AN ARGUS SPECIALIST PUBLICATION

SEPTEMBER 1985

95p

Personal Computing Today

WORDPROCESS ON THE C64!
It's easy with our listing

WIN THE MODEM OF THE YEAR
Acorn, Atari, Commodore, Apple, Sinclair and IBM owners — don't miss this chance!

GAMES, MUSIC EDUCATION . . . AND LOTS MORE

HOW TO SELL A PROGRAM

An author's advice — and tips from software houses



AVAILABLE FOR COMMODORE 64 WITH DISC DRIVE

Disc Drive?

DOMESTIC DRIVE
BUSINESS DRIVE

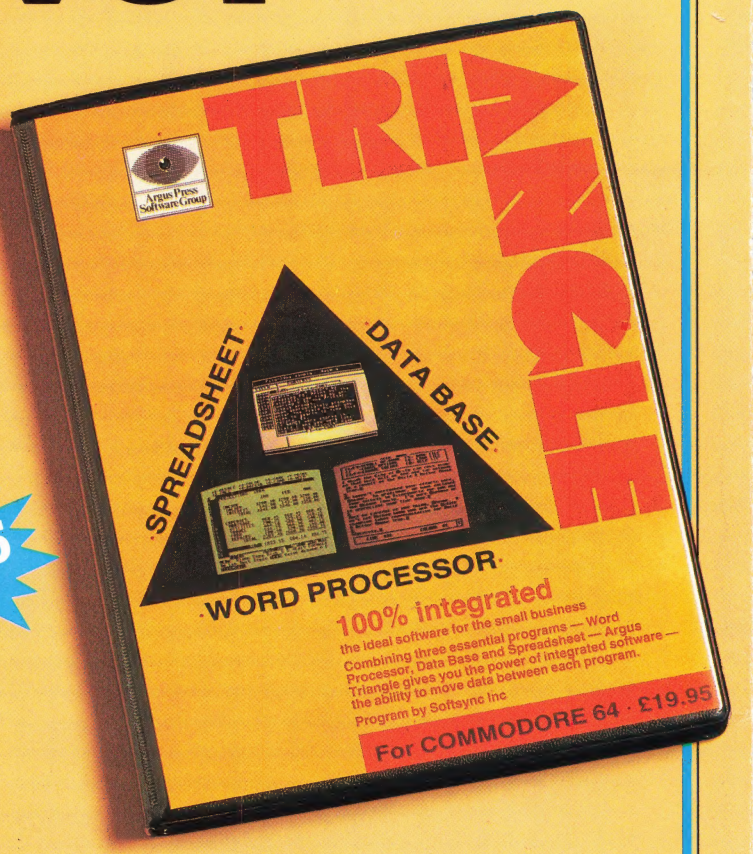
Realise the full potential of your disc drive with Triangle.

A powerful, fast and totally integrated suite of programs — WORD PROCESSOR, DATA BASE and SPREADSHEET.

Keep accounts, set-up household and business cash flow projections, store mailing lists.

All for only **£19.95** RRP

- **Three totally integrated programs**
- **Write and edit text easily**
- **Store your records, lists and inventories**
- **Answer your financial "What If's"**
- **Exchange data between programs**
- **Tutorial and on-screen "Help" windows**
- **100% machine-code for fast response**
- **Comprehensive manual that's easy to follow**



CREATIVE DRIVE

Stretch your mind and your disc drive with Activity Centre.

An integrated, stimulating and wide-ranging toolkit. Write, draw, animate drawings even compose music.

Widen your creative skills, write and illustrate reports for school or at work. Whether you're 7 or 70 years, you'll find Activity Centre great fun.

The price? It's a breakthrough at **£14.95** RRP

- **Write and illustrate text**
- **Paint and animate your pictures**
- **Compose your own music**
- **Save and copy images**
- **Large picture library memory**
- **Music fanfare library memory**
- **Easy to use manual**
- **100% integrated**



Argus Press Software Group

ARGUS PRESS SOFTWARE LIMITED

Liberty House, 222 Regent Street,
London W1R 7DB. Telephone 01 439 0666

AVAILABLE FOR COMMODORE 64 WITH DISC DRIVE

Personal Computing Today

VOLUME 4 NUMBER 2
SEPTEMBER 1985

Editor:
Deirdre Boyd

Assistant Editor:
Ione Holmes

Software Assistant:
John Donovan

Group Editor:
Wendy Palmer

Advertisement Manager:
Fulton Hogan

Classified Advertising:
David Horne

Copy Controller:
Lynn Collins

Publishing Director:
Peter Welham

Published by:
Argus Specialist Publications Ltd, 1 Golden Square, London W1R 3AB 01-437 0626

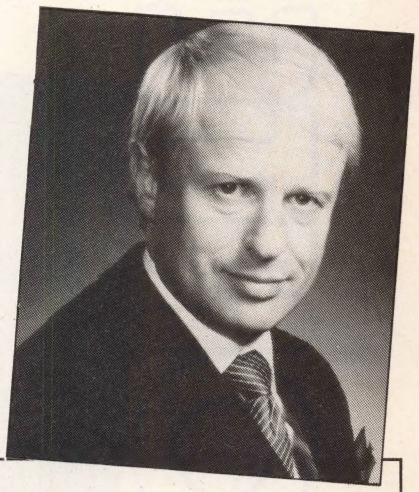
Design by
Argus Design Tel: 01-631 3888

Printed by:
Alabaster Passmore & Sons Ltd. London and Maidstone

Distributed by:
Argus Press, Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS

Personal Computing Today is normally published on the first Friday in the month preceding cover date. 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 Limited. 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 Limited and any reproduction requires the prior written consent of the Company. 1985 Argus Specialist Publications Ltd. All reasonable care is taken in the preparation of the magazine contents, but the publishers cannot be held legally responsible for errors. Where mistakes do occur, a correction will normally be published as soon as possible afterwards. All prices and data contained in advertisements are accepted by us in good faith as correct at time of going to press. Neither the advertisers nor the publishers can be held responsible, however, for any variation affecting price or availability which may occur after the publication has closed for press.

Subscription Rates. UK £14.00 including postage. Airmail and other rates upon application to Personal Computing Today, Infonet Ltd., Times House, 179 The Marlowes, Hemel Hempstead, Herts HP1 1BB. Tel: (0442) 48435.



Each issue of Personal Computing Today devotes its first page to a figure in the computer industry who has a strong opinion on how to improve it — or on its survival. This month, Nick Bessey calls for long-term investment.

THE CHALLENGE FACING HOME-COMPUTER MANUFACTURERS AND RETAILERS

Millions of home computers have been sold — but now research shows that there is demand for better functionality in terms of hardware and software, and a higher degree of pre- and post-purchase advice than is being delivered.

The lesson we have learned at Commodore — which applies to every manufacturer — is that, if we are to enjoy the continued support of the consumer, we must provide highly compatible computers, and give the consumer real reasons to buy our products.

Retailers are realising that the home-computer buyer is now more discerning than ever, and that they must provide the higher quality of in-store presentation that sophisticated products like micros deserve.

This year's panic price cuts did nothing to generate customer confidence. And there is now a need for far greater responsibility and maturity from the people who make and the people who sell micros.

Providing friendlier, more sophisticated computers and easy-to-use software to meet real needs requires long-term investment. Cut-price computers with cut-price service will not generate the margins needed to improve service and keep ahead of demand with technical innovation. Maintaining prices is in the consumer's best interest as it leads to long-term support, upgrade paths and more exciting software.

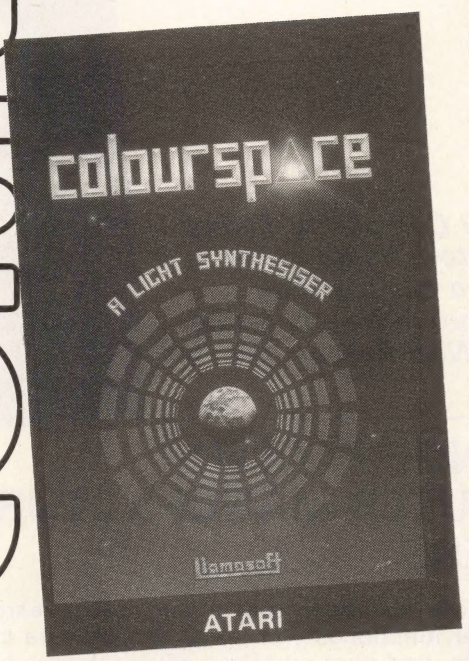
At Commodore we know that, unless we deliver the highest degree of compatibility and long-term support, the consumer will go elsewhere.

The challenge, then, to manufacturers and retailers alike is to deliver these improvements.

Nick Bessey
General Manager
Commodore Business Machines (UK)

ABC

CONTENTS



news

News 6
Catch up with events and gossip in the world of computing.

Product news 9
A device which lets six micros share one printer, Logo graphics, disk drives for the long-neglected Oric and Dragon, and a camera which transforms screen data to colour slides — these are just some of the new gadgets for your micro.

features

How to sell your program 12
There is no greater thrill than to walk into your local shop and see *your* program on the shelves. Programmer Michael Gruneberg parts with advice on how to get your programs published.

Write a bestseller 16
What do top software houses look for in a game — be it arcade, adventure or platform?

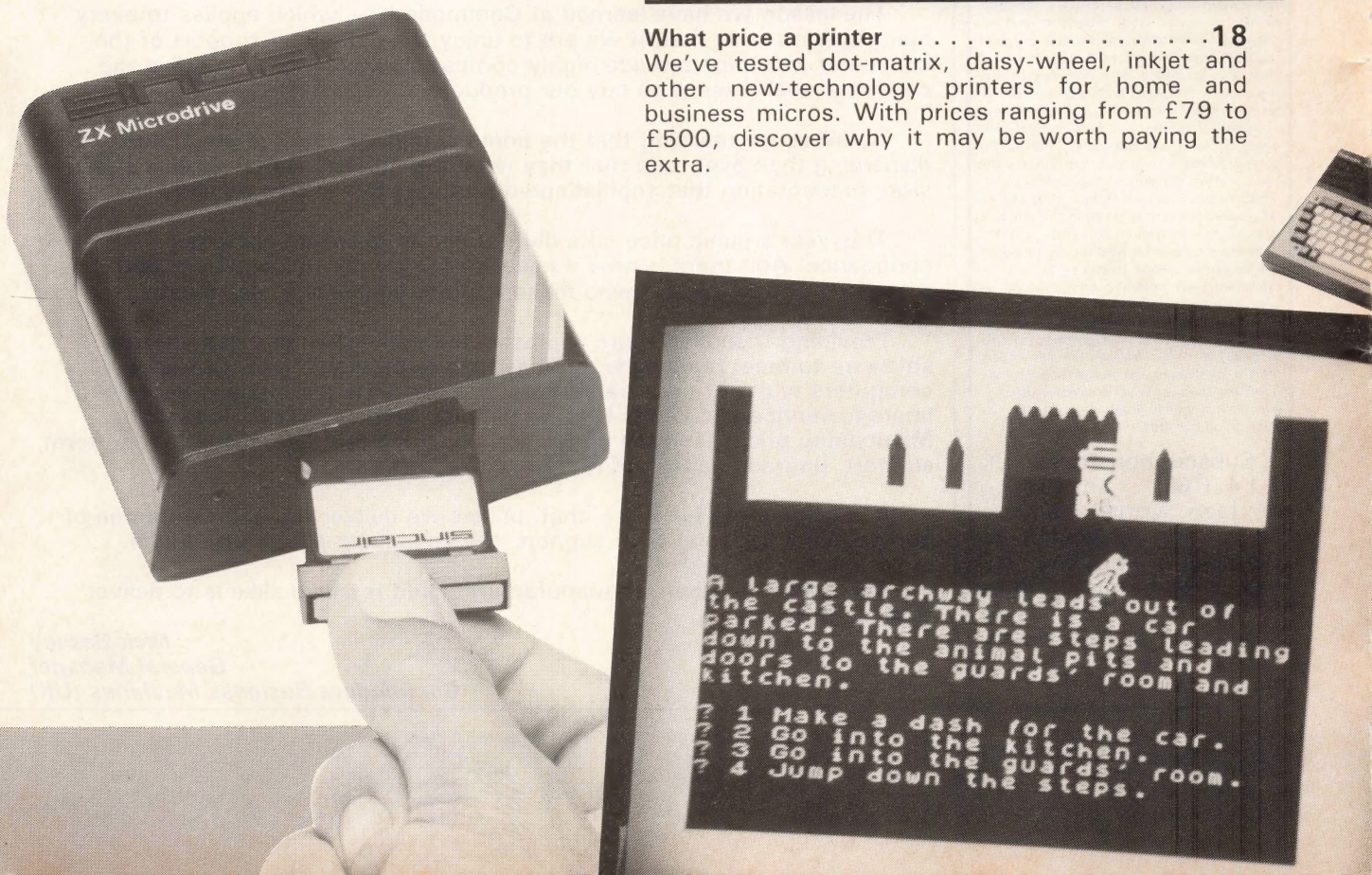
Rock on! 30
Cheap music packages for your micro can help create musicals — this month they go to the Edinburgh Fringe Festival.

From bluff to buff 44
Untangle the confusion of 'computer speak' with this simple explanation of structured programming.

Hacker's world 72
Hints, tips and gossip on the trans-world hobby.

peripherals

What price a printer 18
We've tested dot-matrix, daisy-wheel, inkjet and other new-technology printers for home and business micros. With prices ranging from £79 to £500, discover why it may be worth paying the extra.



education

Blackboard 22
Our education editor introduces the latest robotics package into his school — with unexpected results.

Software selection: education/utilities 23
It's difficult to choose educational packages, particularly when they're designed for someone much younger than you. So let our panel assess them for you.

Eureka! 27
The hunt for good language software leads to a crock of goodies.

programming

The Atari DOSsier 33
Expand the powers of your Atari disk drive with our hints and mini-programs.

MSXemplary graphics 36
MSX graphics statements are easy to use. Start with this guide.

About turn! 40
The second, and final, part of this article shows how you can convert graphics from the many Spectrum listings to the Amstrad.

Pack a byte! 43
How to squeeze as much information as possible into as little memory as possible.

competition

Win the modem of the year! 21
Miracle Technology's WS2000 modem runs on Atari, Apple, BBC, Commodore, IBM and Spectrum micros — and you've got a chance to get one free.

games

Adventure corner 48
Can John Donovan get out of *Lands of Havoc*, past *Assassin*, and survive *Angelique — A Grief Encounter*?

Software selection 49
PCT's expert review team spends time and energy testing games, so that you won't waste yours — or your money.

listings

Commodore 64/128: Super Typewriter 53
Turn your Commodore into a wordprocessor with this short — but effective — listing.

Oric: Home accounts 58
We can't promise you'll keep in the black, but at least your bank manager won't see red!

BBC: Singleminded? 65
One advantage Sinclair machines have over the BBC is single-key entry. Now you, too, can have it at a stroke.

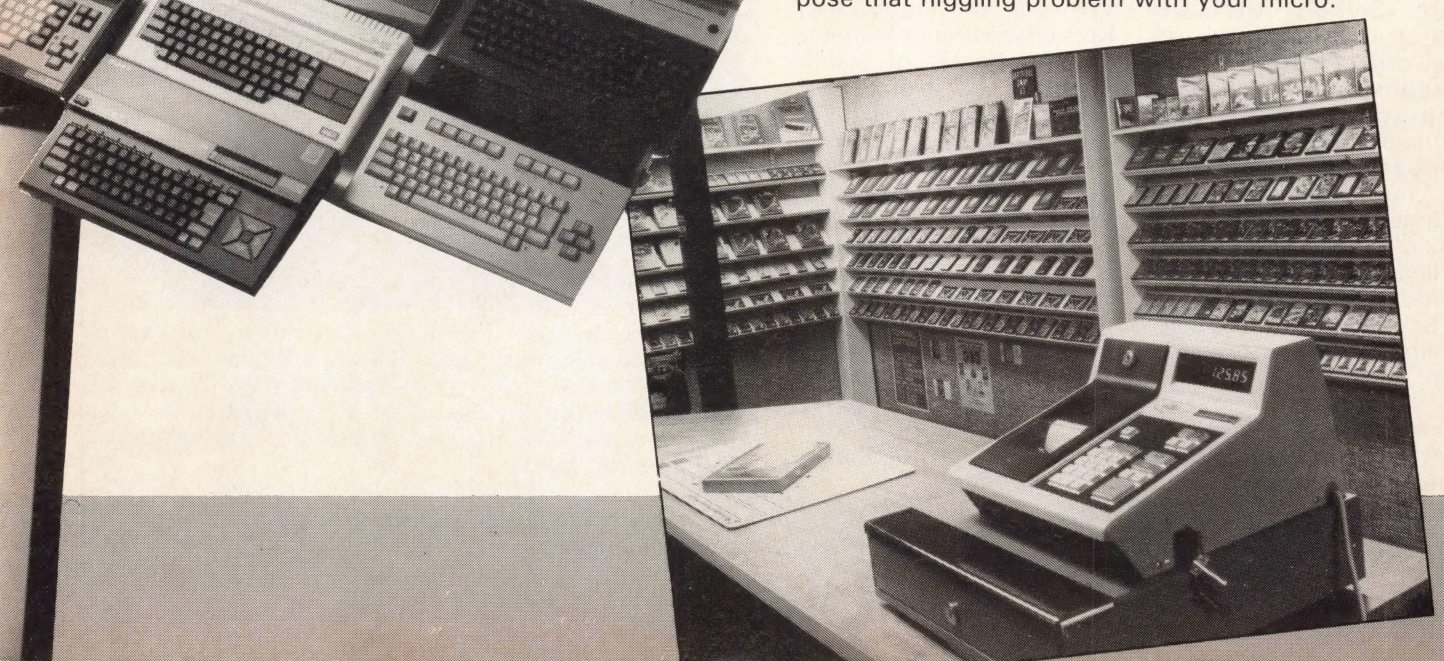
book reviews

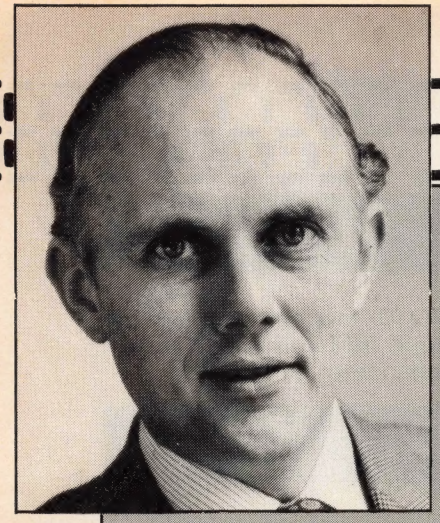
Booklist 67
Want to get down to Basics, expose the C16, make music, drive a disk — and more? Read on to see if these books will help you.

letters

Input 70
Go on — get it off your chest. Vent your views or pose that niggling problem with your micro.







ACORN'S UPS AND DOWNS

Acorn's financial future has been bouncing like a yo-yo. In February, Italian electronics giant Olivetti had to bail the company out with £10.4 million, for which it received 49.3% of the shares. Then, in July, Olivetti injected £4 million, bringing its share to 79.8%.

'We were on the verge of receivership,' Acorn chairman Alex Reid stated. 'If any of the parties involved in the refinancing plan had not agreed, the receiver would have been called in.'

As part of the refinancing package, major creditors agreed to write off debts of £7.9 million — about half what is due. The BBC waived 50% of royalties owed over the last year, and its future royalty rate will be reduced. The creditors are to receive £3.7 million of interest-bearing unsecured loan stock, redeemable in five to seven years, and £700,000 unsecured loan stock which will be repaid as a percentage of future profits.

Barclays Bank chipped in by doubling Acorn's banking facilities to £16 million.

As part of the deal with the BBC, Acorn will produce new models. The Cambridge Work Station was scheduled to be ready as *PCT* went to press at the end of July. Meanwhile, deputy chairman Hermän Hauser promised add-ons for the BBC, denying its production would cease — although 'the main emphasis will be on the BBC+'. Reid added that the Electron 'is still being made and delivered to suppliers'.

Reid denied that price cutting would form part of Acorn's ma keting strategy. But I, like many others, believe it was the BBC's uncompetitive price in a cut-throat market which led to its dilemma in the first place. So it will be interesting to see how new managing director Brian Long — once president of a Massey Ferguson division with an annual turnover of several hundred million pounds — will cope.

AVAST, YE PIRATES!

On 16 July the Copyright (Computer and Software) Amendment Act was finally passed, outlawing unauthorised copying of computer software. Penalties can now run between unlimited fines and up to two years' imprisonment for manufacture, distribution or importing of 'pirate' material. Selling, possessing or exhibiting it can incure a fine of up to £2,000.

The back-up for the new legislation is provided by an enforcement unit headed by former chief superintendent Bob Hay. His list of credits include his work at the command centre during the Iranian embassy siege and peacekeeping in London's Southall district during the Sikh unrest. Pirates beware!

The act was initiated by FAST, the Federation Against Software Theft, and is believed to be the fastest to get through parliament, having the support of all factions. 'Software theft costs the industry over £150 million per annum, and this act has removed any doubt about copyright protection for computer software,' explained its sponsor, Corby MP William Powell.

Information is forthcoming from the industry, police and trading standards officers, but prosecutions remain the responsibility of individuals. 'Obviously, this legal procedure is now a lot easier, because you don't have to prove your point in law,' Artic managing director Richard Turner praised.

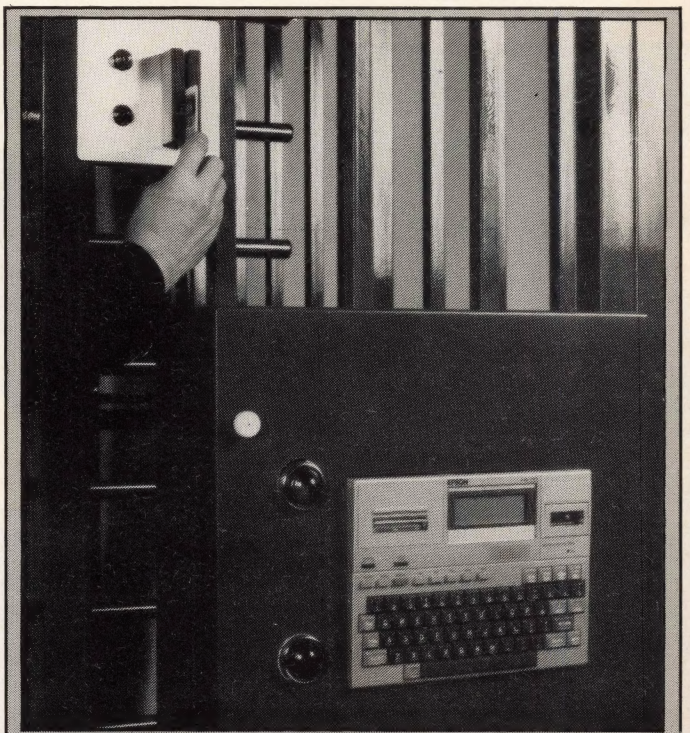
The seriousness of piracy was summed up by Virgin Games founder Nick Alexander. 'A year ago there were 300 software houses. Today there are about 100. Piracy is a major factor.'

By Andrew Selwood

WINNERS ALL!

Our June competition — with Seiko's RC1000 watch-cum-terminal as first prize — drew a large response from readers hoping to own this unusual add-on. An impartial passer-by was dragged into the office to plunge his arm into the box of correct entries, and pulled out that from Edwin Jones of Stonehaven in Scotland. The RC1000 is already ticking towards him.

Twenty runners-up each received a piece of software for their micro. They are: JW Smallwood of Chippenham, Andrew Tolley of Wolverhampton, Alex Slater of Hemel Hempstead, Robert Williams of Pontypridd, Kerry Bytheway of Quarry Bank, I Sandy of Handsworth, P Tavernier of Cressing, DJ Roberts of Stevenage, Neil Parker of Highbridge, Mark Lissak of London, T Dutton of Westerham, Jonathan Evans of Clwyd, Richard Kemley of Blaircowrie, Fraser Harkins of Stewarton, Michael Shelton of Halstead, Andrew McKay of Eccles, Gary Gittings of Sudbury, Frank Botchey of London, Mark Purden of Jersey and JC Shuker of Stovebridge.



Football gate of the future? Pessimistically confident that football identity cards will be enforced, Pass UK designed a low-cost access-control system to check them. Based on the Epson HX-20 portable micro, it uses a magnetic encoder similar to that used to check credit cards.

'Dedicated systems at each turnstile provide security, segregation, cash auditing and emergency turnstile release,' Epson enthused. And, heeding the Bradford fire disaster, stewards can use emergency pass cards to open extra exits.

MSX GETS DOWN TO BUSINESS

Spectravideo — the company largely responsible for defining the MSX standard — has developed a business version of the MSX micro. Called the X'Press, it has an inbuilt 3.5" disk drive, RS232C interface and 80-column card, and includes MSX-DOS and CP/M system disks as well as MSX disk Basic. It comes in a neat carrying case, accompanied by manuals on the three systems and the machine, and reference manuals on CP/M commands and Basic. All this is expected to be sold at the competitive price of £400.

The official launch of the X'Press is at the Berlin show in September, but Spectravideo hopes to have it available before then.

Inside the machine is 64K Ram, 16K video Ram and 32K Rom. MSX disk Basic is onboard, while the MSX-DOS disk has a memo-writer, disk manager, spreadsheet and database. The CP/M 2.2 disk also has utilities.

If you don't have a disk inserted on power-up, you automatically get MSX disk Basic. If you want MSX-DOS or CP/M, the appropriate disk should be inserted. There is no reset button to switch between modes, but you can get into disk Basic by answering a prompt in MSX-DOS.

The 73-key keyboard is not as cluttered as its predecessor, the SVI-728, but lacks its numeric keypad. There is also a cartridge port on the top of the micro. One drawback — which Spectravideo said should be corrected by the time the X'Press is in the shops — is that there is no £ sign.

On the right side, you will find two joystick ports, a cassette port and the disk drive. At the back are ports for RS232C output, second disk drive, Centronics printer, RF output, audio, video and power, as well as the on/off switch. There is a very handy extra: a handle which protects the ports when folded but which can be pulled out to prop the keyboard at an angle.

The inclusion of MSX-DOS means that the X'Press can avail of the wide MS-DOS software base used by most 16-bit micros, and CP/M software runs into thousands of packages. So the slow development of MSX software should not hinder this machine.

Spectravideo's SVI-728 did not sell well. But the X'Press is aimed directly at the increasing business 'middle' sector and is competitively priced. If marketed properly, this could turn out to be a winner.

IN BRIEF . . .

PCT will be at London's Olympia PCW Show from 4 to 8 September with its sister magazines: *Your Commodore, A&B Computing, Home Computing Weekly, MSX User, ZX Computing* and *Computer Gamer*. There'll be demos of popular games and special offers such as subscription discounts. Come round and chat.

Whoops! The August issue of *PCT* featured a photograph of *Bits'n Pieces* star Bobby Hains — but the series was credited to Granada instead of Grampian TV. Hope you tuned into the right channel.

Assassin assassinated! If you want to buy Robico's game featured on page 48, note that it is now sold as 'Rick Hanson'. Apparently, ICI had registered the word 'assassin' as a software trade mark.

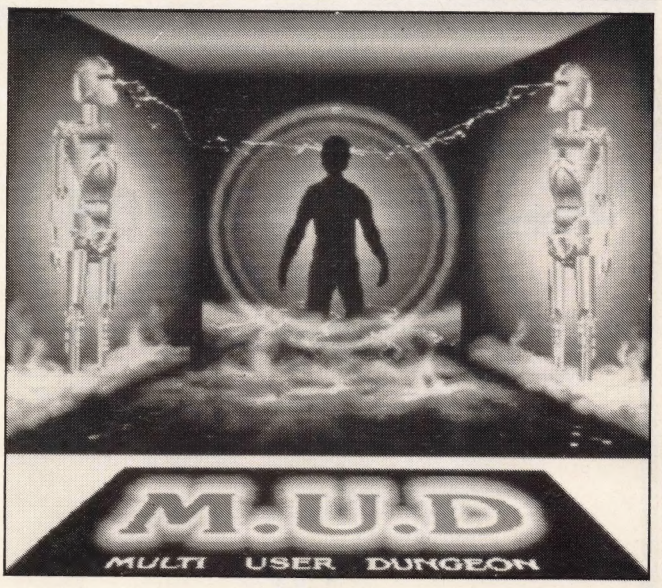
Oric — still no decision. Oric owner Eureka is still dithering between a choice of two after-sales service agencies for the UK. It hopes to settle the matter by the end of August.

AMSTRAD AND ENTERPRISE EXPAND THEIR RANGES

Amstrad is about to release its new micro but is keeping its cards close to its chest — details of its CPC464 and CPC664 micros were given to the press only as they were being delivered to the shops.

A lot of 'disinformation' is floating about, but our sources told us that the new machine is dedicated to business, being a 6800-based 16-bit with 128K Ram. It is compatible with its predecessors, and its operating system was compared with the Super Brain. But, unlike its predecessors, the new model does not come with a monitor, and an inbuilt disk drive is optional. The end of August will prove how accurate those sources are.

Enterprise, too, is keeping quiet, saying that it does not want to reveal details of its disk drive controller as 'we want it to work with all systems from launch'. It hopes to be in that happy position in September. 'We want to increase our immediate software platform substantially,' a spokesman added. CP/M will in incorporated amongst the goodies.



MUD — British Telecom's Multi-User Dungeons — will be up and running at the beginning of September. The largest multi-user computer game in the world, it took authors Roy Trubshaw and Richard Bartle five years to develop. You, too, can join in if you own a Spectrum, Commodore 64, Atari, Amstrad, BBC, QL, IBM or compatible, Apricot, Apple, Macintosh, TRS-80, VIC-20 or Dragon and have a modem with terminal software. Telephone 01-608 1173 for details.

AN EXCITING NEW DISC UTILITY for BBC 'B'

100% effective

REPLAY

Vine Micros

THE LEADERS IN TAPE TO DISC TRANSFER FOR THE BBC COMPUTER

THIS IS A UNIQUE HARDWARE/ROM-BASED SYSTEM

£35·00 inclusive of VAT and post

featuring:–

* *100% TAPE TO DISC TRANSFER*
* *MID-PLAY 'STOP AND SAVE' FACILITY (no more TAPE data files)*
* *MOST DISC TO DISC TRANSFER*
* *DISPLAYS A 'STOPPED' SCREEN (ready for printing)*
* *MID-GAME PAUSE FACILITY*

FOR BBC 'B' 1.20 O.S. – FITTED WITH THE FOLLOWING DISC FILING SYSTEMS

Type R8271	Type R1770	Type R2791
Acorn DFS/DNFS Watford Single Density and Pace Single Density	*Opus 3.45 and Solidisk*	*Opus 3.15*

The **REPLAY** system solely provides a facility for an individual user. Resulting discs are not recognised by a different computer.
We are offering **£8.00** trade-in value to **T.D.ROM** users who may wish to upgrade to **REPLAY**.

* **Easy to fit with detailed installation instructions, no soldering.** *

Available by post from:

VINE MICROS, MARSHBOROUGH, NR. SANDWICH, KENT CT13 OPG.
Telephone: (0304) 812276
(or your local Dealer)

TD ROM C

Have you just bought a disc drive and wish that more of your software was on disc? Then sit back and let the T.D.ROM transfer most of your tapes to disc easily and efficiently. The T.D.ROM handles any length of program and can save to any disc. To suit DFS types as for REPLAY. Economically priced at:

£18.00 including VAT and post
(Further details available)

ADDCOMM

This versatile ROM will suit all your TOOLKIT and GRAPHIC requirements. Forty extra commands are added to BBC BASIC including LOGO GRAPHICS. Highly rated in reviews and real value for money at:

£28.00 including VAT and post
(Brochure available)

VISA

4MATION Educational Resources

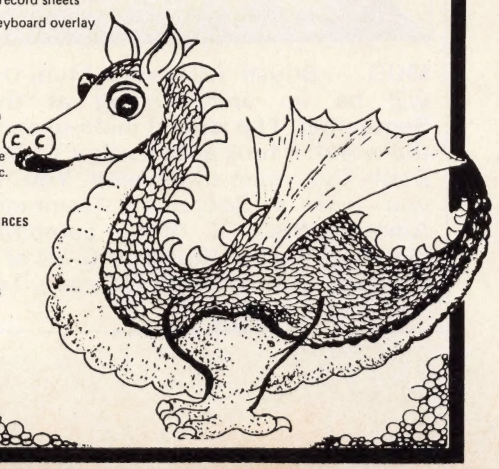
Beyond GRANNY'S GARDEN is:

DRAGON WORLD

For children aged 6+ Dragon World will provide the basis for a whole-class project to last for many weeks encouraging activities right across the curriculum.

The package includes:-

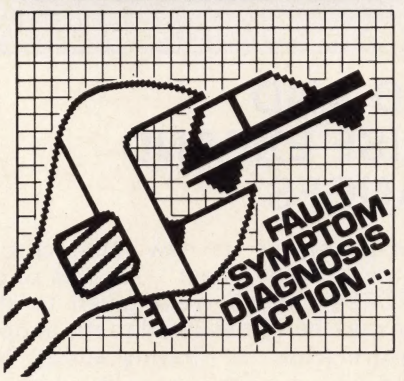
an audio tape of the exciting story and original music for dance/drama
a two-part adventure
four supplementary programs including a music synthesizer
an illustrated story book
a comprehensive User's Manual
a Book of Ideas containing 100+ suggestions for class activities
illustrated record sheets
Concept keyboard overlay



BBC B version now available.
RML version in preparation
£17.65 + VAT for Disc
£16.00 + VAT for Cassette
Add £3.50 for 3" Disc.

4MATION EDUCATIONAL RESOURCES
LINDEN LEA
ROCK PARK
BARNSTAPLE N.DEVON
EX32 9AQ TEL: (0271) 45566

CAR CURE



CAR CURE will transform your micro into an **expert mechanic** to help you **find faults** on your car and **determine the cure.**

The program will equally well help you **learn** about the faults cars can suffer, their symptoms and cures.

SPECIAL OFFER
£1.50 OFF prices shown below
THIS MONTH ONLY

★ **COMPREHENSIVE**
+ 900 faults
+ 300 symptoms
+ 90 parts
+ 1000's of screens of information

★ **POWERFUL FEATURES**
+ List faulty parts for every symptom
+ Display faults for each part
+ Find how to cure the fault
+ Trace symptoms of any fault
+ Identify faults causing 2 symptoms
+ Mark items for easy reference
+ Eliminate faults checked OK
+ Review progress on summary screen
+ Call help if in difficulty
+ Save diagnosis on tape

★ **EASY TO USE**
+ Colour coded menus to guide you
+ Available commands always displayed
+ Single key commands
+ Special beginners mode

CAR CURE comes complete with a comprehensive **5000 word User Guide**. Available direct from us on cassette for **48K Spectrum** at only £9.95 and on disc for **BBC Micro** at only **£11.95**.
SEE REVIEW ON PAGE 23

DEPT PCT
4 CLARENCE DRIVE
EAST GRINSTEAD
WEST SUSSEX
RH19 4RZ
Tel: 0342 28188

SIMTRON
PROGRAMS TO HELP YOU

Product news

Cliff Joseph collars the computer manufacturers to find the newest accessories for your micro.

MIRACLE MODEMS

Miracle Technology has upgraded its range of WS3000 modems. The basic V2123 has now been joined by the V22 and V22bis models which give you 1200 or 2400 baud full duplex, respectively.

The V22 retains all the features of the earlier V2123, plus Viewtext for Prestel, at a cost of £495. The V22bis has all those features, plus 2400 baud, full duplex CCITT standard, and costs £650 excluding VAT.

Other features in the range include Protect 3000 'anti-hacker' data protection, autodial/autoanswer, telex, Prestel, Telecom Gold and user-to-user/user-to-mainframe software. If you want more details, contact Miracle Technology at St Peter's Street, Ipswich IP1 1XB (tel: 0473 50304).

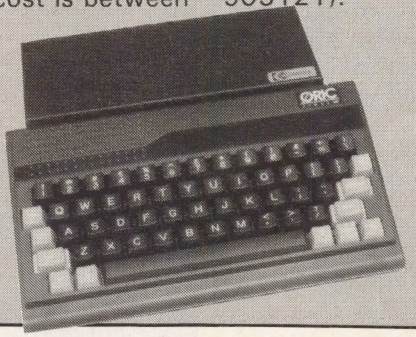


ORIC AND DRAGON GET DISKS



Cumana, which produces disk drives for many micros, has released a pair for the Oric and Dragon. The units, called Cumana Disk Controllers, can be 40- or 80-track, single- or double-sided drives. They come complete with Rom-based cartridge adaptor, user's guide and connection. The cost is between £179.95 (single drive, single-sided with 40 tracks) and £354.95 (dual drive, double-sided with 80 tracks).

Contact Cumana at Pines Trading Estate, Broad Street, Guildford, Surrey (tel: 0483 503121).



THE FAMILY THAT PRINTS TOGETHER . .

Nighthawk Electronics has come up with an interesting little device — a parallel printer switch which allows up to six computers to be connected to one printer. The switch works on nearly all machines with a Centronics printer port, and doesn't need additional software.

The switch is connected to the micros by an ordinary ribbon cable, then to the printer by a metre-long cable. It obviously allows only one computer to print out material at any one time, and deals with the computers on a first-plugged-in first-printed basis. Now all those squabbles over who gets to use the printer first can be solved — assuming, of course, that you are a family or business lucky enough to own more than one!

The printer switch is available in a range from £139 for a two-line model to £265 for the full six-line version. More details are available from Nighthawk Electronics, PO Box 44, Saffron Walden, Essex (tel: 0799-40782).

'64 GETS SERIOUS

Two companies have joined those providing 'serious' business software for the C64.

First Publishing, in addition to its books for the C64, has released *PowerPLAN 64*, described as 'a state-of-the-art, fully integrated graphics and spreadsheet program'. First Publishing managing director Sara Galbraith claims that *PowerPLAN 64* is better than similar packages costing up to £100 — it costs only £44.99.

Precision Software chips in with a low-cost version of the world-beating database system, *Superbase Starter*'. This is a complete computer filing system which, with Precision's *Stepping Stone* software packages can provide cashbook, sales daybook, stock records, and many other features. *Superbase Starter* costs £39.95, and the *Stepping Stones* packages £9.95 each.

Contact Precision on 01-330 7166 and First on 01-580 8418.



D.I.Y.

Anyone interested in an introduction to 'basic digital electronics and computer building blocks' can now buy Educational Electronics' *Introducing Microelectronics* package. It contains a circuit board, components and a 136-page booklet so you can conduct your own microelectronics investigations. The entry fee is £59.50 including VAT, and the place to send any money or enquiries is Educational Electronics Ltd, 28 Lake Street, Leighton Buzzard, Beds.

SAY 'CHEESE'

Video system specialist Reflex has developed a camera which, when linked to an IBM PC, can turn the machine's screen display into colour slides. The camera is the Celtic VFR 2000.

It needs no additional software, and uses standard 35mm slide film. It can produce colour slides even when the display is in black and white, using a process known as CEM (colour enhanced monochrome).

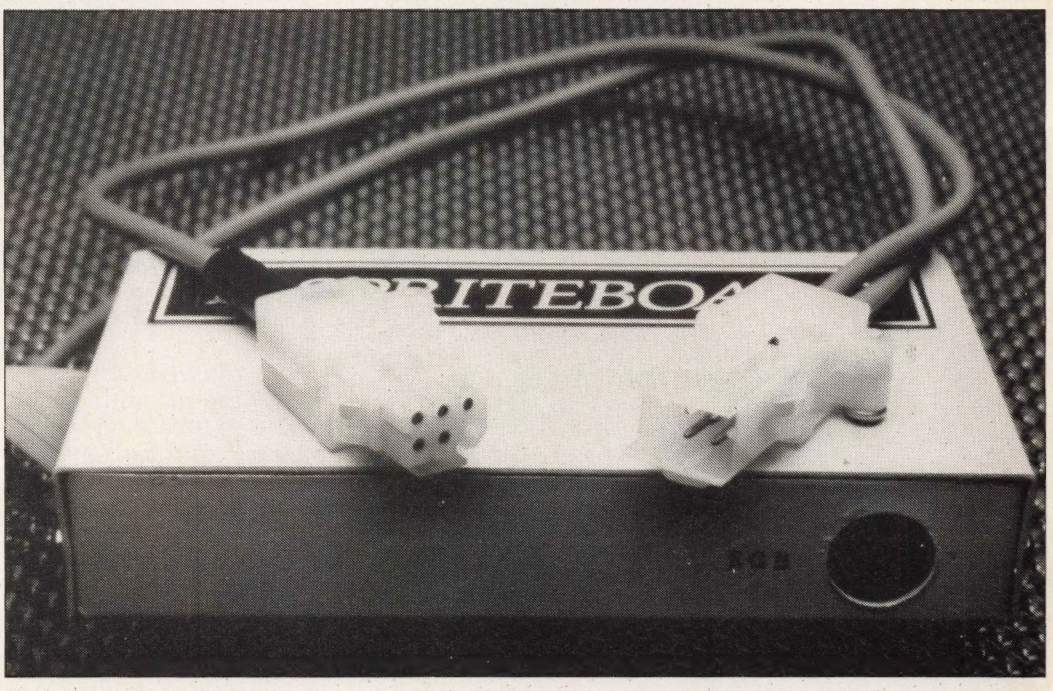
The only drawback is that the VFR2000 costs £2,795. But if you're interested, get in touch with Reflex at Wellington Industrial Estate, Basingstoke Road, Spencer's Wood, Reading (tel: 0734 884611).

LOGO ACQUIRES GRAPHICS

Logotron, the first company to provide a full version of Logo for the BBC, has launched its Sprite Board for the same machine, and has already won orders for it worth over £200,000. The Spectrum version is due later this year.

Logotron managing director Michael Hayward claims that the board 'allows even the computer novice easy access to true animation, and in a fraction of the time taken when using tedious machine code!'

The board, which plugs into the BBC's mgh bus, enables you to control up to 32 programmable sprites at one time. It also provides for up to 64 sprite designs, which can be manipulated with the board's sprite editor.



The Sprite Board costs £179.95 inc VAT. Or you can buy the Board with Logotron Logo for £239.95. Logotron also does a special deal for schools as Logo is such a popular teaching aid, charging just £175 for the Board and bundled Logo.

Contact Logotron Ltd at Ryman House, 59 Markham Street, London SW3 (tel: 01-352 1088).

POWERTRAN PRODUCES FIRST INDUSTRIAL TRAINING ROBOT

Powertran Cybernetics has produced the first SCARA (selective compliance assembly robot arm) robot for use in industrial training. Called IVAX, the robot is designed to be used as a training device for courses in robotics, mechanics, control systems, and other educational and industrial fields.

With its Z80A microprocessor, IVAX can be controlled directly from Basic or machine code, and is modelled on industrial robots — but at a fraction of the cost. It's available either in kit form or ready built. For details of price and more technical info, contact Powertran Ltd at Portway Industrial Estate, Andover, Hants (tel: 0264-64455).

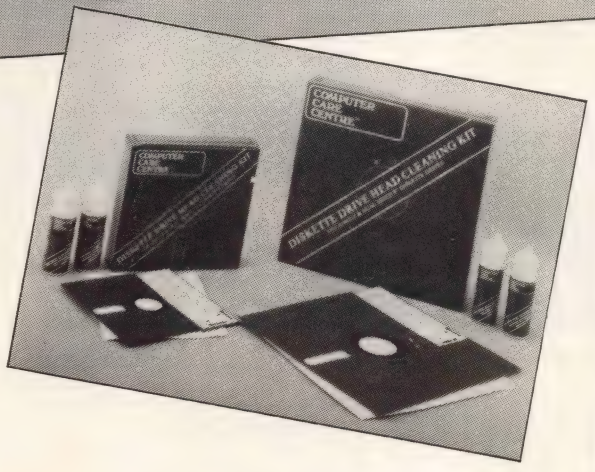


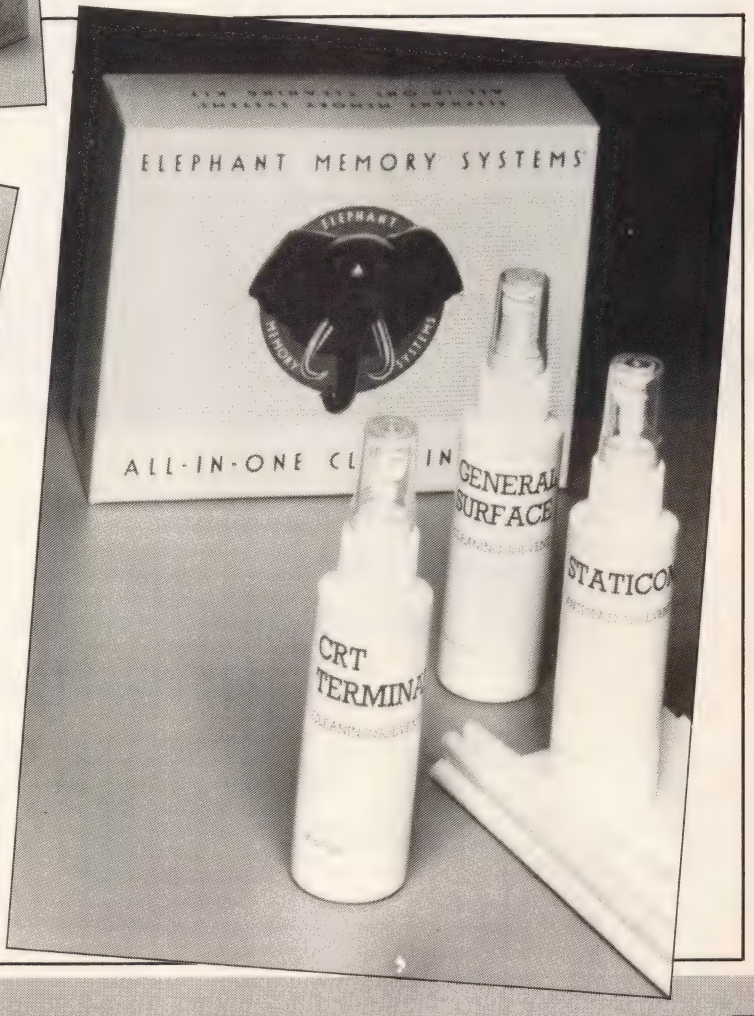
CGL'S NEW CHAMPION

CGL, which markets a wide range of joysticks/hand controls under the WICO brand name, has produced a 'budget' hand control — joystick, in other words — for playing all your budget software games with.

Called the CGL Champion, and priced at £11.99, it has a contoured hand grip, auto-fire switch and two fast-fire buttons. It is compatible with many major systems. Check with CGL on 0628-72448.







Computer care products seem to be an up and coming thing at the moment. Elephant Memory Systems, Rapidata and Nu-Way Styli Components have all released cleaning/care kits.

Rapidata weighs in with its non-smelly, anti-static kits, and disk-drive head-cleaning kits. Elephant's range of cleaning fluids and cloths attack dust, 'caked on dirt' and static. And Nu-Styli hopes to clean up with its Keyboard Cleaners and Personal Computer Cleaning Sets. If you can't decide which kit you need, ring Elephant on 0923-41244, Rapidata on 0635-61086 or Nu-Styli on 021-454-5955. I'm sure someone will be happy to make up your mind for you.

Feature

How to sell your program



'There is no greater thrill than walking into your local stockists and seeing your software on the shelves!'

Michael Gruneberg has had his programs published on 12 micros by nine companies. He passes on his advice on how to get your programs published, too.

Four years ago I developed a method of teaching foreign languages — *Linkword* — which has now been published on some 12 machines by nine companies. What follows is based on my publishing experience over the last few years.

Preparing the presentation. As many people wish to publish software, and more software is submitted to publishers than will see the light of day, it is important that you establish the potential worth of your own software before you approach a publisher. For educational software, this should involve evaluation — preferably independent — of the educational value of what you have written. This could be reports by six or so teachers, based on classroom use.

Who to approach. Clearly, who to approach in the present economic climate is a major headache. Both small

dubious companies and large established companies with seemingly sound reputations are folding in alarming numbers. By and large, the golden rule must be to approach first a large company with an established reputation. But beware! Even large companies can be unreliable or unsound.

I prefer to first approach a software company directly associated with, or owned by, a hardware manufacturer because they are likely to publicise their products in catalogues sold with new machines.

How to approach a company. My preference is to use the telephone. Ask to speak to the software evaluator and explain to him or her quickly and accurately who you are and what your product is. It is at this point that the value of an independent evaluator comes in. In my own case, for example, I had an independent evaluation from Thompson Holidays that Linkword was about three to four times faster than conventional methods of language learning, and this had a marked effect on catching the interest of software evaluators. I have found that only if a strong interest is expresed at this stage is there much point in sending the software for review.

Is it ethical to approach more than one software house at a time? Software houses are generally very good at giving an answer quickly, in two or three weeks. But there is no more immorality in approaching more than one software house than in seeking more than one offer for a house you are trying to sell.

With some software there is an advantage in the author demonstrating the program to the software evaluator. I always took the opportunity to personally show how *Linkword* worked and to see the software evaluator if there was a strong interest. This policy paid strong dividends, as I was able to demonstrate exactly how *Linkword* worked and to deal with any problems the potential publisher saw.

The contract. Assuming all goes well and you are offered a contract, you must suppress your joy and consider that contract. Almost everyone negotiating a contract for the first time thinks the royalty figure is the only thing that matters. They couldn't be more wrong. Of course, the royalty figure in the percentage of sales you receive is important, but other points are equally so.

First, it is vitally important that you do not give all your rights to any one company for all computers for all time. For example, if you have designed your program for a BBC, ensure that the rights to publish are for that machine only and not for, say, a Spectrum as well. One of the advantages of approaching a software company directly associated with a particular machine — Commodore software for the Commodore, Amsoft for the Amstrad, and Acornsoft for the BBC, for example, is that they are interested only in publishing for one machine. You must ensure that you are free to sell rights to other machines if you are offered a contract from such a company.

Of major importance, too, is an advance. It is vital, particularly if you do not know the company well, that you are paid in advance on royalties. If the company goes bust, then at least you will have some reward for your effort. And paying an advance is likely to put pressure on the company to bring your program out as soon as possible. It is amazing how many companies sit on programs they have agreed to publish. A time limit to date of application, after which rights come back to you, is an important part of the contract.

There are a few more contract points. First, given the high failure rate of software houses, you must have your rights returned immediately a company ceases trading or becomes insolvent. You should not allow publishing rights in the company you sign a contract with to be transferred or sold to another company without your permission. Otherwise your rights or your programs might be sold by a company about to go into liquidation.

Second, you should ensure that you are given a statement of account and any royalties due every three months after publication, even if no money is due to you. If companies fail to pay you in this time, they are in breach of contract and you can bring pressure on them to pay royalties owed.

'A golden rule is: never be embarrassed about talking about money and terms'

Third, you must always have the right to inspect the books of the software house as they relate to your royalties, to check whether you are being paid the correct amount. The knowledge that you have the right to check is in itself likely to be a disincentive to 'creating accounting'.

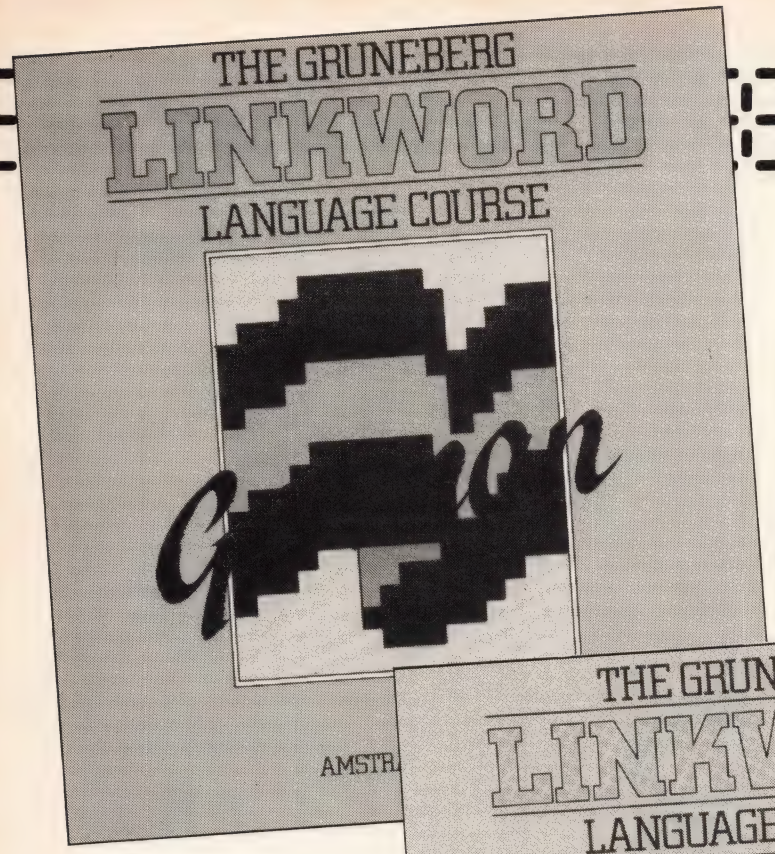
Fourthly, give rights only to the program being published, with options on similar programs if you must. Do not give rights to characters or themes — such as Mr Men or Minder. Confine your rights to computer presentation only.

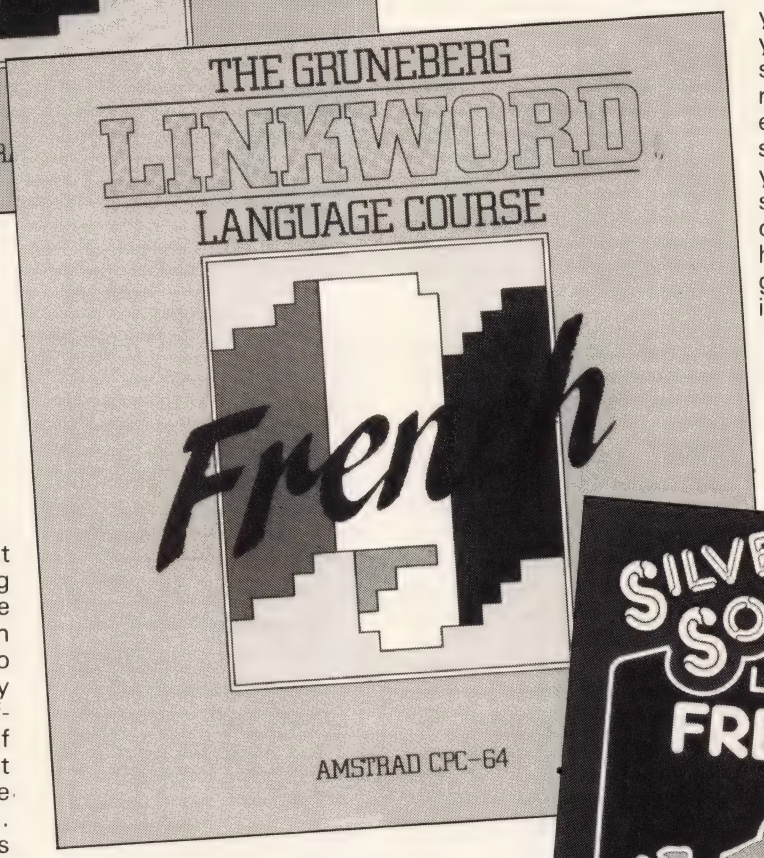
Another golden rule is never be embarrassed about talking about money and terms. Software houses are business concerns which want to publish your product for profit. You must ensure both that you will receive a reasonable profit and that the software company will see profit in publishing.

What, then, is a reasonable royalty to expect? Royalties are either a percentage of money received by the software house or a percentage of the recommended retail price. A software house usually sells its software to a wholesaler for about half the price asked for in the shops — this is the money received. Normally a royalty payment of between 15% and 25% of moneys received is given to the author as royalties. Where the royalty is based on the recommended retail price, the usual royalty payment is between 7 ½% and 12 ½%. It is not advisable to accept flat rate payments, such as 25p per copy sold, because of inflation. You should also make sure that no hidden deductions, such as packaging, are included.

In most contracts there is a clause about foreign rights. It is reasonable to expect at least 50% of foreign receipts. But if your program has great potential abroad you might wish to grant UK rights only. For most programs, however, the question of foreign rights will not be a major issue.

Given the problems I have outlined above, the question arises as to whether you should employ a solicitor to draw up a contract. I would emphatically say no for the ordinary 'one off' program. Solicitors are likely to cost you a considerable part of your royalties, and if you deal with a reputable software house it should agree to most of the points discussed. Of course, if you have wor-





Ensure that the rights to publish are for one machine only'

ries about the contract you are offered, seeking professional help might be advisable. My own philosophy has been to sign contracts with many software houses on different machines so that, if some contracts turned out badly, at least some would turn out fruitful. This is exactly what has happened.

Possibly the most difficult problems concern adequate promotion of the program and providing financial statements and royalties in time. As far as lack of information is concerned, there is no substitute for phoning regularly to ask what is happening. At least the software house will know you are still interested in your program!

To be fair to the software companies, it is unrealistic to expect them to advertise heavily when software sales are low and when buyers in some of the multiple stores are reluctant to stock 'slow selling' educational software. On the other hand, without promotion the exercise is a waste of time, as potential customers will not know of the existence of your program.

The very least you can expect is that all relevant magazines and journals are supplied with a review copy, but in my experience not every software house manages to do this. To be fair, most do; you must ensure that this step is taken.



Conclusion. If this article has left you wondering whether software publishing is worth the effort, you must ask yourself the question: what do I want out of publishing software? If all you want is the pleasure of saying 'I published a piece of software', you can ignore all the pitfalls, and not worry if you don't find your program in the shops, receive no royalty and find yourself 'ripped off' with no chance of publishing with another software company because you have tied yourself to your first company for life.

On the other hand, if you want to ensure that your software reaches the shops, that you receive a reasonable return for your efforts, and that at least some people appreciate your work, following some of the points I have outlined should prove helpful. There is no greater thrill than walking into Boots or Menzies or your local stockists and seeing your software on the shelves. Sorry, there is one greater thrill — the shop assistant telling you it is selling well!

OH NO! SOLD OUT!

Your COMMODORE

YOUR BEST INDEPENDENT COMMODORE MAGAZINE

Whatever you do, don't let this happen to you. When you rush down to your newsagents to get your copy of the latest Your Commodore don't suffer the disappointment of being told they've none left.

Why not sit in the luxury of your own home and feel safe in the knowledge that your copies will come neatly wrapped winging their way through the post and find themselves popping through your letter box?

Whatever your interest in the Commodore range of computers and peripherals, you simply cannot afford to miss a single issue. The magazine is packed with news, reviews of the latest software, information about the books for sale, articles to help you with your programming, games to type in and play, useful routines to make your computing life easier, and all sorts of useful, informative and entertaining features. So whether your interest is purely in the latest games available for the VIC 20 or in reading a serious review of the most recent hardware for your Commodore, you must ensure that you read and inwardly digest every issue of Your Commodore.

And it's so easy to do! Just fill out the form below, write a cheque and send it off to the Subscriptions department. You know it makes sense!

SUBSCRIPTION ORDER FORM

Cut out and SEND TO:
YOUR COMMODORE
INFONET LTD., TIMES HSE,
179 THE MARLOWES,
HEMEL HEMPSTEAD,
HERTS, HP1 1BB

Please commence my personal subscription YOUR COMMODORE with the .....issue

SUBSCRIPTION RATES (tick as appropriate)

UK £13.40 for 12 issues ☐

Overseas Surface £15.30 for 12 issues ☐

Overseas Air Mail £31.20 for 12 issues ☐

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)

VISA

Access

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Please use BLOCK CAPITALS and include post codes.

Name (Mr/Mrs/Miss).....
(delete accordingly)

Address .....

.....

Signature _____ Date.....

How to write a bestseller

Phil South lays down his golden rules on writing and selling your computer game.

If you, like me, enjoy a really cracking computer game, you might be harbouring the desire to write a bestseller game, and hook in to the fame and wealth bestowed on those who succeed in this marketplace.

Before we begin, I must tell you that writing a game is hard work. The most difficult thing is to have a truly original idea. Jeff Minter has made some very catchy games. But this doesn't mean that, to catch the eye of the software acquisitions manager at a big software house, you must copy his style. Most buyers would be singularly unimpressed by *Revenge of the Radioactive Guineapigs II*. It's been done, or as near as dammit.

What you need, as a potential software designer, is to know what these people do want to see in a game, and what procedures get you noticed. Do not spoil your chances through faulty planning.

Let's make it simple, by laying down the absolutely indispensable list of golden rules:

* pick a popular computer
* write in machine code
* be original
* document everything
* balance the difficulty levels
* use good music
* the program should be error-checked and user friendly.

Pick a popular computer. No software acquisitions manager in his/her right mind will buy a game if it works only on the Newbrain or Jupiter Ace. If you want your software to sell, you must make it work on one of the top four computers: the Commodore 64, the Sinclair Spectrum, the BBC B/Electron or the Amstrad CPC464 and its newer versions. The Enterprise and MSX micros could also be borne in mind.

The best bet, to my mind, for your first foray into games design is the C64. Not only is it colourful and highly 'games usable' in architecture, but it is the world's best-selling computer. If you write a good game for this baby, it'll go like hot potatoes, both in the UK and the US.

Write in machine code. This may seem obvious, but it's surprising how few programmers see the importance of this. No software house is interested in programs written in anything else. This should be your next step in grabbing their attention: learn assembly language.

This isn't as difficult as it might at first seem. All you need is to buy a good book on 6502 or Z80, called something like *Machine Language for the Absolute Berk*. Such texts usually have the games programmer in mind. So the gap between buying the book and submitting your first megagame is limited only by your capacity to read, digest, and apply the principles of machine code from the book.

The real reasons for using machine code? The obvious answer is that of speed of execution, meaning that your programs run much faster. Speed is a prerequisite of games, otherwise players find them too easy.

The other reasons, of course, are tied up with what you can do in machine code that you can't in slower languages. Fast animation, scrolling the screen in any direction, colourful and graphically detailed sprites and backgrounds are child's play, not to mention control of the SID chip, to produce beautiful music.

Yes, if you want to attract attention to your game, it must be fast, addictive, and — most importantly these days — graphically superior to most on the market.

Originality is all. You must strive to concoct an original scenario, as there is a lot of dross on the market, due entirely to lack of imagination. Think back to recent releases, and ponder how they might have come up with the 'environment' for the plot of the game.

Does the story take place in space? Or on the ground, in a desert, in a forest, in the past or the future? It is under the sea or in the air? Are you driving, flying, swimming or walking? Is it a magical place, or one where technology has broken down? Are there strange creatures or man-like robots?

These are just a few of the concepts that can cross your mind. Believe me, you'll know an original concept when it hits you. It'll grip your heart like a cold vice, and make you break out in a sweat; you won't be able to sleep.

Although the application of ideas requires a lot of talent, ways of getting ideas are simple. Pick up a dictionary. Turn to one or two pages at random, pick out a word on each, and write them down. Does the juxtaposition of these words suggest anything to you? Switch them round; does the meaning change? Try more words, and see if anything else springs to mind. This is a well-tried writer's technique to simulate ideas. Try it! Something may spring out of the murk at you.

Read some fiction. Take your favourite books, and steal a few ideas; see if they suggest anything to you in a similar vein. You'd be surprised how often ideas for bestsellers were found in this way. If you read much science fiction, some of the main culprits will be obvious! Haunt your library, scanning encyclopaedias and reference books, writing down everything you think might make a good game. Do all these things and something will turn up.

Top of the form. Before the software company will accept your submission, it may ask you to fill in some forms. They



contain vital information about you, your game, and the rights to your game, should it be bought. The horrid thing about not following the right procedures in submitting your piece is the possibility it might get sent back, without being looked at. This is a waste of your time and theirs. It's easy to avoid this kind of setback. Just give the publisher a ring. He/she is usually delighted to mail you the appropriate sheets of paper, and information about how to submit your game for scrutiny.

The kind of things your documentation should contain are these:

1. Name, address and telephone number.
2. The title of your program (it may be changed).
3. A brief summary of your game, with particular attention to any unique features.
4. A full description of the program.
5. Any special technical requirements, although it's advisable to avoid anything peripheral except a joystick.
6. The sales pitch. State why it should buy your program rather than any other. What does it have that the market lacks?

Remember, this documentation is your only representative on the aquisitions manager's desk. So make it as neat, organised and persuasive as possible.

You now have in your hands the most accurate information available about how to get your game not only noticed, but bought — and maybe make a successful career from software design.

To sum up, first and foremost you must have an original idea. That's where your success will lie. Come up with a fantasy that most people share. Most people play computer games to escape the mundane world they live in, so give them a fantasy to absorb them: sleek spacecraft, fast racing cars, elegant and graceful steeds, and beautiful maidens. Use your imagination.

Learn machine code. This cannot be overstressed, as without it you have a snowflake's chance in hell against the stiff competition. Take heart from the fact that if you have a reasonable competence in a language like Basic, you can easily make the transition to Assembly language.

The most addictive games are ones that make you use your mind. A large helping of strategy in a game makes it addictive and a real challenge, not just a mildly irritating wrist exercise. In the same way, don't make it so difficult that the player gives up. Playtest your game on your friends, and gauge their reactions to it, balancing challenge and attainment so that success is always just out of reach.

With practice, the player should be able to move on in the game. It's tricky, but a well balanced game is almost always a hit.

You are in this business to entertain, to create an illusion, like a cartoonist or movie maker. Make your graphics mobile and colourful, and the backgrounds suitable and well drawn. If you can't draw too well, what's to prevent a friend who can doing them for you? He'll get a credit on the cassette, at least. The same goes for music. The music must be good these days. If you can't do it, get a friend to do it.

A tune on a video game must be nice to listen to, not too repetitive, and you must give the user the facility to toggle it off. If the tune is unmusical or irritating, either the software house won't buy it in the first place or no one will be able to bear playing it for very long. You want to write a game that no one can leave alone!

Most of all, a program should be well error-checked, and user friendly. If you do your job, the playtesting should weed most of this out. This is most important because, if your program crashes continuously while the software house is looking at it, or it literally can't be operated without a degree course, you are sunk!

OK, that's it. Off you go and write a brilliant game. When it gets into the shops, I'll buy it!

WHAT THE SOFTWARE HOUSES SAY

'There's been a great increase in the quality of submitted software.'
'We've no plans to publish any adventures (in the Sparklers range): arcade games and simulations go down best.'
Gordon Reid, Creative Sparks

'We accept game designs only. They must have a minimum of 200 locations, detailed plot, and plenty of humour. It's as if we were evaluating a novel.'
Margaret Austin, Level 9

'We look for something that's as good as what other companies produce, that we can sell for £2.50 . . . anything that's fun.'
Tony Beckwith, Firebird

'It has to be original, with outstanding graphics. We're incredibly fussy.'
Simon Goodwin, Beyond

'We'll do anything if it's a good game.'
Paula Byrne, Melbourne House

What price a printer?

Jon Revis sets his fingers tapping on dot-matrix, daisy-wheel, inkjet and other new-technology printers, and advises on how to choose one for your needs.

There comes a time in every computer user's life when he or she wishes to put something on paper. It may be a listing from a long program, a copy of a screen from one of your games, or even a begging letter to your bank manager. Whatever the job, you will need a printer — but which one? There are so many types of printer on the market today, most of which are selling at vastly different prices. Hopefully, by the end of this article you should be familiar enough with the different types of printer, their strengths and their weaknesses, that you can part with your money knowing you have bought the right machine for the job.

The printer that we first take a look at is the dot-matrix printer. This is the most common type and is available for almost all home computers. The characters are printed using a vertical column of wires, the print head. The printer can fire one, or any combination, of these wires against the printer ribbon to give a vertical sequence of dots on the paper. Five such sequences of dots are needed to print each character.

Figure 1 shows how the letters E and T are constructed, by moving the print head in 10 separate steps from left to right across the paper.

The number of wires determines the quality of text. A greater number of wires enables the printer to print characters with more definition. This is one of the areas in which the difference in price between dot-matrix printers is most apparent. The dearer printers, such as the Epson FX80, print their characters using a 9×5 dot matrix. This produces a high-quality print. At the cheaper end of the market, printers such as the Seikosha GP80 or GP100 use a 7×5 matrix.

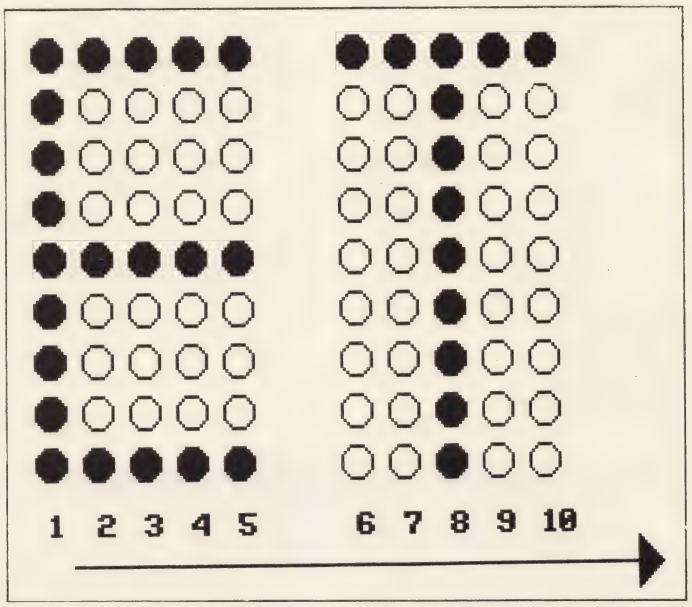
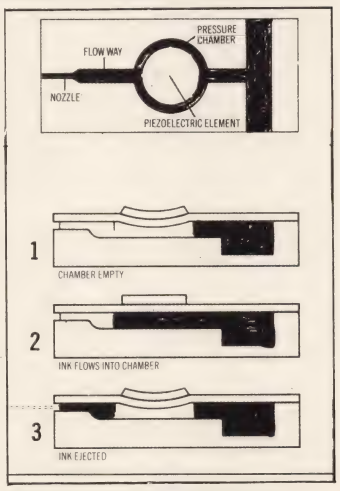


Figure 1: the print head moves in 10 steps

The most obvious difference between the two types is that the Seikosha cannot print letters with true descenders. This means that when printing lower-case letters with 'tails' — p and g, for example — the tails do not descend lower than the bottom of the other letters.

The quality of printout should be given a great deal of consideration when choosing a dot-matrix printer. If you are sure that you will never use your printer for anything other than producing program listings, then feel free to purchase the cheapest printer that you can lay your hands on. But should you intend to show other people the text that you produce — as in reports, articles, letters, or invoices — the least you want is a printer which can print characters with true descenders.



How Epson's SQ2000 inkjet printer works

If you feel a little daunted by the prospect of paying nearly £500 for an Epson FX80, as I did, take a look at some of the many Epson-compatible printers. These use the same control codes as the Epson, so can be used in place of the Epson with programs which tailor their output specifically. Printers such as the KDC FT-5001 for £200, or the Star SG-10 NLQ for £300, considerably undercut the Epson FX80 but produce excellent printouts. One very nice feature of the Star printer is its NLQ (near letter quality) mode of operation. The printer can redefine its character set so that the text produced is almost as good as that produced by typewriter or daisy-wheel printer.

Another feature limited by price is the speed at which printing takes place. Using the Seikosha as our example of a cheap printer, you will

find that its print speed is about 30 characters per second. This may be adequate for printing out listings of short programs, but it is unbearably slow when you are waiting for a 2,000-word article to be printed. A printer at the Epson FX80 end of the market bangs out text at a rate of 160 characters per second. Part of this increase in speed is because the print head prints when travelling from both left to right and right to left.

The final point worth noting about dot-matrix printers is that they can reproduce screen graphics when used with a suitable screen-dump program. This is possible because the characters are not fixed, as with a daisy-wheel. So the wires can reproduce the pixels lit on the screen.

Users of Commodore computers have several printers produced by Commodore. These do not need a special interface and can reproduce the Commodore graphic-character set. On the minus side, they tend to be overpriced and of poor quality.

A second option is to purchase another brand of printer and use it via an interface. One cheap alternative is the Alphacom 81. This dot-matrix printer produces 80-column printouts on 22cm wide paper, handles the Commodore graphics character set, and prints characters with true descenders — all for £99. The accompanying interface can be bought for £25. To prove that shopping around can pay dividends, I recently saw the Alphacom in a local branch of Curry's at a special price of £49.

The second type of printer which we are covering is the daisy wheel. Due to their mechanical complexity, and corresponding high price, daisy wheels have until now been found solely in the office environment.



Star's SG-10: undercutting the Epson FX80

The characters are moulded onto plastic blocks on the ends of the spokes of a wheel which resembles a daisy (hence the name). The wheel is rotated until the required character is in position. The spoke is then struck from behind, causing the character to hit the back of the ribbon and print the character on the paper. Using moulded characters enables the printer to produce text which is of the same quality as a typewriter.

One fiddly point about the daisy-wheel design comes to light when you wish to alter the typeface in mid-document, for example printing one paragraph in italics. With a daisy wheel you must stop printing, and replace the present daisy with one containing italic characters. Having printed the paragraph, printing must be halted again to swap the daisy wheels again. This problem does not occur with a dot-matrix machine as a printer control code can be embedded in the text. When this is encountered, the printer is instructed to change the way in which it prints its characters and produces an italic typeface until further notice.

The main drawback with daisy-wheel printers is speed. One cheap daisy wheel which has been recently released, the £329 Dyneer DW16, prints at the phenomenal speed of 16 characters per second. If quality of printout is your only need and you are willing to wait for a quarter of an hour or so whilst a document is printed, then by all means plump for the daisy wheel.

Commodore micros need a special interface to benefit from good printers



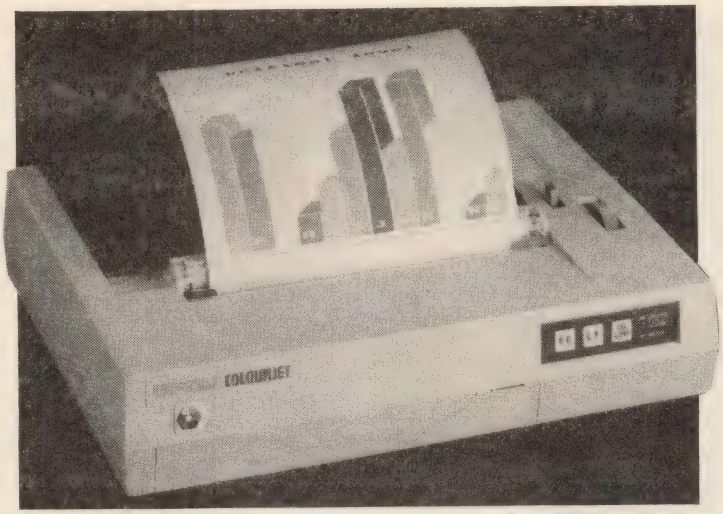
One final point before we leave daisy wheel printers is the fact that they can't produce screen dumps. Every character is constructed as a moulded piece of plastic, so they cannot produce the individual dots needed for individual screen pixels.

Moving away from the traditional type of printer, let's look at some of the newer designs. The first of these is the inkjet printer. One printer from this category is the £575 Integrex Colourjet. As you can probably deduce from the name, the Integrex produces multi-coloured printouts. The best that you can expect from any of the dot-matrix printers mentioned is a screen dump in several shades of grey.

The inkjet printer constructs its characters in the same way as a dot matrix by building each character from several columns of dots. But its method of producing the dots is different. The inkjet printer has no ribbon, but holds coloured inks in powdered form in a cartridge which is moved backwards and forwards across the paper. An electrical discharge vaporises small quantities of the ink, which is then expelled through a fine jet onto the paper, creating a dot.

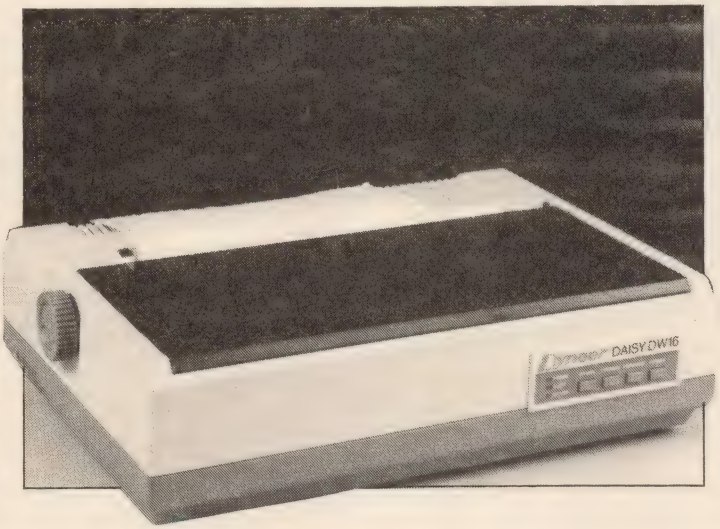
The inkjet method of printing is very quiet compared to dot-matrix and daisy-wheel printers. On the minus side, the printing speed is very slow and the quality of the printout can be affected by the absorbancy of the paper. If the paper used is too absorbent, the image may 'blot'. If it is too glossy, the characters may merge. When used with the correct paper the Integrex Colourjet can produce results comparable with the better dot-matrix printers, and all in glorious technicolour.

Another device based on the electrical discharge principle is the latest addition to the Epson range, the P-40P which is priced at £99. Unlike the inkjet printer, the P-40P has no ink cartridge. It directs its spark directly at a roll of special heat-sensitive paper.



The Integrex Colourjet: expensive but in glorious technicolour

The Dyneer DW16: only 16 characters per second



The unit is very compact (21 ×13 ×4cm) and is powered by built-in rechargeable batteries which give power for six hours' printing. Its diminutive size makes the P-40P an ideal choice for the businessman to use with a portable computer. The major limitation with this size of printer is that the paper is only 11cm wide, and expensive.

Readers may remember the thermal printer produced by Uncle Clive Sinclair for use with his computers — fortunately, the P-40P does not produce the same 'smudge matrix' printout. The little Epson gives the quality users have come to expect from such a stable. The default printout is 40 characters wide. But when used in condensed-text mode it can happily fit 80 readable characters on a single line, and still retain its descenders!

Speedwise, the P-40P is not the fastest printer I have met. But for quality and portability it is definitely a winner.

The third offering in the 'unconventional' category is the CPP-40 from Micro Peripherals. It is a multi-coloured portable printer costing £79. As with the Epson P-40P, the CPP-40 has a built-in rechargeable battery and comes with a mains adapter.

The CPP-40 uses 11.5cm wide paper, and prints using a series of four pens of different colours. These are held in a rotating barrel. A set of dip switches on the bottom of the machine can switch between 40- and 80-column printing modes. The quality of the printout is very impressive. Even in 80-column mode the characters are well defined, though users with below-average eyesight may have difficulty reading the text.

If you want to buy a printer the best advice that I can give is to describe what you intend to use the printer for. The daisy wheel produces a very high-quality printout but can print only text, not screen dumps. The dot matrix is an excellent all-round machine but the standard of printout can vary tremendously; one of the Epson-compatible printers would be a very strong favourite for this category.

At present, the other types of machines are very much minority groups. Until they increase in popularity, the standard of after-sales backup may be a little suspect. And the availability and cost of the special paper and ink cartridges must be taken into consideration.

No matter which printer you decide to buy, if you have read this article carefully it will serve you well for many years to come.

Win the modem of the year!

Access the world's information technology with the WS2000.

Miracle Technology's WS2000 modem was the only one to reach the finals of the British Micro Computing Awards 1985 Peripheral Of The Year, the 'Oscars' of the industry. In both business and home markets, it has outsold and outperformed its rivals. And its versatility has led to uses as diverse as sending data to HQ on Operation Raleigh, carrying despatches from war correspondents for CBS News, and helping to track basking sharks via satellite for BBC programme *Wildtrack*.

You can see most of the WS2000's features in the photograph below. It is multi-speed: 300 baud full duplex, 600 and 1200 baud half duplex, and 1200 baud with 75 baud back channel. So you can access Prestel/Micronet, Telecom Gold, bulletin boards and other users with it.

It can switch between CCITT standards for countries adopting the European communications standard, and Bell standards for countries adopting the US standard. These cover nearly every country in the world.

You can control the WS2000 from a software program featuring auto answer and autodialling. Or you can buy it as a manual-selection modem for £154.73. Interfacing it to your BBC, Atari, Commodore, Spectrum, Apple or IBM micro costs from £30 to £60, depending on make. Now you have a chance to win one for yourself, opening communications worldwide!

All you need to do is fill in the answers to the crossword below and send them with the coupon to us. Each clue consists of two parts: the first is a cryptic clue, and the second is a straightforward definition whose answer can be found in a good computer dictionary. The first correct entry drawn on the closing date wins the modem complete with relevant interface.

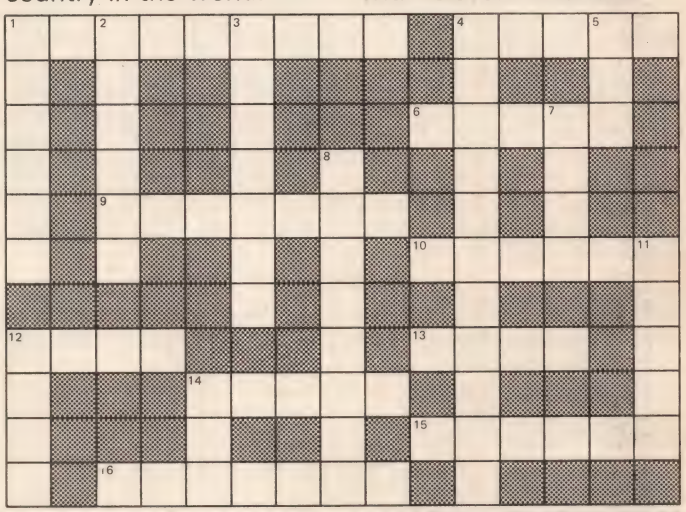
Across

1,4 down. Regulator with reverse stand — modem stands for it (9/11)
4. Lower social group — but standard for games playing (1-4)
6. Clever: a modem which dials, redials, auto-answers or disconnects and defines data formats (5)
9. See 2 down.
10. Dupe is outside left at ten — carrying two channels simultaneously (6)
12. See 14 across.
13. Upper class nips in bad surroundings — 300, 600 and 1200 per second (4)
14,12 across. Begins complex crossword in target time before clanger; or goes from European communications standard to US standard (5,4)
15. Accommodates after the bulletin (6)
16. St Peter's there — and Miracle Technology lives there also (7)

Down

1. Documentation done by hand? (6)
2,9 across. A straight link — this modem hooks up electronically (6,7)
3. 12 down, 14 across — initials form a word (7)
4. See 1 across.
5. It's a drug, this short variable resistor (3)
7. Fish for a tape-bearing spool? (4)
8. Hearing — the coupler which connects through signals in the air (8)
11. The unknown Pole, going into a horizontal dimension (1-4)
12. Incomplete baby goes to a model; but maybe this body will give approval to other modems (4)
14. Starts chasing poor schedules — measure of a printer's speed? (3)





NAME . . . . . .

ADDRESS . . . . . .

. . . . . POSTCODE . . . . . .

I own the following micro: . . . . . .

To: WS2000 Modem Competition, Personal Computing Today, 1 Golden Square, London W1R 3AB.

COMPETITION RULES

1. Closing date for the competition is 30 September 1985. Entries postmarked after that date are ineligible.
2. Entries will not be accepted from employees of Argus Specialist Publications Ltd, their design house and printers, or from those of Miracle Technology. The restriction also applies to employees' families.
3. Entries must be on this coupon; photocopies are ineligible.
4. Entries should be addressed to WS2000 Modem Competition, Personal Computing Today, 1 Golden Square, London W1R 3AB.
5. The winner will be notified by post, with the result published in a future issue of *Personal Computing Today*.
6. The editor's decision is final and no correspondence will be entered into.

Details of Miracle Technology's modems can be obtained if you write to them at St. Peter's Street, Ipswich IP1 1XB.

Blackboard

Phil Tayler introduces the latest robotics package into his school, with unexpected results.

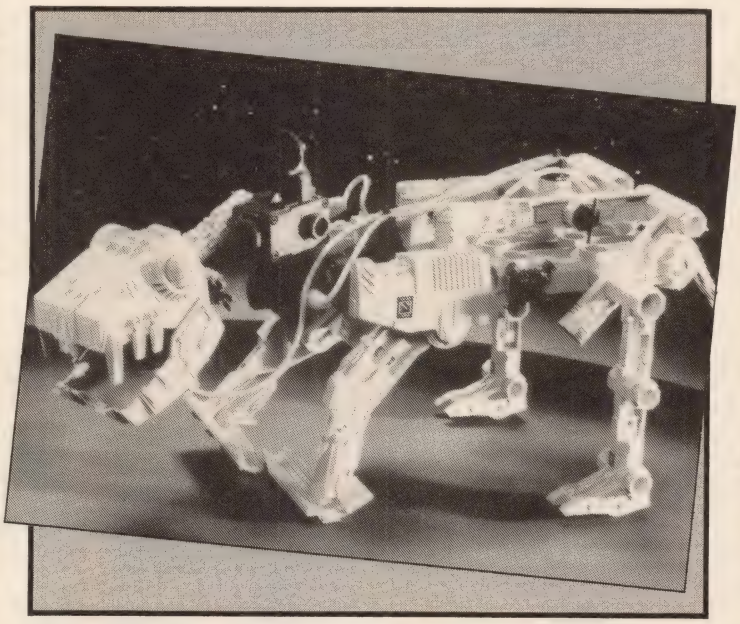
I want to write Blackboard this month without mentioning computers at all! Those who know me realise I have been programmed to mention computers at least once a minute, so let's see how far we get. What has distracted me from my keyboard? I've recently been playing (!) with an interesting package from toy manufacturer Milton Bradley, perhaps better known as MB Games. It is one of a set of packs under the general title Robotix, and allows children to design and build robot-like figures, which use bi-directional motors.

The potential of the kit is huge. The set I reviewed has four such motors which can be used to power wheels, legs, a winch and so on. There are also many interlocking plastic pieces, which have stood up perfectly to the very close attentions of my class, and which can be made up in an almost countless number of shapes.

The power from four batteries is taken to a neat control console which allows five motors to be controlled independently or simultaneously — there are add-on packs consisting of additional motors and constructional items, allowing vastly complex designs. The motors may all be used on one model, or in a combination which can make up a scene.

Design became interesting to the children as they played, and I feel the kits will prove very valuable in schools. Their price will not make the kits top sellers in the home market. The kit I received was priced at £50; other less expensive kits are also available, with less motors and other parts.

There is a possibility of an interface to a micro (there, I've mentioned one!). This would greatly improve the package, as would slightly longer power cables to the motors. But overall the kit provides an interesting and enjoyable lead into the field of robotics.



The kit raises an interesting point. The plastic interlocking pieces which make up the figures and machines are rather reminiscent of *Star Wars*, and calculated fairly accurately to appeal to boys. In my own family, my two sons were both immediately interested, but my daughter showed little enthusiasm. This position was to an extent mirrored in my classroom. The boys tended to build models more quickly, and with more imagination, perhaps because they have had more experience with constructional toys. That in turn made me wonder why it is that girls seem less likely to succeed on the school micro in the average primary classroom.

My class of eight-and nine-year olds have spent some time lately making up screens about their town on a Viewdata program, for these to be displayed on Open Evening. The fourth years are also involved, and have produced imaginative and colourful pages. Most of the successful designs are those of boys. I would imagine the picture is repeated in many schools, and I'd be interested to hear from any schools who have ideas or views to share on the subject.

I've just updated our class lists, which we hold on disk. Actually, they are held on a file which loads into *Inform*, a program I have made reference to before. We don't hold any private information about the children — which would be awkward, anyway, under the Data Protection Act — but merely their name, class, year, sex and date of birth. The latter is held in reverse order (year-month-date) so that sorting can be done either alphabetically or chronologically. With a printer attached, a list is instantly available of all boys, all second years, all infants or whatever.

The field of administration is an exciting one for the school micro, and not only with datafiles. The office must issue dozens of nearly identical 'standard letters' over a year, and a word processor will surely be a part of every school office. Again, how does the micro help in your school adminstration? Which programs do you use, and what are they like?

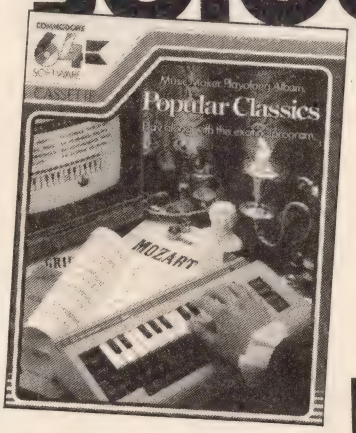
This month has raised a number of questions, and your views are always welcome. I'd like to feature a few letters on the page soon, so write in with your problems, brainwaves or whatever! The Blackboard page awaits.....

Phil Tayler is deputy head teacher at Holland Park County Primary School, Clacton-on-Sea.

Software selection

Title: Playalong Popular Classics
Software house: Commodore (UK), 1 Hunters Road, North Weldon Industrial Estate, Corby, Northants
Price: £9.99, cassette

Machine: Commodore 64



I have always been a fan of classical music, but my delight on receiving this program turned to disappointment after loading it. Fortunately, as I continued playing with it, it became more and more interesting.

Basically, it plays jazzed-up versions of twelve classic tunes. When you get bored with the computer's rendition, you can adapt it in various ways. You can follow an illuminated keyboard on screen with a preprogrammed accompaniment or you can choose your tempo and watch synchronised graphics.

I have mixed feelings about this rather expensive package which includes a music book and hints on playing. I didn't like some of the computer's adaptations of the classics and found the 'single key' and 'rehearsal' modes unresponsive. I did enjoy adapting the music myself and the graphics are good.

If you think this package appeals to you, I recommend you try it before buying. **A.B.**

STAR TABLE	
Screen display	* * *
Usefulness	* * *
Ease of use	* * *
Overall	65%

Title: Car Cure
Software house: Simtron, 4 Clarence Drive, East Grinstead, West Sussex RH19 4HZ
Price: £9.95, cassette

Machine: Spectrum 48K

Car Cure gives a step-by-step guide to finding the faults with your car, and suggests 'cures'.

While the program is loading, a maze is shown. This maze is different each time — a good idea but I found it a bit hard to follow.

The program can begin with either the symptoms or parts section. The symptoms section gives a list of, obviously, symptoms. These become progressively more detailed. The parts section lists possible sources of the fault. When one part has been identified as the problem, a list of faults appears. Decide which fault applies to your car, and the computer suggests the action to be taken.

The procedure may seem complicated, but the commands are shown continually on screen. Each section has a different background colour for easy identification.

The program is designed for everyone, from beginners to experienced drivers. In some places, the information is basic and in others advanced. You need a good knowledge of the engine and bodywork of your car to check the parts and look for faults given in the program.

Car Cure could be improved by including diagrams, and by dealing with specific makes of car. Overall, it is a good fault finder, but could do more to explain how to check and repair your car. **A.B.**

STAR TABLE	
Screen display	* * *
Usefulness	* * *
Ease of use	* * *
Overall	75%

Title: Kon Tiki
Software house: Golem Ltd, 77 Qualitas, Bracknell, Berks RG12 4QG
Price: £10.95, cassette (£12.95, disk)

Machine: BBC Model B

One of the most important criteria when I review educational software is the answer to the question 'Could this be done better any other way?' Well, in this particular program the answer is undoubtedly 'Yes', for it attempts to simulate the journey made by Thor Heyerdahl on his raft. Perhaps the authors have deliberately set out to show how terribly tedious the whole journey was, with over a hundred days at sea. If this is so, they have succeeded brilliantly, for the sheer boredom certainly came through to my children!

Each group has 15 minutes at the keyboard, starting from a previous position if they have already saved one using their identification code. Rather sadly, their participation consists merely of entering a direction for the steering oar, with the computer then deciding a new position.

Occasionally the children have the excitement of a short extract from Thor's diary being printed on the screen, to copy into their books. I am afraid that it appears to me to be an absolute waste of time, and an excellent way of showing children how boring computers can be.

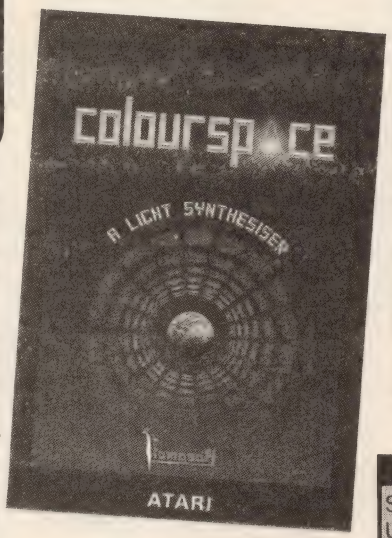
The program is accompanied by a poorly duplicated A4 booklet, which could have been very useful, containing maps, drawings and notes on the raft and animals the children might meet.

All in all, it is the sort of program that I tend to delete from my disks, as the space itself is more valuable to me than the program. **P.T.**

STAR TABLE	
Screen display	* *
Educational value	*
Ease of use	* *
Overall	35%

Title: Colourspace
Software house: Llamasoft, 49 Mount Pleasant, Tadley, Hants
Price: £7.50, cassette

Machine: Atari



Colourspace is the Atari version of *Psychedelia*, a light synthesiser designed to accompany music. But to do justice to the Atari's display chip, the program includes many more features.

I was slightly apprehensive about reviewing a program written by someone whose favourite word for the week was 'zarjaz' and whose aim seemed to be to freak people out, but *Colourspace* is excellent.

The program allows two types of drawing, foreground and background, so you can keep a display in the foreground while creating the background in time to a piece of music.

As well as drawing, *Colourspace* can produce interesting effects. These include stroboscopics where the current colourform (better known as ink!) is flashed several times a second and dynamic colourflows (sometimes referred to as ooze) where the colourflow 'pulsates' through a given rhythm of colour.

The program has 80 'presets' which are used to store colourflows. These can be called up with a single keypress. Although preset with the author's own creations, they can be redefined by the user. Screen resolution and screen symmetry are also controllable.

I can't think of many practical uses for the program but it is great fun to play around with. I know I'll load it up whenever I want to demonstrate my Atari's zarjaz graphics to my friends! **M.D.**

STAR TABLE	
Screen display	* * * * *
Usefulness	* *
Ease of use	* * * * *
Overall	85%

Title: Microscript
Software house: Amsoft, Brentwood House, 169 Kings Road, Brentwood, Essex
Price: £49.00, disk

Machine: Amstrad CPC464, CPC664

Microscript is a new word processor program for the Amstrad. The disk comes with a large manual giving all the instructions needed to use the package.

As with all word processors, the micro becomes an intelligent typewriter, and can be used for letter writing, reports, invoices and statements. In fact anything that you might use a typewriter for can be done with this package — and a lot more!

The program supports all the usual features such as creating the document, formatting the text on screen, corrections to the text, reformatting the document, tabs and columns, insertion, deletion and movement of blocks of text, merging files, printing the documents, file handling on the disks, changing codes and functions, and the ability to do calculations and five functions with the calculator up to 14 digits long.

The manual could have been a little clearer in explaining some aspects of the program, but with a little patience and practice there won't be too many problems.

The only real drawback with the program is its price. Apart from file handling, I found that it did no more than programs costing about £20, and so it is difficult to recommend. **P.S.**

STAR TABLE	
Screen display	* * * *
Usefulness	* * *
Ease of use	* *
Overall	70%

Title: Sheep Dog Trial
Software house: Jacaranda Software, Baffins Lane, Chichester, West Sussex
Price: £19.95, cassette

Machine: BBC Model B

Each player controls a sheep dog and sends him out into the field to muster a group of sheep the same colour as the dog. The sheep must be herded into a pen. This is done by typing in grid references, to which the dog moves. The computer keeps count of the number of moves your dog makes, and how long it takes you. Just as in a real sheep dog trial, time is of the essence.

When this was tried in the classroom it was interesting to watch the children developing round-up skills. They started by moving the dogs haphazardly, but soon realised that certain techniques worked more effectively, like grouping sheep first and then moving them in a bunch. There are difficulties too; if one dog lands on a square next to another dog, they'll fight . . . and the dog who starts it gets sent back to the paddock!

A nice touch is that occasionally both players have to work together, rather than against each other. For example, when a group of different coloured sheep get together, it is difficult for one dog to split them up.

The game was most successful with children at the top end of junior school. Younger children like pictorial variety in their computer diet, and this game has only one screen. It teaches little about grid references, but is valuable in the area of manoeuvre planning and quick thinking in order to beat the clock. **M.K.**

STAR TABLE	
Screen display	* * *
Educational value	* * *
Sound	* *
Overall	65%

Title:	GraphiQL
Software house:	Talent Computer Systems, Curran Buildings, 101 St. James Road, Glasgow G4 0NS
Price:	£34.95, microdrive cartridges
Machine:	Sinclair QL

The QL's mode 8 colour screen display offers 256 x 256 pixel resolution; each pixel can be one of eight colours. With this range, the QL has the capability of producing some quite spectacular displays. For the budding artist, the prospect of producing high-resolution colour pictures on the QL is quite daunting, but *GraphiQL* makes it simple.

GraphiQL provides the user with about 50 high level commands to enable complex pictures to be produced with relative ease. The instruction booklet is easy to follow, and little practice is needed to master the program.

GraphiQL is supplied on two microdrive cartridges. One contains the main program and the other three demonstration pictures which, on a colour monitor, are truly impressive.

The program's positive features include colour and texture filling of shapes, a magnification facility and rubber banding. On the negative side, there is no on-screen response to key presses (an annoying omission, at times), a printer dump facility which cannot be accessed from the main program, and the price.

While QL software producers appear to expect a premium price for their products, £35 for *GraphiQL* is about four times greater than similar products for other home micros, which is ridiculous. **D.N.**

STAR TABLE	
Screen display	* * *
Usefulness	* *
Ease of use	* * * *
Overall	30%

Title:	Micropen
Software house:	Amsoft, Brentwood House, 169 Kings Road, Brentwood, Essex
Price:	£49.00, disk
Machine:	Amstrad CPC464, CPC664

Micropen is a new database program designed to run from disk. It is a very powerful package that can store amazing numbers of records because it is limited not by memory size but by the size of the disk.

The disk contains three programs: one for creating and maintaining databases, one for repairing or altering the system and dealing with faulty files, and the last is a word processor for creating input and output forms and sending out mail shots or standard letters containing information from the database. Also contained on the disk is a sample or demo program.

You have complete control of each layout and the fields in that layout. However, no more than 100 fields or no more than 1024 characters may be stored for each layout. But an amazing number of 32750 records are allowed on one database (an advantage over tape-based databases!). You are also able to edit and amend the screen layout and records, and print out and search for data or records.

The program is quite easy to use, but again the manual takes for granted that you already know something about using a database program. But patience and thumbing through the manual a few times should help.

Once again my only reservation is the price — all £49 of it — surely it can't be justified. **P.S.**

STAR TABLE	
Screen display	* * * *
Usefulness	* * * *
Ease of use	* *
Overall	80%

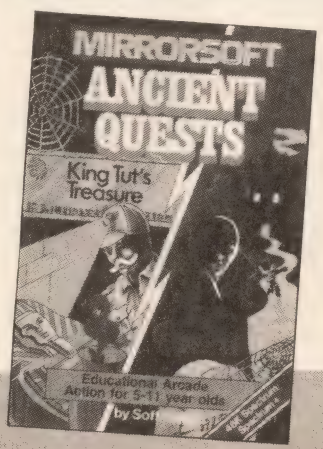
Title:	Ancient Quests
Software house:	Mirrorsoft, Maxwell House, Worship Street, London EC2
Price:	£7.95, cassette (£10.95, disk)
Machine:	Spectrum 48K

Ancient Quests comprises two quite separate and involved games for children aged between five and nine.

King Tut's Treasure is the better of the two. It involves moving Professor Diggins and his metal detector around an ancient ruin, making him dig up objects when his detector bleeps. There are many scenes, and once you move into a setting where an object is buried, a 'clue scale' helps identify the exact spot for the dig. When all the objects have been retrieved they are returned to the door where the game starts, and the door rumbles open to reveal the buried treasure.

There is an assortment of 'creatures' in the various locations. These must be avoided, and it's not a good idea to crash into walls, palm trees or pyramids, either, because you have only four lives and it's all too easy to end up in the Chamber of the Dead!

The 'objects' form the educational content of the game, and there are various options of shape matching, simple fractions and decimals, matching shapes with names, and so on. Every ability level is catered for. There are easy or hard games, four or six objects to collect, and optional help in getting out of the Chamber of the Dead.



The Count (Dracula, of course) suffers from brief and inadequate instructions, especially for children. It is similar to the above game in that a character is moved around (a castle this time), and there are six problems to be solved involving counting and the four basic maths rules. The graphics are good but the game is a little confusing to play at first.

These games won't teach children a great deal, but they are good for reinforcement of skills . . . and fun to play. **M.K.**

STAR TABLE	
Screen display	* * * *
Educational value	* * *
Ease of use	* * * *
Overall	80%

Title: The Code Machine
Software house: Picturesque, 6 Corkscrew Hill, West Wickham, Kent BR4 9BB
Price: £19.95, cassette

Machine: Amstrad CPC464

An assembler is a program which converts assembly language mnemonics (vaguely understandable) to machine code (complete gibberish). It is for this reason that an assembler is an absolute necessity when developing machine code programs. *The Code Machine* consists of two programs, an assembler and a machine code monitor.

Using the assembler program is a two-stage process. First you enter the assembly language mnemonics using an editor program. These mnemonics are known as the source code. The code is entered along with line numbers and can be edited, renumbered or saved in the same way as a Basic program. The second stage is the actual assembly of this code to produce the object code, which is the true machine code.

The monitor program becomes vital when your new machine code masterpiece grinds to a halt for no apparent reason. The monitor provides facilities to execute your program step by step while displaying the contents of each of the CPU's registers. A faster method of achieving the same result is to insert a breakpoint into the section of code which is most likely to be the cause of the problem. The program will execute until this breakpoint is reached, then stop and display the CPU's registers.

The disassembler part of the program converts object code back to assembly language mnemonics. This allows the user to examine both own and other people's machine code programs. However, it refuses to disassemble itself (surprise, surprise!).

The Code Machine is a functional package with few bad points, but it lacks the polish of some of its rivals. Take a look at *Maxam* by Arnor before making your final decision. **J.R.**

STAR TABLE	
Screen display	* * *
Usefulness	* * *
Ease of use	* * *
Overall	65%

Title: Invostat
Software house: Amsoft, Brentwood House, 169 Kings Road, Brentwood, Essex
Price: £29.95, cassette

Machine: Amstrad CPC464



When Amstrad released the CPC464 it intended that it should be used in some capacity as a business machine. It has now released programs aimed at the small business; *Invostat* is one such project.

Invostat will produce invoices, statements and credit notes for your business which can then be printed and despatched to your customers. The program is menu driven, providing the non-computing businessman with easy access to all of the program's facilities.

When reviewing this type of program I always look at its crashability rating. This is important where company records are at stake; in the case of *Invostat* I found it impossible to crash even when stooping to my dirtiest tricks.

Documentation is provided in the form of a slim eight page booklet, the contents of which are both concise and informative. Within 30 minutes of loading the program the user should be quite capable of producing the whole range of documents available.

Each of your customers has a client file in which all previous transactions are stored. Having accessed a particular client's file you are at liberty to view in detail any of the previous transactions. These are professionally presented in 80-column mode and can be sent to the printer if required.

All of the documents produced can be printed immediately or the information stored on tape for printing at a later date.

Invostat is an easy to use yet very professional product, well worth the consideration of any business user. **J.R.**

STAR TABLE	
Screen display	* * * *
Usefulness	* * * *
Ease of use	* * * *
Overall	90%

Title: Dragon World
Software house: 4 Mation, Linden Lea, Rock Park, Barnstaple, Devon
Price: £18.40, cassette (£20.30, disk)

Machine: BBC Model B

4 Mation has been steadily building a reputation for excellent software for the BBC, and this program extends it further. It is in many respects a sequel to their enormously successful *Granny's Garden* adventure.

The children have to try to return to Dragon World after a terrible fight between the dragons over some musical teeth. These teeth need to be located, and many other trials lie ahead for the children as they search for the treasures to make the dragons happy once more. As in its previous adventures, 4 Mation makes excellent and extensive use of mode 7 graphics which appeal greatly to children. Sound is also very important in the program, and there is a separate composing program which is quite interesting to use, and quite powerful.

As always, the documentation is first rate. Included in the package are a teacher's guide, a book full of ideas and the script of the beginning of the adventure. An audio cassette of the story is also included, and there are overlays for use on the Concept Keyboard (if you've got one!).

The age range suggested by the authors is six plus, although I feel that infants may find some of the ideas a little difficult. Certainly, my eight- and nine-year olds are totally involved, and the program is leading to much creative and language work, as well as other ideas touching all areas of the curriculum.

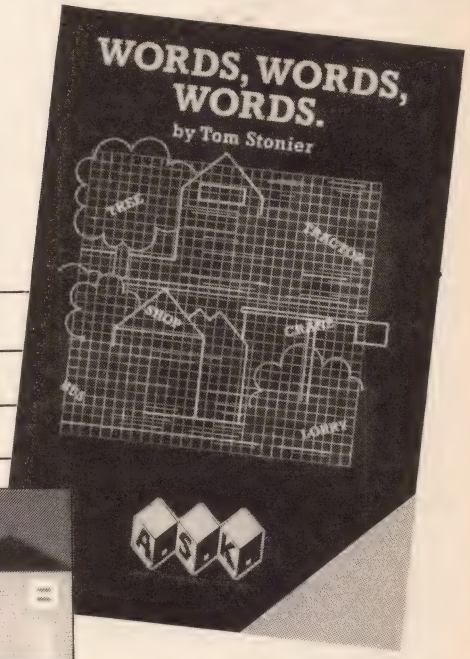
The screen displays are clear and easy to read, and the children are able to move through the stages with little interference from adults.

I reviewed the disk version, which allows groups to start without all the fuss of tape loadings. I would suggest the extra cost is well worth while.

If you are looking for an adventure which will stimulate and excite your class, you will not do better than this. **P.T.**

STAR TABLE	
Screen display	* * * *
Educational value	* * * * *
Ease of use	* * * * *
Overall	98%

Eureka!



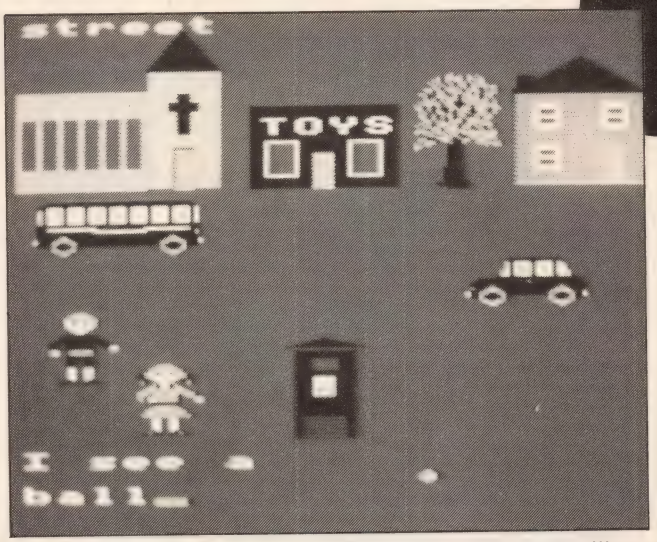
Phil Tayler hunts for good language software for children of infant- and junior-school age, and nominates his favourites.

Some language software has proved to be most effective in the classroom, and many packages have been well used in our infant department.

Happy Letters by Bourne gives levels of letter recognition, with options to link upper/lower case, or to show the relationship between the letters on the screen with the keyboard. The program contains an extremely valuable monitor screen, which shows the adult the responses made by up to five children, with the mistakes clearly displayed. The letters needing extra assistance can then be easily identified.

Another use of the computer in classrooms is adventures or simulations, in which the children can discuss their plans to succeed in a world they could not experience in their real lives. Until recently, there have been no obvious adventures designed for infants, but I have been looking at a very simple concept by Eagleswing. *Princess Rebecca* is exceptionally easy for young children to use, as the only response on their part is through the cursor keys. They must follow the yellow brick road as they search for the princess.

A map of the road is printed in a text window at the bottom of the screen. The main part of the screen shows some of the best mode 7 graphics I have seen in educational programs. As the child moves round the scenes change. They also vary, depending on the state of the story. For instance, the wizard is not present the first time the child visits his house, but the next time he is able to give the child a spell.



Most infants should be able to move through the program fairly quickly, and then want to write about and draw their adventures. The oral work which would ensue is also most valuable.

Another adventure which many other teachers have found excellent in use is *Granny's Garden* (BBC/RML) by 4Mation. I have reservations. There are instances when a question is asked — 'Do you want to go in the cave?' — and some children wish to say 'no'. But the program will not allow such divergent thinkers, there only being one way through the story.

Database Publications publishes *The Magic Sword* for the BBC, Electron and Amstrad micros, and this again is ideally suited to infants. The hero on this occasion is the delightfully named Prince Fred, who must rescue Princess Poppy. Again there are clues to be had, but this time from the book which comes with the disk or tape. The graphics are bold and colourful, rather than beautifully detailed, and the child's response is made from a simple range of keys. It aided my children in their early attempts into adventures, having none of the frustrations of other programs, and is heartily recommended.

Podd from ASK has received much praise in the educational press, being an extremely original idea which children respond to well. I was not particularly impressed at first, but the children have taken to it with enthusiasm and determination. Podd is a happy little man who knows about 100 actions, and shows those that the child types correctly. The amount of discussion and thought which results makes this a good option for the classroom.

For further spelling practice, I suggest that teachers look at *Words, Words, Words* from the same software house. The locations each have their own vocabulary. Objects are drawn in beautiful detail, then the child enters a word (there are often alternatives). When all the objects are present, the scene becomes brilliantly animated, with one particular object moving to the next set. All stimulate much oral and written language.

Moving to slightly older 'junior' children, wordprocessors have possibilities ranging from ▷

presentation of stories to exercises in procedure, or anything between. But there are few teachers with the time or inclination to become fully conversant with a complex and sophisticated WP program like *View*, from Acornsoft.

We come to the main question to be asked about such a program: how easy is it to use? Criteria include screen layout, documentation and the presence or otherwise of helpful features, such as justification and block moves.

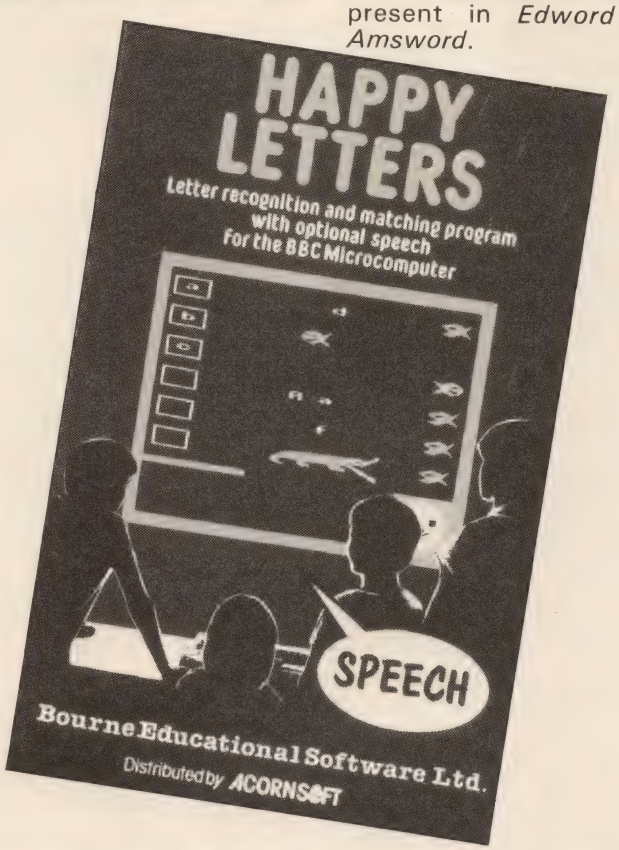
I have been using *Edword* by Clwyd Technics on the BBC for a while. It is a Rom-based program accompanied by a battery of exceptionally clear and detailed booklets, which cater for most teachers' needs. Most will do well to work logically through the spirally bound self-instruction guide, in which it is suggested that each page is read and commands entered exactly as written, with no pages skipped. This is sound advice, for there is a wealth of information to take in. 240 pages later, the reader may move on to the more concise reference guide.

I use it in conjunction with a disk system, and documents are automatically saved on completion. The screen can be formatted with options of either 40- or 80-character width. One most useful feature is an ERROR/HELP feature on f4, which points out the cause of any error. Text may be emphasised or underlined, and may also be moved easily. There is a word-search facility, with an option to replace one word by another.

Edword has been written specifically for use in schools, and the expertise to operate it smoothly and confidently comes surprisingly quickly. It can do everything you are likely to want in a classroom, although it is limited to modest documents, holding about five pages in 80-column mode. And being in Rom, it is instantly available.

Incidentally, I write my articles on an Amstrad CPC464, using the excellent *Amsword* program. It has all the features needed to make it user-friendly, including automatic right-justification and word-wrapping, so that words do not end up stranded across two lines. In addition, the user can customise the program, and save the new version to tape/disk. Again, it is unable to handle long documents, but all the *Edword* 'block move' facilities are present, allowing text to be manipulated almost indefinitely. There is a superb HELP facility on ESC, and a window of information at the top of the screen. Amsoft also produces a cut-price version, called *Easi-Amsword*, but it does not compare with *Amsword*.

A later arrival on the scene is *Wordpower*, from Ian Copestake, available soon for the BBC micro, on disk or diks and Rom. Some readers might have used *Wordsworth*, the previous word processor from the company, a program which gained some popularity in schools. This version adds several more sophisticated features to those present in *Edword* or *Amsword*.



I strongly recommend the disk and Rom option, as the disk alone offers only a 40-column display on a BBC, because of memory restraints. But all the additional memory from a second processor, or on a BBC+, can be used to make all modes available. On the standard Beeb, far more in evidence in schools than any enhanced models, the disk option provides a much reduced text capacity — as little as 1,250 words. With the Rom also fitted, the text capacity moves to nearer 4,000 words, and 80 columns are possible on all models.

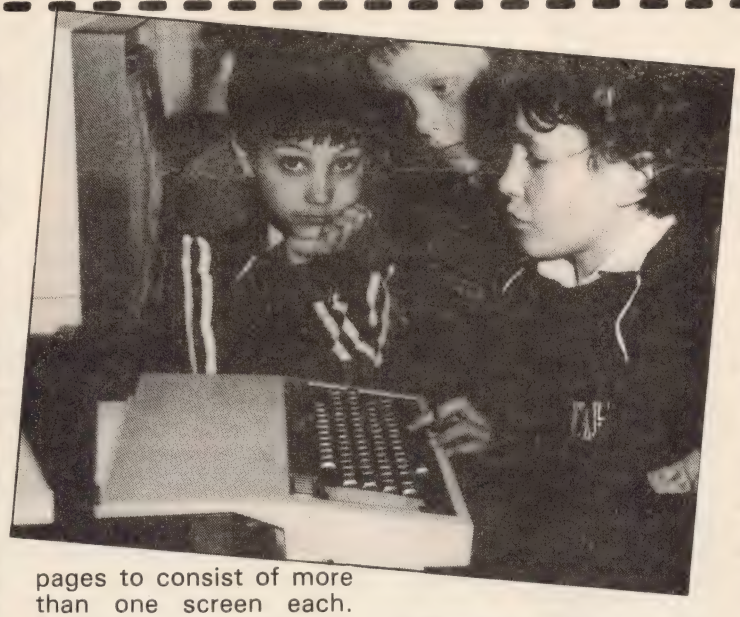
It is possible to write in columns up to 250 characters, should your printer be able to handle this, using different formats during a piece of writing. This is achieved by defining rulers, which decide the text width, in conjunction with an advanced system of 'line slicing'. Character widths depend on the mode selected. But with the disk only, no option except mode 7 is practicable.

I must confess that I found *Wordpower* much less user-friendly than either of the others, perhaps because of its broader range of features. On several occasions, altering the text produced strange effects elsewhere in the piece, such as spaces in the middle of words, and some failing with the word-wrapping. I managed to sort out these problems. But it could cause problems for many less experienced teachers.

I appreciate that this program is not aimed especially at schools, but after the success of *Wordsworth* in some schools, I have looked with a practising teacher's eye. The market for whom it is written will doubtless enjoy its additional features, such as the ability to perform background printing.

I was quite interested to receive a copy of *Tombs of Arkenstone* from Arnold Wheaton Software. This BBC disk can be used either as a straight adventure (yes you've guessed, in the Tombs of Arkenstone!) or as a vehicle to make up your own adventure, set in a time and place of your choosing.

My enthusiasm soon evaporated, for the adventure is one of the least exciting I've seen for a long time. The option to



compose a story is a little better, allowing children to express their creative ideas in another original way. But the action is constrained to take place in the same system of rooms, connected in the same manner each time. Flexibility required is absent, so it is to be hoped that there will soon be similar programs which allow the child to create a personal environment.

A very exciting possibility is presented by a Viewdata-type program, and I have been using *Mikefax* by Colin Rowling Software, *Edfax* by Tecmedia and *Telebook* by 4Mation. They all use the same idea, that of using mode 7 graphics and text to create a series of pages which can be accessed directly from the disk, or which can be cycled in a preset sequence. There is, of course, nothing particularly startling in mode 7. But many teachers and other users tend to steer clear of the more complex features it can provide.

Each program comes with a sample range of screens which show the system to good advantage and are not too difficult. The system of storing pages on the disk is different, with all the *Edfax* pages together in a file called Pages, *Telebook* storing each 'book' as one file in a similar way, and *Mikefax* holding each page as a separate file named by its page number. This has an advantage in that pages can easily be copied, deleted or whatever — but the directory also has a habit of filling up rapidly!

There are strong points in each program. *Edfax* has a very easily used cycling capability, in which many pages can be set to appear in a predetermined sequence. *Mikefax* initially has a somewhat more awkward system, requiring linked pages to consist of more than one screen each. Editing a mistake, or otherwise editing a screen, is where *Mikefax* wins hands down, having several additional functions such as Delete Line or Insert Space.

My copy of *Mikefax* also includes a super utility called Convert, which I believe is usually available only on versions sold under licence to LEAs. It allows *Mikefax* and *Edfax* pages to be converted between the two, and allows pages from H & H Software's *Story* to be put into *Mikefax* form (and therefore *Edfax*).

Telebook has four modes which can be toggled through the TAB key, with the text mode including elementary but helpful wordprocessing features. These include automatic word-wrapping, allowing the child to concentrate on the text rather than the screen format. This underlines the rather different approach taken by 4Mation, that of providing a fresh medium and stimulus for the child to use in written work. These word processing features put the program into a somewhat different type, and it is well worth looking at for that reason. It also has the most comprehensive documentation, with Mike Matson's fresh view of education shining through as usual!

The educational potential of this type of program is vast. Many schools produce duplicated magazines, and this allows the pages to remain bright, topical, updated and colourful instead of slowly becoming dog-eared and ignored. A project can be presented to other classes in the school, or pages from new additions to the library can be written and illustrated. I have found that less able children, often frustrated by an inability to put thoughts into writing, have found a medium where polishing and honing can be made without a trace of earlier mistakes being left on the screen.

One thing is certain. Any one of these three would be a marvellous buy for most schools, being simple in concept yet vastly variable in the ways they could be used. *Edfax* claims to be uncopiable, and thus the main program needs to be loaded in each time from the master disk, with the pages probably being stored on another. *Mikefax* and *Telebook* explain how to make a backup copy of the parts of the program necessary for editing and composing. So a school with several micros could have different versions of the program running simultaneously. If I was pressed for a choice, I would spend my money on *Mikefax*, but would also suggest teachers try to see all three working.

WHERE TO GO

The following are the addresses of the software houses mentioned in the article.

Applied Systems Knowledge, London House, 68 Upper Richmond Rd, London SW15 2RP.
Clwyd Technics Ltd, The Coach House, Kelsterton Rd, Flint, Clwyd CH6 5TH.
Amsoft Ltd, Brentwood House, 169 Kings Rd, Brentwood, Essex CM14 4EF.
Acornsoft Ltd, Betjeman House, 104 Hills Rd, Cambridge CB2 1LQ.
4Mation Software, Linden Lea, Rock Pk, Barnstaple, Devon EX32 9AQ.
The Eaglewing Company, 23 Brendan Way, Westcliff-on-Sea, Essex.
Bourne Educational Software, distributed through John Wiley & Sons, Shripney Rd, Bognor Regis, West Sussex.
Database Publications Ltd, Europa House, 68 Chester Rd, Hazel Grove, Stockport SK7 5NY.
Tecmedia Ltd, 5 Granby St, Loughborough LE11 3DU.
Colin Rowling Software, 2 Tamar Rd, Haydock, St. Helens, Merseyside WA11 0HR.
Ian Copestake, 23 Connaught Crescent, Brookwood, Woking, Surrey GU24 0AN.

Rock on!

Even the cheap music packages for your micro can be used to make a musical — Phil South brought his to the Edinburgh Festival.

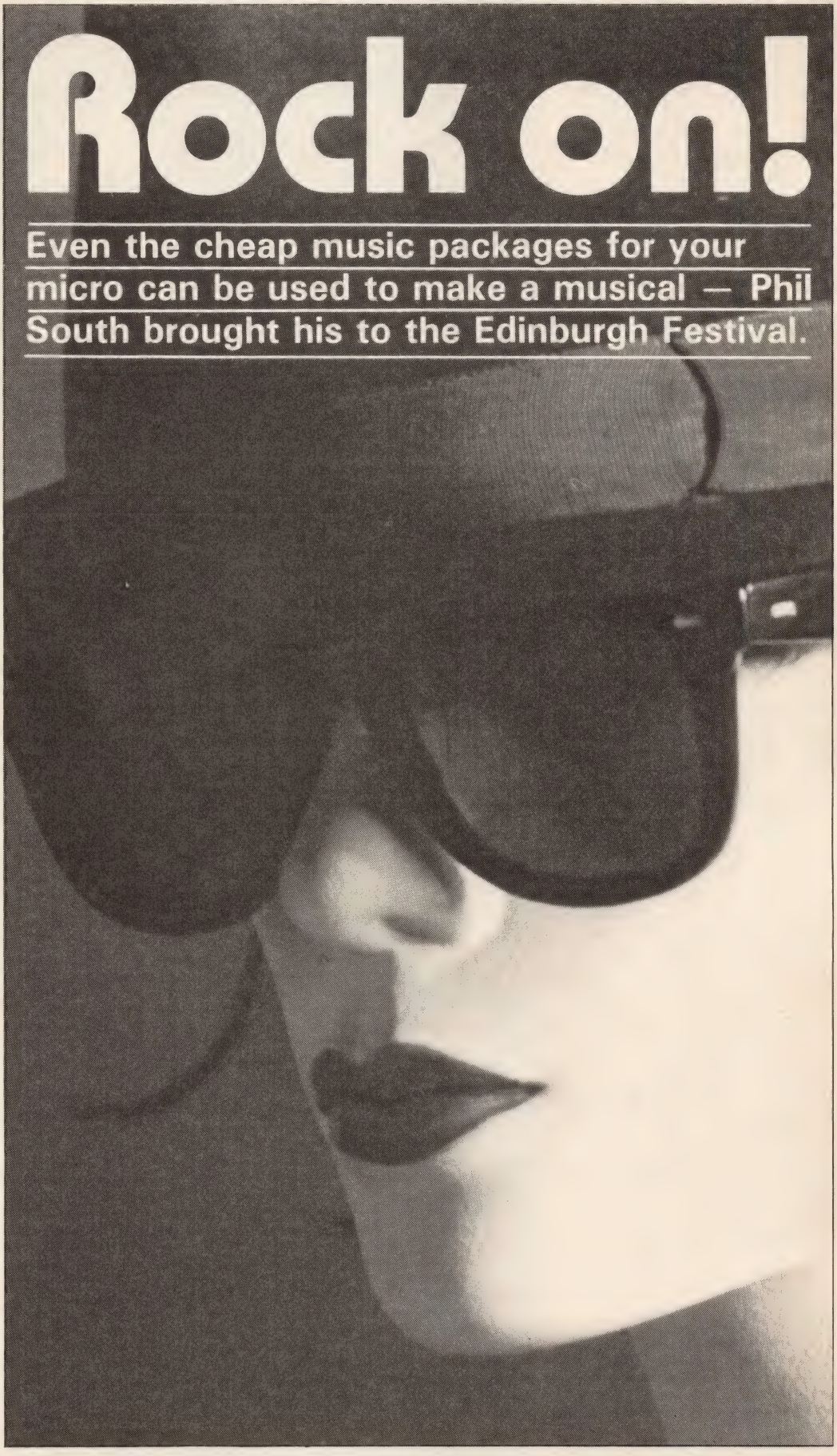
'The future is a great metal city. Its bristling tentacles reach around the globe, fouling all but the most arid, godforsaken deserts, and the densest, dankest forests. The people of Earth have become ragged and bizarre, fearing nothing but boredom and seeking what they know to be only temporary solace. Natural death has been cheated by science, but death it seems has the last laugh, as humans lose their humanity to machines . . .'

So begins *Maxine*, a good old-fashioned story of life, death, love and violence in the 23rd century. It is also a futuristic musical which uses the many commercial, inexpensive music packages for home micros.

I find myself in a weird situation. I wrote a play for the stage, an ambition I've held for a number of years. But when I started out on the project, it seemed to have a life of its own. The original boy-meets-girl story grew daily to include laserbeams, computer music and Maxine, a real android. (What? I hear you cry, you cannot be serious! But it is true.)

In the last two weeks of March, the show played to packed houses in a theatre near Hemel Hempstead. It was so well received that a booking was made for a week at the Richmond Demarco Theatre, Edinburgh, as part of the Edinburgh Fringe Festival in August. How was it all done?

Showroom dummy. In my living room there stands a showroom dummy. I called her Maxine, and originally bought her merely for decoration (*You're potty — Ed*). A shop dummy is an unsettling thing to have around the house. Every time I walked into the room holding a cup of tea she startled me into spilling it! But that got me thinking. I



Maxime: the problem was getting artificial control systems to move limbs twice their size



A scene from the show: 'you can use computers intelligently in music, and come up with a human product'

was looking over the first drafts of the show, and made an inevitable connection.

Robots, or to be specific androids (human-like robots), would have a similar effect. It looks human, but it's too motionless.

I set to work.

All the best set plans . . . Building a robot seems like an easy task. But, believe me, it requires a lot of know-how — the building of an android doubly so! We tried all kinds of designs before we gave up. Control was the problem. How could artificial control systems move limbs twice their own size and mass, while fitting them all into the body shell of an android five feet high? We had to admit that, although we were essentially artists with a slightly scientific background, none of us had the engineering skills to pull it off. We temporarily shelved the problem to concentrate on the rest of the show.

Music maestros, please. I am a musician. I can play most instruments if I have to, which I do to write songs. I thrash at a drum set, tickle the ivories or strum a guitar and record it. Although I know enough about musical notation to know what it all means, I don't know enough to write it down directly. This is where my computers stepped in to help.

Music composer programs are alright if you know musical notation. I don't so I passed them by — until I got a package to review one day, *Music Typewriter* by Romantic Robot. I fished my old 48K Spectrum out of the hall cupboard and tried it out. The results were amazing.

It plays a tune for you. You put all the notes on the staff (that is what you call those five lines you put the notes on), and it plays it back to you. If it sounds wrong, you change it a bit until it's better. The upshot of all this trial and error is that, not only do you learn about musical notation, but you also have a score (a piece of music on paper) which you can print on any dot-matrix printer.

The next package I found invaluable was *MusiCalc* by US company Waveform. You may have heard a lot about this program (suite of programs, to be exact). It is very much the definitive music program on the Commodore 64.

MusiCalc is, apart from any consideration of ability, the most attractively packaged piece of software I've ever seen. You've seen picture discs in your record shops? The *MusiCalc* disks have pictures on them, and very pretty ones they are, too! In view of the price (£45 rrp for the system master disk), it should be. But, as I said, it is the best music software for the Commodore. Until Hyrbid Technology's *Music 500* comes out for the 64, I'll stick to *MusiCalc*!

It is a synthesiser and sequencer program, neatly laid out, and very easy to use. The controls are constructed like those on a real synth, and the sounds you can get are the best you can expect from the humble SID chip. It comes with 32 sample preset sounds and 32 preset tunes.

The system disk is called *MusiCalc 1*. I wondered about this, until I came to the second disk in the system, *MusiCalc 2 — Scorewriter*. This disk carries three enhancements to the *MusiCalc* system on it. *Scorewriter* generates scores, or sheet music, from your *MusiCalc* files; *List Maker* links files to make complicated long pieces, impossible with just the system disk alone; and *E. Sequencer* sequences the files linked by *List Maker*.

The third disk in the system is *MusiCalc 3 — the Keyboard Maker*. This program allows you to set your keyboard to an enormous range of non-European scales, such as Balinese, Japanese or Indonesian. Eighty such scales are included on the disk, with hundreds more on a disk called *Scales of the World*.

As well as the three *MusiCalc* disks, there is a range of musical 'templates'. These are alternatives to the presets you get when you boot up the system master.

In the studio. Having generated some ideas for the songs, I took my computer-written sheet music and some croaky demos to my musical collaborator, Matthew Cowe. By the light of day he is a graphic artist, by night he is one of the best musicians I know. He took my musical sketches and mumblings, and turned them into arranged songs, which we then took into the studio. Or rather, the studio came to us, in the form of a Greengate DS:3 plus Apple IIe computer and John Molloy, one half of innovative rock band Mainframe.

All the songs in the show are influenced by the glam rock of the seventies: folk like Gary Glitter, David Bowie, Marc Bolan and T Rex, and my favourites Roxy Music and Brian Eno. We used the DS:3 to give our demos that authentic sound.

The DS:3 (Digital Sound Sampling Sequencer = DSSS = DS:3) is a board which you slot into your computer, and a full-size piano-style keyboard. It samples sounds: that means it records them digitally, and allows you to play or sequence them from the piano-style keyboard, or the Apple's qwerty keyboard.

We took our first samples from a David Bowie track, genuine 1970s drums for our title song *Maxine Machine*. For the Electric Prince's (a

kind of Gary Glitter character) song, we sampled the original Glitterband drum sound. It's not as simple as that, of course. But, as John said at the time, 'it's so easy to learn, even you could do it, Phil!'

Genius of dance. I left it to John to come up with some modern dance tracks for the show. With a few minor adjustments for timing, the tapes were used in the show in their original versions. They were composed and played using the DS:3 and a Juno 60 synthesiser.

Sue Mercer (ex-Leicester Polytechnic of Expressive Arts) then choreographed the dances. The combination of the two elements was exciting to watch, especially the scene in which Maxine meets Dax, the leading man, and they express their feelings for each other in dance.

Incidentally. The actor who played the electric prince, Ian Nichols, expressed his desire to create incidental music for the show, to cover such things as scene changes. He recorded them using his new MSX computer, the Yamaha CX-5M. This must be the nicest sounding computer around, as it has a Yamaha DX9 synth as its base.

Ian's music was a cross between the soundtrack of *Forbidden Planet* and the kind of music you hear on *The Twilight Zone* or *Outer Limits*. He sold his Korg Poly 800 to buy his computer; it was money well spent.

The robot cavalry. During one rehearsal for the show, Matt Cowe was tuning up his bass, when a very thin man with a straggly grey beard and glasses came up to him. 'Hello, I wonder if you could help me? I make robots . . .' he said. Matt nearly fell over. His visitor's name was Peter Holland, and besides being an artist he makes robots for such clients as Grundig and the people responsible for the Green Cross Code.

Peter made Maxine in two weeks, giving her a light styrofoam body and control lines made of the wire used to fish for sharks and marlins at sea! (Why the dickins didn't we think of that?) A mould was made of Emma Milward's (the dancer who plays Maxine at other times in the show) face, and the body was shaped to her build. A control wire ran up the microphone lead held by the robot, down to the control box in the hands of the operator backstage.

We had our android. And the show was due to go on in two weeks.

Just a simple love story. Most musicals are pretty light in the plot department, and Maxine is no different. It is a simple boy-meets-girl, boy-meets-robot-and-forgets-girl, boy-loses-robot-and-remembers-girl story. The script was written on an Apple II with *Applewriter* wordprocessor, and took me about five months to complete. I say 'complete', but as I'm writing this I have a pile of paper next to my computer consisting of ideas for re-writes.

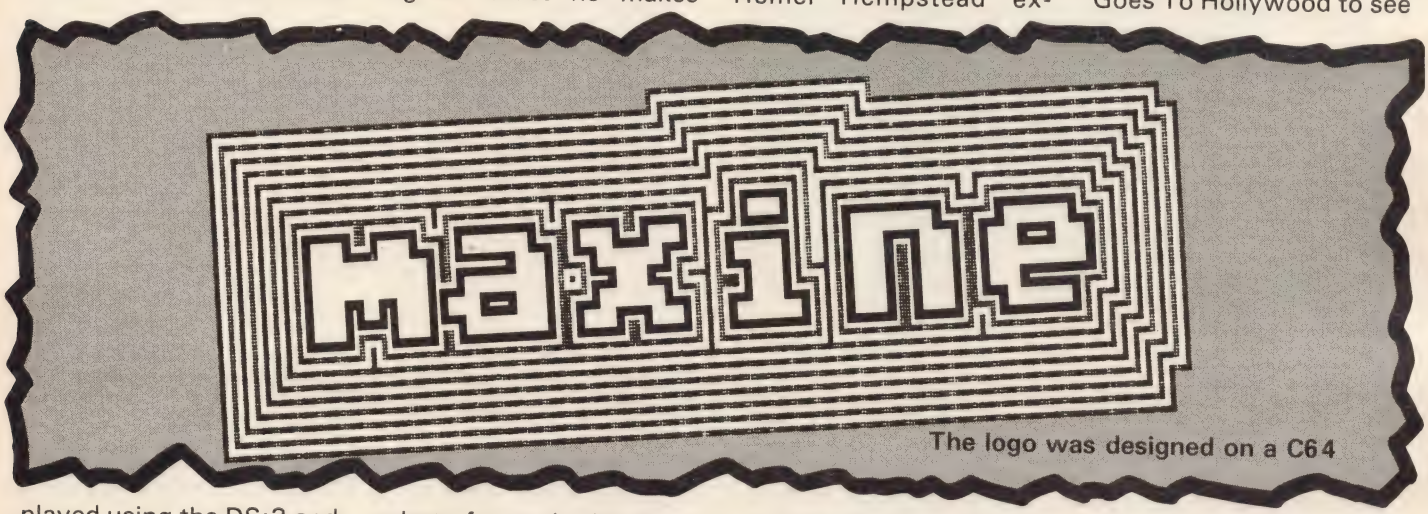
It'll be alright on the night. It was. The attendance of the two-week run of Maxine at the Boxmoor Hall Arts Centre in Hemel Hempstead exceeded attendances at our previous productions at this venue. The Hill O'Beans is the theatre company to which we all belong. I was a founder member in 1983, when the company took a production of the old movie classic *Casablanca* to the Edinburgh Fringe of that year. It is run to promote theatre in general, and specifically to give young, or in my case medium-young, people a chance to get into theatre.

You must know the adage 'never work with children or animals'. I would definitely extend that to include robots. They malfunction, break down, generally do what you don't want them to do — and yet they manage to upstage you!

The pay off. Computers played a strong part in the assembly of the *Maxine* project, I even designed the logo on the Commodore 64. But they didn't create it; they merely extended our capabilities, and allowed us more time to consider the creative decisions. All the songs (except the dance tracks) are played by live musicians, but the demos were greatly helped by the involvement of our computers.

That's how I see the future of computers in the arts, as a tool to extend what can be done with our creative urges. You've only got to look at Frankie Goes To Hollywood to see that; it was nearly just another R'n'B band until it discovered the Fairlight CMI synthesiser. You can use computers intelligently in music, and come up with a human product.

On the menu for my next article, we will be chewing over some interfacing facts, a peek at one or two ways of linking your computers to the outside world of synthesisers and drum machines. See you then.



The logo was designed on a C64

Maxine — A Futuristic Musical can be seen in the last week of the Edinburgh Fringe Festival at the Richard Demarco Theatre, Edinburgh Arts Centre, St. Andrews Hall, St. James' Place, Edinburgh. Tickets available from the Boxmoor Arts Centre (0442) 64048, from the Richard Demarco Theatre, or from Fringe Society on 031-226-5257.

The Atari DOSsier

Beginners are not limited to the DOS which comes with their Atari disk drive. Mark Hutchinson explains

Most people who buy an Atari disk drive seldom, if ever, think of DOS (disk operating system) except to initialise a new disk or use it for a directory listing. Because of this, certain limitations are never noticed. These limitations do not detract from the power of DOS; it is just that a standard had to be set and DOS 2.0 is it. I hope you will find this article helpful in adding to your DOS.

In 1978, a firm called Shepardson Microsystems Inc (SMI) was approached by Atari to set up a possible Basic. Not only did it build a Basic language, but it went on to set up *File Manager* to change the operating system (OS) into DOS.

Since that time, SMI employees have gone their own ways. The best known of them, Bill Wilkinson of Optimized Systems Software, has brought us Basic A + OS/A+, MAC 65 and Action. The beauty of Atari DOS is that it is Ram resident, so can be altered by the simple expediency of POKE statements. You can obtain a listing for DOS in *Inside Atari DOS* by Bill Wilkinson (Compute books), so you will not need a disassembler.

In case you do not already know, your disk is divided into 40 concentric rings or 'tracks', each one sub-divided into 18 divisions called sectors. Each sector can store 128 bytes of information. Your program uses only 125, the other three noting the file number, where the next sector in the file is located, the end of file (EOF), and how many bytes were used in the sector.

There are 720 sectors on the disk, but you cannot access them all. Because of a bug in the system which counts sectors from 0 to 719, and the fact that the sectors are numbered from 1 to 720 so the disk drive will not accept that there is a sector 0, you cannot use sector 720. There are ways round this.

You can fool the computer into using sector 720 for, say, hiding your name on the disk in case of ownership disputes. Nine of the remaining sectors are used by the *File Management System* (FMS) to take care of the housework, and the last three sectors for the boot load.

Because the disk directory is used so often by DOS, it is positioned at the centre of the disk to give a short average search time. The sectors involved are 361 to 368 and each sector contains eight file names of 16 bytes each. The first five bytes give current information about the file.

The first of these is a flag to show things such as whether the file is open or locked. The next two bytes give the number of sectors in the file, and the last two give the starting address sector number. The remaining 11 bytes give the file name and extension.

To complement the directory is sector 360, known as the volume table of contents (VTOC) and sometimes as the sector bit map. This map has 90 bytes, each of whose eight bits represent a sector on the disk (0 if used, 1 if unused). Other bytes tell FMS if it is DOS 1.0 or DOS 2.0, total number of sectors and free sectors. These are some free bytes here for you to use for identification purposes.

The diagrams make this a lot clearer. DOS is divided into two parts. The first is DOS.SYS, which is loaded during the initial boot process, and allows commands such as RUN, SAVE, DOS. The second part, DUP.SYS or disk utility package, is loaded into the computer with the DOS command. This allows directory listing, lock/unlock, copying, and the like.

The good point of this design is that the DOS package is Ram resident and is not wiped clean when you load DUP.SYS. This means that, if you use option H to write DOS files, the amendments are also written.

Now that we have an understanding of the set-up of a disk let us find out how to change it to suit ourselves.

Probably the most famous DOS amendment is by Wilkinson. If you listen to your drive operate you may detect different 'tones' as data is transferred. One of the differences occurs when the computer detects a mass of data to transfer. In this case, it goes into what is referred to as BURST I/O, where the drive operates much faster reads and writes.

Unfortunately, a bug in this operation causes writes to be right and reads to be wrong! So up Wilkinson came with a list of POKEs, as follows:

```
10 FOR P=1 TO 8 : READ
   D : POKE 2591+P, D :
   NEXT D
20 DATA 130,19,73,12,
   240,36,106,234
30 POKE 2625, 16 :
   POKE 2773, 31
```

As you may know, DOS 2.0 accepts only upper-case letters for the filename, no imbedded spaces, no punctuation, no control characters. There is a specific routine which checks legality of the characters. Location 3818 holds the lower ATASCII limit (normally 65) and 3822 holds the

upper limit (91). If we change these to, say, 33 and 123 respectively, we can now include upper and lower case, numbers and punctuation. You could, of course, include almost all the character set, but this may prove to be a problem in certain cases.

The wild cards can also be changed, if so desired, by POKE 3783 with whatever ASCII value you want. But this causes trouble when you try to run a program that uses '*' as a wild card.

The Atari is usually set to have three files open at any one time (IOCB 0 is used for the screen and is not usable by the programmer). Each file has a buffer of 128 bytes. So if you do not use all the files, or if you use more, you can POKE 1801 with the number of files you need.

If, like many of us, you have only one drive you can save 128 bytes by changing the drive default value of two drives in location 1802 to one. For this, you must set the bits. For one drive set bit 0, for two drives set bits 0 and 1, etc, and POKE in the decimal equivalent:

1 Drive, Bit 0, Dec. 1
2 Drives, Bits 0,1, Dec. 3
3 Drives, Bits 0,1,2, Dec. 7
4 Drives, Bits 0,1,2,3, Dec. 15

If the drives are not numbered sequentially you must set the appropriate bits.

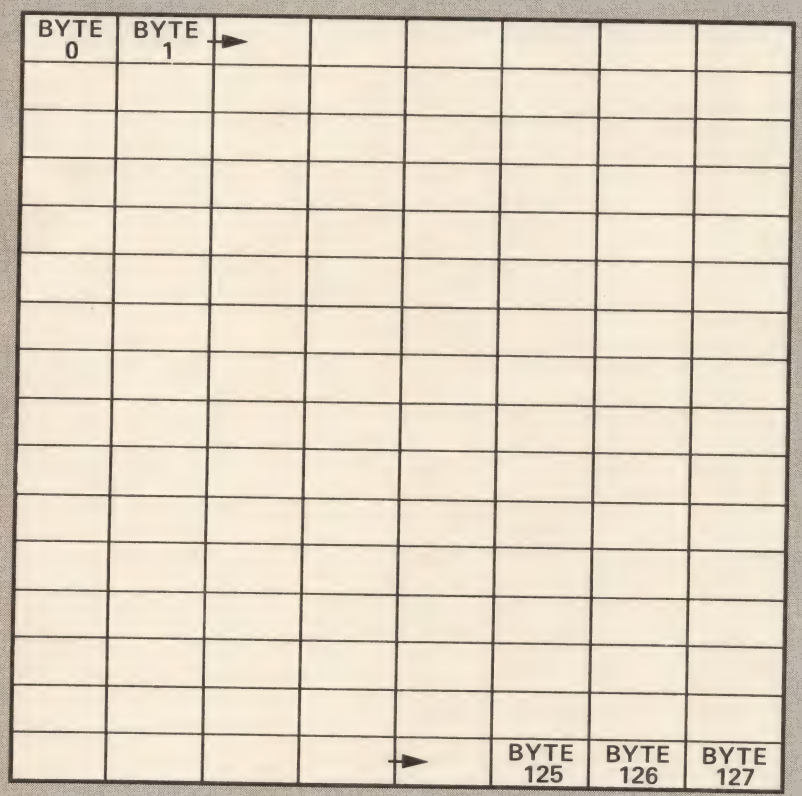
When you wish to increase the I/O rate, but are not worried about DATA errors, you can change the default value of location 1913 from 87 to 80. If you are using K-DOS this is location 1907. The upgrade to the analog or 'Grass Valley' configuration means that the error rate has been drastically reduced, so you can have confidence in this change.

The analog upgrade enhances the power supply, read/write amplifiers and greatly cuts down motor speed fluctuations. This is a point you will notice if you run an RPM check using the newer 1050 and the older 810 (endearingly termed 'the old Rom clunker'). Open the door of the drive and look in and up. If you see a circuit board, you have the newer model. Or, if you dare take it apart, the speed control on the back board is a large white wheel in the old model, and a small green potentiometer in the enhanced model.

The disk menu screen display is stored in locations 7951 to 8278. So if you want a customised screen alter these bytes.

Unfortunately, like Apple, IBM and Cromenco models, the Atari 810 has its own 6507 processor which does not allow custom formatting. There was a rumour of an E Rom revision to do this, but so far I have seen no confirming of this. If you feel like experimenting with the format routine, it starts at 9856. You can buy a program from a US firm to custom format, but it uses its own replacement Rom and is very expensive.

One customising routine used by many commercial software houses is to allocate the directory to another sector. Location 4226 is the pointer to the directory and, because it can store only up to 256, we are limited as to what sectors we can use. The default



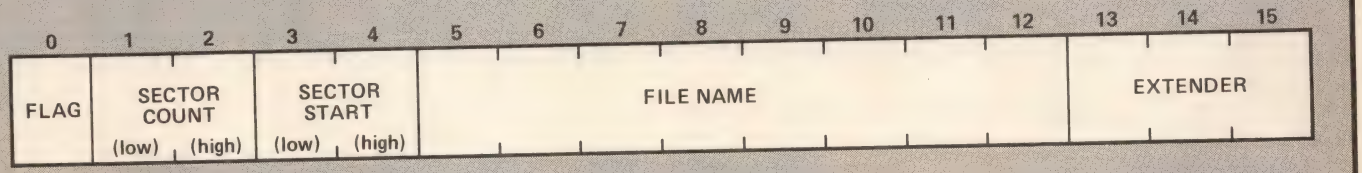
BYTE 125. Number of bytes used in the sector

BYTE 126. File number and high two bits of next sector

BYTE 127. Low eight bits of next sector number

All others are data bytes

Figure 1: Typical sector layout



FLAG
0 = Entry never used
1 = Open for output
2 = Created by DOS 2.0
32 = Locked
64 = In use
128 = Deleted

SECTOR COUNT
Number of sectors in file (low, high)

SECTOR START
Address of file start sector (low, high)

Figure 2: Directory entry for a file

value is 105, pointing to sector 361. The formula for working out what to POKE into this location is: new sector number + 105 — 361.

You must do this for a new disk, otherwise you need to zap the old directory to clear it. To carry this a stage further, you could, with the help of a sector copier, move an empty VTOC sector to your disk and delete DUP.SYS (which loads the DOS menu). Then use an AUTORUN.SYS file to boot your disk, keep the directory a secret, and show 707 free sectors.

This is a good time to mention that dreadful noise you hear when loading certain programs. During the boot, the program looks for a certain sector(s) with bad or unreadable bytes. This is called 'bad sector' copy protection and is quite sore on the drive motors. It is outmoded by today's standard of disk protection, anyway. If you use a drive-head cleaner you will clear the machine 'snark' continuously.

When DOS is called, the value of locations 5446 and 5450 enter DUP.SYS. These also update locations 10 and 11. You can use 10 and 11 to vector a DOS call to your own routine. But to make it permanent, you should POKE 5446 and 5450 with your vector value and use option H. You can use a USR call to the value of 10 and 11, or 5446 and 5450, instead of using the command DOS, if you have a program already loaded; you need a MEM.SAV on the disk so as not to wipe out the program. If there is no program loaded and you have previously called DOS, you could also use a USR call to locations 24 and 25 to get an instantaneous DOS menu.

You have probably come up against the tedious problem of having two similarly named files. If you have had this problem and have not owned a sector modifier, you may know that DOS loads the first file of the name that it comes to. This is a nuisance if you want to access the other file. What you normally do is to use option E to rename both files and interrupt the process during the slight pause when the drive is looking at the directory for the second file. This is where the sector modifier pays for itself.

The solution White found was to change the source code by POKE 3117,234 : POKE 3118, 234 : X = USR(8309) and Return, again using option H. This is commonly called 'patching'.

Whilst on the subject, if there are any programmers out there who would like to patch something to DOS rather than change it, you might be interested to know that the file management system Ram ends at location 5377, and the DUP.SYS Ram starts at location 5440. It's only 60-odd bytes, but enough for a simple routine.

Finally, back to something easier from Gordon L Banks of Analog. If you are fed up with having an AUTORUN.SYS file (ARS) and not remembering what it is for, then try this. DOS checks if there is a file with the same name that is stored in locations 5903 to 5914; if so, it boots that file. All you need do is to amend these locations to the ASCII value of the file you wish it to load. This means that this DOS RUNs only a specifically named file, so you are back where you started.

But if you left location 5309 alone, and amended the other locations to read ? or *, any file starting with 'A' is loaded. The variations are immense. Will you load anything starting with A, AU, or anything ending in ARS, or a file 'A*.SYS, or any combinations of the above?

Hopefully, this article shows that the beginner is not limited to the DOS that has come with the drive.

VTOC BYTES

Byte No.	
0	— DOS type (0 = DOS 2.0)
1-2	— Total number of sectors
3-4	— Number of unused sectors
5	— Reserved
6-9	— Unused
10-99	— Bit map; each bit represents a sector (0 = used, 1 = unused)
100-127	— Unused

Fig. 3: volume table of contents sector.

MSXemplary graphics

Brian Boyde-Shaw continues his series on MSX, probing the mechanics behind simple graphic statements.

This month, we investigate the graphics statements available with MSX Basic, and their easy use. Later in the series we shall look at more complicated graphics statements, and routines to make full use of them.

Usually, depending on the type of statement and the resolution, graphics are created by filling in, or switching on, screen pixels one after the other until the shape you want is completed. In screens 0 and 1 no graphics are possible, except those created with the built-in character graphics from the keyboard. Screen 1 has, of course, the sprite facility. This is useful for livening up text, and will be dealt with in next month's article about text in general.

Graphics are available with screens 2 and 3. In screen 2 you have 256 × 192 pixels to plot with, but with screen 3 only 64 × 48. So lines and points in screen 3 are chunkier than in screen 2, the main reason it is called low resolution. A plotted graphics point in screen 2 is 16 times smaller than a plotted graphics point in screen 3.

Plotting points. To plot, or switch on, a single point or pixel we use the statement PSET or PRESET. These stand for Pixel SET or Pixel RESET, as in this example:

```
10 screen2:color4,3,3
20 pset(100 ,100)
30 pset(150,100)1
99 goto99
```

The co-ordinates in the brackets put two coloured dots in the middle of the screen. The colour of the dot is determined by the number after the comma in the PSET statement. In line 20 there isn't a number after the comma, so the dot is in the colour in the last COLOR statement (or PSET or PRESET statement). Change the colour of the foreground in line 10, and the colour in line 30, to prove it. If you want to see a bigger single dot, use SCREEN3 in line 10.

PSET and PRESET need not be used for drawing lines, as LINE is much quicker. But they are useful for building multicoloured areas and single spots, and erasing in a fine-brush fashion.

There is not much difference between the PSET and PRESET statements, except that PSET sets a pixel to the foreground, or specified, colour, and PRESET to the background, or specified, colour. For example, change the above short program to:

```
10 screen3:color4,3,3
20 preset(100,100)
30 pset (150,100),1
99 goto99
```

Notice that the blue dot disappears when the program is RUN, because the pixel has been set to the background colour code 3.

Change line 20 to: 20 preset(100,100),6 and notice the difference.

When the colour of the pixel is specified in the statement, both PSET and PRESET act in the same way. A good demonstration of the effects of PSET and PRESET is to move a large pixel across the screen in low resolution:

```
10 screen3:color4,3,3
20 forx = 0to255step4
30 pset(x,100),1
40 ford = 0to250:next
50 preset(x,100)
60 next
99 goto99
```

Of course, using sprites is a lot better! Each statement in Basic has its own particular use, but it is interesting to experiment with them to see what they could do.

One snag with plotting points, or pixels, is that you have 'colour or pixel creep'. Foreground colour information sets eight pixels on one row on screen at a time. So if less than eight pixels are plotted at one time, other pixels on that row may also be changed at the same time.

Notice I said 'may'. If this could be a nuisance, care must be taken to ensure that graphics screens are drawn so that colour changes finish at an eight-pixel-wide boundary, starting from the left of the screen. To demonstrate this, use this example, which also shows the difference between PSET and other graphics statements such as LINE and PAINT:

```
10 screen2:color4,3,3
20 fory = 56 to 86
30 forx = 48 to 78
40 pset(x,y),1
50 next:next
60 z = 56:forw = 48 to 78
70 pset(w,z),15
80 z = z + 1:next
99 goto99
```

You will see the block of black colour drawn in row blocks of eight pixels, then the creep for the diagonal when it reaches the bottom right-hand corner. Experiment with the coordinates on lines 20 and 30, and the z on line 60 to correspond with the y, to see how the creep works.

You can also change to low resolution by changing line 10 to SCREEN3, and seeing the effect. For example, change line 20 to: 20 fory = 53 to 86. This takes effect only when plotting over previously set pixels, and does not affect the background colour. Change line 60 to: 60 z = 56:forw = 48 to 178 and notice that the diagonal runs down without any creep.

This effect is also given in the following short program:

```
10 screen2:color15,3,3
20 fory = 25 to 86
30 forx = 74 to 96
40 pset(x,y),1
50 next:next
60 forz = 74 to 96
70 forw = 25 to 86
80 pset(z,w)
90 ford = 1to10:next
100 preset(z,w),1
110 next:next
120 goto120
```

The trick is to position your main colour block, or previously plotted area of colour, where changes have only the desired effect. To show this, experiment with the coordinates



on lines 20 and 30, and 60 and 70, remembering to keep the xs and ys equal on both lines. Change also to the low-resolution screen and notice the difference. Can you work out how to get rid of the repeats in the same column?

More examples of this difficulty crop up in the CIRCLE statement later in this article.

The PSET statement is also very useful for drawing graphs. The following short program draws a curve based on the square root of a number:

```
10 screen2:color1,4,4
20 n=1:X=30
30 pset(x,180-sqr
   (n)*10),1
40 x=x+1:n=n+1
50 if n<>192 then30
99 goto99
```

Before continuing have a look at SCREEN3.

The following program draws sine waves, where n affects the number of waves drawn:

```
10 screen2:color,15,15
20 draw"bm38,10c1d
   120u61r190"
30 n=12:forx=40 to
   225
40 pset(x,70+50*sin
   (x/n))
50 next
99 goto99
```

The number 50 in line 40 affects the amplitude, or height, of the wave drawn, and the number 70 its starting position.

Try changing line 20 to: 20 draw"bm40, 10c1d120u61r190". See the affect of the creep on the X axis of the graph. Position is all important! Don't forget to have a look at SCREEN3.

Associated with PSET and PRESET is the POINT function, which reads the colour of a pixel anywhere on the viewing screen. It is usual to define a variable to do this:

```
c=point(x,y)
if c=background colour
then continue
```

C is the colour code read, and x and y the coordinates on or off the screen.

If the pixel has not been set or filled in at the point read, the code returned will be the background colour. If the point is set, the code of the colour set will be returned. If x and/or y are off screen, −1 will be returned. Naturally, the function should be used only with screens 2 and 3. If used with screens 0 and 1 you can expect strange codes to be returned.

Drawing lines and circles. To draw a straight line on the screen, we can plot all the points necessary:

```
10 screen3:color1,15,15
20 forx=50 to 149
30 pset(x,25),6
40 next
99 goto99
```

This draws a red line, 100 pixels long, on the screen. But we had to resort to a FOR. . . . NEXT loop to do it.

The LINE statement does the job for us without a FOR. . . . NEXT loop, and does it a lot

quicker. This is because it calls up a built-in machine code routine, and each pixel does not have to be plotted individually from Basic. Add line 50 and notice the difference in speed: 50 line (50,50)–(149,50).

Notice also that the colour of the second line is as in the colour statement in line 10. The colour called up by the PSET statement has no affect when LINE is used.

The LINE statement draws a line from the first set of x and y coordinates to the second. Notice the brackets and the dash. To change the colour of the line we must add a comma and a colour code to the LINE statement: 50 line ((50,50)–(149,50),4

You can draw lines anywhere you like on the screen by using separate LINE statements. But you use individual coordinate pairs to draw shapes, the start of each line being the present position of the graphics cursor. This can be positioned using either PSET as in line 30 or LINE as in line 50. For example, change line 50 to 50 line–(149,50),4 and we have the beginnings of a shape.

To achieve the same shape, lines 20 to 40 can be changed to one line 20, and lines 30 and 40 deleted: 20 line (50,25)–(149,25),6.

To complete the shape add line 60: 60 line–(50,50),8:line–(50,25),10. This produces a rectangle with four differently coloured sides.

The same rectangle can be produced with just one LINE statement, but the sides must all be the same colour. Add: 70 line (50,125)–(149,150), 12. This draws a diagonal of the same size as the first rectangle. Run it and see.

If you don't like low resolution, change to SCREEN 2 in line 10. Notice that each diagonal plot is eight pixels long! It all depends on the angle of the diagonal drawn.

Now change line 70 by adding the letter 'b': 70 line(50,125)–(149,150),12, b. This draws a rectangle based on the diagonal in line 70.

The rectangle can now be fitted in by adding f to the b: 70 line (50,125)–(149,150), 12,bf. If you leave out the colour code, the comma must remain: 70 line (50,125)–(149,150),,bf. But you get your COLOR statement code colour, not the last LINE statement colour! It is one of the fastest ways to fill an area with colour.

To move this block of colour is quite easy. Add lines:

```
65 y=125:forx=50 to
   255
75 line(x,y)–(x,y+25),
   15:next
```

These move it off the right edge of the screen, also showing that these graphics statements can control 'objects' out of range of the normal screen coordinates. All the extra program lines do is draw a white line at the left side of the block each time it draws another block one-pixel strip to the right.

Another method is to draw a while line at the left edge of the previously drawn block, and an extra coloured line on the right, using these extra program lines instead:

```
75 y=125:forx=50 to
   255
80 line(x,y)–(x,y+25),
   15
85 line(x+100,y)–
   (x+100,y+25),
   4:next
```

It is no quicker or smoother, but shows how simple MSX graphics are from Basic. Colour code 0, for transparent, can be used instead of the background colour to make lines disappear.

This is only a simple shape. More complicated routines will be explained in an article on graphics later in the series.

Interesting effects can be achieved by adding the word STEP to the line statement. For example, change line 80 to: 80 linestep(x+100,y)–(x+100,y+25),6: next.

A simple right-angled triangle can be drawn:

```
10 screen2:color1,15,15
20 line(40,10)–(40,60)
30 line–(100,60)
40 line–(40,10)
99 goto99
```

To fill it, we need to use the PAINT statement. If this is not used correctly, it can cause a lot of trouble and frustration.

We can draw the same triangle by using the STEP keyword. When used with a pixel-plotting graphics statement, it tells the computer to plot a pixel, or draw a line a given number of pixels in the x direction with a given number in the y direction. The x and y 'coordinates' are not a particular position on the graphics screen, but a given number of pixels relative to the last graphics point used.

Look at this example:

```
10 screen2:color1,
   15,15
20 line(100,60)–
   (40,60)
30 line-(40,10)
40 line-step(60,50)
99 goto99
```

Line 40 tells the computer to draw a line to a point 60 pixels downwards and 50 pixels to the right from the last point plotted. You can now draw a filled-in box in green at this new point by adding line 50: 50 line-step(35,45),12, bf. The box will be 35 pixels wide by 45 pixels high.

We can also make the false coordinates negative, and move the box round our triangle, as in: 50 line-step (–35,–45)12,bf.

The PAINT statement needs only a pair of coordinates to tell the computer where to start filling in the screen with solid colour. The colour can be specified. If it isn't it fills the shape with the present graphics colour. The program

```
10 screen2:color12,15,
   15
20 draw"bm81,80c4r
   60d40160u40"
30 paint(83,82)
99 goto99
```

does not fill the rectangle the DRAW statement has drawn. This is because the colour drawn is code 4, the colour specified in the COLOR statement is code 12, and no colour has been assigned in the PAINT statement. Instead, it fills the screen 15 second later.

Change line 30 to 30 paint(83,82),4 and all will be well. Notice the difference in time it takes to fill the rectangle between the LINE . . . BF statement and the PAINT statement.

Failure to position the PAINT coordinates correctly also leads to trouble. Change line 30 to 30 paint(70,80),4 and you end up with a blue screen and a white rectangle!

I always add at least two pixels to the start coordinates of my shape to avoid any difficulty.

Any problem making the border of a shape the same colour as the filling is overcome with SCREEN3. For example, change lines 10 and 30 to:

```
10 screen3:color 12,15,
   15
30 paint(93,92),4
```

Notice that the coordinates move further into the shape, and the shape's border lines are thicker.

Now change it to 30 paint(93,92),6,4 which fills the blue-border rectangle with red, code 6. If you want to do this, you must add the colour code of the border and an extra comma to the statement.

Finally, let's move to the CIRCLE statement. This is similar to the LINE statement, but cannot automatically fill a shape



with colour. For example,

```
10  screen2:color12,15,
    15
20 circle (128,96),10
99 goto99
```

draws a small circle in the centre of the screen, where the two numbers in brackets are the normal screen coordinates, and the first number after the comma is its radius in MSX units.

The circle is not a true circle but some sort of an ellipse, depending on your TV. To get a true circle, you must use the aspect ratio of the statement. On my equipment, an aspect of 1.4 to 1 seems to work well; you must experiment. Changing line 20 to

```
20  circle(128,96),10,,,,
    1.4/1
```

should do the trick — don't miss out the four commas.

This aspect ratio changes the relationship of the vertical axis of the circle to the horizontal axis. Here it makes the vertical 1.4 times bigger than the horizontal, and so produces a true circle.

If you want coloured circles to make a pattern, try this:

```
15 for r=0to96
20  circle(128,96),r,5,,,
    1.4/1
25 next
```

It shows you that there are always some pixels not filled in when you draw a circle, and that the fourth number is the one to assign a colour for your circle.

Try changing line 20 to get a spectacular effect: 20 circle(128,96),r, r/7,,,1.4/1. You get all the MSX colours, and you can experiment with the divisor for the colour code to change this effect, too.

If you want separate circles, try this:

```
15 c=1:for r=0 to 96
    step4
20 circle(128,96),r,c,,,
    1.4/1
25 c=c1:ifc 15 then
    c=1
30 next
```

The spaces between the remaining three commas are for drawing incomplete circles given in radians, so pi on its own is only half a circle. It must be written as 22/7, or its decimal equivalent of 3.142.

We put the start point for our incomplete circle between the first two of the set of three commas, and the end point between the next pair, working anti-clockwise from its rightmost point. This point is written as 0 (zero).

Try changing line 20 to: 20 circle(128,96),r,c, 0,22/7,1.4/1 and you get a half circle. Change to 20 circle(128,96),r,5,0, 11/7,1.4/1 and you get a quarter circle only.

Change the start point to: 20 circle(128,96),r,5, 0.7,2.1,1.4/1 and you get a fan.

Delete lines 15, 25 and 30, and change line 20 — all you get is an arc: 20 circle(128,96),50,4, 0.7,2.1,1.4/1

To finish off this month's article, try the following program. It uses nearly all we have looked at this month and makes use of the creep effect.

```
10   screen2:color1,
     12,12
20   x=40:z=x:for
     loop= 1to8
30   line(x-9,95)-
     (x+15,96),14
40   circle(x,96),40,0,
     22/7,5/1
50   paint(x,94)14:x=x
     +17:next
60   circle(z-10, 10
     6),16, 4
70   paint(z-8,104),4
80   circle(x+25,
     104),12, 13
90   paint(x+27,
     102),13
100  circle(x+25,
     98),4,1
110  paint(x+27,96),1
120  line(30,96)-(200,
     116),4,bf
130  goto130
```

About turn!

Converting graphics from the Spectrum can be complex — but many things are very easy, Steve Lucas promises.

In the July issue of *PCT*, I took a close look at how to convert a Basic program written for the Sinclair Spectrum so that it would run on an Amstrad computer, and showed that there were several functions in Spectrum Basic not available to the Amstrad user.Conversion of graphics is more complex than general programs, but many things are very easy to convert.

If you try to convert a listing without having seen the effect on the host computer, life will be much harder than if you have a good idea of what the picture looks like on the Spectrum's screen. My general plan of action when converting to the Amstrad is to examine the Spectrum listing closely, and make the conversions in the following order.

* Find and convert any user-defined graphics (UDGs).
* Convert the screen coordinates of any high-resolution graphics commands.
* Convert graphics dealing with straight lines.
* Convert curves, etc.

Approaching the task in this order has the advantage of leaving the most difficult part until last. Spectrum Basic has extensive graphics commands to handle arcs and circles. These need extensive rewriting to work on the Amstrad. But user-defined graphics are comparatively easy.

User-defined graphics. Characters on both machines are defined on an 8 x 8 grid, but the details of programming differ greatly. Before converting a listing, you need to understand the way in which graphics characters are defined on the Spectrum.

Supposing you wanted to define the character in diagram 1. You first need to store the data for this character as a series of binary numbers in DATA lines. Each pixel in the character can then be represented by either a zero or a one.

The following listing shows how this information would be read and POKEd into memory to define the character.

```
10 REM * * Spectrum Program * *
20 DATA BIN 00000000
30 DATA BIN 00000000
40 DATA BIN 00000010
50 DATA BIN 00111100
60 DATA BIN 01010100
70 DATA BIN 00010100
80 DATA BIN 00010100
90 DATA BIN 00000000
100 FOR n=0 TO 7
110 READ a
120 POKE USR"P"+n,a
130 NEXT n
```

The eight lines of BINary data contain the equivalent of the character lines in diagram one. These eight binary numbers are read in line 110 and POKEd into the address in line 120.

The first byte, or group of eight bits, is POKEd into the memory location specified by USR"P". The second byte goes into USR"P"+1, and so on. This use of the USR command is very different from the way in which it is used in Locomotive Basic, but conversion to the SYMBOL command is easy. To convert the character, you don't need to understand how the Spectrum listing works, as long as you can recognise that part of the listing dealing with character definitions. But to satisfy your curiosity, the USR command converts the string argument into the address of the first byte in memory for the corresponding UDG.

Amstrad users will, no doubt, be familiar with converting a diagram like ours into a UDG using the SYMBOL command. The first task is to decide which character number you want to define the character as. There is no need to convert it to the same number used in the Spectrum listing, as long as you remember which number you have used.

Using characters above about 240, rather than redefining the standard character set, has the distinct advantage of using less memory. If you wish to define it as

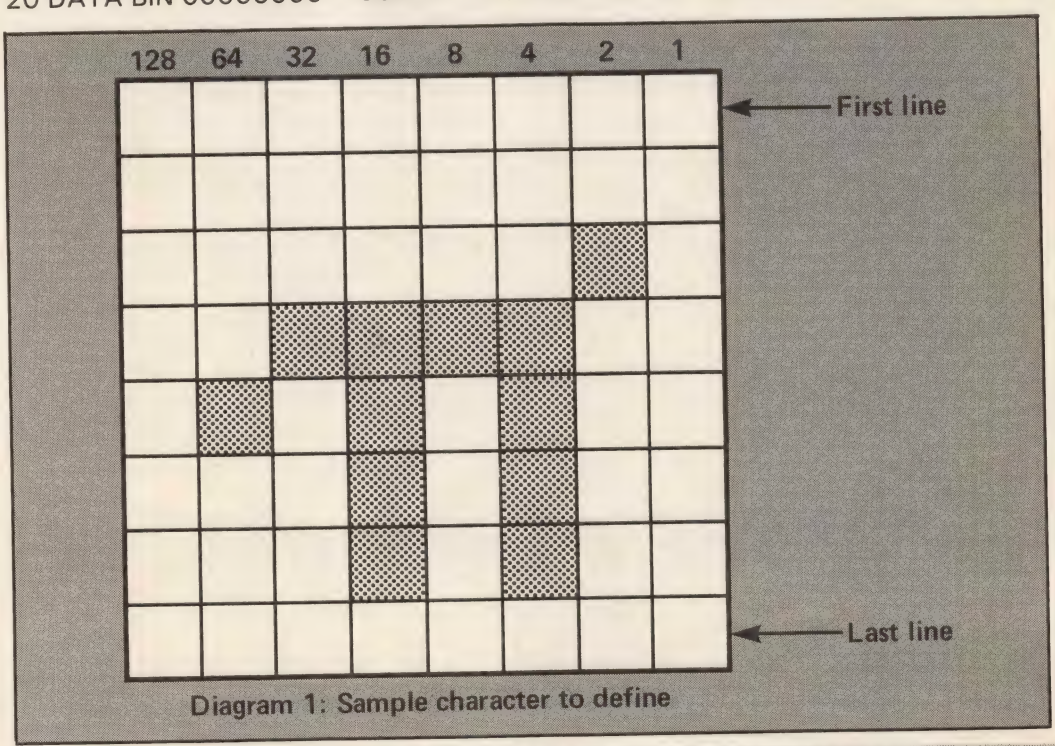
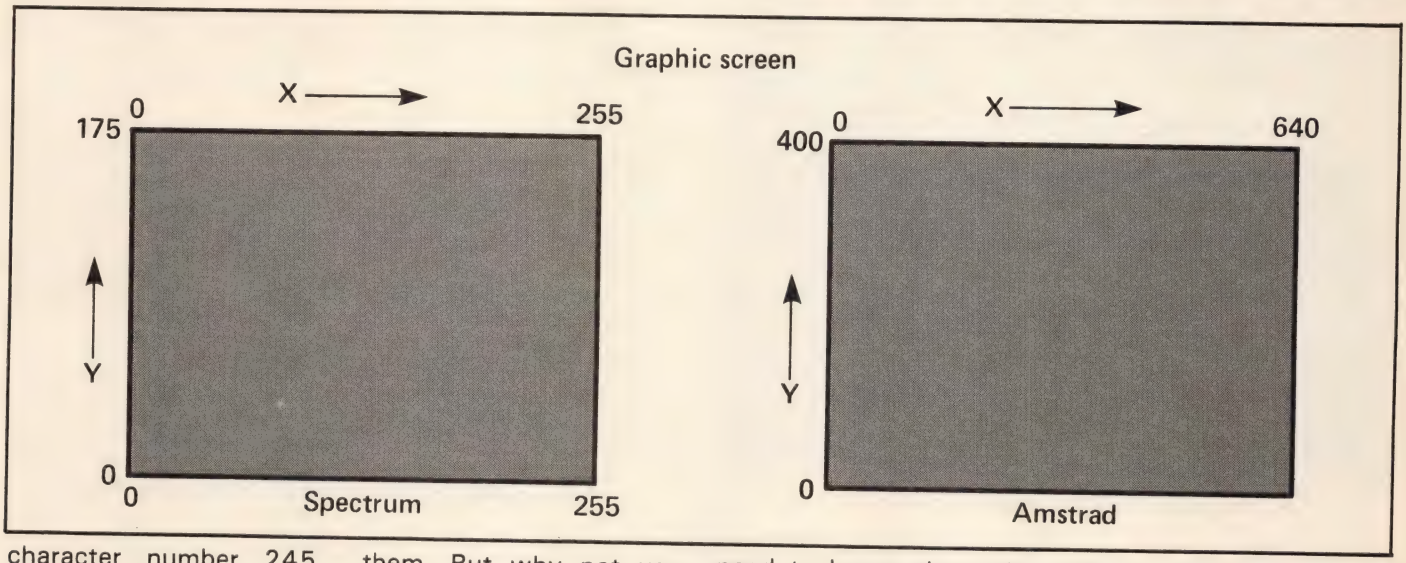


Diagram 1: Sample character to define



character number 245, say, you need to use a command such as SYMBOL AFTER 244 before making the conversion.

Amstrad version:

```
10 REM * * Amstrad version
20 SYMBOL AFTER 244
30 SYMBOL 245,0,0,2,60,84,20,20,0
```

In case you are unsure from where the numbers following SYMBOL 245 come, they are the denary (base 10) equivalent of the binary numbers in the Spectrum listing.

So how do you set about converting the BIN statements into these eight numbers? The first thing to do is to put the numbers 128,64,32,16, 8,4,2 and 1 above the binary number, and add the equivalent base 10 number of each column where a 1 occurs in the binary number. To illustrate this, consider the fourth line of BINary numbers.

	128	64	32	16	8	4	2	1
Binary number	0	0	1	1	1	1	0	0

You will see that 1 occurs in columns 32, 16, 8, 4. So the denary (base 10) equivalent is 32+16+8+4=60.

Conversions like this are not hard once you have got the hang of them. But why not use your Amstrad to do it for you with the STR$ function? I wouldn't recommend that you include this in the converted program because it uses a lot of memory. But typing the binary number at the keyboard and writing the equivalent is a lot quicker, and less error prone, than doing the calculation manually.

Supposing you want to find the denary equivalent of the binary number 1010100. You would type in PRINT STR$(&X1010100) and press Enter. The &X in the statement tells the computer that the number is in binary. When you have got used to converting binary to denary numbers, converting UDGs from one machine to another is very easy.

High-resolution graphics. Diagram 2 shows the screen layout of both Spectrum and Amstrad micros. You'll notice that the graphics origin in both cases is the bottom left of the screen, compared to the text origin which is at the top left.

The first thing you'll need to know about the coordinates is the conversion scale for the two machines. The Y coordinates of the Spectrum go from 0 to 175; on the Amstrad, they go from 0 to 400. Now, 400 divided by 175 is 2.2857142 so this is the scaling factor to use when converting the Y coordinates of any high-resolution graphics.

In practice, it makes little difference whether we use 2.25 or 2.28 or 2.2, as long as we don't use a number bigger than the exact number. This could take the graphics off the edge of the Amstrad's screen.

Similarly, the X coordinates on the Spectrum go from 0 to 255, and on the Amstrad from 0 to 640. This time, the conversion factor is 640÷255 which is almost equal to 2.5. If, for example, you want to find the point on the Amstrad's screen equivalent to the coordinate (191,159), multiply 191 by 2.5 and 159 by 2.28 to get the coordinate (477,362).

If you try the conversion, you'll notice that I've rounded the numbers down to the nearest whole number. Rounding up may take the coordinate off the edge of the screen if the pixel is near an edge on the Spectrum.

It is interesting to note that there are two extra lines on the Spectrum at the bottom of the screen which cannot normally be PRINTed or PLOTted on. I have not taken these into account when looking at the conversion of coordinates.

Plotting Pixels. The PLOT command on both machines is very similar, but the coordinates need to be scaled by the factors calculated above. To illustrate this, consider the Spectrum listing below.

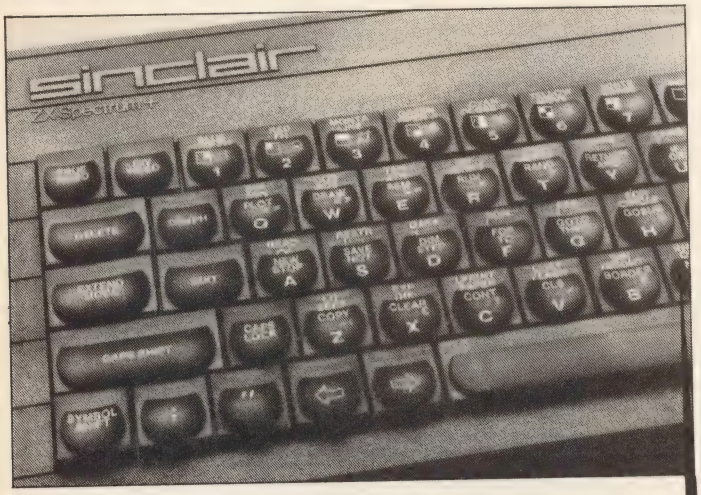
```
10 REM * * Spectrum version * *
20 FOR x=0 TO 255
30 PLOT n, 88 + 80
    * SIN(n/128 * PI)
40 NEXT n
```

The only changes needed to this listing are in line 20. This needs to be changed to:

```
10 REM * * Amstrad version * *
20 FOR x=0 TO 640
```

and line 30, where you need to apply the scaling factor to the numbers.

Note that colour is selected in the PLOT and DRAW commands on the Amstrad by adding a third number at the end of the statement rather than using the INK command, as on the Spectrum. In practice, it is usually better to ignore any INK command in the Spectrum program and add colours to your Amstrad listing to suit the monitor used. Line 30 could then become something like: 30 PLOT n,200+182*SIN(n/-



291 * PI),2

Drawing lines. Both machines use the DRAW command to draw a straight line, but the format is different. The starting point on both machines is the last defined cursor position. If you have not previously moved the cursor in the program, the line is drawn from location (0,0).

On the Spectrum, the DRAW x,y command draws a line x pixels to the right and y pixels up from the previous point plotted. On the Amstrad, the DRAW command draws the line to the absolute point specified. To make life easier, the Amstrad does have a DRAWR command which is the exact equivalent of the Spectrum DRAW command.

For example:

```
5 REM * *Spectrum version * *
10 PLOT 200,100
20 DRAW 40,30
```

would become:

```
5 REM * *Amstrad version * *
10 PLOT 500,228
20 DRAWR 100,68
```

Life starts to become difficult when curves are plotted using a third number in the DRAW command on the Spectrum. There is no command to draw either a circle or an arc of a circle on the Amstrad. The following commands need patience and experimentation to convert.

Commands which can't be directly converted. Draw x,y,a — this draws a curved line where the third number (a) specifies the number of radians through which the arc must be drawn. CIRCLE x,y,r — this draws a circle with centre x,y and radius r.

If a program contains either of these commands, you must write a short routine to produce the same effect. As an example, consider the following Amstrad listing:

```
1 REM * *Amstrad version * *
2 CLS: DEG
10 FOR r=1 TO 360
20 PLOT 320,200
30 PLOT 320+190 * COS(r), 200+190 * SIN(r)
40 NEXT r
```

This listing draws a circle with centre 320,200 and of radius 190. It can be adapted to draw arcs by changing line 10. If, for instance, you changed line 10 to 10 FOR r= 120 TO 240, an arc would be drawn.

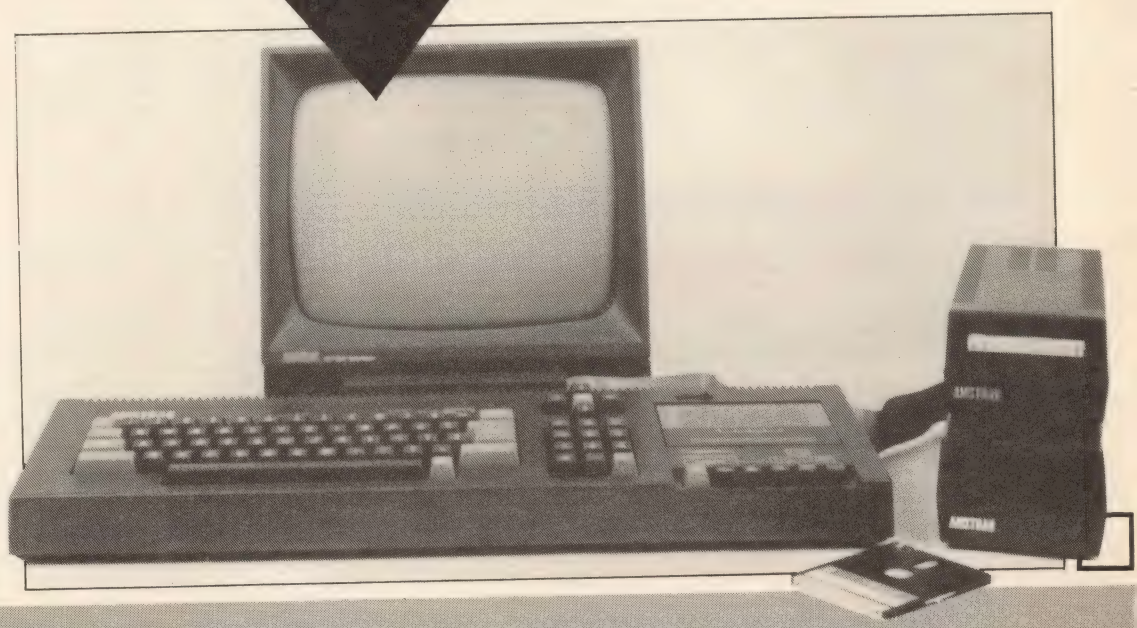
Changing the PLOT command in line 30 to DRAW results in a solid circle being drawn. In practice, you need to spend a great deal of time experimenting with the above listing.

From this point on, things are not so simple. There is no golden rule to aid conversion of a program from the Spectrum to the Amstrad. The Basics of both machines allow excellent pictures to be built, but there is no easy answer. It is easier to convert a listing if you have a picture in your mind of just what effect the Spectrum program produces. My own approach, when I have converted the UGDs and the straight-line graphics, is to experiment with the rest of the program until the picture looks right rather than trying to make a direct conversion.

Sound. Amstrad micros offer far superior sound facilities than the Spectrum. Converting simple BEEP commands is pointless. You would be better advised to forget those parts of a Spectrum listing which deal with sound. Write your own to make full use of the facilities of your Amstrad.

Conclusions. You will have realised by now that there is no easy answer to converting a Spectrum listing. If you are a good programmer, you can convert anything — but then you wouldn't need these articles to guide you in the right direction. Before beginning any conversion, you do need to know your own limitations and search through the Spectrum listing carefully for any problem areas. When you spot any of the 'nasties', you must decide whether the trouble is worth it. It can be frustrating to spend 10 hours converting a program, only to find that you're not up to converting a small section.

If I've made it sound as if the Spectrum offers superior graphics to the Amstrad, I must apologise. The two machines use different commands to handle them, and in many cases it is easier to rewrite a section than to convert. That way you can make better use of the Amstrad's graphics.



Pack a byte!

Dave Carlos explains byte packing — or how to get as much data as possible into as little memory as possible.

Many programmers will have bumped against the problem of trying to squeeze as much information as they can into the least memory. One good method is 'byte packing'.

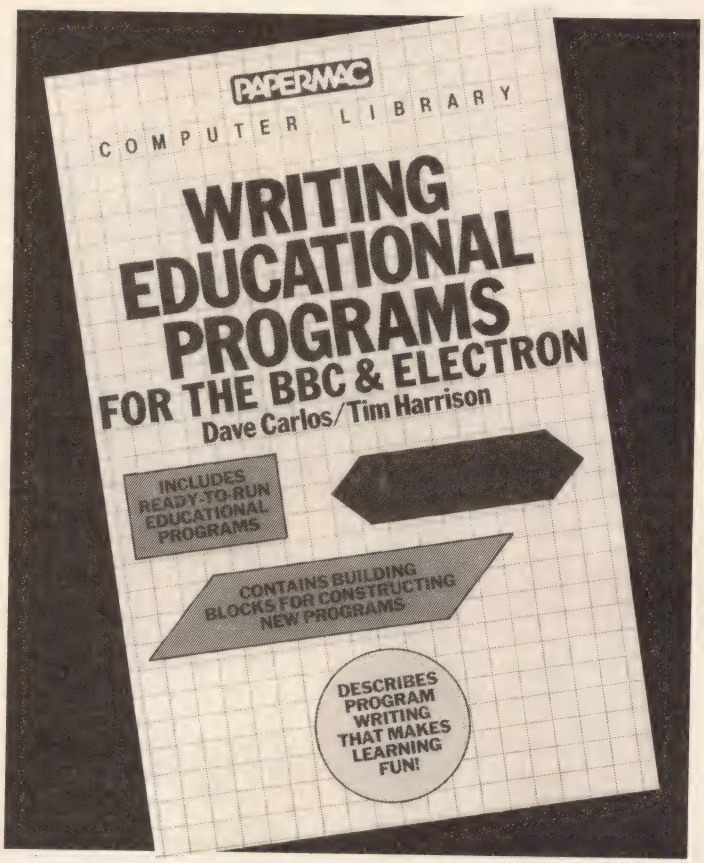
Before you can appreciate this technique, you need to understand what a byte and a bit are. You probably know that a computer's memory is made up of thousands of little switches, each of which can be set to on or off. These are called bits; when you get a collection of eight of them, they are known as a byte. Guess what number less than eight — usually four — is called? Well done, a nibble!

Each bit in memory can be set to 1, which is on, or 0, which is off. They are called two-state devices because of this. By combining the eight bits and a little binary arithmetic, we discover that a number up to 255 can be stored in just one byte.

In my program *Humber* in the July issue of *PCT*, we needed to store not one but three numbers in one byte. How could this be done?

Humber is an educational program in which you must build roads across varying terrains to link towns. The cost of each unit crossed varies from £1 million to £10 million: our first number. The terrains crossed — agricultural land, woods, towns and river — also had different colours to represent difficulty level: our second number. And checking if a road was built was made with key presses: our third number. To store the three numbers, we must make a list of their maximum values.

In our case, the first value can be anything from 1 to 10 which can be converted and held in only four bits (2 to the power 4 equals 16, but 2 to the power 3 equals only 8). The second set of values, the colours, can range from 0 to 3 only. They can be represented by two bits (2 to the power 2 equals 4). The final value to be stored is the in_route 'flag'. A flag is a single bit which can be only true (on) or false (off). So this needs only one bit.



Adding all these requirements gives a total of seven. That means all these values can be squeezed into a single byte, or eight bits.

Imagine that you are devising a numerical code to send or store this information. Start by writing the value of the units crossed as a decimal number. It could be 6 for a £6 million project or 10 for a £10 million project.

Add to this a value representing the colour but coded as follows. Colour 0=0, colour 1=16, colour 2=32 and colour 3=48. These are all multiples of 16. Remember that this should be added to the starting value.

Finally, there is the 'already built flag' to incorporate in the calculation. This is done on the following basis. If the area is not already built over, add 0 times 128 to the total. That is, add 0 or leave it alone! If the area is built over, add 1 times 128 to the total so far.

Just to convince you that the 'quart into a pint pot' routine works, let's take two examples. A river location of value 6 has a colour blue, 1, and has been built over. This means that its value after being converted to our code will be 6+(1x16)+(1x128) =6+16+128=150. Notice that this is not above 255 so can be contained easily in one byte.

If we take a simpler case, a town square to be built over has a value of 10, a colour of black, 0, and hasn't been built over. So its coded value is 10+(0x16)+(0x128)=10+0+0=10.

All we need to do now is to explain the maths involved in getting the values out of the byte again. But if you can code something you should be able to decode it by simply reversing the process. This is all we do in the program. But to save space — in a really tight memory, otherwise we wouldn't need to do all this extra work — we can use some concepts of 'Boolean algebra'. It is sufficient to say that they simply reverse the coding we have just explained.

That's all there is to it. But look at your past programming... I bet that you have used the technique before!

Dave Carlos and Tim Harrison are co-authors of Writing Educational Programs for the BBC & Electron, *published by Papermac.*

From bluff to buff

David Nowotnik unravels the often confusing 'computer speak' with an explanation of structured programming.

BLUFF

Don't panic! Just read these simple definitions before starting the article.

Command: a computer instruction in machine language.

Function: A definition of the value of a dependent variable based on the value of an independent variable.

Read: interpret, or get data from.

String: a set of consecutive, adjacent similar items.

REM or REMark statement: a program line which serves as a note to the user about what is happening in the program.

GOTO: an instruction for the program to branch.

GOSUB: an instruction to go to a subroutine.

Loop: a sequence of instructions repeated until a terminating condition or result is met; reusing instructions in this way, rather than duplicating them, saves memory.

With Basic as the universal language of home and business micros, it is little wonder that most novice programmers first program micros with this language.

The reason it is so popular is straightforward — it is much easier to use than any other all-purpose language. Basic has a simple but comprehensive vocabulary of commands and functions. These 'words' of the language are combined with simple rules of grammar (syntax). And line numbers make it easy to edit, add, or delete 'phrases' when composing the 'text' (the program).

The general simplicity of Basic makes it almost ideal for the first time programmer, but several criticisms have been levelled by many professional programmers. The most common criticism is the lack of 'structure'. Lack of structure, so we are told, makes it difficult to 'read' program listings, so inhibiting anyone except the author understanding how the program works and, if necessary, improving or debugging programs (removing errors which cause the program to go wrong).

As the hardware side of micros has improved over the past few years, so have the operating systems and Basic interpreters. Attempts have been made to allow amateur programmers become more professional in their approach.

A structured approach to writing Basic programs isn't all that difficult. It means taking a little more care and planning. The examples in figures 1 and 2 show some differences between the structured approach and the more common, amateur, method of writing in Basic.

Take the example of a simple routine to convert a decimal number into hexadecimal (a number with a base of 16). There are a few ways to do this; I've adopted the following sequence:

1. Divide the number by 16, and work out the remainder
2. Convert the remainder to a character, so that 0 = ''0'', 8 = ''8'', 10 = ''A'', 13 = ''D'', etc.
3. The new number is the integer of number divided by 16; if this is not 0, go back to 1
4. If the number is now 0, the conversion is complete; as long as the string of characters is in the correct order, you can use the result!
5. Hexadecimal numbers are usually written in blocks of two digits. So, to complete the conversion, a leading 0 is added to the string if the string is of uneven length.

Given the basic conversion method, the

Figure 1: an unstructured decimal to hexadecimal routine for the Spectrum

program listing

```
100 INPUT a
110 GO SUB 1000
120 PRINT a$
500 STOP
1000 REMark routine to convert dec to hex
1010 LET a$=""
1020 GO TO 1100
1050 LET z=y+48+7*(y>9)
1060 LET a$=CHR$(z)+a$
1080 GO TO 1140
1100 LET x=INT(a/16)
1110 LET y=a-x*16
1120 LET a=x
1130 GO TO 1050
1140 IF a<>0 THEN GO TO 1100
1150 IF LEN(a$)=0 THEN a$="00"
1160 IF LEN(a$)/2<>INT(LEN(a$)/2) THEN a$="0"+a$
1170 RETurn
```

Figure 2: a structured decimal to hexadecimal routine for the QL

program listing

```
100 INPUT a
110 PRINT hex$(a)
1000 REMark procedure to convert dec to hex
1010 DEFine FuNction hex$(n)
1020 LOCal a$,y,z
1030 a$="": REMark empty string to build hex number
1040 REPeat loop
1050     IF n=0 THEN EXIT loop
1060     y=n MOD 16: REMark y=remainder of a/16
1070     a$=CHR$(y+48+7*(y>9))&a$
1080     REMark   convert y to a character and add
1090     REMark   it to string a$.
1100     n=INT(n/16): REMark  Next hex digit.
1110 END REPeat loop
1120 z=(LEN(a$)=0)+2*(LEN(a$)/2<>INT(LEN(a$)/2))
1130 REMark z=1 if len(a$) is 0; z=2 if len(a$)=odd
1140 SELect ON z
1150   =1: a$="00"
1160   =2: a$="0"&a$
1170 END SELect
1180 RETurn a$
1190 END DEFine hex$
```

average Basic programmer will probably type in lines which he/she hopes will carry out the conversion. In the process, something of the nature of figure 1 might result.

Figure 1 is a piece of Basic for the Spectrum, written badly to emphasise what might happen from lack of planning. The routine works — after much trial and error, and several line changes/additions/deletions. But it is difficult to see at first glance how it does so.

The routine jumps about — termed 'spaghetti programming', as the program follows an almost random path through the listing — caused by the author changing his/her mind about how to tackle the problem. It is only a short routine, but it takes time to see how the method has been converted to Basic code.

By now, many programmers may (justifiably) feel insulted by associating the listing in figure 1 with their abilities as programmers. But remember: figure 1 is meant to illustrate the worst of common programming practise.

Now take a look at figure 2. This piece of Basic written for the QL carries out the same conversion. It is longer, but should be far easier to follow once you are familiar with all the commands that the QL's SuperBasic has to offer.

The first thing to notice is the abundance of REM statements. This is not necessarily a feature of structured programming, but it is a general good practice to enable programs to be clearly understood. The listing in figure 1 would be helped by a few REM lines, but it would still suffer from the jumps created by GOTOs.

One feature of structured programming is the absence of GOTOs or ▷

GOSUBs. The program should flow naturally from beginning to end, with any loop structures clearly visible by the identification of program lines such as lines 1040 to 1100.

The program should be able to do without line numbers altogether. Forth and Logo are two structured languages which do not have line numbers, and Pascal has them only for the convenience of editing lines; this should also be the case for well-written Basic programs.

SuperBasic on the QL is specifically designed for structured programming. While SuperBasic has GOTO and GOSUB in the command set, there should be no need to use them. Let's take a look at figure 2 in more detail to see how the QL can support structured programming.

Line 100 (INPUT a) is common to all variants of Basic, but line 110 is unusual. It refers to a function called 'hex$'. This is a standard function built into SuperBasic, but a user-defined one. It is defined in the Basic routine, lines 1000 to 1190.

The QL can allow the user to define his/her own commands and functions. To continue the analogy used at the beginning of this article, if commands and functions are the 'words' which are combined to form the text of the program, then the QL allows the user to increase that vocabulary. Just as a dictionary defines new words using words that should already be familiar to the user, so the QL can be taught new words using existing words (commands and functions).

The user-defined commands (called procedures) and functions must have a fairly independent existence. Take a look at figure 2 to see what this means.

The start of the definition of hex$ is on line 1010. It specifies the name of the new word and what parameters need to be given to this particular function to make it work. The function converts a decimal number to hexadecimal, so a decimal number (variable n) has to be 'fed' to the function to make it work.

The defined function is made independent from the rest of the program by the LOCAL command. The new function uses variables, defined by the LOCAL command, which ensures their independence from all others in the program, even variables of the same name used elsewhere.

The variable n is also a local variable; notice line 110, where the hex$ function is used. This sends the variable a to the function hex$, where it is converted to the local variable n for manipulation in that function. The variable a remains unchanged.

The abundance of REM statements should make the rest of the user-defined function in figure 2 very easy to follow — although MOD, REPeat, EXIT, and SELECT may not be clear to non-QL users. Once the hex number has been built-up in the local variable a$, this string is returned (line 1180) at the end of the function to the main routine.

As well as the smooth 'flow' of the defined function from top to bottom, notice the indentation of the loop, which marks its beginning and end. When activated, the EXIT command (line 1050) sends the program to the line which immediately follows the end of the REPeat loop (line 1110). SuperBasic also offers the SELECT command, and IF, THEN, ELSE, END IF structures which help to make GOTO unnecessary.

A structured program is made of one or more user-defined commands and functions. Planning is essential to decide what activities need to be combind to produce the new 'words', then how the new words are combined to produce the program. In structured Basic programming, it is usual to define one word which pulls together all major operations which form the program. RUN in Basic is one example: it causes the program to operate.

In planning a program there are two approaches. You can start by deciding what will be the simplest operations carried out by the program, and write and code these as new procedures or commands. These new words can then be used to define more new words, which link the simple logical operations. More words are added to link these higher definitions until, eventually, all the operations of the program are combined to make an effective whole. This approach has been termed the 'bottom up' approach.

The opposite can also be done: the 'top down' method. It can be likened to the organisation of a business. At the top of the management tree is the MD, who decides what the business should be doing, and (in broad principle) how it should be done. To achieve this, the MD sets up departments, each with specific aims, which should be combined to meet the aims of the business. In each department, a head sets up sections, whose individual activities combine to meet the department's aims. Finally, each section has workers, who work together to meet the section's aims.

The top-down programming method works in exactly the same way. Decide on the aim of the program, break that down into the major operations, then break down each of these operations into manageable progressively smaller units, which can be defined as individual procedures or functions.

The choice of top-down or bottom-up programming is very much a personal one. You'll probably adopt one naturally without realising it. The important feature of both is planning. Code on paper first, making sure that you have a logical flow through the routine. Only when all this is complete should you punch keys to see if your paperwork does allow you to create the program which you have planned.

Try adopting these few simple rules, and you should write better, more 'readable' Basic programs.



ON SALE FROM 23 AUGUST

Next Month*



The October issue of *Personal Computing Today* is a must for all micro owners, whether interested in games, education or business. Make sure of your copy by handing the form below to your local newsagent.

GET THE LOW-DOWN ON LINGO

If you want to start programming, our cover stories are for you. They tell how to choose the right language for what you want to do, and what the pros and cons of the different ones are. They review the most popular language packages on the market. And they speculate what the computer languages of the future will be.

ROBOTS — THE £2 BILLION INDUSTRY?

Domestic robots will grab a massive £2 billion market share in the next five years — and the robot industry will emerge as greater than even the car industry. So state the manufacturers, despite the company closures here and in the US. Read how they plan to do it.

HACKER'S WORLD

We reveal hints and gossip from this fascinating sector, and bring the latest network news.

DRIVING LICENCE

Disk drives add power and speed to your micro. We review some of the latest and bestselling ones for the most popular micros.

TEACH YOURSELF TO PROGRAM

Don't miss October's guide for beginners and serious programmers alike. Create advanced graphics on the Atari or experiment with sound on the MSX. Convert the many and cheap TI99/4a listings to the Amstrad in the first of a two-part series. And discover what's so different about Commodore's new Basic 7.

STARTING UP

Tired of games? Or want to use your micro for business but don't know what you need? Our new series for business aspirants assesses how you can set up a basic management system — micro, monitor, disk drive and software — for well under £1,000.

REGULARS

Save time and money with our reviews of new books, games and educational and business packages . . . type in our arcade and utility listings . . . catch up with the news on products and personalities . . . demystify 'computerese' with the help of From Bluff To Buff . . . and enter our big-prize competition.

*Every effort is made to ensure that the individual articles mentioned appear in the next issue, but they may be altered due to unforeseen circumstances.

COMING NEXT MONTH TO A NEWSAGENT NEAR YOU!

GIVE THIS TO YOUR NEWSAGENT

Please reserve me a regular copy of Personal Computing Today

Name .....

Address .....

.....

.....

.....

Adventure corner

Can John donovan make an Exodus, pass the Assassin, and avoid Angélique — a grief encounter?

Greetings! I have an offering from across the Atlantic this month — *Exodus*, which is the third (why aren't the other two here?) in the Ultima Series. There is another classic text-only adventure from Robico and an oddity from Nemesis called *Angélique — a grief encounter*.

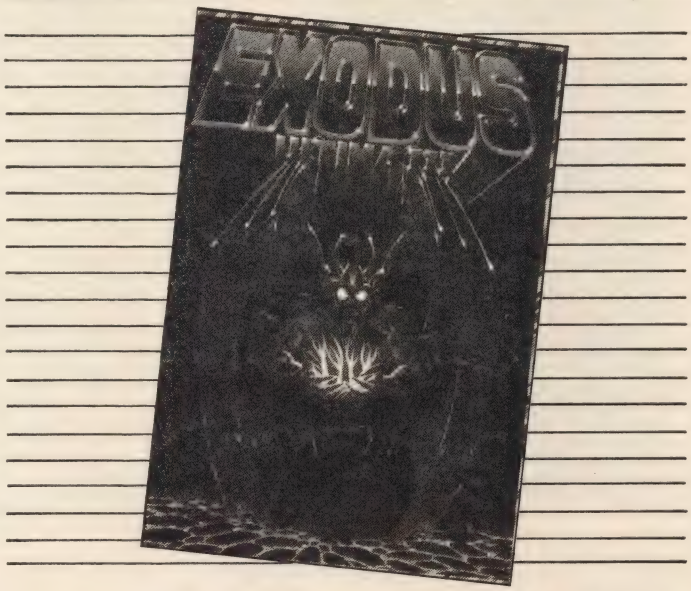
First, let's look at *Exodus*. This £19.95 US import was first released on Atari micros but has been converted to the Commodore 64. The game is a sort of hybrid, made of the best bits of *Dungeons and Dragons*, some arcade and a lot of adventure.

The first thing to do is create characters. You do this by dividing 50 points between four attributes: strength, dexterity, intelligence and wisdom. Dependent on the strengths and weaknesses of a character, his/her class is chosen. For example, a wizard needs a high intelligence but, as he can use only a dagger, strength is not as important as it would be for a warrior. Once the player has four such characters, he/she can begin.

The band appears on a large map with a little man holding a sword. To the north lies a castle and a town. You can issue instructions for the man to move or to arm his men. At this point, my advice is to run to the nearest town as quickly as possible.

Now the game takes the form of a map exploration, with the band nipping in and out of the nearest town, killing some orks, then running quickly for cover. In this way, they amass gold to buy better weapons and armour.

When the band is strong enough, it can journey overland to another town. The problem here is that, on a long trip, more than one attack from foes can be expected. You must have



enough strength to withstand them without outside help.

Ultimately, the band must venture through one of the mysterious 'moon gates' which takes them to the more interesting sections of the land — and of the game. There are dungeons and a ruined castle to be explored before you meet the big baddy — Exodus.

This is an original and interesting game. It isn't a classic adventure but it is one that I recommend.

Robico is a small firm which specialises in machine-code games for the BBC. Robert (he of the Robico name) wrote *Island of Xaan*, a classic text game about which I was somewhat critical, mainly because of its patchy descriptions and poor vocabulary. *Assassin*, selling at £9.95, is a totally different kettle of fish.

The story starts by describing you, Rock Hanson, special agent. You are a man in the Dick Barton mould. Rick appears sporting a light-brown raincoat at a railway station, with absolutely no idea of what to do.I walked boldly out of the waiting room — to be promptly shot by a man with a rifle across the street. Next time, I decided to explore the station, and found myself in a telephone booth with a tape recorder.

Play the tape. You'll hear a message from your boss, telling you what to do and where to find more information. As every good secret agent knows, messages from the boss always self destruct three seconds after the tape ends — so get out fast!

Needless to say, the next set of instructions is in code. This took me 45 minutes to crack.

The game boasts 220 locations, all of which are superbly described, and many different puzzles. I won't tell you more about the game, not because I haven't finished it — I have! — but because games of this quality are becoming rare and I don't wish to spoil it for you.

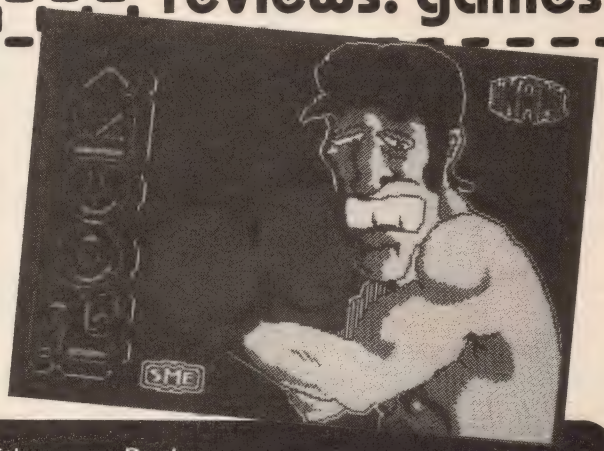
Next comes the oddity: *Angélique — a grief encounter*. Costing £7.95, this runs on the Amstrad and uses a strange window display. A large window shows the usual adventure conversation, and a smaller one shows what you are holding. Other windows each have their own purpose.

The game starts in the crater of a volcano with no visible exits.

Unusually, this game has a heroine, rather than a hero — and she won't let you forget it. For instance, ask her to search her handbag, and she refuses on the grounds that she doesn't like people rummaging through her things. With tact, she can be made to divulge the contents. But she'll refuse to read a letter because 'it's private'.

That's all for this month — do send hints and posers about your favourite adventures and I'll try to include them in next month's column.

Software selection



Title: The Way of the Exploding Fist
Software house: Melbourne House, Castle Yard House, Castle Yard, Richmond TW10 6TF
Price: £9.95, cassette

Machine: Commodore 64

Martial arts fanatic or not, try this karate simulation game. Don't let the slightly bizarre title put you off.

You control your karate fighter with joystick or keyboard. You may play against the computer or a friend. There is also a demonstration mode.

The two characters fight in Japanese settings which make the game both colourful and attractive. The fighters are 'watched' in each scene by a 'judge' who awards points.

The scoring is a little complicated and rather poorly set out onscreen. But the characters are very realistic and can perform 18 manoeuvres including blocks, flying kicks, leg sweeps,roundhouse and even somersaults. Don't worry if some of these terms mean nothing to you. What is important is that you master the great variety of kicks and punches. Herein lies the skill of the game.

Against the computer, you start as a novice and work your way up through the hierarchy of 'Dan' levels, being pitted against increasingly formidable opponents. The objective is to reach 10th Dan.

Graphics are excellent and sound effects even better, with haunting oriental music and realistic shrieks, grunts and thuds to accompany the fight.

This game is highly amusing, original and thoroughly recommended. **D.J.B.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * * *
Ease of use	* *
Overall	80%

Title: Rocky
Software house: Gremlin Graphics, Alpha House, 10 Carver Street, Sheffied S1 4FS
Price: £7.95, cassette

Machine: Spectrum 48K

After some trouble loading the game (it is a hyperload) I was presented with a starting menu laid out very similarly to that of *Monty Mole*. Here the similarity ends.

I must be one of the few people in the UK who has never seen any of the *Rocky* films, so I was interested to see what everybody was raving about. The game, with its well-designed graphics — they look almost digitised! — manages to capture the atmosphere of a real-life fight. The first time I saw 'Jansen Sino', I felt like jumping out of the ring and legging it. I dread to think what 'Fighter Bull' looks like — I have yet to beat Mr Sino.

You can almost feel each blow, as a left hook turns your face to the side with a look of agony on it.

The screen layout consists of the fight on the top two-thirds of the screen, with information and photo of the two fighters in the rings. There is also an energy-bar for you and your opponent. When this bar decreases into the red, another couple of well aimed blows will leave you unconscious. Three knockouts and you're out of the fight!

A nice touch is the bell ringing before each bout. You can see it vibrating as the clanging sound comes out (quietly) from your Spectrum loudspeaker.

This is another addictive sport game and the keys are incredibly easy to use. I think it could be a hit. **D.S.**

STAR TABLE	
Screen display	* * * * *
Addictiveness	* * *
Ease of use	* * * * *
Overall	90%

Title: Quackshot
Software house: Creative Sparks, Thompson House, 296 Farnborough Road, Farnborough, Hants
Price: £2.50, cassette

Machine: Spectrum 48K

This is one of the low price games from a collection called *Sparklers*. If they're all as good as this one, buy them all.

Quackshot is set in a toy factory. It's your first night as nightwatchman and it's not going to be a quiet one. The toy ducks are revolting!

Snakes, ducks, spiders and more roam around a maze, all with a strong dislike for the nightwatchman. You have to get them before they get you. Your weapons are a stun gun which shoots from side to side and duckbuster bombs which destroy all the toys on the screen.

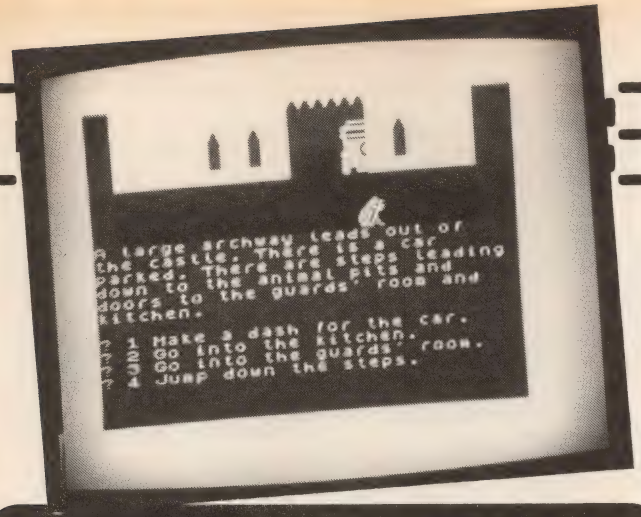
The game consists of 16 mazes which scroll from side to side, with a helpful map displayed at the top of the screen.

Sooner or later, you come to a door which must be unlocked with one of the keys scattered round the maze. Only one particular key opens each door, and you can carry only one key at a time, so you often find yourself backtracking.

The game is well designed with flicker-free graphics and smooth animation. Use of colour is good. Sound includes some short jingles and good game effects.

Quackshot is excellent value for money and a great game. **A.A.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * * *
Ease of use	* * * *
Overall	90%



Title:	Phineas Frogg
Software house:	Widget — through Mirrorsoft, Maxwell House, Worship St, London EC2A 2EN
Price:	£7.95 cassette, £10.95 disk

Machine: Spectrum

Phineas Frogg is a fine gent of an amphibian — with a licence to kill! The evil hamsters of sloth have kidnapped the good professor mole who is working on a new secret wotsit.

The scene is set and the game offers a colourful package. This includes a booklet telling the story, complete with hidden clues and a secret map.

Your task is to rescue the professor. The game starts as you wind your way to the hamsters' castle. You are given a written description of each location as well as some excellent animated graphics.

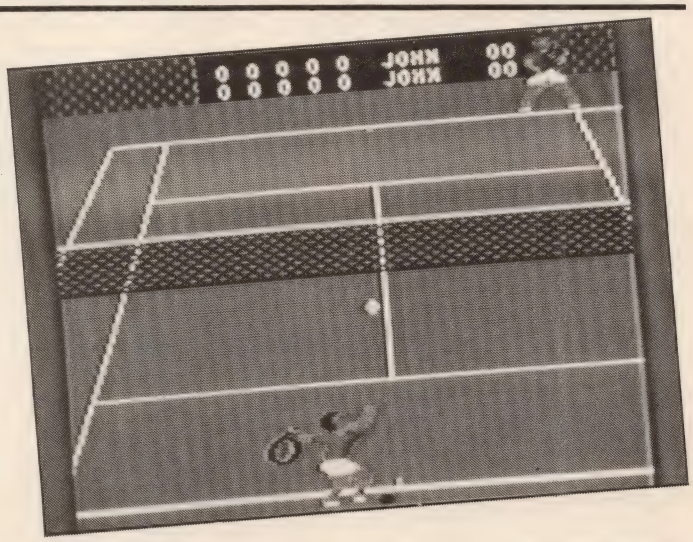
Interspersed with the animated scenes are five small arcade and logic games, including a simple *Frogger*. These are easy but fun.

This game is well thought out with lasting appeal, especially for young people. **P.C.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * *
Ease of use	* * * * *
Overall	73%

Title:	On-Court Tennis
Software house:	Activision (UK), 15 Harley House, Marylebone Rd, London NW1 5HE
Price:	£10.99, cassette

Machine: Commodore 64



On-Court Tennis differs from other C64 tennis games because you don't control the movement of your player round the court. Instead, you control the types of shot you want by pushing the joystick in different directions. This is more difficult than it sounds because you must time your shots accurately. You have a range of shots for rallying, serving and smashing.

There are many game options to choose from, such as demo, one-or two-player games and one, three or five set matches. You can also play on grass, hard court or clay surfaces, each with different characteristics.

There are four players to choose from: John, Jimmy, Bjorn and Ivan. Their styles of play are accurately reproduced.

The graphics are good but some of the animation could be better. The ball increases in size as it rises in the air. This is a nice touch and particularly pleasing when you lob the ball.

This game reproduces the skill and strategy of tennis accurately. At first you can expect to miss the ball often and to make lots of double faults when serving. But when you've mastered the timing you can play a very enjoyable game. **M.H.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * * * *
Ease of use	* *
Overall	95%

Title:	Jump Jet
Software house:	Anirog Software House, Unit 10, Victoria Industrial Park, Dartford
Price:	£9.95, cassette

Machine: Commodore 64 (also Amstrad)

This is a combined flight and combat simulator. You are placed in the pilot's seat of a Jump Jet fighter plane. Your aim is to successfully carry out a mission which entails the seeking and destruction of enemy aircraft.

The number of planes to be destroyed depends upon the skill level, selected from a choice of five. The lowest level has no enemy jets to chase and is to let you practise takeoff, landings and general manoeuvres. On the highest level, that of group captain, you must find and destroy four enemy planes, whilst coping with mountainous seas and gale force winds.

A special feature of the game is the use of speech at some stages of play. This is extremely well done and very easy to understand. I particularly liked the MayDay call before crashing into the ocean.

Three separate screen shots appear throughout the game. Two relate to the landing sequence and the third gives an impressive view of the ocean and sky.

My only criticism of the package is that the instructions could give more help in explaining how to fly the plane.

This is a reasonable game, but cannot be described as a simulator. **G.W.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * *
Ease of use	* * * *
Overall	70%

Title: A View to a Kill
Software house: Domark, 204 Worple Road, London SW20 8PN
Price: £10.99, cassette

Machine: Spectrum 48K

Enter Max Zorin, evil genius. His ambition is to dominate the micro chip industry. It's your job, as super-spy James Bond, to thwart his evil plan.

Zorin has planted an atomic bomb in a mine beneath California's silicon valley. When exploded, it will set off the San Andreas fault.

The game is in three parts, each linked by code numbers. Game one is set in Paris. Your informant has vital details of Zorin's plans. But before he can pass them on, he is murdered by May Day, one of Zorin's henchpersons.

May escapes from the Eiffel Tower by parachute. As she descends to the streets below, you must chase her by taxi.

Having captured May Day, load the next game. Here you and your heroine, Stacey, are in San Francisco's City Hall. Unfortunately, someone has set fire to the building — and to make matters worse, Stacey is trapped in a lift which is stuck between floors.

Assuming barbecue Bond is not on the menu, we now move to game three. In my opinion, this is the best of the bunch.

Bond has finally tracked down the location of the atomic device, in a disused mine under silicon valley. You must locate and disarm the bomb before your time runs out. Hopefully, with the aid of May Day, some code numbers, a few sticks of dynamite and objects you collected, you will get there in time. But watch your step! One false move and it will be a bumpy ride to the bottom.

For the price, this game gives fair value for money. **T.K.**

STAR TABLE	
Screen display	* * *
Addictiveness	* * * *
Ease of use	* * * *
Overall	80%

Title: The Rocky Horror Show
Software house: CRL, CRL House, 9 Kings Yard, Carpenter's Road, London E15
Price: £8.95, cassette

Machine: Commodore 64 (also Spectrum, Amstrad)

This game is based on the cult film of the same name. You first choose whether to be Brad or Janet, the hero and heroine of the film. Then you must search the many rooms of Dr Frank n Furter's castle to locate the dismantled pieces of the De-Medusa machine.

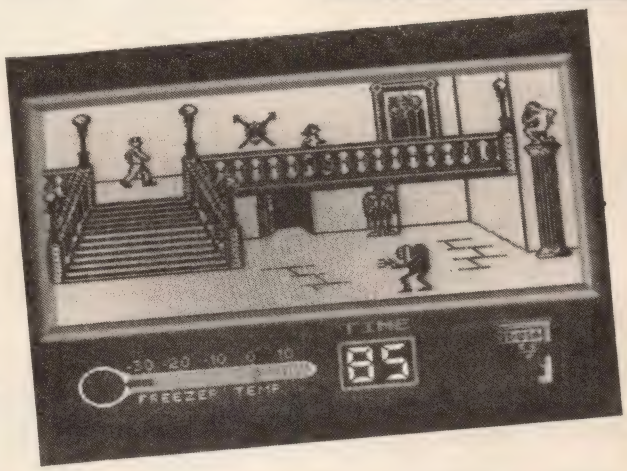
This is the one and only machine that will save your partner who has been kidnapped and turned to stone. Time is not on your side as the castle is really a disguised space ship ready to blast off.

To complicate matters further, Frank's friends are wandering about, eager to hinder you in your quest. Some characters zap you, others strip you — forcing you to shuffle around embarassingly naked searching for your clothes.

The graphics are good but straightforward. Sound is excellent with background music based on a song from the film. There is no joystick option which would have improved game play.

Although the program appears original it is basically a search and collect game, and becomes boring once the novelty has worn off. But it should do well with *Rocky Horror* freaks! **J.G.**

STAR TABLE	
Screen display	* * * *
Addictiveness	* * *
Ease of use	* * *
Overall	65%



Title: The Chess Game
Software house: Micro Classic, Priory Road, Forest Row, Sussex RH18 5JD
Price: £7.95, cassette

Machine: Commodore 64

You are a pawn on a chess board, but it is no ordinary chess board. It is a huge and hellish stadium where all the pieces are against you, the audience is after your blood, and even the board is hostile.

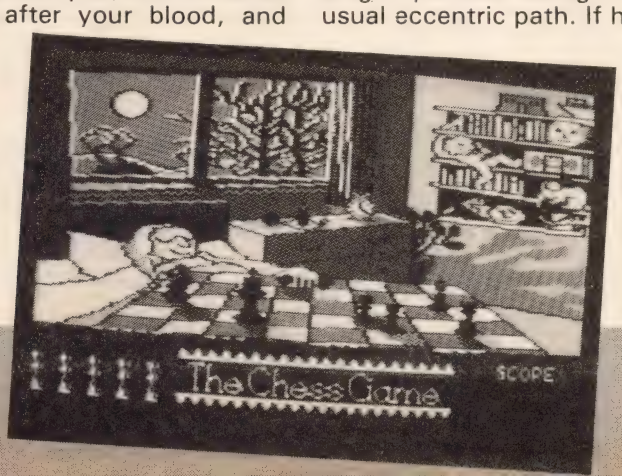
Your first opponent is the knight. As you move about the squares, their colours change and the knight pursues along his usual eccentric path. If he lands on the square you leave, the colour changes again and that square must be avoided for the rest of the game.

As you progress you meet the bishop, the rook, the king and the queen in excellent graphic action.

The concept of this game is novel and the 3D graphics are of a high standard. However, once I'd seen the spectacular title screen and played the game for half an hour I found it slow and monotonous.

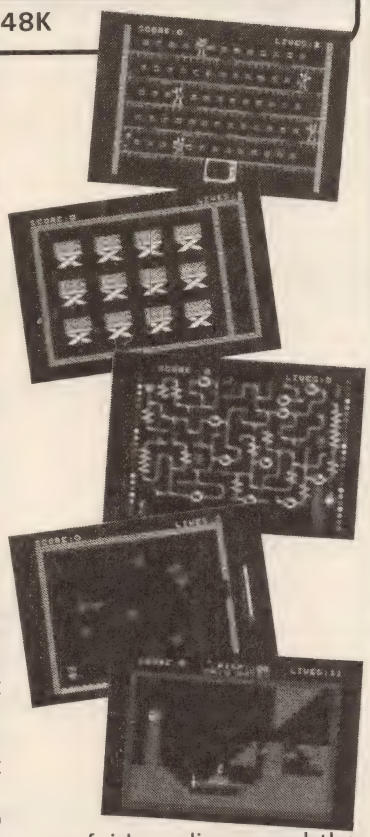
The idea of being chased round a giant chess board by larger-than-life pieces might appeal to some people, but not me. **B.W.**

STAR TABLE	
Screen display	* * * * *
Addictiveness	* *
Ease of use	* * * *
Overall	60%



Title: Charlie and the Chocolate Factory
Software house: Hill MacGibbon, 8 Grafton Street, London W1
Price: £9.95, cassette

Machine: Spectrum 48K



For £10, five games and a book seem like good value. Unfortunately, four of the games are terrible. All of them, though linked with the chocolate factory, are little more than the jerky Basic arcade games you find in most magazines' program listings.

In Game 1, you must twist white piping to help a strange little purple blob escape.

Game 2 is the best on side one. You bounce blueberries into a machine without being killed. Actually, if you stay in the corner of the screen long enough, the blueberries will bounce themselves into the machine without you lifting a finger. (When I said this was the best on side one, I didn't mean it was good.)

In Game 3, you drop nuts to kill squirrels chasing you. It is almost impossible to move, so you tend to wind up dead.

Game 4 is reasonable. You must collect all the Wonka bars (actually flashing Ws) without having your photograph taken by rampant television cameras which rush round after you.

Side two consists of a 43-screen platform game, which is fairly addictive. But it is not up to the high standards of today's gaming public. The graphics are fairly ordinary and the sound is worse.

Charlie and the Chocolate Factory could have made quite an interesting adventure game. Instead, the makers produced a selection of fairly turgid arcade games. It will probably sell well because of its name, but I don't recommend it. **D.W.**

STAR TABLE	
Screen display	* *
Addictiveness	*
Ease of use	* *
Overall	30%

Title: Herbert's Dummy Run
Software house: Mikro Gen, 44 The Broadway, Bracknell, Berks
Price: £9.95, cassette

Machine: Spectrum 48K (also C64)

This game follows the theme of previous Mikro Gen *Wally* titles. The graphics are excellent with large, colourful (as far as possible on the Speccy) and varied locations as well as brilliant animation.

The game is set in a department store. It is 1.00pm and Herbert is in the toy department. You must get him back to his parents in the lost-and-found office by closing time, 5.30pm. This involves solving a difficult adventure and completing several arcade sequences.

Nasties occupy some of the locations and move about the screen. In the bedding department, Zs float in waiting. In the hi-fi department, notes float down, sapping your endurance if touched.

Most of the objects are useful to collect. Herbert can carry only two items at once and they are necessary to complete various tasks. Any food that Herbert passes over boosts playing time.

This game is similar to *Pyjamarama* and is worth buying. I enjoyed playing it very much. **A.O.**

STAR TABLE	
Screen display	* * * * *
Addictiveness	* * *
Ease of use	* * * * *
Overall	90%



Title: The Key Factor
Software house: Amsoft, Brentwood House, 169 Kings Road, Brentwood, Essex CM14 4EF
Price: £8.95

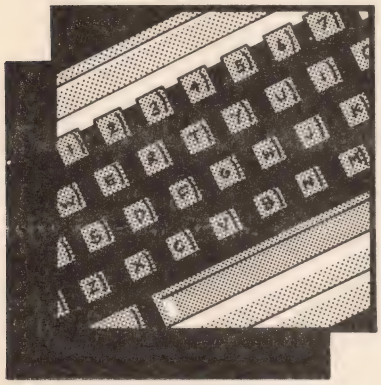
Machine: Amstrad CPC464

This game is designed to improve your keyboard skills. Your task is to shoot down invaders as they descend the screen by pressing the appropriate key to match the letter they are about to hit.

There are five levels of difficulty. In the higher levels, the keys you must press change very frequently. So you can find yourself frantically searching for a particular letter — only to discover that you now need a different one to shoot the alien.

This is an excellent, well designed program with colourful graphics and good sound. It will appeal to young children but is not the kind of game for those who have mastered the keyboard to play again and again. **S.C.**

STAR TABLE	
Screen display	* * * * *
Addictiveness	* *
Ease of use	* * * * *
Overall	80%



Super typewriter

Frustrated with the complex task of printing on your Commodore? Then type in William Fong's listing to turn your micro into a wordprocessor.

CBM64

I bet there are a lot of Commodore 64 owners who have a printer but cannot afford a wordprocessor. And there must be those who have a typewriter but don't want a wordprocessor because they think it is too complicated to use. Well, here's a program to change all that.

Super Typewriter allows you to insert and delete text onscreen. It allows you to set the number of characters you want per line, and word wrap your text. This means that it will not split words on the end of a line, but bring them down automatically to the next line. It allows you to print your corrected copy with just one keystroke — and you can make as many copies as you like.

Type in the program exactly as in the listing. You can, of course, omit the REM statements as they are there only to explain what the reverse symbols are and what the main sections do — half the program is REMs, reversed to be more clearly seen. When you have finished typing, save a copy before running it.

When you run the program the screen should turn black, and a small message should be displayed while machine code is placed in memory.

how it runs

1-8	REM
9-12	Check whether the program has been run before. If so, it skips the section which puts the machine code in memory, so allowing no delays.
13-22	Here the program places the machine code in memory. It adds the data read, and when there is no more data it compares the sum of the numbers with a constant. If they do not match, there is an error and the program stops, displaying a message.
23-52	Data containing machine code. Make sure you type this in very carefully.
53-58	Variables.
59-66	Initialisation. It turns the volume to maximum, makes Basic execute faster, prints in lower case, makes all keys repeat, locks Commodore and Shift keys, and executes machine code at address #49152.
67-81	Display a simple title and allow you to input the right margin and your printer device number.
82-92	'OK.' Ready for your text.
93-102	Routine to get A$; makes a beep, prints the small cursor as an arrow.
103-112	Illegal key presses; go back to get A$.
118-127	Is your key a Delete? If yes, print it, cut the last letter of text, and push the arrow back by printing a space. Go back to get A$.
128-139	Is your line longer than the specified right margin? If yes, use another string.
140-144	Continue, print A$. The text then equals text plus A$. Go back to get A$.
145-158	Subroutine to print text to given device; prints it with full word wrapping.
159-169	Finish printing; reprint text onscreen so another copy can be made.
170-184	If you don't like your letter, you can erase it — here's the routine.

Note: the second time you run it, you won't have this delay.

You are then asked to enter how long each line should be (right margin), and your printer device number (usually 4). The screen should change colour, and you can type away.

You should continue to type even if the words onscreen are split at the edge of the screen. This does not affect the printed text which, as I have said, does not split any words. Do not press the Return key when you get to the end of a line, as this skips a line. Only press Return at the end of a paragraph or when you want a line space!

If you make a mistake, use the Delete key. It deletes as normal until a certain point, depending on where you set the right margin.

When you have finished your text, press function key 1 to print it. When printed, the text remains in memory so you can go through the same procedure to print another copy.

If you don't like the text, press function key 3 to erase it all.

The program is very simple to understand if you know how to use string handling. It uses strings through the program, splitting them, using parts of them, and the like. The string functions have been taken to their limits.

The machine code or data statements print the title and line numbering on top of the screen. The short program runs on the interrupts and continues to do so until you press Run/Stop and Restore.

The listing is well annotated, but this explanation should help a lot. If you are an expert, you can expand the program to become a full word-processor. *Super Typewriter* does not use sprites, sound or graphics so can be converted easily to other machines. The only problem is the machine code; this can be omitted but the display won't be as good.

variables used

TX$(n)	this string contains your text; n represents how many printer lines you have used
TX	length of line so far
LI	lines used
N	number
M	machine-code memory
C	Checksum for data
A	current data
CO	counter
A$	last letter in text
PR	printer device number
MD$	middle of TX$(n)

program listing

```
1 REM
2 REM
3 REM
4 REM
5 REM
6 REM
7 :
8 :
9 REM
10 :
11 IFPEEK(49152)=120THEN55
12 IFPEEK(49153)=169THEN55
13 REM
14 :
15 POKE53281,0:POKE53280,0
16 REM
17 PRINT"
"TAB(13)"BE PATIENT PLEASE"
18 M=49152:C=0
19 READA:IFA=-1THEN22
20 POKEM,A:M=M+1:C=C+A:GOTO19
21 REM
22 IFC<>18035THENPRINT"
DATA ERROR IN MACHINE-CODE....  CHECK!!!":END
23 REM
24 :
25 DATA  120, 169, 13, 141, 20, 3, 169, 192, 141, 21
26 DATA  3, 88, 96, 162, 0, 189, 72, 192, 32, 68
```

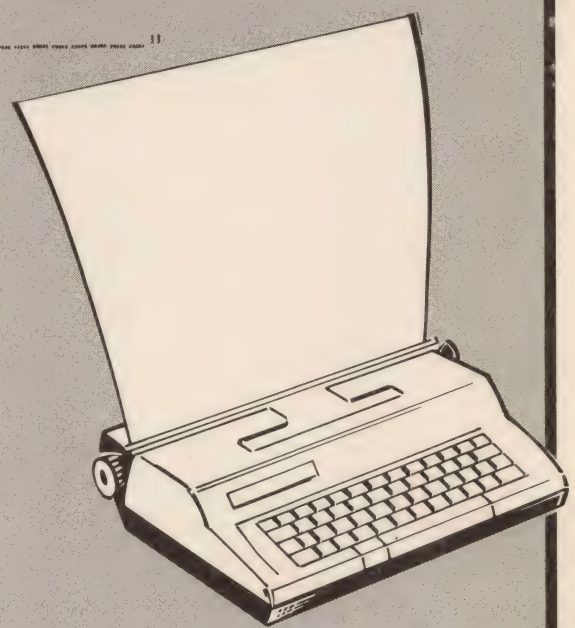
program listing

```
27 DATA  192, 157, 0, 4, 189, 112, 192, 32, 68, 192
28 DATA  157, 40, 4, 189, 152, 192, 32, 68, 192, 157
29 DATA  80, 4, 189, 192, 192, 32, 68, 192, 157, 120
30 DATA  4, 189, 232, 192, 32, 68, 192, 157, 160, 4
31 DATA  232, 224, 40, 208, 206, 76, 49, 234, 24, 105
32 DATA  128, 96, 32, 32, 32, 32, 32, 32, 32, 32
33 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
34 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
35 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
36 DATA  32, 32; 32, 32, 32, 32, 83, 85, 80, 69
37 DATA  82, 45, 84, 89, 80, 69, 87, 82, 73, 84
38 DATA  69, 82, 32, 66, 89, 32, 87, 73, 76, 76
39 DATA  73, 65, 77, 32, 70, 79, 78, 71, 32, 32
40 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
41 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
42 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
43 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32
44 DATA  32, 32, 45, 45, 45, 45, 45, 45, 45, 45
45 DATA  49, 48, 45, 45, 45, 45, 45, 45, 45, 45
46 DATA  50, 48, 45, 45, 45, 45, 45, 45, 45, 45
47 DATA  51, 48, 45, 45, 45, 45, 45, 45, 45, 45
48 DATA  52, 48, 62, 43, 43, 43, 43, 43, 43, 43
49 DATA  43, 43, 43, 43, 43, 43, 43, 43, 43, 43
50 DATA  43, 43, 43, 43, 43, 43, 43, 43, 43, 43
51 DATA  43, 43, 43, 43, 43, 43, 43, 43, 43, 43
52 DATA  43, 60, 0, 0, 255, 255, 0, 0, 255, 255, -1
53 REM
54 REM
55 DIMTX$(999):REM
56 RM=65       :REM
57 CO=-1       :REM
58 LI=1        :REM
59 REM
60 :
61 POKE54296,15:POKE54277,1
62 POKE56325,255
63 PRINTCHR$(8)
64 POKE650,255
65 PRINTCHR$(14)
66 SYS49152
67 REM
68 :
69 POKE53281,11:POKE53280,11
70 REM
71 PRINT"      "TAB(12)"_ORMAT "ROCEDUCES"
72 REM
73 PRINTTAB(12)"
74 REM
75 INPUT"      _IGHT MARGIN     :   65    ";RM
76 REM
77 INPUT"      "RINTER DEVICE #  :   4     ";PR
78 REM
79 PRINT"      RESS [Y] KEY WHEN YOU ARE READY TO TYPE"
```



program listing

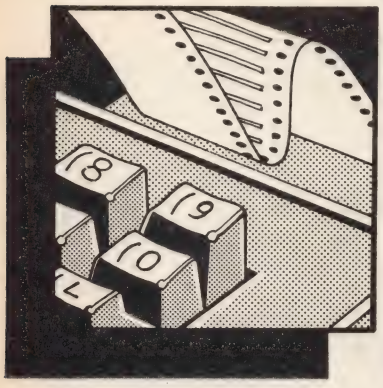
```
80 GETA$:IFA$<>"Y"THEN80
81 POKE53281,6:POKE53280,6
82 REM:
83 :
84 PRINTCHR$(14)
85 REM
86 PRINT"
87 REM
88 PRINT"  RINTER DEVICE = "PR"     [_1]  RINT TEXT"
89 REM
90 PRINT"IGHT MARGIN    = "RM"    [_2]  RASE ALL"
91 REM
92 PRINT"
93 REM:
94 REM:
95 :
96 REM
97 PRINT"  ";
98 GETA$:IFA$=""THEN98
99 REM
100 PRINT"  ";
101 POKE54276,0:POKE54276,33
102 POKE54273,100:POKE54272,100
103 REM:
104 :
105 REM
106 IFA$=""ORA$=""ORA$="@"ORA$=""THEN97
107 REM:
108 IFA$="S"ORA$=""THEN97
109 IFA$<>CHR$(13)THEN114:REM:
110 REM
111 PRINT"  "A$:TX$(LI)=TX$(LI)+A$+"@":LI=LI+1L::REM:
112 GOTO97
113 REM
114 IFA$=""THENGOSUB147:REM:
115 REM
116 IFA$=""THENGOSUB171:REM:
117 TX=LEN(TX$(LI)):REM:
118 REM:
119 :
120 IFA$<>CHR$(20)THEN127
121 IFTX<1THEN97
122 TX=TX-1
123 TX$(LI)=LEFT$(TX$(LI),TX)
124 REM
125 PRINT"  "A$;
126 GOTO97
127 :
128 REM:
129 :
130 IFTX<RM+1THEN142
131 IFRIGHT$(TX$(LI),1)=" "THEN139
132 FORN=TX+1TO1STEP-1:CO=CO+1
```



program listing

```
133 MD$=MID$(TX$(LI),N,1)
134 IFMD$=" "THENN=1:GOTO142
135 NEXT
136 TX$(LI+1)=RIGHT$(TX$(LI),CO)
137 TX$(LI)=LEFT$(TX$(LI),TX-CO)
138 CO=-1
139 LI=LI+1
140 REM[redacted]
141 :
142 TX$(LI)=TX$(LI)+A$
143 PRINTA$;
144 GOTO97
145 REM[redacted]
146 :
147 POKE53281,0:POKE53280,0
148 REM[redacted]
149 PRINT"[redacted]"
150 REM[redacted]
151 PRINT"[redacted]  ARE YOU SURE YOU WANT TO PRINT [Y/N]"
152 GETA$:IFA$=""THEN152
153 IFA$<>"Y"THEN161
154 OPEN4,PR,7
155 FORN=1TOLI
156 PRINT#4,TX$(N)
157 NEXT
158 CLOSE4
159 REM[redacted]
160 :
161 POKE53281,6:POKE53280,6
162 REM[redacted]
163 PRINT"[redacted]"
164 FORN=1TOLI
165 PRINTTX$(N);
166 NEXT
167 A$=""
168 RETURN
169 :
170 REM[redacted]
171 POKE53281,0:POKE53280,0
172 REM[redacted]
173 PRINT"[redacted]"
174 REM[redacted]
175 PRINT"[redacted]ARE YOU SURE YOU WANT TO EARSE ALL TEXT?"
176 REM[redacted]
177 PRINT"[redacted]"TAB(15)"[redacted]RESS [Y/N]"
178 GETA$:IFA$=""THEN178
179 IFA$<>"Y"THEN161
180 FORN=1TOLI
181 TX$(N)=""
182 NEXT
183 LI=1
184 GOTO161
READY.
```





Home accounts

We can't promise you'll stay in the black, but Derrick Hancock's home budget program will certainly help you to keep on the right side of your bank manager!

ORIC 48K

This program allows you to keep control of your accounts, on a month-to-month basis. You can estimate your costs over the next few months, and you can do an annual budget. The program listing shows only one month, so you must duplicate the other 11 and the budget.

The first five items are credit terms and will be added together. The rest are debit, and will be subtracted.

After loading the data, the screen displays the months and the saving option. Pressing the appropriate key displays the required month and four commands on the top status line:

I returns you to the index
E allows you to change items held in data
P prints the account as a hard copy
S saves the screen to tape

When you are in EDIT, the cursor moves quickly and you must type RUN to re-enter the program. Before switching off the computer, you must SAVE the program, which takes about 20 seconds. You could add lines to verify the saving but I've never had any trouble.

Note: the program prints out only items which have a value of more than zero pence, and a value (including zero) must be entered against an item.

Happy accounting!

how it runs

Line	Effect	
150	subroutine —	clear status line
160	subroutine —	dimension arrays read in data
1020-1170		index
1190-1340		reads chosen month into Q$ variable
2000	subroutine —	status line instructions
2010-2170		prints account to screen
2510-2680		prints account to printer
3500-3620		edit lines
3800-4020		status line instructions
4200-4380		tab calculations and rounding procedures
4500-4770		screen-saving routine
5000-5115		23 monthly items

Repeat 11 more months plus budget.

NB: Ensure line numbers correspond with lines 3500-3620.

variables used

Q$(n)	holds name of entries — printed to screen
Q(n)	cash totals of items — printed to screen
A1$	name of month
RO	rounded totals
AZ,ZA	accumulated totals
TA-TB	calculated tab positions
Z$	status instructions
A$-L$	monthly data names
A-L	monthly data cash amounts
M$	budget names
M	budget cash amounts

hints on conversion

The program runs in Basic and so should be easy to convert to other computers. However, attention must be given to screen layout. The DOKE instruction in line 110 speeds up the cursor, and the POKE instruction turns off the sound and cursor.

Paper and Ink are the Oric instructions for background and foreground colours.

```
   AUG85              Entry          Total
                      -----          -----
                                     289.34
Brought Forward
SALARY                539             828.34
MORTGAGE               93.3           735.04
RATES                  36.43          698.61
HALIFAX                55             643.61
DEPOSIT                14             629.61
                                      --------


   SEPT85             Entry          Total
                      -----          -----
                                     629.61
Brought Forward
SALARY                539             1168.61
MORTGAGE               93.3           1075.31
RATES                  36.43          1038.88
HALIFAX                55              983.88
DEPOSIT                14              969.88
                                      --------


   OCT85              Entry          Total
                      -----          -----
                                     969.88
BROUGHT FORWARD
SALARY                539             1508.88
MORTGAGE               93.3           1415.58
RATES                  36.43          1379.15
HALIFAX                55             1324.15
DEPOSIT                14             1310.15
MOTOR INSURANCE        90             1220.15
                                      --------
```

Screen dump from Home Accounts

program listing

```
50 REM ****** Home Accounts ******
60 REM
70 REM *****                  ******
80 REM
85 REM ********** by ***********
90 REM
95 REM **** Derrick Hancock ****
100 CLS
110 DOKE#24E,#0108:POKE#26A,10
120 PAPER0:INK7
130 PLOT13,8,"HOME ACCOUNTS"
140 PLOT10,11,"By Derrick Hancock"
144 REM
145 REM *** CLEAR STATUS LINE ***
146 REM
150 GOSUB 3860
154 REM
155 REM *** INITIALISE DATA   ***
156 REM
160 GOSUB4980
997 REM
998 REM *** INDEX ***
999 REM
1000 CLS
1010 PRINTCHR$(4)
1020 PRINTSPC(10);CHR$(27);"J Home
     accounts.":PRINTCHR$(4)
1030 PRINT:PRINTSPC(2)"A, April."
1040 PRINTSPC(2)"B, May."
1050 PRINTSPC(2)"C, June."
1060 PRINTSPC(2)"D, July.NEXT
1070 PRINTSPC(2)"E, August."
1080 PRINTSPC(2)"F, September."
1090 PRINTSPC(2)"G, October."
1100 PRINTSPC(2)"H, November."
1110 PRINTSPC(2)"I, December."
1120 PRINTSPC(2)"J, January."
1130 PRINTSPC(2)"K, F"
1140 PRINTSPC(2)"L, March."
1150 PRINTSPC(2)"M, Annual Budget."
1160 PRINTSPC(2)"N, Save Programme."
1170 PRINTSPC(2)"Q, End Programme."
1174 REM
1175 REM *** MENU CHOICE ***
1176 REM
1180 GETO$
1190 IFO$<"A"ORO$>"Q"THENGOTO1180
```

program listing

```
1200 IFO$="A"THENA1$="APR85":FORN=1TO23:Q$(N)=A$(N):Q(N)=A(N):NEXTN:GOTO2000
1210 IFO$="B"THENA1$="MAY85":FORN=1TO23:Q$(N)=B$(N):Q(N)=B(N):NEXTN:GOTO2000
1220 IFO$="C"THENA1$="JUNE85":FORN=1TO23:Q$(N)=C$(N):Q(N)=C(N):NEXT:GOTO2000
1230 IFO$="D"THENA1$="JULY85":FORN=1TO23:Q$(N)=D$(N):Q(N)=D(N):NEXT:GOTO2000
1240 IFO$="E"THENA1$="AUG85":FORN=1TO23:Q$(N)=E$(N):Q(N)=E(N):NEXT:GOTO2000
1250 IFO$="F"THENA1$="SEPT85":FORN=1TO23:Q$(N)=F$(N):Q(N)=F(N):NEXT:GOTO2000
1260 IFO$="G"THENA1$="OCT85":FORN=1TO23:Q$(N)=G$(N):Q(N)=G(N):NEXT:GOTO2000
1270 IFO$="H"THENA1$="NOV85":FORN=1TO23:Q$(N)=H$(N)RETURNQ(N)=H(N):NEXT:GOTO200
1280 IFO$="I"THENA1$="DEC85":FORN=1TO23:Q$(N)=I$(N):Q(N)=I(N):NEXT:GOTO2000
1290 IFO$="J"THENA1$="JAN86":FORN=1TO23:Q$(N)=J$(N):Q(N)=J(N):NEXT:GOTO2000
1300 IFO$="K"THENA1$="FEB86":FORN=1TO23:Q$(N)=K$(N):Q(N)=K(N):NEXT:GOTO2000
1310 IFO$="L"THENA1$="MAR86":FORN=1TO23:Q$(N)=L$(N):Q(N)=L(N):NEXT:GOTO2000
1320 IFO$="M"THENA1$="BUGET":FORN=1TO23:Q$(N)=M$(N):Q(N)=M(N):NEXT:GOTO2200
1330 IFO$="N"THENCLS:GOTO4500
1340 IFO$="Q"THENGOSUB 3860:CLS:PRINTCHR$(17):PAPER7:INK0:END
1997 REM
1998 REM *** MAIN PROGRAM ***
1999 REM
2000 S=1:CLS:GOSUB 3800
2010 PRINTTAB(7)A1$TAB(22)"Entry"TAB(32)"Total"
2020 PRINTTAB(22)"-----"TAB(31)"------"
2030 AZ=Q(1):GOSUB 4200
2040 PRINTTAB(3)Q$(1);TAB(TA)RO
2050 GOSUB4310:IFQ(2)=0THENGOTO2070
2060 AZ=Q(1)+Q(2):GOSUB 4200:PRINTTAB(3)Q$(2);TAB(TB)ZATAB(TA)RO
2070 FORN=3TO7:GOSUB4310
2080 IFQ(N)=0THENGOTO2100
2090 AZ=AZ+Q(N):GOSUB 4200:PRINTTAB(3)Q$(N);TAB(TB)ZATAB(TA)RO
2100 NEXTN
2110 FORN=8TO23:GOSUB4310
2120 IFQ(N)=0THENGOTO2140
2130 AZ=AZ-Q(N):GOSUB 4200:PRINTTAB(3)Q$(N);TAB(TB)ZATAB(TA)RO
2140 NEXTN
2150 PRINTTAB(31)"---------"
2160 S=0
2170 GOSUB 3920
2180 GOTO1000
2200 S=1:CLS:GOSUB 3800
2210 PRINTSPC(11)A1$:PRINTSPC(11)"------"
2220 AZ=Q(1):GOSUB 4200
2230 PRINTTAB(3)Q$(1);TAB(TA)RO
2240 FORN=2TO20:GOSUB4310
2250 GOTO2130
2497 REM
2498 REM *** PRINTER ROUTINE ***
2499 REM
2500 S=1:     GOSUB 3800
2510 LPRINTTAB(20)A1$TAB(35)"Entry"TAB(48)"Total"
2520 LPRINTTAB(36)"-----"TAB(49)"------"
2530 AZ=Q(1):GOSUB 4200
2540 LPRINT'1RUN)Q$(1):'TA+16)RO
2550 GOSUB4310:IFQ(2)=0THENGOTO2570
2560 AZ=Q(1)+Q(2):GOSUB 4200:LPRINTTAB(16)Q$(2);TAB(TB+13)ZATAB(TA+16)RO
2570 FORN=3TO7:GOSUB4310
2580 IFQ(N)=0THENGOTO2600
2590 AZ=AZ+Q(N):GOSUB 4200:LPRINTTAB(16)Q$(N);TAB(TB+13)ZATAB(TA+16)RO
2600 NEXTN
2610 FORN=8TO23:GOSUB4310
2620 IFQ(N)=0THENGOTO2640
2630 AZ=AZ-Q(N):GOSUB 4200:LPRINTTAB(16)Q$(N);TAB(TB+13)ZATAB(TA+16)RO
2640 NEXTN
2650 LPRINTTAB(47)"---------"
```

program listing

```
2660 GOSUB3860
2680 GOTO1000
2700 S=1:GOSUB 3800
2710 LPRINTSPC(27)A1$:LPRINTSPC(27)
     "_____"
2720 AZ=Q(1):GOSUB 4200
2730 LPRINTTAB(16)Q$(1);TAB(TA+16)RO
2740 FORN=2TO20:GOSUB4310
2750 GOTO2630
2760 GOSUB3860
2770 GOTO1000
3497 REM
3498 REM *** NEW ENTRY ***
3499 REM
3500 IFO$="A"THEN:CLS:LIST5005-5115
3510 IFO$="B"THEN:CLS:LIST5125-5235
3520 IFO$="C"THEN:CLS:LIST5245-5355
3530 IFO$="D"THEN:CLS:LIST5365-5475
3540 IFO$="E"THEN:CLS:LIST5485-5595
3550 IFO$="F"THEN:CLS:LIST5605-5715
3560 IFO$="G"THEN:CLS:LIST5725-5835
3570 IFO$="H"THEN:CLS:LIST5850-5960
3580 IFO$="I"THEN:CLS:LIST5970-6080
3590 IFO$="J"THEN:CLS:LIST6090-6200
3600 IFO$="K"THEN:CLS:LIST6210-6320
3610 IFO$="L"THEN:CLS:LIST6335-6440
3620 IFO$="M"THEN:CLS:LIST6450-6545
3797 REM
3798 REM *** STATUS LINE ***
3799 REM
3800 Z$="I.Index E.Entry P.Printer
     S.Save screen"
3810 B=48000
3820 FORX=1TOLEN(Z$)
3830 POKEB+X,ASC(MID$(Z$,X,1))
3840 NEXTX
3850 RETURN
3857 REM
3858 REM *** CLEAR STATUS LINE ***
3859 REM
3860 Z$="
     "
3870 B=48000
3880 FORX=1TOLEN(Z$)
3890 POKEB+X,ASC(MID$(Z$,X,1))
3900 NEXTX
3910 RETURN
3917 REM
3918 REM ***     ***
3919 REM
3920 REPEAT:GETK$
3930 UNTILK$="I"ORK$="E"ORK$="S"
     ORK$="P"
3940 IFK$="I"THENGOSUB 3860:GOTO1000
3950 IFK$="E"THENPRINTCHR$(17):GOSUB
     3970:GOTO3500
3960 IFK$="S"THENGOSUB 4750:RETURN
3964 IFK$="P"ANDO$="M"THENGOTO2700
3965 IFK$="P"THENGOTO2500
3967 REM
3968 REM *** STATUS LINE ***
3969 REM
3970 Z$="   Use curser to EDIT.Then
     type RUN    "
3980 B=48000
3990 FORX=1TOLEN(Z$)
4000 POKEB+X,ASC(MID$(Z$,X,1))
4010 NEXTX
4020 RETURN
4197 REM
4198 REM *** TAB (TA) ***
4199 REM
4200 RO=(INT(AZ*10^2+0.5))/10^2
4210 IFAZ>0.01ANDAZ<1THENTA=34
4220 IFAZ>0.99ANDAZ<10THENTA=33
4230 IFAZ>9.99ANDAZ<100THENTA=32
4240 IFAZ>99.99ANDAZ<1000THENTA=31
4250 IFAZ>999.99ANDAZ<10000THENTA=30
4260 IFAZ>9999.99ANDAZ<100000THENTA=29
4270 IFAZ<-9ANDAZ>-100THENTA=32
4280 IFAZ<-99ANDAZ>-1000THENTA=31
4290 IFAZ<-999ANDAZ>-10000THENTA=30
4300 RETURN
4307 REM
4308 REM *** TAB (TB) ***
4309 REM
4310 S=S+1
4320 ZA=(INT(Q(S)*10^2+0.5))/10^2
4330 IFZA<10THENTB=23
4340 IFZA>9.99ANDZA<100THENTB=22
4350 IFZA>99.99ANDZA<1000THENTB=21
4360 IFZA>999.99ANDZA<10000THENTB=20
4370 IFZA>9999.99ANDZA<100000THENTB=19
4380 IFZA<0.99THENTB=24
4390 RETURN
4497 REM
4498 REM *** SAVE PROGRAM ***
4499 REM
4500 CLS
4510 PRINT:PRINT:PRINTSPC(8)"TO SAVE
     PROGRAMME"
4520 PRINT:PRINT" 1 Ensure tape is at
     correct position."
4530 PRINT:PRINT" 2 Press RECORD and
     PLAYBACK buttons"
4540 PRINT"   on the recorder."
4550 PRINT:PRINT" 3 When you are ready
     press 'R'"
4560 PRINT:PRINT" 4 The 'I' KEY will
     return you"
4570 PRINT"   to the Index."
4580 REPEAT:GETK$:UNTILK$="R"ORK$="I"
4590 IFK$="I"THENCLS:GOTO1000
4650 CLS
4660 CSAVE"Home Accounts",AUTO
4670 CLS
4680 PLOT6,12,"Saving now complete"
4700 PLOT6,14,"Switch off the
     recorder."
4720 PLOT6,16,"Press 'I' for the
     Index."
4730 REPEAT:GETK$:UNTILK$="I"
4740 IFK$="I"THENCLS:GOTO1000
4750 GOSUB3860
```

program listing

```
4757 REM
4758 REM *** SAVE SCREEN ***
4759 REM
4760 CSAVEA1$,A48040,E49119
4770 RETURN
4980 DIMA$(30),B$(30),C$(30),D$(30),
     E$(30),F$(30),G$(30)
4985 DIMH$(30),I$(30),J$(30),K$(30),
     L$(30),M$(30),Q$(30)
4990 DIMA(30),B(30),C(30),D(30),E(30),
     F(30),G(30)
4995 DIMH(30),I(30),J(30),K(30),L(30),
     M(30),Q(30)
5000 FORN=1TO23:READA$(N),A(N):NEXTN
5005 DATACARRIED FORWARD,100
5010 DATACREDIT,100
5015 DATACREDIT,70
5020 DATACREDIT,5
5025 DATACREDIT,50
5030 DATACREDIT,10
5035 DATACREDIT,0
5040 DATADEBIT,0.09
5045 DATADEBIT,56.745
5050 DATADEBIT,0.2
5055 DATADEBIT,300
5060 DATADEBIT,0
5065 DATADEBIT,0
5070 DATADEBIT,5
5075 DATADEBIT,0
5080 DATADEBIT,0
5085 DATADEBIT,0
5090 DATADEBIT,0
5095 DATADEBIT,0
5100 DATADEBIT,0
5105 DATADEBIT,500
5110 DATADEBIT,600
5115 DATADEBIT,500.87
5117 REM *** REPEAT FOR OTHER ***
5118 REM *** 11 MONTHS + BUGET ***
5119 REM *** N-B 5 BETWEEN LINES ***
5120 FORN=1TO23:READB$(N),B(N):NEXTN
5125 DATACARRIED FORWARD,100
5130 DATACREDIT,500
5135 DATACREDIT,100
5140 DATACREDIT,50
5145 DATACREDIT,50
5150 DATACREDIT,100
5155 DATACREDIT,100
5160 DATADEBIT,56.25
5165 DATADEBIT,56.25
5170 DATADEBIT,56.25
5175 DATADEBIT,56.25
5180 DATADEBIT,56.25
5185 DATADEBIT,56.25
5190 DATADEBIT,123
5195 DATADEBIT,56.25
5200 DATADEBIT,56.25
5205 DATADEBIT,56.25
5210 DATADEBIT,56.25
5215 DATADEBIT,56.25
5220 DATADEBIT,56.25
5225 DATADEBIT,56.25
5230 DATADEBIT,56.25
5235 DATADEBIT,56.25
5240 FORN=1TO23:READC$(N),C(N):NEXTN
5245 DATACARRIED FORWARD,100
5250 DATACREDIT,500
5255 DATACREDIT,100
5260 DATACREDIT,50
5265 DATACREDIT,50
5270 DATACREDIT,100
5275 DATACREDIT,100
5280 DATADEBIT,56.25
5285 DATADEBIT,56.25
5290 DATADEBIT,56.25
5295 DATADEBIT,56.25
5300 DATADEBIT,56.25
5305 DATADEBIT,56.25
5310 DATADEBIT,56.25
5315 DATADEBIT,56.25
5320 DATADEBIT,56.25
5325 DATADEBIT,56.25
5330 DATADEBIT,56.25
5335 DATADEBIT,56.25
5340 DATADEBIT,56.25
5345 DATADEBIT,56.25
5350 DATADEBIT,56.25
5355 DATADEBIT,56.25
5360 FORN=1TO23:READD$(N),D(N):NEXTN
5365 DATACARRIED FORWARD,100
5370 DATACREDIT,500
5375 DATACREDIT,100
5380 DATACREDIT,100
5385 DATACREDIT,0
5390 DATACREDIT,100
5395 DATACREDIT,100
5400 DATADEBIT,56.25
5405 DATADEBIT,56.25
5410 DATADEBIT,56.25
5415 DATADEBIT,56.25
5420 DATADEBIT,56.25
5425 DATADEBIT,56.25
5430 DATADEBIT,56.25
5435 DATADEBIT,56.25
5440 DATADEBIT,56.25
5445 DATADEBIT,56.25
5450 DATADEBIT,56.25
5455 DATADEBIT,56.25
5460 DATADEBIT,0
5465 DATADEBIT,0
5470 DATADEBIT,56.25
5475 DATADEBIT,56.25
5480 FORN=1TO23:READE$(N),E(N):NEXTN
5485 DATACARRIED FORWARD,100
5490 DATACREDIT,500
5495 DATACREDIT,100
5500 DATACREDIT,50
5505 DATACREDIT,50
5510 DATACREDIT,100
5515 DATACREDIT,100
5520 DATADEBIT,56.25
5525 DATADEBIT,56.25
5530 DATADEBIT,56.25
5535 DATADEBIT,56.25
```

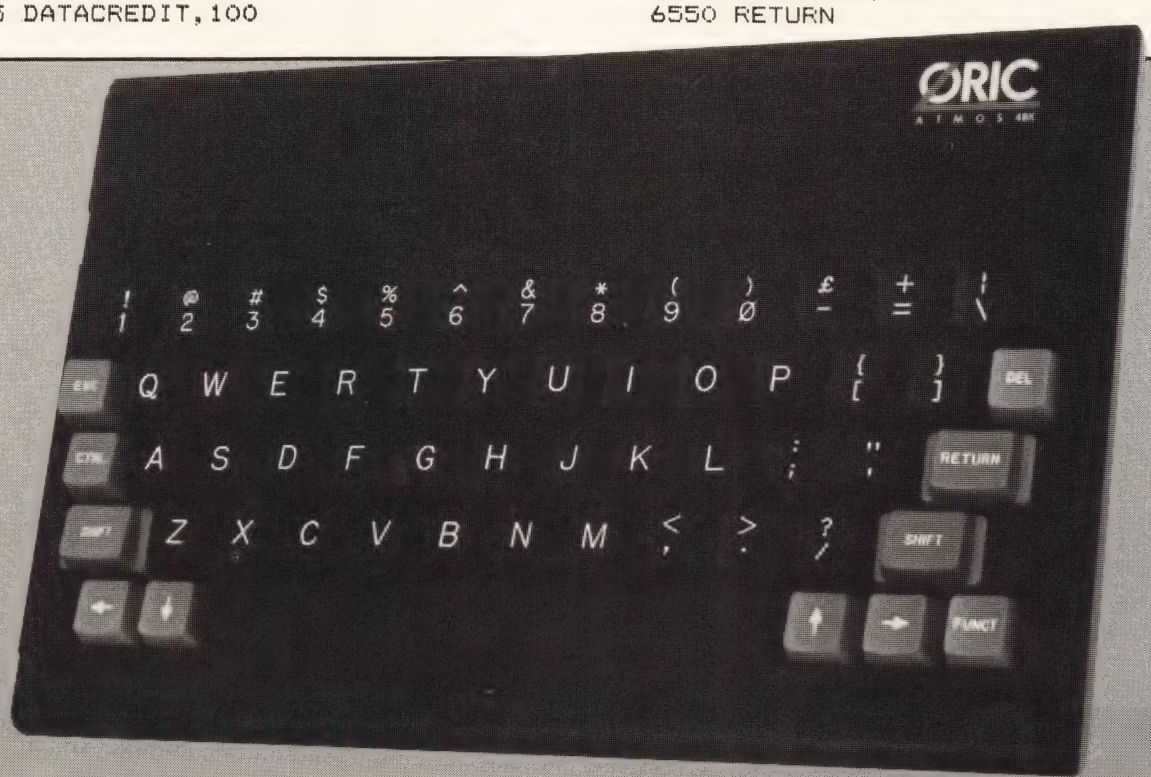
program listing

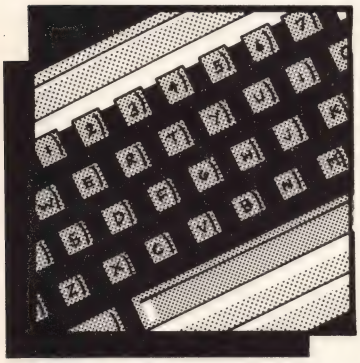
```
5540 DATADEBIT,56.25
5545 DATADEBIT,56.25
5550 DATADEBIT,56.25
5555 DATADEBIT,56.25
5560 DATADEBIT,56.25
5565 DATADEBIT,56.25
5570 DATADEBIT,56.25
5575 DATADEBIT,56.25
5580 DATADEBIT,56.25
5585 DATADEBIT,56.25
5590 DATADEBIT,56.25
5595 DATADEBIT,56.25
5600 FORN=1TO23:READF$(N),F(N):NEXTN
5605 DATACARRIED FORWARD,100
5610 DATACREDIT,500
5615 DATACREDIT,100
5620 DATACREDIT,50
5625 DATACREDIT,50
5630 DATACREDIT,100
5635 DATACREDIT,100
5640 DATADEBIT,56.25
5645 DATADEBIT,56.25
5650 DATADEBIT,56.25
5655 DATADEBIT,56.25
5660 DATADEBIT,56.25
5665 DATADEBIT,56.25
5670 DATADEBIT,56.25
5675 DATADEBIT,56.25
5680 DATADEBIT,56.25
5685 DATADEBIT,56.25
5690 DATADEBIT,56.25
5695 DATADEBIT,56.25
5700 DATADEBIT,56.25
5705 DATADEBIT,56.25
5710 DATADEBIT,56.25
5715 DATADEBIT,56.25
5720 FORN=1TO23:READG$(N),G(N):NEXTN
5725 DATACARRIED FORWARD,100
5730 DATACREDIT,500
5735 DATACREDIT,100
5740 DATACREDIT,50
5745 DATACREDIT,50
5750 DATACREDIT,100
5755 DATACREDIT,100
5760 DATADEBIT,56.25
5765 DATADEBIT,56.25
5770 DATADEBIT,56.25
5775 DATADEBIT,56.25
5780 DATADEBIT,56.25
5785 DATADEBIT,56.25
5790 DATADEBIT,56.25
5795 DATADEBIT,56.25
5800 DATADEBIT,56.25
5805 DATADEBIT,56.25
5810 DATADEBIT,56.25
5815 DATADEBIT,56.25
5820 DATADEBIT,56.25
5825 DATADEBIT,56.25
5830 DATADEBIT,56.25
5835 DATADEBIT,56.25
5840 FORN=1TO23:READH$(N),H(N):NEXTN
5850 DATACARRIED FORWARD,100
5855 DATACREDIT,500
5860 DATACREDIT,100
5865 DATACREDIT,50
5870 DATACREDIT,50
5875 DATACREDIT,100
5880 DATACREDIT,100
5885 DATADEBIT,56.25
5890 DATADEBIT,56.25
5895 DATADEBIT,56.25
5900 DATADEBIT,56.25
5905 DATADEBIT,56.25
5910 DATADEBIT,56.25
5915 DATADEBIT,56.25
5920 DATADEBIT,56.25
5925 DATADEBIT,56.25
5930 DATADEBIT,56.25
5935 DATADEBIT,56.25
5940 DATADEBIT,56.25
5945 DATADEBIT,56.25
5950 DATADEBIT,56.25
5955 DATADEBIT,56.25
5960 DATADEBIT,56.25
5965 FORN=1TO23:READI$(N),I(N):NEXTN
5970 DATACARRIED FORWARD,100
5975 DATACREDIT,0
5980 DATACREDIT,0
5985 DATACREDIT,50
5990 DATACREDIT,50
5995 DATACREDIT,100
6000 DATACREDIT,100
6005 DATADEBIT,56.25
6010 DATADEBIT,56.25
6015 DATADEBIT,56.25
6020 DATADEBIT,56.25
6025 DATADEBIT,56.25
6030 DATADEBIT,56.25
6035 DATADEBIT,56.25
6040 DATADEBIT,56.25
6045 DATADEBIT,56.25
6050 DATADEBIT,56.25
6055 DATADEBIT,56.25
6060 DATADEBIT,56.25
6065 DATADEBIT,56.25
6070 DATADEBIT,56.25
6075 DATADEBIT,56.25
6080 DATADEBIT,56.25
6085 FORN=1TO23:READJ$(N),J(N):NEXTN
6090 DATACARRIED FORWARD,100
6095 DATACREDIT,500
6100 DATACREDIT,100
6105 DATACREDIT,50
6110 DATACREDIT,50
6115 DATACREDIT,100
6120 DATACREDIT,100
6125 DATADEBIT,56.25
6130 DATADEBIT,56.25
6135 DATADEBIT,56.25
6140 DATADEBIT,56.25
6145 DATADEBIT,56.25
6150 DATADEBIT,56.25
6155 DATADEBIT,56.25
6160 DATADEBIT,56.25
```



program listing

```
6165 DATADEBIT,56.25
6170 DATADEBIT,56.25
6175 DATADEBIT,56.25
6180 DATADEBIT,56.25
6185 DATADEBIT,56.25
6190 DATADEBIT,56.25
6195 DATADEBIT,56.25
6200 DATADEBIT,56.25
6205 FORN=1TO23:READK$(N),K(N):NEXTN
6210 DATACARRIED FORWARD,100
6215 DATACREDIT,500
6220 DATACREDIT,100
6225 DATACREDIT,50
6230 DATACREDIT,50
6235 DATACREDIT,100
6240 DATACREDIT,100
6245 DATADEBIT,56.25
6250 DATADEBIT,56.25
6255 DATADEBIT,56.25
6260 DATADEBIT,56.25
6265 DATADEBIT,56.25
6270 DATADEBIT,56.25
6275 DATADEBIT,56.25
6280 DATADEBIT,56.25
6285 DATADEBIT,56.25
6290 DATADEBIT,56.25
6295 DATADEBIT,56.25
6300 DATADEBIT,56.25
6305 DATADEBIT,56.25
6310 DATADEBIT,56.25
6315 DATADEBIT,56.25
6320 DATADEBIT,56.25
6325 FORN=1TO23:READL$(N),L(N):NEXTN
6330 DATACARRIED FORWARD,100
6335 DATACREDIT,500
6340 DATACREDIT,100
6345 DATACREDIT,50
6350 DATACREDIT,50
6355 DATACREDIT,100
6360 DATACREDIT,100
6365 DATADEBIT,56.25
6370 DATADEBIT,56.25
6375 DATADEBIT,56.25
6380 DATADEBIT,56.25
6385 DATADEBIT,56.25
6390 DATADEBIT,56.25
6395 DATADEBIT,56.25
6400 DATADEBIT,56.25
6405 DATADEBIT,56.25
6410 DATADEBIT,56.25
6415 DATADEBIT,56.25
6420 DATADEBIT,56.25
6425 DATADEBIT,56.25
6430 DATADEBIT,56.25
6435 DATADEBIT,56.25
6440 DATADEBIT,56.25
6445 FORN=1TO20:READM$(N),M(N):NEXTN
6450 DATAINCOME,10000
6455 DATAITEM,1000
6460 DATAITEM,700
6465 DATAITEM,2000
6470 DATAITEM,300
6475 DATAITEM,150
6480 DATAITEM,150
6485 DATAITEM,300
6490 DATAITEM,150
6495 DATAITEM,50
6500 DATAITEM,50
6505 DATAITEM,100
6510 DATAITEM,100
6515 DATAITEM,50
6520 DATAITEM,30
6525 DATAITEN,20
6530 DATAITEM,50
6535 DATAITEM,50
6540 DATAITEM,1000
6545 DATAITEM,500
6550 RETURN
```





Single minded?

The Beeb sadly lacks the facility for single key entry of keywords. I P Wrigley solves the problem at a stroke.

BBC owners have many advantages over Sinclair owners, but the Sinclair machines and the Electron have one feature which I feel is sadly lacking on the BBC — the single-key entry of keywords. To remedy this, I have come up with my SKKS, or Single-Key Keyword System program. When this is in operation, the Beeb's Tab key finally finds a use, and acts like the Function key on the Electron. When TAB is pressed in conjunction with any of the letter keys, a complete keyword is printed up.

Loading the program: type the program in and save it as "S.K.K.S." When you want to use it, there are two methods of loading it.

Method 1: if you are not using the red function keys, and you do not require user-defined characters, type CHAIN "S.K.K.S." The program will ask you where the routine should be assembled. Answer it with &B00. The screen will clear and B00 will appear in the top left-hand corner. After a few seconds, a message will appear telling you that the system is in operation.

Type NEW, and you are ready to enter your own programs, using the SKKS. If you press Break, some 'rubbish' characters will appear. To re-enable the SKKS, press Escape then type:

```
*FX219,0
?&220=Z% AND 255
?&221=Z% DIV 256
*FX14,4
*FX14,2
```

Use method 2 if method 1 does not apply. Find the old value of Page (in hex) — ie: type PRINT PAGE. Increase PAGE by &200, ie: type PAGE=PAGE+&200. Then type NEW

CHAIN "S.K.K.S."

When the program loads, it will ask you for a location for the code to be assembled. Answer it with the old value of page+2. For example, if the old value of page was &E00, answer with &E02.

After a few seconds the program will tell you that the system is operative. Do not press Control-break or try to define the Break key (using *KEY 10). If you press Break, the SKKS will re-initialise itself.

Figure 1 shows the keywords that can be generated, and their associated keys. To print these keywords, simply press the Tab key; while it is depressed, press the correct letter key. You must not use the variable Z% in your programs. I have stuck labels on the front of my keys to see at a glance which key generates each keyword.

Figure 1

Key	Keyword
A	AUTO
B	DEF PROC
C	CHAIN"
D	DATA
E	ENDPROC
F	FOR
G	GOTO
H	GOSUB
I	INPUT
J	LOAD"
K	LOCAL
L	LIST
M	MODE
N	NEXT
O	INKEY
P	PRINT
Q	RENUMBER
R	REPEAT
S	STEP
T	THEN
U	UNTIL
V	VDU
W	RUN
X	COLOUR
Y	SAVE"
Z	RETURN

Figure 2

Key	Internal key no.		
		U	&35
		V	&63
		W	&21
A	&41	X	&42
B	&64	Y	&44
C	&52	Z	&61
D	&32		
E	&22	1	&30
F	&43	2	&31
G	&53	3	&11
H	&54	4	&12
I	&25	5	&13
J	&45	6	&34
K	&46	7	&24
L	&56	8	&15
M	&65	9	&26
N	&55	0	&27
O	&36	:	&48
P	&37	;	&57
Q	&10	@	&47
R	&33	SPACE	&62
S	&51	FULL-STOP	&67
T	&23	COMMA	&66

program listing

```
LIST
 1REM"        **********
 2REM"        *S.K.K.S.*
 3REM"        **********
 4REM"
 5REM"COPYRIGHT (C) I.P.WRIGLEY 1984
 6REM
 7REM
 8REM See instructions provided for
 9REM use of this system.
 10REM HELLO
 20PRINT"HELLO"
 30GOTO10
100CLS
110INPUT"Where do you want the routine to be
assembled",A$
120IF A$="" OR LEFT$(A$,1)<>"&" THEN 100
130Z%=EVAL(A$)
135CLS
140FOR pass=0 TO 2 STEP 2
150P%=Z%
160[
170OPT pass
180.start
190STA &F8
200PHP
210PHA
220TYA
230PHA
240TXA
250PHA
260LDA &F8
270CMP #&4
280BEQ syncevent
290.inputevent
300LDX #&60 EOR &80
310LDA #&79
320JSR &FFF4
330CPX #&E0
340BNE exit
350LDA #&7A
360JSR &FFF4
370TXA
380LDX #&00
390.check
400CMP table,X
410BEQ ok
420.missout
430INX
440LDY table,X
450CPY #&0
460BNE missout
470INX
480LDY table,X
490CPY #&0
500BNE check
510.exit
520PLA
530TAX
540PLA
550TAY
560PLA
570PLP
580RTS
590.ok
600INX
610LDA table,X
620CMP #&0
630BEQ changeflag
640STX &F8
650TAY
660LDX #&0
670LDA #&8A
680JSR &FFF4
690LDX &F8
700JMP ok
710.changeflag
720LDA #&FF
730STA &F9
740JMP exit
750.syncevent
760LDA &F9
770BEQ exit
780LDA #&0
790STA &F9
800LDA #&8A
810LDX #&0
820LDY #&7F
830JSR &FFF4
840JMP exit
850.table
860]
870NEXT pass
880W%=P%
890REPEAT
900READ A$
910IF LEFT$(A$,1)="&" THEN ?W%=EVAL(A$):W%=W%+1:GOTO900
920IF A$="0" THEN ?W%=0:W%=W%+1:GOTO 970
930FOR A=1 TO LEN A$
940?W%=ASC(MID$(A$,A,1))
950W%=W%+1
960NEXT A
970UNTIL ?(W%-1)=0 AND ?(W%-2)=0
980?&220=Z% AND 255
990?&221=Z% DIV 256
1000*FX219,0
1010*FX14,4
1020*FX14,2


1030CLS:PRINT"S.K.K. SYSTEM BY I.P.WRIGLEY IS NOW     OPERATIVE"
1040IF Z%<&E00 THEN END
1050*KEY10 !M!M!M!M*FX219,0!M?&220=Z% AND 255!M?&221=Z% DIV 256!M*FX14,4!M*FX14
,2!MPAGE=Z%+&200!MCLS:P."S.K.K. SYSTEM OPERATIVE"!M
1060DATA "&41","AUTO ",0,"&64","DEF PROC",0,"&52","CHAIN""",0,&32","DATA ",0,"&
22","ENDPROC",0,"&43","FOR",0,"&53","GOTO",0,"&54","GOSUB",0,"&25","INPUT",0
1070DATA"&45","LOAD""",0,"&46","LOCAL",0,"&56","LIST",0,"&65","MODE",0,"&55","N
EXT",0,"&36","INKEY",0,"&37","PRINT",0,"&10","RENUMBER ",0,"&33","REM",0
1080DATA"&51","STEP",0,"&23","THEN",0,"&35","UNTIL",0,"&63","VDU",0,"&21","RUN"
,0,"&42","COLOUR",0,"&44","SAVE""",0,"&61","RETURN",0,0
```

variables used

If you want to change the keywords available, or to add more keywords, you will need to alter the DATA elements. These consist of the internal key number in speech marks for the key, a comma, the keyword to be produced in speech marks, a comma, a zero, and another internal key number.

The last two items on the last Data line must be two zeros separated by a comma. See figure 2 for internal key numbers of the keys A to Z, plus some others not used in the standard program.



how it runs

Simply, the program sets up a table of data, with the keywords which are to be generated. It then intercepts the 'Character Entering Input Buffer' event, and checks to see if the Tab key and another are held down. If this is so, it checks the table and prints out the required keyword. It also sets a flag at location &F9 (unused by O.S. 1.2).

The program also intercepts the 'Start of Vertical Sync' event once every 1/50th second. This routine checks if the flag is set and, if it is, resets it and enters a delete character into the input buffer. If it did not do this, the keyword printed out would be followed by the letter which was pressed, for example PRINTP or RETURNZ.

Booklist

COMMODORE 16 EXPOSED
Authors: Richard Woolcock & Cameron Duffy
Publisher: Melbourne House, Castle Yard House, Castle Yard, Richmond
Price: £6.95

This is a general work book for the C16. It follows the usual format with a good introductory chapter on the computer system and Basic programming. Then follows the inevitable dictionary of Basic commands.

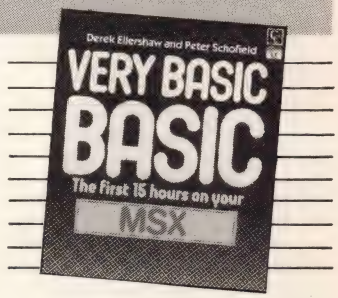
Advanced Basic techniques include some very interesting routines and utilities, and a guide to speeding up Basic programs. Sound and graphics are explained, from Basic and machine code, including sections on how the hi-res screen is bit mapped, and how to use interrupts to play music. There is also a look at C16 peripherals.

The rest of the book is more advanced, dealing with number systems, assembly language, memory organisation and typing in machine code programs and Kernal routines.

The appendices are mostly copied from the C16 manual, with a few extras including a Tedmon command summary and a colour mix chart.

This book provides a lot of useful information not found in the users' manual and is well worth reading if you want to make the most of your Commodore 16. **T.B.**

VERY BASIC BASIC
Authors: Derek Ellershaw & Peter Schofield
Publisher: Century Communications, Portland House, 12-13 Greek Street, London W1
Price: £2.95



Very Basic Basic, subtitled 'The first 15 hours on your MSX', is a well presented introduction to MSX Basic. It is intended for those who find the manual difficult to follow, and as such the style is simple, easy to follow and free from technical jargon.

The book does not attempt to tell you everything about MSX Basic, but to give a working basis from which you can move to more complicated books. In this it succeeds admirably.

The book starts by explaining how to use the keyboard and works through to simple data handling. It explains the most common Basic commands, and chapters on sound and graphics are particularly interesting. Although more complex programming is not dealt with, there are some ideas on how to adapt programs to suit your own needs.

Short programs (explained line by line) clearly illustrate the topics dealt with and exercises are set to aid your understanding. Presentation is superb: each individual point is labelled in the margin, everything is indexed and the listings are easy to read.

This is a good book for the beginner to start with before delving into something meatier. **P.S.**

MICRO MUSIC FOR THE COMMODORE 64 AND BBC COMPUTERS
Author: Gary Herman
Publisher: Papermac, Macmillan, 4 Little Essex Street, London WC2R 3LF
Price: £8.95

One of the major strengths of the Commodore and BBC is their excellent sound facilities. This has resulted in quite a large number of books being written for both machines covering the subject of sound and music making. I found it surprising that the author of *Micro Music* decided to cover both machines in one book.

Would the owner of a BBC be prepared to pay £8.95 for a book when half of it is dedicated to the Commodore? This thought was uppermost in my mind when I first looked at it.

Now I've read it, I think for this book they would. It contains not just details of using the sound facilities of each machine, but looks at the techniques of producing computer music. It starts off by discussing the history of electronics in music and the development of the synthesiser. There is an introduction to the principles of sound and acoustics before the book gets down to the nitty gritty of getting music out of your micro.

Unless you can already program in Basic the book will be of little use except to copy the many listings into your machine. Someone with a reasonable knowledge of Basic and a desire to produce good music from their BBC or Commodore will find this book excellent.



The book is easy to follow with many examples to encourage the reader to experiment. Each character covers a particular aspect of computer music ranging from simple tunes to sound effects such as ring modulation and echo. There are many music utilities included, some of which are quite sophisticated programs, such as a sequencer and sound effects generator.

The book has an excellent appendix which discusses the sound hardware of both the BBC and Commodore. There is some in-depth information on the 76489 and 6851 sound chips, and how they are programmed and interfaced. **P.R.**

book reviews

THE AMSTRAD CPC4645 DISC SYSTEM
Author: Ian Sinclair
Publisher: Collins Professional & Technical Books, 8 Grafton Street, London W1
Price: £7.95



I wasn't as impressed with this book as I was with Ian Sinclair's previous title for the Amstrad, although the technical content and style of writing are as impressive as ever.

My lack of enthusiasm stems from the fact that learning to program in Basic machine code or any language is something of a challenge, and you can get a great deal of satisfaction from the finished results. But learning to use a disk drive is like learning to use any piece of hardware — once you know the ins and outs it becomes just another piece of equipment.

The book deals with disk drives and assumes that the reader has little or no knowledge of disk operating systems. It looks at two operating systems, AMSDOS and CP/M, and gives details of how to use them.

Readers will gain most from the programs written by Ian Sinclair, such as a tape header reader, a file editor, a database and a tape to disk copier (protected Basic only), which the software houses will love! The programs are well written and well explained, and can be used without understanding how they work.

This is a good book for beginners but with a little bit of something for everyone. **I.D.**

FACTS FROM FIGURES WITH YOUR MICRO
Author: Terence Ryan
Publisher; Sigma Press, 5 Alton Road, Wilmslow, Cheshire
Price: £6.95

How much does a computer programmer earn? Which is the most expensive supermarket? How far does a car go on one gallon? Questions like these are simple enough to ask. The problem is that you get many answers, not just one. The study of statistics deals with how to get a few meaningful numbers (the 'facts' of Terence Ryan's title) from many 'figures'.

This book is a practical and comprehensive guide to statistical concepts for the micro-owner. It is based on and gives details of a menu-driven program in module form written for a Commodore 64 in Basic. Anyone who is used to struggling with magazine listings (an art in itself) should be able to use it, though I admit I did not try it on my Spectrum. The blurb does say it's been tested. You don't have to type in the whole program.

Even without typing anything in, the book is an interesting read. For example, a sample of 10 components from a machine gives 10 diameters in inches (0.11, 0.12, 0.103 . . .) all quite close to the desired diameter of 0.1. Is the machine okay? Or has it wandered off? If, like me, you were vaguely thinking back to some sort of Gauss bell curve, and hoping that means and standard deviation would see you through, you are only partly right. One module of the program will give you those, but another will also give you a t-test result for a 99% confidence prediction.

The sort of problems dealt with in the book vary from getting the data into ascending order or doing a histogram, right up to regression analysis with Durbin-Watson and von Neumann statistics.

If you are in business, and like doodling at home on the micro, then all sorts of applications will arise — how about getting a time-based trend for your own salary? **T.R.W.**

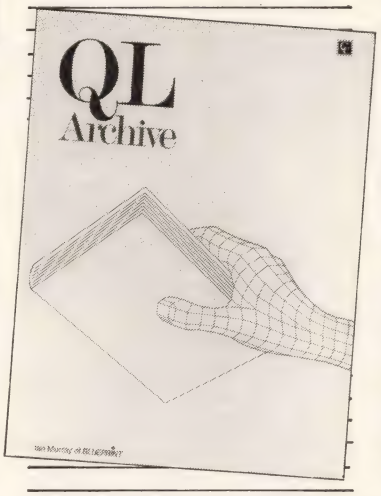
QL ARCHIVE
Author: Ian Murray
Publisher: Century Communications, Portland House, 12-13 Greek St, London W1
Price: £14.95

QL Archive follows the same style as two earlier books in the series on *Quill* and *Easel*. Those earlier books were a mere 150 pages in length. *QL Archive* more than doubles that, just to emphasise that *Archive* on the QL is far more complex to use than the other QL application software.

The layout and style of the book are very simple. A step-by-step approach is used throughout, with the left hand margin containing the keypresses required to achieve the effect outlined in the text.

Although this is a book for beginners, there is no explanation of simple file structures (files, records or field). The book launches straight into the formation of a simple file. For the beginner who might have some difficulty defining 'database', this is a rather serious omission.

From that point onwards, however, the book achieves what it sets out to do — to provide a simple explanation of a rather complex application. The book is comprehensive; all aspects of *Archive* are covered, from single-file formation to multiple-file applications and complex screen layouts of record 'cards'.



The simplified approach ('press this key, then that key . . . .') worked well in the companion books in the series on *Quill* and *Easel*. Whether the same approach is quite so successful on the more complex subject of *Archive* is doubtful. But if you have struggled with the *User Guide*, and other books on *Archive*, it might help you break through the barrier of basic understanding. **D.N.**

SOFTAID

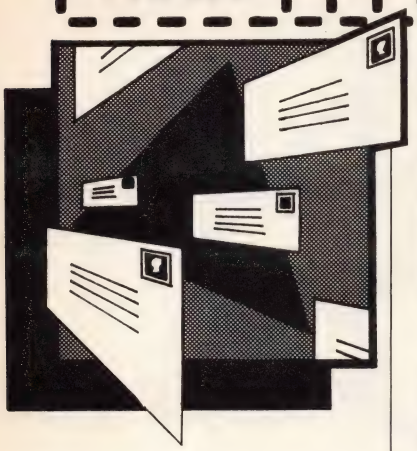
Feed The World



48K SPECTRUM

Spellbound — Beyond
Starbike — The Edge
Kokotoni Wilf — Elite
The Pyramid — Fantasy
Horace Goes Ski-ing — Melbourne House/Psion
Gilligan's Gold — Ocean
Ant Attack — Quicksilva
3D Tank Duel — Real-Time
Jack and the Beanstalk — Thor
Sorcery — Virgin
and
Feed the World
the hit single
from Band Aid

Available from all good software stores now. Price £4.99.
All proceeds to
Bob Geldof Band-Aid Ethiopian Appeal Fund



Input

Air your views or ask for help with micro problems — just write to 'Input', PCT, 1 Golden Square, London W1R 3AB.



Dear PCT
I read with interest Ray Elder's article on upgrading Spectrums. Perhaps it would be useful for the many Spectrum users to mention my past two years' experience with the Microdrives.

I started with Interface 1 and one drive placed conventionally at the side of the Spectrum using the short ribbon cable supplied. The monitor was a small Telefunken TV placed about 1ft away. Loading times varied up to a minute. I was unable to find the cause of this unpredictability, like many other Spectrum users. I have also resorted to tape backup of important files.

Last year, while in the UK, I purchased an additional drive and a DK'tronics keyboard and decided to use a 1ft ribbon cable to place the drives on top of the keyboard. The result was disastrous. Drive 2 would not load Drive 1 and gave 42K free for a newly formatted cartridge; programs and data were scrambled and lost. Eventually, I started to suspect a radiation perturbance of the Microdrives after shielding the ribbon cable without success.

Covering a raised section (where the Sinclair logo and ZX Microdrive are embossed) and the back with aluminium foil seemed to solve all previous problems. I have used the system for more than seven months and have not had a single hang-up. The system is used every day with a graphic *Stock Portfolio Management, Omnicalc, Tasword* and *+80 Finance Manager*. My conclusion is that the Microdrive is not adequately shielded and it is this factor rather than the concept which makes it unreliable.

I hope that this information will assist some of my fellow Spectrum owners using Microdrives. I shall be glad to hear comments.

Otto Molnar
Strand 7140
South Africa

Dear PCT
I saw that you printed my letter in the June edition of *PCT*. Unfortunately, I must apologise for a mistake: the changes to *Zodiac* are correct but *Flight* is wrong.

The correct changes are:
Lines 5 and 2020: delete call.
Line 8195: change CALL #E804 to CALL #E93D.
Change all PLOT 1s and 0s to PLOT 2.
Line 6200: change IF XO = <0 . . . . . to — IF XO = <1 . . . . .
Line 7030: change FOR Z = 1 . . . . . to — FOR Z = 2 . . . . .
Line 8010: as above.

John Pearson
Horsham
West Sussex

LETTERS

Dear PCT
I am the proud owner of an Amstrad CPC464. I've had no problems with it, but I'd like two extra things.

First, I have seen many computer listings whose commands are, obviously, different from Amstrad's. I don't know how to change the commands, and so cannot use the listings. I wonder if you could give me the meanings of these commands — most computer users probably have this problem.

Secondly, I cannot find much information on computers here. Would you please give my address to *PCT* readers who would like to correspond with me.

Ibrahim Sen
8.Yurt, 615/4
ODTU — Ankara
Turkey

Your first problem seems to have been solved already! The July issue of PCT had an article on how to convert BBC listings to the Amstrad micros. The August issue showed how to convert Spectrum listings, and this issue how to convert Spectrum graphics commands. This will be followed by an article on converting TI99/4a programs.

As to your second point — well, here's the address. It's now up to PCT readers.

Dear PCT
I joined the computer revolution when Mr Sinclair offered me the power to control my life, if not a certain London power station, with his hi-tech 1K ZX81 for the unheard-of price of £80. Since then I have tried a variety of micros and am currently the owner of an Amstrad CPC464 with disks and a couple of home-brewed goodies. Today the years of dreaming of PDP11s running *Space Invaders*, the sleepless nights full of LD IX,(nn)s, and the plans to hook Sizewell up to a V24 lead have all come together in the culmination of my hobby. I'm writing to your magazine.

I usually find *PCT* enjoyable and informative (the main reasons why I buy the magazine). It is well worth the three 74LS02s that it costs. Except, that is, for pages 32 to 34 of the June issue.

Has Brian Boyde-Shaw purchased an example of the wonderful MSX standard? Does he now have to justify to himself how wise he was by writing articles that prove



how inferior other machines are? If the article is written for other MSX owners, of course, the MSX standard comes complete with a built-in inferiority complex. To prospective purchasers this series presents a biased and distorted view where the author has told some naughty porky pies.

Firstly, to put two minor points straight, the Amstrad *does* support the WHILE...WEND structure. And, while I would agree that MSX graphics do not use any user Ram, I would be *most* grateful if Mr Boyde-Shaw could inform me where to obtain a video chip with all 16K video Ram onboard.

As the series is devoted to Basic, I won't mention how to obtain split-screen modes on the Atari and Amstrad — which Mr B-S points out cannot be mixed. Nor will I mention the Amstrad's totally soft, and programmable, keyboard or the Memotech's colossal memory, none of which is used as video Ram, or...

It is also worth noting that the more sophisticated the hardware of a machine is then the more complex is the software required to use the given facilities. But a high-level language can get so complex that it runs slowly. As the BBC runs a fair bit quicker than the MSX machines and has rather more things to control, it is only fair to expect that stuff like VDU commands can get a bit tortuous. Even so, Mr B-S admits that the MSX sound command is complex, so why mark down the BBC for the same thing?

Finally, I shall follow this series with interest to see what Mr Boyde-Shaw makes of Basic interrupts, 80-column text, user definable characters (all 256 of them), and the like. Don't get me wrong. Much as I like my own computer or drool over the new BBC B+, I would be the first to point out their faults (like a seven-bit centronics port aargh...). Cannot Mr Boyde-Shaw bring himself to admit that MSX is not quite the best thing since heated cathodes?

Space and editorial discretion probably preclude you from publishing this letter in your magazine, but I would hope that the points raised are brought to the attention of your readers.

Simon Gamble
Reading
Berkshire

Brian Boyde-Shaw replies:
In reply to Mr Gamble's interesting but overstated letter, I would not say that my remarks indicated that other micros are inferior to the MSX standard. My articles are not written to prove this — far be it from me to attempt any such project. And so to being biased and distorted — Mr Gamble bases his statement on very little evidence.

I meet many people in the insular world of home computing who can't remove their blinkers. MSX Basic is, in my opinion, very user friendly and goes a long way towards removing the mystique of programming for the uninitiated.

While Amstrad does not support REPEAT... UNTIL I do admit that it supports WHILE... WEND, and apologise for my small error. And I did not mean to infer that by using the words 'built in' that the Ram was actually built into the video chip, but built into the video system!

I have no intention of marking down the BBC sound facilities. But it isn't quick and easy on the BBC to produce a recognisable tune as the MSX PLAY statement does. If we are to spread the gospel and remove the mystique that surrounds using and programming a home computer, it must be quick and easy for the home market in general. We are not all sophisticated hobbyists nor aspiring academics.

Finally, I admit that all home computers have their faults, and I shall be the first to point this out regarding the MSX as the series continues.

Hacker's world

Jez looks into hacking's history, recalling two novice methods to 'discover' passwords and scan hidden data on Prestel.

Last month, I discussed the ethics of hacking, or how to have fun with your hobby without causing the disastrous havoc that gives hacking a bad name (such as deleting hospitals' patient records). In this article, we look at the common methods which have been used in the past to work out passwords.

It is very difficult to 'work out' a password. It is far easier to 'discover' it. Password-generating programs can be written very easily in Basic on any micro. These merely try logging on to a system with various incremental passwords. A program might start trying with 'AAA' and progress through 'AAB' to 'ZZZ'. This takes ages, and isn't a great help for your phone bill!

After going through all the three-letter passwords, you progress to the four-letter ones, and so on. Anyone with any sense would have a six-character, or larger, password.

The myth that passwords are based on 'Joshua' or the owner's middle name are largely out of date. People — quite rightly — are more sensible with their passwords, so pick long words or even random sets of letters. It is no longer any good trying to guess someone's password, because you could be there all day and have nothing to show for it.

One way known to be used by hackers of discovering a password is to fake the login prompt of a computer system. This often looks similar to 'Please sign on' or 'Please login'. Pressing Return a few times merely repeats the message. So the only way to proceed is to sign on, by specifying your account and password.

The sneaky idea behind this hack was to write a program which simulated the text the computer displayed just before the user logged in. The login prompt would have to be printed, and repeated every time the user pressed Return — just like the real thing. The difference was that, after the user signed on by giving the password, this nasty program would have stored this away for later reference and perusal.

I have often seen a vacant terminal at college waiting for an unsuspecting teacher to login to what they think is a disconnected terminal — but was already logged into someone else's account while pretending to be vacant. Most accounts belonging to the staff of universities and colleges have 'high privilege' levels. Once inside one of their (master) accounts, the hacker can usually create more — with realistic names and titles, of course, so that they will not be recognised in a hurry.

England's major databases are not immune to hackers, either. Bugs exist in the systems, and these can greatly help a determined hacker. One such bug existed on Prestel; it allowed you to see pages which you normally had no access to. These could include top-secret pages containing privileged accounts.

The technique for using this bug was so simple that even novices used it to great advantage. First, they logged in during the day, because Prestel runs very slowly due to overuse and lousy hardware. Once in, they roamed round as usual on their ID or even a friend's (with permission, of course!). Pressing Hash to pass to the next page may have been blocked on some pages to conceal stuff which shouldn't be seen. Problem was — and is — the system is so slow (and I mean S-L-O-W!) that it takes forever to realise the user shouldn't have seen the page.

Okay, so 'forever' is a bit of an exaggeration. Prestel takes a few extra milliseconds before it realises its mistakes. In the meantime, it can accidentally display a small portion of the next frame, usually only half a line of text. You can have another go by typing *00 (star-zero-zero) to redisplay your current frame. While redrawing, past hackers pressed Hash again and again. Eventually, jotted down (on paper) were the contents of the next frame.

Many hackers consider this a megablunder by the Prestel computer, but Prestel is quick to deny such a bug exists.

This 'look behind the frame' technique allowed hackers to see information placed on the system by an IP (information provider) which was not ready for user access (boring stuff). But occasionally they had the good fortune to stumble over information left by Prestel employees. The best pages to look under were Prestel's own service pages, like telex and mailbox menus. There was sure to be something you weren't supposed to see. It has not been unknown to see new IDs using this method, and many people have taken advantage of this gaping hole in Prestel's security.

Personal Computing Today

CLASSIFIED DEPT. (PCT) A.S.P. Ltd., 1 Golden Square, London W1.
CLASSIFIED RATES — 40p private per word. Semi display £9 per single column centimetre. Series rates on application. Please note that all classified advertisements under £20 must be prepaid. All cheques or postal orders to be made payable to A.S.P. Ltd.,

Telephone David 01-437 0699

SOFTWARE

HOME ACCOUNTS. Put your house in order! Comprehensive coverage of bank accounts, credit cards, HP, etc. Inbuilt accuracy check. Projects cashflow for any period. Available for CBM 64, VIC 20, BBC & Electron. £8.45. FREE details from Discus Software, FREEPOST, Windmill Hill, Brixham, TQ5 9BR. Tel: 080 45 55532.

ASTROLOGY

for beginners. Special started 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 most have 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 PCT)
67 Peascroft Road, Hemel Hempstead, Herts
HP3 8ER.
TEL: (0442) 51809.

"WORKSHOP" 48k Spectrum. PCD, triangles, arcs, sheet metal, rectangular/square to round. Design office and shop floor. £4 cassette. E. Morris, 18 Glebe Road, Purley, Reading RG8 8DP.

CLUB

AMSTRAD/ATMOS/ORIC 1 Club. Large selection of software available. Send SAE to: 26 Beechley Drive, Cardiff.

REPAIRS

COMPUTER REPAIRS

Commodore 64 — **£15**, Vic 20 — **£12.50**, Spectrum — **£10.50**, QL — **£30**, 1541 Disk Drives — **£20**, and all Commodore systems. Send faulty machine with note describing fault to:
TRIDENT ENTERPRISES LTD
37 LINDEN HOUSE, COMMON ROAD, LANGLEY, SLOUGH.
TEL: 0753 48785
Overseas enquiries welcome. (Above prices are exclusive of parts and VAT).

ACCESSORIES

NEW

TAPE COPIER 7

Yes it can be done! Make security BACK-UP copies of the MAJORITY of your SPECTRUM programs including HIGH SPEED loaders, and JERKY leaders with TC7. A good "clean" copy which will hopefully suit your system. Other methods have proved limited. Ideal for school use.
★ includes BAUD RATE (speed) measurer.
★ can save high speed/jerky parts in "normal" form for DRIVE transfer.
★ so many extra features we can't possibly list them here. Just try it and see. As the market leader we can't afford to let you down.
COST ONLY £6.50 (or £7.50 with MT6 — see below)

NEW

TAPE TO DRIVE TRANSFER OF RECENT PROGRAMS

Yes SPECTRUM owners now have a **LOW COST** way of transperring many of even the **LATEST** high speed/jerky programs to their drives. You will be amazed! Other methods are expensive or very limited.
Firstly you need MD1b (for m-drive) or WD1b (for Wafa/or Disc drive). They give you the tools to convert programs so that they will run on your drive. Includes ability to compe with v.long programs, can split/chop bytes in one go, "VAL" creator, REMkill, make Visible, etc, etc. **FULL** manual (highly rated by CRASH) with example transfers, (also need TC7 for latest programs). **MD1b or WD1b cost £6.99** "Manages more programs" — Your Spectrum. (TC7 also needed for latest programs).
EXTRA program to CONVENIENTLY transfer converted bytes directly to m/drive with many other features (e.g. alter program names, stop programs, etc) buy MT6 at £3. (Wafadrive version is called WAFATAPE at £4).

NEW

LERM INFORMATION SHEETS

Each sheet gives you the DETAILED instructions required to each transfer 8 of the more POPULAR programs. They cost £2 each plus SAE. Up to sheet 5 available UPDATE SERVICE: for latest version on your LERM product send old tape + large SAE to get a £2 REDUCTION. OVERSEAS: add £1 Europe, £2 others for each product. SAE for enquiries.
ALL PROGRAMS CARRY OUR MONEY BACK GUARANTEE (not updates).

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 GAMES

ATARI SOCCER Manager: 4 divisions, transfers, promotion, relegation, F.A., all European cups, save game, injuries, substitutions. Much more. 48k. Send £9.00 to R.J. Rigden, 1 The Cottage, Sidcup Place, Kent.

SOFTWARE

SUPER POWER

ROM SOFTWARE FOR THE AMSTRAD CPC 464
FULFILS THE PROMISE OF YOUR AMSTRAD
SUPERPOWER ROM-CARD
–7 Sockets for 8K or 16K eproms (external cased unit with bus extension) £39.95.
SUPERPOWER ROM-BASED SOFTWARE
Disk User's Utilities ● Mailing List/Club Membership ● Programmer's Toolbox ● Assembler/Disassembler/MC Code Monitor ● All £39.95 each.
Word Processor, Database, Spreadsheet, Graphics/Statistics – With Data Interchange. Prices to be announced. Available Aug./Sept.
Order today. Phone 0532 434006 or by mail order from MicroPower Ltd., Sheepscar House, Sheepscar Street South, Leeds LS7 1AD. P&P 95p. Access/Visa.

REPAIRS

THE COMPUTER HOSPITAL
Sick Spectrum Specialists
Any cure **£18.50** inclusive, 48k fitted **£29.50** inc., 48k kit **£23.50** inc. Composite output for most modern TVs **£7.50** inc. 10% discount for existing customers. Trade enquiries welcome.
Invicta House, Station Road, Keyingham, Hull HU12 9SZ.
TEL: 09644 3354 Hrs Ans



CLASSIFIED ADVERTISING TERMS & CONDITIONS

Our terms for new advertisers (semi-display and lineage) are strictly pro-forma payments until satisfactory reference can be taken up (excluding recognised advertising agencies). Cheques and PO's should be crossed and made payable to:
ARGUS SPECIALIST PUBICATIONS LTD.,
and send together with the advertisement to:
THE CLASSIFIED DEPT., L/H, NO: 1 GOLDEN SQUARE, LONDON, W1R 3AB.
There are no reimbursements for cancellations. Advertisements arriving too late for a particular issue will be inserted in the folowing issue unless accompanied by instructions to the contrary.
All advertising sales are subject to Government regulations concerning VAT. Advertisers are responsible for complying with the various legal requirements in force eg: The Trade Description Act, Sex Discrimination Act & The Business Advertisements (Disclosure) Order 1977.
FULL TERMS & CONDITIONS OF ADVERTISING AVAILABLE ON REQUEST.

ORIC

ORIC 1/ATMOS FROM FGC

Extracts from FGC Summer Supersaver list. All Atmos compatible other than stated.
Chopper, Land of Illusion,Don't Press Q, Xenon-1, Dambusters **£6.95**.
Lone Raider, Manic Miner, Hellion, Paras and Special OPS (now Atmos), Superfruit, Gravitor, Ghostman, Atmos ext basic **£5.95**.
Waydor (graphics adventure), Krillys, Hunchback (Oric 1), Skramble, Arena 3000, any channel 8 mysterious adv **£4.25**.
Spooky Mansion, Space Shuttle, Harrier, Scubadive, Johnny Reb, Hopper, Digger **£3.25**.
Summer Supersaver: 'Four Packs' of top games at £11 and £17.
ACCESSORIES, SERIOUS SOFTWARE, BOOKS, FOURTEEN EDUCATIONAL TITLES FROM £3.69.
Free delivery. Send Chq/PO or SAE for details of full list of bargains from 99p to:
FGC (Mail Order), 20 Meadowcroft, Euxton, Charley, Lancs PR7 6BU.
TRADE ENQUIRIES WELCOME TO: FGC DISTRIBUTION AT ABOVE ADDRESS.

ADVENTURES

AMSTRAD CPC464 ADVENTURES
Angelique: A grief Encounter..... **£6.50**
"I recommend it highly"
PCW March '85
"A festival of entertainment"
HCW March '85
Send for details of other adventures.
Chqs/PO's for despatch by return to:-
NEMESIS (PCT), 10 Carlow Road, Ringstead, Northants NN14 4DW.

TO FILL THIS SPACE PHONE DAVID ON 01-437-0699

ATARI

ATARI 400/600/800 XL XE OWNERS UTILITIES

BOUGHT A DISK DRIVE?
Stuck with programs on tape. TRANS-PAC solves it. TRANS-PAC, REV 2.
New released! Six disks with more progs, more up-dates, including multi-tape to auto-boot disk, multi-tape to menu, disk to tape, tape to tape, disk to disk. Multi-move of files comes packaged on three double-sided Memorex disks.
£25 post free
All post free. Mail order only.
For full lists send s.a.e. 24 hour despatch.
STOCKSOFT
15 Woodbrooke Road, Birmingham B30 1UE.

STATIONERY

DATALIFE BY VERBATIM
5¼ DISKETTS.
5 YEAR WARRANTY
S/Side S/Density 40 Track
£1.79 E.A.!
D/Side D/Density 40 Track
£2.22 E.A.!
Other Format on Request
Please enc. S.A.E. Tel enq welcome **01-698 7560.**
Cheque/PO payable to:

Stylus Stationers,
1 Honley Road, Catford, London SE6 2HZ. (P&P inc)

It's easy to complain about advertisements. But which ones?

Every week millions of advertisements appear in print, on posters or in the cinema.

Most of them comply with the rules contained in the British Code of Advertising Practice.

But some of them break the rules and warrant your complaints.

If you're not sure about which ones they are, however, drop us a line and we'll send you an abridged copy of the Advertising Code.

Then, if an advertisement bothers you, you'll be justified in bothering us.

The Advertising Standards Authority.
If an advertisement is wrong, we're here to put it right.

ASA Ltd, Dept 2 Brook House, Torrington Place, London WC1E 7HN

This space is donated in the interests of high standards of advertising.

CLASSIFIED COUPON

If you have something to sell now's your chance! Don't turn the page — turn to us!
Rates of charge: 40p per word per issue (minimum of 15 words).Please state classification and post to **PERSONAL COMPUTING TODAY, CLASSIFIED DEPT., 1 GOLDEN SQUARE, LONDON W.1.**

Please place my advert in **PERSONAL COMPUTING TODAY** for ..... issues commencing as soon as possible.

I am enclosing my Cheque/Postal Order/International Money Order for: (delete as necessary) £..... (Made payable to A.S.P. Ltd)

VISA OR Debit my Access/Barclaycard (Delete as necessary)

We welcome Access

ADVERTISERS PLEASE ADD 15% VAT

All classified advertisements must be paid for in advance.

Please use BLOCK CAPITALS and include post codes.

Name (Mr/Mrs/Miss/Ms) (delete accordingly) .....
Address .....
.....
.....
Signature..... **Date**.....
Daytime Tel. No. .....

Ace down the line.

Punch data down the line with Miracle Technology's Modem WS2000.

WS2000 links your computer into the world-wide telephone network — the world of PRESTEL, Telecom Gold, Telex, MICRONET, databases and bulletin boards.

Flexible, versatile, high quality WS2000 wins every point, as over 11,000 business and home users know. MICRONET recommended, PCN Best Buy, British Microcomputing Awards Finalist — WS2000 plays its rivals off the court!

Our match winner, with BT telephone lead, mains power supply and comprehensive manual, costs £129.95 ex. (£154.73 inc. VAT & UK delivery). We can also supply the necessary leads and software for most computers — an unbeatable service!

Plus Free Quarter's Micronet Subscription



Prestel is a Registered Trade Mark of British Telecommunications PLC
Micronet 800 is the Trading Style of Telemap Ltd and British Telecommunications PLC

*APPROVED for use with telecommunications systems run by British Telecommunications in accordance with the conditions in the instructions for use.

FINALIST BRITISH MICRO COMPUTING AWARDS 1985

MIRACLE TECHNOLOGY

MIRACLE TECHNOLOGY (UK) LTD ST PETERS STREET IPSWICH IP1 1XB ENGLAND
✆ (0473) 216141 6 LINES TELECOM GOLD 79: KEY 001 Telex 946240 CWEASY G 19002985



ATARI ST

520ST

POWER WITHOUT THE PRICE

THE NEW ATARI 520ST

Under the new leadership of Jack Tramiel (former boss and founder of Commodore Business Machines), Atari Corporation have marked their entry into the world of business/personal computers with a machine which leaves the competition standing. Tramiel's slogan 'Power Without the Price' has been implemented in the manufacture of the new 512K Atari 520ST colour computer which offers the user amazingly high performance at an incredibly low price. Launched as a work-station, this new system incorporates seven software packages as well as the 520ST computer with 512K RAM, mouse controller, high resolution monochrome monitor (640x400), 95 key keyboard (with 18 key numeric keypad), MIDI interface, GEM and a 500K 3½ inch disk drive, all for the package price of only £651.30 (+VAT = £749). Dubbed the 'Mac beater' and the 'Jackintosh' (after Atari's Chief, Jack Tramiel), Atari's new machine has been directly compared with the Apple Macintosh RRP £2595 (+VAT = £2985) which offers similar features and capabilities but at a much higher price. Favourably reviewed by the UK's highly critical specialist computer press, the 520ST is likely to make a great impact in this country as a sophisticated alternative to an IBM PC, APRICOT or APPLE MACINTOSH. Unlike its overpriced competitors, the Atari 520ST can be linked up to a colour monitor to unleash a choice of up to 512 colours. The addition of colour brings out the full potential of graphics packages such as GEM.

USER FRIENDLY GEM OPERATING SYSTEM

The power of the ST is harnessed and made user friendly by the new operating system 'GEM' from Digital Research. GEM stands for Graphics Environment Manager and allows a user friendly colour or B/W graphics interface which closely resembles that of the Macintosh. This similarity extends to the use of moveable resizeable windows, icons to represent objects such as disks and disk drives, and the use of pull down menus and a mouse. The advantage of all this is that the computer becomes extremely easy to use. GEM has now been implemented for the Acorn, ACT, Atari, IBM, ICL, and Olivetti. Software written for GEM on one computer should also run under GEM on another computer. This will enable the market to quickly produce a large library of standard interchangeable software.

FREE SOFTWARE AND FUTURE EXPANSION

The Atari 520ST comes supplied with seven free software packages as listed below: 1) TOS -Tramiel Operating System based on CPM 68K. 2) GEM Graphics Environment Manager by Digital Research (DR) giving a WIMP (Window, Icon, Mouse, Pull down menu) environment. 3) DR GEM Paint for creating graphics masterpieces. 4) DR GEM Write for word processing. 5) Logo learning language to enable you to write your own programs easily using turtle graphics. 6) DR Personal Basic a powerful user friendly version of the Basic programming language. 7) BOS operating system giving you access to dozens of business applications packages already available on the market. Designed with future expansion in mind, the ST also features a host of different interfaces to the outside world and an impressive list of accessories is planned. Atari will soon be releasing a 1000K (1MB) 3½ inch disk drive, and a 15MB hard disk storage system as well as a mass storage compact disk (CD) player capable of storing an entire 20 volume encyclopedia on one disk. A full range of inexpensive printers are planned including dot matrix, daisywheel and thermal colour printers. With its unbeatable graphics, speed and software at a price which is far below that of any comparable personal computer currently on the market, the ST is all set to do battle with the competition. To receive further details of the ST from Silica Shop, just fill in the coupon below with your name and address details and post it to us.

Silica Shop Price: £651.30 + £97.70 VAT = £749.00 This price includes:

- ★512K RAM
- ★B/W MONITOR
- ★MOUSE
- ★500K 3.5" DISK DRIVE
- ★GEM
- ★KEYBOARD (95 KEYS)

£749

ATARI 520ST SPECIFICATION

MEMORY
512K RAM (524,288 bytes)
16K ROM expandable to 320K
Port for addnl 128K plug-in ROM cartridges
200K TOS operating system

GRAPHICS
Individually addressable 32K bit-mapped screen with 3 screen graphics modes:
320x200 pixels in 16 colours (low resolution)
640x200 pixels in 4 colours (med resolution)
640x400 pixels in monochrome (high res)
16 shades of grey in low res mode
512 colours available in low/medium res
8 levels of each in red, green and blue

ARCHITECTURE
4 custom designed chips:
GLUE Chip - MMU Memory Mngmnt Unit
DMA Controller - Graphics Processing Unit
16/32 bit Motorola 68000 processor at 8MHz
eight 32 bit data registers
eight 32 bit address registers
16 bit data bus/24 bit address bus
7 levels of interrupts/56 instructions
14 addressing modes/5 data types

DATA STORAGE
High speed hard disk interface
Direct memory access 1.33 Mbytes per second
CD (Compact Disc) interface
Built in cartridge access
Dedicated floppy disk controller

DISK DRIVE
500K (unformatted) s/sided 3½" floppy d/drive
349K (formatted) storage capacity

SOUND AND MUSIC
Sound Generator
Frequency control from 30Hz to above audible
3 voices (channels) in wave shaping sound in addition to a noise generator
Separate frequency and volume controls
Dynamic envelope controls
ADSR (Attack, Decay, Sustain, Release)
Noise generator
MIDI interface for external music synthesizers

KEYBOARD
Separate keyboard microprocessor
Standard QWERTY typewriter styling
Ergonomic angle and height
95 keys including 10 function keys
Numeric keypad — 18 keys including ENTER
One touch cursor control keypad

MONITOR
12" screen - high res monochrome monitor
640x400 monochrome resolution

VIDEO PORTS
Display - Low Resolution - 40 columns
Med/High Res - 40/80 plus cols
Medium res RGB (Red/Green/Blue) output
High resolution monochrome (Black & White)

COMMUNICATIONS
Bidirectional centronics parallel interface for printers, or modems capable of input/output
RS232C serial modem/printer interface
VT52 Terminal Emulation Software
Maximum Baud Rate up to 19,200
High speed hard disk interface
Floppy disk controller (Western Digital)
2 joystick ports (one for 2 button mouse)
MIDI interface for external music synthesizers

GEM WIMP ENVIRONMENT
WIMP - Window Icon Mouse Pop-down menus
Two button mouse controller
Icons/Pull down menus/Windows
GEM VDI - Virtual Device Interface
GEM AES - Application Environment Services
GEM BBT - Bit Block Transfer
Real time clock & calendar

SOFTWARE
GEM environment
with user friendly Macintosh style operation
'TOS' - Tramiel Operating System
Atari's own system based on CP/M 68K with hierarchical directory & file structure plus a host of MS DOS & UNIX command structures
'BOS' - Business Operating System
to run any standard BOS business programs
GEM desktop
with GEM PAINT graphics mgmt system and GEM WRITE word processor
Personal BASIC and DR Logo
originally written by Digital Research (DR) Very much like those on other machines except for the extensive use of pull down menus, mouse control and windows

VARIOUS
Dimensions: 470mmx240mmx60mm
Replaceable external power supplies
Expansion: 3½" floppy disk drives 500K/1,000K (two drives can be connected)
3½" 15Mb hard disk
CD (compact laser disc)
Dot matrix & d/wheel prntrs (black)
Thermal dot matrix (colour)
RGB & monochrome monitors

LANGUAGES
BASIC & LOGO supplied
Many others will soon be available, including: Assembler, BCPL, C, Cobol, Compiled Basic Lisp, Modular-2 and Pascal

Note: Some of the above specifications are pre-release and may therefore be subject to change

MACINTOSH v F1e v 520ST

"Imagine a Fat Mac - the 512K Apple Macintosh - but with a bigger screen, a far bigger keyboard with numeric keypad, cursor and function keys, and colour. That gives you some idea of what the Atari 520ST is like, except for two important things. First the Atari seems faster. Second the Atari system is about one third of the price."
June 1985 - Jack Schofield - **PRACTICAL COMPUTING**

FEATURES OF BASIC SYSTEM	APPLE MACINTOSH	APRICOT F1e	ATARI 520ST
Price Includes B/W Monitor	YES	NO - extra £200	YES
Keyboard size mm (LxDxH)	330x147x50	450x167x28	470x240x60
Keyboard size ins (LxDxH)	13x5¾x2	17½x6½x1	18½x9½x2½
3½" D/Drive (Unformatted)	500K	500K	500K
3½" D/Drive (Formatted)	399K	315K	349K
WIMP (Window, Icon, Mouse ...)	Apple	ACT - Activity	GEM
Real-time Clock	YES	YES	YES
Polyphonic Sound Generator	YES	NO	YES
RS232 Serial Port	YES	YES	YES
Centronics Parallel Printer Port	NO	YES	YES
Dedicated Floppy Disk Controller	NO	YES	YES
Hard Disk DMA Interface	NO	YES	YES
Full stroke keyboard	YES	YES	YES
Number of keys on keyboard	59	92	95
Numeric Keypad	NO	YES (16 Keys)	YES (18 keys)
Cursor Control Keypad	NO	YES	YES
Function keys	NO	10	10
16-bit processor	68000	Intel 8086	68000
Processor running speed	8MHz	4.77MHz	8MHz
RAM size	512K	256K	512K
Number of graphics modes	1	4	3
Number of colours	Monochrome	16	512
Max Screen Resolution (pixels)	512 x 342	640 x 256	640 x 400
Mouse included	Single Button	NO - extra £95	Two Button
Replaceable External Power Pack	NO	NO	YES
Cartridge Socket	NO	NO	YES
Joystick Ports	NO	NO	YES (two)
MIDI Synthesiser Interface	NO	NO	YES
Monitor Size	9"	9" - extra £200	12"
RGB Video Output	NO	YES	YES

System Cost with: Mouse - Monochrome Monitor - 512K RAM - 500K Disk Drive

Price of basic system (exc VAT)	£2595+VAT	£595+VAT	£652+VAT
+ Mouse	Included	£95+VAT	Included
+ Monochrome Monitor	Included	£200+VAT	Included
+ Expansion to 512K RAM	Included	£295+VAT	Included
Price of complete system (exc VAT)	£2595+VAT	£1185+VAT	£652+VAT
PRICE rounded down including VAT	**£2,984**	**£1,362**	**£749**

PRESS COMMENT

"Atari's new corporate image as an aggressive low cost computer maker is likely to mirror that of Commodore where Mr. Tramiel established the maxim that 'Business is war'."
August 21st 1984 **FINANCIAL TIMES**

"This is the only personal computer I know of that comes with a Midi interface as standard."
Peter Bright March 1985 **PERSONAL COMPUTER WORLD**

"The (GEM) version running on the Atari 68000 machines will have the additional advantage of leaving the PC version standing."
April 6th 1985 **PERSONAL COMPUTER NEWS**

"It would seem that GEM offers the ideal operating system."
March 7th 1985 **POPULAR COMPUTING WEEKLY**

"I found it (GEM) extremely easy to use and was very impressed with the way in which it disguises the unfriendly hardware and operating systems lurking under the surface."
Peter Bright Feb 1985 **PERSONAL COMPUTER WORLD**

"The electronics in the machine are a work of art ... The heart of the 520ST is a Motorola 68000, one of the most powerful 16-bit processors around and in many respects it is close to being a 32-bit chip ... when the machine appears in the shops, I'll be at the front end of the queue to buy one."
Peter Bright June 1985 **PERSONAL COMPUTER WORLD**

"This machine is significantly more powerful than an IBM PC ... If it's possible to design a sure-fire winning machine, this is it."
May 11th 1985 **PERSONAL COMPUTER NEWS**

"... the use of GEM makes the new range of Atari computers se similar to the Macintosh (with the added attraction of colour), that they are already being called 'Jackintoshes'."
May 2nd 1985 **COMPUTING**

"The new Atari ST computers truly represent to the consumer what Jack Tramiel is saying — easy-to-use computing power without the price."
March 1985 **ANALOG COMPUTING**

"It (the ST) uses the most modern technology that is affordable, in a package that gives a professional impression."
May 23rd 1985 **POPULAR COMPUTING WEEKLY**

"The Atari ST is one of the most elegant designs I have seen ... Atari has used an original and elegant method of memory management which should make the ST faster than any other PC on the market - in any price bracket ... The 64K dollar question is would I go out and spend money for one? To which the only answer is 'Try and stop me!'"
John Lambert July 1985 **ELECTRONICS & COMPUTING**

"The 520ST is technically excellent ... The 520ST hardware is the new standard by which others will be judged."
July 1985 **YOUR COMPUTER**

SILICA SHOP

WE ARE THE UK's No1 ATARI SPECIALISTS

At Silica we have been successfully dedicated to Atari ever since their products first appeared on the UK market. We can attribute our success largely to the Atari specialisation which we practice and to the user back-up we provide. Rest assured that when you buy a piece of Atari hardware at Silica you will be fully supported. Our mailings giving news of software releases and developments will keep you up to date with the Atari market and our technical support team and sales staff are at the end of the telephone line to deal with your problems and supply your every need. With our specialist bias, we aim to keep stocks of all the available Atari hardware, software, peripherals and accessories. We also stock a wide range of Atari dedicated books and through us, the owners on our list can subscribe to several American Atari dedicated magazines. We can provide a full service to all Atari owners and are now firmly established as the UK's NUMBER ONE Atari specialists. Here are just some of the things we can offer to our customers.

- ★ FREE POST & PACKING ON MAIL ORDERS
- ★ FREE NEXT DAY SECURICOR DELIVERY
- ★ INFORMATION MAILING SERVICE
- ★ TECHNICAL SUPPORT TEAM
- ★ HIGHLY COMPETITIVE PRICES
- ★ AFTER SALES SUPPORT SERVICE
- ★ REPAIR SERVICE ON ATARI PRODUCTS

If you would like to be registered on our mailing list as an Atari computer owner, or as a person interested in buying an Atari machine, let us know. We will be pleased to keep you up to date with new Atari developments free of charge. So, return the coupon today and begin experiencing a specialist Atari service that is second to none.

SILICA HOTLINE 01-309 1111

SILICA SHOP LTD, 1-4 The Mews, Hatherley Road, Sidcup, Kent, DA14 4DX

SEND FOR FREE ATARI ST LITERATURE

To: Silica Shop Ltd, Dept PCT 0985, 1-4 The Mews, Hatherley Road, Sidcup, Kent, DA14 4DX

PLEASE SEND ME FREE LITERATURE
ON THE NEW ATARI 520ST COMPUTER

Mr/Mrs/Ms: ..... Initials: ..... Surname: .....

Address: .....

.....

.....

.....

Postcode: .....

Do you already own a computer
If so, which one do you own? .....