's monster stands over 10 feet tall and is made of different types of plastics and rubber over a steel skeleton. He's got all the usual and vital parts which move like eyes, jaw, neck, waist and leg and various bits of him light up in a special sequence, just to give an even more grisly effect.

At the moment the mighty monster lurks within the Terror Castle at Great Yarmouth's

WS News

Marine Parade. Why not take a trip to the seaside and pay him a visit?

ELECTRON FEVER

Acorn's new Electron computer will be on sale in August.

That's the latest update on the Acorn computer which was first announced just before the Spectrum's launch in the early summer of 1982. Since then the Electron vanished into the depths of Acorn Computer for development purposes.

But now its on the brink of being unveiled and reports are that to look at it's very similar to the BBC Microcomputer having the same amount of memory as the Model B and is built around the same processor. Colour-wise it's the same shade of cream as its sister machine the BBC B, but there is green lettering and the Acorn logo on the brown strip. The Electron measures 13 in wide, 2 in high and 6½ in deep, it's a bit slimmer than the BBC micro.

Acorn has based its design both externally and internally on the BBC micro it developed for the television corporation. By all accounts it has a slower processing speed than its rival Beeb machine, it only has one sound channel and lacks the Mode 7 graphics facility. Standard interfaces for colour TV, composite video and PAL or RGB monitors and cassette input have been incorporated into the Electron.

It has a 'proper' keyboard with full-travel keys unlike other low-cost computers like the Spectrum. There are 10 programmable function keys available on the Electron which are controlled by the caps lock key. All told there are 29 keys and a single key entry system for BBC BASIC commands like CHAIN, RENUM, MODE, COLOUR and LIST.

And according to David Johnson-Davies, managing director of Acornsoft; 'The BBC's software is 90% compatible with the Electron. There will be a new range of software, they will have identical titles of the BBC range but there will also be minor changes'. We can't wait to see it.

● Latest figures for BBC Micro computer's production stands at 17,000 a month from June. This is only in the UK but the increase gives an indication as to the speed at which they are now selling. And another major shop is now stocking Acornsoft products for the BBC Micro computer. W H Smith has added Acornsoft's range of software to its Spectrum range in selected branches. It is hoped that more branches of the store will stock Acornsoft's products so watch out in your local Smiths for displays of them.

CHIPCHAT SHOW

A two hour special phone-in show is planned by the BBC as part of the giant corporation's computer literacy project.

The live show will be on your screens on the morning of Sunday October 2 later this autumn. Experts will be on hand to answer any telephone queries which come through from viewers on the subject of micro computers. Ian McNaught-Davis is presenting this mammoth show, viewers of the BBC Computer Programme will recognise his face from that series. Helping him out will be a panel of computer experts as well as some micro users and a studio audience.

As all electronic equipment tends to become temperamental under the eagle eye of the studio cameras this could prove to be a very interesting show.

What the team want to do with the phone-in is to gather queries from people right now. They will sift through all the letters that are sent to them and select the most interesting ones which can be dealt with on the live programme. So if you've got a micro problem and would like it answered on the show write it down, along with other suggestions or comments, to Micro Special, PO Box 7, London W3 6XJ. Don't forget to put a daytime telephone number with your letter.

If you're question is chosen to be used in the programme one of the team will get in touch with you nearer the date to finalise arrangements. In addition the telephone lines will be open for a couple of hours before the show in on the air and any stunningly interesting queries will be dealt with almost directly during the show.

Additional features of the show will include demonstrations of the equipment on show in the studio – providing they behave themselves. People on the panel of experts include John Coll, the man who wrote the BBC Micro manual, freelance software writer Ian Trackman, David Ellis whose expertise lies in music on computers, graphics expert from Middlesex Polytechnic John Vince; computer journalists Malcolm Peltu and Henry Budgett; a manager from an Information Technology Centre; Chris Webb; Prestel's Richard Hooper; a representative from the Microelectronics Project, Richard Fothergill; and somebody from the BBC's Ceefax service.

Although a live show of this kind has never been done before estimates for audience figures run to the three million mark, which is pretty good for a Sunday morning programme. The brains behind this show belong to Patrick Titley and David Allen who were closely involved in the series The Computer Programme and Making the Most of the Micro.

● Plans are afoot for three more computer programmes to be shown on BBC television. These include a series of six episodes each of which will last 25 minutes, to be called the Electronic Office; Computer Club is aimed at young computer users and the third is to be called Computers in Control which will feature the BBC buggy.

IS THAT CLEAR?

You can now get clear, high quality printed listings of your computer programs – without the aid of a printer!

All you have to do is to get in contact with Beebprint, who will despatch, by return of post, a printout of any 1200 Baud cassette. All material is handled with the strictest of confidence and your tapes are, of course, returned with the listing.

Tapes recorded with the 0.1 Operating System should use the Cassette Bug-fix recommended by Acorn, but if you're not sure about this then a stamped-addressed envelope sent to Beebprint will get you full details of the Bug-fix. Beebprint also allow you to have screen dumps of your program in action – all you have to do is to add an extra line in your listing at the point at which you want the screen reproduction. Again, words with Beebprint will make this task simplicity itself.

The costs involved are £1.95 for each program listing (maximum block count – &20) with a charge of 20 pence for every nine blocks over and above the initial 20. High resolution screen dumps will cost you 50 pence each, and if you want to send in 300 Baud tapes all costs are 50% extra.

For further information on this new printing service, contact Beebprint Specialist Computer Services, 19 Orchard Way, Hurstpierpoint, West Sussex BN6 9UB. Telephone enquiries can be made on 0273 833397.

ON SHOW

If you always spend hours walking around exhibitions trying the stands that highlight peripherals and software for the BBC Micro, you'll be pleased to hear that an exhibition is being held wholly aimed at BBC Micro users.

The Acorn User Exhibition is to take palce at the Cunard International Hotel, London, between the 25th and 28th of August. Scheduled during the school summer holidays, the organisers are expecting a large proportion of the estimated 20,000 visitors to be schoolchildren, their parents and teachers. Because of this, great emphasis will be placed on educational computing, as well as the more established area of business and personal computing.

So, if you want to get your hands on an Electron, try out the new BBC second processors and see the BBC Buggy in action, make a date in your diary.

CONTINUED OVER

News Ne

For further information get in touch with the organisers on 01-930 1612. They'll be only too pleased to help you.

HANDS ON HOLIDAY

Dolphin Activities are an organisation formed last year with assistance from the Department of Industry's Education Unit and Information Technology '82 to operate educational holiday day camps with a strong emphasis on 'hands on' computer holidays.

These holidays proved so successful that this year Dolphin have developed a broader range of holidays and substantially increased their capacity. They expect over 3,500 children to attend their computer camps this summer, but this year they are launching a new scheme to involve the parents, helping them to break down the 'technology gap'.

With around £100,000 of micros and interfacing equipment at each centre, including BBC Micro related equipment, Dolphin attempt to make the courses fun and informative. The courses are broken down into three main areas? robotics, psychobionics and computing. The computing courses are organised to help beginners learn BASIC and for the more experienced programmers, there are tutorials on the higher level languages such as Pascal and LOGO.

Everyone has the opportunity to learn the techniques of interfacing, to control and even build robots. And should you get tired of taking in all that technology, you could always take part in some of the sporting and creative activities available at the camps such as video film and cartoon making, windsurfing, go-carting and horseriding.

The camps themselves are all set in the heart of the British countryside and all offer a well-planned week or weekend.

For further information on these computing holidays, contact Dolphin Activities Ltd, 68 Churchway, London NW1 1LT or telephone 01-387 5602.

Laserbug is the newsletter of the Independent National BBC Microcomputer Users Group. If you want the best source of information on the BBC Micro you can't do without Laserbug No matter what your interest – hardware, software, business, games or education then Laserbug has something for you.

Also, Laserbug has available many special offers including dust covers (for computer, monitor, printer, disks), cassette leads and 1.2 ROMS FOR ONLY £5.50 INCLUSIVE – THE CHEAPEST PRICE ANYWHERE! (Members Only)

Laserbug defies description – send off for a sample copy and you'll find that it sells itself to you. See one and you'll be hooked for life!!!

Please supply me with

[ ] more details about Laserbug and your special offers
[ ] a sample copy for £1.00 and an A4 SAE (17p postage)
[ ] 1 UK 12 Month Subscription for £12.00
[ ] 1 UK 6 Month Subscription for £6.00
[ ] 1 Overseas Surface Mail Subscription for £14.00
(air mail rates on application)

Please send the goods to:

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

.....

I enclose a cheque/PO for £ p made payable to LASERBUG.

Please send the form to LASERBUG Dept. B, 10 Dawley Ride, Colnbrook, Slough, Berks., SL3 0QH.

Superior Systems presents Sabresoft

For the BBC Model B Microcomputer

FLEXIBASE

The most flexible database yet seen on the BBC. Individual records can be as long or short as you wish. Full machine code alphabetic, numeric or date sort on any field. Search function gives 16 search criteria with output option to screen or printer. A most useful program for all homes or small businesses.

£9.95 pp & ins. 75p

DISC RECOVERY

Reads complete disc surface byte by byte allowing backup facility on ANY disc, no matter how well protected. If you accidentally delete a file then all is not lost. Simply read the disk sector by sector until you find your "deleted" file. Similarly for corrupt or damaged discs, you may read the disc up to the point of corruption and again after the corrupted sector. By using the DMON program these blocks may be relocated and pieced together avoiding lengthy re-typing. Unfortunately the section following a corrupt point or media fault is not readable. **£9.95 pp & ins. 75p**

SIMPLYCALC

The master spreadsheet package. An essential software item for business and home allowing your imagination to solve any sort of financial or numeric problems. Simple accounts, price lists, job costing, profit and loss estimating etc. etc. are all available. All calculating or formulae is done automatically, no need to keep pressing the CALC key. Full replicate, global command, justification, insert etc.

£9.95 pp & ins. 75p

FRUIT MACHINE

A cheaper alternative to the compulsive gamblers desire to keep feeding the fruit machine with 10p's. Fruit machine code, colour graphics make an exciting package with enough nudge control to give a better return than most one arm bandits. **£4.95 pp & ins. 75p**

ABM Model A or B

A highly entertaining full machine code colour space spectacular. Protect your universe from alien attack using laser energy, refreshed from your own cities. **£4.95 pp & ins. 75p**

DMON

The best machine code monitor and disassembler available on the BBC. Features include:- Full disassembler with ASCII dump. Full Hex dump with ASCII Conversions both with printer options.
Reads any ROM currently plugged into machine and option to more blocks of memory to more convenient locations. Edit any byte of RAM. Run machine code sub routines from specified locations with option to set all registers and check their state at end of routine.
1.0 or 1.2 O.S. only ** Full machine code program in less than 2K **

£7.95 pp & ins. 75p

LOONY LANDER

Full Colour fast action machine code game. Land your moon buggy but with a difference. Moon craft meets rather awkward opposition but with your destroy laser capability it shouldn't prove too tricky. Skilful operators actually return to mother ship. Highly recommended.

£7.95 pp & ins. 75p

COSMIC ASTEROIDS

Wandering around in space can be rather problematic when you are set upon by aliens and asteroids. Many happy hours can be spent on this entertaining machine code program. High Scores etc. **£4.95 pp & ins. 75p**

Send 25p stamp for full BBC Microcomputer accessory and software list.

DEALER ENQUIRIES WELCOME **Superior Systems Ltd.**

178 West Street, Sheffield S1 4ET
Tel: (0742) 755005

HOME STUDY COURSES

30 Hour BASIC

A beginner's BASIC programming course. Available in 3 editions: Standard, ZX81 or Spectrum.

Structured Programming in BASIC

A second stage BASIC programming course.

Beyond BASIC on the Beeb:

6502 Assembly Language Programming. An easy introduction to assembly language.

All 3 courses available now as NEC correspondence courses. Write for free leaflet/enrolment form. (30 Hour BASIC text only is available, price £5.95 post free.)

MICROTRUST SOFTWARE

30 Hour BASIC

2 cassette tapes containing 62 programs from 30 Hour BASIC, for BBC Micro use only. Boxed with instruction booklet.

£11.96 inc. VAT (post free).

Crossword Puzzler

Programs to create and play puzzles plus 4 sample crosswords, boxed with instruction booklet. Available in BBC Model B and Spectrum editions.

£5.00 inc. VAT (post free).

NATIONAL EXTENSION COLLEGE
Dept 47, 18 Brooklands Avenue, Cambridge CB2 2HN

THE NEW AND UNIQUE C.A.D. PROGRAMME FOR THE BBC (32K)

AT A VERY AFFORDABLE PRICE

Ideal for teachers, designers, artists, technical drawing and numerous offer applications including your own form and stationery design etc. This programme must be seen to be appreciated — your imagination is the only factor to limit its individual applications.

- Modes 0, 1, 2, 4, 5 (can be changed when programme is running).
- Multiple display of arrays enables infinite complexity.
- **FUNCTIONS:** Line, rectangle, triangle, circle, text (upper and lower case) and colour pallet (8 colours and flashing).
- **DRAWING AIDS:** Alignment grid, circle copy, delete, free memory, purge memory, variable cursor speed, clear screen and redraw.
- Shapes can be filled or outlined (no need for Fill Routines).
- Save and load to tape in about 20 seconds or to disk in about 2 seconds. Disk version provides screen save which takes about 3 seconds for a full screen.
- SPECIAL FACILITY — Rubber band mode — A very flexible and variable line drawing facility — must be seen.
- Free "redraw" routine to enable the pictures created to be displayed in your own programmes.

Grafkey (for keyboard input) **£7.95**
Grafstik (for joystick input) **£7.95**
Grafdisk (combines both) **£12.95**

SYNTHESISER — define & save up to 16 envelopes which can be used in the second program to play not just single notes but chords to. Octaves can be changed using cursor keys. **CASSETTE £7.95 DISK £10.95**

THE KEY enables you to copy sectors and tracks from disk to disk plus backup most protected disks. MUST NOT BE USED ILLEGALLY. DISK ONLY £12.95

JOYSTICK
A pair of fully proportional joysticks. Compact, handy sized. Smart injection moulded cases **£17.95**

CASSETTE LEADS with Motor Control
DIN to DIN & Remote (BBC ORIC or LYNX) **£2.95**
DIN to 3 jacks (BBC ORIC or LYNX) **£2.95**
Dragon Lead (DIN to DIN or DIN to 3 jacks) **£2.95**
TEXAS TI 99/4A (Single Recorder) **£4.95**

MONITOR LEADS
BNC to Phono **£2.95**
Phono to Phono **£1.50**
6 pin DIN to DIN RGB LEAD **£2.95**
SERIAL PRINTER LEAD (Acorn Specs) **£7.50**
PARALLEL PRINTER LEAD **£13.95**

All Leads are Top Quality and carry an Unconditional Guarantee.

Access

CLARES MICRO SUPPLIES
222 TOWNFIELD ROAD,
WINSFORD, CHESHIRE, CW7 4AX
TEL: 060 6551374

Space Chaser

John Wilson

Watch out for the homing rocket that's out to get you in this fast action arcade style game.

An amazing maze game to key-into your BBC Micro is here in the form of Space Chaser.

Based around the theme of the incredibly popular arcade game PacMan this game will keep you enthralled and glued to your screens for hours. It runs on the BBC Model B Micro but unfortunately, there wasn't enough memory to include instructions in the program. To overcome that you'll find all you need to know about the program in the following.

As in the PacMan game the player's objective is to manoeuvre a single, fast and very intelligent homing rocket around the maze. This replaces the four ghosts in PacMan which chase you round the maze. You must avoid the rocket, eating up the paths full of dots on your way. If you run into the rocket you will be instantly destroyed.

You start off the game with five ships and 100 units of fuel. As you progress through the game your fuel, of course, decreases. To re-fuel your ship you must try to run over a strategically positioned star before the rocket does. If not you are wiped out immediately. On clearing the first screen of dots another one appears containing even more dots to absorb. Bonus points are gained for any unused fuel after you have cleared each screen.

The control keys are:

'A'	Left
'D'	Right
'W'	Up
'X'	Down
'S'	Accelerate

To help with converting the program, a list of the procedures and what they do are given in Table 1.

The only problem I can see in the program, is the entering of the maze between lines 1990 and 2180. The 'a' and the '/' are not available directly from the keyboard. (Do not use the ones on the keyboard.) When in Modes 0-6, the pattern should appear as a maze of blocks and dots. To enter the correct characters (at codes 224 and 225), a user defined key must be used. First of all type.

```
*KEY0 I !#
*KEY1 I I a
```

Now, every time an 'a' appears, press f1 and every time a '/' appears press f0. To help with entering the maze, I suggest that you enter:

```
3ll STOP return
GO TO 200 return
3ll        return
```

This will set up the defined characters. If you then change to Mode 4, you will see a block appear if you press f0, or a dot appear if you press f1. You can then easily enter the maze as already described.

This method of entering 'user defined' characters into a program directly, although a little complex, is much easier than typing in about three lines of CHR$ commands, for every line of the maze.

To save memory, the information about the maze is stored as a group of memory locations, starting from 2%, instead of an array. The information stored in these locations, tells the computer the best direction to an exit in the corridor and whether it holds a dot. So, in order to modify the maze, the data statements, between lines 920 and 1110 must also be changed as well.

Each number in the maze data, is split up into groups of bits, which mean a certain direction. These are as follows:

binary	direction
00	Down
01	Right
10	Up
11	Left

The information is split up as in

Bit	Use
0	)
1	)level number of the tunnel
2	)
3	)Direction to nearest
4	)exit up
5	)Direction to nearest
6	)exit up
7	)contents of location; 1 = Dot 0 = empty

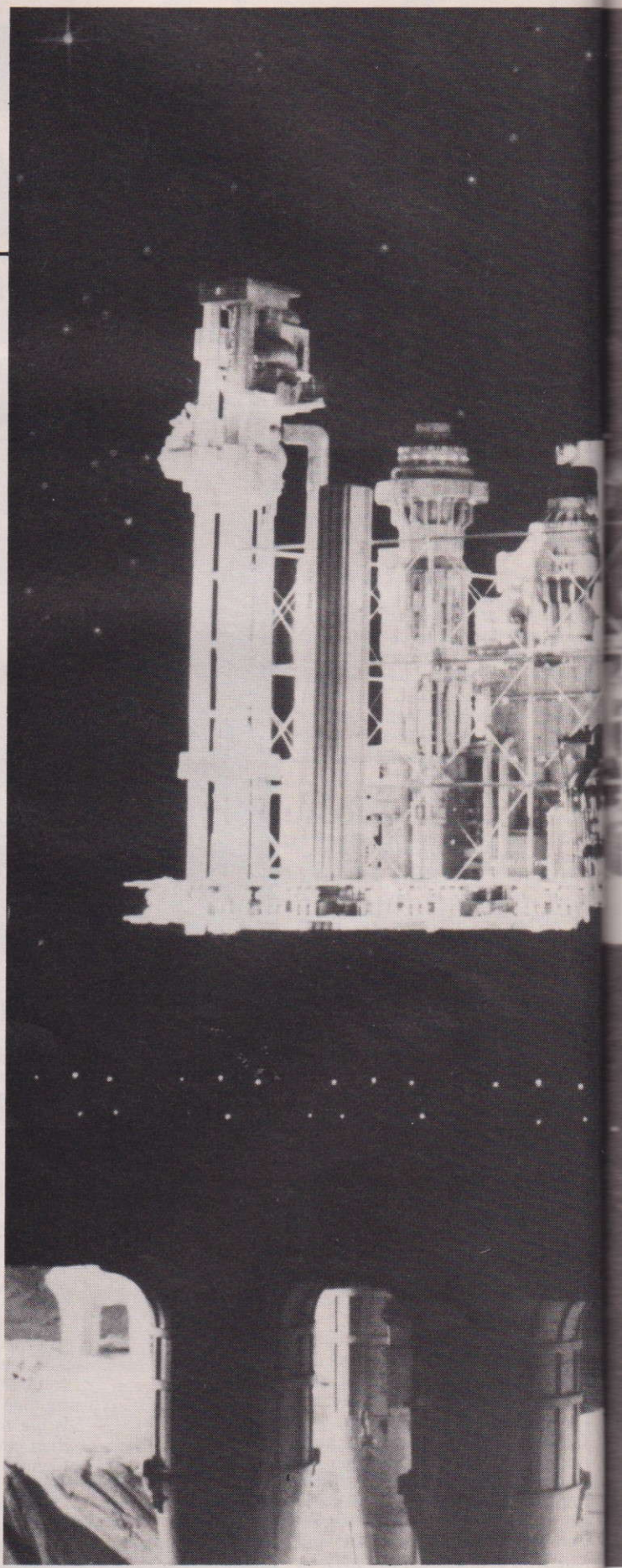
PROCEDURES

Procedure function	Use
PROSETUP	This sets up the maze in memory from the data statements.
PROCEXPLOSION	Do an explosion
PROSCREEN	Print up the partial maze.
FNP	Returns the character at the given coordinates.
PROCDOT	You have run over a dot.
PROCNEXT	You have completed a screen.
PROSTAR	You have run over a star.
PROCEND	You have been caught by the rocket.
PROCSAME	Are you in the same corridor as the rocket?
PROCLESS	Is the rocket in a closer corridor to the centre than you?
PROCBIG	Is the rocket farther away from the centre than you?
FNO	Contents of maze at that position, i.e. nearest directions for exits down.
FNI	Direction to nearest exit up, for a given location in the maze.
PROC FANFARE	Play a tune.

CONTINUED OVER

PROGRAM LISTING

```
 10 DIM Z% 400,J% 10
 20 ON ERROR GOTO 2700
 30 MODE7
 40 *FX11,1
 50 PRINT ''CHR$141;CHR$133;"S P A C E  C H A C E R"
 60 PRINT CHR$141;CHR$133;"S P A C E  C H A C E R"
 70 PRINT ''CHR$141;CHR$133"     H A L L  O F  F A M E."
 80 PRINT CHR$141;CHR$133"     H A L L  O F  F A M E."
 90 PRINT '''CHR$141;"HIGH SCORE = ";H%
100 PRINT CHR$141;"HIGH SCORE = ";H%
110 PRINT '''CHR$141;"Press the SPACE BAR to start."
120 PRINT CHR$141;"Press the SPACE BAR to start."
130 REPEAT UNTIL GET$=" "
140 G%=0
150 D%=1
160 F%=100
170 Q=0:W%=0
180 S%=0:SH%=5:X%=1:Y%=6:X1%=0:Y1%=0
190 MODE2
200 VDU 23,224,255,255,255,255,255,255,255,255
210 VDU 23,225,0,0,0,24,24,0,0,0
220 VDU 23;8202;0;0;0;
230 VDU 23,226,24,60,36,102,126,255,189,153
240 VDU 23,227,7,28,126,207,207,126,28,7
250 VDU 23,228,153,189,255,126,102,36,60,24
260 VDU 23,229,224,56,126,243,243,126,56,224
270 VDU 23,230,0,16,24,38,100,24,8,0
280 VDU 23,231,16,16,56,56,108,124,214,198
290 VDU 23,232,0,3,15,60,246,60,15,3
300 VDU 23,234,192,240,60,111,60,240,192,0
310 VDU 23,233,198,214,124,108,56,56,16,16
320 PROCSETUP
330 P%=J%
340 [OPT0
350 LDA#135:JSR &FFF4:STX &70:RTS
360 ]
370 PROCNEXT
380 A$=INKEY$(0):*FX15 1
390 P%=P%+1
400 COLOUR 132
410 COLOUR 7
420 PRINT TAB(0,28);"SCORE = ";S%;" ";
430 PRINT TAB(0,30);"FUEL = ";F%;" ";
440 COLOUR 128
450 COLOUR 6
460 PRINT TAB(X%,Y%);" ";
470 IF A$="A" AND FNC(X%-1,Y%)<>0 THEN X1%=-1:Y1%=0:D%=2
480 IF A$="D" AND FNC(X%+1,Y%)<>0 THEN X1%=1:Y1%=0:D%=4
490 IF A$="W" AND FNC(X%,Y%-1)<>0 THEN X1%=0:Y1%=-1:D%=1
500 IF A$="X" AND FNC(X%,Y%+1)<>0 THEN X1%=0:Y1%=1:D%=3
510 IF FNP(X%+X1%,Y%+Y1%)=224 THEN X1%=0:Y1%=0
520 X%=X%+X1%:Y%=Y%+Y1%
530 R%=?(Z%+X%+(Y%-5)*20):T%=(R% AND 7)
540 IF X%=M% AND Y%=N% THEN PROCEND
550 K%=FNP(X%,Y%)
560 IF K%=225 THEN PROCDOT
570 IF K%=230 THEN PROCSTAR
580 PRINT TAB(X%,Y%);CHR$(225+D%)
590 IF INKEY(-82) AND F%>0 AND P% MOD 2=0
    THEN SOUND 0,20,5,5:F%=F%-1:GOTO 380
600 COLOUR 7
610 COLOUR 128
620 IF (?(Z%+M%+(N%-5)*20) AND 128) THEN PRINT
    TAB(M%,N%);CHR$225 ELSE PRINT TAB(M%,N%);" ";
630 IF O%=1 THEN M1%=0:N1%=-1
640 IF O%=2 THEN M1%=-1:N1%=0
650 IF O%=3 THEN M1%=0:N1%=1
660 IF O%=4 THEN M1%=1:N1%=0
670 IF ?(Z%+M%+M1%+(N%+N1%-5)*20)=0 THEN
    M1%=0:N1%=0:O%=(O%+RND(3)) MOD 4 +1
680 M%=M%+M1%:N%=N%+N1%
690 IF N%=14 AND M%>X% THEN O%=2:GOTO 790
700 IF N%=14 AND M%<X% THEN O%=4:GOTO 790
```



```
 710 IF M%=9 AND N%<Y% THEN O%=3:GOTO 790
 720 IF M%=9 AND N%>Y% THEN O%=1:GOTO 790
 730 U%=(?(Z%+M%+(N%-5)*20) AND 7)
 740 IF U%=T% THEN PROCSAME
 750 IF U%>T% THEN PROCLESS
 760 IF U%<T% THEN PROCBIG
 770 COLOUR 2
 780 COLOUR 128
 790 PRINT TAB(M%,N%);CHR$(230+O%);
 800 IF M%=X% AND Y%=N% THEN PROCEND
 810 IF P% MOD 7=0 THEN P%=P%+1:GOTO 600
 820 GOTO 380
 830 STOP
 840 DEFPROCSETUP
 850 RESTORE 920
 860 FOR Y=0 TO 19
 870 FOR X=0 TO 19
 880 READ A
 890 IF A<>0 THEN A=A+128
 900 ?(Z%+X+Y*20)=A
 910 NEXT X,Y
 920 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
 930 DATA 0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0
 940 DATA 0,1,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,1,0
 950 DATA 0,1,0,2,2,2,2,2,2,10,2,2,2,2,2,2,2,0,1,0
 960 DATA 0,1,0,2,0,0,0,0,0,3,0,0,0,0,0,0,0,2,0,1,0
 970 DATA 0,1,0,2,0,3,3,3,3,11,3,3,3,3,3,0,2,0,1,0
 980 DATA 0,1,0,2,0,3,0,0,0,4,0,0,0,0,3,0,2,0,1,0
 990 DATA 0,1,0,2,0,3,0,4,4,12,4,4,4,0,3,0,2,0,1,0
1000 DATA 0,1,0,2,0,3,0,4,0,5,0,0,4,0,3,0,2,0,1,0
1010 DATA 0,1,2,34,3,35,4,36,5,5,5,5,68,4,67,3,66,2,1,0
1020 DATA 0,1,0,2,0,3,0,4,0,5,0,0,4,0,3,0,2,0,1,0
1030 DATA 0,1,0,2,0,3,0,4,0,5,0,0,4,0,3,0,2,0,1,0
1040 DATA 0,1,0,2,0,3,0,4,4,20,4,4,4,0,3,0,2,0,1,0
1050 DATA 0,1,0,2,0,3,0,0,0,4,0,0,0,0,3,0,2,0,1,0
1060 DATA 0,1,0,2,0,3,3,3,3,3,19,3,3,3,3,3,0,2,0,1,0
1070 DATA 0,1,0,2,0,0,0,0,0,0,3,0,0,0,0,0,0,2,0,1,0
1080 DATA 0,1,0,2,2,2,2,2,2,18,2,2,2,2,2,2,2,0,1,0
1090 DATA 0,1,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,1,0
1100 DATA 0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0
1110 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1120 ?(Z%+X%+(Y%-5)*20)=?(Z%+X%+(Y%-5)*20)-128
1130 ENDPROC
1140DEFPROCEXPLOSION(X%,Y%)
1150 X%=X%*64+32
1160 Y%=(31-Y%)*32+16
1170 LOCAL X1%
1180 X1%=0
1190 GCOL 0,X1% DIV 32
1200 MOVE X%-X1%,Y%-X1%
1210 DRAW X%-X1%,Y%+X1%
1220 DRAW X%+X1%,Y%+X1%
1230 DRAW X%+X1%,Y%-X1%
1240 DRAW X%-X1%,Y%-X1%
1250 SOUND 0,17,5,1
1260 X1%=X1%+8
1270 IF X1%+X%<1280 AND X%-X1%>0 AND Y%+X1%<864 AND
     Y%-X1%>192 THEN 1190
1280 GCOL0,0
1290 X1%=0
1300 MOVE X%-X1%,Y%-X1%
1310 DRAW X%-X1%,Y%+X1%
1320 DRAW X%+X1%,Y%+X1%
1330 DRAW X%+X1%,Y%-X1%
1340 DRAW X%-X1%,Y%-X1%
1350 X1%=X1%+16
1360 IF X1%+X%<1280 AND X%-X1%>0 AND Y%+X1%<864 AND
     Y%-X1%>192 THEN 1300
1370 ENDPROC
1380 DEFPROCSCREEN
1390 COLOUR 132
1400 CLS
1410 COLOUR 128
1420 COLOUR 7
1430 FOR Y=0 TO 19
```

CONTINUED OVER

PROGRAM LISTING

```
1440 FOR X=0 TO 19
1450 K%=?(Z%+X+Y*20)
1460 PRINT TAB(X,Y+5);
1470 IF K%=0 THEN PRINT CHR$224;:GOTO 1490
1480 IF (K% AND 128) THEN PRINT CHR$225; ELSE PRINT " ";
1490 NEXT X,Y
1500 COLOUR 132:COLOUR6
1510 FOR I=1 TO SH%
1520 PRINT TAB(I*2,2);CHR$226
1530 NEXT
1540 ENDPROC
1550 DEFFNP(X%,Y%)
1560 PRINT TAB(X%,Y%);
1570 CALL J%
1580 =?&70
1590 DEFPROCDOT
1600 IF ?216=32 THEN ?216=16 ELSE ?216=32
1610 SOUND 1,25,200,1
1620 S%=S%+5
1630 G%=G%+1
1640 ?(Z%+X%+(Y%-5)*20)=?(Z%+X%+(Y%-5)*20)-128
1650 IF G%=194 THEN PROCNEXT
1660 ENDPROC
1670 DEFPROCNEXT
1680 PROCFANFARE
1690 G%=0
1700 COLOUR 132
1710 CLS
1720 COLOUR 15
1730 IF S%=0 THEN 1770
1740 B=F%*2
1750 PRINT TAB(2,10);"BONUS = ";B
1760 S%=S%+B
1770 PROCSETUP
1780 GOSUB 1890
1790 FOR I=1 TO Q
1800 X=RND(18):Y=RND(20)+5
1810 IF FNP(X,Y)<>225 THEN 1800
1820 COLOUR 128
1830 COLOUR (RND(4)+7)
1840 PRINT TAB(X,Y);CHR$230;
1850 NEXT
1860 Q=Q+1
1870 O%=2:M%=18:N%=23
1880 ENDPROC
1890 COLOUR 132
1900 CLS
1910 COLOUR128
1920 COLOUR7
1930 COLOUR 132:COLOUR6
1940 FOR I=1 TO SH%
1950 PRINT TAB(I*2,2);CHR$226
1960 NEXT
1970 PRINT TAB(0,5);
1980 COLOUR 128:COLOUR7
1990 PRINT "''''''''''''''''''''";
2000 PRINT "'aaaaaaaaaaaaaaaaa'";
2010 PRINT "'a'''''''a'''''''a'";
2020 PRINT "'a'aaaaaaaaaaaaa'a'";
2030 PRINT "'a'a'''''a''''''a'a'";
2040 PRINT "'a'a'aaaaaaaaa'a'a'";
2050 PRINT "'a'a'a'''a''''a'a'a'";
2060 PRINT "'a'a'a'aaaaa'a'a'a'";
2070 PRINT "'a'a'a'a'a''a'a'a'a'";
2080 PRINT "'aaaaaaaaaaaaaaaaa'";
2090 PRINT "'a'a'a'a'a'''a'a'a'a'";
2100 PRINT "'a'a'a'a'a''a'a'a'a'";
2110 PRINT "'a'a'a'aaaaa'a'a'a'";
2120 PRINT "'a'a'a'''a''''a'a'a'";
2130 PRINT "'a'a'aaaaaaaaa'a'a'";
2140 PRINT "'a'a'''''a''''''a'a'";
2150 PRINT "'a'aaaaaaaaaaaaa'a'";
2160 PRINT "'a'''''''a'''''''a'";
2170 PRINT "'aaaaaaaaaaaaaaaaa'";
2180 PRINT "''''''''''''''''''''";
2190 RETURN
2200 DATA 999,1,1,3,5,2,3,2
2210 DEFPROCSTAR
2220 G%=G%+1
2230 SOUND 1,17,50,2
2240 L=POINT(X%*64+32,(31-Y%)*32+16)
2250 S%=S%+(12-L)*50
2260 F%=F%+50
2270 ?(Z%+X%+(Y%-5)*20)=?(Z%+X%+(Y%-5)*20)-128
2280 IF G%=194 THEN PROCNEXT
2290 ENDPROC
2300 DEFPROCEND
2310 PROCEXPLOSION(X%,Y%)
2320 SH%=SH%-1
2330 IF SH%=0 THEN 2570
2340 PROCSCREEN
2350 X%=1:Y%=6
2360 U%=1
2370 ENDPROC
2380 DEFPROCSAME
2390 IF X%=M% AND Y%>N% THEN O%=3
2400 IF X%=M% AND Y%<N% THEN O%=1
2410 IF Y%=N% AND X%>M% THEN O%=4
2420 IF Y%=N% AND X%<M% THEN O%=2
2430 ENDPROC
2440 DEFPROCLESS
2450 IF FNI<>0 THEN O%=FNI
2460 ENDPROC
2470 DEFPROCBIG
2480 IF FNO(M%,N%+1)>U% THEN O%=3
2490 IF FNO(M%,N%-1)>U% THEN O%=1
2500 IF FNO(M%-1,N%)>U% THEN O%=2
2510 IF FNO(M%+1,N%)>U% THEN O%=3
2520 ENDPROC
2530 DEFFNO(X%,Y%)
2540 I%=?(Z%+X%+(Y%-5)*20)
2550 IF I%=0 THEN I%=U%
2560 =(I% AND 7)
2570 MODE7
2580 PRINT ''''"Your score was ";S%
2590 IF S%>H% THEN PRINT
     "This is the highest score so far.":H%=S%:GOTO 2610
2600 PRINT "But the high score is still at ";H%
2610 PRINT '''"Another game ?"
2620 A$=GET$
2630 IF A$="Y" THEN RUN
2640 IF A$="N" THEN 2660
2650 GOTO 2620
2660 MODE7
2670 *FX12 0
2680 *FX15 1
2690 END
2700 IF ERR=25 THEN GOTO ERL
2710 IF ERR=17 THEN 30
2720 *FX15 1
2730 *FX12 0
2740 REPORT
2750 PRINT " in line ";ERL
2760 ?216=32
2770 END
2780 MODE7
2790 PRINT ''''
2800 DEFFNI
2810 =INT((?(Z%+M%+(N%-5)*20)/7) AND 15)
2820 DEFPROCFANFARE
2830 RESTORE 2900
2840 READ P,D
2850 IF P=999 THEN L=0 ELSE L=-15
2860 IF P=-1 THEN ENDPROC
2870 SOUND 1,L,P,D
2880 SOUND 1,0,0,3
2890 GOTO 2840
2900 DATA 97,15,97,5,101,5,999,5,101,5,97,5,101,10,97,2,
     89,5,81,5,77,10
2910 DATA -1,-1
2920 DEF FNC(X%,Y%)
2930 =?(Z%+X%+(Y%-5)*20)
```

Digging Deeper into Discs

N. Fox

Much more can be made of the BBC Micro if you can afford the added extra of disc drives. Here the system is explored to help you find out what you can do with it.

The BBC Micro is one of the best personal computers available, but it becomes even better when used with a disc drive. Cassette tape leaves a lot to be desired. It takes a long, long time to load, and is sometimes difficult to transfer from one cassette recorder to another. And for all its reliability, who hasn't seen the dreaded error messages from time to time. It isn't vital, as a hobbyist, to have a dual disc unit; you only have one cassette machine, don't you? I recently bought myself the single drive and spent a few hours exploring its capabilities, with the aid of the manual which is not really up to the standard of the User Guide.

CLEAR THINKING

First let's clear up a misconception. Many people shake their heads sadly over the amount of memory taken up by the disc unit. As you know, when you add a disc drive to the BBC Micro you also have to add a disc interface. This is not just a whim on the part of Acorn, it's needed because it contains the disc operating system (strictly speaking called the Disc Filing System, or DFS in this case) which does all the things a cassette unit can do and a lot more besides.

Once you have the interface installed, when the computer is switched on it automatically starts up in disc mode. To use the cassette deck you have to type *TAPE. Only one of the operating systems, tape or disc, can be used at any one time so you don't lose any memory at all because of this. But life is never simple, and unfortunately the disc drive does in fact take up 2.75K of RAM for its own purposes — or so the manual tells us. As we shall see this is only partly true, and in many situations you can rescue most of this lost memory.

RECLOSING THE DISC

Let's start with a quick look at the disc itself. It is a standard 5¼" diameter size, and the single drive uses single-sided discs each holding 100K bytes (102400 characters). The information is held on circular tracks of which there are 40 and a little calculation shows that each track is only about ½ millimetre wide. This is one reason why discs should be treated with care and not dipped in coffee or used as ashtrays! Each track is divided into 10 parts called 'sectors', and each sector holds 256 bytes. A sector is a kind of fundamental unit; most activities seem to involve one or more whole sectors at a time, as we shall see. Things which are stored on the disk are known as 'files', whether they are BASIC programs or genuine files of data. Each one has a file-name, so your program called FRED has the file-name FRED.

The disk operating system, like the machine operating system, has a number of commands, which are really utility programs. They are listed in Table 1, and can be used alone or inside a BASIC program. All can be abbreviated to save typing and * for example, has the same effect as *CAT (which lists the catalogue of files on the current disc). With 27 different commands available what more can we ask? Well, one which isn't there is STAT which in CP/M can tell you how much empty space still exists on the disc. To find this information on the BBC machine you have to use *COMPACT which removes any free space between files, pushes it all to the end of the disc and then tells you how much there is. However, one or two niggles apart, it's a good system and certainly provides most of the things you need to get started.

LOSS OF MEMORY

Now we can get back to the memory problem. The first two sectors on the disc contain the catalogue, a list of all files present on the disc with data about their length, where they are to be loaded in memory and so on. Two sectors can only hold 512 bytes and because of this restriction only 31 file-names can be used on one disc. I don't find this a problem!

On any occasion, when the drive is first used it copies the catalogue data, two sectors of it, into the 2.75K of RAM I mentioned earlier, which starts at address &E00 (3584 decimal). This is where the tape programs normally start loading and the area from here to address &1900 (6400 decimal) is the 2.75K which is apparently lost to the user. It is in fact 11 sectors long, and since the catalogue has taken up 2 of them we still have 9 sectors left. These are mainly for holding information about any data files you may be using, and since up to 5 data files can be in use at any time the remaining 9 sectors can rapidly get used up. But... if you are not using data files at all, only reading in a BASIC program, then 0.75K (3 sectors) of the RAM is needed by the disc system. The RAM from &1100 (4352 decimal) is simply empty. If you want to use it, because your BASIC program is particularly long then type PAGE=&1100 before loading the program. This provides an extra 2K of memory!

A FILING SYSTEM

Why did Acorn call their disc operating system a Disc Filing System? Firstly, no doubt to be different, but secondly because it contains the bones from which a skeleton file handling system can be put together. It all hinges on something called a 'pointer', which is a system variable with the permanent variable name PTR#X. If this variable contains a number it represents the number of bytes from the beginning of a file, the file itself being identified by the value of X. Without going into a great amount of detail here's how it works.

As a practical example let's set up a simple file to hold a list of club members' names. Not very useful but easy to describe. Suppose we allow 18 characters per name (well, we might have a Cholmondeley-Smith!) then, as stored on a BBC disc. The file would need 20 characters per record, the two extra characters showing that the record is of string type and the length of the string. Ten records would then occupy 200 bytes, a hundred would take up 200 and so on. To locate record number 20 we set PTR#X to 400 (20 records at 20 bytes each) and it will then point to the start of this record.

To read the record using BASIC we use INPUT#X, and PRINT to show it on the screen. At the start of the program the file has to be 'opened' and identified as X, and at the end of the program it has to be closed. So the complete program to identifty and show on the screen the 20th record would look like the routine in Listing 1.

```
10  X=OPENIN("FRED")
30  PTR#X=400
40  INPUT#X,B$
50  PRINT B$
70  CLOSE#X
```

You can see that the pointer allows immediate access to any record in the file, a truly 'random-access' method in this respect. The only thing to remember is where each record starts and to make this easier it is best to make all records the same length. Also remember that you have to add 2 bytes to a field containing a string, and just to add to the confusion you must know that integer fields occupy 5 bytes and real numbers (non-integers) 6 bytes. So a typical name and address

CONTINUED OVER

record for a home filing system might look like Fig. 1.
Searching for a particular record can be done by reading the surname, say, of the first record, testing it for a match with the one you are looking for and moving the pointer to the next record if it isn't the correct one. There is more to designing a filing system than just using the pointer and it is even possible to use assembler programs for file handling. One such program is very useful since it allows any particular sector to be read from the disc into memory, modified if necessary and written back again. This could form the basis for a disc recovery program since it allows deleted files to be loaded even if the catalogue entry has gone.

FIG. 1.

Field	Characters	Field length (bytes)
Surname	18 letters	20 bytes
Initials	4 letters	6 bytes
House number	4 numbers	5 bytes
Street	18 letters	20 bytes
Town	18 letters	20 bytes
County	18 letters	20 bytes

SECTOR LOADING

The machine operating system provides a number of routines which can be 'called' from an assembler program. This one is OSWORD, which is called at location &FFF1. Before doing this the contents of the accumulator (A), the X and Y registers (X and Y) have to be set. For this program A is set to &7F, and X and Y to the low and high bytes of an area of memory which is to contain the instructions for the call to act upon. I usually use the spare space in the first page of memory starting at &70, and the information we have to store there is shown in Table 3.

PROGRAM EXAMPLE

The example program I have given takes the first sector on the first track, both start at zero, and loads it into main memory starting at address &4000. Since addresses for this purpose must be four bytes long (to allow for expansion) we have to write it as &FFFF4000, the &FFFF part signifying that the location is on the main board. The three parameters are track and sector numbers and a third meaning '1 sector of 256 bytes'. When the program is run and the appropriate track and sector numbers entered, the sector is loaded into memory and can then be examined with any dissassembler. It makes life easier if you chain the dissassembler to the short assembler program before running it.

This very brief look at the DFS shows that the BBC Micro is living up to its reputation for expansion possibilities, flexibility and above all interest for the enthusiastic amateur.

SECTOR LOADING

address	data	value
&70	drive number	zero for one drive
&71	)where the sector	zero
&72	)is to go	&40
&73	)in	&FF
&74	)memory	&FF
&75	number of parameters	three
&76	read or write?	&53(read) or &4B(write)
&77	track number	as you like
&78	sector number	as you like
&79	third parameter	&21

PROGRAM LISTING 1

```
 8 PRINT"      A disc examination program by":PRINT
10 PRINT" Norman.W.Fox.":PRINT
12 PRINT"      Enter the Track number":PRINT
14 INPUT T
18 PRINT"      Enter the Sector number":PRINT
20 INPUT S
24 ?(&70)=&00
26 ?(&71)=&00
28 ?(&72)=&40        40 ?(&78)=S          52 LDY #&00
30 ?(&73)=&FF        42 ?(&79)=&21        54 LDA #&7F
32 ?(&74)=&FF        44 DIM D 100         56 JSR &FFF1
34 ?(&75)=&03        46 P%=D              58 RTS
36 ?(&76)=&53        48 [                 60 ]
38 ?(&77)=T          50 LDX #&70          62 CALL D
```

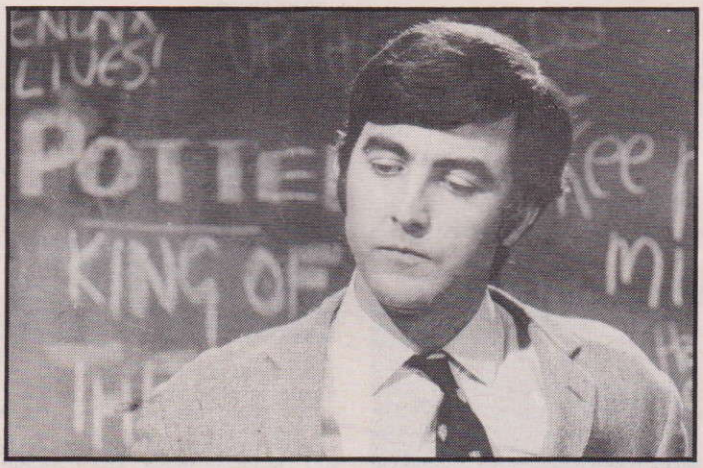
PROGRAM LISTING 2

```
>RUN
      A disc examination program by
 Norman.W.Fox.
      Enter the Track number
?0
      Enter the Sector number
?0
1AB8                 1ABE 20 F1 FF        LDY #&00
1AB8 A2 70           1AC1 60              LDA #&7F
1ABA A0 00           >                    JSR &FFF1
1ABC A9 7F           LDX #&70             RTS
```

Maths Teacher

T. Gallagher

Presenting an exciting maths tutorial package for your children to learn addition and subtraction, 'space invaders' style!



Learning maths often causes heartache and despair to young children — but those days are no more with the help of this fun to use educational program.

In 'Plus and Minus' all the numbers and '+', '−' and '=' signs float down the screen randomly and with random timing. What the player has to do is fire missiles at selected objects from a laser gun located at the side of the screen. When you have successfuly shot down a number or a sign they are collected at the bottom of the screen. You must keep capturing until 12 have been caught.

The program continues by asking the player to form an equation with the numbers and signs available, under the following rules.

1. The equation must contain an 'equals' (=) sign.
2. It may contain either a '+' or a '−' on each side of the '='.
3. Points are given for numbers or signs used in a correct equation.
4. Bonus points are given if all the collected numbers and signs are used in the equation.

The points system means that it can be more rewarding to collect a few symbols and use them all (even if it means something like 123 = 123), than to collect the full quota of 12. On the other hand, '=' is essential! Points are deducted for unused symbols!

The program uses the non-flashing colours of Mode 2, and random numbers for colour, number or sign, and timing between moves down the screen.

The missiles are defined in VDU23 statements, and VDU30 is used to return the text cursor to the top of the screen when required.

To ensure that keys pressed are not repeated, thus upsetting the shooting, the command * FX 11,0 is used.

Shooting the number wanted is not as easy as it sounds! They have a habit of drifting down the screen just when you think that the one you want has been cornered.

PROGRAM LISTING

```
  10 MODE2
  20 VDU23,224,64,32,30,31,31,30,32,64
  30 VDU23,225,0,1,30,126,126,30,1,0
  40 DIM X(100),Y(100),C(100),CH$(100)
  50 PROCARRAYS
 110 PROCTITLE
 120 CLS
 130 MODE7
 140 PROCINSTRUCT
 150 MODE2
 180 PROCSCREEN
 190 PROCWAIT
 200 *FX 11,0
 210 S=0:S$=""
 220 H=2:J=22
 230 FORI=2TOH:IFY(I)=-10RCH$(I)=""THEN380
 240   T=INT(RND(1)*5)+4
 250   FORK=1TOT
 260     COLOURC(I):PRINTTAB(X(I),Y(I));CH$(I)
 270     G=INKEY(0):IFG=70 THEN PROCSHOOT:GOTO310
 280     IFG=69THENCLS:I=H:GOTO400
 290     TIME=0
 300     REPEAT UNTIL TIME =5
 310     PRINTTAB(X(I),Y(I));" "
 320     Y(I)=Y(I)+1:COLOURC(I):PRINTTAB(X(I),Y(I));CH$(
I)
 330     IFY(I)<24THEN370ELSEY(I)=Y(I)-24:K=T
 340     PRINTTAB(X(I),24);" "
 370     NEXTK
 380   VDU30
 390   IFS=12THENI=H:GOTO 510
 400   NEXTI:IFG=69 THEN530
 500 H=H+1:IFH>100THEN510ELSE230
 510 CLS:PRINTTAB(1,1);"YOU HAVE THE MAXIMUM NUMBER OF S
YMBOLS"
 520 PRINTTAB(1,3);S$:PROCWAIT
 530 PROCEQUALS
 540 IFN$="OK"THEN600 ELSE860
 560 GOTO990
 600 MODE7
 610 PRINTTAB(1,1);"YOUR EQUATION SHOULD CONTAIN AS
   MANY OF YOUR SYMBOLS AS POSSIBLE."
 620 PRINTTAB(1,3);"FOR A ";CHR$(129)"CORRECT ";CHR$(135
)"EQUATION YOU WILL          SCORE:-"
 630 PRINTTAB(5,6);"20   FOR ONE = (ONLY 1')"
 640 PRINTTAB(5,8);"20   FOR EACH + OR -"
 650 PRINTTAB(5,10);"10   FOR EACH NUMBER USED"
 660 PRINTTAB(5,12);"5   SUBTRACTED FOR EACH UNUSED TERM
"
 670 PRINTTAB(1,20);"PRESS ANY KEY TO CONTINUE"
 680 A$=GET$:IFA$=""THEN680
 690 MODE2
 700 L=LEN(S$):K=1
 710 SUM$="................."
 720 CLS:PRINTTAB(1,2);S$
 730 PRINTTAB(1,6);SUM$
 740 PRINTTAB(1,9);"WHICH TERM FOR        SPACE ";K
 750   PRINTTAB(1,13);"TYPE X TO END":PRINT
 760   INPUTSYM$:IF SYM$="X"THEN 850
 780 I=1
 790 IF MID$(S$,I,1)=SYM$THEN820
 800 I=I+1:IFI<13THEN 790
 810 PRINT"TRY AGAIN":GOTO720
 820 SUM$=LEFT$(SUM$,K-1)+SYM$
 830 S$=LEFT$(S$,I-1)+" "+RIGHT$(S$,L-I)
 840 K=K+1:IFK<L+1 THEN 720
 850 PRINTTAB(1,6);SUM$:PROCCHECK
 860 IFN$="Y" THEN CLS:PROCARRAYS:GOTO180 ELSE 990
 990 END
1000 DEFPROCSCREEN
1030 PRINTTAB(1,28);"................."
1040 COLOUR1
1050 FORI=3TO22:PRINTTAB(1,I);CHR$(224):NEXT
1060 COLOUR3
1070 ENDPROC
1100 DEFPROCWAIT
1110 FORT=1TO3000:NEXTT
1120 ENDPROC
1200 DEFPROCSHOOT
1210 PRINTTAB(1,J);" "
```

CONTINUED OVER

PROGRAM LISTING

```
 1220 L=2:M=1:COLOUR1
 1230 PRINTTAB(L-1,J);" ";CHR$(224)
 1240 L=L+1:IFL<19THEN1230
 1250 PRINTTAB(18,J);" "
 1260 IFY(M)=J THEN1280
 1270 M=M+1:IFM<I+1THEN1260ELSE1370
 1280 IFCH$(M)=" "THEN1310
 1290 S=S+1:COLOUR C(M):PRINTTAB(S,25);CH$(M)
 1300 IFS>12THEN1390
 1310 S$=S$+CH$(M)
 1320 Y(M)=-1
 1330 M=M+1:GOTO1260
 1350 VDU30
 1360 :CH$(M)=" "
 1370 J=J-1:IFJ>2THEN1390
 1380 FORB=3TO22:COLOUR1:PRINTTAB(1,B);CHR$(224):NEXT:J=2
2
 1390 ENDPROC
 1600 DEFPROCEQUALS
 1610 I=1
 1620 F$=MID$(S$,I,1)
 1630 IF F$ ="=" THEN N$="OK":GOTO1690
 1640 I=I+1:IFI<19THEN1620
 1650 COLOUR7:PRINTTAB(1,7);"WITHOUT AN = YOU    CANNOT M
AKE AN      EQUATION!"
 1660 PRINTTAB(1,10);"SCORE 0"
 1670 PRINTTAB(1,14);"DO YOU WISH TO TRY  AGAIN?(Y/N)"
 1680 INPUTN$
 1690 ENDPROC
 1700 DEFPROCCHECK
 1710 T=LEN(SUM$):SCORE=0:S=LEN(S$)
 1720 REM AMOUNT TOBE SUBTRACTED
 1730 X=1
 1740 IF MID$(S$,X,1)=" "THEN1760
 1750  SCORE=SCORE-5
 1760 X=X+1:IFX<S+1 THEN 1740
 1770 REM CHECK LHS
 1780 L=1
 1790 IF MID$(SUM$,L,1)="="THEN1810
 1800 L=L+1:IFL<T+1 THEN 1790
 1810 EQ=L:SC=20
 1820 K=1:P1=0:M1=0:FAULT=0
 1830 IF MID$(SUM$,K,1)="+"THEN P1=K:SC=SC+20:GOTO1850

 1835  IF MID$(SUM$,K,1)="-"THEN M1=K:SC=SC+20:GOTO1850

 1840 K=K+1:IFK<EQ  THEN 1830
 1850 FORP=P1+1TO EQ-1: IF MID$(SUM$,P,1)="+"OR MID$(SUM$
,P,1)="-" THEN FAULT=1 ANDP=EQ-1:NEXTP
 1860 IF FAULT=1 THEN2040
 1865 IF M1=0AND P1=0 THEN LHS=VAL(LEFT$(SUM$,EQ-1)):GOTO
1900
 1870 IF M1>0 AND P1=0 THEN 1890
 1880 LHS=VAL(LEFT$(SUM$,P1-1))+VAL(MID$(SUM$,P1+1,EQ-1-P
1)):GOTO1900
 1890 LHS=VAL(LEFT$(SUM$,M1-1))-VAL(MID$(SUM$,M1+1,EQ-1-M
1))
 1900 REM RHS
 1910 K=EQ+1:P2=0:M2=0
 1920 IF MID$(SUM$,K,1)="+"THEN P2=K:SC=SC+20:GOTO1950

 1930 IF MID$(SUM$,K,1)="-"THEN M2=K:SC=SC+20:GOTO1950
 1940 K=K+1:IFK<T+1 THEN1920 ELSE1960
 1945 FAULT=0
 1950 FOR P=K+1 TO T: IF MID$(SUM$,P,1)="+"OR MID$(SUM$ P
,1)="-" THEN FAULT=1 AND P=T:NEXT P
 1960 IF FAULT=1THEN2040
 1970 IFM2=0 AND P2=0 THEN RHS=VAL(RIGHT$(SUM$,T-EQ)) :GOT
O2000
 1980  IFM2>0AND P2=0 THEN1990
 1985 RHS=VAL(MID$(SUM$,EQ+1,P2-EQ-1))+VAL(RIGHT$(SUM$,T-
P2)):GOTO2000
 1990 RHS=VAL(MID$(SUM$,EQ+1,M2-EQ-1))-VAL(RIGHT$(SUM$,T-
M2))
 2000 IF LHS = RHS THEN2010 ELSE 2040
 2010 IF SC=60 THEN SCORE = SCORE+(T-3)*10:GOTO2100
 2020 IF SC=40 THEN SCORE = SCORE+(T-2)*10:GOTO2100
 2030 IF SC=20 THEN SCORE = SCORE+(T-1)*10:GOTO2100
 2040 PRINTTAB(1,16);"FAULTY EQUATION"
 2050 PRINT;"SCORE 0":GOTO2200
 2100 COLOUR 12:PRINTTAB(1,17);"EQUATION CORRECT"
 2110 PRINT:PRINT:PRINT"SCORE ";SC+SCORE
 2120 BONUS=0
 2130 IFS=T AND T<=10 THEN BONUS=100:GOTO2160
 2140 IFS=T AND T<=15 THEN BONUS=200:GOTO2160
 2150 IFS=T AND T=18 THEN BONUS=300
 2160 IF BONUS=0 THEN 2180
 2170 PRINT"EXTRA ";BONUS;" POINTS FOR USING ALL TERMS"
 2180 PRINT"TOTAL SCORE ";SC+SCORE+BONUS
 2200 PRINT:PRINT"DO YOU WISH TO TRY  AGAIN?(Y/N)":INPUTN
$
 2210 COLOUR7
 2250 ENDPROC
 2300 DEFPROCTITLE
 2310 FORI=1TO80
 2320   X=INT(RND(1)*19):Y=INT(RND(1)*31)
 2330    COLOURC(I):PRINTTAB(X,Y);CH$(I)
 2340   TIME=0
 2350   REPEAT UNTIL TIME = 5
 2360   NEXTI
 2370 PRINTTAB(5,12);"          "
 2380 COLOUR11:PRINTTAB(6,13);"PLUS     "
 2385 PRINTTAB(5,14);"          "
 2390 COLOUR9:PRINTTAB(6,15);"  AND    "
 2395 PRINTTAB(5,16);"          "
 2400 COLOUR10:PRINTTAB(6,17);"    MINUS "
 2405 PRINTTAB(5,18);"          "
 2410 FORT=1TO5000:NEXT
 2420 COLOUR11:PRINTTAB(6,13);"   G.W.   "
 2430 COLOUR9:PRINTTAB(6,15);"GALLAGHER"
 2440 COLOUR10:PRINTTAB(6,17);"   1982   "
 2450 FORT=1TO5000:NEXT
 2460 ENDPROC
 2500 DEFPROCINSTRUCT
 2510 PRINTTAB(1,1);"You are required to form a balanced
   equation."
 2520 PRINTTAB(1,3);"You will first have to shoot down th
e"
 2530 PRINTTAB(1,4);"numbers and symbols you wish to use.
"
 2540 PRINTTAB(1,6);"There is a maximum number of 12 term
s."
 2550 PRINTTAB(1,8);"You may include either one + or one
-   on each side of the equal sign."
 2560 PRINTTAB(1,11);"Points are scored :-"
 2570 PRINTTAB(5,13);"20  for one = "
 2580 PRINTTAB(5,14);"20  for + or -"
 2590 PRINTTAB(5,15);"10  for each number used"
 2600 PRINTTAB(5,16);"-5 for each term collected
      and not used"
 2610 PRINTTAB(1,18);"There are extra marks for using all
 the terms in a correct equation."
 2620 PRINTTAB(1,20);"An incorrect equation scores 0 mark
s"
 2630 PRINTTAB(1,22);"PRESS THE SPACE BAR TO CONTINUE"
 2640 G=GET:IFG<>32 THEN 2640
 2650 CLS:PRINTTAB(1,1);"The lowest missile will always f
ire     first."
 2660 PRINTTAB(1,4);"There is no limit on the number of
   missiles you may fire."
 2670 PRINTTAB(1,7);"There IS a limit to the number of
   symbols which will appear."
 2680 PRINTTAB(1,12);"Press the key ";CHR$(129);"'F'"CHR$
(135)" to fire "
 2690 PRINTTAB(1,15);"Press the key ";CHR$(129)"'E'"CHR$(
135)" when you have          enough terms"
 2700 PRINTTAB(1,22);"PRESS THE SPACE BAR TO CONTINUE"
 2710 G=GET:IFG<>32 THEN 2710
 2720 ENDPROC
 2800 DEFPROCARRAYS
 2810 FORI=1TO100
 2820   X(I)=INT(RND(1)*12)+5
 2830   C(I)=INT(RND(1)*6)+1
 2840   Y(I)=0
 2850   R=INT(RND(1)*14)+1
 2860   IFR<10THENCH$(I)=STR$(R):GOTO2910
 2870   IFR=10THENCH$(I)="0":GOTO2910
 2880   IFR=11THENCH$(I)="-":GOTO2910
 2890   IFR=12THENCH$(I)="+":GOTO2910
 2900   IFR=13 OR R=14 THEN CH$(I)="="
 2910   NEXTI
 2920 ENDPROC
```

Questions & Answers

J Ruston

Each issue our resident expert, Jeremy Ruston, will answer a selection of technical queries which are causing you problems.

Q. What is the TAB key for? Mine doesn't seem to work.

A. *The TAB key is primarily designed for use under word processing programs – for example, it behaves like the tab key in a typewriter if you use View or Wordwise. In BASIC, you can test for the TAB key being pressed using GET or INKEY (it returns a code of 9), but that's about all. You can make the TAB key return any code (not just 9) by executing an *FX 219, followed by the code you want it to return. Thus, *FX 219,42 will make the TAB key return the ASCII code of an asterisk. This will make the asterisk available without resorting to the shift key. You can also make the TAB key behave like a function key. If you instruct it to return a code between 128 and 143 (inclusive) it will mimic like function key number 128-code. Thus, *FX 219,143 will make the TAB key be function key 15, which is normally buried under a cursor key. (None of this works on Operating System 0.10.)*

Q. Why doesn't the BBC Micro pump its sound effects through the TV speaker, in the same way as the Commodore VIC 20 does?

A. *A great many BBC Micros are used with monitors which don't have a speaker available, so it was necessary to use an internal speaker. I haven't heard of anyone carrying out a modification to use the TV speaker. The BBC Micro does incorporate a volume control, which can be increased as needed. This also has the effect of increasing the level of the buzz usually heard from the speaker.*

Q. I have a disc-based computer, fitted with Wordwise. This gives me a single spare ROM socket. I'm a bit worried that I'm going to run out soon. Will Acorn be selling (or at least advertising) an expander board?

A. *Acorn have been known to mention such a device, but for the moment, Watford Electronics (Cardiff Road, Watford) sell a device to extend the number of sockets to the maximum of 16. It costs £19.95. They claim RAM can be used instead of ROM, giving a 48K computer. Home users should note that this won't give more space for BASIC programs, but it will allow program developers to put things in ROM with greater ease.*

Q. How can I print on the bottom right-hand character position in MODE 3? VDU 5 doesn't seem to work properly.

A. *The easiest way to do this is to print at position (79,23) as opposed to the target location of (79,24), and then scroll the screen down a line, using 'VDU 30,11', to place the character in its final position.*

Q. When I have Shift Lock engaged on my BBC Micro, the red function does not work. Why?

A. *The function keys do not work because under Operating System 1.00 and above (one of which you obviously have) the function keys give special codes when used with the Shift key, which comes to the same thing as using Shift Lock. Thus, this is not so much a bug, more a trap for the unwary.*

Q. I have written a program for my Model B BBC Micro which I think is worth selling. The trouble is, I don't know how to see if it will run on a Model A – is there any way of turning my Model B into a Model A?

A. *If you are using the old operating system, you will have to pull out some RAM chips to achieve this – which is not a good idea if you intend to do it often. On the new (post 1.0) operating systems, typing '*FX 254,64', followed by 'CTRL-break', which gives the "BBC Computer 16K' message, will turn your Model B into something approximating a Model A. The only way to turn it back into a real Model B is to type '*FX 254,128', followed by another hard reset. Really, the advantages of doing so are pretty minimal.*

Q. Whilst doing a dump of the contents of my Operating System 1.00 EPROM, I came across references to commands called '*LINE' and '*CODE'. Trying these still gave a 'bad command' error, but they must do something. What?

A. *'*CODE' and '*LINE' are used to transfer information across the Tube. That's all you really need to know.*

Q. I have heard that my new operating system contains some new 'PLTO' options designed to fill areas with colour. Is this true, and if so, how are they used?

A. *You are correct about the new 'PLOT' options. They are numbers &48-&4F Hex and &58-&/F Hex. Watch out for an in-depth article on how to use them (which is not easy) in a future issue of* **A&B Computing.**

Q. What does the speech synthesiser for the BBC Micro consist of, and what exactly does it do?

A. *The first BBC Micro speech synthesisers are now being shipped to customers. They consist of two chips that are plugged into the large empty sockets at the left hand side of the printed circuit board. If you have an Issue 3 board (or earlier), some minor modifications need to be carried out on the board, usually by your dealer. (You can discover what issue your computer is by taking the cover off the machine and examing the board for a message, usually found in the upper right hand corner of the board). Thus, apart from the modifications, the fitting process is similar to that required to install Wordwise, or the new operating system.*

Once the chips are installed in the board, you can start making your computer speak immediately. All the different words in the standard vocabulary are assigned a different number. For example, the number 179 corresponds to the word 'COMPUTER'. To make the computer say a specific word, you simply use a statement of the form:

```
SOUND – 1,179,0,0
```

The first parameter tells the computer to use the primary vocabulary chip, while the second tells the computer to use the 179th word in the vocabulary. The last two parameters are not used at present. (The first parameter can take on other values, which are detailed in 'The Speech System User Guide', which allow speech to occur from a vocabulary held in RAM, so that the user can invent his own words).

The 'phrase number' in the standard vocabulary can fall in two ranges. Normally, numbers from 127 to 291 are used. These numbers correspond the entire vocabulary arranged in alphabetical order. Alternatively, using numbers from 32 to 126 gives words linked to their ASCII representation. For example, 42 is the ASCII code for an asterisk (or multiplication symbol). Thus,

```
SOUND –1,42,0,0
```

gives the word 'TIME' (you can add an 's' to the end to make it 'TIMES'). Parameters from 0 to 31 have odd effects. I particularly enjoy the effect of:

```
REPEAT SOUND –1,9,0,0:
SOUND –1,13,0,0: UNTIL
FALSE
```

The primary vocabulary is pretty comprehensive, but has some surprising omissions ('ME', 'MY' and 'TURN' – vital for games, I

would have thought) and some even more surprising inclusions ('ENGAGED', 'BETWEEN' etc). The size of the vocabulary can be increased by making things plural, or combining words. For example, 'IN' and 'KEY' give, not surprisingly, 'INKEY'.

Finally, the speech system is modelled on the voice of Kenneth Kendall, an ex-BBC newsreader. In practice, it just sounds like anyone else. Generally, intelligibility is good.

As a postscript, when I fitted my synthesiser, I fitted two 12 inch wires across C12. By altering the length of these wires, while they are twisted together, it is possible to alter the pitch of the voice. It doesn't quite sound like Selina Scott, but it is a start!

• For an in-depth analysis on the new speech chip turn to our special review (see 'Contents' for page number).

Q. I keep reading about various unknown *FX calls, and I am frankly annoyed that they are not described by Acorn. Will a complete list of *FX calls ever be made public?

A. *Apparently, a technical manual is under production at Acorn, but I wouldn't hold your breath. Here at A&B Computing, we are trying to put together a complete list for OS 1.20. When this mammoth task is completed, the list will be published in these pages. So try to contain yourself for a little longer.*

Q. I have heard that it is possible to tweak MODE 7 into displaying 80 character lines. Is this true?

A. *This is categorically not true – the SA5050 character generator used in MODE 7 cannot operate at a high enough speed. If you want to display 80 columns in as little memory as possible, I suggest you investigate the possibility of designing a mode with, say, 16 lines of 80 characters. This will take up 10K of RAM, compared to 16K for MODE3 and 20K for MODE0. Details on designing your own display modes are included in my book 'The BBC Micro Compedium'. I believe that 'Beebug' did a much shorter article on the subject, some months ago.*

Q. What languages besides BASIC can be run on the BBC Micro?

A. *FORTH is available from at least four sources, most of which are advertised in this magazine. I have used the Acornsoft version, but cannot comment on the others. The 'rqFORTH', from Level 9 Computing, has enthusiastic followers, mainly due to their FORTH toolkit, which seems to offer many powerful features.*

Acornsoft are selling a version of LISP (a powerful artificial intelligence language). I haven't seen it yet, but it is apparently reasonably standard. Acornsoft are also getting ready a version of BCPL (a powerful language, from which 'C' was derived) which will be in ROM form. There is a whole department of Acornsoft dedicated to producing languages, so we can expect some more over the next few months. However, their performance in supplying manuals leaves a lot to be desired.

So, there are not a great many different languages available at the moment. In a few months time, the Z80 second processor will begin to be readily available, which will allow a whole gamut of languages to be used. The price of the Z80 box (£299) includes one of several popular CP/M software packages, and several languages will be available in this way.

Q. Is it possible to swap the 2MHz 6502 inside my BBC Micro for a 4MHz version, to make things run twice as fast?

A. *I don't know if anyone has tried this, but I suspect it will not be possible, or at least, if it is possible, it will be a tricky job. For example, all the memory chips (both RAM and ROM) will have to be replaced with faster chips, which will prove expensive. On the other hand, the BBC Micro does not use software timing loops (it uses 6522 timers instead), which will make the software side easier, assuming the hardware could be arranged. Naturally, there is nothing to stop you pulling out the 2MHz chip in your computer and replacing it with a 4MHz version, but unless you carry out some other modifications, there will be no speed increase.*

Q. At my local computer club, I saw an Apple computer being demonstrated with incredibly realistic photos of a girl on the screen. I would love to be able to do the same thing on my BBC Micro, but I can't see how they could put such a photograph into a computer in the first case.

A. *These photographs are basically formed by aiming a TV camera at something, then feeding the output of the camera into a special box which converts the data into information the computer can understand. The whole box of tricks costs about £1000 (for the Apple). A similar system for the BBC Micro was demonstrated on the BBC TV series 'Making the Most of the Micro'. If you get a chance, watch the repeat in November to see it in action.*

I don't know of any similar commercial system for the BBC Micro, but I am sure that someone is working on it.

If you have a problem to do with any aspect of your BBC Micro why not jot it down and send it to Jeremy Ruston. In each issue of *A&B Computing* **he will answer a batch of letters. Send your query to:–**

Q&A
A&B Computing
145 Charing Cross Road,
London,
WC2H 0EE.

The Hobbit: Another Adventure?

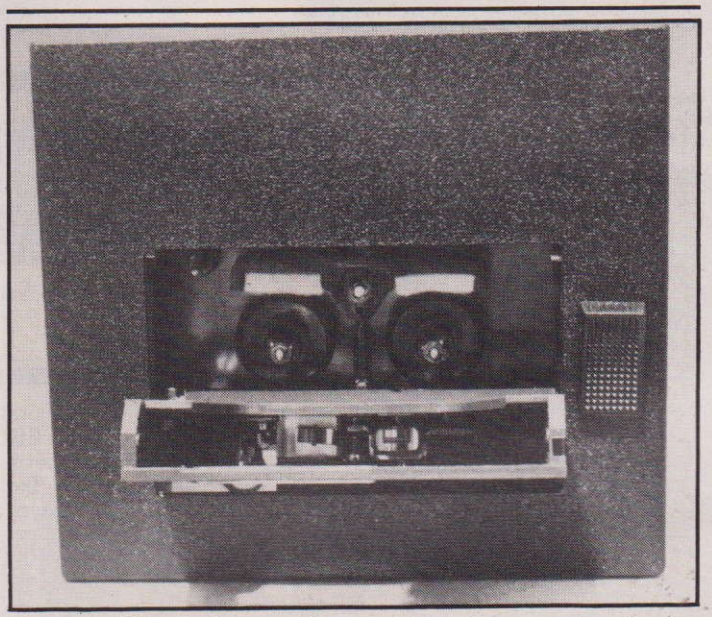
Henry Budgett

If you find tapes take too long to load on your BBC Micro and you cannot afford discs, the Hobbit drive could save you both money and time.

For the domestic BBC Micro user there must come a point when the realisation dawns that cassettes are just not going to be of use any more. The commercial user has already moved on to discs either through the BBC's own drives or with the Torch disc pack, but the home user has been content to struggle with the 1200 baud cassette system.

The problem you must face is, how can you justify buying something that costs nearly as much as the computer again? The alternative approach to discs is to go it alone and add a proprietary drive which will be cheaper but requires a good knowledge of soldering and interfacing. Something that many people will rightly be nervous of doing.

So, is there another way of adding a mass storage system onto your computer that will be faster than tape, as easy to use as discs and yet still leave you with both arms and legs attached? The answer may well lie in a device called the Hobbit from Ikon Computer Products. It's not actually a new device, it has been available for the NASCOM computer for around a year and the Philips digital cassette mechanism is even older. But this is the first time that it has been implemented with a complete operating system. And, at £135 plus VAT it appears to be ideally placed between the existing tape system and the expensive disc.



OPEN THE BOX

The Hobbit is a small black box. It comprises a black metal and plastic cube 90mm by 95mm by 110mm with the tape loading door at the front, four rubber feet at the bottom and two sockets at the back. Also included in the package are a manual, two leads and a PROM containing the device's operating system. Both cables are fairly short; indeed the data cable was somewhat shorter than the power lead, which means that the unit has to stand fairly close to the BBC Micro but that's not too much of a problem.

The standard cable set will cater for two drives which means that expansion to a dual drive system only requires the purchase of a second Hobbit, the operating system will happily handle both. The cassette supplied with our review drive was, although of exactly the same size as a dictation machine tape, of digital quality rather than audio. Given the speed that the system operates at I would highly recommend that these are used even though they do cost nearly £3 each.

ERECTING THE HOBBIT

Installing the Hobbit is no easy task. Mainly this is the fault of the manual, of which more later, rather than the basic ideas behind the system. If you have any worries about opening the box of your precious BBC Micro then you can send the whole lot to Ikon and they'll install it for a £5 note (plus VAT of course!). If you have no qualms about opening the box then the installation will take some five to ten minutes of your time, but don't rush it or you could damage your micro.

The first task is to make sure that the power is off and the plug is removed from the socket. Now remove the top cover of the BBC Micro and the keyboard assembly, a total of four screws and three nuts and bolts. You should now be able to see the ROM socket area at the front of your machine on the right-hand side. The ideal place to fit the operating system PROM is IC100 but it can go in IC101 or IC88 if required.

Plugging the PROM in is quite hard work, the sockets Acorn chose are not the nicest, so make very sure that all the pins are straight on the clip before you give it that final push home. Depending on where the PROM has been fitted you will now have to reposition one of two links and possibly cut the lead on one component. Now you can re-assemble the keyboard and case and proceed to plug the leads in.

First to go in is the data lead which fits into the user port socket underneath the micro, this should be poked into place with the clips provided on the socket. The other end of this lead goes to the socket at the back of the Hobbit. The power cable fits into the slightly precarious auxilliary power socket under the micro and care must be taken to ensure that the polarising lug on the plug is on the side nearest to the computer or spectacular damage may result. In theory it shouldn't be possible to plug it in the wrong way round but... With the power cable attached to the Hobbit you can now turn everything back on.

IS THE HOBBIT SWITCHED ON?

Instead of the familiar, friendly greeting from the BBC Micro on the screen you should now have the following:

```
BBC COMPUTER
MOS V1.2 (C) L.J.WANT & A.A.UNWIN 1982
IKON...TEL 099 421 515
BASIC
```

This indicates that all is well. If you don't get this then turn off and check everything again very carefully!

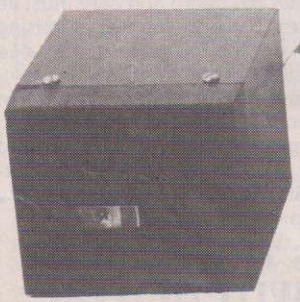
What you now have is a very fast tape system which will obey (almost) all the commands that you have previously used on the cassette tape system. However, there is one major change to be made to the way you operate. Each blank tape must first be initialised and each side of the tape needs to be given a unique name. This is achieved by using the *FORMAT FFF command which initialises the tape and gives it a volume name which can be up to eight letters specified by FFF. As a confidence check type *CAT once you've formatted a blank tape and this will show you what's on it. Don't be perturbed by noises coming from the Hobbit drive, it's under complete control of the ROM and knows what it's doing!

CONTINUED OVER

Technical Data	Ikon Hobbit
Drive	Philips digital micro cassette
Speed	750 bytes per second
Capacity	50K per side
Access time	25 seconds average
Number of files	138 total 5 active simultaneously
Buffer size	750 bytes per file 2 buffers initialised automatically
Costs	£135.00 plus VAT £17.50 plus VAT per six tapes
Supplier	Ikon Computer Products Kiln Lake Laugharne Carmarthen Dyfed SA33 4QE 099421-515

MAKING USE

The sheer speed of operation of the Hobbit should totally change the way you use your BBC Micro in that the handling of larger data files becomes a realistic operation time-wise and doesn't take all day. All the basic operating system commands you previously used on cassette are present and correct (with one exception that I'll cover in a moment) so you can SAVE and LOAD programs as normal.



However, to add to the facilities there are a number of extra functions available. The catalogue of programs stored on the tape can be accessed directly from the index display so it is possible to *CAT the tape and then simply type either the name or file number of the program you want and press f8 to LOAD or f9 to LOAD and RUN. If you want to return to the conventional cassette system then *TAPE will achieve this or you could simply type *BBC and the Hobbit Operating System (HOS for short) disappears completely. (Note: *HOBBIT will get it back after a *TAPE but only a Break will restore it from a *BBC command.) Later versions of the HOS use 0 and the Break key instead of *BBC to reset the system.

The *LOAD, *SAVE, *RUN, *SPOOL and *EXEC funtions are all implemented as usual and should need no further comment from me. However, there are a number of totally new functions available which the cassette user will not have encountered. *COPY will either duplicate a file on the same drive or copy a file onto a second drive. This is the first command that incorporates the @ Inhibit Check symbol. If you are copying a file and the HOS finds that there is already a file with that name in the catalogue it checks with you first to make sure you wish to overwrite the existing file. You can also suppress this check so it does it automatically. You can *RENAME files on the tape, *DELETE a single file or *KILL the whole tape. Once a file is deleted it isn't removed from the tape, it is merely struck out of the index list, so Ikon have thoughtfully provided a *RECOUP command to recover the last file deleted. This only works if you do it immediately, a SAVE or *SAVE will overwrite the actual file area and all will be lost.

One of the apparent quirks of the Hobbit system is that, because it works on the directory or index system, it can appear to take longer to LOAD something than it took to SAVE it or, indeed, *vice versa*. This is because it has to look up the file in the directory and then spool through the tape until it reaches the correct point. This shouldn't be much of an inconvenience as the transfer rate for information is 750 bytes per second. Compare that to the fastest BBC cassette speed of 120 bytes per second and you'll see why Ikon claim that it is halfway between tape and disc in operating speed. Each tape can hold around 100K of information with a maximum of 138 files.

It is in the area of files, specifically data files, that the small differences between the HOS and the BBC's Cassette Operating System or COS occur. The commands affected are the OPENIN and OPENOUT file operators. Under the HOS they have the same function and need to be qualified with one of the four control codes provided. The default values for OPENIN and OPENOUT are different in that the first gives access for both reading and writing and the latter only provides write access to the file. However, the OPENOUT command can be restricted to either read or write only by using the W or R control code.

The major problem is that while the Hobbit can handle up to five files simultaneously it only initialises sufficient buffer space for two. Yes, even the Hobbit eats more of your precious RAM and as anyone who has read Tolkein's tale knows they like regular feeding! Each buffer must be 750 bytes so the inital memory loss is some 1.5K, which is less than discs would take but still painfully expensive for the user. With all five file buffers initialised (the manual tells you how), you've lost nearly 4K of user memory so it is well worth making sure that you are doing things as efficiently as possible when dealing with data files.

MOANS ABOUT THE MANUAL

That was the good news, now for the bad. The manual is a poor thing indeed. Currently standing at some 16 pages of poorly laid out and, even worse, badly thought out A5 sheets, it doesn't do the Hobbit justice at all. The instructions for installation are not good, it doesn't even appear to warn you to turn the BBC Micro off before you start, and the diagram showing the location of the various components is a hand drawn scrawl.

A re-write and re-packaging

operation is called for here, decent examples should be included and it would be helpful if someone checked the spelling too. It would also be nice if Ikon could see their way to including some of the routines for data file handling on a tape and supplying this with the rest of the package because this area is the least well explained of the lot and yet will probably be the reason many home users buy the system.

TECHNICAL FACTS

The idea of using a very fast digital cassette for personal computers is not exactly new. Indeed, the Philips drive that the Hobbit uses has been around for three years now but has seldom been implemented. Ikon themselves produced a naked module for the NASCOM range of computers last year which seemed to gain some acceptance, but everyone else carried on with the domestic cassette. This year, however, we have seen the Epson HX-20 and the Sharp PC-1251 making use of the micro cassette and there are others to come too.

The advantages to be gained are enormous, in terms of speed of operation, physical size and the power of the operating system that can be provided — it has no match in tape based form. Hybrid devices such as Sinclair's long awaited Microdrive and the BATS/NCI 3" disc have yet to be proven in terms of reliability and don't exist in sufficient quantity.

The question that must be asked is whether Ikon have managed to link the micro cassette to the BBC Micro in the best possible way. There are two areas where they appear to have run into problems and, sadly, not actually told anyone about it. Because the tape needs a timing signal written onto it to enable it to find any program or data file Ikon 'borrow' the BBC's timer and corrupt its contents. It should have been possible to use the system clock and to generate a timing signal within the Hobbit rather than to mess up a system variable.

The other reported area of complaint is that the Control B copy facility fails to operate as normal when the Hobbit is installed. I couldn't establish the truth of this claim as at the time of the review our printer was taking its annual holiday and Ikon reckon that it doesn't happen on their serial printer anyway. If it is correct that this facility is disabled then it would be kind of Ikon to warn us and explain how to get around it with the VDU2 and VDU3 commands, see pages 404-408 of the **User Guide**. Apart from these niggles they appear to have provided as good an implementation of the unit as possible.

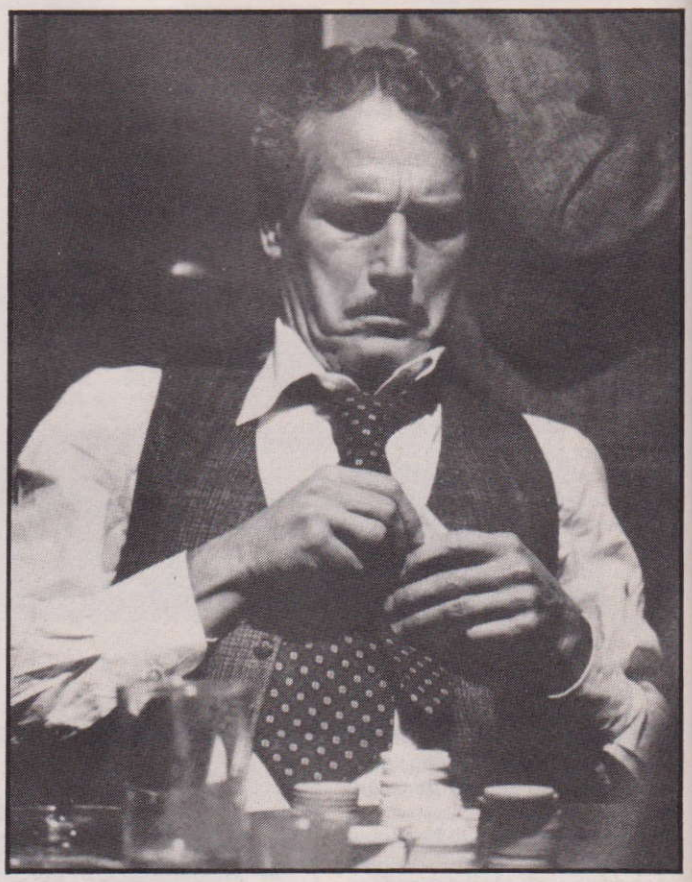
LAST WORDS

Given that your usage of the BBC Micro is being limited by the amount you can store and access within a reasonable time and that you either cannot afford discs or simply don't think that you really need them, the Hobbit is probably your only real alternative. It really is halfway between the two in terms of price, speed and capabilities.

My only regrets are that the appearance and usability of the excellent Philips drive are let down by a shoddy casing, a poor manual and a generally unprofessional appearance to the whole thing. Given a nice case, a better manual and an overall uplift in quality it deserves to succeed as the price is right and the HOS flexible enough to satisfy the market for which it has been designed. Ikon could, given a little attention to detail, have a real winner on their hands provided, of course, someone else doesn't beat them to it with a 3.5" Sony-type drive! With technology progressing as quickly as it currently is, that's quite possible.

Card Sharper

Freda Perrow



Here are two card playing games for the price of one to run on your BBC Micro.

If you fancy yourself as something of a card sharp, then here is a tremendous double bill for you and your BBC Micro. The first program is based on the old favourite, Pairs, in which you have to match up two cards. However, there's an exciting difference to this version which you'll soon find. The other program has been developed from the game of Snap and includes a number of different shaped space vessels! So, pack away your deck of cards and get typing.

PAIRS

Pairs is a version of a card game which has been modified for the BBC Micro and at the same time has had a bit of spice added to it by introducing a little flutter between each round.

Instructions are briefly given at the start of the program but if you wish to save a bit of typing these can be omitted by removing 'Mode 7: Procheader' from line 50 and by deleting lines 1170 onwards. A more detailed explanation is given here.

IN CREDIT

The player is given a starting credit of £25 and 48 cards are dealt out in four rows of 12. Odds are selected at random and fall between two to one and seven to one. These can be altered if required by changing line 850. At the start of the game the player stakes a bet of between £1 and £5. Because at this stage, you have no idea of the layout of the cards it is advisable to place a low bet. Next, the first card is selected by typing in the required card's position. For example, the top right-hand card would be L1, whilst the bottom left-hand card is A4.

The selected card is then displayed. At this stage the player may be given the option of doubling or trebling the stakes. If you have this opportunity, only use it if you think you may know the position of a matching card.

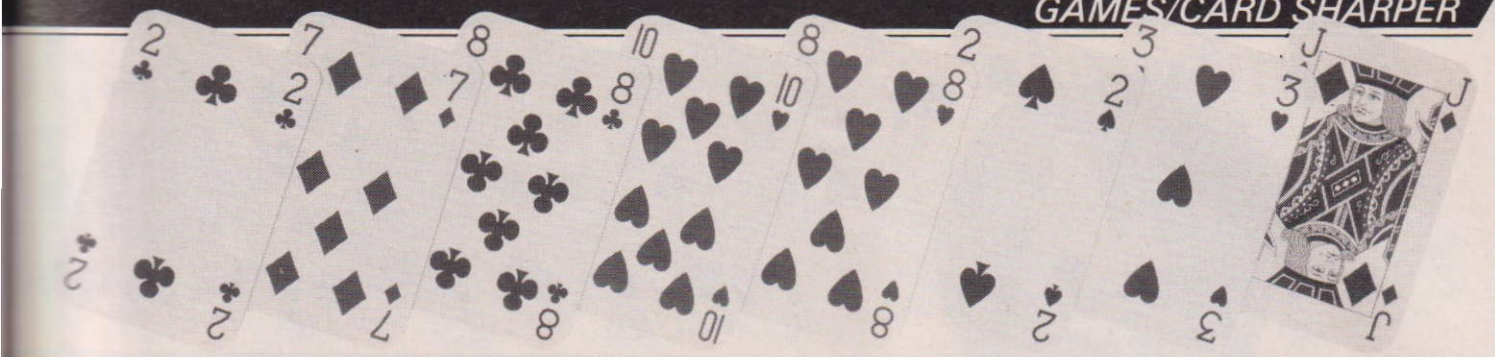
If the two selections are the same your winnings are added to your credit and the two cards are removed from the screen. If the two cards don't pair up they are simply turned face down again and the game continues. The game is completed when all cards have been paired up or you go bust. Try to accumulate as much cash before all the cards are gone.

IN COMMAND

The program follows the standard BBC Micro's layout with the main body contained up to line 150 and then using procedures to perform the necessary tasks.

The '*TV0,1' command turns off the interlace and helps to produce a steady picture and the '*TV255,1' drops the screen format by one line. Both these commands only take effect after a mode change. Note also that '*TV0,1' has no effect in Mode 7.

A point worth remembering about the '*FX' and 'TV' commands is that anything after them on a line is ignored so you can only use one per line. If other instruc-



tions are on the line you must place the '*FX' commands last. If a line contained '*TV0,1: *TV255,1' on the same line then only the first command would be implemented.

LINE DESCRIPTION

Lines	Description
50-150	Main body of the program.
50	Seeds random number generator and turns off interlace.
60	Drops screen format by one line.
70	Selects graphics mode, turns off cursor and sets text window.
80	Changes palette colours.
90	Dimension arrays.
100	Initial setting of variables.
110	Plays game until all cards are gone or player goes bust.
170-210	DEFPROCSETUP.
180	Combines text and graphics cursors.
190	Sets out letter and number matrix for card positions.
200	Places cards on the screen.
230-280	DEFPROCCARD. Select colour of card; if F=3 card is cleared from the screen, if F=1 then white card is printed and if F=0 then card is printed face down.
300-330	DEFPROCSHOW(j)
310	Joins cursors and determines colour of printed card.
320	Prints face on card.
330	Separates cursors.
350-460	DEFPROCSHUFFLE.
360	Sets up local variables.
380	Reads data.
390-440	Shuffle cards.
450	Sets up pattern on backs of cards.
500-540	DEFPROCLOOK Select which card is turned over for examination.
560-740	DEFPROCEXAMINE (main action proceedure).
570	Displays initial credit.
600	If card already selected or gone asks for selection.
610	Draws and shows card.
620	Checks to see if stake can be altered.
630-660	If it can, then carries out alteration.
670	Displays new credit and asks for next selection.
680-720	Repeat action as for first selection.
730	Checks for like cards and adjusts variable as required.
760-780	DEFPROCCLEAR. Turn cards back to face down or removes them if F=3.
800-820	DEFPROCWAIT(t). Delay loop.
840-900	DEFPROCPLAY.
850	Selects odds.
860-880	Print out playing details.
890	Calculates credit after stake.
960-1060	Various sound effects — the procedures are self explanatory.
1080-1160	Finishing mesages.
1180-1230	DEFPROCHEADER Starting instructions.

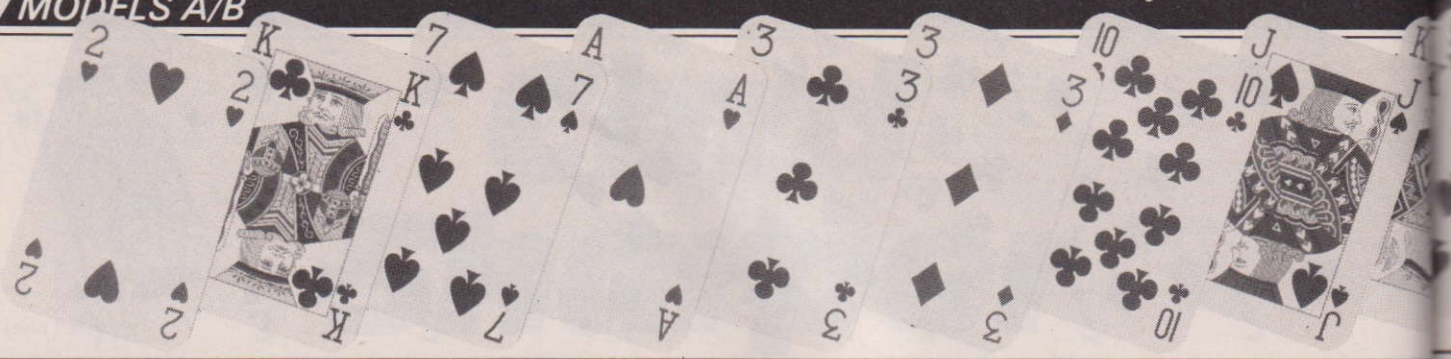
VARIABLES USED

F	Determines how the card is drawn: F=0 — card face down; F=1 — card face up; and F=3 — removes card.
GM	Number of cards paired.
CL	Credit left.
BS	Information of card character.
AS	Information of card character colour.
A,S	
N$(12)	
S$(12)	Arrays used in the card shuffling sequence.
C$	
SP$	
X1,Y1	Temporary variables stored because X and Y alter but old values are required again at a later stage.
BACK$	Used for production of pattern on back of cards. Pattern if made up from three CHR$ 37 characters. These are the '%' symbol.

PROGRAM LISTING: PAIRS

```
   10 REM ********PAIRS*******
   20 REM ****FREDA PERROW****
   30 REM *TERRICOMS SOFTWARE*
   40 REM ***APRIL 1983 (C)***
   50 X=RND(-TIME):MODE7:PROCHEADER:*TV0,1
   60 *TV255,1
   70 MODE5:VDU23,10,32,0,0,0,28,0,31,19,20
   80 VDU19,128,4;0;19,2,0;0;
   90 DIMA(48),A$(48),B$(48),S(48),N$(12),S$(4),C$(4,12),
SP$(48)
  100      F=0:CM=0:CL=25:PROCSHUFFLE:PROCSETUP
  110 REPEAT:PROCPLAY:PROCWAIT(2):PROCCLEAR:UNTIL CM=48 O
R CL<1
  120 IFCM=48THEN PROCGOOD ELSE PROCBAD
  130 PRINT''TAB(3)"PRESS KEY 'Y'""  FOR ANOTHER GAME"
  140 A$=GET$:IF A$="Y" THEN RUN
  150 CLS:COLOUR129:CLG:MODE7:END
  160 :
  170 DEFPROCSETUP
  180 VDU5:GCOL0,3:C=1
  190 FORY=983TO448STEP-145:MOVE640,Y:PRINTC:C=C+1:NEXT:C
=65:FORX=50TO1150STEP100:MOVEX,448:PRINTCHR$C;:C=C+1:NEXT
:VDU4
  200 FORY=923TO488STEP-145:FORX=50TO1150STEP100:PROCCARD
    :NEXT,
  210 ENDPROC
  220 :
  230 DEFPROCCARD
  240 IF F=3 THEN GCOL0,0 ELSE GCOL0,3
  250 MOVEX,Y:DRAWX,Y+90:DRAWX+55,Y+90:PLOT85,X,Y
  260 DRAWX+55,Y:DRAWX,Y:PLOT85,X+55,Y+90:IFF=1THENENDPRO
C
  270 VDU5:IF F=3 GCOL0,3:MOVEX,Y+60:PRINT":"ELSE GCOL0,1
:MOVEX,Y+25:PRINTBACK$
  280 VDU4:ENDPROC
  290 :
  300 DEFPROCSHOW(J)
  310 VDU5:IFA$(J)="R"THENGCOL0,1ELSEGCOL0,2
  320 MOVEX,Y+50:PRINTB$(J)
  330 VDU4:ENDPROC
  340 :
  350 DEFPROCSHUFFLE
  360 LOCALS%,T%,U%:S%=1
  370 PRINTTAB(0,4)"PLEASE WAIT  WHILE I""SHUFFLE AND D
EAL THE""TAB(8)"CARDS"
  380 FORT%=1TO 4:READS$(T%):NEXT:FORT%=1TO12:READN$(T%):
NEXT
  390 FORU%=1TO 4:FORT%=1TO12:C$(U%,T%)=N$(T%)+S$(S%)
  400 NEXT:S%=S%+1:NEXT
  410 FORT%=1TO48
  420 U%=RND(12):S%=RND(4):IF C$(S%,U%)=""THEN420
  430 SP$(T%)=C$(S%,U%):C$(S%,U%)=""
  440 B$(T%)=LEFT$(SP$(T%),1):A$(T%)=RIGHT$(SP$(T%),1):NE
XT
  450 BACK$=CHR$37+CHR$11+CHR$8+CHR$37+CHR$11+CHR$8+CHR$3
7
```

CONTINUED OVER



PROGRAM LISTING

```
  460 COLOUR129:ENDPROC
  470 :
  480 DATA R,B,R,B,A,2,3,4,5,6,7,8,9,J,Q,K
  490 :
  500 DEFPROCLOOK
  510 X=GET:IFX<650R X>76 SOUND1,-15,100,5:GOTO510ELSE SO
UND1,-15,180,3
  520 Y=GET:IFY<490R Y>52 SOUND1,-15,100,5:GOTO510ELSE SO
UND1,-15,180,3
  530 X=X-64:Y=Y-48:CARD=(Y-1)*12+X:X=X*100-50:Y=1068-(Y*
145)
  540 ENDPROC
  550 :
  560 DEFPROCEXAMINE
  570 PROCCREDIT:PRINTTAB(0,8)"SELECT FIRST CARD"
  580 PROCLOOK
  590 X1=X:Y1=Y:CARD1=CARD
  600 IF A(CARD1)=1 THEN VDU7:GOTO580
  610 PROCCARD:PROCSHOW(CARD):GAMBLE=RND(4)
  620 IF GAMBLE=2 AND STAKE*2<CL THEN PRINTTAB(0,8)"DOUBL
E YOUR STAKES?":GOTO640
  630 IF GAMBLE=3 AND STAKE*3<CL THEN PRINTTAB(0,8)"TREBL
E YOUR STAKES?"ELSE 670
  640 PROCCHANGE:A$=GET$:IFA$<>"Y" AND A$<>"N" THEN 640
  650 IF A$="Y" AND GAMBLE=2 THEN STAKE=STAKE*2:PRINTTAB(
0,5)"NEW  STAKE.....";STAKE:CL=CL-(STAKE/2)
  660 IF A$="Y" AND GAMBLE=3 THEN STAKE=STAKE*3:PRINTTAB(
0,5)"NEW  STAKE.....";STAKE:CL=CL-(STAKE/1.5)
  670 PROCCREDIT:PRINTTAB(0,8)"SELECT SECOND CARD "
  680 PROCLOOK
  690 X2=X:Y2=Y:CARD2=CARD
  700 IFX2=X1 AND Y2=Y1 THEN VDU7:GOTO680
  710 IF A(CARD2)=1 THEN VDU7:GOTO680
  720 PROCCARD:PROCSHOW(CARD)
  730 IF B$(CARD1)=B$(CARD2)PROCWON:F=3:CM=CM+2:STAKE=ODD
S*STAKE+STAKE:CL=CL+STAKE:A(CARD1)=1:A(CARD2)=1 ELSE PROC
FAIL:F=0
  740 ENDPROC
  750 :
  760 DEFPROCCLEAR
  770 X=X1:Y=Y1:PROCCARD:X=X2:Y=Y2:PROCCARD
  780 ENDPROC
  790 :
  800 DEFPROCWAIT(T)
  810 LOCALZ:Z=TIME:REPEAT UNTIL TIME-Z>T*100
  820 ENDPROC
  830 :
  840 DEFPROCPLAY
  850 ODDS=RND(6)+1
  860 CLS:PRINT"ODDS OFFERED ARE ";ODDS;":1":PROCCREDIT
  870 PRINTTAB(0,5)"YOUR STAKE.....";
  880 STAKE=GET:IF STAKE>CL+48 OR STAKE<49 OR STAKE>53THE
N 860 ELSE STAKE=STAKE-48:PRINT;STAKE:VDU7
  890 CL=CL-STAKE:F=1:PROCEXAMINE
  900 ENDPROC
  910 :
  920 DEFPROCCREDIT
  930 PRINTTAB(16,3)SPC(4)TAB(0,3)"YOUR CREDIT IS ";CL
  940 ENDPROC
  950 :
  960 DEFPROCWON
  970 FORS=50TO200STEP10:SOUND1,-10,S,1:SOUND2,-10,S+25,1
:NEXT
  980 ENDPROC
  990 :
 1000 DEFPROCFAIL
 1010 FORS=50TO20STEP-5:SOUND1,-10,S,5:SOUND2,-10,S-15,5:
NEXT
 1020 ENDPROC
 1030 :
 1040 DEFPROCCHANGE
 1050 FORL=1TO3:FOR S=100TO190STEP30:SOUND1,-15,S,2:NEXT,
 1060 ENDPROC
 1070 :
 1080 DEFPROCGOOD
 1090 CLS:PRINTTAB(4,1)"GREAT STUFF!!"''
 1100 PRINT" YOU'VE SCOOPED ";CL
 1110 ENDPROC
 1120 :
 1130 DEFPROCBAD
 1140 CLS:PRINTTAB(2,1)"SUCKER!! YOU'VE"'
 1150 PRINT"  LOST EVERYTHING"
 1160 ENDPROC
 1170 :
 1180 DEFPROCHEADER
 1190 PRINTTAB(11,1)CHR$130;CHR$141;"<< PAIRS >>"TAB(11,2
)CHR$130;CHR$141;"<< PAIRS >>"
 1200 PRINTTAB(2,4)"PAIRS COMBINES BOTH MEMORY AND SKILL"
'" WITH  A  CERTAIN AMOUNT OF  LUCK. YOU'" START  WITH
ONLY `25  OF  CREDIT. THE"'" IDEA IS TO PLACE  YOUR BET AN
D TRY TO"'" MATCH UP PAIRS OF CARDS."'
 1210 PRINT"CARD SELECTION IS BY LETTER  AND NUMBER"'"FOR
 EXAMPLE 'B3'.  AFTER YOUR FIRST PICK"'"YOU MAY FIND YOU H
AVE A CHANCE TO ALTER"'"YOUR STAKES.  YOUR INITIAL STAKE M
UST BE"'"1  TO  `5.  YOUR AIM IS TO TRY TO CLEAR"
 1220 PRINT"THE DECK OF CARDS  AND AT THE SAME TIME"'TAB(
2)"ACCUMULATE AS MUCH CASH AS YOU CAN"'''TAB(13)"GOOD LUC
K"'''CHR$136;CHR$131"PRESS THE SPACE-BAR TO START GAME."
 1230 A=GET:ENDPROC
```

SPACE MATCH

Space match is an updated version of the timeless game of Snap. The idea of producing a simple game of Snap was nearly thrown out before it got passed the doodling stages, but encouraged by my two young children who always seemed to be playing it with cards I persevered and came up with the following program. So, if you have young children, grandchildren or are simply very young at heart, give it a try — my kids don't use cards now!

There is nothing too spectacular about the program, but it can be easily altered to suit individual tastes. For example, I have only used five different shapes of space vessels but you could have more shapes and change the concept completely by designing your own ideas into lines 1170 to 1370. If you use more than five shapes, don't forget that your chances of 'snap' will be reduced.

In this basic program, I have not included a shuffle routine because I thought that simple random selection of numbers was quite adequate. However, I have included line 50 to 'seed' the random number generator to prevent the same sequence of random numbers occuring if the game was played from a 'cold' start each time.

SPACE STRUCTURE

50-180	Main body of the program.
50	Seeds random number generator.
60	Sets mode and turns off auto repeat of keys.
70	Turns off flashing cursor.
80	Displays instructions.
90	Selects graphics mode.
100	Alters colour pallette.
110	Sets up screen and deals out cards.
120	Initial setting of variables.
130-150	Main loop — repeats until player has no cards left.
160	Displays finishing message.
200-210	Function calculations for back of cards.
230-270	DEFPROCCARD(X,Y). Draw a card at the position specified by X and Y.
290-330	DEFPROCLAYOUT. Print out information required during the game.
350-400	DEFPROCDEAL. Just for show really! Deals out 26 cards to each player.
420-460	DEFPROCPLACE.

430 Checks for cheating and determines what shape is printed.
450 Picks a random colour and prints selected shape.
480-530 DEFPROCLHP (left-hand player).
490 Determines a random card for playing.
500 Checks if player has a card left to play.
510 If you have, you play one from your pile.
520 and place it on the table, also, checks for snap.
550-600 DEFPROCRHP (right-hand player).
As lines 480-530 but for right-hand player.
620-660 DEFPROCSCORE.
Print number of cards each player has left.
680-730 DEFPROCPLAY.
Print number of cards each player has left.
690 Indicates when to play a card.
700 If AD = 1 then auto dealing has been selected and cards will be played at random intervals.
710 If manual dealing has been chosen, the micro waits for space bar to be pressed before playing.
750-800 DEFPROCCHEAT.
760 The left-hand checks if player has pressed the 'snap' key out of turn and if so deducts one card from that pile..
770 As 760 but for the right-hand player.
820-870 DEFPROCSNAP.
830 Checks for cheating again.
850-860 Detect which player was quickest on the button and then proceeds to that players 'PROCWIN'.
890-950 DEFPROCLHW (left-hand win).
900 Removes played cards.
910 Adjusts score.
920-930 Place cards on the left-hand player's pile.
970-1030 DEFPROCWAIT(s).
Delay loop.
1090-1150 DEFPROCDONE.
Game over message.
1170-1240 DEFPROCINIT.
Set up user defined characters.
1260-1370 DEFPROCSELECT.
Create the shapes of the space vessels to be printed on the played cards.
1390-1490 DEFPROCINST.
Display initial instructions.

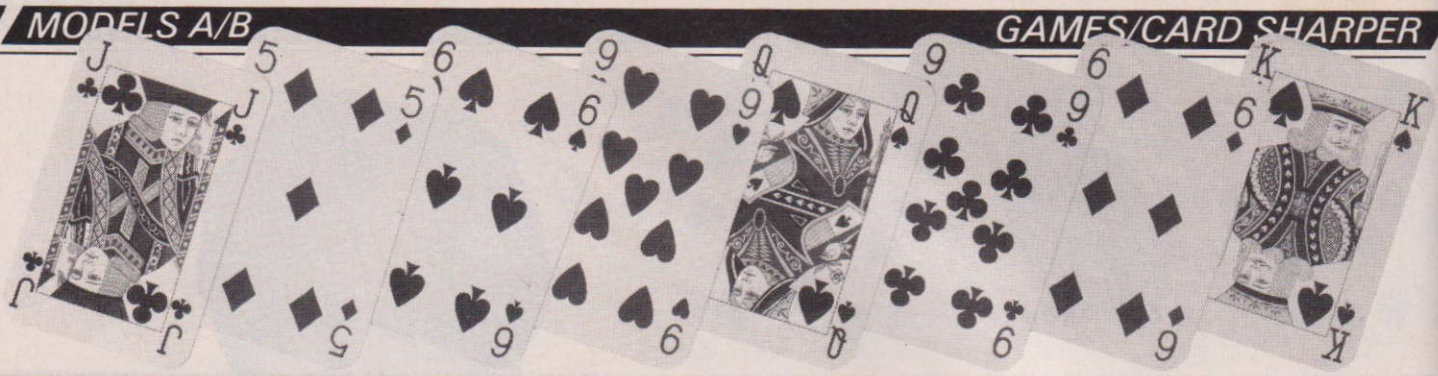
VARIABLES USED

Pick	Shape printed on played cards.
F	Determines what type of card is printed (face up or down).
C	Cheat indicator.
CLL	Cards left — left-hand player.
CLR	Cards left — right-hand player.
AD	Automatic dealing flag.
P	Playing order flag.
SH	Shape of space vessel to be printed.

PROGRAM LISTINGS: SPACE-MATCH

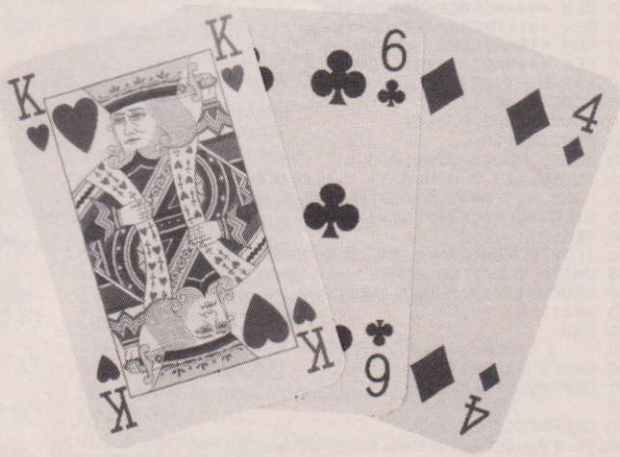
```
10 REM *****SPACE-MATCH****
20 REM ****FREDA PERROW****
30 REM *TERRICOMS SOFTWARE*
40 REM ***MARCH 1983 (C)***
50 X=RND(-TIME)
60 MODE 7:*FX11 0
70 VDU23;8202;0;0;0;
80 PROCINST
90 MODE5:VDU23;8202;0;0;0;
100 VDU19,128,2,0,0,0,19,2,0,0,0,0
110 PROCINIT:PROCLAYOUT:PROCDEAL
120 PICK1=6:PICK2=7:C1=0:C2=0
130 REPEAT
140 IF P=0 PROCLHP ELSE PROCRHP
150 UNTIL CLL<1 OR CLR<1
160 PROCWAIT(1):MODE7:PROCDONE
170 *FX12 0
180 END
190 :
200 DEFFNH=250-Z*30
210 DEFFNV=350-Z*40
220 :
230 DEFPROCCARD(X,Y)
240 IF F=3 GCOL0,0:F=1 ELSE GCOL0,3
250 VDU25,4,X;Y;25,5,X;Y+350;25,5,X+250;Y+350;25,85,X;Y
;25,5,X+250;Y;25,5,X;Y;25,85,X+250;Y+350;:IFF=1THENENDPRO
C
260 GCOL0,2:FORZ=0TO4:SOUND1,-10,RND(100)+130,RND(2):VD
U25,4,X;Y+Z*40;25,1,FNH;FNV;25,4,X+Z*30;Y;25,1,FNH;FNV;25
,4,X;Y+FNV;25,1,FNH;-FNV;25,4,X+Z*30;Y+350;25,1,FNH;-FNV;
:NEXT
270 ENDPROC
280 :
290 DEFPROCLAYOUT:VDU5
300 MOVE0,540:PRINT"PLAYER":MOVE100,480:PRINT"ONE"
310 MOVE900,530:PRINT"PLAYER":MOVE1020,470:PRINT"TWO"
320 MOVE340,350:PRINT"CARDS LEFT"
330 ENDPROC
340 :
350 DEFPROCDEAL
360 F=0:FOR C%=1 TO 26 STEP 2
370 PROCCARD(30,20):PROCCARD(1000,20)
380 CLL=C%+1:CLR=C%+1
390 VDU4:PRINTTAB(5,23);CLL:PRINTTAB(13,23);CLR:VDU5:NE
XT
400 ENDPROC
```

CONTINUED OVER



PROGRAM LISTING

```
  410 :
  420 DEFPROCPLACE(X,Y,P)
  430 PROCCHEAT:PROCSELECT:VDU5
  440 FOR A=130 TO 300 STEP 170
  450 GCOL0,RND(3)-1:MOVE X+40,Y+A:PRINT;SH$:NEXT:VDU4
  460 ENDPROC
  470 :
  480 DEFPROCLHP
  490 PICK1=RND(5):SH=PICK1:PROCPLAY
  500 IF CLL<2 THEN F=3 ELSE F=0
  510 PROCCARD(30,20):F=0:CLL=CLL-1:PROCSCORE:PROCCARD(35
0,650)
  520 F=1:PROCCARD(350,650):PROCPLACE(350,650,0):P=1:PROC
SNAP
  530 ENDPROC
  540 :
  550 DEFPROCRHP
  560 PICK2=RND(5):SH=PICK2:PROCPLAY
  570 IF CLR<2 THEN F=3 ELSE F=0
  580 PROCCARD(1000,20):F=0:CLR=CLR-1:PROCSCORE:PROCCARD(
650,650)
  590 F=1:PROCCARD(650,650):PROCPLACE(650,650,1):P=0:PROC
SNAP
  600 ENDPROC
  610 :
  620 DEFPROCSCORE
  630 VDU4
  640 PRINTTAB(5,23);SPC(2):PRINTTAB(13,23);SPC(2)
  650 PRINTTAB(5,23);CLL:PRINTTAB(13,23);CLR :VDU5
  660 ENDPROC
  670 :
  680 DEFPROCPLAY
  690 VDU4:PRINTTAB(8,30)"PLAY"
  700 IF AD=1 THEN PROCWAIT(1-RND(1)):GOTO 720
  710 IF INKEY(-99) THEN 720 ELSE 710
  720 PRINTTAB(6,30);SPC(7):VDU5
  730 ENDPROC
  740 :
  750 DEFPROCCHEAT
  760 IF INKEY(-98) THEN VDU4:SOUND1,-8,25,2:PRINTTAB(6,3
0)"<CHEAT":C1=1:GOTO760
  770 IF INKEY(-105)THEN VDU4:SOUND1,-8,25,2:PRINTTAB(6,3
0)" CHEAT>":C2=1:GOTO760
  780 PRINTTAB(6,30);SPC(7):VDU5
  790 CLL=CLL-C1:CLR=CLR-C2:C1=0:C2=0
  800 ENDPROC
  810 :
  820 DEFPROCSNAP
  830 PROCCHEAT:IF PICK1<>PICK2 ENDPROC
  840 IF PICK1<>PICK2 ENDPROC
  850 IF INKEY(-98) THEN PROCLHW:ENDPROC
  860 IF INKEY(-105) THEN PROCRHW:ENDPROC
  870 GOTO 850:ENDPROC
  880 :
  890 DEFPROCLHW
  900 F=3:PROCCARD(350,650):F=3:PROCCARD(650,650)
  910 FOR S=80 TO 180 STEP10:SOUND1,-10,S,1:NEXT:PROCWAIT
(0.5)
  920 CLL=CLL+(52-CLL-CLR):P=0
  930 F=0:PROCCARD(30,20)
  940 PROCSCORE:PICK1=6:PICK2=7
  950 ENDPROC
  960 :
  970 DEFPROCRHW
  980 F=3:PROCCARD(350,650):F=3:PROCCARD(650,650)
  990 FOR S=80 TO 180 STEP10:SOUND1,-10,S,1:NEXT:PROCWAIT
(0.5)
 1000 CLR=CLR+(52-CLL-CLR):P=1
 1010 F=0:PROCCARD(1000,20)
 1020 PROCSCORE:PICK1=6:PICK2=7
 1030 ENDPROC
 1040 :
 1050 DEFPROCWAIT(S)
 1060 W=TIME:REPEAT UNTIL TIME-W>S*100
 1070 ENDPROC
 1080 :
 1090 DEFPROCDONE
 1100 *FX 15 0
 1110 IF CLL<CLR I=30:L=0 ELSE I=0:L=30
 1120 FOR R=10 TO 20:PRINTTAB(I,R)CHR$(135+RND(2));CHR$(1
28+RND(8));"I WIN!":PRINTTAB(L,R);CHR$136;"I LOSE!":NEXT
 1130PRINTTAB(3,22)CHR$134;"PRESS RED KEY F0 FOR ANOTHER
GO"
 1140 *KEY0 RUN|M
 1150 ENDPROC
 1160 :
 1170 DEFPROCINIT
 1180 VDU23,224,1,3,7,15,31,63,127,255
 1190 VDU23,225,255,255,255,126,126,255,255,255
 1200 VDU23,226,128,192,224,240,248,252,254,255
 1210 VDU23,227,255,165,165,165,165,165,165,255
 1220 VDU23,228,255,127,63,31,15,7,3,1
 1230 VDU23,229,255,254,252,248,240,224,192,128
 1240 ENDPROC
 1250 :
 1260 DEFPROCSELECT
 1270 F$=STRING$(3,CHR$8):F$=CHR$10+F$
 1280 G$=CHR$224+CHR$225+CHR$226
 1290 H$=CHR$228+CHR$225+CHR$229
 1300 I$=CHR$227+CHR$227+CHR$227
 1310 J$=CHR$9+CHR$227+CHR$9
 1320 IF SH=1 SH$=G$+F$+I$+F$+CHR$228+CHR$225+CHR$229
 1330 IF SH=2 SH$=J$+F$+G$+F$+CHR$225+CHR$225+CHR$225
 1340 IF SH=3 SH$=G$+F$+H$+F$+1$
 1350 IF SH=4 SH$=H$+F$+I$+F$+G$
 1360 IF SH=5 SH$=J$+F$+CHR$9+CHR$225+CHR$9+F$+G$
 1370 ENDPROC
 1380 :
 1390 DEFPROCINST
 1400 CLS:PRINTTAB(0,1)CHR$141;"T E R R I C O M S  S P A
C E M A T C H":PRINTTAB(0,2)CHR$141;"T E R R I C O M S  S
 P A C E M A T C H"
 1410 PRINTTAB(0,4)CHR$131;"WELCOME TO THE TIMELESS GAME
OF SNAP. "'CHR$131;"IN THIS VERSION OF THE GAME THERE ARE"
'CHR$131;"FIVE DIFFERENT CARDS. SAME SHAPES BUT"'CHR$131;
"DIFFERENT COLOURS ALSO COUNT AS 'SNAP'"'
 1420 PRINTCHR$134;"THE 'SNAP' KEYS ARE 'Z' FOR THE LEFT"
'TAB(20)CHR$134;"'?' FOR THE RIGHT"'
 1430 PRINTCHR$131;"EACH PLAYER IS DEALT  26  CARDS."'CHR
$131;"TWO AT A TIME - IT'S MAGIC FOLKS!"''CHR$134;"A GAME
 ENDS WHEN A PLAYER HAS NO MORE"'CHR$134;"CARDS LEFT. DON'
T CHEAT YOU LOOSE CARDS"'
 1440 PRINTCHR$131;"DO YOU WANT THE COMPUTER TO PLAY THE"
'CHR$131;"CARDS FOR YOU?"':P$=GET$:IF P$<>"Y" AND P$<>"N"
 THEN 1400
 1450 IF P$="Y" THEN AD=1:PRINTTAB(5)CHR$130;"O.K.  I'LL D
O THAT FOR YOU."' ELSE AD=0:PRINTCHR$130;"O.K.  USE THE SP
ACE BAR TO PLAY CARDS"'
 1460 PRINTTAB(4)CHR$134;CHR$136;"WHO GOES FIRST <L OR R>
 ?":P$=GET$
 1470 IF P$<>"L" AND P$<>"R" THEN 1400
 1480 VDU7:IF P$="R" THEN P=1 ELSE P=0
 1490 ENDPROC
```



Mailsort

MAILSORT

Have you got any bright ideas on how to make the most of your BBC Micro? Any programming tips you think will help your fellow readers improve their listings? If you have, why not write in and tell us.

Just as a small incentive, from the next issue we'll be including a star letter, the author of which will receive £10. So, if you're saving up for a new add-on or a piece of software, why not share your ideas with the readers of *A&B Computing* and profit from your knowledge!

IMPRESSIONS

Dear A&B Computing,
As a prospective BBC Micro user, I eagerly bought your first issue and although finding its contents to be worthwhile, I found it marred by the assumption that all its readers were fairly proficient with BBC BASIC.

For example, nowhere in the magazine was the word 'PROC' explained properly — a word that I have never encountered before.

Your 'Welcome' section gives the impression of a magazine with something for everyone — but here sits a competent BASIC programmer who thinks otherwise.

I feel that this first issue would have been much better received as a second or third issue as its function as an introductory issue has not, in my opinion, been fulfilled.

Being a former ZX81 user and an avid fan of *ZX Computing*, I cannot help but find its new-born sister to be rather perplexing and something of a let-down.

Yours faithfully,
GRAHAM NIMMO,
Fife,
Scotland.

I am sorry to hear that you were disappointed with the first issue of A&B Computing and appreciate the comments you make in your letter.

Being a regular reader of ZX Computing, which is another specific magazine, you will have noticed that there are foibles of ZX programming peculiar to that range of machines. The same is true with BBC Microcomputers. Unfortunately, because you are unfamiliar with the BBC Micro there are bound to be things which are a blurr to you, which will become clear with constant use of the machine. We will, however, try to explain most terms that crop up in the magazine so that the whole spectrum of computer owners are not in the dark.

If you do decide to buy a BBC Microcomputer I am sure you will find that it is not a perplexing machine to use. Finally, here is an explanation of the word 'PROC' mentioned in your letter:
PROC is the shorthand notation for procedure. Basically, a procedure can be thought of as an extended subroutine. At the end of most BBC programs you will find a DEFPROC statement. This identifies the beginning of the subroutine and gives it a name. To delineate the end of the code an ENDPROC statement is used. This can be thought of as RETURN in other BASICs. To call a procedure you use the PROC statement together with the name of the routine. This will then transfer the operation of the program to that procedure. It will continue operating here until it meets an ENDPROC statement.
Thanks for your letter. Ed.

BUZZ OFF

Dear A&B Computing,
I wonder if any of your readers are wondering about some of the spurious sounds you can get out of your BBC Micro. I found I could pick up our local BBC radio station when touching components in the sound amplifier area of my micro!

Also, the unpleasant buzz emanating from the small speaker seems to be common to most of the BBC Micros I have encountered. However, the following cure may be of interest to other owners.

By soldering a 10nF (0.01uF) capacitor between the exposed ends of the vertically mounted resistors, R8 and R11, I found the buzz was greatly reduced in level, without affecting the audio output. These resistors are located at the bottom left-hand corner of the circuit board as viewed from the front of the machine, above and below the LM324N IC.
Yours faithfully,
W N WATKINS,
Belfast.

A SOFT TOUCH

Dear A&B Computing,
A group of teachers and ex-pupils of the Tabor High School, Braintree have set up an an organisation called 'Taborsoft'. We are partially funded by the MEP to produce Biological Software for schools. As well as producing packages for the MEP we are able to supervise programmers as they produce other work for us.

I'm writing this letter to you to let you know of our existence. Obviously because we're supported by the MEP and the teachers at the same time, we have no particular need to be commercial and therefore rather give a service to teachers or anyone else interested in software for the BBC Computers.

Perhaps you can let your readers know of our existence and also that we have available a few programs which we are sure will be useful. Our aim is to produce a suite of programs which will give a total revision for any student at school doing 'O' and 'A' levels. The two we have produced already are one on the heart and one on transpiration and our latest which is a massive program on Classification which basically will classify any animal or plant you happen to name simply by answering questions and put it in its appropriate group. It's a 28K package. What we're going to do is sell these for £2.50 each. We will provide the cassette providing the buyer provides a SAE. Our name is 'TABORSOFT' and cheques can be sent to the above address. You are of course under no obligation to print this, but if you think it would be useful to your readers, either as students or teachers, we would be more than happy for you to do so.
Yours faithfully,
CHRISTOPHER J SMITH,
Braintree.

IN THE PICTURE?

Dear A&B Computing,
I was very interested in the first issue of your magazine, many of the programs in which showed great promise.

After many hours spent keying in 'Cells and Serpents' we managed to get the program working satisfactorily. However, my two sons, aged eight and six, complained that there were no pictures.

You did say that if any readers took up the challenge and re-wrote the program to get in touch with you . . . so here I am. We have now got a version of the game with pictures of the monsters, treasure and rooms (in MODE 7 graphics) and also sounds relating to the monsters, spells, exit, etc. We do have a disc system, but have managed to get the program working without it moving down. This we have done by loading at &1100, where it will just work on our Model B (the top being &7000).
Yours faithfully,
MRS H BROWN,
Stanford-le-Hope,
Essex.

NEW! ZX SPECTRUM TAPE NOW READY!
NEW! EXPANDED DISC VERSIONS FOR APPLE, PET AND SHARP!

SOFTWARE FROM A&B

THE VALLEY

What are you... Barbarian or Wizard?

Choose your character type carefully... Barbarians recover quickly but their magic doesn't come easily. A Wizard? Slow on the draw and slow to mature... but live long enough and grow wise enough and your lightning bolts are almost unstoppable...

The Valley is a real-time game of adventure and survival. You may choose one of five character types to be your personal 'extension of self' to battle and pit your wits against a number of monsters. Find treasure, fight a Thunder-Lizard in the arid deserts of the Valley, conquer a Kraken in the lakes surrounding the dread Temples of Y'Nagioth or cauterise a Wraith in the Black Tower. In fact live out the fantasies you've only dared dream about. BUT BEWARE... more die than live to tell the tale!

You've read the program (Computing Today — April '82) ... Now buy the tape. Tape versions (£11.45 each inc P&P and VAT) available for: ZX Spectrum (48K), Atari 400 and 800 (32K), Tandy TRS-80 Model 1 Level 2, BBC Model A and B, Sharp MZ-80A, Sharp MZ-80K (18K), VIC-20 (with 16K RAM pack) and PET (New ROM, 16K RAM minimum). Disc version (£13.95 each inc P&P and VAT) available for: Apple II (DOS 3.3), Sharp MZ-80A, Sharp MZ80K and PET 8032 (8050 drives).

A complete reprint of the Valley article is also available for those wishing to do their own conversions for £1.95.

Fill in the coupon and return it to CT Software, ASP Ltd., 145 Charing Cross Road, London WC2H 0EE and become one of the many to play... The Valley...

Please send me the following versions of The Valley program:

Tape .....**@£11.45 all inclusive.** **I enclose cheque/PO for £..... (payable to ASP Ltd).**
Disc .....**@£13.95 all inclusive.** **OR Debit my Access/Barclaycard (delete as necessary)**
Article reprint .....**@£1.95 all inclusive.**

Please use BLOCK CAPITALS
Name (Mr/Mrs/Miss).....

Address.....

.....Postcode.....

Signature..... Date.....

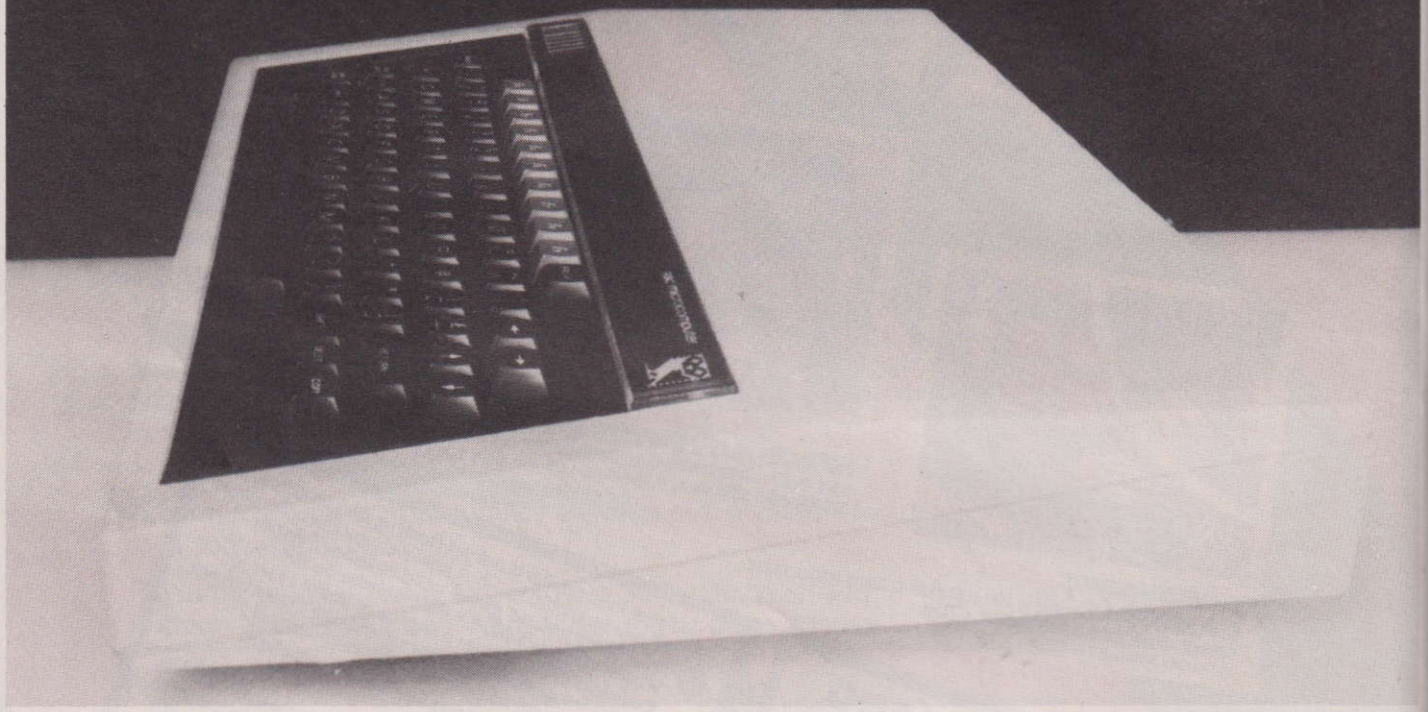
Please allow 21 days for delivery

ASP SOFTWARE

$tringing Along $ $

R. Metcalfe/P. Smith

Here's a clever utility program which will allow you to find lines within your listings quickly and easily.



Are you fed up with spending hours searching for specific lines in your BASIC programs? This find utility program is the answer to your problems.

It cleverly allows any string to be entered as a target and displays the number of any line in which the string is found. Once that's done it returns to BASIC and allows the lines to be listed. A text window is set to prevent the line numbers being scrolled off the screen when listing. BASIC keywords cannot be found because they are stored as tokens. The program as listed assembles from location &D00 and may be saved and run as a machine code program from that location.

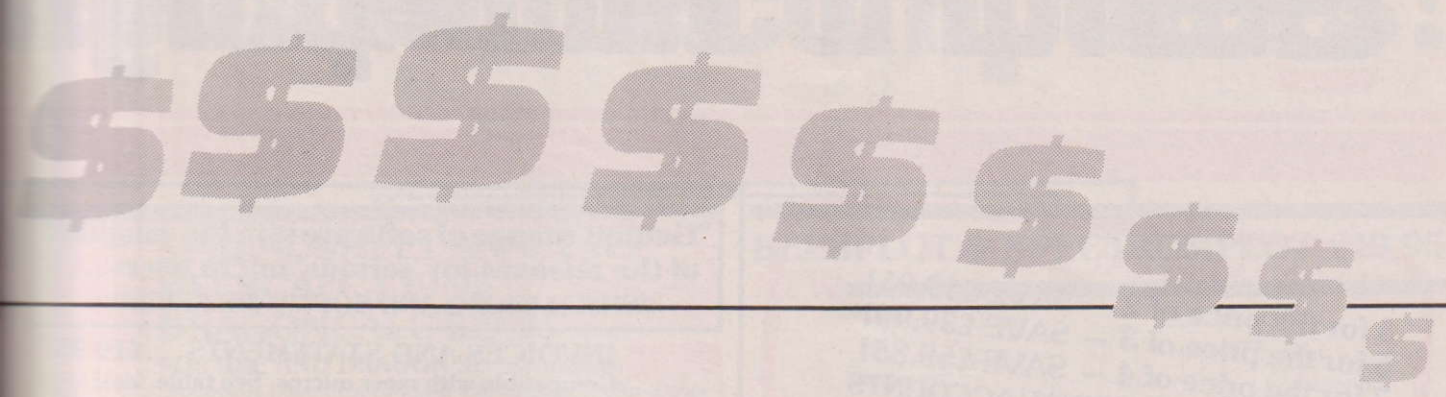
PROGRAM NOTES

The following are a few explanatory comments and user modifications.

After using the FIND utility the text window left can be removed by typing CTRL Z. Pressing ESCAPE will jump out of the FIND utility and return to BASIC. f5 will call FIND again (provided it was defined at line 10). The total length of the assembled code allows space for the cassette bug fix required by the 0.1 MOS. The FIND utility works on BBC Model A or B and has been tested on the 0.1 and 1.2 operating systems.

LINE DESCRIPTION

10	Sets user key f5 to call the routine and may be omitted.
50	The machine code is assembled from this address which may be altered. If this is done then the address in line 10 should also be changed.
60 to 110	These are MOS and BASIC routine addresses.
150	During assembly this prints the last address used by the assembled code and will depend on the string length set in line 170.
160-190	Sets up the block used by the OSWORD call.
170	Sets the maximum length of the target string allowed.
320	This sets the value of the high byte of PAGE to &EO. If you have a disc system then this should be set to &19 in order to set PAGE to &1900. Any value of PAGE can be set and can be changed after assembling the programme by noting the location of the code in memory and entering a new value using the Indirection Operators as explained in the USER GUIDE.
1360 to 1390	Sets pointers to memory areas used at the end of the assembled code.
1420 to 1460	A function to emulate the DEFM assembly psuedo-operation.



PROGRAM LISTING

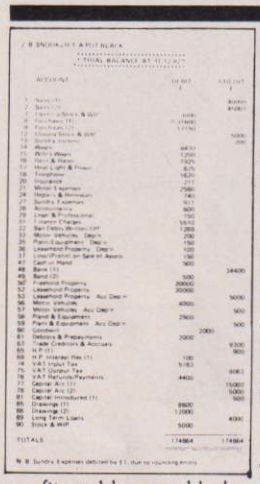
```
10 *KEY5CALL&D00|M
20 REM MAC. CODE FIND UTILITY
30 REM BY ROD METCALFE & PHIL SMITH
40 REM 7 JAN 83
50 START=&D00:REM MACHINE CODE ASSEMBLED FROM THIS ADDRESS
60 OSWORD=&FFF1
70 OSBYTE=&FFF4
80 OSNEWL=&FFE7
90 OSWRCH=&FFEE
100 OSASCI=&FFE3
110 BSWRDC=&98F1:REM PRINTS LINE NO. IN DEC
120 PMSG=&70
130 PLINE=&72
140 S=0:GOSUB240:REM FIRST PASS OF ASSEMBLER
150 PRINT"END=";MSG3+15:REM PRINTS LAST ADDRESS USED
160 BLOCK?0=MSG3 MOD256:BLOCK?1=MSG3 DIV256:REM BLOCK USED BY OSWORD
170 BLOCK?2=15:REM MAX NO OF CHARS IN TARGET STRING
180 BLOCK?3=23:REM MIN ASCII
190 BLOCK?4=126:REM MAX ASCII
200 S=3:GOSUB240:REM SECOND PASS OF ASSEMBLER
210 END
220
230
240 P%=START
250 [OPTS
260 LDA #26
270 JSR OSWRCH
280 LDA #12
290 JSR OSWRCH
300 .LOOP LDA#0
310 STA &72
320 LDA #&0E
330 STA &73
340 \ASK FOR TARGET
350 LDA #MSG1 MOD256
360 STA PMSG
370 LDA #MSG1 DIV256
380 STA PMSG+1
390 LDY #00
400 JSR OUTMSG\PRINTS 'FIND?'
410 \SET UP BLOCK
420 LDX #BLOCK MOD256
430 LDY #BLOCK DIV256
440 LDA #00
450 \GET TARGET
460 JSR OSWORD
470 BCS ESCAPE
480 \SEARCH LOOP
490 .SEEK LDY #1
500 LDA (PLINE),Y
510 CMP #255\END OF PROG?
520 BEQ DONE
530 LDX #0\POINT TO START OF TARGET
540 LDA MSG3,X
550 CMP #13
560 BEQ BASIC
570 LDY #4\POINT TO LINE START
580 .NX LDA (PLINE),Y\GET A CHAR
590 CMP #13\END OF LINE?
600 BEQ NXLINE\THEN NOT THIS LINE
610 CMP MSG3,X
620 BEQ CHARM
630 LDX #0
640 INY\ELSE POINT TO NEXT CHAR
650 JMP NX\AND DO IT AGAIN
660 .BASIC RTS
670
680 .CHARM
690 INX
700 LDA MSG3,X
710 CMP #13
720 BEQ MATCH
730 INY
740 JMP NX
750
760 .MATCH
770 JSR OUTLINE
780 JMP NXLINE
790
800 .OUTMSG\MESSAGE INDEX IN Y
810 LDA (PMSG),Y
820 CMP #13
830 BEQ ENDMSG
840 JSR OSASCI
850 INY
860 JMP OUTMSG
870 .ENDMSG RTS
880
890 .NXLINE
900 TYA
910 PHA
920 CLC
930 CLD
940 LDY #3
950 LDA (PLINE),Y
960 ADC PLINE
970 BCC NOCARY
980 INC PLINE+1
990 .NOCARY
1000 STA PLINE
1010 PLA
1020 TAY
1030 JMP SEEK
1040
1050 .OUTLINE
1060 TYA
1070 PHA
1080 LDY #1
1090 LDA (PLINE),Y
1100 STA &2B
1110 INY
1120 LDA (PLINE),Y
1130 STA &2A
1140 JSR BSWRDC
1150 JSR PSPC
1160 PLA
1170 TAY
1180 RTS
1190
1200 .PSPC
1210 LDA #32
1220 JSR OSWRCH
1230 RTS
1240
1250 .ESCAPE\DEAL WITH ESCAPE KEY
1260 LDA #126
1270 JSR OSBYTE
1280 JMP OSNEWL
1290
1300 .DONE\SEARCH COMPLETE
1310 LDY #7
1320 JSR OUTMSG
1330 RTS
1340
1350 ]
1360 BLOCK=P%:P%=P%+5
1370 MSG1=FNDEFM("FIND? ")
1380 MSG2=FNDEFM(CHR$(28)+CHR$(0)+CHR$(24)+CHR$(39)+CHR$(3))
1390 MSG3=P%
1400 RETURN
1410
1420 DEFFNDEFM(M$)
1430 $P%=M$
1440 M%=P%
1450 P%=P%+LEN(M$)+1
1460 =M%
```

Gemini more progra

GEMINI SPECIAL FREE OFFERS
3 for the price of 2 – SAVE £19.95!
5 for the price of 3 – SAVE £39.90!
7 for the price of 4 – SAVE £59.85!
(CASHBOOK & FINAL ACCOUNTS NOT INCLUDED)

CASH BOOK ACCOUNTS PROGRAM FOR BBC 32K, TORCH, SPECTRUM 48K

NEW £59.95

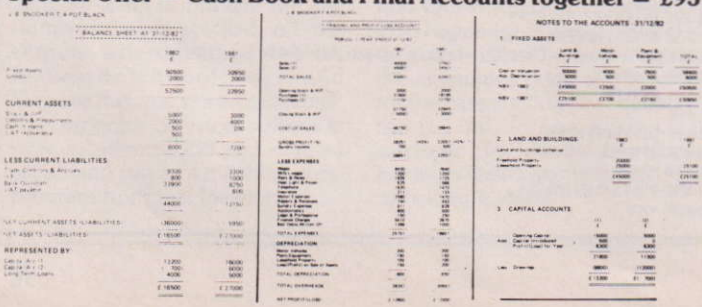


One of the most innovative business programs on the market. Replaces a manual cash book system, e.g. Simplex and 'All-in-One'. Written by practising Chartered Accountants, this practical program is simple to use and will replace your manual cash and bank records. By giving you access to vital management information as and when you want it, it will enable you to keep more positive financial control of your business.
The software is extremely well and lucidly documented, and Gemini provide a full technical back-up and product up-date policy. Take a look at the information this program will provide:-
• Summary of VAT information for VAT returns • Cumulative receipts and payments report analysed over the standard profit and loss and balance sheet headings • Option for departmental analysis of sales and purchases • Audit trail printout of all transactions. • Journal routine for entering transfers between accounts and year end adjustment for debtors, creditors etc. • Trial balance at any interval • Interfaces to 'Final Accounts' program to produce balance sheet and trading and profit/loss account etc. • Spectrum version may be used with Sinclair OR 80 column printer.

FINAL ACCOUNTS PROGRAM FOR BBC 32K, TORCH, SPECTRUM 48K £59.95

NEW

Requires Cash Book module. This program will take your cash book data to the logical conclusion of balance sheet, trading and profit/loss account and notes to the accounts i.e. fixed assets, land and buildings and capital accounts. Final accounts (BBC version) links to 'Beebplot' for graphic data presentation.
Format: Torch disk, BBC disk/cassette, Spectrum cassette.
Special Offer – Cash Book and Final Accounts together – £95



"Gemini's range of software is in the vanguard of the releases for 'serious' micro users . . ."
(WHICH MICRO AND SOFTWARE REVIEW)

INVOICES AND STATEMENTS . . . £19.95

Compatible with most micros. See table. Ideal for the small business. A complete suite of programs together with generated customer file for producing crisp and efficient business invoices and monthly statements on your line printer. All calculations include VAT automatically, and the program allows your own messages on the form produced. This program gives you superb presentation and saves time on one of the most tedious tasks in the office.

COMMERCIAL ACCOUNTS . . . £19.95

Compatible with most micros. See table. A gem of a program, all for cassette, with the following features:- Daily Journal. Credit Sales. Cash Sales. Credit Purchases. Purchases—other. Sales Ledger. Purchase Ledger. Bank Account. Year to date summary. A fully interactive program suitable for all businesses. Files can be saved and loaded and totals from one file carried forward to another on cassette. Particularly useful from a cash flow point of view, with an immediate accessibility to totals for debtors and creditors. Bank totally supported with entries for cheque numbers, credits and, of course, running balance.

MAILING LIST . . . £19.95

Compatible with most micros. See table. A superb dedicated database to allow for manipulations of names and addresses and other data. Gemini's unique 'searchkey' system gives you a further ten 'user-defined parameters' to make your own selections. Features include the facility to find a name or detail when only part of the detail is known, it will print labels in a variety of user specified formats.

DATABASE . . . £19.95

Compatible with most micros. See table. The program that everyone needs, the most valuable and versatile in your collection. Facilities include sort search, list print if required. Can be used in place of any card index application; once purchased you can write your own dedicated database to suit your particular needs with a limitless number of entries on separate cassettes.

STOCK CONTROL . . . £19.95

Compatible with most micros. See table. Dedicated software with all that's necessary to keep control of stock. This program will take the tedium out of stock control and save time and money. Routines include stock set up, user reference number, minimum stock level, financial summary, line print records, quick stock summary, add stock, delete/change record and more.

HOME ACCOUNTS . . . £19.95

Compatible with most micros. See table. Runs a complete home finance package for you with every facility necessary for keeping a track of regular and other expenses, bank account, mortgage, H.P., etc. This program also allows you to plot graphically by Histograms your monthly outgoings.

WORD PROCESSOR . . . £19.95

Compatible with most micros. See table. This program features routines found in much larger and more expensive packages with a typical word length of 5-6 letters it allows for around 1000 words in memory at one time. Ideal for the user who requires a simple program to write letters on his computer. Features include, block delete, block insert, search and replace, edit text, display text and more.

"Simple to use ....."
"Ideally suited to the way most offices run....."
PERSONAL COMPUTER NEWS

ns for more computers!

SPREADSHEET ANALYSIS BEEBCALC £19.95 DRAGONCALC £19.95

NEW

FOR BBC AND DRAGON 32. Spreadsheet processors have proved to be important tools for using micros in business, scientific and domestic financial applications.

```
POSITION A1 RC SPACE 5185 T
   II     A   II   B   II   C   II
 1   -J.B. SNOOKER T/A POT-BLACK
 2   PROJECTED CASH FLOW        YEAR      ENDED
 3
 4                              Oct.      Nov.
 5                               £         £
 6   INCOME
 7   Sales                     11786     10944
 8                            ------    ------
 9
10   REVENUE EXPENDITURE
11   Purchases                   500       500
12   Advertising                 500      1000
13   Director's salary          1596      1596
14   Salaries                   2216      2216
15   Rent
16   Telephone                              300
17   Insurance                              200
18   Printing, stationary                   400
19   Repairs & renewals
20   Hire of equipment            60        60
COMMAND BCDEFGPRSTW*?
```

Without any programming knowledge at all, you may:-

- Set up a computerised spreadsheet, with chosen row and column names.
- Specify formulae relating any row or column to any other.
- Enter your source data and have the results calculated.
- Save the results on tape (or disk– BBC) for later reloading and manipulation.
- Print the tabulated results in an elegant report format.
- Experienced users may access saved files and write their own reporting or graphics presentation programs for the results.

Some typical applications:—

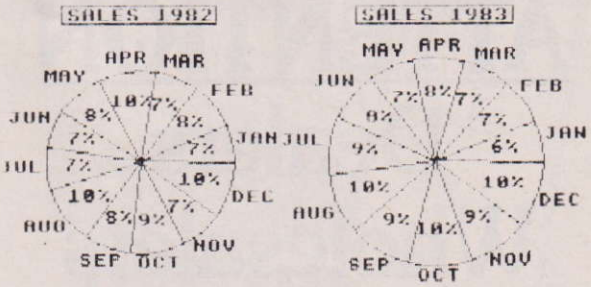
- Small business accounting applications, e.g. profit and loss statements and cashflow projections, break-even analyses etc.
- Investment project appraisal– anything from double glazing to oil rigs!
- Comparing rent/lease/buy options.
- Processing the results of scientific experiments or field studies.
- Engineering calculation models.
- In fact, anything that involves repeated re-calculation of results presented in tabular or spreadsheet format.

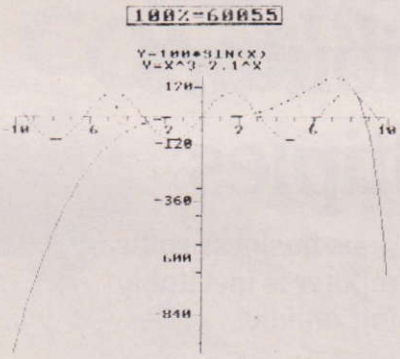
Program Availability Chart:-

	Database	Stock Control	Mailist	Invoices & Statements	Spread sheet Analysis	Cashbook Accounting	Word processor	Home Accounts	Commercial Accounts	Plot	Final Accounts
Sinclair Spectrum 16k or 48k	●	●	●			●		●	●	●	●
Dragon 32k or 64k	●	●	●	●	●			●	●	●	
VIC 20 16k + I	●	●	●	●				●	●		
Sinclair ZX81 16k + I	●										
Grundy Newbrain	●		●								
Sharp MZ80A	●	●	●	●				●	●		
Sharp MZ80K	●	●	●	●				●	●		
Sharp MZ80B	●	●	●	●				●	●		
BBC Micro model A or B 32k	●	●	●	●	●	●	●	●	●	●	●
Atari 400/800	●										
Torch	●	●	●	●	●	●	●	●	●	●	●
Epson HX-20	●	●	●								
Commodore 64	●		●					●			

BEEBPLOT & SPECTRUMPLOT £19.95

NEW





Important new additions to the Gemini family. Present numeric and string data together in easily-understood pie chart, histogram or graph format. Beebplot has a built-in interface to Beebcalc, and both Beebplot and Spectrumplot have built in interfaces to the Final Accounts program of Cashbook. The facility for mathematical function plotting is also provided. The BBC version has a high resolution screen dump for the Epson or CP-80 printers, and the Spectrum version dumps to the Sinclair printer via the 'copy' key. A very useful program that will give superb results either from direct input of data from the keyboard or via simple access to other software data files. A must for business and education.

Dealer/Trade enquiries invited– generous trade discounts for quantity.
Special ACCESS card instant sales hotline for prompt despatch...
24 hr Ansaphone Service.
All enquiries other than credit card sales to 03952-5832

Tel: 03952 5165

Gemini. Functional Software Specialists. 9, Salterton Road, Exmouth, Devon. EX8 2BR

PLEASE SEND URGENTLY

_____ _____
_____ _____
_____ _____
_____ _____

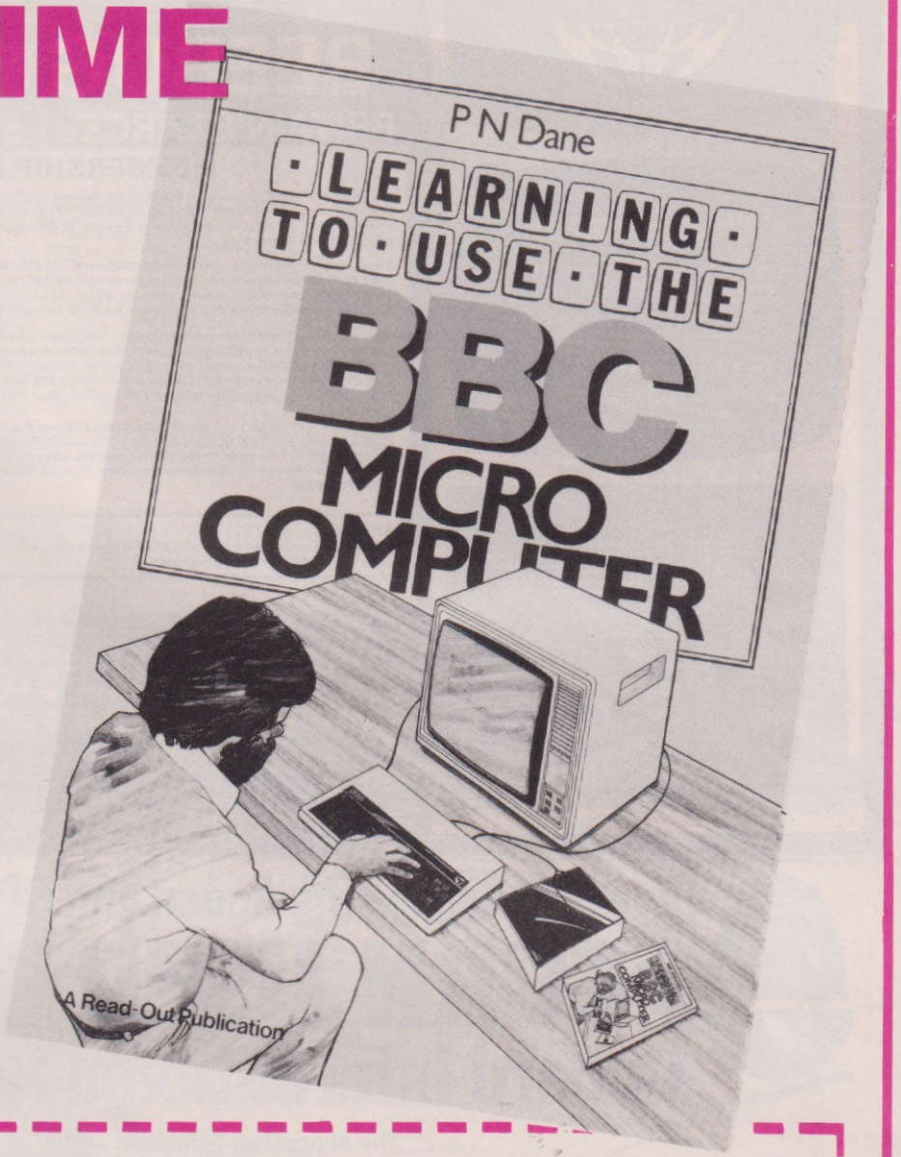
(Please note: Items as priced except BBC/TORCH DISKS: please add £4 extra for 40 track and £5 for 80 track format)

Name _____
Address _____
Machine Type _____ **Memory Size** _____
I enclose _____
Make cheques and postal orders payable to Gemini Marketing Ltd.
Access Number _____
Signature _____

ACCESS only by Telephone

Gemini
MARKETING LIMITED
Gemini. Functional Software Specialists, 9 Salterton Road, Exmouth, Devon EX8 2BR

A NEW BOOK FOR FIRST TIME USERS



"Learning to Use the BBC Micro" is in a new series of books which introduces newcomers to the most widely used micros in the marketplace.

The books assume absolutely no knowledge about computers and the reader is shown even the most fundamental operations such as "switching on" and "loading a program". The books lead the reader through simple programming and then on to graphics, with several programs which show how to achieve pictures and even animation!

The user-friendly approach is consistent throughout the text – not only are program listings clearly shown, but in many cases, a photograph is included to show what the program looks like when actually loaded and run!

All books in the series are £5.95 (incl. postage).

Gower

"The Learning to Use" series will bring comfort and help to many first time microcomputer users. The books represent the first genuine approach to assisting the beginner to get to grips with the best selling personal microcomputers . . ."

Communicate

Order Form to: **READ-OUT BOOKS AND SOFTWARE**

8 CAMP ROAD, FARNBOROUGH, HANTS, GU24 6EW.
24 Hour Answering Service. Telephone 0252 510331/2

Name _____

Address _____

_____

Please send me .....copy/ies of

☐ Learning to Use the BBC Microcomputer

@ £5.95 including postage and packing. (Allow 14 days for delivery.)

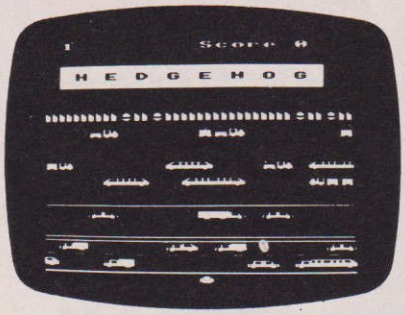
Make cheques payable to Read-Out Publishing Company Ltd.

I enclose my cheque for £.....

Please debit my Access ☐ Number ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Signed _____ Date _____

BBCCOM5



Screen Photographs of programs in BEEBUG

BBC MICRO

REGISTERED REFERRAL CENTRE FOR THE BBC PROJECT

BEEBUG FOR THE BBC MICRO

BRITAIN'S LARGEST SINGLE-MICRO USER GROUP

MEMBERSHIP NOW EXCEEDS 18,000

18,000 members can't be wrong. **BEEBUG** provides the best support for the BBC Micro. **BEEBUG** Magazine - now 64 pages including new product guide supplement - devoted exclusively to the BBC Micro. Programs. Hints & Tips. Major Articles. News. Reviews. Commentary. **PLUS** members discount scheme with National Retailers. **PLUS** members Software Library. 10 Magazines a year. First issue April 1982. Reprints of all issues available to members.

March Issue: Program Features: Life (32k), Artillery Duel (16k/32k), Square Dance, Microsketch (16k) screen drawing program in only one line of code. 3D Rotation (will rotate any object). Printers for the BBC micro - Review of Epson, Seikosha, Tandy and Olivetti. What to do with the new Operating System Chip. Disc Formatter Program, and full Disc instruction set. Newcomers article on Text and Graphics Windows. **PLUS** How to get a new Operating System ROM and a special deal on Wordwise (members only).

April/May Issue Special Anniversary Issue - **Contains index to the whole of BEEBUG Volume 1.** Music Composer - create complex 3 part harmonies with this synthesiser Program. **Colour bar chart generator** program. Beeb implementation of the **Connect-Four Game. Invasion** - a 16k Plus. **Review of Tape Recorders** for the Beeb: a Basic Program Editor, which lists variables and procedures, and executes Find and Replace in a Basic Program; **Reviews of Acornsoft Games** and the **Torch Z80 Disc** Pack. Disc Menu Program. Newcomers introduction to Mode 7. How to save the unsaveable, and a routine to print Double Height Characters in all modes.

June Issue: Program Features **'Return of the Diamond'** A 16k adventure game, **'hedgehog'** a well implemented 'Frogger' type game, and **Ellipto**. Create your own off the shelf sound effects with **Sound Wizard**. Plus articles on **Using Files**, **Rotating and Expanding Characters**, **Using Printers**, and **How to multi-program the User Keys**. Reviews of **The Hobbit Floppy Tape System**, **Adventure Games**, **and a Comparative Review of Wordwise and View**. Plus **FX Call Update**, **Disc Program Auto-relocator**, **Worldwise Update**, and more **BBC Book Reviews**.

STOP PRESS

BEEBUG has negotiated a deal with ACORN over the new 1.2 OPERATING SYSTEM ROM. BEEBUG members are offered the ROM at around half-price. See BEEBUG Feb. issue for details.

SOFTWARE DETAILS

BEEBUGSOFT - BEEBUG SOFTWARE LIBRARY

offers members a growing range of software from £3.50 per cassette. 1. Starfire (32K). 2. Moonlander (16K). 3D Noughts and Crosses (32K). 3. Shape Match (18K). Mindbender (16K). 4. Magic Eel (32K). 5. Cylon Attack (32K). 6. Astro-Tracker (32K). **Utilities**: 1. Dissembler (16K). Redefine (16K). Mind Text Ed (32K). **Applications**: 1. Superplot (32K). 2. Masterfile (32K).

Magazine programs now available on cassette to members at £3.50 inc. VAT & P&P. See April/May issue for details.

13% DISCOUNT TO MEMBERS ON THE EXCELLENT WORDWISE WORD PROCESSING PACKAGE
THIS REPRESENTS A SAVING OF OVER £5.00

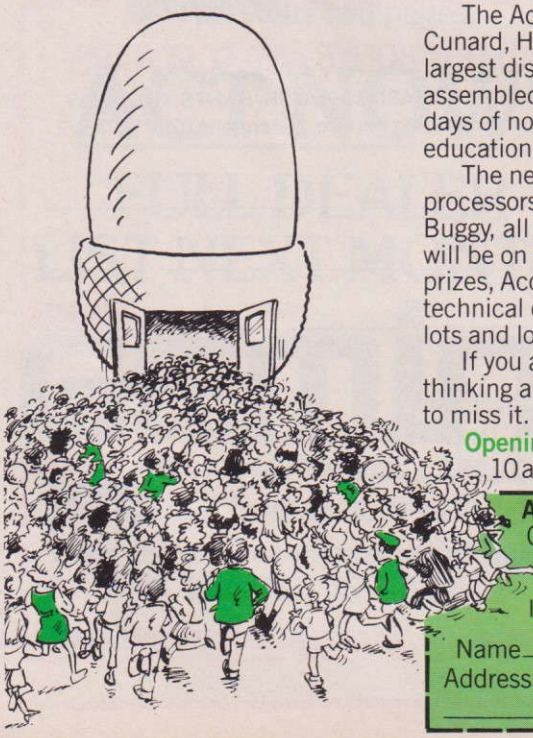
Send £1.00 for SAE for Sample
Membership: UK £5.40 for six months, £9.90 for one year.
Overseas one year only: Europe £16.00, Middle East £19.00, Americas & Africa £21.00, Other Countries £23.00.
Make cheque to BEEBUG and send to: BEEBUG, Dept 15, 374 Wandsworth Rd., London SW3 4TW
For editorial material send to: The Editor, BEEBUG, PO Box 50, St. Albans, Herts AL1 2AR

Everything for the Acorn and BBC microcomputer user.

ACORN USER EXHIBITION

Cunard International Hotel

Hammersmith, 25-28 August 1983



The Acorn User Exhibition at the Cunard, Hammersmith will house the largest display of Acorn products ever assembled under one roof. It will be four days of non-stop entertainment and education for parents and children alike.

The new Electron, the second processors for the BBC micro, the BBC Buggy, all the new software and hardware will be on show. There'll be competitions, prizes, Acorn experts to answer your technical questions, demonstrations and lots and lots of bargains.

If you are an Acorn owner, or just thinking about being one, you can't afford to miss it.

Opening hours: August 25th-27th, 10am-7pm; August 28th, 10am-4pm.

Admission charges: Adults £2 per ticket, Children £1 per ticket.

We have arranged for nearly every exhibitor to redeem the cost of your ticket when you buy something from their stand.

Group rates: 10% discount for parties of 10 or more.

Buses: Frequent services from central London.

Tubes: Hammersmith Broadway – Metropolitan, District and Piccadilly lines.

Car Parking: Several car parks in the immediate area.

For details of exhibition stands and advance ticket sales contact Computer Marketplace Ltd, 20 Orange Street, London WC2H 7ED. Tel: 01-930 1612.

ADVANCE BOOKING COUPON Miss the queue – buy your tickets in advance.
Computer Marketplace Ltd, 20 Orange Street, London WC2H 7ED.

Please send_____ Adult tickets at £2 _____ Childrens tickets at £1

I enclose a cheque/postal order value £____ payable to Computer Marketplace Ltd.

Name_____

Address_____

Electronequip

(Authorised BBC Dealer, and service centre)

NEW PRODUCTS

Utility Disc for BBC Contains VER FORM 35 FORM40 and FORM80 ..... **£9.95**

BBC Sparkjet Printer New quiet printer for BBC. Friction and tractor feed 80cps. **£379.50**

Torch Z80 Disc pack 800K dual disc drive plus Z80 processor with CMP compatible operating system **£897.00**

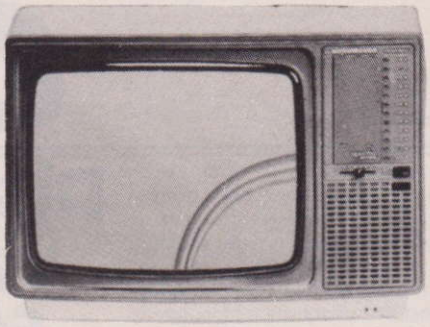
TORCH Computer 800K to 21.4M disc drives. High res. colour monitor. Plus auto-dial modem..... **From £2795.00 + VAT**

New Epson FX80 FX80 160cps printer in stock. Friction and tractor feed + proportional spacing..... **£458.85**

BBC 2	BBC Model B Micro Computer	£399.00
BBC 3	BBC Model A Micro with 32K	£333.50
BBC 4	BBC Model A Micro with 32K and VIA	£339.50
BBC 5	BBC Model B with Disk Interface	£469.00
BBC 6	BBC Model A with Econet Interface	£356.00
BBC 7	BBC Model B with Econet Interface	£456.00
BBC 8	BBC Model B with Disc & Econet Interface	£526.00
BBC 21	BBC Model A to B Upgrade	£99.82
BBC 28	Econet Upgrade for BBC	£92.00
BBC 23	BBC Acorn Memory Upgrade for Model A	£34.50
BBC 27	Disc Upgrade for BBC B (inc fitting)	£92.00
BBC 30	BBC 14" Colour Monitor	£287.50
BBC 33	Sanyo SM12N Green Monitor 15MHz	£90.85
BBC 34	BMC 12E Green Monitor 18MHz	£113.85
BBC 35	Karga K12A 12" Orange Monitor	£129.95
BBC 41	BBC Single 100K 5.25" Disc Drive (AND01)	£265.00
BBC 43	BBC Dual 800K 5.25 Disc Drive (AND02)	£803.85
BBC 44	Single Disc Drive (100K) for BBC (Teac)	£205.85
BBC 45	Single Disc Drive (200K) for BBC (Teac)	£263.35
BBC 46	Single Disc Drive (400K) for BBC (Teac)	£343.85
BBC 47	Dual Disc Drive (200K) for BBC (Teac)	£411.70
BBC 48	Dual Disc Drive (400K) for BBC (Teac)	£526.70
BBC 49	Dual Disc Drive (800K) for BBC (Teac)	£687.70
BBC 50	Epson FX80T 160cps Printer	£458.85
ATM 2	Acorn Atom assembled 12K ram	£184.00
ATM 26	Atom New Power Supply 1.8A	£9.66
ATM 21	Floating Point ROM for Atom	£21.85

Large stocks of software for BBC and Atom. Business, Games and Educational. Send for comprehensive lists.

Large stocks. Prices inclusive of VAT. All prices inclusive of postage except micros. £3.00. All Upgrades etc. are fitted free of charge and the computer fully re-tested. Access and Barclaycard Welcome.



14" Colour portable Monitor/TV

This Monitor/TV us not a modified television as many TV/Monitors are, but a 14" Monitor/TV which has been designed to perform both functions. It has RGB and Composite video and sound. An RGB cable for a BBC is supplied as standard.

£259.90 Trade enquiries welcome

ACORN COMPUTER

Electronequip

BBC

36-38 West Street, Fareham, Hants TO16 0GN (0329) 230670

DON'T MISS THIS INCREDIBLE OFFER!

50 GAMES

£9.95

YES, 50 GAMES! on cassette for all the following:

GALACTIC ATTACK	MAZE EATER	SUBMARINES
SPACE MISSION	ORBITTER	ROCKET LAUNCH
LUNAR LANDER	MOTORWAY	PLANETS
PLASMA BOLT	FORCE FIELD	BLACK HOLE
STARTREK	NIM	DYNAMITE
RADAR LANDING	TUNNEL ESCAPE	DO YOUR SUMS
ATTACKER	BARREL JUMP	DERBY DASH
GALACTIC DOGFIGHT	CANNONBALL BATTLE	SPACE SEARCH
ZION ATTACK	OVERTAKE	UNIVERSE
IVASIVE ACTION	SITTING TARGET	RATS
OXO	SMASH THE WINDOWS	TANKER
BOGGLES	SPACE SHIP	PARACHUTE
PONTOON	JET FLIGHT	JETMOBILE
SKI JUMP	PHASER	HIGH RISE
HANGMAN	INTRUDER	THE FORCE
OLD BONES	INFERNO	EXCHANGE
THIN ICE	GHOSTS	

sinclair SPECTRUM

apple (ON DISC & CASSETTE)

ACORN·ATOM

BBC

DRAGON

ATARI VIC 20

SHARP

ORIC-1

NEW BRAIN

ZX81 LYNX

POSTAGE FREE TO U.K. ORDER DIRECT TO:

U.T.S.
P.O. Box 96
HARROGATE
NORTH YORKSHIRE
HG3 2UZ

PLEASE SEND ME _____

CHEQUE ENCLOSED, VALUE _____

NAME _____

ADDRESS _____

_____

POSTCODE _____ G

cut & post now

BBC Micro-Aid

SOFTWARE - Programs that are guaranteed to run! Save hours of work and worry with these utilities and practical programs on cassette or disc

102	CASHBOOK	Double entry 4 columns with accounts	£ 5.95	B
103	LEDGER	Complements CASHBOOK with ageing & analysis	£ 5.95	B
104	MAILING	Holds 218 addresses. System with 6 options, 2 Sorts, Labels, 2 Searches & Update	£ 5.95	B
105	PAYROLL (W or M)	In two parts to handle weekly or monthly PAYE & NI for 100 employees. Supported.	£13.90	B
105a	Manual	Extra (No VAT)	£ 2.50	B
106	MEMO-CALC	Database/Calcsheet with up to 255 columns, string or numberic data, sorts, searches, calculations, with full print facility.	£ 9.95	B
106a	Manual	Extra (No VAT)	£ 2.00	
201	CARDS	Beat Bruce Forsyth at his own game.	£ 2.95	A B
202	BATTLE	Fast moving game simulation of a minefield.	£ 2.95	B
203	HANGMAN	Word game. French, German, Italian, Spanish.	£ 7.95	B
301	BANNER	Paper printout of large text & graphics.	£ 2.95	A B
302	DISTANCES	Graphic maps of U.K., EUROPE & WORLD. Calculate distance between any two places.	£ 3.95	B
303	FLAGS	Full colour flags of the world & questions.	£ 3.95	B
304	STATPAK	Statistics offering over 30 results.	£ 8.95	B
501	SEARCHBAS	PROC to search a BASIC program & alter it.	£ 1.95	A B
502	PROCVAR	PROC to list variables used in a program.	£ 1.95	A B
503	PROCFLUSH	PROC to clear Resident Integer Variables.	£ 1.00	A B
504	PROCAID	A combination of programs 501, 502 & 503.	£ 3.45	A B
505	DEFCHR	Design, display & store graphic characters.	£ 2.95	A B
506	SORTM/C	Machine code sort for up to 255 integers.	£ 1.00	A B
507	SORBAS	A very fast BASIC sort. 1000 items in 42s.	£ 1.00	A B
508	UTILITY-A	Combination of 501 to 507. Superb value.	£ 5.95	A B
600	FORTH	Second language ROM for either OS.	£34.74	B
601	WORDWISE	ROM Superb fast & easy wordprocessor OS 1.0	£34.74	B
701	BOOKS	Various titles for the BBC Micro from (No VAT)	£ 6.95	
801	CASSETTES	C12 Computer quality boxed in 10's	£ 4.50	
810	DISCS	MEMOREX Soft Sectored 40 track 5.25"	£19.95	
900	NEW EPSON	RX-80 100 CPS 3 Founts Graphics	£379	
901	NEW EPSON	FX-80 F/T, 160CPS, 3 FOUNTS, Graphics	£379.00	
905	SEIKOSHA	GP-100a Printer 50cps, 80 columns, tractor	£195.00	
910	NEW TEAC	Slimline Disc drives suitable for BBC Micro with Power supply. Format disc & Manual.		
	100K	Single Sided, 40 Track. OTHERS AVAILABLE	£199.00	
920	VDU STAND	Stainless steel support protects your micro	£19.95	

ADD VAT TO ALL PRICES. FOR COPY ON DISC ADD £1.50. NO PACKING CHARGE.

If you want further information before parting with your hard earned cash send for our new brochure to:-

Micro-Aid (BC)
25 Fore Street, Praze, Camborne, Cornwall TR14 0JX.
Tel: 0209-831274

No Hard Graft

Norman Fox

Display numerical values in graph form with this simple plotter.

This program uses the screen as a piece of graph paper, plots points using the co-ordinates you give it and joins them up. I might use this kind of graph at home to see how my heating bill has changed over the last five years since I insulated the loft. Similarly at work to plot sales, costs and profits (or losses!) for the last twelve months. I have left some blank lines and some REMs in the listing to show how the program is divided into identifiable chunks.

The first step is to move the graphics origin, normally at the bottom left corner of the screen, to a location which will allow numbers to be printed along the axes of the graph. I chose the point X = 80, Y = 64 for the new origin and set it using the VDU28 instruction in line 100.

Next, we need somewhere to put the text, such as questions and prompts, while the graph is being constructed. Line 110 sets up a 'text window' at the bottom of the screen. You can set it anywhere but it's a good idea to keep it away from the area in which the graph will be plotted. The X-axis and Y-axis can be drawn using those handy MOVE and DRAW instructions.

WHAT'S THE POINT

At this stage, we need to specify how many points are to be plotted. This information is used to set up a loop (lines 565 to 650) asking for each point's co-ordinates to be input. This is so that the program cannot accept more points than you specified. Alternatively, we could simply ask for the first point and ask "any more points?" after each one. In this way, you could plot as many or as few as you liked.

We now come to the problem of dividing up each axis and printing a 'marker' at each division. Markers are just conveniently shaped characters and are placed in position by joining the text and graphics cursors together with the VDU5 instruction. This moves the combined cursor to the appropriate place and finally prints the character.

This use of two cursors is a powerful feature of BBC BASIC since it allows characters to be plotted anywhere on the screen. The **User Guide** has a program which plots a whole sentence in the form of a sine wave!

If you want the axes to have a number beside each marker, answer Y to the question "Label axes?". The numbers are calculated in lines 1070 and 1120. Because of the limited size of a screen I suggest you don't use maximum X or Y values much greater than 1000. Axes do not need to be labelled.

You might decide to have 12 divisions, one for each month, however the program as it stands doesn't provide a facility for printing letters. This would not be difficult to improve by modifying lines 1070 and 1120 in PROClabel. This prints a character instead of a number, using PRINT A$ for example.

DETAILING

You will already have noticed from the listing that the program uses MODE 0. This gives maximum detail on the screen though lacks colour facilities. Since the program was originally designed to produce a paper printout, this is not important. It does cause some difficulty if you want to plot several graphs on the screen at the same time.

Using colour, these can easily be identified, but if colours are not available, what can be done? Well the way it's done on paper is to use different characters for the 'points' of each graph. Also, join up the points with solid lines, dotted lines, dashes and so on. I've used both methods by using PRINT CH$(Z) to print each point, where Z can be the ASCII code for any character you choose.

In the program, Z is either 42 (an asterisk) or 111 (small letter o). Also the PLOT instruction draws solid lines if PLOT5, X, Y is used, or dotted lines for PLOT21, X, Y. So far these are the only two possibilities but more may emerge with the graphics extensions planned by Acorn.

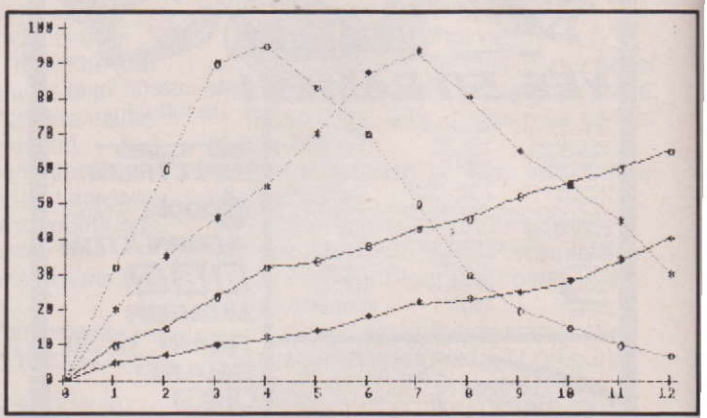
Although these methods can distinguish between three or four graphs on the same screen, they can't cope with many more. The program restricts you to a maximum of four graphs, each of which has to have the same scale. I've included a routine for printing what is on the screen, a 'screen dump', which works with an Epson printer.

As it's written in assembler language, it only takes about two minutes to print the whole screen. It also only works with MODE 0 as it stands, and it can be detached from the program and used as a subroutine. This can be used to print some of the more interesting patterns produced by various graphics programs. Of course, if you don't need the printout routine don't type in the procedure PROCsdump.

IN POSITION

By the way, some of the MOVE instructions have strange looking numbers in them. These are simply to position printed characters at positions which look right (to me!). A graph should be a thing of beauty, if not a joy for forever, and I just changed the numbers until I got the desired effect. This is an example of doing things in a practical way rather than a theoretical one. Don't waste too much time calculating what should happen in a layout, just go ahead and try it.

With the computer's excellent screen editing facilities, altering a BASIC program line only takes a few seconds. Finally, why not use more procedures, for example for drawing the axes? I believe that it really isn't worth using a procedure if the routine is only to be used once in the program. If it will be used several times each run however, then a procedure is worthwhile and does make the program easier to read. Happy graphing!



PROGRAM LISTING

```
>LIST
    1 REM Graph program..Norman.W.Fox
   10 MODE 0
  100 VDU29,80;64;:REM Set graphics origin
  110 VDU28,8,31,35,31:REM Set text window
```

```
110
115 REM Draw axes
120 MOVE 80,64
130 DRAW 1280,64
140 MOVE 80,64
150 DRAW 80,1024
152
155 REM Set variables
160 Z=42:C=0:A=5
200 INPUT "How many points",N
210 INPUT "How many X-div's",XD
220 INPUT "How many Y-div's",YD
230 INPUT "Max. X-value",XM
240 INPUT "Max. Y-value",YM
250
260 REM Print markers on axes
300 VDU5
310 X=1100/XD
320 FOR I=0 TO XD
330 MOVE (X*I)+70,76
340 PRINT "!"
350 NEXT I
360 Y=880/YD
370 FOR I=0 TO YD
380 MOVE 68,(Y*I)+94
390 PRINT "_"
400 NEXT I
410 VDU4
420
450 REM Are axes to be labelled?
500 INPUT "Label axes Y/N",A$
510 IF A$="Y" THEN PROClabel
520
535 REM Put in the actual values
550 PROCvalues
660
700 INPUT "More graphs",A$
710 IF A$="Y" THEN PROCmore
720 INPUT "Printout (printer on?)",A$
730 IF A$="Y" THEN PROCsdump
750 END
800
1000 DEF PROClabel
1005 MOVE 80,64
1010 VDU5
1020 P=1100/XD
1030 FOR I=0 TO XD
1040 MOVE (P*I)-70,45
1070 PRINT INT(I*XM/XD+.9)
1080 NEXT I
1085 MOVE 80,64
1090 P=880/YD
1100 FOR I=0 TO YD
1110 MOVE -100,(P*I)+80
1120 PRINT INT(I*YM/YD+.9)
1130 NEXT I
1140 VDU4
1200 ENDPROC
2000 DEF PROCvalues
2010 MOVE 80,64
2020 FOR I=1 TO N
2030 INPUT "X,Y",X,Y
2040 X=80 + X*1100/XM
2050 Y=64 + Y*880/YM
2060 PLOT A,X,Y
2070 VDU5
2080 PLOT 0,-5,15
2090 PRINT CHR$(Z)
2100 VDU4
2110 PLOT 4,X,Y
2120 NEXT I
2130 ENDPROC
2200
3000 DEF PROCmore
3010 C=C+1
3020 ON C GOTO 3030,3040,3050,3070
3030 Z=111:GOTO 550
3040 Z=42:A=21:GOTO 550
3050 Z=111:A=21:GOTO 550
3070 ENDPROC
3090
5000 DEF PROCsdump
5010 ?(&80)=0:?(&82)=0:?(&83)=0
5020 ?(&81)=&30
5030 VDU1,27,1,65,1,7
5040 DIM CODE 500
5050 P%=CODE
5060 FOR I=0 TO 1
5070[OPT 2
5080 LDA#1:JSR&FFEE:LDA#27:JSR&FFEE
5090 LDA#1:JSR&FFEE:LDA#76:JSR&FFEE
5100 LDA#1:JSR&FFEE:LDA#128:JSR&FFEE
5110 LDA#1:JSR&FFEE:LDA#2:JSR&FFEE
5120 LDX#0:LDY#0
5130 .start LDA (&80),Y
5140 .loop CLC
5150 ROL A
5160 ROL &70,X
5170 INX
5180 CPX #7
5190 BNE loop
5200 LDX #0
5210 INY
5220 CPY #8
5230 BNE start
5240 LDX #0
5250 .char LDA#1:JSR&FFEE:LDA &70,X:JSR&FFEE
5260 INX:CPX #8
5270 BNE char
5280 LDX#0
5290 LDA#8:CLC:ADC&80:STA&80
5300 LDA#0:ADC&81:STA&81
5310 LDA#1:CLC:ADC&82:STA&82
5320 CMP#80:BCC start
5330 LDA#0:STA&82
5340 RTS:]
5350 NEXT
5360 FOR A=1 TO 32
5370 CALL CODE
5380 VDU1,13
5390 NEXT
5400 ENDPROC
```

Launch your BBC micro with the Torch Z80 disc pack.

If you already have a BBC Model B micro, then you know that it is not a toy. But neither is it suitable for business or for serious programming because it lacks the access speeds and massive storage capacity that only floppy discs can provide.

The answer is a TORCH Z80 DISC PACK. The twin discs store up to 800K of data and programs, while the Z80 processor board provides an extra 63K of user memory and Torch's CP/M* compatible operating system – giving access to the largest range of software currently available.

For the business user, this means sophisticated word processing, financial and stock-control packages. For the programmer who has outgrown BASIC, the TORCH DISC PACK can handle more compact, powerful, and specialised languages – such as PASCAL, C, FORTRAN, LISP and COBOL.

The TORCH Z80 DISC PACK comes complete with a range of useful software, including a full disc function test, a character redefinition program, a music generator, a Z80 monitor and system demonstrations.

TORCH gives you the power and programming space to launch your BBC micro into the world of serious computing.

TORCH
COMPUTERS

Abberley House, Great Shelford, Cambridge CB2 5LQ.
Telephone: (0223) 841000 Telex: 818841 TORCH G.

Lighting up Torch Discs

Henry Budgett

We examine the new disc system produced by Torch Computers and enlighten you on its price / performance capabilities.

Torch Computer's disc system for the BBC micro weighs in price-wise at £897 making it an expensive alternative to the BBC disc system. And in view of the fact that the average person on the street will have to get a dealer to install the system, is it really worth the money?

The BBC Micro's interfaces allow you to link it up to many of the 5¼" disc drives now commercially available, including the BBC's own at £265 for 100K disc drive, or £804 for a pair of 400K drives. These all access the BBC Micro's disc operating system, DFS.

Torch's disc system has beaten Acorn to the finishing post in producing an alternative system which allows access to the computer's CP/M, making it a much more useful and versatile system. By linking a Z-80 processor onto the BBC Micro's Tube the industry standard Disc Operating System (DOS) can be used.

Originally this project was to be undertaken by both Acorn and Torch together while they were producing a business computer system called the Torch based around the BBC Micro's main PCB (printed circuit board). Torch was to provide the software and Acorn was to come up with the hardware to link a Z-80 procesor to the machine.

But a few hiccups in the BBC project followed, meaning Acorn's team was unable to finish the Z-80 card in time. The partnership was dissolved and Torch went it alone.

The Torch computer that eventually resulted proved to be an excellent product. But how did the firm solve the problem of fitting a Z-80 card onto the BBC Micro, and what has that got to do with their disc pack?

Obviously once the enlightened Torch team had solved the Z-80 problem for their commercial system, they were free to use exactly the same technique to produce a disc pack for the BBC Micro. This pack would also incorporate a Z-80 processor and sufficient memory to allow the CP/M disc operating system to be used as an alternative to the BBC's DFS.

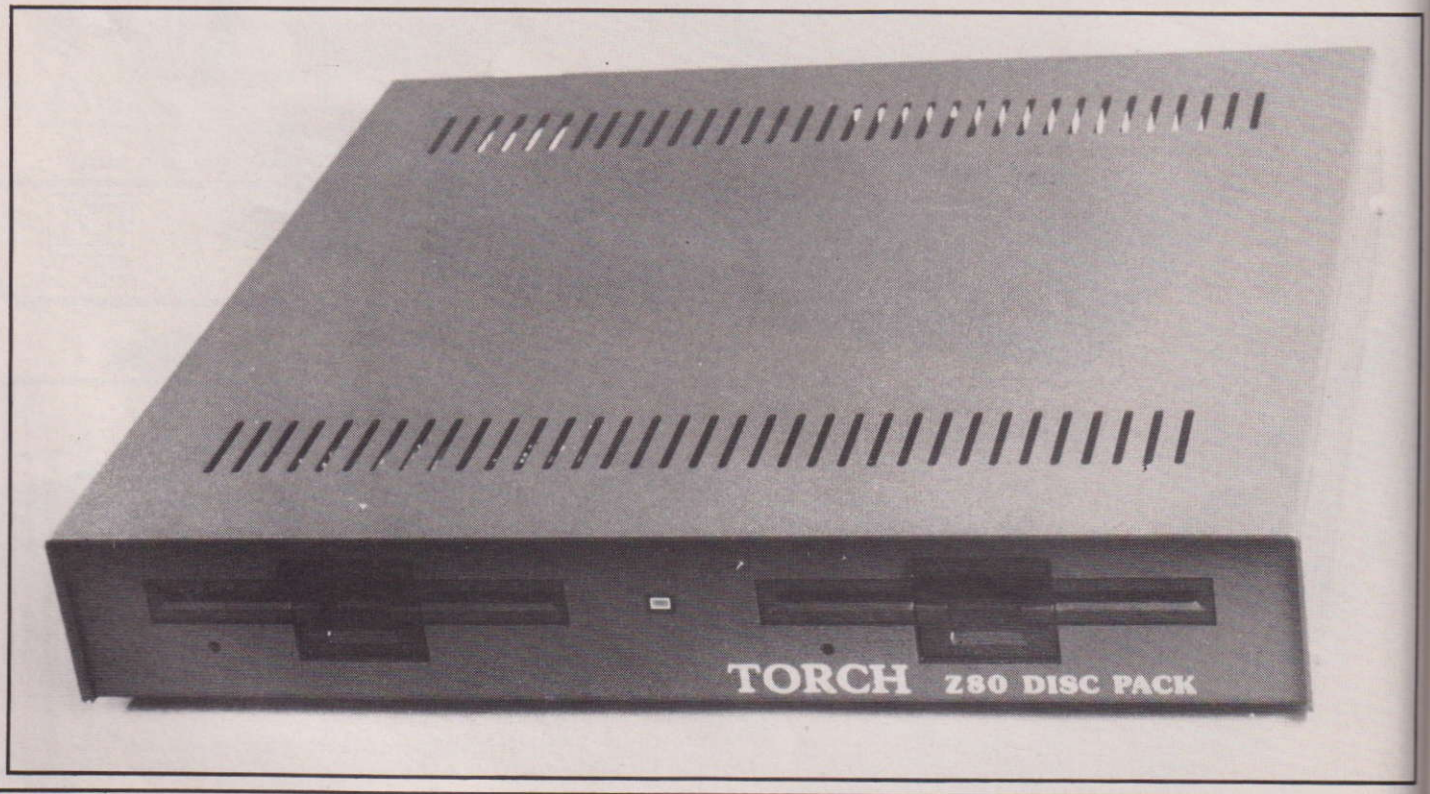
A LIGHT ON THE TORCH

The Torch Disc Pack comprises two 400K 5¼" half-height discs in a slimline brown case that fits neatly under the BBC Micro, a Eurocard sized PCB with the Z-80 and 64K of RAM which lives inside the computer's case, and a price tag of £897. So far, there are no proper manuals ready for the system apart from the Torch computer's own manuals and a small pile of photocopied sheets detailing the installation procedure along with a disc containing a number of utility programs.

Neither of our office BBC computers were fitted with the DFS interface (a vital requirement for the disc system even at £110!) so the entire conversion had to be done before I could set about the task in hand of testing the Disc Pack in action.

But here's a strong word of warning. *Don't* do it yourself unless you are brimming with confidence about installing integrated circuits because the process involves cutting PCB tracks and making wiring changes which can cause all sorts of problems. Your other options is to take it to your local dealers and get them to do it for you.

Although I had access to a BBC Hardware/Service manual



The torch disc system fitted to the BBC micro.

which isn't normally available to the public) the system's instructions proved to be insufficient for me to easily get everything up and running. In fact, the system eventually crashed.

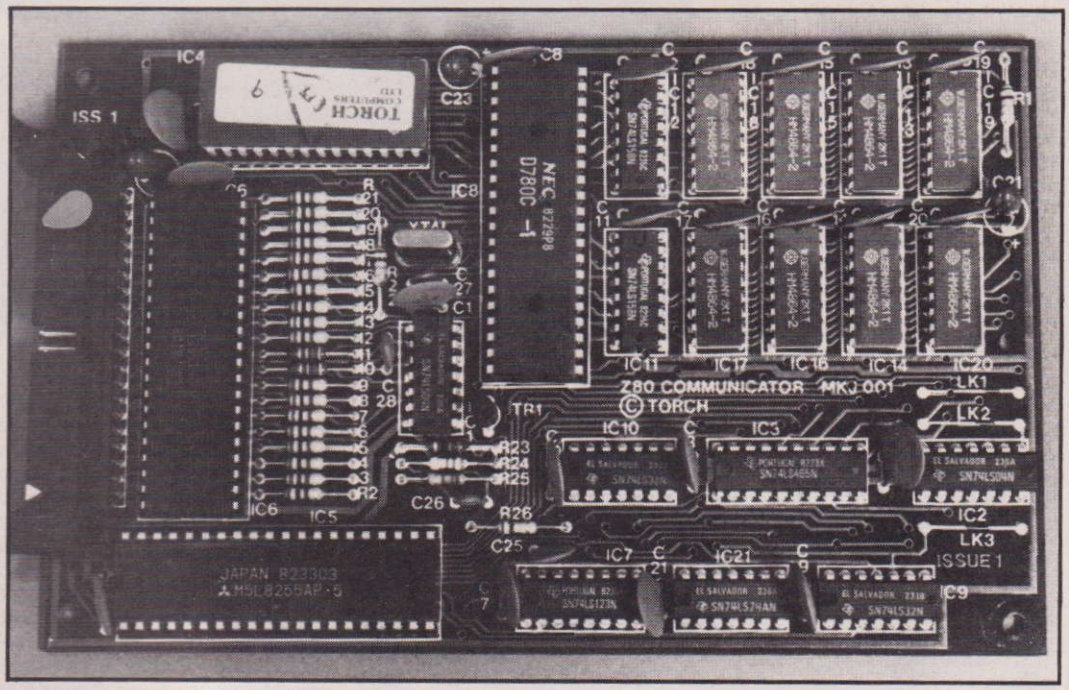
The disc interface went in with only two minor problems. The first was to do with the new Machine Operating System (MOS). You must have MOS 1.20 and this was supplied on a 'butterfly board' which would not fit because a two-pin connector sticking up from the BBC's PCB was in the way. I managed to overcome this problem by carefully filing a suitable notch in the butterfly board so the offending connector no longer fouled the board.

The second problem was that our BBC Micro has Issue 3 PCBs and these needed a wiring modification. Many of you will come up against the same problem. However, neither the manual or the sheets that come with the Torch Disc Pack makes it clear that when you link pin 9 of IC89 to pin 11 of IC78 you must first isolate them from the board by bending the legs out of their sockets and soldering a lead between them. One of these is a 40-pin chip — the disc controller — it's a pretty tricky task.

ASSEMBLY PROBLEMS

The Acorn DFS and its interface was installed and checked before the Torch Disc Pack was fitted, a move that was to prove wise in view of the events that followed. To start with our Disc Pack was supplied without its main lead, even though the packing list was checked off to show that it should be there, not a serious problem but frustrating none the less.

The real problems began when the installation got underway. Following Torch's instruction sheet the first job was to install the CPN (no, that's not a misprint but Torch's CP/M look-alike DOS) ROM into its correct socket. We now need to remove the BBC Microcomputer's power supply, all the necessary voltages are provided by the Disc Pack, no problems here except that the instructions tell you to put the power lead through the Econet hole but both logic and the grommet supplied indicate that they really mean the unused Reset hole and even this has to have the rear panel sticker scraped away to reveal its true size.



A bird's eye view of the board.

CONTINUED OVER

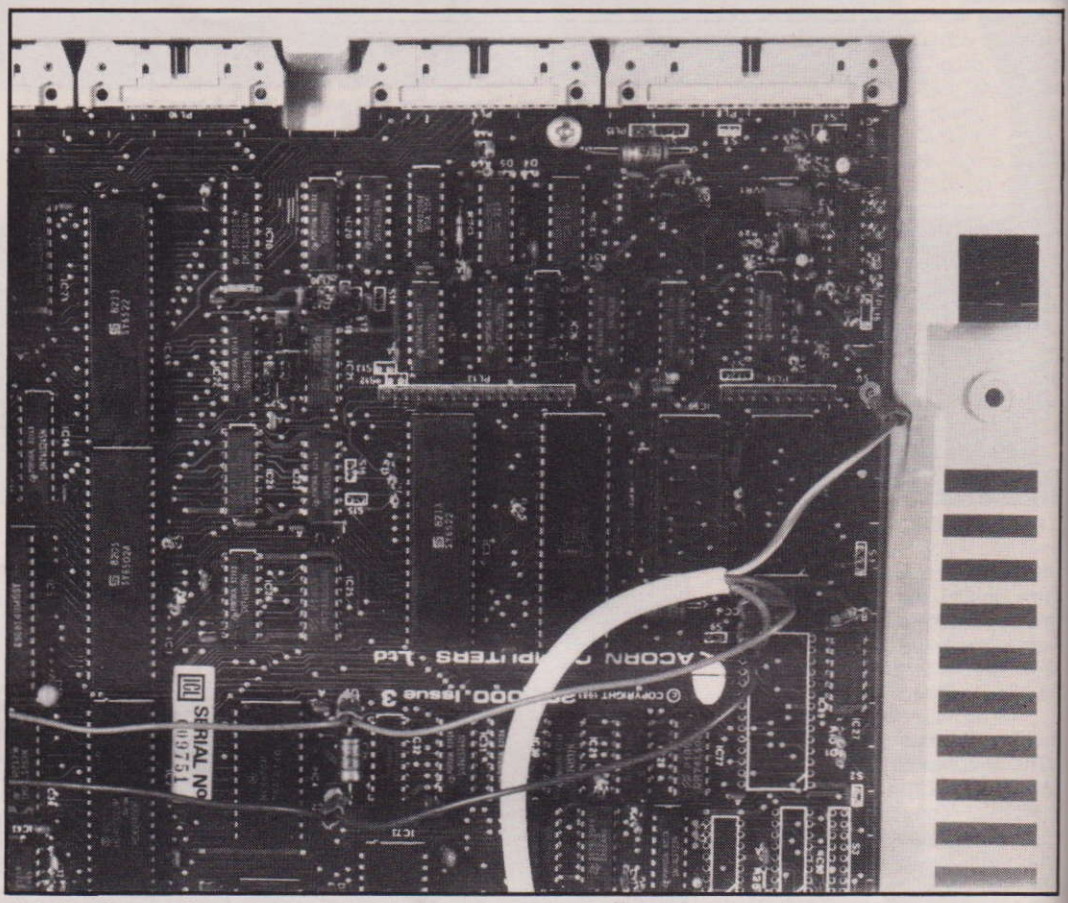
Power supply connections were made and checked, the next step was to fit the Z-80 card. Here we really started to run into trouble. The card itself has to be mounted onto the underside of the BBC Microcomputer's lid. No problems here. Then the short lead must be connected from the card to the Tube connector.

To start with there is not enough room to get the header plug out of the slot in the casing, you need to partially remove the BBC Microcomputers PCB to make it big enough but you can't do so because the new power cable and its grommet foul the back of the board. So you have to free these and get the cable out ready to mate with the Tube connector. The trouble here is that there is no indication of which way round the lead goes, there are no key marked, nothing in the instructions and the diagrams don't help at all. Logic dictates the most sensible way but unfortunately it was wrong!

The cable must have a twist in it to help it fold up inside the case and the user has a 50-50 chance of getting it wrong. With the board installed, hopefully correctly, the Disc Pack cable can be connected to the disc socket under the BBC Microcomputer. But again I found a total absence of indication as to which way up the cable should go. The cable goes upside down, at least according to the conventions laid down by the BBC's own disc drive, and again you've got a 50-50 chance of getting it right, the correct way is for the cable to be closest to the case and then loop down and back to the drives.

The white power cable is now plugged into the back of the Disc Pack, a suitable monitor is connected and power applied. You can forget using your TV set here as the default display Mode used is Mode 0 and televisions simply cannot handle 80 columns of text. Under normal circumstances you would get a double bleep and the screen would display the message:

```
TORCH Computer System (v71)
Acorn DFS
CPN
0A>
```



Circuitry close up 1.

In our case, nothing. After a few false starts and some time with a multimeter we finally resigned ourselves to the fact that our beloved micro had died. So it was time to call in the Torch doctors to carry out a few operations.

DEAD DISCS

Quite why the system gave up the ghost we are still unsure of but it is still undergoing major surgery. The only thing to do was to borrow a spare micro from the aptly named dealer Brainstorm who had supplied us with the disc system. Our grateful thanks go to them.

Once installed I discovered that the master disc supplied but untouched until that time was corrupted beyond salvation due to transit troubles.

With a new disc, a spare BBC Microcomputer and the original Disc Pack the nitty gritty of reviewing could at last begin. The time that this pandemonium and chaos had taken sadly left little time to really evaluate the full potential of the product but I trust that the following will convey some of its capabilities.

The Disc Pack provides the BBC Microcomputer with two 400K disc drives that can be accessed either under Torch's CPN operating system or the BBC's own DFS provided the ROM is fitted. The system's always boots-up in the Torch CPN format so one of the first commands that you learn is *BASIC (or just BASIC) which wil revert it to the DFS; to get back to CPN is simply a matter of typing *CPN.

UP AND RUNNING

The facilities available under CPN include FORMAT to prepare blank discs, DUP to copy discs, DIR to display the directory and TYPE to output the contents of a text or documentation file. Filenames can be up to eight characters long with an optional three character extension, DOC for documentation, TXT for text and DAT for data all being typical examples. Individual files on a disc can be copied with the COPY command and to make life easy a wildcard option is available where * represents any string of characters and ? represents any one character; COPY *DOC TO *TXT would copy all files with the DOC extension into files with the same name but a TXT extension.

Files may also be RENAMd or DELETEd as required. There are no provisions within CPN for directly LOADing or SAVEing programs in BBC BASIC. This is because when CPN is operational

the Z-80 processor is handling all the work and the BBC Microcomputer is just looking after the peripherals. To get BASIC of any sort running under CPN it must be loaded from disc, you can get at the BBC BASIC ROM but only if you either have the DFS ROM fitted or you revert to using tapes. If the DFS ROM is fitted then the system behaves exactly like a BBC disc drive and all the normal DFS functions are available.

As far as a user of commercially available software is concerned CPN is completely compatible with Digital Research's CP/M system which opens up a huge library of applications software and languages to the BBC Microcomputer user. Memory is no problem as the Z-80 card carries its own 64K of RAM and the CP/M programs live and run there, the BBC Microcomputer is simply used to look after the peripheral devices such as the keyboard, screen and printer.

I have indicated that you can use the BBC's own DFS on the Torch Disc Pack, but with the massive proviso that unless you have a BBC Utilities Disc (only provided with official BBC disc drives) all your blanks are going to be formatted to Torch standards and, therefore, cannot be successfully used with the BBC's own 40 track or 80 track drives.

PERFORMANCE AND DOCUMENTATION

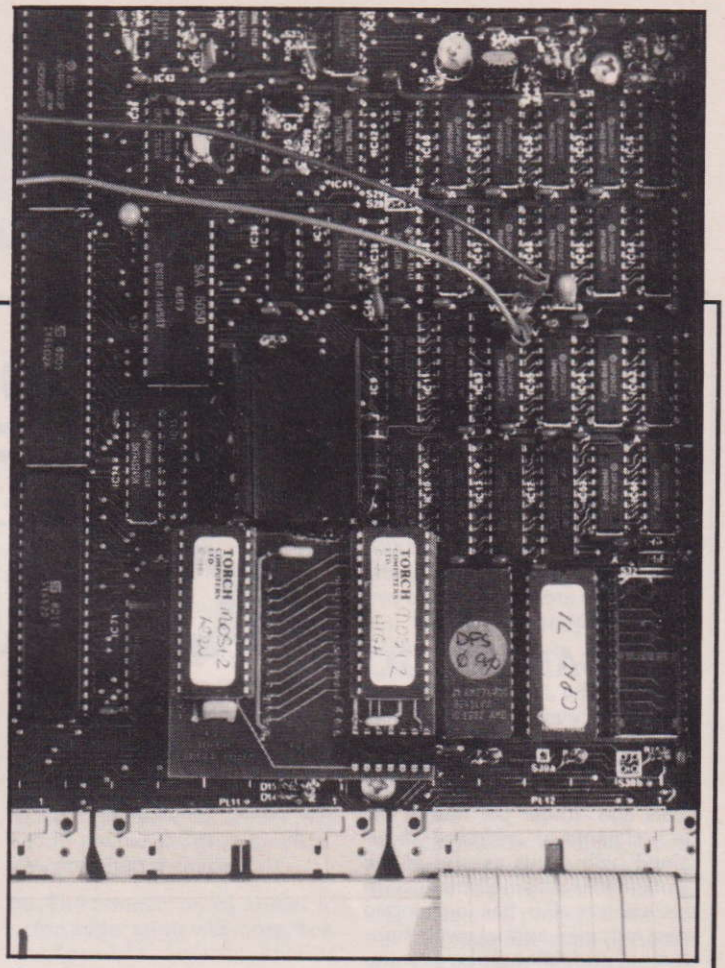
Torch have provided a pair of utility programs to read DFS format discs and save the files in CPN format and vice versa but because the BBC BASIC is tokenised, keywords like PRINT, INPUT, PROC and NEXT are reduced to single bytes or tokens, care has to be taken in how these are used. The entire explanation and instructions for these programs is less than a single screenful of text. Nothing relating to it could be found in any of the documentation supplied.

Sadly the trend of inadequate documentation that has befallen many a good product seems to have affected Torch as well. The Disc Pack is currently being supplied with manuals which relate to the Torch computer system. For example, my disc had neither Executives Aid or Cardex on it, which didn't really matter as I wasn't going to look at the software anyway, but the manuals came as standard. Now these may be good manuals as they stand, the Programmers Guide goes into great detail about the internal operation of CPN and the facilities the software writer could access, but there is very little here that the first time disc user will actually be able to relate to.

The BBC DFS Manual is poor in many respects but at least you can understand it, and these are simply impenetrable to anyone but a systems programmer. This may seem an odd point to raise but, as far as I can understand it, the purpose of buying a CP/M compatible disc system is so that the user will be able to run commercially available software and *not* worry about how to toggle his cache! What is needed is a new manual that covers the installation procedure thoroughly and correctly, although I do feel that this should not be attempted by the end user at all. This introduces the ideas of using discs as opposed to tape and illustrates the facilities and commands available under CPN.

The systems information could then be incorporated into a second manual which would allow Torch to get rid of all the duplicated material they are forced to include for their computer because they are not allowed to supply the BBC User Guide.

As a piece of hardware the Torch Disc Pack is excellently designed and very well put together. It opens up whole new horizons to both business and personal users of the BBC Microcomputer but it is badly let down by the lack of user friendly documentation and a horrendously complicated installation procedure. I would feel much happier if I knew that the money you pay for the Disc Pack included fitting by your local dealer and that owner installation was severely frowned on, possibly to the extent of being prevented completely.

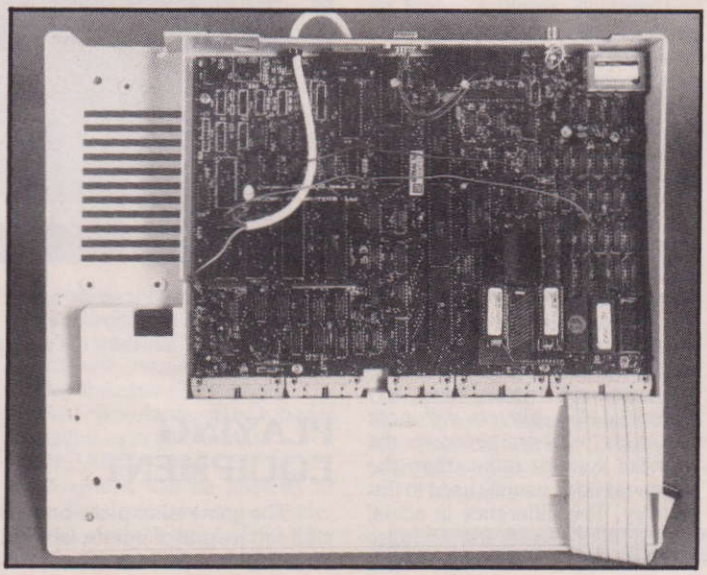


Circuitry close up 2.

In operation, provided one bears in mind its partial incompatability with BBC DFS format discs unless thy are 80 track and the DFS ROM is fitted, it proved faultless. There is one cosmetic blemish lurking in the design and that is that the drives do not always manage to eject the discs when the catch is release. I have no idea why as the problem is intermittent but annoying none the less.

Overall then, this is a product that turns the BBC Microcomputer from a small home computer system with a restricted memory capacity into a machine with considerable potential.

Where the board slots into the computer.



competition

Win £1000 in our software competition

ACORNSOFT

Acornsoft and *A&B Computing*, in co-operation with the British Go Association, are pleased to announce a competition to find the best Go-playing program for the BBC Microcomputer. The ultimate winner, to be decided by a play-off at the London Open Go Tournament in January 1984, will receive a cash prize of £1,000. Here the game of Go and the competition are introduced to you by Charles Matthews.

THE GAME OF GO

Go is the Japanese name for an ancient board game played all over the Far East, and increasing in popularity in the West. Originating in China about 3,000 years ago, under the name 'wei qu' or 'game of enclosing', it attained high status as one of the gentlemanly accomplishments. In this century Go has developed mass popular appeal, with hundreds of professional players competing in Japan, China, Korea and Taiwan. Here in Britain the level of amateur play has risen steadily over the last 20 years, with our representatives now regularly placed in the top ten places of the World Amateur Championship.

BRAIN BRAWN NEEDED

One of the attractions of Go, causing many players to prefer it to Chess, is the depth of intuition and strategic insight required by good players, on top of brute power of calculation. This aspect of the game makes it a real challenge to programmers who have yet to come up with an approach to machine Go-playing at a level beyond that of beginners. Now Acornsoft hope to stimulate the imaginations of owners of BBC Micros to tackle this task – see the full competitiion details below.

Since the game of Go may not be familiar to all readers of this magazine, here are the rules of Go and some explanatory hints. Experienced Go players will note that these rules are based on the Chinese system, rather than the Japanese code usually used in this country. The difference in actual play is slight, and the Chinese rules have definite advantages from a computing viewpoint. The competition will employ a board of size 13 x 13, rather than the standard 19 x 19 size.

PLAYING EQUIPMENT

The game takes place on a 13 x 13 square grid of points, labelled A1, A2, . . . up to M13. Traditionally these points are represented by the intersections of a 13 x 13 grid of lines, as illustrated in Fig. 1. The board is initially empty. Two players play alternately by occupying a vacant intersection with a piece (called 'stone') of their colour (Black or White, with Black starting).

You can play with pencil on squared paper, using noughts and crosses for 'stones' (you'll need a rubber for captures); for better equipment, a cardboard board and plastic stones from the BGA (see below) are convenient, or you can use simple graphics. Millionaires will order slate and shell stones and a *kaya* wood board straight from Japan!

SUMMARY OF THE RULES

The players play single stones alternately on vacant points, restricted only by a special rule ('ko') to prevent repetition of position. Once played the stones are never moved, but can be captured and are them immediately removed from the board. The objects of the game are (i) capture of enemy stones and (ii) occupation and encirclement of territory. A single stone is captured when it is directly surrounded along the lines by opposing stones. Two or more stones of the same colour which are joined together *along the lines* stand or fall together: the whole group is captured if it is directly surrounded along the lines. A player may pass at any time (in Japan, players are forbidden to pass as it is taken as an expression of contempt of the opponent's play, but this problem should not arise with machines!). The game ends when both players pass successively; each player then scores for the total of territory it/he/she has occupied or encircled, and the player with the higher score is the winner. (In tournament play, White receives 5½ points extra to compensate for starting second — this then prevents draws).

DEFINITIONS AND EXAMPLES

Liberties The liberties of a stone are those intersections adjacent to it along the lines of the ruled grid. For example, a stone at A1 B2 and A3, a stone at G7 has liberties at G6, G8, F7 and H7.

Connection Two stones of the same colour are directly connected if each occupies a liberty of the other; thus, stones at B2 and C2 are connected, while stones at B2 and C3 are not directly connected. A collection of stones of

competition

the same colour linked by direct connections is called an *army*.

Capture An army is captured when an enemy stone is played to fill the last liberty of the army. For example, with a Black stone at B2, White at A2, B3, C2, White captures B2 by playing at B1. With Black at B1, B2, C2, D1, D2 and White at A1, A2, B3, C3, D3, E1, Black captures A1, A2 by playing at A3, or White captures all five Black stones by playing at C1.

Suicide Playing your own army into a situation without liberties, is not forbidden under these rules, but the army is immediatly captured, and this would rarely have any point. As was illustrated above, a stone may be played into a position without liberties if this involves capturing an enemy army.

KO This special situation leads to the only rule forbidding play on a vacant point. In positions such as Black A2, White A1, B2, C1 or Black G7, H6, H8, White H7, I6, J7, Black can capture a single White stone (playing at B1 or I7), whereupon White could capture Black immediately, leading to a repetition of position.

The *rule* is: in a position where a *single stone* has been captured, and the capturing stone is subject to immediate recapture, this immediate recapture is forbidden for *one turn only*. In practice, if the recapture is important, the player makes a play in some other part of the board requiring an instant reply ('ko threat') and then recaptures — the boot is then on the other foot. Long sequences of ko threats and recaptures are a common and exciting part of the game, involving delicate calculations of profit and loss.

Safe Groups As the game proceeds, the board gradually fills up. The stones sort themselves into collections of armies which are potentially, if not directly, connected — what Go players call 'groups'. Beginners are often puzzled by the question 'if ultimately every group is surrounded on all sides, how is it possible to escape capture?"

A basic Go skill is to make groups from capture by ensuring they enclose enough empty territory. The principle behind every safe group is the same, and is known as 'two eyes'. For example, the Black group B1, D1, A2, B2, C2, D2, is safe (barring Black blunders), for White needs to play at A1 and C1 simultaneously (as well as at A3, B3, C3, D3, E1, E2) which is impossible. Change the colour of the stone at B2, and White threatens C2, D1, D2 — the 'eye' at C1 has become 'false'. False eyes cause more blood and tears to flow in beginners' Go than any other aspect of the game. A group unable to form two eyes will die.

End of Game We have decided the end of the game by both players passing. When will this happen, in practice? The end of the game can be recognised as follows: (a) neither player can capture further stones of the opponent, or (b) neither player can play further stones forming part of an actually or potentially safe group.

Roughly speaking, if I am to play and the game is really at an end, I see the board divided into the areas occupied by my opponent's groups (which I have reluctantly decided are safe) and the areas occupied by my own safe groups, inside which there is no point playing if they are safe from invasion and I have already captured all the 'stragglers' (opposing stones cut off in my territory and not part of a safe group).

At this stage, then, the spaces in 'no man's land' between the lines have been filled in. There may also, exceptionally, be 'impasse' situations which both players will leave unresolved. Then the scoring system, of one point for each point occupied and one point for empty intersection surrounded, may be put into effect, and the winner declared.

Those who are used to the Japanese scoring system will see that this Chinese-style scoring, replacing the counting of captives by counting for occupation of territory, will usually give a result differing by, at most, one point. It should be said that good players will always agree to play and the plays worth at most one point by mutual consent.

HINTS FOR PLAY AND PROGRAMMING

It is hoped that the illustrative game will be helpful in understanding how a game develops and ends. Here are a few basic points about armies, liberties and capture:

(i) Armies with one liberty may be captured, armies with two liberties are very vulnerable and may be subject by a forcing sequence to capture. (Simplest example, Black C2, White B2, C3, White plays at D2 and Black cannot ultimately save his stone.)

(ii) Saving threatened armies may lead to greater losses — small armies are often sacrificed.

(iii) The connection of armies to form a larger army with more liberties avoids losing stones piecemeal.

(iv) It is normally good to prevent the opponent's armies connecting, especially if one army is short of liberties.

Any good Go program must index armies, their liberties and some measure of their "vulnerability". (A crude approximation would divide number of stones by number of liberties). Forcing sequences of liberty-filling attacks on armies must be explored. Possible connections must be considered.

The territorial aspect of the game is equally important, but is likely to be much harder to program.

RULES FOR TOURNAMENT PLAY

(i) Programs should print out or display their own plays and point out all resultant captures. Programs will *not* be required to display the state of the game. The plays of opponents will be typed in to machines in the format:
A3 or PASS.

(ii) Programs will be allowed at most five minutes to consider plays — after the lapse of this time the program will be deemed to have passed.

(iii) Attempts to play illegal moves, such as occupation of an occupied point, or immediate recapture of a ko, will be deemed passes. Repeated offences may result in forfeiture of the game.

(iv) The tournament Komi will be 5½ points to White.

(v) The rules of Go as defined above will apply in the tournament.

(vi) The organisers will reserve the right to adjudicate any game at any stage, usually after 100 plays.

(vii) Matters concerning adjudication, interpretation of the rules of Go and determination of the relative merit of programs will be at the complete discretion of Acornsoft Ltd in consultation with the committee of the British Go Association.

THE COMPETITION

Entry to the competition will be on the form below, or a form obtained from Go Competition, Acornsoft Ltd, 4a Market Hill, Cambridge, CB2 3NJ. Entry forms must be returned to Acornsoft by 31st August, 1983.

Programs for consideration should be ready for submission to Acornsoft by 1st December, 1983, entrants will be invited to submit programs in the first two weeks of December, by post (on cassette or floppy disc) or in person (if so desired, but by appointment only).

All programs will be returned to authors, after consideration. At most 10 programs will be selected to play a tournament in the first week of January, 1984, in conjunction with the London Open Go Tournament. The authors of the selected programs will receive Acornsoft software of their choice of a value up to £50.

The declared winner of the

CONTINUED OVER

competition

tournament will receive £1,000 cash; the rights to the program and modifications of it will belong to Acornsoft Ltd, the author receiving a standard royalty from any sales.

Collective Entries: Entries of single programs from groups such as Go clubs, schools, etc, will be accepted if accompanied by a list of co-authors, having equal rights in the program, and the nomination of one co-author, with address and 'phone number, for contact and correspondence.

The British Go Association: They have fostered Go in this country for 20 years. They can put you in contact with Go players and clubs, and supply books and equipment. The Membership Secretary of the organisation is Derek Hunter, 60 Wantage Road, Reading, Berkshire.

Book Distributor: Andy Finch, 63a Russell Road, Moseley, Birmingham, B13. Recommended books are: 'Go for Beginners' (Penquin) by K. Iwamoto, 'Go Tutor' at £1 from the BGA. If you fancy having a traditional (it's adviseable to do so) game you can get a set of 361 plastic stones for £7 from them.

For the latest news and developments in this competition, leading up to a full report on the Grand Final in January, keep a regular eye on *A & B Computing*.

A QUIET OPENING (1-26)

Here is an explanation of the game played out here on paper for you to get to grips with strategic techniques. If you are already familiar with the game of Go then this will not be necessary for you, but may be useful to glance over.

Up to 9, the corners are occupied with stones on the third and fourth lines. The formation C4, E3 is a particularly solid way to enclose a corner. The early invasion 13 at K11 deprives Black of the corner. The next few plays show Black avoiding the trapping of his group, which would not be clearly safe.

The marked stone 26 is an indirect attack on the upper right Black group. If Black answered passively at G3, White would build a wall on the fourth line (say at E4) to threaten the upper group at a distance. Instead Black resists, and middle game fighting breaks out.

In the diagram 1, note that there are no stones on the first line, and only two on the second line. Observe also the close connection of stones where the two sides have come into contact, the looser connections elsewhere.

Black	White
1. J10	2. D11
3. C4	4. K3
5. D9	6. F10
7. E3	8. K8
9. J8	10. J7
11. K9	12. K7
13. I8	14. K11
15. L9	16. I12
17. H3	18. F8
19. C7	20. I7
21. G9	22. G10
23. G8	24. F9
25. G7	26. G4

CLOSE FIGHTING (27-62)

The long sequence in Diagram 2from 27 to 48 shows many direct threats to capture, with 44 the first actual capture of the game (notation '×2' for 'captures two stones'). Play 45 at E8 is necessary to save the eight Black stones. Play 48 at K10 cuts off two Black stones; these cannot form a safe group, but are only finally captured in the last few plays of the game. White regards the three isolated stones played in the sequence (E5, E7, G4) as sacrifices, with possible future uses. In the sequence, the combination of White H8 and I9 is a clever tactic — Black cannot reply by taking HB (why?).

After some territorial moves, fighting breaks out again at 62, which initiates a dangerous ko fight.

Black	White
27. H4	28. H5
29. G5	30. G6
31. F5	32. F6
33. F7	34. H6

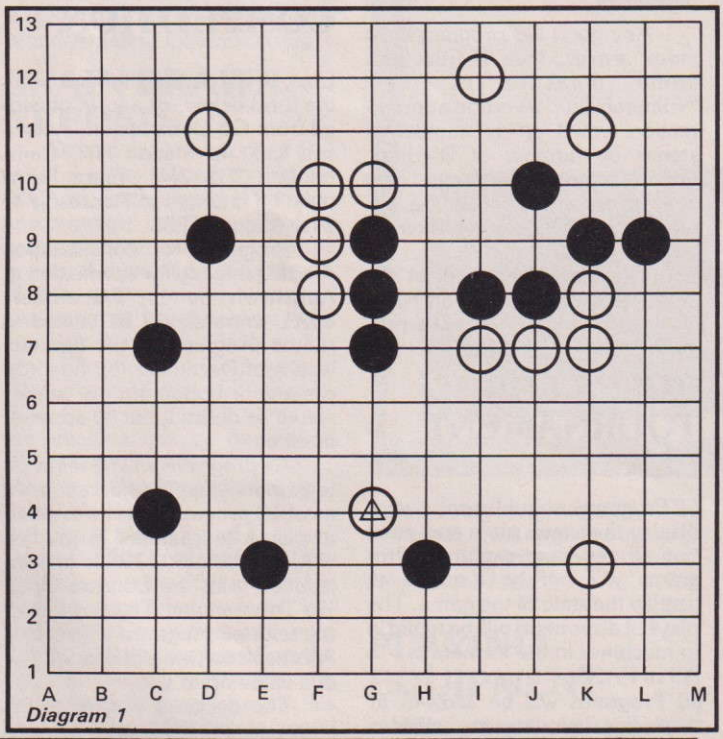


Diagram 1

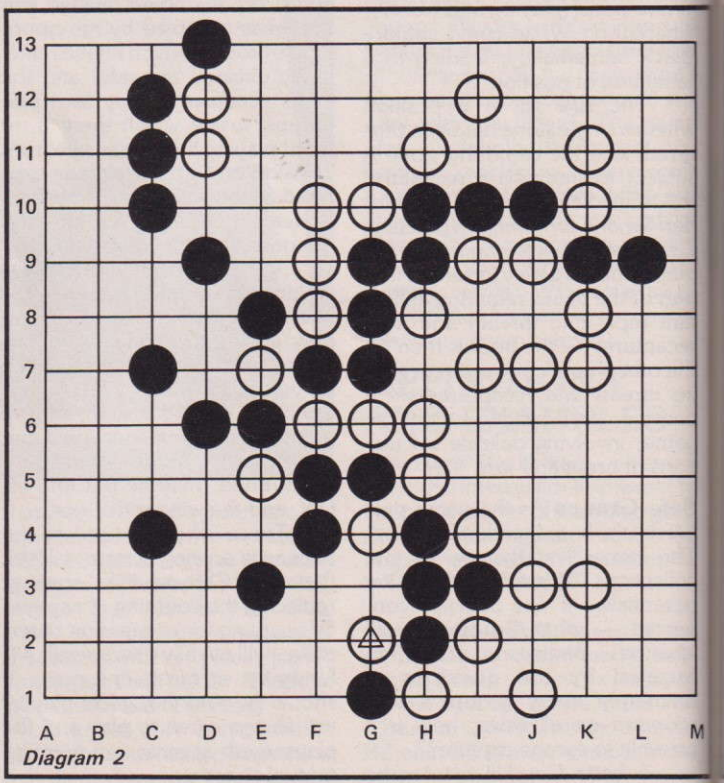


Diagram 2

competition

35. E6	36. E5
37. F4	38. E7
39. D6	40. H8
41. H9	42. I9
43. H10	44. J9 × 2
45. E8	46. H11
47. I10	48. K10
49. C10	50. I4
51. I3	52. J3
53. C11	54. D12
55. C12	56. I2
57. H2	58. J1
59. D13	60. H1
61. G1	62. G2

ENDING THE GAME (63-110)

The ko fight on the lower side is too important to last long. See Diagram 3. The threat 64 at E9 is answered, but White ignores 67, giving up stones on the upper side for most of the lower side. After 69 White has lost seven stones and some territory, but gains immediately by pushing into Black's corner; the exchange overall is reasonable. The presence of E5 means 71 at D2 would be a mistake, so each one of White's sacrificed stones plays a part in the ko fight.

A minor ko fight on the upper side (83-102) is the last feature of the game. The final plays 103-110 would be taken for granted among experienced players. White wins by three points.

This game is a reasonable example of British amateur play. It illustrates the ebb and flow of fighting in Go; as the ancient Chinese put it 'Empires wax and wane, states cleave asunder and coalesce'. Go has always been seen as a dynamic model of the highest level of strategic warfare.

Black	White
63. I1 × 1	64. E9
65. D7 × 1	66. H1 × 1
67. G11	68. G3 × 4
69. G12	70. E2
71. D3	72. F3
73. E4	74. D2
75. C2	76. H12
77. H13	78. J11
79. D1	80. E1
81. C1	82. I13
83. F11	84. G13 × 1
85. F13	86. H7
87. H13 × 1	88. I11
89. E7	90. G13 × 1
91. I8	92. J8 × 1
93. H13 × 1	94. C3
95. B3 × 1	96. G13 × 1
97. F1	98. F2 × 2
99. G13 × 1	100. D4
101. D5 × 2	102. G13 × 1
103. E9 × 5	104. H13
105. E12	106. L10
107. E11	108. L8
109. D10 × 2	110. M9 × 2
Pass	Pass

Score: Black 83
White 86

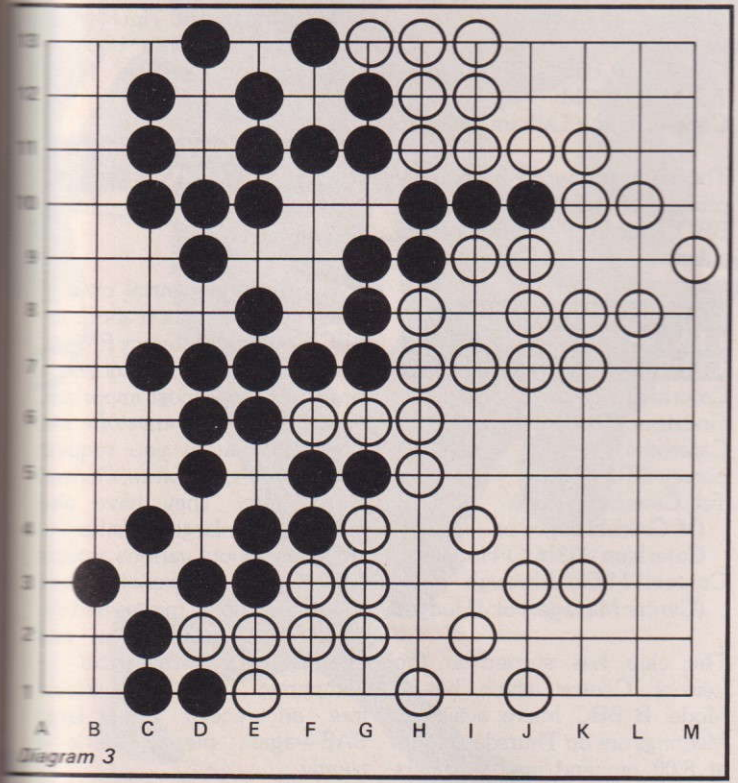


Diagram 3

CONDITIONS OF ENTRY

The competition is open to all UK and Northern Ireland readers of A & B Computing, except employees of Argus Specialist Publications Ltd, their printers and distributors, employees of Acornsoft Ltd, or anyone else connected with the competition.

As long as the correct coupon is used for each entry, there is no limit to the number of entries per person.

The entry form printed below must be returned to Acornsoft by August 31, 1983. Send them to GO COMPETITION, Acornsoft Ltd, 4a Market Hill, Cambridge, CB2 3NJ.

All program submissions must be sent to Acornsoft Ltd by December 16 1983 on cassette or floppy disc. All programs will be returned to the author after consideration.

Acornsoft's decision regarding the qualifying programs is final and neither Acornsoft nor the editor of A & B Computing will enter into any correspondence regarding this.

Group, or collective, entries (as in the case of a club entry) will be accepted providing the entry is accompanied with a list of co-authors who will have equal rights in the program. One of the co-authors must be nominated as a contact and supply both address and telephone number.

Acornsoft reserves the right to market the winning program.

Acornsoft reserves the right to market the winning program and retain the world copyright for it. Acornsoft will pay the winner either onan agreed royalty basis, or if the author so wishes the winning program will be bought for an additional agreed sum over and above the cash prize.

If the winner so wishes he/she can choose to use his/her cash prize to buy £1000 worth of equipment from AcornComputer and Acornsoft.

A & B COMPUTING/ACORNSOFT 'GO' COMPETITION

I wish to enter the competition in which I am required to write a program for the BBC Microcomputer to play the game of 'Go'. I accept the conditions of the competition as set out in the July/August issue of *A & B Computing* and agree to comply with the rules therein.

Name .....

Address .....

.....

.....

Telephone Number: day ..... evening .....

Signature .....

NOTE: This form does not mean that your 'Go' program has to be ready when you send this form in. It is to let the organisers know that you intend to enter the competition. If you don't send the form in your entry will NOT be accepted.

CLUB CORNER

Find about the hottest spots to learn about the BBC Micro. Are there any in your area?

It's easy to feel you are alone when you are sitting in front of your micro wondering why it won't do what you think you've instructed it to do! But you need never be alone again. User groups are springing up all over the place and can be an invaluable source of help, enjoyment and inspiration.

We list here some of the clubs that are particularly interested in the BBC Micro. If you don't see one in your area, why not start one up and let us have details? Please remember though, that this list is by no means comprehensive.

BEEBUG

The National Independent User Group for the BBC Micro
33 St. Julians Road,
St. Albans,
Hertfordshire.
Contact: D E Graham or Sheridan Williams

BEEBUG runs a regular newsletter (10 issues a year) including program listings, hardware and software tips, reviews and advice, all exclusively devoted to the BBC Micro. Membership is available for £4.50 for six months or £8.50 for the full year's subscription.

LASERBUG

10 Dawley Ride,
Colnbrook,
Slough,
Berkshire SL3 0QH.
Contact: Paul Barbour

LASERBUG started off life as a BBC User group based around London, but it is now an international group with members in 14 countries worldwide. A comprehensive magazine is sent to all members monthly. Local meetings are arranged via the newsletter. Annual membership is £12.00 which includes 12 copies of the magazine plus special members only discounts.

NORTHWICH & DISTRICT BBC MICRO USER GROUP

Room B12a,
Northwich City College of Higher Education,
Ipswich Road, Norwich,
Norfolk NR2 2LJ.
Tel: 0603-60011 ext 233
Contact: Paul Beverley

A local support group, membership will cost you £2.00 for the rest of the year unless you are a student or OAP in which case it will cost you £1.00.

COMPUTERTOWN UK!

7 Collins Drive,
Eastcote, Middlesex HA4 9EL.

A nationwide network of computer literacy centres. The idea started in the USA and was brought across and championed by David Tebbutt, the then Editor of *PCW*. Many of the local centres are doing sterling work for the BBC Project and, as the whole idea of CTUK! is to provide free access, they are well worth checking out.

COMPUTER USERS CLUB

72 Sidmouth Road,
Welling,
Kent DA16 1DS.
Tel: 01-304 3910
Contact: Tony Latham

The club produces a monthly printout of software ideas for the BBC Micro, programs and advice on programming technique.

CATERHAM LEISURE CENTRE COMPUTER CLUB

Caterham Leisure Centre,
Godstone Road,
Caterham,
Surrey CR3 6RE.
Tel: Caterham 48304 (M Goldsbrough) or Caterham 43316 (J Hodges)
Contact: M Goldsbrough (Centre Manager) or J Hodges

The club has started at the Leisure Centre which has a Model B BBC Micro available. Meetings are on Thursday nights at 8.00 pm and new members (and their micros) are welcome to come along.

THE FAREHAM AND PORTSMOUTH AMATEUR COMPUTER CLUB

22 Sandy Close,
Petersfield,
Hampshire GU31 4HF.
Tel: 0730-4059 (evenings)
Contact: Peter Smith

Established back in 1980, the Fareham and Portsmouth Amateur Computer Club have recently organised a referral service and a Users Club for the BBC Micro. The group meet at 7.00 pm on the third Monday of each month at the Portchester Community Centre.

INDEPENDENT BBC USERS' GROUP

Dept. 1A,
44-46 Earls Court Road,
London W8 6EJ.

This group has been in operation since October 1981, and is run by Jeremy Ruston and Tim Hartnell who have written a book of programs for the BBC Micro published by the BBC.

THE COMPUTER REFERRAL SERVICE

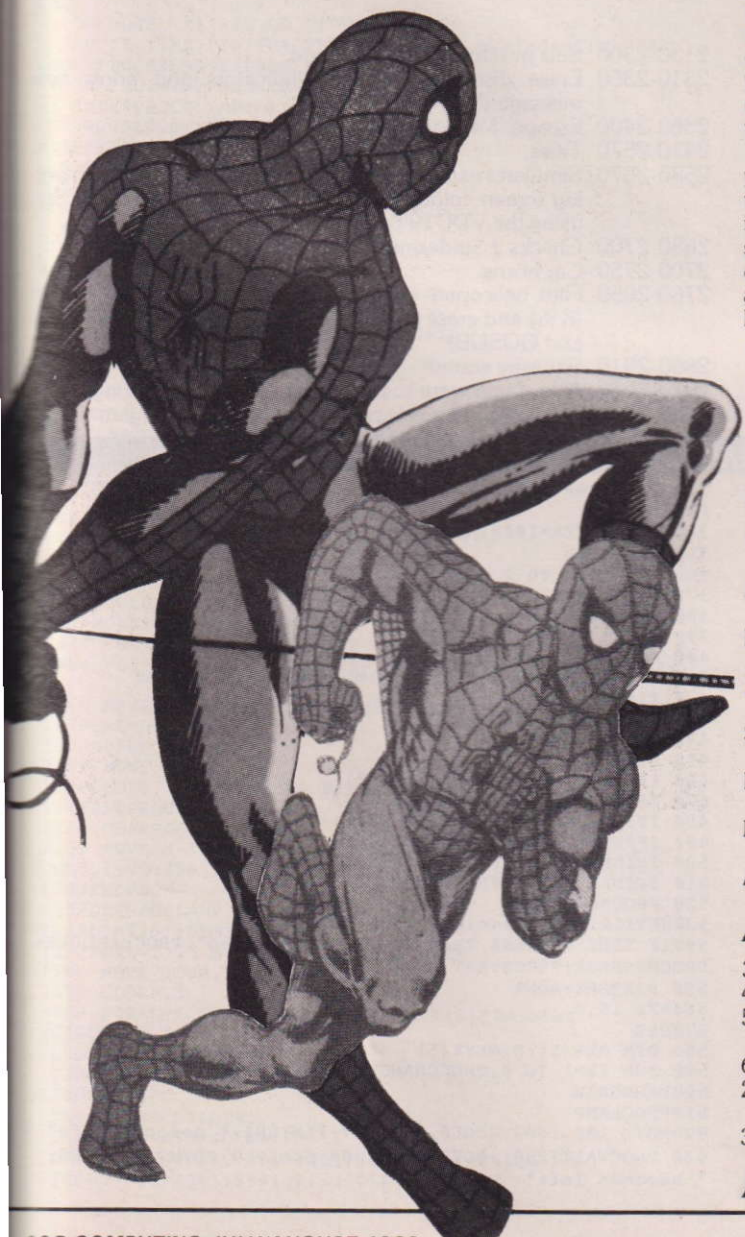
PO Box 7,
London W3 6XJ

This acts as a central clearing house for information about the BBC Computer Literacy Project. Please remember to enclose a large SAE and, most important, to write on your envelope just what information you require: User Groups, Software, General Query, etc. They have also published a large number of factsheets about various aspects of the project: regional groups, suggested books, micros and the small business, micros and education, and jobs in computing. These are available free on receipt of a large SAE—again please mark it clearly.

Spiderman

Richard Jones

Climb into the guise of one of the most reknowned superheros of our time and use your BBC to save the Metropolis. A great game for enthusiasts.



Find and defuse the bomb with only minutes to go till detonation time and do it in the way only Spiderman knows how. This time he's lucky as he has his BBC Micro with him — though that's a bit cumbersome to carry round as he stealthily leaps from rooftop to rooftop.

To help you, there is a Bomb Detector (key D) which tells you how far from the bomb you are and how much time you have left to find it. However, you slow down when you use it. To hinder you there are maniac drivers on the streets. You can move in any direction with the cursor keys. If you press two at once you will go diagonally — but you can't go through buildings!

When the Detector shows you are very close to the bomb, the scene changes and Spiderman is ready to run towards a skyscraper which he must climb to defuse the bomb on top. He will automatically climb if you move him left or right. All easy stuff — except for the huge spiders trying to squash him!

If Spiderman is killed, a helicopter will take him away. But if the time limit expires, the city is blown up.

The second part of the game features animated running and climbing figures which are achieved by alternating between different character strings which are themselves pre-defined characters printed on top of each other.

Speed has been increased by using integer variables and multi-statement lines. Many of the procedures are governed by parameters which make them multi-purpose e.g. PROCMAN (A,B) is a multi-directional movement procedure. Thus one procedure is capable of many on-screen effects.

Lastly, you have a panic button in the Space Bar on the grid, this sends you back to your starting point in the centre (not always a very safe place to be!).

VARIABLES

CARX%, CARY%,; CX%, CY%	Car positions.
P%	Pitch for sound of falling spiders.
DISTANCE%	Distance from bomb on grid.
NOW%	Time remaining.
MX%, MY%	Spiderman's position on grid.
FINISH%	Time limit.
X%, Y%	Spiderman's position in street scene.
SX%, SY%	Spiders position.
D%	Difficulty level.
M$	Message.
A$	Reply to "Another game?".

LINE DESCRIPTION

30	Changes base for random number generator.
40	Enables function key f0 to run program.
50	VDU5 enables one to PRINT a CHR$ at point on graphics screen.
60-250	Defines character set.
260-320	Strings containing graphics for running and climbing figures.
330-440	Initialise variables.

CONTINUED OVER

450-530	Loop for first part of game. Scans keyboard and moves figures according to the parameters set. Moves cars.
530-550	End of loop when position is near to bomb or time limit is reached, in which case relevant message displayed. Clock stopped and remaining time becomes new time limit.
560-650	Variables initialised for second part of game and procedures for drawing new scene.
660-770	Loop for second game. Keyboard scanned, Spiderman position checked, relevant animation procedure triggered. End of loop when position adjacent to bomb or time limit reached. Win procedure. Procedures defined.
780-870	Makes Spiderman climb up or down depending on value of A%.
880-970	Spiderman runs forward or backward depending on value of R%. Chooses correct graphics dependent on A$ and B$. Interspersed PROCSPIDERS keep movement on screen of several items more continuous.
980-1000	Delay loop.
1010-1140	Draw street and skyscraper.
1150-1240	Draw lampost for data.
1250-1350	Draw spiders (number depends on difficulty D%) RND (D%) keeps movement unpredictable.
1270	Note for each spider with pitch change.
1300	Checks if building or ground is reached.
1360-1400	Changes coordinates of spider to begin again at top of screen in random horizontal position.
1410-1430	Checks if spider has hit Spiderman.
1440-1490	Escape. Draws thread and moves S'man to safety.
1500-1530	Procedure if you win.
1540-1670	Draws street grid. Parameters passed to PROCBLOCK which draws triangles to make squares.
1680-1860	Sets new positions of cars.
1870-1980	Moves cars.
1910-1920	If car is off screen, sets new position.
1930-1940	Car horns if car is near Spiderman.
1990-2040	Moves figure around grid.
2020	Checks if figure is hitting a building.
2050-2120	Prints distance from bomb and remaining time.
2180	Formula to measure distance between two sets of polar coordinates —

$$\text{Distance between } P(X_1,Y_1) \text{ and } Q(X_2,Y_2) = \sqrt{(X_2-X_1)^2 + (Y_2-Y_1)^2}$$

2130-2300	Sets position of bomb on grid.
2310-2350	Erase distance and time indicators and prints new message (M$)
2360-2400	Escape. Moves figure to centre of grid.
2410-2570	Titles.
2580-2670	Simulates exploding bomb by sound effect and by changing screen colour (black) to white and back repeatedly, using the VDU19 palette changing command.
2680-2700	Checks if Spiderman is hit by car.
2700-2750	Car horns.
2760-2850	Flies helicopter on same horizontal line as Spiderman (K%) and erases him when they are adjacent (line 2800 and GOSUB).
2860-2910	Squashy sound!
2920-3010	Asks if you want to play again and acts accordingly.
3020-3210	Sets difficulty level and time limit, and starts game.

PROGRAM LISTING

```
>LIST
   10 MODE7
   20 PROCTITLE
   30X=RND(-TIME)
   40*KEY0 MODE4|M RUN|M
   50CLS:MODE2:VDU5
   60VDU23,224,140,140,68,38,30,6,38,86
   70VDU23,225,142,16,32,64,64,64,64,128
   80VDU23,226,48,48,16,152,120,24,24,24
   90 VDU23,227,24,56,80,80,80,80,208,48
  100VDU23,228,24,24,8,156,122,28,24,24
  110VDU23,229,24,56,68,131,65,32,16,48
  120VDU23,230,48,48,16,28,26,25,26,24
  130VDU23,231,56,88,24,24,20,18,17,48
  140VDU23,255,255,255,255,255,255,255,255,255
  150VDU23,232,24,24,16,57,94,56,24,24
  160VDU23,233,24,28,34,193,130,4,8,12
  170VDU23,234,12,12,8,56,88,152,88,24
  180VDU23,235,28,26,24,24,40,72,136,12
  190VDU23,237,36,24,90,60,255,153,165,165
  200 VDU23,238,24,24,60,126,60,60,126,60
  210VDU23,239,0,68,252,255,255,252,68,0
  220 VDU23,240,24,24,24,60,66,24,36,66,66
  230 VDU23,241,199,192,32,31,7,3,1,0
  240 VDU23,242,252,64,64,254,221,143,223,6
  250 VDU23,243,0,16,8,28,62,62,62,28
  260R1$=CHR$228+CHR$8+CHR$10+CHR$229
  270R2$=CHR$230+CHR$8+CHR$10+CHR$231
  280R3$=CHR$232+CHR$8+CHR$10+CHR$233
  290R4$=CHR$234+CHR$8+CHR$10+CHR$235
  300C1$=CHR$224+CHR$10+CHR$8+CHR$225
  310C2$=CHR$226+CHR$10+CHR$8+CHR$227
  320RUB$=CHR$255+CHR$8+CHR$10+CHR$255
  330 DIM CARX%(2):DIM CARY%(2):DIM CX%(2):DIM CY%(2)
,P%(5)
  340 DISTANCE%=1000:NOW%=0
  350 PROCGRID
  360 FOR C%=1 TO 2
  370 PROCCAR(C%)
  380 PROCCROSS(C%)
  390 NEXT
  400 PROCBOMB
  410 MX%=536:MY%=470:GCOL0,6:MOVE MX%,MY%:VDU240
  420 TIME=0
  430 FINISH%=FINISH%*100
  440NOW=FINISH%
  450 REPEAT
  460 IFINKEY(-58) PROCMAN(0,10)
  470 IFINKEY(-42) PROCMAN(0,-10)
  480 IFINKEY(-26) PROCMAN(-10,0)
  490 IFINKEY(-122) PROCMAN(10,0)
  500 IFINKEY(-51) PROCDISTANCE
  510 IFINKEY(-99) PROCPANIC
  520 PROCMOVECROSS
  530UNTILDISTANCE%<=10ORTIME>FINISH%
  540IF TIME>FINISH% THEN M$="Bomb exploded":PROCEXPLODE:
PROCMESSAGE:PROCSTART
  550 FINISH%=NOW%
  560*FX 15,0
  570CLS
  580 DIM SX%(5):DIMSY%(5)
  590 FOR I%=1 TO 4:PROCCHANGECOORDS(I%):NEXT
  600PROCDRAW
  610PROCLAMP
  620MOVE 100,1000:GCOL0,3:PRINT;FINISH%;" seconds left"
  630 PROCWAIT(200):MOVE 100,1000:GCOL0,0:PRINT;FINISH%;
" seconds left"
```

```
  640 X%=1072:Y%=100:GCOL0,6:MOVEX%,Y%:PRINTC2$
  650TIME=0:FINISH%=FINISH%*100
  660REPEAT
  670 PROCSPIDERS
  680 IF INKEY(-26) AND X%>512 AND Y%=100 THEN PROCRU
N(R1$,R2$,-20)
  690 IF INKEY(-26) AND X%=512 ANDY%<580 THEN PROCCLI
MB(20)
  700IF INKEY(-26) AND X%<=512 ANDY%=580 THEN PROCRUN
(R1$,R2$,-20)
  710 IF INKEY(-122) AND X%>=512 AND Y%=100 THEN PROC
RUN(R3$,R4$,20)
  720 IF INKEY(-122) AND X%<512 THEN PROCRUN(R3$,R4$,
20)
  730 IF INKEY(-122) AND X%=512 ANDY%<=580 THEN PROCC
LIMB(-20)
  740 IF INKEY(-99) AND X%=512 AND Y%>100 THEN PROCES
CAPE
  750 UNTIL X%<=72 OR TIME>FINISH%
  760 IF TIME>FINISH% THEN M$="Bomb exploded":PROCEXP
LODE:PROCMESSAGE:PROCSTART
  770 PROCWIN:PROCSTART
  780DEFPROCCLIMB(A%)
  790GCOL0,0:MOVE X%,Y%:PRINT RUB$
  800Y%=Y%+(A%)
  810 GCOL0,6:MOVE X%,Y%:PRINTC1$
  820PROCSPIDERS
  830 GCOL0,0: MOVE X%,Y%:PRINTC1$
  840Y%=Y%+(A%/2)
  850 GCOL0,6:MOVE X%,Y%:PRINTC2$
  860PROCSPIDERS
  870ENDPROC
  880DEFPROCRUN(A$,B$,R%)
  890GCOL0,0:MOVE X%,Y%:PRINT RUB$
  900X%=X%+R%:IFX%>1072 X%=1072
  910GCOL0,6:MOVE X%,Y%:PRINTB$
  920PROCSPIDERS
  930GCOL0,0:MOVE X%,Y%:PRINTB$
  940X%=X%+R%:IFX%>1072 X%=1072
  950GCOL0,6:MOVE X%,Y%:PRINTA$
  960PROCSPIDERS
  970ENDPROC
  980DEFPROCWAIT(Z)
  990FIN=TIME+Z:REPEAT UNTIL TIME>=FIN
 1000ENDPROC
 1010DEFPROCDRAW
 1020GCOL0,2
 1030MOVE1279,36:DRAW0,36
 1040 FOR Y%=516 TO 68 STEP -64:PROCBRICK(0,Y%):NEXT
 1050 FOR Y%=484 TO 100 STEP -64:PROCWINDOW(0,Y%):NEX
T
 1060 MOVE 0,548:GCOL0,7:VDU243
 1070ENDPROC
 1080DEFPROCBRICK(X,Y)
 1090 MOVE X-10,Y
 1100GCOL0,1:FOR B=1 TO 8:VDU255:NEXT
 1110ENDPROC
 1120 DEFPROCWINDOW(X,Y)
 1130 MOVE X-10,Y:GCOL0,1:VDU255,255:GCOL0,3:VDU 255:
GCOL0,1:VDU255,255:GCOL0,3:VDU255:GCOL0,1:VDU255,255
 1140ENDPROC
 1150DEFPROCLAMP
 1160DIMX(6):DIMY(6)
 1170FORL=1 TO 6:READ X(L),Y(L):NEXT
 1180 MOVE 1200,37
 1190 GCOL0,3
 1200 FOR L=1 TO 6:PLOT1,X(L)*20,Y(L)*20:NEXT
 1210GCOL0,7
 1220PLOT81,-20,40
 1230DATA 0,9,-2,2,-2,0,-1,-1,-1,-2,2,0
 1240ENDPROC
 1250DEFPROCSPIDERS
 1260FORQ%=1TORND(D%)
 1270 SOUND&0012,-10,P%(Q%),2:P%(Q%)=P%(Q%)-4
 1280MOVESX%(Q%),SY%(Q%):GCOL0,0:VDU237
 1290 SY%(Q%)=SY%(Q%)-20
 1300IFSY%(Q%)=540ANDSX%(Q%)<512 PROCCHANGECOORDS(Q%)
 1310IFSY%(Q%)=60 PROCCHANGECOORDS(Q%)
 1320PROCCHECK
 1330 MOVE SX%(Q%),SY%(Q%):GCOL0,1:VDU237
 1340 NEXT
 1350 ENDPROC
 1360 DEFPROCCHANGECOORDS(I%)
 1370 P%(I%)=181
 1380SY%(I%)=800
 1390 SX%(I%)=(RND(30)*20)+72
 1400 ENDPROC
 1410DEFPROCCHECK
 1420 IF SX%(Q%)>=X%-32 AND SX%(Q%)<=X%+32 AND SY%(Q%
)<=Y% THEN M$="Spiderman squashed":PROCMESSAGE:PROCSQ
UASH:PROCHELI(X%,Y%):PROCSTART
 1430 ENDPROC
 1440 DEFPROCESCAPE
 1450 MOVEX%,Y%:PLOT21,1072,100
 1460 MOVEX%,Y%:GCOL0,0:PRINTRUB$
 1470 X%=1072:Y%=100:MOVE X%,Y%
 1480GCOL0,6:PRINTC2$
 1490ENDPROC
 1500DEFPROCWIN
 1510 MOVE0,548:GCOL0,4:VDU243
 1520M$="Bomb safe":PROCMESSAGE:PROCSTART
 1530 ENDPROC
 1540 DEFPROCGRID
 1550 FOR X%=0 TO 1000 STEP 200
 1560 FOR Y%=0 TO 800 STEP 200
 1570 PROCBLOCK(X%,Y%)
 1580 NEXT:NEXT
 1590 ENDPROC
 1600DEF PROCBLOCK(X,Y)
 1610GCOL0,1
 1620MOVE X,Y
 1630 DRAWX+120,Y
 1640 PLOT85,X,Y+120
 1650 MOVE X+120,Y+120
 1660 PLOT85,X+120,Y
 1670 ENDPROC
 1680 DEFPROCCAR(C%)
 1690 CARY%(C%)=0-(RND(25)*20)
 1700 R=RND(5)
 1710 ON R GOSUB 1730,1750,1770,1790,1810
 1720 ENDPROC
 1730 CARX%(C%)=136:RETURN
 1740 CY%(C%)=168:RETURN
 1750 CARX%(C%)=336:RETURN
 1760 CY%(C%)=368:RETURN
 1770 CARX%(C%)=536:RETURN
 1780 CY%(C%)=568:RETURN
 1790 CARX%(C%)=736:RETURN
 1800 CY%(C%)=768:RETURN
 1810 CARX%(C%)=936:RETURN
 1820 DEF PROCCROSS(C%)
 1830 CX%(C%)=0-(RND(25)*40)
 1840 R=RND(4)
 1850 ON R GOSUB 1740,1760,1780,1800
 1860 ENDPROC
 1870 DEF PROCMOVECROSS
 1880 FORC%=1TO2
 1890MOVECX%(C%),CY%(C%):GCOL0,0:VDU239:MOVECARX%(C%)
,CARY%(C%):VDU238
 1900 CX%(C%)=CX%(C%)+40:CARY%(C%)=CARY%(C%)+20
 1910 IFCX%(C%)>=1100 PROCCROSS(C%)
 1920IFCARY%(C%)>=920 PROCCAR(C%)
 1930IFCY%(C%)>MY%-60 AND CY%(C%)<MY%+60 AND CX%(C%)>
MX%-200 AND CX%(C%)<MX%-160 PROCBEEP(41)
 1940IFCARY%(C%)>MY%-200 AND CARY%(C%)<MY%-160 AND CA
RX%(C%)>MX%-60 AND CARX%(C%)<MX%+60 PROCBEEP(41)
 1950MOVE CX%(C%),CY%(C%):GCOL0,2:VDU239:MOVECARX%(C%
),CARY%(C%):VDU238
```

CONTINUED OVER

PROGRAM LISTING

```
 1960PROCHIT(CX%(C%),CY%(C%)):PROCHIT(CARX%(C%),CARY%
(C%))
 1970 NEXT
 1980 ENDPROC
 1990 DEFPROCMAN(A,B)
 2000 MOVE MX%,MY%:GCOL0,0:VDU240
 2010 MX%=MX%+A:MY%=MY%+B
 2020 IF POINT(MX%+4,MY%)=1 OR POINT(MX%+52,MY%)=1 OR
 POINT(MX%+4,MY%-36)=1 OR POINT(MX%+52,MY%-36)=1 THEN
 MX%=MX%-A:MY%=MY%-B
 2030 MOVE MX%,MY%:GCOL0,6:VDU240
 2040 ENDPROC
 2050 DEF PROCDISTANCE
 2060 GCOL0,0:MOVE 100,1000:PRINT;DISTANCE%
 2070 GCOL0,0:MOVE700,1000:PRINT;NOW%;" secs"
 2080 DISTANCE%=((((BOMBX%-MX%)^2)+((BOMBY%-MY%)^2)))
DIV100:NOW%=(FINISH%-TIME)DIV100
 2090 GCOL0,2:MOVE 100,1000:PRINT;DISTANCE%
 2100 MOVE100,1000:PRINT;DISTANCE%
 2110 GCOL0,3:MOVE700,1000:PRINT;NOW%;" secs"
 2120 ENDPROC
 2130 DEF PROCBOMB
 2140 R=RND(2)
 2150 IF R = 2 GOTO 2240
 2160 R=RND(5): ON R GOSUB 2190,2200,2210,2220,2230
 2170 BOMBY%=RND(1000)
 2180 ENDPROC
 2190 BOMBX%=136:RETURN
 2200 BOMBX%=336:RETURN
 2210 BOMBX%=536:RETURN
 2220 BOMBX%=736:RETURN
 2230 BOMBX%=936:RETURN
 2240 R= RND(4):ON R GOSUB 2270,2280,2290,2300
 2250 BOMBX%=RND(900)
 2260 ENDPROC
 2270 BOMBY%=168:RETURN
 2280 BOMBY%=368:RETURN
 2290 BOMBY%=568:RETURN
 2300 BOMBY%=768:RETURN
 2310 DEFPROCMESSAGE
 2320MOVE100,1000:GCOL0,0:PRINT;DISTANCE%:MOVE700,100
0:PRINT;NOW%;" secs"
 2330 MOVE700,1000:PRINT;NOW;" secs"
 2340 GCOL0,2:MOVE100,1000:PRINTM$
 2350 ENDPROC
 2360DEFPROCPANIC
 2370 GCOL0,0:MOVE MX%,MY%:VDU240
 2380 MX%=536:MY%=470
 2390 GCOL0,6:MOVE MX%,MY%:VDU240
 2400ENDPROC
 2410DEF PROCTITLE
 2420 FOR T=1 TO 2
 2430 PRINTTAB(5);CHR$141;CHR$129;CHR$157;CHR$134;" S
 P I D E R M A N  ";CHR$156
 2440NEXT
 2450PRINT''CHR$134;"Spiderman must find the Atomic B
omb "
 2460 PRINTCHR$134;"hidden somewhere in New York"
 2470 PRINT
 2480 PRINTCHR$133;"Guide him round the blocks with t
he"
 2490 PRINTCHR$133;"cursor keys and your Detector (KE
Y 'D')"
 2500PRINTCHR$ 131;"When you have a reading of below
10 "
 2510PRINTCHR$ 131;"Spiderman is at the site of the b
omb"
 2520PRINTCHR$ 131;"which will then be displayed"
 2530PRINT'CHR$129;"Guide him left or right to avoid
the"
 2540PRINTCHR$129;"giant spiders protecting the bomb"
 2550PRINTCHR$129;"Spiderman will climb and save the
city!"
 2560 PROCDIFFICULTY
 2570ENDPROC
 2580 DEFPROCEXPLODE
 2590ENVELOPE1,1,1,0,0,10,0,0,127,-1,-1,-1,150,25
 2600SOUND0,1,4,20
 2610 FOR T%=1 TO 10
 2620 VDU19,0,7,0,0,0
 2630 PROCWAIT(5)
 2640 VDU19,0,0,0,0,0
 2650 PROCWAIT(5)
 2660NEXT
 2670 ENDPROC
 2680 DEFPROCHIT(X,Y)
 2690 IF X>=MX%-40 AND X<=MX%+40 AND Y>=MY%-30 AND Y<
=MY%+30 THEN M$="Spiderman run over":PROCMESSAGE:PROC
SQUASH:PROCHELI(MX%,MY%):PROCSTART
 2700ENDPROC
 2710 DEFPROCBEEP(P)
 2720 SOUND1,-10,P,1:SOUND2,-10,P+12,1
 2730 SOUND1,0,0,1:SOUND2,0,0,1
 2740 SOUND1,-10,P,4:SOUND2,-10,P+12,4
 2750 ENDPROC
 2760 DEFPROCHELI(J%,K%)
 2770 FOR H%=0 TO 1100 STEP 10
 2780 MOVE H%,K%:GCOL3,7:VDU241,242
 2790 MOVE H%,K%:GCOL3,7:VDU241,242
 2800IFH%<=J%+20ANDH%>=J%-20 GOSUB 2830
 2810 NEXT
 2820 ENDPROC
 2830IF H%<=MX%+20  AND H%>=MX%-20 GCOL0,0:MOVEJ%,K%:
VDU240
 2840IF H%<=X%+20 AND H%>=X%-20 GCOL0,0:MOVEJ%,K%:VDU
255,8,10,255
 2850RETURN
 2860 DEFPROCSQUASH
 2870SOUND0,-15,3,25
 2880FOR S%=144 TO 100 STEP-1
 2890SOUND1,0,S%,1
 2900NEXT
 2910ENDPROC
 2920 DEFPROCSTART
 2930 PROCWAIT(200)
 2940CLS
 2950 VDU4
 2960 PRINT TAB(0,5);"Another try (Y/N) ?":A$=GET$
 2970 IF A$="N" END
 2980 IF A$<>"Y" THEN 2960
 2990 VDU5:CLS
 3000 RUN
 3010 ENDPROC
 3020 DEFPROCDIFFICULTY
 3030 PRINTTAB(0,20);"How difficult do you want it ?
(1 - 5)"
 3040PRINT "(5 is hardest)"
 3050 D$=GET$:D%=VAL(D$)
 3060 IF D%<1 OR D%>5 THEN 3030
 3070 IF D%=1 FINISH%=210
 3080 IF D%=2 FINISH%=180
 3090 IF D%=3 FINISH%=150
 3100 IF D%=4 FINISH%=120
 3110 IF D%=5 FINISH%=90
 3120 CLS
 3130 *FX15,0
 3140 PRINT'''"You have ";FINISH%;" seconds to save N
ew York"
 3150PRINT'"In REAL trouble use the Space Bar"
 3160PRINT"for (hopefully) a miraculous escape"
 3170PRINT'''''"Press any key to begin"
 3180PRINT'';"f0 will RUN the program at any time"
 3190 *FX15,0
 3200 G$=GET$
 3210 ENDPROC
```

THE BBC FINDS ITS VOICE

J. Petit

Now that Acorn have given a British voice to the BBC Microcomputer it adds a whole new dimension to the machine. We take a look at this new sound chip and bring you a report on it.

The BBC Microcomputer's voice synthesis facility is now available and comes as a set of two chips which plug into sockets which are already available on the BBC Computers printed circuit board. A small modification is required to issue 1, 2 and 3 pcb's and a 1.2 operating system is necessary.

The first chip is a Voice Synthesis Processor (VSP) and the second chip is a Phrase Read Only Memory (PHROM).

The PHROM contains 165 phrases (Listed in Table 1). You will notice from an examination of this table that as well as numbers and letters there are also two tones, two pauses and a selection of words.

Those words which begin or end with the symbol "–", can be combined with other words to form new words. For instance "–ING" can be combined with "PRESS" to form the word "PRESSING".

Each phrase has an associated code and it is this code which is used to identify the phrase to the voice synthesis facility.

HOW IT WORKS

The mechanism which is used to pass phrases to the voice snythesis facility, is a modified SOUND command. For instance, the command SOUND–1,160,0,0 will pass the word 'ACORN' to the voice synthesis facility. The "–" identifies the SOUND command as a voice synthesis command and the "160" identifies the word ACORN. The last two parameters are for future enhancements and must be present for the command to be valid.

Some of the phrases in Table 1 are presented again in Table 2 where they have an alternative code. These are the ASCII codes and each phrase is shown with its ASCII code and associated keyboard symbol. The ASCII codes and phrases have been grouped together in such a way as to have meaningful relationships. For instance capital letters produce spoken letters and the numeric keys produce spoken numbers.

The mechanism for passing these phrases to the voice synthesis facility is similar to that for standard codes. For instance the command SOUND–1,ASC ("h"),0,0 will pass the word HUNDRED to the operating system.

You can see that the voice synthesis facility on the BBC Computer is very simple to use if a little limited in vocabulary. However, you should remember that eventually it will be possible to buy additional PHROMS and extend your vocabulary. These additional PHROMS will plug into a special socket which is located on the left hand side of your computer keyboard.

To show you how you might use this facility in your own programs, we will now consider a few simple routines.

PROGRAMS TO TRY OUT

Have a look at Program 1. With this program you will be able to generate most of the words in Table 2 just by pressing the appropriate key. Don't forget to try the lower case keys.

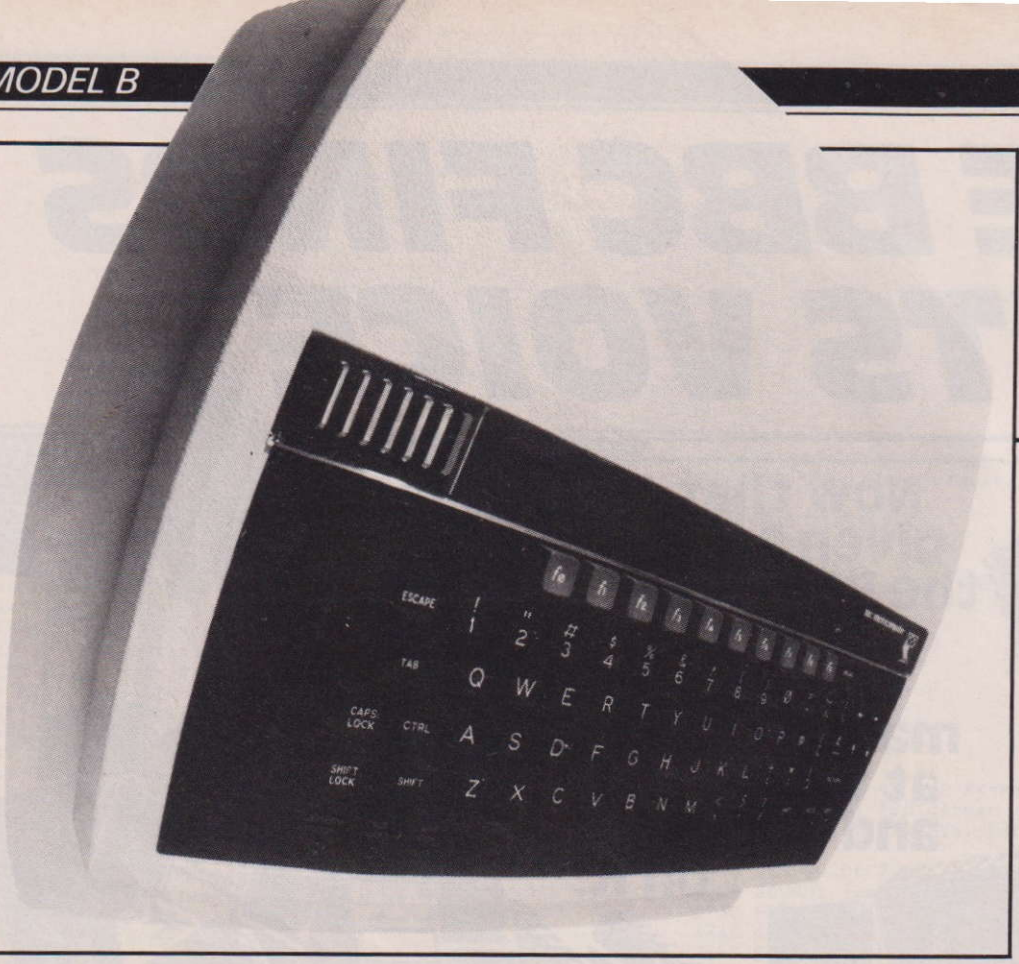
Line 10 waits for a key to be pressed and then puts the ASCII code for that key into a String Variable called WORD$. Line 20 uses this variable in a modified SOUND command and thus passes the phrases associated with the key which has been pressed to the voice synthesis facility where it is pronounced. Line 30 passes control back to Line 10 and the whole process is repeated.

SERIOUSLY SPEAKING

Now consider Program 2. This program will enable you to single step through each phrase in Table 1, and observe the code for each phrase as it is pronounced.

The program consists of a FOR___NEXT loop which is repeated for all values of A% from 127 up to 291. This is the full list of phrase codes from Table 1. Within the FOR___NEXT loop there is a wait for any key to be pressed, a command to print out the current value of A% and a modified SOUND command.

CONTINUED OVER



SENTENCE CONSTRUCTION

We are now ready to start joining words (Phrase Codes) together, to form sentences. Program 3 has a sentence encoded in the DATA Statement at Line 90. The program consists of a FOR____NEXT loop which is repeated seven times, one for each phrase code in the Data Statement. Within the FOR____NEXT loop the next phrase code in sequence is read from the DATA Statement and presented to a modified SOUND command. Thus the phrase codes in the DATA Statement are fed, in order, to the voice synthesis facility and the sentence is read out. At the end of the loop the DATA Pointer is RESTORED and a wait is imposed. Pressing any key will repeat the process.

Try and decode the sentence before you run this program. Then once you have run the program a few times, you should try and modify it to use a new sentence. Just change the codes in the dATA Statement and alter the Terminating Parameter in the FOR____NEXT loop to fit the number of codes you have put into the DATA Statement.

An easier way for users to form their own sentence is to use an array to store each phrase code as it is keyed in.

Consider Program 4. Line 10 Dimensions an array called STORE which will hold ten phrase codes, whilst Line 30 sets up a variable called POINTER which will point to the next empty slot in the array. The main body of the program consists of a REPEAT____UNTIL loop which is escaped when the user keys in "999" in response to the program prompt for more data. Within the loop the pointer is updated and a request is made for a single item of data.

Each item is put into the next empty slot of the array via a variable called WORD. Once the array is full, or the user terminates with a data item of "999" a FOR____NEXT loop is entered which will take each code in turn from the array and present it to a modified sound command. At the end of the FOR____NEXT loop the user is given the option of either repeating the sentence or keying in a new one.

Try this program out, don't forget that you have to enter the phrase codes from Table 1 and not the actual words. Alter the DIM statement in Line 10 if you want longer sentences.

Now let us have a look at a couple of serious applications.

Perhaps the most obvious application of voice synthesis is as spelling tutor. Indeed, one could argue that spelling tuition and testing on a computer, is not really feasible without voice synthesis. So let us have a look at a Spelling Test Program.

A selection of words have been chosen from Table 1 and are presented as a list called DATA FOR PROGRAMS 5 and 6. I have put each word and its associated phrase code in a separate Data Statement. This has been done for clarity only, and the words could be compressed into just a few Data Statements.

Program 5 is a very basic spelling list. The program consists of a FOR____NEXT loop which starts at Line 20 and finishes at Line 130.

Before the loop is entered, a variable called SCORE is set to zero. This variable is incremented by 1 with each correct spelling and a final score out of 10 is printed by 1 with each correct spelling and a final score out of 10 is printed on the screen at the end of the program (Line 230).

The nested FOR____NEXT loop consisting of lines 40,50 and 60 uses a Random Number (Between 1 and 32) generated at Line 30 to read the requisite data into variables WORD and WORD$.

At the end of this exercise the phrase code in variable WORD is used in a modified SOUND command to pronounce the word held in Variable WORD$. The user is then invited to key in the word which has been pronounced (Line 80).

If the user response is correct then the score is updated.

Irrespective of user response the Data Pointer is RESTORED and the process is repeated.

This is a very simple program, but it provides the basis of a very useful spelling test and tutor. Run the program a few times and think about the improvements you might make.

PROGRAM PROBLEMS

There are three major problems with this program:

1) There are no instructions for the user.
2) There is no attempt to correct bad spellings.
3) Words can be repeated within a 10 word session.

Program 6 is a modified version of Program 5 and overcomes the stated problems as follows!

1) Instructions for the user are provided at Line 28.
2) Wrong spellings are corrected at Line 180.
3) The third problem is overcome by using an array called WORD-USED to store the words which have already been spoken. The main loop now uses a control variable called TURN and each time through the loop this control variable is used to store the current word in the array. Thus each time a new word is extracted from the Data Statements a search can be made to see whether that word has been used before.

Program 6 is our first working example of voice synthesis. It is a very basic program and therefore easy to understand. However it is open to considerable refinement and should provide you with a lot of interesting work.

SECOND UP

Our second working example is a Talking Clock. Program 7 has been developed from the Digital Clock program in the BBC MICROCOMPUTER SYSTEM USER GUIDE (Page 131).

The two horizontal lines in Program 8 show how the basic Digital Clock program fits into the

Talking Clock program. All the necessary words for a talking clock are embedded in Data Statements at the end of the program. For ease of use these words are put into two arrays (Lines 10 to 80).

When the program runs Lines 110 to 180 will prompt for the current time, after which the clock (Lines 190 to 260) runs continuously.

Within the clock loop an escape mechanism is activated if any key on the keyboard is pressed (Lines 240 and 250).

When a key is pressed control is passed to procedure VOICE where the variables HOUR and MINUTE are used to pronounce the time.

At the end of the Procedure, control is passed back to the clock loop, ready for another key to be pressed.

Again a very basic programme with a lot of development potential.

PROGRAM LISTINGS

```
  1  REM *****PROGRAM-1*****
 10  WORD$=GET$
 20  SOUND-1,ASC(WORD$),0,0
 30  GOTO 10
 40  REM ********************
  1  REM *****PROGRAM-2*****
 10  FOR A%=127 TO 291
 20    B=GET
 30    PRINT A%
 40    SOUND-1,A%,0,0
 50    NEXT
 60  END
 70  REM ********************
  1  REM *****PROGRAM-3*****
 10  FOR MESSAGE=1 TO 7
 20    READ WORD
 30    SOUND-1,WORD,0,0
 40    NEXT
 50  RESTORE
 60  GO=GET
 70  GOTO 10
 80  END
 90  DATA 205,164,267,170,170,
     175,179
100  REM ********************
  1  REM *****PROGRAM-4*****
 10  DIM STORE(10)
 20  CLS
 30  POINTER=0
 40  REPEAT
 50    POINTER=POINTER+1
 60    PRINT
 70    INPUT"WORD",WORD
 80    STORE(POINTER)=WORD
 90    UNTIL WORD=999 OR POINTER=10
100  FOR A%=1 TO POINTER-1
110    SOUND-1,STORE(A%),0,0
120    NEXT
130  INPUT"REPEAT. YES OR NO",ANSWER$
140  IF ANSWER$ ="YES" GOTO 100
150  GOTO 20
160  REM ********************
  1  REM *****PROGRAM-5*****
 10  SCORE=0
 20  FOR TURN=1 TO 10
 30    NUMBER=RND(32)
 40    FOR A%=1 TO NUMBER
 50      READ WORD,WORD$
 60      NEXT
 70    SOUND-1,WORD,0,0
 80    INPUT ENTRY$
 90    IF ENTRY$<>WORD$ PRINT
     "WRONG":GOTO 120
100    PRINT "CORRECT"
110    SCORE=SCORE+1
120    RESTORE
130    NEXT
140  PRINT "SCORE=";SCORE;"
     OUT OF 10"
240  END
250  DATA 180,CORRECT
260  DATA 140,HUNDRED
270  DATA 206,ILLEGAL
280  DATA 252,RETURN
290  DATA 285,WHICH
300  DATA 229,NUMBER
310  DATA 141,THOUSAND
320  DATA 160,ACORN
330  DATA 155,EIGHT
340  DATA 153,SEVEN
350  DATA 163,AMOUNT
360  DATA 166,ANOTHER
370  DATA 185,DOLLAR
380  DATA 190,ELEVEN
390  DATA 191,ENGAGED
400  DATA 194,ESCAPE
410  DATA 219,MINUS
420  DATA 224,NEGATIVE
430  DATA 239,PARAMETER
```

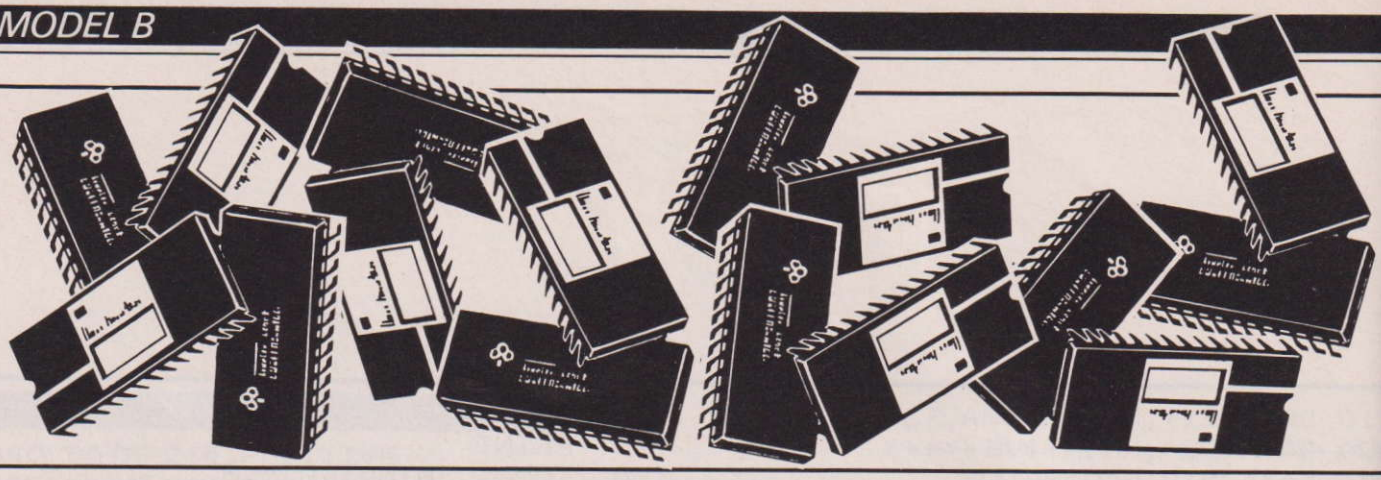
CONTINUED OVER

```
440 DATA 244,POSITIVE
450 DATA 254,RUNNING
460 DATA 258,SECOND
470 DATA 262,SWITCH
480 DATA 273,TWELVE
490 DATA 177,CHARACTER
500 DATA 178,COMPLETE
510 DATA 179,COMPUTER
520 DATA 176,CASSETTE
530 DATA 172,BETWEEN
540 DATA 174,BUTTON
550 DATA 169,AVAILABLE
560 DATA 167,ANSWER
570  REM **********
      ***DATA FOR PROGRAMS 5&6***
250 DATA 180,CORRECT
260 DATA 140,HUNDRED
270 DATA 206,ILLEGAL
280 DATA 252,RETURN
290 DATA 285,WHICH
300 DATA 229,NUMBER
310 DATA 141,THOUSAND
320 DATA 160,ACORN
330 DATA 155,EIGHT
340 DATA 153,SEVEN
350 DATA 163,AMOUNT
360 DATA 166,ANOTHER
370 DATA 185,DOLLAR
380 DATA 190,ELEVEN
390 DATA 191,ENGAGED
400 DATA 194,ESCAPE
410 DATA 219,MINUS
420 DATA 224,NEGATIVE
430 DATA 239,PARAMETER
440 DATA 244,POSITIVE
450 DATA 254,RUNNING
460 DATA 258,SECOND
470 DATA 262,SWITCH
480 DATA 273,TWELVE
490 DATA 177,CHARACTER
500 DATA 178,COMPLETE
510 DATA 179,COMPUTER
520 DATA 176,CASSETTE
530 DATA 172,BETWEEN
540 DATA 174,BUTTON
550 DATA 169,AVAILABLE
560 DATA 167,ANSWER
      **********
```

```
  1  REM *****PROGRAM-6*****
 10 DIM WORDUSED(10)
 20 SCORE=0
 25 FLAG=0
 26 CLS
 27 PRINT:PRINT:PRINT
 28 PRINT"SPELL THE WORDS AS THEY
    ARE SPOKEN BY   THE COMPUTER."
 30  FOR TURN=1 TO 10
 50   NUMBER=RND(32)
 60   FOR B%=1 TO TURN
 70 IFWORDUSED(B%)=NUMBER THEN FLAG=1
 80     NEXT
 90   IF FLAG=1 THEN FLAG=0:GOTO 50
100   WORDUSED(TURN)=NUMBER
130   FOR A%=1 TO NUMBER
140     READ WORD,WORD$
150     NEXT
160   SOUND-1,WORD,0,0
170   INPUT ENTRY$
180   IF ENTRY$<>WORD$PRINT"WRONG
THE CORRECT SPELLING IS (";WORD$;")
":GOTO21(
190    PRINT"CORRECT"
200   SCORE=SCORE+1
210   RESTORE
220   NEXT
230 PRINT"SCORE=";SCORE;" OUT OF 10"
240 END
250 DATA 180,CORRECT
260 DATA 140,HUNDRED
270 DATA 206,ILLEGAL
280 DATA 252,RETURN
290 DATA 285,WHICH
300 DATA 229,NUMBER
310 DATA 141,THOUSAND
320 DATA 160,ACORN
330 DATA 155,EIGHT
340 DATA 153,SEVEN
350 DATA 163,AMOUNT
360 DATA 166,ANOTHER
370 DATA 185,DOLLAR
380 DATA 190,ELEVEN
390 DATA 191,ENGAGED
400 DATA 194,ESCAPE
```

```
410 DATA 219,MINUS
420 DATA 224,NEGATIVE
430 DATA 239,PARAMETER
440 DATA 244,POSITIVE
450 DATA 254,RUNNING
460 DATA 258,SECOND
470 DATA 262,SWITCH
480 DATA 273,TWELVE
490 DATA 177,CHARACTER
500 DATA 178,COMPLETE
510 DATA 179,COMPUTER
520 DATA 176,CASSETTE
530 DATA 172,BETWEEN
540 DATA 174,BUTTON
550 DATA 169,AVAILABLE
560 DATA 167,ANSWER
570  REM **********
  1  REM *****PROGRAM-7*****
110  PRINT "PLEASE INPUT THE TIME"
120  PRINT
130  INPUT "HOURS ",H
140  PRINT
150  INPUT "MINUTES ",M
160  TIME=H*360000+M*6000
170  VDU 23;8202;0;0;0;
180  CLS
190  REPEAT
200    SECOND=(TIME DIV 100)MOD 60
210    MINUTE=(TIME DIV 6000)MOD 60
220    HOUR=(TIME DIV 360000)MOD 12
230    PRINT TAB(7,12)HOUR;";"
       ;MINUTE;":";SECOND;" "
240    UNTIL FALSE
250  REM **********
  1  REM *****PROGRAM-8*****
 10 DIM UNI(12)
 20 DIM MULTI(10)
 30 FOR A%=0 TO 12
 40   READ UNI(A%)
 50   NEXT
 60 FOR A%=1 TO 9
 70   READ MULTI(A%)
 80   NEXT
 90 CLS
100 PRINT
110  PRINT "PLEASE INPUT THE TIME"
120  PRINT
130  INPUT "HOURS ",H
140  PRINT
150  INPUT "MINUTES ",M
160  TIME=H*360000+M*6000
170  VDU 23;8202;0;0;0;
180  CLS
190  REPEAT
200    SECOND=(TIME DIV 100)MOD 60
210    MINUTE=(TIME DIV 6000)MOD 60
220    HOUR=(TIME DIV 360000)MOD 12
230    PRINT TAB(7,12)HOUR;
       ";";MINUTE;":";SECOND;" "
240   VOICEOUT=INKEY(10)
250   IF VOICEOUT<>-1 PROCVOICE
260    UNTIL FALSE
270 DEFPROCVOICE
280 SOUND-1,267,0,0
290 SOUND-1,271,0,0
300 SOUND-1,209,0,0
310 IF HOUR=0 SOUND-1,273,0,
    0:GOTO 330
320 SOUND-1,UNI(HOUR),0,0
330 IF MINUTE=0 SOUND-1,231,0,
    0:GOTO 420
340 FIRST=MINUTE MOD 10
350 OTHER=MINUTE DIV 10
360 IF OTHER=0 SOUND-1,
    UNI(FIRST),0,0:GOTO 420
370 IF OTHER>1 SOUND-1,MULTI(OTHER)
    ,0,0:SOUND-1,137,0,0:SOUND-1,
    UNI(FIRST),0,0
:GOTO 420
380  IF FIRST=0 SOUND-1,264,
    0,0:GOTO 420
390  IF FIRST=1 SOUND-1,190,
    0,0:GOTO 420
400  IF FIRST=2 SOUND-1,273,
    0,0:GOTO 420
410 SOUND-1,MULTI(FIRST),0,
    0:SOUND-1,135,0,0
420 ENDPROC
430 DATA 127,142,143,145,147,149,
    151,153,155,157,264,190,273
440 DATA 0,144,146,148,150,152,154,
    156,158
450  REM **********
```

CONTINUED OVER



CODE	WORD
127	0.125(SHORT PAUSE)
128	0.25(LONG PAUSE)
129	TONE1
130	TONE2
131	-D
132	-ED
133	-ING
134	-S
135	-TEEN
136	-TH
137	-TY
138	-Z
139	0
140	HUNDRED
141	THOUSAND
142	1
143	2
144	TWEN-
145	3
146	THIR-
147	4
148	FOUR-
149	5
150	FIF-
151	6
152	SIX-
153	7
154	SEVEN-
155	8
156	EIGH-
157	9
158	NINE-
159	A
160	ACORN
161	AFTER
162	AGAIN
163	AMOUNT
164	AN
165	AND
166	ANOTHER
167	ANSWER
168	ANY
169	AVAILABLE
170	B
171	BAD
172	BETWEEN
173	BOTH
174	BUTTON
175	C
176	CASSETTE
177	CHARACTER
178	COMPLETE
179	COMPUTER
180	CORRECT
181	D
182	DATA
183	DATE
184	DO
185	DOLLAR
186	DONT
187	DOWN
188	E
189	EACH
190	ELEVEN
191	ENGAGED
192	ENTER
193	ERROR
194	ESCAPE
195	F
196	FEW
197	FILE
198	FIRST
199	FOUND
200	FROM
201	G
202	GOOD
203	H
204	HAVE
205	I
206	ILLEGAL
207	IN-
208	INPUT
209	IS
210	J
211	K
212	KEY
213	L
214	LARGE
215	LAST
216	LINE
217	M
218	MANY
219	MINUS
220	MORE
221	MUST
222	N
223	NAME
224	NEGATIVE
225	NEW
226	NO
227	NOT
228	NOW
229	NUMBER
230	O
231	O'CLOCK
232	OF
233	OFF
234	OLD
235	ON
236	ONLY
237	OR
238	P
239	PARAMETER
240	PENCE
241	PLEASE
242	PLUS
243	POINT
244	POSITIVE
245	POUN-
246	PRESS
247	PROGRAM
248	Q
249	R
250	RED
251	RESET
252	RETURN
253	RUN
254	RUNNING
255	S
256	SAME
257	SCORE
258	SECOND
259	SMALL
260	START
261	STOP
262	SWITCH
263	T
264	TEN
265	THANK
266	THAT
267	THE
268	THEN
269	THIRD
270	THIS
271	TIME
272	TRY
273	TWELVE
274	TYPE
275	U
276	UH
277	UP
278	V
279	VERY
280	W
281	WANT
282	WAS
283	WERE
284	WHAT
285	WHICH
286	X
287	Y
288	YEAR
289	YES
290	YOUR
291	Z
	**********

ASCII CODE	SYMBOL	WORD
33	!	TEN
34	"	TWEN-
35	£	THIR-
36	$	FOUR-
37	%	FIF-
38	&	SIX-
39	'	SEVEN-
40	(	EIGH-
41	)	NINE-
42	*	TIME
43	+	PLUS
44	,	THOUSAND
45	-	MINUS
46	.	POINT
47	/	ON
48	0	
49	1	
50	2	
51	3	
52	4	
53	5	
54	6	
55	7	
56	8	
57	9	
58	:	TONE1
59	;	TONE2
60	<	SMALL
61	=	IS
62	>	LARGE
63	?	WHAT
64	@	AMOUNT
65	A	A
66	B	B
67	C	C
68	D	D
69	E	E
70	F	F
71	G	G
72	H	H
73	I	I
74	J	J
75	K	K
76	L	L
77	M	M
78	N	N
79	O	O
80	P	P
81	Q	Q
82	R	R
83	S	S
84	T	T
85	U	U
86	V	V
87	W	W
88	X	X
89	Y	Y
90	Z	Z
91	[	START
92	\	OFF
93	]	STOP
94	^	THANK
95	_	LINE
96	`	POUND
97	a	AND
98	b	BAD
99	c	CORRECT
100	d	0.125(SHORT PAUSE)
101	e	0.25(LONG PAUSE)
102	f	FILE
104	h	HUNDRED
105	i	IN-
106	j	-ING
107	k	KEY
108	l	ILLEGAL
109	m	MUST
110	n	KNOW
111	o	ONLY
112	p	PRESS
113	q	ON
114	r	RETURN
115	s	-S
116	t	THE
117	u	YOUR
118	v	VERY
119	w	WHICH
120	x	NUMBER
121	y	YES
122	z	-Z
123	{	FIRST
124	|	OR
125	}	LAST
126	~	NOT
103	g	GOOD
**********		

Games of the unexpected

for the unafraid...

THE WHITE BARROWS

Somewhere amid this maze of burial chambers lurks an Evil Sorcerer whom you need to trap. Trouble is, he's protected by Trolls, Dwarves, Serpents and the occasional Dragon or two! Your magic staff will block the tunnel to prevent him escaping unless, that is, he outwits you.

A real brain twister, White Barrows requires both brains and brawn from its players. It's no good just hacking your way through the Barrows and hoping to fall over the Sorcerer. Eventually you'll meet a Dragon, and they don't hack easily! You'll need all your strength and cunning to survive this one for long.

THE WHITE BARROWS Only £6.50 all inclusive!

CELLS AND SERPENTS

More monsters than you ever thought could live behind your keyboard. Wander the hills in search of gold and glory but be very, very careful where you tread! There are things here that will make your wildest nightmares look like Julie Andrews. Fancy meeting a Mind Flyer, for example? Or how about shaking hands with an Asmodeus? (You'll only do that once!) Treasure is here to be found though... the hard way.

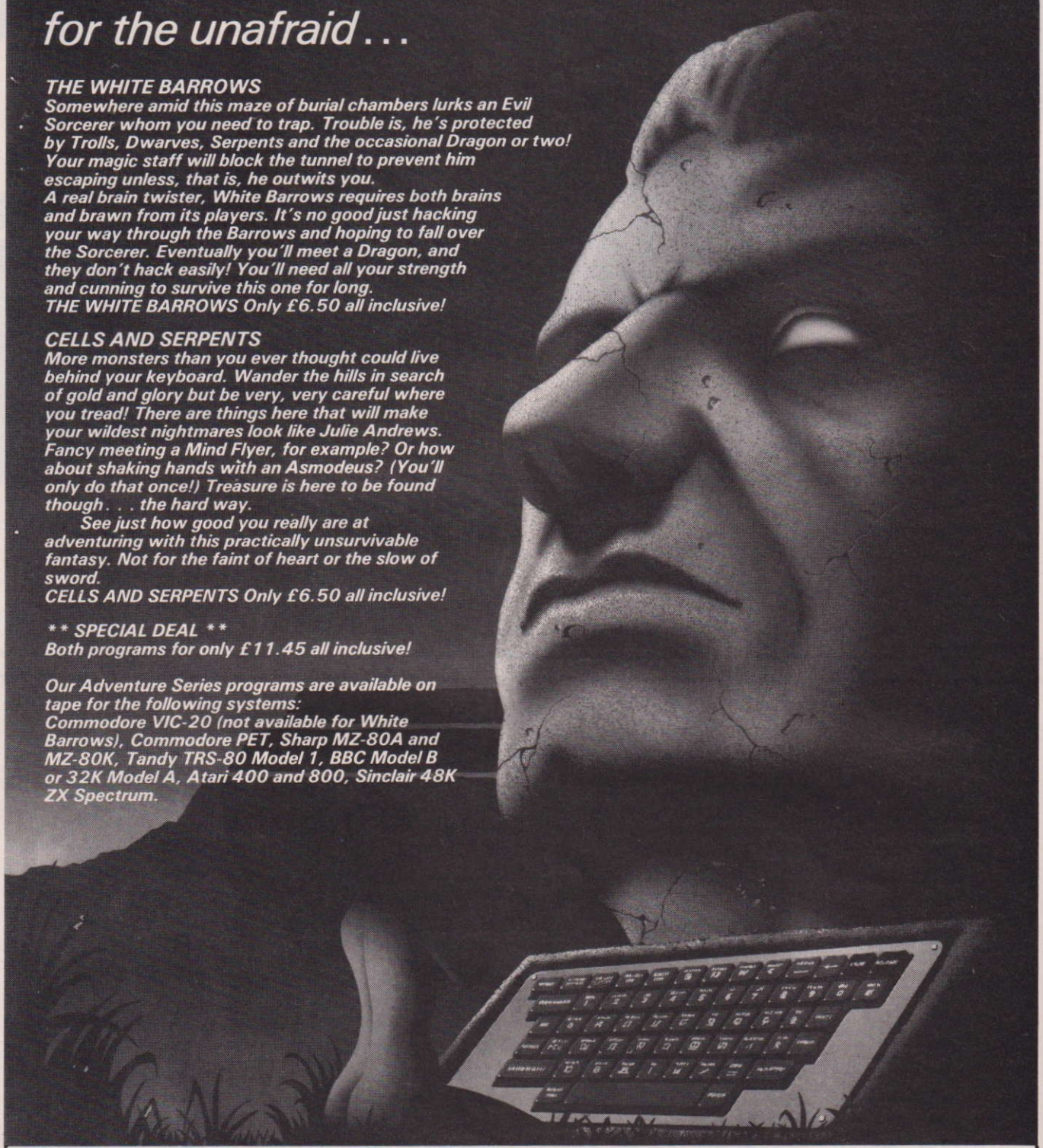
See just how good you really are at adventuring with this practically unsurvivable fantasy. Not for the faint of heart or the slow of sword.

CELLS AND SERPENTS Only £6.50 all inclusive!

*** SPECIAL DEAL ***

Both programs for only £11.45 all inclusive!

Our Adventure Series programs are available on tape for the following systems: Commodore VIC-20 (not available for White Barrows), Commodore PET, Sharp MZ-80A and MZ-80K, Tandy TRS-80 Model 1, BBC Model B or 32K Model A, Atari 400 and 800, Sinclair 48K ZX Spectrum.



ASP Software, ASP Ltd,
145 Charing Cross Road, London WC2H 0EE

Please send me ... tape(s) of the following programs:
The White Barrows ..... @ £6.50 each
Cells and Serpents ..... @ £6.50 each
Both tapes at special price ..... only £11.45
My system is a ..... computer

TRADE ENQUIRIES WELCOME

I am enclosing my Cheque/Postal Order/Money Order (delete as necessary) for £ ... (payable to ASP Ltd)
OR Debit my Access/Barclaycard (delete as necessary)

Please use BLOCK CAPITALS and include your postcode
NAME (Mr/Ms) .....
ADDRESS .....
.....
..... POSTCODE .....
Signature ..... Date .....

Sounding out Loud

P. Richardson

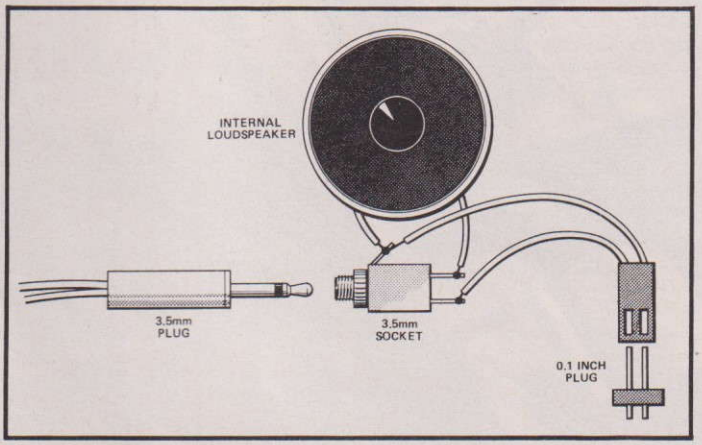
If you're not satisfied with the output of your micro's loudspeaker, have you ever thought of using an external loudspeaker?

The BBC computer is provided with a moderately powerful sound synthesis chip, the 76489 (1C 18) to be precise. It is capable of producing three pure tones and one white noise source, all software definable. Judicious selection of the parameters for these sound channels will produce reasonably accurate impressions of most familiar musical instrument and waveforms. A square wave for example is characterised by a predominance of odd harmonics thus the tone channels would be selected to give frequencies comparable to the tonic, 3rd and 5th harmonics.

Due to both size and cost considerations however, it was impractical to put a high quality loudspeaker inside the computer. The speaker chosen has a cone size a little over three centimetres in diameter. Its small size necessarily limits its operation to treble sounds, frequencies below middle C (53 on the sound 'pitch' command) will be attenuated considerably. Additionally even good loudspeakers are only about 10% efficient. Thus most of the electrical power available from the LM386 driver amplifier (0.25w) will never be turned into sound energy. There are two possible answers to this problem.

Firstly, an external loudspeaker may be attached. This should be larger than the internal speaker but not so large as to be undrivable from the small LM386. I have successfully tried a 5" Goodmans at eight ohms, and an elliptical 5" by 8" of an unknown manufacturer. Both of these gave very good reports of themselves, with at least a four-fold increase in volume due to increased efficiency.

Second, the LM386 will happily drive another external power amplifier which can in turn drive its own loudspeaker. I have tested this with a TDA2030 power amplifier chip running at 10w into eight ohms, a Quad 303 hi-fi amplifier and a 100 watt bass guitar amplifier driving a 15" loudspeaker! Computer games take on a new meaning when you can *feel* the asteroids exploding. In each case the computer's internal volume control, VR1, was used to decrease the output power, thereby decreasing harmonic distortion and avoiding clipping.



HARDWARE CONNECTIONS

The on-board loudspeaker is connected to the PCB using a 0.1" pitch two pin connector (nearest RS equivalent 467-605, which has three ways but still fits). Simple connection to an external loudspeaker may be effected by using one of these connectors on a flying lead and disconnecting the internal speaker. However, a flying lead protruding from the case is undesirable. In addition every time it is necessary to change back to the internal speaker, both the top cover and the keyboard have to be removed to get at the plug, which is a nuisance.

A better alternative is to use a switched 3.5mm socket, RS 478-497. This may be secured in a 6mm hole drilled in the side of the case. Connections should be as shown in Fig.1.
Normally, the internal loudspeaker is connected. However, insertion of a 3.5mm plug will break this circuit and instead direct the output through this plug to either speaker or an amplifier.

LAST WORD

The use of the recommended 3.5mm socket has additional spin-offs. Delicate insertion of a used matchstick will render silent the eternal 'beeping' and background hiss, which for those of us working with many of these machines in schools and computer establishments will be a blessing indeed.

Also it is now possible to mix together sound outputs from several machines. Thus different 'instruments' can be placed on a stereo 'sound stage' with a multi-track mixer. Perhaps one day we shall even see the BBC (Computer) Symphony Orchestra?!

If you have any other ideas for improving the BBC sound facility why not write in and tell us about it?

The BBC Programme - a spotlight on Queen's School

Catherine Robins

The BBC Computer Programme chose Queens School in Richmond's Kew to take a close look at how the school kids use their computers and found that computer games playing is also an excellent teaching tool.

Walking into a primary school and trying to prise the children away from their new computer is a little like trying to disturb a group of football enthusiasts watching the Cup Final! Total concentration and disregard for anything else that might be going on. Such was our first impression when we arrived at Queens Church of England Junior School in Kew to film a story for the series 'Making the Most of the Micro'. The BBC crew, complete with cameras, lights and other filming paraphernalia, was regarded with little more than the odd glance — it would take something a lot more exciting than us to draw attention away from 'Brick-Up', 'Crash', 'Kingdom' and the other games the children were playing!

Queens School has had a BBC Micro for just over two terms now, and both children and staff have had a fair amount of experience in using it. Looking at them now, it's hard to imagine that there were ever any reservations about getting a computer, but as teacher Malcolm Rivers pointed out, it was hard for the staff to see exactly where it could fit into the school curriculum. Added to that was the fact that most of them had had little or no computer training. They were breaking new ground, and Rivers for one had pretty mixed feelings about the idea: "I felt excitement and trepidation. Ex-

citement because I enjoy innovations in the school, and yet I wasn't science or maths trained and so I felt a certain amount of trepidation because I didn't know how I would get on with the machine."

Despite the initial reservations, the computer was duly installed in the school entrance hall, in full view of pupils, staff and parents alike. This was a deliberate move, ensuring that the machine quickly became a part of everyday school life, and not something to be hidden away and feared. And sure enough, the worries were largely unfounded; it took very little time to become familiar with the keyboard, and the children were soon so enthusiastic that a rota system was organised, enabling them to use the computer in small groups.

As we talked to the children it became obvious that the reason for their enthusiasm is that they regard the computer as an elaborate plaything — as they put it, "it gets us off doing real work!" And fair enough, since the programs they are using are games. But, of course, there's more to it than that. Two of the most popular games are 'Crash' and 'Brick-Up', both of which have been specially written for use in schools by the Government-sponsored Microelectronics Education Programme. At the mention of the word 'games', it's hardly surprising that teachers like Malcolm Rivers tend to conjure up visions of interminable Space Invaders which have nothing whatever to do with the school curriculum.

But the thinking behind the MEP project has been that the excitement which so many children derive from playing games like Space Invaders could just as well be built into games which are geared specifically towards developing the sort of process skills which crop up throughout the curriculum — fact-finding, problem-solving, decision-making, organising and so on. Such is the case with both 'Crash' and 'Brick-Up'.

'Crash' is an investigative program, designed to encourage problem-solving: the problem in this case is to find your way around a maze. A vehicle, represented by an arrow, is positioned at the bottom of a grid of squares. Drawn on the grid is an obstacle course, around which the arrow must move without crashing, in order to reach its target at the end. 'Brick-Up' is designed to test children's spelling. With the help of a clue: for instance, "a colourful tropical bird beginning with the letter 'p'", the child is encouraged to guess and spell the word correctly. The reward for a correct spelling is being able to fire an arrow through a wall of bricks until a target on the other side is reached.

The beauty of this game is that a teacher can easily adapt the program to include words which might crop up regularly in a classroom reader. Of particular interest in Queens School is another game, 'Kingdom', which is available as part of the BBC's 'Welcome' package. The child playing this strategy game is the ruler of a small kingdom which he or she must protect against floods and attacks from bands of theives, while keeping an eye on the population level and how much food there is in store. The idea is to survive as long as possible. This has fitted very well into the general theme of 'treasure', which the children have been studying in class. In this case the food becomes the treasure which must be guarded and for which they are responsible.

TESTED SOFTWARE

'Crash' and 'Brick-Up' are part of a much larger collection of materials for teachers and pupils, and are the product of over two years of extensive research and discussion. Before being made available, each of the MEP's programs was tested in schools up and down the country and comments and suggestions from teachers were welcomed. Of course it's still early days, but judging from the reaction at Queens School, the various programs, ranging from maths to spelling have been fitting quite well into the curriculum.

Teacher Jenny Capstick was full of praise for the screen display; "I think the graphics and the colour presentation are excellent for the children, and I think they add to the appeal and help maintain the children's interest". And the children themselves all seem to feel that encountering words or maths problems on the computer helps them to remember them when they write exercises later.

However, Malcolm Rivers is still anxious that the computer

CONTINUED OVER

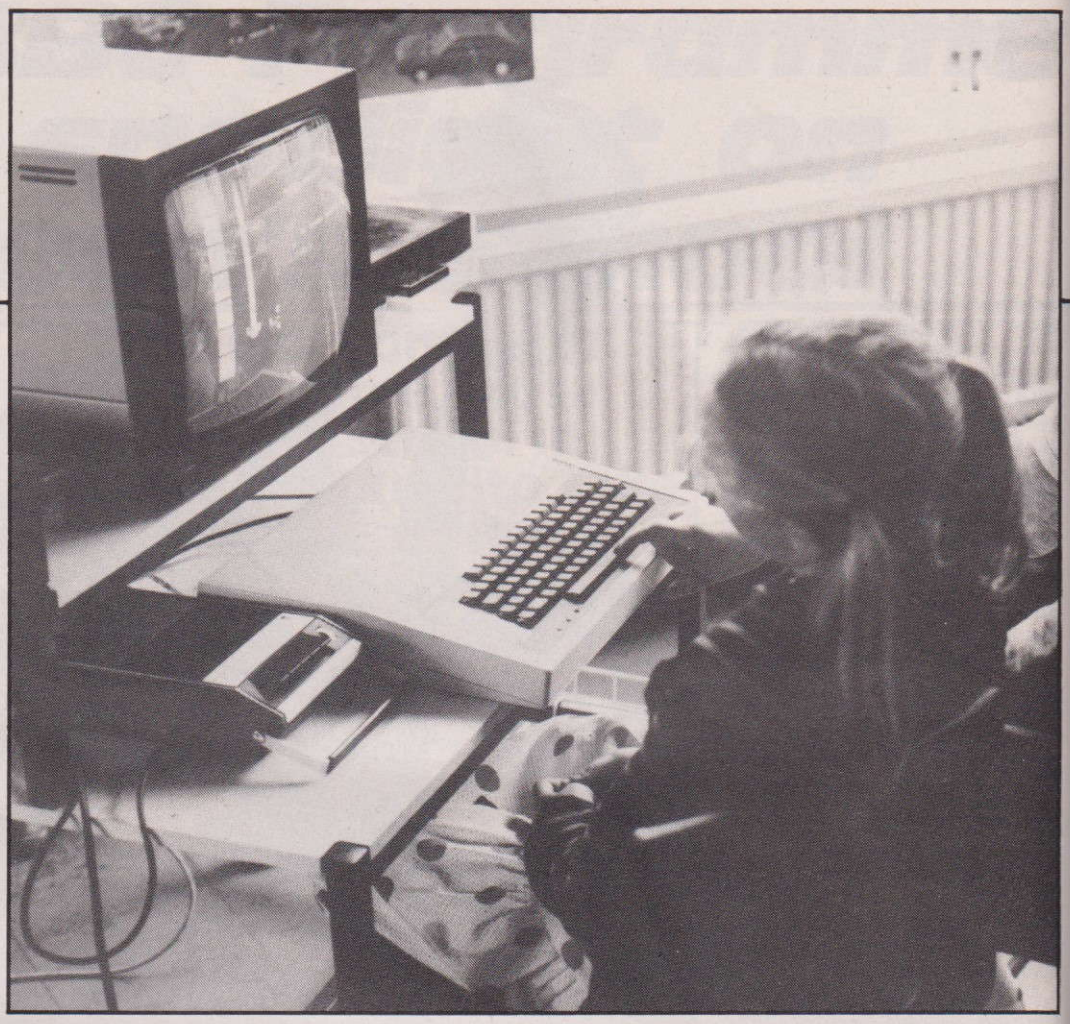
shouldn't dictate the way in which he plans his lessons. He explained that he often relies on the structure of a program which has been written according to someone else's ideas. Those ideas might or might not coincide with his own, or with the level of ability of his pupils. His lessons, he feels, would suffer tremendously if he lost the freedom to be flexible in his teaching because he was constantly adapting his classes to fit in with the computer.

According to Bob Coates, Project Manager at the MEP, this concern is shared by many teachers who recognise the enormous potential of micros and yet find themselves restricted by lack of knowledge or confidence in adapting programs to meet their needs. As well as helping to train teachers, one of the major aims of the MEP has been to encourage them to submit ideas or specifications for material which they feel would be of particular use to them. The idea is that a teacher shouldn't have to be a good programmer — as long as he or she knows where to go for advice there are plenty of people willing to develop ideas to meet the needs of the classroom.

Should sponsorship for the MEP come to an end, Coates and his colleagues are very much aware of the need to maintain links between parents, teachers and local authorities around the country so that such ideas and information might continue to be shared.

RIGHT FORMULA

The idea of using computers in primary schools is still in its infancy, and staff and pupils alike are still learning to come to grips with their machines. Moreover, we can have little conception of just how the development of microelectronics will eventually affect the working lives of today's primary school children. It would be arrogant of anyone to claim to be providing the best and only foundation for the future, but if the reaction at Queens School is anything to go by, organisations such as the MEP are certainly working along the right lines.

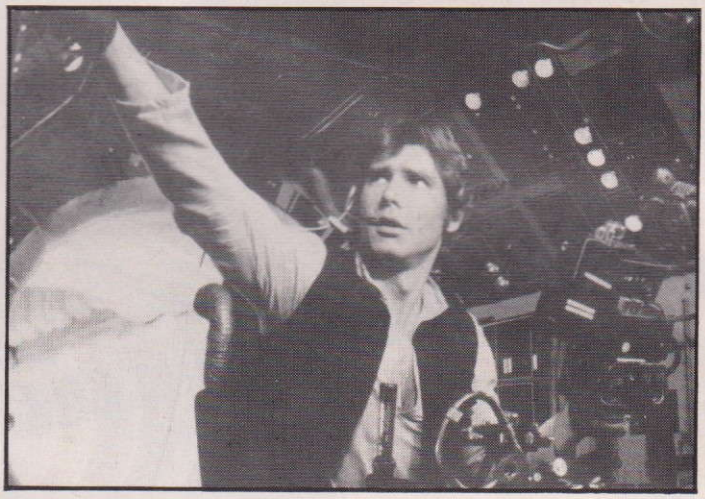


'Glued to the screen' takes on a new meaning!

Light Speed Graphics

J. Ruston

Hans Solo's ship travelled at light speed, try filling your screen as quickly!



Recursion is a programming technique that you may have heard about, but have probably not yet found any use for, nor indeed fully understood its properties.

The academic definition of recursion is that 'a recursive routine is one that is defined in terms of itself'. This does not usually appear to be a very helpful description to beginners in programming. If we rephrase that definition, it becomes relatively easy to see how recursion operates. We shall leave the question of why we should ever wish to use recursion until later on.

To examine the process of recursion, we will look at the problem of adding up all the numbers from 1 to N, where N is an integer input by the user. The simplest (and indeed the fastest) method of doing this is to set up a loop, saying:

```
counter = 0
FOR T% = 1 TO N
counter = counter + T%
NEXT T%
PRINT counter
```

This method works and is quite efficient, taking up only five lines counting the initialisation.

ANOTHER WAY

There is another method. We can define the sum of all the integers up to N as being the number N plus all the integers up ot N – 1. For example, the sum of all integers up to 10, including 10, is 10 plus the sum of all integers up to 9. This may not sound a particularly startling revelation, but it allows us to define a function called FNsum with the argument N, the code of which would consist of:

FNsum(N) = N + FNsum(N – 1)

If one did this it would be found that the memory quickly filled up. There would be a 'no room' error when trying to do the sum of the numbers. The reason for this is that the function always calls itself. We need a cutoff point at some point to stop calling itself. For example, we can recode the function as:

```
DEF FNsum(N)
IF N = 1 THEN = 1 ELSE = N +
    FN + sum(N – 1)
```

In other words, if N is 1 on entry, the answer is returned as 1, otherwise the answer is N + FN sum(N – 1). This formula will work perfectly for low integers. For high integers it fills up memory very perfectly for low integers. For high integers it fills up memory very quickly because for each call to a function or a procedure, the BBC Micro has to use up a certain amount of memory space, depending upon the number of arguments that you pass to the function or procedure and the number of LOCAL variables set up in the function or procedure. If you make a routine call itself, this effect is multiplied many times which will result in a shortage of memory. In this case, recursion is not a viable alternative to the normal programming method we outlined earlier.

USES

To recap, a recursive function or procedure is one that calls itself. However, to ensure that the functions and procedures do not go on nesting themselves for ever and ever, you must have a conditional statement in the function or procedure that will stop it calling itself under certain circumstances. To examine the practical applications of recursion, there are a number of things we can do.

First of all, using recursive programming one can create some very beautiful patterns. In addition, recursion is useful in graphics filling routines. Most microcomputer languages have to be written in a recursive way — for example, the IF statement can be followed with another IF statement, which makes the routine which deals with the IF statement recursive.

ABOUT THE PROGRAM

One very simple pattern generator using recursive programming techniques is shown below. A procedure is set up, the parameters of which are the centre co-ordinates of a square and the length of the sides of a square. When the procedure is called, it draws the square as directed. Next, it exits if the size of the square is relatively small.

Finally, the procedure is called again, ie. it does four recursive calls with the centre of the square being each of the four corners of the original square and the sides of the square being half the size of the original square. The program shown generates quite intricate patterns, with only 10 active lines.

KEYLIST UTILITY ROUTINE

This is a special utility to print the definitions of the user definable function keys.

Many micros besides the BBC Micro provide user definable function keys to help in programming and applications. However, most of these other micros provide a command called KEYLIST, or something similar to print out the current definitions of each key. This feature is lacking in the BBC Micro. However, this short program serves the same function.

The area of memory from &B00 to &BFF (which is where the function key definitions are stored), is not organised in a particularly useful way. The format used for storage is as follows:

The first 16 bytes of this area are set aside as pointers to the start of each function key definition. For example, the first byte of this area (&B00) points to the beginning of the definition of function key 0, the second byte points to the beginning of the definition of functions key 1 and so on, up to key 15.

However, if no function keys are defined, all these pointers will point to the start of the function key definition area. If a function key is defined, any undefined function keys will be adjusted to point to the next free location after the definition in question. The procedure for deciphering this information is pretty complex. Here the algorithm is described.

THE ALGORITHM

In line 90 M% is going to be the number of bytes that are used in all the current function key definitions. M% will be incremented each time a byte is detected. In line 100, a loop through all the possible function keys is set up. In line 110 a blank line is printed to allow for the fact that each function key definition printout ends in a semicolon.

In line 120, if the current function key number is less than 10, then a space is printed, which will simply align the function keys

CONTINUED OVER

neatly. In line 130 a heading for the current key is printed out. In line 140 A% is set to be the pointer byte for the current function key. In line 150 B% is set to be 256.

This is the highest possible value of any of the pointers throughout the function keys. So, in lines 160 to 180 the pointers to the other function keys are examined and if they are less than the value B, then B is set to their value. Thus, this will allow us to find the lowest function key pointer in numerical order. In line 190, if B% has not been altered by this procedure then this function key is empty. So, a jump is made to the end of the current loop and a jump past the next FOR loop.

LOOPS

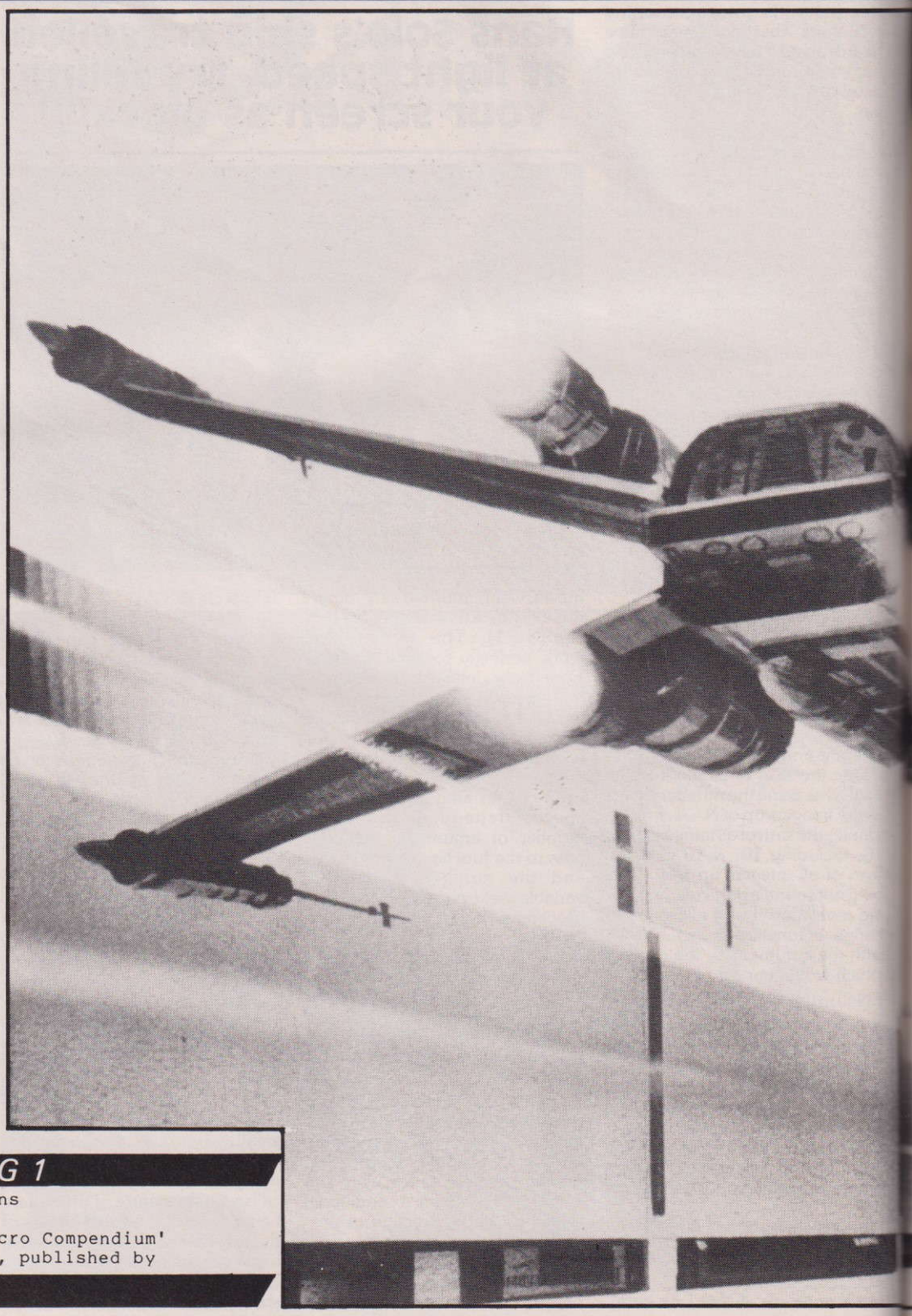
In line 200 another loop is set up, which simply prints out all the characters between the two extremities in the function key definition, using the VDU statement and incrementing M% for each character printed out. This allows the total number of characters used to be worked out. In line 230 a check is made to see if the current character is a control character.

If it is, a vertical bar is used, in the normal way. This allows things like embedded carriage returns to be printed out in the correct manner. In lines 240 and 250, the two loops used in the program are closed. Line 230 prints an indication of the number of bytes used.

A sample run of the program follows the listing. As you can see it also prints out the contents of the function keys 10 to 15, which are normally inaccessible.

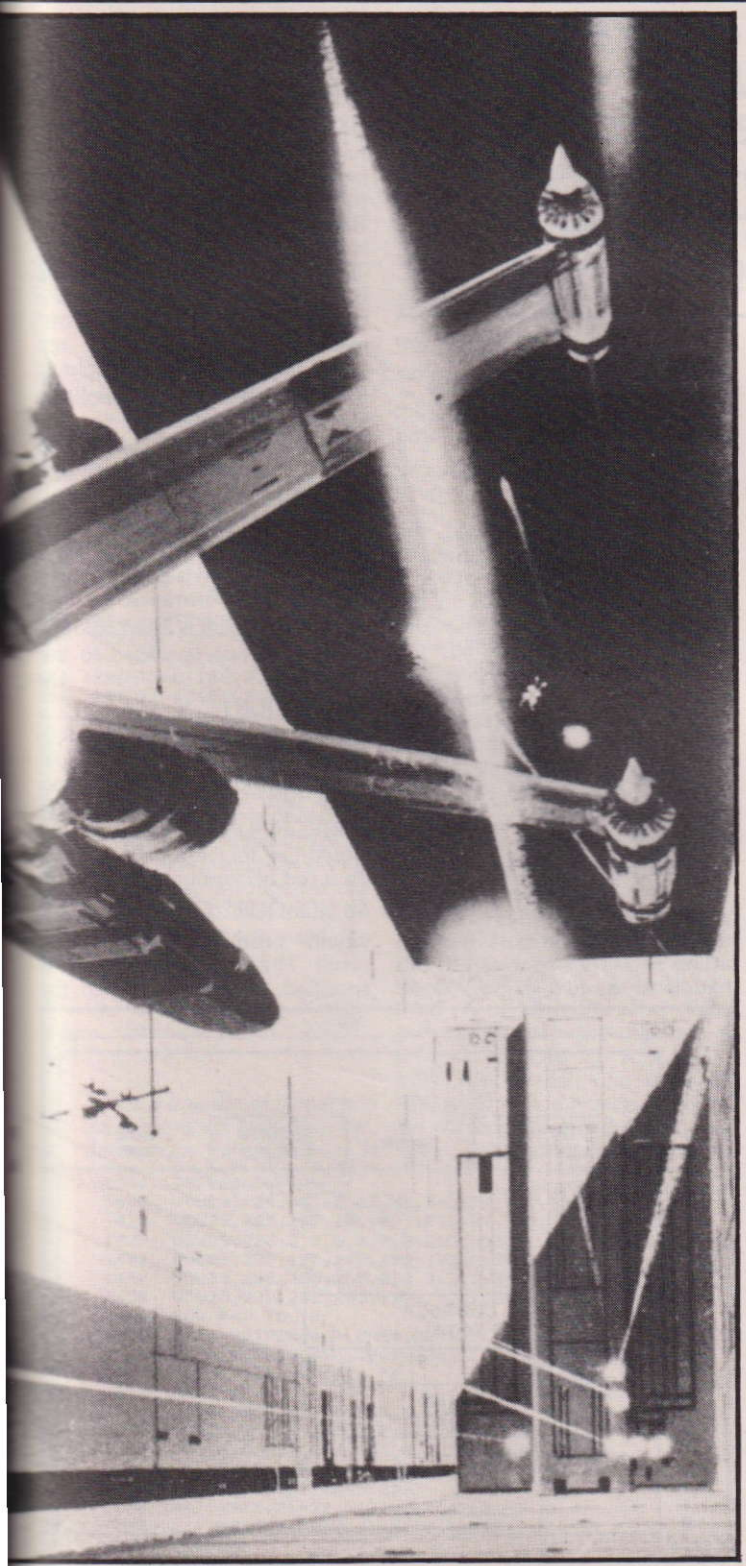
This program has only been tested on OS 1.00 and OS 1.20. Thus it should work on OS 1.10, but it is unlikely to work correctly on OS 0.10.

In use, the program could be converted to assembly language for faster access, or the listing as given could be combined with an application program which utilises function keys.



PROGRAM LISTING 1

```
10 REM Recursive patterns
20
30 REM From 'The BBC Micro Compendium'
40 REM By Jeremy Ruston, published by
```



```
 50 REM Interface
 60
 70 MODE 4
 80 PROCsquare(640,512,250)
 90 END
100
110 DEF PROCsquare(X%,Y%,D%)
120 MOVE X%-D%,Y%-D%
130 DRAW X%+D%,Y%-D%
140 DRAW X%+D%,Y%+D%
150 DRAW X%-D%,Y%+D%
160 DRAW X%-D%,Y%-D%
170 IF D%<16 THEN ENDPROC
180 PROCsquare(X%-D%,Y%-D%,D% DIV 2)
190 PROCsquare(X%+D%,Y%-D%,D% DIV 2)
200 PROCsquare(X%+D%,Y%+D%,D% DIV 2)
210 PROCsquare(X%-D%,Y%+D%,D% DIV 2)
220 ENDPROC
```

PROGRAM LISTING 2

```
  10 REM --- Function key list ---
  20
  30 REM From 'The BBC Micro Compendium'
  40 REM by Jeremy Ruston, published by
  50 REM           Interface
  60
  70 REM Tested on OS 1.00 and OS 1.20
  80
  90 M%=0
 100 FOR K%=0 TO 15
 110 PRINT
 120 IF K%<10 THEN PRINT " ";
 130 PRINT "f";K%;"--> ";
 140 A%=K%?&B00
 150 B%=256
 160 FOR T%=0 TO 15
 170 IF T%?&B00>A% AND T%?&B00<=B% THEN B%=T%?&B00
 180 NEXT T%
 190 IF B%=256 THEN GOTO 250
 200 FOR G%=A% TO B%-1
 210 H%=G%?&B01
 220 M%=M%+1
 230 IF H%>31 THEN VDU H% ELSE VDU 124,H%+64
 240 NEXT G%
 250 NEXT K%
 260 PRINT ';M%;" bytes used."
 270 END
>RUN

 f0--> LIST|M
 f1--> RUN|M
 f2--> SAVE "CURRENT"|M
 f3--> LOAD "CURRENT"|M
 f4--> |BLIST|M|C
 f5--> *WORDWISE|M
 f6--> *WORD|M
 f7--> *CAT|M
 f8--> *FX 5,1|M*FX6|M
 f9--> *FX 5,2|M*FX8,4|M*FX6|M
f10-->
f11-->
f12-->
f13-->
f14-->
f15-->
100 bytes used.
```

In the Beginning

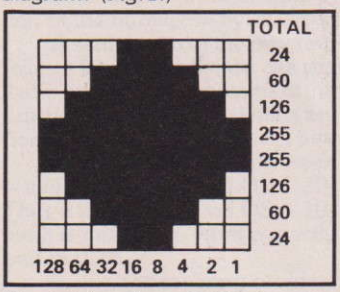
T. Gallagher

Continuing our regular 'Learning BASIC' series, we take the first steps towards defining your own characters and introducing colour.

The first part of this series in the launch issue of *A & B Computing* introduced the idea of using strings to carry information which could then be extracted when required. Not all programs are required to contain information. In some cases the visual impact of what appears on the screen is of greater importance, including the need to catch and hold the attention of the operator, perhaps a child. The addition of sound to the program will add to the effect.

The characters available from the keyboard are not in themselves exciting, so that the facility to design one's own shapes is a very desirable one. A group of character numbers (from 224 to 255) are readily available without disturbing any of the keyboard characters, and more numbers are available if you are prepared to redesign some of the numbers already in use.

To design a character, one needs a piece of squared paper to work on. The character is composed of 8 by 8 dots, which may be used or not as required. Each point on a line has its own numerical value, rising in powers of 2 from the right to the left. The value of a line is then found by adding together the points on that line, as shown in the diagram. (Fig.1.)



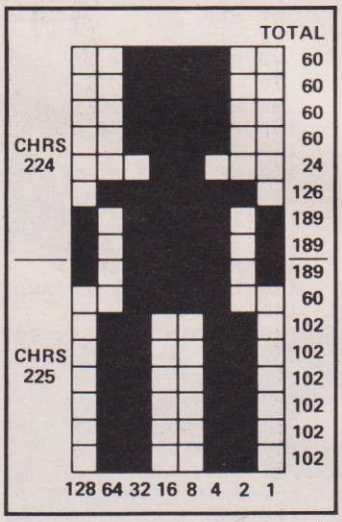
To plot the shape in Fig.1 on the screen, a VDU 23 statement is used.

VDU23,224,24,60,126,255,225,126,60,24

224 is the character number, and the remaining 8 numbers follow in order from the top of the square to the bottom. The computer reads these decimal numbers, converts them to binary, and fills in the dots where the binary '1's would be. For example, 24 (decimal) is 00011000 in binary, which gives the top line in the example.

This character can be reproduced in any Mode except Mode 7. Typing in the line

MODE5: VDU23,224,24,60,126,255,255,126,60,24:PRINT CHR$224,



and pressing return, will clear the screen, and print the shape in the top left hand corner of the screen.

To extend the use of defined characters, a square shape is not always ideal. Figure 2 shows a shape which requires two characters to define it. The print statement would then have to include both parts so that one is immediately underneath the other.

```
10  MODE1
20  VDU23,224,60,60,60,60,24,
    126,189,189
30  VDU23,225,189,60,102,102,
    102,102,102,102
50  PRINTTAB(10,10)CHR$224
60  PRINTTAB(10,11)CHR$225
```

Lines 50 and 60 print the two characters.

This program is in Mode 1. After running it in its present form, change line 10 to read

10 Mode 0

and run it again to see the effect. Making similar alterations to try the other Modes from 0 to 7 will show the differences in the characters produced. The differences are caused by the number of characters per line, and the number of lines per screen. Mode 0 will give a very thin shape, because it uses 80 characters per line, while Modes 1 and 5 use only 20 characters per line, therefore producing a shape 4 times as wide as Mode 0.

Modes 3 and 6, being 'text only' have 25 lines per screen, as opposed to the 32 lines of Modes 0,1,2,4,5, and the extra depth of a line shows up as a gap in the middle of the two shapes. Mode 7 will not produce the shape at all.

COLOUR

So far, all the shapes produced have been in white on a black background, these being the natural colours for all modes. Returning line 10 to Mode 1, we have then four colours available. These are described as 'logical colours', which simply means that there are four 'slots' available, numbered 0 to 3, into which any of the 16 listed colours may be placed. Normally these colours are (in Mode 1)

Logical colour 0 . . . . . black
. . . . . which is listed colour 0
Logical colour 1 . . . . . red
. . . . . which is listed colour 1
Logical colour 2 . . . . . yellow
. . . . . which is listed colour 3
Logical colour 3 . . . . . white
. . . . . which is listed colour 7

If an extra line is added to the program

40 COLOUR1

running the program will result in a red shape, whereas COLOUR 2 would produce a yellow shape. Line 40 has defined the FOREGROUND colour. If you would like to see a yellow figure on a red background, then

40 COLOUR2:COLOUR129

would produce the required result. The background colour is specified by adding 128 to the

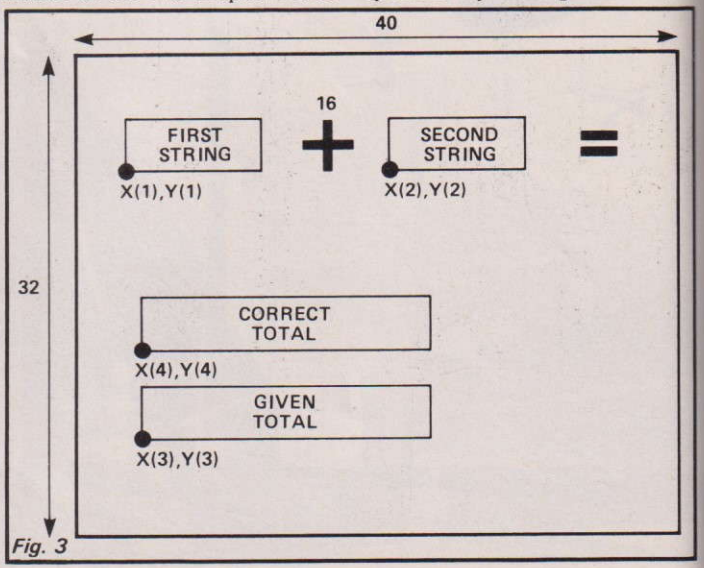


Fig. 3

logical colour.

To see all the colours available, use Mode 2 and change the program as follows.

```
10  MODE2
20  VDU23,224,60,60,60,60,
    24,126,189,189
30  VDU23,225,189,60,102,102,
    102,102,102,102
40  FOR I=1 TO 16:COLOUR1
50  PRINTTAB(I,10)CHR$224
60  PRINTTAB(I,11)CHR$225
70  NEXT
```

This program uses the idea of a LOOP, as introduced in the previous part, to repeat the printing of the figure, side-by-side across the screen until all the available colours have been used.

Returning to the use of strings, and the STRING$, the lines

```
15  DIM A$(10),B$(10)
50  FOR A=1 TO 10
60  A$(A)=STRING$(A,CHR$224)
70  B$(A)=STRING$(A,CHR$225)
80  NEXT A
90  R=INT(RND(1)*10)+1
100 PRINTTAB(10,10)A$(R)
110 PRINTTAB(10,11)B$(R)
```

will fill an ARRAY A$ with strings of the character 224, and an ARRAY called B$ with strings of the character 225. An array is really a storage area of a defined size. If you wish to keep values or strings available for use throughout the life of a program, they must be stored, and arrays are used for that purpose.

A$(10) sets aside 11 storage spaces, labelled A$(0), A$(1), A$(2).....A$(10).

This is a one-dimensional array, but there is no restriction on the dimensions or number of elements stored in an array, except that memory runs out very quickly! In some computers, an array of 10 elements or less would not need to be dimensioned before use, but the BBC Micro computer insists on *all* arrays being declared.

USING THE SHAPES IN A PROGRAM

The ideas discussed so far are sufficient to set up a simple program, which can then be added to if the program seems to be useful. The objective is to write a program which will use random strings of shapes to test simple addition, restricting the numbers to be added together to integers from 1 to 10 inclusive. The program consists of:

1. Setting up a string of random length, and asking for the number of shapes contained in it,
2. Repeating the process with a second string,
3. Completing the diagram with a + and =, and asking for the total.

THE SCREEN

In Mode 1, the screen offers 32 lines of 40 spaces. In Fig.3, the first string of shapes is placed at the point X(1),Y(1), and the second at X(2),Y(2). The total given from the keyboard starts a string of shapes at X(3),Y(3), while the correct total starts at X(4),Y(4). If the answer given is not correct, the spare shapes fall down, which needs two more character definitions (232 and 233), giving a figure lying in a horizontal position.

The '+' and '=' from the keyboard are too small in proportion to the shapes, and new ones are defined (226 to 231). (See Sheet 9.)

All these definitions are set up in a PROCEDURE, which can then be called upon at the beginning of the program. This procedure is called PROCDEFINE.

The random number calculations which determine the lengths of the first two strings are also contained in a procedure. Two random numbers are found, and their integerpart stored in D(1) and D(2).

The random number generator RND(10) will generate an integer value between 1 and 10 inclusive.

```
1600  DEFPROCRANDOM
1610  FOR D=1 TO 2
1620  R=RND(10):D(D)=R
1630  NEXT D
1640  ENDPROC
```

```
1390  DEFPROCDEFINE
1400  VDU23,224,60,60,60,60,24,126,189,189
1410  VDU23,225,189,60,102,102,102,102,102,102
1420  VDU23,226,3,3,3,3,3,3,255,255
1430  VDU23,227,192,192,192,192,192,192,255,255
1440  VDU23,228,255,255,3,3,3,3,3,3
1450  VDU23,229,255,255,192,192,192,192,192,192
1460  FOR A=1 TO 10
1470  A$(A)=STRING$(A,CHR$(224))
1480  B$(A)=STRING$(A,CHR$(225))
1490  C$(A)=STRING$(A," ")
1500  NEXT A
1510  VDU23,230,0,0,0,0,0,0,255,0
1520  VDU23,231,0,255,0,0,0,0,0,0
1530  VDU23,232,1,252,255,3,3,255,252,1
1540  VDU23,233,192,32,239,255,255,239,32,192
1550  D$=CHR$(232)+CHR$(233)
1560  X(1)=2:X(2)=20:X(3)=2:X(4)=2
1570  Y(1)=5:Y(2)=5:Y(3)=20:Y(4)=15
1580  ENDPROC
```

The program contained in Listing 1 will run to give a simple test of addition from the keyboard. The reader who decides to try it will almost certainly want to alter it to fit his own ideas.

Some possible alterations are:

1. A change of shape, or even a choice of shape, given at the beginning of the program.
2. Use of subtraction, multiplication or division, although the latter two would probably require larger numbers than used here. If the random number generator is altered, some alteration to the screen positions might be necessary.

CONTINUED OVER

THE PROCEDURES

PROCWAIT (1000 to 1040)	waits for a key to be pressed before continuing the program. Something of this kind, with either a GET or INKEY is necessary in most programs.
PROCDELAY (1100 to 1120)	includes a short time lag, required to ensure that changes on the screen are visible.
PROCDEFINE (1390 to 1640)	includes the definitions of the characters used, and fills the arrays A$, B$, C$, X and Y.
PROCOVER (1800 to 1850)	contains the messages used when the total typed in is not correct.
PROCRIGHT (1900 to 1920)	sound for the right answer.

PROGRAM DETAILS

1. PRINT TAB(X,Y) begins to print at the point which has coordinates (X,Y,) when the origin of coordinates is at the top left hand corner of the screen.

The ';' between items to be printed can be omitted here, since all the print statements are related to TAB statements. In general, ';' is used to ensure that no gap is left before printing the next character or statement.

2. The arrays set up in line 20 are:—

AS(10),BS(10) store strings of shapes

6S(10) stores strings of blanks to overwrite messages on the screen.

D(2) stores the two random numbers to be used.

X(4),Y(4) store the coordinates of the points at which the strings are to begin.

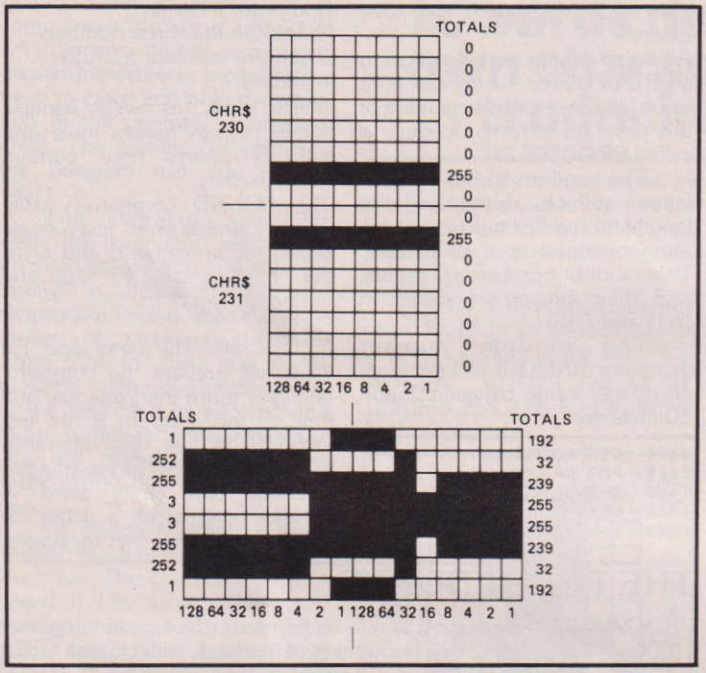
N(3) store the three numbers which are inputted from the keyboard.

Notice that, as with variable names, arrays names must end with S unless they contain numbers only.

The SOUND commands used are very simple ones, just a single sound for movement, and a rising or falling scale for right and wrong answers.

3. Line 50. The command FX 11,0 will prevent the computer taking in more than one key at a time. If the program is run first without this line, the difference will be seen. A heavy-handed operator can easily send in several repeats of a letter or number, bringing error messages.

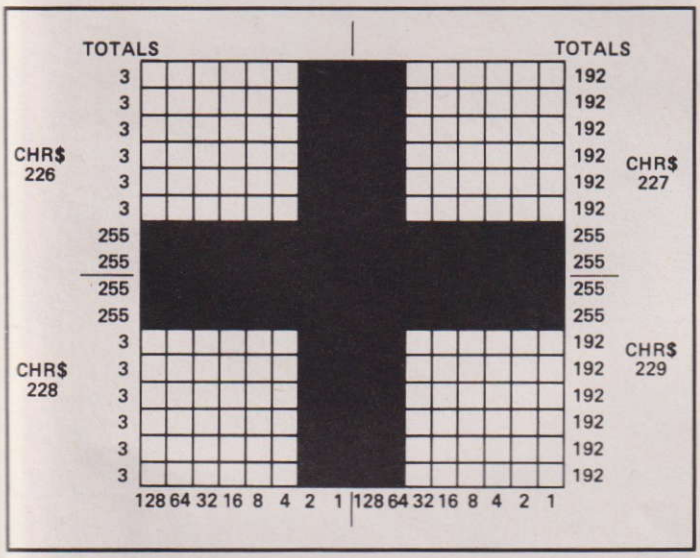
3. Adding more elaborate sound effects.
4. Adding an opening sequence or title. Listing 2 shows how to use CHR$(141) to give double height letters in Mode 7 (PROCDBL). The title given, with my name, can easily be altered to use *your* title and *your* name. PROCTITLE fills the screen with figures before the main part of the program begins.



PROGRAM LISTING 1

```
LIST
   10 MODE7
   20 DIMA$(10),B$(10),C$(10),D(2),X(4),Y(4),N(3)
   30 PROCINTRO
   40 MODE1
   50 *FX11,0
   90 PROCDEFINE
  100 PROCTITLE:PROCWAIT
  110 PROCRANDOM
  120 CLS:COLOUR1:PRINTTAB(X(1),Y(1));A$(D(1));TAB(X(1),Y
(1)+1);B$(D(1))
  130 COLOUR3:PRINTTAB(5,8);"How many?"
  140 INPUTN(1):IF N(1)=D(1) THEN 180 ELSE 150
  150 PRINTTAB(5,8);"Try again."
  160 PROCDELAY
  170 PRINTTAB(5,8);C$(10);TAB(0,9);C$(10):GOTO 130
  180 PRINTTAB(5,8);C$(10);TAB(0,9);C$(10)
  190 COLOUR1:PRINTTAB(X(2),Y(2));A$(D(2));TAB(X(2),Y(2)+
1);B$(D(2))
  200 COLOUR3:PRINTTAB(20,8);"How many?"
  210 INPUTN(2):IF N(2)=D(2) THEN 250 ELSE 220
  220 PRINTTAB(20,8);"Try again."
  230 PROCDELAY
  240 PRINTTAB(20,8);C$(10);TAB(0,8);C$(10);TAB(0,9);C$(1
0):GOTO 200
  250 PRINTTAB(20,8);C$(10);TAB(0,9);C$(10)
  260 PRINTTAB(13,5)CHR$(226)TAB(14,5)CHR$(227)TAB(13,6)C
HR$(228)TAB(14,6)CHR$(229)
  270 PRINTTAB(31,5)CHR$(230)TAB(31,6)CHR$(231)
  280 PRINTTAB(0,25);"How many?":INPUTN(3):IF N(3)<21 AND
 N(3)>0  THEN 290 ELSE 285
  285 PRINTTAB(0,25);C$(10)C$(10)TAB(0,26)C$(10)C$(10):GO
TO 280
  290 COLOUR2:PRINTTAB(X(3),Y(3));STRING$(N(3),CHR$(224))
  300 PRINTTAB(X(3),Y(3)+1);STRING$(N(3),CHR$(225))
  310 COLOUR1:FORT=1TO D(1)
  320   PRINTTAB(X(1)+T-1,Y(1));C$(1);TAB(X(1)+T-1,Y(1)+1
);C$(1)
  330   PRINTTAB(X(4)+T-1,Y(4));CHR$(224);TAB(X(4)+T-1,Y(
4)+1);CHR$(225)
  340   SOUND2,-15,1,5:PROCDELAY:NEXT
  350 FORT=1TO D(2)
  360   PRINTTAB(X(2)+T-1,Y(2));C$(1);TAB(X(2)+T-1,Y(2)+1
);C$(1)
  370   PRINTTAB(X(4)+D(1)+T-1,Y(4));CHR$(224);TAB(X(4)+D
(1)+T-1,Y(4)+1);CHR$(225)
  380   SOUND2,-15,1,5:PROCDELAY:NEXT
  390 IF N(3)=D(1)+D(2) THEN 600
  400 IF N(3)<D(1)+D(2) THEN 500
  410 DIF=N(3)-D(1)-D(2)
  420 COLOUR2:PRINTTAB(X(3)+N(3)-DIF,Y(3));C$(DIF);TAB(X(
3)+N(3)-DIF,Y(3)+1);C$(DIF)
  440 PRINTTAB(X(3)+N(3)+3,Y(3)+3);STRING$(DIF,D$)
  450 C=1:PROCOVER
  490 PROCWAIT:CLS:GOTO 120
  500 DIF=D(1)+D(2)-N(3)
  510 COLOUR1:PRINTTAB(X(4)+N(3),Y(4));C$(DIF);TAB(X(4)+N
(3),Y(4)+1);C$(DIF)
  530 PRINTTAB(X(3)+N(3)+3,Y(3)+3);STRING$(DIF,D$)
  540 C=2:PROCOVER
  590 PROCWAIT:CLS:GOTO 120
  600 COLOUR3:PRINTTAB(0,25);"WELL DONE!";TAB(0,26);C$(4)

  610 PROCRIGHT:PROCWAIT:CLS
  620 PRINT TAB(2,4)"ANOTHER GAME? Y OR N"
  630 INPUT T$:IF T$="Y" THEN CLS:GOTO 110
  640 IF T$="N" THEN 990
  650 GOTO 620
  990 END
 1000 DEFPROCWAIT
 1010 PRINTTAB(0,28);"PRESS THE SPACE BAR TO CONTINUE"
 1020 Z$=GET$:IFZ$=""THEN1010
 1030 IFZ$<>CHR$(32)THEN1010
 1040 ENDPROC
 1100 DEFPROCDELAY
 1110 FOR DELAY = 1 TO 1000:NEXT
 1120 ENDPROC
 1200 DEFPROCTITLE
 1210 CLS:T1$=CHR$(224)+" ":T2$=CHR$(225)+" "
 1220 FORI=0TO9:FORJ=1TO7
 1230       COLOUR1:PRINTTAB(I*2,J*4);T1$
 1240       PRINTTAB(I*2,J*4+1);T2$
 1250       COLOUR2:PRINTTAB(38-I*2,J*4);T1$
 1260      PRINTTAB(38-I*2,J*4+1);T2$
 1270      NEXTJ:PROCDELAY
 1275    SOUND2,-15,1,5
 1280    NEXTI
 1290 PROCDELAY:PROCDELAY
 1300 ENDPROC
 1390 DEFPROCDEFINE
 1400 VDU23,224,60,60,60,60,60,24,126,189,189
 1410 VDU23,225,189,60,102,102,102,102,102,102
 1420 VDU23,226,3,3,3,3,3,3,255,255
 1430 VDU23,227,192,192,192,192,192,192,255,255
 1440 VDU23,228,255,255,3,3,3,3,3,3
 1450 VDU23,229,255,255,192,192,192,192,192,192
 1460 FORA=1 TO10
 1470   A$(A)=STRING$(A,CHR$(224))
 1480   B$(A)=STRING$(A,CHR$(225))
 1490   C$(A)=STRING$(A," ")
 1500   NEXT A
 1510 VDU23,230,0,0,0,0,0,0,255,0
 1520 VDU23,231,0,255,0,0,0,0,0,0
 1530 VDU23,232,1,252,255,3,3,255,252,1
 1540 VDU23,233,192,32,239,255,255,239,32,192
 1550 D$=CHR$(232)+CHR$(233)
 1560 X(1)=2:X(2)=20:X(3)=2:X(4)=2
 1570 Y(1)=5:Y(2)=5:Y(3)=20:Y(4)=15
 1580 ENDPROC
 1600 DEFPROCRANDOM
 1610 FORD=1TO2
 1620   R=RND(10):D(D)=R
 1630   NEXT D
 1640 ENDPROC
```

```
1700 DEFPROCDBL(W$,U,W)
1710 PRINTTAB(U,W);CHR$(141);W$
1720 PRINTTAB(U,W+1);CHR$(141);W$
1730 ENDPROC
1800 DEFPROCOVER
1810 FORI=97 TO 25 STEP -4:SOUND1,-15,I,1:NEXT
1820 IFC=1 PRINTTAB(0,25);"Your number was too large!":G
OTO 1840
1830 IFC=2 PRINTTAB(0,25);"Your number was too small!"
1840 PRINTTAB(0,26);"Try again!"
1850 ENDPROC
1900 DEFPROCRIGHT
1910 FORI=25 TO 97 STEP 8:SOUND1,-15,I,1:NEXT
1920 ENDPROC
2000 DEFPROCINTRO
2010 PROCDBL("ADDITION",12,10)
2020 PROC    ("G.W.GALLAGHER",20,16)
2030 PROCDBL("1983",25,20)
2040 PROCDELAY:PROCDELAY:PROCDELAY
2050 ENDPROC
```



PROGRAM LISTING 2

```
LIST
  20 DIMA$(10),B$(10),C$(10),D(2),X(4),Y(4),N(3)
  40 MODE1
  50 *FX11,0
  90 PROCDEFINE
 110 PROCRANDOM
 120 CLS:COLOUR1:PRINTTAB(X(1),Y(1));A$(D(1));TAB(X(1),Y
(1)+1);B$(D(1))
 130 COLOUR3:PRINTTAB(5,8);"How many?"
 140 INPUTN(1):IF N(1)=D(1) THEN 180 ELSE 150
 150 PRINTTAB(5,8);"Try again."
 160 PROCDELAY
 170 PRINTTAB(5,8);C$(10);TAB(0,9);C$(10):GOTO 130
 180 PRINTTAB(5,8);C$(10);TAB(0,9);C$(10)
 190 COLOUR1:PRINTTAB(X(2),Y(2));A$(D(2));TAB(X(2),Y(2)+
1);B$(D(2))
 200 COLOUR3:PRINTTAB(20,8);"How many?"
 210 INPUTN(2):IF N(2)=D(2) THEN 250 ELSE 220
 220 PRINTTAB(20,8);"Try again."
 230 PROCDELAY
 240 PRINTTAB(20,8);C$(10);TAB(0,8);C$(10);TAB(0,9);C$(1
0):GOTO 200
 250 PRINTTAB(20,8);C$(10);TAB(0,9);C$(10)
 260 PRINTTAB(13,5)CHR$(226)TAB(14,5)CHR$(227)TAB(13,6)C
HR$(228)TAB(14,6)CHR$(229)
 270 PRINTTAB(31,5)CHR$(230)TAB(31,6)CHR$(231)
 280 PRINTTAB(0,25);"How many?":INPUTN(3):IF N(3)<21 AND
 N(3)>0  THEN 290 ELSE 285
 285 PRINTTAB(0,25);C$(10)C$(10)TAB(0,26)C$(10)C$(10):GO
TO 280
 290 COLOUR2:PRINTTAB(X(3),Y(3));STRING$(N(3),CHR$(224))
 300 PRINTTAB(X(3),Y(3)+1);STRING$(N(3),CHR$(225))
 310 COLOUR1:FORT=1TO D(1)
 320   PRINTTAB(X(1)+T-1,Y(1));C$(1);TAB(X(1)+T-1,Y(1)+1
);C$(1)
 330   PRINTTAB(X(4)+T-1,Y(4));CHR$(224);TAB(X(4)+T-1,Y(
4)+1);CHR$(225)
 340   SOUND2,-15,1,5:PROCDELAY:NEXT
 350 FORT=1TO D(2)
 360   PRINTTAB(X(2)+T-1,Y(2));C$(1);TAB(X(2)+T-1,Y(2)+1
);C$(1)
 370   PRINTTAB(X(4)+D(1)+T-1,Y(4));CHR$(224);TAB(X(4)+D
(1)+T-1,Y(4)+1);CHR$(225)
 380   SOUND2,-15,1,5:PROCDELAY:NEXT
 390 IF N(3)=D(1)+D(2) THEN 600
 400 IF N(3)<D(1)+D(2) THEN 500
 410 DIF=N(3)-D(1)-D(2)
 420 COLOUR2:PRINTTAB(X(3)+N(3)-DIF,Y(3));C$(DIF);TAB(X(
3)+N(3)-DIF,Y(3)+1);C$(DIF)
 440 PRINTTAB(X(3)+N(3)+3,Y(3)+3);STRING$(DIF,D$)
 450 C=1:PROCOVER
 490 PROCWAIT:CLS:GOTO 120
 500 DIF=D(1)+D(2)-N(3)
 510 COLOUR1:PRINTTAB(X(4)+N(3),Y(4));C$(DIF);TAB(X(4)+N
(3),Y(4)+1);C$(DIF)
 530 PRINTTAB(X(3)+N(3)+3,Y(3)+3);STRING$(DIF,D$)
 540 C=2:PROCOVER
 590 PROCWAIT:CLS:GOTO 120
 600 COLOUR3:PRINTTAB(0,25);"WELL DONE!";TAB(0,26);C$(4)

 610 PROCRIGHT:PROCWAIT:CLS
 620 PRINT TAB(2,4)"ANOTHER GAME? Y OR N"
 630 INPUT T$:IF T$="Y" THEN CLS:GOTO 110
 640 IF T$="N" THEN 990
 650 GOTO 620
 990 END
1000 DEFPROCWAIT
1010 PRINTTAB(0,28);"PRESS THE SPACE BAR TO CONTINUE"
1020 Z$=GET$:IFZ$=""THEN1010
1030 IFZ$<>CHR$(32)THEN1010
1040 ENDPROC
1100 DEFPROCDELAY
1110 FOR DELAY = 1 TO 1000:NEXT
1120 ENDPROC
1390 DEFPROCDEFINE
1400 VDU23,224,60,60,60,60,24,126,189,189
1410 VDU23,225,189,60,102,102,102,102,102,102
1420 VDU23,226,3,3,3,3,3,3,255,255
1430 VDU23,227,192,192,192,192,192,192,255,255
1440 VDU23,228,255,255,3,3,3,3,3,3
1450 VDU23,229,255,255,192,192,192,192,192,192
1460 FORA=1 TO10
1470   A$(A)=STRING$(A,CHR$(224))
1480   B$(A)=STRING$(A,CHR$(225))
1490   C$(A)=STRING$(A," ")
1500   NEXT A
1510 VDU23,230,0,0,0,0,0,0,255,0
1520 VDU23,231,0,255,0,0,0,0,0,0
1530 VDU23,232,1,252,255,3,3,255,252,1
1540 VDU23,233,192,32,239,255,255,239,32,192
1550 D$=CHR$(232)+CHR$(233)
1560 X(1)=2:X(2)=20:X(3)=2:X(4)=2
1570 Y(1)=5:Y(2)=5:Y(3)=20:Y(4)=15
1580 ENDPROC
1600 DEFPROCRANDOM
1610 FORD=1TO2
1620   R=RND(10):D(D)=R
1630   NEXT D
1640 ENDPROC
1800 DEFPROCOVER
1810 FORI=97 TO 25 STEP -4:SOUND1,-15,I,1:NEXT
1820 IFC=1 PRINTTAB(0,25);"Your number was too large!":G
OTO 1840
1830 IFC=2 PRINTTAB(0,25);"Your number was too small!"
1840 PRINTTAB(0,26);"Try again!"
1850 ENDPROC
1900 DEFPROCRIGHT
1910 FORI=25 TO 97 STEP 8:SOUND1,-15,I,1:NEXT
1920 ENDPROC
```

Asteroid Lander

Use your BBC Micro to transport you to space and land a craft.

Double danger and difficulty confront you in Asteroid Lander as your ship hurtles through the darkest depths of space.

Your treacherous mission is to safely reach a scientific outpost located on a large asteroid. The scientists at the station have set up an invisible force-shield to protect their base from space bandits.

The sky is cluttered with small asteroids and it is on one of these that the interplanetary terrorist is based. Can you get your ship, with little fuel left, beneath the scientist's shield before the plasma cannon's lethal laser beam blasts you and your craft to smithereens?

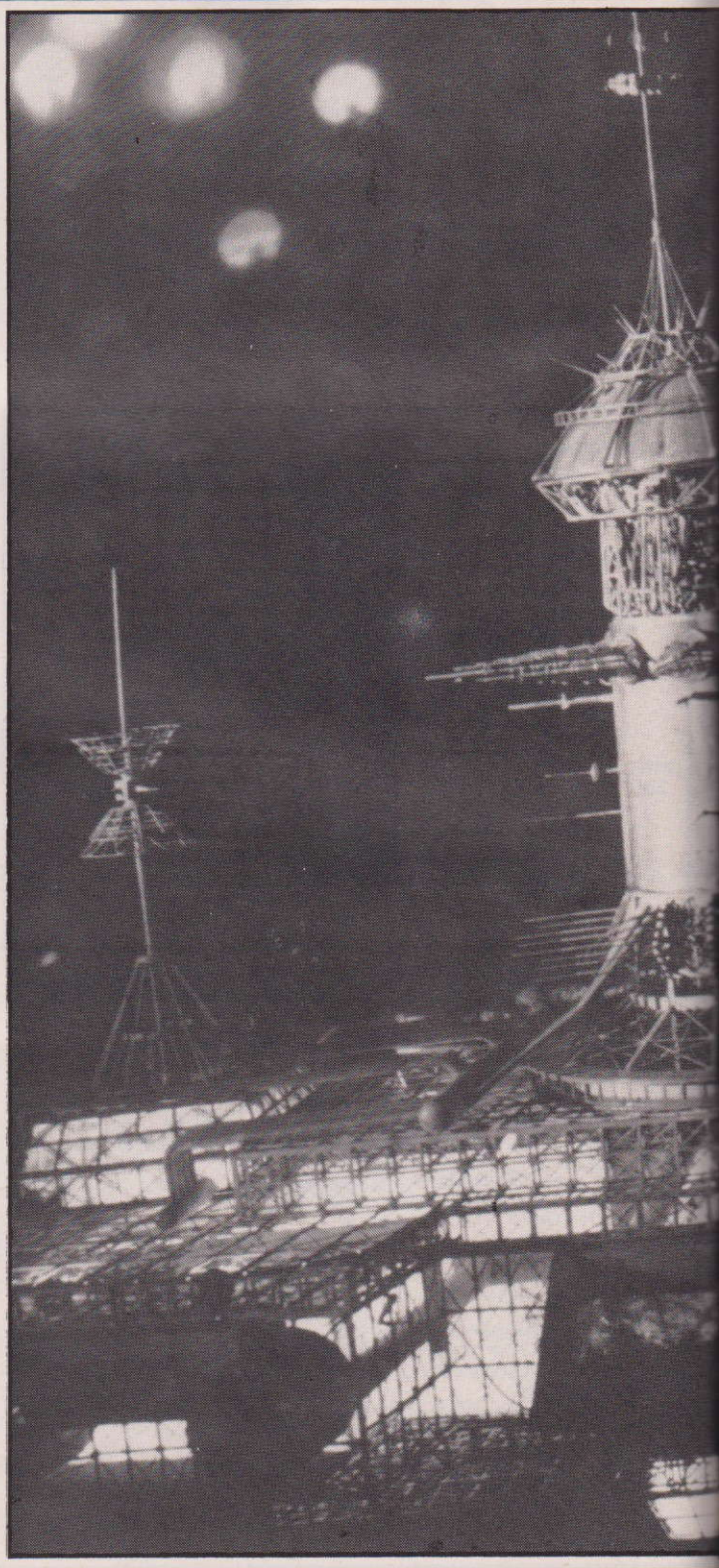
It's a difficult game to play and one requiring a quick finger movement. Level One is hard to handle until you have mastered the controls. Level Three is the most taxing and provides you with maximum frustration. A word of advice — try to keep the lander upright as much as possible to counter gravity pulls.

The best strategy is to find a path as far as possible from the gunman and then hedge hop to the pad beneath the force field.

If you're feeling adventurous, manoeuvre so that the gunman blasts a path through the asteriods for you. It's not advisable at Level One.

The listing is very compact because all non-essential spaces have been omitted to fit the program into the BBC Model A Microcomputer.

So man the controls and happy landings.



PROGRAM DESCRIPTION

The program uses quite a few commands unique to the BBC Micro. VDU 5 and VDU 4 respectively join and separate the graphics and text cursors. When these cursors are joined the move command enables a character to be printed with its upper left most corner at any point on a 1280 by 1024 grid. If this is not possible on your machine then PRINT TAB(X,Y) may be used with X and Y scaled to your machines display; eg if your display is 40 by 25 then X will be INT(LX/32) and Y will be INT(LY/41).

POINT(X,Y) returns the logical colour of the pixel at (X,Y). It should be possible to simulate POINT with PEEK (the scalling mentioned above will also apply).

ASC(d$) is identical to CODE(d$).

VDU 28 defines a text window.

VDU 23 redefines the ASCII character whose code is the number following the 23. The new character is an 8 by 8 grid whose rth row is a representation in binary of the rth number following the code. Thus the syntax is VDU 23, code, row 1, row 2, row 3, up to row 8.

PLOT 69,X,Y prints a point at location X,Y.

PLOT 1,x,y draws a line between locations (X,Y) and (X + x, Y + y) where (X,Y) is the present position of the graphics cursor.

PLOT 85,x,y fills a triangle with vertices (x,y) and the last two places visited by the graphics cursor.

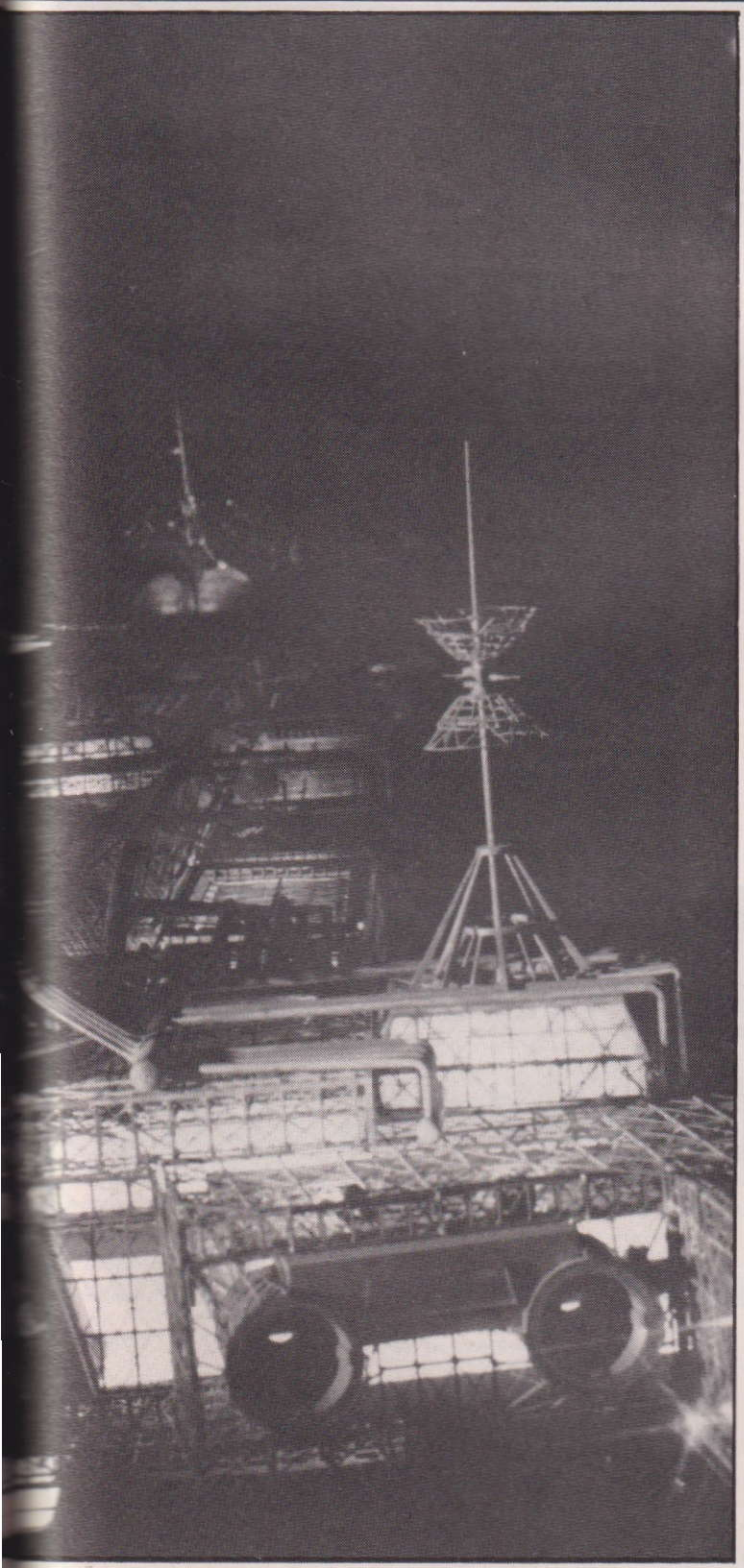
*FX 15,0 clears the keyboard and sound buffers.

SOUND c, v, f, d causes a sound of duration d to be emitted by channel c (white noise if c = 0 and f = 4) while f and v determine frequency and volume respectively.

PROC and ENDPROC can be replaced by a GOSUB to the first line of the procedure, and a RETURN respectively.

REPEAT and UNTIL can be replaced by a single GOTO. Here is an example

```
10 REPEAT
20 ------
```



30 UNTIL condition is true

This can be replaced by (30 IF NOT(condition is true) THEN GOTO 10) TRUE and FALSE always return –1 and 0 respectively.

MODE is used to switch between the various graphics modes of the BBC Micro. In mode 5 there are 4 logical colours which can be thought of as paint pots numbered from 0 to 3. Unless VDU 19 is used to change the colour of paint in a pot then these colours are black, red, yellow and white. GCOL 0,p selects the colour to be used from pot p. GCOL 2,p selects the colour from pot P where P = p added with the colour already on the screen. If you do not have an approximation of the GCOL command then you will have to abandon the special function of line 320. Delete line 220 and replace line 320 with:-

```
320 PRINT
TAB(lx,ly);"  ":PRINT
TAB(LX,LY);CHR$(P):
p = P:lx = LX:ly = LY:A$ =
INKEY$(0):*FX 15,0
```

LX , LY , lx , and ly should of course be scaled appropriately.

Variables Used

D$ = The level of difficulty required by the player.

D, is a numerical variable to which the level of difficulty is passed; it is used to modify, gravity, thrust, number of small asteroids, and the rate of increase of the speed of the plasma bolts.

I = For loop variable used in three loops in the program. It is defined at lines 150, 260, 390.

X = Draw the mountains at the bottom of the playing area.

GX and GY = the horizontal and vertical positions of the gunmans asteroid.

F, defined at line 280, = the fuel remaining.

VX and VY = The horizontal and vertical velocities of the lander respectively.

AX BX AY and BY = The numbers to be added to DX and DY when the lander is turned. The X and Y in the variables name determines which A or B is added to which D, and A is for an anticlockwise turn, whilst B is for a clockwise turn.

DX and DY together fix the direction of any burst of thrust by their addition to VX and VY respectively.

P = The character code of the current image of the lander.

P, defined in line 280 = The previous value of P.

LX and LY respectively are the horizontal and vertical positions of the lander. They are also defined at line 280.

lx and ly = The previous values of LX and LY.

SX and SY = The horizontal and vertical positions of the plasma bolt.

UX an UY = The horizontal and vertical velocities of the plasma bolt.

m = Modify the speed of the plasma bolt.

A$ = Input a character from the keyboard and if it is an 8 or a 0 then the lander turns whilst if it is a 9 then a burst of thrust is imparted to the lander.

sq = Aim the plasma bolt at the lander and ensure that its speed is correct.

LINE DESCRIPTION

20 — 40	Prints instructions 'land on the red pad' and 'Choose difficulty 1, 2, or 3'.
50	The VDU 26 command cancels the effect of the previous VDU 28 command to allow full use of of the screen.

CONTINUED OVER

60 Ensures that the difficluty level input is either 1, 2, or 3.

70 — 80 Define nine characters. The first character is used to represent the small asteroids and the other eight represent the different rotations of the lander.

90 Selects mode time for playing and clears screen.

100 — 140 Draw a frame about the playing area. There are three gauges at the top of the screen to assist the player. From left to right these are vertical velocity, horizontal velocity and fuel.

150 — 210 Draw the frames of the above by the For loop.

230 Prints the F into the the fuel gauge for identification.

240 First draws a line across the screen just below the gauges to complete the frame about the playing area, then paints in the base of the landscape.

250 Paints in the mountains.

260 Places a varying number of small asteroids on the screen depending on the difficulty chosen.

270 Draws in the landing pad.

280 Initialises most of the variables not yet defined.

290 — 370 The section which is executed continuously during the game achieving the motion on the screen.

290 — 310 Blank and update the veritcal velocity gauge.

320 Moves the lander to its new position. It has been specially designed to give a completely flicker free image.

330 — 340 Pass control to the appropriate procedure if a valid input is made.

350 Blanks out the plasma bolt, then updates its position and tests whether it has left the playing area or hit anything. If the test is positive then procedure E at line 550 is executed.

360 Plots the plasma bolt then provides for the effect of gravity on the lander and calculates the lander's new position.

370 Tests four points about the lander to see whether the lander has hit anything. It also tests whether the lander has left the playing area or whether it has been struck by a plasma bolt. If any of these conditions is true then the Repeat Until loop (lines 290 to 370) is executed.

390 Causes the lander to disintigrate if it has not landed lightly on the pad.

400 — 420 Clear the keyboard buffer then print the message "Press escape to step" for four seconds before they return control to line 10.

430 — 500 Defines the procedure T whose function is to turn the lander respecitively clockwise or anticlockwise if either the eight key or the nine key is pressed.

430 — 440 Select the character which represents the new orientation of the lander and update the variables DX and DY which direct any thrust appropriately. The rest of the procedure sets AX, AY, BX, and BY so that DX and DY will be altered correctly the next time the procedure is executed.

510 — 540 Define the procedure t which provides a burst of thrust when the nine key is pressed by altering the lander's horizontal and vertical velocities depending on DX and DY.

510 Emits a tone and leaves the procedure if there is no fuel left.

520 — 530 Deduct the correct amount of fuel and alter the landers vertical and horizontal velocity, then they update the horizontal velocity gauge.

540 Updates the fuel gauge.

560 Defines procedure which blots out anything the plasma bolt has hit, resets the bolt's position to the gun man's asteroid, and aims the bolt at the landers current position.

PROGRAM LISTING

```
L.
   10MODE 7:CLEAR
   20PRINT:PRINT:PRINT CHR$(129):"LAND ON THE RED PAD":PRINT:PRINT:PRI
NT"TURN THE LANDER CLOCKWISE  WITH":PRINT:PRINT"THE 8 KEY. ANTI-CLOCKW
ISE WITH":PRINT:PRINT"THE 0 KEY."
   30PRINT:PRINT"PRESS THE 9 KEY FOR THRUST"
   40PRINT:PRINT:PRINT CHR$(136)"CHOOSE DIFFICULTY 1,2, OR  3"
   50 d$=GET$
   60d=ASCd$-48:IF d<1 OR d>3 THEN GOTO 50
   70VDU 23,224,0,24,60,60,24,0,0,0,23,225,24,60,60,60,24,60,90,90,23,
226,0,56,120,127,124,28,18,16,23,227,0,3,116,255,255,116,3,0,23,228,16
,18,28,124,127,120,48,0,23,229,90,90,60,24,60,60,60,24
   80VDU 23,230,8,72,56,60,254,30,28,0,23,231,0,192,46,255,255,46,192,
0,23,232,0,28,30,254,60,56,72,8
   90 MODE 5
  100MOVE 31,15
  110DRAW 31,1015
  120DRAW 1247,1015
  130DRAW 1247,15
  140DRAW 31,15
  150FOR I=58 TO 826 STEP 384
  160MOVE I,1000
  170 PLOT 1,280,0
  180PLOT 1,0,-44
  190PLOT 1,-280,0
  200PLOT 1,0,44
  210NEXT I
  220VDU 19,2,1,0,0,0
  230PRINT TAB(13,1)"F"
  240MOVE31,940:DRAW1247,940:X=1200:MOVE35,20:MOVE1243,20:PLOT85,35,40
:PLOT 85,1243,40:MOVEX,50
  250REPEAT:X=X-RND(50):PLOT85,X,RND(100)+20:PLOT 85,X+RND(50),20:UNTI
L X<50
  260GX=RND(1000)+100:GY=RND(200)+200:VDU5:MOVEGX,GY:PRINT CHR$(224):F
OR I=1 TO 30+d:MOVE RND(1000)+100,RND(500)+200:PRINT CHR$(224):NEXT I
  270MOVE RND(1000)+140,20:GCOL0,2:PLOT1,0,100:PLOT1,20,0:PLOT1,-40,0
  280F=180:VX=0:VY=0:UX=RND(3):UY=RND(3):AX=-d:BX=d:AY=-d:BY=-d:P=225:
F=P:LX=RND(1000)+140:SX=GX:SY=GY:LY=900:lx=LX:ly=LY:DX=0:DY=d*2:m=10
  290REPEAT VDU4:PRINTTAB(1,1)"    ":TAB(1,1):
  300IF VY<=0 THEN PRINT"^":ELSE PRINT"v":
  310PRINT:ABS(INT(VY)):VDU 5
  320GCOL0,2:MOVElx,ly:PRINT CHR$(F):GCOL 0,1:MOVE LX,LY:PRINT CHR$(F)
:GCOL2,1:MOVElx,ly:PRINT CHR$(F):F=P:lx=LX:ly=LY:A$=INKEY$(0):*FX15,0
  330IF A$="8" OR A$="0" THEN PROCT
  340IF A$="9" THEN PROCt
  350GCOL0,0:PLOT 69,SX,SY:SX=SX+UX:SY=SY+UY:IF POINT(SX,SY)<>0 OR SX<
1200 OR SY<100 OR SY>900ORSY<200THEN PROCe
  360GCOL0,3:PLOT69,SX,SY:VY=VY-d/4:LX=LX+VX:LY=LY+VY
  370UNTIL POINT(LX+32,LY+5)=300RPOINT(LX+15,LY-32)<>10R POINT(LX-5,LY-
16)=30R POINT(LX+47,LY-32)<>10R((LX+32-SX)^2+(LY-16-SY)^2)=10000RLX<400
RLY>940
  380MOVElx,ly
  390IF POINT(LX+32,LY-32)<>2 OR ABS(VX)+ABS(VY)>5THEN SOUND0,-15,4,20
:FORI=1TO100:GCOL0,2:PRINT CHR$(224):;VDU8:GCOL0,0:PRINTCHR$(F);:VDU8:
NEXTI
  400*FX15,0
  410VDU4:PRINT"PRESS ESCAPE TO STOP":TIME=0:REPEAT UNTIL TIME>400:GOT
O 10
  420END
  430DEF PROCT:IF A$="0"THEN P=P+1:DX=DX+AX:DY=DY+AY:IFP=233THENP=225
  440IF A$="8" THEN P=P-1:DX=DX+BX:DY=DY+BY:IF P=224 THEN P=232
  450IF DY>=0THEN AX=-d:BX=d ELSE IF DY<0 THEN AX=d:BX=-d
  460IF DX=d*2THEN BX=-d ELSE IF DX=-d*2 THEN AX=d
  470IF DX>=0 THEN AY=d:BY=-d
  480IF DX<0 THEN BY=d:AY=-d
  490IF DY=d*2 THEN AY=-d ELSE IF DY=-d*2 THEN BY=d
  500ENDPROC
  510DEF PROCt:IF F=0 THEN SOUND1,-5,255,1:ENDPROC
  520F=F-d:VX=VX+DX:VY=VY+DY:VDU4:PRINT TAB(7,1);"    ":TAB(7,1);:IF VX
<0 THEN PRINT"<":ELSEPRINT">";
  530PRINT:ABS(INT(VX))
  540PRINT TAB(14,1);"    ":TAB(14,1):F:VDU5:ENDPROC
  550DEF PROCe:MOVESX-32,SY+16:PRINT CHR$(224):SX=GX:SY=GY:m=m+d:sq=SQ
R((LX-GX)^2+(LY-GY)^2)/m:UX=(LX+32-GX)/sq:UY=(LY-16-GY)/sq
  560IF m>50THEN M=50
  570SOUND0,-10,5,5:ENDPROC
```

Catalogue

D Bishop

You don't need the Cray 1 computer to store information — a BBC will keep catalogues and files like school records, club lists, mailing lists and film lists.

TWO OF A KIND

This program is what is commonly known as a database, allowing you to enter large quantities of data which it will then store away. The software is designed so that once the data has been entered, it will enable the user to manipulate the data easily and quickly, and also to save it on disc or tape for future use.

The uses for this program are infinite — for example, you could use it to keep records of pupils and textbooks in school, addresses and telephone numbers, a company's mailing list, etc. As you can see, the possible uses of the program are manifold.

The program is written for a BBC Micro Model B or Model A with 32K. It has been proven on machines with the 0.1 Operating System and with the 1.2 Operating System. To find what operating system your machine has type: *FX 0 (RETURN). The program has also been proved with discs, as well as tape. I would not recommend that readers with micros other than the BBC try to translate the program directly, line by line, because the BBC has some BASIC which is virtually unique. However, using the listings and notes below as a guide to the structure of a database program and the options one should include in it, then you should be able to write a version for your micro, providing your micro has a fairly large memory.

The program is actually two programs. I have called them CATALOGUE1 and CATALOGUE2, 5.6K and 11.2K long respectively. The former has a pretty title display using Mode 7 graphics and more importantly, the instructions, printed in Mode 1. This program should be entered and saved first. Then the main program, CATALOGUE2, should be entered and saved after the first program. Thus, when you CHAIN 'CAT1', it will load and run; then it will automatically CHAIN 'CAT2'. The reason that the instructions are separated from the rest of the program is that they take up almost 6K of memory, which is better used for storing the data which you enter. CATALOGUE2 will work alone, so if you do not wish to enter CATALOGUE1 then you need not.

CATALOGUE2 is a large program and this limits the amount of memory available for your data. Therefore there are restrictions that must be adhered to. Your data is divided into 'records', which are each divided into 'fields'. Each field must have a name. For example, if you wanted to make a film catalogue, the details about each film are the record. But each record has several fields, for example, four; these contain data on the film title, date when last seen, people who have seen it and a comment — four fields. Each field must have a name, eg FILM TITLE, LAST SEEN, AUDIENCE, and COMMENT. These field names are used throughout the catalogue.

In CATALOGUE2, the maximum number of fields is six, and the lowest number is one. The contents of each field can be between one and twenty-five characters long. You decide these values at the start. I suggest you make the latter short if possible because it saves valuable memory, thus allowing more records to be kept, and it takes less time to record to disc/tape. You could try coding your data, eg instead of putting WORTH SEEING AGAIN, put WSA.

You must also decide on what names to give to each field. The maximum length of name is 10 characters, by default. The program calculates the maximum number of records that it has room for in memory, basing its calculation on how many fields and how many characters per field you want. The more you want of each, the lower the maximum number of records becomes. The maximum varies on a range between 40 and 200 records. Once the catalogue's specifications have been worked out, you may start to fill it with data.

FEATURING

The program has nine major features:

1) View catalogue. This enables you to look through the catalogue, record by record, like flicking through the pages of a book.
2) Search catalogue for a record. You can enter any part of any record and the program will scan the catalogue and present the record which you are looking for; even if you can only remember a few letters it will find a selection of entries similar to these letters.
3) Make a new record. You may add a new record to the catalogue at any time so long as the catalogue is not full, in which case you would start a new catalogue or remove that whole record from the catalogue altogether.
5) Save catalogue onto disc/tape. This option will record your data onto disc or tape, depending on which medium you happen to be using.
6) Verify catalogue on disc/tape. This compares a catalogue on disc/tape with the one in the computer and tells you if they do not exactly match each other. It is useful to do this after you have just saved some data, before you do anything else.
7) Alphabetical sort. You must choose one of your fields for the program to use as the basis for its sort. It will then rearrange the records in the catalogue and put them into alphabetical order. If you have, say, three fields, there are three possible alphabetical orders depending on which field you choose for the subject of the sort.
8) Delete entire catalogue. If you delete the entire catalogue then you must start again by either loading in a catalogue previously saved on disc/tape or designing and filling a new catalogue.
9) Global replace/delete. You must choose one of your fields for the program to work on. You then give the program a piece of data. If you wish to do a global replace, the program will ask for new data with which to replace the old data (in the field which you specified) with the new data. Alternatively, the program will go through the catalogue and delete every record which contains your specified data in your specified field.

More detailed instructions on how to use the program are contained in CATALOGUE1 in the print statements, so I will not go into any more detail here.

VARIABLES USED

ENTRY$(x,y)	This string array holds all the data, where x is the record number and y is the field number.
MAXREC	This variable holds the maximum number of records that there is room for in a catalogue.
NUMREC	This variable holds the number of records in the catalogue at any one time.
NUMFEL	This variable holds the number of fields per record that the user has decided upon.
FELNAME$(x)	This string array holds the names that the user has given to the fields, where x is the field number.
MAXLEN(x)	This numeric array holds the maximum

	lengths that the user has given to the fields' contents where x is the field number.
COL$	This holds the control codes to produce blue print on a cyan background, for one line of text in Mode 7.
FILE$	This holds the name to be used for a file in saving, loading and verifying operations.
ANSWER$, ANSWER	These variables are often used to hold the answer from a user input, temporarily, until the input is acted upon.
N,NO,N1,N2...	These variables are used for different purposes in different parts of the program, often in loops. Other similarly short variables are also very localised in use.

HOW THE PROGRAMS WORK

Readers not familiar with BBC BASIC may find it useful if I outline what is meant by PROC, DEF PROC, ENDPROC, REPEAT, UNTIL and VDU 10, because they occur frequently in the programs. PROCnnn sends execution to a section of program beginning DEF PROCnnn and ending ENDPROC, where 'nnn' is the name. This section of program is executed and then execution returns to the PROCnn.

It is very similar to GOSUB.

REPEAT and UNTIL are put around a section of program, and execution loops around this section, like in a FOR... NEXT loop, until the condition specified after the UNTIL statement becomes true. Execution then goes to the next statement in the program. VDU 10 simply moves the cursor down one line — I use it instead of PRINT.

CATALOGUE

Lines	Description
50	This line may not be necessary on your BBC Micro but it moves all screen display down one line and when in a mode other than Mode 7, it stops the display's vertical wobble (see **User Guide** p435).
60	Switches to Mode 7, draws graphics display, turns off cursor, pauses, turns on cursor and prints prompt.
70	If answer is yes, then switches to Mode 1 and gives instructions.
90	If answer is no or when instructions have been read, then switches to Mode 7, draws graphics display, defines text window, clears it and CHAINs the main program.
100-160	Define procedure 'display'.
110	Colours screen cyan.
130-140	Print the title in double height blue letters.
170-220	Define procedure 'graphic' which prints a strip across screen at level Y.
230-720	Define procedure 'instructions'.
240	Sets COLOUR 1 to cyan and COLOUR 2 to blue.
250	Prints in blue on a cyan background.
260	Prints in white on a black background. COLOUR 1 turns the text cyan and COLOUR 3 turns it white.
730-760	Define procedure 'wait' which is used to break up the instructions into six screenfuls.

CATALOGUE2

Lines	Description
50	This command moves all following screen display down one line.
60	Teletext mode is selected because it leaves the maximum RAM free for users' data.
90-130	This allows the user to load a catalogue from disc/tape immediately upon running the program, as long as the user has a catalogue already saved.
140-230	This section of the program allows the first time user to design the structure of his or her catalogue.
240-260	The total maximum number of characters per record is calculated; N2. MAXREC is calculated from this. DIV means divide and gives an integer as the answer. If MAXREC exceeds 200 then it becomes 200.
290-540	This section is the core of the program. It is a menu of nine options and is only accessible to the user when there is at least one record in the catalogue.
560-730	First of many procedure definitions that make up the rest of the program. Each has a REM statement to clarify the purpose of the procedure.
750-1070	Defines a procedure to load a catalogue from disc/tape.
810	Notice that LEN(FILE$) 3 must not be true. If a number of more than three digits was allowed then this would mean that the whole file name would be more than seven characters long — the maximum length of a file name on discs is seven. Similarly in lines 2030 and 2340.
1090-1240	Define a procedure to allow user to 'flick-through' a catalogue.
1260-1520	This procedure searches the catalogue for a particular record, which the user wishes to see, edit, etc.
1280	The variable FLAG is set to 1.
1370	If a match cannot be found then FLAG is set to 0.
1540-1590	This procedure prints on the screen, record number N4.
1570	The field name is printed in yellow (CHR$(131)) and the entry in white (CHR$(135)). Similarly in line 1800.
1610-1850	This procedure allows the user to change entries or delete records.
1870-1960	This procedure deletes record N1 by moving all records above it in the array down one position in the array and by subtracting one from NUMREC.
1980-2230	Save a catalogue from memory onto disc/tape.
2250-2280	A general purpose error response procedure. It is usually used after illegal numerical inputs.
2300-2630	Compare a catalogue in memory with one on disc/tape.

CONTINUED OVER

2650-2690	Print NUMREC at the bottom of the screen. On the BBC Micro, POS returns the horizontal position and VPOS returns the vertical position of the cursor. So, these values are stored, the printing at the bottom of the screen is done and then the cursor is returned by TAB to its original position.
2710-2770	This section handles errors which occur during saving, loading and verifying. The error codes and their meanings can be found in the **User Guide** (pp 475-482) and in the **Disc System User Guide** (p 77).
2790-3040	This procedure allows global operations on the catalogue in memory.
3060-3170	A much used procedure which gets from the user an input of maximum length N1. It does not allow null responses or leading spaces. It responds to an illegal keystroke by sounding a bleep (VDU 7). It strips trailing spaces off the input as well. The resultant string is B$.
3190-3260	This procedure sets up the ENTRY$ array. Each entry in the array is filled to the maximum length it will ever be, with spaces. This sets aside sufficient space in memory for the users' entries in the catalogue. Each entry in the array is then made null. If this procedure is not done on the BBC Micro and memory is not reserved in this way, the memory rapidly fils with old strings as well as new strings when using the program. This means that valuable memory is wasted. Some micros have an inbuilt 'garbage collection' system to tidy up the memory. The TRS-80 does it automatically and the Apple II has a command FRE.
3280-3530	These two procedures rearrange a catalogue into alphabetical order, using any of the fields as a basis for the sort. Which field is specified by the user. If there is only one field, then a sort is done on this without questioning the user. The algorithm 'Quicksort' is a well-known one, and can be found to be mentioned in any good book which covers methods of sorting. It is typically very fast — faster than the better known and simpler 'Bubblesort'. This implementation is written using integer variables (hence the % symbols) in order to make it even faster.

PROGRAM 1

```
>LIST
   10  REM CATALOGUE1
   20  REM Written by Daniel J. Bishop
   30  REM February 1983
   40  REM Runs on a BBC MICRO Model B
   50  *TV 255,1
   60  MODE 7:PROCdisplay :VDU 23;8202;0;0;0;: FOR PAUSE=
0 TO 5000: NEXT PAUSE:MODE 7:PRINT"Do you wish for instru
ctions ? (Y/N)"
   70  A$=GET$: IF A$="Y" THEN MODE 1: PROCinstructions :
GOTO 90
   80  IF A$<>"N" THEN GOTO 70
   90  MODE 7:PROCdisplay :VDU 28,0,24,39,20 :CLS :CHAIN"
CAT2"
  100  DEF PROCdisplay
  110  FOR Y=0 TO 24: PRINT " ";:VDU 134,157:PRINT" ": NE
XT Y
  120  PROCgraphic(8)
  130  VDU 141,134,157,132:PRINTSPC(12);"CATALOGUE"
  140  VDU 141,134,157,132:PRINTSPC(12);"CATALOGUE"
  150  PROCgraphic(13)
  160  ENDPROC
  170  DEF PROCgraphic(Y)
  180  PRINT TAB(1,Y);CHR$(148);"4";
  190  PRINT STRING$(37,"4");
  200  PRINT TAB(1,Y+1);CHR$(150);CHR$(124);
  210  PRINT STRING$(37,CHR$(124));
  220  ENDPROC
  230  DEF PROCinstructions
  240  VDU 19,1,6,0,0,0,19,2,4,0,0,0
  250  COLOUR 2: COLOUR 129: PRINT SPC(40);"INSTRUCTIONS"
;SPC(28)
  260  COLOUR 3: COLOUR 128
  270  PRINT"This program is called CATALOGUE1. It  cont
ains a graphics display and brief  instructions on the u
se of CATALOGUE2.  When you have finished using this
  program, it will automatically 'chain'  CATALOGUE2."
  280  VDU 10: PRINT"CATALOGUE2 is a database program."
  290  VDU 10:PRINT"Your data is divided into blocks call
ed 'records'. CATALOGUE2 allows 40-200     records to be
kept. Each record has     between 1 and 6 parts, called '
fields'."
  300  VDU 10:PRINT"Next, there follows a more detailed
   description of CATALOGUE2 using extractsfrom the progr
am, which are printed in adifferent colour."
  310  PROCwait
  320  PRINT"If, whilst using CATALOGUE2, you make  an i
llegal response to a question etc., then the computer wil
l tell you so. If  it bleeps this means that you are
  entering too much or too little data. If";
  330  PRINT"this message appears: ";:COLOUR 1:PRINT"The
computer does not understand! Please try again. "
  340  COLOUR 3:PRINT"then this usually means that you ha
ve  entered a number which is too large or  too small.":
VDU 10
  350  COLOUR 1: PRINT"Do you wish to load a catalogue fr
om    disk/tape ?"
  360  COLOUR 3: PRINT"If using the program for the first
 time your answer must be NO, because you do  not have an
y data to load in yet."
  370  VDU 10: COLOUR 1: PRINT"Set up catalogue."
  380  COLOUR 3: PRINT"When using the program for the fir
st    time, you must design the structure of  your catalo
gue. This early part of the  program allows you to do this."
  390  VDU 10: COLOUR 1:PRINT"How many fields per record
(1-6) ?"
  400  COLOUR 3:PRINT"You may decide how many fields you
wish to have, between 1 and 6."
  410  PROCwait
  420  COLOUR 1:PRINT"Please type in the name for field X
"
  430  COLOUR 3:PRINT"X is a number (1-6) in the program.
 Eachfield must have a name, e.g. HEIGHT,    AGE, SEX. A
field name must be no longerthan 10 letters.":VDU 10
  440  COLOUR 1:PRINT"What is max. length of field FFFF
     (1-25) ?"
  450  COLOUR 3:PRINT"FFFF is your field name. You must d
ecidehow many letters you will allow for eachfield. The f
ewer the letters you can    manage with - the better, bec
ause this  allows a greater number of records to bekept -
 up to 200."
  460  VDU 10:COLOUR 1:PRINT"ENTRY Type in the relevant i
nformation  after the prompts."
  470  COLOUR 3:PRINT"Once your catalogue has been constr
uctedthen you may start to fill it with your data.":VDU 1
0
  480  COLOUR 1: PRINT"< PRESS SPACEBAR TO MAKE ANOTHER E
NTRY ><PRESS RETURN TO RETURN TO MAIN MENU >"
  490  COLOUR 3:PRINT"When you are using the program, you
 willcome across messages similar to this.   Choose what
you want to do and press theappropriate key. Unlike most
answers    which you type in, you need not press   RETURN
 afterwards."
  500  PROCwait
  510  COLOUR 1:PRINT"MENU Please choose from the followi
ng   then type the corresponding number."
```

```
  520  COLOUR 3:PRINT"This is the core of the program, to
     which you will often return. Following  this message
 are nine options numbered  1 to 9. Enter your choice.":V
DU 10
  530  COLOUR 1:PRINT"Type in any entry in the record :"
  540  COLOUR 3:PRINT"When searching for a particular rec
ord  which you have made, or changing a       record, you
must give the computer a     clue to which record you are
referring. The computer will search the catalogue  ";
  550  PRINT"and return with all the similar records,e.g.
 you may type JOHN, and the computermay find such entries
 as JO and ELTON    JOHN.":VDU 10
  560  PRINT"When changing a record, you are given   the
choice of changing an entry in a    field or deleting the
 whole record for  good. Enter your choice by typing the
 correct number."
  570  PROCwait
  580  PRINT"When saving your catalogue of records,  your
 catalogue is given a name: FILEx,  where x is a number 1
-3 digits long.    You must choose this number when you a
reasked -"
  590  COLOUR 1:PRINT"What number do you wish to give to
 your catalogue when saved ?"
  600  COLOUR 3:PRINT"You must make a note of this number
,    because it is required also when loadingand verifyin
g.":VDU 10
  610  COLOUR 1:PRINT"ALPHABETICAL SORT Which field do yo
u    wish to do your alphabetical sort on ?"
  620  COLOUR 3:PRINT"This question is followed in the pr
ogramby a numbered list of your field names. The program
 will rearrange all the       records in the catalogue once
 you have  entered your choice, basing its sort on your c
hosen field."
  630  PROCwait
  640  COLOUR 1: PRINT"GLOBAL REPLACE/DELETE"
  650  COLOUR 3:PRINT"In this part of the program, you mu
st   choose a field and enter some data. Thenyou have a c
hoice. (1) The computer willgo through the catalogue, sea
rching for identical data in your specified field."
  660  PRINT"It will replace this data with differentdata
 that you will have just entered.   (2) The computer will
 search the whole  catalogue for identical data in your
  specified field. It will delete every    ";
  670  PRINT"record with this data in this field.":VDU 10
  680  COLOUR 1:PRINT"TOTAL NUMBER OF CATALOGUE RECORDS:
x"
  690  COLOUR 3:PRINT"This message often appears, at the
     bottom of the screen. x is the number.  When have fi
lled the catalogue, you willbe told.":VDU 10:PRINT"Do you
 wish to see the instructions      again ? (Y/N)"
  700  A$=GET$: IF A$="Y" THEN CLS:GOTO 250
  710  IF A$<>"N" THEN GOTO 700
  720  ENDPROC
  730  DEF PROCwait
  740  PRINT TAB(0,31);"< PRESS ANY KEY TO CONTINUE >";:
A$=GET$
  750  CLS
  760  ENDPROC
```

PROGRAM 2

```
   10  REM CATALOGUE2
   20  REM Written by Daniel J. Bishop
   30  REM February 1983
   40  REM Runs on a BBC MICRO Model B
   50  *TV 255
   60  MODE 7
   70  MAXREC=0:NUMREC =0: NUMFEL=0 : COL$=CHR$(134)+CHR$
(157)+CHR$(132)
   80  PROCnumrec
   90  INPUT "Do you wish to load a catalogue from    dis
k/tape ? "ANSWER$
  100  IF ANSWER$= "Y" OR ANSWER$= "YES" THEN PROCdataloa
d : GOTO 300
  110  IF ANSWER$= "N" OR ANSWER$= "NO" THEN GOTO 130
  120  PROCredo: CLS: GOTO 80
  130  CLS
  140  PRINT COL$;"Set up catalogue"
  150  VDU 10
  160  INPUT"How many fields per record (1-6) ? "ANSWER$
  170  NUMFEL=INT( VAL(ANSWER$)): IF NUMFEL <1 OR NUMFEL>
6 THEN PROCredo:CLS:GOTO 160
  180  DIM MAXLEN(NUMFEL),FELNAME$(NUMFEL)
  190  FOR N1=1 TO NUMFEL
  200     PRINT "Please type in the name for field ";N1:P
ROCinput(10):FELNAME$(N1)=B$
  210    PRINT"What is max. length of field ";FELNAME$(N1
):INPUT"(1-25) ? "ANSWER$
  220    MAXLEN(N1)=INT(VAL(ANSWER$)):IF MAXLEN(N1)<1 OR
MAXLEN(N1)>25 THEN PROCredo:CLS:GOTO 210
  230    NEXT N1
  240  N2=0:FOR N1=1 TO NUMFEL:N2=N2+MAXLEN(N1): NEXT N1
  250  MAXREC=6000 DIV N2
  260  IF MAXREC>200 THEN MAXREC=200
  270  PROCfillarray
  280  PROCentry
  290  CLS
  300  PRINT COL$;"MENU"
  310  VDU 10 : PRINT"Please choose from the following th
en   type the corresponding number."
  320  VDU 10 : PRINT"1 View catalogue."
  330  PRINT"2 Search catalogue for a record."
  340  PRINT"3 Make a new record."
  350  PRINT"4 Change or delete a record."
  360  PRINT"5 Save catalogue onto disk/tape."
  370  PRINT"6 Verify catalogue on disk/tape."
  380  PRINT"7 Alphabetical sort."
  390  PRINT"8 Delete entire catalogue."
  400  PRINT"9 Global replace/delete."
  410  VDU 10: PRINT"What is your choice ";
  420  PROCnumrec
  430  INPUT"? " ANSWER$
  440  ANSWER= INT( VAL(ANSWER$) )
  450  IF ANSWER<1 OR ANSWER>10 THEN PROCredo: GOTO 290
  460  IF ANSWER = 1 THEN PROCview : GOTO 290
  470  IF ANSWER = 2  THEN CLS :PRINT COL$;"CATALOGUE SEA
RCH": PROCsearch : GOTO 290
  480  IF ANSWER = 3 THEN PROCentry : GOTO 290
  490  IF ANSWER = 4 THEN PROCedit : GOTO 290
  500  IF ANSWER = 5 THEN PROCdatasave : GOTO 290
  510  IF ANSWER = 6 THEN PROCverify : GOTO 290
  520   IF ANSWER = 7 THEN PROCalphasort:GOTO 290
  530  IF ANSWER = 8 THEN CLEAR : GOTO 60
  540  IF ANSWER = 9 THEN PROCglobal: GOTO 290
  550  REM Proc. for adding a new record to the catalogue
  560  DEF PROCentry
  570  CLS
  580  PRINT COL$;"ENTRY":VDU 10
  590  IF NUMREC=MAXREC THEN PRINT"This catalogue is full
.  Please start a new one.": PRINT"< PRESS RETURN >";:ANS
WER=GET:GOTO 720
  600  PRINT "Type in the relevant information after  the
 prompts."
  610  PROCnumrec
  620  VDU 10 : NUMREC=NUMREC+1
  630  FOR N1=1 TO NUMFEL
  640    PRINT FELNAME$(N1);": ";:PROCinput(MAXLEN(N1)):
ENTRY$(NUMREC,N1)=B$
  650    NEXT N1
  660  VDU10: PRINT"< PRESS SPACEBAR TO MAKE ANOTHER ENTR
Y >";
  670  PRINT"< PRESS RETURN TO RETURN TO MAIN MENU >";
  680  ANSWER=GET
  690  IF ANSWER=13 THEN GOTO 730
  700  IF ANSWER=32 THEN GOTO 570
  710  GOTO 680
  720  CLS
  730  ENDPROC
  740  REM Proc. for loading a catalogue from disk/casset
te
  750  DEF PROCdataload
  760  CLS
  770  PRINT COL$;"CATALOGUE LOAD FROM DISK/TAPE":VDU 10
  780  PRINT "If using tape then position tape in     cas
sette recorder.  Press PLAY button oncassette recorder.
If using disk then  place disk into disk drive."
  790  PROCnumrec
  800  INPUT "What is the number of the catalogue you wis
h to load ? "FILE$
  810  FILE=INT(VAL(FILE$)): IF FILE<0 OR (FILE=0 AND FIL
E$<>"0") OR LEN(FILE$)>3  THEN PROCredo:CLS:GOTO 800
  820  VDU 10 : PRINT "< PRESS SPACEBAR WHEN READY >";
  830  ANSWER =GET: IF ANSWER<>32 THEN 830
```

CONTINUED OVER

```
  840  PRINT ""
  850  ON ERROR  GOTO  2710
  860  FILE$="FILE"+FILE$
  870  X=OPENIN FILE$
  880  PRINT "Loading ";FILE$
  890  NUMREC=0
  900  INPUT#X,NUMFEL: DIM FELNAME$(NUMFEL),MAXLEN(NUMFEL
)
  910  FOR N1=1 TO NUMFEL: INPUT#X,FELNAME$(N1),MAXLEN(N1
):NEXT N1
  920  INPUT#X,MAXREC,N
  930  PROCfillarray
  940  FOR N2=1 TO N: NUMREC=NUMREC+1
  950    FOR N1=1 TO NUMFEL
  960      INPUT#X,ENTRY$(N2,N1)
  970      NEXT N1
  980    PROCnumrec
  990    NEXT N2
 1000  CLOSE#X
 1010  ON ERROR OFF
 1020  VDU 10 :PRINT"The catalogue has been loaded into
    computer memory."
 1030  PROCnumrec
 1040  VDU 10 : PRINT"< PRESS RETURN >";
 1050  ANSWER=GET: IF ANSWER<>13 THEN 1050
 1060  CLS
 1070  ENDPROC
 1080  REM Proc. for enabling user to look through the re
cords of the catalogue
 1090  DEF PROCview
 1100  N=0
 1110  REPEAT
 1120    N=N+1
 1130    CLS
 1140    PRINT COL$; "CATALOGUE"
 1150    VDU 10
 1160    PROCprirec(N)
 1170    VDU 10
 1180    PRINT"< TO CONTINUE PRESS SPACEBAR >"
 1190    PRINT"< TO RETURN TO MENU PRESS RETURN >" ;
 1200    PROCnumrec
 1210    ANSWER = GET
 1220    IF ANSWER <>13 AND ANSWER<>32 THEN GOTO 1210
 1230    UNTIL N=NUMREC OR ANSWER=13
 1240  CLS: ENDPROC
 1250  REM Proc. for scanning catalogue for a particular
record
 1260  DEF PROCsearch
 1270  VDU 10: PRINT "Type in any entry in the record : "
;:PROCinput(25)
 1280  FLAG=1: N3=0: N1=0
 1290  N4=1
 1300  N2=1
 1310  IF LEN(ENTRY$(N4,N2))<=LEN(B$) THEN IF INSTR(B$,EN
TRY$(N4,N2))>0 THEN GOTO 1390
 1320  IF LEN(ENTRY$(N4,N2))>LEN(B$) THEN IF INSTR(ENTRY$
(N4,N2),B$)>0 THEN GOTO 1390
 1330  N2=N2+1
 1340  IF N2<=NUMFEL THEN GOTO 1310
 1350  N2=1: N4=N4+1
 1360  IF N4<=NUMREC THEN GOTO 1310
 1370  IF N3=0 THEN PRINT "Record is not in the catalogue
.": FLAG=0: GOTO 1470
 1380  GOTO 1460
 1390  PROCprirec(N4)
 1400  VDU 10: N1=N4
 1410  PRINT"< PRESS SPACEBAR TO FIND ANY MORE          SI
MILAR RECORDS.  OTHERWISE PRESS        RETURN >";
 1420  ANSWER=GET
 1430  IF ANSWER=13 THEN GOTO 1510
 1440  IF ANSWER<>32 THEN GOTO 1420
 1450  CLS: N3=1:N2=1:N4=N4+1:GOTO 1360
 1460  PRINT "No more similar records in catalogue."
 1470  PRINT"< PRESS RETURN >";
 1480  ANSWER=GET
 1490  IF ANSWER = 13 THEN GOTO 1510
 1500  GOTO 1480
 1510  CLS
 1520  ENDPROC
 1530  REM Proc. for printing an record
 1540  DEF PROCprirec(N)
 1550  VDU 10
 1560  FOR N1=1 TO NUMFEL
 1570    PRINT CHR$(131);FELNAME$(N1);" : ";CHR$(135);ENTR
Y$(N,N1)
 1580    NEXT N1
 1590  ENDPROC
 1600  REM Proc. for changing records in the catalogue
 1610  DEF PROCedit
 1620  CLS
 1630  PRINT COL$;"CHANGE ENTRY"
 1640  PROCsearch
 1650  IF FLAG =0 THEN GOTO 1840
 1660  VDU10
 1670  PRINT"Choose which option you would like from the
list below and enter the correspond-ing number."
 1680  PRINT"1 Delete whole record."
 1690  PRINT"2 Return to the main menu."
 1700  FOR N2=1 TO NUMFEL
 1710    PRINT; N2+2;" Change ";FELNAME$(N2);"."
 1720    NEXT N2
 1730  INPUT"What is your choice ? "ANSWER$
 1740  ANSWER=INT( VAL(ANSWER$))
 1750  IF ANSWER<1 OR ANSWER>NUMFEL+2 THEN PROCredo :CLS:
 GOTO 1660
 1760  IF ANSWER=2 THEN GOTO 1840
 1770  IF ANSWER=1 THEN PROCdelete : IF NUMREC =0  THEN C
LEAR:GOTO 60
 1780  IF ANSWER=1 THEN GOTO 1840
 1790  CLS
 1800  PRINT CHR$(131);FELNAME$(ANSWER-2);" : ";CHR$(135)
;ENTRY$(N1,ANSWER-2)
 1810  VDU 10
 1820  PRINT"Please enter the correct information."
 1830  PRINT FELNAME$(ANSWER-2);" : ";:PROCinput(MAXLEN(A
NSWER-2)):ENTRY$(N1,ANSWER-2)=B$
 1840  CLS
 1850  ENDPROC
 1860  REM Proc. for deleting a record
 1870  DEF PROCdelete
 1880  IF N1=NUMREC THEN GOTO 1950
 1890  REPEAT
 1900    FOR N2 = 1 TO NUMFEL
 1910      ENTRY$(N1,N2)=ENTRY$(N1+1,N2)
 1920      NEXT N2
 1930    N1=N1+1
 1940    UNTIL N1=NUMREC
 1950  NUMREC=NUMREC-1
 1960  ENDPROC
 1970  REM Proc. for saving catalogue to disk/tape
 1980  DEFPROCdatasave
 1990  CLS
 2000  PRINT COL$;"CATALOGUE SAVE TO DISK/TAPE":VDU 10
 2010  PRINT "If you are using tape then position tapein
cassette recorder.  Press PLAY and  RECORD buttons on ca
ssette recorder.  Ifyou are using disk then insert disk i
ntodisk drive."
 2020  INPUT"What number do you wish to give to your cata
logue when saved ? "FILE$
 2030  FILE=INT(VAL(FILE$)): IF FILE<0 OR (FILE=0 AND FIL
E$<>"0") OR LEN(FILE$)>3 THEN PROCredo:CLS:GOTO 2020
 2040  VDU 10
 2050  PROCnumrec
 2060  FILE$="FILE"+FILE$
 2070  ON ERROR GOTO 2710
 2080  X=OPENOUT FILE$
 2090  PRINT "Saving ";FILE$
 2100  PRINT#X,NUMFEL: FOR N1=1 TO NUMFEL: PRINT#X,FELNAM
E$(N1),MAXLEN(N1):NEXT N1:PRINT#X,MAXREC,NUMREC
 2110  FOR N2=1 TO NUMREC: FOR N1=1 TO NUMFEL
 2120      PRINT#X,ENTRY$(N2,N1)
 2130      NEXT N1: NEXT N2
 2140  CLOSE#X
 2150  ON ERROR OFF
 2160  VDU10: PRINT"The catalogue has been saved to
   disk/tape."
 2170  PRINT"< PRESS SPACEBAR TO SAVE IT AGAIN>       < PR
ESS RETURN TO RETURN TO MENU >";
 2180  ANSWER=GET
 2190  IF ANSWER=32 THEN PROCdatasave
 2200  IF ANSWER=13 THEN GOTO 2220
 2210  GOTO 2180
 2220  CLS
 2230  ENDPROC
 2240  REM Proc. for printing exclamation when user puts
in silly answer
```

```
 2250  DEF PROCredo
 2260  VDU 10 : PRINT"The computer does not understand!
     Please try again."
 2270  FOR PAUSE=1 TO 4000: NEXT PAUSE
 2280  ENDPROC
 2290  REM Proc. for comparing catalogue on disk/tape wit
h that in computer memory
 2300  DEF PROCverify
 2310  CLS
 2320  PRINT COL$;"VERIFY CATALOGUE ON DISK/TAPE":VDU 10
 2330  PRINT"If you are using tape then position tapecorr
ectly.  Press PLAY button on        cassette recorder.  I
f using disk then  make sure disk is in disk drive."
 2340  INPUT"What is the number of the catalogue you wish
 to verify ? "FILE$
 2350  FILE=INT(VAL(FILE$)): IF FILE<0 OR (FILE=0 AND FIL
E$<>"0") OR LEN(FILE$)>3 THEN PROCredo:CLS:GOTO 2340
 2360  PROCnumrec
 2370  VDU 10 : PRINT "< PRESS SPACEBAR WHEN READY >";
 2380  ANSWER=GET: IF ANSWER<>32 THEN 2380
 2390  PRINT ""
 2400  ON ERROR GOTO 2710
 2410  FILE$="FILE"+FILE$
 2420  X=OPENIN FILE$
 2430  PRINT "Verifying ";FILE$
 2440  INPUT#X,N1:IF N1<>NUMFEL THEN GOTO 2560
 2450  N1=0:REPEAT:N1=N1+1:INPUT#X,CHECK$,CHECK:UNTIL CHE
CK$<>FELNAME$(N1) OR CHECK<>MAXLEN(N1) OR N1=NUMFEL: IF C
HECK$<>FELNAME$(N1) OR CHECK<>MAXLEN(N1) THEN GOTO 2560
 2460  INPUT#X,N1,N2:IF N1<>MAXREC OR N2<>NUMREC THEN GOT
O 2560
 2470  N1=0:N2=0
 2480  N1=N1+1
 2490  REPEAT: N2=N2+1: INPUT#X,CHECK$: UNTIL CHECK$<>ENT
RY$(N1,N2) OR N2=NUMFEL
 2500  IF CHECK$<>ENTRY$(N1,N2) THEN GOTO 2560
 2510  IF N1=NUMREC THEN GOTO 2530
 2520  N2=0:GOTO 2480
 2530  CLOSE#X
 2540  ON ERROR OFF
 2550  VDU 10: PRINT"The catalogue on disk/tape matches t
hat in the computer.":GOTO 2570
 2560  ON ERROR OFF: VDU10 : PRINT"The catalogue on disk/
tape does not      match that in the computer."
 2570  VDU 10: PRINT "< PRESS SPACEBAR TO VERIFY IT AGAIN
 >    < PRESS RETURN TO RETURN TO MENU >";
 2580  ANSWER=GET
 2590  IF ANSWER=32 THEN PROCverify
 2600  IF ANSWER=13 THEN GOTO 2620
 2610  GOTO 2580
 2620  CLS
 2630  ENDPROC
 2640  REM Proc. for printing number of records
 2650  DEF PROCnumrec
 2660  X2=POS: Y2=VPOS
 2670  PRINT TAB(0,24);CHR$(129);"TOTAL NUMBER OF CATALOG
UE RECORDS: ";NUMREC;
 2680  PRINT TAB(X2,Y2);"";
 2690  ENDPROC
 2700  REM Error handling routine for filing system error
s
 2710  IF ERR=218 OR ERR=6  OR ERR=222 OR ERR=216 OR ERR=
219 OR ERR=217 THEN VDU 10: PRINT "TAPE BAD."
 2720  IF ERR=&D6 OR ERR=&C3 OR ERR=&C6 OR ERR=&BE OR ERR
=&C9 OR ERR=&C7 OR ERR=&C5 OR ERR=&CA THEN VDU 1u:PRINT"D
FS ERROR, CODE ";→ERR
 2730  IF ERR=17 THEN VDU 10: PRINT "ESCAPE pressed."
 2740  VDU 10:PRINT"< PRESS RETURN >";
 2750  ANSWER=GET: IF ANSWER<>13 THEN GOTO 2750
 2760  ON ERROR OFF
 2770  IF NUMREC<=0 THEN GOTO 60 ELSE GOTO 290
 2780  REM Proc. for global operations on the catalogue i
n memory
 2790  DEF PROCglobal
 2800  CLS
 2810  PROCnumrec
 2820  PRINT COL$;"GLOBAL REPLACE/DELETE"
 2830  VDU 10
 2840  IF NUMFEL=1 THEN N0=1:GOTO 2910
 2850  PRINT "Choose which field you are concerned    wit
h:"
 2860  FOR N1=1 TO NUMFEL
 2870    PRINT;N1;" ";FELNAME$(N1)
 2880    NEXT N1
 2890  INPUT"Enter the appropriate number: "ANSWER$
 2900  N0=INT( VAL(ANSWER$)):IF N0<1 OR N0>NUMFEL THEN PR
OCredo:GOTO 2800
 2910  PRINT "Please enter the data to be replaced    or
deleted : ";:PROCinput(MAXLEN(N0)):ANSWER1$=B$
 2920  INPUT"Do you wish to (1) replace, or          (2)
delete ? "N3$
 2930  N3=INT( VAL( N3$))
 2940  IF N3<1 OR N3>2 THEN PROCredo: CLS: GOTO 2920
 2950  IF N3=1 THEN PRINT "Please enter the new part to r
eplace    with : ";:PROCinput(MAXLEN(N0)):ANSWER2$=B$
 2960  N4=0
 2970  REPEAT
 2980    N4=N4+1
 2990    IF N3=1 AND ENTRY$(N4,N0)=ANSWER1$ THEN ENTRY$(N
4,N0)=ANSWER2$
 3000    IF NOT( N3=2 AND ENTRY$(N4,N0)=ANSWER1$ )THEN GO
TO 3030
 3010    N1=N4: PROCdelete :IF NUMREC=0 THEN CLEAR:GOTO 6
0
 3020    N4=N4-1
 3030    UNTIL N4=NUMREC
 3040  ENDPROC
 3050  REM Proc. for handling string input from user of s
pecified length
 3060  DEF PROCinput(N1)
 3070  B$="":N2=0
 3080  N1$=GET$
 3090  IF ( N1$=CHR$(127) OR N1$=CHR$(13) OR N1$=" " ) AN
D N2=0 THEN VDU 7: GOTO 3080
 3100  IF N1$<>CHR$(127) AND N1$<>CHR$(13) AND N2=N1 THEN
 VDU 7: GOTO 3080
 3110  IF N1$=CHR$(127) THEN B$=LEFT$(B$,LEN(B$)-1):N2=N2
-1:GOTO 3140
 3120  IF N1$=CHR$(13) THEN GOTO 3160
 3130  B$=B$+N1$: N2=N2+1
 3140  PRINT N1$;
 3150  GOTO 3080
 3160  IF RIGHT$(B$,1)=" " THEN B$=LEFT$(B$,LEN(B$)-1):GO
TO 3160
 3170  PRINT "":ENDPROC
 3180  REM Proc. to set aside sufficient memory for the E
NTRY$ array
 3190  DEF PROCfillarray
 3200  DIM ENTRY$(MAXREC,NUMFEL)
 3210  FOR N%=1 TO MAXREC
 3220    FOR M%=1 TO NUMFEL
 3230      ENTRY$(N%,M%)=STRING$(MAXLEN(M%)," "):ENTRY$(N
%,M%)=""
 3240      NEXT
 3250    NEXT
 3260  ENDPROC
 3270  REM Proc. for putting catalogue into alphabetical
order using 'Quicksort'
 3280  DEF PROCalphasort
 3290  CLS
 3300  PRINT COL$;"ALPHABETICAL SORT"
 3310  IF NUMFEL=1 THEN N%=1:GOTO 3380
 3320  VDU 10: PRINT"Which field do you wish to do your
    alphabetical sort on ?"
 3330  FOR N1=1 TO NUMFEL
 3340    PRINT;N1;" ";FELNAME$(N1)
 3350    NEXT N1
 3360  INPUT"Type in the correct number: "ANSWER$
 3370  N%=INT (VAL(ANSWER$)): IF N%<1 OR N%>NUMFEL THEN P
ROCredo: GOTO 3290
 3380  VDU 10: PRINT"Sorting..."
 3390  IF NUMREC=1 THEN GOTO 3410
 3400  PROCquicksort(1,NUMREC)
 3410  ENDPROC
 3420  DEF PROCquicksort(P%,R%)
 3430  LOCAL X$,W$,I%,J%
 3440  X$=STRING$(25," "):W$=X$:X$="":W$=""
 3450  I%=P%:J%=R%:X$=ENTRY$((P%+R%) DIV 2,N%)
 3460  REPEAT
 3470    IF ENTRY$(I%,N%)<X$ I%=I%+1:GOTO 3470
 3480    IF X$<ENTRY$(J%,N%) J%=J%-1:GOTO 3480
 3490    IF I%<=J% FOR L%=1 TO NUMFEL:W$=ENTRY$(I%,L%):EN
TRY$(I%,L%)=ENTRY$(J%,L%):ENTRY$(J%,L%)=W$:NEXT:I%=I%+1:J
%=J%-1
 3500    UNTIL I%>J%
 3510  IF P%<J% PROCquicksort(P%,J%)
 3520  IF I%<R% PROCquicksort(I%,R%)
 3530  ENDPROC
```

Software Reviews

This issue a variety of firm's packages are scrutinised.

Title	Knockout
Publisher	The Software House
Machine	Model A & B
Price	£5.95

KNOCKOUT is another version of that industry standard Breakout, where you try to break out of a number of walls, using a bat and ball. It sounds quite simple, but you need to be quick with your actions or you'll never go far enough to get through all the walls to hear the magic sound telling you that you've broken out.

This version of the breakout type game uses Teletext graphics. The colour is really effective. There are six walls and you need to get through all of them to succeed. The best way to reach a high score is when the ball goes behind the blocks. It bounces forwards and backwards increasing your score before it finally comes back.

You use the left and right arrows to move the bat round to catch the ball and send it back. There is a choice of two bat sizes and the speed of the ball can be altered from 1 to 9. To produce a new ball, just press the space bar after you have been wiped out.

If you don't press return at the end of a game, Knockout goes into its demo mode. Knockout is good value for a basic micro game.

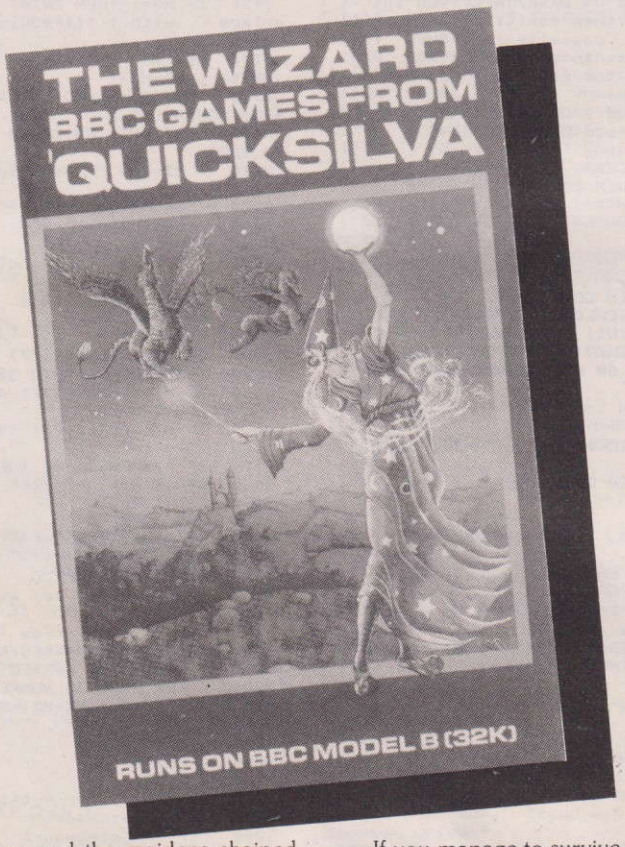
Ratings Table:

SOUNDS	50%
GRAPHICS	60%
DOCUMENTATION	48%
VALUE FOR MONEY	70%
OVERALL	80%

Title	The Wizard
Publisher	Quicksilva
Machine	Model B
Price	£6.95

The **WIZARD** is one of the first three BBC releases from software giants Quicksilva. In it, you play the part of the Wizard and it is your task to prevent an evil demon from capturing the maidens which have been chained to some rocks.

The game, despite sounding like an adventure, is in fact a rather amusing action game. The screen display shows the wizard perched on a rock at the left side of the screen and the maidens chained to rocks away to the right. Above them hover evil winged demons. As the game starts the demons descend and try to grab the maidens. You defend them by shooting down the demons with your wand. The controls enable you to raise or lower the wand and fire.



At first this seems simple but you soon find that your wand has limited power and when that is used up, it takes a long time to recharge. If you let a demon escape, you are in trouble because their next target is you. The demon will move overhead, out of range of your wand and descend. This ends with the demon landing on you and squeezing your brains out in a gory technicolor explosion.

If you manage to survive the first attack wave, then they next try a different tactic. The demons attack in formation from the right side of the screen, sweeping towards you. These two waves repeat themselves, getting nastier each time.

The graphics on the game are good quality and they are complemented by full sound effects. The game isn't particularly fast, but a fair degree of skill is needed to survive for any length of time.

Ratings Table:

SOUNDS	60%
GRAPHICS	75%
DOCUMENTATION	75%
VALUE FOR MONEY	75%
OVERALL	75%

Title	Educational-1
Publisher	Golem Software
Machine	32K Model B
Price	£8.05

This educational tape from Golem Software is aimed at teaching basic numeracy skills to young children. The tape contains six programs. These are Maths 1, Maths 2, Cubecount, Shapes, Memory and Clock. They run on the BBC Model B and all load at 1200 baud.

The first program on the tape is Maths 1. This is designed to help children with simple addition and subtraction. In the program you race the computer along a line by answering questions which the computer sets for you. If you get it right then you move a red ball along a line at the top of the screen. If the answer is wrong then the BBC moves its own ball along a line further down the screen. The first person to reach the winning post is naturally the winner.

The graphics are very basic and the sums set are not exactly demanding. However, given the age range this tape is aimed at, I should think this is acceptable. The second program is virtually the same as the first, except that this time the questions are based on multiplication and division skills.

The theme of the rest of the programs are not so directly mathematical. Cubecount is a counting exercise. As the name suggests, piles of cubes are placed on the screen and the child has to type in how many cubes are shown. This is not as easy as it sounds because of the way the cubes are arranged. Some are hidden, so the child has to work out the total number from the configuration of the cubes.

The last program on the tape is particularly useful. Called Clock, it allows the child to test their recognition of times in relation to the position of hands on a clock face. It offers two options. The first prints the time in hour and minute format and then asks for the hands on the clock face to be moved into the corresponding position. The second option is the opposite of the first. The computer sets the hands on the clock and then asks for the time to be input-

ted, in hours and minutes format.

The programs on the tape were good, but I felt that the maths ones could have been improved by adding a system of correction which would actually help the child learn by his mistakes.

Ratings Table:

SOUNDS	60%
GRAPHICS	65%
DOCUMENTATION	70%
VALUE FOR MONEY	70%
OVERALL	65%

Title	**Skowsh**
Publisher	**Viking Software**
Machine	**Model A or B – Joysticks essential**
Price	**£3.95**

SKWOSH is a computer version of one of the first TV games and follows the original quite closely. The official BBC joysticks (unimaginatively named ANH01) are required to play the game, and two players must take part.

The display, using MODE 4 graphics, is rather rudimentary. White lines border three sides of the blue display. At the bottom of the screen, two short horizontal lines may be moved back and forth by the players. The ball appears as a small white dot poised above one of the bats.

When the appropriate player presses the joystick button, the ball is served. It zooms off towards the top of the screen, bouncing off one of the borders. When it returns, the non-serving player must intercept it with the bat, sending it bouncing back. The game continues until a player fails to intercept the ball with the bat. The other player scores a point. Services are alternated, and the aim is to score eight or more points while remaining two points ahead of your rival's score.

Sound effects are even more basic than those of the original TV game — two kinds of 'beeps' sound as the ball bounces or a player misses a shot. There are ten possible speeds, but at high speeds the graphics are very jerky and the ball seems to bounce before it has made contact with the wall or a bat. It seems a shame that there is no one player option.

Skwosh is a very simple program. It is clearly written using meaningful names for variables and procedures, so that it might be of interest to a budding programmer. The listing is neatly formatted, yet the whole program is only about 3K long. At £3.95, Skwosh is cheap but dull.

Ratings Table:

SOUNDS	10%
GRAPHICS	35%
DOCUMENTATION	50%
VALUE FOR MONEY	40%
OVERALL	30%

Title	**Countdown to Doom**
Publisher	**Acornsoft**
Machine	**Model A or B**
Price	**£9.95**

COUNTDOWN TO DOOM resembles the three previously-published Acornsoft 'Adventure' games. An adventure is a role-playing game, where the player is supplied with a description of an environment, and is invited to suggest some action to take. Actions (such a picking up an item or walking in a given direction) result in changes to the environment. The player's 'score' increases until some set aim has been achieved.

The environment of Countdown to Doom is a classic science-fiction one. While orbiting a planet with the improbable name of Doomawangara (Doom, for short), your spaceship is unfortunately zapped. You crash on the planet's inhospitable surface, where you must hurriedly repair your spaceship before the corrosive atmosphere of Doom makes it impossible for you to escape. Just to complicate matters, the planet is liberally scattered with treasures, which you are reluctant to leave behind. Various alien tricks and traps are distributed across the planet, giving you the chance of a sudden, rather than a lingering death.

Instructions such as 'TAKE GUN', 'QUIT', 'GO NORTH' and so forth may be used. The program is rather dumb and consequently it doesn't understand most instructions, even if you confine yourself to two words. Sometimes it misunderstands a command completely — we typed 'EXPLODE DOOR' and received the reply 'What do you want me to with the explode?'. Other instructions bring the unhelpful reply 'EH?'. The program is written in BASIC but cannot be examined in the usual way.

Countdown to Doom takes over six minutes to load. A partially completed game can be saved on tape if required. If you enjoyed the other Acornsoft adventures you'll probably find this one fun, but Countdown to Doom does not make very good use of the machine's facilities.

Ratings Table:

SOUNDS	0%
GRAPHICS	20%
DOCUMENTATION	75%
VALUE FOR MONEY	70%
OVERALL	65%

Title	**Roadrunner**
Publisher	**A & F Software**
Machine	**Model A or B**
Price	**£6.90**

In **ROADRUNNER** you drive a vehicle around a rectangular course as other cars pass in opposite directions. Now and again drivers shoot at you, aiming to damage your tyres, fuel tank or the twin lasers with which you may fire back. Every so often a group of strange undocumented shapes zoom past. These enemies are shaped like parking-meters and also shoot at you. Other problems include refuelling and 'smoke' which causes some cars to vanish.

Roadrunner is a BASIC game. The graphics of the cars and landscape are quite good, but the animation is unconvincing. Your car does not turn from side to side — it leaps sideways without swerving at all! As you cross from one side of the road to another your car suddenly flips to face the opposite way. There is no 'cornering' effect so that you can turn suddenly through a right angle without ill-effect. The animation is slow and jerky. It is possible to drive sideways through other cars. This usually means damage to your vehicle, but the situation is worse for the other driver, whose car generally splits or vanishes altogether.

Individual sounds and graphics in this game are quite good, but it does not come together as a whole at all. The keys used are not well chosen and there are no alternatives. The display contains an elementary spelling mistake, and the whole game seems to have been rushed to the market without adequate testing.

Ratings Table:

SOUNDS	40%
GRAPHICS	30%
DOCUMENTATION	30%
VALUE FOR MONEY	20%
OVERALL	20%

Title	**Secta Invaders**
Publisher	**Secta Software**
Machine	**32K Model A or B – Joysticks optional**
Price	**£5.95**

SECTA INVADERS is a BBC Micro version of the original 'Invaders' arcade game. When loaded a rather dull page of graphics is slowly drawn, and then a menu (spelt 'menue' on the screen) appears. The menu allows selection of a one-or-two player games, one of five skill levels, and other useful options. The player may elect to use the keyboard or standard joysticks to play. If the keyboard is selected, two combinations of keys are available to suit left or right-handed people. A key to the graphics may be called up and a high-score table can be read or written to tape.

Once the game is started, 64 invaders appear at the top of the screen. Waving their antennae and legs these creatures crawl across the display, slowly descending. They drop two types of bomb — one fast, one slow — down the screen as the player manoevres a laser base and shoots back at them. Irregularly a 'Flying saucer' flies above the melée.

The invaders are all the same colour — blue on a black background. There are no colourful explosions but the graphics work well on a black-and-white screen. However, Secta Invaders is quite well written, using machine-code to emulate all of the features of the arcade original.

CONTINUED OVER

The optional sound effects are different to the arcade game, but still good. It is annoying however, to have to sit through ten seconds of whining and flashing after each laser is destroyed before another is ready for use. Overall though, Secta Invaders is well-produced, and good value for money.

Ratings Table:

SOUNDS	80%
GRAPHICS	65%
DOCUMENTATION	75%
VALUE FOR MONEY	80%
OVERALL	75%

Title	**Crunch**
Publisher	**Oxhey Tutors**
Machine	**Model B**
Price	**£6.50 – cassette**
	£9.50 – disk

CRUNCH is a good example of a well thought out program. It apparently took over nine months to develop and write and this is reflected strongly in the presentation of the program.

Crunch is an educational program designed to teach and improve numeracy skills in children. The program can be used one child at a time or in pairs where the participants compete against each other.

On running the program you are presented with a menu offering a choice of subjects. These are Addition, Subtraction, Division, Multiplication, Mixed Numbers and Directed Numbers. Depending on which one of these options you take you will be presented with another screen which allows you to set the level of the problems, or within what range of numbers the problems should be based around. The final frame asks for an indication of your ability, which can range from weak to super-champ.

Once the level has been selected the game will commence. The question is printed up on the screen and above it is displayed a magenta coloured bar. This bar shows the time you have left to complete the questions. There are 20 questions in all. If you get an answer wrong or take too long in answering it then the question will be held as a correction. If you still have time remaining when you have completed the 20 questions, you will be allowed to go back to any questions you have got wrong and re-answer them. When the time has run out you enter the corrections phase. In this the computer will keep cycling through the questions you failed to answer correctly until you get all of them right.

When this has been completed your score is calculated. If you manage to score 100% then you are immediately placed up a level and the process starts again.

The program is fully error trapped. Only the keys which are necessary to the operation of the program remain active and the Return key isn't implemented to make the program easier to use for non-computer orientated people.

This program offers enough variation and challenge to make it a very useful educational tool.

Ratings Table:

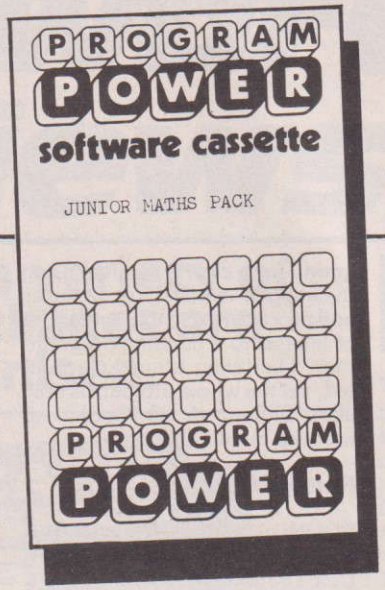
SOUNDS	60%
GRAPHICS	70%
DOCUMENTATION	75%
VALUE FOR MONEY	80%
OVERALL	85%

Title	**Junior Maths Pack**
Publisher	**Program Power**
Machine	**32K Model B**
Price	**£5.95**

THE JUNIOR MATHS PACK is comprised of three programs which are aimed at teaching numeric skills and basic co-ordinates.

All three programs are loaded as one and each is accessed as a selection from a main menu. The first program contained on the tape is called Lander. This is a simple arithmetic program in which you have to prevent an alien from landing. To do this you must answer a question which the computer sets. If you fail to answer correctly the alien moves closer to the surface. A correct answer has the effect of turning on the alien's rocket motor and blasting it further up the screen. The game ends when the alien finally lands.

The second program is designed to co-ordinate notation. It takes the form of a game in which a wizard



has hidden four treasures in a large grid. You enter your guess and provided that you haven't found the treasure, the computer will place a little arrow in the box, pointing to where the treasure is. You then repeat the process for a different box. By repeating this over and over again you can narrow down your search area until you find the treasure. When all four treasures are found the game is over.

Guessses are input as co-ordinate pairs, first the X axis and then the Y axis. Two versions of the program are offered, one which keeps the arrows dislayed on the screen throughout the game and the other which doesn't.

The final program is titled Number Spin. This takes the form of a fruit machine, using numbers instead of fruit. You get Hold and Nudge features and brief instructions which tell you that one of the function keys is a plus, and one a minus. What you are supposed to do escapes me but it appears to work all right.

The graphics used in all the programs were fairly basic and the programs themselves suffered accordingly. Not one of the best educational tapes I have seen.

Ratings Table:

SOUNDS	45%
GRAPHICS	50%
DOCUMENTATION	45%
VALUE FOR MONEY	50%
OVERALL	50%

Title	**Croaker**
Publisher	**Program Power**
Machine	**Model B 32K**
Price	**£6.95**

You guessed it. **CROAKER** is another version of the ever popular Frogger game. For those of you who have not yet encountered this game here is a brief explanation.

You control a small frog who is stuck at the bottom of the screen. In order for him to return safely to his home he must cross a busy road and then jump between logs and turtles across a river. All this has to be achieved for five frogs and against a time limit.

The graphical presentation of the game was good. It used full colour, defined characters for all the game elements, of which all but the frog moved smoothly. If you manage to reach the other side of the river you are rewarded with a beautifully drawn picture of a very smug frog.

Even on the first screen the speed of the traffic and logs is almost too fast. Luckily your frog moves quite quickly but you still have to have fast reflexes to avoid being drowned or squashed. I would have liked to have seen a slightly slower speed to begin with. Someone who had never played this game before could find it very off-putting being repeatedly squashed as soon as you move. I know I would.

The keyboard layout is easy to use. The keys for up/down and left/right are situated at opposite ends of the keyboard.

The speed of the frog also caused me some problems. Having been used to arcade and other computer versions of this game I found myself 'leading' the objects I was jumping to. On other machines this technique works well because the frog doesn't move there immediately. On this version the frog appears in its new position as soon as you push the key. This caused me to come to grief a lot of times before I was able to change my style.

Produced by Program Power, Croaker is a quite acceptable yet not outstanding version of this popular program.

Ratings Table:

SOUNDS	70%
GRAPHICS	75%
DOCUMENTATION	60%
VALUE FOR MONEY	70%
OVERALL	70%

Title	Link 4 Plus
Publisher	ABC Software
Machine	Model A or B
Price	£6.95

LINK 4 PLUS is a computer version of the family game 'Connect Four'. Players take turns to drop counters into one of seven columns. A maximum of six counters may occupy each column. They pile up as they are dropped and the aim is to be the first player to get four of your counters in a row, horizontally, vertically or diagonally.

An entertaining title page introduces Link 4 Plus to the player. These graphics and sounds are the most exciting in the entire package and the main game is tidily and clearly presented. Once the instructions have been shown, the tape is started again and the game program is automatically loaded in their place.

Link 4 Plus follows the rules of the family game. It will act as a games-board for two players, automatically detecting when the game has been won and highlighting the winning line. Alternatively, the computer will play against one person, or even against itself. There are four skill levels, and the computer plays a good game on all except the first. Response time is fast — usually less than ten seconds even at the highest level — and moves are simply entered by pressing the digit key which corresponds to the appropriate column. Unfortunately it is easy to make an accidental move by pressing a key twice.

The main program is written in BASIC, and is about 4K long. It is possible to examine the code, but the program is not very clearly written. There are however, two minor bugs. The program does not check that names are less than 15 characters long, which means that a long name can ruin the graphic display. It also ignores answers such as 'y' or 'n' if they are typed in lower case, although the instructions do mention that.

Link 4 Plus is an entertaining program. If you haven't the time for more complex games like chess, it is a challenging substitute. The rules are simple, but moves can involve considerable thought. The computer plays well and the graphics, while not stunning, are quite acceptable.

Ratings Table:

CLARITY & ADAPTABILITY	45%
SOUNDS	50%
GRAPHICS	60%
DOCUMENTATION	65%
VALUE FOR MONEY	55%
OVERALL	60%

Title	EDG Graphics Package
Publisher	Salamander Software
Machine	Model B
Price	£24.95

The **EDG Graphics Package** from Salamander Software is designed as a comprehensive drawing utility. Using it you can quickly create complex pictures in Mode 0, 1 or 2.

Once loaded you select the mode you wish to start in and the computer sets up the screen display. At the bottom of the screen is a status line. This is comprised of a colour palette, function indicator and coordinate display. The various functions are called by using a system of keyboard mnemonics. Most of these are single key strokes, but a few require the use of the control key.

Your graphics pen is represented by a small flashing cross-hair cursor. This can be moved around the screen using the BBC's cursor keys. It seemed to accelerate as it moved. This feature is useful for quickly moving about the screen, but I found it difficult to control over small distances. This is a pity because most drawing requires accurate positioning of the cursor.

The package offers commands to draw lines, circles and boxes. These can be solid or dotted. The package also offers a fast and accurate fill command. It was nice to see this because on other packages the fill is poorly implemented.

Other functions are shape repitition, text and arc drawing. Using these it is possible to create virtually anything on the screen in whatever colour the mode allows. Colour selection is implemented by moving another cursor across the colour palette.

Once you have created your picture you can save it to tape and reload at a later time. You can also move pictures between modes. To help with copying pictures you can have a grid placed on the screen, thus enabling you to copy pictures bit-by-bit from the original.

The EDG Graphics Package is one of the better drawing utilities available for the BBC. It could have benefitted from a 'Help' command which showed you the action of the commands, but I suppose this would stretch the memory limitations too far.

Ratings Table:

SOUNDS	N/A
GRAPHICS	80%
DOCUMENTATION	65%
VALUE FOR MONEY	70%
OVERALL	75%

Title	Galactic Commander
Publisher	Program Power
Machine	Model B 32K
Price	£6.95

The title **GALACTIC COMMANDER** is not really representative of the game. I was expecting another 'Star Trek' game. Instead the game is a very enjoyable variation on the 'Lunar Lander' theme.

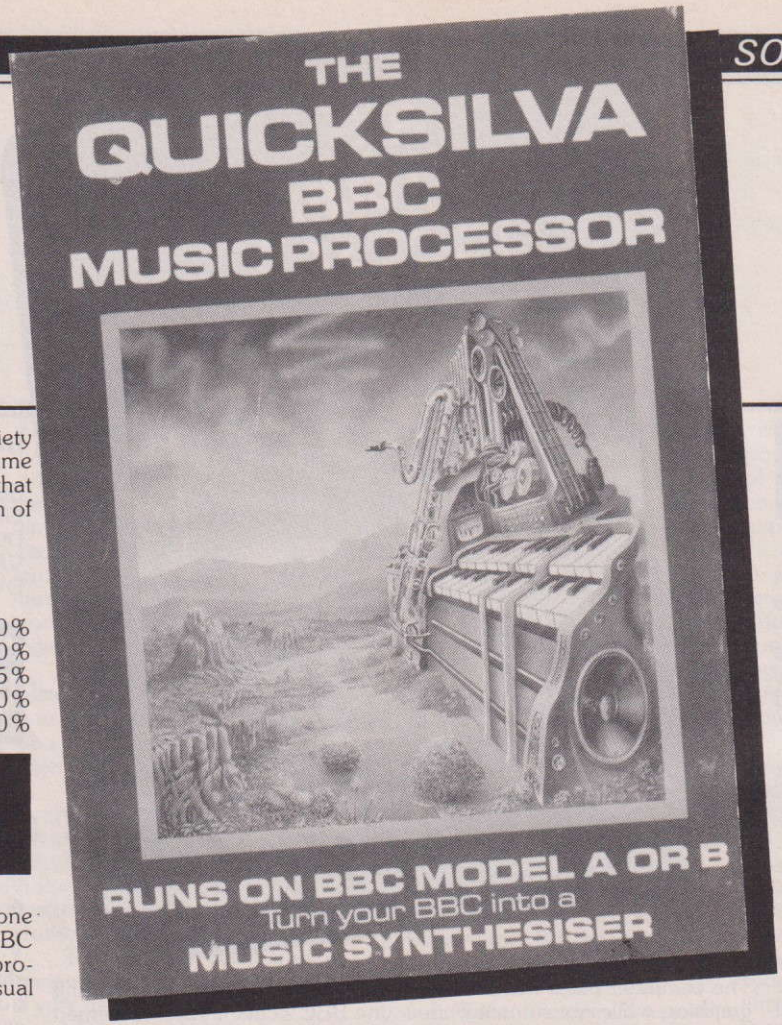
The game has nine levels. You start off trying to land your shuttle safely on the pad under normal gravity conditions. To control the shuttle you use the A and D keys to fire the left and right steering jets. The RETURN key provides control of the main descent engine. As with every lander program you have to lightly drop the shuttle onto its base.

Level 0 is quite easy to accomplish and when you complete it you move up to the next level. Here you are faced with treble gravity so you might plummet to the ground. On higher levels you have to take off from a mother ship, land, pick up cargo, take off and re-dock with the mother ship. This gets increasingly difficult as you have to deal with a rogue gunman who delights in taking the occasional pot shot at you. When you finally reach the last level you have to fly through an underground cavern and avoid moon monsters at the same time. The helpful hint for this section is that it is nearly impossible to accomplish.

The graphics used in the program are superb. Everything is presented in black and white to allow for higher defintion. All the craft and buildings are incredibly detailed and this goes a long way to increase the appeal of the game.

This game is definitely one to get hooked on. The challenge remains through all the levels and it

CONTINUED OVER



is not easy to master. The variety between levels keeps the game fresh and the humerous notes that preface each level add a touch of light relief.

Ratings Table:

SOUNDS	80%
GRAPHICS	90%
DOCUMENTATION	65%
VALUE FOR MONEY	80%
OVERALL	90%

Title	**Starship Command**
Publisher	**Acornsoft**
Machine	**Model B**
Price	**£9.95**

STARSHIP COMMAND is one of the latest releases for the BBC Micro from Acornsoft. The program loads in two parts as is usual and runs on the Model B.

The theme for the game is along the tried and trusted you versus the aliens format. On running the game you are passed through two pages of menus where you can set up various parameters which alter the game play. The action takes place in real time on a single colour, high resolution display. The left side of the screen shows a plan view of the space around your ship. To the right of this are various status displays. At the top is a long range scan, below that is the short range display which includes speed and direction indicators. At the bottom are indicators which display the state of the ship's four energy banks.

The graphics work in such a way that your ship remains stationary in the centre of the screen and steering causes everything else to rotate around you. At this point it is worth mentioning that the quality of all the displays is superb. The programmers have sacrificed the use of colour to provide a lot of extra detail.

At the start of the game your main display shows nothing but the space around you. The long range scan tells a different story. Screaming down on you from the surrounding space are hoards of enemy ships. This is my first criticism of the game. Because you start in the centre of these ships and your ship hasn't got the speed necessary to outrun them, you very soon find yourself surrounded with no way out.

If you have a good understanding of dogfight tactics then you will get on with this game. The alien craft you face behave very intelligently most of the time but occasionally they make some really dumb moves. Unfortunately the lapses don't help you. The two methods of destroying the enemy you can employ are shooting them, which takes about ten direct hits, or ramming.

Ramming a ship should be a last resort as it uses up about half an energy bank and scores you very few points. However, the enemy ships seem to take great delight in flying straight into you and this often is the cause for my rapid demise from the game, very rarely was I 'shot down'.

Your last resort as your energy fails is to launch a escape capsule. If this leaves the battle field safely and you have accumulated enough points starship command will award you with another ship. This is difficult to achieve. Even with a mis-spent youth of video gaming behind me I found myself killed off within the first few minutes more times than I consider reasonable.

Starship Command certainly offers a challenge but I found it frustrating in places, especially when running for my life and avoiding the enemies missiles, only to have some imcompetant starship captain run straight into me.

Ratings Table:

SOUNDS	70%
GRAPHICS	85%
DOCUMENTATION	60%
VALUE FOR MONEY	70%
OVERALL	75%

Title	**BBC Music Processor**
Publisher	**Quicksilva**
Machine	**Model A or B**
Price	**£14.95**

Given the BBC's extensive sound creation and manipulation facilities it is about time that somebody tied them together into a comprehensive package. Music Processor does just that. It turns your BBC Micro into a combined four track digital recorder and four channel synthesiser.

Using the package you are able to compose, edit and play tunes in a bewildering array of voices. Tunes are input into the computers memory by playing them directly on the keyboard. To this end the BBC's keyboard is split into two parts, each covering a different octave range. The function keys control channel selection, envelope shape, noise, volume level, metronome, playback speed and cassette operations. The cursor keys become the tape controls for the digital recorder.

Once you have played your tune into the computer's memory you can rewind the tape back and listen to it. If you hear something that you don't like then you can edit it out or replace it. When you are happy with the music you can change channels and start work on an accompaniment or a harmony. Having finished this you can play both back simultaneously. You continue this process until you have finished your piece.

The program contains predefined envelope shapes to give usable instrument and special effect sounds. If these do not suit the piece of music you want to play you can enter the envelope editor. In this section you can alter any one of the fourteen parameters that go to make up the envelope statement. It is easy to 'fine tune' the sounds by moving backwards and forwards between the main program and the editor, altering values as you go.

For very long pieces you can switch the computer to long play mode. When in this mode each note takes only six bytes of store, as opposed to ten bytes in the normal mode. Once the music has been converted to long play format it can not be edited or changed.

Finally the manual which is supplied with the package is very good. One point, read it completely before starting to use the package, it will save you a lot of time. There are also a few choice typo's mistakes in the manual, including a classic title for one of the prerecorded pieces supplied on the other side of the tape. The second piece is called 'Hark the Herald Angels Sin'.

Overall it is a very good package and a must for anyone musically minded with a BBC Micro.

Ratings Table:

SOUNDS	95%
GRAPHICS	80%
DOCUMENTATION	85%
VALUE FOR MONEY	75%
OVERALL	90%

Machine Specifications

Processor
2MHz 6502A

Memory
16K ROM BASIC
16K ROM Machine Operating System
32K RAM (16K on Model A)

Keyboard
73 key full QWERTY layout, including 10 user-definable function keys, four cursor control keys, two key rollover and auto repeat.

Display
Mixed high resolution graphics and upper and lower case text.
Full Colour on all eight display modes:

0 640x256 two colour graphics and 80x32 text
1 320x256 four colour graphics and 40x32 text
2 160x256 16 colour graphics and 20x32 text
} Model B only
3 80x25 two colour text
4 320x256 two colour graphics and 40x32 text
5 160x256 four colour graphics and 20x32 text
6 40x25 two colour text
7 40x25 Teletext display

UHF, composite video separate RGB and Sync outputs.

External storage
300 baud and 1200 baud interface (with motor control) for standard cassette recorders.

Tone generation
Three-voice music synthesis with full envelope control feeding internal loudspeaker.

Printer interfaces (Model B only)
1 eight bit plus full two-line handshake Centronics port
2 75 bausd to 19,200 baud software selectable serial port to drive RS232 devices with full two-line handshake.

User Input/Output (Model B only)
Eight bit parallel input/output port.

Analogue inputs (Model B only)
Four analogue input for games, paddles or control applications.

Expansion capabilities
Within the computer (at extra cost):

Floppy Disc Interface.
Econet Interface.
Speech Synthesis.
Cartridge ROM Pack Interface.
Tube connector for second processor option.

Second processor option
The computer is designed so that it can be expanded to run with a second processor and considerably expanded memory.

Planned expansion units which are connected via the Tube include:

3 MHz 6502 with 60K RAM.
Z80 with 60K RAM running CP/M.
16 bit processor with 128K RAM.

HARDWARE

Model A BBC Microcomputer System **Price: £299**

A fast, powerfull self-contained computer system generating high resolution colour graphics and capable of synthesising three-part music. The computer is contained in a rigid injection moulded thermoplastic case. The following are contained within the computer thus ensuring the minimum of connecting wires.

- 73 key full travel QWERTY keyboard with 10 user-definable function keys. The keyboard has two key rollover and auto repeat.
- Internal 240/115V power supply is fully encased.
- The internal loudspeaker is driven from a three-voice music synthesis circuit with full ADSR envelope control.
- A fully modulated PAL colour television signal, for connection to a normal domestic television aerial socket, is available through a phono connector.
- A BNC connector supplies a composite video output to drive a black and white or PAL colour monitor.
- A standard audio cassette recorder can be used to record computer programs at 300 or 1200 baud using the CUTS standard tones. The cassette recorder is under full automatic motor control and is connected to the computer via a seven pin DIN connector.
- An interrupt driven elapsed time clock enables real time control and timing of our responses.
- The unit uses a 2 MHz 6502 and includes 16K of Random Access Memory.
- A 16K Read Only Memory (ROM) integrated circuit contains an extensive and powerful Machine Operating System designed to interface easily to high level languages.
- A further 16K 'Language ROM' contains an extremely powerful and fast BASIC interpreter. The interpreter includes a 6502 assembler which enables BASIC statements to be freely mixed with 6502 assembly language.
- Up to four 16K Language ROMs may be plugged into the machine at any time. These four ROMs are 'paged' and may include Pascal, Word Processing, computer aided design software, disc and Econet routines or Teletext acquisition software.
- The standard television output is 625 line 50 Hz, interlaced, fully encouded PAL, modulated on UHF channel 36. Other standards are available.
- The full-colour Teletext display of 40 characters by 25 lines has full character rounding with double height, flashing, coloured background and text — all to the Teletext standard.
- The non-Teletext display modes provide user-definable characters in addition to the standard upper and lower case alphanumeric font. In these modes, graphics may be freely mixed with text. Text characters can be positioned not only on, for example, a 40x32 grid, but at any intermediate position.
- Separate or overlapping text and graphic windows can be easily user-defined over any area of the display. Each of these windows may be filled separately and the text window may be scrolled.
- The Model A is able to support the following modes:

  4 320x256 two colour graphics and 40x32 text.
  5 160x256 four colour graphics and 20x32 text.
  6 40x 25 two colour text.
  7 40x 25 Teletext display.

- All graphics access is 'transparent' resulting in a fast snow-free display.
- Extensive support is provided in the Machine Operating System for the graphics facilities, and this is fully reflected in the BASIC in-

CONTINUED OVER

terpreter. These facilities include the ability to rapidly draw lines and to fill large areas of colour. In addition, very rapid changes of areas of colour can be effected.
— The Model A BBC Microcomputer System can be expanded at any time to the Model B System.

Model B BBC Microcomputer System **Price: £399**

The Model B BBC Microcomputer System is an enhanced version of the Model A Microcomputer but with the following differences:
— 32K Random Access Memory (RAM). This enables all the graphics modes to be used:

0 640x256 two colour graphics and 80x32 text.	(20K)
1 320x256 four colour graphics and 40x32 text.	(20K)
2 160x256 16 colour graphics and 20x32 text.	(20K)
3 80x 25 two colour text.	(16K)
4 320x256 two colour graphics and 40x32 text.	(10K)
5 160x256 four colour graphics and 20x32 text.	(10K)
6 40x 25 two colour text.	(8K)
7 49x 25 Teletext display.	(1K)

— The installed RAM is divided between the high resolution graphics display, the user's program and Machine Operating System variables. The Machine Operating System requires about 3¼K of RAM in the Model A. If higher resolutions are required with large program then the second processor option may be fitted.
— Serial interface to RS423 standard. The new standard has been designed to be inter-operable with RS232C equipment but offers a considerably enhanced spefication — for example, in maximum length of cable and maximum data transfer rates. Baud rates are software selectable between 75 baud and 19,200 baud.
— An eight bit Centronics type parallel printer port is provided with 'Strobe' and 'Acknowledge' lines.
— An eight bit input/output port is also provided.
— Four 12 bit analogue input channels are provided. Each channel has an input voltage range of 0-2.5V and the interval concerter provides a number in the range 0 to 4095 x 16.
— A 1 MHz buffered extension bus is provided for connection to Prestel, Teletext or various other expansion units.
— All interface sockets to external peripherals, are fitted to the Model B. These include R/G/B/Sync for colour monitors, serial interface, parallel printer, disc and tube. Having the interface sockets fitted enables the internal expansions mentioned above to be fitted without further soldering. The Econet interface requires additional components.

EXPANSION

Both Model A and Model B may have the following expansion options fitted internally at purchase, or by Dealers at a later date.

— Floppy disc interface.
— Econet network interface.
— Voice synthesis circuits.
— Cartridge ROM pack interface.
— Various alternative high-level languages in ROM.

External options which plug directly into the machine include:

— Paddles.
— Cassette recorder.
— Black and white and colour monitors and televisions.
— 5¼" single-sided disc drive (100K).
— 5¼" dual double-sided double track density disc drives (800K).
— 80 column dot-matrix printer.
— Daisy wheel printer.
— Teletext acquisition unit
— Prestel acquisition unit

} Both of these enable Telesoftware to be downloaded into the BBC Micro as well as providing access to the normal Teletext/Prestel services. Pages may be 'grabbed' and stored for later use.

— 3MHz 6502 second processor with 60K of RAM.
— Z80 second processor with 60K of RAM and a fully CP/M-compatible operating system.

SOFTWARE

Machine Operating System (MOS)

A 16K ROM is used for the MOS. This software controls all input/output devices using a well defined-interface. The MOS supports the following interrupts:

— Event Timer (used as elapsed time clock).
— Four channel analogue to digital converter.
— Vertical sync.
— Keyboard and keyboard buffer.
— Tube byte transfer.
— Music tone generation and buffer.
— Serial interface, input and output, and buffers.
— Parallel input/output port.

and 'hooks' are provided to support other devices such as:

— Teletext.
— Prestel.
— Econet file system.
— Disc file system.

Many of the operating system calls are vectored to enable the user to change them if required at a future date.

BASIC

The BASIC interpreter is an extremely fast implementation, very close to the Microsoft standard but with numerous powerful extensions:

— Long variable names.
— Integer, floating point and string variables.
— Multi-dimension integer, floating point and string arrays.
— Extensive support for string handling.
— IF...THEN...ELSE.
— REPEAT...UNTIL.
— Multi-line integer, floating point and string functions.
— Procedures.
— Local variables.
— Full recursion on all functions and procedures.
— Effective error trapping and handling.
— Cassette loading and saving of programs and data.
— Full support for the extensive colour graphics facilities.
— Easy control of the built-in music generation circuits.
— Built-in 6502 mnemonic assembler enabling BASIC and assembler to be mixed or pure assembly language programs to be produced.

SERVICE

Technical support and service can be obtained from a number of sources. For details of contacts in your area contact:

BBC Microcomputer System,
PO Box 7,
London W3 6XJ.

EuroCUBE 6502/6809
One small Eurocard - more computer than ever

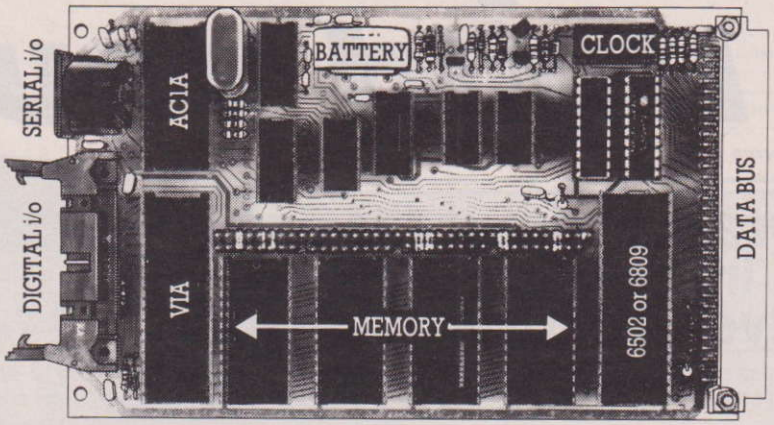
EuroCUBE is just one, small 100×160mm Eurocard but it is an Industrial Computer of exceptional power and versatility.

The user has a choice of 6502 or 6809 microprocessors. Programs can be developed on a full CUBE development Eurorack or on any other microprocessor development system in a high level language such as Industrial BASIC or machine code.

EUROCUBE has four memory sockets whose function and position in memory is fully user selectable. For example, the first socket might contain a 4KB operating system EPROM, the second a BBC BASIC interpreter in ROM, the third the user's BASIC program in whatever size EPROM is convenient, and the fourth might contain 8KB of battery backed non-volatile CMOS RAM.

An on-board calendar clock is also supported by the battery, and provides the time of day to 1/10th second and the date correct for the whole century.

EUROCUBE communicates with the outside world through 20 digital i/o channels, RS423/RS232 serial port and the full CUBE/Acorn data bus through which the entire CUBE/Acorn range of video, disk, analog, solid state switches and other peripherals can be controlled.



£139
Exc. VAT. No charge for Operating System Software

Control Universal's new 150 page catalogue and technical overview is now available free of charge. Contact us for your copy, or ring one of our engineering team to discuss your application.

Control Universal Ltd
The Hardware House

Unit 2, Andersons Court,
Newnham Road, Cambridge CB3 9EZ
Telephone (0223) 358757

EDUCATIONAL SOFTWARE FOR BBC MODEL B (32K)

Suitable for 9-16 year olds

GEOGRAPHY — ITALY: colour graphics, mountains, rivers, cities, etc., tests **£7**

GEOGRAPHY — FRANCE: colour graphics, mountains, rivers, cities, etc., tests **£7**

MATHS — TRANSLATIONS: colour diagrams, explanations, calculations etc., tests **£7**

For use: class instruction by teacher or individual learning by Students

Disc Versions available **£9**

CORONA SOFTWARE
(Dept AB), 73 High Rd., South Woodford, London E18 2QP
Tel: 01-530 3747

Note Invaders (BBC 32K)

At last! Just what music teachers tell us they have been waiting for. NOTE INVADERS by Dereck Riddell does more than any other music teaching program. "Staff" really does teach children (and adults) to read music, while "Invaders" helps increase both speed of reading music and accuracy. Two programs for **£9.25** (inc. VAT) SAE for details to:

CHALKSOFT
(DEPT. AB7)
Lowmoor Cottage, Tondale,
Wellington, Somerset
TA21 0AL

TRADE AND EXPORT ENQUIRIES WELCOME

FOR BBC MICROCOMPUTER

Software

FORTH ROM & documentation..... **£34.72 + VAT**
Manual (for newcomers to FORTH)..... **£6.75**
STATISTICS PACKAGE (on 5¼" Disc).... **£19.75 + VAT**

Hardware & Accessories

TORCH drives & operating system..... **£780.00 + VAT**
Printers drives etc..... **(Please phone for quote)**

FOR EPSON HX20

Software

FORTH ROM — documentation..... **£34.72 + VAT**
(Manual available shortly)

Hardware & Accessories

EXPANSION UNIT, USV Interface etc
..... **(Please phone for quote)**

COMING SOON FOR BBC MICRO

FORTH FLOATING POINT EXTENSION (ROM)
COMPUTER AIDED LEARNING PACKAGE

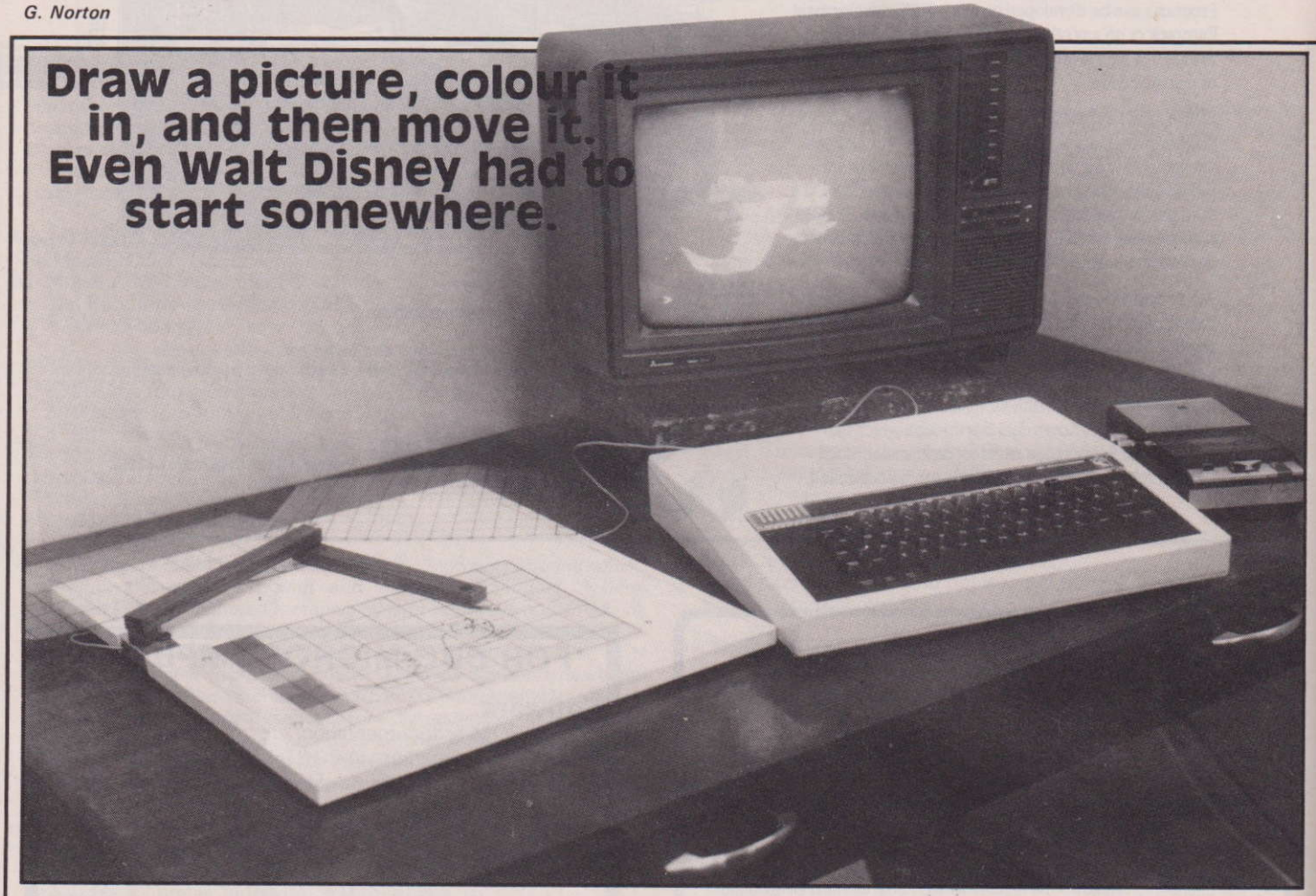
OTHER MACHINES:
Dragon 32, Colon Genie, Apple, SWTPc

H.C.C.S. Associates

533 Durham Road, Low Fell, Gateshead,
Tyne & Wear NE9 5EY Tel: (0632) 821924

Graphics Tablet: Build Your Own

G. Norton



Imagine being able to easily copy diagrams and pictures and colour them just as you would with a coloured pencil or to change scale, rotate and move shapes about the screen until they are in the correct position so that a complex diagram can be built up.

To do this requires some method of interacting with the computer and display. This must be directly related to the diagram to be copied.

Of course there are many ways of achieving this. It might be instructive before we go on to discuss a simple graphics tablet which we can construct ourselves. Similarly, to look at some of the many methods which have been used over the past 30 years.

Naturally, interactive computers have not been available during the whole of this period. It's only during the past few years with the introduction of the microcomputer that such facilities have been generally possible.

FIRST UP

Of all the methods which are available for interaction with the computer and display, the graphics tablet provides the operator with one of the most direct and versatile means of passing information into the computer.

One of the first graphics tablets that I learnt about many years ago, was a simple wooden pantograph used to copy pictures and change scale before colouring them in. This was a simple criss-cross arrangement of wooden slats, fixed at one end and carrying a pencil at the other end. A nail or some such pointer part way along the contraption served as a stylus to trace around the diagram to be copied. No electronic computers then, to help them design!

Modern interactive methods make use of many different phenomena for passing the co-ordinate position of the stylus into the computer. The quantity used in defining this position may be due to some predefined code, conventional keyboard input, time, spatial, or even direct interaction through a video camera.

INTERACTION

One of the simplest methods makes use of the function keys. This may be regarded as the predefined method. For example, we might define KEY0 as *KEY0 MOVE 640, 512: DRAW 800, 900:M to draw some predefined line. Similarly, we might store simple routines such as circles or squares, in the function keys.

Function keys or buttons have also been positioned in conjunction with the visual display unit (VDU) which corresponds to data written in menu form. They have also been used with some mechanically coded overlay. This is so that the system 'knows' the visual data that the button is referring to.

A slightly more flexible approach is to use two or more predetermined keys on the keyboard. This provides 'flying' cursors or draws lines in a particular direction (usually two orthogonal directions) by repeatedly pressing a particular designated key. This is the approach usually adopted in many arcade games.

A particularly fast way of interacting with the screen, consists of touch wires or transparent conductive membranes stretched across the VDU screen. Data is written on the screen immediately over each wire. The system will interpret the operators action mere-

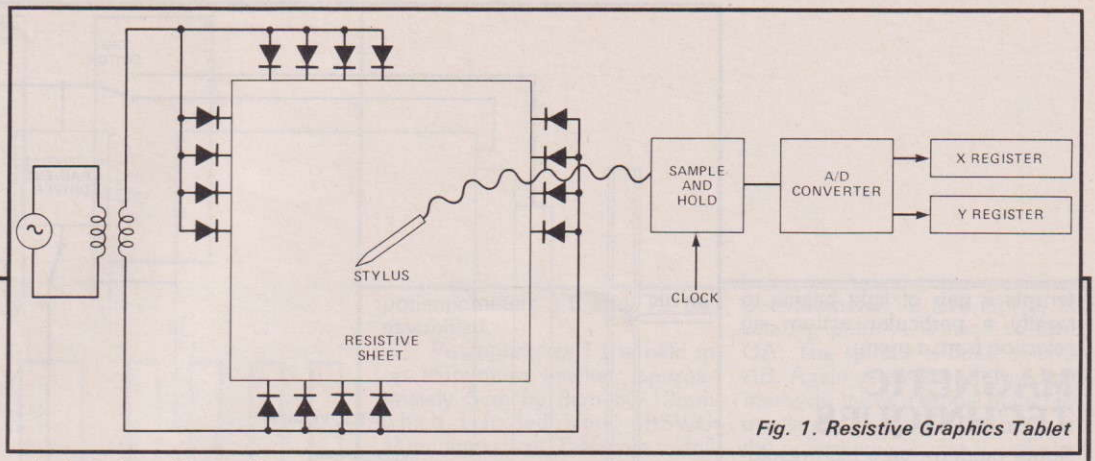


Fig. 1. Resistive Graphics Tablet

ly by momentarily touching the wire as an indication of the command or data requested. Sometimes though, this can be too fast, since operators get so used to the operation that they anticipate the computer's prompt and react too quickly thereby causing an error.

THE SPEED OF LIGHT

A light pen also interacts directly with the graphics screen. It performs a similar function to the stylus on a graphics tablet. If you have a transparent overlay on the screen however, it is not easy to copy diagrams. Modern light pens can be used to draw and select functions from menus written on the screen. It can also read bar codes and pick up colours from a 'pallette' enabling the user to effectively 'paint' on the screen. The pulse detected by the light pen is caused by the momentary flash of ultra-violet radiation. This is emitted when the phosphor of the cathode-ray tube is energised by the transient electron beam. This instant in time can then be used to access the display address for identification or modification.

In raster scan displays, the electron beam systematically covers the whole display 50 times per second. The position of the light pen is uniquely defined in relation to the x,y position on the screen. A counter used to indicate the line number of the raster scan can be used as a measure of the y co-ordinate. The time from the beginning of the line is used as a measure of the x co-ordinate. Resolutions of about 0.2% are obtainable by this approach.

As already mentioned, true graphics tablets have been designed in a variety of forms. Those which are commercially available can be very expensive, costing as much as or often more than the computer itself. In its simplest form, a graphics tablet provides a means whereby a cursor (whether it be a flashing square, crossed lines or an arrow) can be manoeuvred over the screen in response to a mechanical or hand held stylus. Visual feedback allows the operator to adjust for any mispositioning of the cursor. It doesn't really matter in this case whether the tablet itself is non-linear. In fact tests have shown that operators are quite happy to move a stylus in a curve, providing the visual line on the VDU is straight. Joysticks come under this category, as do rolling balls, the light pen and the 'Mouse'.

INVITATION TO THE BALL

The rolling ball and the 'Mouse' are similar in as much as they produce two orthogonal rotary motions to activate sensors. The ball rests against four freely rotating axles and these combine to turn a pair of potentiometers or digitally encoded discs to give a readout of the x and y co-ordinates. The 'Mouse' uses a pair of orthogonal wheels, again providing analogue or digital information into the computer.

When analogue systems are used, an analogue to digital (A/D) converter is required. This translates some voltage derived positional information with an acceptable degree of resolution. Fortunately the BBC Model B (or Model A with upgrade) has an A/D converter built in. A relatively simple but effective graphics tablet using this facility, based on a polar co-ordinate translation system will be described later. Direct rectilinear co-ordinate translation can also be used. Either can incorporate an analogue transducer or digital transducer.

Digital tablets make use of many discrete sensors and transmitters, devices like digitally encoded discs or Moire fringe counters. Digital tablets have the advantage that they are not subject to drift. The resolving power, however, is limited by the packing density that is permissable for each position. Analogue techniques permit greater resolving power but the absolute accuracy is limited by the linearity of the transducer. It is also subject to noise or jitter of the co-ordinate value.

Professional graphics tablets may achieve resolving powers of one in 1000 with an accuracy of 0.01%. Lower cost tablets have accuracies of about 5%. The graphics tablet to be described in this article has an accuracy of less than 5%.

SPECIAL EFFECTS

There are various physical phenomena that are used in graphics tablets. These include such effects as sheet resistance, acoustics, infra-red optical sensors, magnetics and electrostatics (capacitive effects). A simple graphics tablet, for instance, can be constructed using resistive sheet such as Teledeltos paper. This is paper with a graphite coating on its surface. An AC voltage gradient is imposed across the sheet. The potential at any point is detected by a stylus making contact with the surface to give an indication of its x,y co-ordinate. This potential is also detected at the AC rate. Figure 1 shows the principles involved.

The accuracy of such a graphics tablet, however, is dependent on a number of factors. These include the resistive non-linearity of the sheet. A number of alternative methods have been evolved for overcoming boundaries. Biassed diodes are used as feed points down the edges, and distributive resistance coils connected along the sheet boundaries. Others have found ways of using alternating currents and capacitive effects to allow overlays to be used between the stylus and the tablet.

A further alternative is to use two resistive sheets separated by a cellular membrane. This relies on the pressure of the stylus to make contact through the membrane. (Figure 2). A basic problem with the pressure pad is that errors can be caused by applying pressure inadvertently on other parts of the tablet during use. Usually, one has to have something like a hand rest or trolly mounted above the surface.

A number of tablets use a method of measuring the time delay between an acoustic transmitter and receiver fitted to the edge of the tablet. In one type, the receivers are strip microphones, one for the x co-ordinate and one for the y co-ordinate. A stylus which generates a spark, starts counters which are subsequently stopped when the acoustic wavefront meets the microphones. The time delay so measured, gives a measure of the x and y value. Accuracies of 0.1% full scale are claimed for this device. Such a system can also be extended into three dimensions by having strip microphones for each of the three orthogonal axes. Any large object, however, could modify the acoustic wavefronts and cause distortion.

In a similar fashion, piezo-electric transducers have been used. They launch surface waves in the glass surface of a tablet using a high frequency pulse modulated wave. A stylus, or even the application of finger to the surface, reflects the wave back to the source. By alternately reversing the modes of the transducers from transmitters to receivers, the time delay between the transmitter and the reflected pulse again measures the x, y co-ordinate, (Figure 3).

OPTICAL TABLETS

Light pens, of course, come under this category and I have already mentioned these in a previous section. However, infra-red optical tablets have also been devised, using a row of infra-red sensors and transmitters along the edges of the tablet or indeed around the edges of the VDU. These project orthogonal beams of light across the surface of the tube, such that by placing a finger on the screen, it in-

CONTINUED OVER

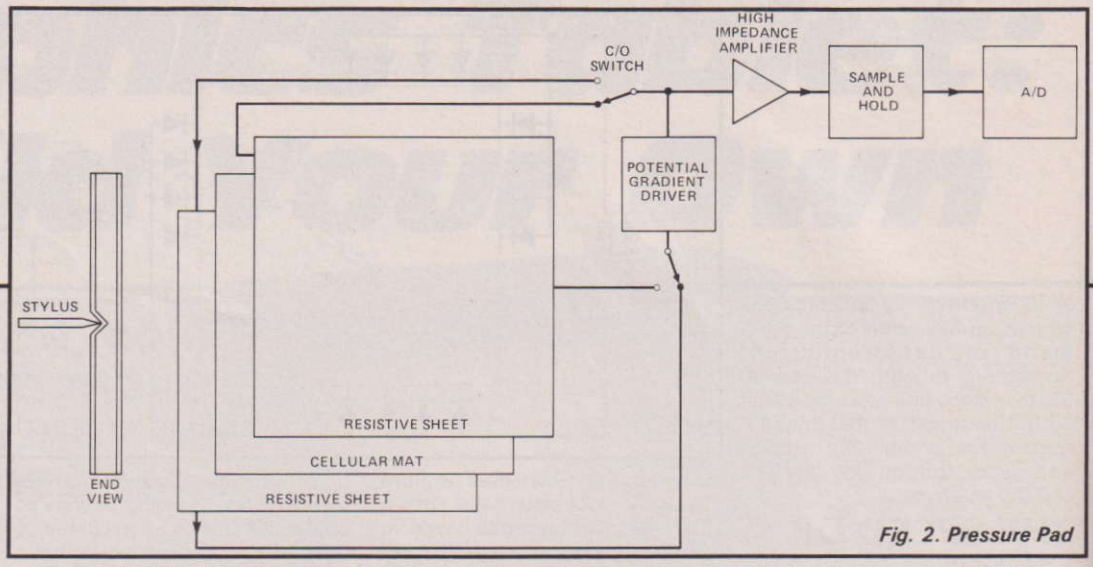


Fig. 2. Pressure Pad

terupts a pair of light beams to signify a particular action, eg selection from a menu.

MAGNETIC TECHNIQUES

Some modern, very high resolution tablets use magnetics as the transmitting and receiving mechanism. They use a tablet which has embedded into it a set of very fine wire loops and a stylus which carries a coil energised with 1 microsecond pulses at a repetition rate of 5KHz. The polarity of the signal received can be deduced whether the stylus is inside or outside a particular loop.

By using a combination of separate loops laid one on top of the other, of the form shown in Fig. 4, the outputs of the windings may be compared with a reference signal and gated appropriately. A code is then generated corresponding to the position of the stylus in the x direction. A similar arrangement is used to cater for the y co-ordinate. A binary unit distance code is then used. With the pattern of windings chosen, a resolution of one in 2^{10} requires 11 windings in each of the x and y directions. Figure 4 shows the arrangement of the windings and how the coded output is derived.

CAPACITIVE TECHNIQUES

Capacitive tablets have been made in much the same way as the magnetic tablets. A series of insulated grid wires in the X and Y directions are connected to a set of overlying and therefore capacitively coupled pads which lie down each side of the tablet. By applying a series of pulses to these pads, a digital code is produced in the stylus. This is directly related to its co-ordinate position on the surface. In practice a Gray-code is used to prevent errors in moving to an adjacent position.

A commercial version of this tablet provided a writing surface of 10 inches square with a resolution of about 1/100th of an inch.

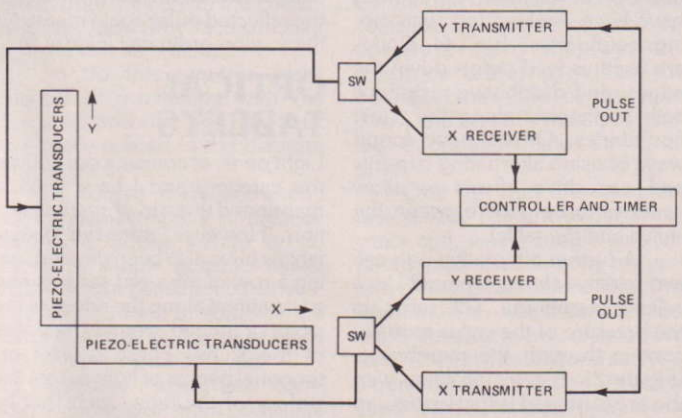
VIDEO GRAPHICS

Just to complete the picture, a modern sophisticated way of transferring diagrams and pictures into the computer, is to use a video digitizer.

Without going into too much detail, the video digitizer accepts an input from a video source such as a closed circuit television camera. It converts the analogue video signal to digital data, and transfers this data to the computer under software control.

The advantages of this system is that you don't have to trace round an object, nor does the object have to be flat. In fact, in principle, you could store an outside scene in the computer. You could then colour it in with the use of a light pen. The only disadvantage though is the cost. Video digitizers plus the camera often cost more than the computer.

Fig. 3 Accoustic Wave Graphics Tablet



A GRAPHICS TABLET FOR THE BBC MICRO

Now that we have had a look at some of the different types of graphics tablets which have been produced over the years, we will now concentrate on one which we can use on the BBC Micro.

Although simple in concept, it uses many of the features discussed in the previous sections. It includes copying diagrams, selecting and filling in colours and selecting commands from a menu, such as accurate line drawing or circle drawing routines, etc. The accuracy isn't as high as some of the more sophisticated units. The graphics tablet nevertheless can achieve better than 5%, which is quite adequate for most purposes such as copying pictures.

This graphics tablet is in effect a large joystick, but capable of more control than the latter. Fortunately, the BBC 'B' (or upgraded 'A') computer has an analogue to digital (A/D) input port to which paddles or joysticks are normally connected. This port is therefore used for this application.

Essentially, the x, y co-ordinates for any plotting point are calculated on a dual polar co-ordinate principle. Figure 5 shows the basic co-ordinate geometry used.

Line OA and AB represent the two respective arms of the plotter. To make life easy, both arms are made the same length, L. The exact length doesn't really matter at present since this is eventually taken care of during calibration. Suffice it to say that the arms should be long enough to reach the extent of the plotting area, but more of that later.

Line OA is free to rotate about O to form an angle with the X axis. AB is free to rotate about A to form an angle θ with OA. Hence B is uniquely defined in polar co-ordinates by the expression:

$$x = L * (\cos(\theta) - \cos(\theta + \psi))$$
$$y = L * (\sin(\theta) - \sin(\theta + \psi))$$

These angles can now be measured quite accurately by using suitable potentiometers at the rotating points and deriving a voltage proportional to the angle moved. This voltage is then passed to the A/D channel in exactly the same way as with joysticks.

The problem, however, is to know exactly what angle is being set up for θ and ψ. Any non-linearity or offset (for instance when OA is in line with the X axis) will grossly distort the picture. For angles of θ near to zero, SIN(θ) increases rapidly, whilst COS(θ) moves only slowly. The potentiometers themselves have a large angular movement and no angular reference point. Therefore there will obviously be undefinable errors.

One way to take account of these errors is to make sure that the potentiometers are highly linear (<0.1%). Also, the mechanical assembly must be highly accurate. This is so that the

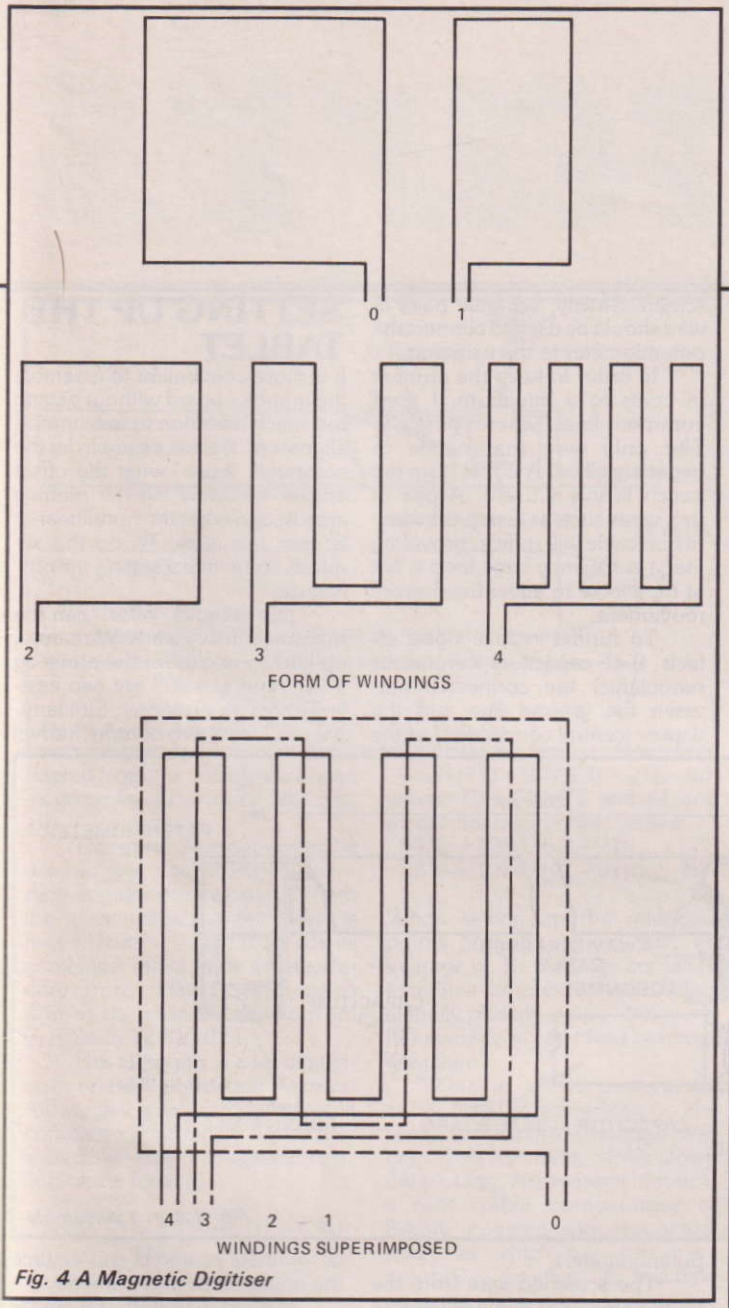


Fig. 4 A Magnetic Digitiser

precise position of the arms and potentiometer angles are all accurately determined and known. But this is exceedingly tedious and expensive. What we want to be able to do, is to put something together in the most convenient way, and let the computer do the work. After all, since we have paid a lot for the computer, it might just as well earn its keep.

This design then, allows us to do just that. Most potentiometers you will find have an electrical travel of somewhere between 240° and 360°. Relatively inexpensive devices can be purchased which have linearities of about +/−5%.

I used two 10K Bourns series 91 potentiometers, which are low torque, plastic film devices with an electrical angle of 240° (4/3 *PI radians). The series 91 potentiometers are housed in a rectangular package. This is convenient as they fit neatly into some rectangular plastic channelling, which I purchased from a local hardware store.

The baseboard is a piece of 12mm by 380mm by 450mm melamine-covered chip board with a plastic edging surround, mitred at the corners to make it neat. The only additional work needed on the baseboard is to make a small rectangular cutout (25mm by 30mm) to accept channel 1 potentiometer (T1) assembly. Four shelf adhesive rubber feet give protection to polished surfaces. Figure 6 shows diagramatically the complete graphics tablet.

Figures 7 and 8 shows how potentiometers T1 and T2 are assembled.

Potentiometer T1 is held in an aluminium bracket, approximately 5cm by 3cm by 12mm which I folded from 16SWG aluminium sheet. Take care to drill all fixing holes and cut the 10mm slot for the potentiometer spindle bush first.

The method of attaching the potentiometer spindle to the graphics arm is a matter of choice according to the means at your disposal. The arms for instance (which were 205mm long) could be constructed from wood (20mm by 10mm) a 6.35mm (¼in.) hole drilled at one end could take the spindle if fixed with a suitable screw.

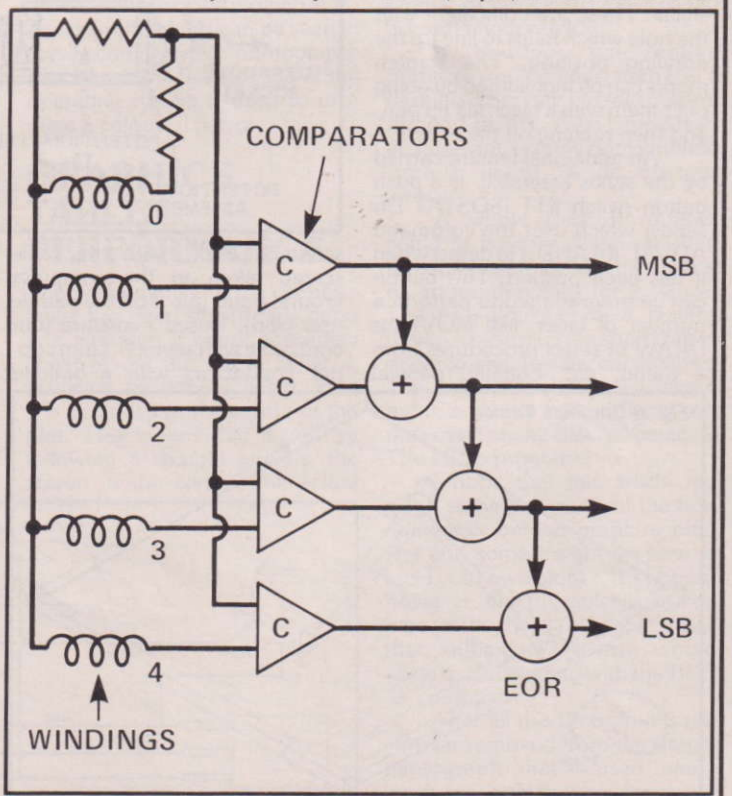
Using plastic moulding, I found it neater and more convenient to machine a piece of rectangular solid PVC on a lathe and assemble the whole unit with one fixing pin. This allowed me the facility of a lifting arm, similar to a record player arm.

In a similar way, the body of potentiometer T2 (see Fig. 8) at rotating point A is fixed to arm OA. The spindle is fitted to arm AB. Again using plastic moulding methods, it was a simple matter to use 'super glue' to fix the body of the potentiometer into the plastic channel. A solid PVC block similar to that at point O provided a means for fixing the spindle. It also provided a sliding 'foot' to keep the arm just a few millimetres off the drawing surface.

A LOW PROFILE

A similar arrangement could also be accomplished using wooden arms. Fixing the body of the potentiometer with either 'super glue' or resin glue. The spindle is now attached to the arm AB in a similar manner to arm OA. If you want to keep the profile low, it may be necessary to reduce the length of the spindle and even cut a few millimetres off the bush. Take care not to cut right through the spindle at this point!

Fixing the spindles with the screw (or pin) is not critical. One



CONTINUED OVER

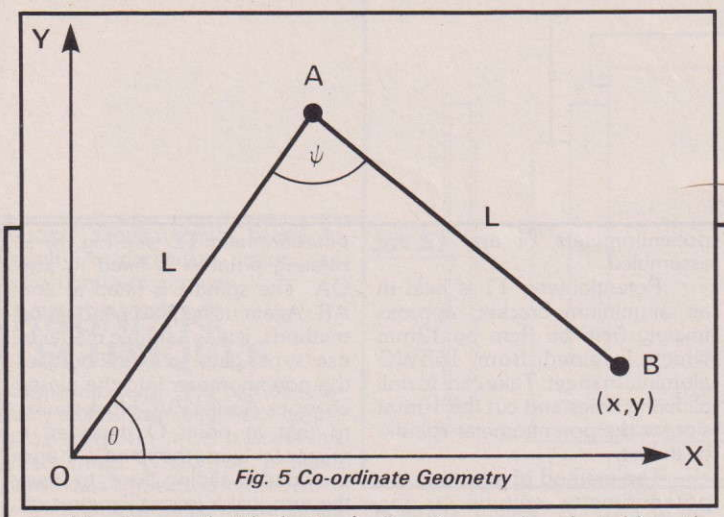


Fig. 5 Co-ordinate Geometry

should make sure that the limits of travel have not been reached for all likely positions of the arms. Arm AB was actually made slightly shorter than arm OA, so that a small plastic stylus could be fixed at its free end. This consisted of a little acrylate assembly, again fixed with 'super glue' (see Fig. 8). It was shaped with a file to form a pointer with a 2.5mm hole drilled in it at the correct distance (205mm) from the axis of potentiometer T2. This hole can either be used to sight through or even used as a convenient location for a pen or pencil. A pair of 'cross hairs' are scratched on the underside of the stylus. These are coincident with the hole which helps to line up the drawing position. The scratch marks can be highlighted by going over them with a black felt tip pen, and then rubbing off the excess.

An additional feature carried by the stylus assembly, is a push button switch (ITT ISOSTAT D6 series) which uses the command ADVAL (0) AND 3 to detect when it has been pushed. This button can be programmed to perform a number of tasks, like MOVE or DRAW or select procedures from a menu, etc. Equally, parallel functions can be selected from the keyboard if desired.

Wiring the potentiometers is a relatively simple matter as can be seen from Fig. 9.

To reduce the effects of extraneous pickup, one should use screened cable, with the cable screen taken to the computer ground point (pin 8 on the paddle user port). I used miniature four core cable with seven 0.1mm copper conductors with a braided screen. Strictly, separate pairs of wire should be used to connect the potentiometer to the user port.

In order to keep the number of cores to a minimum, I used common lines where possible. The only wire that needs to negotiate elbow A is that from the switch (if this is used). A pair of thin wires such as a strip off some ribbon cable will suffice, providing that a sufficiently large loop is left at the elbow to allow freedom of movement.

To further reduce noise effects, 0.1F capacitors (ceramic or monolithic) are connected between the ground line and the wipers (central connection) of the potentiometers.

The screened wire from the potentiometer at A is a little more complicated. This is taken down through a rubber grommet held in the bushing slot of the T1 assembly. This conveniently provides sufficent freedom for the wire and doesn't prevent the arms from moving.

Approximately 1.5 metres of screened cable is taken to the computer via another rubber grommet in the edge of the T1 assembly. Connection to the games paddle port is by means of a 15 pin D type connector with the pin connections as shown in Fig. 9.

The connections to the BBC Micro are such that T1 is connected to ADVAL (1) and T2 is connected to ADVAL (2). The function button is connected to ADVAL (0).

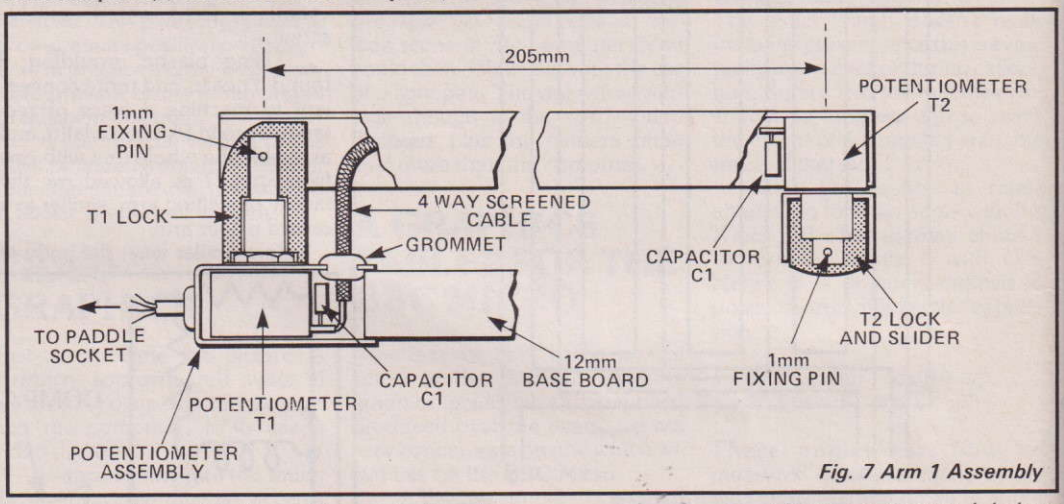


Fig. 7 Arm 1 Assembly

SETTING UP THE TABLET

It is more convenient to assemble the graphics board without paying too much attention to mechanical alignment. It is necessary to let the computer 'know' what the offset angles are, how big the plotting area is, and what the non-linearity is over this area. To do this requires some initial setting up procedure.

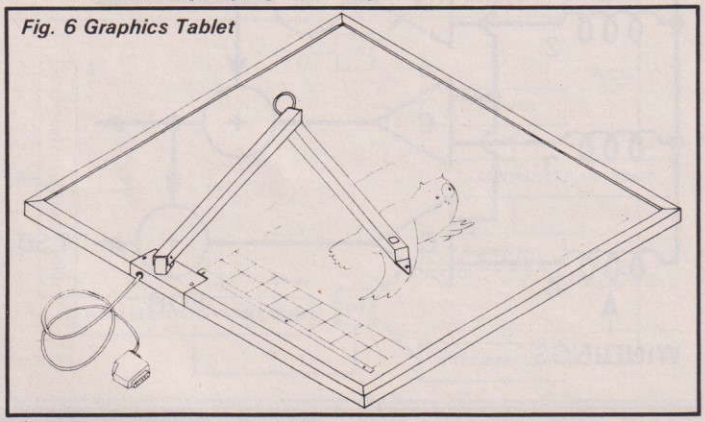
The angular offset can be measured in two ways. We can set up known angles for the arms, eg $\theta = 0°$ and $\psi = 90°$ are two easily locateable positions. Similarly, we can locate two or more known co-ordinate positions and deduce the angles from trigonometry.

Having tried both, I went for the latter, since additional information can be gathered about non-linearities in the x and y directions. Also, the same points can be fixed to describe the plotting area limits.

This fixing of points, however, means that the drawing reference points must always be in the same place. To help me achieve this, I attached a plastic clamp (an old ruler held down with chipboard screws) to the base board. This held in place a plastic overlay with the correct size grid (see page 495 of the BBC **User Guide**) draw on its surface.

Knowing the exact distances for the co-ordinates 0,0; 0,1024; and 1280, 1024 from the graphics arm origin 0, angular constants can be deduced which are sub-

Fig. 6 Graphics Tablet



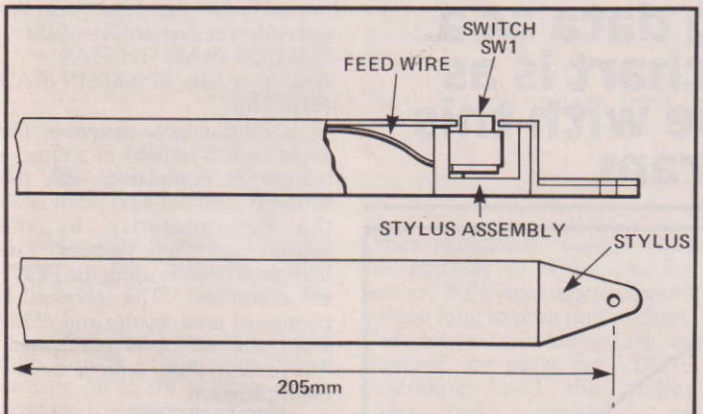


Fig. 8 Arm 2 Assembly

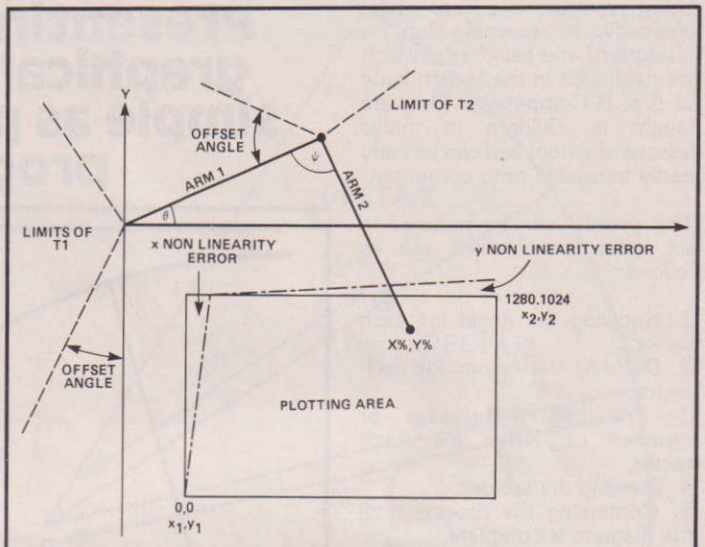


Fig. 10 Angular Co-ordinates

tracted from the measured angles as given by T1 and T2 (see Fig. 10).

This same information can be used to give a measure of secondary angular error related to both the x excursion, as the stylus is moved from 0,0 to 0,1024 and its combined effect with the secondary error as related to the y excursion as the stylus is moved from 0,1024 to 1280,1024.

This all results in a set of position related correction factors. Although it initally produces some complex mathematics, it generally boils down to a manageable solution of the form:

angle T1 = (ADVAL(1) + A%) *N/F.....(3)

angle T2(ADVAL(2) + C%) *N/G.....(4)

where A% and C% are the corrected offset angles and N is a constant involving PI and the potentiometer angular travel (4/3 *PI/65536). F and G are linearity factors.

To obtain the x and y offset co-ordinates, a further calibration procedure is used to determine the limits of the plot, ie x1, y1 at 0,0 and x2, y2 at 1280, 1024. Different positions can of course be used to enlarge or distort pictures, but it is best to stick to the grid size. The calibration procedure thus provides a scale factor S1% and S2% such that the final co-ordinate position is given by:

X% = S1% *(COS(T1) – COS(T3) – x1).....(5)

Y% = S2% *(SIN(T1) – SIN(T3) – y1)...(6)

where T3 = T1 + T2 and x1 and y1 are the origin offset values.

S1% = 1280/(x2 – x1).....(7)

S2% = 1024/(y2 – y1).....(8)

When setting up the reference points and the screen size, an average of 10 readings are taken to reduce the effects of jitter. While actually plotting points, however, the average of only four readings are taken.

This in effect performs a crude form of smoothing on the results and leads to smoother plotting. This inevitably slows down the plotting. An average of four is a reasonable compromise in BASIC coupled with the actual speed at which one can trace around a drawing. A machine code program would obviously permit a much faster response.

By utilizing sub-areas off the main plotting area a range of values can be set up. It may be used in conjunction with either the press button or say, one of the red function keys to select procedures from a menu.

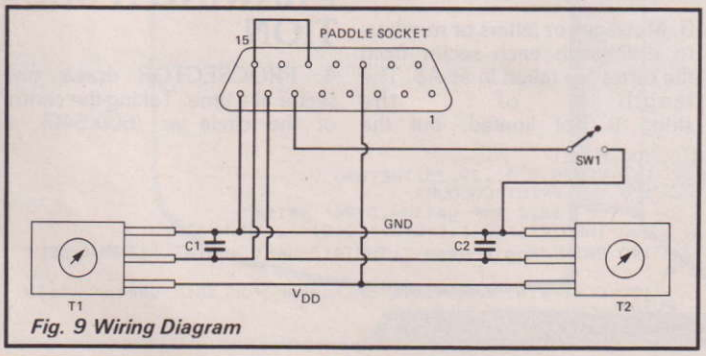


Fig. 9 Wiring Diagram

I have used such an area for selecting one of 16 foreground and background colours. Therefore, I arranged a colour pallette down one side of the plotting area. An example of this can be seen in the accompanying photograph. The fill routine in Issue 1.2 of the operating system is used to produce a colourful parrot.

GRAPHICS TABLET PERFORMANCE

The performance of the graphics tablet has been surprisingly good and although it is not absolutely linear across the whole tablet, it achieves something better than 5% accuracy at the centre of the plot. This means that if you try following a straight line up the screen in the centre, the actual plotted line on the VDU is likely to deviate by +/– 50 pixels in 1280. However this is of no real consequence if you are trying to copy irregular pictures like the parrot and is hardly noticeable.

If, however, you require perfectly straight lines, then use a flashing cross hair cursor to select the end points. A line (or circle, etc) drawing routine implemented from a menu, is an obvious answer.

A whole host of routines could be built up which can then be combined to produce technical or artistic drawings for instance. The list is almost endless. Having drawn a picture, it is a short step to drawing slightly different pictures. The concept of animated movement is shown by the flying owl presented on the BBC's Managing The Micro program.

I hope that this article has given some indication of the fundamentals behind graphics plotters and some insight on how to build your own plotter. It is hoped, however, that the tablet illustrated here will be shortly available. Also, that sufficient software would allow useful plotting with the BBC 'B' computer.

After all this though, it is not very far removed from the simple pantograph that I used many years ago, only the drawing paper has been replaced by a VDU.

Piecharts

Presenting data as a graphical chart is as simple as pie with this program.

Using piecharts is a very useful alternative to recording data on histograms and bar charts (which we dealt with in the launch issue of A & B Computing). They are taught to children in maths lessons at school and can be fairly easily translated onto computer.

This process can be reduced to six main steps which are as follows:

1. Knowing the angle for each sector,
2. Deciding the colour for each sector
3. Providing a message or number or letter for each sector,
4. Drawing the sector,
5. Continuing the process until the diagram is complete,
6. Adding a title.

Steps 1,2 and 3 could be fed in from the keyboard. Since this will be a process repeated several times as the program is run, it is easier to set up a text 'window' which can then be cleared without affecting the rest of the screen.

Working in Mode 1, lines 24-31 in the lower part of the screen, using the full width of the screen, have been set aside as a text window, using the command:

VDU28,0,31,39,25

where 0 is the extreme left,
31 is the bottom line,
39 is the extreme right,
and 25 is the top line of the text window. CLS will now clear this section of the screen only.

CONSIDERING THE STEPS TO BE TAKEN

1. On most of the occasions when piecharts are used, the angles are considered in degrees. It makes sense then to feed the angles in in degrees, using the RAD function to convert each angle to radians when required.

THETA = total angle up to the completion of any section.

DELTA = the angle of the sector being drawn.

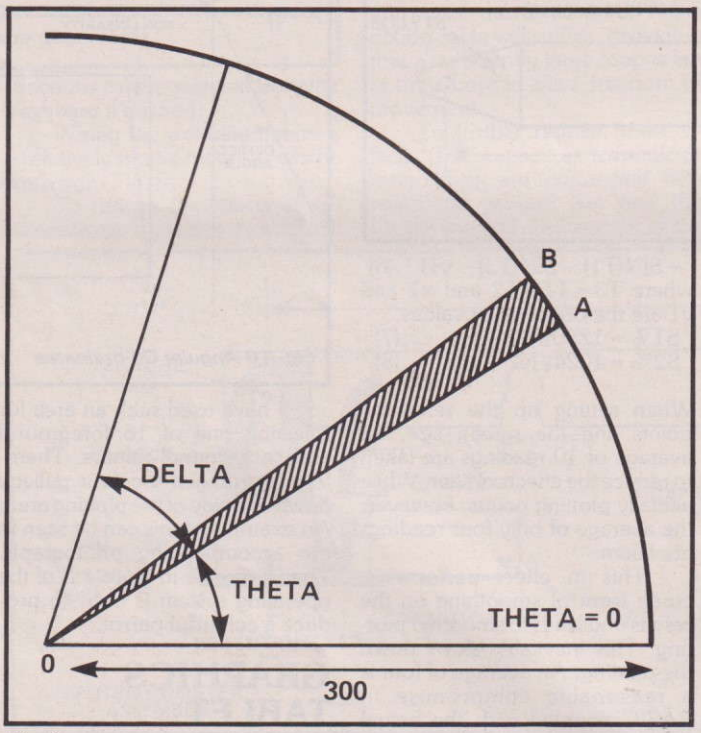


Fig. 1

2. In Mode 1, the logical colours are black, red, yellow and white in that order. Black is left as the background colour, and the colour of each sector is chosen from the keyboard by typing in 1,2 or 3, chosing the logical colour of that number.

3. Messages or letters or numbers to distinguish each sector from the othes are taken in as A$. The length of the string is not limited, but the smaller the angle, the shorter the message should be. Three or four characters should really be regarded as the maximum length if the appearance of the pie-chart is not to be spoiled. (See Listing 1)

DRAWING A SECTION

4. PROCSECTOR draws one sector at a time. Taking the centre of the circle as (600,540), it moves first to the point A on the circumference where X = 600 + 300 COS (RAD (THETA))
AND Y = 540 + 300 SIN (RAD (THETA))
In a loop which increases the angle by 0.5 radians at a time, a triangle is completed with the centre 0, and the next point B on the circumference as the second and third vertices. The triangle is filled in using the PLOT 85 command. The process is continued until, by the end of the loop, the sector is completed. The action then returns to the main program.

PROCMESSAGE is the procedure which places the string A$ on the sector. The two cursors are joined together using the command VDU5, so that the text may be written at the graphics cursor.

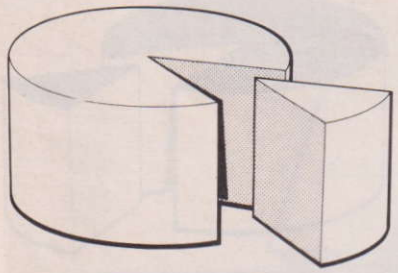
```
160 PROCSECTOR:PROCMESSAGE
170 THETA = THETA + DELTA:
    IF THETA = 360 OR THETA
    > 360\THEN 180 ELSE 120

1000 DEFPROCSECTOR
1010 MOVE600+ 300*COS(RAD
     (THETA)),540+300*SIN(RAD
     (THETA) )
1020 FORT=.5 TO DELTA STEP .5
1030   GCOL0,C
1040   MOVE600,540
1050   PLOT85,600+300*COS(RAD
       (THETA+T)),540+300*SIN
       (RAD( THETA+T))
1060   NEXT T
1070 ENDPROC
```

The message is written at a point inside the sector and positioned as shown in Fig.2. It is difficult to fix an initial position which will suit all sizes of sector, and length of message, but moving slightly to the left of the marked position (subtracting 50 from the y coordinate) gives a reasonable result in most cases. The cursors are then separated again using the command VDU, before returning to the main program.

```
1200 DEFPROCMESSAGE
1210 VDU5:GCOL0,0
1220 MOVE600+ 200*COS(RAD
     (THETA+DELTA/2))-50,540+
     200*SIN(RAD(THETA
     +DELTA/2))
1230 PRINTA$
1240 VDU4
1250 ENDPROC
```

5. The process of completing each sector in turn continues until the value of THETA is 360 degrees or more. (Listing 5)

```
100 MODE1
110 VDU28,0,31,39,25:THETA=0
120 CLS:PRINT"COLOUR?
 1 for red,2 for yellow,3 for white"
130 INPUTC:IF(C-1)*(C-2)*(C-3)<>0 THEN 120
140 PRINT"Angle used ";THETA;" next angle? ":INPUT DELT
A
150 CLS:PRINT"Any title or number for this sector? ":IN
PUTA$
```



```
180 PROCTITLE
200 CLS:PRINT"The pie-chart is complete.                P
ress the space bar to continue"
210 PROCWAIT
220 VDU26:CLS:PRINTTAB(1,1)"Type 1 to try another chart
                2 to end"
230 INPUTN:IF N=1 THEN 120
240 IF N=2 THEN 900
250 GOTO 230
900 END
```

6. Using the VDU5 command again, a title may be added (PROCTITLE) to complete the chart.

PROCWAIT is used to allow time to look at the chart before moving on to try another chart, or to end. (Listing 6)

```
1100 DEFPROCWAIT
1110 X$=GET$:IF X$="" THEN 1110
1120 IF X$=" " THEN 1130 ELSE
     1110
1130 ENDPROC
1300 DEFPROCTITLE
1320 CLS:PRINT"What is the
     title for the chart?":
     INPUT N $
1330 VDU5
1340 GCOL0,3:MOVE0,900:PRINT N$
1350 VDU4
1360 ENDPROC
```

EXTENDING THE CHART

Once the program is running satisfactorily, it can be extended as required. One extension is to feed in the various values used to compose the piechart, and let the computer work out the angles.

In this case the computer must store the values at the beginning of the program. When they are typed in from the keyboard, they are stored in the array X(10), and when all have been stored and added together, a second array stores the corresponding angles in degrees, to be consistent with the rest of the program.

Up to ten values are accepted. If a greater number is required, the array sizes must be increased.

Lines 20-50 allow the choice of putting in angles or values, and send the program to the appropriate line for the next step.

Lines 60-90 take in the values, and add and store them. Only one other alteration is needed, to skip line 150 if the angles are already known. (See program 'Extending the Chart).

Other alterations could include the addition of a key to the sectors, if the messages required are too long to print on the chart. The colour scheme could be changed, by using the VDU19 command, and the angles calculated could be included on the diagram. For the latter, they could be introduced as A$ on each sector, or printed out at the side of the piechart using the VDU5 command again.

SPEEDING UP THE PROGRAM

The use of integer variables when possible speeds up the running of the program. In this case, the time saved is not great but it could be a useful exercise to replace those values which *can* be replaced by integer variables.

BBC BASIC recognises an integer value if the variable name ends with %. The variables which could be changed are

THETA to THETA%
DELTA to DELTA%
C to C%

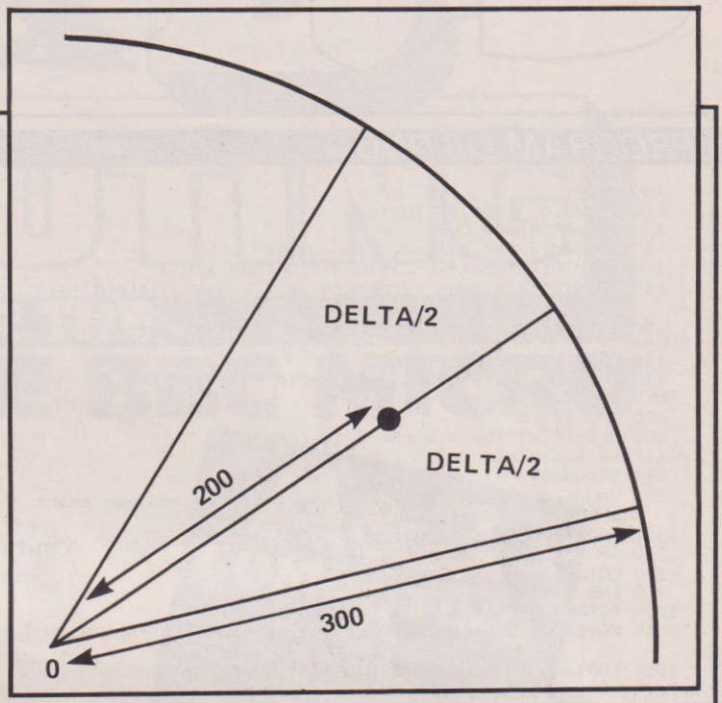
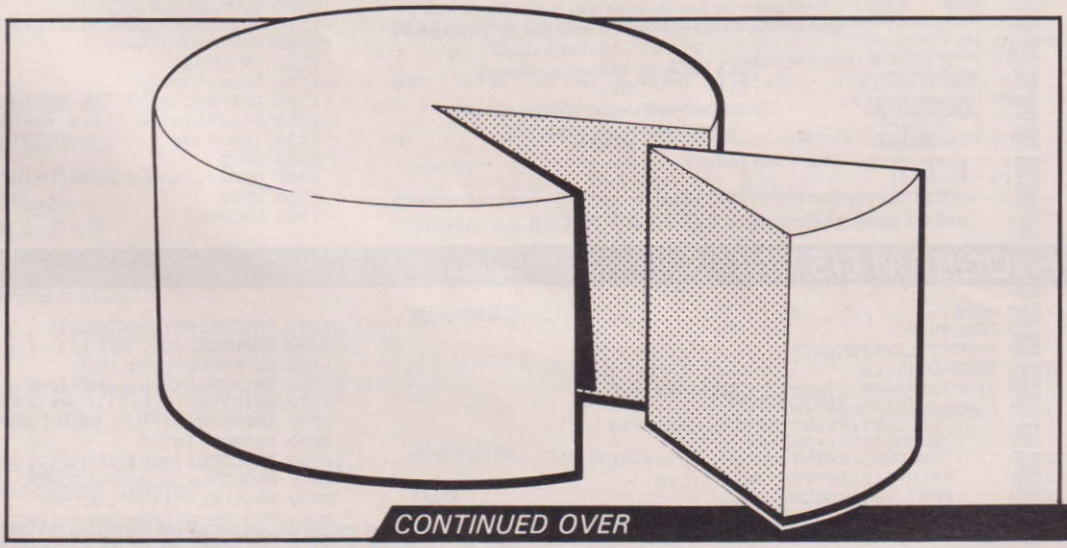


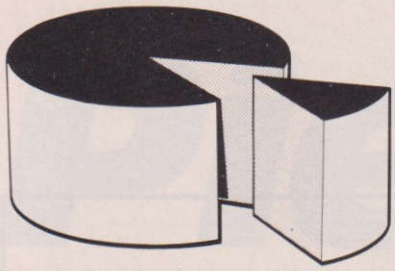
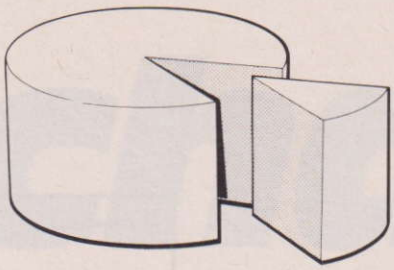
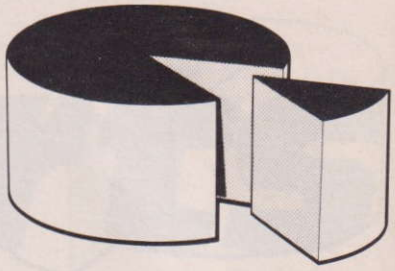
Fig. 2

EXTENDING THE CHART

```
10 DIM X(10),ANGLE(10)
20 CLS:PRINTTAB(1,1);"Do you wish"
30 PRINTTAB(1,5)"1. The computer to work out the angle
s  2. To provide the angles yourself?":INPUTV
40 IF(V-1)*(V-2)<>0 THEN 30
50  ON V GOTO 60,100
60 CLS:N=0:SUM=0:PRINT"Max. number of values 10. If yo
ur list  is complete,please press ="
70 PRINT"A Value please ":INPUTN$:IFN$="=" THEN 90
80 N=N+1:X(N)=VAL(N$):SUM=SUM+VAL(N$):GOTO 70
90 SCALE=360/SUM:FORI=1 TO N:ANGLE(I)=X(I)*SCALE:NEXTI

100 MODE1
110 VDU28,0,31,39,25:THETA=0:I=0
120 CLS:I=I+1:PRINT"COLOUR?
      1 for red,2 for yellow,3 for white"
130 INPUTC:IF(C-1)*(C-2)*(C-3)<>0 THEN 120
140 IFV=2THEN150 ELSE DELTA=ANGLE(I):GOTO 160
```



CONTINUED OVER

PROGRAM LISTING 1

```
  100 MODE1
  110 VDU28,0,31,39,25:THETA=0
  120 CLS:PRINT"COLOUR?
   1 for red,2 for yellow,3 for white"
  130 INPUTC:IF(C-1)*(C-2)*(C-3)<>0 THEN 120
  140 PRINT"Angle used ";THETA;" next angle? ":INPUT DELT
A
  150 CLS:PRINT"Any title or number for this sector? ":IN
PUTA$
  160 PROCSECTOR:PROCMESSAGE
  170 THETA = THETA + DELTA:IF THETA = 360 OR THETA > 360
 THEN 180 ELSE 120
  180 PROCTITLE
  200 CLS:PRINT"The pie-chart is complete.                P
ress the space bar to continue"
  210 PROCWAIT
  220 VDU26:CLS:PRINTTAB(1,1)"Type 1 to try another chart
                  2 to end"
  230 INPUTN:IF N=1 THEN 120
  240 IF N=2 THEN 900
  250 GOTO 230
  900 END
 1000 DEFPROCSECTOR
 1010 MOVE600+ 300*COS(RAD(THETA)),540+300*SIN(RAD(THETA)
)
 1020 FORT=.5 TO DELTA STEP .5
 1030   GCOL0,C
 1040   MOVE600,540
 1050   PLOT85,600+300*COS(RAD(THETA+T)),540+300*SIN(RAD(
THETA+T))
 1060   NEXT T
 1070 ENDPROC
 1100 DEFPROCWAIT
 1110 X$=GET$:IF X$="" THEN 1110
 1120 IF X$=" " THEN 1130 ELSE 1110
 1130 ENDPROC
 1200 DEFPROCMESSAGE
 1210 VDU5:GCOL0,0
 1220 MOVE600+ 200*COS(RAD(THETA+DELTA/2))-50,540+ 200*SI
N(RAD(THETA+DELTA/2))
 1230 PRINTA$
 1240 VDU4
 1250 ENDPROC
 1300 DEFPROCTITLE
 1320 CLS:PRINT"What is the title for the chart?":INPUT N
$
 1330 VDU5
 1340 GCOL0,3:MOVE0,900:PRINT N$
 1350 VDU4
 1360 ENDPROC
```

PROGRAM LISTING 2

```
   10 DIM X(10),ANGLE(10)
   20 CLS:PRINTTAB(1,1);"Do you wish"
   30 PRINTTAB(1,5)"1. The computer to work out the angle
s  2. To provide the angles yourself?":INPUTV
   40 IF(V-1)*(V-2)<>0 THEN 30
   50  ON V GOTO 60,100
   60 CLS:N=0:SUM=0:PRINT"Max. number of values 10. If yo
ur list  is complete,please press ="
   70 PRINT"A Value please ":INPUTN$:IFN$="=" THEN 90
   80 N=N+1:X(N)=VAL(N$):SUM=SUM+VAL(N$):GOTO 70
   90 SCALE=360/SUM:FORI=1 TO N:ANGLE(I)=X(I)*SCALE:NEXTI

  100 MODE1
  110 VDU28,0,31,39,25:THETA=0:I=0
  120 CLS:I=I+1:PRINT"COLOUR?
        1 for red,2 for yellow,3 for white"
  130 INPUTC:IF(C-1)*(C-2)*(C-3)<>0 THEN 120
  140 IFV=2THEN150 ELSE DELTA=ANGLE(I):GOTO 160
  150 PRINT"Angle used ";THETA;" next angle? ":INPUT DELT
A
  160 CLS:PRINT"Any title or number for this sector? ":IN
PUTA$
  170 PROCSECTOR:PROCMESSAGE
  180 THETA = THETA + DELTA:IF THETA = 360 OR THETA > 360
 THEN 190 ELSE 120
  190 PROCTITLE
  200 CLS:PRINT"The pie-chart is complete.                P
ress the space bar to continue"
  210 PROCWAIT
  220 VDU26:CLS:PRINTTAB(1,1)"Type 1 to try another chart
                  2 to end"
  230 INPUTN:IF N=1 THEN 15
  240 IF N=2 THEN 900
  250 GOTO 230
  900 END
 1000 DEFPROCSECTOR
 1010 MOVE600+ 300*COS(RAD(THETA)),540+300*SIN(RAD(THETA)
)
 1020 FORT=.5 TO DELTA STEP .5
 1030   GCOL0,C
 1040   MOVE600,540
 1050   PLOT85,600+300*COS(RAD(THETA+T)),540+300*SIN(RAD(
THETA+T))
 1060   NEXT T
 1070 ENDPROC
 1100 DEFPROCWAIT
 1110 X$=GET$:IF X$="" THEN 1110
 1120 IF X$=" " THEN 1130 ELSE 1110
 1130 ENDPROC
 1200 DEFPROCMESSAGE
 1205 VDU5:GCOL0,0
 1210 MOVE600+ 200*COS(RAD(THETA+DELTA/2))-50,540+ 200*SI
N(RAD(THETA+DELTA/2))
 1220 PRINTA$
 1230 VDU4:ENDPROC
 1300 DEFPROCTITLE
 1310 CLS:PRINT"A title for the chart,please"
 1320 INPUT N$
 1330 VDU5
 1340 GCOL0,3:MOVE0,900:PRINTN$
 1350 VDU4
 1360 ENDPROC
```

PROGRAM LISTING 3

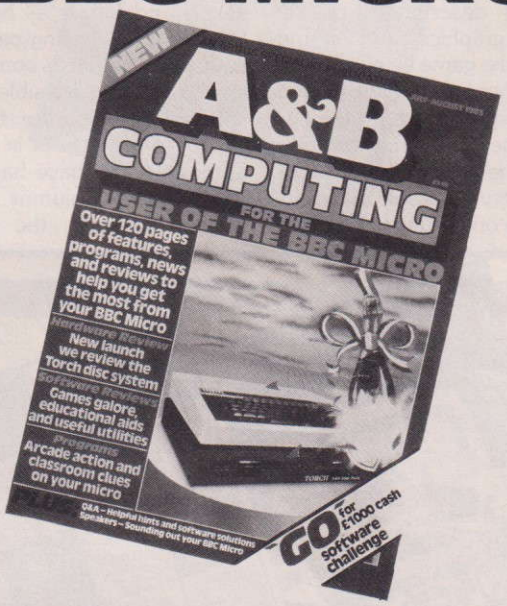
```
   10 MODE7
   30 PROCINTRO
  100 PROCTITLE:PROCWAIT
 1200 DEFPROCTITLE
 1210 CLS:T1$=CHR$(224)+" ":T2$=CHR$(225)+" "
 1220 FORI=0TO9:FORJ=1TO7
 1230       COLOUR1:PRINTTAB(I*2,J*4);T1$
 1240       PRINTTAB(I*2,J*4+1);T2$
 1250       COLOUR2:PRINTTAB(38-I*2,J*4);T1$
 1260      PRINTTAB(38-I*2,J*4+1);T2$
 1270      NEXTJ:PROCDELAY
 1275    SOUND2,-15,1,5
 1280    NEXTI
 1290 PROCDELAY:PROCDELAY
 1350 ENDPROC
 1700 DEFPROCDBL(W$,U,W)
 1710 PRINTTAB(U,W);CHR$(141);W$
 1720 PRINTTAB(U,W+1);CHR$(141);W$
 1730 ENDPROC
 2000 DEFPROCINTRO
 2010 PROCDBL("ADDITION",12,10)
 2020 PROCDBL("G.W.GALLAGHER",20,16)
 2030 PROCDBL("1983",25,20)
 2040 PROCDELAY:PROCDELAY:PROCDELAY
 2050 ENDPROC
```

A&B COMPUTING FOR THE USER OF THE BBC MICRO

A&B Computing is a new bi-monthly publication dedicated to users of the BBC Micro.

Priced at £1.85, forthcoming issues of A&B Computing will contain features on various aspects of the system, courses on improving your techniques in graphics, sound and assembly language, reviews of software and new add-ons, as well as plenty of exciting games and utility programs for you to run. Indeed, everything you'll need to make the most of your BBC Micro.

So, no matter whether you're a raw beginner or an accomplished programmer, we're sure you'll find A&B Computing the magazine you simply can't do without. And, just to prove it, we're offering you the chance to take out a subscription at the same rates that you would pay in the shops and not a penny more.



DON'T MAKE YOUR BBC MICRO MISS OUT, SUBSCRIBE TO A&B COMPUTING TODAY!

Subscription Order Form

Cut out and SEND TO:

A&B COMPUTING
513 London Road
Thornton Heath
Surrey CR4 6AR

Please commence my subscription with the very next issue.

SUBSCRIPTION RATES (tick □ as appropriate)

£11.10 for six issues UK □
£13.40 for six issues overseas surface mail □
£21.00 for six issues overseas airmail □

I am enclosing my (delete as necessary) cheque/Postal Order/International Money Order for £.....
(made payable to ASP Ltd)
OR
Debit my Access/Barclaycard (*delete as necessary)*

BARCLAYCARD VISA

Please use BLOCK CAPITALS and include post codes.

NAME (Mr/Mrs/Miss) .....

ADDRESS .....

.....

..... POSTCODE .....

Signature .....

Date .....

Software Listings

Want a program for your BBC Micro. Look no further than our listings to make your choice.

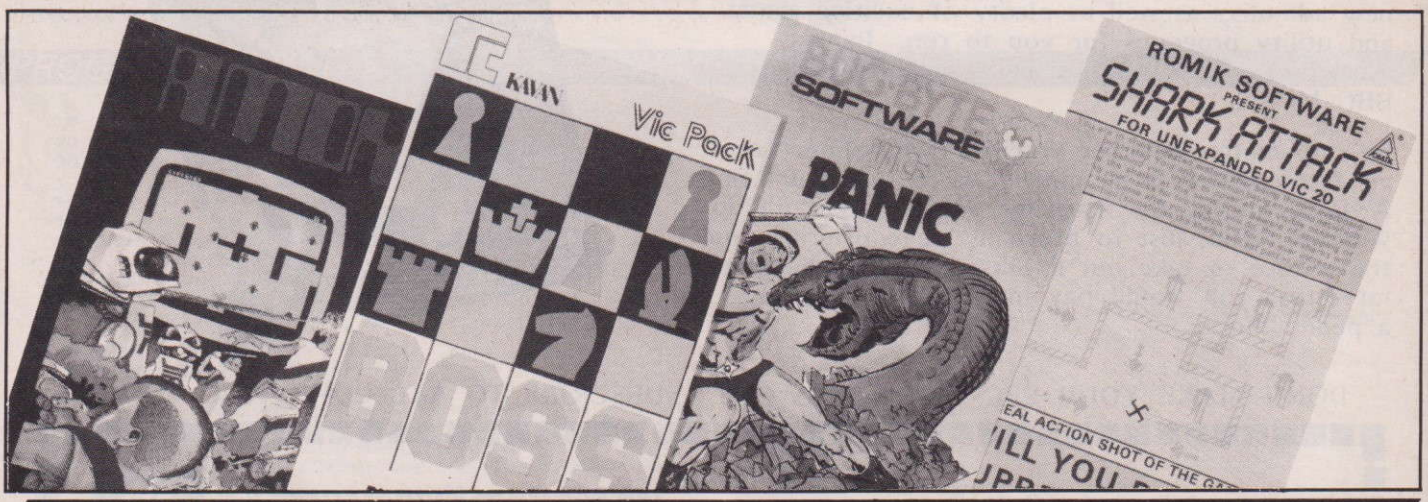
Finding and choosing the correct software for your needs is a daunting task indeed, whether you are looking for a word processing package or for a new game to test your alien-destroying, treasure-seeking, path-finding talents!

Often you can be put off even looking through the pages of advertisements which tempt you with vivid descriptions of the amazing graphics and sound effects of the game being offered, you sit there with pen poised above cheque book and your eye catches the small print that tells you that the game will run on just about every available machine except the one sitting beside you!

But BBC Micro owners despair no longer — help is at hand in the following pages. We have put together as comprehensive a list as possible of the software available for the BBC Micro. In order to fit in as many as possible we have had to use codes in some columns. The title of the software, the memory required to run it, the company that produces it, whether it is tape/cartridge or disc, the supplier and the price, are given for each piece of software listed. The codes used are:

Code	Explanation
Gm	Game
Bs	Business Routine
Ut	Utility (ie programming aid)
Do	Domestic
Ed	Educational
C	Cassette

As you are probably aware new software is surfacing all the time so don't assume that there is no such item as the one you are looking for if it is not included in this list. Also, remember that games in particular that at present run only on a machine other than the BBC Micro may well soon appear in a BBC version. If you are aware of a piece of software that is not listed here, whether you are a user or a producer, feel free to let us know.



SOFTWARE LISTING

Title	Type	Manufacturers	Memory	Software Supplier	Price
Action of the Heart	Ed	Garland Comp.	32K	C JX	£11.76
Adventure	Gm	Micro Power	16K	C GK,NR	£6.95
Adventure	Gm	Program Direct	32K	C NP	£5.99
Adventure	Gm	Odyssey Software	32K	C OG	£4.50
Adventure Quest	Gm	Level 9 Computing	32K	C CU	£9.90
Algebraic Manipulation	Ut	Acornsoft	16K	C AL,GA	£9.95
Algebraic Manipulation	Ut	Acornsoft	32K	C AL,GA	£9.95
Alien Destroyers	Gm	Micro Power	32K	C GK	£6.95
Alien Dropout	Gm	Superior Software	32K	C KH	£6.50
Arcade Action	Gm	Acornsoft	16K	C GA	£11.90
Arcade Pack 1	Gm	Odyssey Software	32K	C OG	£3.60
Arrow of Death (1)	Gm	Digital Fantasia	16K	C NT	£8.95
Arrow of Death (2)	Gm	Digital Fantasia	16K	C NT	£8.95
Angle(4)	Ed	Chalksoft	32K	C KT	£6.95
Append It	Ut	Aztec S/W	16K	C IB	£3.00
Asteroids/Frong	Gm	Aardvark Software	16K	C IU	£4.00
Asteroid Belt	Gm	Electronics Appplied	32K	C IF	£11.50
Asteroid Belt	Gm	Computer Concepts	16K	C GJ	£7.80
Astro Navigator	Gm	Micro Power	32K	C GK	£4.95
Atlantis	Gm	IJK Software	32K	C IT	£6.95
Awari	Gm	Foilkade	16K	C NR	£5.95
Bach (Music)	Gm	Swift Link Software	32K	C OF	£5.00
Backgammon	Ut	Bug Byte	32K	C KP	£8.00
Balloons	Gm	C J E Microcomputers	32K	C NV	£6.00
Basic Maths	Ed	Aztec S/W	16K	C IB	£3.00
Battlefield	Gm	Micro-Aid	32K	C IZ	£2.50
Banner	Do	Micro-Aid	16K	C IZ	£2.95
Beeb-Chase	Gm	Database Software	32K	C NU	£7.50
Beebchat	Gm	Odyssey Software	32K	C OG	£2.25
Beebmunch	Gm	IJK Software	32K	C IT	£5.95
Beebtrek	Gm	Software for All	16K	C KN	£7.95
BEEP-BEEB	Gm	IJK Software	32K	C IT	£3.95
Beowulf Adventure	Gm	Swift Link Software	32K	C OF	£5.00
Bounty Pirates	Gm	Aztec S/W	16K	C IB	£5.50
Breakout	Gm	I.J.K. S/W	16K	C IT	£3.95
Breakout	Gm	Bryants S/W	32K	C HW	£3.75
Carbohydrate Metabolism	Ed	Garland Comp.	32K	C JX	£18.24
Cashbook A	Do	Micro-Aid	16K	C IZ	£3.95

Cashbook B	Do	Micro-aid	16K	C IZ	£3.95
Cat & Mouse	Gm	Micro Power	16K	C GK	£4.95
Cells and Serpents/ Stockmarket	Gm	ASP Software	16K	C OD	£11.45
Cells & Serpent	Gm	Hexagon S/W	16K	C JA	£5.00
Centipede	Gm	Superior S/W	32K	C KH	£7.00
CESIL	Ed	Eduquest	16K	C NW	£19.95
Character Builder	Ut	Davansoft	16K	C NX	£4.95
Characters	Ut	Computer Concepts	32K	C GJ	£6.67
Character Generator	Ut	MP S/W	32K	C JZ	£3.00
Character Generator	Ut	Software for All	32K	C KN	£4.95
Characters	Ut	Computer Concepts	16K	C GJ	£6.67
Characters & Envelope Definer	Ut	Electronics Applied	32K	C IF	£5.50
Chargen	Ut	Odyssey Software	16K	C OG	£2.50
Chess	Gm	Micro Power	32K	C GK	£6.95
Circus	Gm	Digital Fantasia	32K	C NT	£8.95
Claws	Ed	Bryants S/W	16K	C HW	£3.75
Cards	Gm	Micro-Aid	16K	C IZ	£2.95
Cobra/Robo-Swamp	Gm	Software for All	16K	C KN	£6.95
Cobra/Robo-Swamp	Gm	Software for All	32K	C KN	£6.95
Code Race	Gm	Computer Concepts	16K	C GJ	£6.67
Code Race	Ut	Computer Concepts	32K	C GJ	£6.67
Colossal Adventure	Gm	Level 9 Computing	32K	C CU	£9.90
Connect 4	Gm	Database Software	32K	C NU	£5.90
Constellation	Ed	Micro Power	32K	C GK	£5.95
Cookbook Wizardry	Do	Database Software	32K	C NU	£7.50
Cowboy Shoot-out	Gm	Micro Power	32K	C GK	£5.95
Creative Graphic Pack	Ut	Acornsoft	16K	C AL,GA	£9.95
Creative Graphics	Ut	Acornsoft	32K	C AL,GA	£9.95
Crossed Words	Ed	Aztec S/W	16K	C IB	£6.50
Data-Quiz	Ut	Bryants S/W	32K	C HW	£4.88
Database	Bs	Computercat	16K	C IJ	£11.95
Database	Bs	Software for All	16K	C KN	£9.95
Defchr	Ut	Micro-Aid	16K	C IZ	£2.95
Defender	Gm	Acornsoft	32K	C AL,GA	£9.95
Desk Diary	Bs	Acornsoft	32K	C AL,GA	£9.95
Desk Diary	Bs	Acornsoft	16K	C AL	£9.95
Devil's Causeway	Gm	Anirog Computers	16K	C OA	£6.00
Dissassembler	Ut	Micro Power	16K	C GK	£5.95
Dissassembler	Ut	Program Direct	16K	C NP	£3.00
Dissassembler	Ut	Davansoft	16K	C NX	£5.95
Dissassembler	Ut	C J E Microcomputers	16K	C NV	£5.00
Distances	Ed	Micro-Aid	32K	C IZ	£2.95
Digital X-Word Compiler	Gm	N. Darwood	16K	C JB	£6.00
DNA Replication	Ed	Garland Comp.	32K	C JX	£17.65
Dragon Rider	Gm	Salamander Software	32K	C NZ	£6.95
Drawing	Ut	B.B.C.	16K	C KB	£10.00
Driving Test	Ed				
	Gm	Swift Link Software	32K	C OF	£5.00
Dungeon Adventure	Gm	Level 9 Computing	32K	C CU	£9.90
Early Learning	Ed	BBC	16K	C KB	£10.00
Early Numbers	Ed	Bryants S/W	32K	C HW	£4.80
Early Warning	Gm	A&F Software	16K	C GE	£6.00
Educational (1)	Ed	Golem	16K	C OB	£8.05
Educational (2)	Ed	Golem	16K	C OB	£8.05
7 Educational Games	Gm	Micromail	32K	C OE	£5.75
Eldorado Gold	Gm	Program Power	32K	C GK	£7.99
Electric	Ed	Database Software	16K	C NU	£5.50
England	Ed				
	Gm	Swift Link Software	32K	C OF	£5.00
European Studies	Ed	Aztec S/W	32K	C IB	£6.50
Escape from Pulsar 7	Gm	Digital Fantasia	32K	C NT	£8.95
Fairytale	Gm	Molimerx	32K	C AJ	£10.06
Feasibility Experiment	Gm	Ditigal Fantasia	32K	C NT	£8.95
Filer	Bs	Micro Power	16K	C GK	£8.95
Firien Wood	Gm	MP S/W	32K	C JZ	£6.50
Flags	Ed	Micro-Aid	16K	C IZ	£2.95
Flush	Ut	Micro-Aid	16K	C IZ	£1.00
Football Pools Predictor	Do	Mayday Software	16K	C IX	£4.99
Footer	Gm	Micro Power	32K	C GK	£6.95
Frenzy	Gm	Persoft	16K	C IY	£5.75
Frogger (Machine Code)	Gm	A&F Software	32K	C GE	£8.00
Fruit Machine	Gm	Superior Software	32K	C KH	£6.50
Fruity	Gm	Odssey Software	32K	C OG	£2.75

Supplier Code

AJ Molimerx Ltd
1 Buckhurst Road
Town Hall Square
Bexhill-on-sea
East Sussex

AL Acornsoft Ltd
4a Market Hill
Cambridge CB2 3NJ

FY Wida Software
2 Nicholas Gardens
London W5 5HY

GA Eltec Services Limited
217 Manningham Lane
Bradford BD8 7HH

GE A&F Software
83 Hyde Road
Gorton
Manchester M18 7JD

GJ Computer Concepts
16 Wayside
Chipperfield
Hertfordshire WD4 9JJ

GK Micro Power Ltd
8/8a Regent Street
Chapel Allerton
Leeds LS7 4PE

HW Bryants (Educational) Software
1 The Hollies
Chalcraft Lane
North Bersted
Bognor Regis PO21 5SX

IF Electronics Applied
4 Dromore Road
Carrickfergus
County Antrim BT38 7PJ

IT IJK Software
55 Fitzroy Road
Bispham
Blackpool

IU Aardvark Software
15 Queensberry Avenue
Hartlepool
Cleveland TS26 9NW

IV James Hager
7 Basset Street
Camborne
Cornwall TR14 8SW

IW Simon Hessel Software
15 Lytham Court
Cardwell Crescent
Bershire

IX Mayday Software
181 Portland Crescent
Stanmore
Middlesex HA7 1LR

IY Persoft
Freepost
Baildon
Shipley
West Yorkshire BD17 5SX

IZ Micro-Aid
25 Fore Street
Praze-an-Beeble
Camboren
Cornwall TR14 0JX

JA Hexagon Software
17 Cambridge Grove
Otley

JB N Darwood Ltd
Halfacres
Stroud
Petersfield
Hampshire GU32 3PJ

JC Futura Software
63 Lady Lane
Chelmsford
Essex CM2 OTQ

JX Garland Computing
35 Dean Hill
Plymouth PL9 9AF

JY Corona Software
Corona House
21 Tennyson Avenue
London E11 2QN

JZ MP Software & Services
165 Spital Road
Bromborough
Merseyside L62 2AE

CONTINUED OVER

NP Program Direct
37B New Cavendish Street
London W1M 8JR

NR Foilkade
Dept PR14
66 Littledean
Yate
Bristol BS17 4UQ

NT Digital Fantasia
24 Norbreak Road
Norbreck
Blackpool

NU Database Software
97 Defoe Drive
Park Hill
Stoke-on-Trent

NV CJE Microcomputers
25 Henry Avenue
Rustington
West Sussex BN16 2PA

NW Eduquest
Thames Avenue
Windsor
Berkshire SL4 1QP

NX Davansoft
1 Delapoer Drive
Haverfordwest
Dyfed SA61 1HX

NZ Salamander Software
27 Ditchling Rise
Brighton
East Sussex BN1 4QL

KB BBC Publications
British Broadcasting Corporation
35 Marylebone High Street
London W1M 4AA

KH Superior Software
69 Leeds Road
Bramhope
Leeds

KN Software for All
72 North Street
Romford
Essex

KT Chalksoft
Lowmoor Cottage
Tonedale
Wellington
Somerset TA2 10AL

OA Anirog Computers
26 Balcombe Gardens
Horley
Surrey

OB Golem Ltd
77 Qualitas
Bracknell
Berkshire RG12 4QG

OC A Lane (Software)
12/316 Seaside
Eastbourne
East Sussex BN22 7RH

OD ASP Software
145 Charing Cross Road
London WC2H 0EE

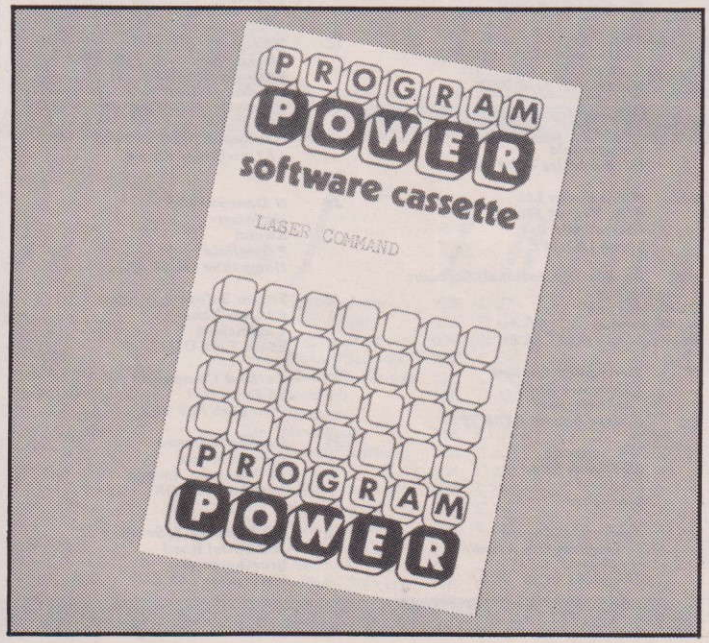
OE Micromail
PO Box 34
Leighton Buzzard LU7 8SJ

OF Swift Link Software
118-120 Wardour Street
London W1V 4BT

OG Odyssey Software
8 Greenbrook Avenue
Hadley Wood
Barnet
Herts EN4 0LS

OH Edu-CAL
28 Ingersoll Road
Shepherds Bush
London W12 7BD
01 743 1579

CU Level 9 Computing
229 Hughenden Road
High Wycombe
Buckinghamshire HP13 5PG



SOFTWARE LISTING

Title	Type	Manufacturers	Memory	Software Supplier	Price
FORTH	Ut	Acornsoft	32K	C AL	£16.85
Fun Games	Gm	BBC	16K	C KB	£10.00
Galactic Commander	Gm	Micro Power	32K	C GK	£6.95
Galaxians	Gm	Superior S/W	32K	C KH	£7.00
Game of Logic	Ed	N Darwood	16K	C JB	£8.00
Games of Logic and cunning	Gm	Golem	16K	C OB	£9.20
Games of Strategy	Gm	BBC	16K	C KB	£10.00
Games Pack II	Gm	Micromail	32K	C OE	£6.75
Genetic Code	Ed	Garland Comp.	32K	C JX	£17.65
Geography Italy	Ed	Corona S/W	32K	C JY	£5.00
Ghost/Diamonds	Gm	A Lane	16K	C OC	£3.00
Ghost Maze	Gm	Software for All	32K	C KN	£6.95
Grand Prix	Gm	Software for All	32K	C KN	£5.95
Golf	Gm	Bryants S/W	32K	C HW	£4.88
Golf	Gm	Bug Byte	32K	C GA	£7.00
Gomoku	Gm	Micro Power	16K	C GK	£3.95
Graphics Package	Ut	Salamander Software	32K	C NZ	£24.95
Graph and Charts Pack	Bs	Acornsoft	32K	C AL,GA	£9.95
Graph and Charts Pack	Bs	Acornsoft	16K	C AL,GA	£9.95
Great Britain Ltd	Gm	SW Hessel S/W	32K	C IW	£5.95
Hangman	Gm	Aztec S/W	16K	C IB	£5.50
Hangman	Gm	MP S/W	32K	C JZ	£4.00
Helicopter Rescue/Tunnel/ Roadrace	Gm	A Lane	16K	C OC	£4.00
Home Accounts	Do	Persoft	32K	C IY	£12.50
Home Finance	Do	BBC	16K	C KB	£10.00
Hydraulics	Ed	Database Software	16K	C NU	£5.50
Inkosi	Gm	Chalksoft	32K	C KT	£5.95
Invaders	Gm	Superior S/W	32K	C KH	£7.00
Invaders	Gm	Software for All	16K	C KN	£6.95
Invaders	Gm	Hexagon S/W	16K	C JA	£6.00
Invaders	Gm	MP S/W	32K	C JZ	£6.50
Invisible Man	Ed	Chalksoft	32K	C KT	£5.95
Inheritance	Gm	SW Hessel S/W	32K	C IW	£5.95
Inheritance	Ed	Garland Comp.	32K	C JX	£34.70
J.R.	Gm	Software for All	32K	C KN	£6.95
Jumbles	Ed	Bryants S/W	32K	C HW	£4.88
Jumbo	Gm	Molimerx	32K	C AJ	£17.25
Junior Maths Pack	Ed	Micro Power	32K	C GK	£5.95
Katakombs	Gm	Golem	32K	C OB	£9.20
La Princesse (French)	Ed	Aztec S/W	32K	C IB	£6.50
Landfall & Serpent	Gm	GT Software	32K	C JW	£6.50
Laser Command	Gm	Micro Power	32K	C GK	£6.95
Letters	Ed	Chalksoft	32K	C KT	£6.95
Library Dewey Classification	Ed	Aztec S/W	32K	C IB	£6.50
Link-4-Plus	Gm	ABC Software	16K	C KR	£6.95
Lisp	Ut	Acornsoft	32K	C AL,GA	£16.85
Lisp	Ut	Acornsoft	16K	C AL,GA	£16.85
Logo 2	Ut	Computer Concepts	32K	C GJ	£10.00
Lunar Lander	Gm	A&F Software	32K	C GE	£6.90
Mailing A	Bs	Micro-aid	16K	C IZ	£3.95
Mailing B	Bs	Micro-Aid	16K	C IZ	£3.95
Martians	Gm	Micro Power	32K	C GK	£5.95
Mastermind	Gm	Micro Power	16K	C GK	£3.95
Maze Invaders	Gm	Micro Power	32K	C GK	£4.95
Maze Man	Gm	C J E Microcomputers	32K	C NV	£6.00
Micro Budget	Do	Micro Power	16K	C GK	£6.95
Metrics (5)	Ed	Chalksoft	32K	C KT	£9.95
Micro Maths	Ed	LCL	16K	C KA	£24.50
Minefield	Gm	Eduquest	16K	C NW	£5.95
Master Copier	Ut	Aztec S/W	16K	C IB	£6.50
Maths Translation	Ed	Corona S/W	32K	C JY	£5.00
Minefield	Gm	A&F Software	16K	C GE	£6.00

Missile Control	Gm	C J E Microcomputers	32K	C NV	£9.00
Mission Impossible	Gm	Aztec S/W	16K	C IB	£6.50
Mixed Games	Gm	IJK S/W	16K	C IT	£3.95
Model A Invaders	Gm	IJK S/W	16K	C IT	£4.95
Model B Invaders	Gm	IJK S/W	32K	C IT	£6.95
Money Box	Ed	Bryants S/W	32K	C HW	£4.88
Monsters	Gm	Acornsoft	32K	C AL,GA	£9.95
Monster Battles	Gm	Bryants S/W	32K	C HW	£4.80
Multiple Choice	Ed	Eduquest	32K	C NW	£25.00
Munchyman	Gm	Micro Power	16K	C GK	£5.95
Music	Do	B.B.C.	16K	C KB	£10.00
Music Maker	Gm	Rainbow S/W	16K	C KS	£3.50
Musical Numbers	Ed	Bryants S/W	32K	C HW	£4.88
Musical Number Box	Gm	Bryants S/W	16K	C HW	£3.75
Music Program	Do	Golem	32K	C OB	£6.90
Mutant Invaders	Gm	IJK S/W	16K	C IT	£5.95
MX 80 Type 3 Screen Dump	Ut	Software for All	16K	C KN	£6.95
Othello	Gm	Computer Concepts	16K	C GJ	£8.95
Othello	Gm	Computer Concepts	32K	C GJ	£8.95
Othello	Gm	Computercat	16K	C IJ	£8.95
Painting	Ut	BBC	16K	C KB	£10.00
Parity	Ed	N. Darwood	16K	C JB	£6.00
Payroll	Bs	Micro-Aid	32K	C IZ	£5.95
Payroll 2	Bs	Micro-Aid	32K	C IZ	£5.95
Peeko Computer	Bs	Acornsoft	16K	C AL,GA	£9.95
Peeko Computer	Ut	Acornsoft	32K	C AL,GA	£9.95
Pete the Plastered Postman/ Asteroid Lander	Gm	ASP Software	16K	C OD	£8.50
Pharoah's Tomb	Gm	A&F Software	32K	C GE	£8.00
Philosopher's Quest	Gm	Acornsoft	16K	C GA	£9.95
Picasso	Ut	Odyssey Software	32K	C OG	£3.00
Picsave	Ut	Hexagon S/W	16K	C JA	£6.00
Powerboat Race	Gm	Futura S/W	32K	C JC	£7.95
Programming Made Easy	ED	Edu-Cal	32K	C OH	£7.90
Princess	Gm	Aztec S/W	16K	C IB	£6.50
Procflush	Ut	Micro-Aid	16K	C IZ	£1.00
Procvar	Ut	Micro-Aid	16K	C IZ	£1.95
Punctuation	Ed	Bryants S/W	32K	C HW	£4.88
Reversi 1	Gm	Micro Power	16K	C GK	£4.95
Reversi 2	Gm	Micro Power	32K	C GK	£4.95
Road Runner	Gm	A&F Software	32K	C GE	£6.90
Roulette	Gm	Micro Power	32K	C GK	£4.95
Searchbas	Ut	Micro-Aid	16K	C IZ	£1.95
Seed Germination	Ed	Garland Comp.	32K	C JX	£18.82
Sequences	Ed	Chalksoft	32K	C KT	£5.95
Sheepdog Trials	Ed	Bryants S/W	32K	C HW	£4.80
Shootout	Gm	MP S/W	32K	C JZ	£5.00
Snake	Gm	Computer Concepts	32K	C GJ	£6.67
Snakes and Ladders	Gm	Swift Link Software	32K	C OF	£5.00
Snapper	Gm	Acornsoft	32K	C AL,GA	£9.95
Sort M/C	Ut	Micro-Aid	16K	C IZ	£1.00
Sortbas	Ut	Micro-Aid	16K	C IZ	£1.00
Spacehawks	Gm	Computer Concepts	32K	C GJ	£7.80
Spacemaze	Gm	Micro Power	32K	C GK	£5.95
Space Academy	Gm	Swift Link Software	32K	C OF	£5.00
Space Warp	Gm	Bug Byte	32K	C GA,EA	£11.50
Space Fighter	Gm	MP S/W	32K	C JZ	£8.50
Space Fighter	Gm	Superior S/W	32K	C KH	£7.00
Space Games Pack 1	Gm	Futura S/W	16K	C JC	£3.99
Space Games Pack 2	Gm	Futura S/W	16K	C JC	£3.99
Space Games Pack 3	Gm	Futura S/W	16K	C JC	£4.99
Space Games Pack 4	Gm	Futura S/W	16K	C JC	£4.99
Space Trek	Gm	Program Direct	32K	C NP	£5.99
Speechparts	Ed	Bryants S/W	32K	C HW	£4.88
Squash	Gm	Aztec S/W	16K	C IB	£5.50
Star Maze	Gm	Database Software	32K	C NU	£7.50
Star Trek	Gm	Micro Power	16K	C GK	£4.95
Startrek/Candy Floss	Gm	IJK S/W	16K	C IT	£5.95
Star Treck	Gm	Hexagon S/W	16K	C JA	£5.50
Statpack	Ed	Micro-Aid	32K	C IZ	£7.95
Storybuilder	Ed	Bryants S/W	32K	C HW	£4.88
Superlife	Gm	Golem	32K	C OB	£9.20
Super Hangman	Gm	IJK S/W	32K	C IT	£3.95
Swamp Monsters	Gm	M P Software	32K	C JZ	£6.50
Swift 1 (Money)	Ed	Swift Link Software	32K	C OF	£10.00
Swift 2 (Four Rules)	Ed	Swift Link Software	32K	C OF	£10.00
Swift 3 (Multiplication)	Ed	Swift Link Software	32K	C OF	£10.00
Swift 4 (Problems)	Ed	Swift Link Software	32K	C OF	£10.00
Swoop	Gm	Micro Power	32K	C GK	£6.95
Tables Test	Ed	Bryants S/W	32K	C HW	£4.88
Tape Copy	Ut	Davansoft	16K	C NX	£7.50
Teacher's Toolkit	Ed	Wida Software	32K	C FY	£30.00
Text Processing Pack	Bs	Eduquest	32K	C NW	£10.00
The Frog	Gm	James Hager	32K	C IV	£6.50
The Golden Baton	Gm	Digital Fantasia	16K	C NT	£8.95
3D Maze	Gm	IJK S/W	32K	C IT	£3.95
The Time Machine	Gm	Digital Fantashia	16K	C NT	£8.95
Timetrek	Gm	Micro Power	32K	C GK	£6.95
Touch Typist	Ed	Computercat	16K	C IJ	£9.95
Tower of Alos	Gm	A&F Software	16K	C GE	£6.90
Towns of Britain	Ed	Aztec S/W	16K	C IB	£6.50
The Computer Programme Programs Vol 2	Do	BBC	32K	C KB	£10.00
The Computer Programme Programs Vol 1	Do	BBC	16K	C KB	£10.00
Threed OXO	Gm	Micro-Aid	16K	C IZ	£2.50
3-D Maze	Gm	Hexagon S/W	16K	C JA	£6.00
3D Maze	Gm	Earthshock S/W	16K	L KI	£3.00
Utilities	Ut	Golem	16K	C OB	£8.05
Utility A	Ut	Micro-Aid	16K	C IZ	£4.95
Valley	Gm	ASP Software	32K	C OD	£11.45
Varkman/Meanies	Gm	Aardvark Software	16K	C IU	£4.00
Vogon Attack	Gm	Micromail	32K	C OE	£6.33
Water Relations of Plant Cells	Ed	Garland Comp.	32K	C JX	£18.82
Where?	Ed	Micro Power	32K	C GK	£5.95
White Barrows/Conquering Everest	Gm	ASP Software	16K	C OD	£11.45
Wordwise	Gm	Computer Concepts	32K	C GJ	£74.75
World Geography	Ed	Micro Power	32K	C GK	£5.95
Zombies	Gm	Micro Power	16K	C GK	£3.95

Mighty Monsters

I. Nicholls

Improve the playability of Acornsoft's game Monsters with the addition of joystick control.

Acornsoft's game Monsters makes marvellous use of graphics and is very well-programmed but its one drawback is that you can only use the keyboard to control the game.

In case you're not familiar with the Monsters game the idea is this: you control a man who begins life on the bottom of a three layered building. Ladders are interspersed between the walled layers, and the man must walk up them and avoid monsters which appear on different levels. You must make the man dig holes in the layers into which the monsters are supposed to fall if they are walking in that direction.

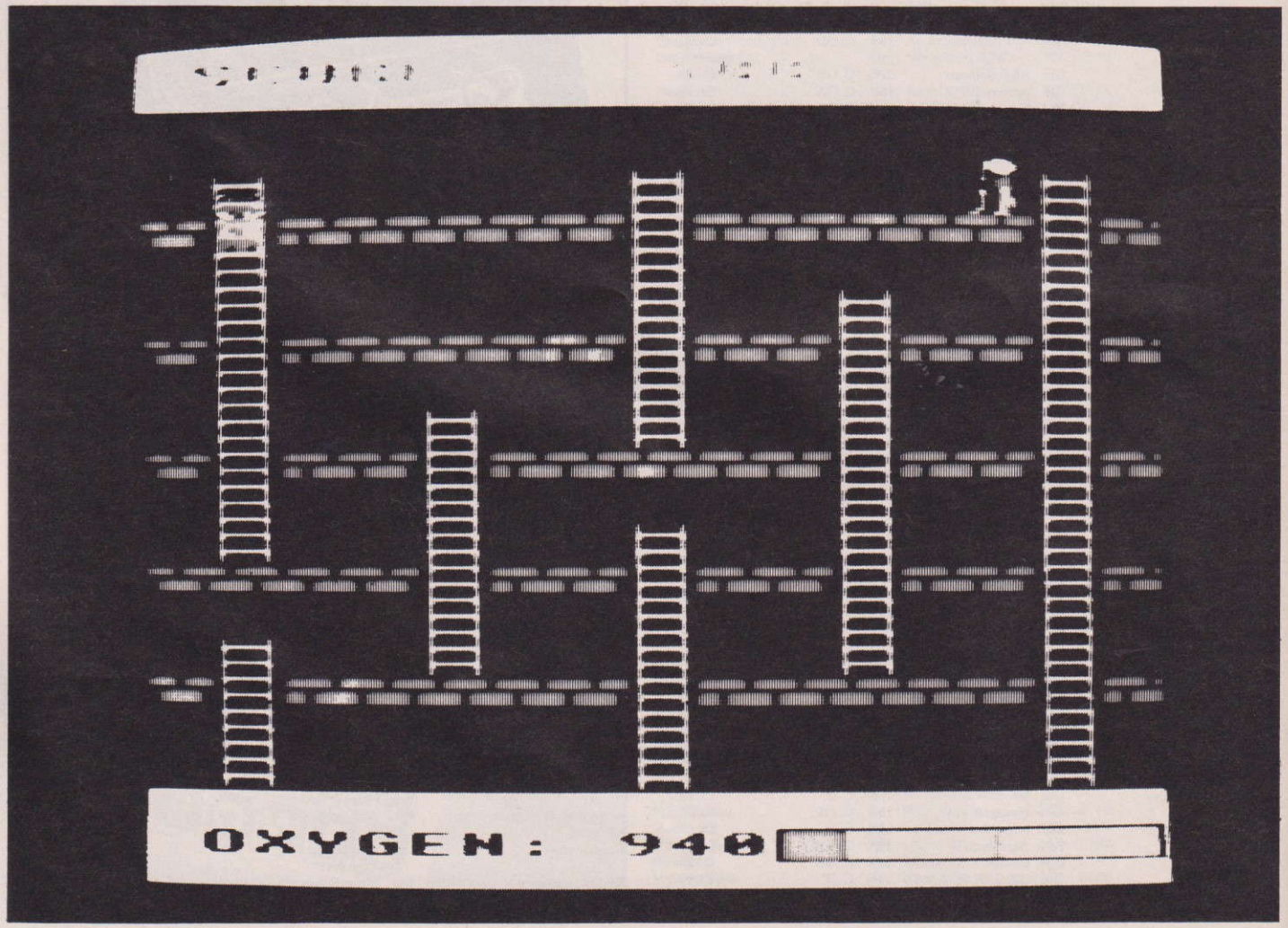
ALL FALL DOWN

When one falls down one of these traps your man has to be on hand to shovel quickly and deftly with his spade to cover over the hole and kill the monster. If you don't do that the chances are that the monster will get you within seconds and you'll be dead. Ultimately, the object is to beat the monsters and reach the top of the layered walls to gain as many points as possible.

You can overcome this problem by using the amendments listed below to the original Acornsoft program which allow you to use a joystick — and don't worry, we've had permission from Acornsoft to do so!

Now you can use your joystick to control the horizontal and vertical movements of the man throughout the game. If you press the 'fire' button on the joystick, the man doesn't move. By pushing the joystick downwards the man will dig a hole, and pushing it upwards makes him fill in a hole. Horizontal movement of the joystick has no effect.

What I have basically done, is to intercept the call to the subroutine in Monsters. This scans the keyboard and makes the program jump to my own subroutine 'JSmod', which has been loaded



into locations &0D00 to &0DA5. An extra line has been inserted into the first program on the tape to load *LOAD "JSmod" * before RUNning the main program.

At present, the modified version doesn't allow you to have the facility to use both keyboard and joystick control, but a further routine could be tacked on to test for either the space bar or the fire button being pressed, and to amend three bytes in Monsters accordingly.

MODIFICATION

1. A 6502 disassembler written for the BBC Micro was used to obtain a listing of the relevant parts of the main program.
2. The first few bytes of Monsters are as follows (I had already found that the execution address of Monsters was &0E02, by *LOADing using the * OPT 1, 2 command to give load and execution address):

```
MEMORY LOCATION     6502 MNEMONIC
(In Hex)
-----------------   -------------

0E02                JMP &1DF5
0E05                LDY #&05
0E07                STY &22D1
0E0A                LDA #&81
0E0C                LDX &22B2,Y
0E0F                LDY #&FF
0E11                JSR &FFF4
0E14                TAX
0E15                LDY &22B1
0E18                STA &06,Y
0E1B                DEC &22B1
0E1E                BPL &0E0A
0E21                RTS
```

The JMP &1DF5 statement jumps to a series of subroutine calls which create the main display of ladders, walls and creatures. The rest of the statements down to RTS are a subroutine which scans six keys on the keyboard (it appears to be called twice in the main program — location &11D0 onwards and location &1628 onwards both contain JSR &0E05). The routine scans the locations &22B7 down to &22B2. By breaking into the program when it was running and printing out the contents of these locations I obtained the following list:

The contents expressed as a negative number correspond to the values to be used with INKEY in BASIC to detect specific key depressions (the keys are shown in brackets).

The OSBYTE call corresponding to INKEY, OSBYTE with A = &81, is being used.

Since the **User Guide** does not give clear details of what is returned in index register X when using this call, I copied it into a BASIC program which printed the value of ?X, and I discovered that ?X = 0 if the particular key is not depressed and ?X = &FF if it is depressed. The function of this subroutine, is to load 0 or &FF into memory locations &06 to &0B in zero page depending on which keys were depressed of the six used in the game.

SAME AGAIN

My machine code patch must perform exactly the same function depending upon the joystick position. I decided that left, right, up and down movements of the joystick would correspond to depressing the keys Z, X, :, respectively on the keyboard. The 'dig' and 'fill' functions would be obtained by depressing the fire button and moving the joystick down or up, respectively.

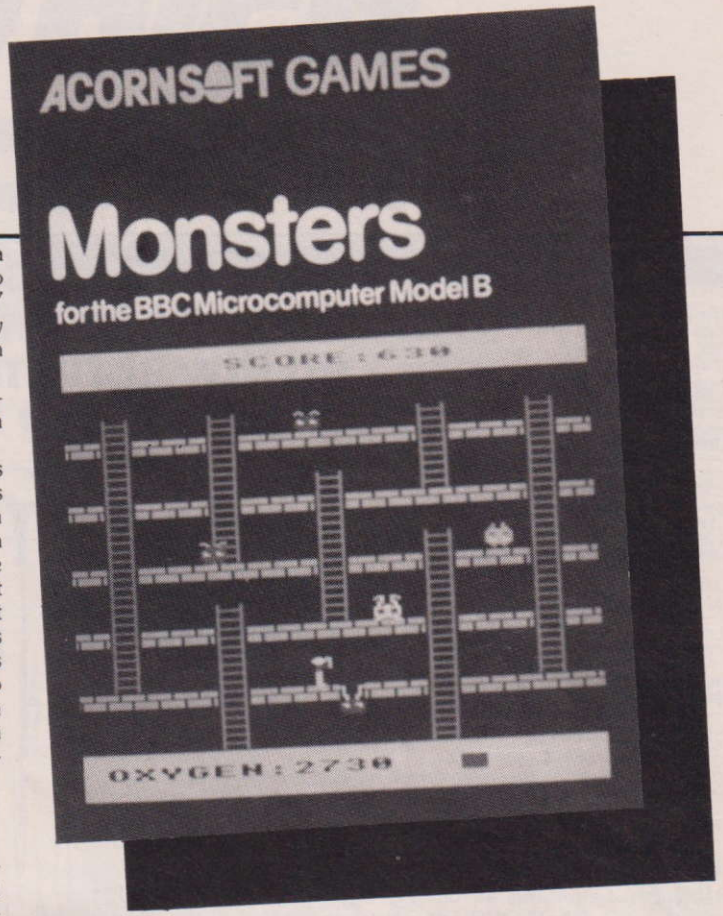
The simplest way of causing my program modification to be used instead of the routine at &0E05 seemed to be to modify bytes &0E05 — &0E07 so that instead of containing A4;05; 8C (LDY &05;STY...), they contained AC;00:0D (JMP &0D00) to give a jump to the modification stored from &0D00 onwards.

The modification itself is as follows, with explanatory comments:

*Line 75 has been added to Monsters 75 XLOAD"JSmod" (it has been saved with * SAVE "MONSTERS" 0E00 + 0800, 0E00 — rather than... 0E00 + 0700...).

PROGRAM LISTING

Loocation	Contents	Contents expressed as a negative number	
&22B2	158	-98	(Z)
&22B3	189	-67	(X)
&22B4	151	-105	(/)
&22B5	183	-73	(:)
&22B6	205	-51	(D)
&22B7	188	-68	(F)

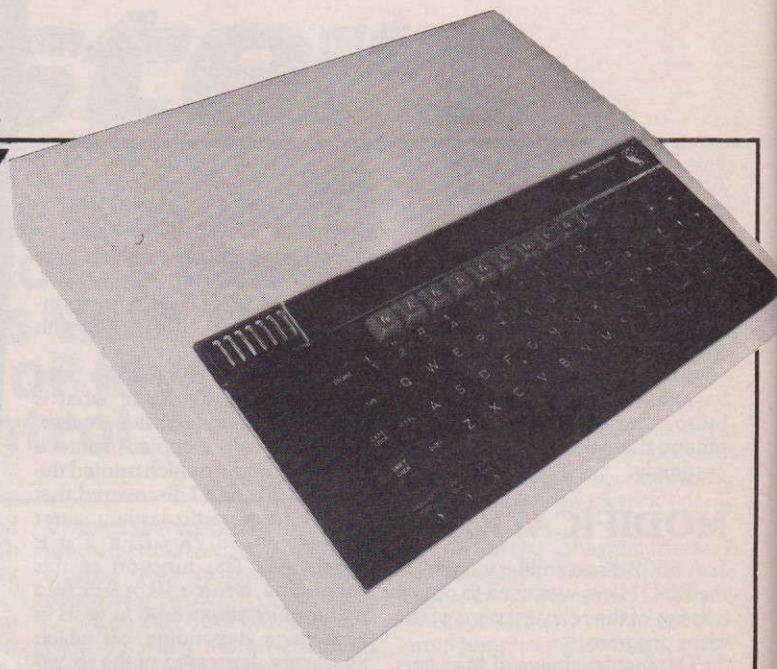


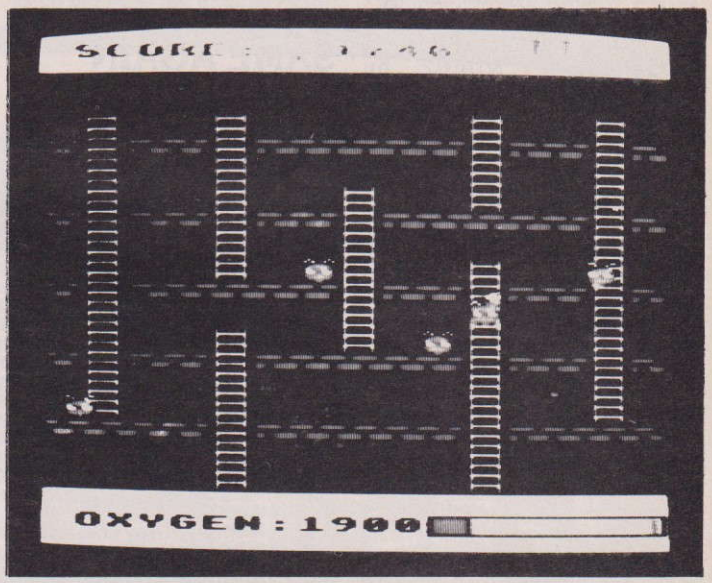

```
-------------------------------
LDA #&00
LDY #&06
.P1 DEX                         Put 0 in locations &06 to &0B
STA &06,X
CPX #&00
BNE P1
-------------------------------
JSR FIRE                        Jump to subroutine to test if fire button
                                pressed (store flag in &22B1)
-------------------------------
LDA #&01
CMP &22B1                       If button pressed, branch to
BEQ PRSD                        subroutine.PRSD
-------------------------------
JMP MAIN                        Otherwise jump to main scanning routine
-------------------------------
.PRSD LDX #&04
LDA #&80                        Read value of ADVAL channel 4 (OSBYTE
JSR &FFF4                       call with A=&80)
-------------------------------
JSR TEST                        Test value returned: store flag in &22B2
-------------------------------
LDA #&01                        Test to see if ?&22B2=1
CMP &22B2
-------------------------------
BEQ P2                          Branch to P2 if ?&22B2=1
BCC P3                          Branch to P3 if ?&22B2>1   (i.e. 2)
-------------------------------
LDA #&FF                        Put -1 in &0A (Joystick moved down)
STA &0A                                    "DIG"
-------------------------------
RTS                             Return to main program
-------------------------------
.P3 LDA #&FF                    Put -1 in &0B (Joystick moved up)
STA &0B                                    "FILL"
-------------------------------
.P2 RTS                         Return to main program
-------------------------------
.MAIN LDX #&03                  Routine which detects horizontal and
LDA #&80                        vertical movement of joystick to move
JSR &FFF4                       player. Read ADC channel 3 (left/right
                                movement)
-------------------------------
JSR TEST                        Test value of ADC channel flag stored in
                                &22B2
-------------------------------
LDA #&01                        Compare value of test flag with 1
CMP &22B2
-------------------------------
```

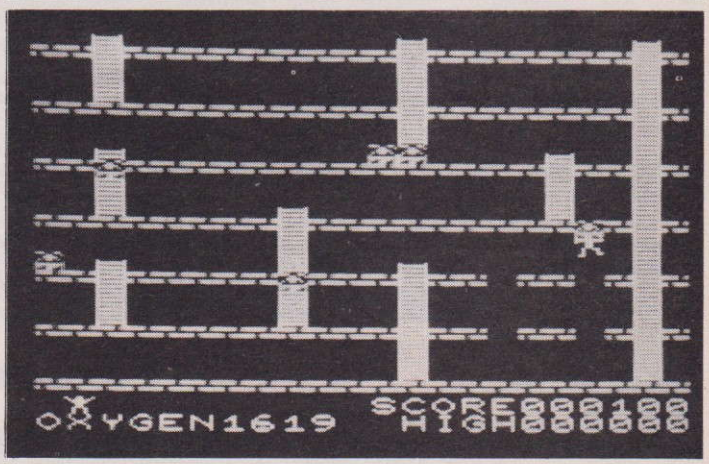
CONTINUED OVER

```
BEQ MAIN2                       Branch to MAIN2 if flag=1 (joystick
BCC M2                          control)
                                Branch to M2 if flag<1 (joystick left)
-------------------------------
LDA #&FF                        Put -1 in &07
STA &07                         (joystick right)
-------------------------------
JMP MAIN2                       Jump to MAIN2
-------------------------------
.M2 LDA #&FF                    Put -1 in &06
STA &06                         (joystick left)
-------------------------------
.MAIN2 LDX #&04                 Read value of ADC channel 4 (up/down
LDA $&80                        movement)
JSR &FFF4
-------------------------------
JSR TEST                        Test value of ADC channel: flag stored in
&22B2
-------------------------------
LDA #&01                        Compare value of test flag with 1
CMP &22B2
-------------------------------
BEQ L2                          Branch to L2 if flag=1 (joystick control)
BCC L3                          Branch to L3 if flag<1 (joystick up)
-------------------------------
LDA #&FF                        Put -1 in &08
STA &08                         (joystick down)
-------------------------------
RTS                             Return to main program
-------------------------------
.L3 LDA #&FF                    Put -1 in &09
STA &09                         (joystick up)
-------------------------------
.L2 RTS                         Return to main program
-------------------------------
.TEST CLC                       Subroutine to test ADC value returned
CPY #192                        - index register Y contains high byte
BCC TI                          of result, therefore:-
LDA #&02                           Y value > 192 ] ADC value > 49152
JMP T3                             Y value < 64  ] ADC value < 16384
.TI CPY #64                        For Y value > 192 ?&22B2=2
BCC T2                             For Y value =<192,>=64 ?&22B2=1
LDA #&01                           For Y value <64   ?&22B2=0
JMP T3
.T2 LDA #&00
.T3 STA &22B2
RTS
-------------------------------
.FIRE LDX #&00                  Subroutine to test if fire button 2 is
pressed
LDA #&80                        Read ADC channel 0
JSR &FFF4
TXA                             Transfer lower byte of value of ADC
channel
                                0 to Acc
AND #&03                        Mask out bits 2 to 7 (to leave value of
                                0,1,2 or 3)
CMP #&02                        Compare Acc with 2 to see if fire button
                                2 is pressed
BEQ F1                          Branch to F1 if button is pressed
-------------------------------
LDA #&00                        Button not pressed - load 0 in &22B1
STA &22B1
-------------------------------
RTS                             Return from subroutine
-------------------------------
.F1 LDA #&01                    Button pressed - load 1 in &22B1
STA &22B1
-------------------------------
RTS                             Return from subroutine

Other memory locations used:-

&22B1 contains result of subroutine FIRE   =1 if button 2 pressed
                                           =0 if button 2 not pressed

&22B2 contains result of subroutine TEST:-

                =2 if value of ADVAL channel >  49152
                =1 if value of ADVAL channel >= 16384,=<49152
                =0 if value of ADVAL channel <  16384
```

* The machine version of "JSmod" is saved on tape after "Monsters" and before monsters. It has been saved with XSAVE"-JSmod" 0D00 0DFF 0D00, to avoid corruption of the last byte at &0DA5.



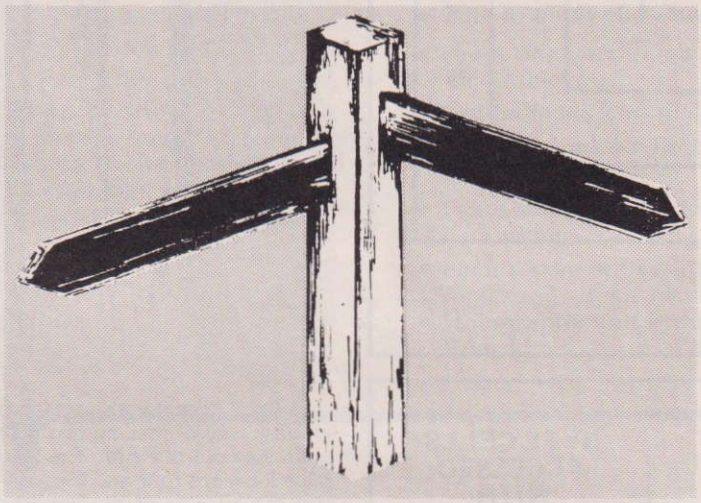




In Which Direction?

B Smith

The first in a new series in which many of the BBC Micro's commands will be examined in great detail.



In this article, which is the first in a series designed to take a simple look at some of the more awe-inspiring commands available on the BBC Micro, I shall try to show how those magnificently named Byte and Word Indirection Operators (? and ! for short!) may be put to some practical use. Both of these commands are peculiar to the BBC Micro, though they are also implemented on the ATOM. You will probably not come across them on other micros, however most have their own equivalent commands.

Okay, so what do these odd looking operators do?

A PEEK AND A POKE

The query operator, '?', can be used to read data from memory, or place data into memory, these two operations being known as PEEK (have a look!) and POKE (jab it in!). Because '?' operates on single bytes, only integer numbers in the range 0 to 255 can be handled.

In program Listing 1, when RUN the asterisk should plot gently across the screen. Line 80 generates a slight delay to enable us to see what would otherwise be a blur! The program as it stands is a little untidy. We can polish it up, by providing the '?' operator with a base address to which we can add an offset. The statement A?M = &FF means poke &FF into the address given by the sum of A + M.

We can therefore rewrite the program much more efficiently using a FOR...NEXT loop to provide the offset from the base address SCREEN shown in Listing 2.

```
?A=?B    i.e.   Poke A equal
                to peek B.
```

Now for some examples 'at the keyboard'. If you're not already there place your Beeb in teletext mode by executing:

MODE 7

then depending on your model enter:

```
Model A        ?&3C1F = ASC"A"
Model B        ?&7C1F = ASC"A"
```

the result should be that the letter 'A' appears at the top right corner of the screen.

To remove the 'A' enter:

```
Model A        ?&3C1F = ASC" "
Model B        ?&7C1F = ASC" "
```

What we have done in these two examples, is to first POKE the letter A, or move correctly, the ASCII code for 'A', into screen memory and then remove it by re-POKEING the same location with a space.

We can use this principle to create the illusion of movement across the screen.

```
10  MODE 7
20  SCREEN=&7C50:REM **
    USE &3C50 FOR MODEL A
30  REPEAT
40  ? SCREEN=ASC "*":REM **
    POKE STAR TO SCREEN
50  ? (SCREEN-1)=ASC " ":REM
    ** REMOVE LAST STAR
60  SCREEN=SCREEN+1:REM **
    INCREMENT SCREEN
70  TIME=0:REM ** SET CLOCK
80  REPEAT:UNTIL TIME>10
90  UNTIL SCREEN>&7C77:REM
    ** &3C77 FOR MODEL A
```

In program Listing 1, when RUN the asterisk should plot gently across the screen. Line 80 generates a slight delay to enable us to see what would otherwise be a blur! The program as it stands is a little untidy. We can polish it up, by providing the '?' operator with a base address to which we can add an offset. The statement A?M = &FF means poke &FF into the address given by the sum of A + M.

We can therefore rewrite the program much more efficiently using a FOR...NEXT loop to provide the offset from the base address SCREEN shown in Listing 2.

```
10  MODE 7
20  SCREEN=&7C50:REM ** &3C50
    FOR MODEL A
30  FOR S%=0 TO 31
40  SCREEN ? S%=ASC "*"
50  SCREEN ? (S%-1)=ASC " "
60  TIME=0
70  REPEAT:UNTIL TIME>10
80  NEXT S%
```

LOOK HERE

As a quick demonstration of the ? operators PEEKing ability, try the following,

```
10  ROM=&8007
20  FOR OFFSET=0 TO 14
30  PRINT CHR$(ROM?OFFSET);
40  NEXT
```

This short routine disassembles a few bytes of the BASIC ROM revealing Acorn's copyright message!

Line 30 of the program illustrates use of the base ? offset feature when PEEKing, and it is therefore perfectly legal to include statements such as A?B = C?D in our programs.

We can use facility to move blocks of memory too and fro'.

Set up one of those delightful function keys thus:

```
*KEY0 CLSIM LISTIM
```

and enter the following as shown:

```
10  SCRNTOP=&7C00:
    SCRNBOT=&7DB8
20  FOR N%=0 TO 200
30  SCRNBOT ? N%=SCRNTOP ? N%
40  NEXT N%
```

The values &3C00 and &3DB8 should be substituted in line 10 for the BBC Model A.

Hit KEY0 and RUN. The listed program should now be copied directly to the lower half of the screen, line 30 POKEing the address SCRNBOT + N with the PEEKed contents of SCRNTOP + N.

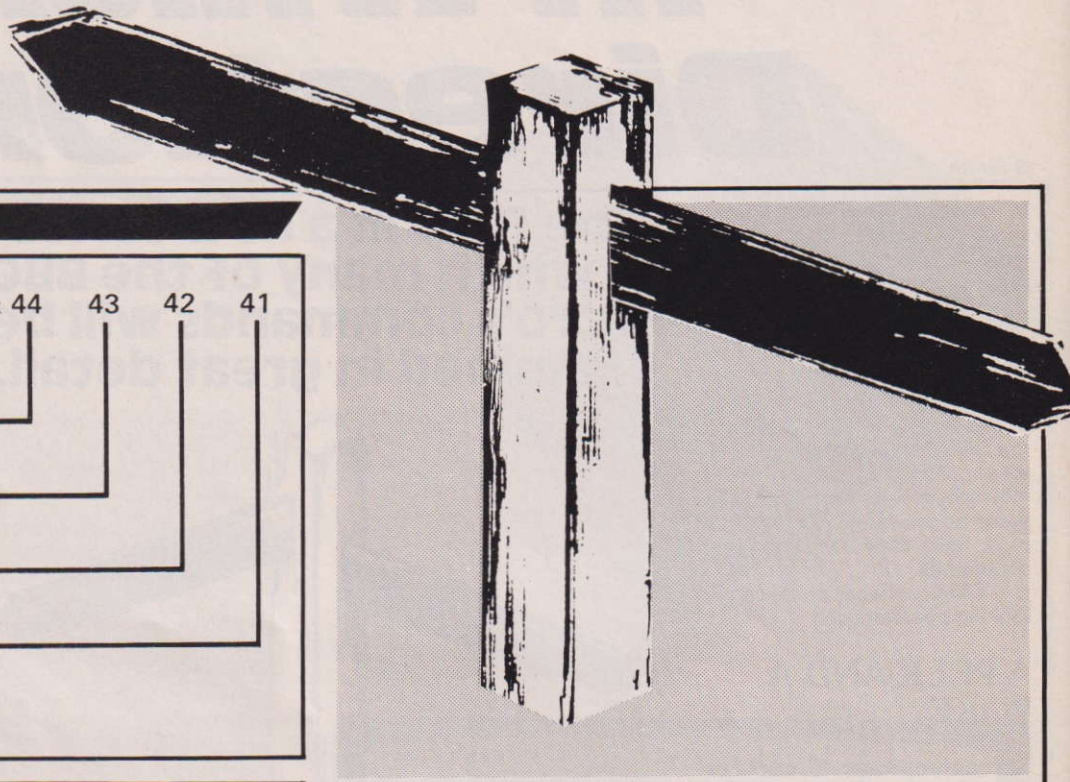
WORDS OF INDIRECTION

The pling '!' operator is simply an extension of the '?' operator, operating on words of up to four bytes at a time rather than single bytes. Similarly, it may also be used to PEEK or POKE information. The letters 'ABCD' can be POKEd onto the screen using the routine in Listing 5.

```
10  MODE 7
20  SCREEN=&7CD0:REM ** &3CD0
    FOR MODEL A
30  ! SCREEN=&44434241:REM **
    D,C,B,A
```

Note that in Listing 5 the ASCII codes for each letter are defined in the reverse order to which they are actually required. (See Fig. 1.)

CONTINUED OVER



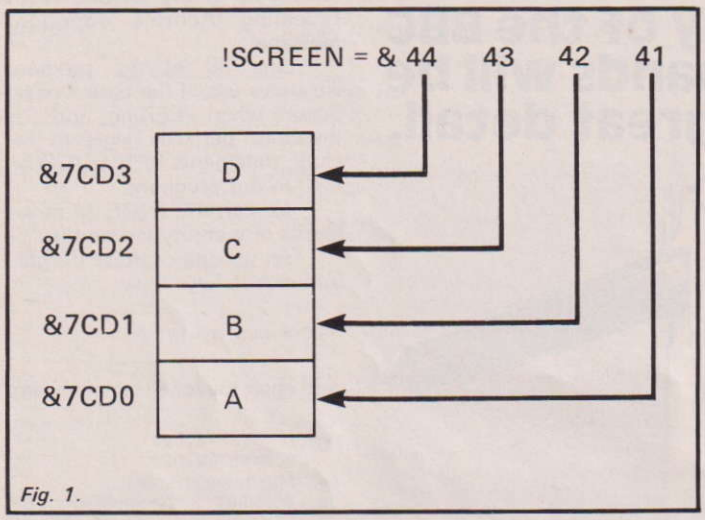
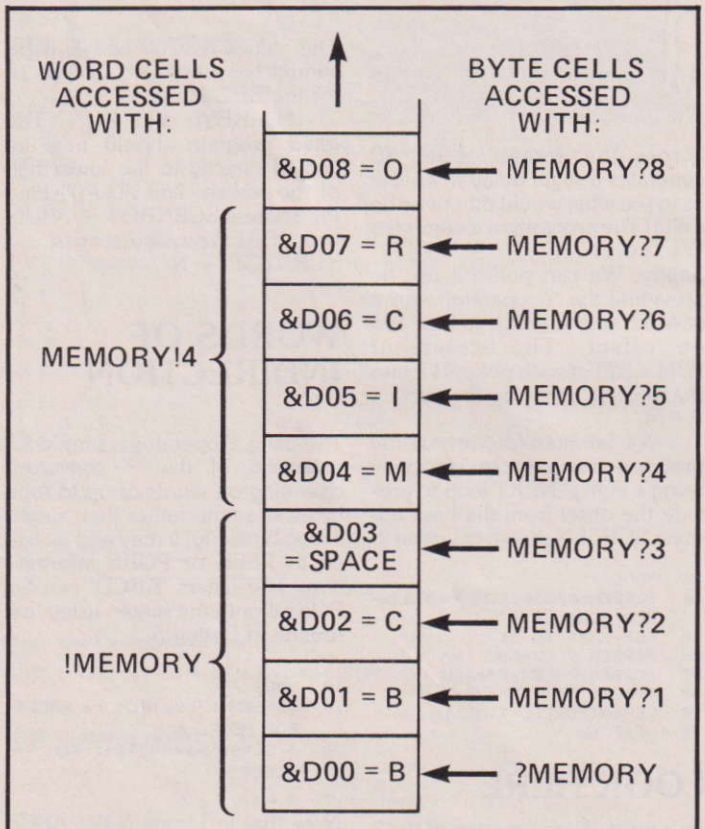


Fig. 1.



ACCESSING MEMORY WITH '?' AND '!'

If MEMORY from 8D00 contains the ASCII string 'BBC Micro' characters can be obtained with the statements shown.

The '!' operator also provides an easy way of reading two byte addresses out of RAM. The zero page locations four and five contain the address of the first free location in the BASIC stack, if required this address can be printed with:

```
PRINT ! 4 AND &FFFF
```

the AND &FFFF is required to 'mask-off' the two bytes at locations 6 and 7 which are not required, incidently these two bytes hold the current address of HIMEM.

BYTEING INTO ARRAYS

Both the indirection operators provide special forms of arrays. These are Byte and Word arrays and they differ from BASIC's normal arrays in two aspects. First, each array element consists of either a single byte or a word of four bytes, and second they are only one-dimensional and may not have subscripts. Because both arrays act in a single dimension they are more correctly termed VECTORS.

Space must be made available for these vectors before they can be used. As with BASIC arrays, the DIM statement can be used, but in a slightly modified form. To dimension 10 bytes enter:

```
DIM X 9
```

Notice that the number of vector elements is *not* enclosed in brackets and a space exists between the variable X and the numeric value, 9.

Alternatively, the variable may be assigned a value corresponding to the start address of an area of memory available for data storage.

In each case each individual element in the vector is accessed by using an offset from the base variable.

Figure 1 shows how the individual elements of a vector may be accessed.

Finally, to illustrate how byte and word vectors may be used, try the following program which uses the word operator to POKE a four byte string into successive memory locations, subsequently printing the string using the byte operator.

```
10  DIM X 9
15  REM ** POKE USING !
    OPERATOR
20  FOR N=0 TO 255 STEP 4
30  X!N=&20434242
40  NEXT
45  REM ** PEEK USING ?
    OPERATOR
50  FOR N=0 TO 255
60  PRINT CHR$(X?N);
70  NEXT
```

If you missed out on Breadboard '79,'80,'81 and '82 . . . . . Now's the time to catch up with BREADBOARD '83

BREAD BOARD

The *premier* electronics show for the enthusiast. Moved this year for your convenience and comfort to the Cunard International Hotel, Hammersmith, London W6

Friday November 25th	10am - 6pm
Saturday November 26th	10am - 6pm
Sunday November 27th	10am - 4pm

Why not bring the family to the show and enjoy a weekend in London? We have arranged a complete hotel package for our visitors to the exhibition. All inclusive rail tickets also available. Send now for details of what *we*, the organisers, can offer *you*. Write to:
Breadboard '83
ASP Exhibitions
145 Charing Cross Road
London WC2H 0EE

This year's features include

- Computer controlled model railway competition. (send off for entry form now)
- Kits to build all sorts of projects.
- Magazines and books covering most aspects of electronics.
- Components and tools at bargain prices.
- Technical advice on electronics projects and Ham Radio technique.
- Clubs/User groups for amateur electronics and computing
- Lectures by professionals on a wide range of topics. eg. Basic Tuition For Beginners,
- Holography and Producing Printed Circuit Boards.

FROGLET FROGLET

The program follows the style of that arcade favourite, Frogger and you must negotiate your green coloured friend across first a road populated with fast moving cars and lorries and then over the river by means of turtles and logs. Frogs may be able to swim but if your turtle decides to sink you'll be swept away by the current. Once across with three out of your four frogs you'll be able to score bonus points by catching the flies that appear over the river but, take care that you don't run out of time!

All in all it's a great, fast moving game that any number of people can play and, just to add to the spirit of the thing the program stores the top ten players' names so you can measure your performance. How much are we charging for this minor masterpiece? Just **£5.99** all inclusive!

To order simply fill in the coupon and send it with your remittance to:
ASP Software, ASP Ltd, 145 Charing Cross Road, London WC2H 0EE

Please send me .... tape(s) of Froglet for my 32K BBC Micro

I enclose my cheque/Postal Order/Money Order (delete as necessary)
for £..... (payable to ASP Ltd)
OR
Debit my Access/Barclaycard (delete as necessary)

FROGLET

Please use BLOCK CAPITALS and include your postcode.
NAME (Mr/Mrs/Miss) . . . . . .
ADDRESS . . . . . .
. . . . . .
. . . . . POSTCODE . . . . . .
Signature. . . . . Date. . . . . .

A & B COMPUTING

Lineage: 35p per word.

Semi display: £8.00 per single column centimetre
Ring for information on series bookings/discounts.

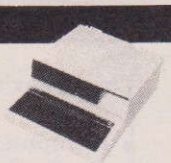
All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01-437 1002 EXT 297

Send your requirements to:
SALLY COLLINS
ASP LTD, 145 Charing Cross Road, London WC2H 0EE.

ACCESSORIES

BBC CONSOLES



The only expandable console for the BBC. The console will house disc drives, 2nd processor teletext adapter etc. With all wiring out of sight within the console. **Coming soon:** an extra bolt on module for extra expansion space. Yes, this console will grow with you. Basic console as shown only **£39.99** + VAT plus £4.00 p&p. Send cheque made to **Silent Computers**. Matching printer stand can double for VDU stand over the micro, only **£14.99** + £2.00 p&p. For more details: **SILENT COMPUTERS, 27 Wycombe Rd., London N17 9XN. Tel: 01-801 3014. MAIL ORDER ONLY. Viewing by appointment only.**

BBC MODEL B GRAPHICS TABLET

PLUG-IN READY TO GO

Trace, draw, save and load screen. Delete, rub-out, instant circles, triangles, polygons, arcs, vertical/horizontal brush lines. Shapes can be dynamically **ROTATED** and filled or part-filled with **COLOUR** or **COLOUR- SHADES**. Cassette based. **£45.00**, p&p included + 15% VAT.

I-TWO
1 Abbots Place, London NW6
TEL: 01-624 5404

BBC USERS. Purpose built mobile work station designed specifically for the BBC plus cassette, disc drive, monitor and printer. Work stations supplied to schools but also ideal for home use. Stylish, rugged construction. 2 heights/lengths. Largest unit priced at £47.95 + p&p. Send now for brochure to Setcraft Sussex Ltd., 32 Walpole Ave., Goring, Sussex.

SOFTWARE APPLICATIONS

BBC "B" Wordprocessor. Suitable for use with any printer. £5.95. Epson Printer Software for BBC, send SAE for list. Processor Applications, 22 Mercer Close, Basingstoke, Hants.

BBC DISC USERS

ANALYSE DISC

• Copy any BBC Disc or any part of a disc. • Recover information for faulty discs. • Protect your own discs against copying.
All of these facilities are available with our analyse disc. Full documentation and user support. **£15.00** inc p&p.
D.A. Computers Ltd, 104 London Road, Leicester. 0533 549407.

SOFTWARE EDUCATIONAL

EDUCATIONAL PROGRAMS

Are you old fashioned enough to feel that ultimately children must sit down and practise arithmetic skills?

Please send for details of our top quality programs which are used in schools throughout the country and are now available for extra tuition in the home.
Ages 4 to 13 years.

COTTAGE SOFTWARE (BBC)
Heather Cottage, Selly Hill, Whitby, N. Yorkshire.

PASS O & A LEVEL CHEMISTRY

8 programmes for individual revision
Tape 1 O-level i) unknowns* ii) equations. **Tape 2** A-level i) inorganic unknowns* ii) physical chemistry **Tape 3** A-level i) aromatic synthesis ii) aliphatic synthesis. **Tape 4** A-level i) organic unknowns (1) ii) organic unknowns (2). *Model B 32k only. 16k otherwise.
Each tape containing two programs is available at **£5** or at **£15** for all four tapes (inc of p&p) from: **E.H.B. (Software) 55 Higham Lane, Tonbridge, Kent.**

BBC SOFTWARE

EDUCATIONAL 1	£8.05
EDUCATIONAL 2	£8.05
FUN WITH WORDS	£8.05
GAMES OF LOGIC	£9.20
KATAKOMBS	£9.20
UTILITIES	£8.05
SUPERLIFE	£9.20

SPECIAL OFFER ANY 3 TAPES £20.70
GOLEM LTD., (Dept A&B), 77 Qualitas, Bracknell, Berkshire RG12 4QG
TEL: (0344) 50720

LOGO Language for 32K BBC- turtle graphics, procedures, parameters, recursion, parallel programming. £8.50 from Summit Software, 96 Toftwood Road, Sheffield S10 1SL.

ADVERTISE YOUR EDUCATIONAL SOFTWARE IN A & B COMPUTING & GET THE BEST RESPONSE PHONE
01 437 1002 **extn 297**

SOFTWARE BUSINESS

BBC Micro A/B Users. Operate your own Teletext compatible information service of up to 9/25 on-line pages with real-time date and clock. Ideal for shops, offices, public institutions, educational establishments etc. The Viewtext package comprises two interactive programs for page creation and display, comprehensive user guide and demonstration pages. Cheques/POs for £9.95 or SAE for details to: IT Services, 27 Waterford Park, Radstock, Avon BA3 3TS.

SOFTWARE GAMES

"SUBMARINE"

Simulated radar-assisted attack by WW2 T-Class S/M on German Battleship 'Tirpitz'. For BBC Micro (32K) **£6.00. SOFT OPTIONS, 6 Remigius Grove, Lincoln.**

REPAIRS & SERVICES

COMPUTER & PERIPHERAL SERVICES, ASHTEAD OFFER

On-site maintenance contracts on BBC Computers and peripherals at 15% of total cost. 12½% on 5 or more. Schools, dealers etc. One-off on-site service £30 plus parts. 24 hour call.

TEL: Ashtead 74297

FOR ALL YOUR NEEDS
ASP CLASSIFIED
01 437 1002

COURSES

PROGRAMMING IN BASIC 1983

Two excellent University taught programmes for

BBC MICRO USERS
17th — 22nd July

PET COMMODORE USERS
24th — 29th July

Residential and non-residential
from £85.00

Details from the Business Manager, University of Manchester Institute of Science & Technology, Sackville Street, Manchester M60 1QD

Courses suitable for teachers and school students. Local Authority grants may apply.

WANTED

Up to 30% Royalties

paid for high quality original software BBC Model B

FAST ACTION ARCADE STYLE GAMES

Silverlind

156 Newton Road, Burton on Trent, Staffs DE15 0TR.

BOOKS & PUBLICATIONS

BBC MICRO INDEX

Where to find that article!

Grouped alphabetical reference to over 1500 published BBC Micro articles, program techniques, hardware and software tips, utility, general, music and games programs, clubs, book reviews, commercial software reviews, special hardware etc. etc.
PRICE £3.00 inc p&p by return.
BEEBDEX
13 Viaduct Road, Brighton, Sussex.

BERKSHIRE

BBC SOFTWARE

[illegible]	£8.05
[illegible]	£8.05
[illegible]	£8.05
[illegible]	£9.20
[illegible]	£9.20
[illegible]	£8.05
[illegible]	£9.20

SPECIAL OFFER ANY 3 TAPES £20.70

GOLEM LTD., 77 Qualitas, Bracknell, Berkshire. RG12 4QG. TEL: (0344) 50720

CAMBRIDGESHIRE

Control Universal Ltd
The Hardware House

Control Universal Limited.
Unit 2, Andersons Court, Newnham Road, Cambridge. Tel: (0223) 358757

Leading Eurocard hardware house, manufacturing the CUBE range of control boards, including BEEB-EX – the 1MHz interface card linking CUBE and the BBC Microcomputer. Distributor of BBC Micros, printers, video monitors, stationery, disks etc. Full technical support is given by our resident electronic engineers. Send for our free 150 page catalogue.

EXETER

COMPUTERLAND
(A & D COMPUTERS)

BBC Micros. Disc Drives. Printers. Cumana Drives. Microvitec Monitors. Software by Acorn. Gemini. Program Power. Bugbyte Software for all etc.

6 City Arcade, Fore Street, Exeter.
TEL: 77117

HAMPSHIRE

R. D. S. ELECTRICAL
AGENT FOR BBC AND ACORN

Full support in hardware and software. OPEN ALL DAY SATURDAY.

157-161 Kingston Road, PORTSMOUTH
Tel: 0705 812478

HEREFORDSHIRE

STEVENAGE

Computers for home and business.
• Disc Drive/Upgrades • Printers • Monitors • Business Packages • Games • Service Contracts •

Q-TEK SYSTEMS
TEL: 0438 65385

ACORN COMPUTER

KENT

MEDWAY COMPUTERS LTD.

• BBC authorized dealers and service centre • Torch Z80 disc pack available • Our own credit facilities • Access and Barclaycard welcome •
We are open 9 till 5.30 six days a week.

141 New Rd., Chatham, Kent ME4 4PT.
TEL: (0634) 826080

LANCASHIRE

MICROROSE LTD

Official dealer and service centre. BBC and Commodore. Full range of games and educational software available for BBC and Commodore.

MAIN STREET, HIGH BENTHAM LA2 7HJ
TEL: (0468) 62180

ACCESS/BARCLAYCARD WELCOME

LEICESTERSHIRE

D. A. COMPUTERS LTD.

Official BBC dealer and service centre. Everything you require for the BBC computer inc. Epson Printers, Shugart Disc Drives, P.L. Digitiser, Micro Vitec, Monitors. Full range of games software and books.

104 LONDON ROAD, LEICESTER.
TEL: (0533) 549407

LONDON

THE VIDEO PALACE

Londons largest home computer store. Model B and Torch. ZX stockists. Full range of games software.

62 Kensington High Street, London W8
TEL: 01-957 8537

D.D.M. DIRECT DATA MARKETING LTD
86/87 WILTON ROAD, LONDON S.W.1
22 WARLEY HILL, BRENTWOOD, ESSEX

• Fully suthorised BBC dealers • Wide range of printers, disc drives etc. • Our own "Wordmaster" word processing package • Dealer enquiries welcome • Full time engineering support and advice centres •

01-834 5096/5016 or 0277 214168/229379

MIDDLESEX

TWICKENHAM COMPUTER CENTRE
Acorn • BBC • Apple • Commodore

Micro Computers for home and business plus peripherals/software and accessories. Always a wide range in stock at:

72 Heath Road, Twickenham, Middx.
TEL: 01-892 7896

SURREY

THE COMPUTER SHOP

J. S. Simnett Computers Ltd., official BBC dealer and service centre. Everything for the BBC and Torch Computers plus peripherals (e.g. Epson FX80 printer **£373** plus V.A.T.)

91 Acre Road, Kingston-on-Thames, Surrey Tel: 01-546 3793

ACCESS/BARCLAYCARD WELCOME

A&B's Nationwide Dealerguide 01 437 1002

SUSSEX

CASTLE ELECTRONICS

Full range of BBC Computers in stock, plus disc drives, printers and monitors. BBC Econet available. Ring for competitive prices **0424 437875**

7 CASTLE STREET, HASTINGS, EAST SUSSEX.

EXPORT ORDERS WELCOME

WALES

TECH NEG TECHNICS CLEWYD LTD

We are Acorn/BBC Econet specialists. Torch and Zenith dealers. Educational discounts available.

The Coach House, Kelstorton Road, Flint, Clewyd CH6 5TH
TEL: DEESIDE (0244) 810518

S.I.R. COMPUTERS LTD

Agents for BBC, Acorn and Torch. Computers readily available plus software, accessories and peripherals.

91 WHITCHURCH ROAD, CARDIFF.
TEL: (0222) 21341

ACORN COMPUTER BBC

WARWICKSHIRE

CARVELLS OF RUGBY LTD
3/7 Bank Street, Rugby CV21 2QE
THE ACORN/BBC SPECIALISTS

We also stock Cumana disc drives, Epson/Olivetti printers, Microvitec Monitors, Torch ZX disc pack + BBC Software and Books.

PHONE: (0788) 65275/6
Half Day Closing Wednesdays

YORKSHIRE

POWER MICRO POWER MICRO POWER

MICRO POWER LTD.

The leading B.B.C. dealer in the North

8/8A REGENT STREET
CHAPEL ALLERTON LEEDS LS7 4PE
TEL. LEEDS 683186 or 696343

3D COMPUTERS

THE HOME COMPUTER SPECIALISTS
Easy parking at all branches

TOLWORTH
230 Tolworth Rise South Tolworth, Surbiton, Surrey
TEL: 01-337 4317

SUTTON
30 Station Road, Belmont, Sutton, Surrey
TEL: 01-642 2534

EALING
114 Gunnersbury Ave, Ealing, London W5 4HB
TEL: 01-992 5855

RICKMANSWORTH
Greystone Works, The Green, Croxley Green, Rickmansworth, Herts
TEL: (0923) 779250

MILTON KEYNES
Unit 1, Heathfield, Stacey Bushes, Milton Keynes
TEL: (0908) 317832

NEWBURY
26 Stanley Road, Newbury, Berks
TEL: (0635) 30047

BBC MICRO

ELECTRON, TORCH Z80 DISCS
CUMANA DISCS, EPSON PRINTERS

Full range of BBC Software and books always in stock.

A & B COMPUTING

CLASSIFIED ADVERTISEMENT — ORDER FORM

1	2	3
4	5	6
7	8	9
10	11	12
13	14	15

35p per word (min 15 words). Simply print your message in the coupon and send with your cheque or postal order made payable to Argus Specialist Publications Ltd to:

CLASSIFIED DEPT., A & B COMPUTING,
145 Charing Cross Road, London WC2H 0EE.
Tel: 01-437 1002

PLEASE STATE CLASSIFICATION REQUIRED

Name . . . . . .

Address . . . . . .

. . . . . .

. . . . . .

Tel.No.(Day) . . . . . .

Please place my advert in A & B Computing for months. Please indicate number of insertions required.

AD INDEX

BBC, DRAGON & ATARI GAMES FROM QUICKSILVA

As part of the plan to expand their range to cover all the most popular personal computers QUICKSILVA are proud to introduce 3 new programs for the BBC Computer, plus a version of their popular "MINED-OUT" game for the Dragon and also a character generating program for the Atari 400/800 "MAGIC WINDOW".

DRAGON

MINED-OUT

On Screen Features:

Tricky Mines, Bonus features, Mine detection indicator, 7 levels of minefield, Safe area, Progressive difficulty, Scoring, High score feature. Full instructions.
Controls: Up-Down-Left-Right.
Special Features: Bill the Worm, Bugs (mines with legs). Action Replay.
Author: **I. & C. Andrew**

ATARI 400/800

MAGIC WINDOW

Define characters on 8x8 grid. Save to tape. Load from tape. Freely merges character sets into existing programs. Hex & Dec. O/Ps. Character manipulation: Mirror, rotate, invert, left, right, up, down, hold, wipe, cancel. Redefine whole character set. Also works in colour mode. Excellent examples supplied.
16K RAM or more.
Author: **M. Walker**

BBC GAMES

MUSIC PROCESSOR

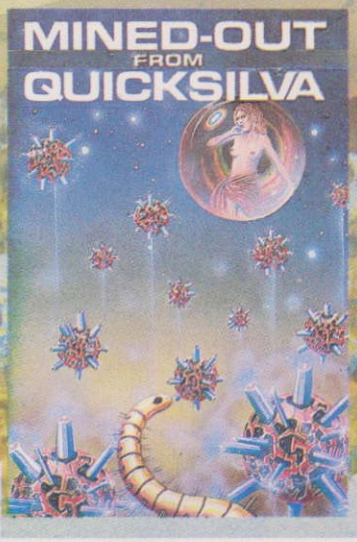
Exploits your BBC music potential to its limits.
Features: Full push button operation, over 30 separate functions for recording, mixing, editing and sound effects etc. 4 channels, 5 octaves, 100,000 envelopes easily selected. Over 2000 note storage capacity. Ready made music files included.
Author: **Andy Williams**
For model A or B

PROTECTOR

Features: Pods, monsters who kill on contact, mine laying UFO's, monsters increase in size as game progresses, new pods every 1000 pts., hyperspace function. Full colour hi-res graphics, sound. Joystick control.
For the model B with 16K + Joystick.
Author: **Andy Green**

WIZARD

The lands most beautiful maidens are being sacrificed by an evil King. Can you be their saviour, fighting off hordes of evil demons armed only with your wand and magical powers. A most compulsive and original game with hi-res colour graphics and full sound.
Author: **A. R. Buckley**
For the model B with 32K



Quicksilva Limited,
Palmerston Park House,
13, Palmerston Road,
Southampton. SO1 1LL
Telephone: (0703) 20169

Please send order to
Quicksilva Limited,
Palmerston Park House,
13, Palmerston Road,
Southampton.

BBC GAMES

Music Processor	£14.95	☐
Protector	£7.95	☐
Wizard	£6.95	☐

DRAGON GAME

Mined-Out	£5.95	☐

ATARI 400/800

Magic Window	£8.95	☐

WARNING: These programs are sold according to QUICKSILVA Ltd's terms of trade and conditions of sale. Copies of which are available on request.

Access BARCLAYCARD VISA

Please send me the games as ticked.
Total cheque/PO. enclosed ..... Cheques payable to Quicksilva
Supplied on Cassette

Name _____

Address _____


ANOTHER FABULOUS B.B.C. PROGRAM FROM BRITAIN'S LEADING SOFTWARE HOUSE
Killer Gorilla
Once again, Killer Gorilla holds captive a young and beautiful heiress. Is the age of CHIVALRY dead? Answer the maiden's cries for help and scale the ironwork tower. Race along Girders, Climb Ladders, Jump Gaps, Leap onto moving Elevators and Career along Conveyors. Dodge or jump the rolling barrels or grab a hammer and smash a few. Watch out for the fireballs and iron beams hurled with animal passion. Sensational, full feature machine code arcade game with four phases, increasing difficulty and speed, bonus points and the highest standard of graphics yet achieved on the BBC micro. Only £6.95
Other programs available: Swoop (B) £6.95/Croaker (B) £6.95/Chess (B) £6.95/Laser Command (B) £6.95/Galactic Commander (B) £6.95/Filer £8.95/Timetrek (B) £6.95/Micro Budget £6.95/Moonraider (B) £6.95/Barrage (B) £6.95/Beebmon (B) £6.95/The Labrynthe of La Coshe (B) £6.95/Draw (B) £8.95/World Geography (B) £5.95/Spacemaze (B) £5.95/Munchyman £5.95/Seek £5.95/Eldorado Gold (B) £5.95/Dissassembler £5.95/Constellation (B) £5.95/Junior Maths Pack (B) £5.95/Where? (B) £5.95/Painter (B) £5.95/Chemistry (B) £5.95/Physics (B) £5.95/Caveman Adventure (B) £5.95/Astro Navigator (B) £4.95/Startrek £4.95. Reversi 1 £4.95/Reversi 2 (B) £4.95/Roulette (B) £4.95
Written any Programs? We pay 20% Royalties for B.B.C. & ELECTRON PROGRAMS
WE Guarantee THAT ALL OUR ADVERTISED PROGRAMS HAVE BEEN COMPLETED AND ARE READILY AVAILABLE
WE ARE AUTHORISED DEALERS FOR ACORN ATOM, BBC MICRO & DRAGON 32
SPECIAL OFFER
Deduct £1 per cassette when ordering two or more.
MICRO POWER LTD.
Dept. ST6
8/8a REGENT STREET,
CHAPEL ALLERTON,
LEEDS LS7 4PE
Tel: (0532) 683186 or 696343
Please add 55p order P & P + VAT at 15%
Access
VISA
Please Note:
All programs are now available at all good dealers or direct from MICRO POWER LTD.
PROGRAMPOWERMICROPOWER