AN ARGUS SPECIALIST PUBLICATION

DECEMBER 1985

A&B COMPUTING

£1.50

FOR USERS OF THE BBC MICRO AND ELECTRON

SIMULATIONS
THE SUBURBAN FOX AND THE FAST CAR

ROM SOFTWARE FOR BEGINNERS

MEMORY MODULE: 64K PERMANENT RAM

UNIX FOR BBC BUSINESS PRODUCTIVITY

SLEUTH II

ALTRA ROMs

FAST FLOOD FILLS

PRINTER BUFFER FOR MODEL B+

BUMPER PACK OF CHRISTMAS GAMES – PLAY SCROOGE!

16K SIDEWAYS RAM FOR £10

SPORTING ACTION: OCEAN'S MATCHDAY FOOTBALL

SIMPLY LIGHT YEARS AHEAD

THE BEST OF BRITISH ROM SOFTWARE

FOR THE BBC AND SECOND PROCESSOR

ADVANCED & POWERFUL BUT EASY TO USE

DUE TO REDUCTION IN ROM COSTS - NEW LOW PRICES!

See us on stand 3 at UMIST show Manchester

ALTRA ENIGMA DISC IMAGER

Enigma Disc Imager is a 16K ROM which will give you 29 additional powerful disc commands. Some of which are: Enigma will IMAGE ANY DISC, Copy sectors, Repair bad tracks, Read deleted or special sectors, Dumps roms to disc, Loads roms from disc to sideways ram, Moves a programme up or down in memory, Unlocks locked tapes, Locks your tapes, Dumps cassette to disc, Global or selective renaming of directories, Writes deleted or special sectors to disc, Track and sector ID editor facilities, Selective file copier, Formats discs to be 40/80 track compatible (requires an 80 track drive with this command), Special disc editor to cope with protected discs, Special formatter to format discs with non standard tracks, Inserts new files into the catalogue, Enigma makes easy work of editing, copying, repairing, etc., of both standard and non standard discs.

ALTRA ENIGMA DISC IMAGER IS TUBE COMPATIBLE

ALTRA PROBE

Altra Probe is a 16K ROM which will give you 59 additional powerful machine commands. Some of which are: Altra Probe will list a basic programme straight from file, Formats basic assembler text output, Dissembles, Unpacks, Super Packs, Relocates, Edits memory, Switches off roms, Lists roms, Copies roms to specified address. Calculates free memory, Copies screen text to printer, Lists all specific types of basic variables and values, Graphics dump for Epson or NEC printers, Calculates and details free space on a disc, On board formatter which will automatically format a dual disc, Onboard formatter and verifier. Repairs bad tracks, Disc sector editor, Turns the tube on and off. These are only a few of the commands available from Altra Probe. Altra Probe makes easy work of editing, programme development, etc.
Altra Probe is available in two versions. PROBE1 and PROBE 2.

ALTRA PROBE 2 IS TUBE COMPATIBLE

ALTRA BASIC ED

Basic Ed, is on 8K ROM which contains 21 additional commands. They are: $ search, List matches, Number matches, Global replace, Selective replace, Bad programme relink, Copy lines from one part of a programme to another. Format a listing, Move, Super pack, Renumber, Table line references, Unpack, Variables X ref, List entire programme, Keyboard immediate mode, Printer on/off, Paged mode on/off, List match lines, Concatenate, Strips rems, spaces etc. Altra Basic Ed, is an essential helpful tool for the Basic/Assembler programmer. Altra Basic Ed. is available in two versions. Basic Ed.1 and Basic Ed. 2.

ALTRA BASIC ED. 2 IS TUBE COMPATIBLE

ALTRA FIRST AID 1.1

First Aid 1.1 is an 8K ROM which contains 19 additional commands. They are: Machine code dissembler, Hex and ASCII dump. Sideways rom to ram copier, Function key predefined facilities, Variable table listing, Memory space free calculator, Checksum calculator, Rom lister, String search, Define function keys with strings, Rem and space Stripper, Bad programme fixer, Clear all variables, Clear ram from &0400 to &8000, Machine code monitor. Lists O.S. calls with their vectors, Lists the first 24 control codes. Altra First Aid 1.1 was designed to assist the Basic/Assembler programmer and the sideways rom user.

ALTRA TUBE EDITOR

ALTRA TUBE EDITOR is a new and unrivalled full labelling Disassembler and editor Rom for the BBC and the second processor. Some utilities of ALTRA TUBE EDITOR are append labels to those in memory, Disassemble in 6502 and 65C02 mnemonics, End file overlay, Delete given label, Delete hex range, Delete markers, Dump memory, Edit memory, Overlay files, Find strings and bytes, Define a hex region, Define a label, List memory, Load labels, List labels, Set memory, Calculate and print values, Print embedded text in a program after a call, Saves Roms, Saves labels. ALTRA TUBE EDITOR also supports many of the 6502 "ILLEGAL" OPCODES.

ALTRA TUBE EDITOR FUNCTIONS IN THE BBC AS WELL AS IN 6502 SECOND PROCESSOR.

ALTRA ROMS

Access

VISA

209 North Street, Leeds LS7 2AA.

Telephone orders please ring: 0532 451508
Send order to: **ALTRA ROMS, 209 North Street, Leeds LS7 2AA.**

☐ I enclose my order for the above products.
☐ *Please send me information sheet on*_____ ☐ *S.A.E. enclosed*

Name _____

Address _____

_____

_____

OFFICIAL ORDERS WELCOME

Please send me:

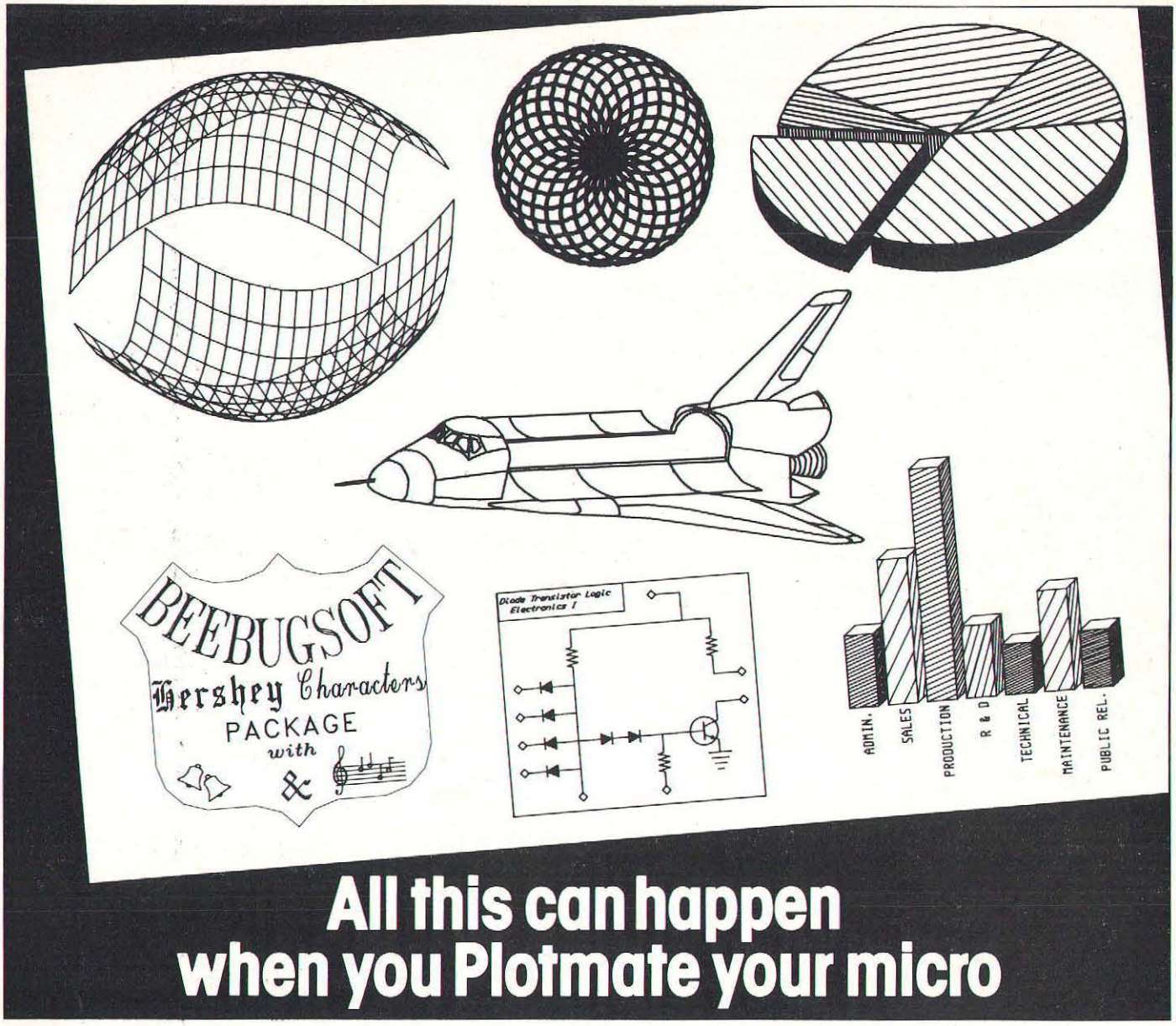
ALTRA TUBE EDITOR	_____	@ £39.00	_____
E.D.I.	_____	@ £39.00	_____
PROBE1	_____	@ £35.00	_____
PROBE2	_____	@ £39.00	_____
F/AID1.1	_____	@ £15.00	_____
BASIC-ED	_____	@ £25.00	_____
BASIC-ED2	_____	@ £29.00	_____
I enclose the sum of		*TOTAL*	£ _____

Cheque, P.O. or Credit Card Number

☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐

DEALER ENQUIRIES WELCOME

ALL PRICES INCLUDE VAT & DELIVERY



All this can happen when you Plotmate your micro

Plotmate is new. Excitingly new.

It's a rugged, responsive flatbed plotter with a lot of built-in magic.

You can use it to create fantastic patterns. Some small as a postage stamp. Or big as an A2 poster.

It'll produce overhead transparencies for you. And all dimensions of graphs, diagrams and charts.

It'll also prepare maps. Then put the contours in.

And that's only part of it.

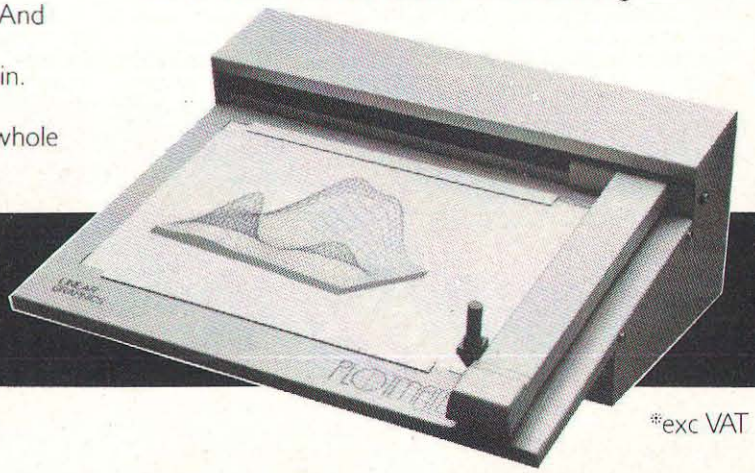
You should see its repertoire of characters – a whole sci-fi population for you to manipulate.

Plotmate software accepts all the colour, vector, character and graphics commands of your BBC micro.

You just plug it in and away you go. All for less than £300.*

Phone 0702 541664 for details. Get Plotmating soon.

PLOTMATE



LINEAR GRAPHICS LIMITED
28 Purdeys Way Purdeys Industrial Estate Rochford Essex
Telephone: 0702 541664 Telex: 995701 Ref 356

*exc VAT

A&B

Volume Three Number

Editor: Mark Webb
Assistant Editor: Fiona J.Eldridge
Group Editor: Dave Bradshaw
Group Managing Editor: Wendy J.Palmer
Advertisement Manager: Jonathan McGarry
Editorial Director: Ron Moulton
Managing Director: Peter Welham

REGULARS

News 6

Delta Card, Super Art, Communications, Torch, Electron Tube and RS232. Superior's Citadel, Repton II and Speech and the Music Maestro.

Soft Sector 22

Scarves on and rattles out. It's Matchday, Ocean's stunning football game for the BBC Micro.

Competition 24

The Wizard from Bored of the Rings sets the challenge.

Down to Business 64

Jon Vogler meets UNIX and comes out on top. The Torch Unicorn proves itself an outstanding computer system for the businessman.

Software Reviews 75

Adventure: Valley of the Lost Souls. Arcade: Video's Revenge. Utility: Replay.

Soft Sale 78

A&B tapes and discs: a massive games compilation at modest cost plus all this year's favourites.

Feedback 81

A&B's communication lines are open for another profitable exchange of views.

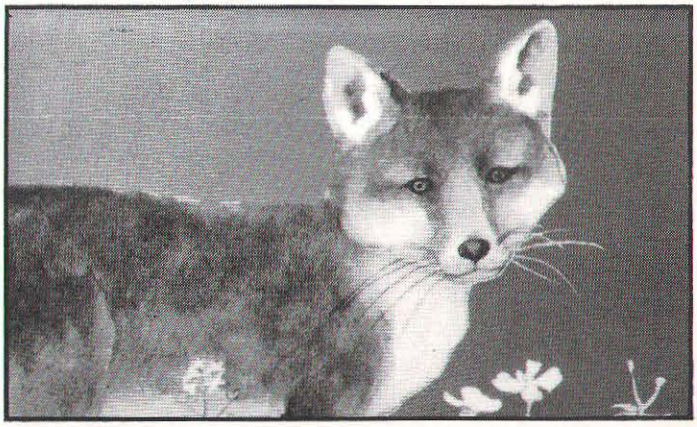
Competition 99

Win a Slogger ROM box/printer interface for your Electron, and some firmware to go in it!

FEATURES

Beginner's ROM 16

All you ever wanted to know about ROM software but didn't dare ask. And a program template to get you started.



Top of the Class 34

Des Thomas takes a trip to the race track in Fast Cars — Maths in Motion and into the sometimes dangerous world of Sly the Suburban Fox.

Articles in the next month column are in an advanced state of preparation but cannot be guaranteed to appear.

A&B Computing is constantly on the look-out for original and well-written articles and programs for publication. Feel free to submit your work to us for consideration for publication.

All submitted material must be in machine readable form. This applies both to programs (in any language) and to documentation, which should be prepared with a BBC or Electron wordprocessor. 5¼ inch disc (40/80) or cassette equally acceptable. Please also include hardcopy and any suitable illustration, photographs and/or screen dumps.

If you are considering submitting material to A&B then please send a S.A.E. for a comprehensive style sheet. It's also sensible to give us a ring before going ahead with any major work.

All submissions will be acknowledged and material returned if not required. On acceptance the copyright in such works which will pass to Argus Specialist Publications Limited will be paid for at competitive rates. All work for consideration should be sent to the Editor at the above address.

Twelve December 1985

Published by Argus Specialist Publications Ltd., Number One, Golden Square, London W1R 3AB. Tel: 01 437 0626.

All work for consideration should be sent to the Editor of A&B Computing at our Golden Square address.

View from the Top 53

How one London school took on the challenge laid down by the advent of microcomputers.

Plotting Success 84

Gordon Taylor turns his hand to getting the best of business graphics out of Plotmate and Interchart.

Cheaper RAM 90

16K Sideways RAM for a tenner? Give this profitable project a try.

Printer Buffer ROM 92

Ian Copestake implements a printer buffer for the BBC Model B+. Acorn didn't supply one, so we have.

REVIEWS

Memory Module 30

A 64K battery backed RAM extension for your BBC.

Rom Report 50

Dave Reeder on a handy half dozen utilities from Altra and Chalice plus A'n'F's Slave II.

Doubling Up 28

Tony Self with the latest UDM double density filing system, the Acorn 1770 upgrade for Model B and a mysteriously fast benchmark. What could it be?

Bookshelf 89

Practical hardware projects and how to get your spreadsheet to perform.

Sleuth 110

Trevor Attewell dons his Deerstalker and draws on his pipe in this investigation of the Beebug utility.

SOFTWARE

Library 26

A bunch of BASIC PROCs and FNs which are guaranteed to come in handy.

Flood Tide 44

Fast flood fill, techniques and examples.

Scruge 56

A Christmas tale, of Scruge, of gold, of strange spirits. Guide Scruge around the caverns in search of seasonal treasure.

Fortune Teller 100

Horrorscope for forecasters of future events.

Ultracalc Printer Driver 105

It needs one. We supply one.

Falcon Pilot 107

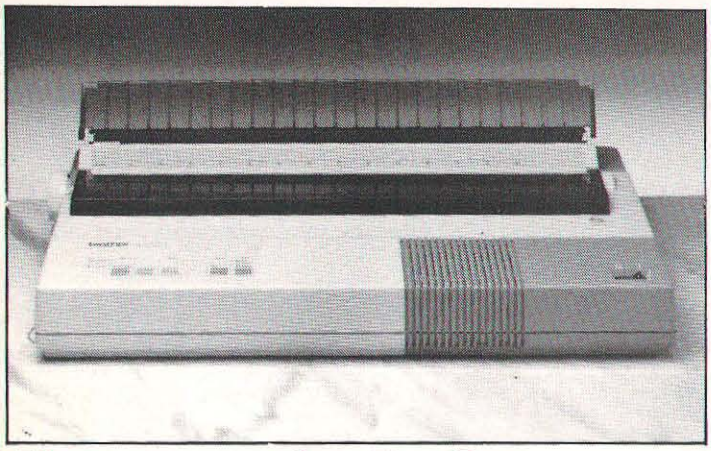
Flying into the jaws of death.

A&B Computing is published monthly on the first Friday of the month preceding cover date. Distributed by SM Distribution Ltd., 16-18 Trinity Gardens, London SW9 8DX. Telephone: 01 274 8611. Printed in the UK by Garnett Print, Rotherham and London.

The contents of this publication including all articles, designs, plans, drawings and programs and all copyright and other intellectual property rights therein belong to Argus Specialist Publications Ltd. All rights conferred by the Law of Copyright and other intellectual property rights and by virtue of international copyright conventions are specifically reserved to Argus Specialist Publications Ltd. Any reproduction requires the prior written consent of Argus Specialist Publications Ltd.

Copyright Argus Specialist Publications Ltd. 1985
Printed in the UK by Garnett Print, Rotherham and London.

News Ne



New generation Brother

Brother, the Japanese company which has arrived quietlybut forcefully on the UK peripherals scene, has launched a 'new generation' dot matrix printer.

The 1509 is a wide carriage (132 character width) printer and is primarily suited to the BBC businessman, although any user could be proud of its capabilities and could benefit from its new features.

The design is notably different from the current dot matrix and its streamlined looks are complemented by a noise level under 50 decibels. At 180 characters per second, the sound is more a light 'buzzing' than the hammering you may be used to. The unit is also very light.

Near Letter Quality is available at 45 cps and is selected by pushing a touch sensitive button on the front panel. A font board is easily accessible for the insertion of EPROMs containing additional typestyles, Gothic, Quadro and Anelia proportional for instance, or alternatively for adding RAM into which custom fonts can be downloaded. Ideal for specialist applications, science and languages, this RAM can also be configured as extra buffer space if required. Clever.

Centronics and RS232 interfaces are standard and the 1509 understands both Epson and IBM control codes.

The 1509 is likely to be the first in a range of new generation printers from Brother. A cut down version of the 1509, possibly 110 character width, is possible. Another outstanding feature is the paper handling. Moving from continuous stationery on a tractor feed to single sheets is a trouble free operation. Plastic supports can be fitted to the back of the printer to control the paper feed, which is handled automatically at the push of a button. An impressive debut for the 1509 at £495. Let's hope that a cut-down version cuts the price tag a little to include a few more serious BBC owners/printer users.

Details about your nearest Brother dealer from Brother Computer Peripherals Division, Shepley Street, Audenshaw, Manchester, M34 5JD. Tel: 061 330 6531.

Le BBC Microcomputer?

There seems to be some confusion over the recent speculation about Acorn and Olivetti trying to strike a deal with Thomson of France.

A licensing deal allowing Thomson to manufacture the BBC Micro would follow similar deals negotiated by Acorn with a Mexican firm and in India.

Olivetti already have a considerable presence in France and Thomson are the biggest French electronics company, themselves about to launch peripherals onto the UK market. France is embarked on an ambitious program of introducing Information Technology into both schools and society as a whole.

Also reported is a possible agreement on a European standard for 32bit computers for further/higher education. Acorn have apparently hopped over the 16bit and would no doubt like to push the BBC plus 32016 and Acorn Cambridge Workstation into European education markets.

Pastiche programs

A new name in educational software emerges in Pastiche Software. Their main product is Grafitti, a combined graphics tablet and software package. Also available is Sums, an arithmetic tester for children incorporating sound, graphics and skill levels. Guitar Tutor offers a tune up facility, a choice of 84 chords, the sounds for a chord and a chance to play along with the computer. Decimal is a question and answer program with four levels of difficulty: recognition of decimal fractions less than one, between 0 and 2, between 0 and 3 and simple addition of decimal fractions less than one.

Information from Pastiche Software, 62 Woodhall Road, Calverley, Pudsey, LS28 5PP.

Compas CAD

Compas means Computer Aided Design for the BBC Micro. A full suite of programs is available for implementing CAD and other subjects including Advanced Manufacturing Technology in the classroom. The initial Compas starter comprises Compas 2D, Compas Advanced and Compas Professional.

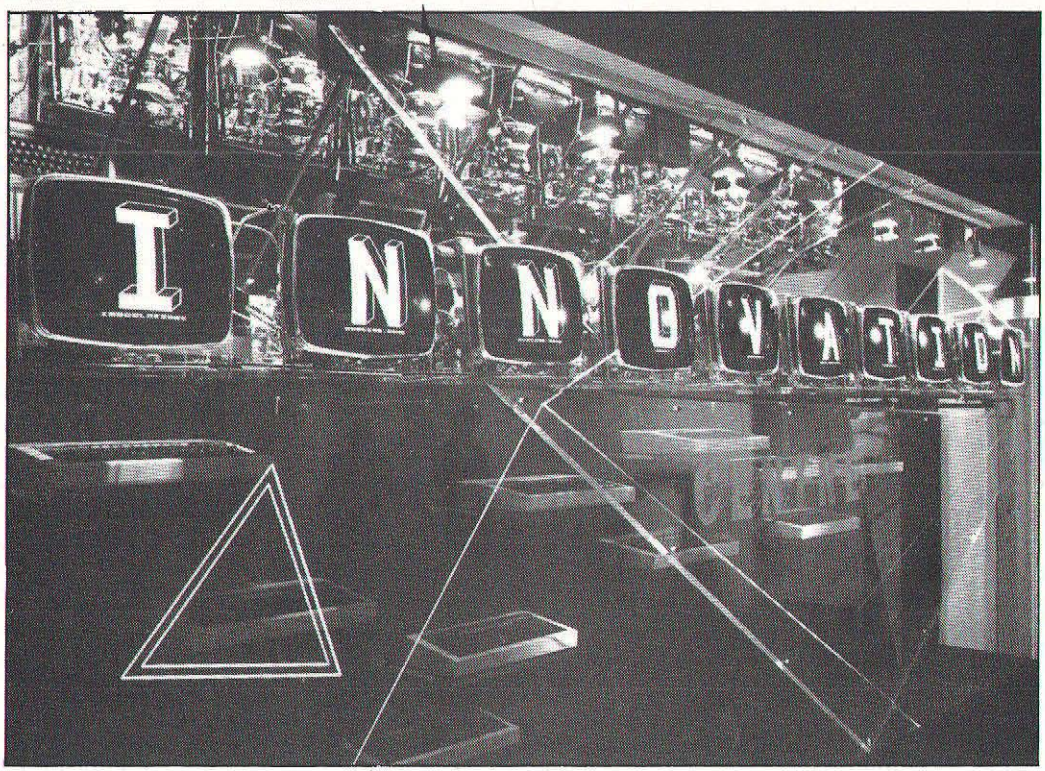
The software is supplied with a mouse which is used for both drawing input and menu selection. the software can be used with Epson compatible printers and a variety of low cost plotters including Graphtec and Penman. The software starts at £390.00.

For the classroom there is a set of workcards and on-screen demonstrations. Teachers notes provide information about the system, ideas for course work and detailed pupil assignments.

Compas starter will supply files to be integrated into Compas Projections and CNC for 3D viewing and CNC machining. Projections gives thickness or height to 2D drawings, which can then be viewed in a variety of standard projections (including Perspective, Isometric, Axonometric etc.) from any angle or eye level as specified by the user.

CNC automatically generates G-codes to drive a CNC lathe. The software also displays tool selection and the machining cycle.

Full details from British Thornton Limited, PO Box 3, Wythenshawe, Manchester M22 4SS. 061 998 1311.



Natty design

Innovation is not unusual down at the Design Centre of London's Haymarket but what about these BBCs? Created by One Point Five, the display is in fact a sign, using conventional BBC Computers in, we think you'll agree, a novel and interesting way.

The sign is a permanent feature of the Innovation Centre and well worth a look when visiting town. It consists of 10 monitors controlled by 10 BBC micros, seen through a perspex screen. A sequence of words beginning with one of the INNOVATION letters changes colour on each screen every 20 seconds.

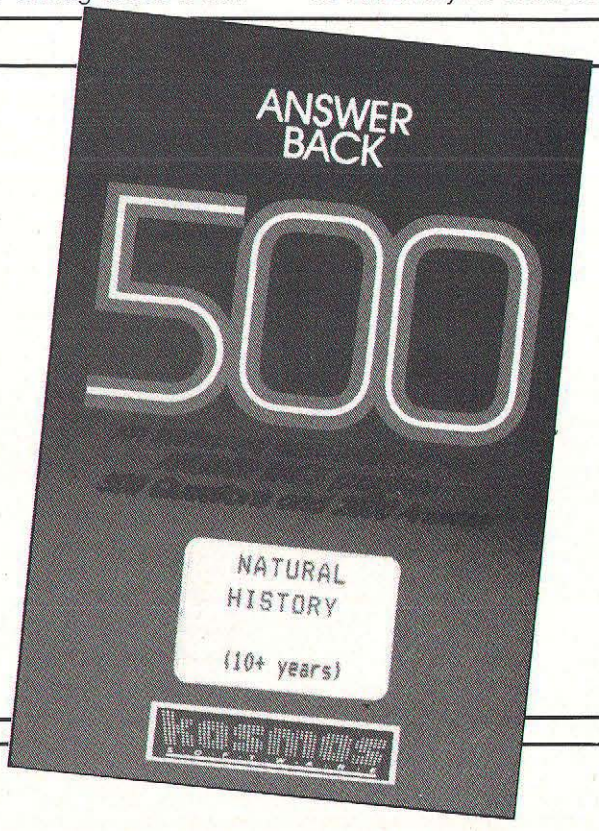
Factfile 500

Kosmos are launching the first nine titles in a vast new range of leisure and educational software modules for the Electron and BBC. This is the Factfile 500 series and twenty will be released before Christmas.

The Factfile 500 packs cost £3.95 each and are used with any of the popular Kosmos Answer Back Quiz master programs. Each contains 500 questions and 2000 multiple choice answers.

The subjects covered are Arithmetic, Spelling, Natural History, General Science, Junior General Knowledge, Senior General Knowledge, Know Scotland, First Aid and Sport.

Details from Kosmos on 05255 3942.



InteresTed in IT?

The BBC, in association with City and Guilds, have launched a new series of programs and course in Information Technology. Inside Information is designed for newcomers to the subject and covers the use of computers in all walks of life.

The self learning package provides a book, computer software, audio cassettes and assessment and certification by City and Guilds. The broadcast series, from which the audio cassettes are edited down, has been on the air since October 15th and continues on Sundays at 4.30pm until December 15th. Presented by Paul Heiney and produced by Chris Stone, the series is a lively and humourous guide to the current uses to which we are putting our computers.

The software ingeniously emulates how a computer works as well as database and wordprocessor operation. An all encompassing book by Jacquetta McGarry is a well illustrated and written introduction to the subject and closely complements the series.

When the student is ready he or she sits the City and Guilds examination, also by computer — the first computer assessment of its kind in this country.

The Inside Information course and package was introduced just as A&B went to press for this issue so for full details, how to take part, where to go for information and a full review of the package, get hold of next month's A&B Computing.

CONTINUED OVER

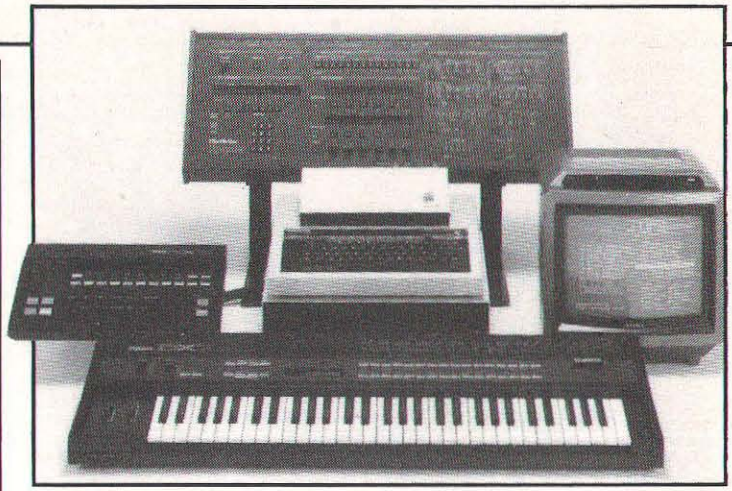
News Ne

September software competition

We were overwhelmed by the response the Software Competition in September's *A&B*. Not only did it give us a valuable guide to the hardware and software that our readers own but it also showed tremendous support for the *A&B* Software Sale: you thought that the prizes were definitely worth winning!

Every month for a whole year, all the software on offer in the *A&B* Software Sale will be sent to the following three lucky winners: *J R Savage* of Kings Lynn, *A Hardy* of Droitwich and *John Stonier* of Welwyn Garden City.

The first 50 entries we received each qualify for a free *A&B* cassette: J A Green, Leigh; Paul Stewart, Redhill; Keith Newham, West Wickham; Darren Draper, Stevenage; Peter Jennings, St Albans; Nancy H Aberle, South Nutfield; R W Rose, Marlborough; M C Daly, Harrogate; S Prendergast, St Albans; E R C Holland, Ashford; J Collins, Castleford; S Teal, Shipley; William Sinclair, Orkney; J Burgess, Oxford; Robert Busschots, The Netherlands; Colin Read, Upminster; Stuart Howard, Bolton; Simon Moss, Preston; J F Halam, London E17; Jeff Standard, Luton; Gareth Sullivan, South Glamorgan; C B & S E Pointeer, Reading; D Low, Glasgow; James T Sullivan, Edinburgh; S Andrews, Croydon; J Robinson, Dukinfield; A B Reader, Godalming; B Gilbert, Bury St Edmunds; M Carter, Donnington; J G Clemas, Worcester; S Willis, Swansea; Pauline Harris, Chelmsford; J M C Barber, Malvern; John Goubin, Belgium; Adam Slatter, St Albans; S Prendergast, Ivybridge; G Ursell, Wickford; Philip Miller, Sheerness; Alex Daly, Glasgow; J Simpson, Shoreham-by-Sea; C E Soan, Bath; W Kermode, Liverpool; B W Truscott, Bristol; B Macdonald, Pangbourne; David S Ferguson, Glasgow; D Kerr, Swansea; A A Shalabi, Hounslow; A G Bowers, Stroud; A Stefanuti, Bradford; William Kwok, Bristol.



Soft music

Electromusic Research has improved on its range of MIDI software for the BBC Micro and EMR MIDI interface.

***Composer* features the 'dynamic' use of memory to optimise the number of MIDI 'events' available across six possible tracks. The interface also allows for the control of some non-MIDI Roland/Korg rhythm units. Notes are entered step by step from the BBC keyboard or are 'recorded' from a MIDI keyboard instrument. Compositions can then be edited in fine detail and sent to cassette or disc for permanent storage.**

***Notator* takes MIDI tracks and prints them out on a dot matrix printer in standard musical notation — for those humans left in the band! *Editor* provides enhanced single step editing of tracks and of MIDI 'voices'. *Toolkit* is a library of useful extensions for EMR MIDI users and *MIDIBANK* a file handling system for MIDI sound and sequence libraries. *Vumusic* is a graphics package which responds to the MIDI music being played with a 'graphic interpretation'. Designed for live performance and the use of large projection screen.**

Details from Electromusic Research, 14 Mount Close, Wickford, Essex, SS11 8HG.

Enlarging Electronically

Peripherals for the Electron are now emerging through the support of companies *Advanced Computer Products* and *PACE Supplies*.

Advanced Computer Products have already demonstrated the massive speed increase provided for the Electron in *all* Modes by the addition of a 6502 second processor. The interface plugs into the back slot of a Plus One and operating software is supplied in a cartridge in the other. Graphics routines and serious applications run in a third of their normal time and of course ROM format software can be run from disc in the second processor.

PACE have revealed their RS232 interface for the Electron. It's housed in the standard black plastic moulding which plugs into the Plus One. The software will be Commstar with a Mode 7 emulation. PACE will probably offer a value added package including interface, software and Nightingale modem, one of the most popular combinations currently in use.

The interface opens up the world of Prestel, Micronet, Microlink, local authority information, bulletin boards (Micro Live, Gnome at Home and other private boards), and company databases (Acorn, Technomatic, BBC Publications). You can even order by credit card and download free or purchased software, enter competitions and play multi-user games.

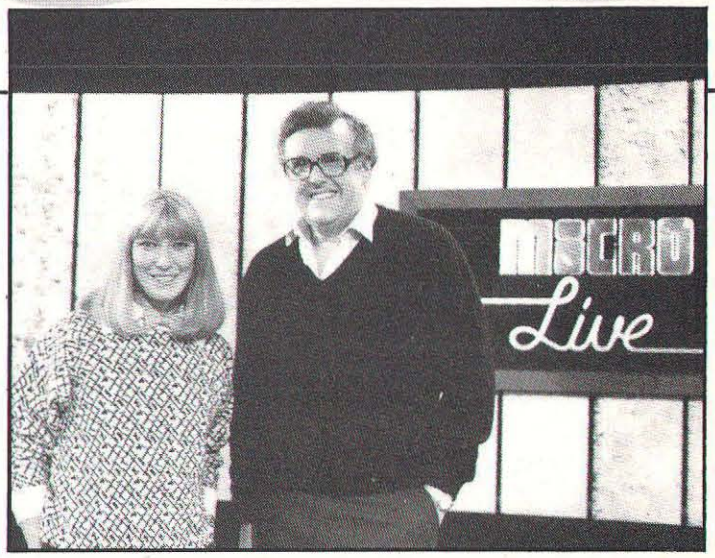
The message seems to be that Acorn cannot support the machine themselves but that they are doing the decent thing by releasing designs and technical information to the companies prepared to service the growing numbers of users who have had the good sense to buy an Electron.

ws News

Micro Live returns

Micro Live has returned to our television screens for a new weekly series. The program sets out to cover as broad a spectrum of computer use as you could imagine, from choosing a painting in an art shop to running the world's most advanced University.

It will also deal with the applications nearer to home in business, industry and as a hobby. There will be features on artificial intelligence and on the realities of computing in business. Presenters are the great Ian McNaught-Davis, Lesley Judd and Fred Harris. Freff rejoins the team from the USA. Micro Live goes out at 7.00pm Fridays on BBC 2 and is repeated at 5.30pm Mondays on BBC 2.



Owl on the air

BBC Publications have now 'gone live' with their brand new Viewdata bulletin board OWLTEL. Anyone who has a Prestel type modem can ring 01 927 5820 and browse through a thoroughly entertaining board run by software manager David Atherton.

The software is the very popular Communitel package providing information pages and a response page, where you can leave messages, suggestions and, naturally, your order and credit card number.

The board currently displays the range of BBC Publications' products, games software, educational packages and books. Details of educational software are right up to date with the Microtechnology pack, Geography and future release Deutsche Direkt, for use in conjunction with the schools television program. No details on books but titles include more in the 'mastering' series, on assembler and discs plus an interesting book on the BBC Micro in small business. You know where you will see reviews first so I'll keep quiet.

There's a set of information screens on Viewdata and a tutorial on how to use Owltel. Further entertainment can be had from Owltel with news items and features. Current news includes the latest on an Ultracalc utilities disc which includes graphic presentation and the good news that it will be *free* to Ultracalc users. Ring BBC Publications now to reserve your copy!

Some tempting sounding features are planned and the two currently on the board are very useful and, in the case of "Who's who at BBC Publications", amusing. The information on licensing of software for multi-user sites makes fascinating reading in the light of the 128K micro. Its sideways RAM 'sockets' are very valuable on a network or large group of stand alone micros using licensed ROM software.

OWLTEL is a welcome addition to the growing number of bulletin boards for BBC owners. Give it a try.

In brief

Logotron's Advanced Logo is to have Control Logo packaged with it on one floppy disc for the price of £15 + VAT (including manual).

Advanced Logo is a command extension to Logo developed alongside educationalists. Control Logo supplies a set of control oriented commands. It has developed out of the Walsall Logo Project. Both extensions are a standard part of the Logotron Sprite Logo system (see the offer in *A&B Computing* October).
Information from Logotron on 01 352 1088.

Single-side drive owners will be aware of the possibility of using both sides of a single floppy by cutting a notch in both sides of their discs. The Disk Nibbler has been designed for them and is now available with a £2 discount and a free disc from Computatill Limited *Give them a ring on 061 652 8006.*

Great Exhibition 2001! What will you be doing in 2001? A gentleman called Evan Steadman will, he hopes, be watching his dream come true in the form of a Great Exhibition.

Mr Steadman is a successful exhibition organiser and recently launched the build up to the 2001 show, a show he hopes will parallel the Great Exhibition of 1851, 150 years earlier. Mr Steadman has set about his task with vigour, imagination and a sense of national pride. The show, he believes, will reflect the emerging industries of information technology and electronic engineering.

The concept is ambitious and vast amounts of money and high level support will have to be won over. If it happens, perhaps the BBC Model Z will be there, on the stand of Olivetti, the Italian subsidiary of Amstrad International...

Torch have added a 1Mb option to their TX professional range. The TX20 runs UNIX, CPN and BBC MOS in the similar manner to the Torch Graduate.

Further UNIX software is becoming available for the Graduate, including Informix, the multi-user relational database, Ultracalc spreadsheet, Uniplex II integrated wordprocessor and high level language Prolog. Watch out for a Micro Prolog on the BBC Micro soon.

Torch have also upgraded their ZEP100 pack with the addition of he Plus100 pack, a ROM based CP/M compatible operating system designed for faster program running and disc access. There's also a comprehensive new user guide.

CONTINUED OVER

News Ne

There are now over 15,000 ZEP100s in use with BBC micros, largely in education and research, where computers are networked with Torchnet. Major users inlcude Cumbria Educational Authority, who have recently istalled five major networks totaling 100 stations and Hereford and Worcester Educational Authority, who have standardised on Torchnet.

Datalines offer a full range of modems for business use.
For a free catalogue, call Julia on 0908 311077.

Panasonic are now producing floppy discs in all sizes to complement their drive mechanisms. Also new to the market are a number of monitors, LCD screens and printer mechanisms.

Software Limited are supporting Dbase II, III and Lotus 1,2,3 owners with a range of utility software.

Salford University are currently into a series of 20 weekly seminars on information with speakers from leading companies and organisations such as Hewlett Packard, Britiish Aerospace, the National Computing Centre, Marconi and British Telecom. Although part of the MSc in Computer Science course, they are open to the public. Well done Salford.

The seminars are designed as introductory modules to subjects such as office automation, data protection, speech processing technology, artificial intelligence in maufacturing and CAD and CAM. They take place on Thursdays during term time from 4.40pm to 6.30pm and the fee for individual seminars is £2.00. A transferable season ticket costs £25.00 and concessions are available. Supporting course notes and refreshments are included in the fee. Tremendous value when you compare the cost of commercial get togethers of this kind. Don't miss it if you are in the Salford area.
Information from John Sweeney, Dept of Mathematics and Computer Science, University of Salford, M5 4WT. 061 736 5843.

Swadelands School in Lenham, Maidstone, have collected £1,500 worth of software from UK companies in response to a plea from the islanders of Niue in the South Pacific, who felt that their children were missing out on computer studies.

The students used the A&B software listing to choose software and companies to approach. The software houses listed therein reacted magnificently, donating 'a mountain of software'.

Copies of A&B magazine and software are also winging their way to Niue, which, Victor Young of Swadelands tells us, means (roughly translated): 'Hey look fellers! Coconuts!'.

An amazing range of exhibitors are already lined up for the Barbican based High Technology and Computers in Education Exhibition which takes place on between the 22nd and 25th of January 1986.

Over 200 companies in all are expected with over 3000 products and the show is being actively promoted abroad by the British Council.

Deva Electronic Controls have produced a low cost 6502/65C02 Emulator. 'DEVICE' can work in conjunction with a BBC Micro to allow the development engineer to run a target system at full speed whilst maintaining total control over the processor's action. It can be driven from any computer with RS232/423 and BBC software can be supplied.
Details from Paul Shuffleton on 051 356 3767.

Two new and interesting printers from Micro Peripherals. The Juki Daisywheel 6200 has a 3K buffer, 32 cps operation, diablo ribbon and protocols, up to 197 characters per line and quiet performancce. Price: £499+VAT.

The Daisy Junior is an amazing £179. It has 137 columns, 7K memory, friction or pin feed and prints at 14 cps.
Information from Ian Donaldson on 1256 473232.

Sandy Dow, ex Acorn and R H Electronics (which he set up himself), has joined Mushroom Electronics, the Bedford based BBC Micro specialists. Paul Vaughan said "The appointment of a director with Sandy's experience illustrates our determination to expand in both the hobby and business computer areas."

Xyllyx have added a new set of communications packages to their product range.

The BBC pack costs £99.00 and includes 1200/75 modem, software and connecting cables.
Details from them on 01 377 1982.

Honeyfold have released a full three dimensional graphics development system that allows the user to create, display, modify and store three dimensional graphics images. These can then be included in the user's own software, either from BASIC or machine code.

Noise on the line

Just as British Telecom announce major price changes in an upwards direction, we have so many more reasons for plugging in our modems for late night stints on the bulletin boards and public databases.

Prestel Education have announced the launch of their full telesoftware service. Some programs available through the service offer 50% discounts on normal publisher's prices. Further software packages written by local education authorities and schools are free. Some are published directly by Prestel Education.

Software can be downloaded immediately and has been thoroughly tested in schools. As well as software there is a magazine section and technical help desk via Prestel mailbox. The telesoftware complements School Link, the educational magazine, the Signpost careers guide, the ECCTIS courses guide and ED.IT, a current awareness service for all

ws News

teachers involved with Information Technology. Why not dial up now?

British Telecom have taken a majority share holding in Farmlink Services, operators of the Prestel Farmlink farm management service. This is the Prestel videotex database for farmers, including commercial information and a wide range of specialist software.

Tubelink is widening its coverage to take advantage of what is now the most popular area for BBC Micro enthusiasts, especially those with second processors.

Prestel's electronic mailbox facility is available to anyone with a problem related to the BBC. The ROM compatibility list is very useful and there are three subsections, Ramlink, Mouselink and Pascalink. Mouselink recently featured *A&B's* Edge Detection program in its software list.

Worldwide Telex is now available from within Prestel. It's just like sending any other message on Prestel. Telex Link automatically transmits messages to the telex number given by the user, and keeps the user informed of progress by sending messages back through the Prestel Mailbox facility. The user is charged only when a successful delivery has been made.

Incoming telexes appear in users' mailboxes. Tariffs range from £0.50 in the UK to £2.00 for North America and £5.00 Maritime.

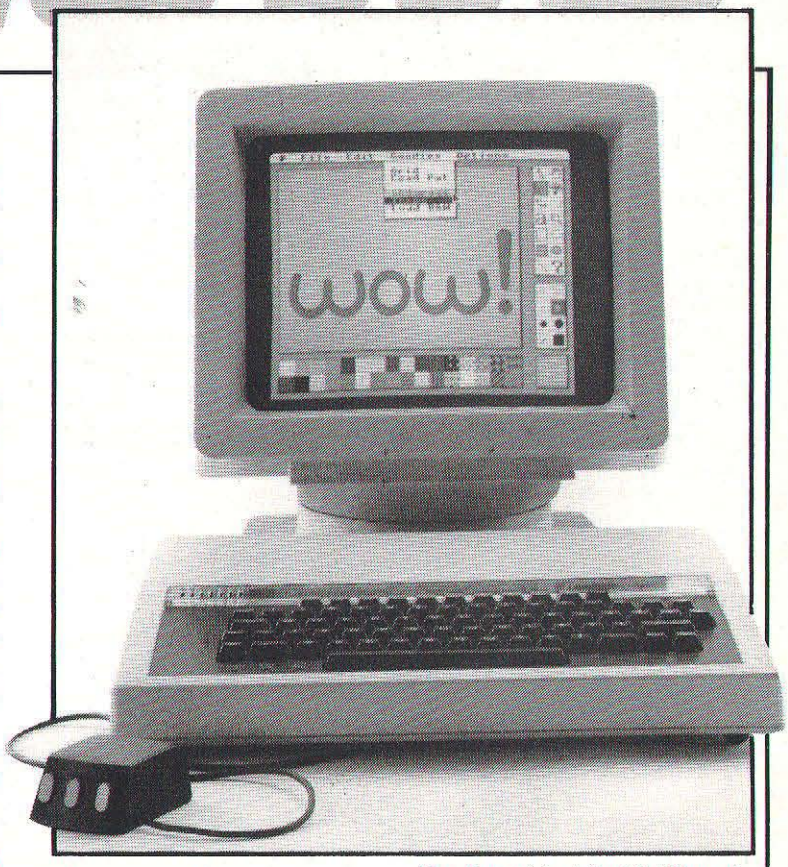
Computer News has now joined the Jobsearch area of Micronet. These computing industry, accounting and secretarial jobs are advertised on Micronet and Prestel. Further enhancements will include a special school leavers spot.

A.I. comes to Micronet with the Artificial Intelligence Quotient. *8006055 gets you an introduction to Micro-Prolog. There will be an A.I. forum and lists of books and software of interest to inquisitive Micronetters and in future, a series of BASIC programs illustrating A.I. principles.

Top Ten modems subscribing to Micronet in August 1985 is headed by old favourite VTX5000 (231) with Pace Nightingale, Minor Miracles WS2000, Tandata, Watford and Demon also featuring. An indication of how BBCs are dominating the comms scene.

William Powell, the sponsor of the Private Members Bill on software piracy, is due to appear on October 29 between 7.00pm and 8.30pm on *Celebrity Chatline.*

SWAFAX is a new viewdatabase for you to try on 0622 850440. It's the work of kids aged 11 to 13 from Swadelands school in Kent. Anyone of school age is invited to send contributions to the database. The school is especially interested in satellite communications and there's software for you to download in order to recieve screen images of weather maps. There is also regular news about the progress being made by Ashford School as they prepare their experiments for the Space Shuttle in 1986. Companies are invited to get in touch if they wish to set up a specialist database through SWAFAX.



New software diet for AMX Mouse

More goodies for the AMX Mouse soon to appear for Christmas include a 16K EPROM *Super Art.*

AMX Super Art, pictured here, works in Mode 1 and offers a full range of shading as well as the familiar mouse drawing facilities, which 10,000 users now enjoy. Super Art is designed to operate in A4 size screens in order to standardise artwork printouts for uses beyond the computer.

Also new for Christmas are AMX Pagemaker, 3D Zicon, Database and Xam. Pagemaker is billed as the great contender to Mirrorsoft's Fleet Street Editor. Nick Pearson of AMS has been heard to say that he relishes taking on the mighty Maxwell House! Pagemaker is also A4 standard size, allowing the blending of text (in 16 different fonts) and pictures drawn using the mouse or perhaps brought in from video digitisers etc.

3D Zicon supplies a full 3D graphics drawing environment for Mouse. Database emphasises user friendliness and full random access capability (disc users only therefore). A nice feature in the £24.95 package is the fact that the screen actually looks like a card index!

AMX XAM (say it out loud to get the full benefit!) is an interesting looking idea. The program displays pictures and text, prompting for answers and supplying a multiple choice format on screen. AMX Super Art can presumably be used to create pictures for use in XAM and no doubt it marks the answers for you as well. XAM is going to revolutionise a few classrooms around the country by the looks of things.

CONTINUED OVER

News Ne

All the hits and more

A musical instrument learning system which is taking schools by storm is to be released in the form of four Micro Maestro packs for public consumption this Christmas.

Each of the four packs will concentrate on a specific group of instruments and contains Mupados software to run on the BBC Micro or Electron, an audio backing cassette and an instruction booklet. The packs will be available through music shops, HMV, Virgin shops, Boots and Smiths.

The Micro Maestro range is a unique combination of software and audio backing track. Following the completion of an agreement with Warner Brother Publishing, Micro Maestro provides the backing for top tunes, Ghostbusters, Superman, Chariots of Fire, Hello (Lionel Ritchie), Merry Christmas (War is Over) and, rumour has it, a Madonna song. The backing tracks have been laid down by professional musicians in a 16 track studio at Mupados in Wales, where the 'land of song meets new technology' and the music has been broadcast on the Welsh fourth channel.

Mupados stands for Music Play And Display Operating Systems. The software, which has proved an overwhelming success with teachers and pupils, displays the musical scores on screen, plays the music simultaneously, gives a continuous guide to the rhythm, changes tempo, selects bars of the music and repeats selected bars.

When the player has mastered the tune with the help of the micro, he or she can go on to solo with the audio backing. The four packs cater for keyboard instruments, concert pitch instruments, recorder, guitar, violin, flute, oboe, cello, trombone, double bass and viola, and B flat instruments such as trumpet, clarinet, euphonium etc. The fourth is a slightly cut down version of the schools recorder tutor.

In strict educational terms there is a small revolution occuring in the teaching of music with a new GCE/CSE syllabus geared more to the *playing* of music. Mupados neatly fits in with this philosophy and allows anyone to learn a musical instrument in an entertaining and contemporary manner.

The Micro Maestro packs cost £14.99. If you need further information then call Mupados on 0570 422877.

In brief

Go dutch? The Dutch are perhaps the most enthusiastic supporters of the BBC Micro on the continent. This is reflected by the shows and clubs which they organise. One such is the Hobby Computer Club. The Venlo section organises their fifth international Computer Fair on February 22 1986.

It takes place in the conference centre "De Maaspoort" in Venlo and there will be exhibitions by dozens of hobbyists and user groups (a prize is offered for the best application). Also represented are companies selling peripherals, books and software, regional schools. Then there's a non-stop film show, lectures, refreshments and a creche with qualified baby sitter.

Information about the show from J Peeters, Molenstraat, 5993 ZB Maasbree, The Netherlands. Tel: 4765 1693.

Tandata Marketing has launched a new triple standard modem, incorporating the three most commonly used communications speeds in one box.

The Tm 602 combines V22 (1200/1200 bps full duplex), V21 (300/300 bps) and V23 (1200/75, 75/1200 and 1200/1200 half duplex). Complex duplex! Autodial and auto answer are standard plus auto-log-on to almost any database. The Tm602 is BABT approved and offers good value for a high performance at £465+VAT.

Details from Tandata Marketing, Albert Road North, Malvern, Worcs. WR14 2TL. Tel: 06845 68421.

Robocom , known for their Bitstik graphics software, have implemented the system on the IBM PC. I wonder if it works on a Torch Graduate?

Namealikes Robcom are marketing an head alignment kit for the BBC (though no mention of the Electron). It's cheap at £4.95 for alignment tape, cleaning tape and screwdriver. It is not software controlled (as with the Global Operation Caretaker) but relies on an audio output judged to be correct by the user.

Christmas time brings shows, shows, and more shows to London. Following the demise of the organisers of the last Acorn User Show in August (despite its apparent success), Editionscheme have stepped in with a new show at the Central Hall, Westminster, on 22nd and 23rd of November. Over 50 companies will be exhibiting, including Chalice Software, Gemini, Griffin, Logotron, Modem House, Private Tutor, Direct Disk, Ramamp, Shards, Wigmore, Duckworth and — at last the games are back — Micropower.

Also on show will be the London Rock Shop's UMI-2B, demonstrated by their expert musicians and giving hands on experience to BBC music makers.

Details about UMI-2B from London Rock Shop, 26 Chalk Farm Road, London NW1. 01 267 7851.

Citizen printers are to be handled in this country by CPU Peripherals, alongside their Epson range. They are also responsible for a new range of ICI optical discs.

Details on 04862 23411.

Chipclub is a new club for schools and individuals who wish to make children "computer aware". Run by Alpha Computer School, resources include an electronics kit, a magazine designed for the 9-13 age group and many programs covering topics such as databases, barcodes, teletext, computer matching, measuring light intensity and the speed of moving objects.

Details from Alpha at Ebernoe House, 33 Chesswood Road, Worthing, Sussex, BN11 2AA. Tel Worthing 210701.

ws News

Micro video

New products from Video Electronics Limited are the VEL Electronic Palette and the VEL Beeb-Lock system. The Beeb-Lock can synchronise the RGB output of the BBC micro to a composite 625 line PAL video source and mix the computer picture with the video picture.

The BBC Micro is used in this way as a low cost caption and graphics generator as it is already in local television. Pictures from videodisc players can be mixed with captions and instructions generated by the micro to add a new dimension to interactive programming.

The Electronic Palette gives total control over the BBC computer graphics display, offering 4000 shades of colour. VEL are mainly a video technology company but are working steadily into the BBC Micro field. VEL are at Wigan Road, Atherton, Manchester M29 0RH.More information about their work on 0942 882332.

Big name games

The Christmas bonanza for games goes on unstilted in 1985. Mosaic Publishing have the Level 9 produced *Diary of Adrian Mole* for the family market while Superior Software have three stunners in Citadel, Repton II and Speech.

CITADEL

A 120 screen graphic arcade/adventure in the Castle Quest mode. Produced by a new and original band of designers/programmers from Denmark, Citadel contains a huge number and variety of puzzles and problem solving exercises involving the movement and carrying of objects.

The player wanders the castle, the grounds, the dungeons beneath and can swim in a lake (even splashing into the water!). An island location is accessible across an ocean, where a temple and a guardian monk await. Outerspace, a witches hut, Stonehenge, and a pyramid are some of the locations and chrystals and secret rooms all play a part in the brand new game, Citadel, available for BBC and Electron.

REPTON II

Players of Repton have long been asking Superior for a follow up. They have been working on the sequel for some time and here it is, a fairly sensational game in the style of Repton but with a whole lot more packed into the machine.

The game is bigger with 16 levels, each level consisting of a 32 x 32 matrix with 8 x 8 showing on screen at any one time. The graphics are big and bold and the boulders are especially stunning with colour mixing. The quirky Repton II character is made up of a number of sprites and he characteristically glances from side to side.

Throughout the game there are transporters which move you to different parts of the screen. Each transporter can only be used once. The graphic effect is similar to hyperspace in Elite. On screen zero a large space is reserved for the construction of the sentence *Repton II is ended*. These pieces are gathered on your way around the game and the puzzles are ingenious, some with a time limit althought there is no overall time limit as in Repton.

A status screen can be accessed from yellow diamond shapes on some screens. Other objects include eggs which hatch giant ants, spririts which are difficult to see but hold the key to the diamonds in the gates, a key and skulls which are fatal.

The spirits have to be shunted through the gates where they are killed and the diamond revealed. All the diamonds must be found to dislodge the skulls on screen zero, then all the transporters must be used, all the monsters killed and all the earth collected. Then all you have to do is dodge the meteor shower!

After much playing and testing no one has come near to finishing the game *yet*. But Superior are prepared for it to happen and are giving away T-shirts to the first 100 players to crack it. These 100 will then be placed into a £200 draw.

Repton II is available for BBC and Electron (£9.95 cassette) and on disc for the BBC (£11.95).

SPEECH

On the intro files to both the above games, Superior will be demonstrating a remarkable new form of speech production for the BBC, totally achieved in software.

Later this year the Speech cassette (£9.95) will appear. It enables straightforward English language conversions using a form of phoneme production. 7K of code means that the overheads are not too high for incorporation into existing or potential programs. The educational value could be enormous.

The pitch of sounds can be varied and certain syllables accented as appropriate to simulate spoken language. It is possible to add to the internal dictionary of the software.

Five support programs are supplied on the tape, a relocater, demonstration, file saver, Wordwise and View converter (speak your texts!), and an educational spelling game. Sounds sensational (literally!)

ADRIAN MOLE, AGED 13¾

Every other entertainment medium has portrayed Sue Townsend's teenage character and now, last but ont least, he is the subject of a computer game. BBC luminaries such as Bruce Smith can tell you his date of birth without hesitation.

Based on *The Diary of Adrian Mole Aged 13¾* and *The Growing Pains of Adrian Mole,* the game takes players into the same situations and asks them to cope. It presents events in Adrian's life and offers a choice of responses, covering a whole year in 3 month chunks.

The BBC version is, in common with other micro versions, a Level 9 production in association with Mosaic Publishing, combining pictures and text. The BBC version is last out because of the memory limitations of the machine. Cassette will cost £9.95, disc £12.95 and a Diary Kit £14.95 and £17.95.

CONTINUED OVER

News

Watford bonanza

It's very difficult to pull together the strands of what Watford Electronics are getting up to as they conclude 1985 with a flourish. A diversity of products emerge, from a £12 ROM to a £100 plus "turbo" for your BBC Model B.

FIRST THE SOFTWARE

Wordaid is the major new utility ROM, designed to complement Wordwise Plus with a variety of functions. The software's existence in ROM is largely a matter of convenience for the user. All functions are implemented from segment programs which are shuffled into segments 4 and 5 as required.

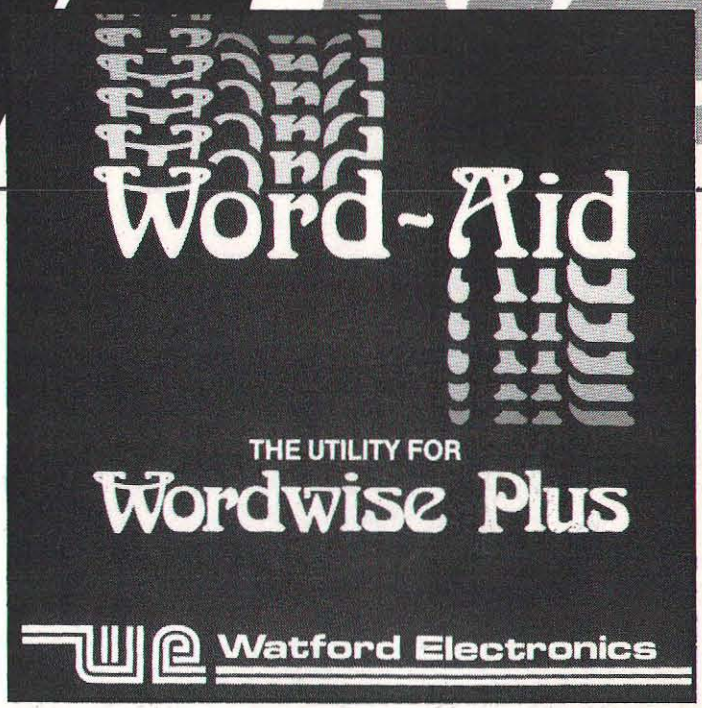
Wordaid is: an alphabetic sort (vital), free transfer of text to and from segments (useful), chapter marker, predefined printer code sequences on function keys, search and display, embedded command removal, multiple copies, and clear segment/text (all convenient but not difficult to program).

The address finder and mail merger could be useful BUT stand by for a review in A&B's Plustalk to see if it competes with Beebug's Wordease. The continual processing/file selection and print/preview operations set themselves up in competition with Paul Beverley's Continual Processing ROM. We shall see...Certainly £12+VAT sounds fair value. Further software now available includes the Diagnostic Disc — a chance to find out whether you bought a duff BBC! (Only joking!) It's a good example of the sort of software checks that an engineer regularly performs on micros and peripheral equipment and could prove a beneficial teaching tool.

Under development is a real time diary — the EXECUTIVE ROM — with which you can stamp your disc files for reference purposes, much as with the Cumana Electron interface. And those teachers who enjoy the task of putting together beginners courses in electronics/computer interfacing and control, will be glad to hear of a series of self-contained kits which Watford are about to launch.

ROM/RAM ENHANCEMENTS

A two pronged attack on the seemingly limitless market for sideways ROM/RAM now emerges from the High Street. Where do all these ROMs spring from. Our expert Dave Reeder did some research and came up with over 150 currently available for the BBC. ZAnd Watford plans to supply enough sockets to go round. The new emphasis however is on sideways RAM. The ROM/RAM board takes the standard four sockets up to eight. Alongside an additional eight banks of sideways RAM can be plugged into the board (8x16 = 128K). ROM format software is then loaded from disc as required. Eight banks carrying software can sit happily side by side. They can all be read by the OS but at any one time just one is selected for writing to.



In addition to the four ROM sockets and four banks of dynamic RAM, Watford have retained the popular CMOS RAM option. Two sockets, representing the upper and lower halves of a 16k bank can take two 8K EPROMs which act as a single 16K bank or two 8K CMOS RAMs (for which this combination was designed). The CMOS RAM can then be battery backed, can contain start-up procedures for your micro, a diary facility perhaps, change of Mode and so on, can be used to test ROM software (for which a read protect switch can be provided).

The board allows flexible entry level. With 32K RAM it costs £45, with 64K, £69, with 128K, £109. Users can always buy further RAM to upgrade their board to the 128K level. CMOS RAM, battery backup and write protect switch are extra but the complete setup amounts to £125, less than £1 per K.

Watford also provide the battery backup option on their "solderless" sideways ROM board, a straight no fuss upgrade to a 16 socket Model B for £35. Socket 14 of this board is designed to take a variety of EPROM types: 2716, 2732, 2764, 27128, 6116, 6264.

MODEL B+ DFS

Quite rightly, Watford have also turned their attentions on the Model B+, bringing out a version of their 1770 based DDFS (Double Density Filing System). The 1770 is capable of double density but the Acorn DFS doesn't exploit it. That's left to the ADFS, which, though displaying superior features, does produce a format incompatible with the DFS.

Watford's DDFS successfully and automatically handles single and double density filing, reading and writing, has a well developed 8271 emulation, offers built-in utilities (sector editor formatter) and a large catalogue (62 file) facility. It's a strong contender as a Double Density upgrade to the Model B+, especially for those who find the DFS a suitable filing system and are unwilling to relearn with the ADFS. The filing system ROM and manual cost £35+VAT.

WS N

DELTA CARD "TURBO" FOR MODEL B

Souping up the Model B is usually a task for software but Watford have taken a hardware step towards extra processing speed with their Delta Card.

For "£100 + VAT as near as matters", the Delta card almost halves the standard BASIC benchmarks (average of results) — 15.32 down to 7.8 and knocks 25% off the 6502 second processor timings — 10.27 down to 7.8.

How does it work? Simply by replacing the vital components of central processing unit and main memory RAM with a new generation of chips capable of operating at higher speeds. Out goes the 6502, in comes either a 65C02 or 65112-4. Out goes the slow on-board memory, and in comes Delta's own 64K of fast RAM.

To get the most out of the fast RAM, the first operation of Delta on power up is to copy the Operating System into its own RAM. The same ROM software that achieves this also supplies utility commands, including one to copy the current language into the equivalent of a 16K Delta RAM bank. This could be BASIC, Wordwise, Logo, anything you wish. This 32K of RAM can be write-protected to ensure that it is not accidentally (or deliberately in the case of protected ROMs) overwritten.

The bottom 32K can be used in two different ways. Delta RAM is normally allocated to all memory below screen memory. Thus zero page, the stack, your BASIC program and workspace, are all in fast RAM. The normal screen memory is used and Delta will not therefore upset any memory-mapped screen handling and Mode is automatically worked out. The processor is working at 4Mhz and only has to slow down when talking to the *old* bits of the BBC. The clocking circuitry uses PAL technology to automatically select whether the processor should run at 1Mhz (access to memory mapped I/O, 2Mhz (accessing ROM and RAM on the BBC board) and 4Mhz (accessing Delta RAM).

The Delta processor's clock is phase-locked to the existing 1Mhz clock on the BBC's board to avoid any possibility of timing problems. Even when switching straight from 4Mhz to 1Mhz the board will automatically synchronise with the correct part of the 1Mhz cycle.

A further Delta command is supplied to switch back into slow (normal BBC) mode. This is necessary for the running of games or programs with critical software timing loops. If the BBC's own timers are used by the software then it will not be affected by the Delta's extra speed. The BBC's own timers operate at normal speed and thus the keyboard repeat rate and TIME act as normal.

The Delta card simply plugs into the 6502 socket on the BBC board via a ribbon cable. It has one flying lead which attaches to a pin with a special clip. Power is derived from existing power leads on the circuit board and an unfortunate side effect is that things get too hot if you try and run twin disc drives off the auxilliary power socket.

The card works alongside all other major add-ons, ROM boards and second processors, 6502 and Z80. Software such as Bitstik still requires the second processor to operate but, since it uses operating system routines, runs very much faster. Imagine the mouse drawing software, not always the most rapid, a favourite utility ROM and your own BASIC programs all going twice as fast!

Watford also supply special software to allow their shadow RAM card to work in conjunction with Delta. In this case, Delta processes away up to &8000 and shadow RAM is paged in when the screen is accessed.

Delta could revolutionise the way we look at software. Much of the programmer's effort is put into getting speed out of the computer. With Delta, the effort can go into design and originality. High level languages will produce acceptably fast software, even games.

4Mation's Box of Treasures

A new classroom resources pack is about to emerge from the highly original 4Mation stable. *Box of Treasures* will, among other things, provide the core them for a lengthy project on boxes.

As well as the software the pack will contain a 60 minute audio tape containing two stories and one play, all three featuring music and sound effects. The reading material will include software manual, inspirational poems on the boxes theme and a project book full of ideas, tips and suggestions for the successful organisation of a 'boxes' project.

The software itself will consist of five modules. First is the adventure, *The Box of Treasures*. Second is a Mode 7 wordprocessor, teletext graphics editor and a program for combining the two, including two frame animation. Third is an 80 column 'form processor' for creating record sheets, data sheets, blank forms etc.

Fourth is *Box Draw*, a graphics design program, using boxes as building blocks in complex designs. Designs can be moved about, scaled and dumped to a printer. Fifth is *Box Maker*, a box making program. A number of options can be taken to produce a variety of box shapes, colours and types, a source of stimulus for research into packaging of all sorts and for creative writing. Watch out for a review in a future *A&B Computing*.

Up the jungle

Operation Raleigh put their BBC Microcomputers (and one Electron) to good use on this year's expeditions to Costa Rica, Honduras, Roatan and Belize and Panama.

The SES Sir Walter Raleigh sailed with three disc based machines on board and ship administration (customs lists, ship inventory) carried out with the help of a printer. More ambitious were attempts at satellite communication.

The ship was linked to Central Headquarters and the outposts in Leeds and Hull through Prestel Marine Information Management Centre and the Times Network for Schools. As well as communications, the BBCs are used for programme development, display of Operation Raleigh information, word processing and research record keeping. In Hull a graphics mapping system using the Acorn Bitstik is being developed to show computerised maps of project areas.

The BBC was run off battery or generator in conditions of high temperature and humidity without any problems and Hull are developing a 'jungle box' for future expeditions.

Beginners' ROM

John Kortink

Implementing ROM software, advice and a program template for beginners.

One of the most powerful features of the BBC micro is the facility to use sideways ROMs. These ROMs, fitted in their own sockets inside the beeb, can contain many kinds of application software: utility software, hardware drivers for things such as EPROM programmers, filing systems such as the Disc Filing System or even complete languages such as BASIC, PASCAL, FORTH, BCPL, LOGO, etc.. They can also hold games or other (BASIC) programs which are then instantly available to the user via the standard RFS Rom Filing System in the OS, which works much like the cassette system.

Sideways ROMs exist in memory from address &8000 to &BFFF, one of them being active at a time in the memory map. The OS and hardware can handle 16 sideways ROMs, but the standard hardware in the BBC is 'wired' for four ROMs. Expansion boards will allow all sixteen ROMs to be used but in theory the number of ROMs can be infinite.

Creating your own ROM software is therfore very tempting and this article will try to tell you how to do just that. Unfortunately the following does require some knowledge of machine code. If you seriously want to start programming sideways ROMs, 'The Advanced User Guide' by Bray, Dickens & Holmes, published by The Cambridge Computer Centre is invaluable. It contains important information on this subject and is constantly used as a reference. Of course you also have to have an EPROM programmer to 'blow' the EPROMs for you.

I will first discuss how the Operating System handles the sideways ROMs and then explain how to design the programs, giving a base for ROM software with useful machine code routines to do standard tasks for you.

HOW THE OS DEALS WITH SIDEWAYS ROMS

First of all any sideways ROM has to be recognised by the Operating System. Whenever BREAK is pressed or the system is reset, the OS will look for rightly coded sideways ROMs. The first bytes in the ROM, starting from location &8000, should contain the following codes (a clear example of this is in the program):

00-02 contain the machine code instruction to JuMP to the language entry, the entry which will be called by the OS when the ROM is selected as a language.
03-05 contain JMP service entry, called when the ROM is asked to perform a service, such as processing unrecognised commands or printing its contents (*HELP), as used by most available sideways ROMs.
06 ROM type byte (discussed below), giving information about the entries the ROM has and indicating whether or not it can be relocated to be used in a second processor.
07 The 'offset pointer', being the offset from &8000 of the locations where the essential copyright codes are by which the OS recognises a ROM.
08 A byte indicating the version number of the ROM. It is ignored by the OS.
09 A title string, which is printed on the screen when the OS selects the ROM as a language. In the BASIC ROM, for example, this string is "BASIC", which you normally see when you press BREAK.

The locations of the three other possible 'strings', directly following the string at location 09 are not *precisely* defined:

v0-.. an optional version string, starting with a &00-byte. The error pointer (locations &FD/&FE) will point to this string when a language is entered.
c0-c3 should contain the essential copyright codes, on which ROMs are primarily recognised. c0 is pointed to by byte 07 as explained above. The bytes should be: &00, &28 (ASCII "("), &43 (ASCII "C"), &29 (ASCII ")") thus making "(C)".
c4-.. a copyright message, terminated by a &00-byte. The BASIC 2 ROM for example contains "1982 Acorn" after the copyright codes.
s0-s5 directly following the copyright message, these bytes can contain a relocation address for the second processor when the ROM type byte indicates it.

The language entry is only called when the ROM is a language and indicates it in its ROM type byte. The service entry however is regularly called, whenever the OS requires service.

The ROM type byte (byte 06) has set bits meaning the following things:

bit 7 If set, the ROM has a service entry. All ROMs should have this bit set, as the OS recognises BASIC by the fact that it does not have a service entry.
bit 6 If set, the ROM has a language entry. It will be entered on a hard reset (CTRL-BREAK) as a language when it has the highest priority of the language ROMs. The ROM in socket number 15 (the highest number) is the rightmost socket in the computer and has the highest priority, i.e. is always considered first (figure 1).
bit 5 If set, means the ROM contains a language that is assembled to second processor memory and that it contains the relocation address required for copying over the Tube. The service part should be assembled normally, as it is only accessed by the normal processor.
bit 1 Must be set.

The ROM type byte is the byte you will find at location &2A1+xx (OS 1.20) where xx is the ROM socket number. You can for example switch 'off' a ROM by making the relevant byte zero, so the OS will think it's not there.

ENTRANCE

The sideways ROMs can be entered in three ways: via a service call to the service entry, via the language entry and via an extended vector. When the ROM is entered, location &F4 always contains the ROM socket number.

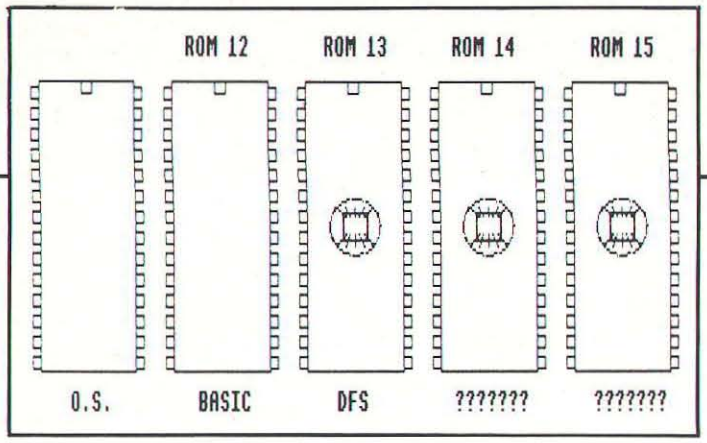
The language entry point is entered with &01 in the accumulator and the OS does not expect a return from this entry. The language should reinitialise the stack and probably start reading commands typed in from the keyboard after initialising itself. Languages are free to use locations &00-&8F and &400-&7FF as their workspace.

The extended vectors are vectors similar to the normal OS vectors in page &02, but point into the sideways ROMs. This is a powerful extension. When a vector is used, it replaces the normal vector. The vectors are primarily used by filing systems in ROM which should handle all filing system vectors.

To make a vector point into a sideways ROM, consider each normal vector to have a number n. This vector is thus located at &200+2*n. To make this vector point into a sideways ROM, make the vector at &200+2*n point to &FF00+3*n. The OS processes extended vectors from there. Then ask for the exact start of the extended vector table by using OSBYTE &A8 (168) with X=&00 and Y=&FF. This returns the address in X (Low byte) and Y (High byte). In OS 1.20 this is &0D9F. Call this V and finally set the vector at address V+3*n to: Low byte address in ROM, High byte address in ROM and the ROM number (held in &F4) respectively. Every call to the normal vector will now go into your ROM, directly. If you have a DFS fitted, you can have a look at the table.

AT YOUR SERVICE

The service entry is the main entry point. It is called regularly for service tasks. This includes processing unrecognised *-commands, strange interrupts, the *HELP



command and unrecognised OSWORD/OSBYTE calls. On entering the service entry the 6502 registers contain the necessary codes:

A contains the number of the service type requested.
X contains the ROM number of the current ROM (also in &F4).
Y can contain a parameter for the required service

The calls the OS makes to the service entries of the sideways ROMs can be executed from the keyboard or programs, using OSBYTE &8F (143) with the service type in X and eventually a parameter in Y.

If a ROM does not want to react to a service call it should return with RTS and all registers preserved.

When a ROM has provided the service however it should set the accumulator to zero, so the command won't be offered to other ROMs. The main service types, code in the accumulator, are given now:

00 Nothing. ROMs should not react to this call as it's already been handled by another ROM.
04 Unrecognised command. Any unrecognised * commands are offered to the sideways ROMs first, then to the current filing system. Disc users who make typing mistakes will know this well, as the DFS will immediately search for a file on the disc using any faulty command.

Languages should get their selection command through this service call. The command to be interpreted is pointed to by locations &F2 and &F3 plus the Y register. The string is terminated by a normal &0D Return byte.
05 Unrecognised interrupt. The OS received an unexpected interrupt from somewhere. If the ROM processes this call, it should still return with RTS, not RTI.
06 BRK ececuted. A BRK instruction has been executed and ROMs are informed before the current language. Location &F0 contains the stack pointer after the BRK and &FD/&FE point to the error number. OSBYTE &BA (186) with X=0 and Y=&FF will give you the ROM number of the ROM which was active when the BRK was executed in X.
07 Unrecognised OSBYTE call. A, X and Y are in locations &EF, &F0 and &F1 respectively.
08 Unrecognised OSWORD call. A, X and Y are in locations &EF, &F0 and &F1 respectively. OSWORD calls with A>&E0 are not sent to the ROMs but to the user vector at &200. OSWORD 7 (SOUND) will cause this call when an unrecognised channel number (&2000 to &FEFF) is used.
09 *HELP command. ROMs should allow all other ROMs to respond to this call, especially when no ROM-specific string follows the *HELP command. The string after the *HELP command is pointed to by &F2 and &F3 plus the Y register. Any ROM should display its name and the commands to which it will respond.

These are the most useful service calls, more calls can be found in the Advanced User Guide.

I will mainly concentrate on making a ROM which will execute unrecognised commands, as this is used by almost all utility ROMs on the market. How to handle a language is much more difficult. The language entry is mentioned for some completeness, should you want to do that more difficult task. Now all the important and difficult OS protocols have been explained, I will now discuss how to design a utility ROM which will respond to * commands. As this involves most of the things mentioned above, you will have a good understanding of implementing sideways ROM software after following this application. Then maybe you could even start designing your own language!

A UTILITY ROM APPLICATION

The phases in design of a utility ROM are of course a personal matter, but three 'phases' in the design can normally be recognised:

1. The design of the communication routines of the ROM with the OS, guiding the right service commands to the right places, including the unrecognised * commands and the *HELP instruction.
2. The design of the main routine that handles the actual * commands i.e. compares the command with any commands in ROM and, if it recognises a command, calls the routine that executes it.
3. The design of the actual command routines, combined with the design of the 'system' routines which can decode any parameters for commands, handle errors, print messages etc...

The first thing to do is to define the first few bytes of the ROM, the ones that contain all the OS protocol codes and strings. The base program gives a good example of this. You can adjust anything you like to suit your own ideas, so just try your own design, but watch the protocols. As the program is full of remarks adjustments should be easy. Unfortunately the program only works directly with BASIC 2, because of the EQU instructions and the use of O%.

The second task in phase 1 is to design the routines for communication with the OS. The program contains a simple service entry, catching codes 4 and 9, the codes for unrecognised commands and the *HELP command respectively. The routine for *HELP is included and the unrecognised commands will be sent to the command interpreter, also included in the program, which will compare the unrecognised command with commands it can execute, and call the right routines to execute the recognised commands (take a look at the simple 'tree' structure in figure 2). It uses a table of commands which can easily be extended with other command names. Detailed explanations of all these things are in the program.

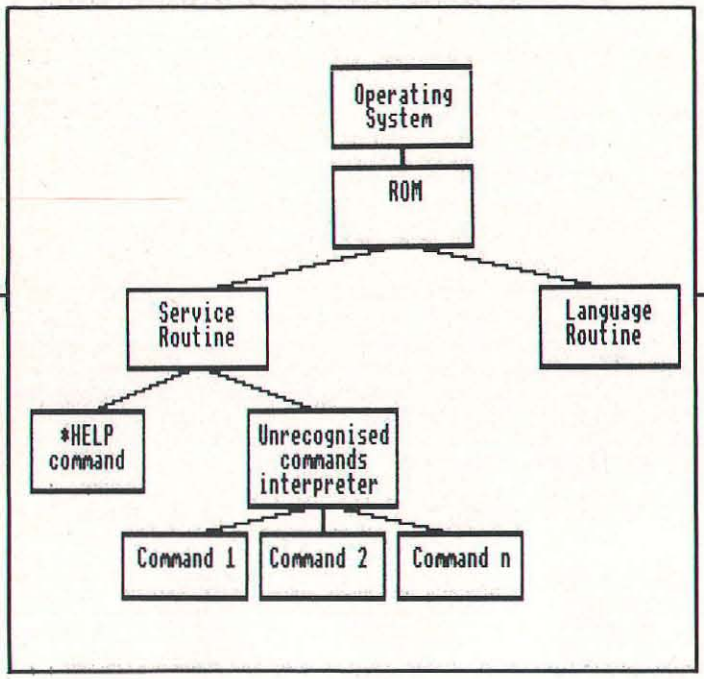
The command interpreter has now completed phase 2. All you have to do now is design your own commands and put them in the program, giving them suitable names. The example commands following the command interpreter in the program just do an RTS, with which all command routines should return. Of course you have to fill in the routines yourself, RTS alone doesn't really do that much, so now it's all up to you! The program, when RUN, will ask you if you want a tryout of the commands or not. When you select a tryout, you can type in a command and the program will tell you whether it was recognised or not.

A full list of commands will always be given with the HELP command. Just type in the commands, without any leading spaces or the normal *. For example, change the RTS after the '.plop' label, indicating the example PLOP command, with LDA—7:JSR&FFEE:RTS and type in the command 'PLOP' in tryout mode. You should then hear the familiar beep and the computer will print 'Executed!'.

When a command is not recognised it will print 'Bad command'. When you don't select tryout mode, the computer will generate the exact code to be put into the ROM and tell you from where in memory it is stored. This code is thus ready to be blown into an EPROM! The whole program is self-explanatory and it shouldn't prove difficult to make your own adjustments and changes to the program. You will soon discover how exciting it is to create your own commands! Happy blowing!

John Kortink's own utility ROM containing a host of useful commands for BASIC and machine code programmers will be documented next month. So if you need further inspiration, return to his article in the January issue of A&B Computing. You never know, you may soon be sending in an example of your own work in sideways ROM!

CONTINUED OVER




```
  10REM Example Utility ROM base
  20REM (C)1985 JK-programs
  30
  40
  50PRINT'"Want a tryout (Y/N) ?";
  60REPEATP=INSTR("YyNn",GET$):UNTILP
  70Tryout=(P<3)
  80IF Tryout PRINT"Y"'"Tryout selected" ELSE PRINT"N"
'"No tryout"
  90
 100PROCassembler
 110
 120IF Tryout PROCtryout
 130
 140PRINT"ROM code assembled from"'"&6000 to &";~O%-1
 150
 160END
 170
 180DEFPROCassembler
 190FOR Option=4 TO 6 STEP2
 200P%=&8000:O%=&6000
 210  IF Tryout P%=&6000
 220[OPT Option
 230
 240
 250
 260
 270
 280.header_codes
 290              \ ROM header codes
 300.language_entry
 310JMPlanguage
 320              \ Jump to language
 330              \ entry routine
 340.service_entry
 350JMPservice
 360              \ Jump to service
 370              \ routine
 380.ROMtype_byte
 390EQUB&82
 400              \ The ROM type byte.
 410              \ This byte is as
 420              \ follows in binary:
 430              \ 10000010, thus
 440              \ indicating a
 450              \ service entry and
 460              \ no language entry
 470              \ or relocation
 480              \ address
 490.Copyright_offset
```

```
 500EQUB Copyright_string MOD 256
 510              \ The Copyright
 520              \ offset pointer, the
 530              \ pointer to the
 540              \ copyright codes at
 550              \ the label
 560              \ 'Copyright_string'
 570.Version_number
 580EQUB&04
 590              \ The Binary version
 600              \ number, for example
 610              \ &04, interpreted as
 620              \ 'Version 4'
 630.Title_string
 640EQUS"My own Utility ROM":BRK
 650              \ The title string,
 660              \ printed when the
 670              \ ROM is selected as
 680              \ a language, with a
 690              \ &00 byte as the
 700              \ string terminator
 710.Version_string
 720EQUS"1.20"
 730              \ An optional version
 740              \ string
 750.Copyright_string
 760BRK:EQUS"(C)"
 770              \ The essential
 780              \ Copyright codes
 790              \ starting with a &00
 800              \ byte
 810.Copyright_message
 820EQUS"1985 Examplesoft":BRK
 830              \ A copyright message
 840              \ terminated with a
 850              \ &00 byte
 860
 870
 880
 890
 900
 910.language
 920RTS
 930              \ The language
 940              \ routine
 950
 960
 970
 980
 990
1000.service
1010              \ The service routine
1020CMP#&04:BNEnot_command:JMPcommand
1030              \ Check for code
1040              \ unrecognised
1050              \ command
1060.not_command
1070CMP#&09:BNEnot_help:JMPhelp
1080              \ Check for code
1090              \ *HELP
1100.not_help
1110RTS
1120              \ Exit service
1130              \ routine
1140
1150
1160
1170
1180
1190.help
1200       \ *HELP routine
1210      PHA:TXA:PHA:TYA:PHA
1220       \ Push registers on stack
1230      LDY#0
1240.he0 LDAtitle,Y:INY:JSR&FFE3
1250      CMP#&0D:BNEhe0
1260       \ Print title string
1270.he2 LDY#0
1280.he4 LDAcmandos,Y:BPLhe6
1290      CMP#&FF:BEQhe9:LDA#&0D:INY
1300.he6 JSR&FFE3:INY:BNEhe4
1310       \ Print commands
1320.he9 PLA:TAY:PLA:TAX:PLA:RTS
1330       \ Exit *HELP routine
1340       \ (and thus the ROM)....
```

```
1350.title
1360EQUS"Utility ROM 1.20":EQUB&0D
1370      \ Title string
1380
1390
1400
1410
1420.cmandos
1430      \ The command table.
1440      \ This table contains the
1450      \ commands followed by their
1460      \ addresses in the ROM.
1470      \ The end of the table is
1480      \ indicated by an &FF byte.
1490      \ Note that the first byte
1500      \ of the address is the high
1510      \ byte, and thus always &80
1520      \ or greater ('Minus')
1530
1540      \ Example command names
1550      \ and their addresses
1560
1570EQUS"PLOP":EQUW FNaddr("plop")
1580EQUS"ZAPPER":EQUW FNaddr("zapper")
1590EQUS"OINK":EQUW FNaddr("oink")
1600EQUS"PLOK":EQUW FNaddr("plok")
1610EQUS"CRASH":EQUW FNaddr("crash")
1620
1630      \ etc. etc.
1640      \ (more commands can be
1650      \ added very easily)
1660EQUB&FF
1670      \ end of table
1680
1690
1700
1710
1720
1730.command
1740      \ Command interpreter
1750      \ Remember 'F2 and &F3
1760      \ plus Y point to the
1770      \ command to be interpreted
1780     PHA:TXA:PHA:TYA:PHA
1790      \ push registers on stack
1800.in0 STY&82:LDX#&FF
1810      \ Store Y and prepare X
1820      \ X points into the command
1830      \ table
1840.in1 LDY&82:DEY
1850      \ Get original Y position
1860      \ and prepare for first INY
1870.in2 INY:INX
1880      \ Increment both pointers
1890      \ to next characters
1900     LDAcmandos,X:BMIin4
1910      \ Get current character
1920      \ in table. BMI detects
1930      \ the address after the
1940      \ command, indicating a
1950      \ succesful match, or the
1960      \ end of the table
1970     CMP(&F2),Y:BEQin2
1980      \ Compare the characters
1990      \ and if they match, carry
2000      \ on comparing until the
2010      \ address is reached
2020.in3 INX:LDAcmandos,X:BPLin3:INX
2030      \ No succesful match, so
2040      \ skip over bytes
2050      \ in command table to get
2060      \ X to the next command
2070     LDA(&F2),Y:CMP#ASC".":BNEin1
2080      \ If current character not
2090      \ a full stop, carry on
2100      \ comparing. This detects
2110      \ abbreviations
2120     INY:DEX
2130     LDAcmandos,X:BNEin6
2140      \ put Y to next character
2150      \ and A to high byte of
2160      \ address, then join the
2170      \ 'succesful match'
2180.in4 CMP#&FF:BEQin7
2190      \ if the end of the table
2200      \ is reached without a match
2210      \ then exit with carry set
2220      \ (note it is already set
2230      \ by the CMP) indicating an
2240      \ unsuccesful search
2250.in6
2260  ]:IF Tryout [OPT Option:SEC:SBC#&80-&60   ELSE [
OPT Option
2270     STA&81
2280      \ Store high byte of address
2290     LDAcmandos+1,X:STA&80
2300      \ Store low byte of address
2310     STY&82
2320      \ Put away current Y
2330     JSRin8
2340      \ Call the command
2350      \ routine
2360     CLC
2370      \ indicate succesful
2380      \ execution
2390.in7 PLA:TAY:PLA:TAX:PLA
2400      \ restore registers
2410     BCSexi:LDA#0
2420      \ If the command has been
2430      \ executed, zero accumulator
2440.exi RTS
2450      \ exit the command routine
2460      \ (and thus the ROM)....
2470
2480.in8 JMP(&0080)
2490      \ jump indirect to the
2500      \ command address
2510
2520
2530
2540
2550
2560      \ Then the command routines
2570      \ follow.....................
2580
2590.plop RTS
2600      \ of course you have to fill
2610      \ in the routines yourself
2620      \ Just RTS doesn't do that
2630      \ much!
2640
2650.zapper RTS
2660
2670.oink RTS
2680
2690.plok RTS
2700
2710.crash RTS
2720
2730
2740
2750]NEXT
2760ENDPROC
2770
2780DEFFNaddr(Address$)
2790A%=EVAL(Address$)
2800  IF Tryout A%=A%+&8000-&6000
2810= A%DIV256+(A% AND&FF)*256
2820     REM Reverse High/Low bytes
2830
2840
2850
2860DEFPROCtryout
2870     REM Try out commands
2880PRINT"Tryout mode"
2890REPEATPRINT"Command?"
2900INPUTLINE$&7A00
2910     REM Input command line
2920IF$&7A00="HELP" A%=9:CALLservice:UNTIL FALSE
2930     REM *HELP instruction
2940?&F2=&00:?&F3=&7A:Y%=0
2950     REM Set pointers to command
2960A%=4:IF(USRservice AND&FF)=0 PRINT"Executed!" ELSE
PRINT"Bad Command"
2970     REM Set A to code for
2980     REM unrecognised command (4)
2990     REM and call service routine
3000UNTIL FALSE
3010ENDPROC
```

We thought it was about time we put you in the picture.

When we introduced our AMX Mouse to BBC Micro users, the response was phenomenal.

And no wonder!

It was regarded as 'pushing the BBC to its limits' and hailed by the press as 'probably the best input device that has appeared recently'. The AMX Mouse brought to BBC Micro users the same sophisticated, positive control that had, until then, been the province of the more expensive computers, like the Macintosh.

Now we announce a new enhanced software package which, can be used with a traditional keyboard or joystick but truly comes into its own when used with the AMX Mouse.

AMX Super ROM and Super Art adds colour to the features of the original AMX ROM, on a new 16K ROM.

It dramatically extends the original AMX ROM facilities to include Pull Down Menus, over lapping colour windows, colour patterns and an extended Icon set in all graphic modes.

Specially written to be usable with the vast majority of BBC Systems including 2nd processors, the Aries RAM Board, the BBC + Shadow mode, tape, DFS and ADFS.

The kernel of the Super Art program is included in the 16K ROM, providing tape and disc users with a Mode 1 colour drawing program.

But Super Art really comes into its own when used on a disc based system; a host of additional facilities can be selected direct from the screen.

Colour pictures can be created in dot resolution up to A4 size by scrolling the screen window, over the picture.

Other facilities include colour pattern editing – use of icons, additional text styles, various copy facilities and a superb colour Zoom.

Pictures can be printed out in colour (Integrex) or shades of black and grey on Epson compatible printers.

The results can be astonishing! – you can even store your finished 'masterpieces'.

The AMX Icon Designer facility gives you further freedom to design and store your own custom icons for use in your own programs.

This fabulous new package costs only £89.95 – a price as remarkable as the package itself – and includes a fully illustrated operating manual.

If, however, you are afraid of mice, you can purchase just the software package plus the 16K Super ROM for just £49.95.

You will still achieve some pretty remarkable results using your keyboard or joystick – not nearly as good as with the Mouse of course – but you can always add one later on for just £40.

Then you'll really be 'in the picture'.

AMX MOUSE

IMAGINATION AT YOUR FINGERTIPS.

BOTH THESE EXCITING PACKAGES ALONG WITH OUR SUPERB RANGE OF AMX ROM AND SUPER ROM COMPATIBLE SOFTWARE ARE AVAILABLE FROM ALL GOOD COMPUTER DEALERS OR BY SENDING CHEQUE OR P.O. TO FREEPOST ADVANCED MEMORY SYSTEMS LTD., GREEN LANE, APPLETON, WARRINGTON, WA4 5NG.
FOR INSTANT ACCESS OR VISA ORDERS, RING (0925) 602959/602690.

Now you can stretch your imagination, without stretching your budget.

AMX PAGEMAKER
£49.95 INC VAT, 2 ROMS + DISC

From Advanced Memory Systems comes an irresistable selection of Icon driven software that AMX Mouse, ROM and AMX Super ROM owners can really get their teeth into!

Seven incredible programs, so comprehensive, they will more than satisfy even the most demanding BBC Micro user and when coupled with the AMX Mouse will broaden the horizons of the BBC Micro to new, undreamed of standards of sophistication.

AMX PAGEMAKER.

A fantastic high resolution A4 Typesetting and Artwork system.

Comprising 2 ROMS (8K and 16K) 'Pagemaker' features a fast scrolling A4 page.

There is a wide selection of typefaces and user definable fonts.

You can load in Wordwise and View files or type directly onto screen.

An incredible 'Cut and Paste' facility includes copying, rotating, stretching and zooming.

For the professional touch there is N L Q high quality dump, and you can dump any part of, or full A4 page.

AMX 3D ZICON.

Now, with a 3D ZICON you can transform three dimensional vector data and translate them into on-screen graphics.

You can Print or Plot graphics to suitable peripherals and produce output suitable for other AMX programs such as ART – UTILITIES and Paintpot.

There is even the facility to sketch in two dimensions and convert to three dimensions and auto swaps generator.

Some interesting example objects on file include – helicopter – jet plane and dodecahedrom.

AMX DATABASE.

Database is a brand new super filing program featuring 1000 records and unlimited fields per record.

It has a powerful search and sort facility and user friendly edit and display.

In fact it is probably the most advanced user friendly database available today.

Special features include: User definable screen formrat for record display – random access files – super fast sort – flexible printer dump and screen display.

AMX XAM.

A brand new and exciting educational program with maximum flexibility, allowing a multitude of options and variations.

With XAM you can create a multiple choice exam which would not be alterable by the examinee.

And you can incorporate within the questions, your own pictures, created on AMX ART, building the finished picture into the quiz and providing an extra and fascinating aspect to the examination.

AMX DESK.

The complete home or office acquisition.

You will wonder how you ever managed without AMX Desk.

Just point your Mouse and select the following additional aids:

MEMO-PAD for up to three pages of text

TELEPHONE/ADDRESS BOOK with 200 or 400 scrolling entries

DIARY with three months calender

ALARM to remind you about appointments

CALCULATOR – Yes! A full function calculator that cannot be lost or suffer flat batteries, and any window can be dumped to your printer at the touch of a button.

THE ORIGINAL AMX ART PLUS UTILITIES.

Now available as a single package – two of AMX's most popular programs.

AMX ART is a superb mode 4 graphics program, making full use of windows, icons, menus and pointers, it allows you to create your own pictures and the host of graphic features in AMX UTILITIES including a fantastic zoom facility for accurate amendments to your ART pictures – one dot becomes a character.

You can also move, copy, rotate, reflect, enhance circles, draw ellipses and arcs, even print icons.

Border can be added and pictures stripped and centred for saving and printing. All pictures can be loaded into AMX PAGEMAKER, AMX 3D ZICON, AMX XAM, AMX PAINT POT, and AMX SUPER ART.

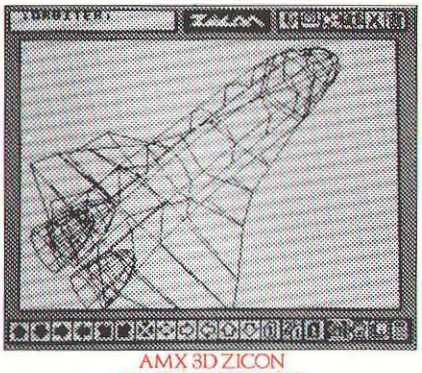
AMX PAINT POT.

The pictures you have created with AMX ART and enhanced with AMX Utilities can now be coloured in, using PAINT POT.

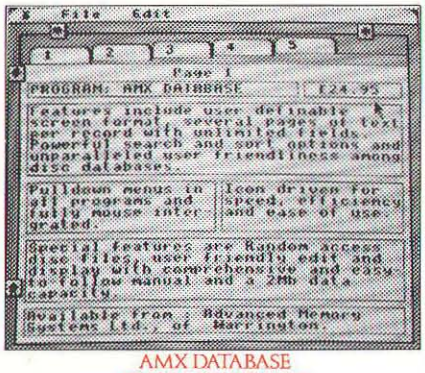
They will burst into life with rich, vibrant colours, and with a palette of 256 different shades you should not be short of inspiration.

You can even store up to 20 Illustrations in Picture Show mode, then display them in sequence.

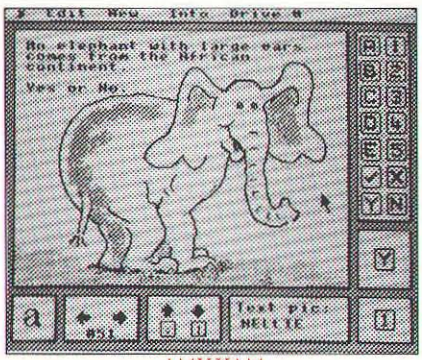
To start you off we have included eight exciting pictures in the program.



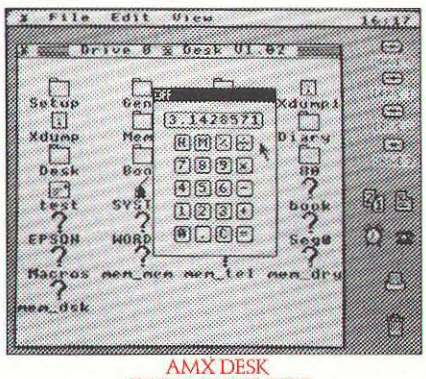
AMX 3D ZICON
£24.95 INC VAT, DISC



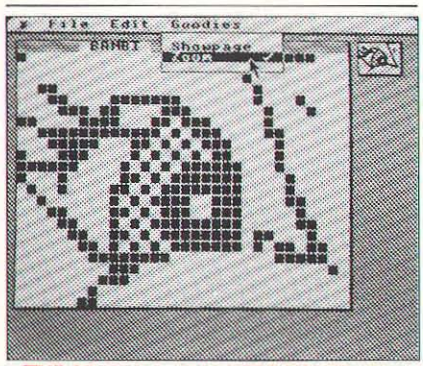
AMX DATABASE
£24.95 INC VAT, DISC



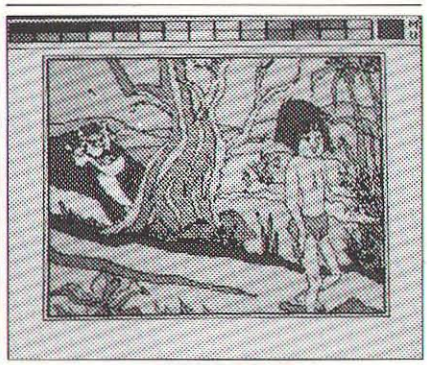
AMX XAM
£24.95 INC VAT, DISC



AMX DESK
£24.95 INC VAT, DISC



THE ORIGINAL AMX ART PLUS UTILITIES
£14.95 INC VAT, DISC



AMX PAINT POT
£14.95 INC VAT, DISC

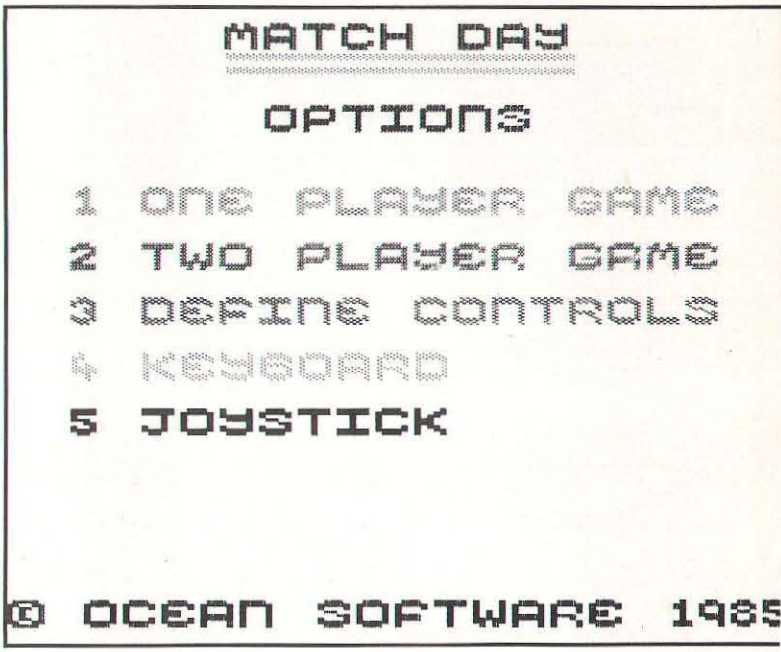
AMX SOFTWARE

IMAGINATION AT YOUR FINGERTIPS.

ALL THESE EXCITING AMX ROM AND AMX SUPER ROM COMPATIBLE SOFTWARE PACKAGES ARE AVAILABLE FROM GOOD COMPUTER DEALERS OR BY SENDING CHEQUE TO PO TO FREEPOST ADVANCED MEMORY SYSTEMS LTD, GREEN LANE, APPLETON, WARRINGTON WA4 5NG.
FOR INSTANT ACCESS OR VISA ORDERS TELEPHONE (0925) 602959/602690.

Soft Sector

Division One for Ocean's Matchday as football returns to the BBC for the first time this season!



Into the top league. That's where Matchday is leading. That's where Ocean have taken the BBC with this football simulation.

The special screen format of Matchday displays an atypical Saturday afternoon, but a scene fairly well known to players of sports simulations on other micros. A stand in the background is full of spectators, some weird characters there are too. There's an electronic scoreboard which records the goal tally and counts down the time. No 45 minutes each way in this game. The Acorn and Ocean logos are painted on the hoardings.

The players are about to move into position on the ultra green sward. Wembley eat your heart out. It must be synthetic! Remarkably quiet for a big match.

Getting some life out of a micro must be very difficult. Some real vitality I mean. Many sports simulations fall flat. They just cannot cast the spell on the players which makes them forget how exciting the *real* thing is. Matchday wins through because the programmers and designers have created a very playable game, recogniseably football.

FOUL! REFEREE?

Matchday can involve you (presumably as player/manager) versus the computer or versus a mate. The keys are user defineable as in every good two player game. It's joystick compatible so you can lay back and play it cool like Glen Hoddle or get stuck in like Peter Reid.

The smooth sideways scrolling, clever sprite movement and design and character detection are all elements in the success of Matchday. The sprites are rather cherubic but clever design takes away the usual flat impression. The angle of the pitch helps give the effect of depth. And of course there are multiple shapes per sprite to create the movement on the ball. The goalkeepers are particularly leaden in their movement, crouched ready to spring. Hit the fire button and they dive left or right for spectacular saves.

The designers have come up with a system results in pretty realistic play. The player's attention is focussed in the area of the ball carrier. This is the sprite currently under control. The opposition have control of their nearest player. if you are playing the computer, then the old adage 'don't hang onto the ball too long' applies. Best to knock it out to the wings, Brian.

Jinking, dribbling and shielding the ball can all be achieved with considerable realism. What you can't do, once you get into the opposition penalty area, is take a dive! There are no fouls in this game. Therefore there is no ref. So no appealing for hand ball. Where in reality you would have obstructed or brought a man down, in Matchday, the chubby midfielder, defender or striker just rides the tackle and gets a head start on the path to goal.

SET PIECES

The ball can go out of play. The computer takes a mean corner to the far post but, alas, no Terry Butcher to nod it down, alas no *proper* heading at all in Matchday. You can intercept the ball with your body or head and use this technique to push the ball past the opposing player.

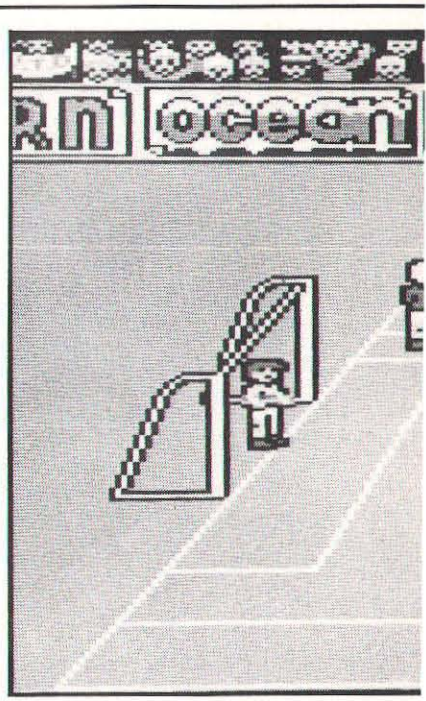
If it's a goal kick, a defender steps up. It's difficult to get a decent clearance out of him and relieve the pressure area on your penalty area. He's a bit like a table football player with a weak left wrist in defence. The knack is to take a run at the ball and time your fire (or as it should be renamed for Matchday 'boot') key.

If the ball goes for a throw-in, a player automatically jogs over to take it. He doesn't wipe the ball on his shirt, nor does he grab a few yards wandering down the touchline. You can control the direction of the throw and a quick upfield throw can set an attack rolling.

Lifting the ball over defenders' heads is not difficult. But nor is lifting it over the bar! As you practice, the right moments to shoot become easier to judge. The first area to improve is your ability to quickly assess which of your players is now under keyboard or joystick control. This player can change quite rapidly as the ball bounces around out of tackles near the penalty area. The sooner you are in control, the quicker you can act.

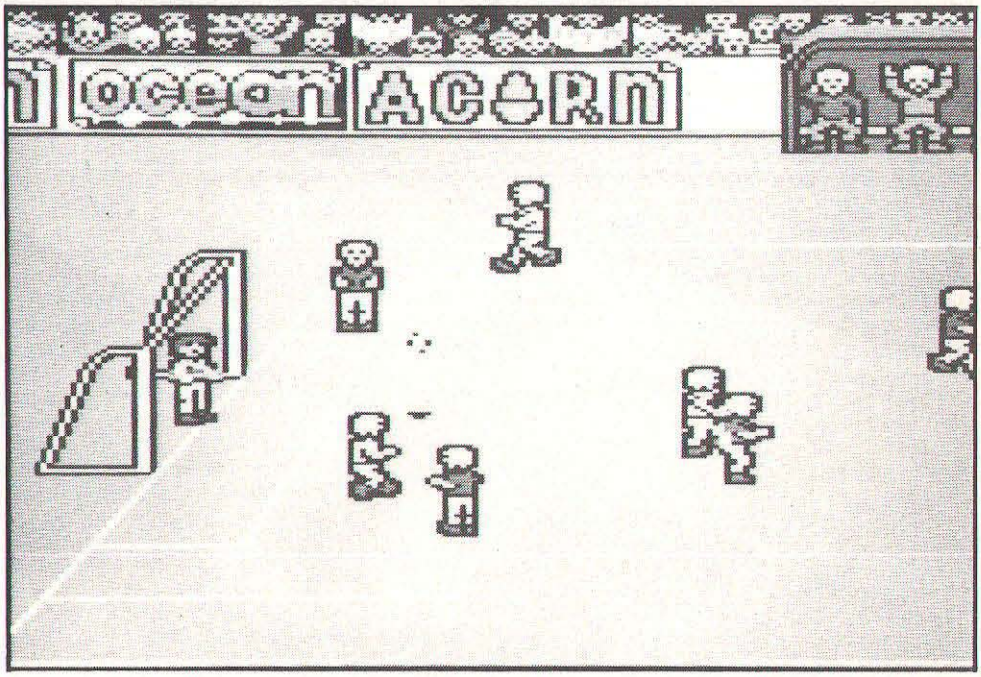
Matchday is very playable. Artificial of course but very entertaining, especially in two player mode. It's the first sports simulation on the BBC which has really succeeded in providing a playable game on screen. Football Manager is excellent in its way but in Matchday, you are the player on the field. It's great running down the wing looking to centre the ball and your first goal, well, a dream come true! If you are a football fan then Matchday will not disappoint. If you are a games player then Matchday should prove a new challenge to your skills in a brand new environment.

It's appropriate that Ocean, who hale from Manchester and have Liverpool connections should be the ones to bring football coverage back onto the BBC this season. Three points to Matchday. A real winner.



ocean
HOME:00
AWAY:01
TIME:30
E.R CORP.
ACOR

ACORN


ocean
ACORN

Competition

A description defying Adventure. And you might be lucky enough to win it!

Bored of the Rings has already been hailed as a classic, we reckon. It must have been, somewhere.

It's the all time antidote to goblins, Gandalfs and gaping caverns.

Apparently a small gnome-like figure was seen climbing the staircase to SilverSoft and , when accosted, threw down a small plastic case while reciting the following verse:

I will return in two weeks
This game for to seek
And if it's not in the shops
The industry's in for the chop.

He wandered off muttering threats about Middle Earth software exports and bouncing elephants.

Silversoft took this threat very seriously indeed and thus *Bored of the Rings* was published.

The wizard you can see on this page is an important character in the game *Bored of the Rings*. He has just paraded around the Personal Computer World Show for eight hours and is beginning to look a little tatty. It's also very hot inside that getup.

What you have to do to win a copy of *Bored of the Rings* is to invent an original and amusing caption for our Wizard. Just what was he saying (he was a she on one day of the show!) as our photographer snapped this picture?

Write your caption on a postcard or envelope if you don't wish to cut out the form below. Don't forget to include your name and address.

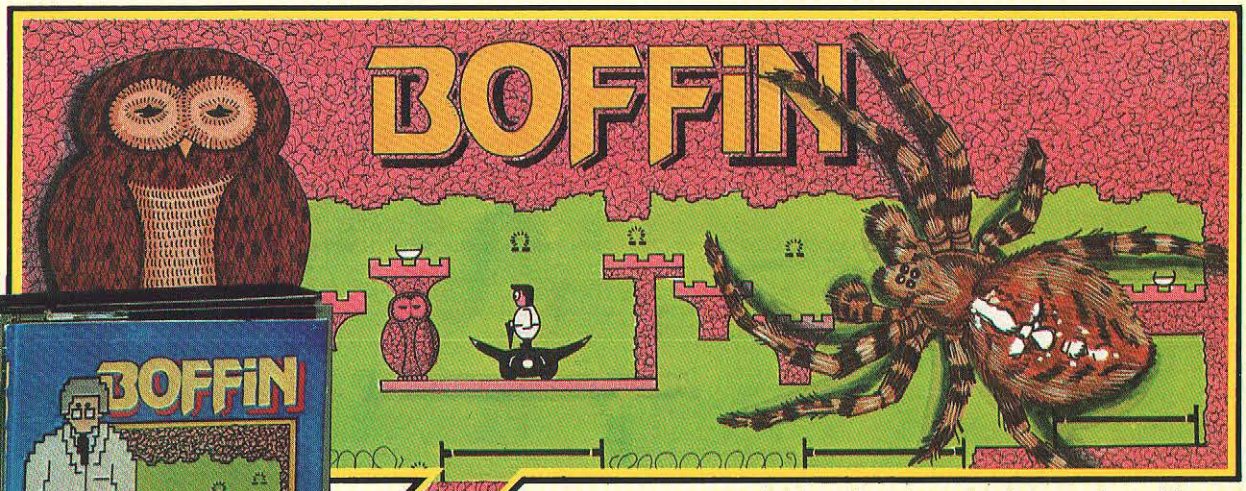
I think the Silversoft Wizard is saying: .....
.....

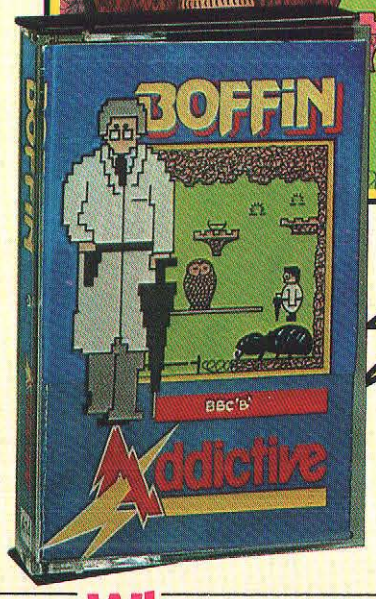
Name .....
Address .....
.....
.....
.....
Postcode .....

Send this coupon (or separate sheet) to Bored of the Rings, A&B Computing, Number One Golden Square, London W1R 3AB. The final date for your entries is 10th December 1985.

Can YOU avoid the fangs of a deadly spider?

Addictive launch BOFFIN
A Mad Professor on the Loose





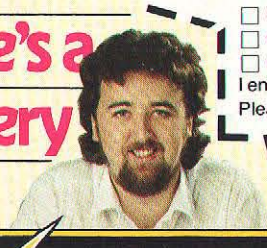
In this action-packed, superbly animated arcade game you have to make crucial decisions to determine the fate of Professor Boffin as he journeys through hazardous caverns – one slip and a horrible doom may await him! Boffin is armed with an umbrella which he can put up to break a fall or a leap from a great height, and trampolines are to be found in certain of the caverns which help him to "bounce" his way to safety. However, there is added danger as hideous creatures are lurking waiting for Boffin to stray too close to their fateful grasp!
Boffin is available for BBC 'B' and Electron priced £9.95.

ALSO AVAILABLE

STRINGER
Stringer is an arcade adventure game in which you are a press photographer set the task of snatching a shot of a Film Star in an hotel. The game is tantalisingly playable, graphically excellent and, of course – Addictive!
Available . . . Commodore 64 £6.95

FOOTBALL MANAGER
The All Time Classic
Available . . . Spectrum 48K £6.95 . . . Commodore 64 £7.95 . . . Amstrad CPC 64 £7.95 . . . BBC 'B' £7.95 . . . Electron £7.95 . . . Dragon £5.95 . . . Vic 20 £5.95 . . . ZX81 £5.95

When you open an Addictive Game, there's a new world of discovery waiting for you.



To: Addictive Games Ltd, 7a Richmond Hill, Bournemouth, Dorset BH2 6HE

Please send me:

☐ cassette(s) for Boffin for ..... (computer)
☐ cassette(s) for Stringer for ..... (computer)
☐ cassette(s) of Football Manager for ..... (computer)
I enclose cheque/postal order for* £ .....
Please debit my Access card No. .....

*Delete as applicable

Name .....
Address .....
.....
Postcode ..... Tel: .....

Access orders can be telephoned through on (0202) 296404

For overseas orders, please add £1.50.

Addictive

• Addictive Games •
7a Richmond Hill, Bournemouth BH2 6HE
Telephone: 0202 · 296404.

Library

S. Taylor

Useful multi-purpose functions and procedures.

When writing programs in BASIC, I find that a number of similar operations are needed in a variety of different programs. It is good programming practice to write a library of routines for such operations and to incorporate them into new programs as required. The fourteen FUNCTIONS and PROCEDURES shown are for a selection of common operations which I hope readers will find useful, either as they stand or with some modifications.

PROCRT This PROCEDURE simply causes a wait in a program. It prints RETURN at the bottom right of the screen to remind the user to press this key when he is ready to continue. It is used when information has been presented on the screen and the user needs time to read it. &D is the hexadecimal number of the RETURN key. Any other key could be used by changing this number and the PRINTed message. Key numbers in hexadecimal are on page 497 of the User Guide.

FNSC This FUNCTION takes a character string and picks out its first character. If the character is a lower case letter, it is converted to an upper case letter by subtracting 32 from its ASCII number. Its purpose is to identify single character responses to requests such as menu selections. The conversion to upper case is to simplify the identification of menu items using IF statements where only the upper case need be tested.

PROCTL A title or other message may be incorporated into a PROCEDURE as illustrated by PROCTL, which is used to display a title as often as necessary. SPC4 needs to be adjusted according to the length of the title and its position across the screen. PROCTL can also be used in printed output.

PROCCT This PROCEDURE clears the screen before using PROCTL to display a title at the top.

PROCGE An error message for general use is created by this PROCEDURE. Insert your own message after PRINT.

PROCES A signal may be printed whenever an error occurs by using this PROCEDURE. Clearly, any signal could be used. In MODE 7, any colour or flashing colour code could preceed the signal. Personally, I use PROCGE followed by a brief error message when my programs detect an error for which a specific message would help the user. Notice the ; after the message to suppress the new line.

FNNR(J$,J,K) Every programmer should be aware that anyone using a computer may accidently enter a number which is outside the limits acceptable to the program. Consequently, the program might collapse or not work correctly. FNNR allows a program to ask for a number within a specified range. Three parameters are required: J$ is the prompt describing the number, J is the minimum acceptable input and K is the maximum acceptable number. J and K may be negative but K must be greater than J. If a number outside the range is typed in, then the FUNCTION tells the user the limit and asks for the input again. The input may be any BASIC expression using + − / * ß ( ) and the figures 0 to 9. However, spaces and variable names are rejected. For example, if a program needs an interest rate to be specified, the program might be written as:

INTRATE = FNNR("Specify your interest rate, %",2,25)

where the programmer considers that 2% and 25% are the minimum and maximum suitable values. It will not be possible to input a number outside this range. Possible inputs would be 8.25 or 7.5*(1.1+.25). The latter expression would be evaluated and INTRATE would have the value 10.125.

The length of an object might be input using

LENGTH = FNNR("Length in inches",5,125)

PROGRAM LISTING

```
30000DEFPROCRT:LOCALJ%:REM Wait (B233)
30010PRINTTAB(33,24);"RETURN";:REPEATJ%=GET:UNTILJ
%=&D:ENDPROC (8E83)
30050DEFFNSC(J$):LOCALJ%:REM Single capital (6B2E)
30060J%=ASC(J$):IFJ%>96ANDJ%<123J%=J%-32 (8669)
30070=CHR$(J%) (79B8)
30100DEFPROCTL:REM Title (C95D)
30110PRINT'SPC4"Spur and Helical Gear Design"'
:ENDPROC (1B77)
30150DEFPROCCT:REM Clear, title (444A)
30160CLS:PROCTL:ENDPROC (A8F5)
30200DEFPROCGE:REM General error (410B)
30210PROCES:PRINT"Incorrect input":ENDPROC (845B)
30250DEFPROCES:REM Error signal (867D)
30260VDU7:PRINT" _________ ";:ENDPROC (BBD9)
30300DEFFNNR(J$,J,K):LOCALL,L$,J%,K%,L%,M%:REM Num
ber in range J to K (395A)
30310@%=&808:PRINTJ$;:INPUT" ? "L$:L%=LENL$:IFL%=0
L=0:GOTO30315 (6F43)
30311K%=0:FORJ%=1TOL%:M%=ASCMID$(L$,J%) (0D63)
30312IFNOT(M%>39ANDM%<44ORM%>44ANDM%<58ORM%=94)K%=
1:J%=L% (635A)
30313NEXT:IFK%=1PROCGE:GOTO30310ELSEL=EVALL$ (A21C
)
30315IFL<J PROCES:PRINT"Minimum ";J:GOTO30310 (517
D)
30320IFL>K PROCES:PRINT"Maximum ";K:GOTO30310ELSE=
L (6082)
```

PROGRAM LISTING

```
30350DEFFNIR(J$,J,K):LOCALL:REM Integer in range (
FD86)
30360L=FNNR(J$,J,K):IFL<>INTL:PROCES:PRINT"Not int
eger":GOTO30360ELSE=L (F935)
30400DEFPROCHD(J%):REM Heading (5786)
30410IFJ%=0:CLS:PRINTSPC9""STRING$(20,"_"):FORK%=0
TO1:PRINTCHR$141SPC10"S    H    G    D":NEXT:PRINT
SPC9""STRING$(20,"_"):GOTO30430 (0185)
30420PROCPN:PRINT':VDU1,27,108,20,1,27,78,4,1,14,1
,27,69:PRINTSPC2STRING$(15,"_")':VDU1,14:PRINTSPC3
"S   H   G   D":VDU1,14:PRINTSPC2STRING$(15,"_")':
VDU1,27,70 (2D07)
30430PROCTL:PRINTSPC7"Gear teeth to BS436:1967"''S
PC7"20 degree pressure angle"'' (A18B)
30440PRINTSPC3"(C)  Doug Walton and Stan Taylor"''
SPC3"Mechanical Engineering Department"''SPC7"Univ
ersity of Birmingham"'' (2CC7)
30450PRINTSPC8"SI units, BBC, Nov 1984"''""STRING$
(38,"_")':PROCPF:ENDPROC (283C)
30500DEFPROCCF:VDU23;8202;0;0;0;:ENDPROC:REM Curso
r off (0786)
30510DEFPROCCN:VDU23;29194;0;0;0;:ENDPROC:REM Curs
or on (22A3)
30520DEFPROCPN:*FX3,10 (ACB6)
30530ENDPROC:REM Printer on (4FEA)
30540DEFPROCPF:*FX3 (8DCD)
30550ENDPROC:REM Printer off (1C38)
30560DEFPROCFL:REM First letter in reply (2D17)
30570PRINT'"First letter ? ";:X$=FNSC(GET$):ENDPRO
C (B6F9)
```

to which the user might type 800/25.4 if his length is 800 mm.

FNIR(J$,J,K) This is similar to FNNR except that it rejects non-integer numbers and is useful where a whole number must be input. It uses FNNR.

For example

DAYS = FNIR("Number of whole days",7,28)
will not accept 18.25 as an input.

PROCHD(J%) This is an example of my own personal program heading. No readers will want this as it stands but you may glean some useful ideas from it. It is a full screen display which I use for a gear design program in MODE 7. It is mostly PRINT commands but notice that PROCHD(0) sends the display to the screen and PROCHD(1) sends the display to an EPSON RX printer. Line 30410 is omitted from printer output and line 30420 is excluded from screen output. Lines 30430 onwards are output to whichever device is selected.

On the screen, S H G D is in double height characters. The VDU command in line 30420 indents the printing by 20 spaces, sets a skip over perforation, sets enlarged print mode and emphasised print mode. This VDU command is suitable for an RX printer. As in some of the other PROCEDURES, I have used colour codes in the printing to make the display more attractive. Unfortunately, these do not show on the printed output so I leave readers to make their own colour arrangements. Whenever I write a new program, I simply change the relevant lines of this PROCEDURE.

PROCCF and PROCCN These PROCEDURES switch the cursor off and on respectively. I dislike a flashing cursor except where it is useful.

PROCPN and PROCPF The printer is switched on and off as required by these two PROCEDURES respectively. They are used in PROCHD. The screen is disabled when the printer is on so that the printed material is not displayed on the screen.

PROCFL This is my menu reply PROCEDURE. This asks for the first letter of a menu item and identifies the key pressed in reply without RETURN being used. After use, X$ contains the first letter of a menu item which can be identified using IFs. FNSC and GET$ are used. For the identification of menu items, I use the first letter of a representative word but it is sometimes difficult to select menu lists with all different letters.

GENERAL COMMENTS

The line numbers start at 30000 so that the FUNCTIONS and PROCEDURES can be appended at the end of any program. They may conveniently be stored on tape or disc using *SPOOL "filename", followed by LIST and then *SPOOL to close the file, as explained in the User Guide page 402. To add the routines to any program, at any stage of preparing it, simply *EXEC "filename". Alternatively, the routines could be typed using WORDWISE and saved in a file. Again *EXEC "filename" would incorporate the routines into a program.

Notice that some of the routines use other routines so if you only wish to use a few of them, make sure that you have included any others which may be necessary. The routines have been written to use the minimum amount of computer store by using short variable names, no superfluous spaces and a minimum number of line numbers. The REMs may, of course, be omitted.

I hope that readers will find some or all of the routines useful with modifications where appropriate. Perhaps the ideas presented will encourage readers to develope their own program libraries to suit their personal computing uses.

Doubling Up

Tony Self

Simplfied volume handling from UDM plus part 1 of the official upgrade from Acorn

UDM UPDATE

In the October issue of A&B,Clive Grace commented in Feedback that the UDM DDFS was one of the best around, but couldn't they do something to clear up some of the gremlins which existed in the version 2.00. Well Clive must be psychic, as UDM have recently released a version 3.1 which should put matters right.

UDM have made some very nice improvements to this version, which certainly makes their system one of the easiest to use for a novice.

All the major changes centre around the expanded volume facility in double density mode — unfortunately, they have done nothing to improve on the maximum number of files available in single density, which still remains at the standard 31.

Version 2.00 gave you the facility to *EXPAND the disc catalogue in double density mode, increasing the maximum number of files to 120, by adding three extra volumes. Each volume was capable of holding 30 files. These could be selected by *VOLUME ‹drv›‹vol› (where vol was a letter in the range A — D). Version 3.1 uses the same system, but makes the volumes easier to access. In fact the user need not know how the volumes are organised at all.

The *EXPAND command has been removed and replaced by an option available after formatting. The formatting command itself has has been improved upon, as it now displays the track being formatted during the operation. I think it is always reassuring to know a program is actually doing something. After the formatting has been completed you are asked whether you require expanded catalogues and then whether you wish to format another disc — again a useful addition.

TRANSPARENT VOLUMES

The best improvements, however, are to the way the system manages the volumes for you. In version 2.00, if you knew a file was on a disc, but were unsure about which volume, you could invoke the *FIND command to search all the volumes for you. On finding the file, that volume would be selected in preparation for a load command. Well version 3.1 does not have this command, instead you just load the file. If the file can not be found on the current volume it automatically selects the next volume and tries again. It does this for all four volumes, before issuing a file not found error. Similarly, when saving files, if the current catalogue is full, it will select the next volume and so on until it finds free space or issues a catalogue full error. In both situations the volume actually loaded from or saved to becomes the current volume. Of course you still have the option of manually selecting the volume with *VOLUME, which now accepts lower case as well as upper, but I suspect that this command will become more or less redundant to most users.

Finally a new command has been added — *CATALL, which, as you might imagine, will list all the catalogues on an expanded disc. This command works in a very neat way. It starts by cataloguing the current volume. Pressing the space bar will then display the next volume's catalogue and so on. However, pressing any other key will exit the routine and leave you in the last volume displayed. The key pressed is also put into the keyboard buffer, so it can form part of the next command you are going to enter (i.e. "C" if you are going to CHAIN a program).

UDM have obviously gone to some trouble to listen to their customers and the comments made in the press, and have come up with a very worthwhile upgraded DDFS Rom. The package now also includes a disc sector editor which operates in either density, an essential tool for the serious disc user.

I would certainly recommend that all existing UDM users upgrade their system and that readers contemplating purchasing disc systems should seriously consider this option.

ENTER ACORN

A little late in the day, but nevertheless welcome, Acorn have brought out their official 1770 upgrade. Although this kit does not provide for double density operation in itself, it is the first step to take if you wish to avail yourself of the Acorn ADFS. Basically the kit replaces the old 8271 and with a disc controller board holding a WD1770 as used in the BBC B+ and many of the independant double density boards around, such as Opus and Solidisk. The kit is very similar as well, comprising the board of two TTL chips, a couple of links (if you are lucky) and a 16K EPROM.

If you read my review in the May issue, you might remember my comments about Solidisk's ingenuity in using shorted polyester capacitors as links. Well Acorn have to take the biscuit. I quote from their covering letter which came with the kit which only included one wire link — "these are in short supply, please make another up using a staple of other suitable material".

Fitting the kit is very straightforward, helped emensely by the excellent fitting instruction booklet. The DFS EPROM supplied was the version 2.20, which is an update on the EPROM originally supplied with the B+. The documentation supplied with the kit is excellent. The 93 page DFS user guide is the one which has been issued for sometime now, but addendum sheets are included covering the new commands available on the 1770 system.

There is little point in me covering the facilities of this DFS as they have already been covered in articles on the B+. Suffice it to say that buying this upgrade on its own would be pointless if you already have an Acorn DFS, as you will undoubtedly find that some of your existing protected software will not run on the 1770 system. However for new users and those who wish to take advantage of Acorn's ADFS this is probably the right way to go. Also new software becoming available on disc should be written taking into account Acorn's new standard.

I hope that in the next issue I will be able to report on the ADFS itself. Although I have already recieved a copy of the ROM I am still awaiting a copy of the utilities disc and the user guide.

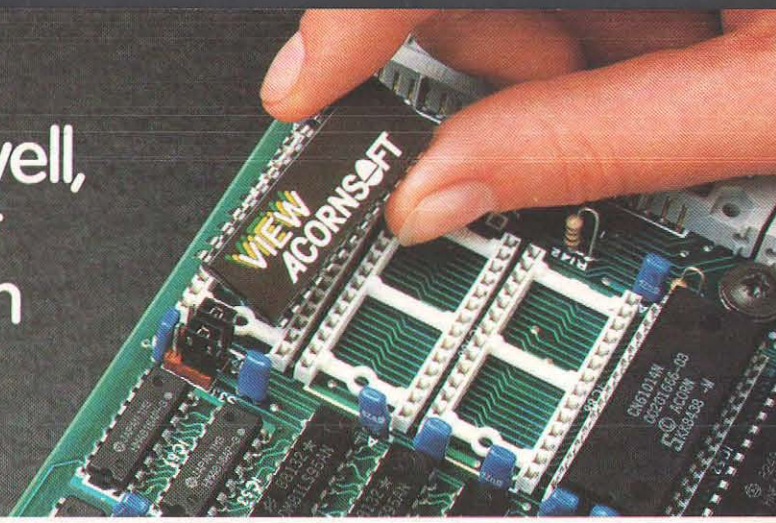
As a final note I have included in Table 1 benchmarks covering these products, plus the figures for the Kenda DMFS, kindly supplied by Allen Hardy, and some mystery figures. Make sure you get next month's issue to see how these figures are obtained.

Table 1 Benchmarks

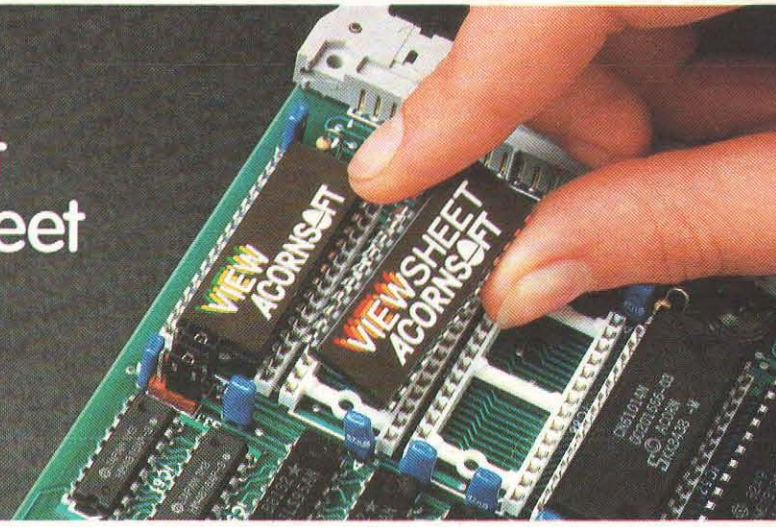
TABLE 1 — Benchmark timings

Benchmark	Acorn single	UDM DDFS single	3.10 double	KENDA DMFS 1.04 single	double	????
1	5.59	4.42	4.13	7.17	6.01	0.19
2	5.39	3.79	3.92	5.56	4.34	0.17
3	1.87	0.88	0.81	60.58	60.59	0.49
4	6.95	5.77	5.47	10.96	8.89	4.59
5	6.63	5.47	5.34	5.78	5.52	2.34
6	10.57	9.43	8.76	14.82	12.02	6.15
7	9.97	8.65	8.36	8.93	8.54	2.85
8	3.58	2.79	2.47	5.41	4.71	1.59
9	3.23	2.26	2.33	2.60	2.68	1.05

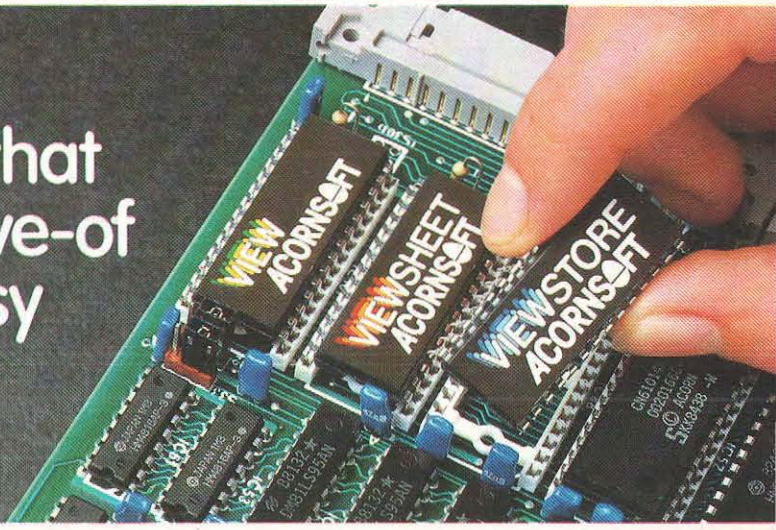
"VIEW handles very well, and is certainly the best word-processing system of its size..." Practical Computing.



"……probably the most sophisticated spreadsheet yet developed on a home micro..." Accountancy Age.



"ViewStore achieves that highly desirable objective-of being powerful and easy to use..." A & B Computing.



Why everyone's so happy to express their point of view.

The View range is the largest range of software available for the BBC Micro. And the most sophisticated.

Hardly surprising, since the View range comes from Acornsoft, and is designed to use the capabilities of the BBC Micro rather more fully than any other software package. Indeed, VIEW was awarded the 1984 British Microcomputing Award for home software.

The View range also includes ViewIndex, ViewSpell, ViewPlot and a Printer Driver Generator, all of them powerful, sophisticated and, above all, friendly to use.

Why not fill in the coupon (or call us on 0223 214411) for more details? And then you'll be able to form your own point of view.

Please send me information on the View range.

Name

Address

Post Code

AB 1285

Send to: Acornsoft Limited Cambridge Technopark 645 Newmarket Road Cambridge CB5 8PD.

ACORNSOFT

Memory Module

Mark Webb

A fast, permanent 64K RAM module for filing, program overlay, print buffer and high resolution graphics.

True to their word, Permanent Memory Systems delivered their 64K memory module in time for the December deadline. The product specificaion had set the imagination going back in August and I was eager to give it a try.

What with Challenger, the Opus RAM disc and the BBC+ 128K, Solidisk Sideways and Watford's offering in the pipeline, RAM, the expansion of, is this autumn's theme tune to which we reviewers dance. It's about time of course. 32K has been a bad joke for some time now.

The especially interesting aspect of PMS's module is the NICAD (Nickel Cadmium) battery which keeps the 64K fresh and your data uncorrupted.

There are two main advantages derived from this non-volatile nature. Firstly portability. The module can be unplugged (it plugs via ribbon cable into the 1Mhz bus) and taken away in your pocket or bag, or in a neat case that PMS will supply for an extra few quid. This is useful if you regularly use a BBC at work and wish to safely transfer programs or data to a home based system, and vice versa. Useful but not revolutionary, since floppies are fairly easily transported.

Secondly, programs and data can remain in the module when the computer has been switched off, awaiting your return after lunch, or the next teacher who wishes to use the demonstation programs installed, or a colleague to whom you wish to leave a message in a predetermined file or on booting into the module.

MANAGING THE MEMORY

There's plenty of memory around these days but what can you do with it? Acorn supply the chips and the ideas but not the software. If you want a printer buffer in your Plus you have to buy A&B Computing and read Ian Copestake's article and use his firmware.

When you buy your PMS module, you get quite a bit of software with it, and necessarily so. Like shadow RAM, paged RAM, mapped in 256 byte chunks and with a paging register to destinguish between them, is awkward to work with. PMS have thought through the possible applications and come up with working software and some nice demonstrations of what might be done with the extra RAM.

The two main supplied programs are the RAM Operating System (ROS) and the Printer Buffer. These are accompanied by a number of file transfer utilities and demonstrations. Software comes on tape or 40 track disc with 80 track converter (redundant if you take the good advice to hold the master and use backups for work).

The manual is the first place to go. It's got al the formalities: 6 month guarantee, what you should get for your money, "a manual (obviously you found it!)". As well as the odd touch of humour (entry in the index: Recursion — see recursion), there are plenty of dire warnings: "IMPORTANT". Nice that the authors have considered the novice. One side effect of using RAM for filing is that everything happens very fast. Some software wipes the whole module and, unlike with discs, there is no time to reach for the Break key to forstall that stupid command you just typed in.

After a browse, I plugged in and performed the recommended checks, checking RAM, switching off the BBC for five minutes or so and then checking the pattern retained by the battery refreshed RAM. Passed with flying colours.

RAM DISC

In these days when games are appearing on EPROM and hard discs ae selling below £500 the poor old floppy could be on its way out already, to be replaced by the RAM disc. Permanent or not, RAM is fast filing. The key to using the module in this fashion is the ROS. It's installed by CHAINing an initialisation program. PMS have chosen &A00-&AFF for the loader which copies the 1K of ROS code into &E00-&1200. This loading process occurs on a Break or *ROSON (*R.). The space for the Break and OSCLI intercepts are defined when ROS is intalled and you get the opportunity to redefine this area if the 42 bytes clash with software you wish to run. Default position is the area for cassette header for BPUT files. The second option available is the NMI workspace (disc/net), safe as you are now using the RAM disc. &1200 to &1900 is not now used so, if you haven't reset PAGE (it is not done automatically) this space could be available for program use.

Once the ROS is in the module (always appearing first and locked), it is initialised with CALL &FDFD. ?FCFF=0 must first be issued if the BBC has just been turned on and ROS (safe in the module) is to be initialised.

These commands can be put in an EXEC file along with other commands to very rapidly set up the BBC with permanently stored programs. Next month A&B Computing publishes Alan Rowley's perpetual diary program. This works marvellously in the module. On auto booting a disc, ROS is initialised and your diary instantly displayed. Searching entries and making entries is equally fast. This is possible because ROS supports random access filing.

Similarly, Jonathan Evan's Wordwise Plus (*Plustalk A&B October)* utilities can be rapidly moved from PMS RAM to Wordwise Plus segments for execution. Two files can be open at one time and this allows transparent continuous processing of wordprocessor files of up to 50K long (with the appropriate software). The two files limit makes the movement of individual files to and fro, from RAM to disc and vice versa difficult to manage. A utility FILECPY takes all the hard work out of the disc to RAM. The flow in the opposite direction is more laboured with LOADing from RAM into main memory followed by *DI. (the new minimum abbreviation in ROS) and a SAVE.

Once ROS is in the module, a number of filing commands are at your disposal, most of them familiar from the Disc Filing System. Some immediate differences are: number of files limited only by memory, 15 character file names, upper and lower case treated differently (difficult to get used to!). Directories don't operate although you can give files common prefixes for similar though not so convenient effect.

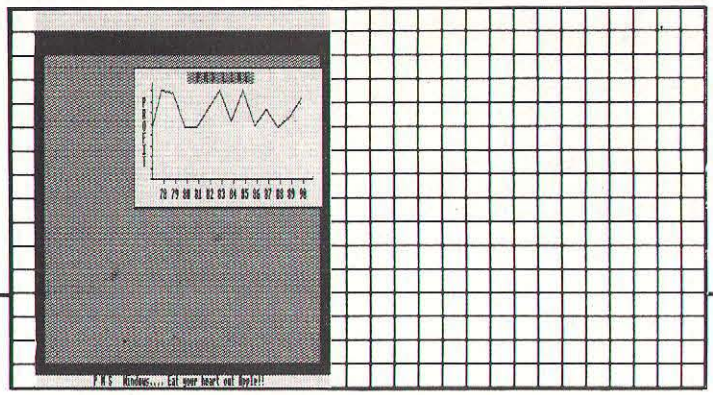
You don't have to "format" the RAM. The catalogue is organised as a linked list of entries, each entry having a pointer, pointing to the next entry. The list is terminated by a zero pointer. Each file is preceded by a block of information containing the pointer, the name, the load and execution addresses, and the length. The execution address is currently restricted to two bytes and is therefore not second processor compatible. A Tube version of ROS is under development.

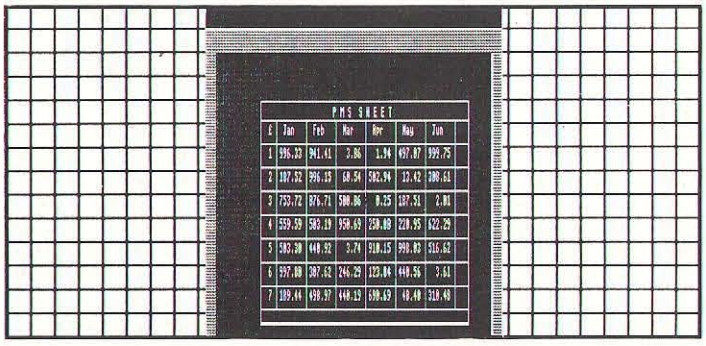
The DFS utilities (if installed) are available to the module. ROS doesn't respond to new ADFS or DFS 2.0/2.2 (BBC+) commands. Bytes free and used are given on a *RCAT or after *RCOMPACT. LOAD, SAVE, *LOAD, *SAVE, *SPOOL, *EXEC and *RUN are supported. *ACCESS can be used to lock a file in the normal way with an L attribute. It can also delete a file with a D attribute. The equivalent *DELETE can also be used. The files remain visible to a *RCAT until a *RCOMPACT, SAVE or *SAVE. D and L can be removed with straight *ACCESS *filename.*

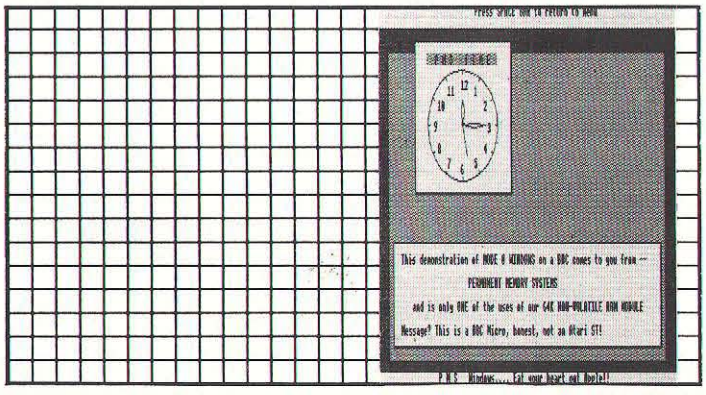
Auto-booting is catered for. R and Break are depressed together to EXEC the !Boot file.

PRODUCTIVE RAM

To further make use of the module, PMS supply a printer buf-







fer. Most of the printers in use with the BBC Micro have no buffer at all or perhaps, at most, 2K. If your BBC is used for program development, documentation, business wordprocessing or report writing, database listing or spreadsheet presentation then producing hard-copy is a natural extension to your activities. A 63K buffer allows you to feed the printer very large quantities of text without tieing up your BBC for long periods of time. The new Amstrad wordprocessor system does the very same thing with part of its 128K. It makes a surprising difference. No more frustrating waits watching the tractor mechanism feed the paper through. A facility well worth having if you can.

We have already mentioned FILECPY, the utility for selective copying of files from disc to RAM module. This program is very friendly with choice of drive, plenty of prompts and information on screen. Less good are the RAM to disc and disc to RAM imagers. The former just gets on with it, RAM to current drive, so be prepared. The complete image of the module also takes some minutes to complete with an Acorn DFS 0.9 (still Acorn's *standard* DFS). The quoted timings of 40 to 50 seconds are perhaps attainable with DFS 1.2 and it takes about a minute on DFS 2.0 (BBC+).

Further utilites include RBACKUP, which uses the module memory to make disc backups. Single drive users need less source disc/destination disc swaps and all disc users can *RBACKUP without corrupting main memory. This is a real time saver if you are wordprocessing or developing a program and require security backups.

There are also SWAPIN and SWAPOUT, programs which move the *complete* BBC Micro memory into and out of the module.

This can be used for security purposes, before running some suspect code for instance, but also to allow rapid transfer between applications. A piece of software (with data and variables intact) can be briefly filed away, while an alternative is run, and then swapped back, allowing the user to continue where he or she left off. You could break off from a spreadsheet to quickly jot a note or search your database for an entry and then straight back into the spreadsheet without having to reload data from disc.

If you program a key to issue the * command for you, then your final lap around Silverstone in Revs or the penultimate screen of Boffin need not perhaps be fatally interrupted by the girl (or boy) friend's telephone call! You can *SWAPOUT, take the call, *SWAPIN and go on playing where you left off.

DEMONSTRATIONS

OVERLAY and WINDOWS are the two main demonstration programs. Both could form the basis of working software, especially OVERLAY which provides the wherewithall to overlay BASIC functions and procedures. This is a fairly wellknown and used technique but takes on new attractions with the speed acquired by using RAM rather than disc. A machine code utility intercepts procedure or function calls, chooses the correct routine from a library in the module and places it into the program at the appropriate point, then returning control to BASIC, which knows no different. Clare's procedure library utility, Profile, might provide a useful alternative when installed in the PMS module.

The WINDOWS display makes impressive use of the extra RAM to provide fast graphics in the Macfadintosh style. It goes to prove that if BBC owners want this style of software front end then it is available to them. It would certainly make sense for current mouse software to try and utilise the extra RAM for Mode 0, 1 and 2 displays. The problem for software houses of course is how many? That doesn't stop programmers giving their own software developed for the module a bit extra on screen. And WINDOWS makes this possible with BASIC procedures to open and close windows for graphics and text display.

Further help for programmers lies in the manual with examples of how to read and write the module, illegal and Tube compatible methods, BASIC and machine code. It suggests studying ROS and PBUFF to see how machine code can run *in* the module!

RAMIFICATIONS

The benefits of PMS RAM are immediate for those who press their BBCs into service for text processing and programming. The module is selling into industrial and scientific establishments where the speed of RAM as opposed to discs is vital, in logging measurements and controlling experiments, in developing and testing firmware.

The manual is to be updated to incorporate the demonstrations and new utilities. Perhaps RAM]DSK could be made more friendly. The source programs are all supplied so modifications to suit individual taste are a possibility. And PMS are going to support owners of the module with regular updates of software at media and packaging only prices. There are already new possibilities in the pipeline.

I found it very useful, for instance, in writing this article. Auto backing up was going on every five minutes without even the whirr of a disc drive. The first draft made its way to the printer while I started on the revisions. I could send it to the office via the modem while I choose some illustrations but in fact, on this occasion, I will switch off my BBC, disconnect the module and pop it into my briefcase. The sooner to be on the pages of A&B Computing.

If you would like to find out more about the PMS 64K RAM Module then ring Permanent Memory Systems on 03552 32796. They can be found at 38 Mount Cameron Drive North, St.Leonards, East Kilbride, G74 2ES.

INTEGRATION—THE FACTS

INTER-SHEET and INTER-CHART are the first of a new generation of integrated packages for the BBC Micro. They are more powerful and much easier to use than any other similar application program.

The ROM-LINK concept is unique to this range of programs; it ensures that all ROM-LINK programs are totally compatible and truly integrated.

The word 'integrated' has been much abused recently. In our eyes 'integrated' should mean that all programs in the range should share a common data format, that it should be possible to transfer data directly from one program to another (not just by saving and loading) and it should be possible to have multiple programs resident in the machine at once. The integrated programs in the ROM-LINK series are the only ones that can do this.

The ROM-LINK system allows up to 16 packages to be resident in memory at the same time. It makes no distinction between any of the packages, so that you could have up to 16 separate spreadsheets stored in memory or (more usefully) a few spreadsheets, a few graphs and a couple of documents being word processed. It is a simple matter to transfer data from any package into the one in current use, and because the packages are resident in memory, the data transfer is very fast. It is also possible to use data from other packages without actually transferring, thereby saving memory.

Put simply, this means that INTER-SHEET can use multiple spreadsheets and use data from one sheet in another sheet, and INTER-CHART can plot graphs using data taken directly from a spreadsheet. When INTER-WORD and INTER-BASE are available then these will be able to use data from the other packages in the same way.

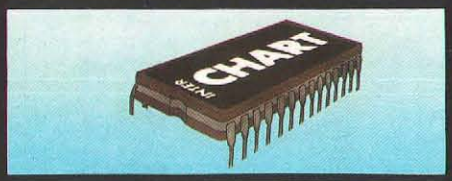
INTER-CHART

FACTS

The chart and graph plotting program

This general purpose graph and chart plotting program allows the user to display data in the most easily understood manner—graphically.

Two quite distinct sorts of graphs are catered for. 'Labelled' graphs that always have fixed increments on the X axis like yearly sales figures, and 'scaled' or scatter graphs. These allow both the X and Y values to be specified.



Data can be entered into the system from three distinct sources: Typing at the keyboard is the simplest method, but data can be read from ASCII files produced by WORDWISE, VIEW or a programming language such as BASIC. Because INTER-CHART is ROM-LINK compatible it can also import its data directly from other ROM-LINK packages such as INTER-SHEET.

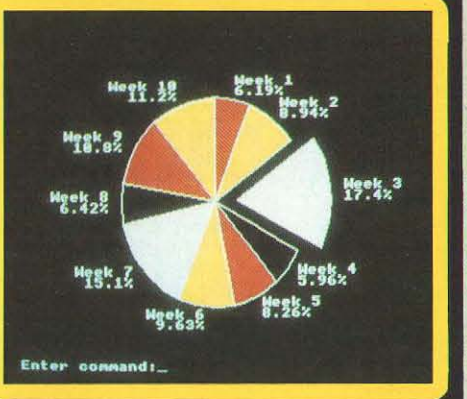
Once the data has been entered a straightforward editor allows any previous data to be altered or new values to be inserted at any position.

TEXT WITH GRAPHS: Labels are normally printed along the X-axis for lines and bars. Pie charts have their labels and values printed around the chart. These may be optionally switched off. With a pie chart the values may be replaced with a percentage figure. All graph displays can have additional labels or messages printed at any position on the screen using a powerful label editor. This allows headings to be easily added to graphs. These labels are stored with each graph when saved.

COMPATIBLE WITH: All shadow RAM boards and the 6502 2nd processor. However the ROM-LINK facilities are not available when using a 2nd proc., these include multiple INTER-CHARTS and direct data transfer from other ROM-LINK programs.

PRICE: £36.80 incl.

INTER-CHART allows up to 16 graphs to be held in memory at the same time as one 'package'. Each graph is named and, once entered, any named graph can be selected from a simple menu display.



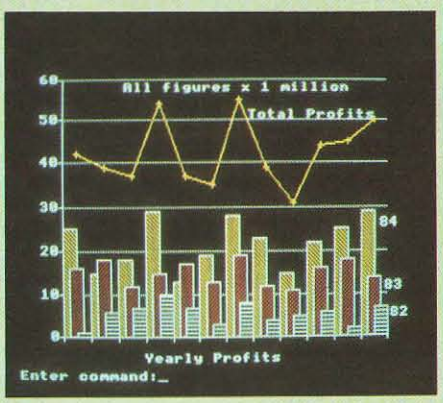
SCREEN DISPLAY: The graphs can be displayed in any of the BBC's graphics modes. The colours of the bars, lines or segments can be altered in a simple table of colours. This also allows special striped colour combinations and for bar charts a variety of cross hatching styles is available. It is also possible to use a VDU19 statement to alter the colour palette.

DISPLAY OPTIONS: Where appropriate (not pies, etc) either axis can be displayed as a log or linear scale, and there is an option to display dotted graduation lines on either axis. Pie segments and bars can be highlighted (segment 'pulled out').

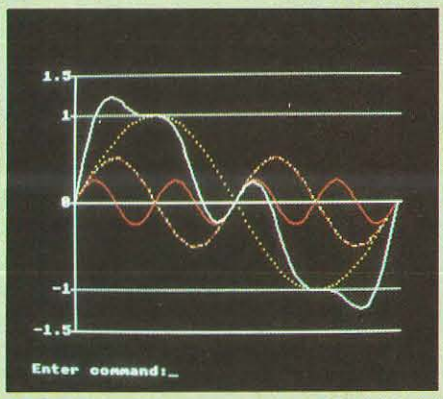
PRINTERS: INTER-CHART has a built in screen dump routine for all EPSON matrix printers and compatibles. It is possible to use other screen dump routines from within (or external to) INTER-CHART for other printers or plotters.

FILING OPERATIONS: Any named graph can be saved or loaded individually to disc or cassette, or all graphs can be saved together in one file.

GRAPH TYPES: INTER-CHART supports the three main graph types, lines, bar charts and pie charts. Scatter graphs are also catered for and can be displayed as points or lines.



SCALING: INTER-CHART automatically scales the graph display to cover the full range of values entered. INTER-CHART can cope with numbers in the range 2E-39 to 2E+38. It is possible however to force the scale independently on the X and Y axis to show any specified range, including the option for log scales on either or both axes.



FUTURE RELEASES

Both INTER-SHEET and INTER-CHART are available now. INTER-BASE and INTER-WORD are under development at the moment, and are likely to be released in early 1986. Both are fully ROM-LINK compatible so that they will be totally integrated with the others in the range.

INTER-WORD is an 80 and 105 column word processor (not 40 columns). This will be a 'WYSIWYG' word processor showing the page on screen exactly as it would appear on the page, including page breaks, headings/footings and centered lines, justification, margins, etc. It will also handle 'continuous' documents on disc up to the capacity of your filing system.

Like all of our products this will be smooth and very fast in operation, and of course very easy to use. Editing operations will be similar to WORDWISE but this is not totally WORDWISE compatible. There will be an exchange arrangement available to WORDWISE owners—more details available nearer the release date.

AVAILABILITY: Winter 85/86
PRICE: Approx. £49.00 + VAT

INTER-BASE will simply be the most powerful database available. This will operate with all disc filing systems and have a very large data handling capacity. It will contain a powerful BASIC-like programming language with full string and mathematical operations.

This will work at 2 levels, a simple user interface will allow card-index like operations but, via the programming language, the system can be easily tailored to the individual needs. Up to 250 fields per record, each field up to 32K long. This would enable the user to store complete documents in the database. Variable length fields and records.

AVAILABILITY: Early 1986
PRICE: Approx. £59.00 + VAT

INTER-SHEET

The electronic spreadsheet program

FACTS

FILING OPERATIONS: Load and Save entire sheet. Save specified part of sheet. Load part of sheet to any sheet position. Spool sheet to file. Save all ROM-LINK packages. Load all ROM-LINK packages.

IMPORT/EXPORT: Export any part of sheet for use in word processors. Import sheets created on word processors. Import data directly from other ROM-LINK packages.

PRINTING: Print all or specified part of worksheet. Print with or without sheet borders. Print with or without held lines (windowing). Sheets titles can be printed and control codes can be sent to the printer. Print all formulae. The pound sign code can be altered to a sequence of codes for different printers.

AUTO-STEP: Whenever an entry has been made the cursor can automatically move one box up, down, right or left.

CELL DISPLAY: Any box (cell) can be displayed left or right justified. Labels can also be centered. Numbers may be displayed in three formats, exponent, fixed or general format. The number of digits is controllable from 0 to 10.

LOCKING: Any box, row or column or all boxes can be locked to prevent accidental alteration.

LABELS: Formulae can refer to other boxes in the normal way (e.g. A1 or G5) but can also refer to boxes in a much more meaningful way via labels (e.g. JAN SALES), the box referred to is at the intersection of the column and row of the specified labels. Labels are not restricted to being placed on the borders of the sheet only.

REPLICATION: Any box or area can be copied to any other box or area on the sheet, giving total flexibility. The replication is very fast—see comparisons. Single key operations for inserting or deleting a row or column. Formulae can automatically be altered, if desired, relative to their new position.

OPTIONS: A variety of spreadsheet defaults can be altered, such as the default box justification, number format, sheet area. Screen colours are easily alterable.

WINDOWING: A unique system that allows any row or column (or multiple rows and columns) of the sheet to appear at a fixed position on screen, still allowing the sheet to scroll underneath. Far easier to use and more flexible than other windowing systems.

INTER-SHEET and the rest of the ROM-LINK series belong to the 2nd generation of application programs for the BBC Micro. They are larger, more powerful and easier to use than any previous products of this sort.

ROM-LINK

INTER-SHEET offers many quite unique features.

1. The fastest spreadsheet—see comparison box
2. The largest capacity—see comparison box
3. 40, 80 and special 105 column screen mode
4. Multiple spreadsheets in memory at once
5. Data transfer between sheets in memory
6. Data transfer to and from word processors such as WORDWISE and VIEW
7. Menu operation and fully prompting commands
8. Complete documentation. 2 manuals, an introduction and a reference, are supplied along with a quick reference card.
9. Will integrate directly with INTER-CHART and other ROM-LINK compatible products.

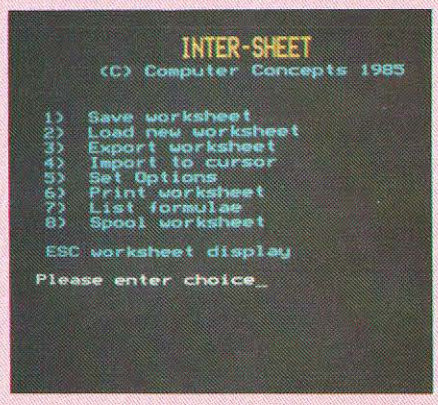
Some further facts about INTER-SHEET:

CURSOR MOVEMENT: WORDWISE-like cursor movement allowing the user to move left or right one box, one screenful or to the extremes of the used sheet area. A further command allows immediate movement to any specified box.

SCREEN DISPLAY: Single key switch for 40, 80 and a special 105 column screen mode. This allows much more of the spreadsheet to be visible on screen at once. The 105 column mode requires no special monitors—if you can read the 80 column mode then the 105 will work perfectly well.

SHEET SIZE: 64 columns by 255 rows. Each box can hold up to 80 characters. Column widths are variable up to 31 characters wide. There is no gap between boxes so labels can continue across boxes.

MATHS: A full range of maths functions is included. There are also additional special spreadsheet functions such as AVERAGE, COUNT, IF . . ., LOOKUP, MAX, MIN, SUM, BOX and SQRS (sum of the squares). Numbers can range from 2E-39 to 2E+38 and can be printed to 10 significant figures.



ROM BASED SOFTWARE FOR THE BBC MICRO

The first published review of INTER-SHEET says:

"... It offers higher capacities than any previous Beeb spreadsheet, but it runs significantly faster than Lotus 1-2-3 and Symphony on the IBM-PC."

"INTER-SHEET is quicker, has more features and is cheaper than either of its competitors. If the remainder of the "Inter-" series are as good as this Acornsoft and the rest better look to their laurels. Software of such quality can only make the Beeb even more attractive as a business machine."

Gordon Taylor—A&B Computing Sept. 85

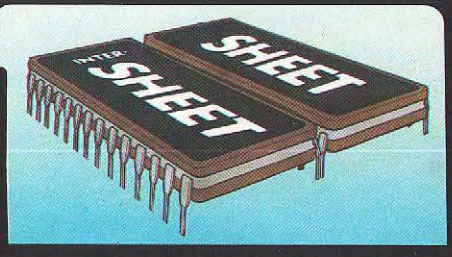
PRESS

Spreadsheet comparisons

Times in seconds	VIEW-SHEET	ULTRA-CALC 2	INTER-SHEET
Replicate SIN(PI) into boxes A1 to Z50	66	65	7
Recalculate the above	28	33	5
Number of bytes free with above sheet	4664	1415	8400
Price (Exclusive of VAT)	£52.00	£69.13	£49.00

COMPATIBLE WITH: Shadow RAM boards such as the Aries B20 and Watford electronics 32K RAM board. Because the ROM-LINK system needs to switch ROMs INTER-SHEET is not compatible with the 2nd processor, although a HI version of INTER-SHEET will be available for the 6502 2nd processor (No ROM-LINK facilities and no 105 screen mode). INTER-SHEET is supplied in 2 ROMs, but if you only have 1 socket then only the ROM-LINK facilities are lost (multiple spreadsheets etc).

PRICE: £56.35 incl.



For the BBC MICRO
Available from all good software dealers or directly from ourselves

Computer Concepts

Gaddesden Place, Hemel Hempstead, Herts HP2 6EX (0442) 63933

Access VISA

Top of the Class

Des Thomas

Two of the Best for 85.

"A good primary school classroom should be an exciting place where children have the opportunity to do stimulating things. In summer, perspiring teachers can be seen shepherding their charges around zoos, castles and other places of interest. The object of all this activity is the enlargement of experience and increase of motivation.

It is also possible to claim that the presence of a micro in a classroom can have much the same effect. It can be used in a wide variety of ways, providing a diversity of experience and differing forms of motivation. One use of the micro is in simulations.

A computer simulation is the creation of a "world" in which the children are required to function, limited by some of the constraints present in the real world which the creators of the simulation have elected to copy. In doing so the children are expected to exercise a variety of skills and experience different situations in which they are required to act." Ian Whittington in *Micros in the Primary Classroom*, Ron Jones (Editor), published by Edward Arnold.

What's your favourite memory of your days in the primary school? That's a question I often ask when ex pupils pay us a visit. The answers are many and varied, but usually they have to do with a particular teacher, something a little out of the ordinary – experiences on a school journey, an achievement – winning an individual or team competition, or succeeding in an activity which previously had been found to be difficult – learning to swim. One answer I received recently took me by surprise – "Going round the world". Well, we had been on board SS Uganda, but that was a Baltic cruise. Then the penny dropped, the lad was talking about the computer simulation, *Round the World*. It had been the first time he'd used a computer and had been so fascinated that he'd persuaded his parents to buy him one and now, not surprisingly, he can teach me a thing or two!

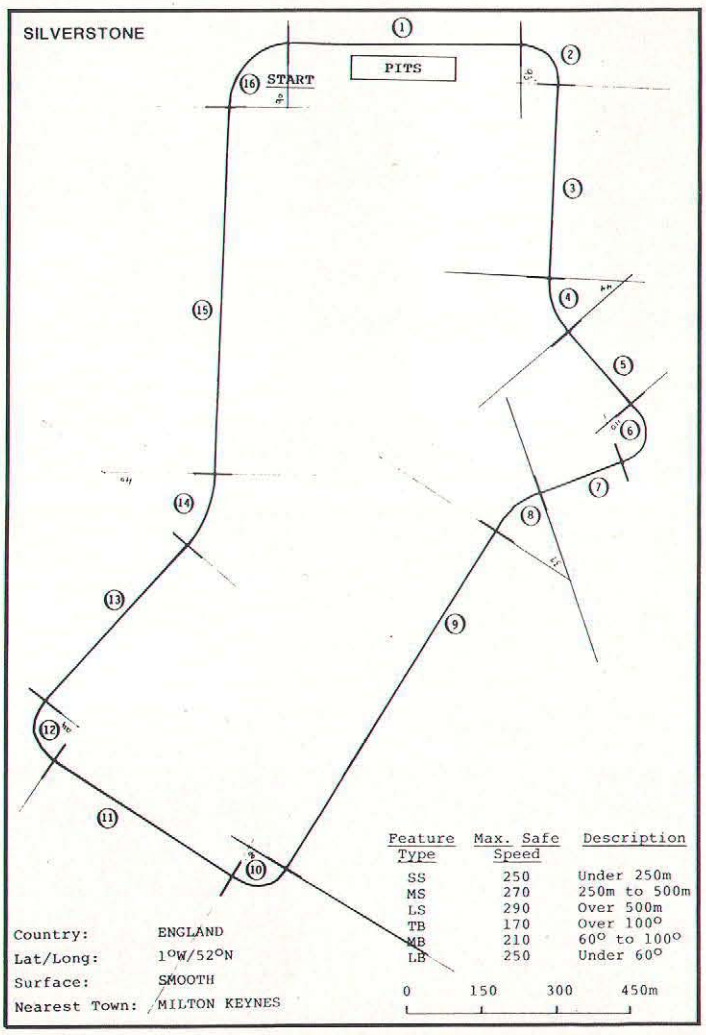
Computer simulations are used in most technical and business spheres – the training of airline pilots and spacemen, weather forecasting or business management. Wherever fast reactions are required for changing situations, the simulators can offer an economical way of training in a controlled environment. In schools, where more emphasis is being placed on teaching children to cope with a rapidly changing world, such skills as decision making, fluency of thought, and formulating, testing and refining hypotheses are also important. The use of simulations offers an ideal tool or, as one writer described, "...a window through which children can observe the world outside the classroom – and in some cases that world looks out on the world of the future. They stimulate the child's imagination with useful spin offs on other school work which requires imagination and flair. Their use of facts, figures and graphics render them multi-descriptive, forcing the child to use several modes in order to comprehend the situation and scenarios being set: these are continually changing depending to a large extend on the decisions being taken."

In primary schools where a topic work approach is used, simulations can be of particular significance. They offer multi-disciplinary activities in which there are often no right or wrong answers, enable youngsters to learn from their mistakes and encourage them to exchange ideas and help one another; they give opportunities for creative language work, motivate investigation and research away from the computer and good record keeping; and they provide a more stimulating method of learning, where skills and concepts are grasped because they are needed and remembered because they have been practised in meaningful situations.

Programs such as *Mary Rose*, *Saqqara* and *Adventure Island* have become early classics – all ideas developed by those kings of simulation, Barry Holmes and Ian Whittington – and have been followed by *Dinosaurs* and *Archaeology* among others. *Educational Computing's Top 20* for 1984 included five simulations: *Into the Unknown*, *Archaeology*, *Osprey*, *Viking England* and *Fletcher's Castle – no room for Flight Path* or *Kontiki*. *Surprisingly, the educational Top Ten in a recent edition of another magazine didn't include one simulation, but maybe that's because they've concentrated on educational software for the home market. If that's so then, mums, dads and children, you don't know what you've been missing.*

Certainly, I've got two simulations I expect to be well-placed in any list of educational programs for 1985: Cars – Maths in Motion and *Suburban Fox. Perhaps not surprisingly, one member of the Holmes/Whittington team is involved in the first, while the second originates from Newman College. I had expected my title to be "Simulations: Three of the Best" but, unfortunately, Police: Language in Action* from CSH is still having its final touches added. It's likely to be available by the time you read this. Previews I've seen suggest that it's going to be one to look out for in 1986 – further information as soon as a copy's available.

Fig.1 Silverstone Circuit Plan



WORKSHEET 1

Name	ACORN	Driver's Name	Fiery Fred
Team No.	1	Temperament	Daring

Circuit	Silverstone	Country	England

Laps 20

STEWARD'S REPORT - THE WEATHER

Time	Description
1400	Heavy rain
1415	Rain
1430	Occasional showers
1445	Dry
1500	Warm & dry
1515+	Occasional showers

WORKSHOP ADJUSTMENT

Engine Tuning	6	7	8	8							
Aerodynamic Downforce	5	4	3	4							
Suspension Adjustment	0	0	0	0							
Gear Box Ratios	7	6	5	6							

SAFE SPEED CALCULATION

Feature Category	Max. Speed Bee Plan	Performance Percentages									Safe Speed		
SS	250	97	97	97	97						243		
MS	270	89	92	94	92						248		
LS	290	82	86	90	87						251		
TS	170	84	80	76	79						134		
MS	210	95	92	83	91						191		
LS	250	97	97	97	97						243		

Fig.2 Worksheet 1 — A back marker on the grid, i.e. the writer!

CARS – MATHS IN MOTION

Let me twist my first question: what's your least favourite memory of primary school. To some extent your answer is likely to reflect your age. Sorry if that hurts but, if you're my vintage, you're likely to be thinking about preparations for "the scholarship" or "the eleven plus" and those pages of long division, converting pounds to tons and division of fractions — do you know to this day why you change the division sign to multiply and invert the fraction?

Perhaps it's not surprising that lads who left the classroom barely able to subtract three from five had no difficulty when placed in front of a dart board at their local. I'm not sure whether it's the winning of the game or the thought of the free pint at the end, but the motivation's there! At my old school there used to be a regular roundup of the school phobics at the local snooker hall, but I bet they knew a great deal about angles and relative motion. You won't have to think too hard to come up with other sporting activities which could have a spinoff in the classroom.

Critics of simulations argue that they should not replace first hand experiences. True, but advantages include participation in activities that would normally be too expensive or dangerous. Take for example Grand Prix Racing. How would you like to prepare and race a car around Silverstone? Not very likely you may think considering the cost of a set of tyres let alone the car! Sit a nine or ten year old — or take yourself back in time — in front of a computer, load *Cars – Maths in Motion*, and let the countdown begin.

The package comes complete with authentic plans of ten well-known international tracks and worksheets which enable the teams to record information to help in the setting up of the cars. Each track has up to 20 features — long, medium or short straights, tight, medium or long bends.

The teams have to decide which category each feature on the circuit falls and enter their results on the worksheet.

To decide on the type of bends, the feature markers on the plan have to be extended until they intersect and the angle measured or estimated and compared with the given limits.

In order to make decisions about the amount of fuel needed and whether a pit stop will be required for additional fuel or to change the tyres, the track must be measured on the plan and the scale used to calculate the length of the race, which will depend on the number of laps the "track officials" decide upon.

Before a race each team has to adjust its engine, aerodynamic, suspension and gearbox. These will depend on the track — has it got long fast straights or a lot of tight bends, and the condition of the track surface. The car's performance for each feature is produced as a result of these adjustments and is expressed as a percentage, and the safe speed is found by multiplying this figure by the maximum performance speed.

Have you noticed the number of mathematical concepts used so far? If the team's not satisfied with the performance figures, it can always go back to the workshop and adjust the variables until the optimum figures are obtained or the time for the practice lap(s) has arrived. By the way, the data for each track is genuine too, being based on information provided by Team Lotus GB Ltd.

Now for the test — the practice lap! The team have already nominated the driver — is it to be Saucy Sue the Maniac or Cautious Colin? The results are displayed as a driver's report, detailing any problems encountered feature by feature, eg satisfactory, 1 in 20 chance of crash, engine overheating, engine blowout — rather expensive! If the practice lap is successfully completed, a lap time appears on the screen and, if it's the fastest time recorded to date by the team, it is counted as the team's qualifying time and determines the position on the starting grid. Again, if the team's not happy, the car can be returned to the workshop for some overtime.

Surprising how happily the mechanics give every spare minute to prepare "their" car — try that with a page of maths!

Race Day arrives. Safety Checks — forget one at your peril, choice of tyres — check the weather forecast on the worksheet, confirm pit-stops, if any, and amount of fuel to start the race. Of course, there's not much fun in racing one car, so the program allows up to 15 teams to participate. This is organised in the standard Cambridgeshire Software House format, which means that all the work not requiring a computer can be done away from the keyboard. When all the teams have prepared their cars, the race itself can be run. This requires no input from the groups as they'll have already entered their strategy, but worksheets are required to note the progress or otherwise of each group's car.

The final grid positions are displayed and a countdown to the start of the race begins in Grand Prix style. Whilst a lap is in progress, the lap report screen is displayed which shows a map of the circuit and reports any incidents which occur. An end of lap screen shows the current position, the lap speed, total time and lag time for each car.

The race is run over the number of laps originally selected and at the end the final positions and times are displayed. A message of congratulations to the winning team ends the program and the class/family left to work out their team championship positions. I must admit, I quite expected someone to complain that they wanted to see their car racing round the circuit, but they were too intent on reading and noting the times and incidents and using their imagination!

I mentioned earlier the excellent handbook with ideas for developing the project. I wonder how many of you know where Kyalami is, how to get there, or how much it's likely to cost? What do you know about the history of the motor car or racing cars? Where does fuel come from? How do gears work? What does streamlining mean? If you're still short of ideas, there's that very good BBC TV series *Craft, Design and Technology*, to help, and don't forget that box of Lego Technics!

If that sounds a little like a full term's project for a class, you're probably right. Nevertheless, if your children — don't forget it's equality of the sexes these days —

CONTINUED OVER

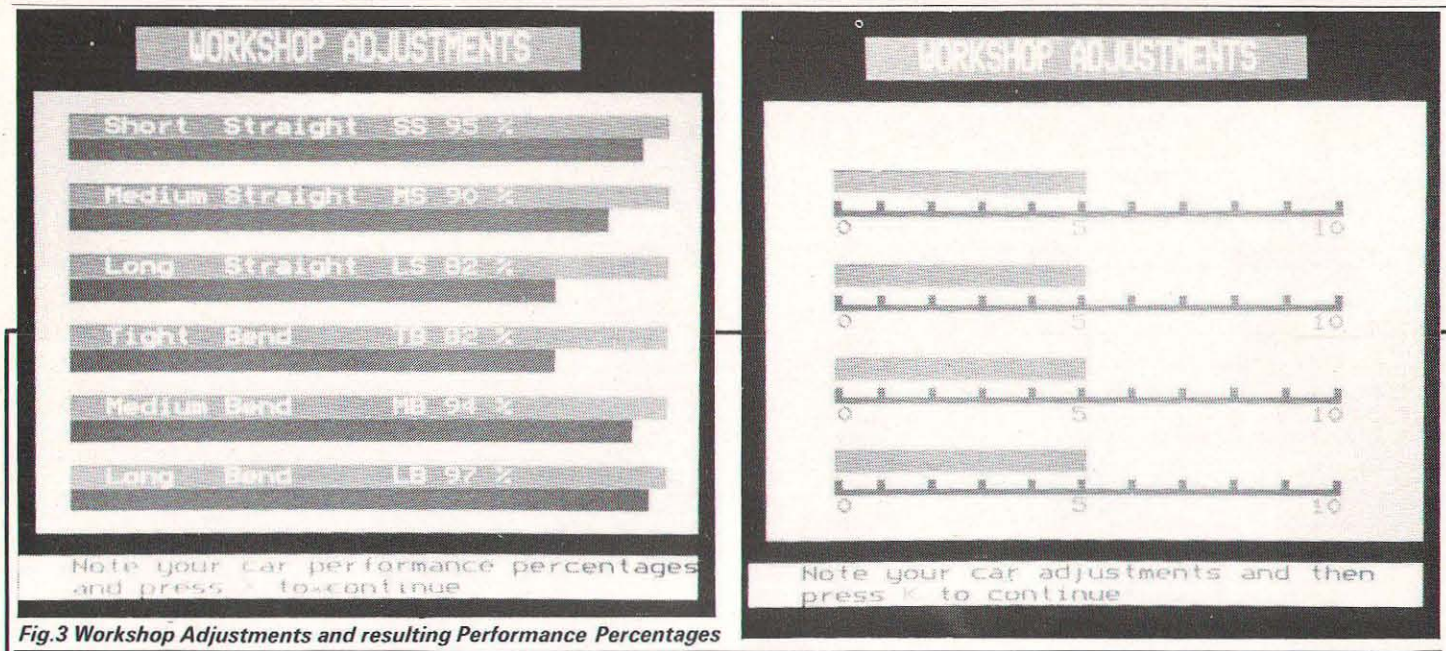


Fig.3 Workshop Adjustments and resulting Performance Percentages

aren't keen on doing that extra maths homework, don't go out and buy the first jazzy drill and practice program you find on the shelf, because really you're giving more of the same medicine. Take them to a race track, or watch a meeting on TV, then produce this program. They'll be making all those calculations and genning themselves up on background information before you know it.

A few words of warning! Don't lose too much beauty sleep getting in extra practice laps so that you can beat them. Just in case I've overemphasised the motivational effect of this program on those who find difficulty in settling down to work, let me assure you that even Sir Keith would be hard pressed to fault it — it's very much a case of "fine tuning", so even those whizz kids will be kept on their toes.

SUBURBAN FOX

So, you don't like getting your hands dirty, you don't fancy all that noise, and you're definitely not prepared to let your little darlings loose on tuning your C registration BMW — you're obviously not a teacher! Of course, if they get hooked on a simulation, you don't know where it'll end — a gokart with your lawnmower engine is the least you can expect! If a stroll through England's leafy glades with a camera slung over your shoulder and a pair of binoculars in your hand is more to your liking, ie you're an environmentalist, then *Suburban Fox* will be an interesting possibility — although I'm not sure you'll be strolling for long with a pack of hounds or a farmer with a shotgun on your trail.

I believe I'm right in saying that this is the first simulation published by Ginn that hasn't been developed by Cambridgeshire Software House, ie Holmes and Whittington. It comes from David Jackson, who conceived the idea while attending the computer course at Newman College, and there's been a great deal of talk about it for over eighteen months now. It was suggested as "a definite contender for the program of 1985" some six months before it was published.

Perhaps it's because enthusiasts have had to wait so long to get hold of the program that it's had something of a mixed reception among my contacts. One reviewer suggested, "This is an outstanding role-playing simulation...Suburban Fox really shows the sort of quality software we can expect to use with our micros". A few colleagues have expressed disappointment and consider there's too much of a games element in it for their liking. In fairness, one must say straight away that in his introduction David Jackson states quite clearly, "This program is designed to appear as much like a game as possible, yet it is a game that requires a carefully thought-out strategy to be played with any degree of success".

So should it be considered as a game or a simulation? Ian Whittington, in the article I quoted from in my introduction, clears up this point very neatly: *"A simulation attempts more than a game. It is true that its use without thought can reduce it to the level of a game....If this happens then much of its power as an educational tool will be lost. The simulation attempts to engage the participants in the consideration of the problems existing in that "world" as simulated."*

Suburban Fox certainly does that and a great deal of thought has been given to the production of the package, so if it is treated at the lower level, then it'll be an opportunity missed.

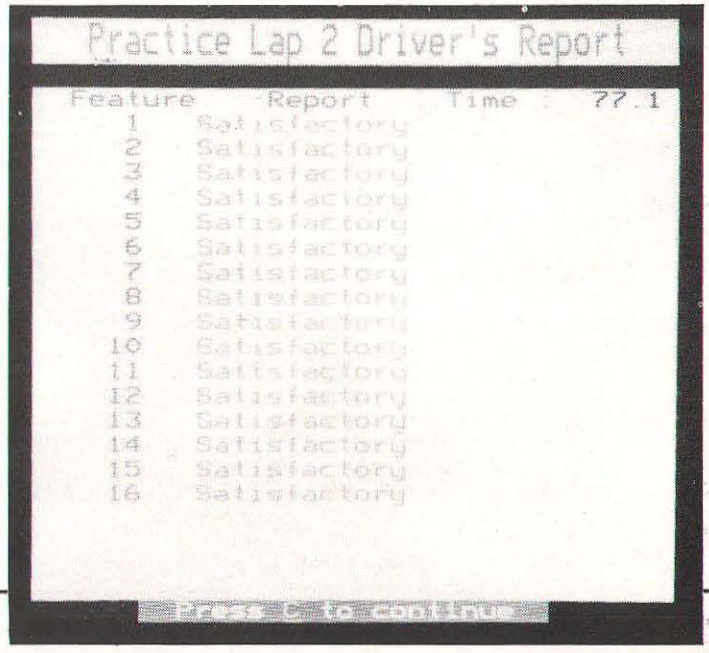
"Wiley had slept curled up, cat-like, in a bare hollow in Theaker's Wood. It had been a restless sleep, and more than once he had opened his eyes, pricked up his ears and pointed his wet nose into the wind to check the sounds and scents it carried. Each time his senses had told him of no danger and he'd settled back into uncomfortable sleep. The pale sunshine slanted through the branches. Wiley shivered as evening brought a chill. He yawned, stretched and sniffed again. He was hungry. It was time to hunt..."

So begins a delightful story, nicely illustrated with black and white photographs, specially written by David Jackson for use with the computer program. Anyone reading this book, which contains considerable background and/or vital information for playng the simulation, cannot fail to develop empathy, and it provides an excellent introduction to the program. Two copies of this and another book — The Fox (Cambridge University Press Pole Star Series), a large full-colour poster, 12 thematic activity cards and first class teachers' notes provide a resource component which forms an essential part of the package.

The program is aimed at children from the age of 10 upwards, working in small groups of three of four. It places them in the role of the modern fox and the intention is to give them first hand experience of the problems and hazards that a fox might encounter. Wits, cunning and senses have to be used to survive for as long as possible and, hopefully, find a mate.

The program has a very friendly "setting up" procedure — fairly

Fig.4 Practice Lap — Driver's Report



similar to Cambridgeshire Software House's. The first option allows the teacher and/or group to list the names of those foxes (max. 19) who have not found a mate or been killed and whose current situations are therefore saved on the disc. The second permits a new group to begin the program or a group to call up an already existing fox to continue the programs, ie the sort of procedures which should be standard in all simulations intended for use in the classroom but are so often missing! The information screen, which lists all the input instructions that can be used during the program, follows from this and is also available during the running of the program by pressing I. Typing H for HELP at anytime displays an index of words associated with foxes or connected with the characters the fox may encounter. You are then asked: "Which subject do you want help with?" By typing the name, you are given the relevant page numbers in the two books to enable you to look up the information — a very pleasing touch which will help the development of research skills.

The simulation begins with a fox randomly placed on a blank map — a 10 x 10 grid — with no idea of his surroundings. In the process of finding food, water and shelter, he uncovers a map of the city suburb, complete with all its hazards — farmers, dogs, cats, trains, snares and hunts. As well as the map, the main screen display has four other "components": a box coloured light or dark blue with either a sun or crescent moon symbol depicting day/nightime respectively and an arrow plus number signifying the wind direction and strength; the clock which depicts the number of days, hours and minutes the fox has survived — the program finishes if it survives more than 99 days; yellow and blue bars denote the fox's food and water supply; and lastly a text window where any messages are printed.

As the fox moves around the map, various symbols, scenes and dangers are revealed, but he can make himself aware of his surroundings by using his three main senses — sight, smell and hearing. Any significant object that emits a sound which the fox can hear — limited usually to two squares in any direction — will be printed as a message in the text window. The fox can LOOK in any square adjacent to his present position by pressing L followed by one of the directional keys for the eight compass points and, again, anything seen will be shown in the message area. By pressing S for SCENT, which is the most important of the fox's senses, he is able to detect any smell blown up, across and even down wind. Pressing X at any point of the simulation allows the fox to choose whether to eat — assuming he's found or caught food, drink — when at a source of water, dig an earth, carry — any food he finds but doesn't wish to eat, or sleep — a safe place?

Having used his senses to find his bearings, the fox moves around the map by means of the cursor keys. On the main map, the fox can

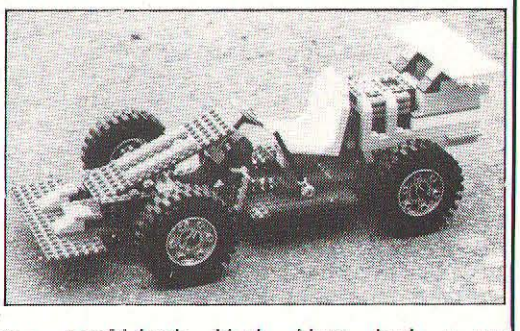
CONTINUED OVER

GRAND PRIX NEWS 20p

1st July, 1985

New William's Special

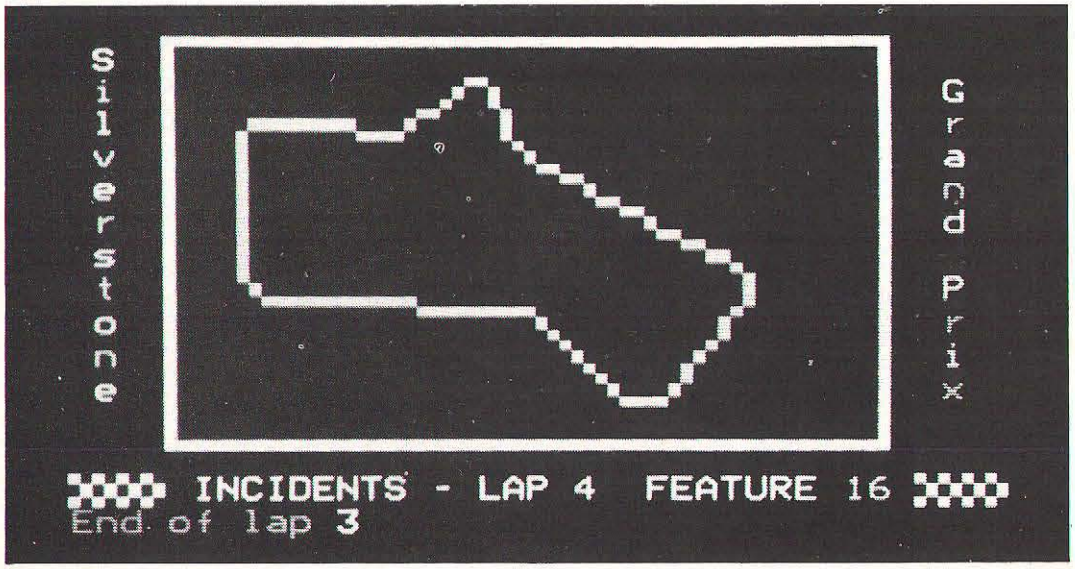
In the early hours of yesterday morning our ace reporter was at the Silverstone track to see the unveiling of a brand new car. It was designed and built by ex-world champion Paul Williams and is said to include several new safety features. The chassis is constructed from a special plastic reinforced by one of the alloys first used in the space program When questioned about performance, Paul was not prepared to give any details. However, Ryan Fleetwood-Jones, who has been engaged as number one driver for the William's team in 1986 was very excited about the team's prospects. He said that he was confident that they had a car that would win the world and manufacturers' championships.

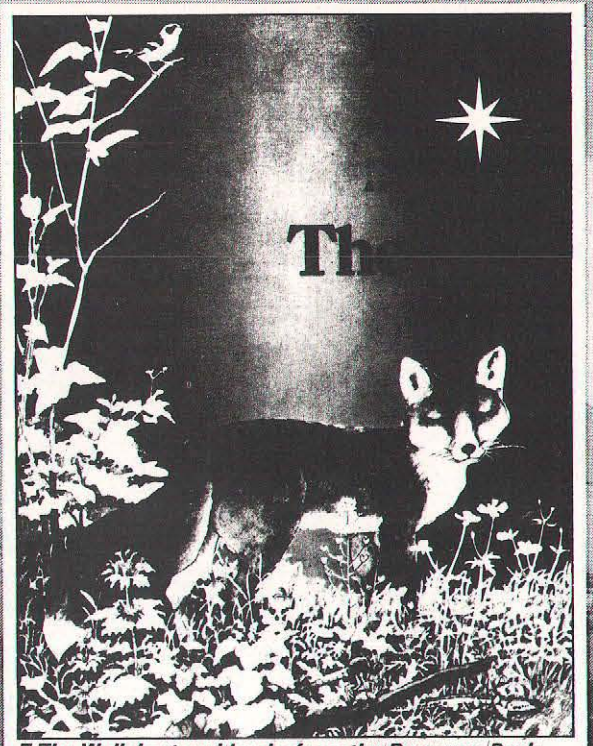


Support British technology

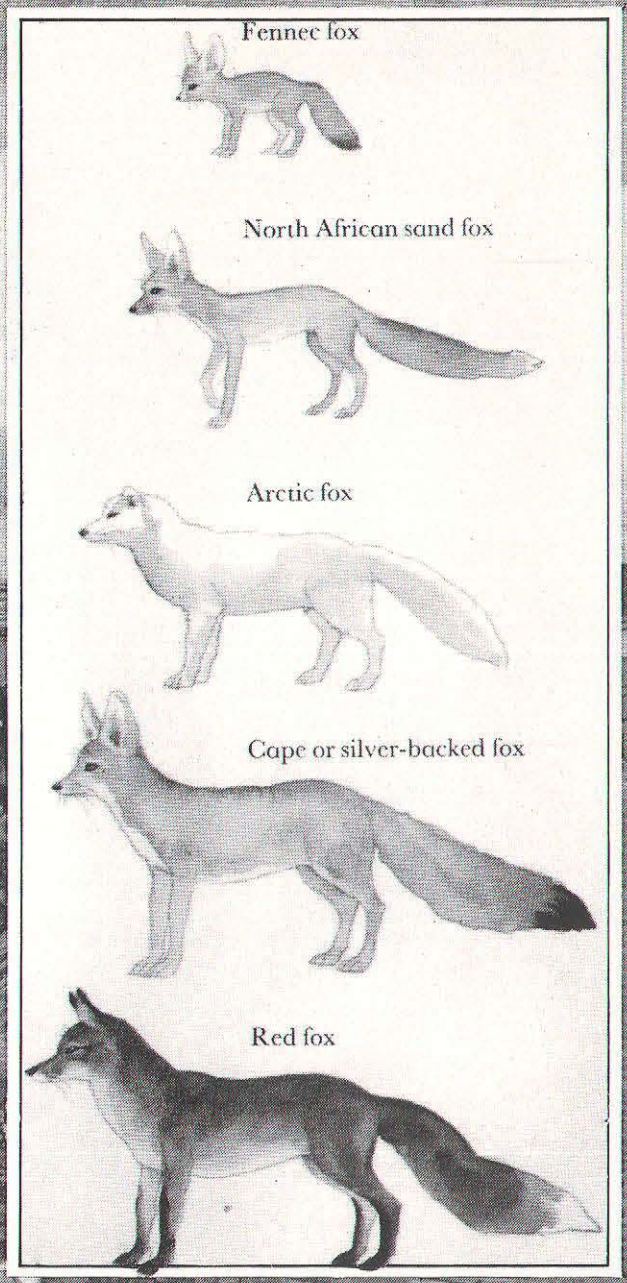
Fig 6.GRAND PRIX NEWS Use of Front Page from MAPE TAPE 2. A photograph of Paul's Lego Technic Model has been placed in the area normally used for drawings.

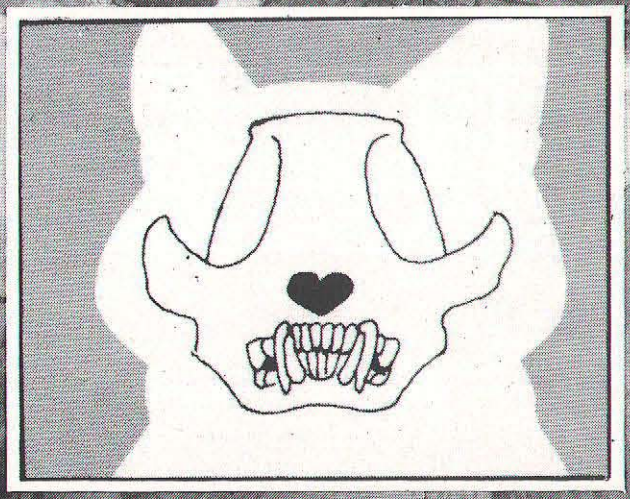
Fig.5 The Race





7. The Wallchart and books from the Resource Pack





CONTINUED OVER

move horizontally and vertically only. When he moves onto a square representing a subsidiary scene – two housing estates, a rabbit warren, a hypermarket, a farm, wood and marsh, which are randomly placed each time the program is run – the main grid is replaced by an enlargement of that scene and now he can move diagonally as well.

Are you a cunning fox? Will you survive the hunt, be shot by the farmer or run over (at times I thought I was back on my Grand Prix Circuit), starve or die of thirst, or will you survive and find a mate? The final paragraph of the introduction to the teachers' notes summed it up very nicely for me: *"In using the simulation, it is hoped that the children will develop a deeper understanding of the environment, not merely with respect to the fox, but to all living things associated with it. This, of course, includes the part played by man. For example, the children ought to develop an awareness of their own senses and compare and contrast them with those of a fox and other living things. It is also important that the children discuss the moral and ethical questions raised by the simulation – for example, man's wastefulness and bloodsports – and that they appreciate the differing opinions and views that people hold. But the overriding aim of Suburban Fox is to provide a meaningful context in which to place children in situations requiring problem solving and decision making and so guide them towards developing ways of uncovering and examining evidence and seeking solutions. It is hoped as well that they thoroughly enjoy themselves while doing so!"* Used in this way by an enthusiastic teacher, it'll create a great deal of interest and excitement and ideas of creative writing and drama, art and craft, history, maths, and natural history will soon be flowing.

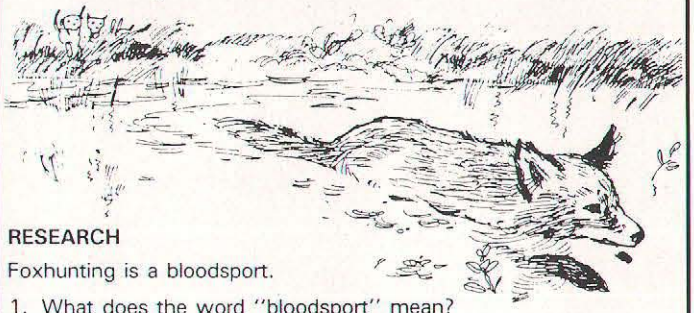
Suburban Fox

Activity Card 6

Foxhunting and bloodsports

THINKING and DISCUSSING

1. In your *Suburban Fox* book, Wiley tries a number of tricks to throw the hunt off his scent. What are they?
2. How does Vulpa the vixen in *The Fox* (Pole Star Book) escape from her hunt?
3. In your *Suburban Fox* book, how does Joe Theaker try to get revenge for his slaughtered chickens?
4. Not all farmers like foxhunts. Discuss some of the reasons why a farmer might not allow a hunt to cross his land.



RESEARCH

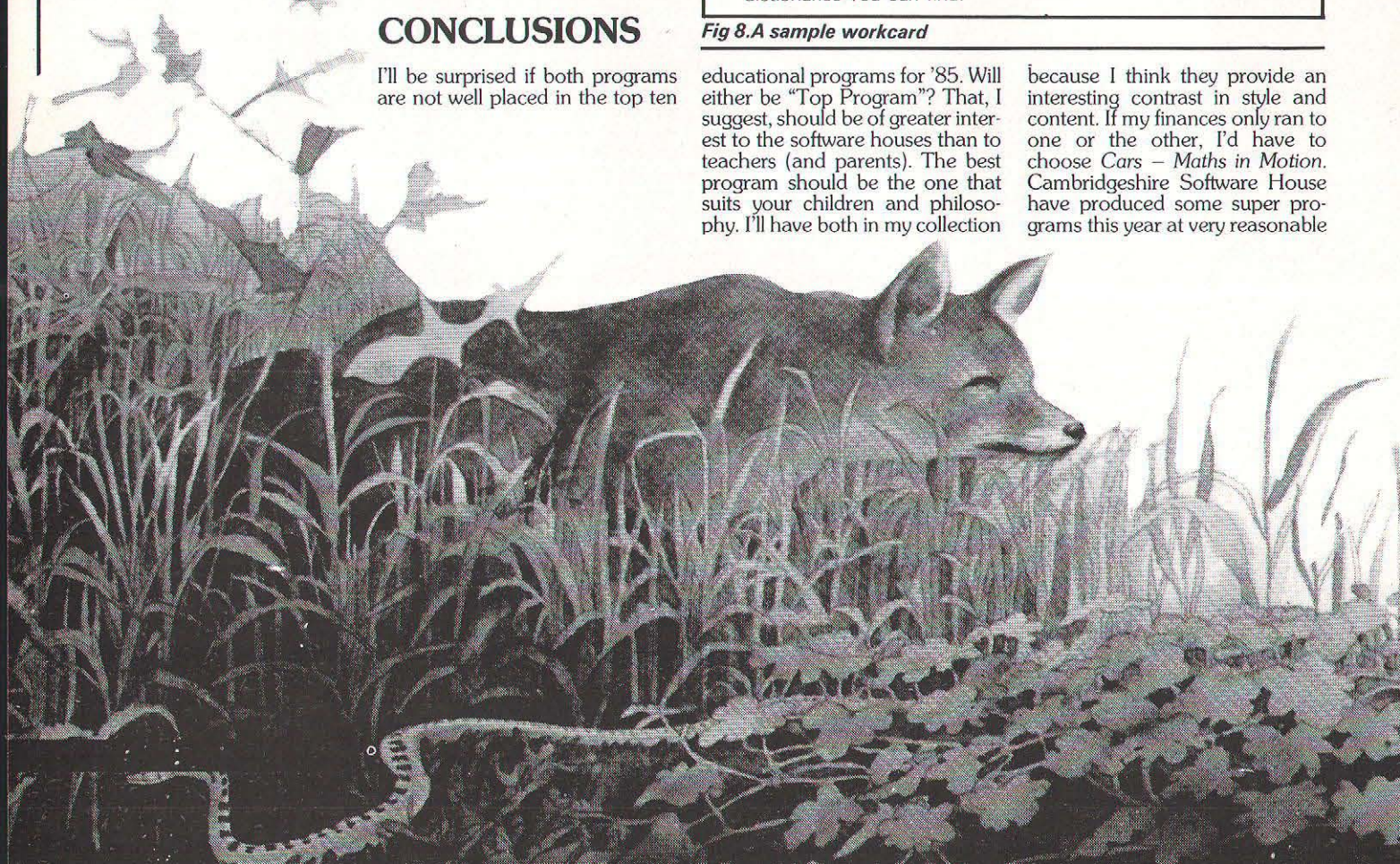
Foxhunting is a bloodsport.

1. What does the word "bloodsport" mean?
2. List five other bloodsports and find out a little about each of them.
   (a) What is the prey?
   (b) Who or what does the killing?
   (c) What is the reason for the killing?
3. Angling is a very popular sport. Do you think it is a bloodsport? Why?
4. In your *Suburban Fox* book, you may have found some strange words about foxhunting (see especially Chapter Two). Write them down and find out what they mean, using the Glossary in the back of the book and any other reference books or dictionaries you can find.

Fig 8.A sample workcard

CONCLUSIONS

I'll be surprised if both programs are not well placed in the top ten educational programs for '85. Will either be "Top Program"? That, I suggest, should be of greater interest to the software houses than to teachers (and parents). The best program should be the one that suits your children and philosophy. I'll have both in my collection because I think they provide an interesting contrast in style and content. If my finances only ran to one or the other, I'd have to choose *Cars – Maths in Motion*. Cambridgeshire Software House have produced some super programs this year at very reasonable



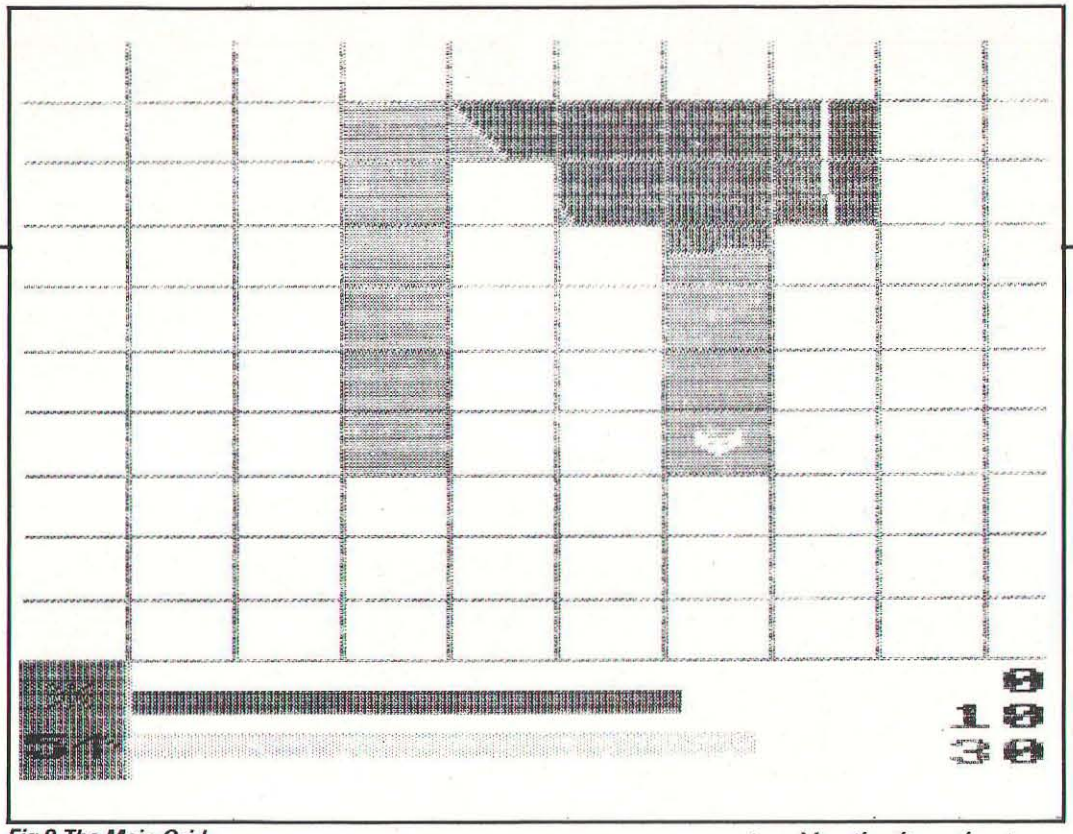


Fig 9. The Main Grid

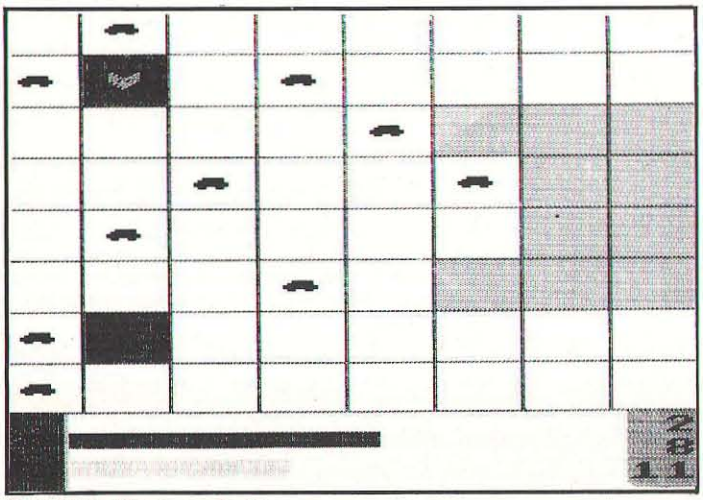


Fig 10. The Hypermarket Scene

prices. My wife often refers to me as the Welsh Jew, so maybe the value for money of their materials has something to do with my choice, but I think it's the possible links with CDT (Craft, Design and Technology) that's got me really excited. Just in case the use of those three letters kills it stone dead for you, perhaps you'd prefer to think in terms of maths, science and craft! On second thoughts, a few pounds has never prevented me from obtaining anything I've wanted for my youngsters and I'm sure most primary teachers are the same. So, is it to be a quick jumble sale, a visit from the school photographer or a chat with a group of supportive parents which will provide you with the wherewithal to give your youngsters *two* experiences they'll remember long after they've left your school?

Teachers and students interested in the use of computers in the classroom would do well to read a very thoughtful dissertation by Barry Holmes, Ian Whittington and Stephen Fletcher — *The Child, the Teacher and the Micro: Using Simulations in the Classroom.* As one might expect, it is full of anecdotes and reflections on their own experiences with children and computers but, as the title suggests, they've got their order of priority right. It's very readable and at just £3.95 it's not going to break the bank. Available from Cambridge Scholastic Services, The Town Hall, St. Ives, Cambs.

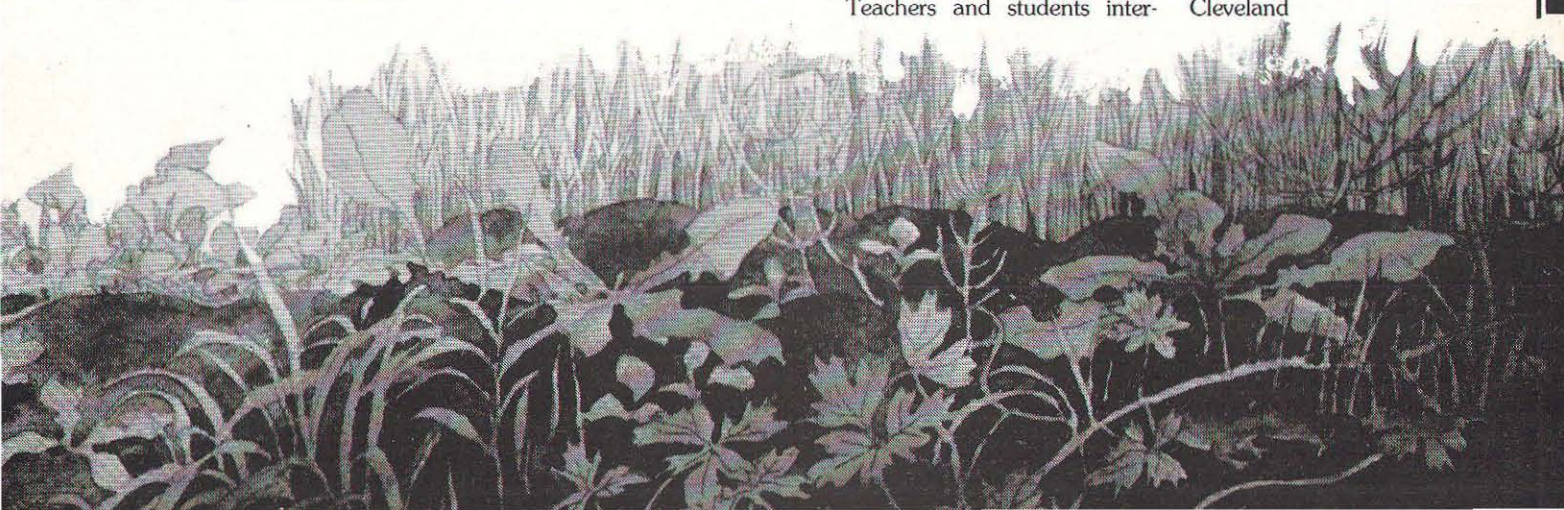
PROGRAM DETAILS

Cars — Maths in Motion (£20.00 + £1.00 p&p +VAT), Cambridgeshire Software House, The Town Hall, St. Ives, Cambridgeshire.

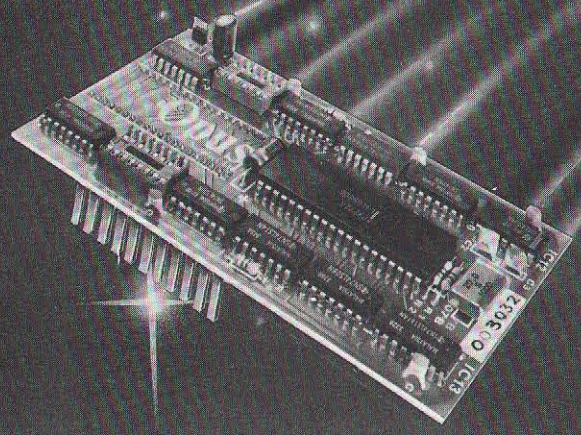
Suburban Fox
Complete: £30.00 + VAT Software: £13.00 + VAT Resource Component: £17.00 + VAT
Ginn & Co. Ltd., Prebendal Ho., Parson's Fee, Aylesbury, Bucks

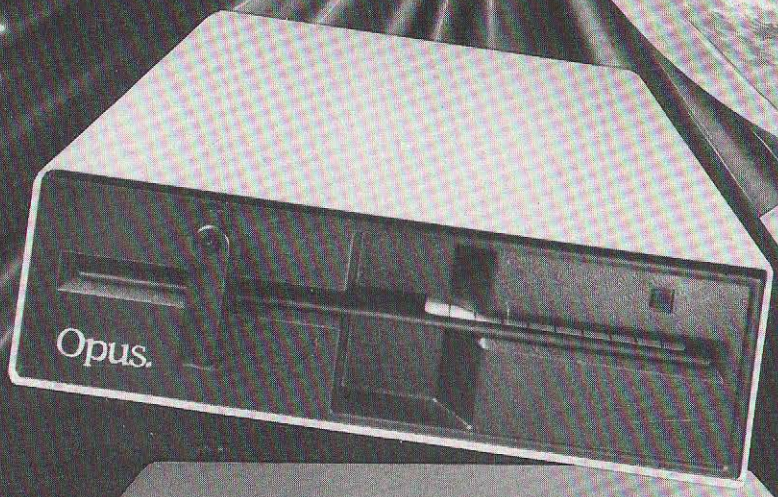
Round the World (£28.00 +VAT), Ginn & Co. Ltd.

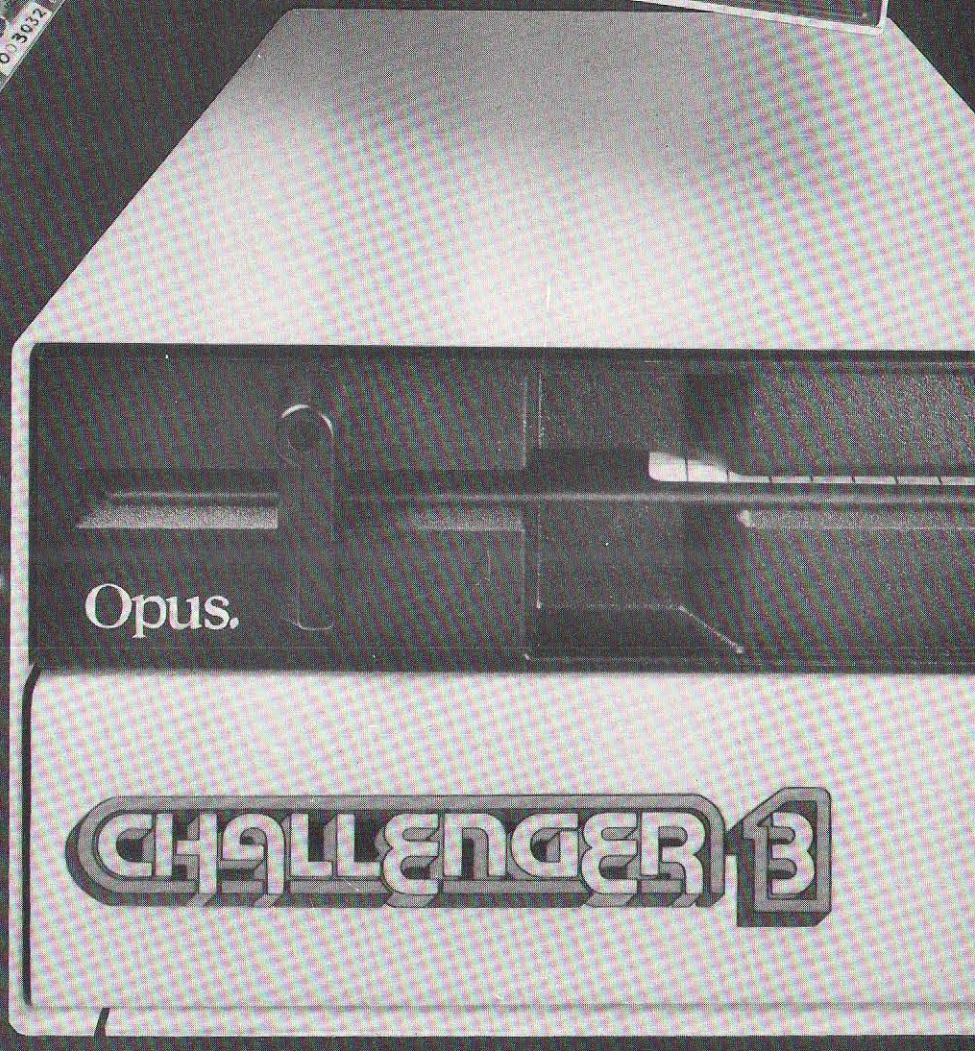
Front Page
MAPE Tape 2. A new version of this program is expected on MAPE Tape 3, which is distributed free of charge to members of MAPE (Micros and Primary Education). Details of membership from: Ann Liddle, Pentland Primary School, Pentland Avenue, Billingham, Cleveland

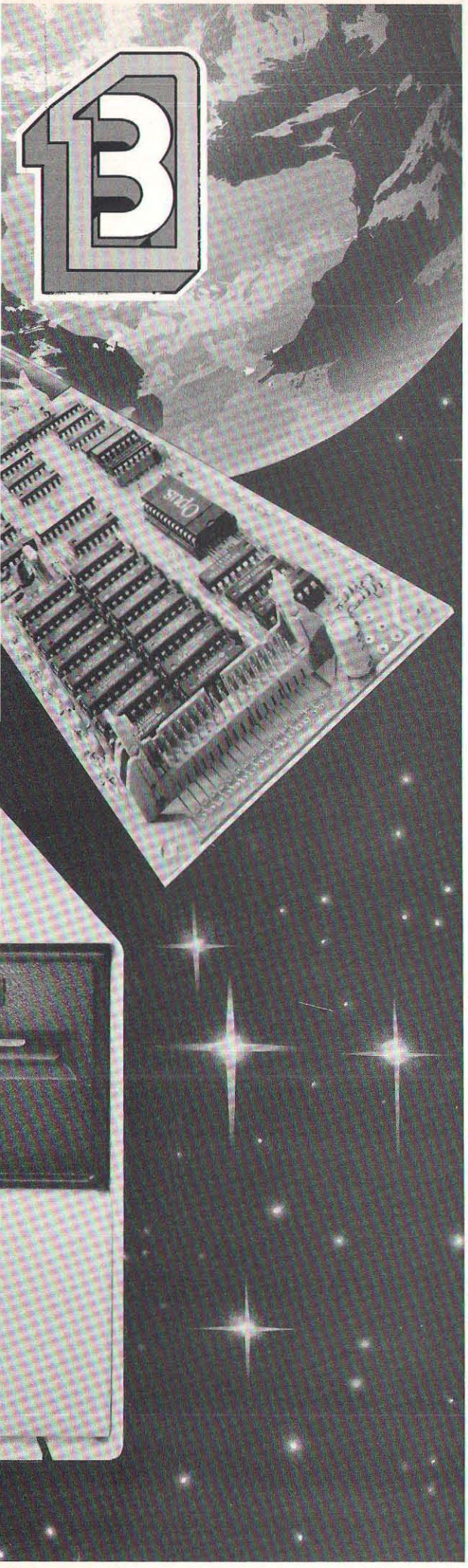


CHALLENGER









Simply light years ahead of the opposition

Opus announce the biggest disc drive breakthrough for years – CHALLENGER. It could make every other disc system for the BBC micro outdated overnight.

An amazing 3-in-1 design has produced Britain's first complete plug-in disc system for the BBC computer. No more complicated interface connections, soldering or track cutting, Challenger is an all-in-one disc system ready to plug straight into your computer. It's the simplest disc link-up ever and a winning combination that no other system can match.

5¼" DOUBLE-SIDED 40/80 TRACK DISC DRIVE

DOUBLE DENSITY DISC INTERFACE

SUPER FAST SOLID STATE RAM DISC – Up to 20 times faster than disc drive and providing a massive 256 K of RAM

All this in a single unit for an amazing £249.95.

And just look at the storage capacity of this remarkable stand-alone system.

Our 5¼" disc drive and built in double-density disc interface – an extended version of the highly praised Opus DDOS, voted fastest system by 'Acorn User' and favourite choice of 'A & B Computing' – combine to give you a powerful 1 Megabyte system. But Challenger has a third dimension, a super fast Solid State RAM DISC for instant access to an additional 256 K of RAM. Effectively a lightning fast second disc drive which can be upgraded to ½ Megabyte capacity for only £49.95. It makes Challenger superior to a dual disc drive, giving faster access and less wear and tear on your drive.

Challenger uses the same disc controller chip as the new BBC B+ computer making it compatible with all leading software titles. Call us for our full compatibility listing.

We've provided a selection of utilities on ROM and Challenger's disc drive is 40/80 track switchable. The extended DDOS takes no User RAM, PAGE, the position at which BASIC programs load, is left unchanged, normally staying at &OEOO, and Challenger dynamically moves files on the disc to overcome the 'can't extend' problem.

Challenger's price of £249.95 includes our unbeatable two year warranty, a full operating manual, VAT and free delivery. Call us now on 0737-65080 for details of our nationwide dealer network or simply post the coupon below.

OPUS SUPPLIES THE DRIVE

The Opus disc drive range has few rivals for quality, reliability, and sheer value for money. We've taken the best in Japanese disc drive technology and backed it with Britain's longest guarantee – an amazing two years. For full details of our extensive range of half-height 5¼" disc drives together with information on your nearest stockist, 'phone 0737-65080.

OPUS – BRITAIN'S LEADERS IN DISC DRIVE INNOVATION

To: Opus Supplies Limited, 55 Ormside Way, Holmethorpe Industrial Estate, Redhill, Surrey.
Please rush me the following: (PRICES INCLUDE VAT AND CARRIAGE)

_____ **Challenger Unit(s) £249.95 each**

_____ **RAM DISC upgrade(s) to 512K at £49.95 each**

I enclose a cheque for £_____ **or please debit my credit card account with the amount of £**_____

My Access ☐ **Barclaycard** ☐ **no. is:** ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

NAME _____

ADDRESS _____

_____

TELEPHONE _____

Opus.
Opus Supplies Ltd.

Fast Flood

Shingo Sugiura

Still struggling with PLOT85? This utility could be the answer to your prayers.

Using computer graphics can be one of the most fascinating aspects of computing. The BBC micro with its multitude of bit-mapped graphic modes and an eight colour palette is particularly suited for creating sophisticated and colourful pictures. Even though BBC BASIC supports many very powerful graphics commands, there is one rather surprising ommision — a flood fill facility.

Fair enough, there is a triangular fill facility (PLOT85) which can theoretically be used to create any solid shape but it is difficult to fill in anything more than the simplest of shapes with this command. With a flood fill facility, there are no such problems. All you need to do is to specify the starting point and the routine will fill in the enclosed region in the current foreground graphics colour.

ALGORITHMS

There are two main algorithms for flood fill routines. One is a recursive method and the other, non-recursive. The recursive method is very easy to understand and program (at least in BBC BASIC and any *truly* high level languages such as Pascal) but it requires a considerable amount of memory and is difficult to implement in machine code. A non-recursive routine is given in Listing 1.

When you type in the listing and run it, it should draw a picture of a house (as shown in Figure 1) and fill it in various colours. Don't worry too much about the garish colours or the feeble picture!

The algorithm makes use of a FIFO (first in, first out) buffer. This method is efficient in terms of memory usage but slow in BASIC. However, the algorithm itself is very efficient, compact and should allow you to grasp the workings of the routine (a detailed breakdown is given in the *How it works* section). Before moving onto the fast, compact, all singing, all dancing assembly language routine, I'll explain the syntax of the BASIC version.

SYNTAX OF ROUTINE

The fill routine is called with a line of the form **PROCfill(X,Y,C)** where (X,Y) is the co-ordinate of the starting point and C is the colour with which you want to fill. The initial point is first checked to see if it's black. If not, the routine is exited immediately as it is if C=0, so don't be too surprised if nothing happens on calling the routine.

As with all flood fill routines, it is very important that the region to be filled in is completely enclosed. If there is even one pixel which is unfilled, the fill routine will leak and probably fill the whole screen!

LISTING 1

```
  10REM Paint routine vers.1 (ED1E)
  20REM By Shingo Sugiura (657B)
  30REM (C) August 1985 (EDCC)
  40  (8A16)
  50MODE2 (7022)
  60VDU23;8202;0;0;0; (EB4E)
  70DIM ystack%(128),lstack%(128),rstack%(128),ws
% 8 (244C)
  80RESTORE (2F96)
  90REPEAT (324A)
 100READ A%,B%,C% (D14C)
 110IF A%>0 PLOTA%,B%,C% (D101)
 120UNTILA%<0 (74FD)
 130PROCcloud(150,900,130,70) (83E5)
 140PROCcloud(1100,900,130,70) (CE46)
 150PROCfill(100,100,2):PROCfill(900,100,2) (AB61
)
 160PROCfill(150,900,7):PROCfill(1100,900,7) (DE9
E)
 170PROCfill(400,600,6):PROCfill(800,600,6) (5F85
)
 180PROCfill(600,400,1) (C4FA)
 190PROCfill(600,800,4) (7FD0)
 200PROCfill(500,600,5) (BA45)
 210PROCfill(100,350,6) (657E)
 220A=GET (777E)
 230END (580F)
 240DEFPROCcloud(X%,Y%,SX%,SY%) (5A9E)
 250VDU29,X%;Y%; (9730)
 260X1%=SX%+SX%/7:Y1%=0 (F5C3)
 270FOR I=0 TO 6.3 STEP0.1 (6B60)
 280X%=SX%*COS(I)+SX%/7*COS(I*7) (2B73)
 290Y%=SY%*SIN(I)+SY%/7*SIN(I*7) (F0C3)
 300MOVEX1%,Y1%:DRAWX%,Y% (17B3)
 310X1%=X%:Y1%=Y% (0E61)
 320NEXT (D2FD)
 330VDU29,0;0; (2DFC)
 340ENDPROC (E9AA)
 350DATA 4,0,300,5,1280,300,4,300,300,5,300,700 (
C994)
 360DATA 4,900,300,5,900,700 (FF3D)
 370DATA 4,250,700,5,950,700 (DD9A)
 380DATA 5,600,900,5,250,700 (B2FF)
 390DATA 4,500,300,5,500,500,5,700,500,5,700,300
(E5A8)
 400DATA 4,350,550,5,350,650,5,450,650,5,450,550,
5,350,550 (607E)
 410DATA 4,750,550,5,750,650,5,850,650,5,850,550,
5,750,550 (354A)
 420DATA 4,500,300,5,400,0 (6305)
 430DATA 4,700,300,5,800,0 (72CE)
 440DATA -1,-1,-1 (61CD)
1000DEFPROCfill(IX%,IY%,C%) (CA78)
1010IFC%=0 ENDPROC ELSEGCOL0,C% (2028)
1020M%=&FFFF:W%=2^((?&355 MOD3)+1) (4340)
```

FASTER FILL ROUTINE

Listing 2 gives the source code for the fill routine — this time in assembly language. Type it in and save it before you run it. This is very important because any typing mistakes may (and probably will) render the program useless.

Replace the BASIC fill routine in Listing 1 with the new routine, change line 70 to read **PROCassemble,** and run it. The familiar picture of the house should be drawn and filled in but you should notice a distinct improvement in speed.

You may be wondering why this version takes up so many lines when compared with the BASIC version. This is because the number of mnemonics per line was deliberately kept own in order to aid legibility. The object code itself takes up less than 0.5K! It resides in a section of memory allocated by DIM code 500. If you want to compile the code to some other address, bear in mind that &280 bytes (five &80 byte blocks, pointed to by variables **ystack, lstackl, lstackh, rstackl** and **rstackh**) must be provided for the buffer. As it stands, the whole of page 6 (the input buffer) and page &A and the first &80 bytes of page &C (user defined characters) is used.

SHADES

One interesting modification to the algorithm allows pictures to be filled in with shades as well as solid colours. Type in Listing 3 and run it. It will draw the house but should fill it in subtle (depending on your

LISTING 1

```
1030IX%=IX%AND(&10000-W%):IY%=IY%AND&FFFC (A1A1)
1040IFPOINT(IX%,IY%)<>0 ENDPROC (A8FD)
1050F%=1:Z%=0 (78C2)
1060PROCline(IX%,IY%,0) (ABCD)
1070REPEAT (0A36)
1080PROCunqueue (4E11)
1090REPEAT (2B77)
1100IFUX%<rx% PROCline(UX%,y%+4,0) (DBA0)
1110IFDX%<rx% PROCline(DX%,y%-4,1) (581B)
1120UNTILDX%>=rx% AND UX%>=rx% (97F4)
1130UNTILF%=(Z%+1)AND127 (BB3A)
1140ENDPROC (F8D5)
1150DEFPROCline(tx%,ty%,flag%) (DA96)
1160IFPOINT(tx%,ty%)>0 PROCscan:ENDPROC (1FD4)
1170PLOT77,tx%,ty% (E581)
1180X%=ws%:Y%=ws% DIV256 (3C3A)
1190A%=&D:CALL&FFF1 (574E)
1200PROCqueue(ty%,!ws%ANDM%,(ws%!4 ANDM%)+W%) (18
98)
1210PROCscan (473D)
1220ENDPROC (A36F)
1230DEFPROCscan (AF21)
1240PLOT92,tx%,ty% (37F9)
1250R%=(!&310 ANDM%)+W% (65CA)
1260IFflag%=0 UX%=R% ELSE DX%=R% (F6A7)
1270ENDPROC (C417)
1280DEFPROCqueue(ay%,lx%,rx%) (16CA)
1290Z%=(Z%+1)AND127 (3A37)
1300ystack%(Z%)=ay% (E32C)
1310lstack%(Z%)=lx%:rstack%(Z%)=rx% (FF45)
1320ENDPROC (16B7)
1330DEFPROCunqueue (4C24)
1340y%=ystack%(F%) (613B)
1350rx%=rstack%(F%) (EEC7)
1360UX%=lstack%(F%):DX%=lstack%(F%) (6912)
1370IFy%=0 DX%=rx% ELSEIFy%=1020 UX%=rx% (6152)
1380F%=(F%+1)AND127 (FC4B)
1390ENDPROC (99E0)
```

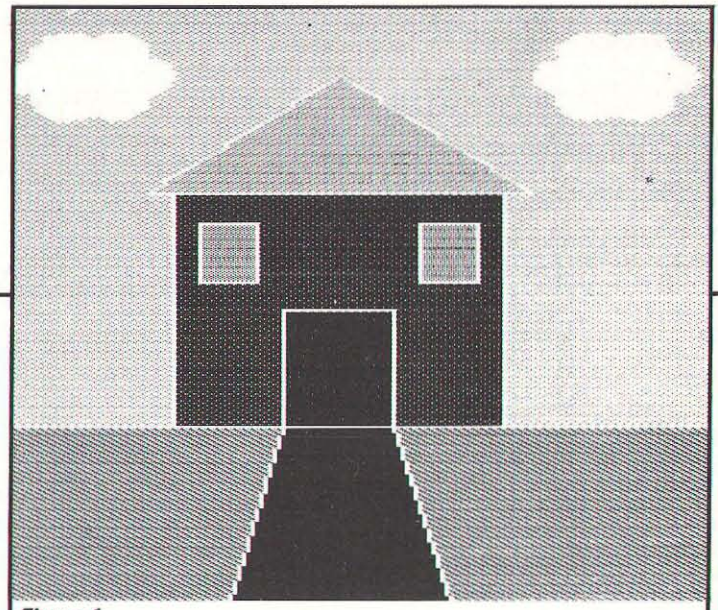


Figure 1

taste!) shades. The modifications are minor and coding it in assembly language should present no problems.

The new de-luxe shade fill routine takes the form **PROCfill(X,Y,C1,C2).** As before, (X,Y) specifies the initial position and as you might have noticed, the shades are made up from two colours. These two colours are specified by C1 and C2. They may of course be the same (to create solid colours, as before) but neither of them can be zero, ie black. This option extends the number of colours available in MODE2 to a staggering 28 (or a mind-numbing 120 if you get kicks from filling pictures with flashing colours). In modes 1 or 5, the number of colours is only increased to six and it isn't useful at all in modes 0 or 4.

EVEN FASTER FILL ROUTINE

All the routines introduced so far will work in all graphics modes and they all depend on standard operating system routines for plotting and drawing lines. These operating system routines spend quite a bit of time checking for window edges, off screen co-ordinates etc. As such, there are bound to be some of you craving for even faster fill routines.

Listing 4 gives the source code for a MODE2 only fill routine which pokes directly to screen memory (shock horror!). Type it in and save it. Tag it onto the end of Listing 1 and change line 70 to read **PROCassemble** as before and run it. Fast eh? In fact, I'm going to stick my neck out and say that it is the fastest fill routine ever devised for the Beeb.

COMPARISON

You may be interested in a little survey I've conducted regarding the speed of various other fill routines that are either commercially available or published in rival mags. The times taken to fill in the picture of the house by the various routines are given in Figure 2. Computer Concept's Graphics Extension ROM fill routine (written in machine code) is excrutiatingly slow. In fact, it's slower than the fill routine given in Listing 1 which is written in BASIC! The fill routine by Beebug is also very slow even though it is written in machine code.

Acorn User's fill routine is rather special. Not only can it handle shades (not simple chequered patterns as in Listing 3 but the pattern may be designed by the user) but the shade may contain the background colour. For this reason, the algorithm is far more time consuming than mine — for good reason I might add.

Although my fill routine (Listing 2) is the fastest of the lot, this does not mean that it's perfect. In fact, I'm quite certain that some of you will find ways of improving the algorithm even further.

NB The size of the buffer used by the routines may seem rather large and in fact, they will never ever be used up. If memory is tight, you may like to modify the programs so that they use smaller buffers.

LISTING 2

```
1000DEFPROCfill(X%,Y%,C%) (47BE)
1010GCOL0,C% (1CD9)
1020W%=2^((?mode MOD3)+1) (C307)
1030X%=X%AND(&10000-W%):A%=FNhi(X%):Y%=Y%DIV4 (E4
78)
1040CALLfill (B6BE)
1050ENDPROC (0CAA)
1060DEFPROCassemble (9873)
1070DIM code 400 (B157)
1080oswrch=&FFEE:osword=&FFF1 (36FC)
1090first=&2A:last=&2B:yloop=&2C:flag=&2D:pixel=&
2E (5EF1)
1100ux=&2F:dx=&31:rlimit=&33 (F929)
1110param1=&4F:param2=&51:ws=&53 (F3A6)
1120ystack=&A00:rstack1=&A80:rstackh=&600 (E8BF)
1130lstack1=&680:lstackh=&C00 (9DF2)
1140mode=&355:col=&359 (4330)
1150FORpass=0TO2STEP2:P%=code (B8FF)
1160[OPT pass (DC17)
1170.fill (4712)
1180  STA param1+1 (BAC8)
1190  STX param1 (E145)
1200  STY param2:JSR gety (D226)
1210  LDA col:BEQ quit (2960)
1220  JSR point (8745)
1230  BNE quit (4666)
1240  LDY mode (B0BB)
1250  LDA table,Y (E3C4)
1260  STA pixel (12D8)
1270  LDY#0 (4BC2)
1280  STY last (9E9D)
1290  INY:STY first (31FD)
1300  JSR fline (2601)
1310.repeat (DA1A)
1320  JSR unqueue (DB90)
1330.loop (ED00)
```

CONTINUED OVER

LISTING 2

```
1340  LDA ux:CMP rlimit (63A5)
1350  LDA ux+1:SBC rlimit+1 (B6F3)
1360  BCS noup (511B)
1370  LDA ux:STA param1 (C592)
1380  LDA ux+1 (5605)
1390  LDX yloop:INX (8F3B)
1400  LDY#0 (A5DA)
1410  JSR line (4CF2)
1420.noup (8E2E)
1430  LDA dx:CMP rlimit (C3A7)
1440  LDA dx+1:SBC rlimit+1 (3436)
1450  BCS nodown (2099)
1460  LDA dx:STA param1 (0155)
1470  LDA dx+1 (03CD)
1480  LDX yloop:DEX (822C)
1490  LDY#1 (CD28)
1500  JSR line (2889)
1510.nodown (7D4E)
1520  LDA dx:CMP rlimit (F8D2)
1530  LDA dx+1:SBC rlimit+1 (AFAC)
1540  BCC loop (8AAE)
1550  LDA ux:CMP rlimit (FD9F)
1560  LDA ux+1:SBC rlimit+1 (AE05)
1570  BCC loop (7485)
1580  LDY last:INY:TYA:AND#127 (5A73)
1590  CMP first:BNE repeat (CF77)
1600.quit (BD19)
1610  RTS (329B)
1620.gety (F7CD)
1630  LDA#0:STA param2+1 (1AFC)
1640  ASL param2:ROL param2+1 (2ACF)
1650  ASL param2:ROL param2+1 (E7BF)
1660  RTS (5E7A)
1670.point (314C)
1680  LDX#FNlo(param1) (F0FB)
1690  LDY#FNhi(param1) (5F6A)
1700  LDA#9 (E191)
1710  JSR osword (7116)
1720  LDA param1+4 (B8FB)
1730  RTS (5F20)
1740.line (7602)
1750  STA param1+1 (4745)
1760  STX param2 (2642)
1770  JSR gety (E534)
1780.fline (B545)
1790  STY flag (F7E7)
1800  JSR point (077F)
1810  BEQ lcont (AEC4)
1820  JMP scan (6218)
1830.lcont (F720)
1840  LDA#77:JSR plot (97DF)
1850  LDX#FNlo(ws) (66D8)
1860  LDY#FNhi(ws) (1B86)
1870  LDA#&D (55E3)
1880  JSR osword (0284)
1890  LDY last (7C2D)
1900  INY (6796)
1910  TYA (26D8)
1920  AND#127 (DF47)
1930  TAY:STY last (B337)
1940  JSR codey (B79B)
1950  STA ystack,Y (3F1F)
1960  LDA ws:STA lstackl,Y (7EEC)
1970  LDA ws+1:STA lstackh,Y (1E75)
1980  LDA ws+4:CLC:ADC pixel:STA rstackl,Y (F563)
1990  LDA ws+5:ADC#0:STA rstackh,Y (9AD1)
2000  JSR gety (68BD)
2010.scan (1C63)
2020  LDA#92:JSR plot (F9F0)
2030  LDA flag:CLC:BNE down (71E5)
2040  LDA &310:ADC pixel:STA ux (FC79)
2050  LDA &311:ADC#0:STA ux+1 (D9E5)
2060  RTS (FA1C)
2070.down (8508)
2080  LDA &310:ADC pixel:STA dx (6125)
2090  LDA &311:ADC#0:STA dx+1 (618F)
2100  RTS (0711)
```

LISTING 2

```
2110.plot (2AB0)
2120  PHA (2792)
2130  LDA#25:JSR oswrch (9296)
2140  PLA:JSR oswrch (2E7E)
2150  LDX#0 (9B44)
2160.sloop (31AC)
2170  LDA param1,X:JSR oswrch (F7F4)
2180  INX:CPX#4 (DA05)
2190  BNE sloop (8F93)
2200  RTS (6931)
2210.codey (393A)
2220  LSR param2+1:ROR param2 (0728)
2230  LSR param2+1:ROR param2 (CA58)
2240  LDA param2 (9356)
2250  RTS (4D8B)
2260.unqueue (7804)
2270  LDX first (A137)
2280  LDA ystack,X:STA yloop (E7D0)
2290  LDA rstackl,X:STA rlimit (53FA)
2300  LDA rstackh,X:STA rlimit+1 (AF77)
2310  LDA lstackl,X (629D)
2320  STA dx:STA ux (5F56)
2330  LDA lstackh,X (6DB4)
2340  STA dx+1:STA ux+1 (3898)
2350  LDX first (B26A)
2360  INX:TXA:AND#127 (1312)
2370  STA first (A462)
2380  LDA yloop:BNE notzero (7243)
2390  LDA rlimit:STA dx (E064)
2400  LDA rlimit+1:STA dx+1 (AD72)
2410  RTS (017D)
2420.notzero (FA5B)
2430  CMP#255:BNE not255 (CB54)
2440  LDA rlimit:STA ux (229B)
2450  LDA rlimit+1:STA ux+1 (E0F4)
2460.not255 (D3A4)
2470  RTS (D990)
2480.table (22C6)
2490]NEXT (86DB)
2500RESTORE2550 (0FD6)
2510FORA%=0TO5 (CE32)
2520READB%:A%?table=B% (523E)
2530NEXT (D88F)
2540ENDPROC (3D95)
2550DATA 2,4,8,0,4,8 (C701)
2560DEFFNlo(number) (152F)
2570=number MOD 256 (5FB8)
2580DEFFNhi(number) (7ECB)
2590=number DIV 256 (D8DF)
```

LISTING 3

```
  10REM Paint routine vers.3 (AD98)
  20REM By Shingo Sugiura (657B)
  30REM (C) August 1985 (EDCC)
  40  (8A16)
  50MODE2 (7022)
  60VDU23;8202;0;0;0; (EB4E)
  70DIM ystack%(128),lstack%(128),rstack%(128),ws
% 8 (244C)
  80RESTORE (2F96)
  90REPEAT (324A)
 100READ A%,B%,C% (D14C)
 110IF A%>0 PLOTA%,B%,C% (D101)
 120UNTILA%<0 (74FD)
 130PROCcloud(150,900,130,70) (83E5)
 140PROCcloud(1100,900,130,70) (CE46)
 150PROCfill(100,100,2,3):PROCfill(900,100,2,3) (
0555)
 160PROCfill(150,900,7,7):PROCfill(1100,900,7,7)
(DFC3)
 170PROCfill(400,600,6,6):PROCfill(800,600,6,6) (
44B3)
 180PROCfill(600,400,1,4) (5B70)
 190PROCfill(600,800,3,6) (D6CD)
 200PROCfill(500,600,5,1) (C4F6)
 210PROCfill(100,350,6,7) (2C47)
 220A=GET (777E)
```

LISTING 3

```
  230END (580F)
  240DEFPROCcloud(X%,Y%,SX%,SY%) (5A9E)
  250VDU29,X%;Y%; (9730)
  260X1%=SX%+SX%/7:Y1%=0 (F5C3)
  270FOR I=0 TO 6.3 STEP0.1 (6B60)
  280X%=SX%*COS(I)+SX%/7*COS(I*7) (2B73)
  290Y%=SY%*SIN(I)+SY%/7*SIN(I*7) (F0C3)
  300MOVEX1%,Y1%:DRAWX%,Y% (17B3)
  310X1%=X%:Y1%=Y% (0E61)
  320NEXT (D2FD)
  330VDU29,0;0; (2DFC)
  340ENDPROC (E9AA)
  350DATA 4,0,300,5,1280,300,4,300,300,5,300,700 (
C994)
  360DATA 4,900,300,5,900,700 (FF3D)
  370DATA 4,250,700,5,950,700 (DD9A)
  380DATA 5,600,900,5,250,700 (B2FF)
  390DATA 4,500,300,5,500,500,5,700,500,5,700,300
(E5A8)
  400DATA 4,350,550,5,350,650,5,450,650,5,450,550,
5,350,550 (607E)
  410DATA 4,750,550,5,750,650,5,850,650,5,850,550,
5,750,550 (354A)
  420DATA 4,500,300,5,400,0 (6305)
  430DATA 4,700,300,5,800,0 (72CE)
  440DATA -1,-1,-1 (61CD)
  450DEFPROCfill(IX%,IY%,C1%,C2%) (90DD)
  460IFC1%=0 OR C2%=0 ENDPROC (2EC5)
  470M%=&FFFF:W%=2^((?&355 MOD3)+1) (F7F2)
  480IX%=IX%AND(&10000-W%):IY%=IY%AND&FFFC (7BC3)
  490IFPOINT(IX%,IY%)<>0 ENDPROC (1E4F)
  500F%=1:Z%=0 (ED4D)
  510PROCline(IX%,IY%,0) (FCDA)
  520REPEAT (D4D9)
  530PROCunqueue (DB05)
  540REPEAT (02AE)
  550IFUX%<rx% PROCline(UX%,y%+4,0) (6F12)
  560IFDX%<rx% PROCline(DX%,y%-4,1) (33F0)
  570UNTILDX%>=rx% AND UX%>=rx% (858A)
  580UNTILF%=(Z%+1)AND127 (1587)
  590ENDPROC (1F79)
  600DEFPROCline(tx%,ty%,flag%) (AA48)
  610even%=(((ty%/4)AND1)=0) (857F)
  620IFPOINT(tx%,ty%)>0 PROCscan:ENDPROC (A76B)
  630IFeven% GCOL0,C1% ELSEGCOL0,C2% (9393)
  640PLOT77,tx%,ty% (2693)
  650X%=ws%:Y%=ws%DIV256 (3603)
  660A%=&D:CALL&FFF1 (951F)
  670L%=!ws%ANDM%:R%=ws%!4 ANDM% (A209)
  680IFeven% GCOL0,C2% ELSEGCOL0,C1% (51CF)
  690MOVE(L%+W%)AND(M%-W%),ty%:PLOT21,R%,ty% (C974
)
  700PROCqueue(ty%,L%,R%+W%) (4E3B)
  710PROCscan (2827)
  720ENDPROC (DBDD)
  730DEFPROCscan (1C70)
  740PLOT92,tx%,ty% (CD24)
  750R%=(!&310 ANDM%)+W% (30C8)
  760IFflag%=0 UX%=R% ELSEDX%=R% (FD4D)
  770ENDPROC (BCA5)
  780DEFPROCqueue(ay%,lx%,rx%) (886F)
  790Z%=(Z%+1)AND127 (1F9D)
  800ystack%(Z%)=ay% (21B3)
  810lstack%(Z%)=lx%:rstack%(Z%)=rx% (3E4A)
  820ENDPROC (BD9F)
  830DEFPROCunqueue (EF6A)
  840y%=ystack%(F%) (C275)
  850rx%=rstack%(F%) (2C58)
  860UX%=lstack%(F%):DX%=lstack%(F%) (A81D)
  870IFy%=0 DX%=rx% ELSEIFy%=1020 UX%=rx% (F3F5)
  880F%=(F%+1)AND127 (3ED4)
  890ENDPROC (32C8)
```

LISTING 4

```
 1000DEFPROCfill(X%,Y%,col) (1519)
 1010X%=X%DIV8:Y%=(1023-(Y%AND&FFFC))DIV4 (F56F)
 1020GCOL0,col:CALLfill (6B9D)
```

LISTING 4

```
 1030ENDPROC (A93B)
 1040DEFPROCassemble (26D9)
 1050DIM code 500 (1295)
 1060ytemp=&2A:xtemp=&2B:mask=&2C:cmask=&2D (FB6C)
 1070param1=&2E:param2=&2F:flag=&30 (CC2A)
 1080x=&31:y=&32:yloop=&33:rx=&34:lx=&35 (B10D)
 1090address=&4F:first=&51:last=&52:dx=&53:ux=&54:
rlimit=&55 (AB5F)
 1100col=&359:ystack=&600:lstack=&A00:rstack=&C00
(C801)
 1110mode=&355:table=&C375 (D546)
 1120FOR pass=0 TO 2 STEP2:P%=code (4F93)
 1130[OPT pass (2BFD)
 1140.fill (3F34)
 1150  STX param1:STY param2 (5925)
 1160  JSR point (31A7)
 1170  BNE nofill (FB32)
 1180  LDA col:BEQ nofill (0705)
 1190  STY last:STY flag (C676)
 1200  INY:STY first (1E43)
 1210  JSR fline (67FA)
 1220.repeat (AD2B)
 1230  JSR unqueue (DF49)
 1240.loop (8399)
 1250  LDA ux:CMP rlimit (58D0)
 1260  BCS noup (80B8)
 1270  STA param1:LDX yloop:DEX (A316)
 1280  STX param2 (9B58)
 1290  LDY#0 (3B1C)
 1300  JSR fline (2601)
 1310.noup (7689)
 1320  LDA dx:CMP rlimit (0729)
 1330  BCS nodown (6D4F)
 1340  STA param1:LDX yloop:INX (C120)
 1350  STX param2 (70AA)
 1360  LDY#1 (F554)
 1370  JSR fline (52EE)
 1380.nodown (AD42)
 1390  LDA dx:CMP rlimit (CBA7)
 1400  BCC loop (CCEB)
 1410  LDA ux:CMP rlimit (006C)
 1420  BCC loop (8718)
 1430  LDY last:INY (27F1)
 1440  CPY first (AC1B)
 1450  BNE repeat (6946)
 1460.nofill (7AA5)
 1470  RTS (A1B6)
 1480.fline (77AC)
 1490  STY flag (A541)
 1500  LDX param1:LDY param2 (39CB)
 1510  JSR point (44AE)
 1520  BEQ fi_cont (2B9D)
 1530  JMP scan (6CB8)
 1540.fi_cont (455B)
 1550  LDY param2 (F13F)
 1560  JSR line:TYA (D787)
 1570  LDX last:INX (4218)
 1580  STX last (CF92)
 1590  STA ystack,X (DFC9)
 1600  LDA lx:STA lstack,X (E950)
 1610  LDA rx:STA rstack,X (660D)
 1620  JMP update (0E17)
 1630.unqueue (ECD8)
 1640  LDX first (78F7)
 1650  LDA ystack,X:STA yloop (68CA)
 1660  LDA rstack,X:STA rlimit (0090)
 1670  LDA lstack,X (FAD1)
 1680  STA ux:STA dx (4224)
 1690  INC first (27FE)
 1700  LDA yloop:BNE notzero (00F2)
 1710  LDA rlimit:STA ux (CA92)
 1720  RTS (EB2C)
 1730.notzero (8D8E)
 1740  CMP#255:BNE not255 (8BA6)
 1750  LDA rlimit:STA dx (A22E)
 1760.not255 (7F29)
 1770  RTS (CF96)
```

CONTINUED OVER

LISTING 4

```
1780.point (075C)
1790  JSR calc (D2EF)
1800  LDY#0 (2977)
1810  TXA:AND#1 (F58C)
1820  BNE rpoint (A1F7)
1830.lpoint (560F)
1840  LDA(address),Y (84DD)
1850  AND#&AA:RTS (4BF9)
1860.rpoint (FCF2)
1870  LDA(address),Y (03B8)
1880  AND#&55:RTS (09C1)
1890.scan (A92E)
1900  TXA:AND#1:BEQ cont_scan (8FDF)
1910  JSR inc_ad (612E)
1920.cont_scan (5059)
1930  JSR lpoint:BEQ noscan (B6C1)
1940  INX:JSR rpoint:BEQ noscan (1EF9)
1950  JSR inc_ad (60C8)
1960  CPX#160:BNE cont_scan (11B9)
1970.noscan (2991)
1980  STX rx (30A6)
1990.update (F8E0)
2000  LDA flag:BNE down (0CD4)
2010  LDA rx:STA ux:RTS (D6A2)
2020.down (0D62)
2030  LDA rx:STA dx:RTS (2E46)
2040.calc (F75E)
2050  STY ytemp (7C73)
2060  LDA#0 (51AD)
2070  STA xtemp (EA8F)
2080  TYA (C17E)
2090  LSRA:LSRA (C410)
2100  AND#&FE (772B)
2110  TAY:LDA table,Y:STA address+1 (0385)
2120  LDA table+1,Y:STA address (3BA1)
2130  LDA ytemp:AND#7 (3E40)
2140  STA ytemp (9C7D)
2150  TXA:AND#&FE (CFF7)
2160  ASL A:ROL xtemp (DA44)
2170  ASL A:ROL xtemp (8511)
2180  ORA ytemp (D0CC)
2190  ADC address:STA address (71D6)
2200  LDA address+1 (2860)
2210  ADC xtemp (2928)
2220  ADC#&30:STA address+1 (F400)
2230  RTS (9566)
2240.plot (59C0)
2250  LDA#&AA:STA mask (03BA)
2260  LDA col:AND#&55:STA cmask (5D72)
2270  TXA:AND#1:BNE right (89C7)
2280  LDA#&AA:STA mask (E50D)
2290  ASL cmask (233A)
2300  SEC:ROL mask (BF4C)
2310.right (AF06)
2320  LDY#0 (57B7)
2330  LDA(address),Y (8D24)
2340  AND mask (157B)
2350  ORA cmask (6F98)
2360  STA(address),Y (B199)
2370  RTS (2030)
2380.line (0FAB)
2390  STX x:STY y (E7F1)
2400  JSR calc (AAB5)
2410  LDY#0 (AF10)
2420  TXA:AND#1:BEQ rloop (E7C6)
2430  JSR rpoint (2C59)
2440  BNE eor1 (C906)
2450  JSR plot:JSR inc_ad (0100)
2460  CPX#160:BEQ eor1 (43F7)
2470.rloop (B42E)
2480  LDA(address),Y:BNE rlcont (4D21)
2490  LDA col:STA (address),Y (E29A)
2500  INX:JSR inc_ad:BNE rlcheck (83E7)
2510.rlcont (320B)
2520  JSR lpoint:BNE eor1 (222A)
2530  JSR plot (864F)
2540  INX:JSR rpoint:BNE eor1 (5AA6)
2550  JSR plot:JSR inc_ad (1D5A)
2560.rlcheck (58F0)
2570  CPX#160:BNE rloop (B1C5)
2580.eor1 (D732)
2590  STX rx (F31A)
2600  LDX x:BEQ noleft (561D)
2610  DEX:LDY y (58D1)
2620  JSR calc (6243)
2630  LDY#0 (B2F9)
2640  TXA:AND#1:BNE lloop (CB09)
2650  JSR lpoint (13C9)
2660  BNE eol1 (6B63)
2670  JSR plot:JSR dec_ad (184C)
2680.lloop (9B4B)
2690  LDA(address),Y:BNE llcont (FC9F)
2700  LDA col:STA(address),Y (9086)
2710  DEX:JSR dec_ad:BNE llcheck (27AB)
2720.llcont (7694)
2730  JSR rpoint:BNE eol1 (103E)
2740  JSR plot (6C87)
2750  DEX:JSR lpoint:BNE eol1 (D62D)
2760  JSR plot:JSR dec_ad (A4E6)
2770.llcheck (3F94)
2780  CPX#255:BNE lloop (5939)
2790.eol1 (D86E)
2800  INX (1A13)
2810.noleft (F387)
2820  STX lx (BF63)
2830  LDY y:LDX x (D7E2)
2840  RTS (BD04)
2850.inc_ad (B5D4)
2860  CLC (49AB)
2870  LDA address (EB50)
2880  ADC#8 (636B)
2890  STA address (CC91)
2900  LDA address+1 (26D8)
2910  ADC#0 (5F77)
2920  STA address+1 (F225)
2930  INX (C3A4)
2940  RTS (98E4)
2950.dec_ad (4057)
2960  SEC (0282)
2970  LDA address (9609)
2980  SBC#8 (6726)
2990  STA address (B1C8)
3000  LDA address+1 (8D73)
3010  SBC#0 (EDAB)
3020  STA address+1 (598E)
3030  DEX (5ACC)
3040  RTS (7A64)
3050]NEXT (7C3C)
3060ENDPROC (4D46)
```

Robico Software

3 Fairland Close, Llantrisant, Mid Glamorgan. CF7 8QH Tel (0443) 227354

Brilliant Games for the BBC B

THE NEW TEXT ONLY, DISC-BASED MEGA-ADVENTURE!

ENTHAR SEVEN

40 Track: £17.95, 80 Track: £16.95

"An absolute knock-out - we couldn't have given it more marks if we tried!... BBC cassette owners - go out and buy a disk drive now. Disk owners read on, then go and buy the game!... Personal Rating: 10 doesn't start to say it!" (C. & V.G.)

"I doubt that a better game has been written... If you own a disc drive and like adventure games, then don't hesitate to buy ENTHAR SEVEN - you won't regret it." (A & B Computing)

It is the distant future ... You are on board a fragile Interplanetary Space Hopper which orbits a small, Earth-like planet known as Enthar Seven. Your task is to reach the planet's surface before the Hopper's orbit decays and the spacecraft spirals into oblivion!

Once on Enthar Seven, you must explore the planet and find and board an Escape Vessel.

Enthar Seven is written entirely in machine code and uses disc filing techniques to provide a massive scenario. The game has an advanced command interpreter which can understand complete sentences and multiple statements, and compresses the text stored on disc (using the same methods as in our acclaimed cassette based games) to approximately half its original length. When expanded there is well over 100K of text!

With 450 atmospherically described locations to visit, 80 objects to examine, an extensive vocabulary and approximately 1200 lengthy messages, we believe that Enthar Seven is the outstanding BBC adventure to date.

ISLAND OF XAAN

Cassette: £7.95, 40 Track: £9.95, 80 Track: £9.95

"I suspect this could prove to be one of the best text adventures on the BBC." (P.C. Weekly)

"Excellent game... Write more!" (A & B Computing)

Escape from the forbidding Island of Xaan by finding a disguise and enough treasure to pay your passage on a sailing ship. Machine code, text compression and about 180 locations make this a must for all adventurers.

RICK HANSON

Cassette: £9.95, 40 Track: £11.95, 80 Track: £11.95

"If you want to know how good an adventure can be, buy RICK HANSON and find out!" (A & B Computing)

"Good to see there are still a few programs around that make you envious of BBC owners." (P.C. Weekly)

As Rick Hanson, Special Agent, your mission is to assassinate an evil General and save the world from the final holocaust! Rick Hanson, a machine code, compressed text adventure, has 220 locations and an advanced sentence interpreter.

BANJAX

Cassette: £9.95, 40 Track: £11.95, 80 Track: £11.95

"Some of the best graphics in a BBC game since Frak." (Your Computer)

"Some of the best Mode 1 graphics I have ever seen." (Micro User)

"The graphics are superb... Virtually every screen is different... I strongly urge you to buy this game." (A & B Computing)

Banjax is a machine code, arcade adventure with a massive 240 locations and brilliant MODE 1 graphics. These combine to give the "feel" of classic adventuring. Will Banjax the Bear enter the Inner Sanctum of the Golden Temple of the Land of Mystery?"

BEELINE

Cassette: £7.95, 40 Track: £9.95, 80 Track: £9.95

"I'll give it 10" (S.W. Echo)

This is a fun, machine code arcade game with smooth, colourful graphics, 18 playing screens, a practice screen, choice of starting screen, monitor and T.V. modes and two levels of play at a giveaway price!

* All software despatched first class by return *

ROM Report

Dave Reeder

Losing track of the number of utility ROMs around? Here's another half dozen recent releases!

The flow of utility ROMs for the BBC micro continues apace — new languages, new implementations of utilities, new programming aids, etc. There are now so many that no one magazine could hope to review them all; in fact the vast majority never receive a review anywhere. That said, I'm looking here at a round half dozen of well-advertised ROMs that offer much the same set of utility commands and so can be usefully compared.

In fact, it becomes increasingly difficult to imagine what extra or unusual commands any ROM can add to the number already available. Nevertheless, the software houses keep trying and certainly each new release does appear to offer a different hand of the utility cards any self-respecting Beeb user needs to play the programming/memory examination game.

FIRST AID

In a world of ROMs hovering around the £20-£30 range, it is refreshing to find a utility ROM costing a mere £15 -though, of course, this is for 8K of utilities only.

First Aid 1.1 to give it it's full name provides a set of useful programming utilities for the BASIC or m/c programmer: none of them outstanding but all useful and adequately implemented. Typical of the genre is the *DIS command, providing a standard m/c disassembler displaying address, data in hex, ASCII of data plus m/c mnemonics of the data. This is scrollable as is *HEX, the slightly more interesting hex and ASCII dump routine. What makes *HEX slightly unstandard is the combination of this command with *SIDE which copies any sideways ROM into RAM starting at &3000, there examinable by *HEX.

A range of useful if standard commands follow: *FKEY to display function key definitions, *VTOP to print top of variable

Figure 1: the simple machine code monitor in "First Aid".

```
*MON

Stack

  SP X  Y  A  NV-BDIZC
  FF 07 04 FF 01110001

Memory          HEX                ASCII
FFF8: 08 02 00 0D CD D9 1C DC  ........
0000: 02 1A 02 1A 00 7C 00 7C  ..... .
0008: 00 00 01 00 07 63 6B 0A  .....ck.
0010: 0D 0A 02 1A 0A 01 33 B4  ......3.
0018: 1A 00 07 00 00 1A 00 00  ........
0020: 00 46 46 FF 00 00 00 40  .FF....(
0028: FF 20 00 00 00 00 2E 2E  . .....
0030: 2E 20 37 43 20 20 2E 00  . 7C  ..

C Continue E End M Memory

A Alter R Registers J JSR
```

Figure 2: "The Scythe" provides a run-down of on-board ROMs.

```
*ROMLIST
0     SPEECH 1.0 (C)DCW 8K
1     SLAVE  (C) 1985 J.Aughton,G.Pennington,I.Pium
arta 16  K
2     BASIC Extensions (C)1984 R.T.Florance & R.J.H
arrison 8K
3     WORDWISE-PLUS (C) 1984 CC 16K
4     ROM MASTER (C)1985 Chalice 8K
5     SCYTHE (C)1985 Chalice 8K
6     Altra PROBE  (C)1984 Altra 16K
7     FIRST AID 1.1 (C)1985 ALTRA / M.P.Hudson 8  K

8     DISC DOCTOR  (C)  COMPUTERCONCEPTS 8K
9     ALTRA TUBE EDITOR (C)ALTRA ROMS A.M.Lord 1985
 8K
10     PRINTMASTER (Epson)  (C)84 Computer Concepts
 M.Bateman P.Hiscock
8K
11     BeebMan (C) Slogger 1985 8K
12     DFS,NET (C)ROFF 16K
13     Upgrade II 8.39 (C) 1984 R T Clayton 8K
14 *    BASIC (C)1982 Acorn 16K
15     Empty
```

```
*FKEYS
*KEY  0 AUTO
*KEY  1 RENUMBER M
*KEY  2 LOAD "
*KEY  3 CHAIN "
*KEY  4 SAVE "
*KEY  5 OLD M
*KEY  6 MODE6 M
*KEY  7 *FIND M
*KEY  8 LIST M
*KEY  9 RUN M
*KEY  10
*KEY  11
*KEY  12
*KEY  13
*KEY  14
*KEY  15

f: user  Sf: &80  Cf: &90  SCf: off
```

Figure 3: function key definitions thanks to "Probe".

table, *FREE to display HIMEM/LOMEM, *ROMS to display sideways ROMs, etc. What does seem more useful is the inclusion of the ability to use *First Aid 1.1* as a language, an effective REM stripper and a Bad Program fix which will also remove hidden control codes from REM statements allowing you to list some unlistable programs.

Nothing outstanding here but it may well serve your purpose for a general programmer's utility ROM, though I suspect *Brom* or *Slave* will get more sales.

TUBE EDITOR

Despite being advertised as a ROM utility package for the Second Processor user, this ROM is merely an update of Altra's *BASIC Editor* and so my comments are mostly applicable to that as well.

Probably the most unusual thing about this ROM is its astounding price -£39« which is a horrifying sum for even a full labelling disassembler and editor. Admittedly it is a very interesting disassembler in that, as well as the usual assembler display in 6502 mnemonics, it is 'intelligent' as it hones in on any labels and ensures disassembly starts with them.

Thus, whilst a normal disassembly might look like this:

```
2000 NOP  0C .
   NOP  A2 .
   NOP  03 .
  2003 ASL A 0A .
```

the *Tube Editor* version would be:

```
2000 ?D?  0C .
 TEST 2001 LDX -3 A2 .
  2003 ASL A 0A .
```

and so on. As well as this rather more understandable format, the utility will also cover extended 6502 mnemonics, including the 'illegal opcodes' used rather naughtily by many programmers. Like most commands you can also disassemble the Second Processor's 65C02 CPU, which as you will know contains instructions not in the normal 6502.

There are the expected label appending and deletion commands here as well as a full set of memory dumps, edits, etc. The ROM's ability to search through memory and offer easy-to-use utilities is impressive and its disassembly display is perhaps the clearest I can recall.

However, despite the ability to track down and identify illegal opcodes, there is little here that is not already available many times over.

PROBE

The basic Probe 1, from Altra, is a BASIC and disc editor; there is also a *Probe 2* which at an extra cost (£39 as opposed to £29) applies a similar set of utilities across the tube for Second Processor owners. To be honest, *Probe* is a bit of a rag-bag of utilities, covering BASIC, disc drives, printers and what are called 'general purpose'.

There is, for instance, a formatter of assembly listings which usefully includes text line numbers; a BASIC program lister which lists from a filing system without loading it first and thus overwriting one already in memory; a disassembler; a disc sector editor; a function keys listing; a disc formatter; a memory editor; a block memory mover; a disc track repairer; a ROM identifier; a disc verifier; and so on. None of these are outstanding but all have fairly common uses.

Of far more interest is the very extensive BASIC editor which works directly on a program's tokenised form in memory. This provides several useful commands such as a bad program relink, a formatted program lister which puts every statement on a new line together with indentations and a partial line renumbering routine (one of the more obvious deficiences of the BBC editor).

There is a very extensive search facility, allowing matching lines to be listed, specific lines numbers to be printed, global and selective replacements. These act rather like a word processor on any BASIC program and the ROM is as easy to use here as you could wish. An extensive packer routine is also included which pulls back memory by removing REM statements, removing unnecessary spaces, removing comments from assembler listings, shortens variable,FN and PROC names to one or two letters wherever possible and, lastly, concatenates lines.

ROM MASTER

Once again we have a ROM whose list of commands suggests a fairly standard ROM control ROM — we've seen this before and will no doubt do so again. However, at £16, it is cheap enough to track down for one or two features of interest.

Firstly, though, for the sake of completeness let's detail the kind of commands you're offered in case you don't already have them in another form. *Rom Master* provides control over other ROMs on board (either the standard number or via an expansion board) allowing identification, enabling, disabling, transfer of ROM contents to sideways RAM, ROM disassembly, ASCII sequence search and so on.

Strangely, there are some more general commands here as well: loading or saving a set of function key definitions, for example. But (and here we venture into somewhat unusual territory) you can also display any variable in binary/decimal/hex, calculate any CRC value or search for a sequence of bytes between any two memory addresses. So, for instance, you could search for the end of a corrupt BASIC program to discover where TOP should be with ***BFIND 1900 7C00 13 255** .

Also very bizarre is the inclusion of an encription device which encodes a file with a given password upon saving to disc or cassette — this is claimed to be an unbreakable device but then we've heard that one before. This won't stop the very dedicated but it would allow you to keep a good degree of control over material you wished to keep secret. A nice touch, if rather out of place in this standard set of ROM control utilities.

THE SCYTHE

I've heard this described as a pirate's dream and certainly the opportunity for misuse of this ROM is considerable. Produced by Chalice, this is compatible with their *Vector* tape to disc utility and users of that tool should note that *The Scythe* will allow some programs unusable off disc with *Vector* ,by creating a !BOOT file accessing the ROM menu. An

CONTINUED OVER

interesting touch as is the ability to complement *Vector* in the transfer of very long files from tape to disc from within *The Scythe's* TDISK programming language.

Usual powerful utility commands such as a hex/ASCII memory dump, disassembler of any ROM, bad program repairer, ASCII sequence searcher, byte searcher, disc formatter, ROM enabler or disabler, disc verifier and disc free space display. All well implemented, of course, but it is in the extra commands that *The Scythe* scores well.

You can, for example, spool disassembled code to any filing system (including from paged ROMs); this can then be execed back as a BASIC program compatible with the assembler for easy editing and the code recreated by running the BASIC program thus formed. Or you might care to try the *XLIST command, which acts on a BASIC program by replacing embedded control codes with the '!' character – thus allowing (in conjunction with the *BAD command) the listing and alteration of commercial software. This is a strong utility with many potential uses and, if used with the ROM's memory shifting and swapping commands, offers a wide freedom in examining software – both tape/disc and on sideways ROM.

However, don't assume for a moment that this is only likely to be of interest to the software crackers stealing hard work and money from manufacturers – delving and trying things out is the best way to learn more about the workings of your computer and BBC BASIC and you could do worse than use these well-designed utilities to start you down that road.

Lastly, a very useful utility is included which allows you to build a variety of simple but powerful commands. You can load a specified number of sectors from disc into various sector repair, editor, backup or protection cracker utilities and the command *SECTLOAD will load all sectors specified (ignoring errors) and will then return to the disc fault after loading all possible information. Very useful and, in combination with the other utilities, very powerful.

SLAVE II

For most BBC users I know, *Slave* is, next to *Brom* and *Disc Doctor*, the single most valuable and used ROM in the machine. For sheer range of utilities and power of commands it would be hard to beat.

Yet, like so much with the BBC micro, nothing can be taken as standard and what may at one moment seem like the greatest hardware or software advance will shortly become merely the standard to beat. Enter *Slave II* from A'n'F: not a radical reworking of the ROM, but still worth considering as an extra purchase or upgrade (details not yet known but it seems likely that A'n'F will offer some kind of trade-in).

Besides some minor tinkering with a few of the commands (mainly tidying up rather than bug removing), the main changes come in the BASIC editor and the disc editor. The first adds some extra options such as automatic renumbering when adding a line number already existing, a faster way of moving the editing cursor around the program and screen and so on.

The disc editor, on the other hand, is almost completly reworked. You can now read, edit and rewrite to disc any disc sector on any single, double, 40 or 80 Acorn compatible single density DFS. The display is in four main areas: the top line shows current drive, etc together with an ASCII representation of the byte currently being edited; the main display is an 8 byte per line hex dump, with current editing location highlighted and printable/non-printable characters displayed; to the right are the ASCII equivalents of the printable characters; finally, at the bottom, is a prompt for editing commands. The whole display is clear and the new commands for disc sector manipulation are a delight to use. No doubt you can think of uses for this yourself but an obvious one would be to replace 'unhelpful' bytes inserted by security conscious programmers – possible as long as the rewritten bytes fill the gap exactly. An excellent update.

Figure 4: the display provided by "The Tube Editor".

```
*XREF
KEY10PA*   110
M          110
MO         110
X          210   220   250   260   370
           380   410   420   540
Y          160   170   180   320   330
           340
init       100   610
text        90   150
```

```
*STATUS 4
   Socket number:4
       Rom title:ROM MASTER
  Binary version:0  %00000000
  Version string: 1.00
Copyright string:(C)1985 Chalice
         Length:8K
   Service entry:&802B
  Language entry:No language entry
 Tube relocation:No tube address
```

Figure 5: full information about a specified ROM from "ROM Master".

CONCLUSION

Is there anything here of sufficient interest to the experienced ROM user? I think the answer is probably 'no', although all these ROMs have features which repay close attention. As always, their ultimate utility will depend on your needs and, to a large extent, on the utilities currently in your collection.

Slave II and *The Scythe* both contain very powerful 'accessing' tools and will certainly strip away 'problems' that may have kept you in despair. *Tube Editor* and *Probe 1 are well designed and carefully thought out, though I believe their price will count against them. Finally, Master* and *First Aid* both have the excellent virtues of being cheap and useful.

If pushed to choose one, then without hesitation I'd pick *Slave II*, even though I've not seen the price on it yet!

First Aid 1.1
£15
Altra Roms, 209 North Street, Leeds LS7 2AA (Tel: 0532 451508)

ROM Master
£16
Chalice Software, 3 Merlin Way, Covingham, Swindon SN3 5AN (Tel: 0793 615026)

Tube Editor
£39
Altra Roms
(Note: BASIC Editor 1 and BASIC Editor 2 are £19 and £29 each)

Probe 1
£29
Altra Roms
(Note: Probe 2 is £39)

The Scythe
£24
Chalice Software

Slave II
A'n'F Software, Unit 8, Canal Side Industrial Estate, Woodbine Street East, Rochdale, Lancs (Tel: 0706 341111)

View From The Top

Mike Kent

Micros in our primary schools. Are we making the most of them?

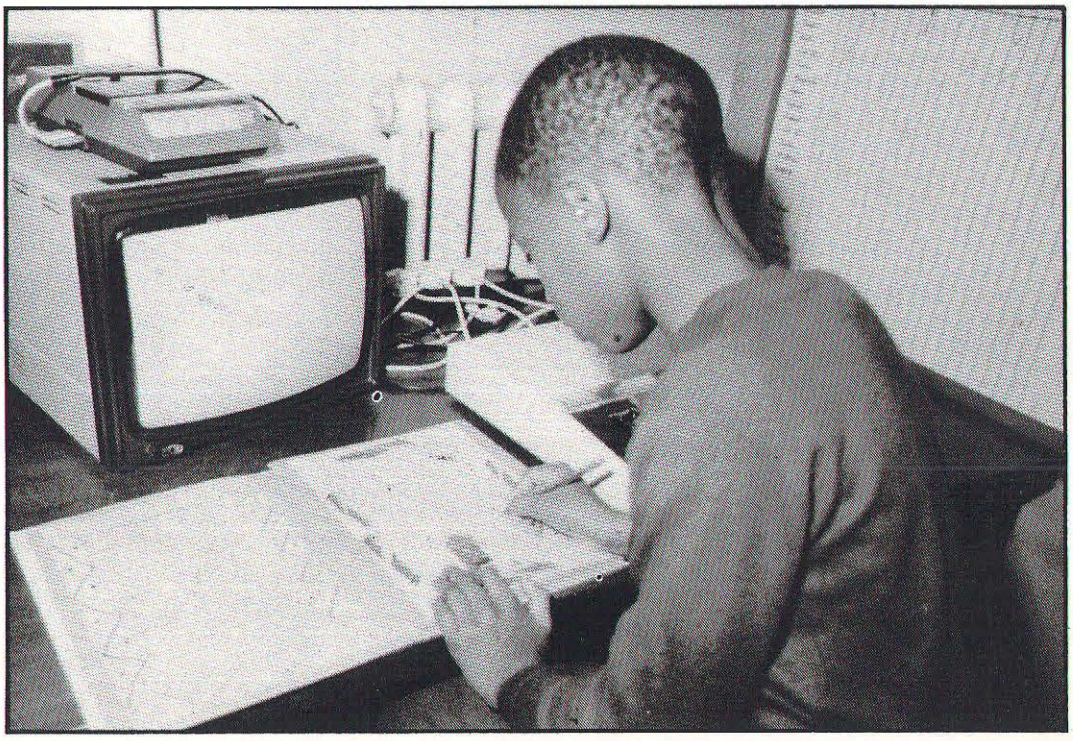
Four years ago, throughout England and Wales, a survey of computing in primary schools was carried out. Thirty two schools owned a computer. By January this year the figure had risen to 26,000 machines, a quite staggering increase by any standards.

This doesn't mean to say, of course, that all the computers are actually being used. How many, I wonder, have ended up in stock cupboards after a few teachers have given the proliferation of connecting wires a cursory glance and decided that computing wasn't for them...and thus, the children they teach? Some, I suspect, may still be lying dormant in their cardboard boxes...

Like many thousands of other schools in the early eighties who had decided to take advantage of the Department of Education scheme for purchasing a computer, we approached the idea with a mixture of excitement and apprehension. As a staff, our combined experience and knowledge of computing was minimal, and reaction to the idea of even buying a computer was decidedly mixed, with several teachers suggesting that the money could more profitably be spent in other ways. However, after much discussion, it was decided that our children should indeed become computerate, and that we should learn alongside them.

The first problem came in deciding which machine to buy. The choice under the DoE scheme narrowed it down to three; the Spectrum, the RML 480z, or the BBC. Since the BBC had received outstanding reviews and had been selected by nearly all the authorities as the ideal machine for Primary use, we tended to favour it. However, on learning that ILEA had put its weight behind the 480z, it seemed sensible to take the advice offered, and opt for this machine instead.

The afternoon our Link arrived was an exciting one, as we unpacked the boxes watched by children who no doubt wondered if their teachers were capable of handling this new technology. No technofear for them, of course. Most of them were already cracking up enormous scores with arcade games on their home Spectrums.



Since the hefty instruction manual made as much sense as a quick lesson in brain surgery to us raw beginners, it was some time before we were in a position to actually connect up, let alone compute, and by five o'clock the less faithful had shrewdly departed for home. Undaunted, we put a program into the cassette recorder (discs weren't available to us then), gave the leads a final check...and were astonished to find that it all worked!

Although computers are supposed to be reliant on human input, ours quickly proved it had a mind of its own. It soon refused to load programs reliably (usually just at the last byte!), and loading a program of any length seemed to take forever. It also produced strange messages on the monitor, and in general gave the staff the impression of being as user friendly as a stick of dynamite! The starter pack of software supplied with the machine was another cause for concern. One of the programs was excellent, two were interesting, and the remainder were unbelievably banal... mere fourth rate text book material given the gloss of modern technology.

DOUBTS

Even those of us who were keen to promote computers began to have serious doubts. In order to use the computer, a teacher would have to wheel the trolleyful of equipment to the classroom, set it up, choose a program from the narrow range of passable material and then load it, assuming, of course, that the equipment wasn't having one of its temperamental days. And what of the classrooms not on the same floor as the computer? How would we get the trolley up or down the stairs? How many children would miss out on computing in the course of a week because we only had one machine to share between them all? (I *know* you want a go on the computer, Simon, but we only have *one* for all two hundred of you. Never mind, you'll definitely have three minutes on April 3rd 2010...

For half a term we struggled with our one computer, though it tended to be based in the library and was used mostly by small groups of Juniors. Certainly, some worthwhile work was done, mainly by fourth years with LOGO. Though interest amongst the staff was sustained it was obvious that most children hardly saw the

CONTINUED OVER

machine at all, mainly due to its weight and immobility.
We could have left it at that, but the interest aroused had given us a determination to succeed, and after more discussion we decided to adopt a different strategy altogether, though it meant virtually starting all over again. Since the most successful *home* computer was the Spectrum, why shouldn't we capitalise on that, and use the computer most children would be using at home? The Spectrum was small, inexpensive, very light to move about, and had an excellent specification. It had been criticised, in terms of use by children, for its multi-function keys which might prove confusing, and the small rubber keys which could be awkward to use. But then, we reasoned, we often under-estimate childrens' ability anyway, and it was likely that our children could actually handle multi-function keys better than many adults. The small keyboard problem was rapidy discounted. Children tend to have small fingers!

Since the cost of a Spectrum was relatively small, we then had an exciting idea. Why not equip *every* classroom with a Spectrum, from Reception upwards? Ambitious, certainly, but if we could trade back our expensive LINK 480Z.....

We approached the Computing Inspectorate rather bashfully to put forward our idea, but it was rejected with a firmness that surprised us. The Spectrum, we were told, was an unreliable machine. It woud break down frequently or overheat, it wouldn't stand up to a heavy workload, and it wasn't suitable for "serious computing". Since reports from friends and colleagues who owned Spectrums differed from this view (including our Chair of Governors, himself a keen Spectrum enthuiast) I became convinced that we were on the right track. My philosophy was essentially a simple one; by giving a computer to each class, the children would not only have much hands-on experience, but also see the computer as an extremely useful and versatile tool in their learning.

We began to see the acquisition of our computers as a goal to be aimed at over the period of one year. Each class would need a TV/monitor, a cassette recorder or microdrive for loading, somewhere safe to lock the equipment away at night, and a method of keeping it plugged together on a semi-permanent basis. No teacher would delight in spending an hour before school sorting out a mass of tangled wires before plugging in! We would also need a good basic software library organised on a class-loan basis.

The theory seemed fine, and now we had to fund the project. Every penny was important. Tins rattled conspicuously at concerts, raffles were abundant, and our parents were marvellous, helping us to raise over six hundred pounds on a sponsored walk alone. The total crept upwards, and together with a generous cheque from a friend of the school, we were in a position to equip all our classrooms after just seven months. I disappeared into our craft room for a day with several children, and together we constructed eight wooden trays to hold the sets of equipment. The trays were divided into small compartments to hold each item firmly in place, and channels were built to contain the wiring. A 4-way 13 amp block was then screwed to each tray.

As the kits arrived, each tray was fitted with a Spectrum, transformer, thermal printer (given free by the distributor), cassette recorder, and demonstration tape, together with the necessary linking leads.

The beauty of this system was that the tray and its contents were very light to carry, even for a child. The equipment, apart from the TV, stayed permanently connected together, and the time taken by the teacher to 'set up' was absolutely minimal. She had only two things to do; connect the mains lead from the 4-way socket to the wall, and plug the Spectrum's aerial lead into the classroom TV, which, of course, could also be used for watching educational broadcasts.

Security for all this equipment might have presented a major headache, but the help and advice given by the security officer soon ensured that each floor of the school had a large safe. A teacher now had only a few yards to walk in order to store her tray and TV at hometime. This,and the ease of setting up, has made certain the equipment is used regularly.

The software library began to expand quickly. Much of the early commercial material for the Spectrum was as dismal as our initial starter pack, but our experience and knowledge were growing..and so was our ability to select a suitable range of programs across the curriculum for both Infants and Juniors. Currently, software has improved significantly in quality, and the best tapes come complete with a workpack of ideas that can occupy children for some weeks.

There were, of course, some teething problems with our equipment, partly due to the very heavy use the computers were receiving from a multitude of tiny fingers! During the first year, three of our Spectrums had to be returned, though they were back within three weeks. We couldn't grumble at that!

So far, then, the children were using commercial software for exploring adventure games, stimulating writing, creating pleasing patterns with a strong maths content via Logo Challenge, learning about a variety of subjects from road safety to ballooning via simulations..and for practising basic skills. There is still much contention about isolating skills acquisi-

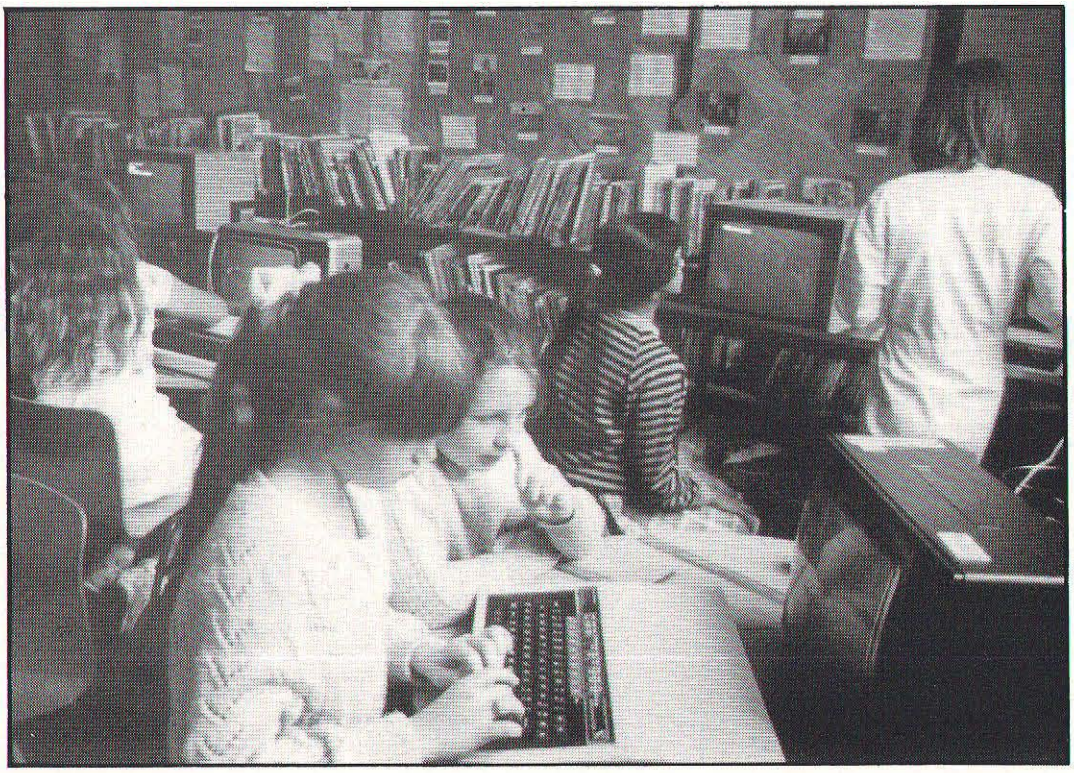
tion from the modern 'broad based' primary curriculm, but it seems logical to assume that children can hardly have an autonomous approach unless they have a healthy 'tool kit' of basic skills with which to explore their environment, and the better 'skills' software tapes can make this a very enjoyable experience.
Naturally, care needs to be taken in selecting software. I recently saw a program which gave the child a couple of minutes to do his tens and units before plunging the world into nuclear war.....

NEXT STAGE, FIRST STEP

Since the revised computer project had taken off so well, in a shorter time than expected, we decided to move a step further. A short while before our project had been initiated, I had bought a BBC Micro for home use, and a colleague had bought a Spectrum. We tended to arrive for school rather tired in the mornings, having spent long nights delving into the fascination of BASIC and making the first hesitant steps towards writing our own educational software. It felt rather like learning a foreign language; tortuous at times but worth it when you got there! From discussing the relative merits of Spectrum versus Beeb, it became obvious that the BBC machine had a lot to offer. It was expensive, and prohibitively so in any quantity, but since the bulk of the quality educational markey was aimed at the BBC, we wished to take advantage of it.
During the course of the next year, we spent part of our school allowance on three BBC's, and after fitting them with Wordwise chips, we were in a position to introduce the children to word processing.

Wordwise is a thoroughly comprehensive word processor, but simplicity itself for children to use as well. After learning a handful of commands, the child can type out his writing, edit it, correct spellings quickly, and experiment with moving sequences of text around. With poetry writing, for example, the word processor really comes into its own. Anyone who has watched a child's face as his or her piece of writing emerges from a printer for the very first time will understand the importance of word processing and the skills it encourages. It is not, of course, a substitute for calligraphy, but an important and exciting extension of it. Since the BBC has such a user-friendly and robust keyboard, children take to typing quickly and easily on it.

It is important to remember that the staff were experimenting and learning, alongside the children. Though my own fascination with computing had begun to fill many of my leisure hours, a project of this kind has to be nurtured slowly. Many teachers are still wary of anything more technical than a tape recorder, though at our own school help is always at hand, and nobody talks in machine code rhetoric. We feel computing must be seen as an activity for all, rather than a technical elite who are able to jargonise skilfully but say very little.



What, then, of the future, for we mustn't simply be content with putting computers within easy reach of our children. There are, I feel, many exciting avenues to be explored.
At present, a group of Juniors spend a morning with me, learning how to write their own programs. BBC BASIC is a very structured dialect, and pleasantly logical to use, but even so I was surprised at how quickly the children began to absorb it. They have experimented with colour and sound, graph work and number crunching, design and animation, and written lots of little routines just for fun...like finding out how much pocket money they are likely to earn over the next decade! It won't be long before they are able to write useful little learning programs for the Infants to try. Certainly, the logic and reasoning power developed by program writing is powerful and readily apparent, stimulating a whole range of new skills.

We must move steadily into the realm of data basing, with its opportunities for classifying, sorting and comparing. We must bring the children to a further understanding of how influential computers are on their lives, and how useful a tool they can be. We must investigate the growing range of peripherals for the computer, which can be helpful to both teacher and child, and we must look to the time when children will not only build models of towns, traffic lights, bridges and lighthouses, but hook a computer up to them and make them work.

We can even extend computer learning into the Nursery, too. At the age of three, my youngest learned most of her alphabet, and how to spell her name, by playing on the computer.

As we have discovered, it is important to make the computer an item of everyday classroom equipment. This, however, is merely the first step.

Mike Kent, Headmaster of Comber Grove Primary School in London, argues that regular hands-on experience is essential for young children, and describes how this was achieved in his own school.

Scruge

Margaret Stanger

Fight the ghosts, explore the caves. An arcade game for your stocking.

On Christmas Eve our hero Scruge finds himself in the middle of a nightmare. He is trying to find four lost gold coins and put them in his safe. Scruge is allergic to the Christmas goodies that surround him and the safe and coins keep moving.

He is chased by the Ghost of Christmas Past but when he is caught he is given a fighting chance. The ghost challenges Scruge to a duel, they miraculously produce swords from their pockets and fight to the death. The progress of the fight and the remaining energy of both parties is displayed on the screen to add to the excitement.

If the ghost wins, or Scruge wanders absentmindedly into the scenery one of four lives is lost. If Scruge survives, his score and experience increase and he lives to fight another ghost if it catches up with him.

After the first coin has been collected and dropped in the safe, the coins and safes become harder to find. The ghosts are replaced by tougher and heavier ghosts, ending with the bondage freak, Marleys ghost and they can all move through the scenery without difficulty.

The scoreboard on the right shows the money collected in the top section. The next section shows the type of ghost and how many have been killed. The third section shows how many lives have been lost. The left hand side of the fourth section shows Scruge's energy. During a fight the right hand section shows how much energy the ghost has to hassle Scruge with.

Apart from the normal movement keys there is the option to freeze (F) or continue (C) the game except during the fighting. Press T to take a coin or D to drop it in the safe. When fighting there is the choice of aiming for the head (H), body (B) or limbs (L). The music can be turned off with Q or turned back on with N.

MOVING ABOUT

Normally in a BASIC program my sprites move eight pixels at a time, and in a machine code game two pixels at a time. As there are only two moving sprites in this game they move four pixels at a time. The ghosts just slide along but Scruge turns to face the way he is going and walking is simulated with two different foot positions.

To find whether Scruge is in a space (colour 0) or in the holly or at a boundary (colour 8) the colour of the screen at his new position is determined by POINTing the screen.

There are 64 rooms altogether but the second 32 are mirror images of the first and constructed from the same data. Each room has eight bytes of data like an enormous user-defined character. Each digit represents a space two characters across by four characters deep. A zero represents a space, and a one an obstacle represented by holly and other Christmas goodies. The scoreboard fills in the space on the right hand side.

CUSTOMISING THE GAME

The game odds could be altered by adjusting line 930 to create ghosts more or less often. Any of the fighting odds could be altered to change the probable outcome of each fight. The visual display of the fighting in PROCSWORD and PROCSWORD2 could be altered to taste. The music could be 'customised' or omitted altogether without affecting the game. The hero and villains could be replaced with your own favourite sprites and the obstacles changed to give your own scenario.

Although this program was developed on the BBC Micro I had the opportunity to RUN it on an ELECTRON. The program worked better on the ELECTRON with the music and the *FX19 calls omitted. Scruge moved rather too slowly at 16 units at a time and moves of 32 units at a time were more acceptable.

GETTING THE ACT TOGETHER

The program SOURCE assembles the machine code,reads in the data and stores it in memory.This data is transferred to location &4000 so that it can be *SAVEd as SCODE.The short header program SCRUGE sets PAGE, LOADs and downloads SCODE and LOADs and RUNs the main game, SCRUGE2.

Type**NEW**
Type in SCRUGE
SAVE "SCRUGE"

Type NEW
Type in SOURCE
SAVE "SOURCE"
RUN SOURCE
Press ‹CTRL› and ‹BREAK›
***SAVE "SCODE" 4000+B00**

PAGE=&1900
Type NEW
Type in SCRUGE2
SAVE "SCRUGE2"

Note that after debugging SOURCE or SCRUGE2 it will be necessary to
Press ‹CTRL› and ‹BREAK›, reset PAGE and type OLD before resaving the program.

To continue debugging LOAD "SOURCE" or CHAIN "SCRUGE" to start again. It may be possible to bypass some of these procedures with a cassette system.

HINTS FOR DEBUGGING

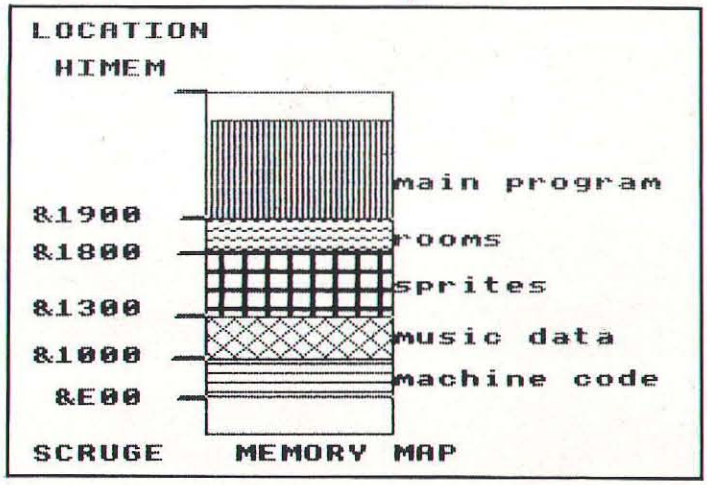
Before running SCRUGE 2 it is possible to check the sprites using:- **FOR S=0TO19: PROCPIC(S*64,1023,S): NEXT** but if some of the sprite data is wrong the game should RUN regardless.

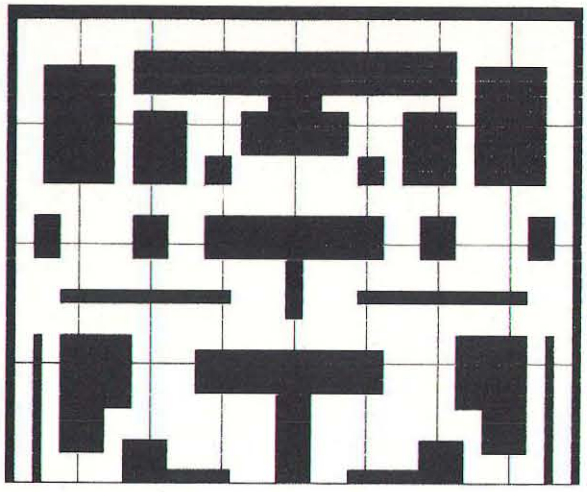
To check the music enter: **REPEAT:PROCTUNE:UNTILFALSE** to make sure that the three parts are correct, and always start and finish together. If the room data is wrong, Scruge could leave a room and find himself in a pile of holly, or in a room with no escape. When I did my debugging I found I needed a short program to print a map of all the 31 rooms together. This program MAP and the printout have been included.

CHAIN "SCRUGE"
Press ‹ESCAPE›

Type in and RUN "MAP" To check the room data if necessary.

To save time when starting to test the program I disabled the ghosts, and put the coins and safes fairly near to the start in room zero.





```
0.
>
   10REM**MAP
   20MODE0
   30VDU23,255,255,255,255,255,255,255,255,255
   40FORY2=0TO3
   50FORY=0TO7
   60FORX2=0TO7
   70X=128
   80REPEAT
   90IF(?(&1800+8*X2+Y+64*Y2)ANDX)>0 PRINTCHR$255;
     ELSEPRINT" ";
  100X=X/2
  110UNTILX=.5
  120NEXT
  130IF(Y2<3 ORY<7) PRINT
  140NEXT:NEXT
  150FORX=128TO992STEP128
  160MOVEX,0:DRAWX,1023:NEXT
  170FORY=0TO992STEP256
  180MOVE0,Y:DRAW1023,Y:NEXT
  190END
```

SCRUGE LISTING

```
   10REM**SCRUGE BY M.STANGER (294B)
   20MODE6:!&FE00=&10200A (264F)
   30PROCINIT (BD8E)
   40PROCINST (F7A5)
   50VDU28,5,23,25,21 (C4E3)
   60*KEY0*LO."SCODE"|MPAGE=&1900|M*LO."SCRUGE2"|M
*T.|MFORI%=0TO&AFCSTEP4:!(&E00+I%)=!(&4000+I%):NEX
T:OLD|MRUN|M (7ABB)
   70*FX138,0,128 (8140)
   80END (CD3D)
   90DEFPROCINST (4FEE)
  100CLS:PRINTTAB(11,1);"SCRUGE -A FANTASY" (DEAB)
  110PRINT'"Scruge has a dream where he has to
  collect four gold coins and put them    in his s
afe, one at a time." (9738)
  120PRINT"If one of the ghosts catches up with hi
mhe must stay and fight.He loses a life  if he run
s out of energy or collides    with the scenery."
(E0E9)
  130PRINT"The game is over when all the coins are
 stashed away or all four lives lost." (8C3C)
  140PRINTTAB(20,12);"KEYS" (3165)
  150PRINT"MOVE"; (7E00)
  160PRINTTAB(10,13);"F";SPC(3);"Freeze";SPC(5);"C
";SPC(3)"Continue" (F55F)
  170PRINTTAB(10)"Z";SPC(3);"Left";SPC(7);"X";SPC(
3)"Right" (BC7B)
  180PRINTTAB(10);":";SPC(3);"Up";SPC(9);"/";SPC(3
)"Down" (42E7)
  190PRINT"MUSIC"; (F26B)
  200PRINTTAB(10);"N";SPC(3);"Noisy";SPC(6);"Q";SP
C(3)"Quiet" (5E4D)
  210PRINT"COINS"; (34D0)
  220PRINTTAB(10);"T";SPC(3);"Take coin";SPC(2);"D
";SPC(3);"Drop coin" (C295)
  230PRINT"FIGHTING  H to hit the Head, B the body
          and L the legs" (BEB4)
  240ENDPROC (5C72)
  250DEFPROCINIT (A0D7)
  260ENVELOPE1,1,0,0,0,0,0,0,127,255,0,255,126,0 (
0C6D)
  270ENVELOPE2,131,255,2,255,1,1,1,127,251,0,251,1
26,55 (5094)
  280VDU23,224,255,129,129,137,157 ,137,129,255 (8
12A)
  290VDU23,225,0,24,60,60,60,24,0,0 (6259)
  300VDU23,226,16,40,86,177,86,40,16,0 (8A2E)
  310VDU23,227,2,6,12,24,48,96,64,0 (7CB4)
  320VDU23,228,0,0,0,126,126,0,0,0 (1249)
  330VDU23,229,0,96,48,24,12,6,3,0 (C044)
  340ENDPROC (E9AA)
```

SCRUGE MAIN PROGRAM

20	Sets the MODE.
30	Accesses initialisation routine.
40	Displays the instructions for the game.
50	Defines a text window for the loading messages.
60	Sets KEY0 to *LOAD and download CODE, and LOAD and RUN Scruge2.
70	Implements the instructions in the KEY 0 buffer.

PROCEDURES

100	PROCINIT
110	Defines labels for machine code routines,sets first flash rate to zero
120	Redefines some colours and puts keyboard auto repeat to minimum
140	PROCROOM(R)
150	Sets first location to HIMEM
160	Sets the base to the first line of data
170	Displays the room using one of the machine code routines
180	Displays the score and cancels any ghost
190-210	Displays coin or safe if any
230	PROCPIC
240	Sets screen location to coincide with graphics coordinates X and Y
250	Displays sprite of number S
260	PROCSETUP Initialises variables, displays first room and Scruge
320	PROCKEY
330	Zeros position changes, new sprite number
340-380	Sets the position changes and new sprite number if a direction key is pressed
390-400	Checks for taking or dropping a coin
410	The freeze and continue routine
420-430	The noise and silence routine
450	PROCSCRUGE
460	Checks for no movement
470	Waits for vertical sync to eliminate flickering
480	Plays tune and waits for vertical sync
490	Blanks out old Scruge
500	Updates position and sprite number
510	Reads palette at new position, and POKEs new sprite
520	Replenishes energy and checks for collision
540	PROCBUMP
550	Checks for coin or safe
560	Checks for collision with scenery
570-600	Scruge has reached the edge of the room, the new room number is calculated from his position
610-620	Scruge is displayed in his new room
630	PROCFALL
640	Scruge is blanked out.
650	Displays a coffin under one of the Scruge pictures
660	Checks for game over
670	Displays new Scruge
680	Replenishes his energy
690	Cancels the ghost
710	PROCFIGHT
720	Blanks out old Scruge
730	Turns Scruge to face ghost
740-760	Calculates and displays the ghosts energy
770	Calculates the experience factor
780	Determines whether Scruge or the ghost starts the fight
790	Scruge's turn: waits for keyboard input of H (for Head), B (for body) or L (for limbs). Aiming for the head can do more damage, but he is more likely to miss.
800	Calculates any damage when aiming for the head
810	Calculates any damage when aiming for the body
820	Calculates the damage when aiming for the limbs, if Z is zero he has missed altogether

CONTINUED OVER

820-830	Caculates the decrease in the ghosts energy
850	Shows Scruge's sword
860	Checks for 'death' of ghost
870-880	Ghost's turn. Calculates any damage
890	Calculates any damage to Scruge
900	Checks for death of Scruge
910	Shows the ghost's sword and goes back to line 790. The procedure is not over until one of the combatants 'dies'
920	PROCGHOST
930	If there is no ghost present, and the odds are unfavourable, no ghost appears
940	Starts a new ghost if necessary
950-960	The ghost moves nearer to Scruge
970	Blanks old ghost
980	Updates ghost's position and displays ghost
990	If Scruge and the ghost are too close they fight to the death
1010	PROCSTART starts a new ghost from one of the sides
1050	PROCTAKE
1060-1080	Skips if Scruge is not near the coin
1090	Cancels ghost if any
1100-112	Updates the score
1130	PROCMONEY Prints the money
1150	PROCDROP
1160-1180	Skips if Scruge is not near the safe
1190	Cancels ghost if any
1200	Blanks the safe
1210Places the next coin	
1220	Updates the score, swaps the ghost for a bigger one
1230	PROCSCORE
1240-1260	Draws the scoreboard frame
1270	Prints coin or safe
1280	Prints money, ghosts killed, and the current ghost
1290	Displays four Scruge's
1300	Prints a coffin under a Scruge for each life lost
1310-1320	Displays Scruge's energy
130	PROCBETTER Increases Scruge's energy if he is flagging
1360	PROCENERGY Updates and dispays Scruge's energy
1400	PROCHASSLE Updates and displays the ghosts energy
1440	PROCGET
1450	Removes keyboard auto repeat
1460	Prompts Scruge to attack
1470	Waits for keyboard input and puts auto repeat back to minimum
1480	Blanks out message
1490	PROCVAN
1500	Blanks out ghost
1510	Updates score, restarts the music
1520	Blanks out the ghost's energy as there is no longer a fighting ghost
1530	PROCHIT Displays a message
1550	PROCSWORD If Scruge hit the ghost his sword is shown, and a channel 0 sound is made
1610	PROCSWORD2 There is a similar display if the ghost hits Scruge

PROCEDURES

90	PROCINST Dislays the instructions.
250	PROCINIT
270-280	Defines the envelopes.
290-330	Defines the coin, safe, coffin and three versions of the sword.

SOURCE LISTING

```
   10REM**SOURCE (BA7A)
   20MODE6 (C941)
   30PROCINIT (BD8E)
   40PROCAS (A5B7)
   50FORX%=0TO&FFCSTEP4:!(&4000+X%)=!(&E00+X%):NEX
T (0E8D)
   60END (14FF)
   70DEFPROCINIT (F58E)
   80FORI%=0TO&5FF:READA%:?(&1300+I%)=A%:NEXT (B38
B)
   90I%=0:FORJ%=0TO&2FF:?(&1000+J%)=0:NEXT (8F9C)
  100REPEAT:READA,B:?(&1000+8*I%)=1:?(&1002+8*I%)=
1:?(&1004+8*I%)=A:?(&1006+8*I%)=B:I%=I%+1:UNTILA=0
:?(&1002+8*I%-8)=0 (B182)
  110!&FF4=&1000 (C4C6)
  120!&FF8=&1000+8*I% (C930)
  130REPEAT:READA,B:?(&1000+8*I%)=2:?(&1002+8*I%)=
2:?(&1004+8*I%)=A:?(&1006+8*I%)=B:I%=I%+1:UNTILA=0
:?(&1002+8*I%-8)=0 (C210)
  140!&FFC=&1000+8*I% (4F9B)
  150REPEAT:READA,B:?(&1000+8*I%)=3:?(&1002+8*I%)=
2:?(&1004+8*I%)=A:?(&1006+8*I%)=B:I%=I%+1:UNTILA=0
:?(&1002+8*I%-8)=0 (F32C)
  160ENDPROC (2117)
  170DEFPROCAS (82C8)
  180OSBYTE=&FFF4:OSWORD=&FFF1 (FA62)
  190FOR PASS=0TO2 STEP2 (3500)
  200P%=&E00 (0576)
  210[OPT PASS (A3F4)
  220.SPIC LDY#31 (3309)
  230.SP1 LDA(&70),Y (6F4A)
  240EOR(&80),Y:STA(&80),Y (F471)
  250TYA:BEQ OUT:DEY:JMP SP1:.OUT RTS (0F19)
  260.LPIC LDY#31 (4777)
  270.LP1 LDA(&70),Y (DEC5)
  280EOR(&80),Y:STA(&80),Y (0D34)
  290TYA:BEQ FEET:DEY:JMPLP1 (8EB1)
  300.FEET CLC:LDA&70:ADC#32:STA&70 (8E11)
  310LDA&71:ADC#0:STA&71 (8590)
  320JSR DOWN1:JMP SPIC (7849)
  330.ROOMUSDOWN LDY#7:.RUSDOWN2 LDA(&74),Y:STA&76
:STA&77 (4187)
  340TYA:PHA:JSR SCAN:PLA:TAY:DEY:TYA:CMP#255:BEQ
F (EA18)
  350CLC:LDA&80:ADC#0:STA&80:LDA&81:ADC#8:STA&81:J
MP RUSDOWN2 (CFEA)
  360.ROOMRWUP LDY#0:.RWUP2 LDA(&74),Y:STA&76:STA&
77 (E53E)
  370TYA:PHA:JSR SCAN:PLA:TAY:INY:TYA:CMP#8:BEQ F
(1667)
  380CLC:LDA&80:ADC#0:STA&80:LDA&81:ADC#8:STA&81:J
MP RWUP2:.F RTS (5608)
  390.SCAN LDX#8:.LOOP LDA&77:AND#128:BEQ SKIP:JSR
 HOLLY (FD8E)
  400.BACK CLC:ASL&77:DEX:TXA:BNE LOOP:RTS (70DD)
  410.HOLLY JSR MOTIF:JSR DOWN1:JSR BORDER (5651)
  420JSRALONG:JSR BORDER:JSR DOWN1:JSR MOTIF:JSR A
LONG:RTS (9EC7)
  430.SKIP CLC:LDA&80:ADC#&40:STA&80:LDA&81:ADC#0:
STA&81:JMP BACK (680E)
  440.BORDER TXA:PHA:STY&79:AND#7:STA&79:CMP#1:BNE
 BORD:LDA#7:STA&79 (CC19)
  450.BORD LDA#&00:STA&70:LDA#&15:STA&71:LDX&79 (8
2D5)
  460.BORD1 TXA:BEQ BORD2:CLC:LDA&70:ADC#&40:STA&7
0 (2D12)
  470LDA&71:ADC#0:STA&71:DEX:JMP BORD1 (6E2F)
  480.BORD2 JSRLPIC:PLA:TAX:RTS (EED0)
  490.MOTIF TXA:PHA:LDA&76:AND#7:STA&76:LDA#&00:ST
A&70:LDA#&15:STA&71:LDX&76 (F448)
  500.M1 TXA:BEQ M2:CLC:LDA&70:ADC#&40:STA&70 (961
E)
  510LDA&71:ADC#0:STA&71:DEX:JMP M1 (CF0B)
  520.M2 JSR LPIC:PLA:TAX:RTS (0AF8)
  530.DOWN1 CLC:LDA&80:ADC#&80:STA&80:LDA&81:ADC#2
:STA&81:RTS (27C0)
  540.ALONG SEC:LDA&80:SBC#&60:STA&80:LDA&81:SBC#7
:STA&81:.A2 RTS (27D6)
  550.TUNE LDA&450:BEQ A2 (86B0)
  560PHA:TXA:PHA:TYA:PHA (8BC8)
  570CLC:LDA&424:ADC#8:STA&422:LDA&425:ADC#0:STA&4
23 (1546)
  580LDA&422:CMP&FF8:BNE VOICE1:LDA&423:CMP&FF9:BN
E VOICE1:JSR BEGIN (EE1C)
  590.VOICE1 LDY#&FF:LDX#250:LDA#128:JSR OSBYTE:CP
X#&0F:BNE VOICE2 (DEA7)
  600CLC:LDA&424:ADC#8:STA&424:LDA&425:ADC#0:STA&4
25 (DAD5)
  610LDA#7:LDX&424:LDY&425:JSR OSWORD (0E82)
  620.VOICE2 LDY#&FF:LDX#249:LDA#128:JSR OSBYTE:CP
X#&0F:BNE VOUT (0A4A)
  630CLC:LDA&428:ADC#8:STA&428:LDA&429:ADC#0:STA&4
29 (A0EB)
  640LDA#7:LDX&428:LDY&429:JSR OSWORD (5815)
  650CLC:LDA&42C:ADC#8:STA&42C:LDA&42D:ADC#0:STA&4
2D (E9CA)
  660LDA#7:LDX&42C:LDY&42D:JSR OSWORD (7E56)
  670.VOUT PLA:TAY:PLA:TAX:PLA:RTS (2DBD)
  680.BEGIN LDX#12 (40C6)
  690.BG2 LDA&FF3,X:STA&423,X (4256)
  700DEX:TXA:BNE BG2:RTS (CF32)
  710]:NEXT (1F6D)
```

```
720!&FE6=SPIC:!&FE8=LPIC (F19A)
730!&FEA=ROOMUSDOWN:!&FEC=ROOMRWUP (8387)
740!&FEE=TUNE:!&FF0=BEGIN (FDC0)
750ENDPROC (3FEB)
760DATA0,0,21,2,2,0,0,0 (BA25)
770DATA21,63,63,63,5,20,60,60 (1395)
780DATA0,42,10,15,10,40,60,60 (515B)
790DATA0,0,0,0,0,0,0,0 (8101)
800DATA0,0,0,0,0,0,5,5 (8DD4)
810DATA45,45,60,60,60,60,15,0 (A9DB)
820DATA60,15,60,60,60,45,0,0 (0BEB)
830DATA0,10,0,0,0,10,10,15 (A198)
840DATA0,0,21,2,2,0,0,0 (9DB4)
850DATA21,63,63,63,5,20,60,60 (739B)
860DATA0,42,10,15,10,40,60,60 (A0FE)
870DATA0,0,0,0,0,0,0,0 (327D)
880DATA0,0,0,0,0,0,0,0 (96A7)
890DATA45,15,45,60,60,60,5,5 (75FE)
900DATA60,60,30,30,60,60,10,15 (7F74)
910DATA0,0,0,0,0,0,0,0 (8CC2)
920DATA0,0,0,0,0,0,0,0 (4D91)
930DATA0,21,5,15,5,20,60,60 (33BA)
940DATA42,63,63,63,10,40,60,60 (5CB5)
950DATA0,0,42,1,1,0,0,0 (D573)
960DATA0,5,0,0,0,5,5,15 (F9E1)
970DATA60,15,60,60,60,30,0,0 (F72D)
980DATA30,30,60,60,60,60,15,0 (7CED)
990DATA0,0,0,0,0,0,10,10 (99AE)
1000DATA0,0,0,0,0,0,0,0 (EA88)
1010DATA0,21,5,15,5,20,60,60 (2FDA)
1020DATA42,63,63,63,10,40,60,60 (DF4C)
1030DATA0,0,42,1,1,0,0,0 (B088)
1040DATA0,0,0,0,0,0,0,0 (29CE)
1050DATA60,60,45,45,60,60,5,15 (2648)
1060DATA30,15,30,60,60,60,10,10 (9C96)
1070DATA0,0,0,0,0.0,0,0 (E89D)
1080DATA0,0,0,0,0,0,0,20 (AF8D)
1090DATA21,63,47,47,5,20,60,60 (73F7)
1100DATA42,63,31,31,10,40,60,60 (09D9)
1110DATA0,0,42,1,1,0,0,40 (353A)
1120DATA20,20,20,0,0,0,0,5 (82DE)
1130DATA60,60,60,60,60,30,10,15 (0AD2)
1140DATA60,60,60,60,60,45,15,0 (48A4)
1150DATA40,40,40,0,0,0,10,0 (CA5D)
1160DATA0,0,0,0,0,0,0,20 (B960)
1170DATA21,63,47,47,5,20,60,60 (8B90)
1180DATA42,63,31,31,10,40,60,60 (C55B)
1190DATA0,0,42,1,1,0,0,40 (7B7D)
1200DATA20,20,20,0,0,0,5,0 (FD02)
1210DATA60,60,60,60,60,30,15,0 (1485)
1220DATA60,60,60,60,60,45,5,15 (B336)
1230DATA40,40,40,0,0,0,0,10 (00E8)
1240DATA0,0,21,2,2,0,0,20 (1DA5)
1250DATA21,63,63,63,5,20,60,60 (DCA9)
1260DATA42,63,63,63,10,40,60,60 (3492)
1270DATA0,0,0,0,0,0,0,40 (BEB4)
1280DATA20,20,20,0,0,0,0,5 (D6F4)
1290DATA60,60,60,60,60,30,10,15 (3E20)
1300DATA60,60,60,60,60,45,15,0 (9B3C)
1310DATA40,40,40,0,0,0,10,0 (F669)
1320DATA0,0,21,2,2,0,0,20 (3BDE)
1330DATA21,63,63,63,5,20,60,60 (C5FC)
1340DATA42,63,63,63,10,40,60,60 (417D)
1350DATA0,0,0,0,0,0,0,40 (3BE1)
1360DATA20,20,20,0,0,0,5,0 (CEE4)
1370DATA60,60,60,60,60,30,15,0 (0DD0)
1380DATA60,60,60,60,60,45,5,15 (3BC8)
1390DATA40,40,40,0,0,0,0,10 (B4F8)
1400DATA0,0,4,12,0,4,12,12 (16C6)
1410DATA4,12,12,12,12,12,12,12 (C4B1)
1420DATA0,8,12,12,8,12,12,12 (37BF)
1430DATA0,0,0,8,0,0,8,8 (BE50)
1440DATA0,4,12,12,12,0,0,0 (BC92)
1450DATA12,12,12,12,12,5,3,3 (36F8)
1460DATA8,12,12,12,12,0,2,2 (7DA3)
1470DATA0,0,8,8,8,0,0,0 (184B)
1480DATA60,40,40,40,61,60,40,40 (264A)
1490DATA60,20,20,62,62,60,20,20 (2937)
1500DATA60,0,0,21,63,60,0,0 (91C1)
1510DATA40,40,40,40,40,40,40,40 (3D37)
1520DATA40,61,60,40,40,40,61,60 (CD61)
1530DATA62,62,60,20,20,62,62,60 (56D9)
1540DATA21,63,60,0,0,21,63,60 (B855)
1550DATA40,40,40,40,40,40,40,40 (5B76)
1560DATA0,0,21,46,0,21,63,12 (6762)
1570DATA4,46,12,12,12,12,12,12 (6A84)
1580DATA0,42,29,12,8,29,29,12 (2327)
1590DATA0,0,0,42,0,0,42,8 (61E9)
1600DATA0,21,63,12,12,0,0,0 (2A49)
1610DATA12,12,12,12,12,5,5,15 (AD10)
1620DATA8,29,29,12,12,0,0,10 (06D5)
1630DATA0,0,42,8,8,0,0,0 (18A3)
1640DATA0,0,0,5,5,0,5,15 (14BC)
1650DATA5,5,5,5,15,15,15,15 (5ED5)
1660DATA0,0,0,5,15,10,15,15 (D5A9)
1670DATA0,0,0,0,0,0,0,10 (8AAF)
1680DATA15,5,0,5,5,0,0,0 (FB46)
1690DATA15,15,15,15,5,5,5,5 (2A5C)
1700DATA15,15,10,15,5,0,0,0 (E127)
1710DATA10,0,0,0,0,0,0,0 (58FD)
1720DATA0,0,0,0,0,0,0,4 (5FA1)
1730DATA0,0,4,4,12,12,12,8 (1C13)
1740DATA0,12,0,0,12,0,8,4 (4656)
1750DATA0,0,8,0,0,8,0,12 (43F3)
1760DATA4,4,4,4,4,12,12,12 (4C85)
1770DATA0,1,1,5,5,1,1,1 (3715)
1780DATA0,7,7,15,15,7,7,7 (BA2E)
1790DATA0,3,3,15,15,3,3,3 (00E1)
1800DATA21,4,4,4,4,12,12,12 (87EB)
1810DATA0,0,0,0,0,8,10,8 (A434)
1820DATA0,0,0,21,5,5,15,15 (1949)
1830DATA0,0,0,0,0,0,10,10 (1719)
1840DATA12,12,29,29,12,12,12,12 (C3E1)
1850DATA13,13,47,47,13,13,13,8 (5955)
1860DATA15,15,15,15,15,15,15,0 (6451)
1870DATA15,15,3,3,3,15,15,0 (1508)
1880DATA63,21,0,0,0,0,21,0 (0861)
1890DATA63,1,43,21,21,21,63,0 (1CC6)
1900DATA47,31,42,0,0,0,63,0 (B403)
1910DATA42,0,0,0,0,0,0,5 (B830)
1920DATA21,0,0,0,0,0,0,0 (7CE7)
1930DATA63,42,21,0,0,0,0,63 (3311)
1940DATA63,7,23,42,42,42,42,63 (7E38)
1950DATA31,42,0,0,0,0,0,42 (4894)
1960DATA6,6,12,8,12,8,8,12 (6DDD)
1970DATA8,12,9,1,8,4,0,4 (65E6)
1980DATA0,8,1,1,12,12,12,12 (76DA)
1990DATA0,12,12,12,8,12,2,6 (F365)
2000DATA2,2,0,4,0,4,0,0 (032B)
2010DATA0,9,1,12,12,12,4,12 (06B5)
2020DATA12,12,12,12,8,12,2,2 (32FB)
2030DATA0,8,0,8,0,4,9,9 (6C18)
2040DATA0,69,138,138,207,207,207,207 (3633)
2050DATA207,207,69,69,207,207,207,207 (2F8C)
2060DATA138,207,0,0,207,207,207,207 (9EC9)
2070DATA0,0,138,138,138,138,138,138 (F46A)
2080DATA207,207,69,69,69,0,0,0 (A77D)
2090DATA207,207,207,207,207,207,69,0 (B995)
2100DATA207,207,207,207,207,207,138,0 (D519)
2110DATA138,138,138,138,138,207,207,69 (107F)
2120DATA0,65,130,130,195,195,195,195 (273E)
2130DATA195,195,65,65,195,195,195,195 (FEF8)
2140DATA130,195,0,0,195,195,195,195 (5A4E)
2150DATA0,0,130,130,130,130,130,130 (D41A)
2160DATA195,195,65,65,65,0,0,0 (E428)
2170DATA195,195,195,195,195,195,65,0 (5F2C)
2180DATA195,195,195,195,195,195,130,0 (7600)
2190DATA130,130,130,130,130,195,195,65 (861C)
2200DATA0,68,136,136,204,204,204,204 (9D0E)
2210DATA204,204,68,68,204,204,204,204 (561C)
2220DATA136,204,0,0,204,204,204,204 (13B2)
2230DATA0,0,136,136,136,136,136,136 (3F16)
2240DATA204,204,68,68,68,0,0,0 (F915)
2250DATA204,204,204,204,204,204,68,0 (2893)
2260DATA204,204,204,204,204,204,136,0 (7FC2)
2270DATA136,136,136,136,136,204,204,68 (259B)
2280DATA0,64,128,128,192,192,192,192 (035C)
2290DATA192,192,64,64,192,192,144,144 (B60E)
```

CONTINUED OVER

```
2300DATA128,192,0,0,192,48,144,96 (D1D5)
2310DATA0,0,128,128,32,128,128,128 (7E35)
2320DATA192,144,144,96,192,16,32,48 (966D)
2330DATA48,144,96,192,144,48,96,0 (F1CC)
2340DATA192,192,192,48,144,96,128,0 (0520)
2350DATA128,128,32,128,128,192,192,64 (08F9)
2360DATA255,128,128,128,143,143,143,143 (86F9)
2370DATA255,0,0,3,243,243,240,243 (C254)
2380DATA255,0,0,255,255,255,0,240 (8310)
2390DATA255,0,0,255,255,255,7,63 (BC67)
2400DATA255,0,0,255,255,255,224,252 (60F4)
2410DATA255,0,0,255,255,255,0,15 (97E9)
2420DATA255,0,0,192,207,207,15,207 (4A28)
2430DATA255,1,1,1,241,241,241,241 (756A)
2440DATA143,143,143,143,128,128,156,156 (EB79)
2450DATA243,243,243,243,0,0,3,3 (6B03)
2460DATA240,240,243,243,0,0,195,195 (CCF1)
2470DATA63,63,128,128,0,0,255,255 (4EF4)
2480DATA252,252,1,1,0,0,255,255 (1586)
2490DATA15,15,207,207,0,0,195,195 (CA63)
2500DATA207,207,207,207,0,0,192,192 (A395)
2510DATA241,241,241,241,1,1,57,57 (40E8)
2520DATA156,128,128,131,128,128,147,147 (D831)
2530DATA3,0,0,255,0,0,252,252 (A99F)
2540DATA195,0,0,255,0,0,0,7 (6782)
2550DATA255,1,1,129,1,0,0,255 (7287)
2560DATA255,128,128,129,128,0,0,255 (4B89)
2570DATA195,0,0,255,0,0,0,192 (7FF9)
2580DATA192,0,0,255,0,0,63,63 (F6EC)
2590 DATA57,1,1,193,1,1,201,201 (39D3)
2600DATA147,147,147,147,147,147,144,144 (BCD7)
2610DATA252,252,252,224,224,231,7,7 (079B)
2620 DATA7,7,0,0,0,192,192,255 (8F74)
2630DATA255,255,3,3,3,3,3,131 (BDB8)
2640DATA255,255,192,192,192,192,192,195 (3A22)
2650DATA192,192,0,0,0,3,3,255 (D4D3)
2660DATA63,63,63,7,7,231,224,224 (335D)
2670DATA201,201,201,201,201,201,9,9 (6B3D)
2680DATA117,5,117,5,145,5,145,5,137,5,129,5,125,5
,117,5,109,5,117,5,125,5,129,5,137,5,145,15 (0A03)
2690DATA117,5,117,5,145,5,145,5,137,5,129,5,125,5
,117,5,109,5,117,5,125,5,129,5,137,5,145,15 (AE67)
2700DATA145,5,149,5,137,5,145,5,149,5,157,5,165,5
,145,5,137,5,129,5,117,5,125,5,129,5,137,5,129,5,1
37,5,145,10 (8D68)
2710DATA149,5,145,5,145,5,137,5,129,5,125,5,117,3
0 (C217)
2720DATA0,5 (F200)
2730DATA117,5,69,20,65,20,69,20,57,20,65,20,49,20
,57,40,69,20,69,10,77,10,69,10,65,10,57,10,49,10,6
5,10,69,10,65,15,57,5,57,30 (31F0)
2740DATA0,5 (5791)
2750DATA117,5,117,20,113,20,117,20,105,20,113,20,
97,20,105,40,117,20,117,10,129,10,117,10,113,10,10
5,10,97,10,113,10,117,10,113,15,105,5,105,30,0,5 (
B8D9)
```

SOURCE VARIABLES

A%	Data read in
I%	Loop variable
J%	Loop variable
P%	Program pointer
X%	Loop variable
PASS	Loop variable for two pass assembly

MAIN PROGRAM

20	Sets the MODE
30	Reads in the data
40	Assembles the machine code
50	Uploads the data and assembled machine code to locations from &4000.

PROCEDURES

70	PROCINIT
80	Reads in the data for the sprites and the rooms
90	Clears an area for the music data
100	Creates an eight byte data block for each note in the first part of the tune. Voice1, envelope 1 and pitch and duration from the data from line 2680
110	Stores the location of the first note in part 1 of the tune
120	Stores the location of the first note in part 2 of the tune
130	Creates an eight byte data block for each note in the second part of the tune. Voice 2, envelope 2 and the pitch and duration from the data from line 2730
140	Stores the location of the first note in part 3 of the tune
150	Creates an eight byte data block for each note in part 3 of the tune. Voice 3, envelope 2 and the pitch and duration from the data from line 2750.

PROCAS:

There are five machine code routines that are used in the main program:-

LPIC	Pokes all the sprite to the screen.
ROOMUSDOWN	Inverts the data for one of the rooms from 0 to 31 to be a room from 32 to 63.
ROOMRWUP	Displays a room from 0 to 31.
TUNE	Plays the tune.
BEGIN	Returns the note numbers of all parts of the tune to their initial values. These five routines share many subroutines.

LPIC

220-250	POKEs half the sprite to the screen.
260-290	Displays the top half of the sprite
300-310	Increases the base and location for the second half
320	Displays the bottom half of the sprite

ROOMUSDOWN

330-350	Scans the 8 data bytes from left to right and bottom to top.The other routines are shared with ROOMRWUP

ROOMRWUP

360-380	Scans the eight data bytes from left to right and top to bottom.
390	Scans the chosen bit of data, and skips if it is zero to leave a blank two character spaces by four character spaces.
400	Shifts the data byte to examine the next bit.
410-420	Displays the chosen four designs if the data bit is set.
430	Increases the screen location for the next bit.
440	ANDs temporary store &79 with 7 to give the border design. Design 7 (holly) was substituted for design 1 (window) because the display looked better that way.
450-480	Displays the two border sprites.
490-520	Displays the two motif sprites, chosen by ANDing the data byte with 7.
530	Increases the screen location, vertically
540	Moves the screen location up and to the right.
550	Compares for silence flag.
560	Transfers the registers.
570-580	Checks for last note of part 1
590	Checks if the voice 1 sound channel is empty.
600	Adds 8 for next part 1 data block.
610	Sounds the next note in part 1.
620	Checks if channel 2 is empty.
630	Adds 8 for next part 2 data block.
640	Sounds note 2.
650	Adds 8 for next part 3 data block.
660	Sounds note 3.
670	Transfers the registers.

BEGIN

680-700	Initalises note numbers I%, J% and K%.
720-740	Stores the locations of these machine code routines. It is not essential, but it makes them easier to find later.

SPRITE DATA

760	&1300	Scruge facing right, position 1
840	&1340	Scruge facing right,position 2
920	&1380	Scruge facing left, position 1
1000	&13C0	Scruge facing left, position 2
1080	&1400	Scruge facing front,position 1
1160	&1440	Scruge facing front,position 2
1240	&1480	Scruge facing back,position 1
1320	&14C0	Scruge facing back,position 2
1400	&1500	Tree
1480	&1540	Window
1560	&1580	Tree with snow
1640	&15C0	Star
1720	&1600	Package
1800	&1640	Bottles
1880	&1680	Glasses
1960	&16C0	Holly
2040	&1700	Ghost of Christmas Past
2120	&1740	Ghost of Christmas Present
2200	&1780	Ghost of Christmas Future
2280	&17C0	Marley's Ghost

ROOM DATA

2360	&1800	Rooms 0 to 7
2440	&1840	Rooms 8 to 15
2520	&1880	Rooms 16 to 23
2600	&18C0	Rooms 24 to 31

MUSIC DATA

2680-2720	Part 1:- The melody of God Rest You Merry, Gentlemen.
2730-2740	Part 2:- The melody of the sinister Dies Irae is used as counterpoint to the Christmas Carol.
2750	Part 3:- The harmony of Dies Irae.

SCRUGE TWO LISTING

```
 10REM**SCRUGE2 (9047)
 20ONERROR GOTO1670 (77BF)
 30PRINTTAB(1,24);'"Press SPACE to play":REPEAT:
UNTILGET=32 (00CE)
 40MODE2:!&FE00=&10200A:PROCINIT (EEF0)
 50PROCSETUP:REPEAT:CALLTUNE:PROCKEY:PROCSCRUGE:
CALLTUNE:PROCGHOST:UNTILLIVES=0 OR MONEY=80 (3875)
 60CLS:FORX=0TO1279STEP64:FORY=32TO96STEP64:FORY
1=0TO896STEP896:PROCPIC(X,Y+Y1,7+RND(8)):NEXT:NEXT
:NEXT (8932)
 70VDU4:!&FE00=&10200A:PRINTTAB(7,6);"SCRUGE":PR
INTTAB(6,11);"SCORE ";MONEY+KILLED (3095)
 80PRINTTAB(4,16);MONEY*1.25;"% COMPLETED":GOTO3
0 (A17F)
 90END (7931)
100DEFPROCINIT (98FF)
110LPIC=&E10:ROOMUSDOWN=&E32:ROOMRWUP=&E57:TUNE=
&F35:BEGIN=&FD7:*FX9,0 (84A7)
120VDU19,10,13,0,0,0,19,8,7,0,0,0,19,13,3,0,0,0,
19,11,3,0,0,0:*FX11,1 (6F84)
130CALL BEGIN:T%=1:ENDPROC (6C16)
140DEFPROCROOM(R) (E327)
150!&80=HIMEM:CLS:CALLTUNE (5A43)
160!&74=&1800+8*(R MOD 32) (7BEC)
170IFR<32 !&74=&1800+8*R:CALL ROOMRWUP ELSE !&74
=&1800+8*(56-8*(INT(R/8))+ R MOD 8):CALL ROOMUSDOW
N (4A3F)
180PROCSCORE:GFLAG=FALSE:IF R<> COIN AND R<> SAF
E ENDPROC (D64E)
190REPEAT:CALLTUNE:XCOIN=32*(RND(29)):YCOIN=64*(
RND(31)):UNTIL POINT(XCOIN+16,YCOIN+16)=0:MOVEXCOI
N,YCOIN:GCOL0,13 (2392)
200IF R=SAFE AND COINFLAG PRINTCHR$224:ENDPROC (
94E8)
210IF R=COIN AND NOTCOINFLAG PRINTCHR$225 (33CA)
220ENDPROC (F9E3)
230DEFPROCPIC(X,Y,S) (14E0)
240!&80=HIMEM+16*(X DIV 32)+640*((1023-Y) DIV 32
) (E24C)
250!&70=&1300+S*64:CALL LPIC:ENDPROC (E77C)
260DEFPROCSETUP (C797)
270VDU5:R=0:S=0:G=0:X=319:Y=639:GFLAG=0:GHOST=16
(D197)
280LIVES=4:ENERGY=63:HASSLE=1 (D21D)
290XCOIN=0:YCOIN=0:MONEY=0:KILLED=0 (3B5E)
300COIN=RND(8+.8*MONEY)-1:SAFE=0:COINFLAG=FALSE
(F95F)
310PROCROOM(R):PROCPIC(X,Y,S):ENDPROC (120A)
320DEFPROCKEY (3AA4)
330DX=0:DY=0:NEWS=S:*FX15,1 (212D)
340IFINKEY(-98) DX=-16:NEWS=2 (F133)
350IFINKEY(-67) DX=16:NEWS=0 (7B73)
360IFINKEY(-88) DY=16:NEWS=6 (60F7)
370IFINKEY(-105) DY=-16:NEWS=4 (50A9)
380IF(DX <>0 OR DY<>0) ENDPROC (A832)
390IFINKEY(-36) PROCTAKE:ENDPROC (93D6)
400IFINKEY(-51) PROCDROP:ENDPROC (88AE)
410IFINKEY(-68):REPEAT:UNTIL INKEY(-83):ENDPROC
(ADCF)
420IFINKEY(-17) T%=FALSE (895F)
430IFINKEY(-86) T%=TRUE:CALL BEGIN (D452)
440ENDPROC (8067)
450DEFPROCSCRUGE (0180)
460IFDX=0:IFDY=0 ENDPROC (5530)
470*FX19 (66F2)
480CALLTUNE:*FX19 (D957)
490PROCPIC(X,Y,S+G) (3414)
500X=X+DX:Y=Y+DY:S=NEWS:G=(G EOR 1) (E5B7)
510W=POINT(X-4,Y):PROCPIC(X,Y,S+G) (4535)
520PROCBETTER:IFW<>0 PROCBUMP (4930)
530ENDPROC (D189)
540DEFPROCBUMP (38CC)
550IFW=13 ENDPROC (D948)
560IFW<8 ANDW>0 PROCFALL:ENDPROC (E0B1)
570IFX+2*DX<16 X=959:R=R-1 (8573)
580IFX+2*DX>991 X=31:R=R+1 (B4CB)
590IFY+4*DY<16 Y=959:R=R+8 (579C)
600IFY+4*DY>991 Y=31:R=R-8 (2071)
610PROCROOM(R):S=NEWS:G=(G EOR 1) (D77D)
620PROCPIC(X,Y,S+G):ENDPROC (6E06)
630DEFPROCFALL:SOUND0,-15,0,10:PROCPIC(X,Y,S+G)
(08E5)
640CALL BEGIN:LIVES=LIVES-1 (5F11)
650MOVE960+(4-LIVES)*64,320:GCOL0,5:PRINTCHR$226
(9E96)
660FORZ%=1TO10000:NEXT:IFLIVES=0 ENDPROC (94E
670X=X-4*DX:Y=Y-4*DY:PROCPIC(X,Y,S+G) (FC35)
680DE=62:ENERGY=0:GCOL0,6:PROCENERGY (97A6)
690IF GFLAG DH=1-HASSLE:PROCVAN (5BEF)
700ENDPROC (5893)
710DEFPROCFIGHT (7A75)
720PROCPIC(X,Y,S+G) (73E4)
730IFSGN(GY-Y)>0 S=6 ELSE S=4 (ABBC)
740PROCPIC(X,Y,S+G):HASSLE=32+8*(GHOST-16) (74C5
)
750GCOL0,POINT(GX+4,GY):MOVE1271,4:MOVE1271,4*HA
SSLE:PLOT85,1155,4*HASSLE (007B)
760MOVE1155,4:PLOT85,1271,4 (9EA5)
770E=39*LOG(1+(MONEY+KILLED)/10)/3.14 (BB8B)
780IFRND(10)<5 GOTO870 (591A)
790PROCGET:RF=RND(10) (163C)
800IFG$="H" AND(RF<5 OR(ENERGY>HASSLE*4)) Z=2:GO
TO830 (EE41)
810IFG$="B" AND(RF<7 ORENERGY>HASSLE*4) Z=1:GOTO
830 (7C1A)
820IFG$="L" AND(RF<9 ORENERGY>HASSLE*4) Z=.5 ELS
E Z=0 (A2D4)
830DH=-INT(((ENERGY*50*RND(1)-(10*HASSLE) +E)/10
0)*Z):IFDH>0 DH=0 (83FB)
840GCOL0,0:IFHASSLE+DH<=1 DH=1-HASSLE (EE64)
850PROCSWORD (9F2D)
860PROCHASSLE:IF HASSLE=1 PROCVAN:ENDPROC (E4DB)
870RF=RND(10):IFRF<5 DE=0:Z=0:GOTO900 (58F4)
880Z=RF/5 (AF0B)
890DE=-INT((((HASSLE*50*RND(1))-(10*ENERGY)-E)/1
00)*Z):IFDE>0 DE=0 (F580)
900GCOL0,0:PROCENERGY:IFENERGY<1 PROCFALL:ENDPRO
C (605E)
910PROCSWORD2:GOTO790 (17C3)
920DEFPROCGHOST (FA3E)
930IF NOT GFLAG AND RND(1)>.5 ENDPROC (3913)
940IF NOT GFLAG PROCSTART (2E96)
950DXG=16*SGN(X-GX):IFABS(X-GX)<33 DXG=0 (97D7)
960DYG=16*SGN(Y-GY):IFABS(Y-GY)<65 DYG=0 (3499)
970PROCPIC(GX,GY,GHOST) (80A3)
980GX=DXG+GX:GY=DYG+GY:PROCPIC(GX,GY,GHOST) (24D
8)
990IFABS(GX-X)<33 AND ABS(GY-Y)<65 PROCFIGHT (D4
7B)
1000ENDPROC (6BD2)
1010DEFPROCSTART (2D8E)
1020GFLAG=TRUE (AC79)
1030SIDE=63+960*(RND(2)-1):MIDDLE=63+16*(RND(60))
:IFRND(2)=1 GX=SIDE:GY=MIDDLE ELSE GX=MIDDLE:GY=SI
DE (C292)
1040PROCPIC(GX,GY,GHOST):ENDPROC (7760)
1050DEFPROCTAKE (C228)
1060IFR<>COIN ENDPROC (9CDD)
1070IFCOINFLAG ENDPROC (9B21)
1080IFABS(X-XCOIN)>32 OR ABS(Y-YCOIN)>32 ENDPROC
(A69E)
1090COINFLAG=TRUE:PROCMONEY:IFGFLAG HASSLE=0:PROC
VAN (6BC3)
```

CONTINUED OVER

```
 1100GCOL3,13:MOVEXCOIN,YCOIN:PRINTCHR$225 (0B25)
 1110MONEY=MONEY+10:PROCMONEY:REPEAT:SAFE=RND(8+.8
*MONEY)-1:UNTIL SAFE<>COIN (B01D)
 1120FORZ%=0TO1:MOVE1119,976:PRINTCHR$(225-Z%):NEX
T:ENDPROC (EFD3)
 1130DEFPROCMONEY (D346)
 1140GCOL3,3:MOVE1119,816:PRINT;MONEY:ENDPROC (140
F)
 1150DEFPROCDROP (8D72)
 1160IFR<>SAFE ENDPROC (C13A)
 1170IFNOTCOINFLAG ENDPROC (25CF)
 1180IFABS(X-XCOIN)>32 OR ABS(Y-YCOIN)>32 ENDPROC
(31B4)
 1190COINFLAG=FALSE:PROCMONEY:IF GFLAG HASSLE=0:PR
OCVAN (DEAA)
 1200GCOL3,13:MOVEXCOIN,YCOIN:PRINTCHR$224 (58B2)
 1210COIN=RND(8+.8*MONEY)-1 (1A39)
 1220MONEY=MONEY+10:PROCMONEY:GHOST=GHOST+1:FORZ%=
0TO1:MOVE1119,976:PRINTCHR$(224+Z%):NEXT:ENDPROC (
BDBD)
 1230DEFPROCSCORE (DADD)
 1240GCOL0,8:MOVE0,0:DRAW0,1023:DRAW1279,1023:DRAW
1279,0:DRAW0,0:MOVE1023,0:DRAW1023,1023 (CEC8)
 1250FORZ%=256TO992STEP256:MOVE1023,Z%:DRAW1279,Z%
:NEXT (D0A8)
 1260MOVE1151,0:DRAW1151,256 (61BC)
 1270GCOL0,3:MOVE1119,976:IFNOTCOINFLAG PRINTCHR$2
25 ELSE PRINTCHR$224 (871C)
 1280PROCMONEY:PROCPIC(1120,720,GHOST):MOVE1023,56
0:PRINT;KILLED (DFF4)
 1290FORZ%=1TO 4:PROCPIC(960+Z%*64,448,0):NEXT (FD
08)
 1300IFLIVES<4 FORZ%=1TO(4-LIVES):MOVE960+Z%*64,32
0:GCOL0,5:PRINTCHR$226:NEXT (2F7A)
 1310GCOL0,6:MOVE1143,4:MOVE1143,4*ENERGY:PLOT85,1
024,4*ENERGY (082B)
 1320MOVE1024,4:PLOT85,1143,4:ENDPROC (12C9)
 1330DEFPROCBETTER (9D01)
 1340IFENERGY>62 OR RND(4)<4 ENDPROC (42DD)
 1350DE=1:GCOL0,6:PROCENERGY:ENDPROC (044F)
 1360DEFPROCENERGY (6FDB)
 1370MOVE1143,4*ENERGY:MOVE1143,4*ENERGY+4*DE:PLOT
85,1024,4*ENERGY+4*DE (B2E7)
 1380MOVE1024,4*ENERGY:PLOT85,1143,4*ENERGY (97ED)
 1390ENERGY=ENERGY+DE:ENDPROC (2E40)
 1400DEFPROCHASSLE (3A00)
 1410MOVE1271,4*HASSLE:MOVE1271,4*HASSLE+4*DH:PLOT
85,1155,4*HASSLE+4*DH (5B1A)
 1420MOVE1155,4*HASSLE:PLOT85,1271,4*HASSLE (A5EB)
 1430HASSLE=HASSLE+DH:ENDPROC (181E)
 1440DEFPROCGET (1B51)
 1450*FX11,0 (75C2)
 1460PROCHIT:*FX15,0 (301B)
 1470G$="":G$=INKEY$(100):*FX11,1 (75FB)
 1480PROCHIT:ENDPROC (C285)
 1490DEFPROCVAN (8902)
 1500GFLAG=FALSE:PROCPIC(GX,GY,GHOST) (C3B6)
 1510IFHASSLE=1 KILLED=KILLED+1:FORZ%=-1TO0:GCOL3,
3:MOVE1023,560:PRINT;KILLED+Z%:NEXT:FORZ%=1TO3000:
NEXT:CALL BEGIN:ENDPROC ELSE DH=1-HASSLE (6FF5)
 1520GCOL0,0:PROCHASSLE:GCOL0,8:MOVE1024,0:DRAW127
1,0:ENDPROC (3661)
 1530DEFPROCHIT (0D16)
 1540GCOL3,6:MOVE1040,656:PRINT"HIT":MOVE1040,624:
PRINT"NOW":ENDPROC (7BE7)
 1550DEFPROCSWORD (7C24)
 1560IFZ=0 ENDPROC (4ECE)
 1570GCOL4,1:IFGY=Y S%=228 ELSEIFGY>Y S%=227 ELSE
S%=229 (2A2B)
 1580MOVEX-16*SGN(X-GX),Y-16-16*SGN(Y-GY):PRINTCHR
$S%:SOUND0,-(8+ENERGY/10),81,Z (DDEF)
 1590FORZ%=0TO300:NEXT:REPEAT:UNTILADVAL(-5)=15 (3
431)
 1600MOVEX-16*SGN(X-GX),Y-16-16*SGN(Y-GY):PRINTCHR
$S%:GCOL0,0:ENDPROC (E6DC)
 1610DEFPROCSWORD2 (E77B)
 1620IFZ=0 ENDPROC (5CFF)
 1630GCOL4,4:IFGY=Y S%=228 ELSEIFGY>Y S%=229 ELSE
S%=227 (8ED4)
 1640MOVEGX+16*SGN(X-GX),GY-16+16*SGN(Y-GY):PRINTC
HR$S%:SOUND0,-(8+HASSLE/10),200,Z (83D8)
 1650FORZ%=1TO300:NEXT:REPEAT:UNTILADVAL(-5)=15 (7
83C)
 1660MOVEGX+16*SGN(X-GX),GY-16+16*SGN(Y-GY):PRINTC
HR$S%:GCOL0,0:ENDPROC (8295)
 1670*FX11,50 (1A3B)
 1680VDU4:COLOUR7 (CD48)
 1690REPORT:PRINT" AT LINE ";ERL (38C4)
 1700END (BDB2)
```

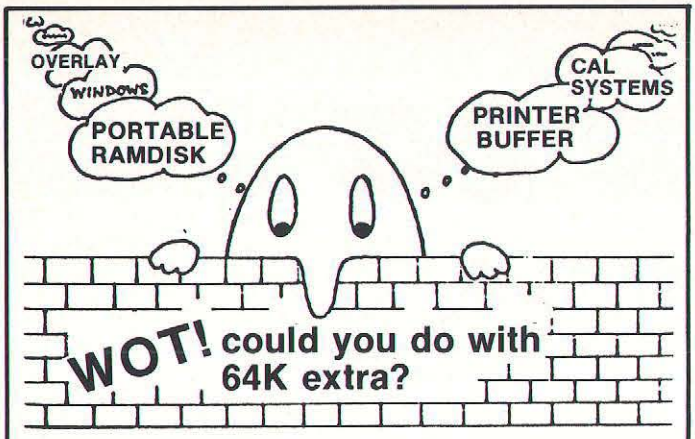
SCRUGE TWO VARIABLES

I%,J%,K%	The current notes used in the machine code routine, TUNE
S%	The current sword character
T%	The noise or silence flag
Z%	Loop variable
COIN	The room where the coin is hidden
COINFLAG	Is TRUE if Scruge is looking for a coin, FALSE if he is looking for a safe.
DX	The change in Scruge's horizontal direction
DY	The change in Scruge's vertical direction
DXG	The change in the ghost's horizontal direction
DYG	The change in the ghost's vertical direction
DH	The change in the ghost7s energy
DE	The change in Scruge's energy
ENERGY	Scruge's energy, decreases during fighting and is replenished when he walks about
E	The experience factor
G	Is odd or even to give the correct feet position for Scruge.
GFLAG	Is TRUE if a ghost is present
GHOST	The sprite number for the current ghost
GX	The ghost's horizontal position
GY	The ghost's vertical position
HASSLE	The ghost's energy during fighting
KILLED	The number of ghosts killed by Scruge
LIVES	The number of lives left
MONEY	There is a score of 10 if a coin is collected, and another 10 if it is put in a safe
MIDDLE	Centre coordinate of ghosts' starting position
NEWS	Sprite number of new sprite
R	Current room
RF	Random factor
S	Sprite number
SAFE	The room the safe is in
SIDE	Side coordinate of ghosts' starting position
W	The colour of palette at Scruge's new position
X	Horizontal coordinate of Scruge's position
XCOIN	The horizontal coordinate of the coin position
Y	Horizontal coordinate of Scruge's position
YCOIN	The vertical coordinate of the coin position
Y1	Vertical offset used to print the scoreboard border
G$	The key pressed during the combat routine

MAIN PROGRAM

20	Accesses error routine if necessary
30	Prints message and waits until SPACE is pressed
40	Selects MODE2, removes the cursor and initialises the program
50	Plays the game until all the money is collected, or all the lives lost
60	Prints a fancy border for the scoreboard
70-80	Prints the score
1670	ERROR routine:- Puts the keyboard auto repeat back to normal

1680-1690 Prints the error message in white



Need more memory on your BBC? — PMS have the answer.
The 64K NON-VOLATILE RAM MODULE
This is no ordinary memory expansion!

* EDUCATIONAL USE - Tough, compact ramdisk, requiring no interfaces or ROMs - use for CAL systems, databases, W/P text storage...
* TRANSPORTATION of programs & data - no compatability problems, works with tape, disk & network BBCs.
* SPECIALIST APPLICATIONS requiring fast data capture or long term data logging.

Module comes with powerful system software, utilities & user guide for **£129.95** + **£2 p&p** (Excluding 15% VAT).
Official Orders accepted. Contact PMS about EDUCATIONAL DISCOUNT scheme.

WRITE OR PHONE FOR MORE INFORMATION

ORDERS AND ENQUIRIES TO:-

PERMANENT MEMORY SYSTEMS

38 MOUNT CAMERON DRIVE, ST. LEONARDS, EAST KILBRIDE G74 2ES

PHONE (03552) 32796

NEW "MICRO-STOCK"

A LOW COST DISC BASED

PROFESSIONAL ACCOUNTING PROGRAM

For The BBC Computer

" Micro-Trader "

INTEGRATED SALES PURCHASE AND NOMINAL LEDGERS

Designed for business use by a business man. " Micro-Trader " is a fully integrated program in which all Sales and Purchase Ledger Transactions are automatically updated to the Nominal Ledger.

" Micro-Trader " offers full Sales and Purchase Ledger facilities including SALES INVOICE and STATEMENT PRINTING with a capacity of 450 accounts and 3000 transactions per month in each Ledger.

Normal Income, Expenditure, Assets, Liabilities & Journal Posting in the Nominal Ledger with full Reporting for individual accounts, Audit Trail, Trial Balance, Profit & Loss and Balance Sheet.

NEW "MICRO-STOCK"

Stock Control program, fully integrated with "Micro-Trader". 4000 Stock Items with user defined Codes. Invoice and Credit Notes and Cash Sales routine. Full Stock Held Reports.

£95.00 + V.A.T.

ONLY £200 + V.A.T.

PHONE TODAY FOR A FREE FACT SHEET

MEADOW COMPUTERS

11, LONDON STREET, WHITCHURCH, HAMPSHIRE, RG28 7LH

Telephone: Whitchurch (025682) 2008

ELECTRON PRODUCTS

The SLOGGER ROMBOX - compatible with PLUS 1 and PLUS 3 ..... **£44.95**
ROMBOX-P - ROMBOX with built-in Printer Interface and FREE Printer ROM (over £20.00 on its own) **£69.95**
Supported by ROM software written specifically for the Electron STARWORD - Wordprocessor, does everything VIEW does plus MUCH MORE (16K ROM) **£34.50**
STARSTORE - Database fully compatible with STARWORD for mailmerging **£29.95**
STARSTORE 2 - Allows up to 65535 very large (18000 chars.) records **£44.95**
ELKMAN - ROM Manager **£17.50**
STARMON - Machine Code Monitor **£22.50**
PDG - Printer Drive Generator for Starword , ...**£9.95**
T2P3 - Tape to Disk ROM **£19.95**
STARGRAPH - Graphics ROM **£21.95**
PRINTER ROM - Buffer & Utility **£24.95**

BBC PRODUCTS

STARWORD - Sophisticated wordprocessor **£34.50**
STARSTORE - Database **£29.95**
STARSTORE - 2 Very powerful database **£44.95**
BEEBMAN - The ultimate Rom Manager **£17.50**
STARMON - Impressive Machine Code Monitor **£24.95**
PDG - Printer Drive Generator for STARWORD ...**£9.95**
TREK - ROM Disassembler with a difference ... **£22.95**
STARGRAPH - Colourful graphics ROM **£21.95**
PRINTER ROM - For all printing requirements .. **£24.95**
DOUBLE DENSITY CP/M for "OPUS DDOS" giving 784K disk space 256 files, 25% speed increase & utilities**£49.50**
DISK EDITOR for "OPUS DDOS" the only single AND double density Sector Editor. Restores bad Sectors**£9.95**

ALSO AVAILABLE

PAGE E00 DDOS & Comprehensive OPUS DDOS USERS BOOK - ALL prices inlcude VAT, P&P (UK)

Contact:

SLOGGER LTD.

107 Richmond Road, Gillingham, Kent ME7 1LX. Tel: (0634) 52303 - 2 Lines

Down to Business

Jon Vogler

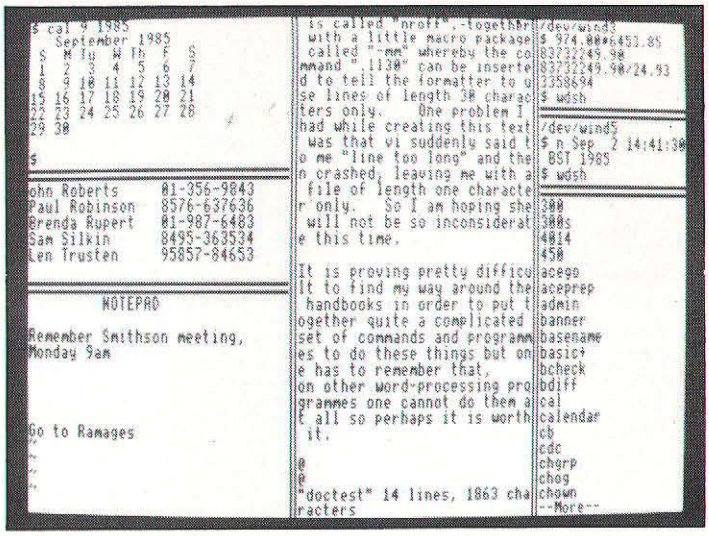
INTRODUCTION

UNIX is the most powerful operating system ever to run on a micro. *Torch* offer it on the ultimate BBC add-on — but not for the faint-hearted.

"I shan't get a car; it's far too difficult to learn to drive." Ever heard that from a businessperson? Under the age of sixty I mean? Perhaps from the bicycle wobblers you pass on your way to work. For most people however, the first few nervous lessons behind the driving wheel, the complexities of gears and clutch, handbrake and hill starts and the tedium of the highway code are worth learning because of the increased speed and capacity and power that a car provides. Trouble is, even cars are not all that fast, especially in traffic jams and by the end of a long trip you are fit for nothing.

You can go by train, relax, stop thinking too hard. There is just as much speed, capacity and power but something else is lost: flexibility. The rails are fixed and someone else plans the timetable without considering your particular needs. All too often, this means you cannot get to exactly where you wish at the time you want, even if you get up at four o' clock in the morning. What is the answer then? Some tycoons are so impressed at the improvements which technology can make to their personal convenience and efficiency that they buy and learn to pilot a personal airplane or helicopter. These people arrive for meetings ontime, fresh, confident. We are no longer surprised that business people should invest thousands of pounds and hundreds of man hours learning time to achieve one ultimate goal: high personal productivity. Why then do these same sultans of business say "I shan't get a *UNIX* system; it's far too difficult to learn"?

Of course many business folk are still at the bicycle stage: their offices use typewriters, filing cabinets, even slide rules or adding machines and those telephones that wear a groove round your index finger. Many who decide they needed the power and flexibility of the micro computer (the office equivalent to the motor car), use *CP/M 80* or *MS-DOS* operating systems and a generation of business programs such as *Wordstar, Visicalc, Dbase II* and *Perfect Software I.*



1. The Unicorn's window facility: seven live tasks all running concurrently and more can run in the background.

They are powerful and flexible but have one problem, you must learn to drive them just as you had to learn to drive a car. Those who persevere find it rewarding, but many funk the effort. They want more speed and power but are unwilling to do the learning and this has provoked the birth of very user friendly operating systems such as *Gem* or *Lisa* which soak up spare memory to create pictures and a clickety mouse to drag them around the screen. Like the train, this approach has one huge disadvantage: although you still get the power and speed and can run everyone's favourite, business program (you know, the ones with trendy names like *Symphony, Framework* or *Jazz*), what is missing is flexibility. If what you want to do is that little bit different; if your business doesn't quite fit the pattern or if you are more adventurous and want to go where others have laid no rails, you find yourself at a dead end.

The other alternative is *UNIX*, which is the equivalent of the personal helicopter. *UNIX* is not a program, or a computer; it is an "operating system", just like *CP/M* or *Gem* or the BBC's own. The difference is that *UNIX* is the most powerful one ever to run on a micro. Not only will it do anything that you wish and go anywhere that you like but taking your employees or business partners along for the ride costs little extra. There is almost total flexibility and huge amounts of power and capacity but both the helicopter's disadvantages: it is expensive and learning to use it needs confidence and a lot of time.

To see whether it was worth the money, and whether its difficulties and complexity had been exaggerated, I wanted to try it myself. Impossible for the owner of a modest 32K *Beeb*? Not at all, because *Torch* offer the ultimate in BBC add-ons: the *Unicorn* which hangs a *Motorola 68000* 32-bit chip (don't worry about the numbers that just means "very powerful indeed!") and a megabyte of memory (that just means 30 times as much as the normal *Beeb*!) and a 20 mega byte hard disc unit (40 times as much as an average floppy disc drive). The advertisements were most persuasive: a system that would run half a dozen different operations at the same time; would display them in separate windows on the screen; could be networked to ten other people who need each have no more than a *BBC B* with an *Econet* fitted and a monitor. All this at around the price of a (20 megabyte) hard disc *ACT Apricot XI* with half as much random memory and no real multitasking capability.

My especial interest was that *UNIX* seemed to solve so many of my current problems. How to look up someone's telephone number on my database when in the middle of word processing this article? How to extract information from the database and put it directly into a report without tedious and elaborate programming beforehand? How to look up my floppy disc catalogue or my bibliography or the calendar for 1988 while compiling a budget spreadsheet? How to insert a couple of tables of data in the text of a technical report? Fantastic if these can be achieved but what about the penalty? what about that notorious user unfriendliness? *UNIX* is huge — the Torch version comes with over 1000 different files and programs (not bad value, around £4 each, inclusive of hardware) and the hand books are about the size of the London telephone directory. So readers will forgive me if this article is split. This month I shall describe the standard *UNIX* system and next month discuss using it in business and running some applications packages on it.

SOME ENCOURAGEMENT

In case it proves difficult to describe this highly technical system , let us start with some encouragement. Fig 1 shows what multi-

tasking windows can really offer a business person. The central window contains the text of an article for *A&B*.

To the left, starting from the top, this month's calendar, so I do not miss your editor's merciless deadline. As well as this year it seems to cope very amply with the year 2010. Then a window on the database giving one or two phone numbers (I needed them in a hurry for someone who just telephoned). At the bottom, my "jotting pad" notes made during the phone call, for later reference.

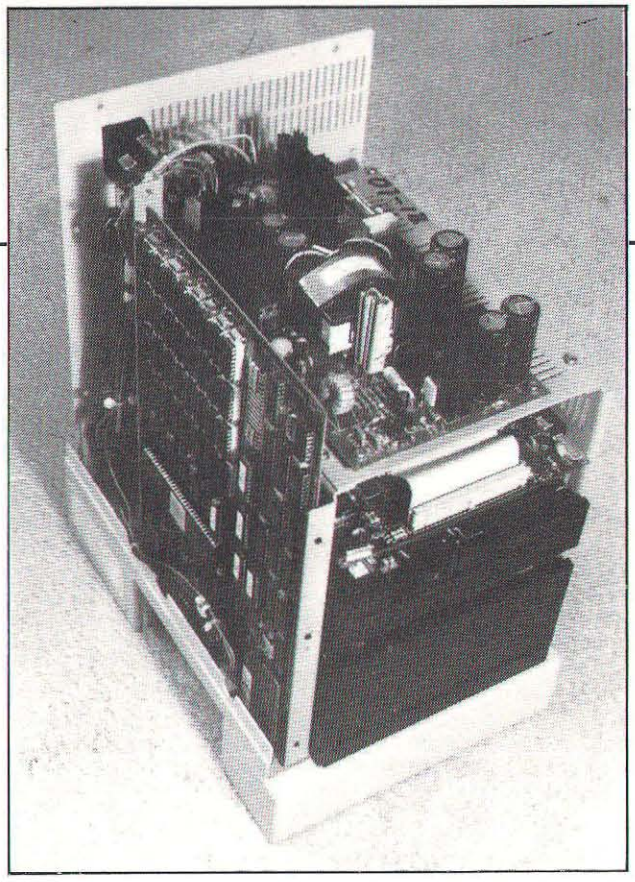
On the right of the screen: at the top the calculator, on which I could work out my income tax or how long it will take the probe to reach Haley's Comet. The calculator program seems quite comfortable working to twenty six decimal places!

Below is today's date and time and, at the bottom a list of disc files, so I can check the name of a file I wish to insert in the article. These are not just dead information on the screen: each is a live program or process currently running.

As well as the seven windows I could have other tasks grinding away in the background: for example calculating a complicated formula that needs to reiterate many times or formatting a document for printing or counting the number of words in an article. You don't have to stop and wait until these processes have finished. Your printer can be printing away in the background and another program could be transmitting data over the telephone or interrogating a data base in California. Think how long it would take, on a normal BBC, to gather all this information and perform all these tasks. That's the high personal productivity that *UNIX* offers.

SOME FEATURES OF UNIX

I shall not try and describe how *UNIX* works. That needs a whole book and some people get quite nutty about it. *UNIX* is full of expressions like the "shell" and the "kernel" which we don't need to go into. Rather let's describe the hardware you get if you buy *Torch UNIX*. You need the 20 megabyte fixed disc (reviewed in *A&B Computing July 1985*); *UNIX* will gobble between half and two thirds of it. That still leaves plenty of capacity for your *CPN* files and programs (CPN is the CP/M look-a-like operating system under which the Torch Z80 runs Wordstar, dBase II, Perfect Writer or whatever); you don't lose access to these because you have *UNIX*.



2. Inside the cabinet of the Torch Unicorn.

Contained inside the hard disc cabinet is the Atlas printed circuit board which contains both the 68000 chip and also the Z80 chip (Fig 2). Also within the same box is the huge random access memory: you have the options of half a megabyte (sufficient for a single user) or a full megabyte (Fig 3), necessary to run an efficient multi-user system. One of the reasons why *UNIX* can run so many processes simultaneously is that all this vast brain is used for the actual processes that are running. *UNIX* loads them from the disc and dumps them again when they are finished so you don't have memory cluttered up with processes which are not actually running. Finally you need a *UNIX* ROM to be fitted in your *BEEB* and, if you want a multi user system, an *ECONET* network. The various costs are shown in the inset box Fig 4.

It is the terminals which are so remarkably economical. They don't even need a second processor: a straight *Beeb* fitted with Econet and a special ROM is quite sufficient. Once *UNIX* is running on the host machine, any terminal can link in quickly and easily. Nor incidentally do you lose access to those beloved BBC games programs. Keys *B* and *Break* will still get you back to the BBC operating system. However, although you can multitask (run several tasks simultaneously) from the terminal, under the Torch system the terminal will not work with separate windows; this facility is only available on the host machine. However, this might even be a good thing, you don't want your secretary writing out his (or her!) weekend shopping list while typing your monthly production report nor the warehouse clerk making changes in the sales ledger. In fact there are no problems of this kind, because *UNIX* offers a sophisticated protection system. Each file can have three different sorts of protection, read, write or execute for three different groups of people: the owner, a defined group to which the owner belongs or anybody else (Fig 5).

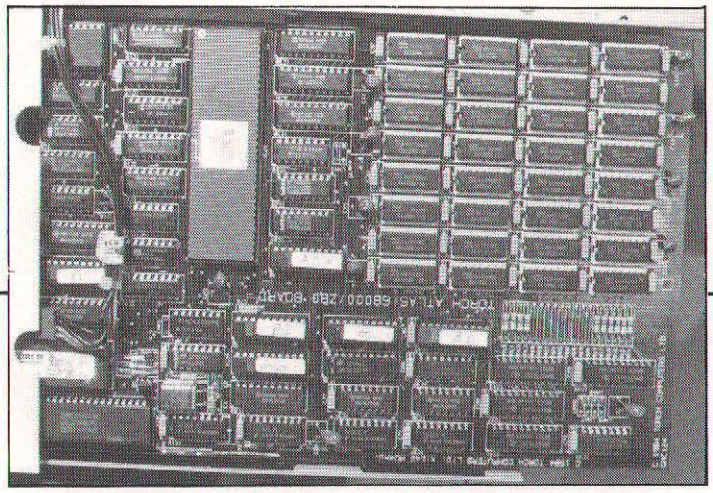
UNIX AND THE USER

UNIX is most often criticized for being unfriendly. People miss the neat displays and helpful error messages so familiar from the BBC. Partly this is because *UNIX*, from within the system, does many of the things for which, with just a BBC, you would need to buy a special program. There is nothing that prevents extravagant displays (except prudent economies of disc and memory space, sometimes ignored in the face of such abundance!) However the lack of error messages is definite. *UNIX* is very terse. If it can't find a file, it doesn't say *"FILE NOT FOUND"* ,it just returns to the prompt – rather like those travelling companions abroad who, as you pass some majestic scene or fascinating old building merely draw on their pipes and grunt!

The unfriendly feeling is emphasized because *UNIX* does every thing in lower case letters and uses extremely concise commands. However this too has merit: it saves one finger constantly hovering over the shift key and reduces the number of key strokes. It is only slightly quicker to run a game by typing in small letters "aliens" than by the familiar BBC: *CHAIN "B:ALIENS"* but when such commands are repeated hundreds of times the difference becomes significant. In fact many of the commands are helpfully mnemonic. If you want to count the number of words in a file called "fred", you simply type in *wc fred*. Likewise the massive handbooks actually look less friendly than they are. They are looseleaf, extremely well subdivided, comprising both separate sheets on each command plus explanatory articles (mostly written by the men who invented *UNIX* at the Bell laboratories in America in the late seventies) and although some are highly technical, there are others designed for the nontechnical and the new user.

Just like using a motor car service handbook, you don't need to understand all the technicalities of the engine to find the bit that tells you how to top up the battery. But

CONTINUED OVER



3. The Unicorn's main board: the thirty two memory chips (top left) give a megabyte of RAM; the huge Motorola 68000 chip (lower centre left) has 64 pins to handle 32 bit words.

if you are a motor mechanic, then you want all the engine detail you can get. However many users will find, as I did, it is helpful to buy one of the many "Teach yourself" books. *Torch* list eight of them and my local bookseller offered a further five!

THE UNIX FILE SYSTEM

Another feature that many people find difficult at first is the concept of "path names". This is also used (but is less essential) by computers (such as the *IBM PC*) that use MS-DOS. Instead of dividing the hard disc into a number of surfaces (B:, C:, etc just like floppy discs) *UNIX* treats the whole huge area as one. That brings the immediate, enormous advantage that you don't have to remember (or care) on which surface your program or file is located. Hundreds of files are all equally accessible. But, as always, there is a penalty to be paid. How do you locate the exact one of a thousand files you need? Imagine you have just arrived in Britain and want to find someone called *Jon Vogler*. You could go to some central government office and look up all the records of all the people who have ever been born in Britain and you might, with difficulty, track him down. Much easier however if you have been given an address. In fact we address letters the wrong way up: to find *Jon Vogler*, the post-office starts at the bottom and progresses upwards. The first information, that he lives in *West Yorkshire*, narrows the search; *Leeds*, as the town in *West Yorkshire*, narrows it further and so on. In fact we could write his address more logically as **West Yorkshire/ Leeds/ Roundhay/ The Avenue/ 40/ Vogler/ Jon.** If there was another Jon: *Jon Smith*, living at *18, The Beeches, Ruislip*, he would be clearly distinguished as **Middlesex/ Ruislip/ The Beeches/ 18/ Smith/ Jon.**

Unix path names work exactly the same way. The place you start from (*Britain* in our example) is known as *root* — like the root of a tree. Then you follow a path name through various branches to get to the file you need. Each of the files can be regarded as a fruit hanging on the tree and various limbs, branches and twigs, from which the fruit hangs, are called "directories". One directory may have several files hanging from it or may have some files and also some subdirectories. A typically *UNIX* path name is **/usr/bin/wc.** The first slash indicates the root. *usr* is like a branch and *bin* a twig (they stand for user and binary and are directories that are present in the *UNIX* structure when you buy it. *wc* is the fruit hanging at the end of the twig, the wordcount program. You can also create your own directories and files.

Supposing in your business the *UNIX* system is going to be used by Tom and Betty. You might create a new directory, called *new*, which would contain all the files added since *UNIX* was received from the supplier. Then in *new* you might create three more directories: one called *common* that anybody could use, one called *tom* and one called *betty* for their sole use. Now suppose both Tom and Betty create a file called *data*, how does the computer know which one to use? Why, because one has the pathname */new/tom/data* and the other has the pathname */new/betty/data*. That is all there is to a pathname, it is simply a file with a rather complete address which enables the computer or its user to find any program or file without difficulty. And of course Betty could lock her data file in such a way that perhaps either she or Tom could both read it but only she could write to it to make alterations.

Every *UNIX* system has also one "super user" who can read or alter every file. And of course *UNIX* offers the normal range of commands to move files from one directory to another, copy them, rename them and alter them in various ways and has a powerful series of "wildcards" so that, for example, specifying the pathname */new/betty/** can refer to every file in the directory */new/betty* if you wish to list or protect them all.

FIG 4 - TORCH UNICORN PRICE LIST		
Item	Memory	Price (excl VAT)
Unicorn	1 Mb RAM	3995.00
Unicorn	Half Mb RAM	3495.00
Multi-user licence	Host ROM	850.00
Terminal pack	Terminal ROM	99.00

4. Prices of the Unicorn

PROCESSING PROGRAMS AND FILES

One reason why *UNIX* is so flexible is that you can do so many things with programs and files. I have already mentioned running them in the background while you do other tasks on the screen. You can also *redirect* them. Suppose you have a file containing a list of names of employees, arranged in order of the date they joined the company and you want to sort it into alphabetical order. The *input* is the list of names, the *process* is to sort but what about the *output*: do you just want it to appear on the screen? Not much you can do with it there. *UNIX* gives you complete freedom to redirect it as you wish: either to the screen or to a printer or to another file or down the telephone wires to a computer in your head office a hundred miles away.

Have you ever tried to make an alphabetic listing of all your BBC disc files? If so you will know how frustrating it is . You can display a catalogue of one disc on the screen and you can print it but you cannot merge it with the contents of twenty other discs or sort them alphabetically without buying a special program (a program which incidentally I have yet to see produced; anyone got one?) *UNIX* makes this easy, without any special program. I was able to type in one single line: **find / -print>/user/jon/filelist** This listed all the 1250 files and directories on the system by name and wrote them into a file (called *filelist* in my personal directory */user/jon*. Alternatively I could have routed them to the printer or put a *tee* (*UNIX* is full of plumbing words) in the system so they went both to the printer and to the file at the same time.

However, suppose I did not want to list them but only to count them. *UNIX* will *pipe* processes: the output from one process flows into another. The command to pipe is the simple vertical bar "|" and I could have piped the output from the file listing described

```
$ ls -l /usr/binlcat|more
total 8615
-rwxr-xr-x  1 bin      other       7108 Aug 22 16:30 300
-rwxr-xr-x  1 bin      other       7224 Aug 22 16:30 300s
-rwxr-xr-x  1 bin      other      14896 Aug 22 16:30 4014
-rwxr-xr-x  1 bin      other       6540 Aug 22 16:30 450
-rwxr-x--x  1 bin      sys       110088 Aug 22 17:33 acego
-rwxr-x--x  1 bin      sys        58747 Aug 22 17:33 aceprep
-rwxr-xr-x  1 bin      other      35084 Aug 22 16:03 admin
-rwxr-xr-x  1 bin      other      10280 Aug 22 15:11 banner
-rwxr-xr-x  1 bin      other         63 Aug 22 15:11 basename
-rwxr-xr-x  1 bin      other      96656 Aug 22 16:46 basic+
-rwxr-x--x  1 bin      sys        45075 Aug 22 17:37 bcheck
-rwxr-xr-x  1 bin      other      15688 Aug 22 16:03 bdiff
-rwxr-xr-x  1 bin      other      10248 Aug 22 15:11 cal
-rwxr-xr-x  1 bin      other        307 Aug 22 15:11 calendar
-rwxr-xr-x  1 bin      other      14828 Aug 22 16:04 cb
-rwxr-xr-x  3 bin      other      34232 Aug 22 16:04 cdc
-rwxr-xr-x  1 bin      other      12696 Aug 22 15:11 chgrp
-rwxr-xr-x  1 bin      other      12480 Aug 22 15:11 chog
-rwxr-xr-x  1 bin      other      12348 Aug 22 15:12 chown
-rwxr-xr-x  1 bin      other      12476 Aug 22 15:12 chsh
-rwxr-xr-x  1 bin      other       9708 Aug 22 15:12 clri
-rwxr-xr-x  1 bin      other      12472 Aug 22 16:31 col
-rwxr-xr-x  1 bin      other      25248 Aug 22 15:12 comb
-rwxr-xr-x  1 bin      other      10844 Aug 22 15:13 comm
-rwxr-xr-x  1 bin      other      19088 Aug 22 16:04 cref
-rwxr-xr-x  1 bin      other      11892 Aug 22 15:13 crypt
-rwxr-xr-x  1 bin      other      16612 Aug 22 15:13 csplit
-rwxr-xr-x  1 root     other      22188 Aug 22 15:51 ct
-rwxr-xr-x  1 bin      other      17832 Aug 22 16:05 ctags
--More--
```

5. File list. The group on the left indicates protection. These files can be read, written or executed by the "owner", executed or read, but not written, by any member of her "group" but only read by anybody else.

above into the wordcount program just by writing: **find / -print/wc.** And if I wanted to sort them into alphabetical order and present them on the VDU, one screenful at a time without "scrolling", I could pipe them through two further processes like this: **find / -print/sort/more.** Now supposing that I had some operation of this type which I wanted to do 10 times a day for a month. Rather than type that rather tedious line in every time, probably getting it wrong now and then, I could turn it into a *shell script.* All this involves is saving it in a file and giving the file a name; in this case let us call it *filecount.* By typing **sh filecount ‹CR›** the *UNIX* shell (which we haven't explained but it is obviously determined to be mentioned!) would run the sequence of commands of its own accord. Suppose however, I only wanted to run the process on working days, not at the weekends. *UNIX* keeps the date permanently recorded (provided you set it correctly when you get started) and, within the shell script system, is a programming language that uses such useful tools (familiar to all who write programs in BASIC) as *if – then loops, while, for* and *until* commands and the rest. So I could easily include in the shell script a clause that says: Only run *filecount* if the day of the week is Monday – Friday. Imagine the length of BASIC program needed to handle that lot. *UNIX* does it in a couple of terse, economical lines but there are no helpful menus; the user, like any helicopter pilot, has to remember precisely which commands to issue to the machine. That is why, to many, it appears unfriendly.

THE C PROGRAMMING LANGUAGE

UNIX is not simply the most powerful operating system available on any micro today. It also includes *C*: one of the most economical and powerful programming languages available. Most modern business applications programs are written in *C* but *C* is also a powerful "number crunching" language. Although *C* can be run on *CP/M* or *MS-DOS* or various other operating systems (but not on the BBC'S operating system) it was invented by the same team who developed *UNIX* and grew up hand-in-hand with it. Most of *UNIX* is written in the *C* language and *C* was developed to operate within the *UNIX* operating system.

C is not a complicated language to learn, particularly if you already know *BASIC* and understand such concepts as loops (*for ...next*) and functions. One of the best books – *Understanding C* by Bruce Hunter, published by *Sybex*, runs to about 300 pages and can be mastered if you spend 2 hours a night on it for a week or two. The reference manual to the *C* language, included within the *UNIX* hand books, extends to only 30 pages and the list of key words is very short.

C programs can be very concise: *C* does in one line what may take a dozen lines of code in *BASIC*. For example a single line of program such as: **while (nbr-read = read (fdi, buffer, BUFSECTS))** is a *while loop* which does several tasks. In the centre is a *read function* that reads an entire file in one pass and puts its contents into a *memory buffer*. Surrounding it is a "dummy variable" called *nbr«read*, waiting to read the message sent by the above read function: a message which might be either "here is the end of the file" or "an error has been encountered". Finally, on the outside, is the *while loop* itself, which keeps the *read function* going and makes it stop when it reaches the end of the file.

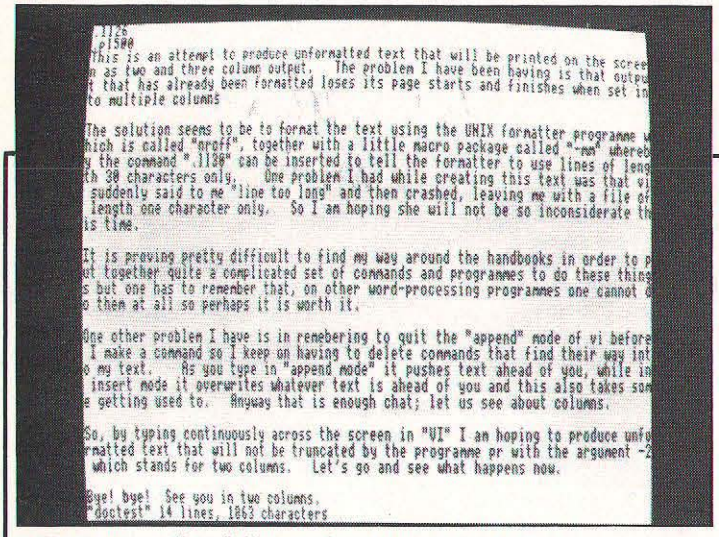
C is a compiled language: the program or *code* that you write, in words such as those shown above, has to be fed through a program called a *compiler* which, stage by stage converts it into *machine code* (the binary language which the computer's central processor actually understands). Only when this has been done for the whole program can it be run. (*BASIC*, in contrast, is an *interpreted language*: it is converted to machine code one line at a time). The advantage is that *C* runs very much more rapidly and uses very little

```
MEMO

To:........ All High-Powered Businesspeople

From:...... Computer Consultant

Subject:... UNIX


Torch's UNIX package offers (at a most competitive
price, particularly if you require several
terminals) a multi-user, multi-tasking, system of
enormous power and flexibility, with windowing on
the main computer. If you find your present system
limited, and if you have the time and confidence to
learn a demanding but extremely rewarding skill,
have a close look at UNIX but checkout the software
(or read Down to Business next month) before
buying.

If you find using computers in your business
difficult, are short of time for learning or are
satisfied to let others decide what your business
needs, keep well away from UNIX.
```

```
INPUT TEXT TO PRODUCE FIG 4 USING tbl

.TS
center doublebox tab (@);
c s s
c | c | c
l | l | n.
FIG 4 - TORCH UNICORN PRICE LIST
_
Item@Memory@Price (excl VAT)
_
Unicorn@1 Mb RAM@3995.00
Unicorn@Half Mb RAM@3495.00
Multi-user licence@Host ROM@850.00
Terminal pack@Terminal ROM@99.00
.TE
```

6. Table pre-processing program "tbl" produced the table in Fig 4 from this input.

CONTINUED OVER



7. The screen editor "vi" uses the entire screen for text.

memory. List it on the screen and the compiled program appears as a confused jumble of codes, so it is far harder to pirate. The disadvantage is that it is tedious to correct the code because, for every alteration, you have to recompile it before testing whether it runs correctly. That, of course, is where the *Torch Unicorn's* multitasking windows facility really comes into its own: you can write the code in one window, compile it in the next, and run it in a third, as near simultaneously as you, the user, can manage. For any business that cannot find, or cannot afford, tailor-made software, the ability to write in *C* is an enormous asset: but only for those with the time and ability to exploit it.

TEXT PROCESSING FACILITIES

Because *UNIX* was originally written to help the engineers at Bell Telephone Labs produce their technical reports, it is particularly strong on text processing facilities. However, these are not like the accustomed wordprocessing packages, such as *Wordstar* or *View*, with all the facilities rolled up behind one single menu.

Instead there are many different components: a variety of editors, two different formatters and several subsidiary programs with special uses. Some of the editors deal with only one line at a time and are virtually obsolescent in these days when we have all become used to full-screen editors. However *UNIX* offers an excellent one called *vi*. To format for a line printer there is a program called *nroff* (pronounced "enroff") and there is a program for formatting photo-typesetting call *troff* ("teeroff"). Any reader who has discovered how difficult it is to get your friendly neighbourhood printer to accept text produced on less powerful word processors should be especially interested in the latter. Both *nroff* and *troff* are difficult programs to use so *UNIX* provides a set of "macros" which tailor them for the average user. In addition there are "pre-processors" for tables of data and for mathematical formulae. Again, anyone who has experienced the problems of incorporating these simply and fluently in word process text will heartily rejoice. The table program is elegant: Fig 6 shows the text I typed in to produce the pricelist table, Fig 4 .

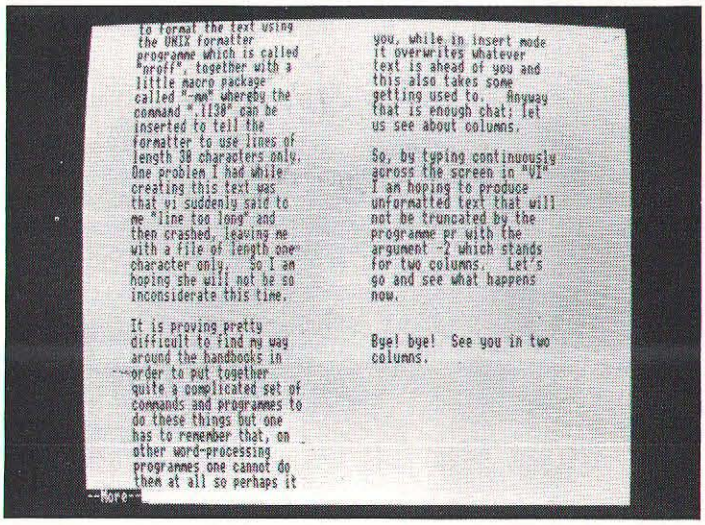
vi

vi has everything you could ask for from a screen editor except one: familiarity. It seems to work quite differently from most programs and it is tempting for the new user to reject it because of this. Typing in is done in a quite separate mode from other activities, although only a single keystroke is needed to switch between the modes. For example if you are in *text mode* and enter the simple command to save your text *":w"* without first touching the escape key to go into *command mode*, then you find you have written ":w" into the middle of your text and, only after some days of practice is it trouble-free to remove it again!

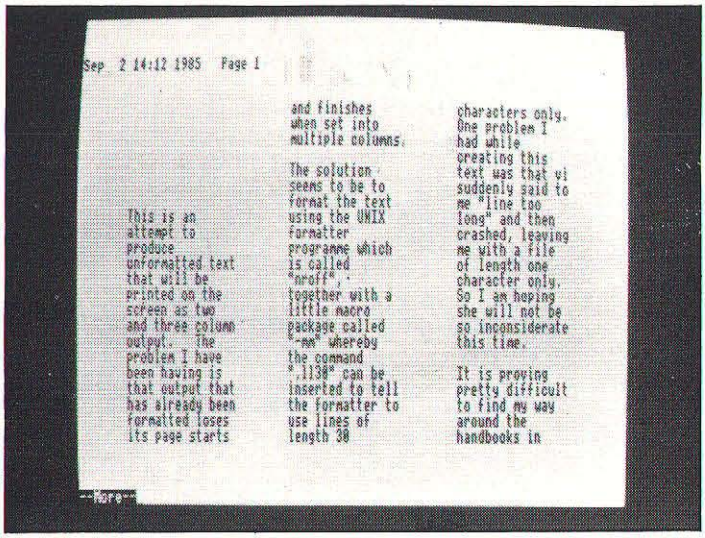
However, once these teething troubles are past, it is a delight. First of all there is a complete range of cursor movements: forwards or backwards by the next letter, word, line, sentence, paragraph, half-screen, screen, section (a section is a portion of text begun with a subheading) and the whole file. You can specify whether your move is to the start or end of the word and can include punctuation or skip over "white space": the net result is that one keystroke serves where other word processors need two or even several and this, as readers will now know, is Vogler's acid test of good editors. One can also move to a specified letter or word or line number. (Lines can be numbered automatically without interfering with the editing). In addition, by prefixing the command with a number, you can move by 5 words, 3 lines, 7 screens etc. There is plenty of screen to move around, because, in contrast to many word processors, (particularly those designed for the IBM PC!) *vi* uses the whole screen depth of 32 lines and the whole width (Fig 7). There is no "wordwrap" but don't worry: *nroff* supplies the necessary justification, filling and hyphenation if required. By prefixing with a "d" any of the movement commands you can delete the text across which the cursor moves. If you want to change a word rather than simply delete it, prefix the movement command with small "c" and you find yourself typing over the unwanted words or sections, marked at the end with a dollar sign to indicate the extent of the deletion, which is completed when you press *escape*, even if the text you are adding is much shorter than that you are taking out.

The arrangements for copying and moving text are particularly powerful because, unlike so many programs which offer only one or at best two or three markers, *vi* gives you 26 markers: an "m" followed by any small letter of the alphabet. To home on a marker you can either prefix the its letter with a "£" sign; or with a single inverted comma sign "'" to return to the start of the line on which it is contained.

To complement the 26 markers there are 27 buffers: one buffer into which any deletion is saved (without you needing to do anything) and the others which need to be specified by a letter of the alphabet. The joy of the unnamed buffer is that, if you carry out deletions and then regret them, the command small "u" restores you to the condition before you started deleting. Of course if you forget what text is in which buffer you would simply open a small window and list them all to jog your memory. Having deleted or copied text



8. The text from Fig 7 converted to two columns by "nroff", the formatter.



9. The same text in three columns.

into a buffer, you can replace it wherever the cursor is and there is also a facility to replace it several times: very useful when writing legal or contractual documents, all of which begin "the contractor shall....."

Most word processors have a facility that enables you to centre the work area (eg the cursor) in the centre of the screen; *vi* goes one better (or two actually) and allows you to postion it either at the centre or at the top or bottom of the screen. Searching and replacing are also easy, with the facility of doing them in reverse and for continuing the search after the first find with a single key stroke. Searches can be instructed to ignore the case (ie treat "happy", "Happy", "HAPPY" as the same). One very useful facility enables you to match braces, (" "") " []). When your cursor is on the opening bracket, typing the command "%" will move the cursor to the matching bracket, particularly useful when writing programs.

Working with multiple files also has undreamed of power. If you want to insert part of one text into another, you just display line numbers and specify the range of line numbers to include. Likewise you can save part of a text or edit only part. If you are really chopping and changing texts about, you could run them in two or three windows with all the line numbers showing in order to carry out a rapid mosaic operation.

vi makes it easy to set out text on the screen: a double sideways arrow will move the complete line of text in either direction and there is a setting command to change the "shift width" distance by which it moves and also to auto indent the following lines if required.

However to me the greatest of all joys with *vi* was that, while still remaining in the word processor, one can still do those useful things which are normally denied. You can set the function keys and this enables you to make full use of them for writing repetitive phrases. However, *vi* has an additional feature, of mapping not only the red function keys, but any key on the keyboard, to contain any phrase or command that you may wish. This gives terrific economy with commonly used phrases or commands. Finally while still leaving the text on the screen you can leap out of word processing, perform a calculation, check on a filename, interrogate a data base, or do anything else you like and then leap back into your word-processing text without losing so much as a character.

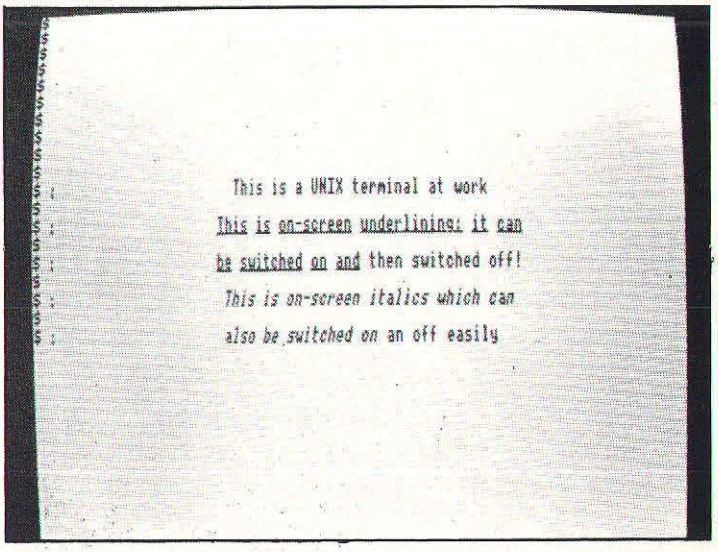
If *vi* is powerful, *nroff* and *troff*, when combined with their various macros, are even more so. They can be summarised in one phrase: anything you can do with words on paper, the *roffs* can do it. The facilities I particularly valued are of adjusting the environments for headings. You can set your own combinations of underlining, bold type, number of lines before or after, indentation etc., and the facility to print text in as many columns as you wish (Figs 8 and 9). It is simple to view the text on the screen before printing: there is none of that nightmare of trying to preview text on the screen, one hand firmly fixed on the shift and control keys, and then being unable to go backwards to see what it said on the previous page. Moreover the underlining and italics can also be viewed on the screen (Fig 10).

One serious deficiency with the text processing facilities is the lack of an adequate manual index. The main part of the *UNIX* hand book, the part that explains the various *UNIX* commands, has an excellent index but there is none for *vi* and, although the hand book sections are detailed and fairly easy to understand, it is very difficult to find the particular activity you want to perform. There is however a useful detachable prompt card which covers the common commands and manoeuvres.

CONCLUSION

I have only glimpsed the full depth of capabilities of the text processing packages and indeed of the rest of *UNIX*. I anticipate it will take years to really get their full power at my fingertips; but what exciting years!. In next month's article I shall discuss some of the applications programs you can run on *UNIX*. Many have been designed to take the complexity and anxiety out of using this operating system.

It took me four or five days of intermittent use to learn as much about *UNIX* as is revealed in this article, and after the first day, I cannot say I found it too terrifying. As with all computer applications, learning your way around the hand book makes life much easier. If you find computers difficult, and have taken a long time to master the programs you are using at present, *UNIX* is not for you. However if you feel at home with present packages but are frustrated by their lack of power and limited capabilities, then you can buy *UNIX* with no hesitation. Finally if your personal productivity is of vital importance and you are prepared to devote money and the necessary time to mastering it (and remember that there are many books, tutorials and courses, as well as *Torch's* extremely helpful customer support unit to help in this) then you should have no hesitation. Like learning to fly a helicopter, it will mean that your ability to do what you want, quickly, efficiently, where you want and how you want, is revolutionized.



10. Underlining and italics can be shown on the screen.

Basic 3 For BBC Computer With Sideways RAM

Until now, only the lucky 32016 co-processor owners could afford to have 10,000 strings or variables at once in their programs. But now with the Solidisk Basic 3, you too, can play with similar facilities.

The Acorn's BAS128, as implemented at present on the 128K BBC PLUS, occupies the main user memory and uses 4 Sideways RAM banks to store both the program and the variables.

The Solidisk BASIC 3 is an overlay for the BBC BASIC 2, occupies 1 Sideways RAM bank, uses other Sideways RAM banks to store variables and leaves the program in the usual main memory.

The Solidisk version has some significant advantages, mainly:

1) The Solidisk BASIC 3 offers more space:

	BBC B	+SWR32	+SWR64	+SWR128	+SWR256
Size for BASIC:	25.5K	37.5K	69.5K	133.5K	261.5K
128K BBC PLUS					

2) The Solidisk BASIC 3 is faster than BAS128, for example in running PCW Basic Benchmark (PCW January 85):

COMPUTER	TIME IN SECONDS	RANK
BBC B + 6502 SECOND PROCESSOR:	9.58	1
BBC B + BASIC 2	14.27	2
BBC B + BASIC 3	18.99	3
ELECTRON	20.44	4
BBC 128K PLUS + BAS128	29.43	5

3) The Solidisk version loads, saves and runs programs in the usual main memory thus requiring no change to the programs.

4) The Solidisk version is available on disk and in ROM (as part of STL TOOLKIT ROM), BAS128 is available only on disc.

Solidisk Sideways RAM is compatible with all issues of BBC computers along with most current add-ons. To date, more than 45,000 Sideways RAMs have been sold.

Sideways RAM is similar to Sideways ROMs such as the BASIC ROM or the DFS ROM but it can be written into. This means that Sideways RAM can run all types of software that are normally available in ROMs.

Solidisk also has a large library of nearly 100 programs to maximise the use of Sideways RAM.

These include important software such as word-processor, database, spreadsheet, spelling, checker, macro Basic, Extended Basic, Virtual Memory, Index, Sprites, RAM Disc, Machine Code monitor, Digitised pictures, SFX etc... and entertaining computer games (Blitz, Meanies, Escape, Seawolf etc).

THE NEW STYLE SWR64 AND 128K:

Solidisk have introduced a new style for the SWR64 and the SWR128.

The new RAM card has the capacity of 64k or 128k, upgradable to 256k and TWO intelligent ROM sockets accepting up to 64k of ROMs.

Each 32k EPROM (or 27256 EPROM) is treated as TWO separate 16k EPROMs and occupies 2 adjacent Sideways ROM banks.

You can use for example STL TOOLKIT + STL WORD PROCESSOR in one, DFS 2.1 and ADFS 2.1 in the other.

If you want to blow 32k EPROMs yourself, you will need the UVIPROM32, a new version of the UVIPROM Eprom Programmer capable of programming the AMD 27256 EPROMs (Programmer: £20.00, Eproms £10.00 each).

SOFTWARE PACK 1
(included with SWR)

The Software package comprises 5 diskettes, formatted either in 40 or 80 track. Please specify when ordering.

Volume 1:
Volume 1 contains all the general applications of Sideways RAM. Menu, Printer Buffer, STL DFS 2.0, STL E00, RAMDISK, RFS (Rom generator), Fast Backup, Quickcopy, Index etc . . .

Volume 2:
Volume 2 contains the Solidisk Wordprocessor, Spelling Checker and the English Dictionary.

Volume 3:
Volume 3 contains the Solidisk Macro Basic (a program generator), a Linker-Editor, VMP the Virtual Memory Program, Sigen, a program to create your own dictionary and a sample French dictionary.

Volume 4:
Volume 4 contains the Solidisk Database.

Volume 5:
Volume 5 contains the Solidisk Sprites System.

SOFTWARE PACK 2
(optional)

Volume 6:
Volume 6 contains the Solidisk Spreadsheet.
Menu driven with standard options to create, edit, recalculate, print, search, sort, report generator and mailmerge.
Unlimited number of rows and columns. Each column can be as small as 2 or as big as 70 characters. All math functions are supported. Complete with home banking and simple portfolio management.

Volume 7:
Volume 7 contains Solidisk Play Tunes System and lots of tunes and write music using ordinary ABCD notation.

Volume 8:
STL Toolkit has 24 star commands (Status, Rwipe, Check, Find, Search and Replace, Mode 8, Join, Keyload, LVAR, Salvage, Expand, Edit etc . . .).
STL SFX is Solidisk screen effects utility. In mode 2, you can scale, move, reverse, mirror etc with superlative ease.
VDURECORDER is another screen aid utility. It remembers everything sent to the screen. When you want it, *REPLAY will show it all over again but with an incredible speed.
Solimon is a machine code monitor for both 6502 and 65C02.

Volume 9:
Volume 9 contains the STL ADFS and disc utilities (see next page).

Volume 10:
Volume 10 contains the Solidisk Teletext Editor.

Megagame Packs:
Around 70 games are currently under compilation.

PRICES

The listed price comprises the Sideways RAM fully guaranteed for 1 year, comprehensive manual and software pack 1 consisting of 5 diskettes formatted either in 40 or 80 track.

— SWR32 + 5 discs (32k unit, by far the most popular): £53.00
— SWR64 + 5 discs (64k unit, new introduction model): £80.00
— SWR128 + 5 discs (128k unit, the second most popular): £125.00
— Any Extra Software Pack (5 discs): £10.00 Specify 40 or 80 tracks.

As Solidisk Software Support Service has to produce in excess of £25,000 diskettes every month, we regret that we can no longer provide personalised service for disk washing. In extreme cases, when you need to update your software diskette, either reorder a new software package (which is always sold to you at nominal media cost i.e. £10.00 for 5 disks, including new manual, or return the old package with £2.00 to cover post and packing. You may avoid post and packing costs if you call at Solidisk's stand at any BBC micro exhibition or at the Solidisk office.

UPGRADING

Upgrading from a smaller model is very simple: you return the old RAM board to us with your payment for the difference.

The listed prices include new manuals when necessary.

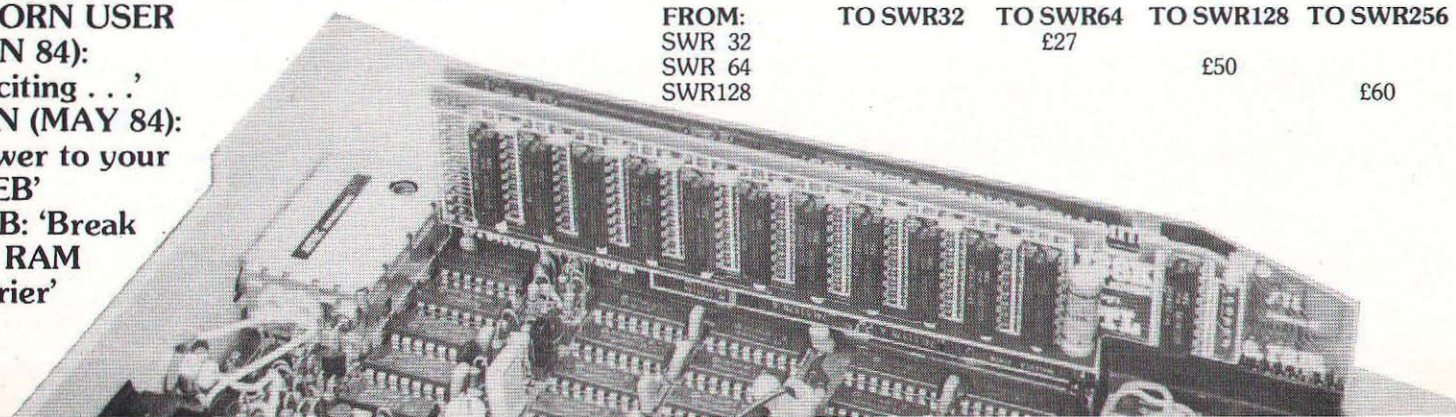
SOLIDISK TECHNOLOGY LIMITED, 17 SWEYNE AVE, SOUTHEND-ON-SEA, ESSEX SS2 6JQ. TEL. SOUTHEND (0702) 354674 (16 lines).

WHAT THE SPECIALISTS SAY:

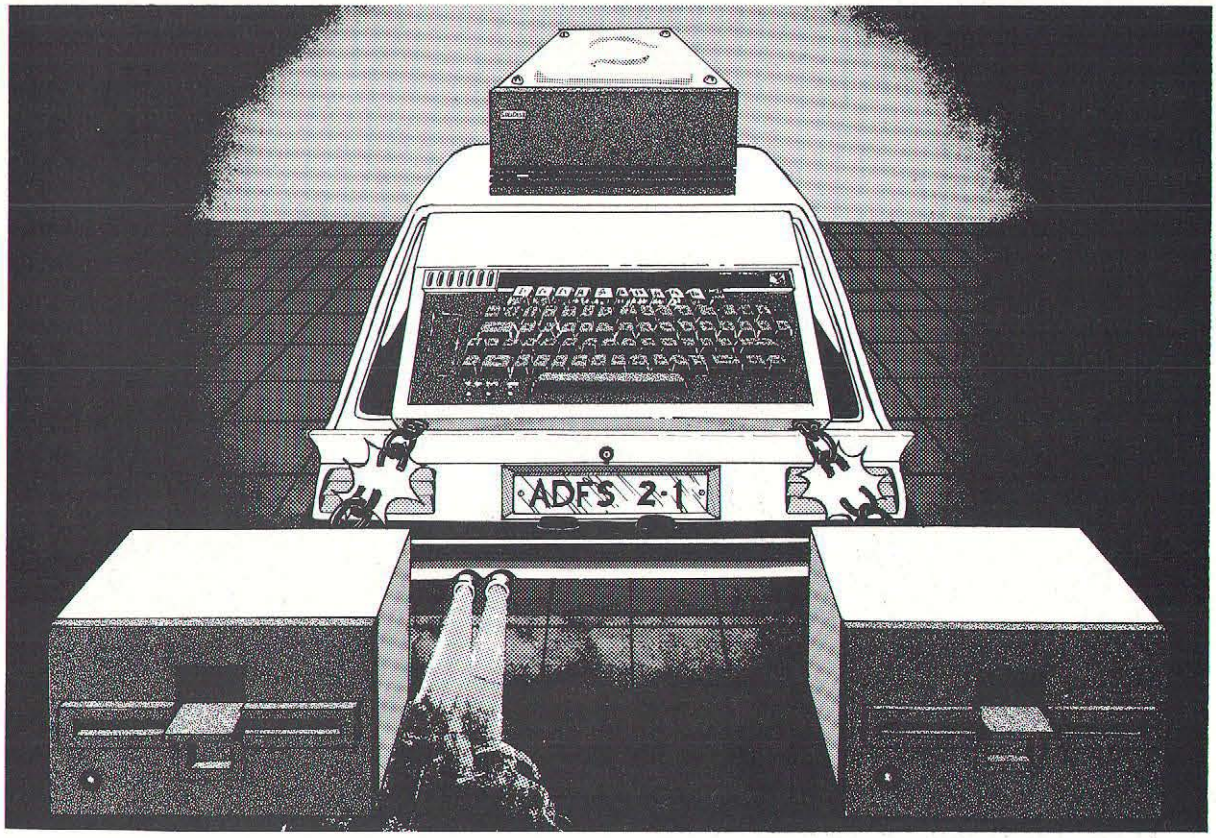
ACORN USER (JAN 84):
'Exciting . . .'
PCN (MAY 84):
'Power to your BEEB'
A&B: 'Break the RAM barrier'

Effective from 15th of JULY 85:

FROM:	TO SWR32	TO SWR64	TO SWR128	TO SWR256
SWR 32		£27		
SWR 64			£50	
SWR128				£60



LOAD AND RUN IN 100 MILLISECONDS (Or Less)...



Would you believe that your BBC Micro will LOAD and RUN a small BASIC program from your Winchester in less than a TENTH OF A SECOND?

Yes, it's perfectly true when you have a Winchester hard disc for it.

How well does your BBC computer compare against the best sellers in 'business micros' such as the IBM PC, the APRICOT, the COMPAQ or the SIRIUS?

Eric Bagshaw, of the prestigious National Computing Centre has devised a series of tests to reflect as closely as possible the performance of various business computers in actual use.

He put computers like the APRICOT, COMPAQ, FORTUNE, IBM, RAIR, SIRIUS, XEROX etc... through the same tests and published the results in the November 84 issue of PCW and the July 85 issue of Practical Computing but he missed out an important one: the BBC computer.

We had to wait until Gordon Taylor, journalist and reviewer for A&B Computing adapted the Bagshaw Disc Benchmarks to run on the BBC. We were quite astonished by the findings. Just look at the figures below:

BENCHMARKS	WITH FLOPPY*	WITH 20 MB HARD DISC**
BM0 *LOAD SIXTEEN	1.94	0.87
BM1 WRITE SEQUENTIAL FILE	9.57	2.86
BM2 READ SEQUENTIAL FILE	4.30	1.84
BM3 WRITE RANDOM FILE	22.62	10.37
BM4 READ RANDOM FILE	10.20	8.54
BM5 MULTI OPEN AND WRITE SEQ.	68.77	19.29
BM6 MULTI OPEN AND READ SEQ.	4.29	1.54
BM7 MULTI OPEN AND WRITE RANDOM	70.63	19.74
BM8 MULTI OPEN AND READ RANDOM	4.33	1.53
BM9 RANDOM READ IN RANDOM FILE	8.48	1.98
BM10 RANDOM WRITE IN RANDOM FILE	24.62	2.65
BM11 FRONT/END SWAP IN RANDOM FILE	368.06	65.18
BM12 CLOSE UP HOLES IN RANDOM FILE	33.66	18.26
BM13 CLOSE ALL DATA FILES	34.64	9.05
TIME IN SECONDS	663.97	163.70

* BBC computer fitted with 1.3 MB Solidisk 5.25" half height floppy disc drives, running Solidisk ADFS 2.1 (double density).
** BBC computer fitted with Solidisk XD20/40 Winchester (20MB), running Solidisk ADFS 2.1.
Now compare these figures with those published for leading business computers*:

COMPUTER		TIME	RANK
BBC with	20 MB**	164	1
FTS	20 MB	187	2
APRICOT XI	10 MB	192	3
SIRIUS	10 MB	203	4
IBM XT	10 MB	254	5

* Source: PCW September 1984.
** BBC computer fitted with Solidisk 20 MB, figure not published in PCW.

The BBC computer with Solidisk equipment is faster than the Fortune, the Apricot, the Sirius and up to 50% faster than the IBM PC-XT!!

At least until the new micros such as the ST520 or the AMIGA gain wide public acceptance (which may never happen), you can be quite sure that your BBC computer will remain the BEST computer around.

This is especially true for those who use the 6502, the Z80* and the 32016 Second Processors and ECONET.

Few have been able to discover the real strength of their BBC system, mostly because of the high price of Acorn Winchesters.
Now with Solidisk, price is no longer a barrier.

You can buy a 10 MB 'Minimum Configuration' at only £399.00 + VAT**. So, rather than waiting for Acorn to reduce their price, ring us today; we will despatch it to you by DATAPOST within 48 hours (credit card orders only).

* Z80 owners: CP/M BIOS 1.20 will run immediately on the Winchester because Solidisk DFS 2.1 has partitioning capability. This means that you can create 'drives' 4, 5, 6 etc... on the Winchester, and backing up your DFS floppy discs onto the Winchester.
Normally, a new MFM BIOS and new BOOT ROM are required.
(**) Minimum Configuration:
1) 10 MB drive, net formatted capacity.
2) Western Digital Hard Disc Controller.
3) Solidisk SASI interface.
4) Solidisk ADFS 2.1.
The minimum configuration is designed to work with all BBC Computers fitted with the later Switched Mode PSU (shiny) as on late issue 3, issue 4 and 7 machines alongside a single low power half height floppy disc drive such as the Mitsubishi in the SOlidisk Disc Offer, with or without a second processor and with or without the new style low power SWR64/128 board.
If you have any other add-on draining power directly from the computer's PSU, you must order the XD-10 or disconnect the floppy disc drive.
PLEASE NOTE: if you intend to use floppy disks with the ADFS, you must also purchase the Solidisk 1770 DFS interface or the 1770 + 8271 DFDC interface.
All Winchester drive offers contain some useful software such as database, wordprocessor and utilities. See full review in A&B Computing, September 85.
Solidisk also has one of the most efficient software support systems in the country; our engineers are experienced, enthusiast and helpful. Should you need assistance or advice, please do not hesitate to contact us and talk about it, we may be able to help since we may know of people using our Winchesters for programming, off loading mainframes, running Viewstore, running mail orders, medical practices etc...

SOLIDISK TECHNOLOGY LIMITED, 17 SWEYNE AVE, SOUTHEND-ON-SEA, ESSEX SS2 6JQ. TEL. SOUTHEND (0702) 354674 (16 lines).

ADVANCED FILING

SOLIDISK 1770 DISC INTERFACE
For the BBC Computer

The Solidisk Double Density 1770 Disc Interface enables the standard BBC B computer to use most current types of disk drive. Only 4 components need to be fitted into existing sockets on the computer and this can be done in a few minutes by novice and expert alike. It now can run all Acornsoft games (such as Elite, Aviator, Revs, Magic Mushroom etc. . . .) and most other protected discs (Castles Quest, Island's Music System etc. . . .)

The Disk filing System provides every facility that you would expect such as built-in disc formatter and verifier, automatic 40/80 track switching, programmable disc speed and is upgradable to the Solidisk ADFS.

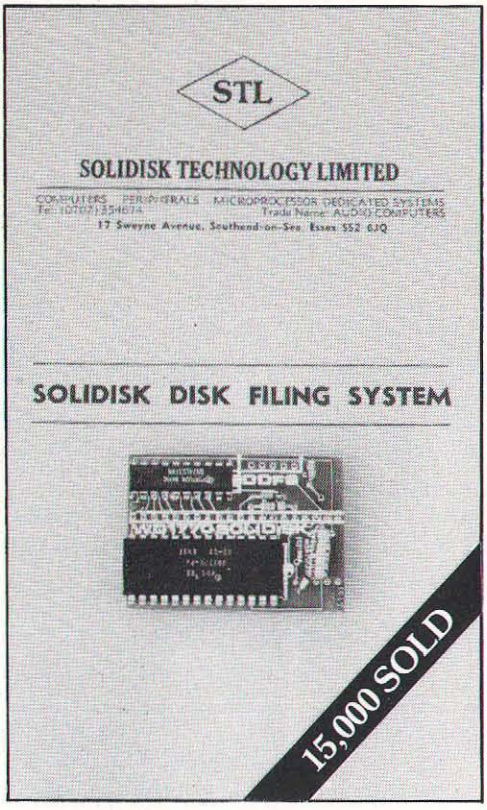
It also has some significant advantages over the Acorn 1770 system, mainly:

1) 60% more storage capacity in double density DFS, giving you up to 640k on a double sided 80 track diskette as opposed to only 400k bytes with the Acorn 1770.

2) Better emulation of the 8271: some protected discs (such as the early Acornsoft's Elite game) would not run on the Acorn 1770.

3) Built-in useful disk utilities such as TAPEDISC, MZAP, DZAP (disc sector editor), RECOVER, RESTORE etc. . . . which would add to cost if bought separately.

PRICE:
The 1770 Double Density Disc Interface costs £45.00 inclusive of VAT, DFS 2.0 ROM, manual and 1 year guarantee.
If you wish, you can also order it with the ADFS ROM and the total cost is £55.00 inclusive.



SOLIDISK ADVANCED DISC FILING SYSTEM
for the BBC Computer

The Advanced Disc filing System (ADFS) is now officially endorsed by Acorn and is certainly going to revolutionise software for the BBC Micro.
The Solidisk ADFS, reviewed in A&B Computing 85, whilst comparable to Acorn's ADFS, is very much designed with floppy disk and DFS software compatibility in mind.
The Solidisk ADFS will run most software available for the BBC computer, including all Acorn languages and the View family, Computer Concepts Wordwise and the INTER family, Gemini's Datagem etc. . . .
It also has PAGE at the lowest value, eg &1900 with one channel. It can realistically support up to 10 opened channels without slowing down the program as it does not swap buffer pages to disc.
It also has utilities such as *FORM, *VERIFY, *BACKUP etc not found in the Acorn ADFS ROM.
It is also very responsive and double safe, with automatic head settling and read after write automatic error correction.
It runs with all Acorn second processors including the latest 32016 mainframe machine (only Z80 CP/M requires new MFM BIOS) and it is fully supported by the Solidisk Toolkit ROM.
The Solidisk ADFS is available for the BBC B or PLUS fitted with the Solidisk 1770 Disc Interface or the Acorn 1770 as a TWO ROM chip set, consisting of the DFS 2.1 and the ADFS 2.1. A 32k ROM may be supplied in lieu of the 2 ROMs on request. It costs £20.00 when purchased separately like this.

SOLIDISK 1770 + 8271 Disc Interface
For the BBC Computer

As Acorn are now backing the Advanced Disc Filing System, many of you, having got the 8271 Disc Interface, wonder what you are going to do next.

If you get the Acorn 1770 upgrade, you will have to part with your 8271 and possibly a few expensive pieces of software such as the old Elite or Replica II which won't run without the 8271. It is still only a single density DFS.

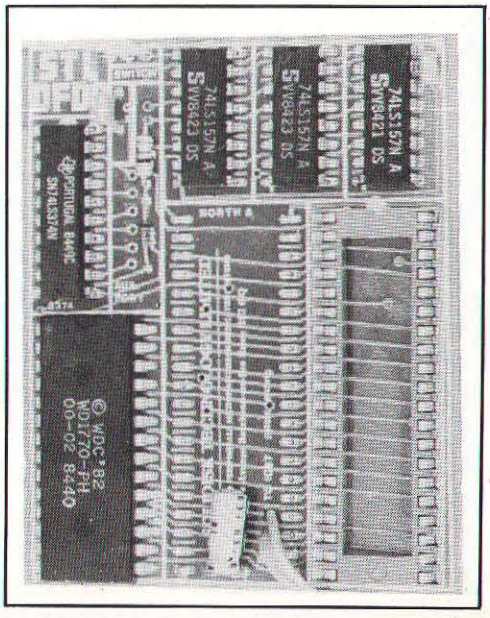
The alternative is to get the Solidisk 1770 + 8271 Disc Interface (known as the DFDC, acronym for Dual Floppy Disc Controller) which will let you keep the 8271 which you already have and add 1770 Double Density capability.

It is low cost, tiday and easy to fit.

At a flick of a switch, you can select either the 8271 or the 1770 Floppy Disc Controller.

You will have the best of both worlds: compatibility with the 8271 and fascination with the 1770 ADFS.

It is probably the fastest, more powerful and versatile Disc System for the BBC.
The DFDC costs £60.00 including VAT, manual, DFS 2.0 ROM and ADFS 2.1 ROM.



SOLIDISK TECHNOLOGY LIMITED, 17 SWEYNE AVE, SOUTHEND-ON-SEA, ESSEX SS2 6JQ. TEL. SOUTHEND (0702) 354674 (16 lines).

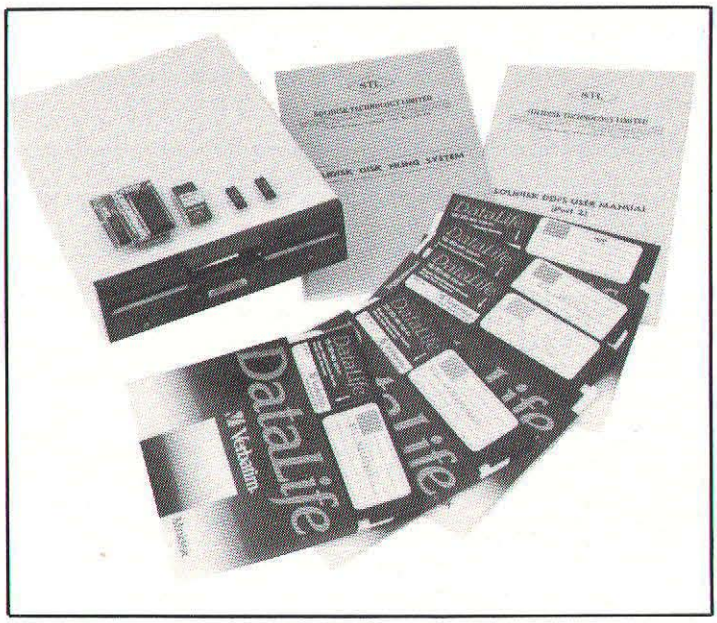
SOLIDISK MITSUBISHI DISC OFFER

A Complete Disc System containing everything you need, including 1 Blank Diskette, at a new low price of only £200.00 inclusive.
The Misubishi Offer Comprises:

— One Mitsubishi MF4853, Double Sided, 80 Track Disc Drive. Cased in beige with all leads.
— One Solidisk DDFS complete Disc Upgrade with 2.0 ROM.
— One Verbatim MD525 Blank Diskette.
— One Software Pack and User Manuals.
— Full one year guarantee.
FREE: ADFS ROM!

HARDWARE:

As described earlier as shown opposite.



THE SOFTWARE:

The software comprises 5 diskettes, formatted in 80 tracks.

Volume 1:
Solidisk Word Processor and Spelling checker with English dictionary. Easier than View, Wordwise or Scribe, you enter the WP by *WP filename, then start typing. What you see on the screen will be exactly printed on paper.
Most useful commands are all displayed on the top part of the screen.
It features 80 column screen throughout with direct on screen justification, automatic margins, page numbering, Wordstar like editing commands for block move, block delete, block copy, print, save, load text to cursor, insert and overwrite, search and replace, total word count, word frequency count, free space, *commands etc. Solidisk WP loads and saves texts in under 1 second flat and also you may save the edited version as many times and under any filename you like.
It has the unusual ability to spellcheck your document without leaving it, simply by typing in Control-U.
It has self dictionary generating capability too. You may create any dictionary you like by just simply entering the words.

Volume 2:
Volume 2 contains Solidisk Database. It is a random access system, completely menu driven and easily customised to suit any particular filing need.
It features unlimited filesize and number of records, supports all maths functions and 80 column screen throughout.
It is simple to understand and to use. You take one of the 15 different options to start. Each option will then lead to a new menu and so on. Mostly you only have to enter an appropriate data or hit the RETURN key.
You can design new databases, list all records, edit them, merge them, split them, making mailshot with Solidisk WP etc. . . We use it to process all your orders.

Volume 3:
Volume 3 contains Solidisk Spreadsheet. Menu driven with standard options to create, edit, recalculate, print, search, sort, report generator and mailmerge.
Unlimited number of rows and columns. Each column can be as small as 2 or as big as 70 characters. All math functions are supported. Complete with home banking and simple portfolio management.

Volume 4:
Volume 4 contains the Solidisk tape to disc program and a selection of computer games.

Volume 5:
Volume 5 contains MBASIC, SIGEN and other disc utilities.

IMPORTANT:

If you upgrade to a twin Mitsubishi disc system, Solidisk will offer you, subject to stock availability, a completely free set of CPU and keyboard case as shown earlier.

You should return your disc drive with payment for another Mitsubishi MF4853 (£160.00) or alternatively, save by ordering right now a Fantastic Offer at only £340.00.

The keyboard is 12 mm lower than the normal BBC case, 2-3 degrees more angled, weighs only 5lbs, does not skid about on the desk while enjoying 2 feet of freedom from the computer unit — and makes a lot of difference to typists (even 2 fingered ones!)

The CPU accommodates 2 disk drives and withstands any heavy weight monitor while providing easy attachment for the whole system, CPU and disk drives, to the desk if need be. Good ventilation to the power supply too and hides all untiday cables away.

PRICE

The complete CPU and keyboard case costs £30.00 inclusive + £3.00 post and packing. The system could be yours free as part of the Mitsubishi twin discs offer.

SOLIDISK TECHNOLOGY LIMITED, 17 SWEYNE AVE, SOUTHEND-ON-SEA, ESSEX SS2 6JQ. TEL. SOUTHEND (0702) 354674 (16 lines).

WIN THE MAZE GAME . . .

Investing in extra equipment for your BBC Micro really needs thinking about; bear in mind, for instance, that for the last five years, all efforts have been put into more memory and faster disks.

The problem is that you may end up with a hotch potch of add-ons from different suppliers, and some may be reluctant to give advice on the whole ensemble.

We should know — here at Solidisk, we see it every day.

So how can you, too, keep pace with the latest technology?

The easy answer is buy important pieces of hardware from Solidisk.

Solidisk design rules adhere strictly to 2 principles:

a) 'Minimum Disruption' to the standard machine.
b) 'Do it in software' if at all possible.

We have Double Density Disc Interface, Winchester, Advanced DFS, Sideways RAM, Shadow/MOS RAM and MODEM and more than 50,000 happy users.

They are happy because their computers have more facilities, run faster, look tidy and usually cost them less money.

For many, it's the support that they receive that they are most happy about.

With over 90 Local Experts covering England, Scotland and Wales, with presence in Holland, France and Australia, Solidisk can offer many users free fitting and advice.

We also make available an ever increasing catalogue of free software to all Solidisk users.

Solidisk Software Support Service can give you the sort of service only matched by the largest companies.

It's up to you to decide. Although they are happy now, many have made the journey twice and they are not ready to do it again.

SOLIDISK LOCAL EXPERTS

Solidisk has a nationwide network of more than 90 local experts and a growing number of experts overseas.

They do not sell our equipment, but they can generally fit it for you absolutely free of charge. If you would like to take advantage of this unparalleled service, ring us once you have our equipment for the name and telephone number of your nearest one. All areas are currently covered, but we still intend to expand the network; if you are interested, please call us for details.

PRICE LIST: (all prices include VAT unless otherwise indicated. We export world wide — contact us for details on prices and delivery).

	Prices	P&P
BBC SIDEWAYS RAM:		
SWR32	£53.00	£2.00
SWR64	£80.00	£2.00
SWR128	£125.00	£2.00
SWR256	£175.00	£2.00
Software pack 2	£10.00	£2.00
NB: all versions of the Sideways Ram come supplied with Software Pack 1 free of charge. Please specify 40 or 80 tracks when ordering. Pack 1 is available without the Sideways Ram at a cost of £10 + £2 P + P.		
1770 DISC INTERFACE:		
1770 Double Density with DFS 2.0 ROM	£45.00	£1.00
1770 Double Density with ADFS	£55.00	£1.00
1770 + 8271 DFDC with ADFS	£60.00	£1.00
DISC OFFERS:		
640k with ADFS Disc Offer (£10.00 Datapost)	£200.00	£3.00
1.3MB with ADFS, CPU and keyboard case (£10.00 Datapost)	£340.00	£5.00
WINCHESTER DISC DRIVES:		
XD-10 without PSU 10MB (Minimum config)	£399.00	+ VAT
XD-10 10MB (with additional PSU)	£460.00	+ VAT
The following prices include keyboard and CPU case:		
XD-20/40 20 MB	£699.00	+ VAT
XD-20/40 TWIN (40 MB)	£1100.00	+ VAT
XD-30/60 30 MB	£900.00	+ VAT
XD-30/60 TWIN (60MB)	£1300.00	+ VAT
All Winchester systems are despatched by Datapost within the UK, included in the prices above. (Please remember that if you also need to use floppy disks in conjunction with these, it is advisable to also purchase our 1770 DFS interface or 1770 + 8271 DFDC interface — see previous pages.)		
EPROMS:		
2764 × 5	£24.00	£1.00
27128 × 3	£25.00	£1.00
27256 (to be used on UVIPROM32)	£10.00	£1.00
UVIPROM Eprom programmer, suitable for 2764, 27128	£19.00	£2.00
UVIPROM32 Eprom programmer, special for AMD27256	£20.00	£2.00
UVIPAC Eprom Eraser	£21.00	£2.00
DISKETTES:		
Datalife MD525 SS/DD	£17.00	£1.00
Datalife MD557 2S/4D	£28.00	£1.00
MISC.		
CPU and KEYBOARD case	£30.00	£4.00
Solidisk TOOLKIT a must for ADFS	£10.00	£1.00
DFS + ADFS if purchased separately	£20.00	£1.00
EFS for the Electron	£59.00	£1.00
Special 3.5" Electron disc offer	£197.00	£3.00
Please circle: BBC Model B/B PLUS/40 tracks/80 tracks.		

Name: _____

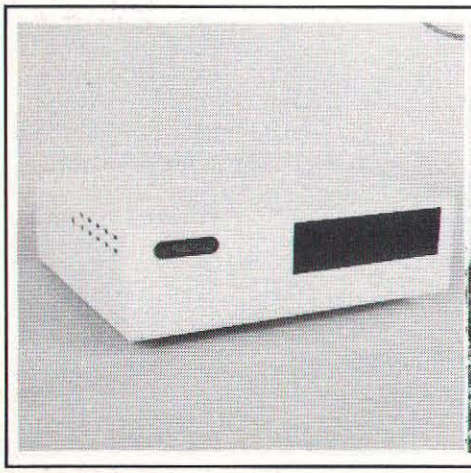
Address: _____

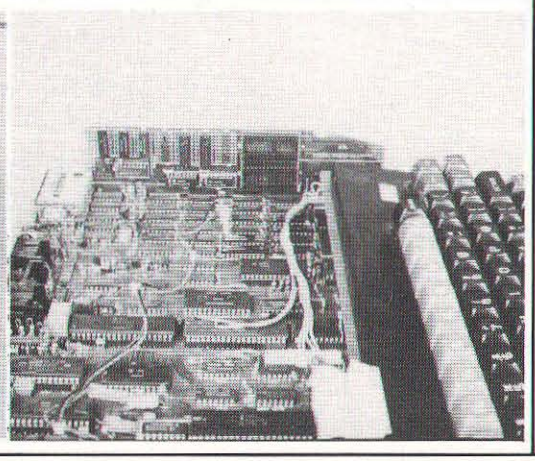
_____

BARCLAY/ACCESS : : : : : : : : : : : : : : : : : :

We reserve the right to change prices and specifications without notice and delivery is subject to availability. Please call our office for confirmation. Callers are welcome Monday to Friday 9.30 to 5.30. Please note that we are closed on Saturdays. Callers requiring fittings or installation checks: it is now essential to ring for an appointment.

SOLIDISK TECHNOLOGY LTD 17 SWEYNE AVENUE SOUTHEND ESSEX SS2 6JQ
TEL: (0702) 354674 (16 LINES)





SOLIDISK TECHNOLOGY

Tel. SOUTHEND (0702) 354674

(16 lines)

Software Reviews

REPLAY

Replay is a magical piece of equipment. Especially interesting for the specialist games player, it has certainly proved useful in reviewing (in its literal sense) all sorts of software, dumping screen (some for A&B) and also in aiding and abetting assaults on high-score tables everywhere.

Of course taking on games with a gadget is not very ethical but when you *know* that you can complete the first fifteen screens without losing a life...

Replay allows you to save a game (or other program) in its entirety to disc. Thus, when you recall the disc file, the software will run from the point at which you saved it! And that's not all. Replay consists of a ROM which has its own socket. This plugs into a spare sideways socket. It's wired up to a couple of points on the BBC board, a link and one of the legs of IC89.

The Replay menu offers choices to format a Replay disc, to create a file, to delete a file, to display a screen. An "empty" file has to be created ready to recieve the contents of BBC memory. This is done a file at a time, one in operation on each disc surface at any one time. You can use up to four disc surfaces.

A Replay file is reloaded by choosing the appropriate letter A,B,C etc from the disc menu display. The number of files per disc varies with the capacity of the filing system. You can have six on your Acorn DFS 80 track drive for instance.

You can reload to continue a game where you left off or you can use the screen load option to dump a screen.

Replay can be summoned at any time by pressing a small switch connected by cable to the ROM. This interrupts whatever is going on on screen. The CAPS LK LED flashes. Hitting function keys 0 to 3 sends the whole memory to the disc surface of your choice. Function key 6 will proceed with the program.

Replay deals with most games. Only special screen formats can cause problems. Some of the more recent are recognised and the correct data applied to the video controller. New games however won't be recognised and you may have to resort to the Edit option. This allows the easy editing of the registers of the video controller.

The manual does its best to explain but you'll need to go to an advanced guide on the 6854 to understand how to fiddle the registers to your needs. Barry Landsberg's article in A&B last autumn will tell you all you need to know.

As well as being great for pausing screens, saving them to go back to and dumping them, REPLAY is a useful tape to disc and disc to disc backup device. Programs saved in this way cannot be used on any other BBC system, even one with a Replay device installed.

Replay is a pretty marvellous device, more than a toy for those who have to document software, very useful if you can only buy a tape version of a program (it doesn't matter what nooks and crannies the code is packed into) and a good opportunity to have fun with the screens of your choice. A liberating piece of software and highly recommended. **AB**

Publisher Vine Micros (0304 812276)
Machine Model B
Price £35.00

Village of Lost Souls

This is very interesting game. It is an orthodox, text only adventure set in a medieval village in, as the publishers put it, a 14th Century slightly different from our own. The accompanying blurb goes on about various magic orders but the play that I have discovered so far does not involve any arbitrary magical solutions, but fairly logical, if hard, problem solving.

An unusual feature of the game is that one can wander into many of the game's 200 plus locations from the starting position without being killed or having to solve problems. One also encounters a huge number of objects whose purpose may remain obscure for a long time. I found the only practical method involved a detailed mapping and cataloguing exercise at the outset. Then you can start thinking about what you ought to be doing. The first puzzles are not too hard but once solved you get clues to much trickier missions. I managed to solve the riddle of the miller's boots, but I am stuck by a number of other puzzles. For example, there is this vampire blocking my way into the crypt. After much industrious searching I managed to approach the beast armed with garlic, a crucifix and a looking glass and what did it do? The stupid creature happily sucked me dry — it obviously hadn't been reading the right books.

The game is fair, in that one can tell what puzzles need to be solved, but it is also very difficult because you have so many options and objects to deal with. Many hours of work have netted me a feeble 250 out of 1200 points so far. I trust that Magus will provide a clue sheet or an efficient hint service for their frustrated witch-finders.

In summary, this is an interesting and challenging adventure game. Not for beginners, but recommended for the seasoned campaigner. **JE**

Publisher Magus
Machine Model B
Price £7.95

One Last Game

I sat for five minutes waiting for this one to load (enduring an appalling, fit inducing flashing screen) then crash — no room! The program does not reset PAGE to &E00 for you, and nor do the instructions tell you to do so before

```
REPLAY System by Vine Micros
(C) 1985 designed by R.P.D Mallett

     Drive: 0 Tracks: 80 Sectors: 10

R   A: FRAK
R   B: CASTLE
R   C: MUSH1
R   D: MUSH2
W   E: REPLAY
    F:

Press appropriate letter to run a
program, or number for a utility :-
1) create file      2) delete file
3) display screen   4) edit file
5) initialise disc  6) change drive
<SPACE> catalogue REPLAY disc
Option ?
```

CONTINUED OVER

CHAINing. Software companies please note — most reviewers have disc machines and this sort of oversight ensures an ill tempered first look at your product.

Fair as always, I reloaded One Last Game later when I had cooled down, but it didn't help me to like it any better. You control a spaceship on the left of the screen and fire at a fleet of alien craft to the right — defined as rather crude sprites. You have to dodge bullets (or laser blasts or whatever) and also the odd spacecraft which breaks off from the main pack. The only controls you have are up, down and fire. There is also a strip of horizontally scrolling landscape at the bottom of the screen to provide some overall effect of motion — the effect is that the wave of alien craft are flying backwards at exactly the same speed with which you are trying to approach them!

I should award some marks for provision of options for joystick control, sound on/off and starting at higher screens, since many games on the market still fail to provide such obvious essentials. That said, the game itself remains a dull implementation of an exceptionally unoriginal theme.**JE**

Publisher Bevan Technology
Machine Model B
Price £7.95

Whist/Bridge

Whist and Bridge are two separate software packages from Livewire which I am reviewing together for convenience — the price quoted is for either program bought separately. There seems in fact to be some common use of program code between the two, as the representation of the hands and the card play are very similar. One odd difference is that while one remains South in both games, the deal rotates only in Whist. In Bridge you are always dealer with the first bid — an annoying limitation.

The programs use the Mode 4, monochrome screen presumably to allow bigger programs, but the displays are quite adequate and foreground and background colours may be adjusted to taste.

The problems lie in the playing strength of the programs. I played a few hands of Whist and noted some rather dubious plays by the program and also was rather irritated that one was given few facilities for learning from the play. For example, you cannot replay a hand or play with the cards open. This applies also to Bridge, which is obviously a more complex program to allow bidding and playing with open dummies.

The Bridge program uses a simple version of the Acol bidding system which is adequately explained in the accompanying documentation. There are options for auto-bidding and card play. I ran a few hands on auto to try to judge the playing strength of the program. Once again I was not impressed, but not being an expert Bridge player I lent the package to a friend who is a county player for evaluation. His assessment was damning — he noted many elementary blunders in bidding and play which all but the weakest beginner would avoid. He emphasised that he was not judging by his own standard but by that of an ordinary club player.

I reported recently on the latest version of the White Knight Chess Program which does reach at least strong club standard in most phases of the game. I believe that programming methods for chess play have been much better researched since this was a major interest in the field of artificial intelligence at one time. Even so, I find it hard to believe that card games cannot be simulated with much greater playing strength than Live Wire have achieved, and I really cannot recommend these programs for any significant entertainment or educational value.

A final complaint. I discovered in the course of the review that the disc version uses the sort of protection that requires the old 8271 floppy disc controller to be present. Hence it would not run with my Solidisk DFS (1770 version) and I assume that it will not run on the BBC Plus either. The company admitted to me on the phone that they had not tested it on the new machine. It really does seem a nonsense to me that companies should still be producing new discs which are incompatible with the hardware of many BBC users.**JE**

Publisher Live Wire
Machine Model B/ Electron
Price £9-95/£11-95

Video's Revenge

Budgie (Budget Software) are one of the companies developing the idea of 'cheap' programs, for which they should be encouraged. Whilst it is worth paying for really outstanding software, such as Elite, the ordinary run-of-the mill program always seems overpriced to me at the typical £8 mark. Video's Revenge is a good example. The program is nothing special but it is no worse than the usual standard and at £3 is definitely fair value for money.

The game is an undisguised alien blaster. Your starship can be moved left and right across the screen but vertical motion causes the whole display to scroll. You shoot at a variety of hostile spacecraft who can destroy you by direct contact or by bombs. One type disconcertingly spawns a bunch of vicious baby ships when you blast it. There is also, apparently, a chance to acquire extra energy by hitting a balloon 12 times, though I have not yet managed to achieve this. Depending upon availability of energy units you can also use a smart bomb to wipe out everything on screen or become temporarily invisible and immune.

The sprite animation and screen scrolling are smooth and flicker free and the general presentation of graphics well up to typical standard of arcade games. Apart from the obvious lack of originality, my main complaints are (i) that I found the game rather too fast for comfortable play and was hampered by rather awkward key controls (there is no joystick option) and (ii) I object to the tedious process of tape loading being unnecessarily lengthened by the *LOADing of a graphics screen of no game significance.

In summary a competent though unexceptional game which provides good value for its low price.**JE**

Publisher Budgie
Machine Model B
Price £2.99

1st in BROMLEY

6502 & Z80 SECOND PROCESSORS
TELETEXT ADAPTORS
TORCH Z80 DISC PACK
MICROVITEC MONITORS
EPSON PRINTERS
GRAF PAD
BITSTIK

EVERYTHING FOR THE BBC OWNER

ACORN
VOLTMACE
QUICKSHOT
JOYSTICKS
RH VIDEO DIGITISER
ACACIA RTC & RAM UNIT
CUMANA & MIDWICH DISC DRIVES
NEW INDEX UTILITY ROM

THE DATA STORE
6 CHATTERTON ROAD, BROMLEY, KENT
01 - 460 8991

A Talent for Inquisition?

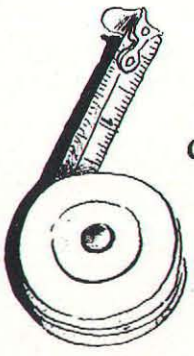
Village of Lost Souls

A machine code adventure
for the BBC B micro OS 1.2 +

send a cheque or PO for £9.95 to

Magus, 4 Toronto Close, Durrington, Worthing, West Sussex BN13 2TD.

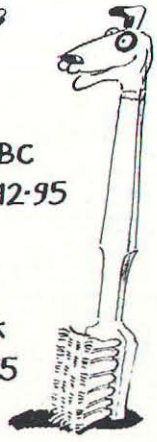
The Secret Diary of Adrian Mole aged 13¾
Sue Townsend



Program by Level 9

SPECTRUM · CBM 64 · AMSTRAD · ATARI · MSX · BBC
CASSETTES £9·95 · DISCS (BBC & CBM 64 ONLY) £12·95

ALSO AVAILABLE
THE ADRIAN MOLE SECRET DIARY KIT
The Computer game + 1986 Diary in one pack
Cassette versions £14·95 Disc versions £17·95



MOSAIC BOOKWARE

Distributed by WHS Distributors, St Johns House, East Street, Leicester
Tel. (0533) 551196

Software Sale

A growing library of useful and entertaining software for BBC and Electron Micros.

New software this month:
A&B Christmas Games Compendium. A massive compilation of games from A&B Computing. A bumper pack of games to see you through Christmas on one tape for the BBC Micro and on another for the Acorn Electron. For the BBC disc drive owner, there's a disc compilation (40 or 80 track)

AYO - an intellectual challenge
Tank Battle - two player game, a shoot out scenario
Backgammon - graphically brilliant, easy to play, great game
Domgrid - dominoes given an ingenious twist
Doodlebug - fast action, tactical contest with the computer
Dunefighters - two buggies skimming the surface of a sand blasted planet. Hunting one another on the video scanners...
Pontoon - excellently presented, a super card game
Scissor - the classic outguessing contest
Space War - addictive, multiple waves of alien ships, scrolling surface and you.
Wiggler - a worm game to test the reactions. Avoid the cans, eat the strawberries.
1984 - sliding block puzzle with a familiar theme but all the old problems of completion!
Scruge is the star performer this month in the sale. It's a super game with Scruge wandering around the caverns in search of gold. To win you have to do battle with the ghosts and that's not easy.

Scruge is first up on our super Christmas games tape for BBC and Electron. All the games listed for the Electron also work on the BBC Micro. The games Falcon Pilot and Fortune Teller from this issue are included on both Electron and BBC tapes. The whole caboodle is naturally on BBC disc (40/80 as you prefer).

CHRISTMAS GAMES
Model B, B+, Electron cassette
(BBC and Electron) £4.00
disc (BBC 40/80) £6.00

Subscriptions to the A&B Computing Software Sale offer considerable advantages over the "month by month" approach. For £48 cassette (£24 for 6 months) and £72 disc (£36 for 6 months) you will recieve *all* the offers in the Software Sale each month. No awkward choices. All the software will be despatched to you automatically. Take a look at the sort of software we are currently offering and then decide.

Documentation is strictly limited since the software is designed to complement the original articles in the magazine. If you don't have the relevant magazine then mention that you require extra documentation and we will supply photocopies.

DOUBLE DUTCH (stochastic generator)/BASIC utilities
FONT Printer Character Generator/Printer Buffer/Old English Font/Graphplot/Sdump
Shadow utilities - Data Logger and Mark Book.

Double Dutch comes with large demonstration files developed within the View wordprocessor. This suite of programs is packaged with BASIC utilities, a full suite of packer, renumber, search and replace etc.

Printer Character Generator (supplied with two example character sets), Printer Buffer, Old English Font, Graphplot and Sdump are designed to act as an accompaniment to Easyword — our full feature wordprocessor. Old English Font provides quality titling, Graphplot, clear graphic output for explaining figures, and Sdump to dump the lot in shades of grey to a printer.

All the above are available on *one* disc (see order form).

The shadow programming combination of Data Logger and Mark Book is available for BBC Plus owners, on disc only. Mark Book is available for second processor owners (6502) without Data Logger. All orders for this suite will also recieve the Double Dutch and FONT suites.

The Easyword backup suite offers lots of possibilities. Why not use Buffer to provide more room for speech. Use our speech synthesiser (coming soon) to "speak" the combinations of sounds stochastically generated by Double Dutch. Use Easyword to prepare text for Double Dutch, spooling it out in the correct form. Buffer will provide a printer buffer for Easyword printouts in the style of your choice (designed with the Printer Character Generator). The building blocks are here, go to work.

IKON A&B's own answer to mouse software with full graphics facilities and icon editing (without separate utility).

TOUCH OF CLASS A very

classy printer graphics program, (Epson compatible) for designing and dumping personalised letter heads.

MOSAIC Multicoloured building blocks towards your own cartoon characters, landscapes and sprites.

CADPACK Computer Aided Design. Sophisticated drawing, filing and retrieving facilities. A full library of drawing routines, scale, post and redraw.

ADVENTURESCAPE An Adventure writing environment, opening up a hitherto closed world of arcane programming. It leaves you free to explore your own ideas, to design an Adventure world, to populate it with characters and objects.

You can spin a tale, weave a patchwork of stirring and inspiring locations, leavw clues, set puzzles and lay traps. A useful software tool, a vehicle for educational "walks".

Xanadu is the first game to be written with Adventurescape and takes you into the mythical world of Kubla Khan. Adventurescape will be fully supported in the future by A&B Computing with new and exciting applications and extensions.

The disc only upgrade to Adventurescape is supplied with all disc purchases and a new disc only Adventure, Amnesia, is available from Phoenix Software, priced £6.00 (cheques made payable to Phoenix Software).

A compilation of adventures, including a number of Adventurescape epics, will appear soon.

GLOBAL VIEW Global View is an educational and graphically stunning representation of the earth's time zones and day and night around the globe. High resolution b/w and colour versions supplied as well as sister program Down to Earth.

BOUNCER Arcade action from down under? Avoid the fruit throwing monkeys to rescue the kanga offspring. Those monkeys are bounders!

ANDROID/ALLY PALLY Android transports you to a desolate alien planet for the ultimate shoot out. While Ally Pally lands you slap bang in the middle of a computer exhibition complete with all the well-known BBC Micro and Electron companies. Beat the crowds, find the kids, top up your failing energy at the bar! BBC only.

GAMES A compilation of Adventure (Grisley Gardens) and arcade. Skoogs takes you into the swamp, Formula One to the track and Star Encounter to the edge of the galaxy. Classic games. Excellent value.

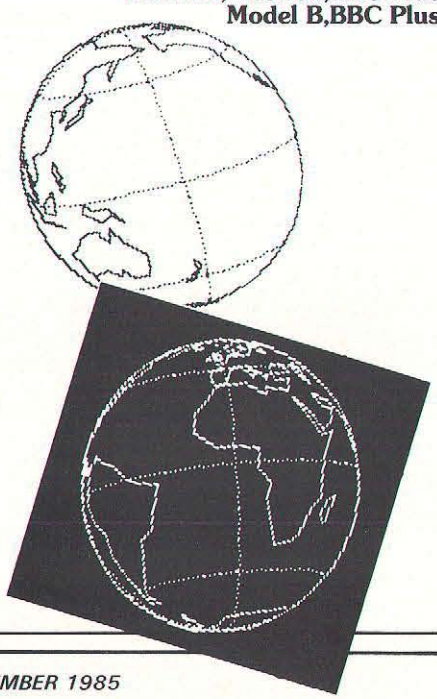
EDUCATIONAL A disc full of educational games with documentation files. From telling the time to interactive punctuation tests to a program generator for computer/audio combination.

MODE 7 A full Mode 7 suite. Scrolling, editing text and graphics, automatic conversion of Teletext screens to BASIC, a Mode 7 screen dump, frame combination.

All disc copies of the Software Sale come complete with our Checker program. So if you do need to type in a listing, Checker will be at your fingertips making de-bugging easy.

1 DOUBLE DUTCH	**Model B,Electron,BBC Plus**
2 PRINTER CHARACTER GENERATOR	**Model B,Electron,BBC Plus**
3 SHADOW PROGRAMMING SUITE	**BBC Plus**
4 IKON/TOUCH OF CLASS	**Model B,Electron,BBC Plus**
5 MOSAIC/CADPACK	**Model B,Electron,BBC Plus**
6 ADVENTURESCAPE	**Model B,Electron, DFS upgrade,BBC Plus**
7 GLOBAL VIEW	**Model B,Electron,BBC Plus**
8 BOUNCER	**Model B,Electron,BBC Plus**
9 ANDROID/ALLY PALLY	**Model B**
10 GAMES	**Model B,Electron**
11 EDUCATIONAL	**Model B,Electron,BBC Plus**
12 MODE 7	**Model B,BBC Plus**

MEDIA/PRICES
1 cassette £4.00
2 cassette £4.00
1/2/3 disc £10.00
4 cassette £4.00
5 cassette £4.00
4/5 disc £10.00
6 cassette £4.00
disc £6.00
7 cassette £4.00
disc £6.00
8 cassette £4.00
disc £6.00
9 cassette £4.00
disc £6.00
10 cassette £4.00
disc £6.00
11 disc £6.00
12 disc £6.00



If you wish to order from the software sale simply fill in the coupon below with the number(s) of the order(s).

Please send me:

__copies of A&B Christmas Games Compendium.

__copies of product number __

__copies of product number __

__copies of product number __

Please open a subscription to the A&B Software Sale for 12 / 6 months.

I enclose a cheque/postal order made payable to ASP for £.....

Name .....
Address .....
.....
.....
.....
Postcode .....

Signed.....
Date.....

Send this coupon together with your cheque/postal order to Software Sale, A&B Computing, Number One Golden Square, London W1R 3AB. All prices include VAT and postage and packing.

Special Offer

Three extra special programs on offer from the A&B software sale

DELIVERY
Not to be missed. Designed for kids, fun for all, a game but educational. The dialogue is humourous and varied, the scenario realistic. An open ended package requiring and instantly gaining the attention of children.

EASYWORD
Like Wordwise, Easyword is a 'natural'. And it's available on the Electron. Straightforward selections from a menu deal with filing and printing. Writing text is not hampered by the need for block formatting or editing. Inserting or deleting text does not cause havoc.

If you want a linespace you just hit the RETURN key. And the cursor keys move you around the text area. Just like Wordwise.

Search and replace and Easyword information are available at the touch of a key. The function keys control the easy editing of text. Sounds easy, and yet all the major functions you would expect from a wordprocessor are available.

Easyword is ideal for producing impressively formatted and perfectly corrected manuals, product catalogues, reports, essays, letters, forms, questionaires, the church gazette, the amateur theatrical programme, ticket to a lunchtime concert, indeed anything you might care to print, even poetry...

The Moving Finger writes; and, having writ,
Moves on: nor all thy Piety nor Wit
Shall lure it back to cancel half a Line,
Nor all thy Tears wash out a Word of it.

OMAR KHAYYAM, Rubaiyat

To quote Ray Hammond in his book about Logo, Forward 100, published by Penguin and reviewed in *A&B Computing:*

"Imagine now the possible effect on a child who starts writing on a word-processor from the age of seven. What will be that child's attitude to writing? How will she feel when her stories appear perfectly printed? If revision is automatic, how will she regard a first draft? What is the impact of her knowledge that she can go back and restructure the first paragraph, or move events around within her piece? How will this affect 'writer's block'? Will she ever regard the story as 'finished', when she can so easily go back to it with her teacher's comments ringing in her ears? A few keystrokes and the composition is rearranged to take account of these suggestions. When she is an adult, how will her attitude to writing differ from that of a child taught to write without the freedom of a word-processor?"

AMNESIA
The disc upgrade to Adventures cape includes a full text editor for the formulation of locations and messages to incorporate into your adventures. The first example of these extra facilities is the psycological thriller, Amnesia. Use of discs results in a truly massive Adventure which should keep you glued to you keyboard this Christmas.

If you wish to order from the software sale simply fill in the coupon below with the number(s) of the order(s).

Please send me:

_____ copies of Delivery

_____ disc disc 40/80 (delete where applicable)@£6.00

_____ cassette@£4.00

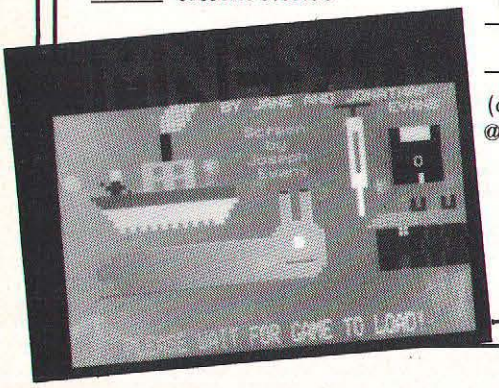
_____ copies of Easyword

_____ BBC disc 40/80 (delete where applicable)@£7.50

_____ cassette@5.00

_____ Electron 3 inch disc@£9.00

_____ cassette@£5.00



_____ copies of Amnesia

_____ disc only 40/80

(delete where applicable) @£6.00

I enclose a cheque/postal order made payable to Phoenix Software

for £.....

Name .....

Address.....

.....

.....

.....

Postcode .....

Signed .....

Date .....

Send this coupon together with your cheque/postal order to
Software Sale, A&B Computing,
Number One Golden Square,
London W1R 3AB.

All prices include VAT and postage and packing.

Feedback

Clive Grace

The Reader Interface that listens to YOU

When opening the mail this month, I was amazed at the number of enquiries regarding machine code, assembly language and the other brands of languages. In fact there are only a few letters this month that do not deal with computer languages and they stick out like a sore thumb! Thanks for writing.

EQUALITY

I'll kick of this month with a letter from *John Rowe* from Devon who has come across a little problem regarding the EQUS command:

"About two months ago I decided to learn how to program in assembler language. Since then, with the help of the user guide, I have learned a great deal. However, I cannot understand how to print strings with the EQUS command incorperated in BASIC 2. Why was it never in BASIC 1?".

The EQUS command was implemented in BASIC 2 because there were a few niggling bugs (ahem! *features*) in BASIC 1. The Assembler was still a bit strange, funny things could happen (at least they did on my early Beeb, even though Acorn used to deny this when I confronted them) and most importantly, there were still some features that Acorn programmers wanted to see, so a new version came out.

To understand how EQUS works, it would be wise to look at the whole subset of the EQU words in the assembler. These words (EQUB, EQUW, EQUD and EQUS) are all used to reserve a special area in memory, this can be altered, but unless you want to write self-modifying code, it is best to let it be.

With the program in figure one, you can see the routine that is used by many people who want to remain compatible with BASIC 1 users, which employs string indirection (which is an idea taken almost directly from the Atom assembler). The example uses the EQUD word, but there is no reason why this cannot be replaced with EQUS, all that happens is that EQUD sets aside 4 bytes of memory (ie EQUate a double word). EQUS reserves the data storage space to hold a string. The really great thing is that you can use as much data as required because EQUS reads until the end of storage space (and can be over a page long).

PARLEZ VOUS LISP?

The next letter comes from a *B. Terry* from York, he has a number of languages to choose from for his studies in Artificial Intelligence.

"I am currently at University studying Artificial Intelligence, and ever since buying my BBC Micro, I have been toying with the idea of using a real AI language. At the moment I am working with a Prolog compiler which does the bulk of my work, although string manipulation is a good deal easier in LISP.

What I am writing for, is a rundown on what features Acornsoft LISP has. As I have heard that it is only a "toy" version of the language, with hardly any of the features of the PDP11/73 version that we use at university I'm unsure as to how useful I will find it".

Acornsoft LISP is indeed a small version of the language. There are a number of LISP standards in the world, but the one most suited, it seems, to working on a microprocessor is MAC-LISP (no, it isn't implemented on the Apple Macintosh!). It lacks the multi-user capabilities, the four terminal controlling drivers and the multi-file access and overlay files.

Acornsoft LISP is even smaller than most micro versions in that it is an interpreted language, suitable only for learning the basic command structure and for writing small programs.

There is also the fact that the LISP compiler is, usually, separate from the text editor. The editor is normally written in LISP, as it is usually good at handling strings and things, but for Acornsoft LISP, again, this is a no-go area.

How good is it? That is of course dependant on how good a programmer you are using LISP. I discussed it with a Senior Lecturer (who is not interested in AI but in program architecture) and owns a BBC Micro. He has nothing but praise for the language. "I often use it for teaching the early concepts of list processing, it is easy to see on the screen as it only works in the 40 column modes and I even use it for examples in my tutorials"

There is also the news that LISP is available on ROM, and can thus be used across the Tube and Shadow RAM, so there is plenty of room to write quite complicated pieces of LISP code. The ROM is a good deal better than the disc version (although I was only able to play with the LISP interpreter with an Acorn flunky singing its praises in my ear).

If you want to use LISP, and I haven't heard of any plans for a BBC PROLOG yet, go for the ROM version, as I am sure that you will need the extra facilities of the interpreter. If you have a 6502 second processor, you should not have any worries about the speed restrictions as the increase puts it well within the speeds of many mini compilers running in the afternoon at 4 O'Clock.

SPREADING THE WORD

Still on the subject of languages, *P Bowman* from Lancaster is thinking of moving to another language:

"After having a go at assembly language, I was quickly confused about how to program in machine code. It seems to me to be so much effort for so little result. Is there anything I can do about learning assembly language easily, like I did with BASIC.

I am thinking of writing in another computer language, perhaps BCPL, as it can create machine code programs with apparently no effort on my behalf.... I feel that too many books say that assembly language is easy and it isn't.

I bought the "Programming the 6502" on the recommendation you gave it in "Feedback" a few issues back. Since then I have been confused and spent several sleepless nights working on an exercise that was fated never to work."

It is a shame that you find assembly language so hard. I would like to think that everybody can program in machine code but, as you have stated, this is not the case.

For you, I would first look at the alternatives. BCPL is a good, if strange language to produce machine code from, but you have to buy the Compiler which is £99, not to mention the calculations package, which is another £30. The biggest setback is the Run-time Module, which will put you back £40. This gives you a licence to copy the Run-time module 100 times per package you write, you have to pay a lot more if you want to sell on a truly commercial level (1000 plus).

You may also find that the language you have bought is totally unsuited for some applications. BCPL is a general purpose language but it has a number of limitations, especially with list-processing and artificial intelligence.

It is a common fact that people have little knowledge of the way a computer handles a program, and that their only means of communicating with the computer is through BASIC keywords and *FX calls, but there is a great deal more to computing than BASIC and what is no more than a glorified POKE command.

Have a look at some books on assembly language to complement the Zax's book you bought (I did state that Zax's book lacks the substance of a teaching book, it is purely a reference guide). Ian Birnbaum's *Assembly Language Programming on the BBC Micro* is a beautiful piece of work. I know so

CONTINUED OVER

many people who were once tied down to BASIC and found the attention to detail in Birnbaum's book what everyone needs.

Do not forget that home computing with the BBC and Electron is also *Fun* , taking upon the challenge to learn what you consider "too difficult" is one of the most rewarding things a person can do if you can learn from the experience.

Computer programming is never easy, unless you want to write a **10 PRINT "HELLO":GOTO 10** program. I can assure you that I have thrown away that "Teach Yourself Assembler" book at least 100 times and walked back later, to pick it up and to solve that damn problem.... it is great fun and expands the mind, not only through personal knowledge, but through patience and dedication, something all of the best programmers have.

MATHEMAGIC!

Almost directly after opening the letter written by Mr Bowman, I came across this wonderful letter and a heartfelt plea from *Alan Bear* for information regarding arithmetic in assembly language for his Electron:

"My Mum and Dad bought me an Electron last month, and I am very pleased with it, I managed to play some games on it, as I got a special bundle of tapes with the computer. Programming is great fun as I write a lot of my computer studies homework on it, and I can load it onto the BBCs at school, unlike the others who use their Spectrums for playing games and nothing else.

I know BASIC already as my dad had an APPLE computer for his work and I am going into "O" Levels this term. I am learning 6502 assembly code (with the Electron) and I got a book which is really good. The big problem is that I cannot find anyone who can tell me what Two's Complement and Binary Coded Decimal is! I asked my Maths Teacher, but she said that it was too complicated for me to understand yet! But I can add up in Binary, which nobody else can do in the class (I think she doesn't know, I don't think she likes computers at all, especially as she is the only Maths Teacher who doesn't use the computer room, and even the games teachers play "Planetoids"!).

PS I think that your "Real Programmers Don't eat Pascal" is funny, although some bits I don't understand, keep it up".

Thanks for your letter Alan and welcome aboard. Your letter has taken pride of place on my pinboard for your comments. I think you must be right, If you can add up in Binary, then you are well on the way to learning how arithmetic is achieved in assembly language, and you never know, you could teach the Teacher about this one.

Two's complement is the way the 6502 normally carries out addition and subtraction operations. In order to understand how It works you must realise that the "most significant bit" (or MSB) is the "sign bit". If the MSB is clear (if it is a "0") then the number is positive, negative numbers are represented by the "complement" of the value plus a "1".

A "complement" is a number that is achieved by changing each bit from a "0" to a "1" or from a "1" to a "0". When a negative number is represented, then this is called "ones complement". This is not very good as there are two ways of showing a "0". You can show it as a positive "0" which means that all the bits are zero, or "clear" and a negative "0", all bits are "1's". By adding a "1" to the complemented value, there is only one way of representing "0", and that is as a "00000000".

Figure two has an example of how numbers are stored in this way, perhaps you would like to write a program to show this, that would really impress your Teacher wouldn't it? (as well as get her in the computer room to see how they can be put to good use). Good luck!

SHADOW LORD

I got this letter from Mr *M Attfield* from Forest Gate, London and he is obviously going to expand his Electron quite dramatically, judging by the questions he wants answered.

"Could you tell me how compatible is Viewstore for the BBC in relation to the Electron, can it be plugged into any of the other interfaces?

I am also getting a disc interface for my Electron but I cannot, as yet decide as to which one is the best. The Cumana looks good, but I will need a Plus One in order to plug it into the interface. I am also very interested in the Plus 3, but a number of factors made me re-think; (1) Acorn's still precarious future; (2) No Tape to Disc utility has yet emerged for transferring my commercial software; (3) The ADFS takes up just under 4K of RAM! so that most of my commercial programs will not run on the ADFS."

Ever since Olivetti threw a lifeline to Acorn for a major part of their shares, Acorn have been quite steady. True, they took a pretty strong blow, I think everybody was shocked by the news that Acorn very nearly went bust, but now they are a good deal steadier than some computer companies around.

BBC VIEWstore will no doubt work on the Electron with a Slogger ROM box, or a Micropulse box, although I cannot guarantee that it will work 100%.

I know many Electron users who have bought VIEW and VIEWsheet for the BBC and plugged them into their Sloggers, even COMAL works as well as LOGO and ISO-PASCAL. But I cannot guarantee that there is not some obscure bug in the Electron OS (either purposely or by accident) put there to disable some BBC software running on the Electron. The big problem will come with the function keys which have different values and require some pre-programming to allow for this discrepancy.

I am sure that if you brought your Electron into a reputable shop, with cash-in-hand for VIEWstore, then they would be only too obliging to try out the combination. After all, if you want to wait for over a year for the release of the software on Plus one cartridge, then fine!.

CLUBBING TOGETHER

User groups were at one time considered unfashionable and very unpopular, or so I thought, until the Sutton Library Computer Club came to my attention. They seem to have a great deal to offer and have some very interesting subjects to talk about. This month in the Sutton meetings, they are having (on the 1st of November) a talk about Prestel and Micronet, and (on the 19th) choosing a Micro. December has a section on Machine Code programming and "An Idiots Guide to the Computer".

Members are encouraged to bring their own micros to meetings, you can even have screen dumps and listings made of your programs (great for sending into A&B if you don't have a printer), this service is for the nominal cost of the paper.

The SLCC is a member of the ALCC (Association of London Computer Clubs) so there won't be any pirating or zapping aliens there, it will be run by enthusiasts and computer buffs, and looks to be great fun, informative and educational.

There are a number of membership levels to suit all ages, as well as a "Family Membership" and special rates for the unemployed and senior citizens so don't think you're too old to get into it!

If you want to write to the SLCC, and they do look to be a very nice bunch of people, then why not write to them (enclosing an S.A.E) to *The Sutton Library Computer Club, 21 Village Row, Sutton, Surrey, SM2 6JZ*. If you are a member of Prestel then why not send them a Mailbox, you can get more information on pages 81021245.

THE REAL PROGRAMMER AT PLAY

This month we look at the Real Programmer's life style, how does he manage to keep his air of weirdness? what does he live on and where does he go..... read on.

Generally, the Real Programmer plays the same way he works

—with computers. He is constantly amazed that his boss actually pays him to do exactly what he would be doing for fun anyway (although he is careful not to express this opinion out loud). Occasionally the Real Programmer does step out of the office for a breath of fresh air and a beer or two. Some tips for recognising the Real Programmer when he is away from the computer room: (1) At a party, the Real Programmers are the ones in the corner of the kitchen talking about Operating System security, and how to get around it.

(2) At a football game, the Real Programmer is the one comparing the plays against his simulations printed on 11" by 14" fanfold paper.

(3) At the beach, the Real Programmer is the one doodling machine code into the sand.

(4) At a funeral, the Real Programmer is the one saying "Poor George, And he almost had the sort routine working before the coronary."

(5) In a grocery store, the Real Programmer is the one who insists on running the cans past the laser checkout scanner himself, because he could never trust keypunch operators to get it right first time.

What is the Real Programmer's habitat, what sort of environment does the Real Programmer function best in? This is an important question for the managers of Real Programmer. Considering the amount of money it costs to keep one on the staff, it's best to put him (/her) in an environment where he (/she) can get his (/her) work done. The typical Real Programmer lives in front of a computer terminal. Surrounding this terminal are:

(1) Listings of all the programs the Real Programmer has ever worked on, piled in roughly chronological order on every flat surface in the office.

(2) Some half-dozen or so partly filled cups of cold coffee. Occasionally, there will be cigarette (and other) butts floating in the coffee. In some cases, the cups will contain orange crush.

(3) Unless he is very good, there will be copies of the OS JCL (Job Control Language) manual and the principles of operation open to some particularly interesting pages.

(4) Taped to the wall is a line-printer Snoopy calendar for the year 1969.

(5) Strewn about the floor are several wrappers for peanut butter filled cheese bars — the type that are made pre-stale at the bakery, so they can't get any worse while waiting for you in the vending machine.

(6) Hiding in the top left-hand drawer of the desk is a stash of double-stuff oreos, for special occasions.

(7) Underneath the oreos is a flow-charting template, left behind by the previous occupant of the office, and not used subsequently.

The Real Programmer is capable of working 30, 40, or even 50 hours at one sitting, under intense pressure, and in fact, he prefers it that way. Bad response time doesn't worry the Real Programmer, it gives him a chance to catch a little sleep between compile and test runs. If there is not enough schedule pressure on a Real Programmer, he tends to make things more challenging by working on some small, but interesting, part of the project for the first nine weeks, then finishes the rest in the last week in a couple of 50-hour stints. This not only impresses the hell out of his manager, who was despairing of ever getting the project out on time, but also creates a convenient excuse for not doing the documentation. In general:

(1) No Real Programmer works 9:00 to 5:00 (unless it is 9:00pm to 5:00am).

(2) Real Programmers don't wear ties. Real Programmers don't wear high-heel shoes.

(3) Real Programmers arrive at work in time for lunch.

(4) The Real Programmer may or may not know his wife's name. He always knows the entire ASCII (or EBCIDIC) code table by heart, however.

(5) Real Programmers don't know how to cook.

Next month, "The Real Programmer at Work"

ADDCOMM FAN

I got an irate letter from *Francis Disney* from Egypt (deepest Cairo) who did not agree with the review we gave to the Vine Micros ADDCOMM ROM:

"In ten years of my programming experience on machines ranging from the IBM 370 down through the range to the HP 9836, (the Beeb is my home computer) I have never heard of a more ignorant comment than the criticism of LGOTO. Has the man never written a program longer than 20 lines?, has he never had to rework other peoples programs? or perhaps the BBC Micro is the only machine he has ever used.

The POP commands seem to be popular (no pun intended) have a look at the "C" programming language and specifically the function of the break command, then write a scathing critique attacking the makers or writers of "C".

The SCALE command was mindlessly criticised for not altering the PLOT, MOVE and DRAW parameters which, he claims necessitates modifying pre-ADDCOMM programs to work on the SCALEd screen. On the contrary, if the program has taken into account a 1280 by 1024 screen you most definitely don't want SCALE to disturb PLOT,MOVE or DRAW. He does not point out that there is a GREPL command which does just what he was complaining that ADDCOMM did not have!

With regard to the CIRCLE and ELLIPSE commands, he criticises ADDCOMM for having too complicated a statement syntax, I fail to see how the statement can be represented in a more simple form.

Finally, to add insult to injury, the reviewer attacks ADDCOMM for not having procedure building routines for his implementation of LOGO, I failed to see any claim to implement LOGO on ADDCOMM. LOGOGRAPHICS yes, but not LOGO, nor FORTH, nor FORTRAN nor COBOL, it surprises me he didn't criticise it for these absences".

In many cases I agree with you Mr Disney, but the fact is that with extensions to BASIC, you either like them or you don't, everybody is entitled to their own opinion, but in a review the facts have got to be correct.

I use ADDCOM and find it very useful and friendly, I class it amongst my ROMs as a damn good bit of software but clearly some people don't. Most people know how to knock out ROMs in their BBC's as the commands do tend to find themselves duplicated and this can be set up in a !BOOT disc or even in the ADDCOMMed program.

In spite of our reviewer's comments, I went out and tried the ROM out for myself for over an hour. I was impressed by the quality of the implementation of most of the commands (although I resort to machine code when POPping things on and off the stack). For the record I am an ADDCOMM fan.

The reviewer's attack was more concerned with the use of PROC, (why use LGOTO when PROC exists?) but I do agree with you that a LABEL definition is an important one, especially since for 99% of commercial BASIC programmers, LGOTO is a widely used feature.

Thank you for your letter Mr Disney, it proves that "Feedback" is here for people like you. If any readers disagree with anything we say, it is up to you to write in and tell us, we are only human.... I think.

Until next month, have a good time. If you see anything or hear any interesting news or simply have a problem with the ol' Acorn computer, why not write to me at **Feedback, A&B Computing, 1 Golden Square, London W1R 3AB**. Have Fun!

Plotting Success

Gordon Taylor

Easy creation of business graphics with Inter-Chart and precision plotting by Plotmate makes for a winning team.

Following general reviews of Inter-Chart in the November issue, and of Plotmate in the September issue, this article looks at their performance together. In a word, it is outstanding — as the illustrations show.

The reader is referred to the review of Inter-Chart in the November issue for a detailed discussion of inputting data, saving it, loading it, and transferring it both via files and by using the in-memory ROM-link "Inter-" system, and outputting via a graphics printer.

This review concentrates on output from Inter-Chart by plotter, specifically the Linear Graphics "Plotmate", which costs £299 plus VAT. This is a flat bed plotter of "A4" size (but capable of producing composite plots up to A2 in size). It is designed for use especially (but not only) with the BBC Micro. To connect to other computers — as well as to enhance its abilities with the Beeb — an "intelligent" interface, containing both ROM and RAM, and costing £169.50 plus VAT, will soon be available.

INSTALLATION

Inter-Chart comes as a Sideways ROM, for plugging into your Beeb or, more likely by now, your ROM or RAM expansion board.

The Plotmate software is currently on disc, which at least avoids having to find another ROM socket. However, a ROM version will be available from November. It will cost only £15, plus VAT, both for registered Plotmate owners and with new purchases.

The Plotmate plotter is connected to the computer just like a printer, and is designed to use a parallel port, so avoiding all problems of serial connection. It is normally connected to the User Port and leaves the parallel port free for your printer. However, it can also use the parallel Printer Port, by means of a different lead and system disc, which are available for £14.50 plus VAT.

COMPATIBILITY

Inter-Chart is already confirmed as compatible with the cassette filing system, the standard Disc Filing System, the Advanced DFS and the Network Filing System/Econet, and on the standard Beeb, the Beeb with add-on Shadow RAM boards (Aries and Watford) and the B Plus. (See the November issue).

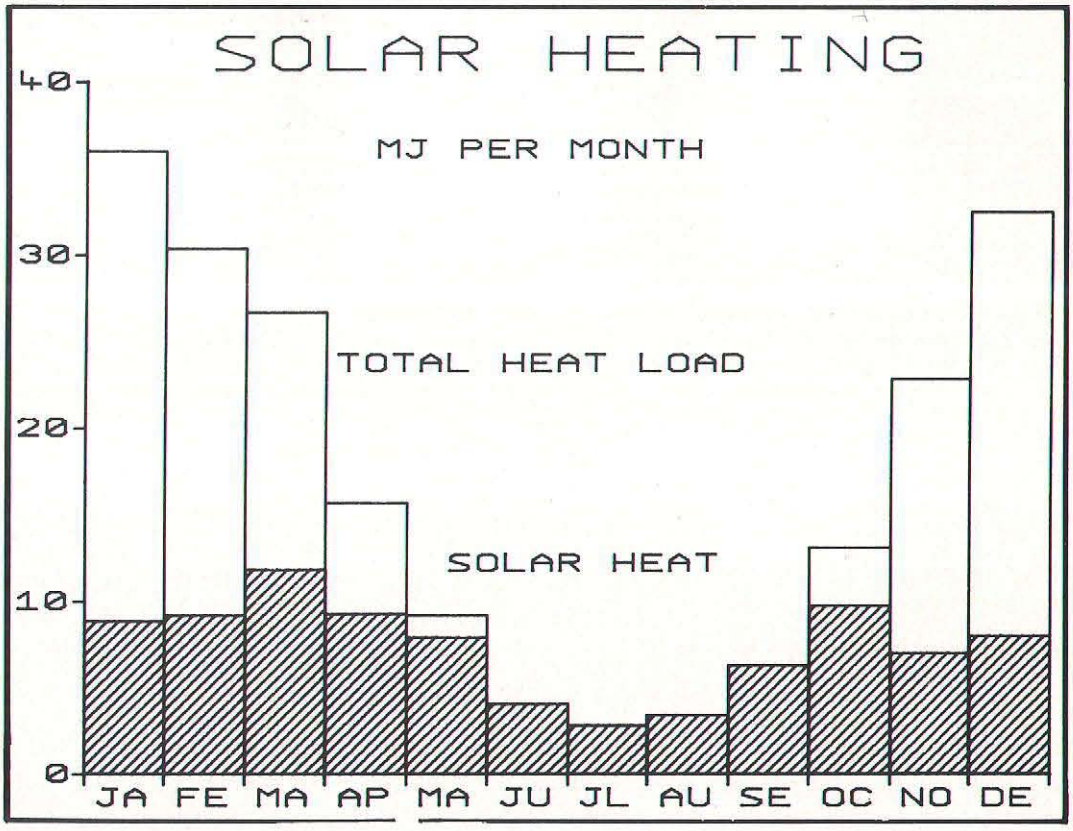
Plotmate is hereby confirmed as compatible with the DFS, ADFS, and NFS/Econet. At present the software is on disc but the ROM version will make it practical to use Plotmate also with cassettes.

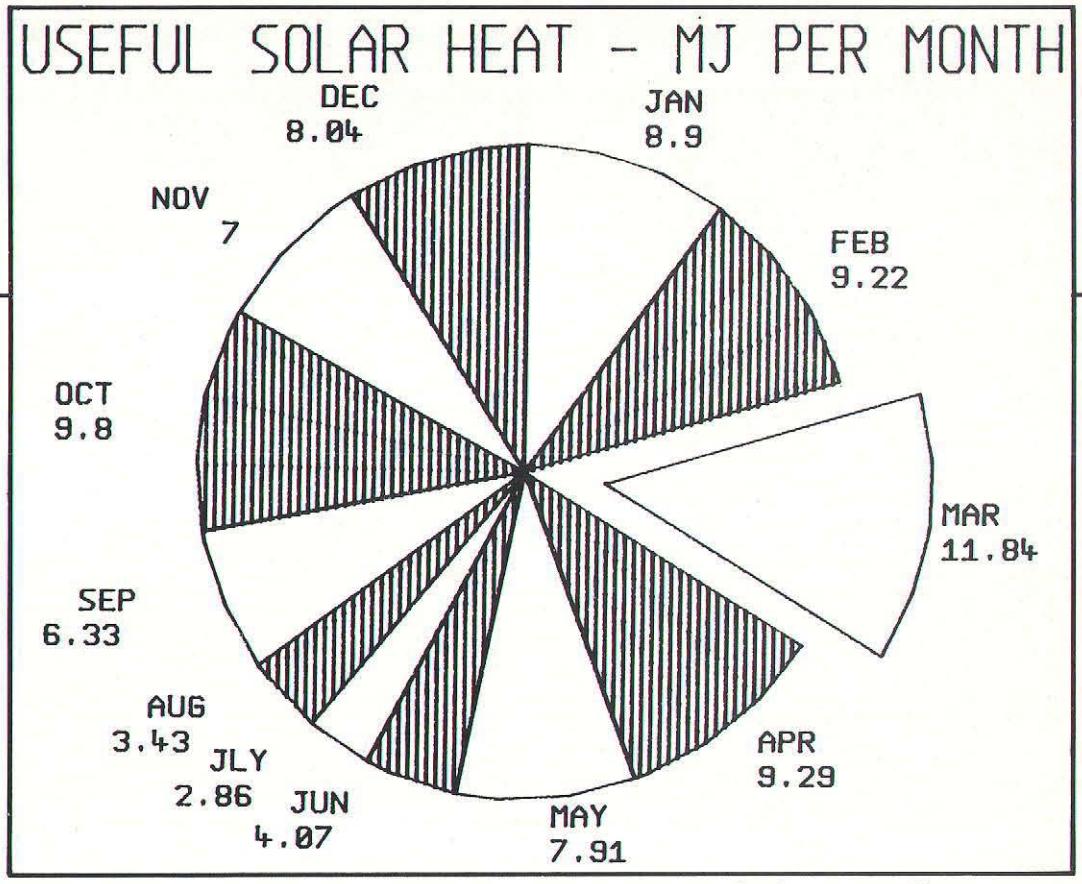
Equally important, Inter-Chart and Plotmate are compatible with each other. Thus far, this applies to the Low version of the Plotmate software, which resides immediately below the Mode 4 graphics screen, from &3400 to &5800. It is expected that the High version, which resides between &5800 and &7C00, just below the Mode 7 screen, will also soon be compatible with Inter-Chart. This disc-based software is expected to continue to be of interest, even after the ROM version becomes available, since it can be downloaded over networks, such as Econet, even to machines not fitted with Sideways RAM.

The Plotmate software is compatible with the B Plus, and hence probably with add-on Shadow RAM boards, such as the Aries B20 (and 32 — see next issue) and the Watford 32. It is also compatible with the 6502 Second (or Co-) Processor. In this case, the Plotmate software still resides in the Beeb itself, "languages", user programs, and data.

Although normally connected to the User Port, Plotmate is compatible with the Solidisk Sideways System, which also uses some pins of this port. The reason is that the Sideways System uses them only briefly for identifying RAM banks during loading and saving operations. This does not affect their use by Plotmate (or other devices) in normal operation.

However, the Plotmate plotter may also be connected to the computer via the parallel printer port. This then leaves the User Port free for any of several input devices, such as the Marconi RB2 Trackball, the AMX or Wigmore House Mouse, or the British Micro Grafpad. This situation can arise when you require both input device and plotter in operation at the same time — as when using Plotmate to digitise e.g. maps or drawings — or





in close succession, as with CAD operations.

GRAPHIC VALUE

Bar and pie charts — like Figs 1 and 2 — are of proven value in business. However, graphical presentation not only shows up anomalies, it even allows you to distinguish between them. Thus in Fig 3, since the right-hand peak has a matching trough, this is very probably due to incomplete filling of the tank. However, the left-hand peak has no corresponding trough, which suggests that one "fill-up" has not been recorded. The difference between the two anomalies would be much harder to spot in a table of figures. Fig 4 in turn shows that the electricity consumption of refrigerators of given size can vary by a factor of two.

PLOTTER PLUSSES

Plotters have advantages over e.g. graphics printers for all three main application areas — business graphics (reviewed here), artists graphics (see the September review) and engineering drawing/CAD, which will be covered in a future article.

A plotter — at least in conjunction with a computer of far-sighted design like the Beeb — has a fundemental advantage over any printer. It can work to the full internal resolution and colours of the computer, whereas the printer is always limited to the actual resolution of the screen.

The Beeb is unusually well-endowed as far as the screen resolution and colours are concerned, with up to 640 x 256 x 2 colours, requiring 20K of screen RAM. However, it supports 1280 x 1024 x 8 colours internally, which would require 640K of screen RAM, and also be very expensive. It would also need a much more powerful processor — faster, to give adequate screen writing speed, and of at least 16-bits to "address" so much RAM.

Hence the relative accuracy of a plotter (on the Beeb, at least) is given by the resolution of its "virtual" screen, which is 1 in 1024 vertically. The actual screen (from which graphics printers take their data) has only 256 lines, only one quarter as good.

Furthermore, lines at any angle appear perfectly smooth. This is because both the X and Y stepper motors are driven together. Of course, the absolute accuracy is still finite, but this is 0.1 mm or 4 thousandths of an inch for the Plotmate. This may be compared with that of a standard dot-matrix printer, which (in the "plotter graphics" mode) gives 576 dots over 8 inches — i.e. 14 thousandths of an inch, only one third as good.

When using colours, a colour printer is constrained to produce just the two, four or eight colours that can be put on the Beeb screen (setting aside flashing or "mixed" colours), whereas a plotter can use any number of colours. Any plotting program for the Beeb can use up to eight colours (whatever the screen mode used), and Plotmate could plot them all. Moreover, colours on a plotter are produced at a single pass, and not by multi-pass mixing.

Hence even the best colour graphics printers are limited by the screen modes of the computers they are linked with. Also (as a result), they are typically little if any higher in resolution than the screens. For example, the Epson JX-80 gives 576 dots across 8 inches in the "Plotter Graphics" mode, and the Canon PJ 1080A gives 640 dots across 8 inches. These are half or less that of the Beeb internally, and hence of Plotmate.

Another advantage of a plotter over a graphics printer is the ability to vary the width of the plotted line — especially below that of one screen pixel. For fibre tipped plotter pens, the range of widths is 0.3 to 0.7 mm. Since the pen width has to relate to the character size to ensure legibility, 0.3 mm is used for small — e.g. A6 — plot sizes, while up to 0.7 mm is used for larger plots, and for shading and solid fills.

Other advantages of a plotter include :

- the ability to change the size or scale of the output freely (without changing the data supplied).
- the ability to produce continuous smooth lines, like those of a good draftsman. (Any limitations are likely to be due to the application and driving software, as opposed to the plotting hardware).
- the ability to produce both cross-hatching patterns (for differentiation in black and white) and multiple colours as required.
- the ability to incorporate legends and titles in a wide variety of styles and sizes etc.
- the ability to include special characters which may be provided by the plotter software.
- the ability to produce output on transparent film for use in overhead projectors. (This is not possible for example with impact printers, but only those using ink jet or laser techniques).

Finally, the Plotmate costs significantly less than e.g. colour printers such as the Epson JX-80 (impact dot-matrix) or the Canon PJ-1080A (Ink Jet), which are only capable of producing replicas of the screen image, implying a much more limited resolution and range of characters, sizes and styles.

THE PROOF

Figs 1 and 2 have been chosen deliberately for comparison with those in the review of Inter-Chart in the November issue, which showed screen dumps produced by a dot-matrix printer (an Epson FX-80) in graphics mode. Comparisons of the line graphs (Figs 3 and 4 here) show the plotted output to even greater advantage. Only a plotter can produce continuous straight lines at any angle, free from jagged steps.

The reason why, in Inter-Chart, the pie chart is slightly polygonal, is that it was originally coded only to produce a screen image, (where this is not apparent). While a more-sided polygon could be adopted, this would require more triangles to be used when shading or filling, and thus take longer to plot. More-

CONTINUED OVER

over, the discontinuities of shading within pie segments are also a function of Inter-Chart, not of Plotmate.

This is proved by the fact that such discontinuities can be avoided by suitable programming e.g. in BASIC. (see the September review).

The Figures also show the ability of the plotter to produce characters in different styles — e.g. enlarged for headings (as on the bar chart, Fig 1 and the pie chart, Fig 2) or rotated sideways for labelling the Y-axis (as in Fig 3).

THE PLOTMATE SOFTWARE

This comes in two alternative versions — actually the same 9K of code "assembled" into two different locations. The "Low" version sits just below the Mode 4 screen area, which allows all output to the plotter to be echoed on the screen, but this reduces the space available for either the user's program e.g. in BASIC or for "packages" in the "Inter-" family — such as graphs in Inter-Chart. The test is, if you have enough room to be able to edit or display in Mode 0, and you then switch to Mode 4, you always have enough space for the Plotmate software. However, plots of even Mode 0 screen images are possible with this software, on machines with Shadow RAM.

The "High" version sits just below the Mode 7 screen area, which — since it is not a graphics mode — allows only text to be echoed on the screen, but neither does it reduce the space for user's programs or "packages" at all (for a standard Beeb). It still reduces the (initially greater) space in a Beeb with Shadow RAM, and runs with BASIC, but not yet with Inter-Chart.

The ROM version will leave the maximum space e.g. for charts in Inter-Chart and programs and data in BASIC, as well as allowing plots of Mode 0 screen images, even with a standard Beeb.

The reason why Plotmate is so versatile, is that it intercepts the OSWRCH vector, and routes it to the plotter. This works provided that the software is "legal", and does not write directly to the screen. In addition, the Plotmate software also recognises all the VDU statements, and responds to several of them. For example, after being suitably enabled, it will prompt the user to change the pen for one of the specified colour. Thus exactly the same Inter-Chart graph or BASIC program may be used to produce plots :

- in black and white only on paper — e.g. for reproduction by photocopying.- in multiple colours on paper — e.g. for use in a report for clients or top management.
- in multiple colours on film — e.g. for presentations using overhead projection.

Earlier versions of the Plotmate software set up the function keys so that the various functions could be called from BASIC and Logo, amongst others. A revised version has been developed which extends this facility to any "language", including Inter-Chart. After the Plotmate software has been loaded into RAM, the functions available on the red keys are :- Enable plotting, disable plotting, enable cursor keys to drive pen, move pen to "home" position (bottom left), lift pen, lower pen, and move pen to "park" position (bottom right).

Since Inter-Chart selects Mode 0 by default for both Edit and Display Modes, and the Plotmate software resides within the Mode 0 screen area of RAM, it needs to be re-loaded from disc quite often. This too may be done by using a function key, f7.

All these functions are available from the Edit and Display modes of Inter-Chart, by pressing Shift-Control as well.

MANUAL AND DEMONSTRATIONS

The manual is in a PVC-bound ring binder, in the same cyan colour as used on the Plotmate itself. The manual has separate numbered sections, doubtless "to allow for future expansion", and a contents list, but — as usual with this format — no index. It includes several short listings to illustrate the full facilities of Plotmate and its software, but as yet, there is no quick reference section (or any separate card). However, the system disc contains several excellent demonstration programs, written in BASIC, which may be listed out, as well as plotted.

PLOTMATE IN USE

Producing presentation quality business graphics is remarkably simple, when using Inter-Chart for creating the chart or graph, and Plotmate for output, especially if you accept the default values. Making use of more of the versatility of Inter-Chart and Plotmate naturally requires more study and effort in the first instance. However, all the operations required can be embodied in an EXEC file for repeated use.

The first step in producing a plot from within Inter-Chart is to create on-screen what (or part of what) you want on paper.

Figs 1 to 4 show what is possible, including the ability of Inter-Chart to produce graphs with logarithmic scales. This is done quite automatically, so it is simple to view the data against both linear and log scales in turn, to see which suits it better. One of the most powerful features of Inter-Chart is the ease with which headings may be editted for best effect, whether on single or even on multiple, overlaid, graphs. The default character size is very close to Mode 4, so editing on-screen gives a good idea of what will come out on paper.

With the plotter switched on and its data lead connected, the software may be loaded into the Beeb simply by pressing Shift-Break, after which plotting should be disabled temporarily with f1. Pens are fitted by twisting into place and the paper is held by flexible magnetic strips. Plotting is best initiated from Display Mode, which allows you to reset the axis ranges from their default values, if required. Furthermore, it is necessary, with the present disc-based, RAM-resident software, to select Mode 4, in order to leave room for the "Low" version. However, this is less of a limitation than it may appear, since the plotter still works with the full 1280 x 1024 internal resolution and all 8 colours if required, and uses its own characters.

The only disadvantage is that any loss of characters (by truncation) from the axis labels shown on the screen cannot be restored by the plotter. With the forthcoming ROM-based Plotmate software, plotting will be possible from any graphics screen mode. Meanwhile, if the disc-based software has been overwritten, it may be re-loaded with Shift-Ctrl-f7.

The secret of understanding how Plotmate works is to realise that it plots on paper exactly as the electron beam writes on the screen, but very much slower. Thus, to initiate plotting, it is necessary to cause the screen to be re-written. This may be done in several ways, without altering the image on the screen, as many of the keyboard commands in Inter-Chart have "no-change" options. Perhaps the most appropriate is f1 — "Select Mode", which returns "Which Mode ?", whereupon you should answer "4".

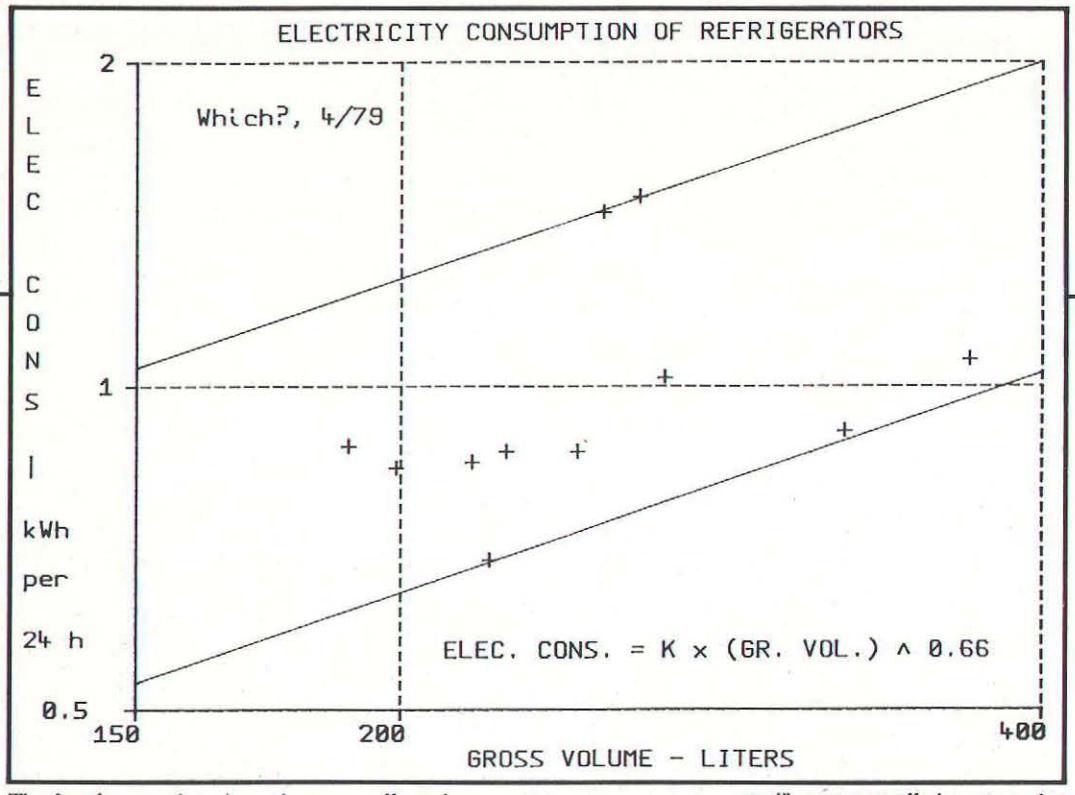
Plotting will then take place, with the image being echoed simultaneously to the screen. Afterwards, it is wise to disable the plotter — with Shift-Ctrl-f2 — to prevent further screen writing being added to the plot.

That is all there is to it, provided you are happy with the default settings of Plotmate.

FANCY PLOTTING

Use of other than the default size and shape of characters — amongst many other things — is enabled by a VDU 23,255 statement, either from within Inter-Chart or from a BASIC program. VDU 23 is normally used to re-program displayed characters, so this choice is quite logical. The second term implies that character 255 is being re-defined, and should not therefore be re-defined by the user's program.

The usage by Plotmate of the VDU 23, n statement is consistent with that of the Beeb, in that it is followed by eight "arguments" or parameters. These are used to define :- Height, Width, Slant, Orientation, Pen Speed, Special Functions, Shading and Scale.



The first four apply only to characters, while Shading applies to both lines (the type of dotted lines) and triangular and rectangular fills (the density of shading or hatching). The fifth argument controls the Pen Speed, but only during actual line drawing operations. The pen always moves at top speed when off the paper and when doing shading.

The sixth argument of the VDU 23,255 statement – the Special Function – may have several values.

When seeking to reset this statement to other than the default values, you must set the Special Function argument to something other than 7. For simple changes to character size etc., SF should be set to 0.

Putting SF=1 allows you to re-origin the plotter – e.g. by enabling the cursor keys, and driving the pen (up) to a pencil mark. This enables multiple plots to be placed alongside each other.

If you enable plotting in multiple colours with SF=4, then any colour changes will be signalled on the screen, with messages in Mode 7. In this case, echoing to the screen will cease, since Mode 7 is not capable of high-resolution graphics.

The seventh argument – the Shading parameter – only qualifies the existing dotted line type; it cannot change a solid line to a dotted one. The reason is simply that the effect is global, and if one line were so changed, all the lines of that graph – including even those in the characters – would be affected.

Both Inter-Chart and Plotmate have facilities designed to help differentiate pie segments and bars in business graphics (and areas generally for Plotmate) by pattern (which shows in black and white), as well as by colour. It turns out that there is no conflict, with Plotmate rendering very well the diagonal shading generated by Inter-Chart. It can be set within Inter-Chart, using Menu option 6, by using Plot Option 1 and a Colour option such as 17. (see Fig 1) Shading of other densities and angles may be achieved with other values of the Colour option.

Alternatively, the Shading parameter within the VDU 23,255 statement in Plotmate may be enabled with SF=8 and then applies wherever triangular and (rectangular) block filling is invoked. Thus it may apply in the segments of pie charts, and in bar charts created using Inter-Chart. With plotters, the least possible shading should be used, since high densities are very time-consuming. In the case of Plotmate, solid filling means drawing lines every 0.1 mm. However, less dense shading is also available.

The Plotmate software has an internal numeric range (or virtual sheet) of +/- 16383 in both X and Y directions. Beyond this, it "wraps around", much like the Beeb screen. This allows it to handle data which is outside the current plotting "window". For example, a partial (or incorrect) plot may leave the A4 page at one point, and return at another, having "visited" a point off the page, but within this virtual sheet. This is known as "clipping" the plotting data, and applies to MOVE and DRAW operations, to triangular and block fills, and to characters.

What the X and Y values represent on the plotter is determined by the eighth argument in the VDU 23,255 statement – the Scaling Factor. With the default value of SC = 120, it will plot X = 0 to 1522 and Y = 0 to 1039 on an A4 sheet. This may be compared with the screen of the Beeb, which shows X = 0 to 1279 and Y = 0 to 1023, expressed in the internal units. Thus this scale is suitable for showing the whole screen image, with a margin around it, and is taken as "full size". Correspondingly, at a reduced scale, such as SC = 64, Plotmate will plot X = 0 to 2854 and Y = 0 to 1948 on an A4 sheet. The scale range is from SC = 1 to 255.

The use of a global scaling parameter means that the overall appearance of a plot can be checked at a small scale, which uses less paper and ink, and can be drawn faster. The final plot can then be done at any other size, while retaining all the proportions, by changing this single parameter.

Plots at reduced scale also allow more than one graph to be placed adjacent on a single A4 sheet. Both the revised origin (for all but the first) and the corresponding scale (smaller than the default of 120) may be set together in a single VDU 23,255 statement. No change to the actual data for plotting is required.

Plotmate plots everything it "sees" on the screen, including graphs overlaid by Inter-Chart. However, the overlays are "transparent", and so you may see details of the graph beneath on the plotter, whereas they are overwritten on the screen. This is especially likely if you are using shaded fills on the plotter to represent solid fills on the screen. Put another way, the plotter cannot erase the image on the paper, while the electron beam can re-write that on the screen.

In practice, this minor (inherent) limitation can usually be overcome – as in the overlaid bar charts in Fig 1. Instead of plotting both graphs at once, they were plotted separately. This avoided the offsetting to left and right that Inter-Chart normally provides (see the Figure in the November issue), and hence no unwanted details of the bottom graph are visible.

When plotting graphs separately on top of each other, you should ensure that the axis ranges are the same for all the "overlaid" graphs. Also you should select graduation lines on only one graph (if any) to reduce the amount of redundant plotting.

ENHANCED PLOTTING

For still greater flexibility when using Plotmate, it may be necessary to resort to BASIC programming (as opposed to using only Inter-Chart). This need not be at all daunting, even to business people, since relevant examples are given in the Plotmate manual. Also, it is always possible to achieve a plot without BASIC programming, or even using the VDU 23,255 statement to re-set anything. These are simply available as options, to improve further the appearance of plots – e.g. for special presentations.

You can always check your BASIC programming very simply by entering MOVE x,y directly from the keyboard. This allows you to check your starting pen position without marking the paper. Another very useful trick when adding to an important graph with either another Inter-

CONTINUED OVER

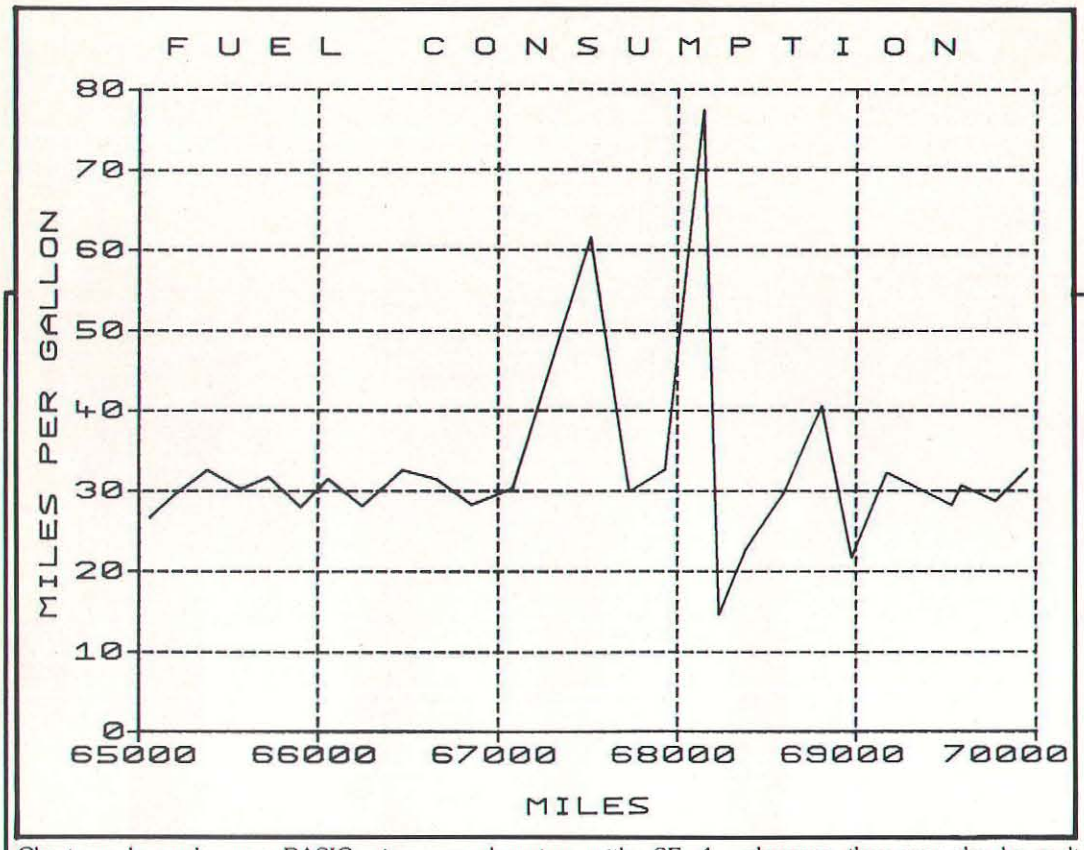


Chart graph overlay or a BASIC program e.g. to plot larger, sideways or special characters, is to cover it with thin scrap paper (fanfold is ideal). This should be secured outside the plot area – e.g. by "Bluetack" – and can then accept your test run, and allow you to check it before the final plot.

The accompanying figures give some idea of the possibilities which may be achieved both by the VDU 23,255 statement alone, and by this in combination with short BASIC programs. Thus Fig 1 used default settings save for the heading, which was produced using the VDU 23,255 statement, and a special overlay, while for Fig 2 the general text has been reduced in width, using the VDU 23,255 statement, and the main heading was produced using a short BASIC program. Fig 3 was also produced using default settings, save for the name of the Y axis, which again required a short BASIC program and Fig 4 was produced with a VDU 23,255 statement to reduce the width of all the text (and so fit it in the space available). This Figure, like Fig 1, is an example of combining several graphs on one plot, just as is possible on-screen within Inter-Chart. Thus the data points and the two boundary lines are from three separate graphs.

In fact, all these special headings could be produced from within Inter-Chart, using the special overlay technique. This involves enabling pen-changing with SF=4, since this results in pauses and prompts at several junctures – notably between drawing the graph proper, and adding the headings. By creating a minimal overlay graph to carry the heading and to be plotted separately, entering an appropriate VDU 23,255 statement, and only fitting a pen for the heading phase, the desired result may be achieved – as in Fig 1.

SPECIAL CHARACTERS

The ultimate example of the enhancements of business graphics produced with Inter-Chart, possible by BASIC programming, when plotted with Plotmate, is the extended character set available. (See Fig 5). The numbering shown is in hexadecimal, but an equivalent table can be produced in decimal. The special characters in the lower half of the table are not directly printable – either by the Beeb on-screen or by any dot-matrix (much less a daisy-wheel) printer without special setting up. However, they may be printed on Plotmate from a BASIC program by spelling them out individually as CHR$(m), CHR$(n) etc. – where m and n are in either decimal or (prefixed by "&", as usual on the Beeb) in hexadecimal. This is consistent with most standard BASICs, but with typical Acorn elegance, they may also be spelt out more shortly as VDU m, n etc.

REPEATED PLOTTING

If you are using Plotmate regularly, you will soon learn the few commands needed for plotting business graphics, and even the syntax of the all-powerful VDU 23,255 statement. Furthermore, for those who are less familiar, EXEC files may easily be set up to control Inter-Sheet, Inter-Chart, and any necessary BASIC programs, which can together take in raw data and deliver a finished plot of presentation quality. (I hope to deal with this in a future article).

The power of EXEC files (which are only possible with disc systems) is that they can call any "language", such as BASIC or Inter-Chart, and run a program or "package" in it. This is possible, without the EXEC file itself being overwritten, since it is read character-by-character from the disc into the keyboard buffer, as though input by a "robot" user.

SPEEDY PLOTTING

Considering the much higher effective resolution, Plotmate is remarkably fast compared with a graphics printer screen dump. This is because a plotter needs only to output the information in an image (e.g. the outline or the graph), whereas a graphics printer dumps the whole screen-full, (and with multiple needle blows per pixel, which effectively increases the time taken).

The main graph of Fig 1 was plotted in only 1 minute, 45 seconds, while the shaded bars took an additional 5 minutes 20 seconds. Even so, a total time of some seven minutes is far less than most users could achieve unaided, especially for such a professional result. Fig 3 was plotted, complete with all the graduation lines and the two headings in just five minutes while plotting Fig 4 took only about four minutes.

LOW COST PLOTTING

All the Figures were plotted on ordinary bond paper, with pens using water-based ink, and the default (highest) pen speed setting. They are thus representative of "good commercial practice". An even more polished appearance can be achieved – e.g. for exhibition displays – by using glossy paper, oil or spirit-based ink, and the slowest pen speed setting.

Amongst the consumables, only the cost of pens is significant – and that not very. They cost around £7.50 for a pack of six.

CONCLUSION

Inter-Chart and Plotmate are even more effective together than they are separately. Any business or academic department with a significant requirement for presentation graphics should invest in such a team. After all, both could easily be afforded by spending only a fraction of what you saved by not buying an IBM!

Inter-Chart is produced by Computer Concepts and is available now from computer dealers. Their address is Gaddesden Place, Hemel Hempstead, Herts. HP2 6EX. Tel: (0442) 63833.

Plotmate and its accessories are available now from computer dealers, and direct from Linear Graphics Ltd. They are at 28 Purdeys Way, Rochford, Essex, SS4 1NE. Tel: (0702) 541664.

Bookshelf

BBC Micro and Electron Book by Gavin Haines, published by McGraw Hill and priced at £11.95 is a standard introduction to the BBC and Electron. Opening up with the "Babbage to BBC" and moving on to a rather mixed bag of material. The sort of hints and tips found in the magazines are combined with sound reference material, especially the breakdown of memory useage.

Unfortunately there's some confusion about filing systems "AFS for use with second processors" — what's that? There's also a bit of a dated feel with no knowledge of the Electron Plus 3.

Still it's an easy read and quite entertaining with Mr Gradgrind (Nickolas Nickelby) and Socrates dragged in at one point.

The book rather lacks the depth that a specialist book can provide and only adds to a pile of "general introductions", none of them able to be comprehensive or up to date.

The cursory notes on disc filing and Econet and an assembly language section which promises well but has to break off just as it gets interesting are cases in point.

Computer Spreadsheets by Nigel Day and Owen Rees, published by BBC Publications for £6.95 is a great book for the uninitiated spreadsheet user. Their power is tempting the BBC businessman but you need a helping hand to get going.

This book throughout sticks to its promise "to play the part of a freind who has used spreadsheets helping you to find all about them" /The early chapters are very helpful and easy to understand. A variety of examples from Ultracalc are displayed. Visicalc and Lotus 123 provide interesting comparisons for the BBC Micro owner using Ultracalc, Intersheet or Viewsheet. He certainly won't feel shortchanged by the facilities offered by the software.

Progress is logical, from calculator type work to formatting screen display, labelling and decimal places. Saving, printing and "templating" are explained before proceding to "What if?" calculations.

Commands, functions and the order of re-evaluation are considered and the usefulness of spreadsheets in loan analysis and trial budgets discussed. Spreadsheets can get unweildy and in "Bigger Models" the book deals with editing features, inserting, moving and copying. At this stage of the book full value is gained by actually practising with the examples on your spreadsheet. They are realistic examples, the sort of thing you would perhaps be devising yourself anyway.

IF...THEN constructs within a model and the use of Ultracalc's "lookup" function are two very important sections, dealt with inadequately in the manuals and perhaps overlooked by many users.

A "worked example" provides a games league model for arcade game players, proving that spreadsheets can be handy for all sorts of application, not just financial.

Appendix 9, a complete design for a horse show (dressage) schedule confirms the flexibility of spreadsheets once again, sitting alongside other appendices on Household Budget, Invoice with Price List, Telephone Directory, Loan Repayment Chart, Expenses Claim Form, Cricket Analysis and Car Expenses.

The book finishes with a look at possible *advanced facilities* and *choosing a spreadsheet program* . An excellent finish to a remarkably useful book. Ultracalc users will get most from it but other spreadsheet owners can easily tailor examples to their own software. Great value if you are *hooked on spreadsheets.*

Practical Hardware Projects by Joe Telford, published by Century Communications. Price: £8.95.Recommended reading for anyone about to embark on the tricky subject o interfacing and hardware project building. It's a big leap from software to hardware. You have get your hands dirty, so to speak. A whole new set of skills have to be learned and this book is less likely to baffle you than any yet published for the BBC Micro.

It assumes no provious knowledge of the subject, repeats information and tips where necessary and uses clear diagrams. The descriptions of equipment leaves no doubt about what you should ask for at your local electronics store.

Interfacing concentrates on the analogue, user and cassette (motor control) ports. Starting with the make-up of suitable connectors, each section, on analogue and on user port, details a number of interesting projects. For analogue port: voltage, ohm, capacitance and amp meters; measurement of heat and light, a touch switch and a paddles/joysticks section. For user port: a coinboard, windspeed meter, piano type keyboard, output buffer, LED indicator, traffic lights simulator and relay board; a combination lock, movement detector and accelerometer.

There's a section on controlling motors, which leads into Beebuggy, its control and monitoring feedback from it. this involves following a white line, an obstacle detector and gradient detector and 'Drop' detector.

The final chapter presents three "upgrades" or modifications for the BBC Micro itself. The hardware disabling of the Break key has and will find applications in schools and where the BBC is used for display purposes. Fitting a DIL switch to the keyboard links enables a choice of screen mode at start-up, auto-boot on Break and selection of disc drive speeds. If you've got shadow RAM or a second processor why not show off and start up in the Mode of your choice?

The last of the three projects is a simple but effective software protection device and will spark off many further ideas. An analogue port dongle doesn't stop those who wish to hack the code but it's a nice project to demonstrate the concept. A list of suppliers and an external power supply are the standard appendices which complete the book.

Because the projects are based on the author's practical experience, they should prove ideal for the classroom with one team debugging the software, another laying out component on Veroboard , coming together to test and then use the resulting device. The hobbyist breaking new ground will also find a friend in this well produced book.

The Programmers' Trouble Shooting Guide for BBC and Electron by Piers Letcher, published by Century, price £7.95 is a "versioned" book, available in similar form for various machines. Although the author claims that problems particular to writing software in BBC BASIC are dealt with, the text doesn't really bear him out.

Indeed the book rather strays from the area indicated by the title and becomes yet another rewrite of the User Guide, a general introduction taking the reader nowhere special. There are certainly a number of peculiarites about programming in BBC BASIC and lots of interesting facts about the implementation of the language, but they are not in this book. They are to be found in numerous specialist magazine articles every month.

The enthusiast will learn nothing from this book, the beginner will regret buying it when he compares some older but better introductions.

Cheaper RAM

Mike Harrison

Ultra cheap sideways RAM.

Sideways RAM on the BBC can be implemented in one of two ways: You can fit two RAM chips into a ROM board, if you have one, or you can buy a RAM-only unit. The first is fine if you already have a ROM board, as the chips are only about £5 each (Sept.). This article shows how these same chips can be used without a ROM board, i.e. in place of a RAM only board. The total cost is about £10 for the RAM chips, plus about 50p for the other chip used.

This cheap solution gives the same facilities as most ROM board based RAM, i.e. Auto-write (any write to memory in the ROM area goes to RAM, so data can be *LOADed into it directly, and machine code (e.g.ROM software) can be assembled directly to the RAM.), and the option to put on a 'Write protect' switch, to protect data stored in the RAM.

The key to the low cost, and ease of construction is the way that it is built: by 'piggy-backing' the two RAM chips, and plugging the bottom one into one of the BBC's ROM sockets.

GETTING TECHNICAL

Each of the RAM chips can hold 8K bytes of data, so two are needed to fill the 16K ROM area. External circuitry is needed to control which RAM chip is active, and to handle the writing of data to the RAM (and disabling of other ROMs when the RAM is written to). These functions are performed by a 74LS158 chip, which is a quad 2-input inverting multiplexer (don't worry if you don't know what that means, this isn't a tutorial on logic design, so just take it from me that it works!).

The circuit is shown in fig.1. IC3a selects which of the 2 RAM chips is accessed. IC1 is active from &A000 to &BFFF, and IC2 from &8000 to &9FFF. IC3b,c determine when the RAM is active – when the 'Chip Select' line from the socket is low, or the Read/Write line is low AND the 'Area Select' line from S21 is low. IC3d disables the ROMs when a Write occurs. The 'Read/Write' line is taken from pin 1 of IC72, as this is right next to the ROM.

Fig.1

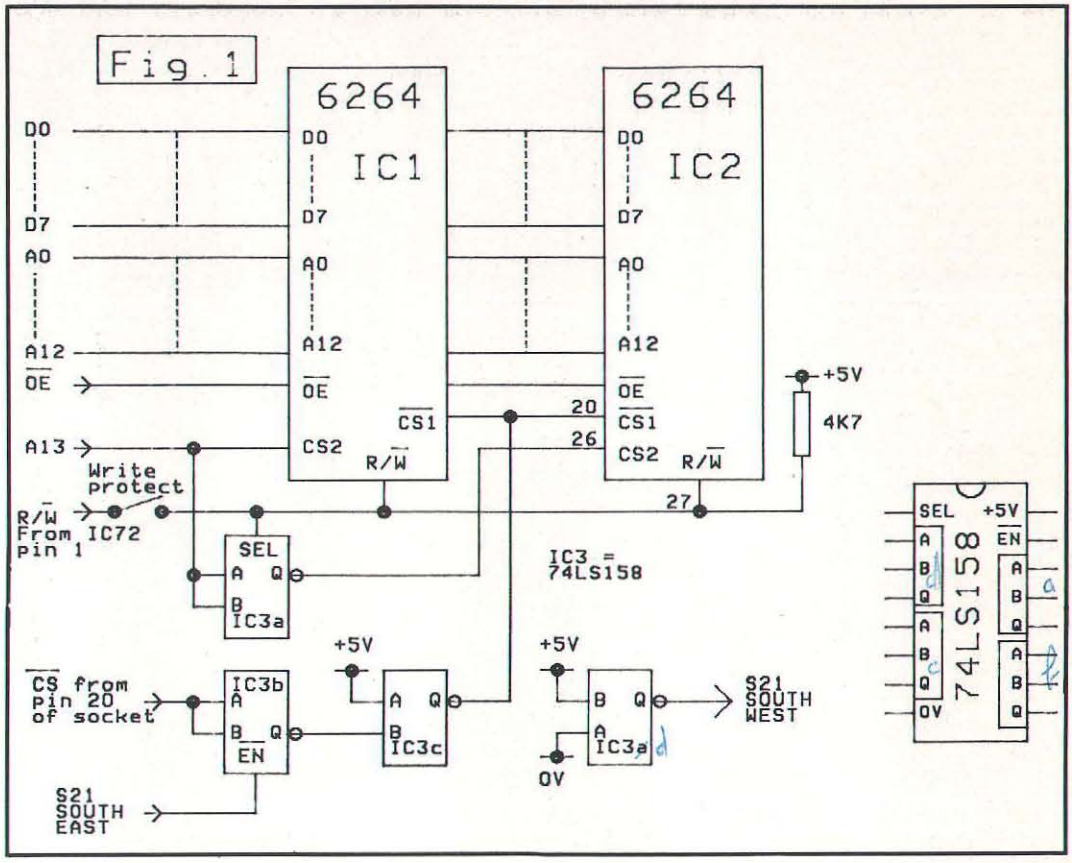
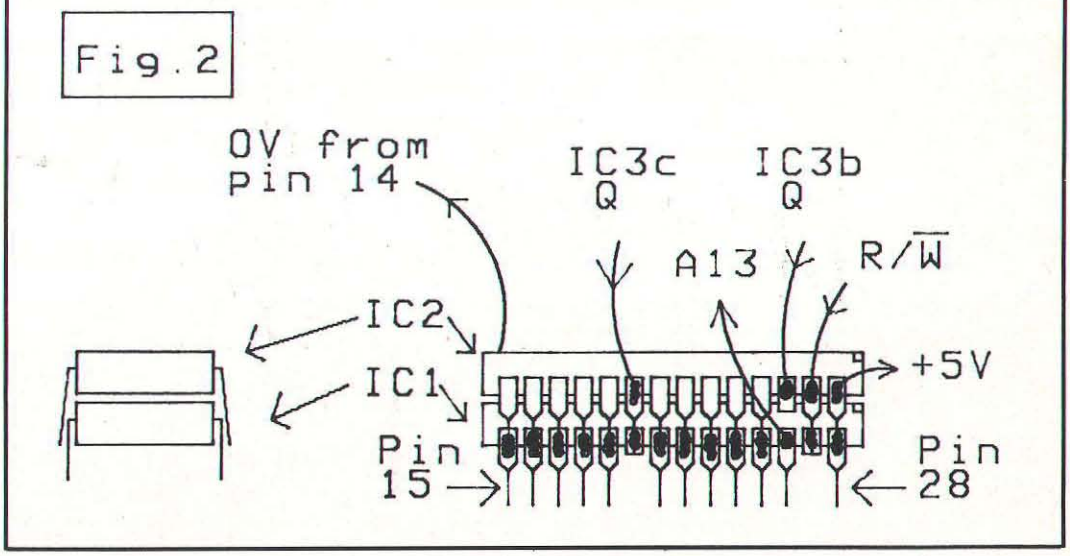


Fig.2



SAVE 30%

KAYSOFT

B.B.C. ONLY

SAVE 30%

JOIN THE MICRO CLUB
£10 LIFE MEMBERSHIP

Introductory Offer Includes:

- ★ TWO FREE GAMES WORTH £14.00
- ★ AN EXTENSIVE RANGE OF ARCADE SPACE AND ADVENTURE GAMES. PLUS BUSINESS EDUCATIONAL AND PROGRAMMING AIDS AND UTILITIES.
- ★ TRY BEFORE YOU DECIDE TO BUY.
- ★ 14 DAYS HIRE FOR £1.50 INC. P.P.
- ★ OPTION TO BUY AT 30% DISCOUNT AFTER DEDUCTION OF HIRE FEE.
- ★ ALL GAMES WITH FULL COLOUR AND DOCUMENTATION, MANY WITH MULTI SCREEN AND SCROLLING AT UNDER £6.00 LESS DISCOUNT.
- ★ CHOOSE YOUR FREE GAMES FROM LIST BELOW:
  Darts · Backgammon · AstroDocker · AstroBlast · Timburst
- ★ CHRISTMAS OFFER:
  ACTION 5 TAPE – 5 SUPER SPACE GAMES
  Breakup · Spacehash · Hellthrush · Destroy · Liquidate

Allow 7 days New Members. S.a.e. for list November

MEMBERS £4.00 + 50p P.P.
OTHERS £6.00 + 50p P.P.
Cheques or Postal Orders to:

KAYSOFT
c/o. LONGACE LTD.
UNIT 1, 5 MILITARY ROAD,
COLCHESTER, ESSEX CO1 2AD
TEL: COLCHESTER 0206 572948 EXT. 1

DEVA
ELECTRONIC CONTROLS LIMITED

DEV-ICE
REAL TIME IN CIRCUIT EMULATION FOR LESS THAN £500

Design and development of microprocessor based systems often encounters problems when software and hardware are brought together.

DEV-ICE offers an excellent cost effective solution.

- *Transparent operation.*
- *Supports 6502 & 65C02.*
- *Resident assembler/ disassembler.*
- *True breakpoint capability.*
- *Delayed (up to 65536 occurrences) breakpoint.*
- *Modify: any memory location, CPU registers or Peripheral I/O device.*
- *Single stepping mode.*
- *No external power supplies.*
- *Target program can be halted and restarted at any position.*
- *Operates with any terminal via RS232 link.*
- *Software packages available to turn the BBC Micro into a powerful development system.*
- *£495 + VAT.*

DEVA
ELECTRONIC CONTROLS LIMITED

Unit 2B Poole Hall Industrial Estate Ellesmere Port
South Wirral Cheshire L66 1ST Tel: 051-356 3767

Vine Micros

THE LEADERS IN
TAPE TO DISC TRANSFER
FOR THE BBC COMPUTER

AN EXCITING NEW DISC UTILITY

100% effective

REPLAY

A 'MUST' for all DISC USERS

THIS IS A UNIQUE HARDWARE/ROM-BASED SYSTEM

BBC 'B' Disc Drive Users - Do you want 100% TAPE TO DISC TRANSFER?
Have you heard about the REPLAY System from VINE MICROS? The *only* system around that can dump anything in the computer memory into disc . . . AND THAT'S NOT ALL! Just look at these extra features. . . .

£35·00 inclusive of VAT and post.

- * GAMES can be stopped and saved directly to disc at any time. This means that you can resume play from *where you left off* and don't need to work through earlier screens time after time.
- * STOPPED SCREENS can be re-loaded from disc and printed via a suitable routine, enabling a permanent record of those high scores!
- * GAME IMPROVEMENTS can be built in! Infinite lives - Hyperspeed - Extra gears - Different background colours - Edit your own improvements or work from the data sheets supplied by us.
- * PAUSE facility built in. * PLUS - most disc to disc transfer.

The **REPLAY** System will not turn you into a pirate! It solely provides a facility for an individual user.

Resulting discs are not recognised by a different computer.

*** Consists of ROM with additional hardware *** *** Easy to fit with detailed instructions, no soldering ***

FOR BBC 'B' with O.S. 1.20 - FITTED WITH THE FOLLOWING DFSs (State type).

Type R8271
Acorn DFS/DNFS Watford and Pace
Single Density

Type R1770
Opus 3.45 and Solidisk DFS.

Type R2791
Opus 3.15.

Type R2793
Opus 3.35.

N.B. If you have an extension ROM board, please make sure that there is still room on the MAIN computer board for the REPLAY ROM
£8.00 DISCOUNT offered for T.D. ROMs to those users wishing to up-grade to REPLAY. (Mail Order Only).

Available by post from:

VINE MICROS, MARSHBOROUGH, NR. SANDWICH, KENT CT13 OPG.

Telephone: (0304) 812276
(or your local Dealer)

VISA

Access

B+ Print Buffer

Ian Copestake

Printer Buffer ROM — real value from your extra RAM.

This article sets out to do two things. To start with, it will show you how to make use of the extra memory on your BBC Model B+, revealing some of the secrets not explained in the User Guide.

It will also show you how to write a program to put on EPROM. If you do not already possess an EPROM programmer, perhaps this will provide you with a good excuse to buy one! The routines in the program are worth studying anyway « you may be able to adapt them for your own purposes.

The program itself provides a very useful and effective facility for anyone who uses a printer. When printing, the BBC Micro places the text to be printed in a printer buffer in the computer's memory. A printer driver program, which is part of the operating system, takes the text out of the buffer again and sends it to the printer. The speed at which it does this depends on the printer: a daisywheel is usually much slower than a dot matrix, for example.

The printer buffer is very small « it can only hold 63 characters at a time « so it will take the computer almost as long to send all the text through the buffer as it takes the printer to put in on paper. Meanwhile, you cannot use the computer for anything else. If the buffer was a decent size, the computer would be free for other work much more quickly. Some printers have a small buffer of their own, which helps a little, and you can also buy add-on buffers at a price. A few applications programs, such as the *Wordpower* wordprocessor, solve the problem by providing background printing. The program described in this article allows you to make your own buffer, holding just under 12k of text, for the cost of an EPROM. You can of course use if for printing BASIC program listings, but it should also work for almost any task involving printing.

WHAT IS SHADOW RAM?

At first sight the Model B+ seems to offer a great deal more RAM than the Model B, but it is not immediately obvious where this extra 32k is or how it can be used. The shadow screen is explained in chapter 42 of the new User Guide, and this accounts for 20k. But what of the remaining 12k?

The User Guide keeps remarkably quiet about it « the 12k of 'shadow' RAM (or 'sideways' RAM or 'paged' RAM) does not get a mention until page 452. It is then disposed of in about a dozen lines, two of which contain the ominous statememt that it is 'not necessarily applicable to other Acorn products'.

This dire warning, which gives a completely new meaning to the term 'volatile RAM', suggests that our Shadow Print Buffer (SPB) program may not work on the BBC Model C « if and when such a machine appears. That should not stop us from using the shadow RAM while it is here, and it is a pity that the User Guide is so coy about it.

In fact the shadow RAM occupies the same position in the computer's memory map as a sideways rom or EPROM would. The BBC Micro allows several different roms to share this space, and can select or 'page in' the particular rom it wants to use at any given moment. The BASIC rom is normally paged in when you switch on, but if you have disc drives the DFS rom will be paged in every time you use them. The shadow RAM on the Model B+ can be paged in and out rather like a rom, which is why it is sometimes called paged RAM.

The total amount of memory space which can be shared by paged roms or RAM is 16 kilobytes, and occupies the addresses from &8000 to &BFFF inclusive. However, there are only 12 kilobytes of shadow RAM, because that is what happened to be left over when the 20k of shadow screen had been provided (32k is a nice round figure where memory is concerned). This leaves 4k of memory, from &B000 to &BFFF, which is not part of the shadow RAM.

What if it were possible to put a machine code program in this spare 4k, which used the 12k of shadow RAM to carry out some background task? It would be like a self- contained unit within the computer, with its own private RAM. Well, it is possible, and this is what the SPB program does.

THE SPB EPROM

In some ways, the SPB program is constructed like any other sideways rom. It is designed to be placed on a 16k EPROM, even though the actual code is less than 1k in length. This is unavoidable, because there has to be some code at each end of the 16k; but EPROMs are becoming so cheap now that this method is still far more economical than buying an add-on buffer.

When the BBC Micro is switched on, or the BREAK key is pressed, the computer looks at any chips plugged into the sideways sockets to see what they are. In order to be recognised, each rom or EPROM must start with a 'rom header' at &8000. This is a set of information in a particular format « the details are explained in *Beginners' ROM* in this issue.

Amongst other things, the rom header tells the computer whether the socket contains a language or a service program (or both). Put very simply, the difference is that a language takes over the whole running of the computer, whilst a service is performed every now and then as a sort of subroutine. Thus BASIC is a language but the DFS is a service. The SPB program is also a service, and should work whatever language you are using.

WHY NOT PUT THE SPB PROGRAM IN SHADOW RAM?

It would be very convenient if we could put the rom header for SPB into the shadow RAM itself, instead of into EPROM, since they both start at &8000. It would be rather fiddly to get it there, because we cannot *LOAD straight into that area; but this method would not work in any case. Although it shares the same address space, shadow RAM is ignored completely when the computer examines the sideways sockets. This is what forces us to put the program in EPROM. The only other approach would be to modify the operating system initialisation routine, which is virtually impossible.

The SPB EPROM therefore starts with a rom header at &8000. The rest of the SPB code is located at &B000, so that it does not conflict in any way with the 12k of shadow RAM. The SPB needs 12 bytes for variables, which it stores at &8000. All the remainder of the shadow RAM, from &800C to &AFFF, can be used for the print buffer. This gives a buffer size of &2FF4 (12,276) bytes.

How can the variables be stored in the same address space as the rom header? The answer is simply that the variables are in the shadow RAM, whereas the rom header is in the EPROM. They will never need to be paged in at the same time.

SO WHY PUT MOST OF THE CODE AT &B000?

It would in fact be possible for the rest of the SPB code to follow immediately after the rom header, occupying the same address space as the print buffer. The EPROM has to be paged in while the code is running, but there are means by which it could read and write the shadow RAM at the same time. These methods are rather long-winded, however, and there is a much better way.

A simple technique, barely hinted at in the User Guide, allows us to page in both the shadow RAM and the EPROM at the same time. While this 'split paging' is in force the operating system will assume that addresses in the range &8000 to &AFFF refer to shadow different commands are used to turn the buffer on and off, or so that the program asks for confirmation if the current buffer is not empty.

The standard print buffer will be re-selected if BREAK is pressed.

The final section of SPB is the one which does all the work. It puts RAM, while addresses from &B000 to &BFFF refer to our EPROM. By putting the main part

of our code at &B000 we avoid the need for any tricky methods of accessing shadow RAM.

SPLIT PAGING

When the operating system wishes to communicate with a particular sideways rom, it selects it by placing the rom's number into a 'paged rom select register', which lives at &FE30 (part of the SHEILA area). A copy of the number is also kept in zero-page location &F4.

The rom number is the number of the socket the chip is plugged into. The socket numbers are listed on page 407 of the B+ User Guide, under *FX142.

The shadow RAM can be selected just as if it were a rom sitting in a socket with the number 128. In fact putting any number from 128 to 255 (that is, with the top bit set) into the paged rom select register will ensure that addresses from &8000 to &AFFF refer to shadow RAM.

But what will happen if a call is made to an address in the range &B000 to &BFFF? Suppose the paged rom select register contains 131 (10000011 in binary). Although bit 7 is set, the call cannot refer to sideways RAM because that stops at &AFFF. The operating system therefore looks at the bottom four bits and assumes they refer to a paged rom. In this case the bottom four bits contain 0011 (binary), or 3 (decimal). The call will therefore be directed to the rom (or EPROM) in socket number 3.

So all we have to do to select our SPB EPROM and shadow RAM at the same time is to find out the socket number of the EPROM, add 128 to set the top bit, and place the resulting number into the paged rom select register.

HOW THE SPB PROGRAM IS ORGANISED

The SPB program has four main sections. First there is the rom header, which we have already mentioned. Then comes another obligatory section which displays a title and version number when *HELP is typed.

The third part contains the initialisation and cancellation procedures. The SPB utility can be initialised at any time after switch-on by typing *SPB. To cancel it, reverting to the standard printer buffer, type *SPB again, and so on. If you wish to experiment you could modify the program so that characters into the buffer and takes them out again, and answers questions about the buffer status when required. The SPB code gains control of these activities because three vectors in page 2 of the computer's memory are altered when SPB is initialised. These vectors are addresses, which normally point to standard operating system routines. The SPB program alters them to point to its own routines. Any other correctly-written program (such as a word-processor or BASIC itself) which wants to do something with a buffer will 'indirect' its calls via these vectors, rather than jumping straight into the standard routines. Alternatively it will use OSBYTE calls, which in turn use the vectors.

VECTORS AND SIDEWAYS ROMS

There is a slight complication when it comes to using vectors with sideways roms and EPROMs. In our case, we wish to make the vectors point to code somewhere above &B000. But the call to our code might come from normal RAM, or another sideways rom, or from the operating system itself. How will the computer know that the vectors are meant to point to our particular chip?

The answer is that another byte of information must be included with each vector, giving the socket number of our EPROM (+128 as described earlier). The combination of a two-byte address with a one-byte rom number is called an 'extended vector'.

There is only room in page 2 for two bytes for each vector, so space has to be reserved elsewhere for extended vectors. The computer decides where this space is, and a special OSBYTE call allows us to discover the start of the extended vector space. Each extended vector takes up 3 bytes of memory; so if the normal vector is located at &0200 + [offset], the corresponding extended vector will be at [start of extended vector space] + [offset * 3/2].

We have to make an entry in the normal vector location as well, telling the operating system that an extended vector is in use. This is done by making the normal vector point to &FF00 + [offset * 3/2].

THE THREE BUFFER VECTORS

Three different vectors are used in handling buffers on the BBC Micro. The buffer insert vector (INSV) is used whenever a character is to be inserted into a buffer. The buffer remove vector (REMV) is used when characters are removed, but also allows a program to find out what the next character is without actually removing it. The code pointed to by these vectors must also indicate when it has failed because the buffer is full or empty.

The count/purge vector (CNPV) can either count the number of items in a buffer, or calculate the amount of free space remaining, or empty the buffer altogether.

Where options are available, the calling code selects them by giving certain values to the Carry or Overflow flags in the status register.

Of course there are many different buffers in the computer's memory. When the SPB program responds to a vectored call it must first of all check that it relates to the print buffer. If it does not, SPB will immediately jump back to the code which would have been executed if SPB did not exist. This is made possible by saving the contents of the three vectors before they are changed.

HOW DOES THE BUFFER WORK?

The print buffer works on a 'first in first out' basis, and has a circular form. Characters do not shuffle forward in a queue, but stay put and wait their turn to be dealt with like customers seated in a restaurant. When the last place is full, the first seats start filling up again.

An 'insert pointer' and a 'remove pointer' always point to the next free place and the next character to be removed, respectively. When one of these pointers reaches the last buffer location it goes back again to the first. If the two pointers have the same value this is taken to mean that the buffer is empty. If you think about this you will see that the buffer can never be absolutely full; one space must always remain empty.

The operating system keeps the pointers in page 2, and uses some neat arithmetic to work out the buffer status from them. Much of this neatness depends on the fact that none of the standard buffers are more than 256 bytes in length, and so the pointers require only one byte. The SPB program, handling a much larger buffer, needs two bytes per pointer. Apart from this, SPB runs the print buffer in much the same way as the operating system would.

HOW TO PRODUCE YOUR SPB EPROM

Listing 1 contains everything needed to assemble the object code for SPB. Make sure you have a disc ready with plenty of free space on it before you start.

Although the actual code is quite short the object file itself will be large, because it must include all the empty space from &800C to &AFFF. To make room for it, you must leave out REM lines and blank lines when typing in the listing. Start with AUTO, and press RETURN straight away for the non-essential lines. Select the shadow screen with *SHADOW if you are not working in MODE 7.

Save the source code file as "S.SPB" in case of errors, then RUN the program. OPT is set to suppress the assembly listing, in order to speed things up, but any assembly errors will be reported.

If all goes well, three hexadecimal values will be displayed on the screen. The first is a simple checksum, and should read &10285. If it does not you have probably made a typing error, so check the listing carefully. The other two numbers are the addresses of the first and last-plus-one bytes of the object code. The difference between them (i.e. the length of the code) should be &3222. To save the object code, type ***SAVE O.SPB ssss eeee** where ssss and eeee are the start and end addresses.

Follow the instructions with your EPROM programmer to produce your 16k EPROM, using O.SPB as the object code file. The last &0DDE bytes of the 16k are not used, so it does not matter whether they are 'blown' or not. Your EPROM programmer may allow you to specify the length of the code, which would be &3222, or fill the unused locations with &FF to speed programming. You are now ready to fit the EPROM in your computer (make sure it is switched off at the time). When you have done this, switch on and type *HELP. Amongst other things you should see "SPB 0.1". If this does not appear, switch off immediately and carefully check the fitting of the EPROM.

To see the shadow print buffer in action, activate it with *SPB, then print out a longish file such as S.SPB. The printer will churn on for several minutes, but the computer will be ready for the next task after a few seconds.

HOW SPB RELATES TO THE OPERATING SYSTEM

SPB can be used in conjunction with the shadow screen, and with

CONTINUED OVER

the 6502 Second Processor. OSBYTE calls will work correctly, and the 'output buffer empty' event is generated when appropriate.

If OSBYTE &80 is used to find out how much space is left in the print buffer, the high byte of the result will be returned in the Y register (with the standard buffer the high byte is always 0).

OSBYTE &98 deserves special mention. There was a bug in OS 1.2 which had more or less been accepted as part of the computer's specification. According to Acorn, this call was supposed to return the value of any character read in the Y register. In fact the character could only be read by a "LDA (&FA),Y" instruction after the call; it could not be read reliably from BASIC.

The bug, caused by a branch instruction which was one out, has been cured in OS 2.0: the Y register does now contain the character read. This has one unfortunate consequence. Not only is the "LDA (&FA),Y" rigmarole unnecessary « it will not work! Any Model B software which used this procedure will not run correctly on the B+.

OTHER WAYS OF ACCESSING THE SHADOW RAM

The SPB program illustrates one way of using all that extra memory. In a future article I will look at alternative methods, and search for a means of loading and saving the shadow RAM.

If you do not have access to an EPROM programmer, or simply wish to save your typing finger, you can obtain a readymade SPB rom by sending £12 to Ian Copestake at 23 Connaught Crescent, Brookwood, WOKING, Surrey, GU24 0AN.

NOTES ON THE SOURCE CODE

The REM lines in Listing 1 always precede the code they describe. The asterisks indicate their level of importance:

580-630 OPT is set to 4 and 6 (rather than 0 and 2) so that the program counter (P%) can start at &8000 while the code is actually stored in RAM (at O%).
670-710 SPB has no language entry point, so the first three bytes of the rom header are set to zero. 1070-1160 This loop fills up the blank space in the object file, between the end of the rom header and the start of the main code.
1180-3700 The operating system offers sideways roms the chance to respond to various 'service calls', identified by a number in the accumulator. SPB responds to numbers 9 ("HELP) and 4 (unrecognised * command).
1380-1400 SPB must not respond, for example, to "*HELP DFS", so we must check that nothing was typed after "*HELP" on the command line. The operating system routine GSINIT does this very conveniently. It requires that '&F2,3 +Y should point to the byte after "*HELP", which is already the case. The Carry flag was set by line 1280 (because A=9), and ensures that GSINIT will check right up to the end of the command line. If the Carry flag was clear, GSINIT would stop at the space before "DFS". If GSINIT finds no more non-blank characters before it stops, i.e. the rest of the command line is blank, the Zero flag is set on return.
1810-1900 When a rom is offered a command which the operating system does not recognise, it is pointed to on entry by '&F2,3'+Y. It is good programming practice to use a label such as "clbase" rather than referring to a location directly. If the location changes in some later version of the BBC Micro, only line 230 will need amending.
1920-1940 As with *HELP, we must check that the user did not type "*SPBXYZ" - this command might belong to some other rom.
1960-2070 Shadow RAM is selected at the same time as our own EPROM, so that we can initialise the variables stored in it. Using OSBYTE &97 is the 'correct' way to write to a SHEILA address, though "STA &FE30" would work just as well.
2110-2140 If SPB has already been turned on the high bytes of the vectors will equal &FF, because they will be pointing to the extended vector processing area.
2390-2480 The contents of our zero-page workspace are saved here and restored on exit, in case SPB has been called from within another program which was using them.
2650-2720 OSBYTE &A8 returns the address of the extended vector space in the X and Y registers.
2740-2760 This is the value we put into romnum at line 2010.
2850-2990 Fiddly arithmetic for the count/purge vector.
3000-3290 The same for the other two vectors.
3390-3460 By making the two addresses equal, we imply that the buffer is empty. They are made to point to the last space in the buffer, but any address within the buffer would do just as well.
3650 We exit with 0 in the accumulator, to show that the unrecognised command has been dealt with. If a lower-priority rom wants to act on "*SPB", it will not get the chance.
3760 "cnp" labels the address pointed to by the extended count/purge vector.
3790-3840 If the X register does not contain 3, this call is nothing to do with the print buffer, and SPB jumps straight to the original vector address.
3860-3890 The Overflow flag decides whether the buffer is to be counted (flag clear) or purged (flag set).
3910-3970 To purge the buffer all we need do is make the insert and remove addresses equal, to indicate that it is empty. It need not actually be wiped.
4110-4270 To count the buffer we start by subtracting the remove address from the insert address. But because the buffer has a circular form (as explained earlier) this may produce a negative result. If the insert address had wrapped right round so that it was only 1 less than the remove address, the result contained in the X and Y registers would be &FFFF, and the buffer would be full (except for the one unusable location). The subtleties of machine-code arithmetic allow us to convert this to the correct result simply by adding the buffer length, and ignoring the final carry. In our example, [&FFFF + length] gives [length-1].
4330-4340 When the buffer is counted, we can return either the number of items in the buffer or the amount of free space remaining. The Carry flag will be clear or set accordingly when cnp is called.
4360-4500 To calculate the free space, we subtract the current values of the X and Y registers from the buffer length, subtracting an extra 1 for the unusable location. The extra subtraction is engineered by clearing instead of setting the Carry flag at line 4420.
4560 "ins" labels the address pointed to by the extended insert vector.
4710-4760 Locations &FA and &FB are the ones used by the operating system for post-indexed indirect addressing when handling buffers.
4780-4820 The insert address already points to the next free location, but this might be the very last one, which cannot be used. To find out, we check whether [insert address + 1] ...
4940-5000 ... is equal to the remove address. If so, the buffer is effectively full. The routine exits after setting the Carry flag at line 5100 to indicate failure. Note that the insert address has not actually been incremented.
4840-4920 This is the code which will make the insert address wrap round to the start of the buffer when it reaches the end.
5170-5200 The insert address is given its new value, which has been wrapped if necessary, ready for the next insertion.
5220-5240 The current insertion however is at the old insert address, which we put at &FA,B earlier.
5430 "rem" labels the address pointed to by the extended remove vector.
5630-5680 The remove address already points to the next character to be removed, unless it is equal to the insert address, indicating that the buffer is empty. In this case the Carry flag is set to indicate failure.
5780-5800 The Overflow flag (unchanged since entry) is set if the next character is only to be examined, or clear if it is actually to be removed from the buffer.
5820-6010 Removing a character from the buffer consists of incrementing the remove address, wrapping it round to the buffer start if necessary.
6030-6240 If this makes the remove and insert addresses equal, the buffer is now empty, and SPB has a duty to generate an output buffer empty event'. This is done by calling OSEVEN, specifying the event number and the print buffer number. (Since the print buffer is an output buffer, we do not have to generate an event when it becomes full).

PROGRAM LISTING

```
  10REM SHADOW PRINT BUFFER: Listing 1 (DE32)
  20  (77F8)
  30  (BD94)
  40REM B+ SHADOW PRINT BUFFER (7DD3)
  50REM Copyright (C) 1985 IAN COPESTAKE (6082)
  60  (3E8D)
  70REM VERSION 0.1 (6433)
  80  (5189)
  90REM *** RESERVE SPACE FOR OBJECT CODE (C026)
 100  (C0C2)
 110DIM obj &3FFF (7168)
 120  (7459)
 130REM *** DEFINE OS ADDRESSES (A7AF)
 140  (B9B7)
 150cnpv=&022E (F492)
 160insv=&022A (5A53)
 170remv=&022C (040E)
 180gsinit=&FFC2 (4ED4)
 190osbyte=&FFF4 (010C)
 200oseven=&FFBF (CF3D)
 210osnewl=&FFE7 (2113)
 220oswrch=&FFEE (36C1)
 230clbase=&00F2 (AE35)
 240romnum=&00F4 (613A)
 250  (AB4B)
 260REM *** DEFINE OTHER CONSTANTS (7171)
 270  (1CD0)
 280bufend=&B000: REM Address of 1st byte after p
rint spool buffer (4E20)
 290codstt=&B000: REM Address of SPB code (C600)
 300  (7222)
 310REM *** LABEL WORKSPACE (ECBC)
 320  (C6B9)
 330REM ** SHADOW RAM WORKSPACE (95BC)
 340  (3B57)
 350REM * RESERVE SPACE FOR OLD BUFFER VECTORS (B
9A1)
 360olcnpv=&8000 (1B1A)
 370olinsv=&8002 (99A3)
 380olremv=&8004 (0E6B)
 390  (2AA4)
 400REM * RESERVE SPACE FOR BUFFER INSERT & REMOV
E ADDRESSES (753F)
 410insadr=&8006 (B38C)
 420remadr=&8008 (91F9)
 430  (A2C6)
 440REM * RESERVE GENERAL WORKSPACE (1D5E)
 450work1=&800A (CB01)
 460work2=&800B (5962)
 470  (EBB3)
```

```
  480REM ** LABEL 1ST BYTE OF SHADOW PRINT BUFFER
(68A2)
  490  (84B7)
  500bufstt=&800C (BA88)
  510  (4F2D)
  520REM ** ZERO-PAGE WORKSPACE (13DC)
  530  (FBB6)
  540zwork1=&70 (2487)
  550zwork2=&71 (68CB)
  560zwork3=&72 (2CA9)
  570  (B2C3)
  580REM *** INITIATE ASSEMBLY (FB06)
  590  (DDC7)
  600FOR pass=4 TO 6 STEP 2 (AACD)
  610O%=obj (D30B)
  620P%=&8000 (AA95)
  630[OPT pass (496B)
  640  (27A4)
  650REM *** ROM HEADER (AB27)
  660  (933F)
  670REM ** LANGUAGE ENTRY NOT USED (A9DD)
  680  (FC3B)
  690.head (BF16)
  700EQUB 0 (D8A2)
  710EQUW 0 (A4E8)
  720  (833A)
  730REM ** SERVICE ENTRY (5ED2)
  740  (7ED4)
  750JMP serv (41B6)
  760  (CA4F)
  770REM ** ROM TYPE: SERVICE ROM (18AF)
  780  (A54B)
  790EQUB &82 (6B31)
  800  (12B4)
  810REM ** COPYRIGHT OFFSET POINTER (8BD1)
  820  (A62F)
  830EQUB copy-head (1423)
  840  (5BC1)
  850REM ** BINARY VERSION NUMBER (AD1E)
  860  (EF5A)
  870EQUB 1 (D1E8)
  880  (805E)
  890REM ** TITLE STRING (D57D)
  900  (4BC4)
  910.titl (504F)
  920EQUS "SPB" (85DE)
  930  (3533)
  940REM ** VERSION STRING (6057)
  950  (C8DD)
  960.vers (D405)
  970EQUB 0 (B00C)
  980EQUS "0.1" (B887)
  990  (1342)
 1000REM ** COPYRIGHT STRING (4305)
 1010  (921B)
 1020.copy (41FD)
 1030EQUB 0 (5374)
 1040EQUS "(C) 1985 Ian Copestake" (3F8D)
 1050EQUB 0 (8503)
 1060  (A599)
 1070REM *** SKIP UNUSED ADDRESSES IN ROM (AF62)
 1080  (CA9D)
 1090.skip (E093)
 1100] (D27A)
 1110FOR F%=skip TO codstt-1 (E45B)
 1120[OPT pass (FA5E)
 1130EQUB 0 (12D3)
 1140] (97F9)
 1150NEXT (BF9C)
 1160[OPT pass (DC17)
 1170  (3685)
 1180REM *** SERVICE ENTRY (D5C1)
 1190  (5981)
 1200REM ** PRESERVE STATUS REGISTER ON STACK (7C6
7)
 1210  (20FB)
 1220.serv (E1AE)
 1230PHP (E16D)
 1240  (A3E2)
 1250REM ** RESPOND TO *HELP (AC9F)
 1260  (1779)
 1270REM * IF *HELP HAS NOT BEEN TYPED, BRANCH TO
init (627A)
 1280CMP #9 (FCDD)
 1290BNE init \ Service type <>9 (D9D8)
 1300  (B3E7)
 1310REM * PRESERVE A,Y,X REGISTERS ON STACK (EB74
)
 1320PHA (1474)
 1330TYA (2A00)
 1340PHA (CC99)
 1350TXA (3881)
 1360PHA (84C2)
 1370  (8465)
 1380REM * IF REST OF COMMAND LINE NOT BLANK, EXIT
 VIA exita (A473)
 1390JSR gsinit \ (C=1) (93A4)
 1400BNE exita \ Z=0 (B4C4)
 1410  (D798)
 1420REM * OUTPUT TITLE & VERSION STRINGS (2BFA)
 1430JSR osnewl (050C)
 1440LDX #0 (B591)
 1450.helpj (E45A)
 1460LDA titl,X (3DF3)
 1470BNE helpl \ A<>0 (F5C6)
 1480LDA #ASC" " \ Substitute space for 0 byte (A6
B9)
 1490.helpl (A97D)
 1500JSR oswrch (DF58)
 1510INX (6B1A)
 1520CPX #copy-titl (BEDB)
 1530BCC helpj \ End of version string not reached
 (A714)
 1540JSR osnewl (EDC1)
 1550  (C79D)
 1560REM * RESTORE REGISTERS FROM STACK; EXIT (363
7)
 1570.exita (4C85)
 1580PLA (C835)
 1590TAX (6EE7)
 1600PLA (A73A)
 1610TAY (21AB)
 1620PLA (EF61)
 1630.exitf (143E)
 1640PLP (5379)
 1650RTS (A22D)
 1660  (52FA)
 1670REM ** RESPOND TO *SPB (2010)
 1680  (3DFE)
 1690REM * IF AN UNRECOGNISED COMMAND HAS NOT BEEN
 TYPED, EXIT VIA exitf (F07F)
 1700.init (DFCE)
 1710CMP #4 (B971)
 1720BNE exitf \ Service type <>4 (631B)
 1730  (8893)
 1740REM * PRESERVE A,Y,X REGISTERS ON STACK (B876
)
 1750PHA (EF15)
 1760TYA (993A)
 1770PHA (A74E)
 1780TXA (8A94)
 1790PHA (7E8C)
 1800  (D371)
 1810REM * COMPARE COMMAND LINE WITH TITLE STRING;
 IF NO MATCH, EXIT VIA exita (419B)
 1820LDX #0 (0887)
 1830.initb (B76A)
 1840LDA (clbase),Y (F069)
 1850CMP titl,X (355C)
 1860BNE exita (5DD1)
 1870INY (2E97)
 1880INX (631A)
 1890CPX #vers-titl (5A57)
 1900BCC initb \ End of title string not reached (
51AF)
 1910  (406D)
 1920REM * IF REST OF COMMAND LINE NOT BLANK, EXIT
 VIA exita (58DA)
 1930JSR gsinit \ (C=1) (7486)
 1940BNE exita \ Z=0 (0F64)
 1950  (0918)
 1960REM ** SELECT SHADOW RAM: SET BIT 7 OF PAGED
ROM SELECT REGISTER (DC96)
 1970  (BD83)
```

CONTINUED OVER

```
 1980REM * SET RAM COPY OF REGISTER (52A1)
 1990LDA romnum (4B2C)
 2000ORA #&80 (0F6E)
 2010STA romnum (F86E)
 2020  (E0B3)
 2030REM * SET SHEILA LOCATION &30 (CBB5)
 2040TAY (A177)
 2050LDA #&97 (2CC4)
 2060LDX #&30 (2172)
 2070JSR osbyte (F01B)
 2080  (C6C2)
 2090REM ** IF PRINT SPOOL BUFFER ALREADY IN USE,
TURN IT OFF (B682)
 2100  (0D58)
 2110REM * IF EXTENDED BUFFER VECTORS ARE NOT SET
UP, BRANCH TO initd (706A)
 2120LDX cnpv+1 (7105)
 2130INX (CCE7)
 2140BNE initd \ Hi byte of vector <>&FF (5B3E)
 2150  (8E41)
 2160REM * DISABLE INTERRUPTS (01B2)
 2170SEI (DB10)
 2180  (9FB2)
 2190REM * RESTORE OLD VECTORS FROM SHADOW RAM LOC
ATIONS olcnpv,olinsv,olremv (6764)
 2200LDA olcnpv (9AC3)
 2210STA cnpv (83E3)
 2220LDA olcnpv+1 (C82E)
 2230STA cnpv+1 (E612)
 2240LDA olinsv (CC10)
 2250STA insv (4330)
 2260LDA olinsv+1 (55D5)
 2270STA insv+1 (AF28)
 2280LDA olremv (AA98)
 2290STA remv (BFFB)
 2300LDA olremv+1 (A7CE)
 2310STA remv+1 (2F66)
 2320  (0B23)
 2330REM * ENABLE INTERRUPTS (6044)
 2340CLI (024E)
 2350  (3CA1)
 2360REM * EXIT VIA exitg (91D7)
 2370JMP exitg (4BB6)
 2380  (2D52)
 2390REM ** PRESERVE CONTENTS OF THE 3 BYTES OF ZE
RO-PAGE WORKSPACE ON STACK (D2E1)
 2400  (11AB)
 2410.initd (4E46)
 2420LDX #0 (9DCD)
 2430.initdb (EF75)
 2440LDA zwork1,X (9E35)
 2450PHA (F913)
 2460INX (5A3D)
 2470CPX #3 (B46E)
 2480BCC initdb \ X<3 (1476)
 2490  (492D)
 2500REM ** PRESERVE OLD BUFFER VECTORS IN SHADOW
RAM LOCATIONS olcnpv,olinsv,olremv (49CC)
 2510  (82B7)
 2520LDA cnpv (91D8)
 2530STA olcnpv (A48B)
 2540LDA cnpv+1 (D9BC)
 2550STA olcnpv+1 (6C74)
 2560LDA insv (510B)
 2570STA olinsv (F258)
 2580LDA insv+1 (934A)
 2590STA olinsv+1 (7B82)
 2600LDA remv (BB1A)
 2610STA olremv (1D6F)
 2620LDA remv+1 (E22E)
 2630STA olremv+1 (9C19)
 2640  (EA3E)
 2650REM ** MAKE (zwork1,2)= ADDRESS OF EXTENDED V
ECTOR SPACE (4517)
 2660  (5EA5)
 2670LDA #&A8 (8BE6)
 2680LDX #0 (4459)
 2690LDY #&FF (62AB)
 2700JSR osbyte (2D95)
 2710STX zwork1 (9839)
 2720STY zwork2 (3EA1)
 2730  (84CC)
 2740REM ** MAKE X= ROM NUMBER FOR EXTENDED VECTOR
S (F196)
 2750  (7922)
 2760LDX romnum (A82F)
 2770  (CDB9)
 2780REM ** DISABLE INTERRUPTS (7026)
 2790  (A2BD)
 2800SEI (9D52)
 2810  (1542)
 2820REM ** UPDATE VECTORS TO POINT TO &FF00+3*N (
VECTORS EXIST AT LOCATION &0200+2*N) (CF10)
 2830REM ** SET EXTENDED VECTORS AT (zwork1,2)+3*N
TO POINT TO SPB CODE (98D7)
 2840  (965B)
 2850LDA #cnpv MOD&100 \ A= lo byte of vector addr
ess (CC47)
 2860STA zwork3 (C61B)
 2870LSR A (4D49)
 2880CLC (9069)
 2890ADC zwork3 \ A=A/2+A (EF51)
 2900STA cnpv \ Lo byte of vector =A (27B8)
 2910TAY (AF6E)
 2920LDA #cnp MOD&100 (582E)
 2930STA (zwork1),Y (388B)
 2940INY (8F06)
 2950LDA #cnp DIV&100 (6561)
 2960STA (zwork1),Y \ Extended vector =cnp (2AE9)
 2970INY (7351)
 2980TXA (21B1)
 2990STA (zwork1),Y \ ROM number =X (9F19)
 3000LDA #insv MOD&100 (59AE)
 3010STA zwork3 (86CA)
 3020LSR A (73B2)
 3030CLC (AA71)
 3040ADC zwork3 (7509)
 3050STA insv (506D)
 3060TAY (210F)
 3070LDA #ins MOD&100 (026E)
 3080STA (zwork1),Y (33F7)
 3090INY (4813)
 3100LDA #ins DIV&100 (5EE0)
 3110STA (zwork1),Y (985C)
 3120INY (908B)
 3130TXA (1BA9)
 3140STA (zwork1),Y (31B0)
 3150LDA #remv MOD&100 (83D3)
 3160STA zwork3 (6E07)
 3170LSR A (AD2E)
 3180CLC (72E9)
 3190ADC zwork3 (9EFB)
 3200STA remv (95F4)
 3210TAY (062E)
 3220LDA #rem MOD&100 (C68D)
 3230STA (zwork1),Y (83E5)
 3240INY (2646)
 3250LDA #rem DIV&100 (FBC2)
 3260STA (zwork1),Y (2A09)
 3270INY (DA11)
 3280TXA (88F1)
 3290STA (zwork1),Y (F07E)
 3300LDA #&FF (AE8B)
 3310STA cnpv+1 \ Hi byte of vector =&FF (6458)
 3320STA insv+1 (3A4F)
 3330STA remv+1 (52CC)
 3340  (F2F8)
 3350REM ** ENABLE INTERRUPTS (0B05)
 3360  (4663)
 3370CLI (363A)
 3380  (2967)
 3390REM ** INITIALISE BUFFER INSERT & REMOVE ADDR
ESSES, =bufend-1 (BC71)
 3400  (159E)
 3410LDX #(bufend-1)MOD&100 (4BE4)
 3420LDY #(bufend-1)DIV&100 (B0EF)
 3430STX insadr (F00D)
 3440STY insadr+1 (5AF7)
 3450STX remadr (09C3)
 3460STY remadr+1 (5777)
 3470  (221C)
 3480REM ** RESTORE OLD CONTENTS OF THE 3 BYTES OF
 ZERO-PAGE WORKSPACE FROM STACK (E10F)
 3490  (4D18)
```

```
 3500LDX #2 (84D8)
 3510.inith (196E)
 3520PLA (3144)
 3530STA zwork1,X (98B2)
 3540DEX (844D)
 3550BPL inith \ X>=0 (AC09)
 3560  (B100)
 3570REM ** RESTORE REGISTERS FROM STACK; PREVENT
OTHER ROMS RESPONDING TO *SPB (8271)
 3580  (DE04)
 3590.exitg (E7BD)
 3600PLA (173F)
 3610TAX (B1ED)
 3620PLA (5F64)
 3630TAY (D9F5)
 3640PLA (8789)
 3650LDA #0 (4D17)
 3660PLP (AB27)
 3670  (90FC)
 3680REM ** EXIT (3C17)
 3690  (FFF8)
 3700RTS (1372)
 3710  (3462)
 3720REM *** BUFFER COUNT/PURGE (75BA)
 3730  (80F9)
 3740REM ** PRESERVE STATUS REGISTER ON STACK (46D
E)
 3750  (7D17)
 3760.cnp (4775)
 3770PHP (73BE)
 3780  (6CE4)
 3790REM ** IF PRINTER BUFFER NOT SPECIFIED: RESTO
RE STATUS REGISTER FROM STACK; TRANSFER TO OLD VEC
TOR (6D22)
 3800  (DB1B)
 3810CPX #3 (BCA0)
 3820BEQ cnpb \ Buffer number =3 (E47E)
 3830PLP (5C9E)
 3840JMP (olcnpv) (7663)
 3850  (5802)
 3860REM ** IF COUNT REQUIRED, BRANCH TO cnpd (5F4
A)
 3870  (EC99)
 3880.cnpb (5A4A)
 3890BVC cnpd \ V=0 (D497)
 3900  (826B)
 3910REM ** PURGE BUFFER (5D1D)
 3920  (36F0)
 3930REM * MAKE BUFFER INSERT ADDRESS = BUFFER REM
OVE ADDRESS (80E5)
 3940LDA remadr (0752)
 3950STA insadr (8DF1)
 3960LDA remadr+1 (F131)
 3970STA insadr+1 (7767)
 3980  (1081)
 3990REM * RESTORE STATUS REGISTER FROM STACK; EXI
T (WITH X,Y UNCHANGED) (574F)
 4000.exith (8B5A)
 4010PLP (23E9)
 4020.exitj (6630)
 4030RTS (2EEA)
 4040  (05E3)
 4050REM ** COUNT BUFFER (0D3E)
 4060  (B178)
 4070REM * DISABLE INTERRUPTS (A9A9)
 4080.cnpd (88C0)
 4090SEI (D77E)
 4100  (15E6)
 4110REM * CALCULATE NUMBER OF ITEMS IN BUFFER (9E
3D)
 4120REM Make X,Y =insadr-remadr (26C4)
 4130SEC (D937)
 4140LDA insadr (0180)
 4150SBC remadr (1822)
 4160TAX (1CC3)
 4170LDA insadr+1 (221B)
 4180SBC remadr+1 (764E)
 4190TAY (514E)
 4200REM If result negative, add buffer length, =b
ufend-bufstt (4746)
 4210BCS cnpe \ Result >=0 (BA4D)
 4220TXA (F9EA)
 4230ADC #(bufend-bufstt)MOD&100 \ (C=0) (1EC8)
 4240TAX (3AB8)
 4250TYA (5F67)
 4260ADC #(bufend-bufstt)DIV&100 (8182)
 4270TAY (E6AC)
 4280  (6C9C)
 4290REM * RESTORE STATUS REGISTER FROM STACK (ADB
0)
 4300.cnpe (D30B)
 4310PLP (4DC9)
 4320  (139D)
 4330REM * IF NUMBER OF ITEMS IN BUFFER REQUIRED,
EXIT WITH X,Y = RESULT (E42B)
 4340BCC exitj \ C=0 (C753)
 4350  (241F)
 4360REM * AMOUNT OF FREE SPACE REQUIRED (A25A)
 4370REM Save status register on stack (EDF2)
 4380PHP (B19F)
 4390REM Make X,Y = free space, = buffer length -
number of items in buffer -1 (98D1)
 4400STX work1 (B710)
 4410STY work2 (CE55)
 4420CLC (B192)
 4430LDA #(bufend-bufstt)MOD&100 (7226)
 4440SBC work1 (50D2)
 4450TAX (52F4)
 4460LDA #(bufend-bufstt)DIV&100 (0B32)
 4470SBC work2 (62B1)
 4480TAY (5722)
 4490REM Exit via exith, with X,Y = result (FD12)
 4500JMP exith (C89D)
 4510  (9A09)
 4520REM *** BUFFER INSERT (0288)
 4530  (2E92)
 4540REM ** PRESERVE STATUS REGISTER ON STACK (81D
C)
 4550  (D37C)
 4560.ins (2ED2)
 4570PHP (0011)
 4580  (C28F)
 4590REM ** IF PRINTER BUFFER NOT SPECIFIED: RESTO
RE STATUS REGISTER FROM STACK; TRANSFER TO OLD VEC
TOR (F137)
 4600  (BBF5)
 4610CPX #3 (3D42)
 4620BEQ insb \ Buffer number =3 (4704)
 4630PLP (B7F2)
 4640JMP (olinsv) (5238)
 4650  (38EC)
 4660REM ** DISABLE INTERRUPTS (AB39)
 4670  (8C77)
 4680.insb (C808)
 4690SEI (0B3E)
 4700  (E285)
 4710REM ** MAKE (&FA,B)= BUFFER INSERT ADDRESS (B
C84)
 4720  (561E)
 4730LDX insadr (CA61)
 4740STX &FA (FD5E)
 4750LDY insadr+1 (17C6)
 4760STY &FB (AAD9)
 4770  (D507)
 4780REM ** MAKE X,Y = BUFFER INSERT ADDRESS +1 (A
9CF)
 4790  (BA03)
 4800INX (EA5F)
 4810BNE inse (208E)
 4820INY (8247)
 4830  (B967)
 4840REM ** IF X,Y NOW EXCEEDS BUFFER END, MAKE X,
Y = BUFFER START (8A55)
 4850  (4489)
 4860.inse (785E)
 4870CPX #bufend MOD&100 (C352)
 4880BNE insf (FA1C)
 4890CPY #bufend DIV&100 (46D9)
 4900BNE insf (2430)
 4910LDX #bufstt MOD&100 (E04A)
 4920LDY #bufstt DIV&100 (A0B8)
 4930  (E017)
 4940REM ** IF X,Y NOW = BUFFER REMOVE ADDRESS, BU
FFER IS ALREADY FULL (FF96)
```

CONTINUED OVER

```
 4950  (1DF9)
 4960.insf (4E6D)
 4970CPX remadr (713B)
 4980BNE insg (0BFC)
 4990CPY remadr+1 (4727)
 5000BNE insg (091A)
 5010  (82CF)
 5020REM * RESTORE X AS BUFFER NUMBER, =3 (539F)
 5030.exitm (64D7)
 5040LDX #3 (646D)
 5050  (CBBA)
 5060REM * RESTORE STATUS REGISTER FROM STACK (B6E
B)
 5070PLP (3327)
 5080  (DA49)
 5090REM * FLAG FAILURE (76D6)
 5100SEC (ED43)
 5110  (DBBF)
 5120REM * EXIT (BF12)
 5130RTS (C329)
 5140  (58A6)
 5150REM ** BUFFER NOT FULL (E29B)
 5160  (EC3D)
 5170REM * UPDATE BUFFER INSERT ADDRESS, = X,Y (0D
F9)
 5180.insg (A861)
 5190STX insadr (6364)
 5200STY insadr+1 (7EA7)
 5210  (302F)
 5220REM * INSERT BYTE INTO BUFFER AT ORIGINAL INS
ERT ADDRESS, =(&FA,B) (527F)
 5230LDY #0 (0352)
 5240STA (&FA),Y (6695)
 5250  (795A)
 5260REM * RESTORE X AS BUFFER NUMBER, =3 (BA0C)
 5270.exitp (5BA0)
 5280LDX #3 (6B8E)
 5290  (A2C5)
 5300REM * RESTORE STATUS REGISTER FROM STACK (60D
9)
 5310PLP (85EA)
 5320  (17A8)
 5330REM * FLAG SUCCESS (BA88)
 5340CLC (58FC)
 5350  (202A)
 5360REM * EXIT (4DF4)
 5370RTS (185F)
 5380  (31D9)
 5390REM *** BUFFER REMOVE (75F4)
 5400  (0D20)
 5410REM ** PRESERVE STATUS REGISTER ON STACK (FC6
4)
 5420  (B9BB)
 5430.rem (1FC4)
 5440PHP (1185)
 5450  (8E39)
 5460REM ** IF PRINTER BUFFER NOT SPECIFIED: RESTO
RE STATUS REGISTER FROM STACK; TRANSFER TO OLD VEC
TOR (9F14)
 5470  (3AA2)
 5480CPX #3 (AA14)
 5490BEQ remb \ Buffer number =3 (C348)
 5500PLP (EDA6)
 5510JMP (olremv) (7061)
 5520  (E0CB)
 5530REM ** DISABLE INTERRUPTS (E329)
 5540  (1D25)
 5550.remb (1EEF)
 5560SEI (C0F3)
 5570  (63D2)
 5580REM ** MAKE X,Y = BUFFER REMOVE ADDRESS (D5EE
)
 5590  (0CD6)
 5600LDX remadr (A6C8)
 5610LDY remadr+1 (6DA5)
 5620  (0B5B)
 5630REM ** IF BUFFER REMOVE ADDRESS = BUFFER INSE
RT ADDRESS, BUFFER IS ALREADY EMPTY: EXIT VIA exit
m (4AF5)
 5640  (F6B5)
 5650CPX insadr (144C)
 5660BNE remc (95ED)
 5670CPY insadr+1 (98EE)
 5680BEQ exitm (BEC0)
 5690  (E746)
 5700REM ** MAKE A= CONTENTS OF BUFFER REMOVE ADDR
ESS (C19C)
 5710  (2CDC)
 5720.remc (AB2F)
 5730STX &FA (C8CB)
 5740STY &FB (F834)
 5750LDY #0 (B25D)
 5760LDA (&FA),Y (83AC)
 5770  (D132)
 5780REM ** IF EXAMINE ONLY REQUIRED, EXIT VIA exi
tt (8741)
 5790  (BE36)
 5800BVS exitt \ V=1 (A09E)
 5810  (09C9)
 5820REM ** REMOVE BYTE FROM BUFFER (0D92)
 5830  (BD52)
 5840REM * UPDATE BUFFER REMOVE ADDRESS (E707)
 5850REM Make X,Y = buffer remove address +1 (396C
)
 5860LDY &FB (9A24)
 5870INX (4E9D)
 5880BNE remd (4E60)
 5890INY (B71C)
 5900REM If X,Y now exceeds buffer end, make X,Y =
 buffer start (F599)
 5910.remd (1381)
 5920CPX #bufend MOD&100 (015F)
 5930BNE reme (4E24)
 5940CPY #bufend DIV&100 (221B)
 5950BNE reme (9231)
 5960LDX #bufstt MOD&100 (5FBE)
 5970LDY #bufstt DIV&100 (7EEF)
 5980REM Make buffer remove address = X,Y (9765)
 5990.reme (B350)
 6000STX remadr (F640)
 6010STY remadr+1 (3903)
 6020  (F067)
 6030REM * IF BUFFER REMOVE ADDRESS <> BUFFER INSE
RT ADDRESS, EXIT VIA exitp, WITH Y= BYTE REMOVED (
2243)
 6040  (0D89)
 6050CPX insadr (A6E4)
 6060BNE exitt (F727)
 6070CPY insadr+1 (C075)
 6080BEQ remg (832E)
 6090.exitt (8688)
 6100TAY (5468)
 6110JMP exitp (8D35)
 6120  (A917)
 6130REM * BUFFER NOW EMPTY (305A)
 6140REM Preserve byte removed on stack (3756)
 6150.remg (06D2)
 6160PHA (F76D)
 6170REM Generate output buffer empty event (D79D)
 6180LDX #3 \ Buffer number =3 (FC57)
 6190LDY #0 \ Event number =0 (1874)
 6200JSR oseven (F38A)
 6210REM Restore byte removed from stack (0D8B)
 6220PLA (408E)
 6230REM Exit via exitt (2FB9)
 6240JMP exitt (6B78)
 6250  (7505)
 6260] (45A7)
 6270NEXT pass (F790)
 6280  (64F6)
 6290REM *** DISPLAY OBJECT CODE CHECKSUM (FDEA)
 6300  (AF6C)
 6310S%=0 (7FB3)
 6320FOR F%=obj TO O%-1 (D820)
 6330S%=S%+?F% (6CC8)
 6340NEXT (AAF8)
 6350PRINT "CHECKSUM = &";~S% (DD2A)
 6360  (5282)
 6370REM *** DISPLAY ADDRESSES FOR *SAVE (CC02)
 6380  (3D86)
 6390PRINT "START = ";~obj (53B4)
 6400PRINT "END = ";~O% (F019)
>
```

Competition

More goodies for Electron owners to win this month.

Have a go at our simple competition and you could soon be plugging a Slogger ROM box with printer interface onto the rear edge connector of your Electron. And Slogger are kindly supplying a bunch of ROM software to get you started.

THE PRIZES

1st Prize:

Rombox with printer interface
Starword wordprocessor
Starstore database

2nd Prize:

Rombox
Starstore database

3rd Prize:

Rombox
Elkman ROM manager

Five runners up prizes of Starmon (machine code monitor) will also be awarded.

We think you'll agree, prizes well worth having a go for. Here's what you must do:

Take a look at the two different screens displaying Electron memory in hexadecimal form and in assembler mnemonics. Don't worry about the validity of the code. Just tell us how *screen two* differs from *screen one*.

Mark the differences or list them clearly on a postcard and send your answers to: **Slogger Competition, A&B Computing, Number One Golden Square, London W1R 3AB.**

```
8000  C9 1 F0 1F 60 EA 60 E  ||||||£||£||
8008  3 42 41 53 49 43 0 28  ||BASIC||(
8010  43 29 31 39 38 33 20 41  C)1983 A
8018  63 6F 72 6E A D 0 0  corn||||||
8020  B8 0 0 A9 84 20 F4 FF  ||||||||  |||
8028  86 6 84 7 A9 83 20 F4  ||||||||  ||
8030  FF 84 18 A2 0 86 1F 8E  ||||||||||||
8038  2 4 8E 3 4 CA 86 23  ||||||||||#
8040  A2 A 8E 0 4 CA 8E 1  ||||||||||||
8048  4 A9 1 25 11 5 D 5  ||||%||||||
8050  E 5 F 5 10 D0 C A9  ||||||||||||
8058  41 85 D A9 52 85 E A9  A||||R||||
8060  57 85 F A9 6 8D 2 2  W||||||||||
8068  A9 EC 8D 3 2 58 4C F8  ||||||||XL||
8070  C2 41 4E 44 80 0 41 42  ||AND||||AB
8078  53 94 0 41 43 53 95 0  S||||ACS||||
8080  41 44 56 41 4C 96 0 41  ADVAL||||A
8088  53 43 97 0 41 53 4E 98  SC||||ASN||
8090  0 41 54 4E 99 0 41 55  ||ATN||||AU
8098  54 4F C6 10 42 47 45 54  TO||||BGET
80A0  9A 1 42 50 55 54 D5 3  ||||BPUT|||
80A8  43 4F 4C 4F 55 52 FB 2  COLOUR|||_
```

Screen 1

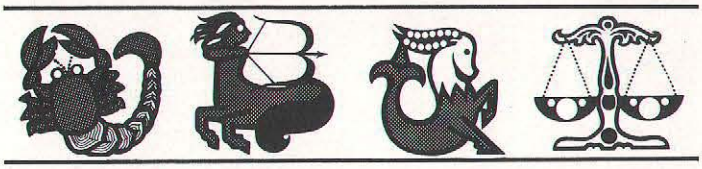
Screen 2

```
8010  C9 1 F0 1F 60 EA 60 E  ||||||£||£||
8008  3 42 41 53 49 43 0 28  ||BASIC||(
8010  43 29 31 39 38 33 20 41  C)1983
8018  63 6F 72 6E A D 0 0  corn||||||
8020  B8 0 0 A9 84 20 F4 FF  ||||||||  |||
8028  86 6 84 7 A9 83 20 F4  ||||||||  ||
8030  FF 84 18 A2  86 1F 8E  ||||||||||||
8038  2 4 8E 3 4 CA 86 23  ||||||||||#
8040  A2 A 8E 0 4 CA 8E 1  ||||||||||||
8048  4 A9 1 25 11 5 D 5  ||||%||||||
8050  E 5 F 5 10 D0 C A9  ||||||||||||
8058  41 85 D A9 55 85 E A9  A||||R||||
8060  57 85 F A9 6 8D 2 2  W||||||||||
8068  A9 EC 8D 3 2 58 4C F8  ||||||||XL||
8070  C5 41 4E 44 80 0 41 42  ||AND||||AB
8078  53 94 0 41 43 53 95 0  S||||ACS||||
8080  41 44 56 41 4C 96 0 41  ADVAL||||A
8088  53 43 97 0 41 53 4E 98  SC||||ASN||
8090  0 41 54 4E 99 0 41 55  ||ATN||||AU
8098  54 4F C6 10 42 47 45 54  TO||||BGET
80A0  9A 1 42 50 55 54 D5 3  ||||BPUT|||
80A8  43 4F 4C 4F 55 52 FB 2  COLOUR|||
```

Fortune Teller

Jonathan and Jane Evans

Look into the future with our horror scope.



This program was written for a fund-raising evening organised by the PTA of our local primary school. It proved so popular that we had to work flat out for two and a half hours to satisfy the constant queue of fee paying customers — mostly, but not exclusively, children. Hopefully other schools can benefit similarly from publication of this program.

So what does it do? It5 generates a personal printout on a dot-matrix printer (see Figure 1) consisting of a drawing of a bat, some accurate calculations and some random nonsense! The computer's "horror scope" is, of course, strictly fgor fun and makes no use of astrological theory or any other doctrine for forecasting the future.

OPERATION

The program needs to be set up by the person operating it before exposure to the customers. When first run it asks whether the printer is on. The program is designed to be used with a printer, since it's the printout which is being sold, but we have also given a non-printer option for testing the program — thus saving a lot of paper!

The main difference is that when the printer is not selected the bat drawing is scaled to 40 columns so that its shape may be checked on screen, whereas an 80 column printout is used on the printer. (Since all text is printed double width on the printer a 40 column display is always generated on the screen.) On the non-printer option one has to press any key to move from the bat to the text and from the text to the next turn. When the printer is selected these GETs are discarded to save time. Time was also a consideration in drawing the bat as text, made up of *s, rather than using the printer's graphic mode.

After choosing whether or not to use the printer, the operator is then required to enter the day's date. This information is used as the basis for several calculations in the production of the printouts.

The program now proceeds to its main loop in which customer details are entered — name, age and birthday — and printouts generated. The program can only be ended by escape action which has been changed to CONTROL-@ to avoid accidents, On escaping, the number of customers served is printed out for the record.

PRINTOUT

When run the program prints out the following text: (i) the name of the customer (as supplied), (ii) his/her birthdate — the year of birth is computed from the information in the program about day and month of birth, age and current date, (iii) birth sign (computed), (iv) lucky number and lucky colour (generated randomly), (v) number of days alive (computed) and (vi) a text message generated by random combinations of three elements.

A particularly popular feature proved to be the calculation of the number of days lived — many children returned to enter the details of teachers and ageing grandparents! The nature and number of text messages can easily be altered to suit requirements (see below under Program Structure).

We included some personal references to leading members of staff in the school which were dropped for obvious reasons from the version presented here. Before looking at the program structure we must first make some comments about the use of a printer.

PRINTER FEATURES

Our school does not have a printer so I lent them my own which was used to test the program, a Shinwa CP-80. The Shinwa uses most of the Epson type codes which are standard on a large range of dot-matrix printers, but in case of problems we will note the special features used.

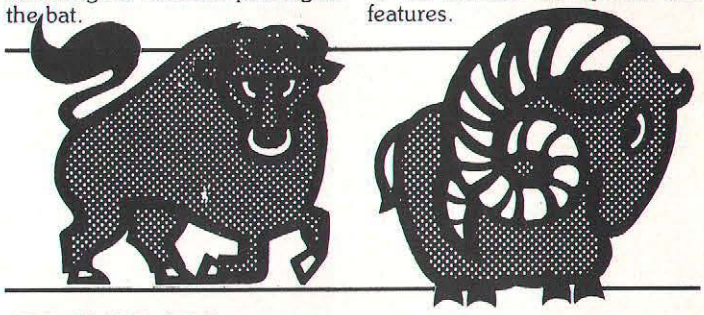
At line 1020 the sequence VDU2,1,14 enables the printer and sends code 14 which means double width until end of line for the heading HORROR * SCOPE. Thus double width is cancelled automatically for the following 80 columns printing of the bat.

At line 1070 the sequence VDU1,27,1,87,1,1 sends the control codes 27,87,1 to the printer (for the uninitiated the BBC requires each printer code to be preceded by VDU1 so that it is not confused with normal control codes by the operating system). Unlike code 14 these enable double width printing until cancelled.

Lines 1490 and 1510 switch italics on and off either side of the procedure call which prints the text message, using codes 27,52 and 27,53 respectively. Since double width is still enabled we get double width italic printing.

Finally, the sequence 27,87,0 at line 1170 is used to cancel double width printing at the end of the cycle.

If you are using a daisy-wheel printer then the bat will still draw since it uses text rather than graphics mode, but you will have to do without the special text features.



PROGRAM STRUCTURE

10-90	Initialisation. The double width heading printed by PROCheader is preserved thereafter by defining the remainder of the screen as a text window at line 90.
100-140	Allows operator to choose printer option and enter current date.
150-200	Main program loop.
230-250	PROCheader. Prints the double width heading for VDU display.
270-400	PROCquestion. Permits entry of details for each customer. These are best typed in by the operator, but the VDU display speeds up collection of the information.
420-630	Data used to print out the picture of the bat.
650-710	PROCdrawbat. Scales display to 40 or 80 columns depending upon state of the Boolean variable "printer" and draws the bat.
730-860	PROCinit. This procedure is called before the main loop starts. It dimensions text arrays to hold the name of lucky colours, signs of the zodiac and

Checked programs can be verified automatically on entry by using the A&B Checker machine code routine. This is published every few months. It is also available on tape for £1 (+ 20p post/packing). Cheques/ postal orders made payable to Phoenix Software. Just 'RUN the program. On entering each line, it is automatically confirmed right or wrong by the four digit number in brackets. A&B makes listings easy to get right.

months of the year and an integer array date% which contains the date boundaries between birth signs. the data is then read into the arrays.

880-910 PROCcompute. Computes birth sign and year of birth.

1000-1190 PROCprintfortune. Prints heading for printer, calls PROCdrawbat, prints name, birth sign etc and then calls PROCprinttext to print out the randomised message.

1210-1340 PROCprint. A useful general procedure for your library. The recursive use of INSTR to find spaces is the fastest means I have found of separating words and printing out text without splitting them across line. For use with a 20 or 80 column mode you need only change the value of the line width counter T% to 19 or 79 respectively.

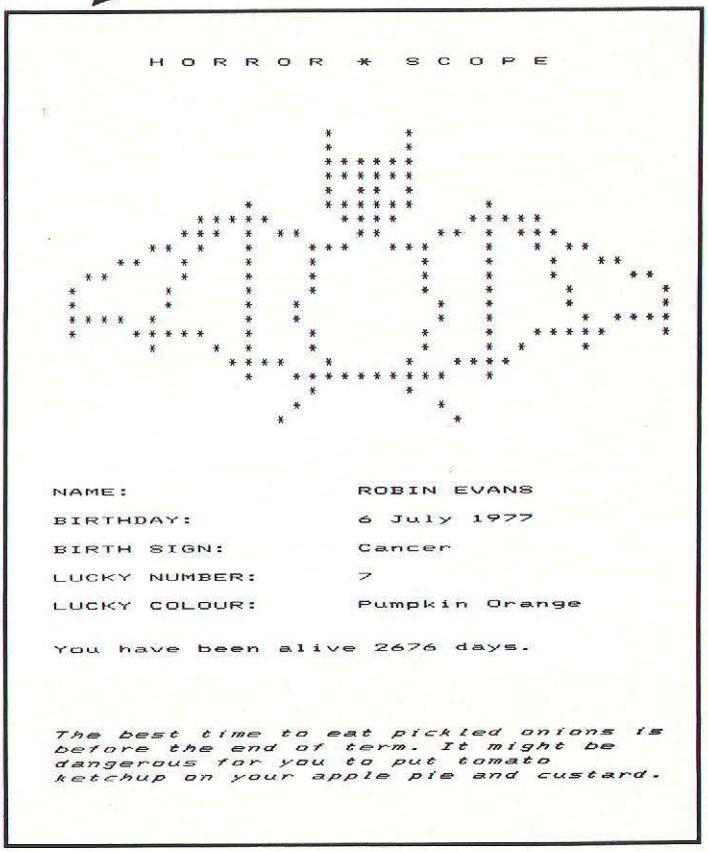
1360-1520 PROCinittext. Run before main loop (called by PROCinit). This procedure dimensions text arrays to hold the first, second and third parts of the messages which are then read into the arrays.

1440-1520 PROCprinttext. Chooses a random first, second and third component for the message from those available in the arrays. Composes the message into a single string variable S$ and calls PROCprint to print it out.

1540-1730 PROCcomputedaysalive,FNmonth,FNyear. Computes the number of days the person has lived, allowing for leap years etc. Programmer might like to try to work out their own algorithm before studying the one we came up with.

1750-1840 Data for messages. The commas separate each part of the message. If you wish to add to the list, study the grammatical structure carefully since the second part both finishes and starts a sentence and any random combination of parts from any line must be grammatical. If you alter the number of messages then the value of ns% at line 1380 must be changed.

1860-1890 PROCerror. When escape is pressed by CONTROL-@ this prints out the number of turns before ending. Type this in early since it also mends the escape key and will report any errors due to typing mistakes.



THIS PROGRAM HAS BEEN

CHECKED!

PROGRAM LISTING

```
10REM HORROR SCOPE (5508)
20REM By Jonathan and Jane Evans (5350)
30REM (461C)
40*FX220,0 (0750)
50ON ERROR PROCerror (A302)
60X=RND(-TIME) (9CED)
70PROCinit (4214)
```

CONTINUED OVER

PROGRAM LISTING

```
  80MODE7:PROCheader (7223)
  90VDU28,0,24,39,2 (EFAD)
 100CLS:PRINT''"Printer on (Y/N) ?"; (8DDC)
 110*FX15,1 (4E6F)
 120REPEAT:G$=GET$:UNTIL G$="Y" OR G$="N":IF G$="
Y" THEN PRINT"Y":printer=TRUE ELSE PRINT"N":printe
r=FALSE (EDEA)
 130K%=0:REM COUNTER FOR NUMBER OF TIMES RUN (58B
9)
 140INPUT''"Today's date"''"    Day";tday%:INPUT'"
  Month",tmonth%:INPUT'"    Year",tyear% (68CB)
 150REPEAT (7287)
 160PROCquestion (5BF4)
 170PROCcompute (47AF)
 180PROCprintfortune (196D)
 190K%=K%+1 (CC16)
 200UNTIL FALSE (C145)
 210: (6F85)
 220: (8415)
 230DEF PROCheader (4562)
 240FOR I=1 TO 2:VDU141,129,157,130:PRINT SPC9;"H
ORROR * SCOPE":NEXT I (287B)
 250ENDPROC (1DD5)
 260: (C196)
 270DEF PROCquestion (CA3A)
 280REPEAT:CLS (C1F6)
 290INPUT''"Please enter your name"''Name$ (5F7C)
 300IF LEN(Name$)>20 THEN PRINT '"Maximum length
20 characters":GOTO 290 (BB95)
 310INPUT'"Enter your age in years"''age%:IF age%
<1 OR age%>99 THEN PRINT'"Don't be silly !":GOTO 3
10 (10B8)
 320PRINT'"Enter your birthdate" (1B77)
 330INPUT'"Day (1-31) ",day%:IF day%>31 OR day%<1
 GOTO 330 (0E12)
 340INPUT'"Month (1-12) ",month%:IF month%<1 OR m
onth%>12 GOTO 340 (5F1D)
 350PRINT''"Are these entries correct (Y/N) ?" (6
E5B)
 360*FX15,1 (E1AA)
 370REPEAT G$=GET$:UNTIL G$="Y" OR G$="N" (E609)
 380IF day%=tday% AND month%=tmonth% PRINT'"HAPPY
 BIRTHDAY !!!":WAIT=INKEY(300) (29DB)
 390UNTIL G$="Y" (CDC8)
 400ENDPROC (A6B8)
 410: (773B)
 420REM DATA FOR BAT (B755)
 430DATA 2,18,23 (A1F9)
 440DATA 2,18,23 (F335)
 450DATA 6,18,19,20,21,22,23 (A3E0)
 460DATA 6,18,19,20,21,22,23 (6E68)
 470DATA 4,18,20,21,23 (DCF6)
 480DATA 8,13,18,19,20,21,22,23,28 (A94E)
 490DATA 14,10,11,12,13,14,19,20,21,22,27,28,29,3
0,31 (C746)
 500DATA 14,9,10,11,13,15,16,20,21,25,26,28,30,31
,32 (9C86)
 510DATA 14,7,8,10,13,17,18,19,22,23,24,28,31,33,
34 (A718)
 520DATA 10,5,6,9,13,17,24,28,32,35,36 (5033)
 530DATA 10,3,4,9,13,17,24,28,32,37,38 (3F70)
 540DATA 8,2,8,13,17,24,28,33,39 (B75F)
 550DATA 8,2,8,13,16,25,28,33,39 (ADE9)
 560DATA 14,2,3,4,5,7,13,16,25,28,34,36,37,38,39
(7D44)
 570DATA 16,2,6,7,8,9,10,13,17,24,28,31,32,33,34,
35,39 (E0DF)
 580DATA 8,7,11,13,17,24,28,30,34 (6AC3)
 590DATA 10,12,13,14,15,18,23,26,27,28,29 (B2DC)
 600DATA 12,13,16,17,18,19,20,21,22,23,24,25,28 (
7067)
 610DATA 2,17,23 (2CC2)
 620DATA 2,16,25 (6E96)
 630DATA 2,15,26 (7F55)
 640: (63A2)
 650DEF PROCdrawbat (339C)
 660RESTORE 430:IF printer THEN scale =2 ELSE sca
le=1 (5D05)
 670FOR Y%=0 TO 20 (3A3F)
 680READ N% (2EBC)
 690FOR I%=1 TO N%:READ X%:PRINT TAB(scale*(X%-1)
);"*";:NEXT I% (B414)
 700PRINT:NEXT Y% (2A29)
 710ENDPROC (1934)
 720: (90F4)
 730DEF PROCinit (681A)
 740ncol%=10 (C552)
 750 DIM colour$(ncol%),zodiac$(12),zdate%(12,2),
month$(12) (2C56)
 760RESTORE 810 (214D)
 770FOR I%=1 TO ncol%:READ colour$(I%):NEXT I% (5
4F9)
 780FOR I%=1 TO 12:READ zodiac$(I%):NEXT I% (93E2
)
 790FOR I%=1 TO 12:FOR J%=1 TO 2:READ zdate%(I%,J
%):NEXT J%,I% (B272)
 800FOR I%=1 TO 12:READ month$(I%):NEXT I% (90E2)
 810DATA Blood Red,Sky Blue Pink,Punk Pink,Slime
Green,Mellow Yellow,Pumpkin Orange,Snow White,Midn
ight Blue,Sooty Black,Murky Purple (F954)
 820DATA Aries,Taurus,Gemini,Cancer,Leo,Virgo,Lib
ra,Scorpio,Sagittarius,Capricorn,Aquarius,Pisces (
8E17)
 830DATA 22,3,21,4,22,5,23,6,24,7,24,8,24,9,24,10
,23,11,23,12,20,1,20,2 (6463)
 840DATA January,February,March,April,May,June,Ju
ly,August,September,October,November,December (916
8)
 850PROCinittext (3DB2)
 860ENDPROC (9B40)
 870: (B124)
 880DEF PROCcompute (003A)
 890col$=colour$(RND(ncol%)):num%=RND(9) (753A)
 900REM COMPUTE BIRTHSIGN (7642)
 910FOR I%=1 TO 12:IF month%=zdate%(I%,2) THEN si
gn%=I% (6E6C)
 920NEXT I% (A41D)
 930IF day%<zdate%(sign%,1) THEN sign%=sign%-1 (C
AC9)
 940IF sign%=0 THEN sign%=12 (5B37)
 950REM COMPUTE YEAR OF BIRTH (8839)
 960IF month%<tmonth% THEN byear%=tyear%-age% ELS
```

```
E IF month%=tmonth% AND day%<=tday% THEN byear%=ty
ear%-age% ELSE byear%=tyear%-age%-1 (7CB5)
  970PROCcomputedaysalive (7199)
  980ENDPROC (C6B7)
  990: (C974)
 1000DEF PROCprintfortune (05DF)
 1010CLS (1699)
 1020IF printer THEN  VDU2,1,14 (5943)
 1030PRINT"      H O R R O R  *  S C O P E"''' (9
9FE)
 1040PROCdrawbat (C7A6)
 1050*FX15,1 (87E4)
 1060IF NOT printer THEN G=GET (3BFF)
 1070IF printer VDU1,27,1,87,1,1 (10E2)
 1080PRINT''''"NAME:                ";Name$ (109B)
 1090PRINT''"BIRTHDAY:           ";day%;" ";month$(m
onth%);" ";byear% (EBB1)
 1100PRINT''"BIRTH SIGN:         ";zodiac$(sign%) (F
BB2)
 1110PRINT''"LUCKY NUMBER:       ";num% (B176)
 1120PRINT''"LUCKY COLOUR:       ";col$ (D754)
 1130PRINT''"You have been alive ";alive%;" days."
 (A82A)
 1140PROCprinttext (42FE)
 1150*FX15,1 (323C)
 1160IF NOT printer G=GET (3539)
 1170IF printer VDU1,27,1,87,1,0,12 (885C)
 1180VDU3,15 (297A)
 1190ENDPROC (D213)
 1200: (841B)
 1210DEF PROCprint(S$) (D941)
 1220REM PRINTS WITHOUT SPLITTING WORDS  (7576)
 1230LOCAL K%,S%,L%,M%,T% (8630)
 1240T%=39 (68AD)
 1250K%=0 (C269)
 1260REPEAT (DF8E)
 1270S%=INSTR(S$," ") (063C)
 1280K%=K%+S%:IF K%>T% THEN PRINT:K%=S% (86F4)
 1290IF S%>0 THEN PRINT LEFT$(S$,S%); (3C29)
 1300S$=RIGHT$(S$,LEN(S$)-S%) (6B7C)
 1310UNTIL S%=0 (B0C1)
 1320L%=LEN(S$):K%=K%+L%:IF K%>T% THEN PRINT (37EC
)
 1330PRINT S$ (665B)
 1340ENDPROC (B326)
 1350: (9CDD)
 1360DEF PROCinittext (C1F2)
 1370RESTORE 1750 (B937)
 1380ns%=10:DIM first$(ns%),second$(ns%),third$(ns
%) (5F4F)
 1390FOR I%=1 TO ns% (4292)
 1400READ first$(I%),second$(I%),third$(I%) (9C75)
 1410NEXT I% (11C9)
 1420ENDPROC (7F7A)
 1430: (7735)
 1440DEF PROCprinttext (2FCC)
 1450LOCAL P1%,P2%,P3%,S$ (A060)
 1460P1%=RND(ns%):P2%=RND(ns%):P3%=RND(ns%) (7853)
 1470S$=first$(P1%)+" "+second$(P2%)+" "+third$(P3
%) (6ED1)
 1480PRINT''''' (62B3)
 1490VDU1,27,1,52      (CA7B)
 1500PROCprint(S$) (CC95)
 1510VDU1,27,1,53 (7442)
 1520ENDPROC (CAA2)
 1530: (7300)
 1540DEF PROCcomputedaysalive (B707)
 1550LOCAL D%,M%,Y% (D6CA)
 1560alive%=FNmonth(month%,byear%)-day% (4176)
 1570IF month%<12 THEN FOR M%=month%+1 TO 12:alive
%=alive%+FNmonth(M%,byear%):NEXT M% (FB48)
 1580IF tyear%-byear%<2 THEN 1620 (05E5)
 1590FOR Y%=byear%+1 TO tyear%-1 (CE2B)
 1600alive%=alive%+FNyear(Y%) (ADFB)
 1610NEXT Y% (5FAD)
 1620IF tmonth%>1 THEN FOR M%=1 TO tmonth%-1: aliv
e%=alive%+FNmonth(M%,tyear%):NEXT M% (7DD8)
 1630alive%=alive%+tday% (0E72)
 1640ENDPROC (9118)
 1650: (883C)
 1660DEF FNmonth(M%,Y%) (BE36)
 1670ON M% GOTO 1680,1690,1680,1700,1680,1700,1680
,1680,1700,1680,1700,1680 (3D60)
 1680=31 (C68E)
 1690IF Y% MOD 4 = 0 THEN =29 ELSE =28 (7737)
 1700=30 (C202)
 1710: (C98A)
 1720DEF FNyear(Y%) (4404)
 1730IF Y%MOD4 =0 THEN =366 ELSE =365 (7255)
 1740: (D579)
 1750DATA Try to clean your teeth,on wet Tuesdays.
 Never ever,run across the road without looking. (
D166)
 1760DATA The best time to eat pickled onions is,d
uring half-term. It is very unlucky for you to,pic
k a fight with an angry lobster. (3458)
 1770DATA A good time to watch television is,every
 other Saturday in November. You should try never
to,eat crisps with a knife and fork. (C634)
 1780DATA A long walk in the fresh air will do you
 good,every Bank Holiday. On no account should you
,wear more than one pair of shoes at a time. (5383
)
 1790DATA A lucky time for you to stand on your he
ad might be,next Boxing Day. Don't forget to remem
ber that it would be terrible for you to,eat too m
any liquorice allsorts on your birthday. (B31F)
 1800DATA Don't forget to brush your hair twenty t
imes,every Monday morning. Be careful never to,go
into a field with an angry bull. (8FA5)
 1810DATA You could learn to tap dance,next Thursd
ay week. However it wouldn't be a good idea to,tur
n cartwheels in the playground in a thunderstorm.
(28E0)
 1820 DATA Learn to spell supercalifragilisticexpi
alidocious,before the end of term. It might be dan
gerous for you to,put tomato ketchup on your apple
 pie and custard. (7EB3)
 1830DATA A good time to eat marmalade sandwiches
with kippers would be,next Friday. Remember never
to,wrestle with a man eating tiger. (0CCA)
 1840DATA Eat up all your cabbage,on Sundays in De
cember when it rains. It would be a good idea if y
ou didn't,leave your chewing gum on your teacher's
 chair. (228F)
 1850: (B12A)
 1860DEF PROCerror (A98F)
 1870*FX220,27 (8358)
 1880IF ERR=17 CLS:PRINT''"Number of turns = ";K%:
END (8BF0)
 1890CLS:PRINT':REPORT:PRINT" at line ";ERL:END (A
297)
```

A&B COMPUTING

FOR USERS OF THE BBC MICRO AND ELECTRON

WHY YOU SHOULD TAKE OUT A SUBSCRIPTION TO A&B COMPUTING:

The magazine is totally commited to supporting users of the BBC Micro Model B, B+ and Acorn Electron computers.

Our broad coverage of events, personalities and product releases is complemented by authoritative reviews in the specialist areas of education and business. The first to explain Level 3 Econet, the first to review Wordwise Plus, the first with Acornsoft Graphics Extension ROM, Viewstore, Solidisc Winchesters, Intersheet and Interchart and so it goes on.

We supply tutorial on programming in BBC BASIC, Logo, and Pascal, reporting how these languages are used in education and other applications.

We don't stop with tutorial. Features on artificial intelligence, computer art, computers and natural language; all have filled the mailbag with response from interested readers. For which we are grateful.

If an important new product appears on the market, A&B is first to tell you all about it. The BBC 128K was launched at the PCW show in early September. By the end of that month we had a full breakdown on *why, how* and *how fast.* Plus news of Le Modem and Challenger.

We do comparative reviews, a dozen printers side by side, accounts software tabulated and explained, *all* the double density filing systems benchmarked and compared.

Nor are we averse to having some fun. We interviewed Commander Glew, one of the first *Elite* , played Combat Lynx, Brian Jacks' Superstar Challenge, Boffin, Beach-Head and Deathstar on the *Electron* as well as the BBC, mapped Atic Atac and kicked around Ocean's Matchday.

Then there's the software we publish every month. It's designed to be useful, to support peripherals by filling gaps in available utilities, to explore areas of programming not normally available at your local computer store. And Checker makes sure it all works for you or it's available on tape or disc if you prefer.

Just a few reasons why it's important to you, as a BBC or Electron user, not to miss a single month of A&B. A back issue may not be available.

MISS A&B MISS OUT – DON'T DELAY, SUBSCRIBE TODAY!

Subscription Order Form
Cut out and SEND TO:

A&B Computing
Infonet Ltd
Times House
179 The Marlowes
Hemel Hempstead
Herts HP1 1BB
(tel: 0442 48435)

Please commence my subscription with the very next issue.

SUBSCRIPTION RATES		(tick □ as appropriate)
£21.80	for 12 issues UK.	☐
£26.00	for 12 issues overseas surface mail.	☐
U.S.A. $34.50		☐
£53.00	for 12 issues overseas air mail.	☐

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 .....

Ultracalc Printer Driver

D.Gibbons

Ultracalc needs a printer driver. A&B provides one.

This simple machine code printer driver, run from disc by the command ***VDU**, allows printer codes (in numeric or alphabetic form) to be sent to a printer as if the routine were part of Ultracalc, thus avoiding loss of data caused by a temporary return to BASIC.

Many reviewers have extolled the virtues of Ultracalc, and it is indeed an extremely powerful and versatile spreadsheet program. With its individually variable column widths, relative and absolute replication, conditional instructions, and a "look-up" facility, it forms a very attractive package. It does, however, suffer from one important drawback; it is so easy to enter a large quantity of data, only to realise that the printer has not been set up in a suitable print style, page length, line length, etc. Ultracalc itself offers no means of issuing printer control codes, so it is necessary to exit to BASIC using the /*BASIC command, and then use the conventional VDU command to send the necessary codes.

Unfortunately, on returning to Ultracalc using *CALC, it is annoying to find that all the data has apparently been lost. The designers of Ultracalc do not appear to have provided a "warm-start" option and, although data can obviously be saved to disc before exiting to BASIC, it is all too easy to forget and, in any case, saving and reloading even a moderate sized sheet can take several minutes.

Fortunately, Ultracalc allows the use of system * commands, in the form /*DRIVE 2, /*CAT, or indeed the /*BASIC command already mentioned. When the majority of such commands (but not /*BASIC, for fairly obvious reasons) are issued from within Ultracalc, control returns to Ultracalc on completion of the task.

Provided the task has not over - written user memory, the spreadsheet data is retained intact because it has not been necessary to issue an *CALC command. In fact, because of the way the "*" commands are handled, it is as if Ultracalc had never relinquished control.

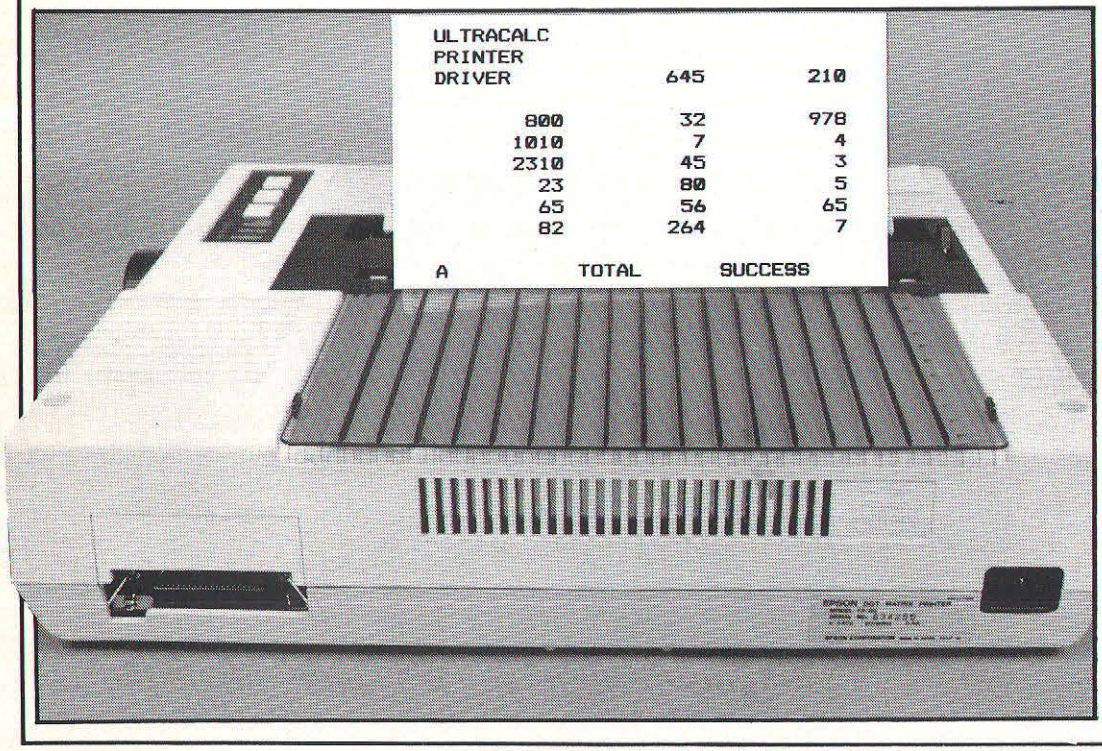
Hence, on a disc system, it is possible to use the command /*PROG to rapidly load and run a machine code program, in an unused area of memory such as the cassette buffer, and then return automatically to Ultracalc on completion.

THE CODE

The short program called VDU, presented here, is just such a program. It has been designed to fit into a single page of memory and loads at &900, although &A00 could equally have been used. As the command /*VDU reloads the program each time it is used, it does not matter if page &9 is used for other purposes in between its use for communication with the printer.

In operation, the program prompts for a string of VDU codes (eg. 27,82,2),which may in fact be separated by commas or spaces, expands these to the required format (ie.2,1,27,1,82,1,2,3), sends these to the VDU drivers using the system routine OSWRCH at &FFEE, and then returns to the calling environment ie. Ultracalc.

For the benefit of users with printer codes tabulated alphabetically, rather than numerically, an automatic option is built into the program making 27,P,3 just as acceptable as 27,80,3; similarly, 27,4 is an alternative to 27,52. This is achieved by checking the number of digits in the first code following an escape (27) code. If there is only one digit, it is assumed to be an ASCII character and is treated as such; if there are two or more digits, these are combined together decimally into a single byte, as are any other subsequent codes until another escape code is encountered.



DOCUMENTATION

The program is reasonably well annotated, but a more detailed explanation follows for the benefit of those unfamiliar with machine code:-

Line 70 makes sure that the printer is switched off so that the initial dialogue is not transmitted to the printer. This is not strictly necessary when used with Ultracalc, but as explained later the program has uses outside Ultracalc.

Lines 80-100 display the prompt "VDU CODES: " using the data string generated by the EQUS statement at line 400. As noted in the program, BASIC-I users must use an alternative form for line 400. In either case, the prompt could be changed according to the user's fancy; even teletext colour control codes could be included by using SHIFTed function keys, as Ultracalc operates in mode 7. BASIC-I users must then ensure that that the P%=P%+11 statement in line 490/400 matches the length of the $P% string.

CONTINUED OVER

Lines 110-130 initialise various locations at the end of the program and, in line 140, a single digit is read from the keyboard and displayed on the screen. This digit is multiplied by 10 and added to the previous value (set to 0 initially in line 120) and the multiplication/addition process is repeated until stopped by a comma, a space, or a return. Each digit has initially an ASCII value (48=0; 57=9) and must be converted to the corresponding actual number before being combined with the other digits (line 210). At the same time, location "num" is incremented (line 230) to keep track of the number of individual digits read.

Lines 150-170 test for comma, space or return and, in either case, the status of the escape flag is eventually checked, the only difference being that, for a return, the return flag is first set. If the escape flag is found to be set, the digit counter is checked and, if only one digit is involved (num=0 because originally set to -1), this single digit is converted back to its original corresponding ASCII character, by adding 48 back again (line 270), on the assumption that it was actually a "letter". As far as it has been possible to ascertain, this approach works for all Epson-type printers, including the Star, Shinwa, etc, and although some single digit numeric codes are possible, their use is limited and, in any case, they never seem to follow immediately after an escape code.

Line 280 stores the final value, depending on the outcome of the above, in an indexed buffer at location "buf" (see line 460). The whole process then repeats from line 120 except that, if a return has been encountered, ie. the return flag is set, a byte value of 13 is added to the end of the buffer and the reading process is terminated. A buffer is used to store the byte values so that all characters can be input and manipulated before the printer is enabled. In this way, only the modified characters are sent to the printer.

Line 320 turns the printer on and the above process more or less goes into reverse. Characters are read back from the buffer (line 330) and are passed, unmodified, to the VDU drivers via OSWRCH, but each one is preceded by a CHR$1 (lines 350-360). The return code at the end of the buffer, when encountered, causes this process to be terminated and the printer is then turned off (line 380). For some printers it is apparently desirable to transmit this final return code to empty the printer buffer and leave the printer in a free state and so it is included here (line 370).

Line 390 transmits a CHR$12, equivalent to CTRL-L, to clear the screen. Note that the program jumps to OSWRCH via a JMP instruction, rather than the usual JSR instruction; it is the RTN instruction at the end of the OSWRCH routine in the OS ROM which actually returns control to Ultracalc.

KEYING IN

When the BASIC program is keyed in (omit the comments to simplify this!) and RUN, machine code should be generated terminating at &9C9 leaving room before the end of the page for over 50 bytes of printer codes — more than enough for most purposes. If this is not exactly so, an error has probably occurred and the program must be thoroughly checked. When satisfactory, save the machine code to disc as VDU, or some other suitable name, using *SAVE VDU 900 9FF. It is not really necessary to save the whole of page &9, but there is little to be gained by not doing so.

In operation from within Ultracalc, it is only necessary to issue a /*VDU command and respond to the prompts. The program can, however, be just as useful outside of Ultracalc — enter *VDU in direct mode, or enter CALL &900 if the program has already been used once and page &9 has not been altered in any way since. However, the speed of discs, and the shortness of the program, make the simpler *VDU approach just as effective.

PROGRAM LISTING

```
   10REM VDU - a printer driver for Ultracalc
   20oswrch=&FFEE
   30osrdch=&FFE0
   40FOR PASS=0 TO 3 STEP 3
   50P%=&900     :REM Store code in Cassette Buf
fer
   60[OPT PASS
   70LDA #3:JSR oswrch     \Make sure printer OF
F
   80LDX #0:STA rtn     \Initialise output point
er and return flag
   90.prn LDA vdu,X:BEQ rdy     \Check for zero
byte terminator
  100JSR oswrch:INX:JMP prn     \Print VDU promp
t string
  110.rdy LDX #0:STX esc     \Initialise buffer
pointer and escape flag
  120.fst LDA #0:STA chr     \Initialise multipl
ication store
  130LDA #255:STA num     \Initialise character
counter to -1
  140.rpt JSR osrdch:JSR oswrch     \Input and o
utput one character
  150CMP #44:BEQ sto     \Check for comma
  160CMP #32:BEQ sto     \Check for space
  170CMP #13:BNE mlt     \Check for return
  180STA rtn:JMP sto     \Set return flag
  190.mlt TAY:LDA chr:ASLA:ASLA:ASLA     \Multip
ly old value by 8
  200ADC chr:ADC chr:STA chr     \Add two more (
x10 in all)
  210TYA:SEC:SBC #48:CLC     \Convert ASCII to a
ctual number
  220ADC chr:STA chr     \Add to store contents
  230INC num:JMP rpt     \Increment character co
unter; go get another
  240.sto LDA esc:BEQ dig     \Test escape flag
  250LDA #0:STA esc     \Reset escape flag
  260LDA num:BNE dig     \Test for single charac
ter
  270LDA chr:CLC:ADC #48:STA chr     \Convert ba
ck to ASCII
  280.dig LDA chr:STA buf,X:INX     \Store final
 byte in buffer
  290CMP #27:BNE nsc:STA esc     \Test for escap
e byte
  300.nsc LDA rtn:CMP #13:BNE fst     \Test if r
eturn just entered
  310STA buf,X:LDX #0     \Terminate buffer; ini
tialise pointer
  320LDA #2:JSR oswrch     \Turn printer ON
  330.snd LDA buf,X:INX     \Get byte from buffe
r
  340CMP #13:BEQ dun     \Test for 13 as termina
tor
  350TAY:LDA #1:JSR oswrch     \Precede each cod
e with VDU1
  360TYA:JSR oswrch:JMP snd     \Transmit VDU co
de
  370.dun LDA #13:JSR oswrch
  380LDA #3:JSR oswrch     \Turn printer OFF
  390LDA #12:JMP oswrch     \Clear screen
  400.vdu EQUS "VDU CODES: "     \Initial prompt
 string (see note)
  410.brk BRK     \Zero byte terminator for prom
pt string
  420.chr NOP     \Multiplication store
  430.esc NOP     \Escape flag
  440.num NOP     \Character counter
  450.rtn NOP     \Return flag
  460.buf NOP     \Start of main buffer
  470]NEXT
  480REM BASIC-I users replace line 400 by:-
  490REM .vdu :] $P%="VDU CODES: ":P%=P%+11:[
  500REM Use *SAVE VDU 900 9FF when saving to di
sc.
```

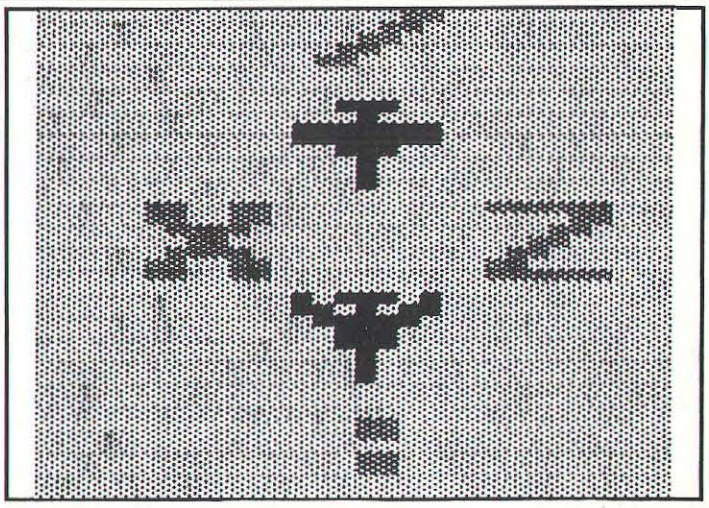
Falcon Pilot

Brian Cassidy

Hi-speed dogfighting in the hitech F111B.

PROGRAM LISTING 1

```
   10REM  F-111B by BRIAN CASSIDY
   20MODE2:VDU19,0,4,0,0,0
   30IF PAGE<>&E00 D%=TRUE ELSE D%=FALSE
   40 VDU23,254,&08,&08,&08,&1C,&1C,&3E,&6B,&5D,23,253,
0,57,125,255,255,125,57,0,23,252,16,56,56,254,254,56,56
,16,23,248,16,56,124,254,254,254,254,254,23,249,254,254
,254,254,254,124,56,16,23,247,8,28,62,62,62,62,28,8
   50 VDU23,240,&08,&08,&08,&1C,&7F,&7F,&08,&1C,23,255,
255,255,255,255,255,255,255,255,23,224,&FF,&FF,&FE,&DC,
&DC,&80,&80,&0,23,225,&FF,&FF,&FF,&F7,&83,&83,&01,&0,23
,226,&FF,&FE,&FE,&7E,&10,&10,&0,&0
   60 VDU 23,227,&7F,&3F,&1F,&1F,&0F,&07,&03,&01,23,230
,&1C,&3E,&7F,&7F,&41,&41,&22,&22,23,231,34,28,28,8,62,8
,20,20
   70GCOL0,3:MOVE 32,991:DRAW 1247,991
   80FOR Q%=2 TO 31:SOUND 1,-15,Q%*4,2
   90VDU4,30,11,5:QQ%=TIME
  100GCOL0,7:MOVE 32,991:DRAW 32,959:MOVE 1247,991:DRAW
 1247,959
  110IF Q%=11 THEN GCOL0,12:MOVE 384,1023:PRINT"B.CASSI
DY"
  120IF Q%=14 THEN GCOL0,2:MOVE 384,1023:PRINT"(C) 1984
"
  130IF Q%=17 THEN GCOL0,6:MOVE 192,1023:PRINT"FOR BBC
MODEL B"
  140IF Q%=20 THEN GCOL0,3:MOVE 448,1023:PRINT"F-111B"
  150REPEAT:UNTIL TIME>QQ%+12
  160NEXT Q%
  170MOVE 32,991:DRAW 1247,991
  180FOR Q=0 TO -15 STEP -0.5:SOUND 0,Q,4,3:NEXT Q
  190CLS
  200VDU19,0,4,0,0,0,5,19,4,0,0,0,0
  210GCOL0,3:PRINT'''"F-111B by ";:GCOL0,5:PRINT"B.CAS
SIDY"'
  220GCOL0,3:VDU248,10,13,249:PRINT" = MERCHANT SHIP  1
"'
  230GCOL0,2:VDU253:PRINT" = BARRAGE BALOON 2"
  240GCOL0,4:VDU247:PRINT" = CORVETTE       4"'
  250GCOL0,7:VDU252:PRINT" = SUBMARINE      7"'
  260GCOL0,6:VDU255,10,13,255,10,13,255,10,13,255:PRINT
" = AIRCRAFT CARRIER"
  270GCOL0,7:MOVE0,850:PLOT5,1279,850
  280MOVE0,910:PLOT5,1279,910
  290MOVE0,311:PLOT5,1279,311
  300MOVE0,292:PRINT" KEYBOARD  JOYSTICK"
  310MOVE640,311:PLOT5,640,32:MOVE0,250:PLOT5,1279,250
  320GCOL0,5:MOVE96,240:PRINT":"
  330GCOL0,3:MOVE96,200:VDU254
  340MOVE96,134:VDU240
  350GCOL0,5:MOVE96,92:PRINT"/"
  360MOVE0,64:PRINT"FIRE=SHIFT"
  370MOVE32,166:PRINT"Z":MOVE160,166:PRINT"X"
  380GCOL0,3:MOVE900,230:VDU254
  390GCOL0,5:MOVE836,180:PRINT"<J>"
  400GCOL0,3:MOVE900,138:VDU240
  410GCOL0,5:MOVE700,64:PRINT"FIRE=FIRE"
  420IF D%=TRUE MOVE 396,96:PRINT"P=PAUSE"
  430IF D%=FALSE MOVE 396,32:PRINT"P=PAUSE" ELSE GCOL0,
12:MOVE 0,32:PRINT"PRESS FIRE TO START":REPEAT UNTIL IN
KEY(-1) OR (ADVAL(0)AND3)<>0
  440VDU 4,28,0,2,19,0
  450IF D%=TRUE THEN PAGE=&1100
  460CHAIN"F-111PG"
```



PILOT BRIEFING

Unfortunately you have been volunteered to fly treacherous missions for the United States Air Force in a specially prepared F-111B, codenamed Falcon. Your aircraft is armed with 32 Vought hypervelocity missiles which must be used wisely to destroy enemy shipping. You can re-arm and refuel the F-111B by landing on United States Navy aircraft carriers. To do this you must first sweep the wings forward for a low and slow approach.

In this configuration, fuel is used up at only half the rate of that of normal flight, but you can collide with enemy shipping. With the wings swept back, you can safely fly over targets except for the barrage balloons which obstruct your flight at all times during these hazardous sorties. You must destroy as much enemy shipping as possible in order to complete your mission and be prompted after landing at your base. You are then issued with progressively upgraded F-111B's until you have attained the skills required by the National Aeronautics and Space Administration for astronaut training.

USING THE PROGRAM

The number of missiles remaining is displayed in red on the left of the screen and the volume of fuel remaining is displayed in magenta on the right. You can control the F-111B from either the keyboard or joysticks. These controls are displayed when the first program is run. Be careful to enter the same name *exactly* each time a particular person plays, so that the computer can keep track of that person's promotions.

PROGRAM NOTES

F-111B consists of two programs. The first displays the scores for a hit on each type of enemy ship as well as the F-111B controls. This program also displays the characters needed for the main program. Because F-111B is in two sections memory is saved so that the game can run on both cassette and disc based machines. (PAGE is automatically set to &1100 before loading the main program onto disc based computers). AUTO can be used to speed typing since both programs have been renumbered. After you have typed in the second program and checked for typing errors remember to remove the REM at the start of line 80.

CONTINUED OVER

Falcon Pilot

FIRST PROGRAM

Lines	
20-60	set up
70-180	title
190-440	instructions
450-460	set PAGE to &1100 if on a disc based computer and CHAIN the main program

SECOND PROGRAM

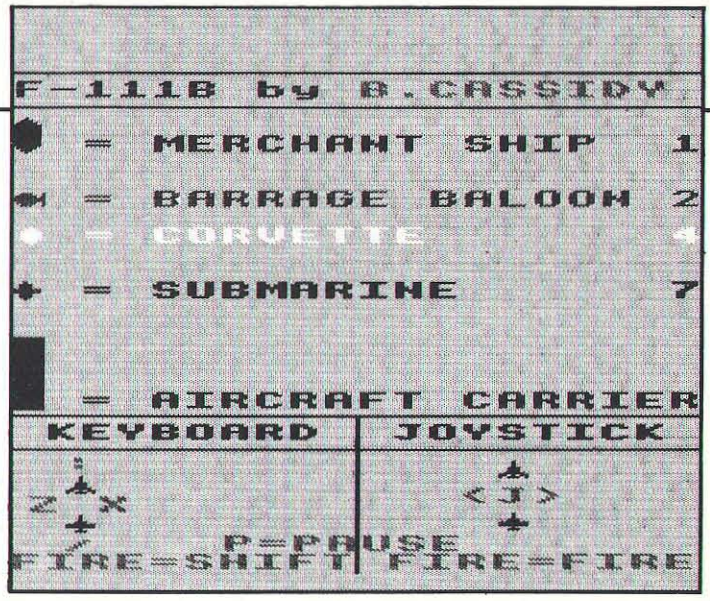
Lines	
20-60	set up
70-150	determine whether player is using the keyboard or joystick
160-190	set up variables and colours
200-650	move F-111B
210-310	print fuel and bomb levels
320-340	pause routine
350	pause routine
360-420	see if F-111B has crashed
430-480	missile firing routine
490-500	see if F-111B has flown off screen
510-610	print on targets
620	check if finished mission
630	slow flight if wings are unswept
640	check if F-111B is out of fuel
660-670	call end of game routine
680-950	landing and promotion routine
1520-1530	data

PROCEDURES

PROCDISPLAY	displays top 10
PROCEJECT	ejection routine
PROCPLAYER	get name of next player
PROCRATING	end of game routine
PROCREFUL	refuelling routine
PROCTOP	determine position in top 10
PROC-MOVE	move F-111B if on keyboard controls

PROGRAM LISTING 2

```
  10REM" F-111B by B.CASSIDY
  20DIM SCORE%(10),NAME$(10),DESCRIPTION$(4),TITLE$(5)
,LEVEL$(10),T$(9),T%(9)
  30FOR Q=0 TO 9:T%(Q)=5:NEXTQ
  40FOR Q%=0 TO 4:READ DESCRIPTION$(Q%),TITLE$(Q%):NEX
T Q%:TITLE$(5)="NASA"
  50MODE2:VDU19,0,4,0,0,0,0,23,1,0;0;0;0;:VDU19,7,4,0,0,
0
  60IF SCORE%(1)>0 CLS:MODE1:VDU4,19,0,4,0,0,0,0,19,2,4,
0,0,0,23,1,0;0;0;0;:PROCDISPLAY
  70REPEAT
  80REM ON ERROR GOTO 90
  90MODE2:PROCPLAYER:VDU23,1,0;0;0;0;
 100 CLS:PRINT TAB(0,10)"PRESS FIRE TO START"
 110COLOUR4:PRINT'"          ON"''"JOYSTICK OR KEYBOARD
"
 120REPEAT:FIRE%=(ADVAL(0) AND 3)
 130UNTIL FIRE%=1 OR FIRE%=2 OR INKEY(-1)
 140IF FIRE%=0 JOYSTICK%=FALSE ELSE JOYSTICK%=TRUE
 150IF FIRE%=2 FIRE%=3
 160CLS:MV=((32760*T%)/54750)
 170VDU5,19,0,4,0,0,0,19,1,3,0,0,0,19,4,0,0,0,0,19,8,3
,0,0,0,19,9,3,0,0,0,19,10,3,0,0,0,19,11,3,0,0,0,19,12,8
,0,0,0,19,13,3,0,0,0,19,14,1,0,0,0,19,15,3,0,0,0
 180C%=0:SCORE%=0:XXX%=254:A%=-1:W%=0:D%=400:CRASH%=0:
LAND%=0
 190PROCREFUEL:P%=640:MOVEP%,96:GCOL4,0:VDU XXX%
 200REPEAT
 210IF JOYSTICK% JUMP%=JUMP%-INT((ADVAL(FIRE%)-32760)*
T%/54750)
 220IF NOT JOYSTICK% PROC_MOVE
 230MOVEP%,96:GCOL4,0:VDU XXX%
 240IF JOYSTICK% AND ADVAL(FIRE%+1)<10380 XXX%=240:D%=
144
 250IF INKEY(-105) XXX%=240:D%=144
 260IF JOYSTICK%=TRUE AND ADVAL(FIRE%+1)>55140 XXX%=25
4:D%=400
 270IF INKEY(-73) XXX%=254:D%=400
 280P%=P%+JUMP%:S%=0
 290VDU4,30:*FX19
 300VDU11,5
 310MOVE P%,96:GCOL4,0:VDU XXX%
 320GCOL0,5:MOVE1248,FUEL%:VDU255:FUEL%=FUEL%-2
 330IF XXX%=240 FUEL%=FUEL%+1
 340GCOL0,14:MOVE-32,BOMBS%:VDU255
 350IF INKEY(-56) MOVE0,30:GCOL0,7:PRINT"  PAUSE: SCOR
ED ";SCORE%:REPEAT UNTIL NOT INKEY(-56):REPEAT UNTIL IN
KEY(-56)
 360 Q%=POINT(P%+4,112):QQ%=POINT(P%+60,112)
 370IF Q%=2 OR QQ%=2 CRASH%=TRUE
 380IF XXX%<>240 THEN 410
 390IF Q%=1 OR Q%=4 OR Q%=7 CRASH%=TRUE
 400IF QQ%=1 OR QQ%=4 OR QQ%=7 CRASH%=TRUE
 410IF POINT(P%+32,112)=6 AND XXX%=240 THEN SOUND3,-15
,255,1:C%=C%+1 ELSE C%=0
 420IF C%=4 PROCREFUEL
 430IF NOT JOYSTICK% AND NOT INKEY(-1) OR BOMBS%<=0 TH
EN 490
 440IF JOYSTICK% AND (ADVAL(0) AND 3)=0 OR BOMBS%<=0 T
HEN 490
 450S%=POINT(P%+32,D%):MOVEP%,D%+16:GCOL0,14
 460PRINT"*":SOUND0,-12,4,1:BOMBS%=BOMBS%-32
 470IF S%=2 OR S%=1 OR S%=4 OR S%=7 SCORE%=SCORE%+S%:S
OUND0,-15,100,2
 480IF S%=6 THEN GCOL0,5:MOVE32,264:PRINT"YOU HAVE BEE
N SHOT"'" DOWN BY USAF"'" YOU ARE A HAZARD"'" TO YOUR O
WN SHIPS":CRASH%=TRUE
 490IF P%<-84 P%=-84:JUMP%=0
 500IF P%>1300  P%=1300:JUMP%=0
 510RANDOM%=RND(100)
 520IF RANDOM%=1 AND A%=-1 THEN A%=RND(1152)+64
 530IF A%=-1 THEN 570
 540MOVE A%,1023:GCOL0,6:VDU255:W%=W%+1
 550IF W%=4 THEN A%=-1:W%=0
 560GOTO620
 570MOVE RND(1152)+32,1023
 580IF RANDOM%>10 AND RANDOM%<18-(T%/5) GCOL0,1:VDU 24
8,8,10,249
 590IF RANDOM%>20 AND RANDOM%<26+T% GCOL0,2:VDU253
 600IF RANDOM%>50 AND RANDOM%<55-(T%/5) GCOL0,4:VDU 24
7:SOUND2,-15,200,1
 610IF RANDOM%>60 AND RANDOM%<63+(T%/5) MOVE RND(1152)
+32,RND(1024):GCOL0,7:VDU252:SOUND1,-15,34,2
 620IF SCORE%>=T%*5 LAND%=TRUE
 630IF XXX%=240 THEN QQ%=TIME:REPEAT:UNTILTIME>QQ%+5
 640 IF FUEL%<0 MOVE128,800:PRINT"OUT OF FUEL":CRASH%=
TRUE
 650 UNTIL CRASH% OR LAND%
 660 IF NOT LAND% PROCRATING:MODE4:VDU 23,1,0;0;0;0;:P
ROCTOP:UNTIL FALSE
 670 COLOUR2:VDU4,30,224,226,224,225,227,227,226,225,2
```




```
24,226,225,224,227,226,226,225,227,226,224,225,5
  680FOR Q%=1 TO 58
  690IF JOYSTICK% JUMP%=JUMP%-INT((ADVAL(FIRE%)-32760)*
T%/54750) ELSE PROC_MOVE
  700MOVEP%,96:GCOL4,0:VDU240,4,30,11
  710COLOUR2:PRINT STRING$(20,CHR$(255)):VDU5
  720P%=P%+JUMP%:MOVE P%,96:GCOL4,0:VDU240
  730IF Q%>3 AND Q%<42 MOVE410,1023:GCOL0,4:VDU255,255,
255:GCOL0,7:MOVE410,991:DRAW410,1023:MOVE602,991:DRAW60
2,1023
  740IF Q%=5 OR Q%=40 MOVE485,1007:DRAW535,1007
  750IF Q%=7 OR Q%=38 MOVE460,1007:DRAW560,1007
  760IF Q%=9 OR Q%=36 MOVE435,1007:DRAW585,1007
  770IF Q%=4 OR Q%=41 GCOL0,8:PLOT69,510,1007
  780IF Q%=6 OR Q%=39 GCOL0,8:PLOT69,476,1007:PLOT69,54
4,1007
  790IF Q%<12 OR Q%>33 THEN 810
  800IF (Q% DIV 3)*3=Q% THEN GCOL0,8:PLOT69,395,1007:PL
OT69,625,1007 ELSE GCOL0,7:MOVE510,990:DRAW510,1023
  810IF P%>410 AND P%<538 AND Q%>40 THEN SOUND1,-15,200
,3:C%=C%+1 ELSE C%=0
  820QQ%=TIME:REPEAT:UNTIL TIME>QQ%+4*C%
  830NEXT Q%
  840IF C%<5 THEN 930
  850MOVE32,900:PRINT"YOU ARE PROMOTED"''"  TO ";TITLE$
(T% DIV 5):T%=T%+5
  860Q=-1
  870REPEAT Q=Q+1
  880UNTIL T$(Q)=NAME$:PRINT SPC(7)NAME$
  890T%(Q)=T%
  900QQ%=TIME:REPEAT:UNTIL TIME>QQ%+200:UNTIL T%=30
  910Q%=125:MOVE32,700:PRINT"CONGRATULATIONS"''"YOU HAVE
 COMPLETED  YOUR TOUR OF DUTY"''"WITH THE U.S.A.F"
  920FOR W%=0 TO 200:SOUND1,-15,W%,2:T%=5:NEXTW%:GOTO 7
0
  930MOVE 32,900:PRINT"AFTER COMPLETING"''"YOUR MISSION
YOU"''"CRASHED ON LANDING":PROCEJECT
  940QQ%=TIME:REPEAT:UNTIL TIME>QQ%+400
  950UNTIL FALSE
  960DEFPROCEJECT
  970X%=P%+32:Y%=100:JUMP=0:QQ%=4*((X% DIV 640)*2-1)
  980GCOL4,0:PLOT69,X%,Y%:PLOT69,X%,Y%-4
  990FOR Q=15 TO 0 STEP -0.3
 1000SOUND0,-INT(Q),4,1
 1010PLOT 69,X%,Y%:PLOT 69,X%,Y%-4
 1020JUMP=JUMP+0.2:X%=X%-QQ%:Y%=Y%+10-INT(JUMP)
 1030PLOT 69,X%,Y%:PLOT 69,X%,Y%-4
 1040NEXT Q
 1050PLOT 69,X%,Y%:PLOT 69,X%,Y%-4
 1060X%=X%-32:Y%=Y%+16
 1070MOVEX%,Y%:VDU230:MOVEX%,Y%-32:VDU231
 1080FOR Q%=1 TO 23
 1090MOVEX%,Y%:VDU230:MOVEX%,Y%-32:VDU231
 1100X%=X%+RND(9)-4:Y%=Y%-8
 1110MOVEX%,Y%:VDU230:MOVEX%,Y%-32:VDU231
 1120QQ%=TIME:REPEAT:UNTIL TIME>QQ%+20
 1130NEXT Q%
 1140ENDPROC
 1150DEFPROCRATING
 1160MOVE P%,96:GCOL4,0:VDU XXX%
 1170MOVE P%,112:GCOL0,1:VDU XXX%
 1180MOVE 256,1000:PRINT"GAME OVER"
 1190MOVE 256,900:PRINT"SCORE=";SCORE%:MOVE 332,700
 1200PRINT"RATING: "''" ";DESCRIPTION$(SCORE% DIV T%);T
ITLE$((T% DIV 5)-1)
 1210PROCEJECT:CLS:ENDPROC
 1220DEFPROCTOP:I%=0
 1230REPEAT:I%=I%+1:UNTIL SCORE%>SCORE%(I%) OR I%=10
 1240IF SCORE%<=SCORE%(I%) THEN PROCDISPLAY:ENDPROC
 1250LEVEL$=TITLE$((T% DIV 5)-1)
 1260CLS:QQ%=I%
 1270REPEAT
 1280H%=SCORE%(I%):H$=NAME$(I%):HH$=LEVEL$(I%)
 1290SCORE%(I%)=SCORE%:NAME$(I%)=NAME$:LEVEL$(I%)=LEVEL
$
 1300SCORE%=H%:NAME$=H$:LEVEL$=HH$
 1310I%=I%+1:UNTIL I%=11
 1320SCORE%=SCORE%(QQ%)
 1330PRINT TAB(7,28)" YOUR SCORE WAS ";SCORE%
 1340PROCDISPLAY:ENDPROC
 1350DEFPROCREFUEL
 1360FOR Q%=1 TO 32:SOUND3,-15,Q%,1
 1370MOVE -32,(Q%*32):GCOL 0,14:VDU255
 1380MOVE 1248,(Q%*32):GCOL 0,5:VDU255
 1390NEXTQ%
 1400JUMP%=0:FUEL%=1023:BOMBS%=1023:ENDPROC
 1410DEFPROCDISPLAY
 1420VDU4:PRINT TAB(8,4)"THE TOP 10 SCORES ARE"
 1430FOR I%=1 TO 10
 1440PRINT TAB(4,I%*2+4);SCORE%(I%);TAB(12,I%*2+4);NAME
$(I%);TAB(22,I%*2+4);LEVEL$(I%)
 1450NEXT I%
 1460PRINT TAB(8,30)"PRESS FIRE TO CONTINUE"
 1470REPEAT
 1480FIRE%=(ADVAL(0) AND 3)
 1490UNTIL FIRE%=1 OR FIRE%=2 OR INKEY(-1)
 1500IF FIRE%=2 FIRE%=3
 1510ENDPROC
 1520DATA AWFUL ,TRAINEE,POOR ,APPRENTICE,FAIR ,PILOT,G
OOD ,INSTRUCTOR,EXCELENT ,TEST PILOT
 1530DEFPROCPLAYER:COLOUR 1:PRINT'''"F-111B ";:COLOUR 5
:PRINT"by B.CASSIDY":VDU31,0,15
 1540COLOUR3:PRINT "NEXT PILOT"''
 1550COLOUR2:PRINT "YOUR NAME PLEASE":VDU31,5,20
 1560*FX15
 1570COLOUR 6
 1580*FX15
 1590INPUT""NAME$
 1600COLOUR7
 1610IF LEN(NAME$)>9 NAME$=LEFT$(NAME$,9)
 1620FOUND=FALSE:T%=5
 1630FOR Q=0 TO 9
 1640IF NAME$=T$(Q) THEN T%=T%(Q):FOUND=TRUE
 1650IF T$(Q)<>"" OR FOUND=TRUE THEN 1680
 1660T$(Q)=NAME$
 1670Q=9
 1680NEXT Q
 1690ENDPROC
 1700DEFPROC_MOVE
 1710IF INKEY(-98) JUMP%=JUMP%-MV
 1720IF INKEY(-67) JUMP%=JUMP%+MV
 1730ENDPROC
```

Sleuth

Trevor Attewell

The BASIC Sherlock Holmes! Have you seen the error(s) of your ways? This ROM should help to track down your misdeeds!

Although BBC BASIC has an error reporting system, it is often less than informative. A typical example is the unhelpful message "No such variable at line WXYZ", where the specified line may contain umpteen possible candidates. Even if the system were perfect, it could hardly be expected to find logical errors if the actual syntax is correct. It may well be worth a few minutes effort to track down the odd error by conventional means, but where there are many, or errors seem obscure, the job can be very frustrating.

BEEBUG have greatly alleviated the problem with SLEUTH, a ROM-based private detective that presents you with every scrap of available evidence about the operation of your program, making it easy to see where things are going wrong. The program is handled statement-by-statement, either continuously at an adjustable speed, or in single steps of one BASIC statement each. If a Procedure has been entered its name will be displayed, together with the current nesting level of any GOSUBs, REPEATS, FOR-NEXT loops and PROCs. The present value of every variable in each statement is given both before and after it is executed.

The program line is listed in full whether it is single or multi-statement. In addition there is a Dual Screen facility to display alternately the SLEUTH control panel and the program screen in which the program actually runs. This is particularly helpful where graphics are in use, since every step of the graphics build-up can be followed in detail. As well as specifying the current state of the program, SLEUTH can alter the value of any variable, update memory, or change the number of the next line to be executed. No actual editing facility is provided, and any necessary program changes must be made in the ordinary way after returning to BASIC.

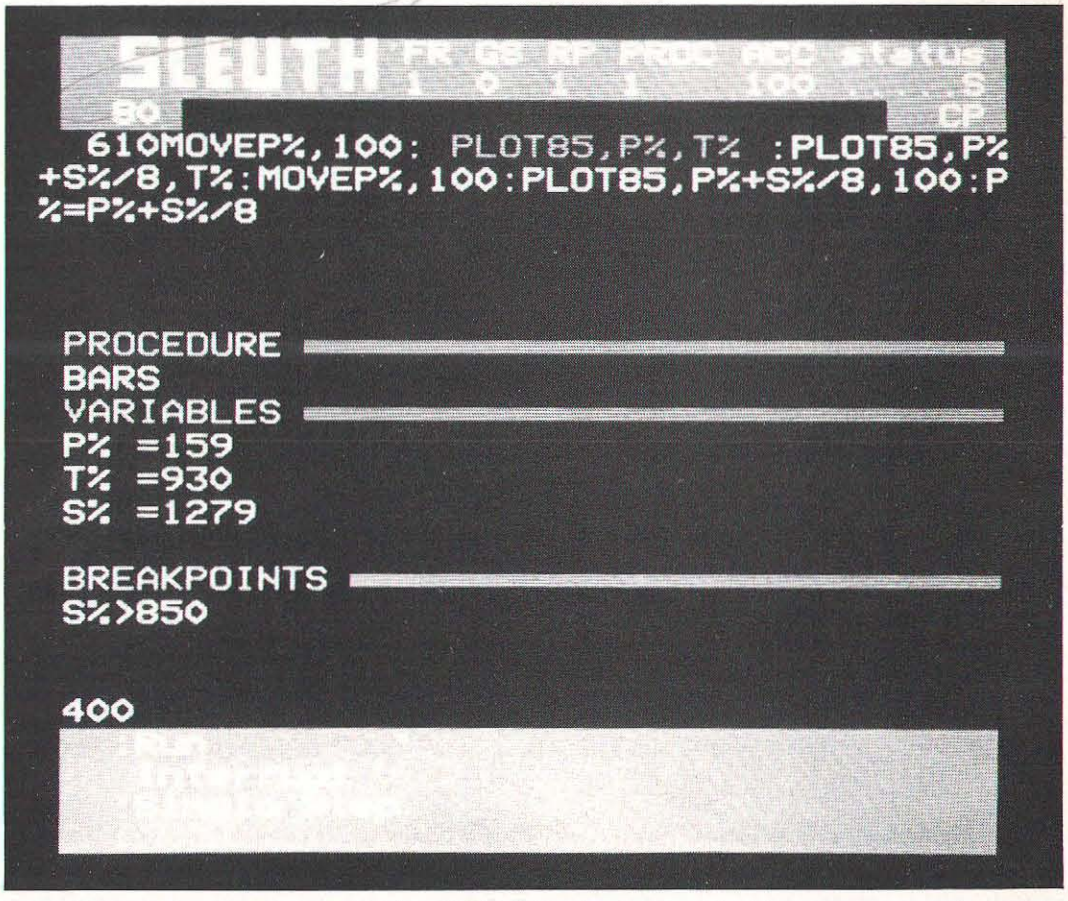
SLEUTH can be entered directly in one of three ways – by typing *SLEUTH from the keyboard, by the same command placed within a program or by external interrupt. In the case of an inserted command the program will run normally until the command is reached, after which control will pass to SLEUTH. External interrupt is achieved by first setting up SLEUTH and then running the program normally. On pressing CTRL@ the program is interrupted and the remainder can be run under the control of SLEUTH, which will initially be in single-step mode.

There are two ways to exit SLEUTH, either by returning control to BASIC with the program intact and accessible, or by exiting to the remainder of the program, which then resumes normal execution from that point.

STATUS SYMBOLS

As soon as SLEUTH is entered the control panel is displayed. At the top is the title and status panel, giving a good deal of condensed information. This includes the page number at which reserved memory for the dual screen facility starts, the nesting depth reached in current FOR/NEXT and REPEAT loops, GOSUBs and PROCs, the relative running speed and the run status. The last named flag indicates whether the program is being run continuously, single stepped, or multistepped. In the first case the program simply runs without stopping, at the speed set, until it either finishes or is stopped by pressing ‹@›.

Single stepping is used for close supervision, each statement being highlighted before execution. Either the control or program screens can be displayed continuously, the default being to alternate them if dual screens are enabled. Multistepping can be engaged at or after the start of a multi-statement block of code (FOR/NEXT, REPEAT, PROCEDURE or GOSUB), and the remainder of the block will be executed in one run. This is most useful if you find yourself flogging painfully through a long loop in which there are no (known!) problems. Other flags which may appear in the status panel include external interrupt (if set), and hard-copy selection (if the program



screen is enabled).

Below the status panel are three windows. Reading downwards, the first is a text window, 7 lines deep, which displays the current program line. Beneath this the second window is divided into three sections. The top section gives the name of the current Procedure, if one has been entered. The names of Functions are not reported, and they are simply executed, statement by statement, in the normal course of events. This merely reflects the fact that although the similarities between Functions and Procedures are often emphasised, they are in fact quite different animals. The middle section of the window lists all the variables which occur in the current line, together with their values. If the variables list exceeds the space available the title of this section changes colour, and cursor keys can be used to scroll up and down the list.

The bottom section of the second window can be toggled between three displays. By default it gives the names and values of up to four variables selectable by the user. These values are displayed continuously as the program runs. Alternatively, this section can display either a Breakpoint Table (which shows the selected breakpoints — see below) or a BASIC Status display. The latter lists the values of the four pseudo-variables PAGE, TOP, LOMEM and HIMEM in hex. It also shows the amount of free memory, both with and without the inclusion of variable space, and the current mode of the program screen (if enabled). The third window at the bottom of the screen is for command entry.

REACHED BREAKING POINT?

A particularly useful feature of SLEUTH is the ability to set conditional or unconditional breakpoints. The latter operate at the start of a specified line number during program execution, and hand over control to the control screen, at which point single-stepping may be selected. Conditional breakpoints stop execution only at the point where the condition is met. They can involve any expression that does not contain BASIC keywords, so (for example) the condition C=100*H%+45 is allowed, but C=SIN(X)*H%+45 is not. In practice this minor restriction is easily sidestepped by adding a statement to the program — in the case quoted we might add K%=SIN(X)*H%+45, and make the breakpoint condition C=K%. The breakpoints can be enabled or disabled as a whole by a breakpoint toggle. Up to 5 unconditional and 3 conditional breakpoints are allowed.

Various commands are available in control screen mode. Only one or two letters need be entered for any of these, the remainder being supplied by SLEUTH itself. Once displayed, DELETE will remove the command and RETURN will execute it. These commands include loading, saving and listing the BASIC program, the last-named engaging paged mode. OS commands can be issued, and the screen is then cleared except for the status and command panels to allow room for a (paged) disc catalog or other display. The use of SHIFT/CTRL will temporarily halt a program being run, while ‹@› stops execution and passes control to the control screen.

Other commands allow variable selection (mentioned above) and updating, by which means the current value of any variable can be altered at will. (G)OTO sends the program to the line specified, and (N)umbers toggles between decimal and hex notation for all variable values. (H)ardcopy toggles printer output, and can be used either to prevent printing where the program turns the printer on or to force printing of all PRINT statements. This control only works when the control screen is enabled in Dual Screen mode. (P)age allows the address of PAGE to be altered, but this must be done with care to avoid corrupting any resident program. The other pseudo-variables cannot be altered from within SLEUTH. (A)ccelerator governs the auto-run speed, specified by a number from 0 to 100, 0 giving about one instruction per second and 100 representing maximum speed, which is naturally very slow compared with BASIC alone. (T)race toggles the option to print the current line number in the top right corner of the program screen, which is useful if this screen alone is in use.

KEEP MEMORY IN MIND

It will often be necessary to take account of the memory locations used as Sleuth's workspace. These are &A00 to &AF, &B80 to &BFF, plus zero page from &4F to &6F, &F8 and &F9. The dual screen facility (if enabled) requires another 1.25k of memory (nearly 5 pages), which can be located where you like, within the usual constraints — by default it is placed immediately above the OSHWM. It can also be in sideways RAM if available. The memory allocation halves the space available for function key definitions, which must be set up after entering SLEUTH. It also makes life difficult for cassette users, since they cannot load programs under Sleuth. It is possible to load the program first, but in practical terms Sleuth is best considered as a disc-user utility.

As BEEBUG wrily point out, memory allocation is always an invidious compromise. If a program uses graphics, it may be possible to rescue some memory (part of which can be allotted to the dual screen facility) by temporarily changing to a lower graphics mode while debugging. If dual screens are invoked while a program is in memory, care must be taken not to overwrite it.

SLEUTH WITH CARE

A little thought must also be given to the contents of programs before running them under SLEUTH, and in some cases minor modifications may be needed. SLEUTH does not deal with assembler code in the same way as it does with BASIC instructions. Instead, assembly takes place in a single operation, except that any BASIC keywords in the listing will be intercepted and displayed. It is anticipated that assembler will be debugged separately using an appropriate machine code monitor, for example BEEBUG's excellent EXMON II. There will be trouble if any machine code overwrites Sleuth's workspace, and the use of DIM statements to produce automatically relocatable machine code above any BASIC program is a wise precaution wherever practicable.

Care must be taken over keyboard inputs to programs — these are allowed, but SLEUTH uses the space bar for single-stepping, so a GET statement which looks for a space bar press may need to be changed temporarily to another character. Because of the slower speed, brief INKEY timings will probably have to be changed for longer ones, and timing loops using TIME must terminate when (for example) TIME›time rather than TIME=time. These loops execute much more slowly than in BASIC, and it is often easier to multi-step the longer ones, or even to update the loop variable to its exit value. Programs should terminate with END. These and other minor quirks or anomalies are dealt with in the comprehensive manual. They cause very little inconvenience in practice.

CONCLUSIONS

This is a most useful and versatile ROM, which can save a great deal of time in debugging BASIC programs, particularly those which sometimes seem to follow only their own peculiarly twisted logic. The few restrictions placed on the program to be debugged are minor, and SLEUTH is quite easy to use — like most utilities it will repay time spent in exploring it fully.

Since it will not be needed every day (hopefully), the command summary in the manual is a very useful reminder, though in my view a separate card would be even more convenient (and hardwearing) for the purpose. SLEUTH does not work across the Tube, but BEEBUG may eventually bring out a version which does, in which case an inexpensive upgrade will be made available to those needing it. I can recommend SLEUTH, and wish that I'd had the benefit of it on more than one well-remembered occasion!

A & B'S NATIONWIDE DEALERGUIDE 01 - 437 0699

WARWICKSHIRE

LEAMINGTON HOBBY CENTRE

Warwickshire's sole official BBC Micro Dealer and Service Centre. Specialists in Monitors, Cumana and BBC Drives, and Epson Printers.

121 Regent Street, Leamington Spa. TEL: (0926) 29211

LONDON

THE VIDEO PALACE

Londons largest Software store. Model B Spectrum. ZX, Commodore 64 and Amstrad stockists. Plus full range of games software.

100 Oxford Street, London W1. Tel: 01-637 0366/01-434 4217/8

LEICESTERSHIRE

MICRO-MAYS
OFFICIAL ACORN/BBC DEALER

BBC Model B, Electron. **Printers:** Epson FX80, RX80, Shinwa CP80, Seikosha GP100, Star DP510. **Disk Drives:** Teac 100K single, twin, Shugart 100K singles, BBC Disks, TV (monitors, accessories and software). Also agents for Atari, Commodore, Sinclair, Oric and Dragon.

MAYS COMPUTERS, 57 Churchgate, City Centre, Leicester LE1 3AL (0533) 22212

SURREY

STATACOM

18 & 20 Grove Road, Sutton, Surrey. Tel: 01-661 2266

Computers and Peripherals
3" Hitachi Drive 5¼" range.
Printers and Monitors
2nd processors, business machines.

GAMES WORLD

Large range of BBC & Electron games always in stock. Phone or send for lists. Mail order via Access/Barclaycard, cheque or cash.

129 King Street, Hammersmith, W6 9JG. Tel: 01 741 4467.

WALES

GWENT COMPUTERS

Everything for the BBC computer.
ACORN AUTHORISED DEALERS
CUMANA DISTRIBUTOR
92 Chepstow Road, Newport NP9 8EE
Tel: 0633 841760

A & B COMPUTING CLASSIFIED ADVERTISEMENT– ORDER FORM

1.	2.	3.
4.	5.	6.
7.	8.	9.
10.	11.	12.
13.	14.	15.

Advertise nationally in these columns to over 100,000 readers for only 40p per word (minimum charge 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 DEPARTMENT A & B COMPUTING
No. 1 Golden Square, London W1.
01-437 0699

Name .....

Address .....

.....

.....

Tel. No. (Day) .....

ADVERTISERS PLEASE ADD 15% VAT

Please place my advert in A & B Computing for ☐ issues. Please indicate number of insertions required.

To appear in the Nationwide Dealerguide at £27 per insertion, simply fill in the details below.

Company Name _____

Address _____

_____

Tel No & Contact _____

Additonal Copy _____

_____

_____

No of Insertions _____

Post to: A&B Computing, Classified Dept., ASP, 1 Golden Square, London W1 or phone 01-437 0699.

A & B CLASSIFIED

Lineage: 40p per word.

Semi display: £9.00 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 0699
EXT 289

Send your requirements to:
CAROLINE FAULKNER, ASP LTD, 1 GOLDEN SQUARE, LONDON W1.

SOFTWARE ACCESSORIES

TURNKEY is a fully assembled M/C system to make all your datafile program disks secure with password protection. In unlocked state disks act normally (full catalogue and full information) but when locked can neither be read from nor written to. Simple Autoboot operation. **Send £12.95 plus 8 letter password, Vass, 1587, Gt. Western Road, Glasgow. Please state 40/80 tracks.**

Disc storage boxes, anti-static lockable with smoked glass cover, 2 sizes available:-
To store 70 disks **£17.95**
To store 100 disks **£19.95**
ED40, 46 The Maltings, Stanstead Abbott, Ware, Herts.

DISC BOXES £64.00 + VAT
ROM BOXES £49.99 + VAT

BBC Peripherals or the Amstrad education scheme and the Amstrad network.
contact:
N COMPUTERS LTD., Churchfield Road, Frodsham, Cheshire WA6 6RD
Tel: (0928) 35700 (10 lines)

BBC/Fischertechnik Robot Interface
4 Outputs for motors, lamps, electro-magnets (MitoM4)
8 Inputs for the connection of switches, relays (E1 to E8)
2 Analogue inputs for potentiometers, photo resistors.
Interface only £39.95 + £1.50 p&p
Interface + kit 30554 £99.95+ £2.50 p&p
Red Giant Software Ltd
3A Oakcroft Close, Pinner, Middx HA5 3YY

BBC/ELECTRON "IMAGE 2"

Without doubt the best back-up copier money can buy. Copes with probably all software. Send SAE for details of incredible features. Unlocks & locks progs. 100% M/C £4.80
PETER DONN (Dept AB) 18 Tyrone Road, Thorpe Bay, Southend-on-Sea, Essex SS1 3HS

ASTROLOGY

ASTROLOGY

for beginners. Special starter pack for only £11.50. Consists of a simple programme to calculate a horoscope, an introductory booklet and 2 self-teaching programmes (how to interpret the horoscope).
NO PREVIOUS KNOWLEDGE REQUIRED
for the B.B.C., A&B, Electron and many other micros, also wide range of programmes for more **Experienced Astrologers**. Cash with order (add 50p outside U.K.) or large SAE for free Catalogue to:
ASTROCALC (Dept A&B), 67 Peascroft Road, Hemel Hempstead, Herts HP3 8ER. Tel: (0442) 51809.

EPROMS

BANK MANAGER

BBC Home Accounts, Random access disk-files, standing orders, budget codes, up to 10 accounts. Balance Sheets, spending graphs £10 add 30p P&P.
Paul Gardiner
103 Elder Close, Badger Farm, Winchester, Hants. SO22 4LL.
Tel: 0962 51443 evenings.

SOFTWARE LIBRARIES

ELECTRON OWNERS We are a software library devoted to software for the Electron Computer. We have at present 80+ titles and we are continually updating with many of the top-selling games. Software lists should soon be well over **100.** Hire rates from 75p per fortnight. Life membership £6.00 (this includes free hire of first program). For further details please write to: Software Listing, Elderhurst, Lobleyhill Road, Meadowfield, Durham DH7 8RQ.

VIDEO

XXX VIDEO

Telephone (day or night)
0373 - 830563
FOR DETAILS

Glossy list from: **XXX Video Club, 2 Streatham High Rd., London SW16**
(You must be over 18 to ask for details)

SOFTWARE LIBRARIES

STAR-SOFT

(BBC + ELECTRON SOFTWARE LIBRARY)
Why buy programs when you can hire at a fraction of the cost?
JOIN THE BBC AND ELECTRON SOFTWARE LIBRARY
★★ Large varied selection of software including best selling titles ★★
★★ Expanding Catalogue ★★
★★ Hire rate from 75p per fortnight ★★
We offer a fast, quality service at competitive rates. Only £6 life membership inc first program free.
Send cheque/PO for £6 payable to **STAR-SOFT** or send stamp for catalogue: (please state whether BBC or Electron)
STAR-SOFT
16, Martland Cresc, Beech Hill, Wigan, Lancs. Tel: 0942 47574

FOR SALE

Printers:

CP-80 80cps Fric & Trac	189.95
CPA-00 100cps Fric & Trac	199.95
BBC Printer Cable 2 mtrs	7.95
3" HITACHI Disc drive cased complete	79.95
3" double sided discs for 2 **8.95** 4	42.95
Cassettes C15's boxed with label & card 3 **1.15** 10 **3.20** 100 **30.00**	
DISC PAYROLL Contracted in or out (Weekly, 2 or 4 Weekly & Monthly)	59.45
DISC ACCOUNTS Double entry Cashbook	36.45
1DISC RANDOM Access Mailing Full Sort	36.45

PRESTEL *2582020

Prices include VAT. Postage & Packing add 50p

Micro-Aid
(BBC)
25 Fore Street, Praze, Camborne, Cornwall. or 0209-831274 with ACCESS

Knights Electronics (Computers & Music)
Computers & Peripherals for the serious business & hobbyist
Our service does not stop at just giving competitive prices and a a fast friendly service.
Special Acorn, Atari 520ST & Amstrad Support
Sections own/other products

BBC 'B'	£277.00
BBC 'B' + DFS	£335.00
BBC 'B Plus'	£395.00
Acorn Music 500	£150.00
5.25" 5¼" Drives 40/80T from	£90.00
Kaga Vision 1	£199.00
Microvitec 1431	£165.00
Phillips 7502 Green	£65.00
Phillips 2007 Col. TV/Mon	£186.00
Canon PW1080A NLQ	£255.00
CCP40 Colour Plotter	£70.00
Epson LX80 NLQ	£197.00
Juki 6000	£195.00
Juki 6100	£295.00
Kaga 810 NLQ	£229.00
MP165 NLQ	£219.00

PLEASE PHONE IF YOU CANNOT SEE EXACTLY WHAT YOU NEED
All prices exclude carriage & VAT. We carry most leading brand names. Please ring to confirm latest prices. You will find us unbeatable. For more information on how to get our regular discount price list & details of our other services, contact:
KECM. 8 Westwood Lane Welling Kent DA16 2HE. Telephone: 01-301-3745 (10am-10pm)
Callers by prior appointment only

FOR SALE. BBC Model B, plus disk drive, games and accessories. Phone Brewood 850959 after 6.00 pm.

BINDERS

ADVANCED/USER GUIDE BINDERS stiff plastic backed, open flat, titled ring binders £3.50 + £1.00 P&P. Hepworth & Co (Dept AB), Waulkmill Farm, Ingersley Vale, Bollington, Cheshire SK1D 5BP.

COURSES

RESIDENTIAL COURSES programming techniques in BASIC. Learn to use your computer on a weekend/holiday course. Details from:- **Jaysoft Microdevelopments, 2 Wester Row, Greenlaw, Berwickshire.**

SOFTWARE

CHALLENGE 'MANDY' TO A GAME OF * * STRIP PONTOON * *

For BBC B, Electron (state which version you require) if you win enough she strips off item by item in superb hi-res graphics with her favourite tune in the background. "Excellent" Mr P. Keiser. Each version costs only £2.99. Send cash/cheque/ P.O. to:-
Dept. A&B
High Voltage, 16 Bridge Rd, Park Gate, Hants SO3 7AE.

TUTORSOFT offer a wide range of Educational, Business and Utility software, from major software houses for BBC and Electron machines. For your exact requirements please send S.A.E. to:
PO Box 103 Bexley, Kent DA5 2JU

TYPING TUTOR (BBC B)
(£12.95 tape £14.95 disc)
Typing tutor is really excellent and the best of its kind that I have seen for any home micro, not just BBC. High recommended..

(A&B COMPUTING)
Need we say more!
Willow Software
The Willows, Wrington Lane, Congresbury, Bristol BS19 5BQ

LE GUITAR TUTOR, a complete description and graphical tutor. Tasteful piece of software incorporating impressive machine code graphics. BBC and Electron – £5.45 Belles-Lettres Software, 106 Swansea Road, Garden Village, Gorseinon, Swansea, SA4 2HQ.

FOR HIRE

MIDLAND COMPUTERS

BBC + ELECTRON USERS
A Subsidiary of Platinum Gold

* OVER 4,000 different titles available for hire for the COMMODORE, SPECTRUM, BBC, AMSTRAD, ATARI and MSX computers.
* Arcade, Adventure, Educational and Business software too.
* Hire Prices from only 75p inc p&p.
* 20% Discount off all purchase software.
* Free Catalogues.
* Free Hints and Tips.
* Fast Return of Post Service (if reserves are given).
* All Games manufactures Originals with full documentation.
* Over 6,000 satisfied members.
* European members welcome.
* Limited Period Only FREE MEMBERSHIP.
* Fully computerised system keeps track of your order. We pride outselves in being the most professional Hire Library in the United Kingdom, with staff of 14 who knows what the word service means. Backed by our own software company and programmers.

National Computer Library
28 College Street, Worcester
WR1 2LS. Tel: 0905 611072.

Please rush me my free catalogue stating machine.

Name: _____

Address: _____

_____

REPAIRS

COMPUTER REPAIRS IN HAMPSHIRE
BBC, CBM 64 from **£19.95**
Spectrum **£19.95** inc VAT, p&p
Also Atari and VIC-20

COMPUTER CARE SOUTH
149 Kingston Rd, Porstmouth, Hampshire
Tel: 0705 825110

SOFTWARE PACKAGE

PEOPLE'S SOFTWARE
10 Lawrence Rd, London SE25 5AA
3 packages for BBC Micro with disc drive: Database, Word-processor, spreadsheet **£5 each** before ordering send S.A.E. for details.

DUPLICATION

jbs records
COMPUTER PROGRAMS
Real-time or (Slow) **High-speed** professional Cassette Duplication and Blanks 1 - 1000+. Computer printed Cassette Labels, BBC Disk Duplication and Unformatted Disks 1 - 500+. Fast security delivery service.
jbs records — a division of FILTERBOND LTD, 19 Sadlers Way, Hertford SG14 2DZ 0992-551188

LISTINGS

BBC AND ELECTRON listings from cassette or disk, 95p each including P&P. Send to S. Chadwick, 2 Collum Lane, Ashby, Scunthorpe DN16 2TA.

BOOKS

We use only 10% of our mental potentials

Einstein

These are the words of Albert Einstein, the greatest physicist of recent times.

L. Ron Hubbard's discoveries in the field of the mind prove today that Einstein was right.

In his book "DIANETICS: The Modern Science of Mental Health" L. Ron Hubbard takes one more giant step in this direction.

He reveals how anyone can use his discoveries to isolate the exact barriers that have so far prevented people from using their mental potentials to the full.

More and more people from all walks of life including technicians, artists, GPs, scientists, workers and managers alike use Dianetics technology today.

The many written reports on their success are the best proof of the quality of this book.

Find our for yourself. Order your copy of this remarkable book today.

Price £3.95 paperback £17.50 hardback. Make cheques and POs payable to S.H.F. Send to Church of Scientology Saint Hill Foundation, FREEPOST (A& B 1) East Grinstead, Sussex RH19 4ZA (no stamp required).

SOFTWARE HIRE

BBC SOFTWARE HIRE: Fast return service. Latest titles S.A.E. for details 141 Brent Lane, Dartford, Kent.

EDUCATIONAL SOFTWARE

BBC B 1.2 **TUXSOFT** **ELECTRON**

Three packs each containing two FULL-LENGTH programs

PACK A: **1. What's The Temperature with The Ghost Game. 2. Norman. Number Man.**
Painless learning and practice of negative numbers starring the fastest thermometer in town,the spook and Nifty Norman.

PACK B: **1. Number Check. 2 Speed Arithmetic with Number Catch.**
Pit your wits with donkeys, ducks, dustbins, dogs. . .Beat the clock and basket the bonus - if you can.

PACK C: **1. Tables Tester with Bonus Shoot Out. 2. Tables Square.**
Easy! Easy! Easy! Amaze your friends with your new brain power, Top the target - miss the missiles - fight the frustration!

The programs are designed to combine important teaching and practice with appealing presentation. All have several levels of difficulty and can keep children & teenagers (even adults!) absorbed for hours.

Prices inc. p&p	Tape	Plus 3 disc	BBC (40)	BBC (80)
Any single pack	£7.95	£10.95	£8.95	£9.95
All 6 programs	£22.95	£24.95	£22.95	£23.95

TUXSOFT, PO Box 1, Tuxford, Newark NG22 OJY

BBC B DISCS. Aesop's Fables £4. French: Time £4. Physics: Mirrors/Lenses £5. Subject Software, 31 Beech Close, Broadstone, Dorset BH18 9NJ.

New Typeasy version 4, fastest fun way to learn to type, BBC disc **£15**, Econet disc **£30.** Cassette (Version 2) BBC and Electron **£6.**
Carswell Computers, Carswell Barn, Faringdon, Oxon SN7 8JN

KITS

KITS & leads for BBC interfaces. Includes inexpensive DC multimeter & continuity buzzer for analog port. SAE for details, M Hughes, 17 Deeside Avenue, Chichester, West Sussex PO19 3QF.

TO FILL THIS SPACE PHONE CAROLINE ON 01-437 0699

UTILITIES

"D-MASTER"
The most powerful Disk-Analyser and Back up program. Copes with 40, 80 or mixed 40/80 tracks, deleted, short tracks, duplicated sector numbers, sectors of any size etc.
This program is not protected like other available products! Price **£7.00**. Many other utilities. Ask for our leaflet.
R-SOFT, 22 Marriots Close, Felmersham MK43 7HD. 0234 781730

2M128 25 ONS Fujitsu £4. Hitachi £4.75. 6264 - LP15 £5 all inc. prices. 10+, less 10% 50+ less 20%. Orders to K. Jones, 31 Padstow Avenue, Fishermead, Milton Keynes.

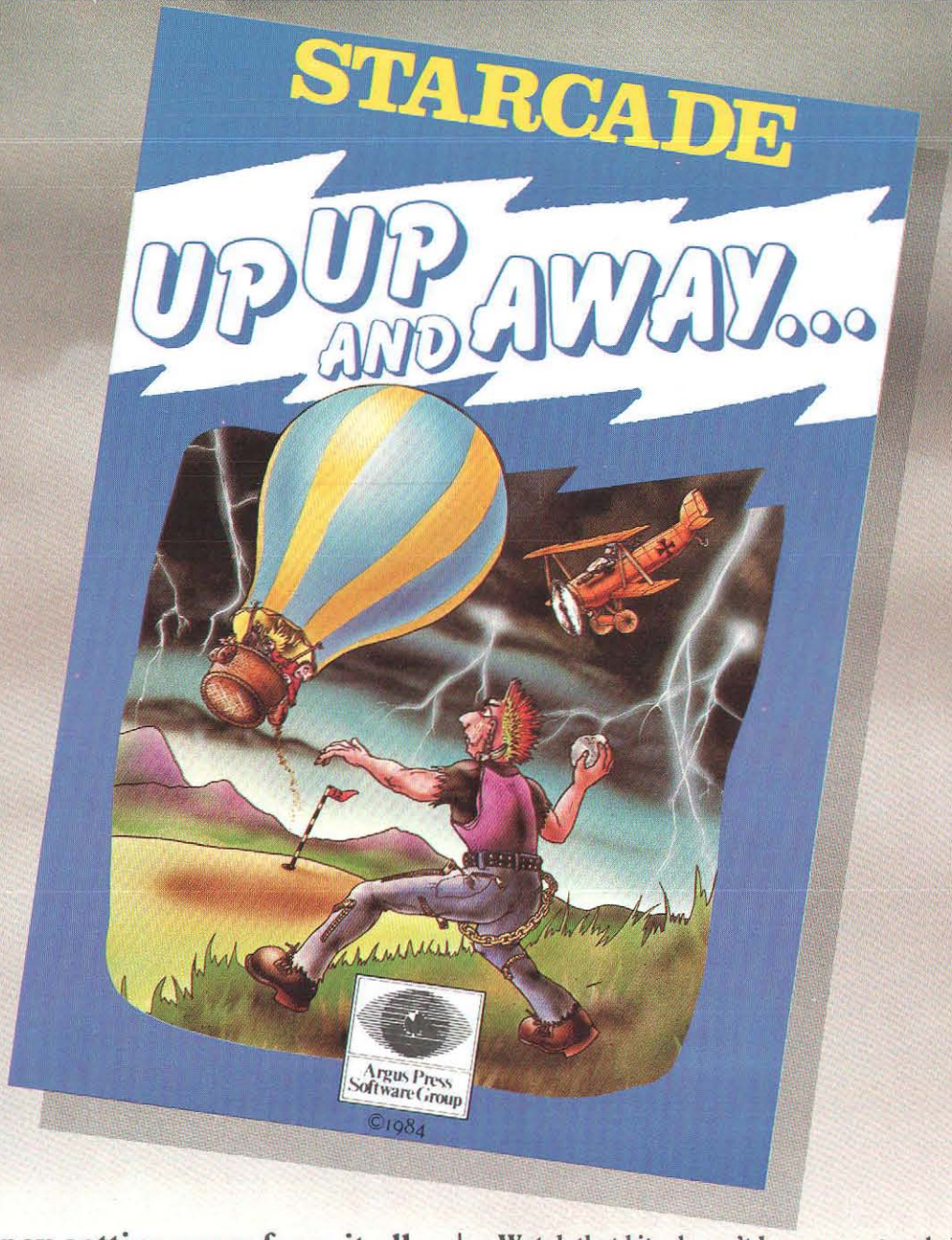
SWOPS

SWOP YOUR TAPES, from 50p. Send an SAE for list to A.P. 45, Victoria St, Blackburn, BB1 6DN.

PHONE CAROLINE ON 01-437-0699

STARCADE

UP UP AND AWAY...



How do you fancy getting away from it all with a trip in our hot air balloon?

Sounds tempting doesn't it. But it takes skill and fast reactions to avoid all the hazards. Are you up to it?

Could you manoeuvre the balloon over the tree tops and mountain peaks, while still keeping an eye on your ballast and your fuel, because when it starts to run low you'll have to descend, and not being distracted by people throwing stones at you, land carefully on the fuel pods. You'll be lucky to rise back up again. But look out!!

Watch that kite doesn't become entangled with you and don't worry about the noise, its only the aeroplanes that come diving and swooping at you. You'll need to rise higher to avoid the turbulence from the windmill blades but as you do you're likely to run into storm clouds and bolts of lightning. Also, the higher you climb in the stratosphere the more difficult the balloon becomes to handle.

By now you'll start to panic!! and wonder how on earth you're going to make it, but then again that's all the fun of the air.

COMMODORE, ATARI, BBC/ELECTRON (£8.95 each)
On disc for
COMMODORE, ATARI, BBC/ELECTRON (£10.95 each)

From Boots and all leading retailers

STARCADE

Argus Press
Software Group

produced by
ARGUS PRESS SOFTWARE
No 1 Golden Square, London W1R 3AB
Telephone: 01 437 0626

SERIOUS SOFTWARE

FOR THE B.B.C. MICRO

TOOLKIT

£27.00

- Adds 27 new commands for Basic program development and debugging
- Automatic error detection
- Full screen editor for Basic
- Host of utilities instantly available

SLEUTH

£29.00

- Revolutionary new debugging tool for Basic programmers
- Dual screen operation
- Single step facility
- Speed adjuster
- Gives YOU control over Basic.

HELP

£25.00

- The Information ROM
- On screen Help facility to aid programming
- Includes Basic keywords • Sound • Colour • Teletext • VDU commands • FX Calls • GCOL • Screen Modes • Memory usage • PAGE/HIMEM etc

EXMON II

£29.00

- Dual screen operation
- Full screen memory editor
- 60 machine code commands
- Invaluable if you wish to program, debug, or learn about assembler

!FREE!

information pack

To find out about these and other Beebugsoft products please phone for a FREE information pack. 24 hour 0494 81 5228

BEEBUG SOFT

DISCMASTER

£19.00

- Eleven powerful utilities to support the disc user:- Disc Editor • String Search • Recover • Movedown • Disc Menu • Indexer • Copier • Link • Overlay • 40 to 80 • Dual Catalogue • Formatter

MASTERFILE II

£19.00

- General purpose file management package
- Allows large amounts of information to be stored and processed
- Extremely flexible
- Easy to use

ORDERING INFORMATION

All these products are available from your local dealer and larger branches of Boots.

Post and packing 50p Prices include VAT

Telephone orders (24 hours) 0494 81 6666

Access

Mail order:- P.O. Box 50 St Albans Herts

For further information telephone (0727) 60263

SPELLCHECK II

£31.00

- ROM-based spelling checker for Wordwise or View
- Extremely fast on test timings Watford 1.3 DFS — 920 wpm Acorn 1.2 DFS + tube — 1450 wpm
- 6,000 word dictionary disc
- Dictionary increased with usage